Note:
Before using this information and the product it supports, be sure to read the general information under “Notices” on page 795.
# Contents

Figures............................................................................................................................................. xvii

Tables.................................................................................................................................................xix

About this document.........................................................................................................................xxi
   Who should read this document.................................................................................................. xxii
   How this document is organized................................................................................................. xxii
   How to use this document............................................................................................................ xxii
   How to contact IBM service......................................................................................................... xxii
   Conventions and terminology that are used in this information.................................................. xxiii
   Prerequisite and related information......................................................................................... xxiii

Summary of changes for SNA Diagnosis Volume 2: FFST Dumps and the VIT........................... xxix
   Changes made in z/OS Communications Server Version 2 Release 3........................................ xxix
   Changes made in z/OS Communications Server Version 2 Release 2, as updated June 2017..... xxx
   Changes made in z/OS Version 2 Release 2, as updated September 2016................................. xxx
   Changes made in z/OS Version 2 Release 2, as updated March 2016........................................... xxx
   Changes made in z/OS Version 2 Release 2................................................................................xxxi
   Changes made in z/OS Version 2 Release 1, as updated February 2015..................................... xxxii
   z/OS Version 2 Release 1 summary of changes....................................................................... xxxiii

Chapter 1. Using FFST dumps................................................................................................................. 1
   First Failure Support Technology (FFST) for VTAM................................................................. 1
      FFST dumps............................................................................................................................. 1
      The generic alert...................................................................................................................... 3
      The symptom string.............................................................................................................. 4
      FFST console......................................................................................................................... 4
      Using the trap module.......................................................................................................... 4
      When to dump coupling facility structures......................................................................... 5
      Activating an APPC sense code trap.................................................................................... 5
   Phantom First Failure Support Technology (PFFST) for VTAM or CSM................................. 6
   Reporting a problem.................................................................................................................... 6

Chapter 2. Using the VTAM internal trace............................................................................................ 7
   VIT control levels.......................................................................................................................... 7
   Selecting the level of VIT Control.............................................................................................. 8
   Interaction of VIT option sets and "Full" VIT Control mode processing..................................... 8
   Example behavior..................................................................................................................... 8
   Activating the VIT...................................................................................................................... 9
   Trace options for the VIT......................................................................................................... 11
   Internal and external trace recording for the VIT...................................................................... 11
      Recording traces in an internal 64-bit common table......................................................... 18
      Recording traces in an external file..................................................................................... 20
   Recording SNAP traces............................................................................................................ 20
   Deactivating the VIT.................................................................................................................. 21
   Extracting VIT information from a dump.................................................................................. 23
   Using module names to isolate VTAM problems................................................................. 23

Appendix A. FFST operations.............................................................................................................. 27
<table>
<thead>
<tr>
<th>Entry Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>EPW9562E - Entry for SDUMP taken for CSDUMP request</td>
<td>149</td>
</tr>
<tr>
<td>EPW9563E - Entry for VTAM CSDUMP trigger</td>
<td>149</td>
</tr>
<tr>
<td>EPW9565E - Entry for VTAM CSDUMP trigger (continuation)</td>
<td>168</td>
</tr>
<tr>
<td>EPW9570I - Entry for VTAM ISTORFBA DUMP</td>
<td>170</td>
</tr>
<tr>
<td>EPW9571I - Entry for VTAM ISTORFBA DUMP (continuation)</td>
<td>171</td>
</tr>
<tr>
<td>EPW9574I - Entry for VTAM FFST</td>
<td>173</td>
</tr>
<tr>
<td>EPW9575I - Entry for VTAM Phantom FFST (continuation)</td>
<td>174</td>
</tr>
<tr>
<td>EPW9576I - Entry for VTAM Phantom FFST (continuation)</td>
<td>174</td>
</tr>
<tr>
<td>EPW9577I - Entry for VTAM Phantom FFST</td>
<td>174</td>
</tr>
<tr>
<td>FFST for MVS transition code function</td>
<td>152</td>
</tr>
<tr>
<td>MVS post dump exit (IEAVTSEL)</td>
<td>152</td>
</tr>
<tr>
<td>CICS dump global exit</td>
<td>154</td>
</tr>
<tr>
<td>MVS dump frequency threshold support</td>
<td>156</td>
</tr>
<tr>
<td>Specifying IBM product identification information</td>
<td>156</td>
</tr>
<tr>
<td>Transition code parameter list</td>
<td>157</td>
</tr>
<tr>
<td>S/390 channel attached device support</td>
<td>158</td>
</tr>
<tr>
<td>FFST controls</td>
<td>159</td>
</tr>
<tr>
<td>FFST probe control</td>
<td>159</td>
</tr>
<tr>
<td>Parameter list control</td>
<td>159</td>
</tr>
<tr>
<td>Parameter list control logic</td>
<td>162</td>
</tr>
<tr>
<td>Appendix B. VTAM internal trace (VIT) record descriptions</td>
<td>167</td>
</tr>
<tr>
<td>**** Entry for SDUMP taken for CSDUMP request</td>
<td>167</td>
</tr>
<tr>
<td>**** Entry for VTAM CSDUMP trigger</td>
<td>167</td>
</tr>
<tr>
<td>**** Entry for VTAM CSDUMP trigger (continuation)</td>
<td>168</td>
</tr>
<tr>
<td>**** Entry for VTAM ISTORFBA DUMP</td>
<td>170</td>
</tr>
<tr>
<td>**** Entry for VTAM ISTORFBA DUMP (continuation)</td>
<td>171</td>
</tr>
<tr>
<td>**** Entry for VTAM FFST</td>
<td>173</td>
</tr>
<tr>
<td>**** Entry for VTAM Phantom FFST (continuation)</td>
<td>174</td>
</tr>
<tr>
<td>ABND entry for abend SNAP routine</td>
<td>177</td>
</tr>
<tr>
<td>ACA1 or ACI1 entry for LU 6.2 authorized IO or LU 6.2 TPIO (Part 1)</td>
<td>177</td>
</tr>
<tr>
<td>ACA2 or ACI2 entry for LU 6.2 authorized IO or LU 6.2 TPIO (Part 2)</td>
<td>181</td>
</tr>
<tr>
<td>ACA3 or ACI3 entry for LU 6.2 authorized IO or LU 6.2 TPIO (Part 3)</td>
<td>182</td>
</tr>
<tr>
<td>ACP1 or ACR1 entry for LU 6.2 user post or RPL exit (Part 1)</td>
<td>182</td>
</tr>
<tr>
<td>ACP2 or ACR2 entry for LU 6.2 user post or RPL exit (Part 2)</td>
<td>186</td>
</tr>
<tr>
<td>ACRC entry for RPL6 return code</td>
<td>188</td>
</tr>
<tr>
<td>ACSN entry for APPC sense code</td>
<td>190</td>
</tr>
<tr>
<td>ACU1 entry for LU 6.2 user exit (Part 1)</td>
<td>191</td>
</tr>
<tr>
<td>ACU2 entry for LU 6.2 user exit (Part 2)</td>
<td>192</td>
</tr>
<tr>
<td>ADE entry for device errors</td>
<td>192</td>
</tr>
<tr>
<td>AFSM entry for altering an FSM state</td>
<td>196</td>
</tr>
<tr>
<td>AI1 entry for authorized IO (Part 1)</td>
<td>198</td>
</tr>
<tr>
<td>AI2 entry for authorized IO (Part 2)</td>
<td>200</td>
</tr>
<tr>
<td>AI3 entry for authorized IO (part 3)</td>
<td>201</td>
</tr>
<tr>
<td>ALSx entry for adjacent link station</td>
<td>202</td>
</tr>
<tr>
<td>AP entry for signals passed between components of the APPN CP</td>
<td>204</td>
</tr>
<tr>
<td>A2 entry for requests and responses with a RUPE</td>
<td>206</td>
</tr>
<tr>
<td>ARB entry for RTP LU-LU session data</td>
<td>206</td>
</tr>
<tr>
<td>ARBB entry for HPR trace option</td>
<td>207</td>
</tr>
<tr>
<td>ARBR entry for HPR trace option</td>
<td>213</td>
</tr>
<tr>
<td>AREL entry for abend RELSTORE</td>
<td>244</td>
</tr>
<tr>
<td>ARP or ARS entry for RTP session data reply (Part 1)</td>
<td>245</td>
</tr>
<tr>
<td>ARP2 or ARS2 entry for RTP session data reply (Part 2)</td>
<td>247</td>
</tr>
<tr>
<td>ARQ entry for RTP session data request (Part 1)</td>
<td>248</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MDIS</td>
<td>MIBDisconnect</td>
</tr>
<tr>
<td>MDEL</td>
<td>MIBSendDeleteRegistration</td>
</tr>
<tr>
<td>MREG</td>
<td>MIBSendRegister</td>
</tr>
<tr>
<td>MQRQ</td>
<td>MIBSendRequest or MIBSendResponse</td>
</tr>
<tr>
<td>MMG</td>
<td>Memory management (Part 1)</td>
</tr>
<tr>
<td>LCSM</td>
<td>MACaddress for DATA SEND/RECEIVE</td>
</tr>
<tr>
<td>ISPx</td>
<td>Invoking an internal shared memory (ISM) Verb (part 1)</td>
</tr>
<tr>
<td>IPTM</td>
<td>Enterprise Extender timer operation</td>
</tr>
<tr>
<td>IRBD</td>
<td>IRB dispatch</td>
</tr>
<tr>
<td>IRBX</td>
<td>IRB exit</td>
</tr>
<tr>
<td>ISPx</td>
<td>Invoking an internal shared memory (ISM) Verb (part 2)</td>
</tr>
<tr>
<td>IUTx</td>
<td>IUT processing (Part 1)</td>
</tr>
<tr>
<td>IUTC</td>
<td>IUTI, IUTM, IUTQ, and IUTS mapping and field descriptions</td>
</tr>
<tr>
<td>IUTC</td>
<td>IUTI, IUTM, IUTQ, and IUTS mapping and field descriptions</td>
</tr>
<tr>
<td>IUTM</td>
<td>IUTI, IUTM, IUTQ, and IUTS mapping and field descriptions</td>
</tr>
<tr>
<td>IUTF</td>
<td>IUTI, IUTM, IUTQ, and IUTS mapping and field descriptions</td>
</tr>
<tr>
<td>IUTR</td>
<td>IUTI, IUTM, IUTQ, and IUTS mapping and field descriptions</td>
</tr>
<tr>
<td>IUTX</td>
<td>IUTI, IUTM, IUTQ, and IUTS mapping and field descriptions</td>
</tr>
<tr>
<td>IUT2</td>
<td>IUT processing (Part 2)</td>
</tr>
<tr>
<td>IUT3</td>
<td>IUT processing (Part 3)</td>
</tr>
<tr>
<td>LCSx</td>
<td>LAN channel station error (Part 1)</td>
</tr>
<tr>
<td>LCS2</td>
<td>LAN channel station error (Part 2)</td>
</tr>
<tr>
<td>LCSM</td>
<td>MACaddress for DATA SEND/RECEIVE</td>
</tr>
<tr>
<td>LDLC</td>
<td>Invoking LDLC</td>
</tr>
<tr>
<td>LDL2</td>
<td>Invoking LDLC (Part 2)</td>
</tr>
<tr>
<td>LKEX</td>
<td>TPLOCK exclusive</td>
</tr>
<tr>
<td>LKSH</td>
<td>TPLOCK shared</td>
</tr>
<tr>
<td>LNKx</td>
<td>Lost trace record</td>
</tr>
<tr>
<td>LOST</td>
<td>Lost trace record</td>
</tr>
<tr>
<td>LSNx</td>
<td>Local SNA record</td>
</tr>
<tr>
<td>LSN2</td>
<td>Local SNA record (Part 2)</td>
</tr>
<tr>
<td>MCO1</td>
<td>MIBConnect (Part 1)</td>
</tr>
<tr>
<td>MCO2</td>
<td>MIBConnect (Part 2)</td>
</tr>
<tr>
<td>MDEL</td>
<td>MIBSendDeleteRegistration</td>
</tr>
<tr>
<td>MMG</td>
<td>Memory management (Part 2)</td>
</tr>
<tr>
<td>MNPS</td>
<td>MNPS macro</td>
</tr>
<tr>
<td>MPDU</td>
<td>MPC PDU processing</td>
</tr>
<tr>
<td>MPD2</td>
<td>MPC PDU processing</td>
</tr>
<tr>
<td>MQRQ</td>
<td>MIBSendRequest or MIBSendResponse</td>
</tr>
<tr>
<td>MRG</td>
<td>Updates to the VTAM topology agent (Part 1)</td>
</tr>
<tr>
<td>MRG2</td>
<td>Updates to the VTAM topology agent (Part 2)</td>
</tr>
<tr>
<td>MSG</td>
<td>Message issued</td>
</tr>
<tr>
<td>MSGS</td>
<td>Message sent (Part 1)</td>
</tr>
<tr>
<td>MT</td>
<td>Module trace</td>
</tr>
<tr>
<td>MU1</td>
<td>LU 6.2 message unit (Part 1)</td>
</tr>
<tr>
<td>MU2</td>
<td>LU 6.2 message unit (Part 2)</td>
</tr>
<tr>
<td>MU3</td>
<td>LU 6.2 message unit (Part 3)</td>
</tr>
<tr>
<td>MU4</td>
<td>LU 6.2 message unit (Part 4)</td>
</tr>
</tbody>
</table>
TPT2 entry for TG modified (Part 2) ................................................................. 642
TPT3 entry for TG modified (Part 3) ................................................................. 643
TREx entry for routing tree Build or Update .................................................. 644
TRMR entry for mark record (Part 1) ............................................................... 647
TRM2 entry for mark record (Part 2) ............................................................... 648
TRM3 entry for mark record (Part 3) ............................................................... 649
TRNM entry for translate message (Part 1) ..................................................... 650
TRN2 entry for translate message (Part 2) ....................................................... 652
TRRT entry for subtrace tree ........................................................................ 653
TSNS entry to trace sense codes .................................................................... 658
UE entry for USER exit (DFASY or RESP) ..................................................... 659
UE1 entry for USER exit (not DFASY or RESP, Part 1) ................................. 660
UE2 entry for USER exit (Not DFASY or RESP, part 2) ................................. 661
ULKA entry for release all locks .................................................................... 662
UNLK entry for release a lock ....................................................................... 663
UP entry for user application program post .................................................. 664
USI or USO entry for APPC commands ........................................................ 666
USI2 or USO2 entry for APPC commands .................................................... 668
UVO or UVI entry for vector information ..................................................... 668
UVO2 or UVI2 entry for vector information .................................................. 671
VCC1 entry for CNCB queued (Part 1) ......................................................... 672
VCC2 entry for CNCB queued (Part 2) ......................................................... 674
VCDQ entry for work element dequeued ...................................................... 675
VHCR entry for invoking a RoCE VHCR operation (part 1) ......................... 682
VHC2 entry for invoking a RoCE VHCR operation (part 2) ......................... 683
VHC3 entry for invoking a RoCE VHCR operation (part 3) ......................... 684
VHC4 entry for invoking a RoCE VHCR operation (part 4) ......................... 684
VHC5 entry for invoking a RoCE VHCR operation (part 5) ......................... 685
VPST entry for VPOST macro ..................................................................... 685
VRSM entry for resume after VWAIT .......................................................... 686
VTAL entry for allocate storage .................................................................. 687
VTFR entry for free storage ......................................................................... 688
VWAI entry for VWAIT macro ..................................................................... 689
WAIT entry for TPWAIT macro ................................................................... 690
XBA1 entry for extended buffer list (Part 1) ................................................ 692
XBA2 entry for extended buffer list (Part 2) ................................................ 693
XBA3 entry for extended buffer list (Part 3) ................................................ 694
XB1I entry for extended buffer list AMU or TSCB ...................................... 695
XB12 entry for extended buffer list AMU or TSCB ...................................... 696
XB13 entry for extended buffer list AMU or TSCB ...................................... 697
XB61 entry for extended buffer list SPAC ................................................... 698
XB62 entry for extended buffer list SPAC ................................................... 699
XB63 entry for extended buffer list SPAC ................................................... 699
XFCF entry for save message (Part 1) ........................................................ 701
XCC2 entry for save message (Part 2) ........................................................ 702
XCF3 entry for join XCF group (Part 1) ........................................................ 702
XCJ2 entry for join XCF group (Part 2) ........................................................ 703
XCLF entry for leave XCF group (Part 1) ...................................................... 704
XCL2 entry for leave XCF group (Part 2) ...................................................... 705
XCFM entry for XCF group member (Part 1) .............................................. 706
XCM2 entry for XCF group member (Part 2) .............................................. 707
XCFR entry for receive message (Part 1) .................................................... 708
XCR2 entry for receive message (Part 2) .................................................... 709
XCSF entry for send message (Part 1) ........................................................ 711
XCS2 entry for send message (Part 2) ........................................................ 712
XCXF entry for VTAM XCF message exit or notify exit driven ................ 713
XID entry for exchange ID (Part 1) ............................................................. 715
XID2 entry for exchange ID (Part 2) ............................................................ 716
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSM probes</td>
<td>789</td>
</tr>
<tr>
<td>Appendix F. Architectural specifications</td>
<td>791</td>
</tr>
<tr>
<td>Appendix G. Accessibility</td>
<td>793</td>
</tr>
<tr>
<td>Notices</td>
<td>795</td>
</tr>
<tr>
<td>Terms and conditions for product documentation</td>
<td>796</td>
</tr>
<tr>
<td>IBM Online Privacy Statement</td>
<td>797</td>
</tr>
<tr>
<td>Policy for unsupported hardware</td>
<td>797</td>
</tr>
<tr>
<td>Minimum supported hardware</td>
<td>797</td>
</tr>
<tr>
<td>Programming interface information</td>
<td>798</td>
</tr>
<tr>
<td>Policy for unsupported hardware</td>
<td>798</td>
</tr>
<tr>
<td>Trademarks</td>
<td>798</td>
</tr>
<tr>
<td>Bibliography</td>
<td>799</td>
</tr>
<tr>
<td>Index</td>
<td>805</td>
</tr>
<tr>
<td>Communicating your comments to IBM</td>
<td>817</td>
</tr>
</tbody>
</table>
Figures

1. Sample VTAM FFST console listing...............................................................................................................4

2. Unformatted VIT records in dump output (MODE=INT)...........................................................................19

3. TRS common topology trace table with two buffers allocated.................................................................740
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Trace options of the OPTION operand</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>Exception conditions always traced by the VIT</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>VIT options and the records they create (API - LOCK)</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>VIT options and the records they create (MSG - XCF)</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>VIT group options</td>
<td>18</td>
</tr>
<tr>
<td>6</td>
<td>Deactivating the VIT</td>
<td>21</td>
</tr>
<tr>
<td>7</td>
<td>VTAM component IDs</td>
<td>24</td>
</tr>
<tr>
<td>8</td>
<td>Standard Keywords Supported by FFST</td>
<td>67</td>
</tr>
<tr>
<td>9</td>
<td>Probe Initialization Return Codes</td>
<td>77</td>
</tr>
<tr>
<td>10</td>
<td>Probe Statement Return Codes</td>
<td>78</td>
</tr>
<tr>
<td>11</td>
<td>Probe Termination Return Codes</td>
<td>80</td>
</tr>
<tr>
<td>12</td>
<td>Dump Return Codes</td>
<td>86</td>
</tr>
<tr>
<td>13</td>
<td>Symptom Record Return Codes</td>
<td>87</td>
</tr>
<tr>
<td>14</td>
<td>Generic Alert Return Codes</td>
<td>89</td>
</tr>
<tr>
<td>15</td>
<td>Exit-type-dependent information for a UE1 entry</td>
<td>661</td>
</tr>
<tr>
<td>16</td>
<td>FFST probe index</td>
<td>747</td>
</tr>
<tr>
<td>17</td>
<td>FFST probe naming conventions</td>
<td>748</td>
</tr>
<tr>
<td>18</td>
<td>FFST probes for APPN over logical link</td>
<td>749</td>
</tr>
<tr>
<td>19</td>
<td>FFST probes for APPN TG management</td>
<td>750</td>
</tr>
<tr>
<td>20</td>
<td>FFST probes for CMIP services (ISTCMCxx)</td>
<td>750</td>
</tr>
<tr>
<td>21</td>
<td>FFST probes for configuration services (ISTCSCxx)</td>
<td>750</td>
</tr>
<tr>
<td>22</td>
<td>FFST probes for control operator (ISTCOCxx)</td>
<td>752</td>
</tr>
<tr>
<td>23</td>
<td>FFST probes for coupling facility services (ISTFSCxx)</td>
<td>753</td>
</tr>
</tbody>
</table>
24. FFST probes for dependent LU server (ISTDLCxx)................................................................. 759
25. FFST probes for directory services (ISTDRCxx)........................................................................ 760
26. FFST probes for half-session services (ISTHSCxx)................................................................. 767
27. FFST probes for Enterprise Extender (ISTAUCxx)................................................................. 767
28. FFST probes for logical link control...................................................................................... 768
29. FFST probes for LU network services (ISTNSCxx)............................................................... 769
30. FFST probes for LU resource manager (ISTRMCxx)............................................................. 769
31. FFST probes for management services transport (ISTMTCxx).............................................. 769
32. FFST probes for network resource management (ISTNACxx).................................................. 770
33. FFST probes for presentation services (ISTPSCxx)............................................................... 770
34. FFST probes for recovery manager (ISTRVMxx)..................................................................... 771
35. FFST probes for session services (ISTSSCxx).......................................................................... 772
36. FFST probes for session services CP-CP (ISTSCCxx)............................................................. 772
37. FFST probes for session services LU-LU (ISTSLCxx)............................................................. 774
38. FFST probes for storage management (ISTORCxx)............................................................... 781
39. FFST probes for topology and routing services (ISTTRCxx)................................................... 781
40. FFST probes for trace services (ISTRACxx)............................................................................. 782
41. FFST probes for TC-DLC (ISTTCCxx).................................................................................... 782
42. FFST probes for transmission subsystem (ISTTSCxx).............................................................. 783
43. FFST probes for the VTAM topology agent (ISTTACxx)......................................................... 784
44. CSM FFST probe naming conventions.................................................................................... 789
45. FFST probes for communications storage manager (IVTSMCxx)......................................... 789
About this document

This document is intended to help system programmers in a VTAM® environment diagnose problems with the VTAM program. Use the document to isolate and identify problems with your VTAM network and to collect appropriate documentation to resolve network problems.

The information in this document includes descriptions of support for both IPv4 and IPv6 networking protocols. Unless explicitly noted, descriptions of IP protocol support concern IPv4. IPv6 support is qualified within the text.

Who should read this document

System programmers should use this document to analyze a VTAM problem, classify the problem as a specific type, and provide information about the problem to an IBM® Support Center representative.

You should be familiar with the VTAM service aids and the procedures for reporting problems to an IBM Support Center representative.

How this document is organized

This document is organized into the following topics:

• Chapter 1, “Using FFST dumps,” on page 1 describes dump procedures.
• Chapter 2, “Using the VTAM internal trace,” on page 7 describes how to use the VTAM Internal Trace (VIT).
• Appendix B, “VTAM internal trace (VIT) record descriptions,” on page 167 describes the VIT records.
• Appendix C, “Internal topology traces,” on page 721 describes the internal topology traces.
• Appendix D, “First Failure Support Technology (FFST) probes,” on page 747 describes the FFST probes that trigger dumps when an unusual condition occurs in VTAM.
• Appendix E, “Communications storage manager (CSM) FFST probes,” on page 789 describes the CSM FFST probes that trigger dumps when an unusual condition occurs in CSM.
• Appendix F, “Architectural specifications,” on page 791 lists documents that provide architectural specifications for the SNA protocol.
• Appendix G, “Accessibility,” on page 793 describes accessibility features to help users with physical disabilities.
• “Notices” on page 795 contains notices and trademarks used in this document.
• “Bibliography” on page 799 contains descriptions of the documents in the z/OS Communications Server library.

How to use this document

Use this document to isolate and identify problems with your VTAM network and to collect appropriate documentation to resolve network problems.

How to contact IBM service

For immediate assistance, visit this website: http://www.software.ibm.com/support

Most problems can be resolved at this website, where you can submit questions and problem reports electronically, and access a variety of diagnosis information.

© Copyright IBM Corp. 2000, 2019
For telephone assistance in problem diagnosis and resolution (in the United States or Puerto Rico), call the IBM Software Support Center anytime (1-800-IBM-SERV). You will receive a return call within 8 business hours (Monday – Friday, 8:00 a.m. – 5:00 p.m., local customer time).

Outside the United States or Puerto Rico, contact your local IBM representative or your authorized IBM supplier.

If you would like to provide feedback on this publication, see “Communicating your comments to IBM” on page 817.

**Conventions and terminology that are used in this information**

Commands in this information that can be used in both TSO and z/OS UNIX environments use the following conventions:

- When describing how to use the command in a TSO environment, the command is presented in uppercase (for example, NETSTAT).
- When describing how to use the command in a z/OS UNIX environment, the command is presented in bold lowercase (for example, `netstat`).
- When referring to the command in a general way in text, the command is presented with an initial capital letter (for example, Netstat).

All the exit routines described in this information are *installation-wide exit routines*. The installation-wide exit routines also called installation-wide exits, exit routines, and exits throughout this information.

The TPF logon manager, although included with VTAM, is an application program; therefore, the logon manager is documented separately from VTAM.

Samples used in this information might not be updated for each release. Evaluate a sample carefully before applying it to your system.

**Note:** In this information, you might see the following Shared Memory Communications over Remote Direct Memory Access (SMC-R) terminology:

- RoCE Express®, which is a generic term representing IBM 10 GbE RoCE Express, IBM 10 GbE RoCE Express2, and IBM 25 GbE RoCE Express2 feature capabilities. When this term is used in this information, the processing being described applies to both features. If processing is applicable to only one feature, the full terminology, for instance, IBM 10 GbE RoCE Express will be used.

- RoCE Express2, which is a generic term representing an IBM RoCE Express2® feature that might operate in either 10 GbE or 25 GbE link speed. When this term is used in this information, the processing being described applies to either link speed. If processing is applicable to only one link speed, the full terminology, for instance, IBM 25 GbE RoCE Express2 will be used.

- RDMA network interface card (RNIC), which is used to refer to the IBM® 10 GbE RoCE Express, IBM® 10 GbE RoCE Express2, or IBM 25 GbE RoCE Express2 feature.

- Shared RoCE environment, which means that the "RoCE Express" feature can be used concurrently, or shared, by multiple operating system instances. The feature is considered to operate in a shared RoCE environment even if you use it with a single operating system instance.

**Clarification of notes**

Information traditionally qualified as Notes is further qualified as follows:

**Attention**
- Indicate the possibility of damage

**Guideline**
- Customary way to perform a procedure

**Note**
- Supplemental detail
Rule
Something you must do; limitations on your actions

Restriction
Indicates certain conditions are not supported; limitations on a product or facility

Requirement
Dependencies, prerequisites

Result
Indicates the outcome

Tip
Offers shortcuts or alternative ways of performing an action; a hint

Prerequisite and related information

z/OS Communications Server function is described in the z/OS Communications Server library. Descriptions of those documents are listed in “Bibliography” on page 799, in the back of this document.

Required information

Before using this product, you should be familiar with TCP/IP, VTAM, MVS™, and UNIX System Services.

Softcopy information

Softcopy publications are available in the following collection.

<table>
<thead>
<tr>
<th>Titles</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Z Redbooks</td>
<td>The IBM Z™ subject areas range from e-business application development and enablement to hardware, networking, Linux, solutions, security, parallel sysplex, and many others. For more information about the Redbooks® publications, see <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> and <a href="http://www.ibm.com/systems/z/os/zos/zfavorites/">http://www.ibm.com/systems/z/os/zos/zfavorites/</a>.</td>
</tr>
</tbody>
</table>

Other documents

This information explains how z/OS references information in other documents.

When possible, this information uses cross-document links that go directly to the topic in reference using shortened versions of the document title. For complete titles and order numbers of the documents for all products that are part of z/OS, see z/OS Information Roadmap (SA23-2299). The Roadmap describes what level of documents are supplied with each release of z/OS Communications Server, and also describes each z/OS publication.

To find the complete z/OS library, visit the z/OS library in IBM Knowledge Center (www.ibm.com/support/knowledgecenter/SSLTBW/welcome).

Relevant RFCs are listed in an appendix of the IP documents. Architectural specifications for the SNA protocol are listed in an appendix of the SNA documents.

The following table lists documents that might be helpful to readers.

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Number</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td><strong>SNA Formats</strong></td>
<td>GA27-3136</td>
</tr>
<tr>
<td><em>TCP/IP Tutorial and Technical Overview</em></td>
<td>GG24-3376</td>
</tr>
<tr>
<td><em>Understanding LDAP</em></td>
<td>SG24-4986</td>
</tr>
<tr>
<td><em>z/OS Cryptographic Services System SSL Programming</em></td>
<td>SC14-7495</td>
</tr>
<tr>
<td><em>z/OS IBM Tivoli Directory Server Administration and Use for z/OS</em></td>
<td>SC23-6788</td>
</tr>
<tr>
<td><em>z/OS JES2 Initialization and Tuning Guide</em></td>
<td>SA32-0991</td>
</tr>
<tr>
<td><em>z/OS Problem Management</em></td>
<td>SC23-6844</td>
</tr>
<tr>
<td><em>z/OS MVS Diagnosis: Reference</em></td>
<td>GA32-0904</td>
</tr>
<tr>
<td><em>z/OS MVS Diagnosis: Tools and Service Aids</em></td>
<td>GA32-0905</td>
</tr>
<tr>
<td><em>z/OS MVS Using the Subsystem Interface</em></td>
<td>SA38-0679</td>
</tr>
<tr>
<td><em>z/OS Program Directory</em></td>
<td>GI11-9848</td>
</tr>
<tr>
<td><em>z/OS UNIX System Services Command Reference</em></td>
<td>SA23-2280</td>
</tr>
<tr>
<td><em>z/OS UNIX System Services Planning</em></td>
<td>GA32-0884</td>
</tr>
<tr>
<td><em>z/OS UNIX System Services Programming: Assembler Callable Services Reference</em></td>
<td>SA23-2281</td>
</tr>
<tr>
<td><em>z/OS UNIX System Services User's Guide</em></td>
<td>SA23-2279</td>
</tr>
<tr>
<td><em>z/OS XL C/C++ Runtime Library Reference</em></td>
<td>SC14-7314</td>
</tr>
</tbody>
</table>

**Redbooks publications**

The following Redbooks publications might help you as you implement z/OS Communications Server.

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>IBM z/OS Communications Server TCP/IP Implementation, Volume 1: Base Functions, Connectivity, and Routing</em></td>
<td>SG24-8096</td>
</tr>
<tr>
<td><em>IBM z/OS Communications Server TCP/IP Implementation, Volume 2: Standard Applications</em></td>
<td>SG24-8097</td>
</tr>
<tr>
<td><em>IBM z/OS Communications Server TCP/IP Implementation, Volume 3: High Availability, Scalability, and Performance</em></td>
<td>SG24-8098</td>
</tr>
<tr>
<td><em>IBM z/OS Communications Server TCP/IP Implementation, Volume 4: Security and Policy-Based Networking</em></td>
<td>SG24-8099</td>
</tr>
</tbody>
</table>
Where to find related information on the Internet

**z/OS**
This site provides information about z/OS Communications Server release availability, migration information, downloads, and links to information about z/OS technology
http://www.ibm.com/systems/z/os/zos/

**z/OS Internet Library**
Use this site to view and download z/OS Communications Server documentation

**IBM Communications Server product**
The primary home page for information about z/OS Communications Server

**z/OS Communications Server product**
The page contains z/OS Communications Server product introduction

**IBM Communications Server product support**
Use this site to submit and track problems and search the z/OS Communications Server knowledge base for Technotes, FAQs, white papers, and other z/OS Communications Server information
http://www.software.ibm.com/support

**IBM Communications Server performance information**
This site contains links to the most recent Communications Server performance reports
http://www.ibm.com/support/docview.wss?uid=swg27005524

**IBM Systems Center publications**
Use this site to view and order Redbooks publications, Redpapers, and Technote
http://www.redbooks.ibm.com/
IBM Systems Center flashes

Search the Technical Sales Library for Techdocs (including Flashes, presentations, Technotes, FAQs, white papers, Customer Support Plans, and Skills Transfer information)

http://www.ibm.com/support/techdocs/atsmastr.nsf

Tivoli® NetView® for z/OS

Use this site to view and download product documentation about Tivoli NetView for z/OS

http://www.ibm.com/support/knowledgecenter/SSZJDU/welcome

RFCs

Search for and view Request for Comments documents in this section of the Internet Engineering Task Force website, with links to the RFC repository and the IETF Working Groups web page

http://www.ietf.org/rfc.html

Internet drafts

View Internet-Drafts, which are working documents of the Internet Engineering Task Force (IETF) and other groups, in this section of the Internet Engineering Task Force website

http://www.ietf.org/ID.html

Information about web addresses can also be found in information APAR II11334.

Note: Any pointers in this publication to websites are provided for convenience only and do not serve as an endorsement of these websites.

DNS websites

For more information about DNS, see the following USENET news groups and mailing addresses:

USENET news groups
comp.protocols.dns.bind

BIND mailing lists
https://lists.isc.org/mailman/listinfo

BIND Users

- Subscribe by sending mail to bind-users-request@isc.org.
- Submit questions or answers to this forum by sending mail to bind-users@isc.org.

BIND 9 Users (This list might not be maintained indefinitely.)

- Subscribe by sending mail to bind9-users-request@isc.org.
- Submit questions or answers to this forum by sending mail to bind9-users@isc.org.

The z/OS Basic Skills Information Center

The z/OS Basic Skills Information Center is a web-based information resource intended to help users learn the basic concepts of z/OS, the operating system that runs most of the IBM mainframe computers in use today. The Information Center is designed to introduce a new generation of Information Technology professionals to basic concepts and help them prepare for a career as a z/OS professional, such as a z/OS systems programmer.

Specifically, the z/OS Basic Skills Information Center is intended to achieve the following objectives:

- Provide basic education and information about z/OS without charge
- Shorten the time it takes for people to become productive on the mainframe
- Make it easier for new people to learn z/OS
To access the z/OS Basic Skills Information Center, open your web browser to the following website, which is available to all users (no login required): https://www.ibm.com/support/knowledgecenter/zosbasics/com.ibm.zos.zbasics/homepage.html?cp=zosbasics
Summary of changes for SNA Diagnosis Volume 2: FFST Dumps and the VIT

This document contains terminology, maintenance, and editorial changes, including changes to improve consistency and retrievability. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Changes made in z/OS Communications Server Version 2 Release 3

This document contains information previously presented in z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT, which supported z/OS Version 2 Release 2.

June 2019
New information
Appendix A, “FFST operations,” on page 27

December 2018
Changed information
Technical and terminology changes about Shared Memory Communications

September 2017
New information
• Communications Server support for RoCE Express2 feature, see the following topics:
  – HCQ entry for invoking a RoCE HCQ operation (Part 1)
  – HCQ2 entry for invoking a RoCE HCQ operation (Part 2)
  – HCQ3 entry for invoking a RoCE HCQ operation (Part 3)
  – HCQ4 entry for invoking a RoCE HCQ operation (Part 4)
  – HCQ5 entry for invoking a RoCE HCQ operation (Part 5)
  – HCQ6 entry for invoking a RoCE HCQ operation (Part 6)
• Improved control over default VTAM VIT options, see the following topics:
  – “VIT control levels” on page 7
  – “Selecting the level of VIT Control” on page 8
  – “Interaction of VIT option sets and “Full” VIT Control mode processing” on page 8
  – “Example behavior” on page 8
• VTAM 3270 intrusion detection services, see the following topics:
  – “FB64 entry for FREEB64 macro” on page 383
  – “GB64 entry for GETB64 macro” on page 396
  – “3270 entry for 3270 Intrusion Detection Services” on page 718
  – “3271 entry for 3270 Intrusion Detection Services” on page 719

Changed information
• Communications Server support for RoCE Express2 feature, see “Trace options for the VIT” on page 11.
• Improved control over default VTAM VIT options, see the following topics:
  – Chapter 2, “Using the VTAM internal trace,” on page 7
  – “Activating the VIT” on page 9
  – “Trace options for the VIT” on page 11
  – “Internal and external trace recording for the VIT” on page 18
  – “Deactivating the VIT” on page 21
• VTAM 3270 intrusion detection services, see “Trace options for the VIT” on page 11.

Changes made in z/OS Communications Server Version 2 Release 2, as updated June 2017

This document contains information previously presented in z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT, which supported z/OS Version 2 Release 2.

New information
• Improved control over default VTAM VIT options, see the following topics:
  – “VIT control levels” on page 7
  – “Selecting the level of VIT Control” on page 8
  – “Interaction of VIT option sets and "Full" VIT Control mode processing” on page 8
  – “Example behavior” on page 8
• VTAM 3270 intrusion detection services, see the following topics:
  – “FB64 entry for FREEB64 macro” on page 383
  – “GB64 entry for GETB64 macro” on page 396
  – “3270 entry for 3270 Intrusion Detection Services” on page 718
  – “3271 entry for 3270 Intrusion Detection Services” on page 719

Changed information
• Improved control over default VTAM VIT options, see the following topics:
  – Chapter 2, “Using the VTAM internal trace,” on page 7
  – “Activating the VIT” on page 9
  – “Trace options for the VIT” on page 11
  – “Internal and external trace recording for the VIT” on page 18
  – “Deactivating the VIT” on page 21
• VTAM 3270 intrusion detection services, see “Trace options for the VIT” on page 11.

Changes made in z/OS Version 2 Release 2, as updated September 2016

This document contains information previously presented in z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT, GC27-3668-03, which supported z/OS Version 2 Release 2.

Changes made in z/OS Version 2 Release 2, as updated March 2016

This document contains information previously presented in z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT, GC27-3668-02, which supported z/OS Version 2 Release 2.
New information

- Shared Memory Communications - Direct Memory Access, see the following topics:
  - “ICR entry for a control register operation” on page 434
  - “ICR2 entry for a control register operation (part 2)” on page 435
  - “ICR3 entry for a control register operation (part 3)” on page 435
  - “IPLE entry for an internal shared memory (ISM) polling operation” on page 451
  - “IPLA entry for an internal shared memory (ISM) polling operation (part 2)” on page 452
  - “ISPx entry for invoking an internal shared memory (ISM) Verb (part 1)” on page 460
  - “ISP2 entry for invoking an internal shared memory (ISM) Verb (part 2)” on page 461
  - “ISP3 entry for invoking an internal shared memory (ISM) Verb (part 3)” on page 461

Changed information

- Shared Memory Communications - Direct Memory Access, see the following topics:
  - “AFSM entry for altering an FSM state” on page 196
  - “IOSP entry for invoking a Peripheral Component Interconnect Express (PCIe) service (Part 1)” on page 445
  - “IOS2 entry for invoking a Peripheral Component Interconnect Express (PCIe) service (Part 2)” on page 446
  - “IOS3 entry for invoking a Peripheral Component Interconnect Express (PCIe) service (Part 3)” on page 446
  - “IUTX mapping and field descriptions” on page 466
  - “IUT6 mapping and field descriptions” on page 468
  - “PCIR and PCII mapping and field descriptions” on page 537
  - “QSRB entry for Queue Service Request Block (SRB) events” on page 553
  - “RPST entry for invoking a RoCE Post command (Part 1)” on page 586
  - “RPSA entry for invoking a RoCE Post command (Part 3)” on page 588

Changes made in z/OS Version 2 Release 2

This document contains information previously presented in z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT, GC27-3668-01, which supported z/OS Version 2 Release 1.

New information

- 64-bit enablement of the TCP/IP stack, see the following topics:
  - “ASN6 entry for ASSIGN_BUFFER requests” on page 254
  - “CHG6 entry for CHANGE_OWNER requests” on page 312
  - “FIX6 entry for FIX_BUFFER requests” on page 386
  - “FRB6 entry for FREE_BUFFER requests” on page 389
  - “GCE6 entry for CSM storage movement” on page 399
  - “GTB6 entry for GET_BUFFER requests” on page 411
  - “IUT6 mapping and field descriptions” on page 468
  - “PAG6 entry for PAGE_BUFFER requests” on page 535
Changed information

- 64-bit enablement of the TCP/IP stack, see the following topics:
  - “ASNB entry for ASSIGN_BUFFER requests” on page 252
  - “CHGO entry for CHANGE_OWNER requests” on page 310
  - “CI2 or CO2 entry for SSCP (RUPE - Part 2)” on page 318
  - “FIXB entry for FIX_BUFFER requests” on page 384
  - “FRFB entry for FREE_BUFFER requests” on page 387
  - “GCEL entry for Get Cell requests” on page 397
  - “GCEx entry for CSM storage movement” on page 398
  - “GTBF entry for GET_BUFFER requests” on page 407
  - “IUTx entry for IUT processing (Part 1)” on page 462
  - “ODPK entry for OSA-Express QDIO or HiperSockets packets (Part 1)” on page 518
  - “PAGB entry for PAGE_BUFFER requests” on page 532
  - “QAPL entry for OSA-Express QDIO or HiperSockets accelerator parameter list (Part 1)” on page 547
  - “CSM probes” on page 789

Changes made in z/OS Version 2 Release 1, as updated February 2015

This document contains information previously presented in z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT, GC27-3668-00, which supported z/OS Version 2 Release 1.

New information

- Shared Memory Communications over RDMA adapter (RoCE) virtualization, see the following topics:
  - “CCR entry for a communication channel operation” on page 270
  - “CCR2 entry for communication channel operation (Part 2)” on page 271
  - “VHCR entry for invoking a RoCE VHCR operation (part 1)” on page 682
  - “VHC2 entry for invoking a RoCE VHCR operation (part 2)” on page 683
  - “VHC3 entry for invoking a RoCE VHCR operation (part 3)” on page 684
  - “VHC4 entry for invoking a RoCE VHCR operation (part 4)” on page 684
  - “VHC5 entry for invoking a RoCE VHCR operation (part 5)” on page 685

Changed information

- Shared Memory Communications over RDMA adapter (RoCE) virtualization, see the following topics:
  - “Trace options for the VIT” on page 11
  - “IOS2 entry for invoking a Peripheral Component Interconnect Express (PCIe) service (Part 2)” on page 446
z/OS Version 2 Release 1 summary of changes

See the Version 2 Release 1 (V2R1) versions of the following publications for all enhancements related to z/OS V2R1:

• z/OS Migration
• z/OS Planning for Installation
• z/OS Summary of Message and Interface Changes
• z/OS Introduction and Release Guide
Chapter 1. Using FFST dumps

This topic covers the FFST dumps that you can use for problem determination for the VTAM program. See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for information about other dumps that can be used for problem determination of the VTAM program.

First Failure Support Technology (FFST) for VTAM

First Failure Support Technology is a licensed program that captures information about a potential problem when it occurs. See z/OS Information Roadmap to determine what document contains more information about FFST.

When a problem is detected, a software probe is triggered by VTAM. FFST then collects information about the problem and generates output to help solve the problem. Based on the options active for the probe, you get a dump and a generic alert. See “The generic alert” on page 3 for information about generic alerts. You also get the FFST EPW message group as shown in the “FFST console” on page 4.

FFST dumps

Each VTAM FFST probe can trip up to five times in 5 minutes before it is automatically turned off. Only one of the five dumps will be produced, limiting the number of dumps that you get if a recurring problem triggers a probe.

Depending on how the dump type was coded in the probe, you can get a full dump or an FFST minidump (partial dump). If the dump type is a full dump, only one full dump is created. If the dump type is a minidump, the FFST minidump is created as is a dump containing the last 4 MB of the VIT. For a listing of the dump type for each probe, see Appendix D, “First Failure Support Technology (FFST) probes,” on page 747.

Dump data set

FFST saves the VTAM FFST minidump on a dynamically allocated sequential data set. The VTAM FFST full dump or the partial dump containing the last 4 MB of the VIT (obtained when an FFST minidump is triggered) is saved on SYS1.DUMPx data sets. You must specify the volume serial number and the UNIT identification information for this data set. Provide this information to FFST on a DD statement in the FFST installation procedure or in the FFST startup command list installed at system installation. A startup command list contains MVS commands to control FFST.

Full dump

If a full dump is created when an FFST probe is triggered, FFST uses the operating system SDUMP macroinstruction to provide a full dump of the address space where the potential problem occurred. A full dump includes selected MVS control blocks, CSA, ECSA subpools (227, 228, 231, and 241), and the PSA.

Formatting a full dump

Use IPCS to view or print the full dump. If you try to use EPWDMPFM to format a full dump, message EPW9561E NOT A VALID FFST DUMP will be issued.

FFST minidump

If the probe is coded as a minidump, an FFST minidump is written to the output data set. See “Sample FFST minidump” on page 2.

An FFST minidump contains general-purpose registers, and selected VTAM control blocks. When an FFST minidump is triggered, an additional dump providing the last 4 MB of the 64-bit common VIT table is also generated by using the system SDUMPX macroinstruction. If the VIT table is larger than 4 MB, the dump contains the VIT table header with the last 4 MB of the VIT.
The probe output data used for VTAM FFST minidumps are found in the data sets that were allocated when VTAM FFST was installed.

**Formatting an FFST minidump**

Use the dump formatting CLIST, EPWDMPFM, to format your VTAM FFST minidump. EPWDMPFM formats your minidump and writes it to a data set that you can view online or print using the IEBPTPCH utility program. (FFST minidumps cannot be processed by the VTAM formatted dump tool.)

**Sample FFST minidump**

See "Sample FFST minidump" for a sample VTAM FFST minidump. "Sample FFST minidump" was produced when VTAM session services CP-CP (SSC) entered an unexpected state on a contention-winner session.

### Sample FFST minidump

**SCUNO - FAILURE ON CONWINNER SESSION**

11/09/92

---

EPW9521I DUMP DATA SET NAME = FFSTDS.MVS42247.VTAM.DMP00033
EPW952I TITLE FROM DUMP = SCUNO - FAILURE ON CONWINNER SESSION
EPW9521 DATE FROM DUMP = 01/11/95, TIME FROM DUMP = 12:00:06
EPW9501I PRODUCT NAME: VTAM
EPW9502I IBM PROGRAM
EPW9503I COMPONENT/PROGRAM ID: 569511701, LEVEL: 301
EPW9504I TYPE OF FAILURE: INCORROUT

REG00 - PROBE PRIMARY SYMPTOM STRING:

PIDS/569511701 LVLS/301 PCSS/ISTSCC09 RIDS/ISTSCUNO

REG01 - REGISTER SECONDARY SYMPTOM STRING:

REGS/GR13 VALU/H0686098 REGS/GR14 VALU/H06860CA REGS/GR15 VALU/H06000000 REGS/GR16 VALU/H06860BA REGS/GR17 VALU/H06860EE REGS/GR18 VALU/H06860EE REGS/GR19 VALU/H06860EE REGS/GR20 VALU/H06860EE

REG02 - DATA COLLECTION WORK AREA:

ASID(X'0012') ADDRESS(060B1000) KEY(00)

REG03 - SPECIFIED DATA STRUCTURE TABLE:

ASID(X'0012') ADDRESS(06999BE8) KEY(00) PREFIXED

REG04 - DEFAULT DATA STRUCTURE TABLE:

ASID(X'0012') ADDRESS(063859BE) KEY(00)

REG05 - AREA AROUND REGISTER 0 - 868D40CA:

ASID(X'0012') ADDRESS(060D03CA) KEY(00)

REG06 - AREA AROUND REGISTER 15 - 00000000:

ASID(X'0012') ADDRESS(00000000) KEY(00) PREFIXED

REG07 - VTAM COMMUNICATION VECTOR TABLE

ASID(X'0012') DATA STRUCTURE DEFINED IN DEFAULT DST

พร้อมจะอธิบายข้อมูลเพิ่มเติมเกี่ยวกับ VTAM FFST Dumps and the VIT หรือมีข้อสงสัยเรื่องมันต่าง ๆ ผู้อ่านสามารถติดต่อเราได้
The generic alert

A software generic alert is built from the symptom record and routed to the NetView program if installed. The generic alert contains:

- The date and time that the probe was triggered
- The system name from the CVTSNAME field
- The product name (VTAM)
- The component identification and release number of the product triggering the probe
- The hardware identification information:
  - Machine type
  - Serial number
  - Model number
  - Plant code
- The dump data set and volume if a dump was taken
- The probe statement identifier
- The probe statement description
- The probe statement severity level

Using FFST dumps 3
The symptom string

The primary symptom string contains the following data supplied by VTAM:

- **PIDS/component ID**: The VTAM component identifier
- **LVLS/level**: The VTAM specification for the product level
- **PCSS/Probe ID**: From the probe that was triggered
- **PCSS/FULL or MINI**: The type of dump taken
- **RIDS**: Module name from the probe that was triggered

FFST console

See Figure 1 on page 4 for a sample console listing for FFST. In Figure 1 on page 4 the FFST program console message group EPW shown informs you that a probe has been triggered and that data is being collected. The **EPW0404I** messages contain the primary symptom string for VTAM.

```
EPW0401I  FFSTPROC: ERROR DETECTION INVOKED BY VTAM 287
EPW0406I  DUMP DATASET IS: USER1.SP41D23.VTAM.DMP00002
EPW0407I  FOUND ON VOLUME: CPDLB2
EPW0404I  Primary symptom string for VTAM follows:
EPW0404I  PIDS/569511701 LVLS/301 PCSS/ISTTSC01 PCSS/VR#HANG PCSS/MINI
EPW0404I  RIDS/ISTTSCRI FLDS/TH4VRSSN VALU/H0000 FLDS/VRBSQRCV
EPW0404I  VALU/H0000 FLDS/VRBDSTSA VALU/H00000000
EPW0701I  END OF MESSAGE GROUP
```

**Figure 1. Sample VTAM FFST console listing**

Using the trap module

The IBM-supplied trap module (ISTRACZT) contains pretested probes that you can use to capture data in places where a probe has not been installed inside of VTAM.

**Before you begin**

You need to obtain the VTAM module name and offset in the VTAM module from IBM Service.

**Procedure**

Perform the following steps to install the trap program:

1. Add an instruction to check the ATCFFST field in the ISTATCVT control block. If ATCFFST is 0, VTAM FFST is not available.

```
BALR R14,R15 -------- 05EF
DC   X'0001' -------- 0001 (default)
```

A 2-byte field containing X'0001' or X'0002' follows the BALR instruction. A value of X'0001' issues the ISTRAC01 probe macro in VTAM with the SDUMP option. An index value of X'0002' issues the ISTRAC01 probe macroinstruction with the VTAM FFST minidump option. The ISTATCVT and the VIT
are included in the minidump output along with the VTAM module list, which contains the five significant letters of a module name, its service level, and its address. Control is returned to VTAM at the address after the 2-byte index.

Results
You know that you are done when you execute the program and the trap is triggered, resulting in an SDUMP or FFST minidump (depending on the option chosen).

Notes:
1. The calling module must save GP register 7.
2. The module is reentrant.
3. AMODE is 24 or 31.
4. All registers except 7 are saved and restored.
5. The module is in LPALIB.
6. If you are trying to invoke the trap module outside of the VTAM environment, you must take the following steps:
   • Follow the list shown above.
   • Be in VTAM key (6).
   • Be authorized.

When to dump coupling facility structures
When using GR, MNPS, TSO/GR, TCP/IP Sysplexports, or TCP/IP Sysplex Wide Security Associations be sure to dump the coupling facility structures involved when documenting problems with those functions. See z/OS MVS System Commands for information about dumping a coupling facility structure.

Activating an APPC sense code trap
The IBM-supplied trap is placed in the VTAM APPC component. This trap contains a pretested probe that provides a full dump when a predetermined sense code is set. The sense code will trigger probe ISTRACZ3.

Before you begin
You need to determine the sense code to be trapped. Sense codes are listed in z/OS Communications Server: IP and SNA Codes.

Procedure
Perform the following steps to activate a trap:
1. Ensure that the FFST program is operational.
2. Ensure that the APPC VIT option is active.
3. Obtain the offset of the ATCASLIP field in the ATCVT. (This field is a fullword.)
4. If you are running an MVS guest on a VM system, using the CP TRACE STORE command, set ATCASLIP to the required sense code.
Results

You know that you are done when the FFST probe ISTRACZ3 is triggered.

Phantom First Failure Support Technology (PFFST) for VTAM or CSM

Phantom First Failure Support Technology (PFFST) captures information about a potential problem when it occurs if FFST is not installed or active. See z/OS Information Roadmap to determine what document contains more information about FFST.

See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for information about other dumps that can be used for problem determination of the VTAM program or CSM.

When a problem is detected, a software probe is triggered by VTAM or CSM. If FFST is not installed or active, then PFFST collects information about the problem and generates output to help solve the problem.

Reporting a problem

Some of the probes that are triggered are not VTAM problems. Analyze the probe output to determine whether a VTAM problem exists before you contact an IBM Support Center representative. For more information about FFST probes, see Appendix D, “First Failure Support Technology (FFST) probes,” on page 747.

Some probes might be triggered by VTAM problems that have already been fixed. To determine whether a problem has already been solved, take the following steps:

1. Search the RETAIN database for occurrences of the symptom string you receive when the probe is triggered. [Also search the Information System (I/S) database or the database used by your organization.]
2. If you find an APAR that applies, apply the fix.
3. If you do not find an APAR and you cannot fix the problem, report it.

For non-VTAM problems, call your IBM branch office. For suspected VTAM problems, do either of the following steps:

- Access IBMLink and search for a similar problem by using the symptom string. If no matches are found, report the problem to IBM by using the electronic technical report (ETR) option on IBMLink.
- Contact the IBM Software Support Center at 1-800-IBM-SERV.

If you call the IBM Support Center, the Center must verify that the documentation collected is adequate to fix the problem and that the problem is a VTAM problem.

If the problem is a VTAM problem, the IBM Support Center opens an APAR against VTAM and includes the symptom string generated by the probe as part of the APAR text.

If the problem is a hardware, network definition, or user definition error, the IBM Support Center representative creates an ASKQ item for VTAM. The ASKQ item includes the symptom string and the solution for the problem, and can be found in the problem determination database (PDDB).
Chapter 2. Using the VTAM internal trace

Most VTAM traces show the information flow between the VTAM program and other network components. However, the VTAM internal trace (VIT) provides a record of the sequence of events within VTAM. These internal events include the scheduling of processes (for example, POST, WAIT, and DISPATCH), the management of storage (for example, VTALLOC), and the flow of internal PIUs between VTAM components.

Together with the operator console listing and a dump, output from the VIT can help you reconstruct sequences of VTAM events and find internal VTAM problems more easily.

This topic includes the following information:

- “Activating the VIT” on page 9
- “Trace options for the VIT” on page 11
- “Internal and external trace recording for the VIT” on page 18
- “Recording SNAP traces” on page 20
- “Deactivating the VIT” on page 21
- “Extracting VIT information from a dump” on page 23
- “Using module names to isolate VTAM problems” on page 23

VIT control levels

VTAM provides two levels of operator control for managing the VIT in internal mode. You can select what level of control is appropriate for your environment. The level of controls primarily affect certain default VIT options (API, CIO, MSG, NRM, PIU, and SSCP). This set of options represent the minimum required options for diagnosing VTAM problems. You can choose one of following two levels of control:

- **Base Control**
  You can choose to allow VTAM to enforce that certain default VIT options remain active at all times. This is the default setting. With this level of control, you cannot use the MODIFY NOTRACE command or the NOTRACE start option to disable these VIT options. If you attempt to disable these VIT options, VTAM accepts the command or start option but immediately re-enables the VIT options.

  When using this “base” level of control, VTAM does not always display the setting of these default VIT options as output to the DISPLAY TRACE, MODIFY TRACE and MODIFY NOTRACE commands. The settings are not considered user controllable and are therefore only displayed if you have explicitly enabled the VIT option by using the TRACE start option or the MODIFY TRACE command. If you later explicitly disable the VIT option by using the MODIFY NOTRACE command, the VIT option is no longer displayed but remains enabled.

- **Full Control**
  You can choose the ability to disable individual VIT options at any time by using the MODIFY NOTRACE command or the NOTRACE start option.

  When using this full level of control, VTAM always displays the current setting of these default VIT options as output to the DISPLAY TRACE, MODIFY TRACE and MODIFY NOTRACE commands.

If you use the VIT in external mode, you always can disable all VIT options.

**Result:** Disabling any of the default VIT options for internal VIT processing can trigger a health check notification (CSV TAM_VIT_OPT_STDOPTS) and impacts VTAM serviceability.
Selecting the level of VIT Control

You can specify the level of VIT control you want by using the VITCTRL VTAM start option.

**VITCTRL=BASE**

Specifies that the operator cannot modify the settings of VIT options API, PIU, SSCP, MSG, NRM and CIO by using the MODIFY TRACE and MODIFY NOTRACE commands or by start options for TYPE=VTAM,MODE=INT processing. In addition, the PSS start option is also started by default, but the operator can modify the use of the PSS VIT option by using the MODIFY TRACE or MODIFY NOTRACE command.

This is the default value.

**VITCTRL=FULL**

Specifies that the operator can modify the settings of all VIT options by using the MODIFY TRACE and MODIFY NOTRACE commands or by start options for TYPE=VTAM,MODE=INT processing.

See VTAM start options in z/OS Communications Server: SNA Resource Definition Reference for additional details.

Interaction of VIT option sets and "Full" VIT Control mode processing

If you specify any VIT option set for internal VIT processing on the TRACE start option or on a MODIFY TRACE command, and you are operating in VITCTRL=FULL mode, VTAM also activates the STDOPTS VIT option set.

For example, the APIOPTS option set includes the API, MSG, NRM, PIU, PSS, SMS, and SSCP VIT options. If you specify MODIFY TRACE,TYPE=VTAM,MODE=INT,OPT=(APIOPTS), VTAM enables tracing for those VIT options, but also for the CIO VIT option, because CIO is part of the STDOPTS option set.

You can disable VIT tracing in this example by using any of the following methods:

- Issue the `MODIFY NOTRACE,TYPE=VTAM,MODE=INT,OPT=ALL` command.
- Issue the `MODIFY NOTRACE,TYPE=VTAM,MODE=INT,OPT=END` command.
- Issue two MODIFY NOTRACE commands:
  - Issue the `MODIFY NOTRACE,TYPE=VTAM,MODE=INT,OPT=APIOPTS` command. This command disables all VIT options in APIOPTS option set that are not also in the STDOPTS option set (basically the SMS VIT option).
  - Issue the `MODIFY NOTRACE,TYPE=VTAM,MODE=INT,OPT=STDOPTS` command. This command disables the remaining VIT options.

Note: When a CSDUMP message or code trigger is active, the VIT MSG option cannot be disabled by the MODIFY NOTRACE command.

Example behavior

Use this table to compare and contrast the two levels of VIT control for VIT processing in internal mode.

<table>
<thead>
<tr>
<th>Action</th>
<th>&quot;Base&quot; VIT control</th>
<th>&quot;Full&quot; VIT control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start VTAM without TRACE, TYPE=VTAM</td>
<td>• VTAM initializes the VIT with the STDOPTS set of VIT options.</td>
<td>• VTAM initializes the VIT with the STDOPTS set of VIT options.</td>
</tr>
<tr>
<td></td>
<td>• DISPLAY TRACE output displays &quot;PSS&quot; as the only active VIT option</td>
<td>• DISPLAY TRACE output displays all seven default VIT options as being active.</td>
</tr>
</tbody>
</table>
### Activating the VIT

You must activate the VIT to record the trace data of the specific events.

- When VTAM is operating with VITCTRL=BASE, you do not need to activate the trace data for the following events, because the data is always automatically recorded in the internal table:
  - API

<table>
<thead>
<tr>
<th>Action</th>
<th>&quot;Base&quot; VIT control</th>
<th>&quot;Full&quot; VIT control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start VTAM with TRACE, TYPE=VTAM, MODE=INT, OPT=(APIOPTS) start option.</td>
<td>• VTAM activates the VIT with all the VIT options defined in the APIOPTS option set, plus the CIO VIT option from the default set.</td>
<td>• VTAM activates the VIT with all the VIT options defined in the APIOPTS option set, plus the CIO VIT option from the default set.</td>
</tr>
<tr>
<td></td>
<td>• DISPLAY TRACE output displays all the VIT options in the APIOPTS option set as active, but does not display CIO.</td>
<td>• DISPLAY TRACE output displays all the VIT options in the APIOPTS option set, plus the CIO option, as active.</td>
</tr>
<tr>
<td>Start VTAM with TRACE, TYPE=VTAM, MODE=INT, OPT=(NRM, PIU)</td>
<td>• VTAM initializes the VIT with the following set of VIT options: API, CIO, MSG, NRM, PIU, SSCP and PSS.</td>
<td>• VTAM initializes the VIT with just the NRM and PIU VIT options.</td>
</tr>
<tr>
<td></td>
<td>• DISPLAY TRACE output displays &quot;NRM PIU PSS&quot; as the active VIT options.</td>
<td>• DISPLAY TRACE output displays &quot;NRM PIU&quot; as the active VIT options.</td>
</tr>
<tr>
<td>Assuming the STDOPTS VIT options are active by default, issue MODIFY NOTRACE, TYPE=VTAM, MODE=INT, OPT=(PSS, SSCP, API)</td>
<td>• Before the MODIFY command, DISPLAY TRACE output indicates only the &quot;PSS&quot; option is active.</td>
<td>• Before the MODIFY command, DISPLAY TRACE output indicates all the STDOPTS options as active.</td>
</tr>
<tr>
<td></td>
<td>• VTAM turns off the PSS VIT option as part of MODIFY processing, but the SSCP and API options remain active.</td>
<td>• VTAM turns off the specified VIT options as part of MODIFY processing.</td>
</tr>
<tr>
<td></td>
<td>• The output for the MODIFY NOTRACE command indicates that no VIT options are active.</td>
<td>• The output for the MODIFY NOTRACE command indicates that &quot;CIO MSG NRM PIU&quot; options are active.</td>
</tr>
<tr>
<td>Assuming the STDOPTS VIT options are active by default, issue MODIFY NOTRACE, TYPE=VTAM, MODE=INT, OPT=ALL</td>
<td>• Before the MODIFY command, DISPLAY TRACE output indicates only the &quot;PSS&quot; option is active.</td>
<td>• Before the MODIFY command, DISPLAY TRACE output indicates all the STDOPTS options as active.</td>
</tr>
<tr>
<td></td>
<td>• VTAM turns off the PSS VIT option as part of MODIFY processing, but the remaining default options are unaffected.</td>
<td>• VTAM turns off all VIT options as part of the MODIFY processing.</td>
</tr>
<tr>
<td></td>
<td>• The output for the MODIFY NOTRACE command indicates that no options are active (&quot;NONE&quot;).</td>
<td>• The output for the MODIFY NOTRACE command indicates that no options are active (&quot;NONE&quot;).</td>
</tr>
</tbody>
</table>

---

**Activating the VIT**

You must activate the VIT to record the trace data of the specific events.

- When VTAM is operating with VITCTRL=BASE, you do not need to activate the trace data for the following events, because the data is always automatically recorded in the internal table:
  - API
Specifying **TRACE TYPE=VTAM,MODE=INT,OPT=STDOPTS** is the equivalent of taking the default for internal VIT tracing. Except for PSS events, the events remain enabled for tracing for internal VIT even if you specify the events on a MODIFY NOTRACE command.

- When VTAM is operating with VITCTRL=FULL, use one of the following options to enable tracing for the default VIT options:
  - Specify no TRACE start option when starting VTAM. VTAM will by default enable tracing for the STDOPTS events for internal VIT.
  - Specify **TYPE=VTAM,MODE=INT,OPTION=STDOPTS** for the TRACE start option when starting VTAM or as the operand on a MODIFY TRACE command. This is the equivalent of explicitly coding **TYPE=VTAM,MODE=INT,OPT=(API,CIO,MSG,NRM,PIU,PSS,SSCP)**.
  - Specify **TYPE=VTAM,MODE=INT,OPTION=<any VIT option set>** for the TRACE start option when starting VTAM or as the operand on a MODIFY TRACE command. VTAM enables tracing for both the events defined in the specified VIT option set and for all the events in the STDOPTS option set.

    For example, the APIOPTS option set includes the API, MSG, NRM, PIU, PSS, SMS, and SSCP VIT options. If you specify **MODIFY TRACE,TYPE=VTAM,MODE=INT,OPTION=(APIOPTS)**, VTAM enables tracing for those VIT options, but also for the CIO VIT option, because CIO is part of the STDOPTS option set.

    If you specify any other value for the TRACE start option, or on the MODIFY TRACE command, VTAM enables just the events that you specified on the OPTION operand.

To activate the internal trace, do one of the following actions:

- If you have not started VTAM and you are starting VTAM and the VIT at the same time, use the TRACE start option and specify the following operands:
  - **TYPE=VTAM**
  - **OPTION=VIT_option**
    
    See “Trace options for the VIT” on page 11 for information about how to specify the OPTION operand to select VIT options.
  - **MODE=VIT_mode**
    
    See “Internal and external trace recording for the VIT” on page 18 for information about how to specify the OPTION operand to select VIT options.

- If you have already started VTAM, use the MODIFY TRACE command and specify the following operands:
  - **TYPE=VTAM**
  - **OPTION=VIT_option**
    
    See “Trace options for the VIT” on page 11 for information about how to specify the OPTION operand to select VIT options.
  - **MODE=VIT_mode**
    
    See “Internal and external trace recording for the VIT” on page 18 for information about how to specify the OPTION operand to select VIT options.

**Notes:**
To prevent the VIT table from being overwritten, VTAM disables the internal VIT when it issues SDUMP and when an FFST probe is tripped.

The minimum trace table size is 4 megabytes. If the trace option default values are running, the table might wrap many times.

CIDCTL FIND macro invocations that are invoked during the process of sending or receiving data are not traced with CDHF or CDNF trace entries unless they result in a nonzero return code.

If you want to use VIT to record 32, 64, 96, or 128 bytes of user-generated information in an SNAP trace, see “Recording SNAP traces” on page 20.

Trace options for the VIT

You can specify the OPTION operand in the TRACE start option or in the MODIFY TRACE command. Deactivate the VIT before you attempt to change an option; otherwise, the options that are currently in effect will remain in effect. See “Deactivating the VIT” on page 21 for more information about deactivating the VIT.

Table 1 on page 11 describes the options that you can specify on the OPTION operand. Select one or more of these options to indicate the VTAM functions you want to trace.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API option (for application programming interfaces)</td>
<td>This option helps you determine whether an application program is causing a problem. API entries are written for RPL macros, RPL exit routines, user exit routines, and user posts.</td>
</tr>
<tr>
<td>APIOPTS option</td>
<td>This option is a collection of multiple VIT options that includes all the individual VIT options required to diagnose potential application program problems. Specifying the APIOPTS option is equivalent to specifying all the following VIT options: API, MSG, NRM, PIU, PSS, SMS, and SSCP.</td>
</tr>
<tr>
<td>APPC</td>
<td>This option helps you determine whether an LU 6.2 application is causing a problem. LU 6.2 entries are written for APPCCMD macro invocations, user posts, and exit scheduling by LU 6.2 code, calls to a security manager for security processing, and message unit transmissions between LU 6.2 components.</td>
</tr>
<tr>
<td>APPCOPTS option</td>
<td>This option is a collection of multiple VIT options that includes all the individual VIT options required to diagnose potential LU 6.2 application program problems. Specifying the APPCOPTS option is equivalent to specifying all the following VIT options: API, APPC, MSG, NRM, PIU, PSS, SMS, and SSCP.</td>
</tr>
<tr>
<td>CFS option (for coupling facility interfaces)</td>
<td>This option helps you determine problems with the VTAM interface with the MVS coupling facility. CFS entries are written when VTAM issues MVS macros to request services related to the coupling facility.</td>
</tr>
<tr>
<td>CIA option (for channel input and output auxiliary)</td>
<td>This option helps you isolate problems related to channel I/O CIA entries. This option presents the remaining trace records from the CIO option.</td>
</tr>
<tr>
<td>CIO option (for channel input and output)</td>
<td>This option helps you isolate problems related to channel I/O. CIO entries are written for attentions, error recovery, interruptions, HALT I/O SVC, and START I/O SVC.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| **CMIP option** (for Common Management Information Protocol Services) | Setting the CMIP option enables the following traces:  
  - Calls from CMIP application programs to the management information base (MIB) application programming interface (API)  
  - Calls to the read-queue exit of the CMIP application program  
  - Topology updates from VTAM resources  
You can use the CMIP option to help you determine whether there is a problem in VTAM or in a CMIP application program.                                                                                                                                                                                                                                                                                  |
| **CPCPOPTS option**       | This option is a collection of multiple VIT options that includes all the individual VIT options required to diagnose potential CP-CP session problems. Specifying the CPCPOPTS option is equivalent to specifying all the following VIT options: API, APPC, MSG, NRM, PIU, PSS, SMS, and SSCP.                                                                                                                                                                                                 |
| **CSM option** (for communications storage manager events) | This option traces the parameter list information that flows across the CSM interface and key internal events (such as pool expansion and contraction) for functions that manipulate buffer states. You can trace and analyze the usage history of a buffer.  
You can also use the CSM trace when VTAM is not operational. An external trace is generated using the VTAM GTF event ID to write trace records directly to GTF in the same format as those recorded using VIT.                                                                                                                                                                                                                                             |
| **CSMOPTS option**        | This option is a collection of multiple VIT options that includes all the individual VIT options required to diagnose potential communications storage manager (CSM) problems. Specifying the CSMOPTS option is equivalent to specifying all the following VIT options: API, APPC, CIO, CSM, MSG, NRM, PIU, PSS, SMS, SSCP, and XBUF.                                                                                                                                                                                                 |
| **DLUROPTS option**       | This option is a collection of multiple VIT options that includes all the individual VIT options required to diagnose dependent LU requester (DLUR) problems. Specifying the DLUROPTS option is equivalent to specifying all the following VIT options: API, APPC, HPR, MSG, NRM, PIU, PSS, SMS, and SSCP.                                                                                                                                                                                                 |
| **EEOPTS option**         | This option is a collection of multiple VIT options that includes all the individual VIT options required to diagnose Enterprise Extender (EE) problems. Specifying the EEOPTS option is equivalent to specifying all the following VIT options: CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, SSCP, and TCP.                                                                                                                                                                                                                      |
| **ESC option** (for execution sequence control) | This option helps you track, in detail, the flow of requests for a given process.                                                                                                                                                                                                                                                                                                                                                      |
| **HPDTOPTS option**       | This option is a collection of multiple VIT options that includes all the individual VIT options required to diagnose high-performance data transfer (HPDT) problems. Specifying the HPDTOPTS option is equivalent to specifying all the following VIT options: CIA, CIO, HPR, MSG, PIU, PSS, SMS, SSCP, and TCP.                                                                                                                                 |
| **HPR option** (for High-Performance Routing) | This option helps you isolate problems related to High-Performance Routing.                                                                                                                                                                                                                                                                                                                                                                                                                                        |
### Table 1. Trace options of the OPTION operand (continued)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HPROPTS option</strong></td>
<td>This option is a collection of multiple VIT options that includes all the individual VIT options required to diagnose High-Performance Routing (HPR) problems. Specifying the HPROPTS option is equivalent to specifying all the following VIT options: API, APPC, CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, and SSCP.</td>
</tr>
<tr>
<td><strong>LCS option</strong></td>
<td>This option helps you isolate problems that occur when an IBM 3172 Interconnect Nways Controller is activating, deactivating, or transferring data. The LCS option enables tracing of data that VTAM receives from an IBM 3172 Interconnect Nways Controller at four levels: LCSX (channel), LCSP (port or adapter), LCSS (SAP), and LCSL (line).</td>
</tr>
<tr>
<td><strong>LCSOPTS options</strong></td>
<td>This option is a collection of multiple VIT options that includes all of the individual VIT options required to diagnose LAN channel station (LCS) problems. Specifying the LCSOPTS option is equivalent to specifying all the following VIT options: CIO, LCS, MSG, NRM, PIU, PSS, SMS, and SSCP.</td>
</tr>
<tr>
<td><strong>LOCK option</strong></td>
<td>This option helps you determine when VTAM modules obtain and release locks.</td>
</tr>
<tr>
<td><strong>MSG option</strong></td>
<td>Specify this option to accomplish the following tasks:</td>
</tr>
<tr>
<td></td>
<td>• Correlate other VIT entries with the console messages, even if you lose the console sheet. MSG entries are written for all messages to the VTAM operator.</td>
</tr>
<tr>
<td></td>
<td>• Match the console log to a surge of activity shown in the VIT. OPER entries are written for all VTAM commands issued at an operator console.</td>
</tr>
<tr>
<td><strong>NRM option</strong></td>
<td>This option helps you follow the services of the network resource management component. These services include the assignment of, references to, and the deletion of certain VTAM resources such as node names, network addresses, and control blocks. NRM entries are written for SRT macros issued by VTAM modules. CIDCTL FIND macro invocations used during the process of sending or receiving data are not traced with CDHF or CDNF trace entries unless they result in a nonzero return code.</td>
</tr>
<tr>
<td><strong>PIU option</strong></td>
<td>This option, like the I/O and buffer contents traces, helps you isolate problems to hardware, to the NCP, or to VTAM. Unlike I/O and buffer contents traces, this option causes PIU entries to be written for all PIUs that flow internal and external to VTAM.</td>
</tr>
<tr>
<td><strong>PSS option</strong></td>
<td>This option helps you track the flow of requests through VTAM. PSS entries are written for the VTAM macros that invoke and control PSS, scheduling, and dispatching VTAM routines.</td>
</tr>
<tr>
<td><strong>QDIOOPTS options</strong></td>
<td>This option is a collection of multiple VIT options that includes all of the individual VIT options required to diagnose queued direct I/O (QDIO) problems. Specifying the QDIOOPTS option is equivalent to specifying all the following VIT options: CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, and SSCP.</td>
</tr>
</tbody>
</table>

**Using the VTAM internal trace** 13
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SMS</strong> option (for storage management services)</td>
<td>This option helps you isolate problems caused by storage shortages. When you specify this option with the SSCP or PSS trace option, it can also help you isolate internal VTAM problems. SMS entries are written when SMS macros are used to request or free fixed-length or variable-length buffers. SMS entries are also written when VTAM expands or attempts to expand a buffer pool.</td>
</tr>
<tr>
<td><strong>SSCP</strong> option (for system services control point request scheduling and response posting)</td>
<td>This option helps you isolate a VTAM problem to a specific VTAM component or module. SSCP entries are written for the request/response units (RUs) sent between VTAM components. This option also records information for the APPN CP.</td>
</tr>
<tr>
<td><strong>STDOPTS</strong> option</td>
<td>This option is a collection of multiple VIT options that includes all of the individual VIT options required to diagnose problems related to high CPU, session services, Open/Close ACB, and D LCS such as multipath channel (MPC) and channel-to-channel (CTC). Specifying the STDOPTS option is equivalent to specifying all the following VIT options: API, CIO, MSG, NRM, PIU, PSS and SSCP. STDOPTS is the default trace options. When VTAM is operating in VITCTRL=FULL mode, recording for the events in the STDOPTS VIT option set is also enabled when any other group option set is enabled. Additionally during VTAM start processing with both a CSDUMP and VITCTRL=FULL start option defined, recording for the events in the STDOPTS VIT option is enabled.</td>
</tr>
<tr>
<td><strong>TCP</strong> option (for use with Enterprise Extender)</td>
<td>This option is used for recording activity related to Enterprise Extender. The trace options record IP address management and timer activity.</td>
</tr>
<tr>
<td><strong>TCPOPTS</strong> option</td>
<td>This option is a collection of multiple VIT options that includes all of the individual VIT options required to diagnose problems related to TCP/IP. Specifying the TCPOPTS option is equivalent to specifying all the following VIT options: CIA, CIO, MSG, NRM, PIU, PSS, SMS, SSCP, and TCP.</td>
</tr>
<tr>
<td><strong>VCNS</strong> option (for VCNS application programming interfaces)</td>
<td>This option helps you determine whether a VCNS application is causing a problem. VCNS entries are written for VCNSCMD macro invocations, user posts, exit scheduling by VCNS code, and work element transmissions between VCNS components.</td>
</tr>
<tr>
<td><strong>XBUF</strong> option (for applications that use the extended buffer list for sending and receiving data)</td>
<td>This option traces the contents of the extended buffer list (XBUFLST). Records are produced to trace these contents from the application-supplied extended buffer list and the internal buffer list that VTAM uses to carry the extended buffer list information. These records store relevant information contained with the extended buffer list, particularly information about CSM usage by VTAM.</td>
</tr>
<tr>
<td><strong>XCF</strong> option (for VTAM use of the cross-system coupling facility)</td>
<td>Specify this option to track VTAM use of the XCF (cross-system coupling facility) MVS macro interface. Each VTAM use of an XCF macro has a VIT entry.</td>
</tr>
</tbody>
</table>
Table 1. Trace options of the OPTION operand (continued)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XCFOPTS</td>
<td>This option is a collection of multiple VIT options that includes all of the individual VIT options required to diagnose cross-system coupling facility (XCF) problems. Specifying the XCFOPTS option is equivalent to specifying all the following VIT options: CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, SSCP, and XCF.</td>
</tr>
</tbody>
</table>

The VIT always traces the exception conditions listed in Table 2 on page 15 and all the default VIT options listed under “Activating the VIT” on page 9.

Table 2. Exception conditions always traced by the VIT

<table>
<thead>
<tr>
<th>Option</th>
<th>Exception conditions traced</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPC</td>
<td>• ACA and ACI entries when following commands are issued:</td>
</tr>
<tr>
<td></td>
<td>– SEND ERROR</td>
</tr>
<tr>
<td></td>
<td>– DEALLOC ABNDxxxx</td>
</tr>
<tr>
<td></td>
<td>– REJECT</td>
</tr>
<tr>
<td></td>
<td>• ACRC and ACSN entries</td>
</tr>
<tr>
<td></td>
<td>• Other entries with nonzero return codes (except RPL6RCSC)</td>
</tr>
<tr>
<td>CFS</td>
<td>Entries with nonzero return codes</td>
</tr>
<tr>
<td>CIO</td>
<td>INOP entry</td>
</tr>
<tr>
<td>CMIP option</td>
<td>The following entries, when they have nonzero return codes:</td>
</tr>
<tr>
<td></td>
<td>• MCO1 and MCO2</td>
</tr>
<tr>
<td></td>
<td>• MDEL</td>
</tr>
<tr>
<td></td>
<td>• MDIS</td>
</tr>
<tr>
<td></td>
<td>• MQRQ</td>
</tr>
<tr>
<td></td>
<td>• MQRS</td>
</tr>
<tr>
<td></td>
<td>• MREG</td>
</tr>
<tr>
<td></td>
<td>• RQE</td>
</tr>
<tr>
<td>LCS</td>
<td>LCSL, LCSP, LCSS, and LCSX entries with nonzero reason codes</td>
</tr>
<tr>
<td>NRM</td>
<td>CDHF or CDNF entries with nonzero return codes</td>
</tr>
<tr>
<td>SMS</td>
<td>Entries with nonzero return codes and EXPN entries if a buffer pool expansion fails</td>
</tr>
<tr>
<td>SSCP</td>
<td>CPI, CPO, and CP2</td>
</tr>
<tr>
<td>(No option)</td>
<td>All SNAP entries and some exception entries“1” on page 15.</td>
</tr>
</tbody>
</table>

Note:
1. The **** (FFST and PFFST), ABND, BUFF, COPY, CMER, CME2, INOP, LOST, MMG, and MM2 trace records are not activated by specific VIT options. They are activated as a result of exception conditions.
Table 3 on page 16 and Table 4 on page 17 list the VIT options and the records that they create. For more information, see the list of notes after Table 4 on page 17.

<table>
<thead>
<tr>
<th>VIT options</th>
<th>API</th>
<th>APPC</th>
<th>CFS</th>
<th>CIA</th>
<th>CIO</th>
<th>CMIP</th>
<th>CSM</th>
<th>ESC</th>
<th>HPR</th>
<th>LCS</th>
<th>LOCK</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT records</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>AIX</td>
<td>AC1x</td>
<td>CFAx</td>
<td>CCR</td>
<td>ADE</td>
<td>MCO1</td>
<td>ESC</td>
<td></td>
<td></td>
<td>LCS</td>
<td>LOCK</td>
</tr>
<tr>
<td></td>
<td>IOx</td>
<td>AC2x</td>
<td>CFCx</td>
<td>CD5Q</td>
<td>ATT</td>
<td>MCO2</td>
<td>ARB</td>
<td>LKE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RE</td>
<td>ACPx</td>
<td>CFDx</td>
<td>C64Q</td>
<td>ERP</td>
<td>MDEL</td>
<td>LKX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>UE</td>
<td>ACRx</td>
<td>CFX</td>
<td>DEVx</td>
<td>HIO</td>
<td>MDIS</td>
<td>LKB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>UX</td>
<td>ACSN</td>
<td>CFFC</td>
<td>INTx</td>
<td>PCX</td>
<td>MQRQ</td>
<td>LKX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MX</td>
<td>ACUT</td>
<td>CFLx</td>
<td>PCIT</td>
<td>MRG</td>
<td>MQRS</td>
<td>LKX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RX</td>
<td>RACR</td>
<td>CFFC</td>
<td>RQX</td>
<td></td>
<td>MREG</td>
<td>LKX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ML</td>
<td>REMQ</td>
<td>CFFx</td>
<td></td>
<td></td>
<td>MRGx</td>
<td>LKX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qx</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>US</td>
<td>CFUS</td>
<td>CFFx</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ESC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Vx</td>
<td>CFVC</td>
<td>CFFx</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LCS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MNPS</td>
<td>CFFx</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNLK</td>
</tr>
</tbody>
</table>

For more information, see the list of notes after Table 4 on page 17.
### Table 4. VIT options and the records they create (MSG - XCF)

<table>
<thead>
<tr>
<th>VIT options</th>
<th>MSG</th>
<th>NRM</th>
<th>PIU</th>
<th>PSS</th>
<th>SMS</th>
<th>SSCP</th>
<th>TCP</th>
<th>VCNS</th>
<th>XBUF</th>
<th>XCF</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT records</td>
<td>MSGx</td>
<td>OPEx</td>
<td>QRYL</td>
<td>TRNx</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BSPx</td>
<td>BSXx</td>
<td>CDHx</td>
<td>NIPx</td>
<td>PROx</td>
<td>RCEx</td>
<td>SRTx</td>
<td>DCOx</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BSSx</td>
<td>NSRx</td>
<td>PIUx</td>
<td>CDNx</td>
<td></td>
<td></td>
<td></td>
<td>DSCx</td>
<td>DTSK</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CDlx</td>
<td></td>
<td>RDSx</td>
<td>EXIT</td>
<td></td>
<td></td>
<td></td>
<td>ETHK</td>
<td>GETS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRNx</td>
<td></td>
<td></td>
<td>IRBx</td>
<td>POST</td>
<td></td>
<td></td>
<td>TRNx</td>
<td>GETS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>QUEx</td>
<td></td>
<td></td>
<td>BSXx</td>
<td>GT64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>REMS</td>
<td></td>
<td></td>
<td>BSSx</td>
<td>ORMG</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SCDM</td>
<td></td>
<td></td>
<td>VSBE</td>
<td>POOF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SBx</td>
<td></td>
<td></td>
<td>VRSW</td>
<td>QREx</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>VAIx</td>
<td></td>
<td></td>
<td>WAIT</td>
<td>RAPx</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>XPST</td>
<td></td>
<td></td>
<td>XPST</td>
<td>RELS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>VTAx</td>
</tr>
</tbody>
</table>

**Note:**

1. The **** (FFST and PFFST), ABND, BUFF, COPY, CMER, CME2, INOP, LOST, MMG, and MM2 trace records are not activated by specific VIT options. They are activated as a result of exception conditions.

2. • For CIO record types ATT, ERP, HIO, INT, SIO, with suffix I, X, or T, and INOP, the events are also captured in the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.
   • For CIA record types INOP, RCPx, RPLx and RPST, the events are also captured in the RUNCB (pointed to by NCBCIOMV).
   • For CIA record type PCIR, the events are also captured in the SRNCB (pointed to by NCBCIOMV).

3. OON and OOX can be generated when the module trace is running.

4. For the IRBx and the SRBx records to be recorded, both the PSS trace option and the PSSTTRACE start options must be specified.

5. For APPC record types REMQ and ACSN, the events are also captured in the ISTRAB.

6. Some trace records are generated only when a subtrace is active. These trace records are the HPR option record types ARBB, ARBR, the CIA option record types QAPL, QDIP, QSRx, RSLK, and the SSCP option record types HLSx, TGVx, TRMx, and TRRx. For more information about subtraces, see z/OS Communications Server: SNA Operation.
Table 5 on page 18 lists the VIT group options and the individual VIT options that are equivalent for each group option.

<table>
<thead>
<tr>
<th>VIT group option</th>
<th>Equivalent to this set of individual VIT options</th>
</tr>
</thead>
<tbody>
<tr>
<td>APIOPTS</td>
<td>API, MSG, NRM, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>APPCOPTS</td>
<td>API, APPC, MSG, NRM, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>CPCPOPTS</td>
<td>API, APPC, MSG, NRM, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>CSMOPTS</td>
<td>API, APPC, CIO, CSM, MSG, NRM, PIU, PSS, SMS, SSCP, XBUF</td>
</tr>
<tr>
<td>DLROPTS</td>
<td>API, APPC, HPR, MSG, NRM, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>EEOPTS</td>
<td>CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, SSCP, TCP</td>
</tr>
<tr>
<td>HPDTOPTS</td>
<td>CIA, CIO, HPR, MSG, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>HPROPTS</td>
<td>API, APPC, CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>LCSOPTS</td>
<td>CIO, LCS, MSG, NRM, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>QDIOPTS</td>
<td>CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>STDOPTS</td>
<td>API, CIA, MSG, NRM, PIU, PSS, SMS, SSCP</td>
</tr>
<tr>
<td>TCPOPTS</td>
<td>CIA, CIO, MSG, NRM, PIU, PSS, TCP</td>
</tr>
<tr>
<td>XCFOPTS</td>
<td>CIA, CIO, HPR, MSG, NRM, PIU, PSS, SMS, SSCP, XCF</td>
</tr>
</tbody>
</table>

**Internal and external trace recording for the VIT**

VTAM can write the VIT trace data to an internal table or to an external device, such as a disk or tape. You can use the MODE operand to choose internal or external trace recording. The MODE operand is available in both the TRACE start option and the MODIFY TRACE command.

- “Recording traces in an internal 64-bit common table” on page 18
- “Recording traces in an external file” on page 20

You can record data externally and internally at the same time. If it is necessary, you can have different sets of trace options active for each mode.

See VIT control levels for details on how VTAM manages the default trace options for internal trace recording.

**Recording traces in an internal 64-bit common table**

To record traces in an internal 64-bit common (HCOMMON) table, which is allocated and initialized in 64-bit common storage, specify the following operands:

- **MODE=INT**
  Specifies that record traces should be recorded in an internal table. The default value of MODE is INT.
  **Note:** You can set this operand either in the MODIFY TRACE command or as a TRACE start option.

- **SIZE**
  Specifies the number of megabytes (4M - 2048M) in storage to be allocated for the internal trace table. If you omit this option, the default value is 4. If there is not enough storage available for the number of megabytes specified, you will receive a message indicating that internal trace activation failed.
You can change the size of the internal trace table by issuing a MODIFY TRACE command with a new SIZE operand. However, if you change the table size while the VIT is running, the current internal trace table is freed, and VTAM starts a new one. The trace information in the current table will be lost.

Example

Figure 2 on page 19 shows an example of VIT records written with MODE=INT and printed in dump output.

Figure 2. Unformatted VIT records in dump output (MODE=INT)

The start of the internal trace table is the header. (The header is not included in the this example.) The header contains the following status information:

**Byte (hex)**

**Contents**

- 00–03
  - Eyecatcher (C’VITH’)

- 04–0F
  - Reserved

- 10–17
  - Control area

- 18–1F
  - Reserved

- 20–27
  - Present wraparound time stamp

- 28–2F
  - Last wraparound time stamp

- 30–37
  - Address of the most recent entry in the table
Recording traces in an external file

To record traces in an external file, specify MODE=EXT in the MODIFY TRACE command or as a TRACE start option.

**Note:** When you specify MODE=EXT, information for the default options is still written to the internal trace table.

Use external mode when you must collect large amounts of trace data, to lessen the chance of wraparound and lost data. However, unless you are using a tape, the trace data can still fill the external DASD or data file and begin overwriting itself if you do not ensure that the external trace file is large enough.

VTAM allocates from common storage area (CSA) the number of 8-K VIT buffers specified on the command or start option. The default value is 2. VTAM now accumulates the VIT records in one of these buffers. When a process has insufficient room in the current buffer, the process writes the entire 8-K buffer to an external trace file using the generalized trace facility (GTF). After the buffer has been written, the process tries again to put its VIT records into the current buffer. If another process finds that there is no current buffer (because of another process writing it to GTF), the process tries to use one of the preallocated buffers. If no other buffers are available, either because they are all in the process of being written to GTF or because BFRNUM=0 was specified, VTAM writes the individual records to the external trace file.

Avoid using BFRNUM=0 because writing to the external trace file involves large system overhead. Also, allot enough 8K buffers to prevent individual records from being written to the external trace file. Depending on the situation, VTAM might write the 8K buffers to the external trace file out of order.

The external trace file contains the same information as the internal trace table, except it does not contain the first-line header record and the BUFF VIT record. The BUFF VIT record is the first VIT record in the 8K buffer. This record contains the buffer sequence number, which you can use to determine whether the 8K buffers were all written in the correct sequence.

The external trace file is produced by GTF, and the default file name is SYS1.TRACE. You can print the internal trace data with IPCS or TAP. If you use IPCS to print the data, specify the GTFTRACE option, and set USR(FE1). See “Formatting and printing trace records” in z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for information about using TAP and IPCS to print output.

Recording SNAP traces

You can use the VIT to record 32, 64, 96, or 128 bytes of user-generated information in an SNAP trace. An SNAP trace places additional information into a VTAM module that might be useful when you are resolving a VTAM problem. The VIT traces the specified bytes and enters the data as an SNAP trace entry in the internal trace table or in the external trace file.

You do not specify the SNAP trace as an option of the MODIFY command. Instead, follow these steps to write your code:

1. Verify that the pointer to the ATCVT (ATCLCPTR) is not 0. ATCLCPTR is at low-storage location X'408'.
2. Verify that ATCRACTR is not 0.
3. Call the internal trace recording routine (ISTRACTR) in a VTAM module. Field ATCRACTR in the ATCVT points to the trace recording routine. For the hex offset, see z/OS Communications Server: SNA Data Areas Volume 1.

**Note:** You cannot record SNAP during the early stages of VTAM initialization, nor can you record them during the late stages of termination.
The SNAP trace requires the following information in registers 1, 14, and 15 on entry to the trace routine:

**Register 1**
Points to the 32, 64, 96, or 128 bytes to be recorded.

**Register 14**
Points to a 2-byte input flag field containing:
- hex 141C for a 32-byte SNAP entry
- hex 142C for a 64-byte SNAP entry
- hex 143C for a 96-byte SNAP entry
- hex 144C for a 128-byte SNAP entry
These flags tell the trace recording routine to record an SNAP trace.

**Two bytes past the address in register 14**
The address to be returned to after recording an SNAP trace.

**Register 15**
Points to the SNAP trace recording routine (ISTRACTR).

When the trace recording routine returns control to the calling module, the contents of registers 6 and 7 are unpredictable. The contents of the other registers are not changed.

The SNAP trace record is unformatted. You should format the first 4 bytes of the SNAP record for a 1- to 4-character EBCDIC record identifier (for example, C"ABCD").

**Guidelines:**
1. The 1- to 4-character EBCDIC record identifier should not match any existing VIT record identifier.
2. For the VIT snap entry to be used by the VIT analysis tool, the VIT entry name must be the first 2 to 4 characters of the VIT entry.

**Deactivating the VIT**

To deactivate specific VIT options, use the corresponding commands that are listed in Table 6 on page 21.

### Table 6. Deactivating the VIT

<table>
<thead>
<tr>
<th>To deactivate these user-selected options</th>
<th>Specify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific options</td>
<td>MODIFY NOTRACE, TYPE=VTAM, OPTION= options</td>
</tr>
<tr>
<td><strong>To deactivate these user-selected options:</strong></td>
<td><strong>Specify:</strong></td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| VIT group options | **MODIFY NOTRACE,TYPE=VTAM,OPTION=group_option**  
Individual VIT options encompassed by each group option specified by the *group_option* value are deactivated. Internal tracing continues for the default options when base VIT control is being used. Tracing also continues for exception records.  
For example, issuing a **MODIFY NOTRACE,TYPE=VTAM,OPTION=EEOPTS** command generates these results:  
- The full set of EEOPTS VIT options (CIA, CIA, HPR, MSG, NRM, PIU, PSS, SMS, SSCP, and TCP) are disabled for external VIT recording.  
- The CIA, HPR, SMS and TCP options are disabled for internal VIT recording, but the remaining options (CIO, MSG, NRM, PIU, PSS, and SSCP) continue to be traced internally because they are part of the STDOPTS default VIT option set.  
  - If VTAM is running in VITCTRL=BASE mode, you cannot disable the remaining VIT options for internal VIT recording.  
  - If VTAM is running in VITCTRL=FULL mode, you can issue **MODIFY NOTRACE,TYPE=VTAM,OPTION=STDOPTS** to disable the remaining events for internal VIT recording. When a CSDUMP message or code trigger is active, the VIT MSG option cannot be disabled by the MODIFY NOTRACE command. |
| External VIT recording | **MODIFY NOTRACE,TYPE=VTAM,OPTION=(..,CIA,..)**  
- If you specify SUBTRACE=ARB, the *vit_option* value must be HPR or one of the group options that include HPR as an individual option equivalent: DLUROPTS, EEOPTS, HPDTOPTS, HPROTOPTS, QDIOOPTS, or XCFOPTS.  
| Internal VIT recording | **MODIFY NOTRACE,TYPE=VTAM,OPTION=(..,CIO,..)**  
- If you specify SUBTRACE=DIO, the *vit_option* value must be CIA or one of the group options that include CIA as an individual option equivalent: EEOPTS, HPDTOPTS, HPROTOPTS, QDIOOPTS, TCPOPTS, or XCFOPTS.  
| **Subtrace options** | **MODIFY NOTRACE,TYPE=VTAM,SUBTRACE=subtrace_option,OPTION=vit_option**  
The subtrace option specified by the *subtrace_option* value is deactivated. The option specified by the *vit_option* value must relate to the *subtrace_option* value in one of the following ways:  
- If you specify SUBTRACE=ARB, the *vit_option* value must be HPR or one of the group options that include HPR as an individual option equivalent: DLUROPTS, EEOPTS, HPDTOPTS, HPROTOPTS, QDIOOPTS, or XCFOPTS.  
- If you specify SUBTRACE=DIO, the *vit_option* value must be CIA or one of the group options that include CIA as an individual option equivalent: EEOPTS, HPDTOPTS, HPROTOPTS, QDIOOPTS, TCPOPTS, or XCFOPTS.  
- If you specify SUBTRACE=TGVC or SUBTRACE=TREE, the *vit_option* value must be SSCP or one of the group options (all of which include SSCP as an individual option equivalent): APIOPTS, APPCOPTS, CPCPOPTS, CSMOPTS, DLUROPTS, EEOPTS, HPDTOPTS, HPROTOPTS, LCSOPTS, QDIOOPTS, STDOPTS, TCPOPTS, or XCFOPTS. |
| Internal only | **MODIFY NOTRACE,TYPE=VTAM,OPTION=END**  
The internal trace table is reallocated to 4 megabytes for the default options. If you are using VITCTRL=BASE, the default VIT options are immediately re-enabled. If you are using VITCTRL=FULL, the default options are disabled.  
| | **MODIFY NOTRACE,TYPE=VTAM,OPTION=ALL,MODE=INT**  
If you are using VITCTRL=BASE, the existing internal trace table is used for the default options. If you are using VITCTRL=FULL, the existing internal trace table is used but the default options are disabled. |
### Extracting VIT information from a dump

Use one of the following methods to analyze dumps.

- Use the dump analysis tools for the VIT:
  
  **VITAL**
  
  Extracts an internal VIT from a dump for use with the VIT analysis tool.

  **VTBASIC**
  
  Displays the VIT table.

  **VTVIT**
  
  Displays the VIT options that were in effect at the time of the dump and whether the trace was running internally, externally, or both.

See "Using VTAM dump analysis tools" in z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for more information about the VTAM dump analysis tools.

- Use this procedure:
  
  1. Examine the internal trace table by taking a dump of VTAM with the appropriate storage area.
  2. Find the internal trace table in the dump by first locating the ATCVT.
     - If low-storage location X'408' is available in the dump, the pointer to the ATCVT (ATCLCPR) is at this low-storage location.
     - If low-storage location X'408' is not available in the dump, use the pointer in the MVS CVT control block extension (CVTATCVT) to find the VTAM control block AVT. Location hex 00 in the AVT points to the ATCVT.
  3. In the ATCVT, field ATCITTBL contains a 64-bit pointer to the trace table. ATCITTBL is located at offset X'8'. The internal trace table contains a 64-byte header followed by 32-byte trace records.

### Using module names to isolate VTAM problems

Many VIT records include the associated module names in EBCDIC, without the IST prefix. For some types of trace records, the sixth letter is also omitted. For example, TSSR is the name of the VIT records for module ISTTSCSR.
You can also determine module names from the ISSR field in some VIT records. If you have a dump, you can find the address in the VTAM module list.

The following names are exceptions to the naming convention:

**CPSS**  
Session Services for CP-CP Sessions PAB

**DII0**  
Disk I/O PAB

**DSME**  
Directory services management exit PAB

**DSVC**  
Directory Services PAB

**LUSS**  
Session Services for LU-LU Sessions PAB

**MSTI**  
MST Multiple Domain Server Router PAB

**MSTR**  
MST Multiple Domain Server Router PAB

**TRS**  
Topology Routing Services PAB

**XPRT**  
Transaction Program PAB

VTAM component IDs are listed in Table 7 on page 24.

**Table 7. VTAM component IDs**

<table>
<thead>
<tr>
<th>ID</th>
<th>Acronym</th>
<th>Component name</th>
</tr>
</thead>
<tbody>
<tr>
<td>@@</td>
<td></td>
<td>All or multiple components</td>
</tr>
<tr>
<td>AC</td>
<td>CONFIG</td>
<td>Activation</td>
</tr>
<tr>
<td>AD</td>
<td>APUNS</td>
<td>Adjacent PU network services</td>
</tr>
<tr>
<td>AI</td>
<td>API</td>
<td>Application programming interface</td>
</tr>
<tr>
<td>AM</td>
<td>ADDR-MGT</td>
<td>Address management</td>
</tr>
<tr>
<td>AP</td>
<td>PSS</td>
<td>Process scheduling services</td>
</tr>
<tr>
<td>AS</td>
<td>APPC-SVC</td>
<td>APPC services</td>
</tr>
<tr>
<td>AT</td>
<td>APPNTGMT</td>
<td>APPN transmission group management</td>
</tr>
<tr>
<td>BS</td>
<td>BFSS</td>
<td>Boundary function session services</td>
</tr>
<tr>
<td>CD</td>
<td>CDRM</td>
<td>Cross domain resource manager</td>
</tr>
<tr>
<td>CF</td>
<td>CFS</td>
<td>Coupling facility services</td>
</tr>
<tr>
<td>CI</td>
<td>OCI</td>
<td>Operator command interface</td>
</tr>
<tr>
<td>CM</td>
<td>CMIP</td>
<td>Common Management Information Protocol</td>
</tr>
<tr>
<td>CN</td>
<td>CNS</td>
<td>Common network services</td>
</tr>
<tr>
<td>ID</td>
<td>Acronym</td>
<td>Component name</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>CO</td>
<td>COPR</td>
<td>Control operator</td>
</tr>
<tr>
<td>CP</td>
<td>SSCP</td>
<td>System services control program functions</td>
</tr>
<tr>
<td>CS</td>
<td>CONFGSVC</td>
<td>Configuration services</td>
</tr>
<tr>
<td>DE</td>
<td>CONFIG</td>
<td>Deactivation</td>
</tr>
<tr>
<td>DL</td>
<td>DLUS</td>
<td>Dependent LU server</td>
</tr>
<tr>
<td>DP</td>
<td>DATACOMP</td>
<td>Data compression</td>
</tr>
<tr>
<td>DR</td>
<td>DS</td>
<td>Directory services</td>
</tr>
<tr>
<td>DS</td>
<td>DSS</td>
<td>Data space services</td>
</tr>
<tr>
<td>EN</td>
<td>ENVIRO</td>
<td>VTAM environment</td>
</tr>
<tr>
<td>EV</td>
<td>ERVR-MGR</td>
<td>Explicit route virtual route (ER/VR) manager</td>
</tr>
<tr>
<td>FD</td>
<td>FMT-DMP</td>
<td>Formatted dump</td>
</tr>
<tr>
<td>FS</td>
<td>CFS</td>
<td>Coupling facility services</td>
</tr>
<tr>
<td>FU</td>
<td>FUNCTION</td>
<td>SSCP functions</td>
</tr>
<tr>
<td>GN</td>
<td>SYSGEN</td>
<td>System generation</td>
</tr>
<tr>
<td>HS</td>
<td>HS</td>
<td>Half-session</td>
</tr>
<tr>
<td>IE</td>
<td>IEF</td>
<td>Inline exit facility</td>
</tr>
<tr>
<td>IM</td>
<td>INT-MAP</td>
<td>Internal mappings</td>
</tr>
<tr>
<td>IN</td>
<td>INIT/TRM</td>
<td>VTAM initialization/termination</td>
</tr>
<tr>
<td>IP</td>
<td>SNAIP</td>
<td>IP network access</td>
</tr>
<tr>
<td>IT</td>
<td>VIT</td>
<td>VTAM internal trace</td>
</tr>
<tr>
<td>IX</td>
<td>INT-MAC</td>
<td>Internal macroinstructions</td>
</tr>
<tr>
<td>LL</td>
<td>LLC</td>
<td>Logical link control</td>
</tr>
<tr>
<td>LM</td>
<td>TPFLOGON</td>
<td>Transaction processing facility (TPF) logon manager</td>
</tr>
<tr>
<td>LR</td>
<td>DLR</td>
<td>Dump/Load/Restart</td>
</tr>
<tr>
<td>LS</td>
<td>LSA</td>
<td>Link services architecture</td>
</tr>
<tr>
<td>LU</td>
<td>LUS</td>
<td>Logical unit services</td>
</tr>
<tr>
<td>MA</td>
<td>MAINT-SV</td>
<td>Maintenance services</td>
</tr>
<tr>
<td>MG</td>
<td>MGMTSVC</td>
<td>Maintenance services</td>
</tr>
<tr>
<td>MS</td>
<td>MGS</td>
<td>Messages</td>
</tr>
<tr>
<td>MT</td>
<td>MST</td>
<td>Management services transport</td>
</tr>
<tr>
<td>NA</td>
<td>NAM</td>
<td>Network address management</td>
</tr>
<tr>
<td>ND</td>
<td>NLDLM</td>
<td>Network logical data manager</td>
</tr>
</tbody>
</table>

Using the VTAM internal trace
Table 7. VTAM component IDs (continued)

<table>
<thead>
<tr>
<th>ID</th>
<th>Acronym</th>
<th>Component name</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO</td>
<td>NOS</td>
<td>Network operator services</td>
</tr>
<tr>
<td>NR</td>
<td>NRM</td>
<td>Network resource management</td>
</tr>
<tr>
<td>NS</td>
<td>LNS</td>
<td>LU network services</td>
</tr>
<tr>
<td>OC</td>
<td>O/C</td>
<td>Open/Close ACB</td>
</tr>
<tr>
<td>OI</td>
<td>OSI-RPI</td>
<td>VTAM OSI RPI</td>
</tr>
<tr>
<td>OR</td>
<td>SMS</td>
<td>Storage management services</td>
</tr>
<tr>
<td>PS</td>
<td>PS</td>
<td>Presentation services</td>
</tr>
<tr>
<td>PU</td>
<td>PUNS</td>
<td>Physical unit services</td>
</tr>
<tr>
<td>PV</td>
<td>PVI</td>
<td>Primitive VTAM interface</td>
</tr>
<tr>
<td>RA</td>
<td>TRACE</td>
<td>Trace services</td>
</tr>
<tr>
<td>RM</td>
<td>LRM</td>
<td>LU resources manager</td>
</tr>
<tr>
<td>RV</td>
<td>RVM</td>
<td>MNPS recovery manager</td>
</tr>
<tr>
<td>SA</td>
<td>SYS-ATTC</td>
<td>System attach</td>
</tr>
<tr>
<td>SC</td>
<td>SSC</td>
<td>Session services CP-CP</td>
</tr>
<tr>
<td>SD</td>
<td>SYSDEF</td>
<td>System definition</td>
</tr>
<tr>
<td>SL</td>
<td>SSL</td>
<td>Session services LU-LU</td>
</tr>
<tr>
<td>SR</td>
<td>SMC-R</td>
<td>Shared Memory Communication over RDMA</td>
</tr>
<tr>
<td>SS</td>
<td>SSCPSS</td>
<td>Session services</td>
</tr>
<tr>
<td>TA</td>
<td>TACMIP</td>
<td>Topology Agent</td>
</tr>
<tr>
<td>TB</td>
<td>TSC-BSC</td>
<td>TSC Binary Synchronous Communication</td>
</tr>
<tr>
<td>TC</td>
<td>TCP-DLC</td>
<td>Data Link Control for TCP</td>
</tr>
<tr>
<td>TD</td>
<td>TSC-DLC</td>
<td>TSC-DLC</td>
</tr>
<tr>
<td>TL</td>
<td>TSC-LAN</td>
<td>Token Ring Local Area Network</td>
</tr>
<tr>
<td>TO</td>
<td>TSO/VTAM</td>
<td>TSO/VTAM</td>
</tr>
<tr>
<td>TR</td>
<td>TRS</td>
<td>Topology routing services</td>
</tr>
<tr>
<td>TS</td>
<td>TSC</td>
<td>Transmission subsystem</td>
</tr>
<tr>
<td>TX</td>
<td>TSX-X25</td>
<td>X.25</td>
</tr>
<tr>
<td>US</td>
<td>USS</td>
<td>Unformatted session services</td>
</tr>
<tr>
<td>VI</td>
<td>VIO</td>
<td>VSAM I/O</td>
</tr>
<tr>
<td>VX</td>
<td>VSAMMACS</td>
<td>VSAM user macroinstructions</td>
</tr>
<tr>
<td>XP</td>
<td>TPS</td>
<td>Transaction program services</td>
</tr>
<tr>
<td>62</td>
<td>APPC</td>
<td>APPC LU6.2</td>
</tr>
</tbody>
</table>
Appendix A. FFST operations

First Failure Support Technology (FFST) is an IBM licensed program that provides immediate notification and first failure data capture (FFDC) for software events. IBM’s SystemView strategy incorporates FFST as part of its problem management discipline. This book applies to the following FFST product running in the specified environments:

- FFST for Multiple Virtual Storage (FFST/MVS) in a Multiple Virtual Storage/Enterprise Systems Architecture (MVS/ESA) environment

FFST/MVS also incorporates its own technology by including software probes in its own code. FFST/MVS processes these software probes using its internal FFDC function. When one of these internal software probes is triggered, FFST/MVS issues a symptom string that describes the event. This symptom string appears in the dump and in the generic alert. The dump is a member of a partitioned data set, which is specified in the FFST/MVS startup procedure. The FFDC messages describe the member name and the data set that contain the dump; you can format the dump using the FFST dump formatting program, EPWDMPPFM.

This appendix explains how to perform the following tasks:

- Use commands to control FFST operation
- Use the different types of FFST output to identify and analyze software events

FFST overview

Advances in hardware manufacturing and technology have enabled the computer industry to vastly improve the reliability of circuitry and reduce hardware cost. Less expensive hardware has stimulated extensive use of circuitry to detect failures or deteriorating circuit performance and 'call home', pointing out what component should be replaced. The results are significant reduction in repair time and even more significant reductions in service skill and labor.

As hardware reliability improves, software problems account for a greater portion of system and application interruptions because software has not enjoyed the same degree of advancement in technology as hardware. Although great strides have been made in quality, often measured as errors per 1000 lines of code, the amount of code and system complexity have increased to make this improvement barely visible. Currently, the industry offers programs based on several different failure capture techniques requiring a variety of personnel skills and system resources to recognize and resolve failures across a system.

There are at least five major problems that exist in the software service arena today:

1. Detecting problems as early as possible before the environment changes
2. Capturing the correct data to debug the software problem- the first time the error occurs
3. Capturing only the data required to debug the error (i.e., minimize the need for full address space dumps)
4. Immediate notification of the error
5. Uniquely identifying the error in order to determine if it is a condition that was already detected and reported to the support organization.

Note: Each place throughout this document, the name FFST implies FFST for MVS except where it is specifically mentioned that it is FFST for MVS or FFST for VM.

FFST provides the following services for IBM products:

**customized dump**

Promotes the collection of only the data required to debug a software problem
symptom string
Provides a unique problem 'label' that can be used to quickly determine if a software problem has already been detected. The symptom string is contained in each output in this list.

symptom record
Error log entry built to IBM's Symptom Record Architecture (SRA) standard and placed in LOGREC.

messages
Indication on the operator console that a problem has occurred and FFST was called to collect the data and report the problem.

network notification
Indication through an System Network Architecture (SNA) Generic Alert that a problem has occurred and FFST was called to collect the data and report the problem. Included in the Generic Alert is key information which includes the machine on which the problem occurred and the name of the dump data set if a dump was requested by the detecting product.

It should be noted that there are situations that will continue to require full address space dumps. For certain types of problems it is very difficult for a programmer to determine what data may be required to diagnose a failure. For these problems, a capture of the complete environment will be required.

IBM programmers continue to improve their defensive programming techniques within their software in order to assure the instances of needing full address space dumps to diagnose a failure will be kept to a minimum.

FFST components

FFST is composed of four components:

software probe
Call statements placed in IBM program products which are used to access FFST services. Each probe statement has a unique identifier, up to eight characters in length. This identifier can be used in controlling the operation of FFST through the MODIFY commands specified in “FFST MODIFY Command Overview” on page 2-3. The first three characters of the probe identifier indicate which product was issuing the call to FFST. For example, if FFST is called with a probe identifier that starts with 'ISTxxxxx', this call was from the VTAM program product.

In order to protect the operating system from the excessive use of resources by FFST, FFST will automatically disable any probe statement that exceeds a using-product specified criteria. If this occurs, FFST will ignore any calls for services by that probe statement with a matching symptom string. After the problem is fixed which caused a probe statement to be executed at a high rate, FFST can be reset to start accepting calls by the probe statement through the FFST MODIFY command set. See “FFST MODIFY command overview” on page 32.

configuration table
A single CSECT module built and shipped with the program product that chooses to use FFST services. It contains information that identifies the using product. Its name is passed to FFST when that program product initializes with FFST. (Each product that requires FFST services must issue an FFST initialization call to inform FFST that it may be requiring FFST services.)

Data Structure Table (DST)
A single CSECT module built and shipped with the program product that chooses to use FFST service. It contains static information which FFST uses to determine what data is to be captured for each software probe used by that product and information used to build the Generic Alert. It minimizes the amount of static data that must be passed in the software probe call. A single product may choose to use more than one DST. The name of the DST which contains the information for a specific software probe is specified in the software probe call.

Problem source identifier (PSI)
The main component of FFST which collects the data specified on the probe statement and generates the diagnostic data outputs specified by the calling product. It utilizes the FFST configuration table to determine the identity of the caller and uses the data structure table (DST) to determine what diagnostic data is to be captured (i.e., customized dump, Generic Alert, console messages, error log...
entry in LOGREC). The processing performed by the PSI is controlled by the FFST MODIFY commands. See “FFST MODIFY command overview” on page 32.

FFST outputs

FFST provides the program product caller with the option of choosing from four diagnostic outputs. Each of these outputs is explained in the sections that follow. Each output is fully controllable through the FFST MODIFY commands which are described in “FFST MODIFY command overview” on page 32.

Customized dump (MVS)

When a software probe is executed and the caller chooses to request a dump, FFST will dynamically allocate a data set and generate an unformatted dump. The name of the data set will be as follows:

user_name.system_name.applid.DMPxxxxx

where:

• user_name is the high level qualifier selected by the customer and entered through the FFST MODIFY command facility. See “FFST MODIFY command overview” on page 32 for more details. The default value FFST uses is ‘FFST’.

  Note: If ‘FFST’ is not an acceptable value, then this name must be changed in order to avoid a dump creation failure due to a data set security violation.

• system_name is the name of the MVS system, taken from SYS1.PARMLIB

  Note: If the name begins with a numeric, an ‘S’ is appended to the beginning of the system name.

• applid is a short name of the using product (e.g., VTAM, NETVIEW) which the using product specifies.

• xxxxx is a sequence number which makes the dump data set name unique

In order to read the dump, the FFST dump formatter EPWDMPFM has to be used. See “Formatting a customized dump for FFST/MVS” on page 54 which describes the use of this tool.

Error log entry

When a software probe is executed and the caller chooses to request an error log entry, FFST generates an error log entry and place sit in the LOGREC data set. This entry, built using the Symptom Record Architecture format, can be formatted and printed using the Environmental Record Editing and Printing (EREP) program. See “Using the symptom string” on page 66.

This entry contains a summation of the problem detected by the caller and includes key information such as the primary symptom string which uniquely identifies the problem and the name of the dump data set (MVS only) into which the dump was placed, if a dump was requested.

Messages

When a software probe is executed and the caller chooses to present problem information through the operator console, FFST generates a series of messages which include the primary symptom string for the problem, the name of the dump data set, if a dump was requested, and the volume serial number on which the dump data set resides. See “Using the console message” on page 72 for details of these messages.

Generic alert

When a software probe is executed and the caller chooses to notify a network operator of the problem, FFST generates an SNA Generic Alert summarizing the problem and passes the alert to the IBM NetView program product for processing. The alert contains information key to the problem and includes the identity of the machine on which the problem occurred, the name of the program product that detected the problem, the date and time of the problem, the name of the dump data set into which the dump was placed, and the primary symptom string. See “Using the generic alert” on page 73 for details of the alert and how it is processed by the NetView program product.
**Probe message log**

In addition to the diagnostic outputs described above, FFST generates a probe execution entry and places it in the FFST probe message log each time a probe is executed. This log can be used to keep a history of all the products that requested FFST services. Through the FFST start up procedure, two logs may be specified: primary and secondary. When the primary is filled, FFST must be switched over to use the secondary log. See “Using the probe message log entry” on page 72 for details of the probe message log and how to use its contents.

**FFST usage**

When a product calls FFST for data capture services, FFST indicates this event through the following message:

```
EPW0401I procname: EVENT DETECTION INVOKED BY applname
```

where *procname* is the name of the procedure that started FFST and *applname* is the name of the calling program product (e.g., VTA M, NetView). This message is a clear indication that FFST was called by a product and there could be a software problem. FFST messages to follow will provide information about the error. Using the information in the following chapters, the FFST outputs can be collected and used when working with the IBM Support Center to determine the cause of the problem. Whenever the FFST MODIFY command facility is being utilized to control FFST support for a specific application, the *applname* value in the EPW0401I message is the value to be used on the APPLID operand.

In addition to program products that use FFST services through software probes, FFST for MVS provides a system monitor capability that watches the programming environment for problems that may occur where FFST is not called to process the data from the situation. This function is provided by watching for dump requests from authorized program products through the MVS Post Dump Exit (IEAVTSEL) facility (PN40734) and watching for dump requests from CICS® transactions through the CICS dump user exit (PN45724). This monitoring capability is known as the FFST Transition Code function and is described in detail in Appendix B, “FFST for MVS transition code function” on page 152.

**FFDC function**

FFST utilizes its own technology to detect internal problems. This function is called FFDC. EPW00xxI messages are issued by FFST which pertain to the FFDC function. When an FFDC probe trips, messages are issued containing the symptom string and a dump is also taken. This dump is a member of a pre-allocated partitioned dump data set, which can also be formatted using the FFST dump formatter.

**Controlling FFST operation**

FFST provides START, MODIFY, and STOP commands that let you control its operation. You can use these commands two different ways

- Through the FFST operator's console
- Through a startup command list that establishes the FFST operating environment at FFST initialization

**UPPERCASE BOLDFACE**

- Indicates actual command names, keywords, or operands. These values must be spelled as shown but can be typed in either uppercase or lowercase.

**lowercase italic**

- Indicates variables that show the type of information required, rather than a specific value. When you type the command, substitute an actual value (usually the name of a resource) for the lowercase italic characters.

**Underscore**

- Indicates the default value that FFST uses if you do not specify another value.
Brackets [ ]
Indicate an optional specification. Any commas, equal signs, parentheses, or other symbols between the brackets are also optional. Do not include the brackets when typing the command.

Vertical bar | Separates the possible options for a single keyword. If a group of options separated by vertical bars appears between brackets, you do not have to choose any of the options in that group; FFST uses the default value.

Equal sign, comma, parentheses, and asterisk = , ( )* Enter these symbols as shown, unless they appear between brackets. When they appear between brackets, you do not have to include them unless you choose to include the associated optional operand.

The sections in this chapter provide the following information:
• General information about how to use FFST commands, including command syntax and definitions of command parameters
• Instructions for using FFST commands from a command console, including specific examples of FFST commands
• Instructions for using FFST commands in a startup command list

Understanding FFST commands
This section provides the following information for the FFST START, MODIFY, and STOP commands:
• The command syntax
• A list of available operands
• An explanation of each of the command parameters

FFST START command overview
When used to control FFST operation, the START command has the following format:

<table>
<thead>
<tr>
<th>Platform</th>
<th>Command Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVS</td>
<td>START|S procname,operands</td>
</tr>
</tbody>
</table>

The FFST START command has the following operands:

[LANG=lang|ENU]
[FMODE=NORMAL|DEBUG]
[PAGE=pagenum|200]
[start=stmemnam|START00]

Note: The EPWFFST sample start-up procedure can be changed to include an additional parameter called START. The value for this parameter should be included as the fourth parameter on the EXEC statement in the JCL.

If you include more than one operand, you must separate the operands with commas. For example:
operand1=option,operand2=option

The following list explains each of the FFST START command parameters:

START|S
The command used to initialize FFST. For FFST/MVS, the FFST subsystem is initialized in its own address space. Once FFST is initialized, it can process triggered software probes which can be controlled by a set of MODIFY commands.

For the first start after initial program load (IPL), the procedure reads the checkpoint data set if the following conditions are true:
The FFSTCKPT DD name appears in the startup job control language (JCL).

The checkpoint data set restores the environment from the last time FFST was active. In addition, whenever FFST is started, the system reads and processes a startup command list, if one exists. For more information, see “Using a startup command list” on page 53.

**proname**

The procedure name for the operation. (This parameter applies to MVS only.) The recommended value is EPWFFST.FFST. If you use EPWFFST.FFST as the procedure name in the START command, you can use FFST as the procedure name in the MODIFY commands. For more information about the procedure name used in MODIFY commands, see “FFST MODIFY command overview” on page 32.

You can find sample JCL for this procedure in the install library on the install tape, along with the other install jobs.

**LANG=**

The keyword used to specify the National Language Support (NLS) abbreviation for the language in which the FFST messages are to appear. The default (and the only value currently allowed for the LANG parameter) is ENU.

**FMODE=**

The keyword used to indicate the type of messages you want FFST to issue. This keyword can have the following values:

- **NORMAL**
  - FFST issues only its standard messages. NORMAL is the default value.
- **DEBUG**
  - FFST issues flow messages in addition to its standard messages. Use this option only when necessary.

**PAGE=**

A keyword, used only for FFST/MVS, that indicates the number of fixed pages of storage to be preallocated for any software probe triggered while running disabled. A value that is too low can result in a software probe not having enough pages available to take an FFST dump. A value that is too high can adversely affect system performance. The default value for the PAGE parameter is 200.

**START=**

A keyword that indicates the FFSTPARM start list member name to be used. The default value is START00.

### FFST MODIFY command overview

When used to control FFST operation, the MODIFY command has the following format:

<table>
<thead>
<tr>
<th>Platform</th>
<th>Command Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVS</td>
<td>MODIFY</td>
</tr>
</tbody>
</table>

The FFST MODIFY command has the following operands:

- \[[ACTION|A=\] DISABLE|DIS| ENABLE|EN| CLEAR|CL| DISPLAY|D|\]
- \[[CHANGE|CH| RESET|R| HALT|\] [QUICK]\]
- \[[APPLID|AP=applname|FFST]\]
- \[[VENDOR|V=vendorname|IBM|IBM CORPORATION]\]
The following MODIFY operands are available for MVS only:

[DUMPQUAL|DQ=dumpqualifier]

[DUMPVOL|DV=dump_volume_serial_number]

The following rules apply when you use these operands:

- If you include more than one operand, you must separate the operands with commas.
- If you include more than one option on an operand, you must separate the options with commas and enclose them in parentheses.

For example:

operand1=option,operand2=(option1,option2),operand3

The following list explains each of the FFST MODIFY command parameters:

**MODIFY**

The command used to communicate with the FFST/MVS program.

**procname**

The procedure name for the operation. (This parameter applies to MVS only.) The recommended value is FFST.

**ACTION**

The keyword used to indicate the action you want to perform.

This keyword can have the following values:

- **DISABLE** (DIS): The parameter used to disable output options, probe statements, and probe message logging.
- **ENABLE** (EN): The parameter used to enable output options, probe statements, and probe message logging.
- **CLEAR** (CL): The parameter used to clear any knowledge of probe statements and to clear a message log which FFST maintains.
- **DISPLAY** (D): The parameter used to display the operating status of FFST, an application known to FFST, or probe statements in an application known to FFST.

**Note:** DISPLAY is the default value for ACTION.

- **CHANGE** (CH): The parameter used to change the destination of FFST dumps and generic alerts.
- **RESET** (R): The parameter used to reset the destination of FFST dumps and alerts to the default value.
- **HALT**

The parameter used to stop the execution of the FFST program once all currently running and pending processes are complete.
QUICK
The keyword used to indicate that FFST will stop running without waiting for all applications to
terminate their interfaces to FFST. This keyword is allowed only with an ACTION=HALT command.

APPLID|AP=
The keyword used to identify the application affected by the command. This keyword can have either
of the following values:

  applname
  The name of the application you want the MODIFY command to affect.

  FFST
  The MODIFY command affects the FFST program.

VENDOR|V=
The keyword used to indicate the application's owner. For IBM products, use IBM or IBM
CORPORATION. For vendor products, use the documented vendor name.

PROBEID|P=
The keyword used to identify the probe statement to be affected by the command. You can use any of
the following values:

  xxxyyynn
  Specifies a single probe statement, where:
  • xxx is the probe identifier prefix (the first 3 characters of the probe identifier).
  • yyy is the source file identifier (the third, fourth, and fifth characters of the probe identifier).
  • nn is the number that uniquely identifies the individual probe statement.

  xxxyyyn
  Specifies a group of probe statements that share the same probe identifier prefix and source file
  identifier, where:
  • xxx is the probe identifier prefix.
  • yyy is the source file identifier.

  xxx*
  Specifies a group of probe statements that share the same probe identifier prefix, where xxx is the
  probe identifier prefix.

  Note: If you type a command that includes a probe identifier prefix that is unknown to FFST, you
  must include the APPLID parameter.

OPTIONS|OPT=
The keyword used to identify the FFST output options affected by the command.
You can use any of the following parameters with the OPTIONS keyword:

  DUMP
  The MODIFY command affects FFST dump support.

  SYMRC
  The MODIFY command affects symptom record support.

  GENAL
  The MODIFY command affects generic alert support.

  SYMST
  The MODIFY command affects symptom string message support.

  SUPDP
  The MODIFY command affects duplicate dump suppression.

  ALL
  The MODIFY command affects all the preceding output options.
**DEFINE**

The keyword used to identify a new application. If you use the APPLID keyword to specify an application that is not known to FFST, include the DEFINE parameter to indicate that it is new.

**LOGID**

The keyword used to indicate that the command is to affect the probe message log function. When you use LOGID you have three options:

- You can type LOGID without specifying a value if you also specify ACTION=ENABLE or ACTION=DISABLE. If you do not specify a value, the FFST probe message logging function will be enabled or disabled.
- You can type LOGID=n or LOGID=FFSTLOGn, where n is a numeric value between 1 and 9. Using this notation means that the log data set specified on DD name FFSTLOGn will be enabled, disabled, or cleared. When a data set is cleared, the clear is scheduled, which means that FFST will not clear the data set at the time of the command, but will consider the data set available the next time it becomes the current data set, and will clear it then.

**Note:** Up to 9 DD statements may be specified for probe message logs in the FFST start-up JCL. However, FFST will only use as many as are sequentially specified, starting at 1. In other words, if FFSTLOG1, FFSTLOG2 and FFSTLOG4 are in the JCL, only FFSTLOG1 and FFSTLOG2 will be used, and are the only ones that you may specify a modify command for.

- You can type LOGID=REUSE if you also specify ACTION=ENABLE or ACTION=DISABLE. This indicates that the REUSE function is to be enabled or disabled. REUSE means that when FFST switches to the next log data set, FFST will reuse that data set (clearing it automatically first), even if it contains log information. If REUSE is disabled, FFST will not reuse a data set that contains any data, unless a CLEAR command has been issued for it.

**Note:** FFST message log switching initially starts with FFSTLOG1 data set, and continues through the last FFSTLOGx DD in the FFST JCL. When that is full, FFST starts over with FFSTLOG1. Also, all log information is stored in the FFST checkpoint data set, so the environment will be restored after recycling of FFST, even after an IPL of the operating system.

**ALRCVID**

The keyword used to identify the NetView alert receiver that is to receive FFST generic alerts. When you use ALRCVID, you have two options:

- If you are using ACTION=RESET, you can type ALRCVID without specifying a value. FFST assumes you want to reset the value to the default of "NETVALRT".
- You can type ALRCVID=alertreceiver where alertreceiver is one of the following values:
  - If you are using ACTION=CHANGE, alertreceiver is the alert receiver you want to use.
  - If you are using ACTION=RESET, alertreceiver is the name of the current alert receiver (the receiver you want reset to the default value).

**DUMPQUAL**

The keyword used to indicate the high-level data set qualifier for all FFST dump data sets. (This parameter applies to MVS only.) FFST dump data set names have the format dumpqual.sysname.prodqual.DMPxxxxx. At FFST initialization, the default dumpqual value is “FFST.”

When you use DUMPQUAL, you have two options:

- If you are using ACTION=RESET, you can type DUMPQUAL without specifying a value. FFST assumes you want to reset the value to the default.
- If you are using ACTION=RESET, dumpqualifier is the name of the current dump data set qualifier (the qualifier you want reset to the default value).

**DUMPVOL**

The keyword used to identify the volume serial number on which FFST dump data sets are allocated. (This parameter applies to MVS only.) When you use DUMPVOL, you have two options:
• If you are using ACTION=RESET, you can type DUMPVOL without specifying a value. FFST assumes you want to reset the value to the default.

• You can type DUMPVOL=dump_volume_serial_number where dump_volume_serial_number is one of the following values:
  – If you are using ACTION=CHANGE, dump_volume_serial_number is the volume serial number you want to use.
  – If you are using ACTION=RESET, dump_volume_serial_number is the name of the current volume serial number (the number you want reset to the default value).

**FFST STOP command overview**

Although the preferred way to stop FFST is to use a MODIFY command with the ACTION=HALT parameter, you can also use the STOP command. The STOP command has the following format:

<table>
<thead>
<tr>
<th>Platform</th>
<th>Command Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVS</td>
<td>STOP</td>
</tr>
</tbody>
</table>

The following list explains each of the STOP command parameters:

**STOP**

The command used to stop execution of the FFST program.

**proname**

The procedure name for the operation. (This parameter applies to MVS only.) The recommended value is FFST.

**Using FFST commands from a console**

You can control FFST operation by issuing commands from an FFST console. The following sections explain how to use these console commands.

**Note:** In this chapter, the phrase "an application known to FFST" refers to an application that has executed a probe statement, or an application specified in an FFST MODIFY command.

**Starting FFST**

When used to start FFST, the START command has the following operands:

```
[LANG=lang | ENU]  
[FMODE=NORMAL | DEBUG]  
[PAGE=pagenum | 200]  
```

For more information about these operands, see “FFST START command overview” on page 31.

**Example 1:**

To start FFST in debug mode, type one of the following commands:

```
MVS
START EPWFFST.FFST,FMODE=DEBUG
```

**Example 2:**

To start FFST/MVS with 100 preallocated pages of fixed storage, type the following command:

```
START EPWFFST.FFST,PAGE=100
```

**Disabling FFST functions**

The MODIFY ACTION=DISABLE command lets you perform the following tasks:

• Disable a probe statement or group of probe statements

• Disable FFST output options
• Disable FFST probe message logging

When you use the MODIFY ACTION=DISABLE command to disable a probe statement or an FFST output option, the APPLID or PROBEID keyword indicates the command's level of control. There are 5 levels of control:

• FFST (the highest level). FFST will not process any probes.
• A specific application. FFST will not process any probes from a specified application.
• A group of probe statements that have the same probe identifier prefix. (The probe identifier prefix is the first 3 characters of the probe identifier.)
• A group of probe statements that have the same probe identifier prefix and the same source file identifier. (The source file identifier is the third, fourth, and fifth characters of the probe identifier.)
• A single probe statement (the lowest level).

If you disable probe statements or output options at one of these levels, the command also affects all levels below that level. In addition, if you want to use a MODIFY ACTION=ENABLE command to enable the disabled probe statements or output options, you must issue that command at the same level. For example, if you disable an output option at the application level, you cannot enable it for a single probe statement in that application by issuing a MODIFY ACTION=ENABLE command at the probe statement level. Nor can you enable the output option for all the probe statements in the application by issuing a MODIFY ACTION=ENABLE command at the FFST level. You must issue the MODIFY ACTION=ENABLE command at the application level. Knowledge of the probe statement and the number of times the probe has tripped is not cleared as with the MODIFY ACTION=CLEAR command.

Disabling probe statements for FFST

When you disable probe statements at the FFST level, FFST does not process any triggered software probes.

When used to disable probe statements at the FFST level, the MODIFY command has the following operands:

| ACTION | A=DISABLE|DIS |
| APPLID | AP=FFST |

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to disable probe statements at the FFST level, type one of the following commands:

**MVS**

MODIFY FFST, ACTION=DISABLE, APPLID=FFST

Disabling probe statements for an application

When you disable probe statements at the application level, FFST does not process any software probes triggered in that application.

When used to disable probe statements at the application level, the MODIFY command has the following operands:

| ACTION | A=DISABLE|DIS |
| APPLID | AP=applname |
| VENDOR | V=vendorname | IBM | IBM CORPORATION |

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to disable all the probe statements in IBMs VTAM application, type one of the following commands:

**MVS**

MODIFY FFST, ACTION=DISABLE, APPLID=VTAM, VENDOR=IBM
Disabling probe statements by probe identifier

When you disable a probe statement or probe statements by probe identifier, FFST does not process any triggered software probes with a probe identifier you specified.

When used to disable probe statements by probe identifier, the MODIFY command has the following operands:

| ACTION | A=DISABLE|DIS |
| PROBEID | P=xxxyyynn|xxxyyyn*|xxx* |
| VENDOR | V=vendorname|IBM|IBM CORPORATION |
| APPLID | AP=applname |
| DEFINE | DEF |

For more information about these operands, see “FFST MODIFY command overview” on page 32.

Example 1:
To disable a single probe statement with the probe identifier ISTTSC07, type one of the following commands:

```mvs
MVS MODIFY FFST,ACTION=DISABLE,PROBEID=ISTTSC07,VENDOR=IBM
```

Example 2:
To disable all the probe statements that have a probe identifier prefix of IST and a source file identifier of TSC, type one of the following commands:

```mvs
MVS MODIFY FFST,ACTION=DISABLE,PROBEID=ISTTSC*,VENDOR=IBM
```

Example 3:
To disable all the probe statements that have a probe identifier prefix of IST, type one of the following commands:

```mvs
MVS MODIFY FFST,ACTION=DISABLE,PROBEID=IST*,VENDOR=IBM
```

Disabling output options for FFST

When you disable an output option at the FFST level, FFST does not generate the output for any triggered probe statements.

When used to disable output options at the FFST level, the MODIFY command has the following operands:

| ACTION | A=DISABLE|DIS |
| APPLID | AP=FFST |
| OPTIONS | OPT= |
| DUMP |
| SYMRC |
| GENAL |
| SYMST |
| SUPDP |
| ALL |

For more information about these operands, see “FFST MODIFY command overview” on page 32.

Example 1:
To disable the generic alert output option at the FFST level, type one of the following commands:

```mvs
MVS MODIFY FFST,ACTION=DISABLE,APPLID=FFST,OPTIONS=GENAL
```

Example 2:
To disable the symptom record and duplicate dump suppression output options at the FFST level, type one of the following commands:
Example 3:
To disable all the output options at the FFST level, type one of the following commands:

```
MVS
MODIFY FFST, ACTION=DISABLE, APPLID=FFST, OPTIONS=(SYMRC, SUPDP)
```

**Disabling output options for an application**

When you disable an output option at the application level, FFST does not generate the output for any probe statements triggered in the specified application.

When used to disable FFST output options at the application level, the MODIFY command has the following operands:

<table>
<thead>
<tr>
<th>ACTION</th>
<th>A=DISABLE</th>
<th>DIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLID</td>
<td>AP=applname</td>
<td></td>
</tr>
<tr>
<td>VENDOR</td>
<td>V=vendorname</td>
<td>IBM</td>
</tr>
<tr>
<td>OPTIONS</td>
<td>OPT= (DUMP SYMRC GENAL SYMST SUPDP)</td>
<td>ALL</td>
</tr>
<tr>
<td>[DEFINE]</td>
<td>DEF</td>
<td></td>
</tr>
</tbody>
</table>

For more information about these operands, see “FFST MODIFY command overview” on page 32.

**Example 1:**

To disable the FFST generic alert output option for IBM's VTAM application, type one of the following commands:

```
MVS
MODIFY FFST, ACTION=DISABLE, APPLID=VTAM, VENDOR=IBM, OPTIONS=GENAL
```

**Example 2:**

To disable the FFST symptom record and duplicate dump suppression output options for IBM's VTAM application, type one of the following commands:

```
MVS
MODIFY FFST, ACTION=DISABLE, APPLID=VTAM, VENDOR=IBM, OPTIONS=(SYMRC, SUPDP)
```

**Example 3:**

To disable all the FFST output options for IBM's VTAM application, type one of the following commands:

```
MVS
MODIFY FFST, ACTION=DISABLE, APPLID=VTAM, VENDOR=IBM, OPTIONS=ALL
```

**Disabling output options by probe identifier**

When you disable an output option by probe identifier, FFST does not generate the output for any triggered probe statement with a probe identifier you specified.

When used to disable FFST output options by probe identifier, the MODIFY command has the following operands:

<table>
<thead>
<tr>
<th>ACTION</th>
<th>A=DISABLE</th>
<th>DIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROBEID</td>
<td>P=xxxxyyy</td>
<td>yyyy*</td>
</tr>
<tr>
<td>OPTIONS</td>
<td>OPT= (DUMP SYMRC GENAL SYMST SUPDP)</td>
<td>ALL</td>
</tr>
</tbody>
</table>

---

**FFST operations**
For more information about these operands, see “FFST MODIFY command overview” on page 32.

Example 1:

To disable the FFST generic alert output option for a single probe statement with the probe identifier ISTTSC07, type one of the following commands:

```mvs
MVS
MODIFY FFST, ACTION=DISABLE, PROBEID=ISTTSC07, OPTIONS=GENAL, VENDOR=IBM
```

Example 2:

To disable the FFST symptom record and duplicate dump suppression output options for all the probe statements that have a probe identifier prefix of IST and a source file identifier of TSC, type one of the following commands:

```mvs
MVS
MODIFY FFST, ACTION=DISABLE, PROBEID=ISTTSC*, OPTIONS=(SYMRC, SUPDP), VENDOR=IBM
```

Example 3:

To disable all the FFST output options for probe statements that have a probe identifier prefix of IST, type one of the following commands:

```mvs
MVS
MODIFY FFST, ACTION=DISABLE, PROBEID=IST*, OPTIONS=ALL, VENDOR=IBM
```

Disabling FFST probe message logging

When used to disable the FFST probe message logging function, the MODIFY ACTION=DISABLE command has the following format:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Operands</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODIFY</td>
<td>F procname , ACTION=A=D,LOGID=n,REUSE</td>
</tr>
</tbody>
</table>

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to disable message logging for DD name FFSTLOG3, type the following commands:

```mvs
MVS
MODIFY FFST, ACTION=DISABLE, LOGID=FFSTLOG3
```

To disable the REUSE function, type the following commands:

```mvs
MVS
MODIFY FFST, ACTION=DISABLE, LOGID=REUSE
```

To disable the FFST's message logging function, type the following commands:

```mvs
MVS
MODIFY FFST, ACTION=DISABLE, LOGID
```

Enabling FFST functions

The MODIFY ACTION=ENABLE command lets you perform the following tasks:

- Enable a probe statement or group of probe statements
- Enable FFST output options
- Enable FFST probe message logging
Note: Probe statements, probe message logging, and all the FFST output options are enabled by default. You only need to enable them if they have been disabled using the MODIFY ACTION=DISABLE command.

When you use the MODIFY ACTION=ENABLE command to enable a probe statement or an FFST output option, the APPLID or PROBEID keyword indicates the command’s level of control. There are 5 levels of control:

- FFST (the highest level).
- A specific application.
- A group of probe statements that have the same probe identifier prefix. (The probe identifier prefix is the first 3 characters of the probe identifier.)
- A group of probe statements that have the same probe identifier prefix and the same source file identifier. (The source file identifier is the third, fourth, and fifth characters of the probe identifier.)
- A single probe statement (the lowest level).

When you issue a MODIFY ACTION=ENABLE command, you must issue it at the same level at which you issued the corresponding MODIFY ACTION=DISABLE command.

For example, if you disable probe statements at the application level, you cannot enable a single probe statement in that application by issuing a MODIFY ACTION=ENABLE command at the probe statement level. Nor can you enable all the probe statements in the application by issuing a MODIFY ACTION=ENABLE command at the FFST level. You must issue the MODIFY ACTION=ENABLE command at the application level.

Enabling probe statements for FFST

When you enable probe statements at the FFST level, FFST processes any triggered software probes that were previously disabled at the FFST level, unless they were also disabled at the application level or by probe identifier.

When used to enable probe statements at the FFST level, the MODIFY command has the following operands:

```
ACTION|A=ENABLE|EN
APPLID|AP=FFST
```

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to enable probe statements at the FFST level, type one of the following commands:

**MVS**

```
MODIFY FFST,ACTION=ENABLE,APPLID=FFST
```

Enabling probe statements for an application

When you enable probe statements at the application level, FFST processes any triggered software probes in the application that were previously disabled at the application level, unless they were also disabled at the FFST level or by probe identifier.

When used to enable all the probe statements at the application level, the MODIFY ACTION=ENABLE command has the following operands:

```
ACTION|A=ENABLE|EN
APPLID|AP=applname
VENDOR|V=vendorname|IBM|IBM CORPORATION
```

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to enable all the probe statements in IBM’s VTAM application, type one of the following commands:

**MVS**

```
MODIFY FFST,ACTION=ENABLE,APPLID=VTAM,VENDOR=IBM
```
Enabling probe statements by probe identifier

When you enable probe statements by probe identifier, FFST processes any triggered software probes previously disabled at the same probe identifier level, unless they were also disabled at one of the following levels:

- FFST level
- Application level
- A higher or lower probe statement level

When used to enable probe statements by probe identifier, the MODIFY command has the following operands:

- **ACTION|A=ENABLE|EN**
- **PROBEID|P=xxxyyyynn|xxxyyy*|xxx***
- **VENDOR|V=vendorname|IBM|IBM CORPORATION**
- **APPLID|AP=applname**

For more information about these operands, see “FFST MODIFY command overview” on page 32.

**Example 1:**

To enable a single probe statement with the probe identifier ISTTSC07, type one of the following commands:

```
MVS
MODIFY FFST,ACTION=ENABLE,PROBEID=ISTTSC07,VENDOR=IBM
```

**Example 2:**

To enable all the probe statements that have a probe identifier prefix of IST and a source file identifier of TSC, type one of the following commands:

```
MVS
MODIFY FFST,ACTION=ENABLE,PROBEID=ISTTSC*,VENDOR=IBM
```

**Example 3:**

To enable all the probe statements that have a probe identifier prefix of IST, type one of the following commands:

```
MVS
MODIFY FFST,ACTION=ENABLE,PROBEID=IST*,VENDOR=IBM
```

Enabling output options for FFST

When you enable an output option at the FFST level, FFST generates the output for any software probe for which the output was previously disabled at the FFST level, unless it was also disabled at the application level, by probe identifier, or in the probe statement.

When used to enable output options at the FFST level, the MODIFY command has the following operands:

- **ACTION|A=ENABLE|EN**
- **APPLID|AP=FFST**
- **OPTIONS|OPT= (DUMP SYMRC GENAL SUPDP ALL)**

For more information about these operands, see “FFST MODIFY command overview” on page 32.

**Example 1:**

To enable the generic alert output option at the FFST level, type one of the following commands:

```
MVS
MODIFY FFST,ACTION=ENABLE,APPLID=FFST,OPTIONS=GENAL
```
Example 2:
To enable the symptom record and duplicate dump suppression output options at the FFST level, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,APPLID=FFST,OPTIONS=(SYMRC,SUPDP)

Example 3:
To enable all the output options at the FFST level, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,APPLID=FFST,OPTIONS=ALL

Enabling output options for an application

When you enable an output option at the application level, FFST generates the output for any triggered software probes for which the output was previously disabled at the application level, unless the output was also disabled at the FFST level, by probe identifier, or in the probe statement.

When used to enable FFST output options at the application level, the MODIFY command has the following operands:

- **ACTION** | \( A = \) ENABLE | EN
- **APPLID** | \( AP = \) applname
- **VENDOR** | \( V = \) vendorname | IBM | IBM CORPORATION
- **OPTIONS** | \( OPT = \) (DUMP | SYMRC | GENAL | SYMST | SUPDP | ALL)

For more information about these operands, see “FFST MODIFY command overview” on page 32.

Example 1:
To enable the FFST generic alert output option for IBM’s VTAM application, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,APPLID=VTAM, VENDOR=IBM,OPTIONS=GENAL

Example 2:
To enable the FFST symptom record and duplicate dump suppression output options for IBM’s VTAM application, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,APPLID=VTAM, VENDOR=IBM,OPTIONS=(SYMRC,SUPDP)

Example 3:
To enable all the FFST output options for IBM’s VTAM application, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,APPLID=VTAM, VENDOR=IBM,OPTIONS=ALL

Enabling output options by probe identifier

When you enable an output option by probe identifier, FFST generates the output for any triggered software probes for which the output was previously disabled at the same probe identifier level, unless the output was also disabled at one of the following levels:

- FFST level
- Application level
• A higher or lower probe statement level
• In the probe statement

When used to enable FFST output options by probe identifier, the MODIFY command has the following operands:

<table>
<thead>
<tr>
<th>Action</th>
<th>Operands</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTION</td>
<td>A=ENABLE</td>
</tr>
<tr>
<td>PROBEID</td>
<td>P=xxxxyyynn</td>
</tr>
<tr>
<td>OPTIONS</td>
<td>OPT=</td>
</tr>
<tr>
<td>DUMP</td>
<td></td>
</tr>
<tr>
<td>SYMRC</td>
<td></td>
</tr>
<tr>
<td>GENAL</td>
<td></td>
</tr>
<tr>
<td>SYMST</td>
<td></td>
</tr>
<tr>
<td>SUPDP</td>
<td></td>
</tr>
<tr>
<td>[ALL]</td>
<td></td>
</tr>
<tr>
<td>VENDOR</td>
<td>V=vendorname</td>
</tr>
<tr>
<td>[APPLID</td>
<td>AP=applname]</td>
</tr>
</tbody>
</table>

For more information about these operands, see “FFST MODIFY command overview” on page 32.

Example 1:
To enable the FFST generic alert output option for a single probe statement with the probe identifier ISTTSC07, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,PROBEID=ISTTSC07,OPTIONS=GENAL,VENDOR=IBM

Example 2:
To enable the FFST symptom record and duplicate dump suppression output options for all the probe statements that have a probe identifier prefix of IST and a source file identifier of TSC, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,PROBEID=ISTTSC*,OPTIONS=(SYMRC,SUPDP),VENDOR=IBM

Example 3:
To enable all the FFST output options for probe statements that have a probe identifier prefix of IST, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,PROBEID=IST*,OPTIONS=ALL,VENDOR=IBM

Enabling FFST probe message logging

When used to enable the FFST probe message logging function, the MODIFY ACTION=ENABLE command has the following format:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Operands</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODIFY</td>
<td>F procname ,ACTION</td>
</tr>
<tr>
<td>,LOGID</td>
<td>L=n</td>
</tr>
</tbody>
</table>

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to enable message logging for DD name FFSTLOG3, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,LOGID=FFSTLOG3

To enable the REUSE function, type one of the following commands:

MVS
MODIFY FFST,ACTION=ENABLE,LOGID=REUSE

To enable FFST's message logging function, type one of the following commands:
MVS
MODIFY FFST, ACTION=ENABLE, LOGID

Note: Enabling a probe message log will make it available for use the next time FFST loops through to that data set. This command no longer causes a change in the log data set being used.

Clearing message logs and knowledge of probe statements
The MODIFY ACTION=CLEAR command lets you perform the following tasks:

- Clear any knowledge of a probe statement. When you clear knowledge of a probe statement, FFST discards all the counter values and any other information associated with that probe statement. The next time the probe statement is executed, FFST treats it as a probe statement that is being executed for the first time. When you clear knowledge of probe statements for FFST or an application, FFST also enables any output options previously disabled at the same level.
- Clear a probe message log and make that message log the active message log.

The following sections explain how to perform these tasks.

Clearing knowledge of all probe statements for FFST
When used to clear knowledge of all probe statements, the MODIFY command has the following operands:

ACTION| A=CLEAR| CL
APPLID| AP=FFST

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to clear knowledge of all the probe statements known to FFST, type one of the following commands:

MVS
MODIFY FFST, ACTION=CLEAR, APPLID=FFST

Clearing knowledge of the probe statements in an application
When used to clear any knowledge of probe statements in a specific application, the MODIFY command has the following operands:

ACTION| A=CLEAR| CL
APPLID| AP=applname
VENDOR| V=vendorname | IBM | IBM CORPORATION

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to clear any knowledge of all the probe statements in IBM’s VTAM application, type one of the following commands:

MVS
MODIFY FFST, ACTION=CLEAR, APPLID=VTAM, VENDOR=IBM

Clearing knowledge of the probe statements by probe identifier
When used to clear knowledge of probe statements by probe identifier, the MODIFY command has the following operands:

ACTION| A=CLEAR| CL
PROBEID| P=xxxxyyy|xxxxyy*| xxx*
VENDOR| V=vendorname | IBM | IBM CORPORATION
[ APPLID| AP=applname ]

For more information about these operands, see “FFST MODIFY command overview” on page 32.
Example 1:
To clear knowledge of a single probe statement with the probe identifier ISTTSC07, type one of the following commands:

MVS
MODIFY FFST,ACTION=CLEAR,PROBEID=ISTTSC07,VENDOR=IBM

Example 2:
To clear knowledge of all the probe statements that have a probe identifier prefix of IST and a source file identifier of TSC, type one of the following commands:

MVS
MODIFY FFST,ACTION=CLEAR,PROBEID=ISTTSC*,VENDOR=IBM

Example 3:
To clear knowledge of all the probe statements that have a probe identifier prefix of IST, type one of the following commands:

MVS
MODIFY FFST,ACTION=CLEAR,PROBEID=IST*,VENDOR=IBM

Clearing a probe message log
When used to clear a probe message log, the MODIFY ACTION=CLEAR command has the following format:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Operands</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODIFY</td>
<td>F procname      ,ACTION</td>
</tr>
</tbody>
</table>

For more information about these operands, see “FFST MODIFY command overview” on page 32.
For example, to schedule the clearing of message log FFSTLOG3, type one of the following commands:

MVS
MODIFY FFST,ACTION=CLEAR,LOGID=FFSTLOG3

Note: Clearing a probe message log will not cause the data set to be immediately cleared, unless it is the current data set. The clear will be done the next time FFST loops through to the specified data set. Also, this command is not needed if you are using the REUSE function.

Displaying status
The MODIFY ACTION=DISPLAY command lets you display the operating status for the following entities:
- FFST
- The applications known to FFST
- Individual probe statements

The following sections explain how to display each of these statuses.

Displaying FFST status
When used to display FFST status, the MODIFY command has the following operands:

[ACTION|A=DISPLAY|D]
APPLID|AP=FFST
[VENDOR|V=vendorname|IBM|IBM CORPORATION]

For more information about these operands, see “FFST MODIFY command overview” on page 32.
For example, to display FFST status, type one of the following commands:
The status display for FFST includes the following information:

- FFST's operating status.
- A list of the applications known to FFST. (Include the VENDOR keyword if you want to limit this list to the applications of a specific vendor.)

FFST/MVS Status Display is an example of FFST status displays.

Displaying application status

When used to display application status, the MODIFY command has the following operands:

- **ACTION** = [DISPLAY|A]
- **APPLID** = applname
- **VENDOR** = vendorname

For example, to display the status of IBM's FFSTV1R2 application, type one of the following commands:

```
MVS
MODIFY FFST, APPLID=FFSTV1R2, VENDOR=IBM
```

When FFST displays operating status for an application, it includes the following information:

- The application's name and operating status
- The probe identifier for each probe statement in the application

FFST/MVS Application Status Display is an example of a status display for IBM application FFSTV1R2.
Displaying probe statement status

When used to display probe statement status, the MODIFY command has the following operands:

- **ACTION**: A=DISPLAY | D
- **PROBEID**: \( \text{P} = \text{xxxyyynn} | \text{xxxyyy*} | \text{xxx*} \)
- **APPLID**: \( \text{AP} = \text{applname} \)
- **VENDOR**: \( \text{V} = \text{vendorname} | \text{IBM} | \text{IBM CORPORATION} \)

For more information about these operands, see “FFST MODIFY command overview” on page 32.

When FFST displays operating status for a probe statement, it includes the following information:

- The probe identifier
- The probe statement's status
- The number of times the probe statement has executed

FFST Probe Statement Status Display (PROBEID=EPWIVP03) is an example of a status display for a probe statement with probe identifier EPWIVP03.

FFST Probe Statement Status Display (PROBEID=EPWIVP03)

FFST Probe Statement Status Display (PROBEID=EPWIVP*) is an example of a status display for a group of probe statements whose probe identifiers begin with the characters EPWIVP.
FFST Probe Statement Status Display (PROBEID=EPW*) is an example of status displays for a group of probe statements whose probe identifiers begin with the characters EPW.

<table>
<thead>
<tr>
<th>FFST Probe Statement Status Display (PROBEID=EPW*)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example 1:</td>
</tr>
<tr>
<td>To display the status of a single probe statement with the probe identifier EPWIVP03, type one of the following commands:</td>
</tr>
<tr>
<td><strong>MVS</strong></td>
</tr>
<tr>
<td>MODIFY FFST,PROBEID=EPWIVP03,VENDOR=IBM</td>
</tr>
<tr>
<td>Example 2:</td>
</tr>
<tr>
<td>To display the status of all the probe statements that have a probe identifier prefix of EPW and a source file identifier of IVP, type one of the following commands:</td>
</tr>
<tr>
<td><strong>MVS</strong></td>
</tr>
<tr>
<td>MODIFY FFST,PROBEID=EPWIVP*,VENDOR=IBM</td>
</tr>
<tr>
<td>Example 3:</td>
</tr>
<tr>
<td>To display the status of all the probe statements that have a probe identifier prefix of EPW, type one of the following commands:</td>
</tr>
<tr>
<td><strong>MVS</strong></td>
</tr>
<tr>
<td>MODIFY FFST,PROBEID=EPW*,VENDOR=IBM</td>
</tr>
</tbody>
</table>

**Changing output destinations**

The MODIFY ACTION=CHANGE command lets you perform the following tasks for the specified platform:

**MVS**

Change the destination of unformatted dumps and generic alerts for FFST or for an application known to FFST.

The following sections explain how to perform these tasks.

**Changing dump destination for FFST/MVS**

When used to change the destination of unformatted dumps for FFST/MVS, the MODIFY command has the following operands:

- **ACTION**: \( \text{A=CHANGE|CH} \)
- **DUMPQUAL**: \( \text{DQ=dumpqualifier} \)
- **DUMPVOL**: \( \text{DV=dump_volume_serial_number} \)
- **APPLID**: \( \text{AP=FFST} \)

For more information about these operands, see “FFST MODIFY command overview” on page 32.

**Note:** You do not have to include both DUMPQUAL and DUMPVOL. You can choose to include just one or the other.

For example, to send FFST unformatted dumps to a data set with high-level qualifier FFSTDUMP and volume serial number R12NB4, type the following command:

**MVS**

MODIFY FFST,ACTION=CHANGE,DUMPQUAL=FFSTDUMP,DUMPVOL=R12NB4
Changing dump destination for a specific MVS application
When used to change the destination of unformatted dumps for a specific application running in an MVS environment, the MODIFY command has the following operands:

```
ACTION|A=CHANGE|CH
APPLID|AP=applname
[DEFINE|DEF]
VENDOR|V=vendorname|IBM|IBM CORPORATION
DUMPQUAL|DQ=dumpqualifier
DUMPVOL|DV=dump_volume_serial_number
```

For more information about these operands, see “FFST MODIFY command overview” on page 32.

**Note:** You do not have to include both DUMPQUAL and DUMPVOL. You can choose to include just one or the other.

For example, to send FFST unformatted dumps for IBM's VTAM application to a data set with high-level qualifier FFSTDUMP and volume serial number R12NB4, type the following command:

```
MODIFY FFST,ACTION=CHANGE,APPLID=VTAM,VENDOR=IBM,DUMPQUAL=FFSTDUMP,DUMPVOL=R12NB4
```

Changing generic alert destination for FFST
When used to change the destination of generic alerts for FFST, the MODIFY command has the following operands:

```
ACTION|A=CHANGE|CH
ALRCVID|AL=alertreceiver
[APPLID|AP=FFST]
```

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to send FFST generic alerts to NetView alert receiver RCV001, type one of the following commands:

```
MVS
MODIFY FFST,ACTION=CHANGE,ALRCVID=RCV001
```

Changing generic alert destination for a specific application
When used to change the destination of generic alerts for a specific application, the MODIFY command has the following operands:

```
ACTION|A=CHANGE|CH
APPLID|AP=applname
[DEFINE|DEF]
VENDOR|V=vendorname|IBM|IBM CORPORATION
ALRCVID|AL=alertreceiver
```

For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to send FFST generic alerts for IBM's VTAM application to NetView alert receiver RCV001, type one of the following commands:

```
MVS
MODIFY FFST,ACTION=CHANGE,APPLID=VTAM,VENDOR=IBM,ALRCVID=RCV001
```

Resetting output destinations
The MODIFY ACTION=RESET command lets you perform the following tasks for the specified platform:

```
MVS
Reset the destination of unformatted dumps and generic alerts for FFST or for an application known to FFST. For FFST, the values are reset to the following FFST defaults:
```

50 z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
Resetting dump destination for FFST/MVS

When used to reset the destination of unformatted dumps for FFST/MVS, the MODIFY command has the following operands:

```
ACTION|A=RESET|R
DUMPQUAL|DQ=[dumpqualifier]
DUMPVOL|DV=[dump_volume_serial_number]
[APPLID|AP=FFST]
```

For more information about these operands, see “FFST MODIFY command overview” on page 32.

You do not have to include both DUMPQUAL and DUMPVOL. You can choose to include just one or the other.

**Example 1:**
To reset the high-level dump data set qualifier to the FFST default, type the following command:

```
MODIFY FFST,ACTION=RESET,DUMPQUAL
```

**Example 2:**
To reset the FFST dump data set high-level qualifier and volume to the FFST default, type the following command:

```
MODIFY FFST,ACTION=RESET,DUMPQUAL,DUMPVOL
```

Resetting dump destination for a specific MVS application

When used to reset the destination of unformatted dumps for a specific MVS application, the MODIFY command has the following operands:

```
ACTION|A=RESET|R
APPLID|AP=applname
[DEFINE|DEF]
VENDOR|V=[vendorname|IBM|IBM CORPORATION]
DUMPQUAL|DQ=[dumpqualifier]
DUMPVOL|DV=[dump_volume_serial_number]
```

For more information about these operands, see “FFST MODIFY command overview” on page 32.

The destination is reset to the FFST value. You do not have to include both DUMPQUAL and DUMPVOL. You can choose to include just one or the other.

For example, to reset the dump data set high-level qualifier and volume for IBM's VTAM application to the FFST values, type the following command:

```
MODIFY FFST,ACTION=RESET,APPLID=VTAM,VENDOR=IBM,DUMPQUAL,DUMPVOL
```

Resetting generic alert destination for FFST

When used to reset the destination of generic alerts for FFST, the MODIFY command has the following operands:

```
ACTION|A=RESET|R
ALRCVID|AL=[alertreceiver]
[APPLID|AP=FFST]
```

FFST operations 51
For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to reset the destination of FFST generic alerts to the FFST default, type one of the following commands:

```
MVS
   MODIFY FFST,ACTION=RESET,ALRCVID
```

### Resetting generic alert destination for a specific application

When used to reset the destination of generic alerts for a specific application, the MODIFY command has the following operands:

```
ACTION|A=RESET|R
APPLID|AP=applname
[DEFINE|DEF]
VENDOR|V=vendorname|IBM CORPORATION
ALRCVID|AL[=alertreceiver]
```

The destination is reset to the FFST value. For more information about these operands, see “FFST MODIFY command overview” on page 32.

For example, to reset the generic alert destination for IBM’s VTAM application to the FFST value, type one of the following commands:

```
MVS
   MODIFY FFST,ACTION=RESET,APPLID=VTAM,VENDOR=IBM,ALRCVID
```

### Stopping FFST

You can stop FFST using any of the following commands:

- **MODIFY ACTION=HALT**

  If you use this command, FFST completes all currently running and pending processes before it stops. Also, if there are any applications that have an active interface to FFST, FFST does not stop running until each of these applications terminates its interface.

  Using the MODIFY ACTION=HALT command is the preferred method for stopping FFST. It has the following formats:

  ```
  MVS
  MODIIFY FFST,ACTION=HALT
  ```

- **MODIFY ACTION=HALT,QUICK**

  If you use this command, FFST stops running after the operator replies to a message which verifies this action. (A product cannot terminate its interface to FFST while FFST is not running.) FFST completes both currently running and pending processes before it stops.

  The MODIFY ACTION=HALT,QUICK command has the following formats:

  ```
  MVS
  MODIFY FFST,ACTION=HALT,QUICK
  ```

- **STOP**

  This command has the same effect as the MODIFY ACTION=HALT,QUICK command, except that FFST does not complete pending processes before it stops. The STOP command has the following formats:

  ```
  MVS
  STOP FFST
  ```

  In an MVS environment, use one of the following commands:

  ```
  STOP FFST
  ```
– Using the MVS CANCEL or FORCE command to stop FFST can cause unpredictable results, including abends, in applications using FFST.

**Using a startup command list**

The FFST startup command list is a partitioned data set specified through a DD statement in the FFST JCL for MVS, through the START00 FFSTPARM file. This list lets you provide FFST with a set of operational control commands that you want it to process when the FFST Problem Source Identifier (PSI) is started. These commands establish the FFST working environment. The commands you can use in this list are explained in “Using FFST commands from a console” on page 36.

When you create this command list, keep the following guidelines in mind:

- For MVS, the DD name is FFSTPARM and the default member name is START00, which can be overridden with a parameter on the MVS START command.
- Any record with an asterisk (*) in column 1 is treated as a comment line and is ignored.
- Any record can be WIDTH=x (starting in column 1) to specify that FFST looks for commands in columns 1 through x of the following records. (The default value for x is 72.)
- Any non-comment record not containing other FFST commands can contain TRN=trnmemnm, where trnmemnm is the FFSTPARM member name used for the FFST transition ABEND code parameter list. The default name is TRNSABCD.
- Any non-comment record not containing other FFST commands can contain HWR=hwrmemnm, where hwrmemnm is the FFSTPARM member name used for the FFST hardware support parameter list. The default name is EPWHWR01.
- If MVS 5.2 is used, any record can contain system symbolics (such as &SYSNAME, &SYSCONLINE, etc.) which will be interpreted (by the ASASYMBM facility) before being processed.

**Using FFST output**

When an event triggers a software probe, FFST can generate the following output to help you identify, track, and analyze the event:

- A customized dump that is smaller than the traditional full-address space dump for MVS. (FFST initially creates an unformatted dump. You can use the dump formatting tool provided with FFST to create a formatted version of the dump. For more information, see “Formatting a customized dump for FFST/MVS” on page 54.
- A symptom string that uniquely identifies the event. FFST places the symptom string in the following output:
  - The formatted customized dump
  - The symptom record
  - The console message
  - The probe message log
  - The generic alert (if you have NetView)
- A console message that contains the application name, probe identifier, and primary symptom string.
- A probe message log entry. The probe message log records each probe statement that is executed.
- A Systems Network Architecture (SNA) generic alert which can be viewed.

In addition, this chapter also documents return codes for FFST initialization, executed probe statements, and FFST termination.

Unless application programmers disabled a particular type of output for a specific software probe during product development, you can enable each of these types of output using a set of FFST commands. You can also use these commands to disable any of the FFST outputs. For more information, see “Controlling FFST operation” on page 30.
This chapter explains how to use the different types of FFST output.

**Note:** FFST also generates various messages that contain information about FFST operation. “FFST messages” on page 80 provides explanations and suggested responses for these messages.

### Using the customized dump

Before you can read an FFST customized dump, you must format it using the FFST/MVS dump formatter, EPWDMPFM, or EPWDMPFV. The following sections explain how to format an unformatted dump and how to use the formatted output.

#### Formatting a customized dump for FFST/MVS

The EPWDMPFM program is an Interactive Problem Control System command list (IPCS CLIST) that lets you format unformatted FFST/MVS dumps. When FFST/MVS generates a customized dump, it saves the unformatted dump in a dynamically allocated data set. EPWDMPFM reads the dump data set and writes the formatted output to an output data set. You can view this data set using the online time-sharing option (TSO) browse function or the MVS utility, IEBGENER.

You can run EPWDMPFM under IPCS or under TSO. When run under TSO (as shown in the examples in this section), EPWDMPFM enters IPCS, formats the dump, and then exits IPCS. In addition, if you want to run EPWDMPFM from a TSO ID, you must first concatenate the FFST/MVS CLIST library to your TSO library list (SYSPROC) and the FFST/MVS panel library to your TSO panel library list (ISPPLIB).

The following sections explain how to perform the following tasks:

- Use the EPWDMPFM program to format an unformatted dump.
- Use the online help provided with EPWDMPFM.
- Print a customized dump after it is formatted.

#### Running EPWDMPFM for FFST/MVS

To format a customized dump with the EPWDMPFM program, perform the following steps:

1. Type EPWDMPFM on the TSO command line, as shown in Invoking EPWDMPFM, and press Enter.

   **Note:** You may also invoke EPWDMPFM from the IPCS command line. In this case, the IPCSPRNT should have already been set up and you may use the default dump as the data set to format.
2. Fill in the fields on the FFST DUMP FORMATTER screen (FFST DUMP FORMATTER Screen for EPWDMPFM).

![FFST DUMP FORMATTER Screen](image)

**Note:** You can process TSO commands from this screen without exiting the EPWDMPFM dump formatting routine.

**DUMP LIBRARY NAME**
Type the name of the file that contains the customized dump. The dump is the input for the dump formatter. (If another dump data set was specified previously, the name of that data set appears in this field. To use a new dump data set, type the new name over the existing name.)

**DUMP MEMBER NAME**
Type the member name for the dump if it is a member of a partitioned data set. (Some components of FFST/MVS use partitioned data sets for dumps taken because of FFST detected problems.)

**OUTPUT DESTINATION**
Type one of the following characters to indicate where you want FFST/MVS to send the formatted output:

- **T**
  EPWDMPFM displays the formatted output on your terminal.

- **P**
  EPWDMPFM places the formatted output in an output data set. If the IPCSPRNT data set is already set up, EPWDMPFM stores the formatted output there. Otherwise, you can specify the name of the output data set on the FFST DUMP OUTPUT DATA SET screen.

- **B**
  EPWDMPFM displays the formatted output on your terminal and places it in an output data set.

Type Y (Yes) or N (No) beside the following options:

**SYMPTOM STRING DATA will be part of output**
If you type Y, the formatted dump includes the primary symptom string and, if one exists, the secondary symptom string.

**FFST WORK AREA will be part of output**
If you type Y, the formatted dump includes the FFST/MVS work area.

**Note:** If a problem occurs during FFST/MVS execution, FFST/MVS support might require the FFST/MVS work area as a debugging aid.
DATA STRUCTURE TABLE(S) will be part of output
If you type Y, the formatted dump includes the data structure table (DST) and the default DST (if a default DST exists).

Note: If a problem occurs during FFST/MVS execution, FFST/MVS support might require the DSTs as a debugging aid.

AREA AROUND REGISTERS will be part of output
If you type Y, the formatted dump includes an X'800' byte area around each register.

DATA STRUCTURES will be part of output
If you type Y, the formatted dump includes the data structures requested by the probe statement.

For an example of a formatted dump that contains this information, see “Using a formatted dump” on page 59.

When you finish, press Enter.

3. If you typed P or B in the OUTPUT DESTINATION field and the IPCSPRNT data set is not already set up, FFST DUMP OUTPUT DATA SET Screen appears.

The OUTPUT DATA SET NAME field contains either a previously specified data set name or a data set name supplied by FFST/MVS.

An FFST/MVS-supplied data set name appears in the format userid.OUTPUT.qual3.qual4.. If the unformatted dump data set is sequential, qual3 and qual4 are the last 2 qualifiers of the unformatted dump data set. If the unformatted dump data set is partitioned, qual3 is the last qualifier of the unformatted dump data set, and qual4 is the dump member name.

Fill in the following fields:

OUTPUT DATA SET NAME
Make sure the name in this field is the name of the data set in which you want EPWDMPFM to store the formatted output.

If you want to change the data set name, type the new name over the existing name. The name you type will appear automatically the next time you use EPWDMPFM.

If a user-specified data set name appears in this field and you want to regenerate the FFST/MVS-supplied data set name, type blanks over the current data set name and press Enter. The FFST/MVS-supplied data set name appears in the field.
**DISPOSITION**

Type one of the following options:

**NEW**

Type NEW if you typed the name of a new output data set in the **OUTPUT DATA SET NAME** field. If you type NEW in the **DISPOSITION** field, you also need to perform the following steps:

a. In the **VOLUME SERIAL WHERE DATA SET WILL BE STORED** field, type the serial number of the volume where EPWDMPFM should store the output data set or leave it blank to use **UNIT(SYSDA)**.

b. In the **NUMBER OF PRIMARY BLOCKS TO BE ALLOCATED** field, type the number of primary blocks EPWDMPFM should allocate for the data set.

c. In the **NUMBER OF SECONDARY BLOCKS TO BE ALLOCATED** field, type the number of secondary blocks EPWDMPFM should allocate for the data set.

**Note:** For **BLKSZ**, **RECFM**, and **LRECL**, EPWDMPFM uses the values required for an IPCSPRNT data set.

**OLD**

Type OLD if you want to replace the contents of the existing output data set with the new formatted output.

**MOD**

Type MOD if you want to append the new formatted output to the contents of the existing output data set.

When you finish, Press **Enter**.

4. If this is the first time the dump formatter is being used or if IPCS cannot find the previously specified dump directory, **FFST DUMP DIRECTORY Screen** appears.

If the dump formatter is being used for the first time, the default dump directory (**userid.DMPDIR**) appears in the **DUMP DIRECTORY** field.

If a user previously specified a dump directory that IPCS cannot find, the name of that directory appears in the **DUMP DIRECTORY** field. (This situation occurs, for example, if you delete the previously specified dump directory.)
Make sure the name in the DUMP DIRECTORY field is the name of the directory you want to use. If you want to change the directory, type the new name over the existing name. EPWDMPFM will automatically use that directory the next time you format a dump.

If you type the name of a new directory, indicate its volume serial number, buffer space, and control interval size by filling in the appropriate fields.

When you finish, Press Enter.

5. Look for the following messages, which indicate when the dump formatting process starts and when it is complete:

<table>
<thead>
<tr>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>EPW9573I ENTERING IPCS</td>
</tr>
<tr>
<td>EPW9574I DUMP FORMATTING IN PROGRESS</td>
</tr>
<tr>
<td>EPW9575I DUMP FORMATTING COMPLETE</td>
</tr>
<tr>
<td>EPW9576I EXITING IPCS</td>
</tr>
</tbody>
</table>

**Note:** Messages EPW9573I and EPW9576I appear only if you are not already running EPWDMPFM under IPCS. If EPWDMPFM does not display any of the messages, an error message should appear below the TSO command line. When an error message appears, see “FFST messages” on page 80 for information about the error; then correct the problem and retry the process.

6. When FFST/MVS message EPW9575I appears indicating that dump formatting is complete, press Enter to use the browse function.

You can view the output online and, if necessary, print it using the MVS program IEBPTPCH. For more information, see “Printing a formatted dump for FFST/MVS” on page 59.

When you finish looking at the formatted dump, press F3 to redisplay the FFST DUMP FORMATTER screen, and perform one of the following steps:

- Repeat the process to format a different dump or format the same dump using different options.
- Press F3 to exit the FFST DUMP FORMATTER screen.

FFST/MVS saves the values specified on this screen as new EPWDMPFM defaults and redisplayes the TSO COMMAND PROCESSOR screen.

**Getting help with EPWDMPFM for FFST/MVS**

FFST/MVS provides help panels to assist you in using the EPWDMPFM program. To access these panels:

1. Perform one of the following steps:

- Press F1 at any time during the formatting process.
- Type help on the TSO command line and press Enter.

A panel similar to the one in Help Panel for the EPWDMPFM SETUP Function appears.
2. Press **Enter** to reveal the next panel (if one exists).

3. When you finish, press **F3** to return to the screen from which you requested help.

**Printing a formatted dump for FFST/MVS**

After you use EPWDMPFM to format a customized dump, you can use the MVS program IEBPTPCH to print the results. See Sample Job for Printing a Formatted Dump for a sample job to print a member from the output data set.

```plaintext
//PRINTMEM JOB (ACCOUNTING),
  //   MSGCLASS=A,CLASS=A,MSGLEVEL=(1,1)
  //STEP1 EXEC PGM=IEBPTPCH
  //SYSPRINT DD SYSOUT=A
  //SYSUT1 DD DSN=USER1.OUTPUT.FFSTV1R2.DMP00127,
       //   DISP=OLD
  //SYSUT2 DD SYSOUT=A
  //SYSIN DD *
    PRINT TYPORG=PS,MAXFLDS=1
    RECORD FIELD=(80)
/*
```

**Using a formatted dump**

When you format a dump for FFST/MVS, IPCS saves the formatted dump in the dump data set you specify. A formatted customized dump can contain the following information:

- Primary and secondary symptom strings. (When it appears in the customized dump, the secondary symptom string contains the register values for the triggered software probe.)
- The FFST work area.
- The applicable DSTs.
- The area around the data registers.
- The data structures specified in the DST.
Sample Customized Dump is a sample dump that contains all the information in the preceding list. The sample is an FFST/MVS dump formatted using EPWDMPFM. For information about specifying which information you want to include in a dump, see “Running EPWDMPFM for FFST/MVS” on page 54.
FFST operations 65
Sample Customized Dump

Using the symptom string

A symptom string consists of a set of keywords and values that appear in the following format:

```
kkkk/vvvvvvvvv vkkk/vvvvvvvvv vkkk/vvvvvvvvv . . .
```

where

- `kkkk` is a keyword.
- `vvvvvvvvv` is the value associated with that keyword.

Table 8 on page 67 provides a list of standard FFST keywords and their meanings.
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>AB</td>
<td>Abend code</td>
<td>AB/U0001</td>
</tr>
<tr>
<td>ADRS</td>
<td>Address</td>
<td>ADRS/000001C0</td>
</tr>
<tr>
<td>DEVS</td>
<td>Device type</td>
<td>DEVS/3380</td>
</tr>
<tr>
<td>FLDS</td>
<td>Field</td>
<td>FLDS/ASCB</td>
</tr>
<tr>
<td>LVLS</td>
<td>Product Level</td>
<td>LVLS/101</td>
</tr>
<tr>
<td>MS</td>
<td>Message</td>
<td>MS/IEF244I</td>
</tr>
<tr>
<td>OPCS</td>
<td>Program OP code</td>
<td>OPCS/02</td>
</tr>
<tr>
<td>OVS</td>
<td>Overlaid storage</td>
<td>OVS/CBLOCK01</td>
</tr>
<tr>
<td>PCSS</td>
<td>Any statement</td>
<td>PCSS/PF10</td>
</tr>
<tr>
<td>PIDS</td>
<td>Product ID</td>
<td>PIDS/569504402</td>
</tr>
<tr>
<td>PRCS</td>
<td>Return, status, condition code</td>
<td>PRCS/0000UNIT</td>
</tr>
<tr>
<td>REGS</td>
<td>Registers</td>
<td>REGS/GR15</td>
</tr>
<tr>
<td>RIDS</td>
<td>Resource identifications</td>
<td>RIDS/NUCLEU#L</td>
</tr>
<tr>
<td>SIG</td>
<td>Signal</td>
<td>SIG/ALARM</td>
</tr>
<tr>
<td>VALU</td>
<td>Field, register value</td>
<td>VALU/B01110101</td>
</tr>
<tr>
<td>WS</td>
<td>Coded wait</td>
<td>WS/E003F</td>
</tr>
</tbody>
</table>

Because duplicate events have identical primary symptom strings, the primary symptom string is the key to distinguishing between a new event and a known or duplicate event. If you request duplicate dump suppression, FFST uses the primary symptom string to identify duplicate events. In addition, you can use the primary symptom string to search the IBM problem database to determine whether the event is a known problem and whether any information about how to solve the problem is available.

The primary symptom string appears in the following FFST output:

- Message EPW0404I. This message appears on the console and in the probe message log. For more information, see “FFST messages” on page 80, “Using the console message” on page 72, and “Using the probe message log entry” on page 72.
- The symptom record. For more information, see “Using the symptom record” on page 68.
- The customized dump, if requested. For more information, see “Using the customized dump” on page 54.
- If you have NetView, the generic alert. For more information, see “Using the generic alert” on page 73.

Sample Primary Symptom String for FFST/MVS is an example of primary symptom strings.

Sample Primary Symptom String for FFST/MVS

```
PIDS/569504402 LVLS/101 PCSS/EPWIVP01 RIDS/EPWIVP FLDS/TEST FLDS/PROBE
FLDS/FOR FLDS/FFSTIVP
```

The primary symptom strings in Sample Primary Symptom String for FFST/MVS provides the following information about the event that caused the probe statement to be executed:

- The product identifier and level of the product that contains the triggered software probe (for FFST/MVS, 569504402, level 101)
• The probe identifier (EPWIVP01)
• The name of the module that contains the triggered software probe (EPWIVP)
• A message indicating that the triggered software probe is a test probe for FFSTIVP.

The secondary symptom string appears in the following FFST output:
• Customized dump
• Symptom record
• Console message
• Probe message log

When it appears in the customized dump and the symptom record, the secondary symptom string contains the register values for the triggered software probe. These register values do not appear in the console message and the probe message log.

Sample Secondary Symptom String is an example of a secondary symptom string without register values.

```
FLDS/CPUMODEL VALU/H3090
```

**Sample Secondary Symptom String**

The secondary symptom string provides additional information about the event, in this case the model number of the central processing unit (CPU).

### Using the symptom record

When a probe statement is executed, FFST generates a symptom record using IBM's Symptom Record Architecture. The following sections explain how to use FFST/MVS symptom records.

#### Using FFST/MVS symptom records

FFST places FFST/MVS symptom records in LOGREC, the MVS error record log. You can format and view a symptom record generated by FFST/MVS using the Environmental Record Editing and Printing (EREP) feature of the MVS operating system. For more information about this feature, refer to the Environmental Record Editing and Printing Program (EREP) User's Guide and Reference Manual (GC28-1378). Sample EREP Job for Printing an FFST/MVS Symptom Record illustrates a sample EREP job you can use to format and print a symptom record generated by FFST/MVS.

```
//STEP EXEC PGM=IFCEREP1,REGION=1024K,PARM=('HIST=N,PRINT=PS,TYPE=S,ACC=N')
//SERLOG DD DSN=SYS1.LOGREC,DISP=(OLD,KEEP),UNIT=DISK,VOL=SER=EREPLB2
//DIRECTWK DD UNIT=VIO,SPACE=(CYL,(5))
//EREPPT DD SYSOUT=*,DCB=BLKSIZE=133
//TOURIST DD SYSOUT=*,DCB=BLKSIZE=133
//SYSIN DD DUMMY
//ACCDEV DD DUMMY
//ACCIN DD DUMMY
```

**Sample EREP Job for Printing an FFST/MVS Symptom Record**

An FFST/MVS symptom record contains the following sections:
• The header
• The search argument abstract
• System environment information
• Component information
• Symptom strings
• Free-format component information
• The summary

The following sections provide examples of the different parts of a symptom record and explain what they mean.
The **FFST/MVS symptom record header**

The header section of an FFST/MVS symptom record provides the following information:

- The date the symptom record was created
- The time and date that the event occurred
- The version and release number of the system control program (SCP)
- The model and serial number of the CPU in which the event occurred

**FFST/MVS Symptom Record Header** is an example of an FFST/MVS symptom record header.

<table>
<thead>
<tr>
<th>TYPE: SYMPTOM RECORD</th>
<th>REPORT: SOFTWARE EDIT REPORT</th>
<th>DAY</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCP: VS 2 REL 3</td>
<td>REPORT DATE: 118 92</td>
<td></td>
</tr>
<tr>
<td>MODEL: 3090</td>
<td>ERROR DATE: 118 92</td>
<td>HH MM SS.TH</td>
</tr>
<tr>
<td>SERIAL: 070115</td>
<td>TIME: 11:09:32.32</td>
<td></td>
</tr>
</tbody>
</table>

**FFST/MVS Symptom Record Search Argument Abstract**

The search argument abstract is identical to the primary symptom string, and IBM customers can use it to search the IBM problem database to see whether the event has occurred before.

**FFST/MVS Symptom Record Search Argument Abstract** is an example of an FFST/MVS search argument abstract.

**SEARCH ARGUMENT ABSTRACT:**

PIDS/569504402 LVLS/101 PCSS/EPWIVP01 RIDS/EPWIVP FLDS/TEST FLDS/PROBE FLDS/FFSTIVP

**FFST/MVS Symptom Record Search Argument Abstract**

FFST/MVS symptom record system environment information

The FFST/MVS symptom record provides the following information about the system environment in which the event occurred:

- The date and time of the event
- The model and serial number of the CPU
- The name of the system, in this case NONAME
- The base control program (BCP), in this case MVS
- The release number or service level of the routine that failed
- The architecture level for the system data and component data
- System data (zeros for FFST)

System Environment Information in an FFST/MVS Symptom Record is an example of system environment information as it appears in an FFST/MVS symptom record.

**SYSTEM ENVIRONMENT:**

CPU MODEL: 3090                  DATE: 118 92
CPU SERIAL: 070115               TIME: 11:09:32.32
SYSTEM: FFSTESA1                BCP: MVS
RELEASE LEVEL OF SERVICE ROUTINE: HBB4410
SYSTEM DATA AT ARCHITECTURE LEVEL: 10
COMPONENT DATA AT ARCHITECTURE LEVEL: 10
RECORD IS ASSOCIATED WITH SVC DUMP
SYSTEM DATA: 00000000 00000000 |........|

**System Environment Information in an FFST/MVS Symptom Record**
FFST/MVS symptom record component information

An FFST/MVS symptom record provides the following information about the component in which the event occurred:

- The component identifier and release number or
- The program identifier and release number or
- A description of the event that includes the following information:
  - The type of event, in this case INCORROUT for an incorrect output error
  - The ID number for the component in which the event occurred
  - The name of the routine that detected the event

Component Information in an FFST/MVS Symptom Record is an example of component information as it appears in an FFST/MVS symptom record.

<table>
<thead>
<tr>
<th>COMPONENT INFORMATION:</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMPONENT ID:</td>
</tr>
<tr>
<td>COMPONENT RELEASE LEVEL:</td>
</tr>
<tr>
<td>DESCRIPTION OF FUNCTION:</td>
</tr>
<tr>
<td>PROBLEM ID:</td>
</tr>
</tbody>
</table>

Component Information in an FFST/MVS Symptom Record

FFST/MVS symptom record symptom string information

An FFST/MVS symptom record provides the following symptom string information:

- The primary symptom string and the secondary symptom string (if a secondary symptom string exists)
- An explanation of each of the values associated with the symptom string keywords

FFST/MVS Symptom Record Symptom String Information is an example of symptom string information as it appears in an FFST/MVS symptom record.

<table>
<thead>
<tr>
<th>PRIMARY SYMPTOM STRING:</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIDS/569504402 LVLS/101 PCSS/EPWIVP01 RIDS/EPWIVP FLDS/TEST</td>
</tr>
<tr>
<td>FLDS/PROBE FLDS/FOR FLDS/FFSTIVP</td>
</tr>
<tr>
<td>SYMPTOM SYMPTOM DATA EXPLANATION</td>
</tr>
<tr>
<td>-----------------------------</td>
</tr>
<tr>
<td>PIDS/569504402</td>
</tr>
<tr>
<td>LVLS/101</td>
</tr>
<tr>
<td>PCSS/EPWIVP01</td>
</tr>
<tr>
<td>RIDS/EPWIVP</td>
</tr>
<tr>
<td>FLDS/TEST</td>
</tr>
<tr>
<td>FLDS/PROBE</td>
</tr>
<tr>
<td>FLDS/FOR</td>
</tr>
<tr>
<td>FLDS/FFSTIVP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECONDARY SYMPTOM STRING:</th>
</tr>
</thead>
<tbody>
<tr>
<td>REGS/GR13 VALU/H02F00E00 REGS/GR14 VALU/H82F006BA REGS/GR15</td>
</tr>
<tr>
<td>VALU/H00000000 REGS/GR00 VALU/H82F006BA REGS/GR01 VALU/H02F006C0</td>
</tr>
<tr>
<td>REGS/GR02 VALU/H00000040 REGS/GR03 VALU/H009F6964 REGS/GR04</td>
</tr>
<tr>
<td>VALU/H009F948 REGS/GR05 VALU/H00F3E88 REGS/GR06 VALU/H02F00548</td>
</tr>
<tr>
<td>REGS/GR07 VALU/H00F6978 REGS/GR08 VALU/H009F190 REGS/GR09</td>
</tr>
<tr>
<td>VALU/H009F01C8 REGS/GR10 VALU/H00000000 REGS/GR11 VALU/H00F3E88</td>
</tr>
<tr>
<td>REGS/GR12 VALU/H22F00548</td>
</tr>
<tr>
<td>SYMPTOM SYMPTOM DATA EXPLANATION</td>
</tr>
<tr>
<td>-----------------------------</td>
</tr>
<tr>
<td>REGS/GR13</td>
</tr>
<tr>
<td>VALU/H02F00E00</td>
</tr>
<tr>
<td>REGS/GR14</td>
</tr>
<tr>
<td>VALU/H22F006BA</td>
</tr>
<tr>
<td>REGS/GR15</td>
</tr>
<tr>
<td>VALU/H00000000</td>
</tr>
<tr>
<td>REGS/GR00</td>
</tr>
<tr>
<td>VALU/H22F006BA</td>
</tr>
<tr>
<td>REGS/GR01</td>
</tr>
<tr>
<td>VALU/H02F006C0</td>
</tr>
<tr>
<td>REGS/GR02</td>
</tr>
<tr>
<td>VALU/H00000040</td>
</tr>
</tbody>
</table>
REGS/GR03          03                   GENERAL PURPOSE REGISTER
VALU/H009F6964     009F6964       ERROR RELATED HEXADECIMAL VALUE
REGS/GR04          04                   GENERAL PURPOSE REGISTER
VALU/H009F6940     009F6940       ERROR RELATED HEXADECIMAL VALUE
REGS/GR05          05                   GENERAL PURPOSE REGISTER
VALU/H009F3E88     009F3E88       ERROR RELATED HEXADECIMAL VALUE
REGS/GR06          06                   GENERAL PURPOSE REGISTER
VALU/H02F0D054     02F0D054       ERROR RELATED HEXADECIMAL VALUE
REGS/GR07          07                   GENERAL PURPOSE REGISTER
VALU/H009F6978     009F6978       ERROR RELATED HEXADECIMAL VALUE
REGS/GR08          08                   GENERAL PURPOSE REGISTER
VALU/H009F3190     009F3190       ERROR RELATED HEXADECIMAL VALUE
REGS/GR09          09                   GENERAL PURPOSE REGISTER
VALU/H809FF1C8     809FF1C8        ERROR RELATED HEXADECIMAL VALUE
REGS/GR10          10                   GENERAL PURPOSE REGISTER
VALU/H00000000     00000000       ERROR RELATED HEXADECIMAL VALUE
REGS/GR11          11                   GENERAL PURPOSE REGISTER
VALU/H009F3E88     009F3E88       ERROR RELATED HEXADECIMAL VALUE
REGS/GR12          12                   GENERAL PURPOSE REGISTER
VALU/H82F09054     82F09054       ERROR RELATED HEXADECIMAL VALUE

FFST/MVS Symptom Record Symptom String Information
For more information about using symptom strings, see “Using the symptom string” on page 66

FFST/MVS symptom record free-format component information

FFST/MVS symptom record free-format component information, as shown in FFST/MVS Symptom Record Free-Format Component Information, is a hexadecimal dump of the symptom record.

FREE FORMAT COMPONENT INFORMATION:
HEX DUMP OF RECORD:
HEADER
+000    4C831800    00000000    0092132F    13214347    |<C.......K......|
+010    FF070115    30900000                            |........        |
SYMPTOM RECORD
+000    E2D9F3F0    F9F0F0F7    F0F1F1F5    FFFFCA5B    |SR3090070115...$|
+010    A5ACEBA5    DAB14602    40404040    40404040    |V..V....        |
+020    4040C6C6    E2E3C5E2    C1F1F5F7    F5F2C8C2    |  FFSTESA15752HB|
+030    C2F4F4F1    F0400080    E2E5C340    C4E4D4D7    |B4410 ..SVC DUMP|
+040    F1F00030    00640070    005D00D4    01A90131    |10.......).M.Z..|
+050    000402DA    00000000    00000000    00000000    |................|
+060    00000000    00000000    00000000    00000000    |................|
+070    E2D9F2F1    F1F0F5F6    F9F5F0F4    F4F0F100    |SR2110569504401.|
+080    F1F0F140    00000000    00000000    00000000    |101 ............|
+090    00000000    00000000    00000000    1DC9D5C3    |.............INC|
+0A0    D6D9D9D6    E4E34040    4040F5F6    F9F5F0F4    |ORROUT    569504|
+0B0    F4F0F140    C5D7E6C9    E5D70000    00000000    |401 EPWIVP......|
+0C0    00000000    40404040    40404040    00000000    |....        ....|
+0D0    00000000    D7C9C4E2    61F5F6F9    F5F0F4F4    |....PIDS/5695044|
+0E0    F0F140D3    E5D3E261    F1F0F140    D7C3E2E2    |01 LVLS/101 PCSS|
+0F0    61C5D7E6    C9E5D7F0    F140D9C9    D3E461C8    |/EPWIVP01 RIDS/E|
+100    C6D3C4E2    61D7D9D6    C2C5A06C    03A4E261    |/FFSTIVP01 RIDS/|
+110    C6D9D940    C6D3C4E2    61C6C6E2    E3C9E507    |/FLDS/TEST      |
+120    40C6D3C4    E216C107    E404D6C4    C5D0A0E5    |/FLDS/CPUMODEL V|
+130    C1D3E461    C83F3F09    F040D9C5    C72E16C7    |ALU/H82F09H REGS/G|
+140    D9F1F340    E5C1D3E4    61C8F0F2    C6F0F0C5    |R13 VALU/H82F00E|
+150    F9D4D09D     C57E261    C70F1F14    0E5C1D03    |00 REGS/GRI4 VAL|
+160    E461C8F8    F2C6F0F0    F6C2C100    D9C5C7E2    |U/H82F0606B REGS|
+170    61C7D9F1    F540E5C1    D3E4C618    F0F0F0F0    |GR15 VALU/H0000|
+180    F9C5C7E2    61C7D9F0    F348E5C1    D3E461C8    |REGS/GRI1 VALU/H02|
+190    C5C7E261    D9F0F540    E0C1D0E4    0C5D1E47    |REGS/G01 VALU/H02|
+1A0    C6F0F0F6    C3F0409D    C5C7E261    D9F0F540    | REGS/G02 VALU/H02|
+1B0    40E5C1D3    E61C8BF0    F0F0F0F0    F0F0F0F0    |VALU/H00000040|
+1C0    D9C5C7E2    61C7D9F0    F348E5C1    D3E461C8    |REGS/G03 VALU/H12|
+1D0    61C8F0F2    F040D9C5    C72E16C7    F0F0F0F0    |REGS/G02 VALU/H02|
+1E0    F9D4D09D     C57E261    C70F1F14    0E5C1D03    |REGS/G03 VALU/H02|
+1F0    61C7D9F1    F540E5C1    D3E4C618    F0F0F0F0    |REGS/G02 VALU/H02|
+200    F9D4D09D     C57E261    C70F1F14    0E5C1D03    |REGS/G03 VALU/H02|
+210    F040D9C5     C5C7E261    D9F0F540    E0C1D0E4    |0 REGS/G05 VALU/H05|
+220    61C8F0F2    F040D9C5    C72E16C7    F0F0F0F0    |/H009F64 REGS/G05|
+230    C7D9F0F6    40E5C1D3    E461C8F0    F2C6F0F0    |GR06 VALU/H02F09|
+240    5C7CEC7E    D9C5C7E2    61C7D9F0    F348E5C1    |054 REGS/G07 VA|
+250    F6F9F7F8    F040D9C5    C70F1F14    0E5C1D03    |LU/H080F0D978 REG|
+260    6261C17D9    F040D9C5    C7E261C7    D9F0F540    |S/G08 VALU/H0600|
+270    C6F3F1F9    F040D9C5    C7E261C7    D9F0F540    |F3190 REGS/G09|
+280    65C1D3E4    61C8F0F2    F9C6C6F1    C3F8A09D    |VALU/H099F1C8 R|
+290    C5C7E261    61C9F0F2    40E5C1D3    E461C8F0    |EGS/G10 VALU/H08|

FFST operations 71
The FFST/MVS symptom record summary provides a summary of each software record. Symptom Record Summary is an example of a symptom record summary.

| TYPE: SYMPTOM RECORD REPORT: SOFTWARE SUMMARY DAY YEAR |
| REPORT DATE: 118 92 |
| SCP: VS 2 REL 3 MODEL: N/A |
| SERIAL: N/A |
| PERIOD FROM: 118 92 TO: 118 92 |
| COUNT OF SYMPTOM RECORDS PROCESSED: 0001 |
| COUNT OF UNIQUE SYMPTOM STRINGS: 0002 |
| PIDS/566528901 LVLS/103 PCSS/ISTTSC01 RIDS/ISTTSCCM FLDS/PIU FLDS/INVALID |
| PIDS/569504401 LVLS/101 PCSS/EPWIVP01 RIDS/EPWIVP FLDS/TEST FLDS/PROBE |

Symptom Record Summary

Using the console message

When a probe statement is executed, FFST displays a message on the operator’s console. This message contains the following information:

- The name of the detecting application
- The probe identifier of the triggered software probe
- The name of the data set and volume that contain the unformatted dump (MVS only)
- The primary symptom string
- The secondary symptom string (if one exists)

Sample FFST/MVS Console Message is an example of an FFST/MVS console message.

```
EPW0401I FFSTPROC: EVENT DETECTION INVOKED BY FFSTV1R2
EPW0401I DUMP DATASET IS: FFST.FFSTESA1.FFSTV1R2.DMP00002
EPW0401I FOUND ON VOLUME: MVSST2
EPW0401I PRIMARY SYMPTOM STRING FOR PROBEID EPWIVP01 FOLLOWS:
EPW0401I PIDS/569504402 LVLS/101 PCSS/EPWIVP01 RIDS/EPWIVP
FLDS/TEST
EPW0401I FLDS/PROBE FLDS/FOR FLDS/FFSTIVP
EPW0401I SECONDARY SYMPTOM STRING FOR PROBEID EPWIVP01 FOLLOWS:
EPW0401I FLDS/CPUMODEL VALU/H3090
EPW0701I END OF MESSAGE GROUP
```

Using the probe message log entry

When a probe statement is executed, FFST adds an entry to the probe message log (for example, FFSTLOG1 or FFSTLOG2). The probe message log entry is identical to the console message. For information about how to use the information in the entry, see “Using the console message” on page 72.

FFST will use up to 9 probe message log data sets. These are specified in the FFST start-up JCL as DD names FFSTLOG1 through FFSTLOG9. FFST will only use the data sets that are sequentially numbered starting with FFSTLOG1. That is, if FFSTLOG1, FFSTLOG2, FFSTLOG3 and FFSTLOG5 DD statements are specified, only FFSTLOG1, FFSTLOG2, and FFSTLOG3 will be used. FFST will not recognized FFSTLOG5 as being specified. FFST initially starts with the FFSTLOG1 data set. When this data set fills up, it issues a message and automatically switches to the FFSTLOG2 data set. When this fills up, FFST will switch to the next data set. When the last data set fills up, FFST will switch back to FFSTLOG1. Each data set must be
defined as fixed, LRECL 80 and BLOCKSIZE any multiple of 80. There is a message log function called REUSE. If this function is disabled, FFST will not reuse a data set that contains data. That is, when a message log switch takes place, and the next log contains data, message log processing will be temporarily suspended. FFST will use that data set when it has been cleared (either with an ACTION=CLEAR modify command or by deleting all data by hand).

FFST log data sets can be individually disabled, enabled or cleared. When a data set is disabled, FFST will skip over that data set when it performs log switching. If it is the current data set, it will switch to the next enabled data set. If an ACTION=CLEAR modify command is entered for a data set, that log will be cleared and reused the next time FFST cycles through to use that data set. The data set is not immediately cleared (unless it is the current data set); it is merely scheduled for clearing. If the REUSE function is enabled, FFST will reuse a log data set even if it contains data. Therefore, the ACTION=CLEAR command should be of no use if REUSE is enabled. For more information on the use of the FFST modify command, see “Controlling FFST operation” on page 30.

Using the generic alert

A software generic alert is a Systems Network Architecture (SNA) function that notifies a network operations center when a software problem disrupts end-user services. If you use NetView in conjunction with FFST, the following events occur when a probe statement is executed:

- FFST generates a generic alert and passes it to NetView on the local processor (the processor where the probe statement was executed).
- Through local NetView, the generic alert can be:
  - Displayed
  - Logged
  - Sent to a NetView focal point

A generic alert generated by FFST contains the following information:

- The processor on which the software application that executed the probe statement
- The identification of the software application that executed the probe statement
- The date and time that the probe statement was executed
- A description of the event that caused the probe statement to be executed
- The event’s primary symptom string
- The name of the data set that contains the customized dump for the event
- The probable cause of the event
- Recommended recovery actions for the event

The following figures represent an example of the NetView generic alert Recommended Action screens.
The following figures represent an example of the NetView generic alert Event Detail Screens.
Note: The preceding screen contains an MVS example. For number 6, UNIT would appear, but MVSST2 would not.
The following figures represent an example of the NetView generic alert product set identification screens.
Note: The preceding screen contains an MVS example.

For more examples of generic alerts, refer to the NetView Operation Primer (SC30-3363) or the NetView Customization Guide (SC31-6016).

**FFST probe return codes**

The following table contains the return codes which are returned on Probe Initialization, Probe Statement Execution and Probe Termination.

<table>
<thead>
<tr>
<th>Hexadecimal Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000</td>
<td>The request was completed successfully.</td>
</tr>
<tr>
<td>00000104</td>
<td>EPWCNTRL INIT requested a conditional wait and FFST is not yet active.</td>
</tr>
<tr>
<td>00000110</td>
<td>The FFST subsystem is not defined to MVS.</td>
</tr>
<tr>
<td>00000210</td>
<td>The FFST initialization stub module is not linked.</td>
</tr>
<tr>
<td>00000310</td>
<td>The FFST interface module is not in an accessible library.</td>
</tr>
<tr>
<td>00000410</td>
<td>The product configuration module is not found.</td>
</tr>
<tr>
<td>00000710</td>
<td>The DCB specified in the EPWCNTRL INIT macro could not be opened.</td>
</tr>
<tr>
<td>00000810</td>
<td>Insufficient storage to allocate a temporary work area.</td>
</tr>
<tr>
<td>00000910</td>
<td>The previous attempt to initialize FFST failed. See the console message for more information.</td>
</tr>
</tbody>
</table>
### Table 9. Probe Initialization Return Codes (continued)

<table>
<thead>
<tr>
<th>Hexadecimal Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000B10</td>
<td>The configuration load module was not link edited with the reusable attribute (REUS).</td>
</tr>
<tr>
<td>0000150C</td>
<td>The EPWCNTRL INIT function was requested specifying a configuration that indicates SHARED, but another configuration with the same prefix that also indicates SHARED is active.</td>
</tr>
<tr>
<td>0000160C</td>
<td>The EPWCNTRL INIT function was requested specifying a configuration that does not match another configuration with the same prefix that is active in the same address space.</td>
</tr>
<tr>
<td>0000170C</td>
<td>The EPWCNTRL INIT function failed because the pre-exit was not found on an accessible library.</td>
</tr>
<tr>
<td>0000180C</td>
<td>The EPWCNTRL INIT function failed because the post-exit was not found on an accessible library.</td>
</tr>
<tr>
<td>0000190C</td>
<td>The EPWCNTRL INIT function failed because the configuration load module was link-edited as re-entrant.</td>
</tr>
<tr>
<td>00001A0C</td>
<td>The EPWCNTRL INIT function failed because a DST specified shared storage or page-fixed storage, but the product is not authorized to use these facilities.</td>
</tr>
<tr>
<td>00001B0C</td>
<td>The EPWCNTRL INIT function failed because the configuration specified that it be page-fixed, but the product is not authorized to perform this function.</td>
</tr>
<tr>
<td>00001C0C</td>
<td>The EPWCNTRL INIT function failed because the configuration is marked shared, but the default DST is not marked shared.</td>
</tr>
<tr>
<td>00001D0C</td>
<td>The first byte of a DST does not contain the correct identifier. The load module specified may not actually be a DST.</td>
</tr>
<tr>
<td>00001E0C</td>
<td>The EPWCNTRL INIT function failed because a DST is marked shared, but the configuration is not.</td>
</tr>
<tr>
<td>00001F0C</td>
<td>The EPWCNTRL INIT function failed because a DST is marked page-fixed, but the configuration is not.</td>
</tr>
<tr>
<td>0000200C</td>
<td>The EPWCNTRL INIT function failed because allocation of a temporary area of storage failed.</td>
</tr>
</tbody>
</table>

### Table 10. Probe Statement Return Codes

<table>
<thead>
<tr>
<th>Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000</td>
<td>The request was completed successfully.</td>
</tr>
<tr>
<td>00000204</td>
<td>The probe statement was disabled by the system operator.</td>
</tr>
<tr>
<td>Return Code</td>
<td>Explanation</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>00000008</td>
<td>A minor error occurred - See the message log for more information.</td>
</tr>
<tr>
<td>0000010C</td>
<td>The probe parameter list contains an incorrect comma.</td>
</tr>
<tr>
<td>0000020C</td>
<td>The probe parameter list does not begin with the required asterisk(“*”).</td>
</tr>
<tr>
<td>0000030C</td>
<td>The probe parameter list contains a DST name that is too long.</td>
</tr>
<tr>
<td>0000040C</td>
<td>The probe parameter list contains a probe identifier that is too long.</td>
</tr>
<tr>
<td>0000050C</td>
<td>The probe parameter list contains a probe identifier that is too long.</td>
</tr>
<tr>
<td>0000060C</td>
<td>The probe parameter list contains a parameter count that is too large.</td>
</tr>
<tr>
<td>0000070C</td>
<td>The probe parameter list does not end with the correct characters (‘.’).</td>
</tr>
<tr>
<td>0000080C</td>
<td>Product initialization was not performed or product initialization failed.</td>
</tr>
<tr>
<td>0000090C</td>
<td>The system is unable to allocate a work area for FFST.</td>
</tr>
<tr>
<td>00000A0C</td>
<td>SETFRR failed.</td>
</tr>
<tr>
<td>00000B0C</td>
<td>The DST is not found.</td>
</tr>
<tr>
<td>00000C0C</td>
<td>The DST is not Release 2 level.</td>
</tr>
<tr>
<td>00000D0C</td>
<td>The DST is marked invalid.</td>
</tr>
<tr>
<td>00000E0C</td>
<td>Information for the probe statement is not found in the DST.</td>
</tr>
<tr>
<td>00000F0C</td>
<td>The cyclic redundancy check (CRC) failed.</td>
</tr>
<tr>
<td>0000100C</td>
<td>The probe identifier describes an entry in the DST that is not a probe entry.</td>
</tr>
<tr>
<td>0000110C</td>
<td>The probe statement does not support AR mode.</td>
</tr>
<tr>
<td>0000120C</td>
<td>A SDBSTR value is inaccessible.</td>
</tr>
<tr>
<td>0000130C</td>
<td>The issuer of the IPROBE macro is in secondary mode.</td>
</tr>
<tr>
<td>0000140C</td>
<td>An unexpected abend occurred.</td>
</tr>
<tr>
<td>0000210C</td>
<td>CNFGLMOD was used on the probe statement, but the probe statement was not issued in TCB mode.</td>
</tr>
<tr>
<td>0000220C</td>
<td>FFST disabled the probe statement because it was issued a specified number of times within a specified time period.</td>
</tr>
</tbody>
</table>
Table 10. Probe Statement Return Codes (continued)

<table>
<thead>
<tr>
<th>Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000510</td>
<td>The FFST subsystem is not active or the FFST virtual machine is not active.</td>
</tr>
<tr>
<td>00000610</td>
<td>The FFST entry point value is 0.</td>
</tr>
</tbody>
</table>

Table 11. Probe Termination Return Codes

<table>
<thead>
<tr>
<th>Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000</td>
<td>The request was completed successfully.</td>
</tr>
<tr>
<td>00000210</td>
<td>The FFST termination stub module is not linked.</td>
</tr>
<tr>
<td>00000310</td>
<td>The FFST interface module is not in an accessible library.</td>
</tr>
<tr>
<td>00000410</td>
<td>The product configuration module is not found.</td>
</tr>
<tr>
<td>00000510</td>
<td>The FFST subsystem is not active or the FFST virtual machine is not active.</td>
</tr>
<tr>
<td>00000810</td>
<td>Insufficient storage to allocate a temporary work area.</td>
</tr>
<tr>
<td>00000A10</td>
<td>The product's interface to FFST is not initialized.</td>
</tr>
</tbody>
</table>

**FFST messages**

This appendix lists in numerical order all the messages that can appear during FFST operation. Along with the actual message text, this appendix provides an explanation for each message, as well as the following information:

**System action**
Indicates how the system responds to the condition that generated the message.

**Operator response**
Indicates how the system operator should respond to the message.

**Programmer response**
Indicates how the programmer should respond to the message.

**Suppression level**
Indicates whether or not the message can be suppressed. The suppression level is one of the following values:

**DEBUG**
The message appears because the DEBUG parameter was included as input for the task that issues the message.

**Suppressible (SUP)**
The operator can suppress the message by entering a MODIFY command.

**Unsuppressible (UNSUP)**
The operator cannot suppress the message. FFST issues the message whenever the conditions that generate the message exist.

**Blank suppression**
Indicates where the message is sent or where it can be viewed. The destination can be any of the following locations:
**Console**

FFST sends the message to the operator's console.

**Terminal**

FFST sends the message to a user's terminal.

**Message log**

FFST records the message in the probe message log.

**Output data set**

EPWDMPFM messages are included in the formatted dump.

An identifier precedes each of the FFST messages in this appendix. This identifier has the following format: EPWaaxxy where:

- EPW is the prefix for all FFST messages.
- aaxx is a unique 4-digit numeric identifier where:
  - aa indicates the process that generated the message:
    - 00: Preinitialization.
    - 02: Initialization.
    - 03: Initialization.
    - 04: Probe statement operation.
    - 05: Message log operation.
    - 06: Command processing.
    - 07: Message processing.
    - 10: Customer Information Control System (CICS) application processing (MVS only).
    - 90: Installation verification program processing.
    - 95: CLIST processing. These messages appear in a dump processed by one of the FFST CLISTs.
    - 99: Debug processing.
  - xx is a decimal value from 01 to 99.
- y indicates the type of message and is one of the following values:
  - I: Information. The message is for information only. No action is required.
  - E: Error. The message indicates that an error occurred.
  - D: Debug. The message appears only when FFST is running in debug mode.

In most FFST messages, the procname variable is the procedure name you designate for FFST in your MVS JCL. **BOLD UPPERCASE CHARACTERS** represent the actual text of the message, while **lowercase italic characters** represent the variable information that is different each time the message appears. When FFST issues a message, it replaces these variables with actual information.
Note: Messages EPW0051 through EPW0066 and message EPW0204 apply only to FFST/MVS.

**EPW0000E** INITIALIZATION FAILED - reason

Explanation
The initialization of FFST failed for one of the following reasons:

**TOO MANY PARAMETERS SPECIFIED**
More than 3 input parameters were specified with the FFST procedure.

**INVALID LANG PARAMETER SPECIFIED**
The specified LANG parameter is invalid. LANG can be only 3 characters long and, when specified, must be the first parameter. If this parameter is not specified, the default value for LANG is ENU.

**INVALID MODE PARAMETER SPECIFIED**
The specified MODE parameter is invalid. MODE can either be NORMAL or DEBUG and, when specified, must be the second parameter. If this parameter is not specified, the default value for MODE is NORMAL.

**INVALID PAGE PARAMETER SPECIFIED**
The specified PAGE parameter is invalid. PAGE can be any number between 1 and 2000 and, when specified, must be the third parameter. If this parameter is not specified, the default value for PAGE is 200.

**INSUFFICIENT STORAGE FOR IFCVT**
Not enough storage was available to allocate the IFCVT control block.

**ESTAE COULD NOT BE ESTABLISHED**
The FFST extended specify task abnormal exit routine could not be established.

**MESSAGE PROCESSING FAILURE**
FFST message processing could not be established.

**FFST NOT AUTHORIZED**
FFST is not an authorized program.

System action
System processing continues, but no FFST processing can occur.

**Command line**

**Operator response**
Give the FFST initialization output to the system programmer.

**Programmer response**
The response depends on the reason for the failure:

**TOO MANY PARAMETERS**
SPECIFIED—Specify up to 3 parameters on the FFST procedure, each separated by a comma.

**INVALID LANG PARAMETER SPECIFIED**
Specify the LANG parameter correctly.

**INVALID MODE PARAMETER SPECIFIED**
Specify the MODE parameter correctly.

**INVALID PAGE PARAMETER SPECIFIED**
Specify the PAGE parameter correctly.

**INSUFFICIENT STORAGE FOR IFCVT**
Ensure that the required storage is available for FFST before it is initialized.

**ESTAE COULD NOT BE ESTABLISHED**
Determine why an ESTAE could not be established.

**MESSAGE PROCESSING FAILURE**
Determine why FFST cannot issue messages.

**FFST NOT AUTHORIZED**
Ensure that FFST is in an authorized library and is linked as an authorized program.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0001E** COMMAND FAILED - reason

Explanation
A MODIFY command failed for the following reason:

**FFST IS NOT ACTIVE**
FFST is not running in the FFST virtual machine.

System action
System processing continues, but no FFST processing can occur.

**Operator response**
Give the FFST initialization output to the system programmer.

**Programmer response**
The response depends on the reason for the failure:
**FFST IS NOT ACTIVE**

Issue the FFST START command.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW0051I FFSTFFDC: INITIALIZATION COMPLETE**

**Explanation**

Initialization of the FFDC function of FFST is complete.

**System action**

Processing continues.

**Operator response**

None.

**Programmer response**

None.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW0052I FFSTFFDC: INITIALIZATION FAILED - CSECT CHECKING OF modulename FAILED**

**Explanation**

Initialization of the FFDC function of FFST failed because a required FFDC module was not correct, where *modulename* is the FFDC module that failed the initialization check.

**System action**

Processing continues.

**Operator response**

Give the FFST initialization output to the system programmer.

**Programmer response**

Verify that the installation of FFST completed successfully.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW0053I FFSTFFDC: INITIALIZATION FAILED - LOAD OF modulename FAILED**

**Explanation**

Initialization of the FFDC function of FFST failed because a required FFDC module could not be loaded, where *modulename* is the FFDC module that failed the load function.

**System action**

Processing continues.

**Operator response**

Give the FFST initialization output to the system programmer.

**Programmer response**

Determine why the FFDC module could not be found in the FFST data sets.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW0054I FFSTFFDC: INITIALIZATION FAILED - funcname PROCESSING FAILURE**

**Explanation**

Initialization of the FFDC function of FFST failed because a required FFDC module was not correct, where *funcname* is the FFDC module that failed the initialization check.

**System action**

Processing continues.
**Explanation**
Initialization of the FFDC function of FFST failed because a required FFDC function failed to complete initialization, where `funcname` is the FFDC function that failed.

**System action**
Processing continues.

**Operator response**
Give the FFST initialization output to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0055I**  
**FFSTFFDC: FUNCTION `funcname` NOT AVAILABLE**

**Explanation**
An FFDC function is not available, where `funcname` is the unavailable FFDC function.

**System action**
Processing continues.

**Operator response**
Give the FFST initialization output to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0056I**  
**FFSTFFDC: TERMINATION COMPLETE**

**Explanation**
Termination of the FFDC function of FFST is complete.

**System action**
Processing continues.

**Operator response**
None.

**Programmer response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0057I**  
**FFSTFFDC: FUNCTION HAS BEEN INVOKED BY FFSTV1R2**

**Explanation**
An FFDC software probe was triggered to gather documentation for a FFST/FFDC problem. Other EPW00xxI messages follow this message.

**System action**
Processing continues.

**Operator response**
None.

**Programmer response**
Give the FFST/FFDC documentation to the system programmer.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.
Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

**EPW0058I**  FFSTFFDC: DUMP CAN BE FOUND IN: *dumpmembername*

**Explanation**
An FFDC software probe was triggered and a dump was taken, where *dumpmembername* is the associated dump member in the dump data set. This message can be issued when EPW0057I is issued.

**System action**
Processing continues.

**Operator response**
Give the FFST/FFDC documentation to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0059I**  FFSTFFDC: ON DATA SET: *dumpdataset*

**Explanation**
An FFDC software probe was triggered and a dump was taken, where *dumpdataset* is the name of the dump data set that contains the dump members. This message can be issued when EPW0057I is issued.

**System action**
Processing continues.

**Operator response**
Give the FFST/FFDC documentation to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0060I**  FFSTFFDC: SOFTWARE PROBE HAS BEEN DISABLED

**Explanation**
The FFDC function of FFST disabled a software probe because it was triggered more than 10 times. This message can be issued when EPW0057I is issued.

**System action**
Processing continues.

**Operator response**
Give the FFST/FFDC documentation to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0061I**  FFSTFFDC: DUMP SERVICES FAILED - RETURN CODE=*retcode*

**FFST operations**  85
An FFDC software probe was triggered and the dump services function failed, where `retcode` is the return code passed back by the dump services function. This message can be issued when EPW0057I is issued. Table 12 on page 86 provides explanations for the possible `retcode` values.

Table 12. Dump Return Codes

<table>
<thead>
<tr>
<th>Reason Code</th>
<th>Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0104</td>
<td>An input/output (I/O) error occurred while FFST was writing the dump. An incomplete dump is available.</td>
</tr>
<tr>
<td>0000</td>
<td>0204</td>
<td>FFST successfully wrote the dump, but it could not write the directory index member (FPSIDINO). A dump member called DUMP xxxx may be available.</td>
</tr>
<tr>
<td>0000</td>
<td>010C</td>
<td>FFST could not write the dump from the dump services member name to dump data set name DUMP xxxx. No dump is available.</td>
</tr>
<tr>
<td>0000</td>
<td>020C</td>
<td>An I/O error occurred while FFST was writing a dump data set. No dump is available.</td>
</tr>
<tr>
<td>0000</td>
<td>030C</td>
<td>Not enough storage was available below the 16MB line to allocate a dump work buffer. No dump is available.</td>
</tr>
<tr>
<td>0000</td>
<td>040C</td>
<td>FFST could not open the specified dump data set. No dump is available.</td>
</tr>
<tr>
<td>0000</td>
<td>050C</td>
<td>No space was available to write the dump on the primary extent, and all 16 extents had been used. No dump is available.</td>
</tr>
</tbody>
</table>

**System action**
Processing continues.

**Operator response**
Give the FFST/FFDC documentation to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0062I**
`FFSTFFDC: SYMPTOM RECORD SERVICES FAILED - RETURN CODE= retcode`
Explanation

An FFDC software probe was triggered and the symptom record services function failed, where retcode is the return code passed back by the symptom record services function. This message can be issued when EPW0057I is issued. Table 13 on page 87 provides explanations for the possible retcode values.

Table 13. Symptom Record Return Codes

<table>
<thead>
<tr>
<th>Reason Code</th>
<th>Return code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0000</td>
<td>The symptom record component completed successfully and the symptom record was recorded.</td>
</tr>
<tr>
<td>0000</td>
<td>0004</td>
<td>One or more errors were detected on the SYMREC macro statement. The entire input record was recorded. The symptom record component processed unsuccessfully for the following reason.</td>
</tr>
<tr>
<td>0164</td>
<td>000C</td>
<td>A serious error was on the SYMREC macro statement. No symptom record was recorded. The symptom record component processed unsuccessfully for one of the following reasons:</td>
</tr>
<tr>
<td></td>
<td>0104</td>
<td>The first 2 bytes of the input symptom record do not contain the SR operand.</td>
</tr>
<tr>
<td></td>
<td>0158</td>
<td>The total length of the input symptom record exceeds the maximum.</td>
</tr>
<tr>
<td></td>
<td>015C</td>
<td>Optional segments of the input symptom record were inaccessible. The record includes the accessible entries of the input symptom record.</td>
</tr>
<tr>
<td>Reason Code</td>
<td>Return code</td>
<td>Explanation</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>0108</td>
<td></td>
<td>The input symptom record does not contain the required entries for section 2.</td>
</tr>
<tr>
<td>010C</td>
<td></td>
<td>The input symptom record does not contain the required entries for section 2.1.</td>
</tr>
<tr>
<td>0114</td>
<td></td>
<td>The input symptom record does not contain the required entries for section 3.</td>
</tr>
<tr>
<td>0128</td>
<td></td>
<td>Portions of the input symptom record were inaccessible to a write request.</td>
</tr>
<tr>
<td>012C</td>
<td></td>
<td>Required portions of the input symptom record were inaccessible to a write request.</td>
</tr>
<tr>
<td>0134</td>
<td></td>
<td>The input symptom record address is in inaccessible storage.</td>
</tr>
<tr>
<td>0144</td>
<td></td>
<td>Program attributes of the job issuing the SYMREC macro are not written using the symptom record component standards.</td>
</tr>
<tr>
<td>0010</td>
<td></td>
<td>A serious error was in the symptom record component. The error is not related to SYMREC macro.</td>
</tr>
<tr>
<td>0014</td>
<td></td>
<td>The symptom record component is not operable.</td>
</tr>
</tbody>
</table>
System action
Processing continues.

Operator response
Give the FFST/FFDC documentation to the system programmer.

Programmer response
Gather the required documentation and contact the IBM support center.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0063I GENERIC ALERT SERVICES FAILED - RETURN CODE=retcode

Explanation
An FFDC software probe was triggered and the generic alert services function failed, where retcode is the return code passed back by the generic alert services function. This message can be issued when EPW0057I is issued. Table 14 on page 89 provides explanations for the possible retcode values.

<table>
<thead>
<tr>
<th>Reason Code</th>
<th>Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0000</td>
<td>An alert was passed to NetView.</td>
</tr>
<tr>
<td>0000</td>
<td>0004</td>
<td>The symptom record was found to be invalid.</td>
</tr>
<tr>
<td>0000</td>
<td>0008</td>
<td>A cause category of UNDETERMINE D was found with other cause categories.</td>
</tr>
<tr>
<td>0000</td>
<td>000A</td>
<td>An invalid cause category was found.</td>
</tr>
<tr>
<td>0004</td>
<td>000C</td>
<td>The specified receiver is not active. The PPI has received a copy of the NMVT, CP-MSG, or data buffer.</td>
</tr>
<tr>
<td>000A</td>
<td>000C</td>
<td>The PPI is available to process user requests.</td>
</tr>
<tr>
<td>000C</td>
<td>000C</td>
<td>Connection is delayed.</td>
</tr>
<tr>
<td>000E</td>
<td>000C</td>
<td>The receiver program is active.</td>
</tr>
<tr>
<td>000F</td>
<td>000C</td>
<td>The receiver program is inactive.</td>
</tr>
<tr>
<td>0010</td>
<td>000C</td>
<td>The receiver program is already active.</td>
</tr>
<tr>
<td>0012</td>
<td>000C</td>
<td>The receiver ECB is not zero.</td>
</tr>
<tr>
<td>0014</td>
<td>000C</td>
<td>The request type is not valid.</td>
</tr>
<tr>
<td>0016</td>
<td>000C</td>
<td>The program issuing this request is not executing in primary addressing mode.</td>
</tr>
<tr>
<td>0017</td>
<td>000C</td>
<td>The user program is not authorized.</td>
</tr>
<tr>
<td>0018</td>
<td>000C</td>
<td>The PPI is not active.</td>
</tr>
<tr>
<td>0019</td>
<td>000C</td>
<td>The ASCB address is not correct.</td>
</tr>
<tr>
<td>001A</td>
<td>000C</td>
<td>The receiver program is not defined.</td>
</tr>
</tbody>
</table>

Table 14. Generic Alert Return Codes (continued)
### Table 14. Generic Alert Return Codes (continued)

<table>
<thead>
<tr>
<th>Reason Code</th>
<th>Return Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>001C</td>
<td>000C</td>
<td>User requests are not supported for NetView V1R2 or earlier releases.</td>
</tr>
<tr>
<td>001E</td>
<td>000C</td>
<td>No data buffer in the receiver buffer queue.</td>
</tr>
<tr>
<td>001F</td>
<td>000C</td>
<td>The receiver buffer is not large enough to receive the incoming data buffer.</td>
</tr>
<tr>
<td>0020</td>
<td>000C</td>
<td>No NetView storage is available.</td>
</tr>
<tr>
<td>000C</td>
<td>0021</td>
<td>The buffer length is not valid.</td>
</tr>
<tr>
<td>0022</td>
<td>000C</td>
<td>The NMVT buffer length exceeds 512 bytes.</td>
</tr>
<tr>
<td>0023</td>
<td>000C</td>
<td>The receiver buffer queue is full.</td>
</tr>
<tr>
<td>0024</td>
<td>000C</td>
<td>ESTAE recovery cannot be established as requested.</td>
</tr>
<tr>
<td>0025</td>
<td>000C</td>
<td>Work area is not on a double word boundary.</td>
</tr>
<tr>
<td>0026</td>
<td>000C</td>
<td>Number of connects allowed to PPI for user exceeded.</td>
</tr>
<tr>
<td>0027</td>
<td>000C</td>
<td>Number of connects to PPI exceeded.</td>
</tr>
<tr>
<td>0028</td>
<td>000C</td>
<td>Invalid SENDER-ID or RECEIVER-ID.</td>
</tr>
<tr>
<td>005A</td>
<td>000C</td>
<td>A processing error has occurred.</td>
</tr>
<tr>
<td>005F</td>
<td>000C</td>
<td>Data transport error; reason in RPB.</td>
</tr>
<tr>
<td>0000</td>
<td>0014</td>
<td>A symptom string entity of less than 5 or more than 15 characters was found.</td>
</tr>
<tr>
<td>0000</td>
<td>0018</td>
<td>Unknown error. The alert was probably not sent.</td>
</tr>
</tbody>
</table>

**System action**
Processing continues.

**Operator response**
Give the FFST/FFDC documentation to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0064I**  
**FFSTFFDC: SYMPTOM STRING:**

**Explanation**
An FFDC software probe was triggered and the symptom string for that software probe is being displayed. Message EPW0065I follows this message.
This message can be issued when EPW0057I is issued.

**System action**
Processing continues.

**Operator response**
Give the FFST/FFDC documentation to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0065I symptomstring**

**Explanation**
An FFDC software probe was triggered and the symptom string for that software probe is being displayed, where is the symptom string for the software probe. Message EPW0064I precedes this message. This message can be issued when EPW0057I is issued.

**System action**
Processing continues.

**Operator response**
Give the FFST/FFDC documentation to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0066I**

**Explanation**
An FFDC software probe was triggered; this is the last message issued for the software probe. This message is issued when EPW0057I is issued.

**System action**
Processing continues.

**Operator response**
Give the FFST/FFDC documentation to the system programmer.

**Programmer response**
Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0201I procname: INITIALIZATION OF FFST IN PROGRESS**

**Explanation**
FFST initialization is in progress. Additional messages can be issued during the initialization process. When the initialization process is complete, FFST issues a message indicating that the initialization process is finished.

**System action**
Processing continues.

**Operator response**
None.

**Programmer response**
None.
Explanation
FFST initialization is complete. Software probes and commands can now be processed.

System action
Processing continues.

Operator response
This message is informational. You can now enter any of the following FFST commands:

- MODIFY ACTION=DISABLE
- MODIFY ACTION=ENABLE
- MODIFY ACTION=CLEAR
- MODIFY ACTION=DISPLAY
- MODIFY ACTION=CHANGE
- MODIFY ACTION=RESET
- MODIFY ACTION=HALT
- STOP

For more information about these commands, see “Controlling FFST operation” on page 30

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

Explanation
FFST initialization failed. The failure occurred for one of the following reasons:

ALREADY ACTIVE
The system already initialized FFST.

ANOTHER FFST IS SET AS THE CMS INTERFACE
Another GCS machine has already initialized itself as the FFST machine.

AUTHNAME COMMAND FAILED
The GCS AUTHNAME command failed.

INSUFFICIENT STORAGE
The system does not have enough storage to allocate the required control blocks and load the required modules.

IUCVINI SET COMMAND FAILED
The GCS IUCVINI command failed.

LOAD FAILED FOR AN FFST MODULE
During initialization, FFST attempted to load one of its modules and the load failed.

MACHEXIT MACRO FAILED
The GCS MACHEXIT command failed.

PROCESSING ERROR
An internal processing error occurred in FFST.

START COMMAND NOT USED TO INVOKE EPWINIT
FFST was not invoked with the MVS START command. The job was submitted or called in some other way.

SUBSYSTEM NOT FOUND
You did not define FFST as a subsystem of MVS.

System action
System processing continues, but no FFST processing can occur.

Operator response
If reason is "ALREADY ACTIVE," this message is informational; no action is required. If reason is any of the other defined values, give the FFST initialization output to the system programmer.

Programmer response
The response depends on the reason for the failure:

ALREADY ACTIVE
None.

ANOTHER FFST IS SET AS THE CMS INTERFACE
Log off the other FFST machine or do not attempt to bring another FFST up.

AUTHNAME COMMAND FAILED
Have the system programmer verify that the FFST saved segment exists.

INSUFFICIENT STORAGE
Ensure that the required storage is available for FFST before it is initialized.
IUCVINI SET COMMAND FAILED
Have the system programmer verify that the FFST saved segment exists.

LOAD FAILED FOR AN FFST MODULE
An FFST module could not be found in the FFST load library structure. See message EPW0207I for the module name. Ensure that this module is available during FFST initialization.

MACHEXIT MACRO FAILED
Have the system programmer verify that the FFST saved segment exists.

PROCESSING ERROR
The FFST/FFDC process should have been invoked. Gather this documentation and contact the IBM support center.

START COMMAND NOT USED TO INVOKE EPWINIT
Invoke FFST using the MVS START command.

SUBSYSTEM NOT FOUND
The FFST/FFDC process should have been invoked. Use this documentation to determine which subsystems have been defined to MVS. Ensure that subsystem FFST is defined.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0204I  procname: MVS LEVEL LESS THAN 2.2 - SYMREC NOT SUPPORTED

Explanation
FFST determined that the level of MVS is lower than Version 2.2. The FFST SYMREC function requires MVS Version 2.2 or newer.

System action
Processing continues, but the SYMREC function of FFST is not active.

Operator response
None.

Programmer response
To use SYMREC, reinstall FFST on an MVS system that is Version 2.2 or newer.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0205I  procname: LOAD FAILED FOR modulename - reason

Explanation
During initialization, FFST failed while attempting to load modulename for one of the following reasons:

COMMAND PROCESSING NOT AVAILABLE
The required command processing subtask could not be invoked.

DUMP SERVICES FUNCTION NOT AVAILABLE
The required dump services subtask could not be invoked.

SYMREC FUNCTION NOT AVAILABLE
The required symptom record services subtask could not be invoked.

GENERIC ALERT FUNCTION NOT AVAILABLE
The required generic alert services subtask could not be invoked.

CHECKPOINT FUNCTION NOT AVAILABLE
The required checkpoint services subtask could not be invoked.

HARDWARE EVENT MONITOR NOT AVAILABLE
The required hardware event services subtask could not be invoked.

System action
System processing continues, but FFST initialization terminates.

Operator response
Report the message and give the FFST initialization output to the system programmer.

Programmer response
One or more required subtasks could not be loaded. Ensure that the FFST startup procedure correctly identifies the load library structure that contains the FFST modules.

Suppression level
UNSUP.
Blank suppression
Yes.

Destination
Console.

EPW0206I procname: MESSAGE LOG
INITIALIZATION FAILED
modulename - reason

Explanation
The message logging function for FFST is disabled. During initialization, FFST did not successfully initialize the FFST message log for the following reason:

UNABLE TO ALLOCATE LOG BUFFER
Not enough storage was available below the 16MB line to allocate the message log buffer.

System action
FFST initialization continues.

Operator response
Report the message to the system programmer.

Programmer response
The response depends on the reason for the failure:

UNABLE TO ALLOCATE LOG BUFFER
Before the system attempts to initialize FFST, ensure that the required storage is available. After correcting this problem, restart FFST to activate the message logging function.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0208I procname: INVALID WIDTH VALUE
IN START MEMBER membername

Explanation
During the processing of FFST startup parameters, the WIDTH parameter in membername was found to be invalid.

System action
FFST initialization continues, but the WIDTH parameter is ignored.

Operator response
Give the FFST output to the system programmer.

Programmer response
Review the WIDTH start parameter in membername and ensure that the value specified for WIDTH is not greater than 80.

Suppression level
UNSUP.

Blank suppression
Yes.

EPW0207I procname: LOAD FAILED FOR
MODULE modulename

Explanation
FFST tried to load module modulename, but the module was not found in any FFST load library.

System action
System processing continues, but FFST processing terminates.

Operator response
Give the FFST output to the system programmer.

Programmer response
Message EPW0203E precedes this message. Ensure that module modulename is available during FFST initialization.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.
**Destination**

Console.

**EPW0209I**  
**proname: INVALID WIDTH VALUE IN START MEMBER **membername**

**Explanation**
This message appears during FFST initialization when commands are being processed from an FFST startup command list. It displays the command currently being processed.

**System action**
Processing continues.

**Operator response**
None.

**Programmer response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
No.

**Destination**
Console.

**EPW0210E**  
**proname: INVALID SYNTAX IN FFSTPARM START MEMBER **memname**

**Explanation**
One or more records in member **memname** in the FFSTPARM data set contains a syntax error. FFST is expecting one of the following on a record:

- A comment, indicated by an asterisk ('*') in column 1
- An indication of the width to be used, indicated by the keyword 'WIDTH=' starting in column 1
- An indication of the transition ABEND code parameter list FFSTPARM member name, indicated by the keyword 'TRN='
- An indication of the hardware support parameter list FFSTPARM member name, indicated by the keyword 'HWR='
- A command that can be interpreted exactly as it would appear on an FFST modify command, after the procedure name

**System action**
The rest of the record containing the syntax error is ignored, the rest of the member is processed, and FFST initialization continues.

**Operator response**
Recycle FFST after the syntax error is corrected.

**Programmer response**
Correct the syntax error.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0250I**  
**proname: FFST INITIALIZATION FOR **applname** COMPLETE

**Explanation**
The initialization of **applname** has completed. This application is now ready to use FFST.

**System action**
Application initialization continues.

**Operator response**
None.

**Programmer response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0251E**  
**proname: FFST INITIALIZATION FOR **applname** FAILED - reason

**Explanation**
The rest of the record containing the syntax error is ignored, the rest of the member is processed, and FFST initialization continues.
Explanation

The initialization of *applname* failed for one of the following reasons:

**ANOTHER SHARED CONFIGURATION FOUND**
The 3-character prefix for *applname* matches another configuration that is marked SHARED, but these configurations do not match.

**UNMATCHING CONFIGURATION USED**
The 3-character prefix for *applname* matches another configuration but does not match the configuration being initialized.

**ALLOCATE FAILED FOR WORK AREA**
The system does not have enough storage to allocate the requested number of FFST work areas.

**LOAD FAILED FOR PRE-EXIT**
The load for the *applname* pre-exit failed.

**LOAD FAILED FOR POST-EXIT**
The load for the *applname* post-exit failed.

**CONFIGURATION IS LINK-EDITED AS REENTRANT**
The configuration for *applname* has been link edited with the REENTRANT attribute.

**NOT AUTHORIZED TO USE SHARED OR FIXED STORAGE**
*applname* is not executing in an authorized mode, but the DST requests a load of this table into the system's common storage area or fixed storage.

**NOT AUTHORIZED TO PAGE-FIX CONFIGURATION**
*applname* is not executing in an authorized mode, but the configuration requests a load into fixed storage.

**LOAD FAILED FOR DST *dstname***
The load for DST *dstname* failed.

**GETMAIN FAILED FOR DST *dstname***
The system does not have enough storage to allocate space for DST *dstname*.

**CONFIGURATION NOT SHARED, UNLIKE DST*dstname***
DST *dstname* is marked SHARED, but the configuration is marked NOT SHARED.

**CONFIGURATION PAGEABLE, UNLIKE DST*dstname***
DST *dstname* is marked FIXED, but the configuration is marked PAGEABLE.

**CONFIGURATION SHARED, UNLIKE DEFAULT DST**
The product's configuration indicates that it is going to be used in a shared environment, so the default DST must also be used in a shared environment. However, the default DST does not indicate this option.

**INCORRECT IDENTIFIER FOUND IN DST**
The referenced DST does not begin with the correct identifier for a DST.

System action

System processing continues, but the initialization of *applname* to FFST fails.

Operator response

Give the FFST output to the system programmer.

Programmer response

The response depends on the reason for the failure:

**ANOTHER SHARED CONFIGURATION FOUND**
Gather the required documentation and contact the IBM support center for *applname*.

**UNMATCHING CONFIGURATION USED**
Gather the required documentation and contact the IBM support center for *applname*.

**ALLOCATE FAILED FOR WORK AREA**
Ensure that sufficient storage is available to execute *applname*. If you cannot correct the problem, gather the required documentation and contact the IBM support center for *applname*.

**LOAD FAILED FOR PRE-EXIT**
Ensure that all load libraries are available for *applname*. If you cannot correct the problem, gather the required documentation and contact the IBM support center for *applname*.

**LOAD FAILED FOR POST-EXIT**
Ensure that all load libraries are available for *applname*. If you cannot correct the problem, gather the required documentation and contact the IBM support center for *applname*.

**CONFIGURATION IS LINK-EDITED AS REENTRANT**
Gather the required documentation and contact the IBM support center for *applname*.

**NOT AUTHORIZED TO USE SHARED OR FIXED STORAGE**
Gather the required documentation and contact the IBM support center for *applname*.

**NOT AUTHORIZED TO PAGE-FIX CONFIGURATION**
Gather the required documentation and contact the IBM support center for *applname*.

**FRR SETUP FAILED**
Gather the required documentation and contact the IBM support center for FFST.

**LOAD FAILED FOR DST *dstname***
Ensure that all load libraries are available for *applname*. If you cannot correct the problem, gather the required documentation and contact the IBM support center for *applname*.

**GETMAIN FAILED FOR DST *dstname***
Gather the required documentation and contact the IBM support center for *applname*. 
CONFIGURATION NOT SHARED, UNLIKE DST

Gather the required documentation and contact the IBM support center for applname.

CONFIGURATION PAGEABLE, UNLIKE DST dstname

Gather the required documentation and contact the IBM support center for applname.

CONFIGURATION SHARED, UNLIKE DEFAULT DST

Gather the required documentation and contact the IBM support center for applname.

INCORRECT IDENTIFIER FOUND IN DST

Gather the required documentation and contact the IBM support center for applname.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0253I   procname: DST dstname FOR applname MARKED INVALID - reason

Explanation
One or more records in member memname in the FFSTPARM data set contains a syntax error. FFST is expecting one of the following on a record:

VALIDITY CHECK FAILED FOR DST
The validity checking algorithm found an invalid value when validating the entire DST.

VALIDITY CHECK FAILED FOR ENTRY entryname
The validity checking algorithm found an invalid value for entry entryname.

LOAD FAILED FOR EXIT exitname
Exit exitname could not be loaded because it was not found in an available library. If DST dstname is marked shared, the exit was not found in an LPALST library (MVS).

System action
System processing and the initialization of applname to FFST continue.

Operator response
Give this message to the system programmer.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

Programmer response
If the reason is LOAD FAILED FOR EXIT exitname, check that the installation of applname was correct. If necessary, gather the required documentation and contact the IBM support center for applname.

EPW0254I   procname: name BEING CHANGED FROM oldname TO newname

Explanation
During the initialization of an application, existing information was changed to match the information from the configuration, where name is either APPLICATION NAME, VENDOR NAME, or PRODUCT LONG NAME, newname is the new name of the application or vendor, and oldname is the old name of the application or vendor.

System action
System processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0270I   procname: ENF LISTEN REQUEST FAILED - reason

Explanation
FFST operations
The FFST transition code issued an ENFREQ macro with the LISTEN option, but it failed for one of the following reasons:

**DUPLICATE LISTEN REQUEST**
Another listen request had been issued from the same place in FFST code.

**ENFDS TABLE IS FULL**
The table that contains the information for the listen has no more room to satisfy the request.

**ENF MODULE ERROR**
An event notification facility (ENF) module failed. Look for another message to explain this failure.

**ENF NOT INITIALIZED**
The ENF function is not available to process the listen request.

**STORAGE NOT AVAILABLE**
Not enough temporary storage is available to process the listen request. System Action: FFST initialization continues.

**THE FFST INTERFACE MODULE IS NOT IN AN ACCESSIBLE LIBRARY**
Module EPWPINIT should be in linklist, but is not found.

**MODULE EPWTRNCF COULD NOT BE FOUND**
The configuration module for the transition code, EPWTRNCF, was not found in the load library.

**INSUFFICIENT STORAGE TO ALLOCATE A TEMPORARY WORK AREA**
Not enough storage is available to process the request.

**EPWCNTRL INIT FAILED PREVIOUSLY**
A previous INIT request was issued and failed. Look for a previous FFST message describing the failure.

**STORAGE NOT AVAILABLE**
Not enough temporary storage is available to process the listen request. System Action: FFST initialization continues.

**System action**
FFST initialization continues.

**Operator response**
Report the message to the system programmer.

**Programmer response**
If the reason is DUPLICATE LISTEN REQUEST, report the problem to FFST support. Otherwise, contact MVS support.

**Suppression level**
UNSUP.

**Blank suppression**
No.

**Destination**
Console.

**EPW0271I procname: EPWCNTRL INIT FAILED - reason**

**Explanation**
The FFST transition code issued an EPWCNTRL macro with the INIT option, but it failed for one of the following reasons:

**THE FFST INTERFACE MODULE IS NOT IN AN ACCESSIBLE LIBRARY**
Module EPWPINIT should be in linklist, but is not found.

**MODULE EPWTRNCF COULD NOT BE FOUND**
The configuration module for the transition code, EPWTRNCF, was not found in the load library.

**INSUFFICIENT STORAGE TO ALLOCATE A TEMPORARY WORK AREA**
Not enough storage is available to process the request.

**EPWCNTRL INIT FAILED PREVIOUSLY**
A previous INIT request was issued and failed. Look for a previous FFST message describing the failure.

**STORAGE NOT AVAILABLE**
Not enough temporary storage is available to process the listen request. System Action: FFST initialization continues.

**System action**
FFST initialization continues.

**Operator response**
Report the message to the system programmer.

**Programmer response**
If the reason is THE FFST INTERFACE MODULE IS NOT IN AN ACCESSIBLE LIBRARY or MODULE EPWTRNCF COULD NOT BE FOUND, check to see that the installation of FFST was successful. Otherwise, contact FFST support.

**Suppression level**
UNSUP.

**Blank suppression**
No.

**Destination**
Console.

**EPW0301I procname: TERMINATION OF FFST IN PROGRESS**

**Explanation**
FFST termination is in progress. Termination started for any of the following reasons:

- The operator entered the MODIFY ACTION=HALT command.
- The operator entered the **STOP** command
- FFST detected an unrecoverable error.

**System action**

FFST initialization continues.

**Operator response**

If the message is the result of a **MODIFY ACTION=HALT** command or a **STOP** command, the operator does not need to respond. If an abnormal termination occurs, use prior messages to determine the cause.

**Programmer response**

None.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW0302I procname: TERMINATION OF FFST COMPLETE**

**Explanation**

FFST termination is complete. Termination occurred either as a result of an unrecoverable error or because an operator entered a **MODIFY ACTION=HALT** or **STOP** command.

**System action**

System processing continues, but no further FFST processing occurs.

**Operator response**

The operator does not need to respond unless FFST needs to be restarted. To restart FFST, follow the normal startup procedure.

**Programmer response**

None.

**Suppression level**

UNSUP.

---

**EPW0303E procname: TERMINATION OF FFST FAILED - PROCESSING ERROR**

**Explanation**

FFST termination failed because of an internal processing error.

**System action**

System processing continues. The operator may need to flush FFST from the system.

**Operator response**

Give the FFST termination output to the system programmer.

**Programmer response**

Gather the required documentation and contact the IBM support center.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW0304E procname: ABEND abendcode IN MODULE modulename AT OFFSET displacement**

**Explanation**

An abend occurred in an FFST module, where:

- **abendcode** is the abend code.
- **modulename** is the module where the abend occurred.
- **displacement** is the offset into the abending module.

**System action**

System processing continues. FFST may recover from the abend and continue processing. If FFST does not recover, it must be restarted.
Operator response
Give the FFST output to the system programmer.

Programmer response
If message EPW0305E does not follow this message, the FFST/FFDC process should have been invoked to provide necessary failure data. Gather this data and contact the IBM support center.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0305E ABEND PSW IS abendpsw

Explanation
An abend occurred in an FFST module, where abendpsw is the abend PSW. This message follows message EPW0304E.

System action
System processing continues. FFST may recover from the abend and continue processing. If FFST does not recover, it must be restarted.

Operator response
Give the FFST output to the system programmer.

Programmer response
Gather the required documentation and contact the IBM support center.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0306E ABEND REGISTERS GR00 THRU GR15 ARE:

Explanation
An abend occurred in an FFST module, where reg is a general register number 00 through 15 and value is the hexadecimal value of the general register. This message follows messages EPW0304E, EPW0305E, and EPW0306E.

System action
System processing continues. FFST may recover from the abend and continue processing. If FFST does not recover, then it must be restarted.

Operator response
Give the FFST output to the system programmer.

Programmer response
Gather the required documentation and contact the IBM support center.
You issued a `MODIFY ACTION=HALT` command, and FFST found that the applications listed in message EPW0625I still have an active interface to FFST.

**System action**

FFST does not stop running until each of the specified applications terminates its interface to FFST.

**Operator response**

If you want FFST to stop running immediately, terminate the specified applications or issue the `MODIFY ACTION=HALT` command with the QUICK keyword.

**Programmer response**

None.

---

You issued a `MODIFY ACTION=HALT,QUICK` command or a `STOP` command, and FFST found that the applications specified in message EPW0625I still have an active interface to FFST.

**System action**

If the following conditions are true, FFST termination begins:

- Before beginning its own termination, FFST is waiting for applications to terminate their interfaces to FFST.
- Application `applname` is the last application with an active interface to FFST.

Otherwise, no action is taken.

**Operator response**

None.

**Programmer response**

None.
Blank suppression
Yes.

Destination
Console.

**EPW0370I**  
**procname**: ENF DELETE REQUEST  
**FAILED - reason**

Explanation
The FFST transition code issued an ENFREQ macro with the DELETE option, but it failed for one of the following reasons:

**ENFDS TABLE IS FULL**
The table that contains the information for the delete has no more room to satisfy the request.

**ENF MODULE ERROR**
An ENF module failed. Look for another message to explain this failure.

**ENF NOT INITIALIZED**
The ENF function is not available to process the delete request.

**STORAGE NOT AVAILABLE**
Not enough temporary storage is available to process the delete request.

**INVALID TOKEN USED**
The ENFREQ macro was issued specifying a token that was invalid.

System action
FFST termination continues.

Operator response
Report the message to the system programmer.

Programmer response
If the reason is **INVALID TOKEN USED**, report the problem to FFST support.
Otherwise, contact MVS support.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Console.

**EPW0371I**  
**procname**: EPWCNTRL TERM  
**FAILED - reason**

Explanation
The FFST transition code issued an EPWCNTRL macro with the TERM option, but it failed with one of the following reasons:

**THE FFST INTERFACE MODULE IS NOT IN AN ACCESSIBLE LIBRARY**
Module EPWPTERM should be in linklist, but is not found.

**MODULE EPWTRNCF COULD NOT BE FOUND**
The configuration module for the transition code, EPWTRNCF, was not found in the load library.

**INSUFFICIENT STORAGE TO ALLOCATE A TEMPORARY WORK AREA**
Not enough storage is available to process the request.

**INITIALIZATION WAS NOT DONE**
The EPWCNTRL INIT was never issued, or it failed when it was issued.

System action
FFST termination continues.

Operator response
Report the message to the system programmer.

Programmer response
If the reason is **THE FFST INTERFACE MODULE IS NOT IN AN ACCESSIBLE LIBRARY** or **MODULE EPWTRNCF COULD NOT BE FOUND**, check to see that the installation of FFST was successful. Otherwise, contact FFST support.
Otherwise, contact MVS support.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Console.

**EPW0401I**  
**procname**: EVENT DETECTION  
**INVOKED BY applname FOR PROBEID :mvprobeid**
Explanation
Application applname issued a probe statement with an identifier of probeid. FFST gathers the requested documentation and may save the information in a dump data set, and/or send the information (via generic alert) to Netview. Messages EPW0402I and EPW0404I should follow this message, unless FFST is suppressing the symptom string messages.

System action
Processing continues.

Operator response
Report the message to the programmer responsible for application applname. (Also include the information in messages EPW0402I and EPW0404I.)

Programmer response
Use the information provided to determine the cause of the problem that triggered the software probe.

Suppression level
SUP.

Blank suppression
Yes.

Destination
Console, message log.

EPW0403E EVENT DETECTION FAILED - reason

Explanation
The processing of a software probe failed for one of the following reasons:

DUMP SERVICES FAILURE
A dump services request failed. See the reason for the failure.

GENERIC ALERT FAILURE
FFST issued a generic alert request that failed. See the reason for the failure.

SYMREC SERVICES FAILURE
FFST issued a SYMREC macro that failed. See the reason for the failure.

PROCESSING ERROR
An internal processing error occurred in FFST.

System action
System processing and FFST probe processing continue.

Operator response
Report the message to the system programmer.

Programmer response
The response depends on the reason for the failure:

DUMP SERVICES FAILURE
Use message EPW0412I that follows to determine the cause of the failure.

GENERIC ALERT FAILURE
Use message EPW0412I that follows to determine the cause of the failure.

SYMREC SERVICES FAILURE
Use message EPW0412I that follows to determine the cause of the failure.
Gather the required documentation and contact the IBM support center.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, message log.

EPW0404I symptomstring
Explanation
This message follows message EPW0402I and is the symptom string for the probe statement identified by message EPW0402I.

System action
Processing continues.

Operator response
Record the symptom string, and give it to the programmer responsible for application applname identified in message EPW0401I. (Also include the information in messages EPW0401I and EPW0402I.)

Programmer response
Use the information provided to determine why the probe statement was issued.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, message log.

EPW0405I PROBEID probeid HAS BEEN DISABLED
Explanation
FFST disabled software probe probeid because it was triggered more than a specified number of times during a specified time period. (The default threshold is 10 times in 10 minutes.) FFST disabled this software probe to reduce the performance impact. Until probeid is enabled with a MODIFY ACTION=ENABLE command, FFST does not issue any more messages related to the same symptom string in message EPW0404I for this software probe.

System action
Processing continues.

Operator response
Report the message to the programmer responsible for application applname specified in message EPW0401I. (Also include the information in messages EPW0402I and EPW0404I.)

Programmer response
Use the information provided to determine why the probe statement was issued.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, message log.

EPW0406I DUMP DATASET IS: datasetname
Explanation
The datasetname variable indicates the name of the data set that contains the customized dump. This variable can be up to 44 characters long. If the data set name is SYS1.DUMPxx, check message IEA911E to ensure that a complete dump is available. If only a partial dump was taken, the SYS1.DUMPxx data sets may not be large enough to hold a complete system dump.

System action
Processing continues.

Operator response
Report the message to the programmer responsible for the application identified in message EPW0401I.

Programmer response
Use the information provided to locate the dump associated with this event.
Explanation
This message is the second in a group of messages beginning with EPW0406I. The \textit{volumename} variable identifies the volume of the data set that contains the customized dump.

System action
Processing continues.

Operator response
Report the message to the programmer responsible for the application \textit{applname} in message EPW0401I. (Also include the information in message EPW0406I.)

Programmer response
Use the information provided to locate the dump associated with this event.

Explanation
This message is issued to indicate any nonfailure reason that occurred during probe processing, where \textit{probereasoncode} can be:

- **ONLY A PARTIAL DUMP IS AVAILABLE**
  - The dump is incomplete and only a partial dump exists.

- **NO DUMP IS AVAILABLE BECAUSE OF A DUMP ERROR**
  - An error occurred during the dump process and no dump is available.

- **PAGE OF STORAGE NOT ACCESSIBLE FOR COPYING**
  - A storage area for the dump is not accessible.

- **INSUFFICIENT STORAGE TO COPY PAGE OF STORAGE**
  - Not enough storage is available to contain the dump.

- **ERROR OCCURRED DURING WRITE TO DUMP DATA SET**
  - An I/O error occurred while FFST was writing the dump.

- **ABEND OCCURRED IN PRE-EXIT**
  - An abend occurred during the process of the pre-probe exit.

- **PROBE ABSTRACT VALUE WAS TRUNCATED**
  - The software probe abstract is longer than 80 characters and has been truncated.

- **SYMPTOM STRING VALUE WAS TRUNCATED**
  - A symptom string value is greater than the maximum and has been truncated.

- **GENERIC ALERT TEXT OVERRIDE VALUE WAS TRUNCATED**
  - A generic alert text override value is greater than the maximum and has been truncated.

- **ABEND0C4-PROBE ABSTRACT NOT AVAILABLE**
  - An abend occurred while FFST was trying to access the storage that contains the software probe abstract.

- **GENERIC ALERT DESCRIPTOR NAME NOT FOUND IN DST**
  - A generic alert description name was not found in the referenced DST.

- **GENERIC ALERT CAUSE NAME NOT FOUND IN DST**
  - A generic alert cause name was not found in the referenced DST.

- **GENERIC ALERT TRUNCATED-EXCEEDS 512 BYTES**
  - The generic alert is greater than 512 bytes and has been truncated.

- **NOT AUTHORIZED TO ACCESS ALTERNATE ASID**
  - The program that issued the software probe is not authorized to access an alternate address space identifier (ASID).

- **ALESERV ADD FAILED FOR PASSED DATASPACE TOKEN**
  - The program that issued the software probe has provided incorrect information to access a dataspace.
ABEND0C4-INVALID POINTER FOUND DURING DATA COLLECTION
An abend occurred while FFST was trying to access storage to be included in the customized dump.

NOT ENOUGH AVAILABLE SPACE IN THE WORK AREA TO CONTINUE
All available space in the software probe work area has been used.

DATA STRUCTURE NAME NOT FOUND IN THE DST
A data structure name was not found in the referenced DST.

ABEND0C4-INVALID POINTER FOUND DURING SYMREC BUILD
An abend occurred while FFST was trying to access storage that contained a value for the symptom string.

DUMP HAS BEEN SUPPRESSED
A new dump has not been taken for this event.

LEVELS NOT FOUND IN SYMPTOM STRING-SET TO 999
The LVLS keyword was not found in the symptom string and has been set to a default value of 999.

System action
System processing and FFST probe processing continue.

Operator response
Report the message to the programmer responsible for the application applname in message EPW0401I.

Programmer response
The response depends on the reason:

ONLY A PARTIAL DUMP IS AVAILABLE
Correct the cause for the partial dump. If it is a SDUMP data set, it may be too small to contain a full dump.

NO DUMP IS AVAILABLE BECAUSE OF A DUMP ERROR
Correct the cause for the I/O error to the dump data set.

PAGE OF STORAGE NOT ACCESSIBLE FOR COPYING
Report this problem to the support organization responsible for the application applname in message EPW0401I.

INSUFFICIENT STORAGE TO COPY PAGE OF STORAGE
Report this problem to the IBM support center.

ERROR OCCURRED DURING WRITE TO DUMP DATA SET
Correct the cause for the I/O error to the dump data set.

ABEND OCCURRED IN PRE-EXIT
Report this problem to the support organization responsible for the application applname in message EPW0401I.

PROBE ABSTRACT VALUE WAS TRUNCATED
Report this problem to the support organization responsible for the application applname in message EPW0401I.

SYMPTOM STRING VALUE WAS TRUNCATED
Report this problem to the support organization responsible for the application applname in message EPW0401I.

GENERIC ALERT TEXT OVERRIDE VALUE WAS TRUNCATED
Report this problem to the support organization responsible for the application applname in message EPW0401I.

GENERIC ALERT DESCRIPTOR NAME NOT FOUND IN DST
Report this problem to the support organization responsible for the application applname in message EPW0401I.

GENERIC ALERT CAUSE NAME NOT FOUND IN DST
Report this problem to the support organization responsible for the application applname in message EPW0401I.

GENERIC ALERT TRUNCATED-EXCEEDS 512 BYTES
Report this problem to the support organization responsible for the application applname in message EPW0401I.

NOT AUTHORIZED TO ACCESS ALTERNATE ASID
Report this problem to the support organization responsible for the application applname in message EPW0401I.

ALESERV ADD FAILED FOR PASSED DATASPACE TOKEN
Report this problem to the support organization responsible for the application applname in message EPW0401I.

ABEND0C4-PROBE ABSTRACT NOT AVAILABLE
Report this problem to the support organization responsible for the application applname in message EPW0401I.

GENERIC ALERT TEXT OVERRIDE NAME NOT FOUND IN DST
Report this problem to the support organization responsible for the application applname in message EPW0401I.
NOT ENOUGH AVAILABLE SPACE IN THE WORK AREA TO CONTINUE
Report this problem to the IBM support center.

DATA STRUCTURE NAME NOT FOUND IN THE DST
Report this problem to the support organization responsible for the application applname in message EPW04011.

ABEND0C4-INVALID POINTER FOUND DURING SYMREC BUILD
Report this problem to the support organization responsible for the application applname in message EPW04011.

DUMP HAS BEEN SUPPRESSED
This is not a problem. Duplicate dump suppression has caused the dump to be suppressed.

LEVELS NOT FOUND IN SYMPTOM STRING
SET TO 999-This is not a problem. LVLS has been set to a default value.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, message log.

EPW0412I epw0403ereason

Explanation
Message EPW04034 has been issued to indicate that an FFST function has failed. This message will identify the reason for the failure, where epw0403ereason can be:

DYNAMIC ALLOCATION RETURN CODE =retcode
Dynamic allocation of a dump data set failed, and retcode is the return code.

DYNAMIC UNALLOCATION RETURN CODE = retcode
Dynamic unallocation of a dump data set failed, and retcode is the return code.

SMS DYNAMIC ALLOCATION FAILURE (97XX), REASON CODE = retcode
Dynamic allocation of a dump data set failed, and retcode is the SMS reason code.

NO DUMP INDEX NUMBER AVAILABLE.
No dump numbers are available.

FAILURE TO OPEN DYNAMICALLY ALLOCATED DUMP DATA SET
FFST is not able to open the dynamically allocated dump data set.

SDUMP SUPPRESSED-ANOTHER SDUMP IN PROGRESS
An SVC dump was suppressed because the capture phase of another SVC dump was in progress.

SDUMP SUPPRESSED-REQUESTED BY INSTALLATION
An SVC dump was suppressed by a request by the installation (for example, DUMP=NO at IPL or CHNGDUMP SET,NODUMP).

SDUMP SUPPRESSED-SLIP NODUMP COMMAND
An SVC dump was suppressed by a SLIP NODUMP command.

SDUMP SUPPRESSED-SYS1.DUMP DATA SET NOT AVAILABLE
An SVC dump was suppressed because a SYS1.DUMP data set was not available.

SDUMP SUPPRESSED BY MVS DAE
An SVC dump was suppressed by MVS DAE processing.

SDUMP FAILURE-PROGRAM NOT AUTHORIZED
An SVC dump failed because the application program was not authorized.

NetView RECEIVER IS NOT ACTIVE
The specified NetView alert receiver is not active.

NetView IS AVAILABLE TO PROCESS USER REQUESTS
NetView is now available to process generic alert requests.

NetView RECEIVER PROGRAM IS ACTIVE
The specified NetView alert receiver program is active.

NetView RECEIVER PROGRAM IS INACTIVE
The specified NetView alert receiver program is inactive.

NetView RECEIVER PROGRAM IS ALREADY ACTIVE
The specified NetView alert receiver program is already active.

NetView RECEIVER ECB IS NOT ZERO
The specified NetView alert receiver event control block (ECB) is not zero.

INVALID REQUEST TYPE
Invalid NetView alert receiver request type.

PROGRAM NOT EXECUTING IN PRIMARY ADDRESSING MODE
The program issuing the generic alert is not executing in primary addressing mode.

USER PROGRAM IS NOT AUTHORIZED
The program issuing the generic alert is not running in authorized mode.
NetView SUBSYSTEM IS NOT ACTIVE
The NetView generic alert subsystem is not active.

ASCB ADDRESS IS NOT CORRECT
The specified ASCB address is not correct.

RECEIVER PROGRAM IS NOT DEFINED
The specified NetView receiver program is not defined.

NetView RELEASE DOES NOT SUPPORT USER REQUEST
The specified NetView release does not support the generic alert request.

NO DATA BUFFER IN THE RECEIVER BUFFER QUEUE
No available buffer in the receiver buffer queue.

RECEIVER BUFFER SIZE TOO SMALL FOR Incoming DATA
The specified NetView receiver buffer size is too small for the incoming data.

NO NetView STORAGE IS AVAILABLE
No available storage in NetView for the generic alert.

INVALID BUFFER LENGTH
The network management vector transport (NMVT) buffer length is invalid.

NMVT BUFFER LENGTH EXCEEDS 512 BYTES
The NMVT buffer length is greater than 512 bytes.

RECEIVER BUFFER QUEUE IS FULL
The specified NetView receiver buffer queue is full.

ESTAE RECOVERY CANNOT BE ESTABLISHED
An ESTAE recovery cannot be established as requested.

INVALID SENDER-ID OR RECEIVER-ID
The NMVT buffer contains an invalid send or receiver identification.

NetView PROCESSING ERROR
A NetView processing error occurred.

DELIMITER NOT FOUND BETWEEN SDB KEYWORDS
No blank delimiter was found between the structure database keywords.

WRITE OF SECTION 1 FAILED
The input symptom record was successfully copied. However, an attempt to write section 1 information from the complete symptom record failed. The area was found nonaccessible to a write request.

SYMPTOM RECORD EXCEEDS MAXIMUM LENGTH
The total length of the input symptom record exceeds the maximum.

OPTIONAL SYMPTOM RECORD SEGMENTS INACCESSIBLE
Optional segments of the input symptom record were found nonaccessible. The record includes the accessible entries of the input symptom record.

SR NOT IN THE FIRST 2 BYTES OF SYMPTOM RECORD
The first 2 bytes of the input symptom record do not contain the SR operand.

INCOMPLETE SECTION 2 OF THE SYMPTOM RECORD
The input symptom record does not contain the required entries for section 2.

INCOMPLETE SECTION 2.1 OF THE SYMPTOM RECORD
The input symptom record does not contain the required entries for section 2.1.

INCOMPLETE SECTION 3 OF THE SYMPTOM RECORD
The input symptom record does not contain the required entries for section 3.

SYMPTOM RECORD SEGMENTS INACCESSIBLE
Portions of the input symptom record were found nonaccessible to a write request.

SYMPTOM RECORD REQUIRED SEGMENTS INACCESSIBLE
Required portions of the input symptom record were found nonaccessible to a write request.

SYMPTOM RECORD IN INACCESSIBLE STORAGE
The input symptom record is in nonaccessible storage.

PROGRAM NOT AUTHORIZED TO ISSUE SYMREC MACRO
The program issuing the SYMREC macro is not authorized.

INSUFFICIENT SPACE IN LOGREC BUFFER
Space in the LOGREC buffer is insufficient to accommodate the symptom record.

INSTALLATION PREVENTED UNAUTHORIZED USE OF SYMREC
The installation prevented the unauthorized caller from writing the symptom record to SYS1.LOGREC.

INSUFFICIENT STORAGE FOR SYMREC SERVICES
The SYMREC macro service routine could not acquire storage for its work area and for a copy of the symptom record.

MOVE OF SYMPTOM RECORD TO LOGREC FAILED
Failure occurred while moving the symptom record to the LOGREC buffer.

LOGIC ERROR IN SYMREC MACRO SERVICES
The SYMREC macro service routine has a logic error.
SYMPTOM RECORD COMPONENT NOT OPERABLE
The symptom record component is not operable.

System action
System processing and FFST probe processing continue.

Operator response
Report the message to the programmer responsible for the application applname in message EPW0401I.

Programmer response
The response depends on the reason:

DYNAMIC ALLOCATION RETURN CODE = retcode
Report this problem to the IBM support center.

DYNAMIC UNALLOCATION RETURN CODE = retcode
Report this problem to the IBM support center.

SMS DYNAMIC ALLOCATION FAILURE (97XX), REASON CODE = retcode
Report this problem to your MVS system programmer.

NO DUMP INDEX NUMBER AVAILABLE.
Report this problem to your MVS system programmer. Any previous dumps that are no longer needed should be deleted.

FAILURE TO OPEN DYNAMICALLY ALLOCATED DUMP DATA SET
Report this problem to the IBM support center.

SDUMP SUPPRESSED-ANOTHER SDUMP IN PROGRESS
Report this problem to the support organization responsible for the application applname in message EPW0401I.

SDUMP SUPPRESSED-REQUESTED BY INSTALLATION
If an SVC dump is desired, change the IPL start parameter or issue a CHNGDUMP command to allow dumps.

SDUMP SUPPRESSED-SLIP NODUMP COMMAND
If an SVC dump is desired, issue a command to reset the SLIP NODUMP command.

SDUMP SUPPRESSED-SYS1.DUMP DATASET NOT AVAILABLE
If an SVC dump is desired, clear out a SYS1.DUMP data set.

SDUMP SUPPRESSED BY MVS DAE
If an SVC dump is desired, report this problem to your MVS system programmer.

SDUMP FAILURE-PROGRAM NOT AUTHORIZED
Report this problem to the support organization responsible for the application applname in message EPW0401I.

NetView RECEIVER IS NOT ACTIVE
The specified NetView alert receiver is not active.

NetView IS AVAILABLE TO PROCESS USER REQUESTS
None.

NetView RECEIVER PROGRAM IS ACTIVE
None.

NetView RECEIVER PROGRAM IS INACTIVE
Activate the NetView alert receiver program.

NetView RECEIVER PROGRAM IS ALREADY ACTIVE
None.

NetView RECEIVER ECB IS NOT ZERO
Report this problem to the IBM support center.

INVALID REQUEST TYPE
Report this problem to the IBM support center.

PROGRAM NOT EXECUTING IN PRIMARY ADDRESSING MODE
Report this problem to the IBM support center.

USER PROGRAM IS NOT AUTHORIZED
Report this problem to the IBM support center.

NetView SUBSYSTEM IS NOT ACTIVE
Verify that the NetView subsystem has been started.

ASCB ADDRESS IS NOT CORRECT
Report this problem to the IBM support center.

RECEIVER PROGRAM IS NOT DEFINED
Verify that the NetView receiver program has been defined to NetView.

NetView RELEASE DOES NOT SUPPORT USER REQUEST
Report this problem to the IBM support center.

NO DATA BUFFER IN THE RECEIVER BUFFER QUEUE
Report this problem to the IBM support center.

RECEIVER BUFFER SIZE TOO SMALL FOR INCOMING DATA
Report this problem to the IBM support center.

NO NetView STORAGE IS AVAILABLE
Report this problem to the IBM support center.

INVALID BUFFER LENGTH
Report this problem to the IBM support center.

NMVT BUFFER LENGTH EXCEEDS 512 BYTES
Report this problem to the IBM support center.

RECEIVER BUFFER QUEUE IS FULL
Report this problem to the IBM support center.
ESTAE RECOVERY CANNOT BE ESTABLISHED
Report this problem to the IBM support center.

INVALID SENDER-ID OR RECEIVER-ID
Report this problem to the IBM support center.

NetView PROCESSING ERROR
Report this problem to the IBM support center.

DELIMITER NOT FOUND BETWEEN SDB KEYWORDS
Report this problem to the IBM support center.

WRITE OF SECTION 1 FAILED
Report this problem to the IBM support center.

SYMPTOM RECORD EXCEEDS MAXIMUM LENGTH
Report this problem to the IBM support center.

OPTIONAL SYMPTOM RECORD SEGMENTS INACCESSIBLE
Report this problem to the IBM support center.

SR NOT IN THE FIRST 2 BYTES OF SYMPTOM RECORD
Report this problem to the IBM support center.

INCOMPLETE SECTION 2 OF THE SYMPTOM RECORD
Report this problem to the IBM support center.

INCOMPLETE SECTION 2.1 OF THE SYMPTOM RECORD
Report this problem to the IBM support center.

INCOMPLETE SECTION 3 OF THE SYMPTOM RECORD
Report this problem to the IBM support center.

SYMPTOM RECORD SEGMENTS INACCESSIBLE
Report this problem to the IBM support center.

SYMPTOM RECORD REQUIRED SEGMENTS INACCESSIBLE
Report this problem to the IBM support center.

SYMPTOM RECORD IN INACCESSIBLE STORAGE
Report this problem to the IBM support center.

PROGRAM NOT AUTHORIZED TO ISSUE SYMREC MACRO
Report this problem to the IBM support center.

INSUFFICIENT SPACE IN LOGREC BUFFER
Report this problem to the IBM support center.

INSTALLATION PREVENTED UNAUTHORIZED USE OF SYMREC
Report this problem to your MVS system programmer.

INSUFFICIENT STORAGE FOR SYMREC SERVICES
Report this problem to the IBM support center.

MOVE OF SYMPTOM RECORD TO LOGREC FAILED
Report this problem to the IBM support center.

LOGIC ERROR IN SYMREC MACRO SERVICES
Report this problem to the IBM support center.

SYMPTOM RECORD COMPONENT NOT OPERABLE
Report this problem to your MVS system programmer.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, message log.

EPW0413I  proname: PROBEID probeid FAILED - reason

Explanation
The software probe with the probe identifier probeid failed for one of the following reasons:

INVALID PARAMETERS
MORE THAN 25 CHARACTERS-More than 25 characters were found in the first parameter passed by the software probe.

INVALID PARAMETERS
FIRST CHARACTER NOT *-The first character of the first parameter passed by the software probe does not contain an asterisk.

INVALID PARAMETERS
DST NAME LONGER THAN 8-The DST name passed by the software probe was more than 8 characters long.

INVALID PARAMETERS
PROBEID LONGER THAN 8-The probe identifier passed by the software probe was more than 8 characters long.

INVALID PARAMETERS
COMMA DOES NOT FOLLOW FLAGS-A comma does not follow the flags passed by the software probe.

INVALID PARAMETERS
COUNT LONGER THAN 3-More than 3 input parameter were passed by the software probe.

INVALID PARAMETERS-CRC VALUE LONGER THAN 7
The CRC value passed by the software probe was more than 7 characters long.

FFST INITIALIZATION OF PRODUCT NOT COMPLETE
The initialization of the product issuing the software probe has not completed.

NO WORK AREA AVAILABLE
No FFST work areas are available to process the software probe.
SETFRR FAILED
FFST was not able to establish an FRR recovery routine.

ESTAE FAILED
FFST could not run its abnormal exit routine.

SPECIFIED DST NOT FOUND
The DST specified by the software probe could not be found in the available libraries.

SPECIFIED DST INCOMPATIBLE WITH PRESENT RELEASE
The DST specified by the software probe is not compatible with the current release of FFST.

INVALID DST SPECIFIED
The DST specified by the software probe is invalid.

PROBE ENTRY NOT FOUND IN SPECIFIED DST
The software probe entry could not be found in the DST specified by the software probe.

PROBE INCOMPATIBLE WITH ENTRY IN DST
The software probe entry is not compatible with the entry in the DST specified by the software probe.

ENTRY IN SPECIFIED DST IS NOT A PROBE
The software probe entry in the DST specified by the software probe is not a software probe entry.

PROBE DOES NOT MATCH AR MODE OF PROBE IN DST
The access Register (AR) mode of the software probe entry of the DST specified by the software probe, does not match the AR mode of the issuing software probe.

PROBE SDBSTR VALUE IS INVALID
The SDBSTR value specified by the software probe is invalid.

PROBE ISSUER IN SECONDARY MODE
The software probe is being issued while running in secondary mode.

PROBE DISABLED - INVOKED TOO MANY TIMES
The symptom string for the probe has been seen by FFST more often than the threshold value allows. Therefore, FFST disabled this probe so no outputs will be generated for it.

UNEXPECTED ABEND OCCURRED
An unexpected abend occurred while processing a software probe.

System action
System processing continues and FFST probe processing continues.

Operator response
Report the message to the programmer responsible for the application applname in message EPW0401I.

Programmer response
The response depends on the reason:

INVALID PARAMETERS-MORE THAN 25 CHARACTERS
Report this problem to the support organization responsible for the application applname in message EPW0401I.

INVALID PARAMETERS-FIRST CHARACTER NOT *
Report this problem to the support organization responsible for the application applname in message EPW0401I.

INVALID PARAMETERS-DST NAME LONGER THAN 8
Report this problem to the support organization responsible for the application applname in message EPW0401I.

INVALID PARAMETERS-PROBEID LONGER THAN 8
Report this problem to the support organization responsible for the application applname in message EPW0401I.

INVALID PARAMETERS-COMMA DOES NOT FOLLOW FLAGS
Report this problem to the support organization responsible for the application applname in message EPW0401I.

INVALID PARAMETERS-COUNT LONGER THAN 3
Report this problem to the support organization responsible for the application applname in message EPW0401I.

INVALID PARAMETERS-CRC VALUE LONGER THAN 7
Report this problem to the support organization responsible for the application applname in message EPW0401I.

FFST INITIALIZATION OF PRODUCT NOT COMPLETE
Report this problem to the support organization responsible for the application applname in message EPW0401I.

NO WORK AREA AVAILABLE
Report this problem to the IBM support center.

SETFRR FAILED
Report this problem to the IBM support center.

ESTAE FAILED
Determine why FFST could not run the ESTAE.

SPECIFIED DST NOT FOUND
Report this problem to the support organization responsible for the application applname in message EPW0401I.
SPECIFIED DST INCOMPATIBLE WITH PRESENT RELEASE
Report this problem to the support organization responsible for the application applname in message EPW0401I.

INVALID DST SPECIFIED
Report this problem to the support organization responsible for the application applname in message EPW0401I.

PROBE ENTRY NOT FOUND IN SPECIFIED DST
Report this problem to the support organization responsible for the application applname in message EPW0401I.

PROBE INCOMPATIBLE WITH ENTRY IN DST
Report this problem to the support organization responsible for the application applname in message EPW0401I.

ENTRY IN SPECIFIED DST IS NOT A PROBE
Report this problem to the support organization responsible for the application applname in message EPW0401I.

PROBE DOES NOT MATCH AR MODE OF PROBE IN DST
Report this problem to the support organization responsible for the application applname in message EPW0401I.

PROBE SDBSTR VALUE IS INVALID
Report this problem to the support organization responsible for the application applname in message EPW0401I.

PROBE ISSUER IN SECONDARY MODE
Report this problem to the support organization responsible for the application applname in message EPW0401I.

PROBE DISABLED - INVOKED TOO MANY TIMES
Report this problem to the IBM support center.

UNEXPECTED ABEND OCCURRED
Report this problem to the IBM support center.

Explanation
FFST temporarily disabled the message log for one of the following reasons:

GETMAIN FAILED FOR LOG CONTROL BLOCK
FFST attempted to allocate storage for a control block to handle the probe message logs, which failed.

NO DATA SETS CAN BE USED
DD name FFSTLOG1 was not defined.

UNEXPECTED ABEND OCCURRED
An unexpected abend occurred during message log processing. After determining and correcting the problem, the operator can enable the log with the MODIFY LOG command.

System action
Processing continues.

Operator response
Report the message to the system programmer.

Programmer response
The response depends on the reason for the failure:

I/O ERROR OCCURRED
Correct the problem that caused the I/O error. When the problem is corrected, the operator can enable the message log with the MODIFY LOG command.

MESSAGE LOG IS FULL
Either copy the message log to another data set or print the message log. After the log is copied or printed, the operator can enable and clear the message log with the MODIFY LOG,CLEAR command.

OPEN FAILED FOR ddname
Correct the problem that caused the failure. When the problem is corrected, the operator can enable the message log with the MODIFY LOG command.

BLOCKSIZE NOT A MULTIPLE OF 80
Correct the problem that caused the failure. When the problem is corrected, the operator can enable the message log with the MODIFY LOG command.

UNEXPECTED ABEND OCCURRED
Report this problem to the IBM support center.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, message log.

EPW0501I     procname: MESSAGE LOG
TEMPORARILY DISABLED - reason

Suppression level
UNSUP.

Blank suppression
Yes.
Destination

Console.

EPW0502I proname: CHECKPOINT FUNCTION UNAVAILABLE - reason

Explanation

reason may be:

OPEN DCB FAILED
During checkpoint processing, an open request for the DCB for the checkpoint data set failed.

System action

Processing continues.

Operator response

Report the message to the system programmer.

Programmer response

The response depends on the reason for the failure:

OPEN DCB FOR OUTPUT FAILED
Report this problem to the IBM support center.

ALLOCATION OF WORK BUFFER FAILED
Report this problem to the IBM support center.

DATA SET FULL-RECOVERY ACTION REQUIRED
Delete and reallocate a larger FFST checkpoint data set.

Suppression level

UNSUP.

Blank suppression

Yes.

Destination

Console.

EPW0503E proname: CHECKPOINT PROCESSING FAILED - reason

Explanation

reason may be:

OPEN DCB FOR OUTPUT FAILED
During checkpoint processing, an open request for output failed for the checkpoint data set.

ALLOCATION OF WORK BUFFER FAILED
During checkpoint processing, the allocation of a work buffer failed.

ALLOCATION OF WORK BUFFER FAILED
The data set is not big enough to hold all the required checkpoint data.

System action

Processing continues.

Operator response

Report the message to the system programmer.

Programmer response

The response depends on the reason for the failure:

OPEN DCB FOR OUTPUT FAILED
Report this problem to the IBM support center.

ALLOCATION OF WORK BUFFER FAILED
Report this problem to the IBM support center.

DATA SET FULL-RECOVERY ACTION REQUIRED
Delete and reallocate a larger FFST checkpoint data set.

Suppression level

UNSUP.

Blank suppression

Yes.

Destination

Console.

EPW0504I proname: FFSTPARM MEMBER memname READ SUCCESSFULLY

Explanation

FFST read member memname from the FFSTPARM data set successfully. This message is issued to confirm that the member read is the one that is desired.

System action

System processing continues, and member memname is interpreted.

Operator response

None, unless memname is not the member that was intended to be used.

Programmer response

None.

Suppression level

UNSUP.

Blank suppression

Yes.
EPW0505I  procname: FFSTPARM MEMBER memname NOT FOUND

Explanation
FFST could not find member memname in the FFSTPARM data set. This message may not indicate an error, as there are default names for the members that can be read, and they may be optional.

System action
Processing continues.

Operator response
None, unless member memname should have been found, or is mis-spelled. In this case, recycle FFST after the system programmer corrects the problem.

Programmer response
Correct the I/O error and have the operator recycle FFST.

Suppression level
UNSUP.

Blank suppression
Yes.

EPW0507E  procname: LOGID ddname MARKED DISABLED - reason

Explanation
FFST found an error with the message log ddname. reason may be:

I/O ERROR OCCURRED
An I/O error other than an out of space condition occurred.

OPEN FAILED
An OPEN request for ddname failed.

DATA SET HAS INVALID ATTRIBUTES
The data set defined on DD ddname is not defined as fixed, logical record length 80, or block size a multiple of 80.

System action
Processing continues, but FFST will not use the log data set defined on DD ddname until the problem is corrected and an ACTION=ENABLE modify command is issued.

Operator response
Report the message to the system programmer, and issue a modify command with ACTION=ENABLE for ddname when the problem has been corrected.

Programmer response
The response depends on the reason for the failure:

I/O ERROR OCCURRED
An I/O error other than an out of space condition occurred.

OPEN FAILED
An OPEN request for ddname failed.
DATA SET HAS INVALID ATTRIBUTES
The data set defined on DD ddname is not defined as fixed, logical record length 80, or block size a multiple of 80.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0508I proname: LOGID ddname FULL - ATTEMPTING NEXT LOG

Explanation
FFST attempted writing messages to log data set ddname, but it encountered an out-of-space condition. Message log processing is switching to the next enabled log data set.

System action
FFST attempted writing messages to log data set ddname, but it encountered an out-of-space condition. Message log processing is switching to the next enabled log data set.

Operator response
None, unless REUSE is not enabled, and case message EPW0510I will be issued. Also, issue a modify command with ACTION=CLEAR to logid ddname if REUSE is not enabled and when the data in the data set is no longer needed.

Programmer response
If necessary, offload the data in the data set defined on DD ddname to a backup data set and request that the operator clear the data set.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0510I proname: MESSAGE LOG SUSPENDED - REUSE NOT ENABLED, AND LOGID ddname NOT EMPTY

Explanation
A message log switch has taken place, and the REUSE function is disabled, but the data set defined on DD ddname still contains data and has been cleared.

System action
FFST message log processing is temporarily suspended, and will resume when logid ddname has been cleared.

Operator response
Issue a modify command with ACTIO=CLEAR for DD ddname when the data set can be reused.
Determine if the data contained in the data set defined on DD ddname can be erased. If so, request that the operator clear the message log.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

**Explanation**

FFST completed processing the command cm, where cm is one of the following commands:

<table>
<thead>
<tr>
<th>DISAB</th>
<th>ENSA</th>
<th>CLER</th>
<th>DISP</th>
<th>CHAN</th>
<th>RESF</th>
<th>HALT</th>
</tr>
</thead>
</table>

**System action**

Processing continues.

**Operator response**

This message is informational. It indicates that FFST completed processing the specified command.

**Programmer response**

None.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

**Explanation**

The system cancelled the command cm because the necessary resources are unavailable. (For example, FFST may not be able to obtain storage for a command control block.)

**System action**

The system does not execute the command. Other processing continues.

**Operator response**

Wait for the necessary resources to become available and reenter the command. If the message appears again, perform the problem determination action.

**Programmer response**

Correct the problem as indicated in the problem determination output.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

**Explanation**

FFST does not recognize parameter parm. This parameter is not valid on any FFST modify command.

**System action**

The system does not execute the command. Other processing continues.

**Operator response**

Try the command again with the correct parameter.

**Programmer response**

None.

**Suppression level**

UNSUP.
Blank suppression
Yes.

Destination
Console.

**EPW0604E**  
**procname**: REQUIRED PARAMETER **parm** MISSING

**Explanation**
You did not include the required parameter **parm** in the command you entered.

**System action**
The system does not execute the command. Other processing continues.

**Operator response**
Try the command again with the required parameter.

**Programmer response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0605E**  
**procname**: CONFLICTING PARAMETERS ENTERED

**Explanation**
The command contains conflicting parameters.

**System action**
The system does not execute the command. Other processing continues.

**Operator response**
Try the command again without the conflicting parameters. For more information about FFST commands, see "Controlling FFST operation" on page 30.

**Programmer response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0606E**  
**procname**: INVALID VALUE FOR PARAMETER **parm**

**Explanation**
The value you used for parameter **parm** was not one of the allowed values for the command you entered.

**System action**
The system does not execute the command. Other processing continues.

**Operator response**
Try the command again with the correct parameter value.

**Programmer response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0607E**  
**procname**: INVALID SYNTAX - **reason**

**Explanation**
You entered a command using incorrect syntax. FFST could not process the command for one of the following reasons:
PARAMETER parm was entered with a value or values that were enclosed in parentheses. Multiple values are not allowed for this parameter.

Missing Right Parenthesis
Multiple values were coded for a parameter (as indicated with a left parenthesis), but no matching right parenthesis was found.

Missing Comma
Multiple values were coded for a parameter, but no comma separated the right parenthesis from the next parameter.

Value Coded for Parameter parm
No value is allowed for parameter parm, but you coded one.

Missing Value for Parameter parm
FFST requires that parameter parm have a value to be coded with it, but no value was found.

System action
The system does not execute the command. Other processing continues.

Operator response
Try the command again using the correct syntax.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

Explanation
You entered an FFST MODIFY command that failed for one of the following reasons:

APPLID or ProbeID Required with Action
You entered a command with ACTION=action, but this command requires that either APPLID or PROBEID or both be coded.

Vendor Required if FASTService Not Installed
FASTService is not installed, but you did not code the VENDOR parameter. VENDOR implies an FFST product; its absence implies FASTService.

Invalid ProbeID
You coded the PROBEID parameter, but its value was invalid for one of the following reasons:

- The value was less than 3 characters.
- An asterisk (*) was found in an invalid position.
- There were characters following an asterisk.

Unknown Appliance - Use "Define" to Define
You coded the APPLID parameter, but the application was not defined to FFST at the time. If you want to define the application name, use the DEFINE parameter to do so.

Unknown ProbeID Prefix
For a command with ACTION=DISPLAY, you coded PROBEID but not APPLID, and the prefix portion of the probe identifier was not known to FFST. If you know which APPLID is associated with this prefix, code APPLID also.

Vendor Not Allowed
You entered a command and used the VENDOR parameter, but that parameter is not allowed with APPLID=FFST (for an ACTION of ENABLE, DISABLE, or CLEAR) or APPLID=FASTSERV (for any action).

ProbeID Not Allowed with APPLID=FFST or APPLID=FASTSERV
You entered a command with APPLID=FFST or APPLID=FASTSERV, but also coded PROBEID.

Options Not Allowed
You entered a MODIFY command with ACTION=DISPLAY, but also coded OPTIONS.

Define Not Allowed
You entered a MODIFY command with ACTION=DISPLAY, but also coded DEFINE.

Unknown Appliance applid
You entered a MODIFY command with ACTION=DISPLAY and APPLID=applid, but applid is not known to FFST.

Unknown ProbeID probeid
You entered a MODIFY command with ACTION=DISPLAY and PROBEID=probeid, but FFST could not find any probe identifiers that match probeid.

FASTService Not Initialized
You entered a MODIFY command with APPLID=FASTSERV or without the VENDOR parameter, but the FASTService product was not initialized.
LOAD FAILED FOR MODULE *modname*
You specified ACTION=CHANGE and GAEXIT=*modname*, but module *modname* could not be loaded.

generic alert exit does not match current- *gaexit*
You entered ACTION=RESET and the GAEXIT parameter with a name, but the name does not match the current generic alert exit name, *gaexit*.

NO GENERIC ALERT EXIT CURRENTLY LOADED
You entered ACTION=RESET and the GAEXIT parameter, but a generic alert exit was not previously loaded.

NetView receiver id does not match current- *alrcvid*
You entered ACTION=RESET and the ALRCVID parameter with a name, but the name does not match the current NetView receiver identifier, *alrcvid*.

DUMP Volume does not match current - *dumpvol*
You entered ACTION=RESET and the DUMPVOL parameter with a name, but the name does not match the current dump volume, *dumpvol*.

DUMP QUALIFIER DOES NOT MATCH CURRENT- *dumpqual*
You entered ACTION=RESET and the DUMPQUAL parameter with a name, but the name does not match the current qualifier, *dumpqual*.

Command too long
The command entered is too long in length.

Logid value required for action=clear
If you wish to clear a log data set, you must specify a log DD name.

Logid cannot be reuse with action=clear
The REUSE value may only be used on a modify command that specifies ACTION=ENABLE or ACTION=DISABLE.

Ddname not found for *ddname*
You entered a modify command indicating LOGID=*ddname*, but the DD statement for this message log does not exist.

System action
The system does not execute the command. Other processing continues.

Operator response
Determine the reason for the failure, and either enter the command correctly or correct the reason for the failure.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0610I     procname: DISPLAY FOR applprobid applprobname follows:
Explanation
This message is the first in a group of messages that are the result of a MODIFY ACTION=DISPLAY command. Additional messages follow this message, where applprobid is either APPLID or PROBEID and applprobname is the application name or software probe identifier.

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0611I     applprobeid counts probe dump symrc genal symst supdp
Explanation
This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command. This message displays the following fields:
applprobeid
   This field contains either APPLID or PROBEID.

COUNTS
   Total and unique count of software probes.

PROBE
   Status of software probe for the application.

DUMP
   Status of dumps for an application.

SYMRC
   Status of symptom records for an application.

GENAL
   Status of generic alerts for an application.

SYMST
   Status of symptom string message EPW0402I for an application.

SUPDP
   Status of dump suppression for an application.

System action
   Processing continues.

Operator response
   None.

Programmer response
   None.

Suppression level
   UNSUP.

Blank suppression
   No.

Destination
   Console.

EPW0613I

Explanation
   This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command. This message displays as a blank separator line.

System action
   Processing continues.

Operator response
   None.

Programmer response
   None.

Suppression level
   UNSUP.

Blank suppression
   No.

Destination
   Console.

EPW0612I   apprbid totct/prbct stat stat stat
            stat stat stat

Explanition
   This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command. This message displays the following fields:

apprbid
   Application name or software probe identifier.

totct
   Number from 0 to 65535 representing the total number of times software probes have been triggered.

prbct
   Number from 0 to 65535 representing the number of different symptom strings generated for triggered software probes.

stat
   Status of the probe or function. An EN as the status indicates that the probe or function is enabled. A DIS as the status indicates that the probe or function is disabled.

System action
   Processing continues.

Operator response
   None.

Programmer response
   None.

Suppression level
   UNSUP.

Blank suppression
   No.
EPW0614I  PRODUCT NAME VENDOR

Explanation
This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command. This message displays the following fields:

APPLID  
Application name

PRODUCT NAME  
Product name

VENDOR  
Vendor name

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Console.

EPW0615I  applid productname vendorname

Explanation
This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command. This message displays the following fields:

applid  
Application name

productname  
Product name

vendorname  
Vendor name

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Console.

EPW0616I  NO applprobeid TO DISPLAY

Explanation
This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command. This message appears if no application or probe identifiers are known to FFST, where applprobeid is either APPLICATION NAMES or PROBEIDS.

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Console.

EPW0617I  THE FOLLOWING PROBEIDS HAVE TRIPPED AT LEAST ONCE:
Explanation
This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command and is issued if software probe identifiers are available to display.

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0618I     probeid probeid probeid probeid

Explanation
This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command and is issued if software probe identifiers are available to display, where probeid is the software probe identifier.

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW0620I     DUM DESTINATION GA EXIT ALRT RCV MSG LOG /STATUS

Explanation
This message is part of a group of messages that are the result of a MODIFY ACTION=DISPLAY command. This message displays the following fields:

DUMP DESTINATION CMS
User id which will be the receiver of all FFST CMS dumps

GA EXIT
Generic alert exit name

ALRT RCV
NetView alert receiver name

MSG LOG
Software probe message log DD name

STATUS
Software probe message log status

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Console.

EPW0621I     DUMPQUAL DUMPVOL GA EXIT ALRT RCV MSG LOG/LOGREUSE

Explanation
This message is part of a group of messages that is the result of a MODIFY ACTION=DISPLAY command, and acts as a header message for message EPW0622I. This message displays the following fields:
**DUMPQUAL**  
High-level dump data set qualifier

**DUMPVOL**  
Dump data set volume serial number

**GA EXIT**  
Generic alert exit name

**ALRT RCV**  
NetView alert receiver name

**MSG LOG**  
Software probe message log status

**LOGREUSE**  
Status of the REUSE message log function

**System action**  
Processing continues.

**Operator response**  
None.

**Programmer response**  
None.

**Suppression level**  
UNSUP.

**Blank suppression**  
No.

**Destination**  
Console.

---

**EPW0622I**  
dumpqual dumpvol gaextnm alrtcvnm msglogst/reusest

**Explanation**  
This message is part of a group of messages that is the result of a MODIFY ACTION=DISPLAY command. This message displays the following fields:

- **dumpqual**  
  High-level dump data set qualifier

- **dumpvol**  
  Dump data set volume serial number

- **gaextnm**  
  Generic alert exit name

- **alrtcvnm**  
  NetView alert receiver name

- **msglogst**  
  Software probe message log status, where status can be:
  - **ENABLED**  
    FFST message logging function is enabled.
  - **DISABLED**  
    FFST message logging function is disabled.
  - **GETMFFAIL**  
    A storage allocation request had failed for a log control block.
  - **SUSPEND**  
    FFST message logging function is not enabled, and the current log data set is not empty.
  - **NODSAVL**  
    No available log data set DD name was found.

- **reusest**  
  Status of the REUSE message log function, where status can be:
  - **ENABLED**  
    FFST message log REUSE function is enabled.
  - **DISABLED**  
    FFST message log REUSE function is disabled.

**System action**  
Processing continues.

**Operator response**  
None.

**Programmer response**  
None.

**Suppression level**  
UNSUP.

**Blank suppression**  
No.

**Destination**  
Console.

---

**EPW0623I**  
THE FOLLOWING APPLICATIONS ARE STILL ACTIVE TO FFST

**Explanation**  
You issued a MODIFY ACTION=HALT command or a STOP command, and one or more applications still have an active interface to FFST. This message is the first in a group of messages. The messages that follow list the applications that still have an active interface.
**EPW0624I**  APPLID PRODUCT NAME VENDOR ASID

**Explanation**
This message is a part of a group of messages that is the result of a MODIFY ACTION=HALT command or a STOP command. It specifies the name of an application that still has an active interface to FFST, where:

- `applid` is the application name.
- `productname` is the name of the product.
- `vendorid` is the vendor name.
- `asid` is the address space in which the application is active to FFST (MVS only).

**System action**
None.

**Operator response**
None.

**Programmer response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0625I**  applid productname vendor asid

**Explanation**
This message is part of a group of messages that is the result of a MODIFY ACTION=DISPLAY command, and acts as a header for message EPW0627I. This message displays the following fields:

- **MSG LOG**
  A list of DD names used by the message log function

- **STATUS**
  The corresponding status of the message log data set.

**System action**
Processing continues.
EPW0627I  logddnam - logstat

Explanation
This message is part of a group of messages that is the result of a MODIFY ACTION=DISPLAY command. This message displays the following fields:

logddnam
A DD name used for the message log function

logstat
The corresponding status of the message log data set, where status can be:

ENABLED, CURRENT
The message log is enabled, and currently being written to.

ENABLED, AVAILABLE
The message log is enabled, and may be used in the future.

ENABLED, FULL
The message log is enabled, but it has filled up. REUSE is disabled, so it must be cleared before it can be reused.

ENABLED, TO BE CLEARED
The message log is enabled, and a modify command with ACTION=CLEAR has been issued for it. REUSE is disabled, but this log may now be reused.

ENABLED, USED
The message log is enabled, and is not the current log, but it has been written to in the past. REUSE is enabled, so this data set may be reused in the future.

DISABLED BY OPERATOR
The operator has issued a modify command with ACTION=DISABLE for this message log.

DISABLED - I/O ERROR
An I/O error has been encountered for this message log.

DISABLED - INVALID ATTRIBUTES
The data set defined for this DD statement was allocated with attributes that are not acceptable for a log data set.

SUSPENDED - DATA SET NOT EMPTY
This message log is the current data set, but REUSE is DISABLED, and the data set is not empty. Message logging is temporarily suspended until the log data set is cleared.

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Console.

EPW0701I  END OF MESSAGE GROUP

Explanation
This message marks the end of a message group. For more information, see the information for the other messages in the group.

System action
Processing continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.
Blank suppression
Yes.

Destination
Console.

**EPW0702E**  INVALID REPLY

**Explanation**
You replied to an outstanding FFST message, but your response was invalid.

**System action**
The response is ignored.

**Operator response**
Look for another iteration of the message for which your response was invalid, and reply to that message with a valid response.

**Programmer response**
The FFST/FFDC process should have been invoked. Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW1001I**  FFSTCICS: EPWTRUE IS NOW ACTIVE

**Explanation**
FFST/CICS initialized a task related user exit (TRUE).

**System action**
Processing continues.

**Operator response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW0799I**  proclname: MESSAGE nnnn ISSUED BUT NOT FOUND

**Explanation**
FFST issues this message whenever a message unknown to FFST is issued. The nnnn variable represents the number of the unidentified message.

**System action**
Processing continues.

**Operator response**
Report the message to the system programmer.

**Programmer response**
The FFST/FFDC process should have been invoked. Gather the required documentation and contact the IBM support center.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW1002E**  LOAD OF EPWCSTUB FAILED, LOAD RESPONSE WAS **abendcode**

**Explanation**
CICS could not load the EPWCSTUB module. The operating system returns abendcode in register 1 and retcode in register 15 after the load fails.

**System action**
Processing continues.
Operator response
Report the message to the system programmer.

Programmer response
Verify that this module is available for CICS to load.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW1003E  cicscmd COMMAND FOR truename FAILED - reason

Explanation
CICS could not initialize a task related user exit (TRUE). The associated CICS command was cicscmd, the TRUE was truename, and the reason for the failure was reason. The CICS transaction issued one of the following commands:

| DISABLE | ENABLE | EXTRACT | LOAD |

The failure occurred for one of the following reasons:

ALREADY ENABLED
CICS already enabled the program identified by truename.

CURRENTLY INVOKED BY ANOTHER TASK
Another CICS task is currently using the program identified as truename.

EPWTRUE HAS NO WORK AREA
CICS did not allocate any save area for EPWTRUE before truename issued the CICS command.

EPWTRUE IS NOT ENABLED
CICS did not enable EPWTRUE before truename issued the CICS command.

EXIT=YES NOT SPECIFIED IN DFHSIT
The DFHSIT being used by CICS does not specify EXIT=YES.

MODULE NOT FOUND OR DISABLED
truename is not in the primary program operator interface task (PPT) or the load library, or its PPT is disabled.

PROGRAM IS NOT ENABLED
CICS did not enable truename before a task issued the CICS command.

PROGRAM NOT AUTHORIZED
CICS failed truename during CICS resource security checking.

UNKNOWN EIBRCODE eibrcode
CICS did not recognize the external interrupt block (EIB) return code eibrcode.

UNKNOWN EIBRESP eibresp
CICS did not recognize the EIB response code eibresp.

System action
Processing continues.

Operator response
Report the message to the system programmer.

Programmer response
The response depends on the reason for the failure:

ALREADY ENABLED
Gather the required documentation and contact the IBM support center.

CURRENTLY INVOKED BY ANOTHER TASK
Gather the required documentation and contact the IBM support center.

EPWTRUE HAS NO WORK AREA
Gather the required documentation and contact the IBM support center.

EPWTRUE IS NOT ENABLED
Gather the required documentation and contact the IBM support center.

EXIT=YES NOT SPECIFIED IN DFHSIT
Specify EXIT=YES in the scanner interface trace (SIT) to be used by CICS.

MODULE NOT FOUND OR DISABLED
Verify that truename has been defined in the PPT, CICS can locate the program, and its PPT is enabled.

PROGRAM IS NOT ENABLED
Gather the required documentation and contact the IBM support center.

PROGRAM NOT AUTHORIZED
Verify that truename can be loaded by CICS.

UNKNOWN EIBRCODE eibrcode
Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about eibrcode.
**UNKNOWN EIBRESP eibresp**

Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about eibresp.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW1011I FFSTCICS: EPWTRUE IS NOT ACTIVE**

**Explanation**

CICS is closing the task related user exit (TRUE). CICS returns control to the program that issued the probe statement.

**System action**

Processing continues.

**Operator response**

Report the message to the system programmer.

**Programmer response**

Verify that the FFST/CICS function is installed correctly.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW1012E EPWTRUE CALL WAS NOT FROM AN APPLICATION PROGRAM**

**Explanation**

An application was not the requester for FFST/CICS. CICS returns control to the program that issued the request.

**System action**

Processing continues.

**Operator response**

Report the message to the system programmer.

**Programmer response**

Verify that the FFST/CICS function is installed correctly.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW1013E UNEXPECTED EPWTRUE CALL**

**Explanation**

The request for FFST/CICS was not for a Task Related User Exit (TRUE). CICS returns control to the program that issued the request.

**System action**

Processing continues.

**Operator response**

Report the message to the system programmer.

**Programmer response**

Verify that the FFST/CICS function is installed correctly.

**Suppression level**

UNSUP.

**Blank suppression**

Yes.

**Destination**

Console.

---

**EPW1014E CICS WAIT EVENT FAILED - reason**
The operating system failed the event for which CICS was waiting for one of the following reasons:

**EIBRESP WAS eibresp**
The EIB response code was eibresp.

**INVALID INTERVAL CONTROL COMMAND**
The application issued an invalid interval command.

**TIME HAS EXPIRED**
FFST did not return control to CICS within one minute.

**System action**
Processing continues.

**Operator response**
Report the message to the system programmer.

**Programmer response**
The response depends on the reason for the failure:

**EIBRESP WAS eibresp**
Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about eibresp.

**INVALID INTERVAL CONTROL COMMAND**
Gather the required documentation and contact the IBM support center.

**TIME HAS EXPIRED**
Determine why FFST is not responding to CICS requests

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**Explanation**
CICS could not attach EPWCSTUB, where respcde is the return code from the attach.

**System action**
Processing continues.

**Operator response**
Report the message to the system programmer.

**Programmer response**
Use the respcde value to determine why the attach failed. For more information about attach return codes, refer to either MVS/ESA Application Development Reference (GC28-1647) or OS/390® MVS Assembler Services Reference (GC28-1910).

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.
Explanation
During termination of the FFST/CICS Task Related User Exit (TRUE), CICS could not delete EPWCSTUB.

System action
Processing continues.

Operator response
Report the message to the system programmer.

Programmer response
The response depends on the reason for the failure:

**EIBRESP WAS eibresp**
Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about eibresp.

**INPUT/OUTPUT ERROR**
Check to see whether the temporary storage data set is full.

**INVALID INTERVAL CONTROL COMMAND**
Gather the required documentation and contact the IBM support center.

**PROGRAM NOT AUTHORIZED**
Verify that CICS can load the FFST program EPWTRUET.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

**EPW1023E**

| FFSTCICS: START FOR TRANSID EPWT FAILED - reason |

**Explanation**
CICS could not complete the CICS START command for transaction EPWT for one of the following reasons:

**EIBRESP WAS eibresp**
The EIB response code was eibresp.

**INPUT/OUTPUT ERROR**
An I/O error occurred during the execution of the CICS START command.

**INVALID INTERVAL CONTROL COMMAND**
The application issued an invalid interval control command.

**PROGRAM NOT AUTHORIZED**
CICS failed the FFST program EPWTRUET during CICS resource security checking.

System action
Processing continues.

**EPW1024E**

| ciscmd COMMAND FOR truename FAILED - reason |

**Explanation**
CICS could not successfully terminate a Task Related User Exit (TRUE). The task issued the CICS command ciscmd for the TRUE truename, where ciscmd is one of the following values:

| DISABLE |
| EXTRACT |

The command failed for one of the following reasons:

**CURRENTLY INVOKED BY ANOTHER TASK**
truename is currently being used by another CICS task.

**EPWTRUE HAS NO WORK AREA**
CICS could not allocate the work area for EPWTRUE when truename issued the CICS command.
EXIT=YES NOT SPECIFIED IN DFHSIT
The DFHSIT being used by CICS does not have EXIT=YES specified.

MODULE NOT FOUND OR DISABLED
truename is not in the PPT or the load library, or CICS disabled its PPT entry.

PROGRAM IS NOT ENABLED
CICS did not enable truename before the task issued the CICS command.

UNKNOWN EIBRCODE eibrcode
FFST did not recognize the EIB return code eibrcode.

UNKNOWN EIBRESP eibresp
FFST did not recognize the EIB response code eibresp.

System action
Processing continues.

Operator response
Report the message to the system programmer.

Programmer response
The response depends on the reason for the failure:

CURRENTLY INVOKED BY ANOTHER TASK
Gather the required documentation and contact the IBM support center.

EPWTRUE HAS NO WORK AREA
Gather the required documentation and contact the IBM support center.

EXIT=YES NOT SPECIFIED IN DFHSIT
Specify EXIT=YES in the SIT to be used by CICS.

MODULE NOT FOUND OR DISABLED
Verify that you defined truename in the PPT, CICS can locate the program, and CICS did not disable its PPT.

PROGRAM IS NOT ENABLED
Verify that CICS did not invoke the transaction EPWT.

UNKNOWN EIBRCODE eibrcode
Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about eibrcode.

UNKNOWN EIBRESP eibresp
Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about eibresp.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW1031I FFSTCICS: GLOBAL USER EXIT exitname IS NOW ACTIVE

Explanation
CICS initialized a global user exit. The exit is exitname.

System action
Processing continues.

Operator response
None

Programmer response
None

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW1032I FFSTCICS: START FOR TRANSID GUET FAILED - reason

Explanation
CICS could not complete the CICS start command for transaction GUET (Global User Exit Transaction) for one of the following reasons:

EIBRESP WAS eibresp
The EIB response code was eibresp.

INPUT/OUTPUT ERROR
An I/O error occurred during the execution of the CICS start command.

INVALID INTERVAL CONTROL COMMAND
The application issued an invalid interval control command.
PROGRAM NOT AUTHORIZED
CICS failed the FFST program EPWCGUET during CICS resource security checking.

System action
Processing continues.

Operator response
Report the message to the system programmer.

Programmer response
The response depends on the reason for the failure:

EIBRESP WAS elbresp
Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about elbresp.

INPUT/OUTPUT ERROR
Check to see if the temporary storage data set is full.

INVALID INTERVAL CONTROL COMMAND
Gather the required documentation and contact the IBM support center.

PROGRAM NOT AUTHORIZED
Verify that CICS can load the FFST program EPWCGUET.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW1034I  cicscmd COMMAND FOR exitname FAILED - reason

Explanation
CICS could not initialize or terminate a global user exit. The associated CICS command was cicscmd, the global user exit was exitname, and the reason for the failure was reason. The CICS transaction issued one of the following commands:

<table>
<thead>
<tr>
<th>DISABLI</th>
<th>ENABLI</th>
<th>EXTRACT</th>
<th>LOAD</th>
</tr>
</thead>
</table>

The failure occurred for one of the following reasons:

ALREADY ENABLED
CICS already enabled the program identified by exitname.

CURRENTLY INVOKED BY ANOTHER TASK
Another CICS task is currently using the program identified by exitname.

EPWCGUE HAS NO WORK AREA
CICS did not allocate any save area for EWPCGUE before exitname issued the CICS command.

EPWCGUE IS NOT ENABLED
CICS did not enable EPWCGUE before exitname issued the CICS command.

EXIT=YES NOT SPECIFIED IN DFHSIT
The DFHSIT being used by CICS does not specify EXIT=YES.

EXITID IS NOT VALID
exitname is invalid.

GALENGTH VALID ON FIRST ENABLE
GALENGTH parameter is valid only on the first enable command.

MODULE NOT FOUND OR DISABLED
exitname is not in the PPT or the load library, or its PPT is disabled.

PROGRAM ALREADY ACTIVE
EPWCGUE is already active.
PROGRAM IS NOT ENABLED
CICS did not enable exitname before a task issued the CICS command.

PROGRAM NOT AUTHORIZED
CICS failed exitname during CICS resource security checking.

UNKNOWN EIBCODE eibcode
CICS did not recognize the EIB return code eibcode.

UNKNOWN EIBRESP eibresp
CICS did not recognize the EIB response code eibresp.

System action
Processing continues.

Operator response
Report the message to the system programmer.

Programmer response
The response depends on the reason for the failure.

ALREADY ENABLED
Gather the required documentation and contact the IBM support center.

CURRENTLY INVOKED BY ANOTHER TASK
Gather the required documentation and contact the IBM support center.

EPWCGUE HAS NO WORK AREA
Gather the required documentation and contact the IBM support center.

EPWCGUE IS NOT ENABLED
Gather the required documentation and contact the IBM support center.

EXIT=YES NOT SPECIFIED IN DFHSIT
Specify EXIT=YES in the SIT to be used by CICS.

EXITID IS NOT VALID
Gather the required documentation and contact the IBM support center.

GALENGTH VALID ON FIRST ENABLE
Gather the required documentation and contact the IBM support center.

MODULE NOT FOUND OR DISABLED
Verify that EPWCGUE has been defined in the PPT, CICS can locate the program, and its PPT is enabled.

PROGRAM ALREADY ACTIVE
Gather the required documentation and contact the IBM support center.

PROGRAM IS NOT ENABLED
Gather the required documentation and contact the IBM support center.

PROGRAM NOT AUTHORIZED
Verify the EPWCGUE can be loaded by CICS.

UNKNOWN EIBCODE eibcode
Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about eibcode.

UNKNOWN EIBRESP eibresp
Gather the required documentation and contact the IBM support center. Refer to the CICS problem determination manual for your level of CICS for information about eibresp.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.

EPW1035I FFSTCICS: EPWCGUE IS NOT ACTIVE

Explanation
CICS is closing the global user exit. CICS returns control to the program that issued the request.

System action
Processing continues.

Operator response
None

Programmer response
None

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console.
**EPW1036I**  **FFSTCICS: EPWCGUE CALL WAS NOT FOR A GLOBAL USER EXIT**

**Explanation**
The call to EPWCGUE was not a global user exit type request. CICS returns control to the program that issued the request.

**System action**
Processing continues.

**Operator response**
Report the message to the system programmer.

**Programmer response**
Verify that the FFST/CICS function is installed correctly.

**Suppression level**
UNSUP.

**Blank suppression**
Yes.

**Destination**
Console.

**EPW9000I**  **prgname: RETURN CODE WAS retcode - rctext**

**Explanation**
During the execution of the installation verification procedure (IVP), an FFST-invoked function did not complete successfully, where *retcode* is the return code received from the FFST function and *rctext* is the text associated with the return code. This message follows messages EPW9002I, EPW9004I, and EPW9006I. *rctext* can be one of the following:

**Note:** In the following text, the term 'the IVP probe' refers to the probe id in message EPW9004I and the term 'the config' refers to the configuration load module used for the IVP (EPWIVCE for FFST/MVS).

**PROBE HAS BEEN DISABLED BY THE OPERATOR**
The operator has issued an FFST modify command to disable the IVP probe.

**INVALID PARAMETERS - FIRST CHARACTER NOT * **
An invalid identifier was found on the control string for the IVP probe.

**INVALID PARAMETERS - DST NAME LONGER THAN 8**
The Data Structure Table name specified on the IVP probe was too long.

**INVALID PARAMETERS - PROBEID LONGER THAN 8**
The probe identifier for the IVP probe was too long.

**INVALID PARAMETERS - COMMA DOES NOT FOLLOW FLAGS**
An invalid delimiter was found in the control string for the IVP probe.

**INVALID PARAMETERS - COUNT LONGER THAN 3**
The count value for the number of parameters passed for the IVP probe was too long.

**INVALID PARAMETERS - CRC VALUE LONGER THAN 7**
The correlator value used for the IVP probe was too long.

**FFST INITIALIZATION OF PRODUCT NOT COMPLETE**
The FFST initialization of FFSTV1R2 has not yet completed.

**NO WORK AREA AVAILABLE**
A GETMAIN of a 16K work area failed when attempting to trip the IVP probe.

**SETFRR FAILED**
For FFST/MVS, the recovery environment setup failed when attempting to trip the IVP probe.

**SPECIFIED DST NOT FOUND**
Data Structure Table EPWIVT could not be loaded.

**SPECIFIED DST INCOMPATIBLE WITH PRESENT RELEASE**
Data Structure table EPWIVT does not indicate a level of release 2.

**INVALID DST SPECIFIED**
Data Structure Table EPWIVT has been previously marked invalid.

**PROBE ENTRY NOT FOUND IN SPECIFIED DST**
Data Structure Table EPWIVT did not contain an entry for the IVP probe.

**PROBE INCOMPATIBLE WITH ENTRY IN DST**
The level of the IVP probe is not the same level in module EPWIVP as it is in Data Structure Table EPWIVT.

**ENTRY IN SPECIFIED DST IS NOT A PROBE**
Enter name of the IVP probe was found in Data Structure Table EPWIVT, but it is not a probe entry.

**UNEXPECTED ABEND OCCURRED**
Some unexpected ABEND occurred in FFST code when attempting to trip the IVP probe.
MISMATCHING SHARED CONFIGURATION FOUND
For FFST/MVS, the level of configuration load module EPWIVCE does not match the level of the main FFST configuration load module, EPWTRNCF.

MISMATCHING CONFIGURATION FOUND IN ADDRESS SPACE
Another config with a prefix of EPW was already loaded.

CONFIGURATION LOAD MODULE LINK-EDIT AS REENTRANT
The config has an attribute of reentrant.

NOT AUTHORIZED TO USE SHARED STORAGE
Module EPWIVP is requesting a service that requires shared storage, but it should not be.

INVALID IDENTIFIER IN DST
Data Structure Table EPWIVT contains an invalid identifier in its first byte.

DST IS SHARED, CONFIGURATION IS NOT
DST EPWIVT is marked as SHARED, but the config is not marked as SHARED.

ALLOCATE FAILED FOR TEMPORARY AREA
The IVP probe failed because a GETMAIN for a temporary area failed.

PROBE DISABLED - INVOKED TOO MANY TIMES
The IVP probe has been tripped more than 10 times in 10 minutes. It has been automatically disabled, since it exceeded criteria specified in the config.

FFST SUBSYSTEM NOT DEFINED
The subsystem name of FFST has not been defined to MVS.

FFST STUB ADDRESS IS ZERO
The probe stub module, EPWSTUB, is not link-edited with load module EPWIVP.

FFST INTERFACE MODULE NOT IN ACCESSIBLE LIBRARY
For FFST/MVS, module EPWPINIT, EPWPITSK, EPWPTERM or EPWSTUB cannot be found in an accessible library.

LOAD OF CONFIGURATION MODULE FAILED
The config cannot be found in an accessible library.

FFST ADDRESS SPACE IS NOT ACTIVE
For FFST/MVS, the task is not started.

ALLOCATE OF TEMPORARY AREA FAILED
FFST interface initialization or termination failed because a GETMAIN for a temporary area failed.

PREVIOUS INIT REQUEST HAD FAILED
FFST interface initialization failed because a previous interface initialization request failed.

PRODUCT NOT INITIALIZED TO FFST
FFST interface termination failed because the interface initialization did not complete.

CONFIGURATION LOAD MODULE NOT LINKED AS REUSABLE
The config does not have REUSE as a link-edit attribute.

UNEXPECTED RETURN CODE RECEIVED
Return code retcode is not recognized by module EPWIVP.

System action
System processing continues.

Operator response
Give the FFST IVP output to the system programmer.

Programmer response
The response depends on what rctext indicates:

PROBE HAS BEEN DISABLED BY THE OPERATOR
Have the operator issue an FFST modify command to enable the IVP probe again and rerun the installation verification program.

INVALID PARAMETERS - MORE THAN 25 CHARACTERS
Contact the IBM support center.

INVALID PARAMETERS - FIRST CHARACTER NOT *
Contact the IBM support center.

INVALID PARAMETERS - DST NAME LONGER THAN 8
Contact the IBM support center.

INVALID PARAMETERS - PROBEID LONGER THAN 8
Contact the IBM support center.

INVALID PARAMETERS - COMMA DOES NOT FOLLOW FLAGS
Contact the IBM support center.

INVALID PARAMETERS - COUNT LONGER THAN 3
Contact the IBM support center.

INVALID PARAMETERS - CRC VALUE LONGER THAN 7
Contact the IBM support center.

FFST INITIALIZATION OF PRODUCT NOT COMPLETE
Contact the IBM support center.

NO WORK AREA AVAILABLE
Make sure there is enough REGION size (FFST/ MVS) available to the installation verification program. If there appears to be, contact the IBM support center.

SETFRR FAILED
Contact the IBM support center.
ESTAE SETUP FAILED
Contact the IBM support center.

SPECIFIED DST NOT FOUND
Verify that module EPWIVT is in load library SEPWMOD3.

SPECIFIED DST INCOMPATIBLE WITH PRESENT RELEASE
Contact the IBM support center.

INVALID DST SPECIFIED
Look for a previous message EPW0253I for EPWIVT, and contact the IBM support center.

PROBE ENTRY NOT FOUND IN SPECIFIED DST
Contact the IBM support center.

PROBE INCOMPATIBLE WITH ENTRY IN DST
Contact the IBM support center.

ENTRY IN SPECIFIED DST IS NOT A PROBE
Contact the IBM support center.

UNEXPECTED ABEND OCCURRED
Look for messages EPW0304E, EPW0305E, EPW0306E and EPW0307E and then contact the IBM support center.

MISMATCHING SHARED CONFIGURATION FOUND
Make sure that the maintenance level of module EPWIVCE matches the level of EPWTRNCF.

MISMATCHING CONFIGURATION FOUND IN ADDRESS SPACE
Contact the IBM support center.

CONFIGURATION LOAD MODULE LINK EDIT AS REENTRANT
Be sure that the config has been installed correctly, with the attribute of REUSE.

NOT AUTHORIZED TO USE SHARED STORAGE
Contact the IBM support center.

INVALID IDENTIFIER IN DST
Contact the IBM support center.

DST IS SHARED, CONFIGURATION IS NOT
Contact the IBM support center.

ALLOCATE FAILED FOR TEMPORARY AREA
Be sure there is enough REGION size (FFST/MVS) available for the installation verification program.

PROBE DISABLED - INVOKED TOO MANY TIMES
Have the operator issue an FFST modify command to enable the IVP probe and rerun the installation verification program.

FFST SUBSYSTEM NOT DEFINED
Follow the installation steps in the FFST program directory before running the installation verification program.

FFST SHARED SEGMENT NOT DEFINED
Follow the installation steps in the FFST program directory before running the installation verification program.

FFST STUB ADDRESS IS ZERO
Look at the latest results of applying maintenance to module EPWIVP to be sure that EPWSTUB is included in the load module.

FFST INTERFACE MODULE NOT IN ACCESSIBLE LIBRARY
Be sure that modules EPWPINIT, EPWPITSK and EPWPTERM are in a LINKLIST library, and that EPWSTUB is in fixed LPA.

LOAD OF CONFIGURATION MODULE FAILED
Make sure that the config is in load library SEPWMOD3.

FFST ADDRESS SPACE IS NOT ACTIVE
Have the operator issue a START command for FFST before running the installation verification program.

FFST APPLICATION IS NOT ACTIVE
Have the operator issue a START command for FFST in the FFST virtual machine before running the installation verification program.

ALLOCATE OF TEMPORARY AREA FAILED
Be sure there is enough REGION size (FFST/MVS) available for the installation verification program.

PREVIOUS INIT REQUEST HAD FAILED
Look at the previous failure of the interface initialization for the IVP.

PRODUCT NOT INITIALIZED TO FFST
Look at the interface initialization failure.

CONFIGURATION LOAD MODULE NOT LINKED AS REUSABLE
Be sure that the config has been installed correctly, with the attribute of REUSE.

UNEXPECTED RETURN CODE RECEIVED
Contact the IBM support center.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, terminal.

EPW9001I progmname: CONTROL INIT SUCCESSFUL
Explanation
During the execution of the installation verification procedure (IVP), the FFST control initialization received a return code of 0. This does not necessarily mean, however, that the initialization function completed successfully. Message EPW0250I indicates successful completion.

System action
System processing continues.

Operator response
Give the FFST IVP output to the system programmer.

Programmer response
Verify that the FFST IVP completed successfully. Review the messages issued and ensure that message EPW9000I was not issued.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, terminal.

EPW9002I prgmname: CONTROL INIT NOT SUCCESSFUL

Explanation
During the execution of the installation verification procedure (IVP), the FFST control initialization function did not complete successfully. Message EPW9000I follows.

System action
System processing continues.

Operator response
Give the FFST IVP output to the system programmer.

Programmer response
Check the return code in EPW9000I to determine why the FFST control initialization function failed. The return code might indicate an FFST installation failure.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, terminal.

EPW9003I prgmname: PROBE probeid SUCCESSFUL

Explanation
During the execution of the installation verification procedure (IVP), a probe was issued and completed successfully, where probeid is the probe identifier.

System action
System processing continues.

Operator response
Give the FFST IVP output to the system programmer.

Programmer response
Verify that the FFST IVP completed successfully. Review the messages issued and ensure that message EPW9000I was not issued.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, terminal.

EPW9004I prgmname: PROBE probeid NOT SUCCESSFUL

Explanation
During the execution of the installation verification procedure (IVP), a probe was issued and did not complete successfully, where probeid is the probe identifier. Message EPW9000I follows.

System action
System processing continues.
Operator response
Give the FFST IVP output to the system programmer.

Programmer response
Check the return code in EPW9000I to determine why the probe failed. The return code might indicate an FFST installation failure.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, terminal.

EPW9005I prgmname: CONTROL TERM SUCCESSFUL
Explanation
During the execution of the installation verification procedure (IVP), the FFST control termination function completed successfully.

System action
System processing continues.

Operator response
Give the FFST IVP output to the system programmer.

Programmer response
Check the return code in EPW9000I to determine why the FFST control termination function failed. The return code might indicate an FFST installation failure.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, terminal.

EPW9010I prgmname: PROGRAM COMPLETE
Explanation
The execution of the installation verification procedure (IVP) is complete.

System action
System processing continues.

Operator response
Give the FFST IVP output to the system programmer.

Programmer response
Verify that the FFST IVP completed successfully. Review the messages issued and ensure that message EPW9000I was not issued.

Suppression level
UNSUP.

Blank suppression
Yes.

Destination
Console, terminal.
DESTINATION
Console, terminal.

EPW9501I PRODUCT NAME: prodname

Explanation
prodname is the name of the product or application that triggered the software probe.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal, output data set.

EPW9502I type PROGRAM

Explanation
The product or application that triggered the software probe is a type program, where type is either IBM or NON-IBM.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal, output data set.

EPW9503I COMPONENT/PROGRAM ID: compid, LEVEL: level

Explanation
The component or program identifier of the product or application that triggered the software probe is compid, and its level is level.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal, output data set.

EPW9504I TYPE OF FAILURE: type

Explanation
The triggered software probe is defined with a category of type.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the SYMPTOM STRING DATA option is N.
Blank suppression
No.

Destination
Terminal, output data set.

EPW9505I PROBE PRIMARY SYMPTOM STRING:

Explanation
The primary symptom string specified by the triggered software probe immediately follows this message.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the SYMPTOM STRING DATA option is N.

Blank suppression
No.

Destination
Terminal, output data set.

EPW9506I PROBE SECONDARY SYMPTOM STRING:

Explanation
The secondary symptom string specified by the triggered software probe immediately follows this message.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the SYMPTOM STRING DATA option is N.

Blank suppression
No.

Destination
Terminal, output data set.

EPW9507I REGISTER SECONDARY SYMPTOM STRING:

Explanation
The secondary symptom string built by FFST that contains the values of all general registers at the time the software probe was triggered immediately follows this message.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the SYMPTOM STRING DATA option is N.

Blank suppression
No.

Destination
Terminal, output data set.

EPW9508I DATA COLLECTION WORK AREA:

Explanation
The 16KB work area used to process the triggered software probe immediately follows this message.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.
**Programmer response**
None.

**Suppression level**
Suppressed when the FFST WORK AREA option is N.

**Blank suppression**
No.

**Destination**
Terminal, output data set.

### EPW9509I  \text{dsttype} DATA STRUCTURE TABLE:

**Explanation**
A DST used during processing of the triggered software probe immediately follows this message, where \text{dsttype} is either SPECIFIED or DEFAULT.

**System action**
Dump formatting continues.

**Operator response**
None.

**Programmer response**
None.

**Suppression level**
Suppressed when the DATA STRUCTURE TABLE(S) option is N.

**Blank suppression**
No.

**Destination**
Terminal, output data set.

### EPW9510I  AREA AROUND REGISTER regnum - regval:

**Explanation**
An area of up to 400 hexadecimal bytes before and after general register \text{regnum} which contains the value \text{regval} immediately follows this message. If the software probe was triggered in AR mode under MVS/ESA, the general register along with its corresponding access register are used to access the storage.

**System action**
Dump formatting continues.

**Operator response**
None.

**Programmer response**
None.

**Suppression level**
Suppressed when the AREA AROUND REGISTERS option is N.

**Blank suppression**
No.

**Destination**
Terminal, output data set.

### EPW9511I  NUMBER OF DATA STRUCTURES OF THIS TYPE EXCEEDED MAX

**Explanation**
The value specified as the maximum count on the definition of the data structure being formatted was exceeded. This data structure was dumped more times than the definition allows.

**System action**
Dump formatting continues.

**Operator response**
None.

**Programmer response**
None.

**Suppression level**
Suppressed when the DATA STRUCTURES option is N.

**Blank suppression**
No.

**Destination**
Terminal, output data set.

### EPW9512I  EXIT ROUTINE CALLED FOR NEXT FFST operations

141
<table>
<thead>
<tr>
<th>Explanation</th>
<th>The DST exit routine specified on the data structure definition was called for processing of the next data structure in a chain.</th>
</tr>
</thead>
<tbody>
<tr>
<td>System action</td>
<td>Dump formatting continues.</td>
</tr>
<tr>
<td>Operator response</td>
<td>None.</td>
</tr>
<tr>
<td>Programmer response</td>
<td>None.</td>
</tr>
<tr>
<td>Suppression level</td>
<td>Suppressed when the DATA STRUCTURES option is N.</td>
</tr>
<tr>
<td>Blank suppression</td>
<td>No.</td>
</tr>
<tr>
<td>Destination</td>
<td>Terminal, output data set.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EPW9513I</th>
<th>EXIT ROUTINE CALLED FOR REF DATA STRUCTURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation</td>
<td>The DST exit routine specified on the data structure definition was called for processing of a referenced data structure.</td>
</tr>
<tr>
<td>System action</td>
<td>Dump formatting continues.</td>
</tr>
<tr>
<td>Operator response</td>
<td>None.</td>
</tr>
<tr>
<td>Programmer response</td>
<td>None.</td>
</tr>
<tr>
<td>Suppression level</td>
<td>Suppressed when the DATA STRUCTURES option is N.</td>
</tr>
<tr>
<td>Blank suppression</td>
<td>No.</td>
</tr>
<tr>
<td>Destination</td>
<td>Terminal, output data set.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EPW9514I</th>
<th>EXIT ROUTINE CALLED FOR DATA STRUCTURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation</td>
<td>The DST exit routine specified on the data structure definition was called on initial processing of the data structure.</td>
</tr>
<tr>
<td>System action</td>
<td>Dump formatting continues.</td>
</tr>
<tr>
<td>Operator response</td>
<td>None.</td>
</tr>
<tr>
<td>Programmer response</td>
<td>None.</td>
</tr>
<tr>
<td>Suppression level</td>
<td>Suppressed when the DATA STRUCTURES option is N.</td>
</tr>
<tr>
<td>Blank suppression</td>
<td>No.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EPW9515I</th>
<th>ID CHECKING FAILED FOR DATA STRUCTURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation</td>
<td>When dumping the data structure, the field specified for identifier checking contained a different value than the one specified in the DST. As a result, FFST dumped up to 800 hexadecimal bytes beginning at the data structure address, and no further REF and NEXT processing occurred.</td>
</tr>
<tr>
<td>System action</td>
<td>Dump formatting continues.</td>
</tr>
<tr>
<td>Operator response</td>
<td>None.</td>
</tr>
<tr>
<td>Programmer response</td>
<td>None.</td>
</tr>
<tr>
<td>Suppression level</td>
<td>Suppressed when the DATA STRUCTURES option is N.</td>
</tr>
<tr>
<td>Blank suppression</td>
<td>No.</td>
</tr>
</tbody>
</table>

| Destination | Terminal, output data set. |
Blank suppression
No.

Destination
Terminal, output data set.

EPW9516I LENGTH WAS FOUND TO BE ZERO

Explanation
When dumping the data structure, FFST determined that the length of the data structure was zero. As a result, FFST dumped up to 800 hexadecimal bytes beginning at the data structure address, and no further REF and NEXT processing occurred.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the DATA STRUCTURES option is N.

Blank suppression
No.

EPW9517I STORAGE UNAVAILABLE DUE TO INACCESSIBILITY

Explanation
When dumping the data structure, FFST determined that, for one of the following reasons, all or part of the storage was not accessible to the program that triggered the software probe:

• The storage in the primary address space is not allocated.
• An alternate ASID was specified, but the program did not have access to it.
• An alternate dataspace was specified, but the program did not have access to it.
• An invalid ASID or dataspace was specified.

• The program was not authorized to use an alternate ASID.
As a result, no storage is dumped, and because this error was encountered, no further REF and NEXT processing occurs. Message EPW9518I follows this message.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the DATA STRUCTURES option is N.

Blank suppression
No.

EPW9518I INVALID ADDRESS WAS address

Explanation
This message follows message EPW9517I. The storage at address address was inaccessible at the time the software probe was triggered.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the DATA STRUCTURES option is N.

Blank suppression
No.
EPW9519I WARNING: warn

Explanation
The exception condition warn occurred while FFST was processing the probe, where warn is one of the following values:

- INVALID POINTER FOUND WHEN BUILDING PARMLIST
- INVALID POINTER FOUND WHEN BUILDING SYMREC
- DATA STRUCTURE NAME NOT FOUND IN TABLE
- RAN OUT OF ROOM DURING DATA COLLECTION
- INVALID POINTER FOUND DURING DATA COLLECTION
- LOAD OF EXIT ROUTINE FAILED
- GENERIC ALERT CAUSE NAME NOT FOUND
- GENERIC ALERT DESCRIPTOR NAME NOT FOUND

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the FFST WORK AREA option is N.

Blank suppression
No.

Destination
Terminal, output data set.

EPW9520I DATA STRUCTURE DEFINED IN DEFAULT DST

Explanation
The data structure was defined in the default DST, not in the specified DST.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

EPW9521I DUMP DATA SET NAME=dsname

Explanation
The original data set to which the dump was written is named dsname.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
Suppressed when the DATA STRUCTURES option is N.

Blank suppression
No.

Destination
Terminal, output data set.

EPW9522I TITLE FROM DUMP=title

Explanation
The dump’s title is title. This title was received from one of the following:

- The abstract from the software probe
- The default abstract from the configuration table or DST
- A default abstract built by FFST

System action
Dump formatting continues.

Operator response
None.
<table>
<thead>
<tr>
<th>EPW9523I</th>
<th>DATE FROM DUMP=<em>date</em>, TIME FROM DUMP=<em>time</em>.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation</strong></td>
<td>The dump data set was written to on date at time.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EPW9524I</th>
<th>NO DATA STRUCTURE TITLE - ENTRY NOT FOUND IN DST</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation</strong></td>
<td>The data structure may have been dumped, but the specified entry was not found in either the specified DST or the default DST.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EPW9550E</th>
<th>UNEXPECTED RETURN CODE retcode RECEIVED</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation</strong></td>
<td>The dump formatter did not have an error message set up for the error condition that was encountered. A return code of \textit{retcode} was received for the operation that failed.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EPW9551E</th>
<th>RETURN CODE retcode ATTEMPTING TO ALLOCATE dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation</strong></td>
<td>The dump formatter stops processing and redispers the last panel displayed.</td>
</tr>
</tbody>
</table>

**System action**
- Dump formatting continues.

**Operator response**
- None.

**Programmer response**
- None.

**Suppression level**
- UNSUP.

**Blank suppression**
- No.

**Destination**
- Terminal, output data set.
**Explanation**
The dump formatter attempted to allocate a dataset, where dataset is the name of a data set or TEMP DUMP DATASET. The ALLOCATE request failed with a return code of retcode.

**System action**
The dump formatter stops processing and redisplays the last panel displayed.

**Operator response**
Report the message to the system programmer.

**Programmer response**
Determine the reason for the allocation failure, and correct the problem.

**Suppression level**
UNSUP.

**Blank suppression**
No.

**Destination**
Terminal.

**EPW9552E IEBGENER FAILED WITH RETURN CODE retcode**

**Explanation**
The dump formatter attempted to copy the specified member of the partitioned dump data set to a temporary sequential dump data set using the IEBGENER utility. This process failed with return code retcode.

**System action**
The dump formatter stops processing and redisplays the last panel displayed.

**Operator response**
Report the message to the system programmer.

**Programmer response**
Determine the reason for the allocation failure, and correct the problem.

**Suppression level**
UNSUP.

**Blank suppression**
No.

**Destination**
Terminal.

**EPW9553E NO OPTIONS ARE SELECTED**

**Explanation**
You entered N for every option on panel EPWFDFMN. To produce any output, you must choose at least one option for the dump formatter.

**System action**
None.

**Operator response**
None.

**Programmer response**
None.

**Suppression level**
UNSUP.

**Blank suppression**
No.

**Destination**
Terminal.

**EPW9554E DATA SET dsname ALREADY EXISTS**

**Explanation**
You selected NEW for the disposition of the output data set on dump formatter panel EPWFDFOD, but the data set name already exists.

**System action**
The dump formatter stops processing and redisplays panel EPWFDFOD.

**Operator response**
Either change the disposition to MOD or OLD, or change the name of the selected output data set.

**Programmer response**
None.
EPW9555E   DATA SET dsname DOES NOT EXIST

Explanation
You selected OLD for the disposition of the output data set on dump formatter panel EPWFDFOD, but the data set does not exist.

System action
The dump formatter stops processing and redisplay panel EPWFDFOD.

Operator response
Either change the disposition to MOD or NEW, or change the name of the selected output data set to an existing data set.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9556E   attr IS REQUIRED FOR A NEW DATA SET

Explanation
The name specified for an output data set is new, and you did not specify one of the required attributes. The missing attribute is attr, where attr is either VOLUME, PRIMARY SPACE, or SECONDARY SPACE.

System action
The dump formatter stops processing and redisplay panel EPWFDFOD.

EPW9557E   RETURN CODE retcode ATTEMPTING TO DEFINE IPCS DUMP DIRECTORY dirname

Explanation
The dump formatter attempted to allocate a new VSAM dump directory with a name of dirname, and it failed with return code retcode.

System action
The dump formatter stops processing and redisplay panel EPWFDFDD.

Operator response
Report the message to the system programmer.

Programmer response
Determine the reason for the allocation failure, and correct the problem.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9558E   RETURN CODE retcode ON ENTRY TO IPCS

FFST operations 147
Explanation
You were not already in IPCS, so the dump formatter attempted to enter IPCS with the IPCS NOPARM command. This process failed with return code retcode.

System action
The dump formatter stops processing andredisplays panel EPWFDFMN.

Operator response
Report the message to the system programmer.

Programmer response
Determine the reason for the failure. The user might not be authorized to access or use IPCS.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9559E RETURN CODE retcode ATTEMPTING TO USE DUMP DATA SET

Explanation
The dump formatter entered IPCS successfully and issued the IPCS SETDEF command to define the dump data set as the default. This command failed with return code retcode.

System action
The dump formatter stops processing andredisplays panel EPWFDFMN.

Operator response
Report the message to the system programmer.

Programmer response
Determine the reason for the failure. The user might not be authorized to access or use IPCS.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9560E macro FAILED ATTEMPTING TO ACCESS DUMP

Explanation
The dump formatter attempted to issue the specified macro, but the macro failed, where macro is one of the following values:

- GETMAIN
- OPEN
- READ

System action
The dump formatter stops processing andredisplays panel EPWFDFMN.

Operator response
Report the message to the system programmer.

Programmer response
Determine the reason for the failure, and correct the problem. If macro is READ, you may have used a partitioned data set without specifying a member name.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9561E NOT A VALID FFST DUMP. CANNOT FIND WORK AREA POINTER

Explanation
The dump formatter looked at the dump header for the pointer to the FFST work area, but found it to be invalid. The dump formatter requires this address to continue dump formatting.

Suppression level
UNSUP.
System action
The dump formatter stops processing and redisplays panel EPWFDFMN.

Operator response
Verify that the data set used as the dump data set actually contains an FFST raw dump.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9562E   WORK AREA NOT IN DUMP

Explanation
The dump formatter could not locate the FFST work area in the dump data set.

System action
The dump formatter stops processing and redisplays panel EPWFDFMN.

Operator response
Verify that the data set used as the dump data set actually contains an FFST raw dump.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9563E   DUMP FORMATTING CANCELLED - PASSWORD MISMATCH

Explanation
The password you typed on panel EPWFDFMN does not match the password in the dump.

System action
The dump formatter stops processing and redisplays panel EPWFDFMN.

Operator response
Get the correct password from the product’s support group.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9565E   SPECIFY Y OR N FOR option

Explanation
While using the EPWDMPFV dump formatter for VM, you did not use a valid value for option, where option is one of the following values:

- SYMPTOM STRING DATA
- FFST WORK AREA
- DATA STRUCTURE TABLE(S)
- AREA AROUND REGISTERS
- DATA STRUCTURES

System action
The dump formatter redisplays the FFST DUMP FORMATTER screen.

Operator response
Type either a Y (for yes) or an N (for no) beside the specified option.

Programmer response
None.

Suppression level
UNSUP.
Blank suppression
Yes.

Destination
Terminal.

EPW9570I  ALLOCATING TEMPORARY
SEQUENTIAL DATA SET

Explanation
You entered a member name for a partitioned dump
data set. Because IPCS requires sequential dump data
sets, the dump formatter is attempting to allocate a
temporary sequential data set to copy the member
into.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9571I  IPCS DUMP DIRECTORY
dirname
HAS BEEN DEFINED

Explanation
You are allocating a new dump directory named
dirname entered on panel EPWFDFDD. The allocation
of the VSAM cluster was successful.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9572I  ENTERING IPCS

Explanation
You were not already in an IPCS session. The dump
formatter is temporarily entering an IPCS session to
complete the dump formatting.

System action
Dump formatting continues.

Operator response
None.
EPW9574I  DUMP FORMATTING IN PROGRESS

Explanation
Setup completed successfully, and dump formatting has begun.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9575I  DUMP FORMATTING COMPLETE

Explanation
All requested output options have been sent to the requested output destination.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9576I  EXITING IPCS

Explanation
You were not already in an IPCS session when the dump formatter was invoked. The formatter entered the IPCS session temporarily (see message EPW9573I) to format the dump, and now is terminating the IPCS session.

System action
Dump formatting continues.

Operator response
None.

Programmer response
None.

Suppression level
UNSUP.

Blank suppression
No.

Destination
Terminal.

EPW9577I  DELETING TEMPORARY DATA SET

Explanation
You entered a member name of a partitioned dump data set, so the dump formatter allocated a temporary sequential data set (see message EPW9570I) and copied the dump into it (see message EPW9571I). The
Dump formatter is now deleting this temporary data set because it is no longer needed.

**System action**
Dump formatting continues.

**Operator response**
None.

**Programmer response**
None.

### FFST for MVS transition code function

In order to enhance the value of FFST, system monitor functions have been included within FFST. These monitor functions allow FFST to be involved in problems for which FFST is not directly called. Monitoring capability is provided through two user exit facilities in the MVS environment: MVS post dump exit (IEAVTSEL) and a CICS global user exit. Each allows FFST to be involved in situations where a dump is being requested.

The support provided by FFST for this function is limited to the generation of an SNA Generic Alert. The alert provides notification to a network operator that a dump was requested by a product. Each monitor function is fully controlled by an FFST filter. Details of these two monitoring functions are documented below.

### MVS post dump exit (IEAVTSEL)

PTF UN53954 (APAR PN40734) adds a new feature to FFST. It utilizes the MVS/ESA Post Dump Exit facility (IEAVTSEL) to trip an FFST software probe when this user exit is called by MVS. This new facility will provide FFST support when an SVC dump or SYSMDUMP is taken by an authorized product.

This new routine supports a parameter list for filtering in order to suppress the calling of FFST for dump situations of little interest. Filtering can be performed using the ABEND code (for ABEND conditions) and you can also specify whether or not a probe should be tripped if this new routine gets control even though the MVS Dump Analysis and Elimination (DAE) facility of MVS has suppressed the dump because it was found to be a duplicate problem.

When an authorized product issues an SDUMP or SYSMDUMP, an FFST routine gets control and extracts key information from the MVS dump header and system diagnostic work area (SDWA). This information is passed to FFST through one of eight FFST probes. See Illustration of the FFST probe IDs used by Transition Code. FFST builds a Generic Alert which is passed to a network management product such as NetView to inform network operations of a possible program problem.

To install and activate this new function, the following steps must be performed:

1. Install PTF UN53954. (The PTF cover letter also contains these instructions.)
2. Make the FFST dump exit code, EPWTRNDX, known to MVS by modifying the IEAVTSEL object code in SYS1.LINKLIB through the use of the SPZAP program or through SMP. The following entry must be placed in this module:

   ```
   C5D7E6E3 D9D5C4E7 80000000
   (EPWTRNDX)
   ```

   **Note:** 80000000 can also be 00000000

   See the “MVS/ESA Installation Exits”, GC28-1637, for details on how to perform this task.
3. If you do not want a probe to trip for situations where DAE has suppressed the dump, update your FFST Transition Code parameter list with the following information:
Where an 'S' in column 1 indicates that if DAE suppresses the dump, FFST should also suppress the probe; 'cccccccccc' is the IBM 9-character component ID of the product for which the probe should be suppressed. An '*' in column 3 indicates that all products will have the probe suppressed if DAE suppresses the dump.

Note: Another way to do this would be to use all zeros as the flag indicators in IEAVTSEL which would prevent this exit from being called when DAE suppresses a dump.

You may include as many of these statements as required. They may be intermixed with the 'I' and 'E' records currently supported by this parameter list.

4. Create one or more entries in the FFST Transition Code parameter list, member TRNSABCD of the EPWPARM data set, to suppress FFST from processing certain ABEND conditions for specific products. Each entry has the following format:

<table>
<thead>
<tr>
<th>COLUMN(S)</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I - Include only the specified ABEND codes for the specified Component ID</td>
</tr>
<tr>
<td></td>
<td>E - Exclude only the specified ABEND codes for the specified Component ID.</td>
</tr>
<tr>
<td></td>
<td>* - Interpreted as a comment record and the record is ignored.</td>
</tr>
<tr>
<td>2</td>
<td>Must be blank</td>
</tr>
<tr>
<td>3-11</td>
<td>The Component ID of the product.</td>
</tr>
<tr>
<td>12</td>
<td>Must be blank</td>
</tr>
<tr>
<td>13-x</td>
<td>One or more ABEND codes for which probe control is required. Each entry must have the format 'tcccc' where 't' is either an 'S' for system ABEND or a 'U' for user ABEND; 'cccc' is up to a four-character ABEND code. Each entry must be separated by a blank.</td>
</tr>
</tbody>
</table>

5. Re-IPL your system.

Note: Each time the FFST Transition Code parameter list is changed, FFST must be deactivated and then reactivated before the changes will take affect.

Examples of filter statements can be found in “Transition code parameter list” on page 157.

Further probe control is available by using the MODIFY probe control facility of FFST. The probe identifier values utilized by the FFST Transition Code and a description of when they are tripped are listed below:

EPWTRN05
This probe is tripped when the Transition Code gets control from the post dump exit for a non-ABEND condition, a product component ID was found, and it was for a dump that DAE did not suppress.

EPWTRN06
This probe is tripped when the Transition Code gets control from the post dump exit for a non-ABEND condition, a product component ID was NOT found, and it was for a dump that DAE did not suppress.

EPWTRN07
This probe is tripped when the Transition Code gets control from the post dump exit for an ABEND condition, a product component ID was found, and it was for a dump that DAE did not suppress.

EPWTRN08
This probe is tripped when the Transition Code gets control from the post dump exit for an ABEND condition, a product component ID was NOT found, and it was for a dump that DAE did not suppress.
This probe is tripped when the Transition Code gets control from the post dump exit for a non-ABEND condition, a product component ID was found, and it was for a situation the DAE suppressed the dump.

This probe is tripped when the Transition Code gets control from the post dump exit for a non-ABEND condition, a product component ID was NOT found, and it was for a situation that DAE suppressed the dump.

This probe is tripped when the Transition Code gets control from the post dump exit for an ABEND condition, a product component ID was found, and it was for a situation that DAE suppressed the dump.

This probe is tripped when the Transition Code gets control from the post dump exit for an ABEND condition, a product component ID was NOT found, and it was for a situation that DAE suppressed the dump.

A quick summary of these Probe IDs and the conditions under which they are tripped follows:

CICS dump global exit

APAR PN45724 adds a new feature to FFST. It utilizes a CICS global exit that gets control when a CICS transaction issues a dump. This new feature causes an FFST software probe to execute when this user exit is called by CICS.

This new routine supports a parameter list for filtering to suppress the calling of FFST for dump situations of little interest. Filtering can be performed using the ABEND code (for ABEND conditions).

To correctly install this update you will need to reference the program directory that was shipped with FFST and reference the, and perform the following steps:

• Define the FFST CICS global user exit, EPWCGUEI, in your initialization program list table (PLT). Add the following statement after the entry for EPWTRUEI:

  DFHPLT TYPE=ENTRY,
  PROGRAM=EPWCGUEI

• Define the FFST CICS global user exit, EPWCGUET, in your termination program list table (PLT). Add the following statement after the entry for EPWTRUET:

  DFHPLT TYPE=ENTRY,
  PROGRAM=EPWCGUET

• Define the FFST CICS global user exit programs EPWCGUE, EPWCGUEI, and EPWCGUET and transaction GUET as CICS resources. Define the programs and transaction using the CEDA transaction:

  CEDA DEFINE PROGRAM(EPWCGUE)
  LANGUAGE(ASSEMBLER)
  RESIDENT(YES)
  GROUP(EPWFFST)
  CEDA DEFINE PROGRAM(EPWCGUEI)
  LANGUAGE(ASSEMBLER)
  GROUP(EPWFFST)
  CEDA DEFINE PROGRAM(EPWCGUET)
  LANGUAGE(ASSEMBLER)
  GROUP(EPWFFST)
  CEDA DEFINE TRANSACTION(GUET)
  PROGRAM(EPWCGUET)

• Use FFST installation job EPW12011 as a model to correctly link edit modules EPWCGUE,
EPWCGUEI, and EPWCGUET. Modify this job as follows:
change EPWTRUEI to EPWCGUEI
change EPWTRUET to EPWCGUET
change EPWTRUE to EPWCGUE
add RENT to the link-edit parm options for each step
change RMODE=24 to RMODE=ANY in step3

Run this job.

In order to test the installation of this new function, run the FFST CICS installation verification program (CIVP). This program invokes the CICS dump facility which will drive this new function. When CIVP is run, an additional output will be produced because this transaction invokes the global user exit. The output is a primary and secondary symptom string.

The format of the primary and secondary symptom strings generated by this new function follows:

**PRIMARY SYMPTOM STRING:**

- PIDS/programid
- LVLS/lvl
- RIDS/applid
- RIDS/moduleid
- AB/Uabcode
- PCSS/transid

where:

- programid = 9 character component ID OR 8 character application name
- lvl = 3 character release/level value
- applid = 8 character application name
- moduleid = 8 character detecting module name
- abcode = 4 character transaction ABEND code
- transid = 4 character transaction identifier

**SECONDARY SYMPTOM STRING:**

- PCSS/date
- PCSS/time
- PCSS/dumpid
- PCSS/userid
- PCSS/termid
- PCSS/systemid

where:

- date = date of transaction dump
- time = time of transaction dump
- dumpid = transaction dump identifier
- userid = user identifier or *USERID*
- termid = terminal identifier
- systemid = system identifier

Add entries in member TRNSABCD in the FFST EPWPARM data set to register CICS applications or filter CICS ABEND codes. This global user exit is given the transaction name GUET, which is used as the application name. For applications that are IBM-written, code a registration record to make a correlation between the given application name and the IBM program id (component ID). You should also code the long name and vendor ID for the application.
**MVS dump frequency threshold support**

APAR PN51328 adds a new feature to FFST/MVS. It utilizes the ENF LISTEN function for the x'47' record in MVS 5.2. After the PTF is installed and FFST is started, an exit gets control when MVS DAE detects that a threshold has been reached which pertains to a certain number of instances of a symptom string seen by DAE in a certain amount of time. When this threshold is reached, the FFST exit gets control, converts the MVS symptom string to its RETAIN equivalent, and executes an FFST software probe.

You may add entries in the transition code parameter list (default name TRNSABCD) to register products in the same way as described in "MVS post dump exit (IEAVTSEL)" on page 152. The probe identifier values utilized by this new function and a description of when they are tripped are listed below:

**EPWTRN15**
This probe is tripped when the Transition Code gets control via the MVS ENF 47 exit for a non-ABEND condition and a product component ID is found.

**EPWTRN16**
This probe is tripped when the Transition Code gets control via the MVS ENF 47 exit for a non-ABEND condition and a product component ID is not found.

**EPWTRN17**
This probe is tripped when the Transition Code gets control via the MVS ENF 47 exit for an ABEND condition and a product component ID is found.

**EPWTRN18**
This probe is tripped when the Transition Code gets control via the MVS ENF 47 exit for an ABEND condition and a product component ID is not found.

A quick summary of these probe IDs and the conditions under which they are tripped follows:

**Specifying IBM product identification information**

Both the MVS post dump exit and the CICS dump exit contain most of the information FFST requires to perform its functions. Some of the information missing is the three-character REL number which indicates the release level of the product and a short descriptive name of the product (i.e., APPLID). In order to pass this information to FFST, the FFST parameter list supports a registration record which has the following format:

```
<table>
<thead>
<tr>
<th>COLUMN(S)</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 R</td>
<td>Registration record - used to pass product identification information to FFST for Transition Code probe statements.</td>
</tr>
<tr>
<td>2 Must be blank</td>
<td></td>
</tr>
<tr>
<td>3-11 The component ID of the product. For CICS transactions, this is the component id that FFST will use when a matching application identifier is found.</td>
<td></td>
</tr>
<tr>
<td>12 Must be blank</td>
<td></td>
</tr>
<tr>
<td>13-15 The three-character REL number of the product which</td>
<td></td>
</tr>
</tbody>
</table>
```

A quick summary of these probe IDs and the conditions under which they are tripped follows:
indicates its release number or level.
16 Must be blank
17-24 Application identifier - short name (For CICS transactions, this is the program name to match against for registration).
23 Must be blank
26-55 Application identifier - long name
56 Must be blank
57-72 Vendor name

Examples of registration statements can be found in “Transition code parameter list” on page 157

**Transition code parameter list**

The FFST Transition Code parameter list, (a member of the FFSTPARM data set, the default name being TRNSABCD) can be used to control the Transition Code function of FFST and assign identification information required by FFST. Sample Parameter List for FFST Transition Processing is an example of this parameter list:

```plaintext
************************************************************************
**** THIS IS A SAMPLE PARAMETER LIST FOR FFST TRANSITION PROCESSING ****
************************************************************************
************************************************************************
* THE FOLLOWING RECORD TYPES ARE SUPPORTED                             *
*                                                                      *
* - COMMENT RECORD                                                      *
* E - EXCLUDE RECORD                                                    *
* I - INCLUDE RECORD                                                    *
* R - REGISTRATION RECORD                                               *
*                                                                      *
************************************************************************
************************* EXCLUDE RECORD FORMAT ************************
************************************************************************
*                                                                      *
* COLUMN(S)                                                            *
*                                                                      *
* 1 - RECORD TYPE - 'E' FOR EXCLUDE                                    *
* 2 - BLANK                                                            *
* 3-11 - PROGRAM IDENTIFIER                                            *
* 12 - BLANK                                                           *
* 13-72 - ABEND CODE(S)                                                *
*                                                                      *
************************************************************************
* EXAMPLE:                                                            *
*                                                                      *
*......................................................................*
*This next record will exclude Transition Code support for all products* *
*that don't pass MVS a valid component ID when the MVS dump facility is* *
called. (FFST defaults the component ID to '999999999'). This next record is HIGHLY recommended. *
*......................................................................*
E 999999999 * 1230008                                                  *
*......................................................................*
*This next record will exclude Transition Code support for all products* *
*from which a S0122 ABEND (operator cancel) occurred.                  *
*......................................................................*
E * S0122 1230008                                                      *
*......................................................................*
*This next record will exclude Transition Code support for the product* *
*with a component ID of 569511111 and ABEND codes S0FCA and S0FDA.     *
*......................................................................*
E 569511111 S0FCA S0FDA                                                 *
*......................................................................*
*This next record will exclude Transition Code support for the CICS Transaction 'TRN1' when any user ABEND starting with the value 'A' occurs. *
*......................................................................*
E TRN104402 UA*                                                        *
************************************************************************
************************* INCLUDE RECORD FORMAT ************************
************************************************************************
*                                                                      *
* COLUMN(S)                                                            *
*                                                                      *
* 1 - RECORD TYPE - 'I' FOR INCLUDE                                    *
* 2 - BLANK                                                            *
* 3-11 - PROGRAM IDENTIFIER                                            *
*                                                                      *
************************************************************************
**FFST operations 157**
Sample Parameter List for FFST Transition Processing

S/390 channel attached device support

ServiceView is IBM’s strategic approach to promote commonality in solutions for the support and service of both hardware and software products developed by the IBM Corporation. ServiceView implementations will be provided in a series of stages. Stage 1 addresses the centralized management of S/390® channel-attached devices through the use of the System Network Architecture (SNA) Generic Alert.

FFST provides this function through a Small Programming Enhancement (SPE) (PTF UN57519) called FFST S/390 Channel Attached Support which works with the IBM NetView program product to generate and processes these Generic Alerts. This SPE provides Generic Alert support for the following devices:

- **DASD**
  - 3990 models 001, 002, 003, 006, 007
  - 9343 models C02, C04, D04
  - 93419393
  - 3390 models 001, 002, 003, 009
  - 9035 model 002
  - 9345 models 001, 002
  - 9391 models 001, 002
  - 9395 model 001
- **ESCON channels**
FFST provides this support through a new FFST module, EPWSVC76, loaded from LPALIB by NetView initialization and then called by NetView when a LOGREC record is generated to report an error for one of the above devices (PTF UW00254 and Netview Release 2.3 and higher). EPWSVC76 receives control from NetView and is passed the record. This module places the record on an FFST data queue where it is dequeued and processed by FFST. When FFST receives the record, it generates an SNA Generic Alert from its contents. Probable cause and recommended repair action information along with error sense data are placed in the Generic Alert which is then sent to NetView by FFST through the NetView Program to Program Interface (PPI).

**FFST controls**

FFST provides two ways in which Generic Alert generation for a supported LOGREC record can be controlled. The following sections detail this control.

**FFST probe control**

FFST creates Generic Alerts for host channel-attached devices through a series of FFST probes - one for each device type supported. Each one of these probes are shipped as active. To completely disable Generic Alert support for one or more of the supported device types, the following MODIFY command can be entered at the MVS console:

```
F ffstproc ,ACTION=DISABLE,PROBEID=EPWHW$xx,VENDOR=IBM
```

where 'xx' is:

- 'A3' for DASD devices
- 'A2' for ESCON channel
- 'ED' for ESCON Director
- 'PP' for printer devices (permanent errors)
- 'PT' for printer devices (temporary errors)

To enable the support, the following MODIFY command can be entered at the MVS console:

```
F ffstproc ,ACTION=ENABLE,PROBEID=EPWHW$xx,VENDOR=IBM
```

where 'xx' is:

- 'A3' for DASD devices
- 'A2' for ESCON channel
- 'ED' for ESCON Director
- 'PP' for printer devices (permanent errors)
- 'PT' for printer devices (temporary errors)

See “FFST MODIFY command overview” on page 32 for a complete description of the FFST MODIFY commands.

**Parameter list control**

In order to provide a more granular filter criteria, the FFST S/390 Channel Attached Support SPE utilizes a parameter list to determine if a Generic Alert is to be generated for a specific device and/or for a specific
This parameter list resides within the partitioned data set specified by the FFSTPARM DD card within the FFST start up procedure. The default name for the member containing this parameter list is EPWHWR01 (which can be overridden in the FFSTPARM start list) and each record within the member has the following format, as illustrated below.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1      | Action code. One of the following values:  
I (Include) generate the Generic Alert if the criteria data in this record matches the information in the LOGREC record.  
E (Exclude) don't generate the Generic Alert if the criteria data in this record matches the information in the LOGREC record.  
* Comment record, all information in this record is ignored. |
| 2      | Blank |
| 3-4    | Record type.  
'A3' indicates that this record is to control an alert for a DASD device. (The LOGREC record generated for a DASD error is a type A3 record.)  
'A2' indicates that this record is to control an alert for an ESCON channel. (The LOGREC record generated for an ESCON channel error is a type A2 record.)  
'30' indicates that this record is to control an alert for a problem reported by an OBR (type 30) LOGREC record. (The ESCON Director and printer devices report their problems through an OBR record and are currently the only device types supported.) |
| 5      | Blank |
| 6-9    | Machine type. One of the following values:  
tttt a four character machine type value (e.g., 3990, 9033) which indicates either the failing machine type value (when columns 3-4 are set to either 'A3' or '30') or the machine type value of the local device to which the ESCON channel is attached (when columns 3-4 are set to 'A2').  
* indicates all machine types. |
<p>| 10     | Blank |</p>
<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
</table>
| 11-13   | Model number. One of the following values:  
|         | mmmm       |
|         | a three character model number value (e.g.,  
|         | 001, 002) which indicates either the failing  
|         | machine model number (when columns 3-4 are  
|         | set to either 'A3' or '30') or the machine model  
|         | number of the local device to which the ESCON  
|         | channel is attached (when columns 3-4 are set  
|         | to 'A2').  
|         | *          |
|         | indicates all model numbers or that a model  
|         | number is not applicable. |
| 14      | Blank       |
| 15-21   | Serial number. One of the following values:  
|         | sssssss    |
|         | up to a seven character serial number which  
|         | indicates either the failing machine serial  
|         | number (when columns 3-4 are set to either  
|         | 'A3' or '30') or the machine serial number of the  
|         | local device to which the ESCON channel is  
|         | attached (when columns 3-4 are set to 'A2').  
|         | *          |
|         | indicates all serial numbers. |
| 22      | Blank       |

The remaining data in the parameter list record is device type dependent. The format for each device type is illustrated below.

**DASD parameter list record**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
</table>
| 23-26   | Symptom Code. One of the following values:  
|         | sssss      |
|         | a four hexadecimal character symptom code  
|         | from offset 22-23 in the DASD 32-byte sense  
|         | code.  
|         | *          |
|         | indicates all symptom codes or that a symptom  
|         | code is not applicable. |
| 27-80   | Ignored, can be used for comments. |
### ESCON channel parameter list record

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
</table>
| 23-26  | One of the following values:  
  - **tttt**: a four character machine type value (e.g., 3990, 9033) of the remote device to which the ESCON channel is attached.  
  - ***:** indicates all remote machine types.  
  - **cc**: a two character CHPID value, followed by two blanks, which specifies the ESCON channel for which this parameter list record applies. If a CHPID value is specified, then the remaining fields do not apply and are ignored. |
| 27     | Blank       |
| 28-30  | Model number. One of the following values:  
  - **mmm**: a three character model number value (e.g., 001, 002) of the remote device to which the ESCON channel is attached.  
  - ***:** indicates all model numbers or that a model number is not applicable. |
| 31     | Blank       |
| 32-38  | Serial number. One of the following values:  
  - **sssssss**: up to a seven character serial number of the remote device to which the ESCON channel is attached.  
  - ***:** indicates all serial numbers. |
| 39-80  | Ignored, can be used for comments. |

### ESCON director parameter list format

There are no additional fields utilized for an ESCON Director problem.

### Printer device parameter list format

There are no additional fields utilized for a printer problem.

### Parameter list control logic

When FFST receives a LOGREC record for a device it supports, it uses the information from the parameter list to determine if the Generic Alert should be generated. If the criteria specified in a parameter list record (e.g., machine type, model number, symptom code) matches the data in the LOGREC record, then the Action code (position 1 in the record) is examined. If it is an 'I', then a Generic Alert is generated. If it is an 'E', then a Generic Alert is not generated. (A record with an asterisk in column 1 is ignored.) All records in the parameter list are searched for matching criteria. The last record with a matching criteria
will be used by FFST to control the generation of the Generic Alert. If no match is found or if this
parameter list does not exist, then the Generic Alert is generated.

The following is a sample parameter list:

************************************************************************
* THIS IS A SAMPLE START UP PARAMETER LIST FOR THE FFST CHANNEL   *
* ATTACHED SUPPORT FEATURE.                                           *
************************************************************************
******
****** A2 (ESCON) RECORD CONTROL ******
******
******
************************************************************************
* The following control records are examples of how to control the     *
* generation of an SNA Generic Alert for ESCON channel problems.       *
* If SNA Generic Alert support is to be completely disabled for any    *
* ESCON channel problem, enter the following MODIFY command at the MVS *
* console:                                                             *
*                                                                      *
* F FFST,A=DIS,V=IBM,PROBEID=EPWHWISA2                                 *
*                                                                      *
* Selective alert control for the A2 records is provided on a basis of *
* the machine type, model number, serial number of the machine that     *
* detected the problem (i.e., Incident node) and on the basis of the    *
* Attached node using either the machine type, model number, and serial*
* number, or the CHPID of the channel that had the error.              *
************************************************************************
* This first record is a control record that will stop an alert being  *
* generated for a problem with an ESCON link between a 9021 CPU, model *
* number 480, serial number 12082, and a 9033 switch, model 001, serial*
* number 0950002.                                                      *
************************************************************************
* Action Record Incident Incident Incident Attached Attached Attached  *
* Code   Type   Machine  Model    Serial   Machine  Model    Serial    *
*                 Type    Number   Number   Type     Number   Number     *
* (1)     (3)     (11)      (15)    (23)     (28)     (32)       *
*                  |       |        |       |        |        |       |
* __|       |       |        |       |        |       |       |
* |  ________|       |        |       |        |        |       |
* | |   _____________|        |       |        |        |       |
* | |   |    _________________|       |        |        |       |
* | |   |    |    ____________________|        |        |       |
* | |   |    |    |     _______________________|        |       |
* | |   |    |    |     |    |    ______________________|       |
* | |   |    |    |     |    |    ______________________________|
* | |   |    |    |     |    |    |
* | V   V    V    V     V    V    V
E A2 9021 480 0040161 9033 001 0950002
*...|....1....|....2....|....3....|....4 <====column scale
************************************************************************
* This next record is a control record that will stop an alert being  *
* generated for a problem with an ESCON link between a 9021 CPU, model *
* number 480, serial number 0040161, and any device to which the CPU is*
* attached.                                                            *
************************************************************************
E A2 9021 480 0040161 9033 001 0950002
*...|....1....|....2....|....3....|....4 <====column scale
************************************************************************
* This next record is a control record that will stop an alert being  *
* generated for a problem with an ESCON link between a 9021 CPU, model *
* number 480, serial number 0040161, on CHPID 2A.                    *
* (Note, the CHPID value must be 2 characters in length. This         *
* distinguishes it from an Attached Machine Type value which is        *
* 4 characters in length.)                                            *
************************************************************************
E A2 9021 480 0040161 2A
*...|....1....|....2....|....3....|....4 <====column scale
************************************************************************
******
****** A3 (SIM) RECORD CONTROL ******
******
******
************************************************************************
* The following control records are examples of how to control the     *
* generation of an SNA Generic Alert for a problem occurring in a DASD *
* device. If SNA Generic Alert support is to be completely disabled    *
* for all DASD devices, enter the following MODIFY command at the MVS *
* console:                                                             *
*                                                                      *
* F FFST,A=DIS,V=IBM,PROBEID=EPWHWISA3                                 *
*                                                                      *
* Selective alert control for DASD devices is provided on a basis of   *

FFST operations 163
* the machine type, model number, serial number of the machine that detected the problem.

* This first record is a control record that will stop an alert from being generated for a problem on a 3390 DASD, model number 003, with a serial number of T3505 and a symptom code (byte 22-23 of the sense field in the A3 record) of X'2313'.

* This first record is a control record that will stop an alert from being generated for a problem on a 3390 DASD, model number 003, with a serial number of T3505 and a symptom code (byte 22-23 of the sense field in the A3 record) of X'2313'.

<table>
<thead>
<tr>
<th>Action Code</th>
<th>Record Type</th>
<th>Machine Type</th>
<th>Model Number</th>
<th>Serial Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(3)</td>
<td>(6)</td>
<td>(11)</td>
<td>(15)</td>
</tr>
</tbody>
</table>

E A3 3390 003 T3505 2313

This next record is a control record that will stop an alert from being generated for a problem on a 3990 DASD, model number 001, all serial numbers, all symptom codes (byte 22-23 of the sense field in the A3 record).

<table>
<thead>
<tr>
<th>Action Code</th>
<th>Record Type</th>
<th>Machine Type</th>
<th>Model Number</th>
<th>Serial Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(3)</td>
<td>(6)</td>
<td>(11)</td>
<td>(15)</td>
</tr>
</tbody>
</table>

E A3 3990 001

This next record is a control record that will stop an alert from being generated for a problem on a 3990 DASD, all models, all serial numbers, with a symptom code (byte 22-23 of the sense field in the A3 record) of X'9980'.

<table>
<thead>
<tr>
<th>Action Code</th>
<th>Record Type</th>
<th>Machine Type</th>
<th>Model Number</th>
<th>Serial Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(3)</td>
<td>(6)</td>
<td>(11)</td>
<td>(15)</td>
</tr>
</tbody>
</table>

E A3 3990 * 9980

These next two control records will stop the generation of Generic Alerts for problems on all 9343 DASD devices except for a model C04 for which Generic Alerts will be generated.

<table>
<thead>
<tr>
<th>Action Code</th>
<th>Record Type</th>
<th>Machine Type</th>
<th>Model Number</th>
<th>Serial Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(3)</td>
<td>(6)</td>
<td>(11)</td>
<td>(15)</td>
</tr>
</tbody>
</table>

E A3 9343 * Exclude all 9343 problems

I A3 9343 C04 * Include all 9343 model C04 problems

The following control records are examples of how to control the generation of an SNA Generic Alert for OBR records. If Generic Alert support is to be completely disabled for OBR records, enter the following MODIFY command:

```
F FFST,A=DIS,V=IBM,PROBEID=EPWHwSxx
```

where "x" is:

- 'ED' - for ESCON Director problems
- 'PP' - for printer problems (permanent errors)
- 'PT' - for printer problems (temporary errors)

Selective alert control for Long OBR records is provided on a basis of the machine type, model number, and serial number of the machine that detected the problem.

This first record is a control record that will stop an alert from being generated for an ESCON Director with a machine type of 9032, a model number of 001, and a serial number of 0000324.

<table>
<thead>
<tr>
<th>Action Code</th>
<th>Record Type</th>
<th>Incident Machine Type</th>
<th>Incident Model Number</th>
<th>Incident Serial Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>(3)</td>
<td>(6)</td>
<td>(11)</td>
<td>(15)</td>
</tr>
</tbody>
</table>

E A3 9032 001 0000324
Sample Startup Parameter List for FFST Channel Attached Support
Appendix B. VTAM internal trace (VIT) record descriptions

This appendix contains the VTAM internal trace (VIT) record descriptions. The entries are listed alphabetically by entry name.

For more information about VIT options, see “Trace options for the VIT” on page 11.

**** Entry for SDUMP taken for CSDUMP request

<table>
<thead>
<tr>
<th>Entry:</th>
<th>**** (CSDUMP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT option:</td>
<td>None (Generated by VTAM)</td>
</tr>
<tr>
<td>Event:</td>
<td>ISTRACSW disabled VIT to issue SDUMPX request for CSDUMP command.</td>
</tr>
<tr>
<td>VIT processing module:</td>
<td>ISTRUCTR (SNAP trace recording routine)</td>
</tr>
<tr>
<td>Control is returned to:</td>
<td>ISTRACSW</td>
</tr>
</tbody>
</table>

The **** (CSDUMP) trace record is written when VTAM disables the VTAM internal trace (VIT) to take dump for CSDUMP request.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C“****”</td>
</tr>
<tr>
<td>04–1F</td>
<td>SDUMP TAKEN FOR CSDUMP REQ</td>
</tr>
</tbody>
</table>

**** Entry for VTAM CSDUMP trigger

Entry:         **** (CSDUMP)
VIT option:    None
Event:         CSDUMP trigger set by one of the following items:
    • Modify CSDUMP command message, sense, or default dump trigger.
    • CSDUMP start option message or sense trigger.
VIT processing module:
ISTRACSW

Control is returned to:
Caller of ISTRACSW

These trace records are issued when a CSDUMP triggers a dump.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;****&quot;</td>
</tr>
<tr>
<td>04</td>
<td>Blank</td>
</tr>
<tr>
<td>05–0B</td>
<td>Triggers:</td>
</tr>
<tr>
<td></td>
<td>• DEFAULT - Default CSDUMP</td>
</tr>
<tr>
<td></td>
<td>• MSG - Message trigger CSDUMP</td>
</tr>
<tr>
<td></td>
<td>• SENSE - Sense trigger CSDUMP</td>
</tr>
<tr>
<td>0C-1F</td>
<td>CSDUMP INVOKED</td>
</tr>
</tbody>
</table>

**** Entry for VTAM CSDUMP trigger (continuation)

Entry: **** (CSDUMP)

VIT option:
None

Event: Continuation

These records are a continuation of the CSDUMP trace record. These records display the General register content.

Start record:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;****&quot;</td>
</tr>
<tr>
<td>04</td>
<td>Blank</td>
</tr>
<tr>
<td>05–0B</td>
<td>Triggers:</td>
</tr>
<tr>
<td></td>
<td>• DEFAULT - Default CSDUMP</td>
</tr>
<tr>
<td></td>
<td>• MSG - Message trigger CSDUMP</td>
</tr>
<tr>
<td></td>
<td>• SENSE - Sense trigger CSDUMP</td>
</tr>
<tr>
<td>0C-1F</td>
<td>CSDUMP INVOKED</td>
</tr>
</tbody>
</table>

**** MODIFY CSDUMP INVOKERS REGS FOLLOW
**First register group record:**

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

| R14 | R0 | R1 | R2 | R3 | R4 | R5 |

**Second register group record:**

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

| R6 | R7 | R8 | R9 | R10 | R11 | R12 |

**Byte (hex) Contents**

**00–03**  
Record ID: C"****"  

**04**  
Blank  

**05–1F**  
CSDUMP INVOKERS REGS FOLLOW  

**First register group record:**

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

| R14 | R0 | R1 | R2 | R3 | R4 | R5 |

**Second register group record:**

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

| R6 | R7 | R8 | R9 | R10 | R11 | R12 |

**Byte (hex) Contents**

**00–03**  
Record ID: C"****"  

**04–07**  
Contents of register R14  

**08–0B**  
Contents of register R0  

**0C–0F**  
Contents of register R1  

**10–13**  
Contents of register R2  

**14–17**  
Contents of register R3  

**18–1B**  
Contents of register R4  

**1C–1F**  
Contents of register R5  

**Second register group record:**

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

| R6 | R7 | R8 | R9 | R10 | R11 | R12 |
**Contents of register R7**

**Contents of register R8**

**Contents of register R9**

**Contents of register R10**

**Contents of register R11**

**Contents of register R12**

**End record:**

```
0 1 2 3 4 5 6 7 8 9 A AB BC CD DE EF F
0 1 2 3 4 5 6 7 8 9 A AB BC CD DE EF

**** ENDCSDUMP INVOKERS REGS
```

**Byte (hex)**

**Contents**

**00–03**

Record ID: C“****”

**04**

Blank

**05–1F**

END CSDUMP INVOKERS REGS

---

****** Entry for VTAM ISTORFBA DUMP**

**Entry:**

**** (ISTORFBA)

**VIT option:**

None (Generated by VTAM)

**Event:**

ISTORFBA detected the available buffer count in the BPCB or PXB is incorrect. The registers are included in the VIT and a dump is taken.

**VIT processing module:**

ISTRACTR (SNAP Trace recording routine)

**Control is returned to:**

ISTORFBA

These trace records are issued with SNAP trace entries when ISTORFBA detects the available buffer count in the BPCB or PXB incorrect.
Byte (hex)  
Contents

00–03  
Record ID: C"****"

04–1F  
ISTORFBA DUMP invoked

**** Entry for VTAM ISTORFBA DUMP (continuation)

Entry:  
**** (ISTORFBA)

VIT option:  
None (Generated by VTAM)

Event:  
Continuation

These records are a continuation of the ISTORFBA DUMP trace record. These records display the General 
register content.

Start record:

 Byte (hex)  
Contents

00–03  
Record ID: C"****"

04–1F  
Start delimiter

First register group record:

 Byte (hex)  
Contents
00–03
Record ID: C“****”
04–07
Contents of register R14
08–0B
Contents of register R0
0C–0F
Contents of register R1
10–13
Contents of register R2 Save area register 13
14–17
Contents of register R3
18–1B
Contents of register R4
1C–1F
Contents of register R5

Second register group record:

| 1F  | 1E  | 1D  | 1C  | 1B  | 1A  | 19  | 18  | 17  | 16  | 15  | 14  | 13  | 12  | 11  | 10  | 09  | 08  | 07  | 06  | 05  | 04  | 03  | 02  | 01  | 00  |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 1F  | 1E  | 1D  | 1C  | 1B  | 1A  | 19  | 18  | 17  | 16  | 15  | 14  | 13  | 12  | 11  | 10  | 09  | 08  | 07  | 06  | 05  | 04  | 03  | 02  | 01  | 00  |

****     R6  R7  R8  R9  R10  R11  R12

Byte (hex)

Contents

00–03
Record ID: C“****”
04–07
Contents of register R6
08–0B
Contents of register R7
0C–0F
Contents of register R8
10–13
Contents of register R9 BPCB or PXB pointer with wrong available buffer count
14–17
Contents of register R10
18–1B
Contents of register R11
1C–1F
Contents of register R12

End record:
Byte (hex)
Contents
00–03
Record ID: C“*****”
04–1F
End delimiter

**** Entry for VTAM FFST

Entry:
**** (FFST)

VIT option:
None (Generated by VTAM)

Event:
VTAM probe triggered

VIT processing module:
ISTRACTR (SNAP trace recording routine)

Control is returned to:
ISTRACZE

The **** (FFST) trace record is written when VTAM disables the VTAM internal trace (VIT) to collect information for a probe that has been triggered. This entry is not associated with any VIT options but is recorded when an unusual condition triggers a probe instruction.

Byte (hex)
Contents
00–03
Record ID: C“*****”
04
Blank
05–1E
C“VIT DISABLED FOR VTAM/FFST”
1F
Blank
**Entry for VTAM Phantom FFST**

**Entry:**
**** (Phantom FFST)

**VIT option:**
None

**Event:**
FFST IPROBE trip in programming product

**VIT processing module:**
- VTAM — ISTRAZPF
- CSM — IVTSMZPF

**Control is returned to:**
Module that issued the IPROBE macro.

These trace records are issued when an IPROBE is tripped, and First Failure Support Technology (FFST) is not active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;****&quot;</td>
</tr>
<tr>
<td>04–1F</td>
<td>Probe name in first record identifies what product. For example:</td>
</tr>
<tr>
<td></td>
<td>• ISTxxxxx is for VTAM</td>
</tr>
<tr>
<td></td>
<td>• IVTxxxxx is for CSM</td>
</tr>
</tbody>
</table>

**Entry for VTAM Phantom FFST (continuation)**

**Entry:**
**** (Phantom FFST)

**VIT option:**
None

**Event:**
Continuation

These records are a continuation of the PFFST trace record. These records display the General register content of the module that attempted the IPROBE (general registers R14, R0 through R12, followed by a delimiter record).

**Start record:**
PFFST INVOKER REGS FOLLOW

**First register group record:**

Byte (hex)
Contents
00–03  Record ID: C"****"
04–1F  Start delimiter

**Second register group record:**

Byte (hex)
Contents
00–03  Record ID: C"****"
04–07  Contents of register R14
08–0B  Contents of register R0
0C–0F  Contents of register R1
10–13  Contents of register R2
14–17  Contents of register R3
18–1B  Contents of register R4
1C–1F  Contents of register R5
Byte (hex)
Contents
00–03  
Record ID: C“****”
04–07  
Contents of register R6
08–0B  
Contents of register R7
0C–0F  
Contents of register R8
10–13  
Contents of register R9
14–17  
Contents of register R10
18–1B  
Contents of register R11
1C–1F  
Contents of register R12
End record:

ATCPFSAV, located in the ATCVT control block, contains the callers Register 13. This pointer field can be located in the PFFST dump.
ABND entry for abend SNAP routine

- **Entry:**
  - ABND

- **VIT option:**
  - None (Generated by SNAP routine)

- **Event:**
  - Abend

- **VIT processing module:**
  - ISTRACTR (SNAP trace recording routine)

- **Control is returned to:**
  - VTAM abend recovery routine (many possible)

This trace record is written when an abend occurs in a VTAM module. This entry is not associated with any VIT options but is recorded as an exception condition when an abend occurs.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ABND&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Abend completion code</td>
</tr>
<tr>
<td>08–0B</td>
<td>PST address or 0</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Request parameter header (RPH) address or 0</td>
</tr>
<tr>
<td>10–13</td>
<td>SDWA address or 0</td>
</tr>
<tr>
<td>14–17</td>
<td>SDWA flags or 0</td>
</tr>
<tr>
<td>18–1F</td>
<td>Abend PSW or 0</td>
</tr>
</tbody>
</table>

ACA1 or ACI1 entry for LU 6.2 authorized IO or LU 6.2 TPIO (Part 1)

- **Entry:**
  - ACA1 or ACI1

- **VIT option:**
  - APPC

- **Event:**
  - LU 6.2 authorized IO or LU 6.2 TPIO (Part 1)

- **VIT processing module:**
  - ISTRACAC
Control is returned to:
   ISTAICAR

The ACA1 trace record shows LU 6.2 requests under the authorized path or LU 6.2 requests not under the authorized path but issued by programs running under an authorized key. The ACI1 trace record shows LU 6.2 requests running under a normal path.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;ACA1&quot; for authorized path</td>
</tr>
<tr>
<td></td>
<td>• C&quot;ACI1&quot; for normal path</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Exit definition (RPLEXTDS)</td>
</tr>
<tr>
<td>06</td>
<td>Option code byte 1 (RPLOPT1)</td>
</tr>
<tr>
<td>07</td>
<td>Option code byte 6 (RPLOPT6)</td>
</tr>
<tr>
<td></td>
<td>Contains the RPLXBFL flag (bit 5) that indicates whether the application is requesting HPDT services.</td>
</tr>
<tr>
<td>08–0B</td>
<td>RPL address</td>
</tr>
<tr>
<td>0C</td>
<td>APPCCMD CONTROL operand value:</td>
</tr>
<tr>
<td></td>
<td>X'10' ALLOC</td>
</tr>
<tr>
<td></td>
<td>X'11' PREALLOC</td>
</tr>
<tr>
<td></td>
<td>X'12' SENDFMH5</td>
</tr>
<tr>
<td></td>
<td>X'20' RESETRCV</td>
</tr>
<tr>
<td></td>
<td>X'30' DEALLOC</td>
</tr>
<tr>
<td></td>
<td>X'31' DEALLOCQ</td>
</tr>
<tr>
<td></td>
<td>X'40' OPRCNTL</td>
</tr>
<tr>
<td></td>
<td>X'50' PREPRCV</td>
</tr>
</tbody>
</table>
APPCCMD QUALIFY operand value:

- **NULL** (X'00')
- **ABNDPROG** (X'01')
- **ABNDSERV** (X'02')
- **ABNDTIME** (X'03')
- **ABNDUSER** (X'04')
- **ANY** (X'05')
- **CNOS** (X'06')
- **CONFIRM** (X'07')
- **CONFRMD** (X'08')
- **DATA** (X'09')
- **DATACON** (X'0A')
- **DATAFLU** (X'0B')
- **DEFINE** (X'0C')
- **DISPLAY** (X'0D')
0E–0F
User buffer length (RPLBUFL) for receive RPL or 0

10–13
Address of SEND data or RECEIVE buffer

If HPDT services are requested, the RPL area contains the extended buffer list area. The contents are recorded in the XBA1, XBA2, and XBA3 entries.

14–17
APPCCMD flags (RPL6FLGS) or 0

18–1B
User field for LU 6.2 commands when CONTROL equals ALLOC, PREALLOC, or RCVFMH5. CID for LU 6.2 commands when CONTROL equals OPRCNTL, and QUALIFY equals either ACTSESS or DACTSESS. 0 for LU 6.2 commands for which one of the following conditions is true:

- CONTROL equals OPRCNTL, and QUALIFY equals either CNOS, DEFINE, or DISPLAY.
- CONTROL equals RECEIVE or RCVDEXPD, and QUALIFY equals ANY or IANY.
- CONTROL equals REJECT, and QUALIFY equals SESSION or CONVGRP.
- CONTROL equals SETSESS, and QUALIFY equals RESUME.

Conversation ID for all other LU 6.2 commands

1C–1F
Sense data (RPL6SNSO) or 0

**ACA2 or ACI2 entry for LU 6.2 authorized IO or LU 6.2 TPIO (Part 2)**

**Entry:**
ACA2 or ACI2

**VIT option:**
APPC

**Event:**
LU 6.2 authorized IO or LU 6.2 TPIO (Part 2)

**VIT processing module:**
ISTRACAC

The ACA2 trace record is a continuation of ACA1. The ACI2 trace record is a continuation of ACI1.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;ACA2&quot; for continuation of ACA1</td>
</tr>
<tr>
<td></td>
<td>• C&quot;ACI2&quot; for continuation of ACI1</td>
</tr>
<tr>
<td>04–07</td>
<td>Address of the issuer of the APPCCMD macroinstruction</td>
</tr>
<tr>
<td>08–0F</td>
<td>When CONTROL=ALLOC, PREALLOC, or REJECT, and QUALIFY=CONVGRP, then 08–0B is the conversion group ID (CGID), and 0C–0F is 0.</td>
</tr>
</tbody>
</table>
• When CONTROL equals ALLOC or PREALLOC (except when QUALIFY equals CONVGRP) or when CONTROL equals OPRCNTL, and QUALIFY equals either CNOS, DEFINE, or DISPLAY, then 08–0F is the mode name for LU 6.2 commands.

• When CONTROL equals SETSESS or when CONTROL equals REJECT, and QUALIFY equals SESSION, then 08–0F is the session instance identifier for LU 6.2 commands.

10–17
Local logical unit name for LU 6.2 commands when CONTROL equals ALLOC or PREALLOC or when CONTROL equals OPRCNTL, and QUALIFY equals either CNOS, DEFINE, DISPLAY, or RESTORE. Otherwise, this field contains blanks or 0.

18–1F
Partner logical unit name for LU 6.2 commands when CONTROL equals ALLOC or PREALLOC or when CONTROL equals OPRCNTL, and QUALIFY equals either CNOS, DEFINE, DISPLAY or RESTORE. Otherwise, this field contains blanks or 0.

**ACA3 or ACI3 entry for LU 6.2 authorized IO or LU 6.2 TPIO (Part 3)**

**Entry:**
ACA3 or ACI3

**VIT option:**
APPC

**Event:**
LU 6.2 authorized IO or LU 6.2 TPIO (Part 3)

**VIT processing module:**
ISTRACAC

The ACA3 trace record is a continuation of ACA2. The ACI3 trace record is a continuation of ACI2.

```
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;ACA3&quot; for continuation of ACA2</td>
</tr>
<tr>
<td></td>
<td>• C&quot;ACI3&quot; for continuation of ACI2</td>
</tr>
<tr>
<td>04–17</td>
<td>0</td>
</tr>
<tr>
<td>18–1F</td>
<td>Partner network identifier for the LU 6.2 commands when CONTROL equals ALLOC or PREALLOC (except for QUALIFY=CONVGRP) or when CONTROL equals OPRCNTL, and QUALIFY equals either CNOS, DEFINE, DISPLAY, or RESTORE. Otherwise, this field contains blanks or 0.</td>
</tr>
</tbody>
</table>
```

**ACP1 or ACR1 entry for LU 6.2 user post or RPL exit (Part 1)**

**Entry:**
ACP1 or ACR1

**VIT option:**
APPC
Event:
LU 6.2 user post or RPL exit (Part 1)

VIT processing module:
ISTRACAC

Control is returned to:
ISTAICPT and ISTAPCSX for ACP1
ISTAPCSX and ISTAICRX for ACR1

The ACP1 trace record is written when an ECB is posted. The ACR1 trace record is written when an RPL exit is dispatched. These records signal that the APPCCMD macro has completed execution and show the data returned to the user application program.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• &quot;ACP1&quot; for ECB posting</td>
</tr>
<tr>
<td></td>
<td>• &quot;ACR1&quot; for RPL exit dispatching</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>APPCCMD returned indicators (RPL6RTUN)</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>1...</td>
<td>FMH5 received indicator</td>
</tr>
<tr>
<td>.1...</td>
<td>LOGON received indicator</td>
</tr>
<tr>
<td>..1...</td>
<td>SIGNAL received indicator</td>
</tr>
<tr>
<td>06</td>
<td>VTAM return code (RPLRTNCD)</td>
</tr>
<tr>
<td>07</td>
<td>VTAM feedback code (RPLFDB2)</td>
</tr>
<tr>
<td>08–0B</td>
<td>RPL address</td>
</tr>
<tr>
<td>0C</td>
<td>APPCCMD CONTROL operand value:</td>
</tr>
<tr>
<td>X'10'</td>
<td>ALLOC</td>
</tr>
<tr>
<td>X'11'</td>
<td>PREALLOC</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>--------------------</td>
</tr>
<tr>
<td>12</td>
<td>SENDFMH5</td>
</tr>
<tr>
<td>20</td>
<td>RESETRCV</td>
</tr>
<tr>
<td>30</td>
<td>DEALLOC</td>
</tr>
<tr>
<td>31</td>
<td>DEALLOCQ</td>
</tr>
<tr>
<td>40</td>
<td>OPRCNTL</td>
</tr>
<tr>
<td>50</td>
<td>PREPRCV</td>
</tr>
<tr>
<td>60</td>
<td>RCVFMH5</td>
</tr>
<tr>
<td>70</td>
<td>RECEIVE</td>
</tr>
<tr>
<td>71</td>
<td>RCVEXPD</td>
</tr>
<tr>
<td>80</td>
<td>REJECT</td>
</tr>
<tr>
<td>90</td>
<td>SEND</td>
</tr>
<tr>
<td>91</td>
<td>SENDEXPD</td>
</tr>
<tr>
<td>A0</td>
<td>SETSESS</td>
</tr>
<tr>
<td>B0</td>
<td>TESTSTAT</td>
</tr>
</tbody>
</table>

**APPCCMD QUALIFY operand value:**

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>NULL</td>
</tr>
<tr>
<td>01</td>
<td>ABNDPROG</td>
</tr>
<tr>
<td>02</td>
<td>ABNDSERV</td>
</tr>
<tr>
<td>03</td>
<td>ABNDTIME</td>
</tr>
<tr>
<td>04</td>
<td>ABNDUSER</td>
</tr>
<tr>
<td>05</td>
<td>ANY</td>
</tr>
<tr>
<td>06</td>
<td>CNOS</td>
</tr>
<tr>
<td>07</td>
<td>CONFIRM</td>
</tr>
</tbody>
</table>
0E
Option code byte 6 (RPLOPT6)
Contains the RPLXBFL flag (bit 5), which indicates whether the application is requesting HPDT services.

0F
0

10–11
LU 6.2 primary return code, RCPRI (RPL6RCPR field in the RPL extension). For more information about RCPRI return codes, see z/OS Communications Server: SNA Programmer's LU 6.2 Reference.

12–13
LU 6.2 secondary return code, RCSEC (RPL6RCSC field in the RPL extension). For more information about RCSEC return codes, see z/OS Communications Server: SNA Programmer's LU 6.2 Reference.

14–17
APPCCMD flags (RPL6FLGS)

18–1B
Conversation ID or 0

1C–1F
Sense data returned (RPL6SNSI) or 0

ACP2 or ACR2 entry for LU 6.2 user post or RPL exit (Part 2)

Entry:
ACP2 or ACR2

VIT option:
APPC

Event:
LU 6.2 user post or RPL exit (Part 2)

VIT processing module:
ISTRACAC

The ACP2 trace record is a continuation of the ACP1 trace record. The ACR2 trace record is a continuation of the ACR1 trace record.
Byte (hex)
Contents
00–03
  Record ID:
  • C"ACP2" for continuation of ACP1
  • C"ACR2" for continuation of ACR1
04–07
  ECB address for ACP2. RPL exit address for ACR2
08–0B
  RPL6 address
0C
  APPCCMD CONTROL operand value:
    Bit
    Meaning
    X'10'
    ALLOC
    X'11'
    PREALLOC
    X'12'
    SENDFMH5
    X'20'
    RESETRCV
    X'30'
    DEALLOC
    X'31'
    DEALLOCQ
    X'40'
    OPRCNTL
    X'50'
    PREPRCV
    X'60'
    RCVFMH5
    X'70'
    RECEIVE
    X'71'
    RCVEXPD
    X'80'
    REJECT
    X'90'
    SEND
    X'91'
    SENDERXPD
    X'92'
    SENDRCV
    X'A0'
    SETSESS
    X'B0'
    TESTSTAT
0D
RPLEXTDS contains the RPLXSRV flag (bit 7), which is set if VTAM accepts all CSM buffers from the application. If no CSM buffers are specified by the application, this field is not set.

0E–0F
Indicators for the type of information received or 0

Bit
Meaning
1... DATA for byte 0E; PARTIAL_PS_HEADER for byte 0F
.1... DATA_COMPLETE
..1... DATA_INCOMPLETE
...1... SEND
.... 1... CONFIRM
.... .1... DEALLOCATE
.... ..1... LOG_DATA
.... ...1... PS_HEADER
1... PARTIAL_PS_HEADER
.xxx xxxx
0

10–13
Address of SEND data or RECEIVE buffer (RPLAREA)
If OPTCD=XBUFLST is specified, this field contains the extended buffer list area. The contents of the extended buffer list are recorded in the XBA1, XBA2, XBA3 entries.

14–17
RPL record length

18–1B
Conversation group ID (CGID) when CONTROL equals ALLOC, PREALLOC, or RCVFMH5; otherwise, 0

1C
RAB conversation FSM

1D
RAB router FSM

1E
RAB error or failure FSM

1F
RAB saved router FSM

ACRC entry for RPL6 return code

Entry:
ACRC
VIT option:
APPC

Event:
APPCSNRC macro

VIT processing module:
ISTRACAC

Control is returned to:
Issuer of the APPCSNRC macro

This trace record is written when an APPC module issues a macroinstruction to set an RPL6RC nonzero return code. The condition that this entry records is an exception and is recorded regardless of whether the APPC VIT option is active.

<table>
<thead>
<tr>
<th>Byte (hex) Contents</th>
<th>00–03</th>
<th>Record ID: C&quot;ACRC&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>04</td>
<td>ID</td>
<td>is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–06</td>
<td>0</td>
<td>Instance identifier in invoking module</td>
</tr>
<tr>
<td>07</td>
<td>Conversation ID</td>
<td></td>
</tr>
<tr>
<td>0C–0F</td>
<td>Half-session ID</td>
<td></td>
</tr>
<tr>
<td>10–13</td>
<td>Invoking module name</td>
<td></td>
</tr>
<tr>
<td>14–17</td>
<td>Address of control block containing return code</td>
<td></td>
</tr>
<tr>
<td>18–1B</td>
<td>Return code (RPL6RC)</td>
<td></td>
</tr>
<tr>
<td>18–19</td>
<td>Primary return code (RPL6RCPR)</td>
<td></td>
</tr>
<tr>
<td>1A–1B</td>
<td>Secondary return code (RPL6RCSC)</td>
<td></td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
<td></td>
</tr>
</tbody>
</table>
ACSN entry for APPC sense code

Entry:
ACSN

VIT option:
APPC

Event:
APPCSNRC macro

VIT processing module:
ISTRACAC

Control is returned to:
Issuer of the APPCSNRC macro

This trace record is written when an APPC module issues a macroinstruction to set a nonzero sense code. It is treated as an exception condition and is always traced if the VIT is active, regardless of the VIT options specified.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ACSN&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–06</td>
<td>0</td>
</tr>
<tr>
<td>07</td>
<td>Instance identifier in invoking module</td>
</tr>
<tr>
<td>08–0B</td>
<td>Conversation ID</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Half-session ID</td>
</tr>
<tr>
<td>10–13</td>
<td>Invoking module name</td>
</tr>
<tr>
<td>14–17</td>
<td>Address of control block containing sense code</td>
</tr>
<tr>
<td>18–1B</td>
<td>Sense code or internal return code</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>
ACU1 entry for LU 6.2 user exit (Part 1)

Entry:
   ACU1
VIT option:
   APPC
Event:
   LU 6.2 user exit (Part 1)
VIT processing module:
   ISTRACAC
Control is returned to:
   ISTAICUE

This trace record is written when an LU 6.2 user exit (either ATTN or TPEND) is dispatched. If both the API and APPC trace options are active, and a TPEND user exit is dispatched, VTAM generates user exit trace records for both API and APPC.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ACU1&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
</tbody>
</table>
| 05         | Exit code for user exit:  
   X'0C' for ATTN  
   X'0D' for TPEND |
| 06–07      | 0 |
| 08–0B      | The type of exit determines the value of this field. |
|            | • For ATTN exit, the exit subtype (CNOS, FMH5, or LOSS) |
|            | • For TPEND exit, byte 8 is the reason code: |
| Code       | Description |
| X'00'      | Standard HALT command issued to close the network. |
| X'04'      | HALT QUICK command or VARY INACT,I or F command issued for the application program name. |
| X'08'      | HALT CANCEL command issued or VTAM terminated abnormally. |
X'0C'
Alternate application issued an OPEN ACB for the same ACB this application has opened.

0C–0F
  0

10–17
  PLU name for ATTN exit; application program ID for TPEND exit

18–1F
  SLU name for ATTN exit; 0 for TPEND exit

ACU2 entry for LU 6.2 user exit (Part 2)

Entry:
  ACU2

VIT option:
  APPC

Event:
  LU 6.2 user exit (Part 2)

VIT processing module:
  ISTRACAC

This trace record is a continuation of the ACU1 trace record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ACU2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Exit address</td>
</tr>
<tr>
<td>08–0F</td>
<td>Mode name for ATTN exit; 0 for TPEND exit</td>
</tr>
<tr>
<td>10–17</td>
<td>0</td>
</tr>
<tr>
<td>18–1F</td>
<td>Network identifier for ATTN exit; 0 for TPEND exit</td>
</tr>
</tbody>
</table>

ADE entry for device errors

Entry:
  ADE

VIT option:
  CIO

Event:
  Failure occurs when trying to activate a device (Instance 0), deactivate a device (Instance 1), allocate a device (Instance 2), or deallocate a device (Instance 2).

VIT processing module:
  ISTRACCI
**Control is returned to:**

ISTINCAV

This trace record contains information about an error when VTAM attempts to activate a device (Instance 0), deactivate a device (Instance 1), allocate a device (Instance 2), or deallocate a device (Instance 2). The three instances of the trace record follow.

This record is treated as an exception entry and is always traced regardless of the VIT options specified.

Instance 0:

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ADE</th>
<th>COS</th>
<th>D</th>
<th>ID</th>
<th>0</th>
<th>R</th>
<th>T</th>
<th>Y</th>
<th>P</th>
<th>E</th>
<th>DLR</th>
<th>CUA</th>
<th>ADDRESS</th>
<th>SRB</th>
<th>ADDRESS</th>
<th>IOSB</th>
<th>ADDRESS</th>
<th>RPH</th>
<th>ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00–02**

Record ID: C"ADE"

**03**

Code: X'F0' for activate device (DLRPLFCD)

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

0

**06**

Return code (DLRPLRCD)

**07**

Device type

**08–09**

Channel unit address

**0A**

UCBINFO return code or 0

**0B**

UCBINFO reason code or 0

**0C–0D**

Data device address (DLRDATAD)

**0E–0F**

Flags:

**Bit**

**Meaning**

1... 1.... 1....

MPC indicator

.1... 1.... 1....

TCP/IP legacy DLC

..1.... 1.... 1....

IDX indicator

...1.... 1.... 1....

TCP/IP CTC DLC
TCP/IP CLAW DLC

TCP/IP LCS DLC

TCP/IP CDLC DLC

TCP/IP HYPERchannel DLC

QDIO indicator

QDIO indicator

Not used

10–13
XCNCB address

14–17
SRB address

18–1B
IOSB address

1C–1F
RPH address

Instance 1:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;ADE&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Code: X'F1' for deactivate device (DLRPLFCD)</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Return code (DLRPLRCD)</td>
</tr>
<tr>
<td>07</td>
<td>Device type</td>
</tr>
<tr>
<td>08–09</td>
<td>Channel unit address</td>
</tr>
<tr>
<td>0A–0D</td>
<td>0</td>
</tr>
<tr>
<td>0E</td>
<td>Flags:</td>
</tr>
</tbody>
</table>
Bit  
Meaning  
1... ....  
MPC indicator  
.1. ....  
TCP/IP legacy DLC  
..1. ....  
IDX indicator  
...1 1111  
Not used  

0F  
UCB attention table index  

10–13  
DCB address  

14–1B  
0  

1C–1F  
RPH address  

Instance 2:  

<table>
<thead>
<tr>
<th>ADE</th>
<th>COS</th>
<th>ASID</th>
<th>0</th>
<th>RC</th>
<th>TYPE</th>
<th>DEV</th>
<th>EHDR</th>
<th>LINE</th>
<th>RDTE ADDRESS</th>
<th>UCB ADDRESS</th>
<th>RUPE ADDRESS</th>
<th>RUPE RUCODE OR 0</th>
<th>RPH ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>C D E F</td>
<td>0 1 1 1</td>
<td>1 1 1 1</td>
<td>1 1 1 1</td>
<td>1 1 1 1</td>
</tr>
</tbody>
</table>

Byte (hex)  
Contents  

00–02  
Record ID: C"ADE"  

03  
Code: X'CA' for allocate device or X'CD' for deallocate device (DLRPLFCD)  

04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.  

05  
0  

06  
Return code (DLRPLRCD)  

07  
Device type  

08–09  
Device address  

0A  
RDTE entry type or 0  

0B  
RDTE header type or 0  

0C–0F  
Line RDTE address
AFSM entry for altering an FSM state

Entry:
AFSM

VIT option:
SSCP

Event:
Alteration of an FSM state

VIT processing module:
ISTRACSC

Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is written when the current state of an FSM changes.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 – 03</td>
<td>Record ID: C&quot;AFSM&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Data link control type:</td>
</tr>
<tr>
<td>A</td>
<td>ATM</td>
</tr>
<tr>
<td>E</td>
<td>Enterprise Extender</td>
</tr>
<tr>
<td>L</td>
<td>LAN (External Communication Adapter)</td>
</tr>
</tbody>
</table>
Shared Memory Communications

FSM type:
A  AAL FSM (ATM only)
D  LDLC FSM
E  LDLC XID FSM
I  Shared Memory Communications - Direct Memory Access (SMC-D) FSM
L  Link FSM
P  Port FSM
R  RDMA over Converged Ethernet (RoCE) user FSM
S  Shared Memory Communications over Remote Direct Memory Access (SMC-R) FSM
X  XID FSM

Old state

New state

Work element type:
01  ISTRPH
17  ISTAUCPL
40  ISTLSPL
54  ISTRUPE
58  ISTTQE
99  IUTTIPAC
9A  ISTTSSPL
9B  ISTLSCB

0 or instance of the SETAFSM macro in the module

Address of the control block containing the FSM
10–13
Return address of the module that changed the FSM state

14–17
Work element address

18–1B
Name of the module that changed the FSM state

1C–1F
Request parameter header (RPH) address

---

**AI1 entry for authorized IO (Part 1)**

**Entry:**
AI1

**VIT option:**
API

**Event:**
Authorized IO (Part 1)

**VIT processing module:**
ISTRACAP

**Control is returned to:**
- ISTAICIR for:
- ISTOCCSM for:
  - SEND
  - INQUIRE
  - RECEIVE
  - INTRPRET
  - RESETSR
  - RVCMD
  - SESSIONC
  - SENDCMD
  - SETLOGON
  - SESSIONC (BIND)
  - SIMLOGON
  - OPNDST
  - REQSESS
  - OPNSEC
  - TERMSESS
  - CLSDST

This trace record shows API requests under the authorized path or API requests not under the authorized path but issued by programs running under an authorized key. The API routine ISTAICIR or ISTOCCSM
writes the entry and then queues the RPL to the correct PAB. For an explanation of the RPL fields, see z/OS Communications Server: SNA Data Areas Volume 1.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;AI1&quot;</td>
</tr>
<tr>
<td>03</td>
<td>RPL request type (RPLREQ)</td>
</tr>
<tr>
<td>X’15’</td>
<td>SETLOGON</td>
</tr>
<tr>
<td>X’16’</td>
<td>SIMLOGON</td>
</tr>
<tr>
<td>X’17’</td>
<td>OPNDST</td>
</tr>
<tr>
<td>X’19’</td>
<td>CHANGE</td>
</tr>
<tr>
<td>X’1A’</td>
<td>INQUIRE</td>
</tr>
<tr>
<td>X’1B’</td>
<td>INTRPRET</td>
</tr>
<tr>
<td>X’1F’</td>
<td>CLSDST</td>
</tr>
<tr>
<td>X’22’</td>
<td>SEND</td>
</tr>
<tr>
<td>X’23’</td>
<td>RECEIVE</td>
</tr>
<tr>
<td>X’24’</td>
<td>RESETSR</td>
</tr>
<tr>
<td>X’25’</td>
<td>SESSIONC</td>
</tr>
<tr>
<td>X’27’</td>
<td>SENDCMD</td>
</tr>
<tr>
<td>X’28’</td>
<td>RCVCMD</td>
</tr>
<tr>
<td>X’29’</td>
<td>REQSESS</td>
</tr>
<tr>
<td>X’2A’</td>
<td>OPNSEC</td>
</tr>
<tr>
<td>X’2C’</td>
<td>TERMSESS</td>
</tr>
</tbody>
</table>
| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.
AI2 entry for authorized IO (Part 2)

Entry:  
AI2

VIT option:  
API

Event:  
Authorized IO (Part 2)

VIT processing module:  
ISTRACAP

This trace record is a continuation of the AI1 entry. It shows additional information about the PLU and SLU in a session established or terminated by a SIMLOGON, OPNDST, CLSDST, REQSESS, OPNSEC, or TERMSESS macroinstruction. This information includes some of the parameters exchanged and the names of the PLU and SLU.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>RETURN ADDRESS</td>
</tr>
<tr>
<td>0</td>
<td>LOGON MODE NAME OR 0</td>
</tr>
<tr>
<td>0</td>
<td>APPLICATION LU NAME (IF AVAILABLE)</td>
</tr>
<tr>
<td>0</td>
<td>PARTNER LU (IF AVAILABLE)</td>
</tr>
</tbody>
</table>

05  
Exit definition (RPLEXTDS)

06  
Option code byte 4 (RPLOPT4)

07  
Option code byte 1 (RPLOPT1)

08–0B  
RPL address

0C–0F  
NIB address or CID

10  
Third byte of RH (RPLRH3)

11  
Send/receive type (RPLSRTYP)

12  
VTAM flags (RPLVTFL1)

13  
Post/respond flags (RPLVTFL2)

14  
RU chain position (RPLCHN)

15–17  
RU control codes (RPLCNTRL)

18–1B  
VTAM options (RPLOPTC2, which corresponds to RPLOPT5 through RPLOPT8) (See z/OS Communications Server: SNA Programming for additional information.)

1C–1F  
VTAM options (RPLOPTC3, which corresponds to RPLOPT9 through RPLOPT12) (See z/OS Communications Server: SNA Programming for additional information.)
00–02
Record ID: C"AI3"

03
0

04–07
Address of the issuer of the macro

08–0F
For SIMLOGON, OPNDST, CLSDST, or REQSESS: Logon mode name (NIBLMODE) or 0 if the NIB is not available
For OPNSEC or TERMSESS: 0

10–17
Application LU name if available

18–1F
Partner LU name if available

**AI3 entry for authorized IO (part 3)**

Entry:
AI3

VIT option:
API

Event:
Authorized IO (Part 3)

VIT processing module:
ISTRACAP

This trace record is a continuation of the AI2 entry. It shows additional information in a session established or terminated by a SIMLOGON, OPNDST, CLSDST, REQSESS, or OPNSEC macroinstruction.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;AI3&quot;</td>
</tr>
<tr>
<td>03</td>
<td>0</td>
</tr>
</tbody>
</table>
| 04–07      | For SIMLOGON, CLSDST, or REQSESS: Address of user data (RPLAREA)  
For OPNDST or OPNSEC: Address of BIND (NIBNDAR) or 0 if the NIB is not available |
| 08–0F      | Target LU network identifier (when for CLSDST PASS) or:  
Byte (hex) Contents |
08–09
Count of node initialization blocks (NIBs) in NIB list (when not for CLSDST PASS)

0A–0B
Count of node initialization blocks (NIBs) in NIB list with NIBRPARM=0 (when not for CLSDST PASS)

0C–0F
0

10–17
Target logical unit name, if available, or 0

18–1F
Partner LU network identifier, if available

**ALSx entry for adjacent link station**

**Entry:**
ALSA, ALSD, ALSP, ALSR, ALSS, or ALSV

**VIT option:**
SSCP

**Event:**
ALSLIST macro

**VIT processing module:**
ISTRACSC

**Control is returned to:**
Module invoking the INTRACE macro that caused the record to be produced.

The adjacent link station (ALS) trace record contains information about the adding, deleting, replacing, selecting, or validating of an adjacent link station.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td>C&quot;ALSA&quot;</td>
<td>Add an adjacent link station.</td>
</tr>
<tr>
<td>C&quot;ALSD&quot;</td>
<td>Delete an adjacent link station.</td>
</tr>
<tr>
<td>C&quot;ALSP&quot;</td>
<td>Search for an APPN link station in the cross-domain resource's ALS list.</td>
</tr>
<tr>
<td>C&quot;ALSR&quot;</td>
<td>Replace an adjacent link station.</td>
</tr>
<tr>
<td>C&quot;ALSS&quot;</td>
<td>Select an adjacent link station.</td>
</tr>
</tbody>
</table>
C"ALS V"
   Validate an adjacent link station.

04
   ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
   Return code from ALSLIST macro invocation

06
   Session control block identifier or 0

07
   Flags (from the following list):

   Bit  Meaning
   1.... The type of add is dynamic.
   ...1. Autologon processing requested.
   ....1 Verify that the add is required.
   ...1 Waiting autologon requests must be processed.
   .... xx... Connection type of PU whose address is given in PUADDR:
       00 Unknown
       01 LEN
       11 APPN
   08–0F
   ALS name or 0
   • For ALSA, ALSR, or ALSV: New ALS name passed on the add, replace, or validate function
   • For ALSD or ALSP: 0
   • For ALSS: ALS name returned on the select function

10–13
   Return address of the caller of the ALSLIST function

14–17
   Address of the cross-domain resource passed to the ALSLIST function

18–1F
   ALS name, PU address, or 0
   • For ALSA or ALSP: 0
   • For ALSD or ALSR: Old ALS name passed on the delete or replace function
   • For ALSS or ALSV: 0 and PUADDR
      - 18–1B: 0
      - 1C–1F: The PU address that is returned for the select or validate function
AP entry for signals passed between components of the APPN CP

Entry:
AP

VIT option:
SSCP

Event:
Signals passed between components of the APPN CP

VIT processing module:
ISTRACSC

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced.

This trace record provides information about inbound request and responses which are processed by the CP. If this entry is associated with an event failure, nonzero sense code, it is generated regardless of the SSCP option. It is recognized as an exception code and is traced when the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–01</td>
<td>Record ID: C&quot;AP&quot;</td>
</tr>
<tr>
<td>02</td>
<td>To process anchor block (PAB) identification byte:</td>
</tr>
<tr>
<td></td>
<td>Code (hex)</td>
</tr>
<tr>
<td></td>
<td>ID</td>
</tr>
<tr>
<td>&quot;A&quot;</td>
<td>Advanced program-to-program communication (APPC)</td>
</tr>
<tr>
<td>&quot;B&quot;</td>
<td>CMIP services session layer</td>
</tr>
<tr>
<td>&quot;C&quot;</td>
<td>Session services for CP-CP sessions (SSC)</td>
</tr>
<tr>
<td>&quot;D&quot;</td>
<td>APPN directory services (DR)</td>
</tr>
<tr>
<td>&quot;E&quot;</td>
<td>Directory services management exit (DSME)</td>
</tr>
<tr>
<td>&quot;F&quot;</td>
<td>CMIP services association control function, association control service element, directory service element, presentation</td>
</tr>
<tr>
<td>&quot;G&quot;</td>
<td>VTAM topology agent (TOPO)</td>
</tr>
<tr>
<td>&quot;H&quot;</td>
<td>Half session (HS)</td>
</tr>
</tbody>
</table>
"I"  Disk input or output
  **Note:** This code is an uppercase letter I.

"J"  Management services transport (MST) main processing

"K"  CMIP services initialization, termination, and command processing

"L"  Session services for LU-LU sessions (SSL)

"M"  Management services transport (MST) initialization processing

"N"  Node operator services

"O"  CMIP services management information base (MIB) controller

"P"  SSCP functions

"Q"  CMIP services scoping and replication

"R"  CMIP services event filtering and forwarding

"S"  System services control point (SSCP)

"T"  Topology and routing services (TRS)

"U"  Recovery RTP context manager (RCM)

"Y"  Recovery manager (RVM)

"X"  Transaction programs (XP)

"Y"  Recovery PAB (RPAB)

"Z"  Dependent LU server (DLUS)

"L"  LU server (LUS)
  **Note:** This code is a lowercase letter L.

03  From process anchor block (PAB) Identification byte. The codes are the same as byte 2.

04  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05–07  Last three characters of the module invoking APSEND. (The characters *KWN* indicate that the module name is not known.)

08–0B  Address of the issuer of the APSEND module

0C–0F  Sense code or 0
10–1F
First 16 bytes of the interprocess signal (IPS). For a list of interprocess signals, see z/OS Communications Server: SNA Data Areas Volume 2.

A2 entry for requests and responses with a RUPE

Entry:
  A2

VIT option:
  SSCP

Event:
  Requests and Responses with a RUPE

VIT processing module:
  ISTRACSC

This trace record is a continuation of the AP entry.

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>1A</td>
<td>1B</td>
<td>1C</td>
<td>1D</td>
<td>1E</td>
<td>1F</td>
</tr>
</tbody>
</table>

A2

30 MORE BYTES OF THE INTERPROCESS SIGNAL (IPS)

Byte (hex) Contents
00–01 Record ID: C"A2"
02–1F 30 more bytes of the interprocess signal

ARB entry for RTP LU-LU session data

Entry:
  ARB

VIT option:
  HPR

Event:
  Any LU-LU session data sent across an RTP connection

VIT processing module:
  ISTITCHP

Control is returned to:
  ISTRPCRS

This trace record is written when LU-LU session data is sent across an RTP connection. It contains statistical information about the LU-LU session data.
Byte (hex)

Contents

00–03
  Record ID: C"ARB"

04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
  0

06
  Trace instance

07
  Information Flags and Burst Multiplier

  Bit
  Meaning
    xx.. ....
    Adaptive Rate Based Algorithm Used
      00
        ARB Mode Algorithm
      01
        ARB Responsive Mode Algorithm
    ..xx xxxx
    Burst Multiplier (RPN_BURST_MULT)

08–0B
  Send rate in bytes per second

0C–0F
  Burst interval in milliseconds

10–13
  Bytes sent in the burst interval (RPN_BYTES_SENT)

14–17
  Bytes left to send (RPN_BYTES_TO_SEND)

18–1B
  Size of the network layer packet (NLP)

1C–1F
  RPNCB address

ARBB entry for HPR trace option

Entry:
  ARBB
VIT option:
   HPR

Event:
   All RTP connections using Responsive Mode ARB and ARBB active.

VIT processing module:
   ISTITCHA

Control is returned to:
   The module that issued the INTRACE macro.

This trace record is written when RTP connections are using Responsive Mode ARB and ARBB option is active. The trace record has six instances included as follows.

Instance 1:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBB&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08</td>
<td>ARB rate status (MRP_NLH_RATE)</td>
</tr>
<tr>
<td>09</td>
<td>Rate of last ARB measurement sent (ARB_LAST_RATE)</td>
</tr>
<tr>
<td>0A–0B</td>
<td>0</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Delay Change Sum (DCS) — accumulated network delay in milliseconds (ARB_ACCUM_QTIME)</td>
</tr>
<tr>
<td>10–13</td>
<td>Current delay change (ARBLV_CURR_QTIME)</td>
</tr>
<tr>
<td>14–17</td>
<td>Elapsed time in milliseconds since the last ARB rate request (ARBLV_ETIME_IN_MS)</td>
</tr>
<tr>
<td>18–1B</td>
<td>ARB request sender's measurement interval (SMI) in microseconds (THD_ARB_CURR_MEASURE_INT)</td>
</tr>
</tbody>
</table>
**Byte (hex)**

**Contents**

**00–03**
Record ID: C"ARBB"

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
0

**06**
Trace instance

**07**
0

**08**
MRP_RCV_RATE_STAT

**09**
Rate of last ARB measurement sent (ARB_LAST_RATE)

**0A–0B**
0

**0C–0F**
Delay Change Sum (DCS) — accumulated network delay in milliseconds (ARB_ACCUM_QTIME)

**10–13**
DCS boundary in milliseconds dividing region 1 and 2 (ARB_GREEN_THRESHOLD)

**14–17**
DCS boundary in milliseconds dividing region 2 and 3 (ARB_MAX_QTIME_RETN)

**18–1B**
DCS boundary in milliseconds dividing region 3 and 4 (ARB_MAX_QTIME_RETN * ACM_UPPER_LIM)

**1C–1F**
RPNCB address

**Note:** The following ARB rate replies are returned to the sender of the ARB rate request depending on the region in which the DCS lies:

<table>
<thead>
<tr>
<th>DCS in region</th>
<th>Rate reply returned</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NORMAL</td>
</tr>
<tr>
<td>2</td>
<td>RESTRAINT</td>
</tr>
<tr>
<td>3</td>
<td>SLOWDOWN 1 or SLOWDOWN 2</td>
</tr>
</tbody>
</table>
### DCS in region

**Rate reply returned**

4

RESTRAINT

Instance 3:

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARBB</td>
<td>ID</td>
<td>0</td>
<td>INSTANCE</td>
<td>FLAGS</td>
<td>CUT</td>
<td>00</td>
<td>ALLOWED</td>
<td>SENDING</td>
<td>RATE</td>
<td>INCREMENT</td>
<td>INCREMENT</td>
<td>TREND</td>
<td>DECREMENT</td>
<td>TREND</td>
<td>RPNCB</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03

Record ID: C“ARBB”

04

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.

05

0

06

Trace instance

07

ARB flags passed on the ARB segment (THD_ARB_FLAGS)

08

Indicates size of rate cut for a SLOWDOWN (send_rate = send_rate — (send_rate / acm_cut_rate)) (ACM_CUT_RATE)

09

Flag values used for ARB processing (ARB_FLAGS)

0A–0B

0

0C–0F

ARB allowed sending rate in kbps (ARB_ALLOW_SEND_RATE)

10–13

Current increment value for allowed send rate in kbps (ARB_CURR_RATE_INC)

14–17

Number of consecutive sending rate increments (ARB_INC_TREND)

18–1B

Number of consecutive sending rate decrements (ARB_DEC_TREND)

1C–1F

RPNCB address

Instance 4:
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03 | Record ID: C"ARBB"
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05 | 0
| 06 | Trace instance
| 07 | 0
| 08 | Flag values used for ARB processing (ARB_FLAGS)
| 09 | 0
| 0A–0B | 0
| 0C–0F | Flag values used for ARB processing in kbps (ARB_CURR_RATE_INC)
| 10–13 | Number of consecutive sending rate increments (ARB_INC_TREND)
| 14–17 | Number of consecutive sending rate decrements (ARB_DEC_TREND)
| 18–1B | Receiving rate of the sender in kbps (THD_ARB_REC_RATE)
| 1C–1F | RPNCB address

Instance 5:
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBB&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>ARB allowed sending rate in kbps (ARB_ALLOW_SEND_RATE)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Maximum burst time allowed in milliseconds (ARB_BT_MAX)</td>
</tr>
<tr>
<td>10–13</td>
<td>Minimum burst time allowed in milliseconds (ARB_BT_MIN)</td>
</tr>
<tr>
<td>14–17</td>
<td>Minimum increment value for allowed send rate in kbps (ARB_MIN_RATE_INC)</td>
</tr>
<tr>
<td>18–1B</td>
<td>Shared link rate in kbps (ARB_MAX_SEND_RATE)</td>
</tr>
<tr>
<td>1C–1F</td>
<td>RPNCB address</td>
</tr>
</tbody>
</table>

Instance 6:
00–03
Record ID: C"ARBB"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Trace instance

07
0

08–0B
Smoothed Round Trip Time in milliseconds (RPN_SRTTs)

0C–0F
Short Request timer in milliseconds (RPN_SHORT_REQ_T)

10–13
Counter for adjustment of Base ARB measurement interval (ARB_TIME_ADJUST_M_INT)

14–17
Interval in milliseconds for ARB measurement requests (ARB_MEASURE_INT)

18–1B
0

1C–1F
RPN CB address

**ARBR entry for HPR trace option**

**Entry:**
ARBR

**VIT option:**
HPR

**Event:**
All RTP connections using Responsive Mode ARB and ARBR active.

**VIT processing module:**
ISTITCHA

**Control is returned to:**
The module that issued the INTRACE macro.

This trace record is written when RTP connections are using Responsive Mode ARB and ARBR option is active. The trace record has 38 instances included as follows.

**Instance 0:**

```
0 1 2 3 4 5 6 7 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1}
Byte (hex)
Contents
00–03
Record ID: C"ARBR"
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
0
06
Trace instance
07
0
08–09
Aggressiveness indicator, the range is 0 (most aggressive) – 1000 (least aggressive) (ARB2_FCURVE_NOW)
0A–0B
Determines how fast F_curve value increases (ARB2_ALPHA_NOW)
0C–0F
Number of bytes drained from leaky bucket during last interval (DRAIN_IN_BYTES)
10–13
ARB allowed sending rate in kbps (ARB_ALLOW_SEND_RATE)
14–17
Target sending rate in kbps (ARB2_TARGET_RATE)
18–1B
Number of bytes that can be placed in the leaky bucket (ARB2_LOSS_BUCKET_CONTENTS)
1C–1F
RPNCB address

Instance 1:

| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|

Byte (hex)
Contents
00–03
Record ID: C"ARBR"
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
0
06
Trace instance
ARBR flags passed on the ARB segment (THD_ARB_FLAGS)

ARB request correlator sent on ARB segment (THD_ARB_CORRELATOR)

Current ARB request correlator (ARB2_REQUEST_CORRELATOR)

Local request correlator working field (ARBLV_REQ_CORRELATOR)

Previous ARB request correlator (ARB2_LAST_REQ_CORRELATOR)

Current receiver threshold in microseconds (ARB2_RCVR_THRESHOLD)

Minimum Receiver threshold in microseconds (ARB2_RCVR_THRESHOLD_MIN)

Maximum Receiver threshold in microseconds (ARB2_RCVR_THRESHOLD_MAX)

Instance 2:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"ARBR"
| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05         | 0        |
| 06         | Trace instance |
| 07         | 0        |
| 08         | 0        |
| 09         | 0        |
0A–0B 0
0C–0D Previous ARB request correlator (ARB2_LAST_REQ_CORRELATOR)
0E–0F Local ARB parity bit (ARBLV_TEMP_ARB_PARITY)
10–11 Local last ARB parity bit (ARBLV_TEMP_LAST_RCVD_PARITY)
12–13 Local request correlator working field (ARBLV_TEMP_REQ_CORRELATOR)
14–17 0
18–1B 0
1C–1F RPNCB address

Instance 3:

<table>
<thead>
<tr>
<th></th>
<th>ARBR</th>
<th>ID</th>
<th>0</th>
<th>INSTANCE</th>
<th>0</th>
<th>IDLE TIME</th>
<th>CURRENT TIME</th>
<th>LAST DATA RECEIVED</th>
<th>IDLE TIME THRESHOLD</th>
<th>MAX SEND RATE</th>
<th>RPNCB ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>C&quot;ARBR&quot;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>08–0B</td>
<td>Amount of time idle in milliseconds since last data received (ARBLV_IDLE_TIME)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0C–0F</td>
<td>Current time stamp value (ARBLV_CURR_TIME)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10–13</td>
<td>Time stamp value of last data received (ARB2_LAST_DATA_RCVD)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14–17</td>
<td>Idle time in milliseconds threshold (INC_IDLE_TIME_THRESH(ARBLV_MAXSNDI))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Shared link rate table index (ARBLV_MAXSNDI)

RPNCB address

Instance 4:

<table>
<thead>
<tr>
<th>ARBR</th>
<th>ID</th>
<th>INSTANCE</th>
<th>ELAPSED TIME</th>
<th>POTENTIAL DCS</th>
<th>POTENTIAL ERROR COUNT</th>
<th>DELAY CHANGE BAR</th>
<th>DC STANDARD DEVIATION</th>
<th>RPNCB ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

00–03

Record ID: C"ARBR"

04

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05

0

06

Trace instance

07

Flag values used for ARB processing (ARB_FLAGS)

08–0B

Elapsed time in microseconds since last ARB status request (ARBLV_ETIME_IN_MS)

0C–0F

Potential DCS — accumulated network delay in microseconds (ARB2_POT_ACCUM_QTIME)

10–13

Potential Count of errors detected (ARB2_POT_ERROR_COUNT)

14–17

Delay Change bar value in microseconds (ARB2_DC_BAR)

18–1B

Delay Change standard deviation in microseconds (ARB2_DC_SDEV)

1C–1F

RPNCB address

Instance 5:
ARBR I

N
S
T
A
C
E
D
I
D
P
A
R
V
L
A
G
S
F
L
A
G
S
0
I
D
T
A
C
E
D
I
D
P
A
R
V
L
A
G
S
F
L
A
G
S

Byte (hex)

Contents

00–03
Record ID: C"ARBR"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Trace instance

07
0

08–0B
ARB2_POT_DC_BAR (Potential Delay Change bar value in microseconds)

0C–0F
ARB2_POT_DC_SDEV (Potential Delay Change standard deviation in microseconds)

10–13
ARB2_POT_DC_THRESH (Potential Delay Change threshold in microseconds)

14–17
ARB2_DC_SDEV (Delay Change standard deviation in microseconds)

1C–1F
RPNCB address

Instance 6:

Byte (hex)

Contents
00–03
Record ID: C"ARBR"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Trace instance

07
ARB_FLAGS (Flag values used for ARB processing)

08–0B
ARB2_DCS_BAR (Delay Change Sum bar value in microseconds)

0C–0F
ARBLV_AVG_COEFF (Average Coefficient based on elapsed time between ARB status requests)

10–13
ARB2_POT_ACCUM_QTIME (Potential DCS — accumulated network delay in microseconds)

14–17
ARB2_DCS_SQUARED_BAR (Delay Change Sum bar value squared in milliseconds)

18–1B
ARB2_ACCUM_QTIME_MAX (Maximum Delay Change over last 30 ARB status requests in microseconds)

1C–1F
RPNCB address

Instance 7:

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARBR</td>
<td>I</td>
<td>D</td>
<td>0</td>
<td>INSTANCE</td>
<td>0</td>
<td>DCS</td>
<td>DC_THRESHOLD</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>RPNCB ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
Contents

00–03
Record ID: C"ARBR"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Trace instance

07
0

08–0B
Delay Change Sum(DCS) — accumulated network delay in microseconds (ARBLV_CURR_QTIME)
<table>
<thead>
<tr>
<th>Bytes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0C–0F</td>
<td>Delay Change threshold in microseconds (ARB2_DC_THRESH)</td>
</tr>
<tr>
<td>10–13</td>
<td>0</td>
</tr>
<tr>
<td>14–17</td>
<td>0</td>
</tr>
<tr>
<td>18–1B</td>
<td>0</td>
</tr>
<tr>
<td>1C–1F</td>
<td>RPNCB address</td>
</tr>
</tbody>
</table>

**Instance 8:**

| 0 | 0 | 0 | 1 | 2 | 0 | 4 | 0 | 6 | 0 | 7 | 0 | 9 | 0 | A | 0 | B | 0 | C | 0 | D | 0 | E | 0 | F | 1 | 1 | 1 | 1 | 2 | 3 | 1 | 4 | 1 | 5 | 6 | 1 | 7 | 1 | 9 | 1 | A | 1 | B | 1 | C | 1 | D | 1 | E | 1 | F |

**Byte (hex) Contents**

**00–03**
- Record ID: C“ARBR”

**04**
- ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.

**05**
- 0

**06**
- Trace instance

**07**
- Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS)

**08–0B**
- Potential Delay Change Sum (DCS) bar value in microseconds (ARB2_POT_DCS_BAR)

**0C–0F**
- Potential DCS bar value squared in milliseconds (ARB2_POT_DCS_SQUARED_BAR)

**10–13**
- Delay Change Sum variance (ARBLV_DCS_VARIANCE)

**14–17**
- Maximum Delay Change Sum recorded in microseconds (ARB2_DCS_MAXOVERALL)

**18–1B**
- Maximum Delay Change in microseconds over last 30 ARB status requests (ARB2_ACCUM_QTIME_MAX)

**1C–1F**
- RPNCB address

**Instance 9:**
**Byte (hex) Contents**

**00–03**
Record ID: C"ARBR"

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
0

**06**
Trace instance

**07**
0

**08**
Flag values used for ARB processing (ARB_FLAGS)

**09**
Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS)

**0A**
Potential flag values used for Responsive Mode ARB processing (ARB2_POT_PHASE_FLAGS)

**0B**
Global ARB segment flags (MRP_ARB_FLAGS)

**0C–0F**
Potential DCS — accumulated network delay in milliseconds (ARB2_POT_ACCUM_QTIME)

**10–13**
Last Delay Change Sum(DCS) in microseconds (ARBLV_LAST_ACCUM_QTIME)

**14–17**
Receiver Threshold value in microseconds (ARB2_RCVR_THRESHOLD)

**18–1B**
Shared link rate in kbps (ARB_MAX_SEND_RATE)

**1C–1F**
RPN CB address

**Instance 10:**
Byte (hex)
Contents
00–03
  Record ID: C"ARBR"
04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
  0
06
  Trace instance
07
  0
08
  Flag values used for ARB processing (ARB_FLAGS)
09
  Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS)
0A
  Potential flag values used for Responsive Mode ARB processing (ARB2_POT_PHASE_FLAGS)
0B
  Global ARB segment flags (MRP_ARB_FLAGS)
0C–0F
  Smoothed sending rate in kbps (ARB2_SMOOTHED_RATE)
10–13
  Elapsed time in microseconds since last ARB status request (ARBLV_ETIME_IN_MS)
14–17
  Potential maximum (ARB2_POT_MAX_DCS_WINDOW)
18–1B
  Maximum Delay in microseconds Change Sum recorded (ARB2_POT_DCS_MAX_OVERALL)
1C–1F
  RPN CB address

Instance 11:
Byte (hex)
Contents
00–03
Record ID: C"ARBR"
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
0
06
Trace instance
07
0
08–0B
Last current byte received (RPN_LAST_BYTE_RCV)
0C–0F
Byte received during last ARB status request (ARB2_LAST_BYTE_RCV)
10–13
Bits of data received over last interval (ARBLV_BITS_OF_DATA)
14–17
Potential smoothed sending rate in kbps (ARB2_POT_SMOOTHED_RATE)
18–1B
Maximum window size over last 10 ARB status requests (ARB2_POT_MAX_DCS_WINDOW)
1C–1F
RPNCB address

Instance 12:

Byte (hex)
Contents
00–03
  Record ID: C"ARBR"

04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
  0

06
  Trace instance

07
  0

08–0B
  Potential Delay Change Sum bar value in microseconds (ARB2_POT_DCS_BAR)

0C–0F
  Present Receiver Threshold maximum in microseconds (ARB2_PRESENT_THR_MAX)

10–13
  Computed threshold variance in microseconds for this link (ARB2_ANTICIPATED_VAR)

14–17
  Fraction that indicates number of increments related to window size (value is 0–1000) (ARB2_N_FRACTION)

18–1B
  Fraction that indicates number of increments over time (value is 0–100) (ARB2_N_TREND)

1C–1F
  RPNCB address

Instance 13:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBR&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Dictates how long the computer stays active (values 0–100) (ARB2_RECOVERY_FACTOR)</td>
</tr>
</tbody>
</table>
0C–0F
  Potential Delay Change Sum bar value in microseconds (ARB2_POT_DCS_BAR)

10–13
  Original Threshold maximum in microseconds (ARB2_ORIG_THR_MAX)

14–17
  Potential Present Receiver Threshold maximum in microseconds (ARB2_POT_PRESENT_THR_MAX)

18–1B
  Potential threshold variance in microseconds for this link (ARB2_POT_ANTICIPATED_VAR)

1C–1F
  RPNCB address

Instance 14:

<table>
<thead>
<tr>
<th>ARBR</th>
<th>I D</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>00</td>
<td>00</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>1A</td>
<td>1B</td>
<td>1C</td>
<td>1D</td>
<td>1E</td>
<td>1F</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
  Contents

00–03
  Record ID: C"ARBR"

04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
  0

06
  Trace instance

07
  0

08–0B
  Dictates how long the computer stays active (values 0–100) (ARB2_RECOVERY_FACTOR)

0C–0F
  Potential Present Receiver Threshold minimum in microseconds (ARB2_POT_PRESENT_THR_MIN)

10–13
  Potential Present Receiver Threshold maximum in microseconds (ARB2_POT_PRESENT_THR_MAX)

14–17
  Potential threshold variance in microseconds for this link (ARB2_POT_ANTICIPATED_VAR)

18–1B
  Original Threshold maximum in microseconds (ARB2_ORIG_THR_MAX)

1C–1F
  RPNCB address

Instance 15:
<table>
<thead>
<tr>
<th>ARBR</th>
<th>ID</th>
<th>INSTANCE</th>
<th>POTENTIAL MIN THRESHOLD</th>
<th>POTENTIAL MAX THRESHOLD</th>
<th>POTENTIAL THRESHOLD VARIANCE</th>
<th>SAVED MIN THRESHOLD</th>
<th>SAVED MAX THRESHOLD</th>
<th>RPNCB ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00–03**

Record ID: C"ARBR"

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

0

**06**

Trace instance

**07**

0

**08–0B**

Potential Present Receiver Threshold minimum in microseconds (ARB2_POT_PRESENT_THR_MIN)

**0C–0F**

Potential Present Receiver Threshold maximum in microseconds (sARB2_POT_PRESENT_THR_MAX)

**10–13**

Potential threshold variance in microseconds for this link (ARB2_POT_ANTICIPATED_VAR)

**14–17**

Saved Receiver Threshold minimum in microseconds (ARB2_SAVED_THR_MIN)

**18–1B**

Saved Receiver Threshold maximum in microseconds (ARB2_SAVED_THR_MAX)

**1C–1F**

RPNCB address

**Instance 16:**

<table>
<thead>
<tr>
<th>ARBR</th>
<th>ID</th>
<th>INSTANCE</th>
<th>WINDOW SIZE</th>
<th>INCREMENTS TO WINDOW SIZE</th>
<th>NUMBER INCREMENTS</th>
<th>LAST FRACTION</th>
<th>INCREMENTS OVER TIME</th>
<th>RPNCB ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**
00–03
Record ID: C"ARBR"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Trace instance

07
0

08–0B
Number of ARB status requests used in calculating Receiver thresholds (ARB2_WINDOW_SIZE)

0C–0F
Fraction that indicates number of increments related to window size (value 0–1000) (ARB2_N_FRACTION)

10–13
Number of consecutive sending rate increments (ARB_INC_TREND)

14–17
Last N_Fraction (value is 0–1000) (ARBLV_N_FRACTION_LAST)

18–1B
Fraction that indicates number of increments over time (value is 0–100) (ARB2_N_TREND)

1C–1F
RPNCB address

Instance 17:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| ARBR | ID | 0 | Instance | 0 | WINDOW SIZE | NUMBER INCREMENTS | INCREMENTS OVER TIME | POTENTIAL MAX THRESHOLD | ORIGINAL MAX THRESHOLD | RPNCB ADDRESS |

Byte (hex)
Contents

00–03
Record ID: C"ARBR"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Trace instance

07
0

08–0B
Number of ARB status requests used in calculating Receiver thresholds (ARB2_WINDOW_SIZE)
**0C–0F**
Number of consecutive sending rate increments (ARB_INC_TREND)

**10–13**
Fraction that indicates number of increments over time (value 0–100) (ARB2_N_TREND)

**14–17**
Potential Present Receiver Threshold maximum in microseconds (ARB2_POT_PRESENT_THR_MAX)

**18–1B**
Original Threshold maximum in microseconds (ARB2_ORIG_THR_MAX)

**1C–1F**
RPNCB address

**Instance 18:**

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBR&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Potential Receiver Threshold value in microseconds (ARB2_POT_RCVR_THRESHOLD)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Minimum Receiver Threshold value in microseconds (ARB2_RCVR_THRESHOLD_MIN)</td>
</tr>
<tr>
<td>10–13</td>
<td>Maximum Receiver Threshold value in microseconds (ARB2_RCVR_THRESHOLD_MAX)</td>
</tr>
<tr>
<td>14–17</td>
<td>Potential Present Threshold maximum in microseconds (ARB2_POT_PRESENT_THR_MAX)</td>
</tr>
<tr>
<td>18–1B</td>
<td>Original Threshold maximum in microseconds (ARB2_ORIG_THR_MAX)</td>
</tr>
<tr>
<td>1C–1F</td>
<td>RPNCB address</td>
</tr>
</tbody>
</table>

**Instance 19:**
### Byte (hex)
**Contents**

<table>
<thead>
<tr>
<th>Byte</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBR&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08</td>
<td>ARB request correlator sent on ARB segment (THD_ARB_CORRELATOR)</td>
</tr>
<tr>
<td>09</td>
<td>Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS2)</td>
</tr>
<tr>
<td>0A–OB</td>
<td>Current ARB request correlator (ARB2_REQUEST_CORRELATOR)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Shared link rate in kbps (ARB_MAX_SEND_RATE)</td>
</tr>
<tr>
<td>10–13</td>
<td>ARB allowed sending rate in kbps (ARB_ALLOW_SEND_RATE)</td>
</tr>
<tr>
<td>14–17</td>
<td>Target sending rate in kbps (ARB2_TARGET_RATE)</td>
</tr>
<tr>
<td>18–1B</td>
<td>Smoothed actual sending rate in kbps (ARB2_SMOOTH_ACTUAL_RATE)</td>
</tr>
<tr>
<td>1C–1F</td>
<td>RPN CB address</td>
</tr>
</tbody>
</table>

**Instance 20:**
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03     | Record ID: C"ARBR"
| 04        | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05        | 0
| 06        | Trace instance
| 07        | 0
| 08–0B     | Smoothed Round Trip Time in milliseconds (RPN_SRTT)
| 0C–0D     | ARB Burst interval in milliseconds (ARB2_BURST_INTVL)
| 0E–0F     | Aggressiveness indicator, range is 0 (most aggressive) – 1000 (least aggressive) (ARB2_FCURVE_NOW)
| 10–11     | Determines how fast F_curve value increases (ARB2_ALPHA_NOW)
| 12–13     | Fraction of the current send/target rate (value 0-1000) (ARB2_DOWN_GAMMA)
| 14–15     | Number of aggressive steps taken (ARB2_AGGRESSIVE_STEPS)
| 16–17     | 0
| 18–1B     | 0
| 1C–1F     | RPN CB address

Instance 21:
Byte (hex)  
Contents  
00–03  
Record ID: C"ARBR"

04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05  
0

06  
Trace instance

07  
0

08  
ARB request correlator sent on ARB segment (THD_ARB_CORRELATOR)

09  
Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS2)

0A  
0

0B  
0

0C–0D  
Current ARB request correlator (ARB2_REQUEST_CORRELATOR)

0E–0F  
0

10–13  
0

14–17  
0

18–1B  
0

1C–1F  
RPNCB address

Instance 22:
Byte (hex)
Contents

00–03
Record ID: C“ARBR“

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X‘FF’.

05
0

06
Trace instance

07
0

08–0B
0

0C–0F
ARB allowed sending rate in kbps (ARB_ALLOW_SEND_RATE)

10–13
Target sending rate in kbps (ARB2_TARGET_RATE)

14–15
Number of consecutive slowdowns while using above 10% of shared link rate (ARB2_SLOW_COUNT)

16–17
Number of consecutive ARB status requests while in conservative phase (ARB2_CONSERVATIVE_CNT)

18–1B
0

1C–1F
RPNCB address

Instance 23:

Byte (hex)
Contents
00–03
Record ID: C"ARBR"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Trace instance

07
0

08
Flag values used for ARB processing (ARB_FLAGS)

09
ARB flags passed on the ARB segment (THD_ARB_FLAGS)

0A–0B
Aggressiveness indicator, range is 0 (most aggressive) – 1000 (least aggressive) (ARB2_FCURVE_NOW)

0C–0F
Number of bits in burst size based on link rate (ARBLV_BURST_SIZE_MIN)

10–13
Current minimum burst size (ARB2_BURST_SIZE_MIN)

14–17
Fractional value used to determine rate increase or decrease (ARBLV_GAMMA_VALUE)

18–1B
Elapsed time in milliseconds since last ARB status reply received (ARBLV_ETIME_IN_MS)

1C–1F
RPNCB address

Instance 24:

<table>
<thead>
<tr>
<th>ARBR</th>
<th>ID</th>
<th>0</th>
<th>INSTA</th>
<th>0</th>
<th>BI</th>
<th>DE</th>
<th>CURR</th>
<th>SET</th>
<th>RA</th>
<th>CH</th>
<th>MG</th>
<th>RPNC</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>A</td>
<td>B</td>
<td>0C</td>
</tr>
<tr>
<td>0F</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>A</td>
<td>B</td>
<td>1C</td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

00–03
Record ID: C"ARBR"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Trace instance
07
0
08–0B
Bits of data sent over the last interval (ARBLV_BITS_OF_DATA)
0C–0F
Number of bits in burst size based on link rate (ARBLV_BURST_SIZE_MIN)
10–13
Current sending rate of last interval (ARBLV_CURRENT_SEND_RATE)
14–17
Fractional value used to determine rate increase or decrease (ARBLV_GAMMA_VALUE)
18–1B
Minimum Gamma value (ARBLV_GAMMA_MIN)
1C–1F
RPNCB address

Instance 25:

<table>
<thead>
<tr>
<th>ARBR</th>
<th>ID</th>
<th>INSTANCE</th>
<th>FLAGS</th>
<th>NEXT_BYTE</th>
<th>LAST_BYTE_SENT</th>
<th>SMOOTH_SEND_RATE</th>
<th>MAX_SEND_RATE</th>
<th>RPNCB_ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
Contents
00–03
Record ID: C"ARBR"
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
0
06
Trace instance
07
0
08
Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS)
09
Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS2)
0A–0B
Number of instances where allowed sending rate is above shared link rate (ARB2_HIGH_STEPS)
0C–0F
Sequence number of next byte to be transmitted (RPN_NEXT_BYTE_XMIT)
10–13
Last byte transmitted during previous ARB rate reply (ARB2_LAST_BYTE_SENT)
### VTAM Internal Trace (VIT) Record Descriptions

#### Instance 26:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBR&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08</td>
<td>Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS)</td>
</tr>
<tr>
<td>09</td>
<td>Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS2)</td>
</tr>
<tr>
<td>0A</td>
<td>Shared link rate table index (ARBLV_MAXSNDI)</td>
</tr>
<tr>
<td>0B</td>
<td>0</td>
</tr>
<tr>
<td>0C–0D</td>
<td>Number of instances where allowed sending rate is above &quot;shared link rate&quot; (ARB2_HIGH_STEPS)</td>
</tr>
<tr>
<td>0E–0F</td>
<td>0</td>
</tr>
<tr>
<td>10–13</td>
<td>Shared link rate in kbps (ARB_MAX_SEND_RATE)</td>
</tr>
<tr>
<td>14–17</td>
<td>Smoothed actual sending rate in kbps (ARB2_SMOOTH_ACTUAL_RATE)</td>
</tr>
<tr>
<td>18–1B</td>
<td>0</td>
</tr>
</tbody>
</table>
### Instance 27:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBR&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–09</td>
<td>Number of aggressive steps taken (ARB2_AGGRESSIVE_STEPS)</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Number of instances where allowed sending rate is above &quot;shared link rate&quot; (ARB2_HIGH_STEPS)</td>
</tr>
<tr>
<td>0C–0D</td>
<td>Number of consecutive slowdowns while using above 10% of shared link rate (ARB2_SLOW_COUNT)</td>
</tr>
<tr>
<td>0E–0F</td>
<td>Number of consecutive ARB status requests while in conservative phase (ARB2_CONSERVATIVE_CNT)</td>
</tr>
<tr>
<td>10–11</td>
<td>Aggressiveness indicator, range is 0 (most aggressive) – 1000 (least aggressive) (ARB2_FCURVE_NOW)</td>
</tr>
<tr>
<td>12–13</td>
<td>0</td>
</tr>
<tr>
<td>14–17</td>
<td>Fractional value used to determine rate increase or decrease (ARBLV_GAMMA_VALUE)</td>
</tr>
<tr>
<td>18–1B</td>
<td>Fraction of the current send/target rate (value 0 – 1000) (ARB2_DOWN_GAMMA)</td>
</tr>
<tr>
<td>1C–1F</td>
<td>RPNCB address</td>
</tr>
</tbody>
</table>

### Instance 28:

For Instance 28, the data is not provided in the document.
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03     | Record ID: C"ARBR"
| 04        | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05        | 0
| 06        | Trace instance
| 07        | 0
| 08        | Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS)
| 09        | Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS2)
| 0A–0B     | Number of consecutive ARB status requests while in conservative phase (ARB2_CONSERVATIVE_CNT)
| 0C–0F     | Target for sending rate in kbps (ARBLV_TARGET_RATE)
| 10–13     | Current sending rate in kbps (ARBLV_CURR_RATE)
| 14–17     | Shared link rate in kbps (ARB_MAX_SEND_RATE)
| 18–1B     | 0
| 1C–1F     | RPNCB address

Instance 29:
Byte (hex)  
Contents  
00–03  
   Record ID: C"ARBR"  
04  
   ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.  
05  
   0  
06  
   Trace instance  
07  
08–09  
   Number of consecutive ARB status requests while in conservative phase (ARB2_CONSERVATIVE_CNT)  
0A–0B  
   0  
0C–0F  
   Random value in the range 0–16 (ITSUDATA)  
10–13  
   Smoothed Round Trip Time in milliseconds (RPN_SRTT)  
14–17  
   Smoothed Round Trip Time with a lower bound of 100 milliseconds (ARBLV_X)  
18–1B  
   Random Threshold value used to determine when to leave conservative phase (ARB2_RANDOM_THRESH)  
1C–1F  
   RPNCB address  

Instance 30:

<table>
<thead>
<tr>
<th>ARBR</th>
<th>ID</th>
<th>INSTANCE</th>
<th>FLAG</th>
<th>CPU RATE</th>
<th>SLOW COUNT</th>
<th>TARGET SEND RATE</th>
<th>CURRENT SEND RATE</th>
<th>GAMMA VALUE</th>
<th>RPNCB ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Byte (hex)  
Contents  
00–03  
   Record ID: C"ARBR"  
04  
   ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.  
05  
   0  
06  
   Trace instance
Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS)

Flag values used for ARB processing (ARB_FLAGS)

Aggressiveness indicator, range is 0 (most aggressive) – 1000 (least aggressive) (ARB2_FCURVE_NOW)

Determines how fast F_curve value increases (ARB2_ALPHA_NOW)

Number of consecutive slowdowns while using above 10% of shared link rate (ARB2_SLOW_COUNT)

Target for sending rate in kbps (ARBLV_TARGET_RATE)

Current sending rate in kbps (ARBLV_CURR_RATE)

Fractional value used to determine rate increase or decrease (ARBLV_GAMMA_VALUE)

RPNCB address

Instance 31:

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03

Record ID: C"ARBR"

04

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05

0

06

Trace instance

07

0

08–0B

0

0C–0F

Local variable used to compute decreased target rate (ARBLV_X)

VTAM internal trace (VIT) record descriptions 239
10–13
Local variable used to compute decreased target rate (ARBLV_B)

14–17
Local variable used to compute decreased target rate (ARBLV_C)

18–1B
Fractional amount of shared link rate used (value 0-1000) (ARBLV_SHARED_RATE_FRACTION)

1C–1F
RPNCB address

Instance 32:

```
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBR&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>Flag values used for Responsive Mode ARB processing (ARB2_PHASE_FLAGS)</td>
</tr>
<tr>
<td>08–09</td>
<td>Number of consecutive slowdowns while using above 10% of shared link rate (ARB2_SLOW_COUNT)</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Determines how fast F_curve value increases (ARB2_ALPHA_NOW)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Target for sending rate in kbps (ARBLV_TARGET_RATE)</td>
</tr>
<tr>
<td>10–13</td>
<td>Local variable used to compute decreased target rate (ARBLV_B)</td>
</tr>
<tr>
<td>14–17</td>
<td>Local variable used to compute decreased target rate (ARBLV_C)</td>
</tr>
<tr>
<td>18–1B</td>
<td>0</td>
</tr>
<tr>
<td>1C–1F</td>
<td>RPNCB address</td>
</tr>
</tbody>
</table>
```

Instance 33:
### VTAM Internal Trace (VIT) Record Descriptions

#### Byte (hex)
**Contents**

| 00–03 | Record ID: C"ARBR"
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05 | 0
| 06 | Trace instance
| 07 | 0
| 08–0B | Minimum number of steps to reach target rate (multiplied by 1000) (ARBLV_KUP_MIN)
| 0C–0F | Maximum number of steps to reach target rate (multiplied by 1000) (ARBLV_KUP_MAX)
| 10–13 | Number of steps to be taken to reach target rate (multiplied by 1000) (ARBLV_KUP_NOW)
| 14–17 | Accumulated trans time forward in microseconds (ARB_ACCUM_TTIME_FRW)
| 18–1B | Accumulated trans time reverse in microseconds (ARB_ACCUM_TTIME_REV)
| 1C–1F | RPNCB address

#### Instance 34:

| 00–03 | Record ID: C"ARBR"
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05 | 0
| 06 | Trace instance
| 07 | 0
| 08–0B | Minimum number of steps to reach target rate (multiplied by 1000) (ARBLV_KUP_MIN)
| 0C–0F | Maximum number of steps to reach target rate (multiplied by 1000) (ARBLV_KUP_MAX)
| 10–13 | Number of steps to be taken to reach target rate (multiplied by 1000) (ARBLV_KUP_NOW)
| 14–17 | Accumulated trans time forward in microseconds (ARB_ACCUM_TTIME_FRW)
| 18–1B | Accumulated trans time reverse in microseconds (ARB_ACCUM_TTIME_REV)
| 1C–1F | RPNCB address

---

**VTAM internal trace (VIT) record descriptions** 241
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Flag values used for ARB processing (ARB_FLAGS)

Burst interval, which is the maximum of SRTT and 100 milliseconds (ARBLV_BURST_INTERVAL)

Number of steps to be taken to reach target rate (multiplied by 1000) (ARBLV_KUP_NOW)

Minimum ARB burst time duration in milliseconds (ARB_BT_MIN)

Number of bits in burst size based on link rate (ARBLV_BURST_SIZE_MIN)

RPNCB address

Instance 35:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: &quot;ARBR&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Smoothed Round Trip Time in milliseconds (RPN_SRTT)</td>
</tr>
</tbody>
</table>
0C–0F
  Smoothed deviation of Round Trip Time in milliseconds (RPN_SMOOTH_DEV)

10–13
  Short Request time in milliseconds (RPN_SHORT_REQ_T)

14–17
  Current Round Trip Time in milliseconds (ARB2_CURRENT_RTT)

18–1B
  REFIFO time value in milliseconds (RPN_REFIFO_T)

1C–1F
  RPN CB address

Instance 36:

| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| ARBR | ID | INSTANCE | FCURVE | IDLE | TIME | ALLOWED | SEND | RATE | TARGET | SEND | RATE | CURRENT | ROUND | TRIP | RPNCB | ADDRESS |

Byte (hex)
  Contents

00–03
  Record ID: C"ARBR"

04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
  0

06
  Trace instance

07
  0

08–09
  Aggressiveness indicator, range is 0 (most aggressive) – 1000 (least aggressive)
  (ARB2_FCURVE_NOW)

0A–0B
  0

0C–0F
  Amount of time in milliseconds idle since last data transmitted (ITSUDATA)

10–13
  ARB allowed sending rate in kbps (ARB_ALLOW_SEND_RATE)

14–17
  Target sending rate in kbps (ARB2_TARGET_RATE)

18–1B
  Current Round Trip Time in milliseconds (ARB2_CURRENT_RTT)

1C–1F
  RPNCB address
Instance 37:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ARBR&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–09</td>
<td>ARB Burst interval in milliseconds (ARB2_BURST_INTVL)</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Aggressiveness indicator, range is 0 (most aggressive) – 1000 (least aggressive) (ARB2_FCURVE_NOW)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Shared link rate in kbps (ARB_MAX_SEND_RATE)</td>
</tr>
<tr>
<td>10–13</td>
<td>Smoothed actual sending rate in kbps (ARB2_SMOOTH_ACTUAL_RATE)</td>
</tr>
<tr>
<td>14–17</td>
<td>Target sending rate in kbps (ARB2_TARGET_RATE)</td>
</tr>
<tr>
<td>18–1B</td>
<td>Number of bits in burst size based on link rate (ARBLV_BURST_SIZE_MIN)</td>
</tr>
<tr>
<td>1C–1F</td>
<td>RPNCB address</td>
</tr>
</tbody>
</table>

**AREL entry for abend RELSTORE**

**Entry:**
AREL

**VIT option:**
SMS

**Event:**
Abend RELSTORE

**VIT processing module:**
ISTRACSM
Control is returned to:
ISTORAPR

This trace record identifies the buffers that are released by VTAM when a VTAM application program is terminated.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;AREL&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Control block ID index value</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>PST address</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of buffer being released</td>
</tr>
<tr>
<td>10–13</td>
<td>Address of the caller of the abend RELSTORE routine</td>
</tr>
<tr>
<td>14–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

ARP or ARS entry for RTP session data reply (Part 1)

Entry:
ARP or ARS

VIT option:
HPR

Event:
Any session data reply is sent over an RTP connection using Responsive Mode ARB

VIT processing module:
ISTITCHR

Control is returned to:
ISTRPCAR

This trace record is written when a session data reply is sent across an RTP connection using Responsive Mode ARB, or when simulating a slowdown reply. It contains statistical information about the session data.
**Byte (hex)**

**Contents**

**00–03**

Record ID:
- C“ARP” for real reply
- C“ARS” for simulated reply

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.

**05**

0

**06–07**

Phase flags (ARB2_PHASE_FLAGS, ARB2_PHASE_FLAGS2)

**Bit**

<table>
<thead>
<tr>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1... .... .... .... Conservation phase</td>
</tr>
<tr>
<td>.1... .... .... .... Medium aggressive phase</td>
</tr>
<tr>
<td>..1... .... .... .... Send setup segment</td>
</tr>
<tr>
<td>...1... .... .... .... High flag</td>
</tr>
<tr>
<td>.... 1... .... .... .... Delay change sum flag</td>
</tr>
<tr>
<td>.... 1... .... .... .... Upward cycle flag</td>
</tr>
<tr>
<td>.... ..1... .... .... .... Idle - no data has been sent recently</td>
</tr>
<tr>
<td>.... .... 1... .... .... .... Startup</td>
</tr>
<tr>
<td>.... .... ..1... .... .... Last parity bit received on an ARB rate request</td>
</tr>
<tr>
<td>.... .... .... 1... .... .... Current ARB request parity bit</td>
</tr>
</tbody>
</table>
High steps (ARB2_HIGH_STEPS) – the number of successive normal rate increases above the current shared link rate

Aggressive steps (ARB2_AGGRESSIVE_STEPS)

Measurement interval in milliseconds (ARB2_MEASURE_INT)

Burst interval in milliseconds (ARB2_BURST_INTVL)

Downward gamma value (ARB2_DOWN_GAMMA)

Gamma value

NCB address

Request parameter header (RPH) address

**ARP2 or ARS2 entry for RTP session data reply (Part 2)**

**Entry:**
ARP2 or ARS2

**VIT option:**
HPR

**Event:**
Any session data reply is sent over an RTP connection using Responsive Mode ARB

**VIT processing module:**
ISTITCHR

This trace record is a continuation of the ARP or ARS entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;ARP2&quot; for real reply</td>
</tr>
<tr>
<td></td>
<td>• C&quot;ARS2&quot; for simulated reply</td>
</tr>
<tr>
<td>04–05</td>
<td>Current F-curve value (ARB2_FCURVE_NOW)</td>
</tr>
</tbody>
</table>
06–07
  Current alpha value (ARB2_ALPHA_NOW)

08–0B
  Current tracking speed value

0C–0F
  Maximum send rate in kbps (ARB_MAX_SEND_RATE)

10–13
  Allowed send rate in kbps (ARB_ALLOW_SEND_RATE)

14–17
  Target rate in kbps (ARB2_TARGET_RATE)

18–1B
  Smooth actual rate in kbps (ARB2_SMOOTH_ACTUAL_RATE)

1C–1F
  Smooth round-trip time in milliseconds (RPN_SRTT)

ARQ entry for RTP session data request (Part 1)

Entry:
  ARQ

VIT option:
  HPR

Event:
  Any session data request is received over an RTP connection using Responsive Mode ARB

VIT processing module:
  ISTITCHR

Control is returned to:
  ISTRPCAR

This trace record is written when session data is received across an RTP connection using Responsive Mode ARB. It contains statistical information about the session data.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;ARQ&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Trace instance</td>
</tr>
</tbody>
</table>
Code
Meaning
C
  Computer (values before computer code executes).
D
  Drift detected.
N
  Entry (values on entry).
R
  Drift reduction.
X
  Exit (values on exit).

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0
06–07
Code
Meaning
N, X
  Phase flags (ARB2_PHASE_FLAGS, ARB2_PHASE_FLAGS2).
C
  Window size (ARB2_POT_WINDOW_SIZE).
D, R
  0

Phase flags:

Bit
Meaning
1... .... ....
  Conservation phase
.1... .... ....
  Medium aggressive phase
..1... .... ....
  Send setup segment
...1... .... ....
  High flag
.... 1... ....
  Delay change sum flag
.... .1... ....
  Upward cycle flag
.... ..1... ....
  Idle - no data has been sent recently
.... .... 1... ....
  Startup
.... .... 1... ....
  Last parity bit received on an ARB rate request
.... .... .1... ....
  Current ARB request parity bit
<table>
<thead>
<tr>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>08–0B</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Code</strong></td>
<td><strong>Meaning</strong></td>
</tr>
<tr>
<td><strong>C, N, X</strong></td>
<td>Receiver threshold in microseconds (ARB2_POT_RCVR_THRESHOLD).</td>
</tr>
<tr>
<td><strong>D</strong></td>
<td>Current accumulated queuing time in microseconds (ARBLV_CLOCK_DRIFT). This is the minimum of the ACCUM QUEUING TIME and DCS BAR.</td>
</tr>
<tr>
<td><strong>R</strong></td>
<td>Current accumulated queuing time in microseconds (ARBLV_CLOCK_DRIFT). This is the absolute value of ACCUM QUEUING TIME plus 1500 microseconds.</td>
</tr>
</tbody>
</table>

| **0C–0F** | |
| **Code** | **Meaning** |
| **C, N, X** | Maximum present threshold in microseconds (ARB2_POT_PRESENT_THR_MAX). |
| **D** | Elapsed time, in microseconds, since a drift condition was detected. |
| **R** | ACCUM QUEUING TIME has been rounded back to 0. |

| **10–13** | |
| **Code** | **Meaning** |
| **C** | Minimum present threshold in microseconds (ARB2_POT_PRESENT_THR_MIN). |
| **N, X** | Anticipated variance in milliseconds (ARB2_POT_ANTICIPATED_VAR). |
| **D, R** | Amount of clock drift, measured in microseconds, allowed per second. |

| **14–17** | |
| **Code** | **Meaning** |
| **C** | Delay change sum variance. |
| **N, X** | Delay change sum bar in microseconds (ARB2_POT_DCS_BAR). |
| **D, R** | This value has a range of 2 to 20000. The default is 20000, which is associated with a drift of 50 microseconds per second. The drift rate, in microseconds, can be determined by dividing one million by the ARB_DRIFT_FACTOR. |

| **18–1B** | NCB address |
| **1C–1F** | Request parameter header (RPH) address |
ARQ2 entry for RTP session data request (Part 2)

Entry:
ARQ2

VIT option:
HPR

Event:
Any session data request is received over an RTP connection using Responsive Mode ARB.

VIT processing module:
ISTITCHR

This trace record is a continuation of the ARQ entry. This trace record is written only for the Entry and Exit instances.

<table>
<thead>
<tr>
<th>00–03</th>
<th>04–07</th>
<th>08–0B</th>
<th>0C–0F</th>
<th>10–13</th>
<th>14–17</th>
<th>18–1B</th>
<th>1C–1D</th>
<th>1E–1F</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARQ2</td>
<td>CURRENT QUEUING TIME OR INCRE'T TREND</td>
<td>ACCUM QUEUING TIME</td>
<td>RECEIVER THRESH MAXIMUM OR RATE SCALER</td>
<td>RECEIVR THRESH MINIMUM OR ERROR COUNT</td>
<td>TARGET RATE</td>
<td>RECOVERY FACTOR</td>
<td>LAST REQUEST CORRELATOR</td>
<td>0</td>
</tr>
</tbody>
</table>

Byte (hex)
Contents
00–03
Record ID: C"ARQ2"

04–07
Current queuing time in milliseconds for Entry. Increment trend (ARB2_POT_INC_TREND) for Exit.

08–0B
Accumulated queuing time in microseconds (ARB_ACCUM_QTIME for Entry, ARB2_POT_ACCUM_QTIME for Exit)

0C–0F
Receiver threshold maximum in microseconds (ARB2_RCVR_THRESHOLD_MAX) for Entry. Rate scaler for Exit.

10–13
Receiver threshold minimum in microseconds (ARB2_RCVR_THRESHOLD_MIN) for Entry. Error count (ARB2_POT_ERROR_COUNT) for Exit.

14–17
Target rate (ARB2_TARGET_RATE)

18–1B
Recovery factor (ARB2_POT_RECOVERY_FACTOR)

1C–1D
Last request correlator (ARB2_LAST_REQ_CORRELATOR)

1E–1F
0
**ASNB entry for ASSIGN_BUFFER requests**

**Entry:**
ASNB

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=ASSIGN_BUFFER

**VIT processing module:**
ISTITCCS

**Control is returned to:**
IVTSMCBF or IVTSM6BF

**Notes:**
- IVTSMCBF processes ASSIGN_BUFFER requests in 31-bit addressing mode.
- IVTSM6BF processes ASSIGN_BUFFER requests in 64-bit addressing mode.

This record provides the status of an IVTCSM REQUEST=ASSIGN_BUFFER macroinstruction. A pair of ASNB records is created for each IVTCSM REQUEST=ASSIGN_BUFFER macroinstruction. The first record, along with the ASN2 continuation records, is created when the macroinstruction is issued and records the buffer token provided. The second record is created when the macroinstruction completes and records the new buffer tokens and additional information.

---

### Byte (hex)

**Contents**

<table>
<thead>
<tr>
<th>00–03</th>
<th>Record ID: C&quot;ASNB&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace record flags:</td>
</tr>
<tr>
<td></td>
<td>B'0... ....' Indicates that this is the last record for this event.</td>
</tr>
<tr>
<td></td>
<td>B'1... ....' Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.</td>
</tr>
<tr>
<td></td>
<td>B'.1... ....' Traces the issuance of the request.</td>
</tr>
<tr>
<td></td>
<td>B'.0... ....' Traces the completion of the request.</td>
</tr>
<tr>
<td></td>
<td>B'.1... ....' Indicates that the buffer list is in 64-bit storage.</td>
</tr>
</tbody>
</table>
Trace record number to correlate all the entries for this particular event.

Type or state of buffers allocated

- X'80'  
  Fixed

- X'20'  
  Eligible to be made pageable

Owner ID. When traced on issuance, the owner ID is always 0.

Return Code

Reason Code

Address of utility routine caller or return address of the issuer of the IVTCSM macroinstruction

Last buffer list entry. When traced on issuance, the last entry is nonzero whether an error occurred or not. On completion of assigned buffer, the last entry is 0 if the return code is 0 and nonzero if the return code is nonzero.

When an error occurs in 64-bit addressing mode, the last buffer list entry here is X'FFFFFFFF'. The ASN6 entry follows with the 64-bit address of the last buffer list entry.

Number of buffers requested to be assigned

THREAD value, if specified, or 0 if THREAD is not specified. The THREAD value is used only to correlate this trace record to a specific IVTCSM macroinstruction.

ASN2 entry for ASSIGN_BUFFER requests

Entry:
  ASN2

VIT option:
  CSM

Event:
  IVTCSM REQUEST=ASSIGN_BUFFER

VIT processing module:
  ISTITCCS

This trace record is a continuation of the ASNB trace record. It contains information about the buffer images that were assigned. Note that two buffer tokens are present in the trace record for each buffer list entry. This is because a token is received on input and a new token is created to represent the new image of the buffer on output.
### Byte (hex)
#### Contents

**00–03**
- Record ID: C"ASN2"

**04–05**
- 0

**06**
- Trace record flag:
  - B'0... ....'
    - Indicates that this is the last record for this event.
  - B'1... ....'
    - Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

**07**
- Trace record number to correlate all the entries for this particular event

**08–13**
- Buffer token contained in the buffer list entry

**14–1F**
- Buffer token contained in the buffer list entry or 0 if there was an odd number of buffers in the list

### ASN6 entry for ASSIGN_BUFFER requests

**Entry:**
- ASN6

**VIT option:**
- CSM

**Event:**
- IVTCSM REQUEST=ASSIGN_BUFFER

**VIT processing module:**
- ISTITCCS

This trace record is a continuation of the ASNB trace record. It contains the 64-bit buffer list entry address. This entry is present only when the ASSIGN_BUFFER request failed with the error return code in 64-bit addressing mode.

### Byte (hex)
#### Contents

**00–03**
- Record ID: C'ASN6'
ATSK entry for attach of a subtask

Entry:
  ATSK
VIT option:
  PSS
Event:
  ATTACH of a subtask
VIT processing module:
  ISTRACPS
Control is returned to:
  The module that issued the ATTACH

This trace record is written when a VTAM module issues an ATTACH macro to start another subtask. This record is generated after the ATTACH completes to ensure that completion information is included in the entry. Consequently, this entry can sometimes appear after the BTSK entry that is generated by the subtask it is attaching.

Not all subtask events generate this entry.

| 00 | 00 | 00 | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| ATSK | I | D | R | C | 0 | TASK NAME | TCB ADDR | 0 | ISSUER ADDRESS | REG 1 |

Byte (hex)

Contents

00–03
  Record ID: C"ATSK"

04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
Return code from the ATTACH macro
06–07
0
08–0F
Name of the subtask being attached
10–13
TCB address returned from the ATTACH macro
14–17
0
18–1B
Address of the issuer of the ATTACH macro
1C–1F
Register 1

**ATTx entry for attention**

**Entry:**
ATT

**VIT option:**
CIO

**Event:**
Attention

**VIT processing module:**
ISTRACCI

**Control is returned to:**
ISTTSCCA or ISTTSCLA

ATTI is written when an attention interrupt occurs for a channel-attached communication controller or an SNA cluster controller.

ATTL is written when an attention interrupt occurs for a channel-attached non-SNA device (device defined in a local non-SNA major node).

ATTT is written when an attention interrupt occurs for a TCP/IP DLC connection.

ATTX is written when an attention interrupt occurs for a channel-to-channel attachment, including an IBM 3172 Interconnect Nways Controller.

For record types with suffix I, X, or T, the CIO events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>ATTI</td>
</tr>
<tr>
<td>2</td>
<td>ATTL</td>
</tr>
<tr>
<td>4</td>
<td>ATT</td>
</tr>
<tr>
<td>6</td>
<td>ATTX</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>12</td>
<td>A</td>
</tr>
<tr>
<td>14</td>
<td>B</td>
</tr>
<tr>
<td>16</td>
<td>C</td>
</tr>
<tr>
<td>18</td>
<td>D</td>
</tr>
<tr>
<td>20</td>
<td>E</td>
</tr>
<tr>
<td>22</td>
<td>F</td>
</tr>
<tr>
<td>24</td>
<td>CSW</td>
</tr>
</tbody>
</table>

For record types with suffix I, X, or T, the CIO events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.
00–03
Record ID:

- C"ATTI" for ICNCB
- C"ATTL" for LDNCB
- C"ATTT" for RWNCB
- C"ATTX" for XCNCB

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Link station state (NCBLNKST) for ICNCB, LDNCB, and RWNCB; station state (XCNSSFSM) for XCNCB

06
CCW opcode or channel or flag byte (XCNFLAG2)

07
0 or flag byte (XCNFLAG3)

08–0B
Channel device name in EBCDIC (either a device address or device number)

0C–0F
NCB address

10–13
Flag bytes (NCBFLAGS)

14
0 or flag byte (XCNFLAG) or for ATTT:
- For the CDLC DLC: C
- For the LCS DLC: L
- For the CLAW DLC: W
- For the CTC DLC: X
- For the HYPERchannel DLC: H

15
I/O completion code (IOSCOD in the IOSB)

16–17
Sense data (IOSSNS of the IOSB)

18–1F
Channel status word from the IOSB

**BSPx entry for ADD, DELETE, and FIND macros**

**Entry:**
- BSPA, BSPD, or BSPF

**VIT option:**
- NRM

**Event:**
- BSBPCID ADD, DELETE, and FIND macros

**VIT processing module:**
- ISTRACNR

**Control is returned to:**
- ISTTSCPU

This trace record is written when ADD, DELETE, and FIND operations are performed for the BSBPCID tree. It shows:
- Key used for the invocation
- Information from the boundary session block (BSB) indicating the state of the session

Use this entry to track activity on a session.

### Byte (hex) Contents

<table>
<thead>
<tr>
<th>00–03</th>
<th>Record ID:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C&quot;BSPA&quot; for BSBPCID ADD</td>
</tr>
<tr>
<td></td>
<td>C&quot;BSPF&quot; for BSBPCID FIND</td>
</tr>
<tr>
<td></td>
<td>C&quot;BSPD&quot; for BSBPCID DELETE</td>
</tr>
</tbody>
</table>

| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'. |
| 05 | Primary logical unit (PLU) state |
| 06 | Macro return code |
|    | **Code** |
|    | **Meaning** |
| X'00' | Successful invocation |
| X'04' | Boundary session block (BSB) not found |
| X'08' | Duplicate input address |
| X'12' | Address not valid |

| 07 | IP flags, if this BSB represents a SNA/IP session. 0 otherwise. |
| 08–0B | Address of the issuer of the macro |
| 0C–0F | Boundary session block (BSB) address |
| 10–17 | Procedure-correlation identifier (PCID) |
| 18–1B | Boundary session block (BSB) flags (FLAG1, FLAG2, FLAG3) |
| 1C–1F | Request parameter header (RPH) address of module issuing the call |
BSSx entry for Add, Delete, and Find

Entry:
BSSA, BSSD, or BSSF

VIT option:
NRM

Event:
BSBSA ADD, DELETE, and FIND macros

VIT processing module:
ISTRACNR

Control is returned to:
ISTTSCB2

This trace record contains information about BSBSA ADD, DELETE, or FIND macros issued by a VTAM module.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03 | Record ID:  
  • C"BSSA" for BSBSA ADD  
  • C"BSSD" for BSBSA DELETE  
  • C"BSSF" for BSBSA FIND |
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'. |
| 05–07 | 0 |
| 08–0B | Address of the issuer of the macro |
| 0C–0F | Subarea address |
| 10–11 | Subarea element address |
| 12–13 | Rex element address |
| 14–17 | Boundary session block (BSB) address or 0 |
| 18–1B | BSB Flags (FLAG1, FLAG2, FLAG3) |
**1C**
PLU state

**1D**
Return code

*Note:* It is not required that users of the product know the meaning of this internal VTAM return code. When required, the product support organization may use it to assist in internal flow diagnosis.

**1E–1F**
Rex element address index

---

### BSXx entry for Add, Delete, and Find

**Entry:**
- BSXA, BSXD, or BSXF

**VIT option:**
- NRM

**Event:**
- BSBREX ADD, DELETE, and FIND macros

**VIT processing module:**
- ISTRACNR

**Control is returned to:**
- ISTTSCB1

This trace record contains information about BSBREX ADD, DELETE, or FIND macros issued by a VTAM module.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>00–03</th>
<th>Record ID:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• C&quot;BSXA&quot; for BSBREX ADD</td>
</tr>
<tr>
<td></td>
<td>• C&quot;BSXD&quot; for BSBREX DELETE</td>
</tr>
<tr>
<td></td>
<td>• C&quot;BSXF&quot; for BSBREX FIND</td>
</tr>
</tbody>
</table>

| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF' |

| 05–07 | 0 |

| 08–0B | Address of the issuer of the macro |

| 0C–0D | LFSID |

| 0E–0F | PU element address |
10  
CRA flags
11–13  
0
14–17  
Boundary session block (BSB) address or 0
18–1B  
BSB Flags (FLAG1, FLAG2, FLAG3)
1C  
PLU state
1D  
Return code

Note: It is not required that users of the product know the meaning of this internal VTAM return code. When required, the product support organization may use it to assist in internal flow diagnosis.

1E–1F  
0

**BTSK entry for begin a subtask**

**Entry:**

BTSK

**VIT option:**

PSS

**Event:**

Begin a subtask

**VIT processing module:**

ISTRACPS

**Control is returned to:**

The subtask that is just starting

This trace record is written when a VTAM subtask begins execution. This record can appear in the internal trace table before the ATSK entry for the ATTACH of the subtask.

Some subtasks start without generating this entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: &quot;BTSK&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>Name of the subtask that was entered</td>
</tr>
</tbody>
</table>
10–13
TCB address of the current subtask

14–17
Entry point into the subtask

18–1B
0

1C–1F
Contents of register 1 upon entry

BUFF entry for 8K external buffer

Entry:
BUFF

VIT option:
None (generated when VIT is running externally using 8K buffers)

Event:
External VIT tracing

VIT processing module:
ISTRACRR

Control is returned to:
ISTRACRR

This trace record is the first record in an 8K external VIT buffer. It determines if the 8K buffer was written to the external trace file in the proper sequence. It also indicates whether enough 8K buffers were specified on the TRACE start option or the MODIFY TRACE command by counting the records that were individually GTRACEd.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: &quot;BUFF&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06–07</td>
<td>Current BFRNUM value (as specified on TRACE start option or MODIFY TRACE command)</td>
</tr>
<tr>
<td>08–0B</td>
<td>Count of VIT records individually GTRACEd because of a lack of 8K buffers</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Buffer sequence number</td>
</tr>
<tr>
<td>10–17</td>
<td>Time stamp of first VIT record in the buffer</td>
</tr>
<tr>
<td>18–1B</td>
<td>0</td>
</tr>
</tbody>
</table>
C64Q entry for CDSG queue manager events (IUTC64QM macro) (Part 1)

Entry:
C64Q

VIT option:
CIA

Event:
Queue management using IUTC64QM

VIT processing module:
ISTITCSH

Control is returned to:
IUTC64QM issuer

This trace record is written when the IUTC64QM macro is run. The record is primarily used to identify the addition or removal of elements to or from a queue header.

Restriction: This record is not written for a Get_Element_Count request or when a Remove_Element request fails.

<table>
<thead>
<tr>
<th>00 00 00 01 02 03 04 05 06 07 08 09</th>
<th>00 00 00 01 02 03 04 05 06 07 08 09</th>
</tr>
</thead>
<tbody>
<tr>
<td>C 6 4 Q I D 0 F U N C T I O N</td>
<td>RETURN ADDRESS</td>
</tr>
<tr>
<td>B E F O R E S Y N C</td>
<td>AFTER SYNCH</td>
</tr>
<tr>
<td>0 ADDED / REMOVED ELEMENT ADDRESS</td>
<td>or</td>
</tr>
<tr>
<td>or 0 IUTC64QH ADDRESS</td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

**00–03**
Record ID: C"C64Q"

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05–06**
0

**07**
Function:

*C'A'*
For Add_Chain

*C'R'*
For Remove_Element
C'S'
   For Synch_byte_only

08–0B
   Address of caller (within IUTC64QM expansion)

0C
   Contents of synch byte before CDSG

0D
   Contents of synch byte after CDSG

0E–0F
   0

10–18
   Element address:
   • For Add_Chain, address of first element in chain
   • For Remove_Element, address of element removed
   • For Synch_byte_only, 0

19–1F
   CDSG queue manager header address (IUTC64QH)

C642 entry for CDSG queue manager events (IUTC64QM macro) (Part 2)

Entry:
   C642

VIT option:
   CIA

Event:
   Queue management using IUTC64QM

VIT processing module:
   ISTITCSH

This trace record is a continuation of the C64Q entry.

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| C | 6 | 4 | 2 | 0 | FIFO QUEUE ANCHOR | LIFO QUEUE ANCHOR | F | C | I | O | F | U | N | T | L | I | O | F | U | N | T | RPH ADDRESS |

Byte (hex)
   Contents

00–03
   Record ID: C"C642"

04–07
   0
08–0F
Address of first element on FIFO queue after the CDSG
10–17
Address of first element on LIFO queue after the CDSG
18–19
Number of elements on FIFO queue after CDSG (X’FFFF’ if 65,525 or more elements are on the queue)
1A–1B
Number of elements on LIFO queue after CDSG (X’FFFF’ if 65,525 or more elements are on the queue)
1C–1F
Request parameter header (RPH) address

CCI or CCO entry for SSCP (RUPE — Part 1)

Entry:
CCI or CCO

VIT option:
SSCP

Event:
Requests/responses with a RUPE (Part 1)

VIT processing module:
ISTRACSC

Control is returned to:

ISTINCCI for inbound processing
ISTINCCO for outbound processing

This trace record provides information about outbound processing done by ISTINCCO and inbound processing done by ISTINCCI. ISTINCCO processes a request; ISTINCCI sends a response to the requestor.

If this entry is associated with an event failure (that is, the sense data is nonzero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition and, therefore, is traced whenever the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CCI&quot; for inbound processing</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CCO&quot; for outbound processing</td>
</tr>
<tr>
<td>03</td>
<td>Flags</td>
</tr>
</tbody>
</table>
• 80 = response RU
• 40 = sense traced
• 00 = request RU

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Control block ID=X'54'

06
Flags (CPCBFL)

07
Return code (CPCBRC)

08–0B
Save area address

0C–0F
Save area ID (bytes 4, 5, 7, and 8 of the module name if available)

10–13
Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)

14–19
For sender, network address at origin (RUPE)

1A–1F
For sender, network address at destination (RUPE)

CC2 entry for SSCP (RUPE — Part 2)

Entry:
CC2

VIT option:
SSCP

Event:
Requests/responses with a RUPE (Part 2)

VIT processing module:
ISTRACSC

This trace record is a continuation of the CCI or CCO (RUPE) entry.

| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| CC2 | C | B | I | D | FIRST 24 BYTES OF RU | SENSE DATA OR 0 |

Byte (hex)

Contents

00–02
Record ID: C"CC2"

03
Control block ID=X'54'

04–1B
First 24 bytes of RU

1C–1F
Sense data or 0
CCI or CCO entry for SSCP (NCSPL — Part 1)

Entry:
CCI or CCO

VIT option:
SSCP

Event:
Requests/responses for an NCSPL (Part 1)

VIT processing module:
ISTRACSC

Control is returned to:

ISTINCCI for inbound processing
ISTINCCO for outbound processing

This trace record provides information about outbound processing, which processes a request, and inbound processing, which sends a response to the requester.

If this entry is associated with an event failure (that is, the sense data is nonzero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition and, therefore, is traced whenever the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID:</td>
</tr>
</tbody>
</table>

- C"CCI" for inbound processing
- C"CCO" for outbound processing

<table>
<thead>
<tr>
<th>03</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>80 = response RU</td>
<td></td>
</tr>
<tr>
<td>40 = sense traced</td>
<td></td>
</tr>
<tr>
<td>00 = request RU</td>
<td></td>
</tr>
</tbody>
</table>

| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'. |

| 05 | Control block ID=X'60' |

| 06 | Flags (CPCBFL) |

| 07 | Return code (CPCBRC) |

| 08–0B | Save area address |

VTAM internal trace (VIT) record descriptions 267
0C–0F
Save area ID (bytes 4, 5, 7, and 8 of the module name if available)

10–13
Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)

14–17
CPCB operation code (see z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)

18
WTD type

19
0

1A–1F
Network address at destination

**CC2 entry for SSCP (NCSPL — Part 2)**

**Entry:**
CC2

**VIT option:**
SSCP

**Event:**
Requests/responses for an NCSPL (Part 2)

**VIT processing module:**
ISTRACSC

This trace record is a continuation of the CCI or CCO (NCSPL) entry.

| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 00 | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B |

**Byte (hex)**
Contents

00–02
Record ID: C"CC2"

03
Control block ID=X'60'

04–07
RDTE state (in control block RPRE)

08–0B
CPCB WTD (first 4 bytes)

0C–1F
0

**CCI or CCO entry for SSCP (not RUPE or NCSPL)**

**Entry:**
CCI or CCO
VIT option:
SSCP

Event:
Requests/responses for neither RUPE nor NCSPL

VIT processing module:
ISTRACSC

Control is returned to:

ISTINCCI for inbound processing
ISTINCCO for outbound processing

This trace record provides information about outbound processing, which processes a request, and
inbound processing, which sends a response to the requester.

If this entry is associated with an event failure (that is, the sense data is nonzero), this entry is generated
whether the SSCP option is in effect or not. It is treated as an exception condition and, therefore, is traced
whenever the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–02      | Record ID:  
|            | • C"CCI" for inbound processing  
|            | • C"CCO" for outbound processing |
| 03         | Flags  
|            | • 80 = response RU  
|            | • 40 = sense traced  
|            | • 00 = request RU |
| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'. |
| 05         | Control block ID (Control block IDs are shown in z/OS Communications Server: SNA Diagnosis Vol 1,  
|            | Techniques and Procedures.) |
| 06         | CPCB flag (CPCBFL) |
| 07         | Return code (CPCBRC) |
| 08–0B      | Save area address |
| 0C–0F      | Save area ID (bytes 4, 5, 7, and 8 of the module name if available) |
| 10–13      | Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT) |
CCR entry for a communication channel operation

Entry:
CCR
VIT option:
CIA
Event:
communication channel operation for an IBM 10 GbE RoCE Express feature that operates in a shared RoCE environment

This trace record is written when a communication channel operation is performed during the activation of a 10 GbE RoCE Express feature that operates in a shared RoCE environment.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03</td>
<td>Record ID: C&quot;CCR&quot;</td>
</tr>
<tr>
<td>04-05</td>
<td>ID is the primary address space ID (ASID).</td>
</tr>
<tr>
<td>06</td>
<td>0</td>
</tr>
<tr>
<td>07</td>
<td>Operation code for the communication channel command that was issued</td>
</tr>
<tr>
<td>08-09</td>
<td>Return code</td>
</tr>
<tr>
<td>0A-0B</td>
<td>Reason code</td>
</tr>
<tr>
<td>0C-0F</td>
<td>0</td>
</tr>
<tr>
<td>10-18</td>
<td>Address of the PFCTE control block that represents this 10 GbE RoCE Express feature</td>
</tr>
</tbody>
</table>
**CCR2 entry for communication channel operation (Part 2)**

**Entry:**
CCR2

**VIT option:**
CIA

**Event:**
communication channel operation for an IBM 10 GbE RoCE Express feature that operates in a shared RoCE environment.

This record is a continuation of the CCR trace record.

<table>
<thead>
<tr>
<th>C C R 2</th>
<th>OPCODE AND RETRIES</th>
<th>INITIAL VALUE</th>
<th>COMMAND VALUE</th>
<th>COMPLETION VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 0 0 0 0 0</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>0 0 0 0 0 0 0</td>
<td>1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00-03**
Record ID: C"CCR2"

**04-07**
Operation code and operation retries values

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-11</td>
<td>Represents the operation code performed</td>
</tr>
<tr>
<td>12-31</td>
<td>Represents the number of retries before the command completes</td>
</tr>
</tbody>
</table>

**08-0F**
The value of the communication channel before the command is attempted

**10-17**
The value stored by z/OS Communication Server into the communication channel to execute the command

**18-1F**
The value of the communication channel after the command completes

---

**CDHx entry for CIDCTL HOSTNODE Add, Change, Delete, and Find macros**

**Entry:**
CDHA, CDHC, CDHD, CDHF

**VIT option:**
NRM
Event:
CIDCTL HOSTNODE ADD, CHANGE, DELETE, FIND macros

VIT processing module:
ISTRACNR

Control is returned to:
ISTNACRT

This trace record is written whenever a CIDCTL HOSTNODE ADD, CHANGE, DELETE, or FIND macro is invoked. CIDCTL FIND macro invocations executed during the process of sending or receiving data are not traced unless they result in a nonzero return code.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CDHA&quot; CIDCTL HOSTNODE ADD</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CDHC&quot; CIDCTL HOSTNODE CHANGE</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CDHD&quot; CIDCTL HOSTNODE DELETE</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CDHF&quot; CIDCTL HOSTNODE FIND</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>NATFLAGS byte from NATBL</td>
</tr>
<tr>
<td>07</td>
<td>Return code</td>
</tr>
<tr>
<td>08–0B</td>
<td>Return address of the invoker of the CIDCTL macro</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of the control block that was added, found, or changed; 0 for delete</td>
</tr>
<tr>
<td>10–11</td>
<td>Index portion of subarea address</td>
</tr>
<tr>
<td>12–13</td>
<td>Subarea portion of subarea address</td>
</tr>
<tr>
<td>14–15</td>
<td>Element portion of subarea address</td>
</tr>
<tr>
<td>16–17</td>
<td>0</td>
</tr>
<tr>
<td>18–1B</td>
<td>Host node table entry address</td>
</tr>
</tbody>
</table>
CDNx entry for CIDCTL NEXTNODE Add, Delete, and Find macros

Entry:
CDNA, CDND, CDNF

VIT option:
NRM

Event:
CIDCTL NEXTNODE ADD, DELETE, FIND macros

VIT processing module:
ISTRACNR

Control is returned to:
ISTNACRT

This trace record is written whenever a CIDCTL NEXTNODE ADD, DELETE, or FIND macro is invoked. CIDCTL FIND macro invocations executed during the process of sending or receiving data are not traced unless they result in a nonzero return code.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CDNA&quot; CIDCTL NEXTNODE ADD</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CDND&quot; CIDCTL NEXTNODE DELETE</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CDNF&quot; CIDCTL NEXTNODE FIND</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>NATFLAGS byte from NATBL</td>
</tr>
<tr>
<td>07</td>
<td>Return code</td>
</tr>
<tr>
<td>08–0B</td>
<td>Return address of the invoker of the CIDCTL macro</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of the control block that was added or found; 0 for delete</td>
</tr>
<tr>
<td>10–11</td>
<td>Index portion of subarea address</td>
</tr>
</tbody>
</table>
**12–13**
Subarea portion of subarea address

**14–15**
Element portion of subarea address

**16–17**
0

**18–1B**
Host node table entry address

**1C–1F**
Address of RPH

---

### CDN2 entry for CIDCTL NEXTNODE FIND (Part 2)

**Entry:**
CDN2

**VIT option:**
NRM

**Event:**
CIDCTL NEXTNODE FIND macro

**VIT processing module:**
ISTRACNR

This trace record is a continuation of the CDNF entry. This entry is present only when the subordinate host node table entry address is found and is associated with a boundary logical unit control block (BLB).

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'CDN2'</td>
</tr>
<tr>
<td>04–07</td>
<td>Subordinate host node table entry address.</td>
</tr>
<tr>
<td>08–0B</td>
<td>Boundary Logical Unit control block address or 0.</td>
</tr>
<tr>
<td>0C–0F</td>
<td>0</td>
</tr>
</tbody>
</table>

---

### CDSQ entry for CDS queue manager event (IUTCDSQM macro)

**Entry:**
CDSQ

**VIT option:**
CIA

**Event:**
Queue management using IUTCDSQM

**VIT processing module:**
ISTITCOD
Control is returned to:
IUTCDSQM issuer

This trace record is written when the IUTCDSQM macro is run. It is primarily used to identify the addition or removal of elements to or from a queue header. **Exception:** This record is not written for a Get_Element_Count request or when a Remove_Element request fails.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CDSQ&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05-06</td>
<td>0</td>
</tr>
</tbody>
</table>
| 07         | Function:  
  • C'A' for Add_Chain  
  • C'R' for Remove_Element  
  • C'S' for Synch_byte_only |
| 08–0B      | Element address:  
  • For Add_Chain, address of first element in chain  
  • For Remove_Element, address of element removed  
  • For Synch_byte_only, 0 |
| 0C         | Contents of synch byte before CDS |
| 0D         | Contents of synch byte after CDS |
| 0E         | Number of elements on FIFO queue after CDS (X'FF' if there are 255 or more elements on the queue) |
| 0F         | Number of elements on LIFO queue after CDS (X'FF' if there are 255 or more elements on the queue) |
| 10-13      | Address of caller (within IUTCDSQM expansion) |
| 14-17      | Address of first element on FIFO queue after the CDS |
**CFAL entry for coupling facility IXLALTER service (Part 1)**

Entry:
- CFAL

VIT option:
- CFS

Event:
- Invocation of the MVS macro IXLALTER

VIT processing module:
- ISTRACCF

Control is returned to:
- ISTFSUAL

This trace entry is written when the IXLALTER macro is used to:
- Initiate an alter of the coupling facility storage attributes
- Stop an alter in progress of the coupling facility storage

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CFAL&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Function request by this invocation of IXLALTER:</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>X'01'</td>
<td>REQUEST(START) to start the alter process.</td>
</tr>
<tr>
<td>X'02'</td>
<td>REQUEST(STOP) stop the alter process.</td>
</tr>
<tr>
<td>07</td>
<td>Event flags</td>
</tr>
</tbody>
</table>
Bit
Meaning

1... ....
Request that the size of the coupling facility structure be changed.

.1... ....
Request that the entry-to-element ratio for a coupling facility structure be changed.

..xx xxxx
0

08–0B
Structure object (CFSSTR) address.

0C–0F
0

10–13
Return address of the invoker of the utility module.

14–17
Return code from the MVS macro IXLALTER. See z/OS MVS Programming: Sysplex Services Reference for an explanation of this code.

18–1B
Reason code from the MVS macro IXLALTER. See z/OS MVS Programming: Sysplex Services Reference for an explanation of this code.

1C–1F
Request parameter header (RPH) address.

CFA2 entry for coupling facility IXALTER service (Part 2)

Entry:
CFA2

VIT option:
CFS

Event:
Invocation of the MVS macro IXLALTER

VIT processing module:
ISTRACCF

This trace entry is a continuation of the CFAL entry.

<table>
<thead>
<tr>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
<th>00 00 00 00</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFA2</td>
<td>T E A R T R E T R A T I O</td>
<td>T E A R T R E T R A T I O</td>
<td>T S S T A T I R Z G U E T U R E</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
Contents

00–03
Record ID: C"CFA2"

VTAM internal trace (VIT) record descriptions 277
If request is to alter the entry-to-element ratio, this is the target entry portion of the ratio.
If request is to alter the entry-to-element ratio, this is the target element portion of the ratio.
If request is to alter the structure size, this is the target structure size.

**CFCE entry for coupling facility request completion exit**

**Entry:**
CFCE

**VIT option:**
CFS

**Event:**
Complete exit driven for a connection to coupling facility structure

**VIT processing module:**
ISTRACCF

**Control is returned to:**
ISTFSXCM

This trace record is written when a previous request to access the coupling facility structure has completed asynchronously.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: &quot;CFCE&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Event flags:</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>1...</td>
<td>This event applies to the new version of the structure.</td>
</tr>
<tr>
<td>.1...</td>
<td>This event pertains to an invocation of the IXLLIST macro.</td>
</tr>
<tr>
<td>..1...</td>
<td>This event pertains to an invocation of the IXLCACHE macro.</td>
</tr>
</tbody>
</table>
This event pertains to an invocation of the IXLLOCK macro.

06 0

07
The connection identifier for the target of this event

08–0B
Structure object (CFSSTR) address

0C–13
The request data specified on the MVS macro invocation identified in the flags at offset X'05' of CFCE

Bit
Meaning
When this bit is on at the completion exit, the buffer contents for the DELETE_ENTRYLIST are traced using the CFLS VIT entries.

14–17
Return code for the MVS macro indicated at offset X'05'

18–1B
Reason code for the MVS macro indicated at offset X'05'

1C–1F
Request parameter header (RPH) address

---

CFCN entry for coupling facility connection service (Part 1)

Entry: CFCN
VIT option: CFS
Event: Invocation of the MVS macro IXLCONN
VIT processing module: ISTRACCF
Control is returned to: ISTFSGCN

This trace record is written when there is an attempted connection to the coupling facility structure with the IXLCONN macro.

---

Byte (hex)
Contents

---

VTAM internal trace (VIT) record descriptions 279
Record ID: C"CFCN"

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Event flags:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>The rebuild process is in progress for the structure.</td>
</tr>
<tr>
<td>.1...</td>
<td>The rebuild stop process is in progress for the structure.</td>
</tr>
<tr>
<td>..1.</td>
<td>This connection attempt caused the structure to be allocated.</td>
</tr>
<tr>
<td>...1</td>
<td>A previously failed connection has been reestablished.</td>
</tr>
<tr>
<td>.1..1</td>
<td>Full connectivity to the rebuild structure cannot be obtained. This bit is valid only if this connection attempt caused the structure to be allocated. This bit is set only if IXLCONN was invoked during rebuild processing.</td>
</tr>
<tr>
<td>...1.</td>
<td>The structure involved is volatile.</td>
</tr>
<tr>
<td>...1.</td>
<td>VTAM did not process this event.</td>
</tr>
</tbody>
</table>

Connection identifier or 0. If IXLCONN was successful, this field contains the connection identifier assigned to this connection. Otherwise this field contains the value 0.

Structure object (CFSSTR) address

The address of the invoker of IXLCONN

Return code for the MVS macro IXLCONN

Reason code for the MVS macro IXLCONN

Request parameter header (RPH) address

CFC2 entry for coupling facility connection service (Part 2)

Entry:
- CFC2

VIT option:
- CFS

280  z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
**Event:**
Invocation of the MVS macro IXLCONN

**VIT processing module:**
ISTRACCF

This trace record is a continuation of the CFCN entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CFC2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>The actual size of the structure allocated, or 0. If IXLCONN was successful, this field contains the actual number of 4K blocks allocated to the structure.</td>
</tr>
<tr>
<td>08–0B</td>
<td>Maximum structure size. This field contains the structure size, in 4K blocks, defined in the active coupling facility resource management policy at the time this structure was allocated.</td>
</tr>
<tr>
<td>0C–0D</td>
<td>The entry portion of the entry-to-element ratio specified on IXLCONN.</td>
</tr>
<tr>
<td>0E–0F</td>
<td>The element portion of the entry-to-element ratio specified on IXLCONN.</td>
</tr>
<tr>
<td>10–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

**CFDS entry for coupling facility disconnection service (Part 1)**

**Entry:**
CFDS

**VIT option:**
CFS

**Event:**
Invocation of the MVS macro IXLDISC

**VIT processing module:**
ISTRACCF

**Control is returned to:**
ISTFSUDC

This trace record is written when a connector disconnects from the coupling facility structure with the IXLDISC macro.
Byte (hex)  
Contents  
00–03  
Record ID: C"CFDS"

04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05  
Macro parameter flags  
Bit  
Meaning  
1 ... ...  
REASON(FAILURE) was specified for the request  
.xxx xxx  
0

06  
0

07  
Connection identifier of the invoker of IXLDISC

08–0B  
Structure object (CFSSTR) address

0C–0F  
0

10–13  
The address of the invoker of IXLDISC

14–17  
Return code for the MVS macro IXLDISC

18–1B  
Reason code for the MVS macro IXLDISC

1C–1F  
Request parameter header (RPH) address

CFD2 entry for coupling facility disconnection service (Part 2)  
Entry:  
CFD2  
VIT option:  
CFS  
Event:  
Invocation of the MVS macro IXLDISC
**VIT processing module:**
ISTRACCF

This trace record is a continuation of the CFDS entry.

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| CFD2 | C | S | D | O | P | I | N | E | S | C | N | E | S | C | O | N | E | S | C | O | N |
|      | C | T | I | N | E | C | T | D | A | T | A | C | O | N | N | E | C | T | O | R | C | O |
|      | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Byte (hex)**

**Contents**

00–03
Record ID: C"CFD2"

04–07
Connector-specific disconnect data

08–0B
Maximum size

0C–1F
0

---

**CFEE entry for coupling facility connection event exit**

**Entry:**
CFEE

**VIT option:**
CFS

**Event:**
Event Exit Driven for a Connection to a Coupling Facility Structure

**VIT processing module:**
ISTRACCF

**Control is returned to:**
ISTFSXEV

This trace record is written when any of the following actions occurs:

- A new connection is established to the coupling facility structure.
- A connection to the coupling facility structure is terminated.
- A connector loses read/write access to the coupling facility structure.
- A state change occurs for the coupling facility structure.
- A user sync point is set by a connector to the coupling facility structure.
- A user sync point has been confirmed by all connectors to the coupling facility structure.
Byte (hex)
Contents
00–03
Record ID: C"CFEE"
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
0
06
Event identifier (see the mapping of IXLYEEPL in z/OS MVS Data Areas at z/OS Internet Library for an explanation of this code.)
07
The event identifier in byte 06 determines the meaning of this field.
Event flags:
• If byte 06 indicates a loss of connectivity event, use the following flags:
  Bit
  Meaning
  1... ....
  The rebuild process is in progress for the structure.
  .1... ....
  The rebuild stop process is in progress for the structure.
  ..1. ....
  This event applies to the rebuild version of the structure.
  ...1 ....
  No action will be taken. MVS will determine action based on policy information.
  .... xxxx
  0
• If byte 06 indicates an existing connection event, use the following flags:
  Bit
  Meaning
  1... ....
  The rebuild process is in progress for the structure.
  .1... ....
  The rebuild stop process is in progress for the structure.
  ..1. ....
  This event does not pertain to a real connection but indicates that all existing connection events have been received.
The connection identified is active.

The connection identified does not have access to the structure.

If byte 06 indicates a disconnected or failed event, use the following flags:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>The rebuild process is in progress for the structure.</td>
</tr>
<tr>
<td>.1..</td>
<td>The rebuild stop process is in progress for the structure.</td>
</tr>
<tr>
<td>..1.</td>
<td>This event applies to the rebuild version of the structure.</td>
</tr>
<tr>
<td>...1</td>
<td>The connection disconnected abnormally.</td>
</tr>
</tbody>
</table>

If byte 06 indicates a rebuild existing connection event, use the following flags:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>The rebuild process is in progress for the structure.</td>
</tr>
<tr>
<td>.1..</td>
<td>The rebuild stop process is in progress for the structure.</td>
</tr>
<tr>
<td>..1.</td>
<td>This event does not pertain to a real connection but indicates that all rebuild existing connection events have been received.</td>
</tr>
<tr>
<td>...1</td>
<td>The connection identified is active.</td>
</tr>
</tbody>
</table>

If byte 06 indicates structure alter begin event, use the following flags:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>The rebuild process is in progress for the structure.</td>
</tr>
<tr>
<td>.1..</td>
<td>The rebuild stop process is in progress for the structure.</td>
</tr>
<tr>
<td>..1.</td>
<td>Structure size will be altered.</td>
</tr>
<tr>
<td>...1</td>
<td>Entry-to-element ratio will be altered.</td>
</tr>
</tbody>
</table>

If byte 06 indicates structure alter end event, use the following flags:
The rebuild process is in progress for the structure.
The rebuild stop process is in progress for the structure.
Attempt to alter structure size.
Attempt to alter entry-to-element ratio.
Alter request able to meet all specified targets.
Alter request able to meet only some specified targets.

If byte 06 indicates a recommended action, use the following flags:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The rebuild process is in progress for the structure.</td>
</tr>
<tr>
<td>1</td>
<td>The rebuild stop process is in progress for the structure.</td>
</tr>
<tr>
<td>.1</td>
<td>Policy is available to determine action.</td>
</tr>
<tr>
<td>.1</td>
<td>Action is disconnect.</td>
</tr>
<tr>
<td>.xx</td>
<td>0</td>
</tr>
</tbody>
</table>

For all other events, use the following flags:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The rebuild process is in progress for the structure.</td>
</tr>
<tr>
<td>1</td>
<td>The rebuild stop process is in progress for the structure.</td>
</tr>
<tr>
<td>.1</td>
<td>VTAM ignored this event.</td>
</tr>
<tr>
<td>.x</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>08–0B</td>
<td>Structure object (CFSSTR) address</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Event sequence number. Each event is assigned a unique sequence number.</td>
</tr>
<tr>
<td>10</td>
<td>The connection identifier for the subject of this event</td>
</tr>
<tr>
<td>11</td>
<td>Connection FSM State. Connection states are listed in the VTAM data map ISTCFCON.</td>
</tr>
<tr>
<td>12</td>
<td>Reason for starting rebuild (see the mapping of IXLYEEPL in z/OS MVS Data Areas at z/OS Internet Library for an explanation of this code).</td>
</tr>
</tbody>
</table>
Reason for stopping rebuild (see the mapping of IXLYEPL in z/OS MVS Data Areas at z/OS Internet Library for an explanation of this code).

- If byte 06 indicates a rebuild event, then this field contains the reason for stopping rebuild (see the mapping of IXLYEPL in z/OS MVS Data Areas at z/OS Internet Library for an explanation of this code).

- If byte 06 indicates structure alter end event, use the following flags:

  **Bit**
  **Meaning**
  1... . . . . Alter request failed because of structure failure.
  .1... . . . . Alter request failed because of loss of connectivity.
  ..1. . . . . Alter request failed because rebuild started.
  ...1 . . . . Alter request failed because ratios specified on alter request are not consistent with structure attributes.
  .... xxxx 0

**14–17**
User Data 1

The contents of this field depend upon the event received and the state of the structure when the event was received.

- If this event pertains to a user sync point, this field contains the connector-defined event for the user sync point set if a user sync point has been set.
- If this event pertains to a disconnect or failed connection event, then this field contains the first 4 bytes of the connector-defined data specified in IXLDISC.
- If this event pertains to a rebuild connects complete event, then this field contains the number of active connectors to the original structure.
- If this event pertains to a rebuild event, other than rebuild connects complete, then this field contains the connector-defined reason for starting rebuild, if a connector-specified reason was given.
- If this event pertains to alter begin event, then the first 2 bytes contain the target entry portion of the entry-to-element ratio. The second 2 bytes contain the target element portion of the entry-to-element ratio.
- If this event pertains to an alter end event and the ratio was changed, then this field contains the current number of entries.
- If this event indicates a structure state change, the first 2 bytes of this field contain a validity flag. The validity flag byte 14 contains:

  **Bit**
  **Meaning**
  1... . . . . Coupling facility operational level of the coupling facility in which the structure resides is valid. If this bit has been set, User Data 2 contains the coupling facility operational level for the coupling facility in which the structure resides.

  Otherwise, this field contains all zeros.

**18–1B**
User Data 2
The contents of this field depend upon the event received and the state of the structure when the event was received.

If this event pertains to a user sync point, this field contains the connector-defined event for the user sync point confirmed if a user sync point has been confirmed.

If this event pertains to a disconnect or failed connection event, then this field contains the last 4 bytes of the connector-defined data specified in IXLDISC.

If this event pertains to a rebuild stop event, then this field contains the connector-defined reason for stopping the rebuild, if a connector-specified reason was given.

If this event pertains to a rebuild connects complete event, then this field contains the number of connectors that successfully connected to the rebuild structure.

If this event pertains to an alter begin event, then this field contains the target structure size.

If this event pertains to an alter end event and the ratio was changed, then this field contains the current number of elements. If the structure size was changed then this field contains the current structure size. If both ratio and size are changed, this field contains the current number of elements.

If this event indicates a structure state change and byte 14 (first byte of User Data 1 field) contains a value of X'80', this field contains the coupling facility operational level. This is the operational level of the coupling facility in which the structure resides.

Otherwise, this field contains all zeros.

1C–1F
Request parameter header (RPH) address

CFER entry for coupling facility connection event exit response service

Entry:
  CFER
VIT option:
  CFS
Event:
  Invocation of the MVS macro IXLEERSP
VIT processing module:
  ISTRACCF
Control is returned to:
  ISTFSURS
This trace record is written when a response is given to an event exit event using the IXLEERSP macro.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 1 2 3 0 4 5 6 0 7 0 8 9 A 0 C D E 0 F</td>
<td>CFER I D S U B J E C T C F S S T R A D D R E S S E N T S E Q N U M B E R R E T U R N A D D R E S S R E T U R N C O D E R E A S O N C O D E R P H A D D R E S S</td>
</tr>
</tbody>
</table>

This trace record is written when a response is given to an event exit event using the IXLEERSP macro.
00–03
   Record ID: C"CFER"

04
   ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
   The connection identifier of the subject of the original event. This field applies to Disconnected/Failed Events, Existing Connection Events, and Rebuild Connect Failure Events only.

06
   Event code (See the mapping of ISTXEEPL in z/OS Communications Server: SNA Data Areas Volume 1 for an explanation of this code.)

07
   Connection identifier of the invoker of IXLEERSP

08–0B
   Structure object (CFSSTR) address

0C–0F
   The event sequence number of the event to which a response is being given. This field applies to Disconnected/Failed Events, Existing Connection Events, and Rebuild Connect Failure Events only. Otherwise, this field contains a 0.

10–13
   The address of the invoker of IXLEERSP

14–17
   Return code for the MVS macro IXLEERSP

18–1B
   Reason code for the MVS macro IXLEERSP

1C–1F
   Request parameter header (RPH) address

**CFFC entry for coupling facility IXLFORCE service**

Entry:
   CFFC

VIT option:
   CFS

Event:
   Invocation of the MVS macro IXLFORCE

VIT processing module:
   ISTRACCF

Control is returned to:  
   ISTFSUFC

**CFLS entry for coupling facility IXLLIST service (Part 1)**

Entry:
   CFLS

VIT option:
   CFS

Event:
   Invocation of the MVS macro IXLLIST
VIT processing module:
ISTRACCF

Control is returned to:
Module invoking the IXLLIST macro that caused the record to be produced.

This trace record is written for a request to access the coupling facility structure using the IXLLIST macro.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"CFLS"
| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05         | 0
| 06         | The function requested by this invocation of IXLLIST:
| X'01'      | REQUEST(READ) specified
| X'02'      | REQUEST(READ_LCONTROLS) specified
| X'03'      | REQUEST(WRITE) specified
| X'04'      | REQUEST(MOVE) specified
| X'05'      | REQUEST(DELETE) specified
| X'06'      | REQUEST(LOCK) specified
| X'07'      | REQUEST(MONITOR_LIST) specified
| X'08'      | REQUEST(READ_LIST) specified
| X'09'      | REQUEST(DELETE_MULTI) specified
| X'0A'      | REQUEST(WRITE_LCONTROLS) specified
| X'0B'      | REQUEST(DELETE_ENTRYLIST) specified
| X'0C'      | REQUEST(READ_MULT) specified
Flags

Bit

Meaning

1... .... Request will complete asynchronously. Bytes X'14' – X'1B' will contain the request data.

.1... .... Request is for the rebuild version of the structure.

..1. .... The buffer list contains real addresses.

...x xxxx

0

08–0B Structure object (CFSSTR) address

0C–0F Buffer object (CFSBUF) address or 0 if not using a buffer object.

10–13 The address of the invoker of IXLLIST

14–17 Return code for the MVS macro IXLLIST or the first 4 bytes of request data. If this IXLLIST invocation will complete asynchronously, this field holds the first 4 bytes of the request data specified on this request. Otherwise it holds the return code for this invocation of IXLLIST.

18–1B Reason code for the MVS macro IXLLIST or the second 4 bytes of request data. If this IXLLIST invocation will complete asynchronously, this field holds the second 4 bytes of the request data specified on this request. Otherwise it holds the reason code for this invocation of IXLLIST.

When this field represents the second 4 bytes of the request data, the tracing of the buffer contents is deferred until the completion exit is driven, as follows:

Bit

Meaning

..1. .... When this bit is on at the completion exit, the buffer contents for the DELETE_ENTRYLIST are traced using the CFLS VIT entries.

1C–1F Request parameter header (RPH) address

CFL2 entry for IXLLIST lock service (Part 2)

Entry:

CFL2

VIT option:

CFS

Event:

Invocation of the MVS macro IXLLIST

VIT processing module:

ISTRACCF

This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a LOCK request.
Byte (hex)
Contents
00–03
  Record ID: C"CFL2"
04
  Function code = X'06'
05
  The value specified for the MODE parameter:
    • X'01'MODE(SYNCSUSPEND) specified
    • X'02'MODE(SYNCEXIT) specified
    • X'03'MODE(AYSNCEXIT) specified
06
  The value specified for the LOCKOPER parameter:
    • X'01'LOCKOPER(SET) specified
    • X'02'LOCKOPER(RESET) specified
07
  The value specified for the LOCKCOMP parameter. This 1-byte value represents the connection
  identifier of the connector that is assumed to own a given lock.
08–0B
  The index of the lock that is being manipulated.
0C–1F
  0

CFL2 entry for IXLLIST list controls services (Part 2)
Entry:
  CFL2
VIT option:
  CFS
Event:
  Invocation of the MVS macro IXLLIST with REQUEST READ_LCONTROLS or WRITE_LCONTROLS
VIT processing module:
  ISTRACCF

This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a
READ_LCONTROLS or WRITE_LCONTROLS request.
CFL2 mode (Part 3)

**Byte (hex)**

**Contents**

**00–03**
- Record ID: C"CFL2"

**04**
- Function code:
  - X'02' REQUEST(READ_LCONTROLS) specified
  - X'0A' REQUEST(WRITE_LCONTROLS) specified

**05**
- The value specified for the MODE parameter:
  - X'01' MODE(SYNCSUSPEND) specified
  - X'02' MODE(SYNCEXIT) specified
  - X'03' MODE(AYSNCEXIT) specified

**06–07**
- 0

**08–0B**
- The list number for which the data is being retrieved or updated

**0C–0F**
- If request is READ_LCONTROLS, the number of entries currently in use on the list; otherwise 0

**10–13**
- If the request is WRITE_LCONTROLS, the LISTKEY value; otherwise 0

**14–1B**
- 0

**1C–1F**
- The first 4 bytes of the 32-byte list descriptor area

**CFL3 entry for IXLLIST service (REQUEST LCONTROLS) (Part 3)**

**Entry:**
- CFL3

**VIT option:**
- CFS

**Event:**
- Invocation of the MVS macro IXLLIST with REQUEST READ_LCONTROLS or WRITE_LCONTROLS

**VIT processing module:**
- ISTRACCF

This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a READ_LCONTROLS or WRITE_LCONTROLS request.
Byte (hex)
Contents
00–03
Record ID: C"CFL3"
04–1F
The last 28 bytes of the 32-byte list descriptor area

CFL2 entry for IXLLIST services (single entry) (Part 2)

Entry:
CFL2
VIT option:
CFS
Event:
Invocation of the MVS macro IXLLIST with REQUEST READ, WRITE, MOVE, or DELETE
VIT processing module:
ISTRACCF
This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a READ, WRITE, MOVE, or DELETE request.

Byte (hex)
Contents
00–03
Record ID: C"CFL2"
04
Function code:
• X'01'REQUEST(READ) specified
• X'03'REQUEST(WRITE) specified
• X'04'REQUEST(MOVE) specified
• X'05'REQUEST(DELETE) specified
05
The value specified for the MODE parameter:
• X'01'MODE(SYNCSUSPEND) specified
• X'02'MODE(SYNCEXIT) specified
• X'03'MODE(ASYNCEXIT) specified

06
If byte X'04' indicates a MOVE or DELETE function, use the following information to determine the DATAOPER parameter:
• X'01'DATAOPER(NONE) specified
• X'02'DATAOPER(READ) specified
• X'03'DATAOPER(WRITE) specified

07
0
08–0B
The list number for which the data is being retrieved or updated

0C–0F
• If the request is READ, WRITE, or DELETE, the Access List Entry Token (ALET) of the data buffers in the buffer object
• If the request is MOVE, the LISTKEY INCREMENT value

10–1F
The value of any ENTRYNAME, ENTRYID, or ENTRYKEY parameter specified

CFL2 entry for IXLLIST services (multiple entries) (Part 2)

Entry:
CFL2
VIT option:
CFS
Event:
Invocation of the MVS macro IXLLIST with REQUEST READMULT, READLIST, or DELETEMULT
VIT processing module:
ISTRACCF
This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a READMULT, READLIST or DELETEMULT request.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CFL2&quot;</td>
</tr>
<tr>
<td>04</td>
<td>Function code:</td>
</tr>
<tr>
<td></td>
<td>• X'08'REQUEST(READ_LIST) specified</td>
</tr>
</tbody>
</table>
• X'09'REQUEST(DELETE_MULT) specified
• X'0C'READ(READ_MULT) specified

05
The value specified for the MODE parameter:
• X'01'MODE(SYNCSUSPEND) specified
• X'02'MODE(SYNCEXIT) specified
• X'03'MODE(AYSNCEXIT) specified

06–07
The number of processed entries. If the number is greater than 65635, the number will be X'FFFF'.

08–0B
The list number for which the data is being retrieved or deleted

0C–0F
Access List Entry Token (ALET) of the data buffers in the buffer object

10–1F
For READLIST request, the 12-byte restart entry ID padded to the right with zeros. For READMULT and DELETEMULT, the 16-byte restart token.

CFL3 entry for IXLLIST services (REQUEST entries) (Part 3)

Entry:
CFL3
VIT option:
CFS
Event:
Invocation of the MVS macro IXLLIST with REQUEST READ, WRITE, MOVE or DELETE, READMULT, READLIST, or DELETEMULT
VIT processing module:
ISTRACCF
This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a REQUEST READ, WRITE, MOVE or DELETE, READMULT, READLIST, or DELETEMULT request. The VTAM internal trace will generate up to five CFL3 entries.

| 0 0 0 0 | 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | 0 0 0 0 |
| CFL3 | 28 BYTES OF BUFFER LIST |

Byte (hex)
Contents

00–03
Record ID: C"CFL3"

04–1F
Up to 28 bytes of the contents of the buffer list

CFL2 entry for IXLLIST delete entry list service (Part 2)

Entry:
CFL2
VIT option:
CFS
**Event:**
Invocation of the MVS macro IXLLIST with REQUEST DELETE_ENTRYLIST

**VIT processing module:**
ISTRACCF

This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a DELETE_ENTRYLIST request.

| 00–03 | Record ID: C"CFL2"
|-------|-----------------|
| 04    | Function code = X'0B'
| 05    | The value specified for the MODE parameter:
| 06–07 | X'01'MODE(SYNCSUSPEND) specified
| 08–09 | Index of the first entry ID processed
| 0A–0B | Index of the last entry ID processed
| 0C–0F | Access List Entry Token (ALET) of the buffer holding the entry IDs to be deleted
| 10–1F | 0

**CFL3 entry for IXLLIST delete entry list service (Part 3)**

**Entry:**
CFL3

**VIT option:**
CFS

**Event:**
Invocation of the MVS macro IXLLIST with REQUEST DELETE_ENTRYLIST

**VIT processing module:**
ISTRACCF
This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a
REQUEST DELETE_ENTRYLIST request. The VTAM internal trace will generate up to 36 CFL3 entries.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"CFL3"
| 04–1F      | Up to 28 bytes of the contents of the buffer containing entry IDs to be deleted. The buffer contents
            traced will begin at the index of the first entry (specified at offset X'04' in CFL2) and end at the index
            of the second entry (specified at offset X'0C' in CFL2) if 36 CFL3 entries have not been generated.

### CFL2 entry for IXLLLIST monitor service (Part 2)

**Entry:**
- CFL2

**VIT option:**
- CFS

**Event:**
- Invocation of the MVS macro IXLLIST with REQUEST MONITOR_LIST

**VIT processing module:**
- ISTRACCF

This trace record is a continuation of the CFLS entry when byte X'06' in the CFLS entry indicates a
MONITOR_LIST request.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"CFL2"
| 04         | Function code = X'07'
| 05         | The value specified for the MODE parameter:

- X'01'MODE(SYNCSUSPEND) specified
- X'02'MODE(SYNCEXIT) specified
- X'03'MODE(AYSNCEXIT) specified
The value specified for the ACTION parameter:
- X'01' ACTION(START) specified
- X'02' ACTION(STOP) specified

The list number for which the monitoring is being started or stopped

The list monitoring vector index being assigned for the list

CFNF entry for event notification

Entry:
  CFNF

VIT option:
  CFS

Event:
  Event Notification Facility (ENF) Exit Driven for Event X'35'

VIT processing module:
  ISTRACCF

Control is returned to:
  ISTFSXEF

This trace record is written when a change in coupling facility resources or in the coupling facility structure has occurred that may now allow a connection to a structure that had previously failed.

| 00  | 01  | 02  | 03  | 04  | 05  | 06  | 07  | 08  | 09  | 0A  | 0B  | 0C  | 0D  | 0E  | 0F  | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 1A  | 1B  | 1C  | 1D  | 1E  | 1F  |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 00  | 01  | 02  | 03  | 04  | 05  | 06  | 07  | 08  | 09  | 0A  | 0B  | 0C  | 0D  | 0E  | 0F  | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 1A  | 1B  | 1C  | 1D  | 1E  | 1F  |
| CFNF| 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   |

Byte (hex)

Contents

00–03
  Record ID: C"CFNF"

04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
  0

06
  Exit function codes pertaining to ENF Events with event code X'35':
  X'01'
    A coupling facility resource has become available that may allow a structure to successfully connect.
X'02'
A specific coupling facility structure has become available.

07
0

08–0B
Structure object (CFSSTR) address

0C–1B
If the exit function code is X'02', this contains the structure name. Otherwise, it contains 0.

1C–1F
Request parameter header (RPH) address

**CFPG entry for IXLPURGE service**

**Entry:**
CFPG

**VIT option:**
CFS

**Event:**
Invocation of the MVS macro IXLPURGE

**VIT processing module:**
ISTRACCF

**Control is returned to:**
ISTFSNPG

This trace record is written when VTAM issues an IXLPURGE macro to purge outstanding accesses to the coupling facility structure.

<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</th>
<th>0 1 2 3 4 5 6 7 8 9 A B C D E F 0 1 2 3 4 5 6 7 8 9 A B C D E F</th>
</tr>
</thead>
<tbody>
<tr>
<td>CF PG I D 0 F L A G S CFSSTR ADDRESS 0 RETURN ADDRESS MVS RETURN CODE MVS REASON CODE RPH ADDRESS</td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

| 00–03 | Record ID: C"CFPG"
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05–06 | 0
| 07 | Event flags:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
</table>

1... ....
   Purged by request ID (REQID)
1.. ....
   Purged by address space token (STOKEN)
..1. ....
   Purged by task token (TTOKEN)
...x xxxx
   0

08–0B
   Structure object (CFSSTR) address.
0C–0F
   0
10–13
   Address of the invoker of IXLPURGE.
14–17
   Return code from the MVS macro IXLPURGE.
18–1B
   Reason code from the MVS macro IXLPURGE.
1C–1F
   Request parameter header (RPH) address.

CFP2 entry for IXLPURGE service (Part 2)

   Entry:
      CFP2
   VIT option:
      CFS
   Event:
      Invocation of the MVS macro IXLPURGE
   VIT processing module:
      ISTRACCF
   This trace record is a continuation of the CFPG entry.

| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 0 1 2 3 4 5 6 7 8 9 A B C D E F |
| 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 |
| CFP2 0 REQUEST IDENTIFIER, ADDRESS SPACE TOKEN OR TASK TOKEN |

Byte (hex)  
Contents
00–03  
   Record ID: C"CFP2"
04–0F  
   0
10–1F  
   If purging by request identifier, the 8-byte request identifier, padded to the right with zeros.
   If purging by address space, the 8-byte address space token, padded to the right with zeros.
   If purging by task, the 16-byte address space token.
CFRB entry for structure rebuild service

Entry:
  CFRB

VIT option:
  CFS

Event:
  Invocation of the MVS macro IXLREBLD

VIT processing module:
  ISTRACCF

Control is returned to:
  ISTFSURB

This trace record is written when a rebuild of the coupling facility structure is started, stopped, or completed, using the IXLREBLD macro.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"CFRB"
| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05         | Macro Parameter Flags

**Bit Meaning**

- **1... .**
  STARTREASON(LOSSCONN) was specified for the request.

- **1... .**
  STARTREASON(STRFAILURE) was specified for the request.

- **1... .**
  STOPREASON(LOSSCONNOLD) was specified for the request.

- **1... .**
  STOPREASON(LOSSCONNEW) was specified for the request.

- **1... .**
  STOPREASON(STRFAILUREOLD) was specified for the request.

- **1... .**
  The rebuild was started or stopped for a connection-specific reason. Bytes X'0C'–X'0F' contain the connection-specific reason.
06
The function requested by this invocation of IXLREBLD:

X’01’
REQUEST(START) was specified.

X’02’
REQUEST(STOP) was specified.

X’03’
REQUEST(COMplete) was specified.

07
Connection identifier of the invoker of IXLREBLD

08–0B
Structure object (CFSSTR) address

0C–0F
Connection-specific reason for starting or stopping the rebuild.

10–13
The address of the invoker of IXREBLD

14–17
Return code for the MVS macro IXLREBLD

18–1B
Reason code for the MVS macro IXLREBLD

1C–1F
Request parameter header (RPH) address

CFTP entry for coupling facility TCP/IP interface requests

Entry:
CFTP

VIT option:
CFS

Event:
Completion of a request from TCP/IP to CFS

VIT processing module:
ISTRACCF

Control is returned to:
ISTFSDPC or ISTFSVPC

This trace record is written when a request from TCP/IP to the coupling facility completes.

 Byte (hex)  |  Contents
-------------|------------------
 00 00 00 00  | CFTP ID FUNCTION OBJECT ADDRESS CFUSR ADDRESS TCP NAME RETURN CODE RPH ADDRESS
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

The TCP/IP function identifier:

- X'01' Sysplex Wide Security Associations
- X'02' Sysplexports

The operation requested by TCP/IP for the following function:

- Sysplex Wide Security Associations
  - X'01' RegisterUser
  - X'02' DeregisterUser
  - X'03' ClaimList
  - X'04' UpdateEntry
  - X'05' DeleteEntry
  - X'06' DeleteMult
  - X'07' FreeList
  - X'08' ReceiveData
  - X'09' RepopulateComplete
  - X'0A' QuiesceComplete
  - X'0B' InitSeq#
  - X'0C' GetSeq#
  - X'0D' QuerySeq#
  - X'0E' ValidateList
  - X'0F' FreeAll
- Sysplexports
  - X'01' RegisterUser
X'02'
  DeregisterUser
X'03'
  AssociateStack
X'04'
  GetEphemeralPort
X'05'
  MarkEphemeralPort
X'06'
  UnassociateStack
X'07'
  FreeList
X'08'
  QueryList
X'09'
  RepopulateComplete
X'0A'
  QuiesceComplete
X'0B'
  GetEphemeralPortBlock
X'0C'
  FreeEphemeralPortBlock
X'0D'
  SetExplicitBindPortRange
X'0E'
  GetExplicitBindEphemeralPortBlock
X'10'
  FreeExplicitBindEphemeralPortBlock

07
  TCP/IP request flags

  Bit
    Meaning
    1...  . . .    Request issued as part of repopulation processing
    .xxx  xxxx

08–0B
  Structure object address

0C–0F
  CFS TCP/IP user control block (CFUSR) address

10–17
  The TCP/IP stack name

18–1B
  Return code for the request

1C–1F
  Request parameter header (RPH) address
CFT2 entry for coupling facility TCP/IP interface requests (Part 2)

Entry:
  CFT2

VIT option:
  CFS

Event:
  Completion of a request from TCP/IP to CFS

VIT processing module:
  ISTRACCF

This trace record is a continuation of the CFTP entry. As many of these as necessary are generated to display the entire parameter list.

| 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 |
|-------------------|-------------------|
| CFT2              | PARAMETER LIST DATA (UP TO 28 BYTES) |

**Byte (hex) **

**Contents**

**00–03**
  Record ID: C"CFT2"

**04–1F**
  28 bytes of the parameter list data for this request

CFTX entry for coupling facility TCP/IP event exit invocation

Entry:
  CFTX

VIT option:
  CFS

Event:
  Invocation of the CFS TCP/IP event exit

VIT processing module:
  ISTRACCF

Control is returned to:
  ISTFSNTX

This trace record is written when an asynchronous event causes CFS to drive the TCP/IP event exit routine.
### Byte (hex)

#### Contents

**00–03**  
Record ID: C"CFTX"

**04**  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**  
The TCP/IP function identifier:

- X'01'  
  Sysplex Wide Security Associations
- X'02'  
  Sysplexports

**06**  
The event code describing the asynchronous event:

- X'01'  
  Quiesce
- X'02'  
  Repopulate/Rebuild
- X'03'  
  Repopulate/Reconnect
- X'04'  
  Deregistered
- X'05'  
  Connection Attempt Failed
- X'06'  
  Received Data
- X'07'  
  DeleteMult Complete
- X'08'  
  FreeList Complete
- X'09'  
  FreeAll Complete

**07**  
Return code:

- X'00'  
  Success
- X'08'  
  Failure
08–0B
Structure object address

0C–0F
Request Correlation ID - Correlates this event with the original request

10–17
The TCP/IP stack name

18–1B
Data buffer address, if event is "Received Data"; otherwise 0

1C–1F
Request parameter header (RPH) address

---

**CFUS entry for user sync point service**

**Entry:**
CFUS

**VIT option:**
CFS

**Event:**
Invocation of the MVS macro IXLUSYNC

**VIT processing module:**
ISTRACCF

**Control is returned to:**
ISTFSUUS

This trace record is written when a connector sets or confirms a user sync point for a coupling facility structure with the IXLUSYNC macro.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CFUS&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Function requested by this invocation of IXLUSYNC:</td>
</tr>
<tr>
<td></td>
<td>X'01'</td>
</tr>
<tr>
<td></td>
<td>SET function requested</td>
</tr>
</tbody>
</table>

![Trace record format diagram]

---

308  z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
CONFIRM function requested
CONFIRMSET function requested

07
Connection identifier of the invoker of IXLUSYNC

08–0B
Structure object (CFSSTR) address

0C–0F
The value of USEREVENT parameter for this invocation of IXLUSYNC

10–13
The address of the invoker of IXLUSYNC

14–17
Return code for the MVS macro IXLUSYNC

18–1B
Reason code for the MVS macro IXLUSYNC

1C–1F
Request parameter header (RPH) address

**CFVC entry for IXLVECTR service**

<table>
<thead>
<tr>
<th>Entry:</th>
<th>CFVC</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT option:</td>
<td>CFS</td>
</tr>
<tr>
<td>Event:</td>
<td>Invocation of the MVS macro IXLVECTR</td>
</tr>
<tr>
<td>VIT processing module:</td>
<td>ISTRACCF</td>
</tr>
<tr>
<td>Control is returned to:</td>
<td>ISTFSLVC</td>
</tr>
</tbody>
</table>

This trace record is written when VTAM issues an IXLVECTR macro to modify or test the list notification vector.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CFVC&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
</tbody>
</table>
Function codes

X’01’
Modify vector size

X’02’
Load and test a range of vector entries

X’03’
Test a single list state.

Structure object (CFSSTR) address.

If byte 06 indicates a modify vector size, then this field contains the new length of the vector.
If byte 06 indicates a load and test function, then this field contains the starting vector index to begin testing. Thirty-two consecutive bits will be tested.
If byte 06 indicates a test function, then this field contains the vector index being tested.

Address of the invoker of IXLVECTR.

Return code from the MVS macro IXLVECTR. See the z/OS MVS Programming: Sysplex Services Reference for an explanation of this code.

If byte 06 indicates a modify vector size function, then this field contains the actual length of the new vector.
If byte 06 indicates a load and test function, then this field contains a 32-bit string where each bit indicates the following information:

0  List is not empty
1  List is empty

Request parameter header (RPH) address.

CHGO entry for CHANGE_OWNER requests

Entry:
CHGO

VIT option:
CSM

Event:
IVTCSM REQUEST=CHANGE_OWNER

VIT processing module:
ISTITCCS

Control is returned to:
IVTSMCCO or IVTSM6CO
Notes:
- IVTSMCCO processes CHANGE_OWNER requests in 31-bit addressing mode.
- IVTSM6CO processes CHANGE_OWNER requests in 64-bit addressing mode.

This trace record provides the status of an IVTCSM REQUEST=CHANGE_OWNER macroinstruction.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CHGO&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace record flag:</td>
</tr>
<tr>
<td>07</td>
<td>Trace record number to correlate all the entries for this particular event.</td>
</tr>
<tr>
<td>08–09</td>
<td>0</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Owner ID</td>
</tr>
<tr>
<td>0C–0D</td>
<td>Return code</td>
</tr>
<tr>
<td>0E–0F</td>
<td>Reason code</td>
</tr>
<tr>
<td>10–13</td>
<td>Address of utility routine caller or return address of the issuer of the IVTCSM macroinstruction</td>
</tr>
<tr>
<td>14–17</td>
<td>Last buffer list entry. The last buffer list entry is 0 if the return code is 0 and nonzero if the return code is nonzero. When an error occurs in 64-bit addressing mode, the last buffer list entry here is X'FFFFFFF'. The CHG6 entry follows with the 64-bit address of the last buffer list entry.</td>
</tr>
<tr>
<td>18–1B</td>
<td>Number of buffers</td>
</tr>
</tbody>
</table>
1C–1F
THREAD value if specified or 0 if THREAD is not specified. The THREAD value is used only to correlate
this trace record to a specific IVTCSM macroinstruction.

CHG2 entry for CHANGE_OWNER requests

Entry:
CHG2

VIT option:
CSM

Event:
IVTCSM REQUEST=CHANGE_OWNER

VIT processing module:
ISTITCCS

This trace record is a continuation of the CHGO trace record. Each CHG2 record traces, at most, two
buffers that were requested for ownership change.

| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | B0 | B1 | B2 | B3 | C0 | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | E0 | E1 | E2 | E3 | E4 | E5 | E6 | E7 | E8 | E9 | F0 | F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 | F10 | F11 | F12 | F13 | F14 | F15 |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  |
| CHG2 | 0 | FLAG | RECORDNUM | BUFFER TOKEN FOR INPUT BUFFER LIST ENTRY | BUFFER TOKEN FOR INPUT BUFFER LIST ENTRY OR 0 |

Byte (hex)

Contents

00–03
Record ID: C"CHG2"

04–05
0

06
Trace record flag:

B'0... ...'
Indicates that this is the last record for this event.

B'1... ...'
Indicates that additional records exist for this event. Use the trace record number from this entry
to locate corresponding continuation records.

07
Trace record number to correlate all the entries for this particular event

08–13
Buffer token contained in input buffer list entry

14–1F
Buffer token contained in input buffer list entry or 0

CHG6 entry for CHANGE_OWNER requests

Entry:
CHG6

VIT option:
CSM
Event:  
IVTCSM REQUEST=CHANGE_OWNER

VIT processing module:  
ISTITCCS

This trace record is a continuation of the CHGO trace record. It contains the 64-bit buffer list entry address. This entry is present only when the CHANGE_OWNER request failed with the error return code in 64-bit addressing mode.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03</td>
<td>Record ID: C'CHG6'</td>
</tr>
<tr>
<td>04-05</td>
<td>0</td>
</tr>
</tbody>
</table>
| 06         | Trace record flags:  
  B'0... ....'  
  Indicates that this is the last record for this event.  
  B'1... ....'  
  Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.  
  B'.1. ....'  
  Indicates that the buffer list is in 64-bit storage. |
| 07         | Trace record number to correlate all the entries for this particular event. |
| 08-F       | 0        |
| 10-17      | 64-bit address of the last buffer list entry. |
| 18-1F      | 0        |

CI1 or CO1 trace entries

This trace record provides information about the inbound and outbound requests sent to session services to begin or end a same-network or cross-network LU-LU session.

The CI1 and CO1 traces have two formats:

- **Format 0** is used for same-network sessions or when VTAM does not know whether the request is for a cross-network session. This format contains parts 1, 2, and 3 of the CIn or COn record.
- **Format 1** is used for cross-network sessions. This format contains parts 1, 2, 3, and 4 of the CIn or COn record.
CI1 or CO1 entry for SSCP (RUPE — Part 1)

Entry:
CI1 or CO1

VIT option:
SSCP

Event:
Requests (Part 1)

VIT processing module:
ISTRACSC

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced

Part 1 of the CI\textsubscript{1} or CO\textsubscript{1} record is for both format 0 and format 1 and contains essentially the same information as the CI\textsubscript{1} or CO\textsubscript{1} entry.

If this entry is associated with an event failure (that is, the sense data is nonzero), this entry is generated whether the SSCP option is in effect. It is treated as an exception condition and, therefore, is traced whenever the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CI1&quot; for inbound processing</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CO1&quot; for outbound processing</td>
</tr>
<tr>
<td>03</td>
<td>Flags</td>
</tr>
<tr>
<td></td>
<td>Bit</td>
</tr>
<tr>
<td></td>
<td>Meaning</td>
</tr>
<tr>
<td>0</td>
<td>Requests</td>
</tr>
<tr>
<td>1</td>
<td>Response</td>
</tr>
<tr>
<td>.0</td>
<td>No sense</td>
</tr>
<tr>
<td>.1</td>
<td>Sense</td>
</tr>
<tr>
<td>.... .00</td>
<td>Format 0</td>
</tr>
<tr>
<td>.... .01</td>
<td>Format 1</td>
</tr>
</tbody>
</table>

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

314 z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
05
Control block ID=X’54’

06–07
Initiation and termination finite state machines (taken from the session information block, SIB)

08–0B
Save area address

0C–0F
Save area ID (bytes 4, 5, 7, and 8 of the module name if available)

10–13
Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)

14–19
Network address at origin (RUPEOAF)

1A–1F
Network address at destination (RUPEDAF)

CI1 or CO1 entry for SSCP (NCSPL — Part 1)

Entry:
CI1 or CO1

VIT option:
SSCP

Event:
Requests (Part 1)

VIT processing module:
ISTRACSC

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0...</td>
<td>Requests</td>
</tr>
<tr>
<td>1...</td>
<td>Response</td>
</tr>
</tbody>
</table>

00–02
Record ID:
- C"CI1" for inbound processing
- C"CO1" for outbound processing

03
Flags

00–02
Record ID:
- C"CI1" for inbound processing
- C"CO1" for outbound processing

03
Flags

00–02
Record ID:
- C"CI1" for inbound processing
- C"CO1" for outbound processing

03
Flags

00–02
Record ID:
- C"CI1" for inbound processing
- C"CO1" for outbound processing

03
Flags
No sense
Sense
Format 0
Format 1

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Control block ID=X'60'

06
Flags (CPCBFL)

07
CPCB return code (CPCBRC)

08–0B
Save area address

0C–0F
Save area ID (bytes 4, 5, 7, and 8 of the module name if available)

10–13
Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)

14–17
CPCB operation code (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)

18
WTD type

19
0

1A–1F
Network address at destination (NCSPLDAF)

CI1 or CO1 entry for SSCP (not RUPE or NCSPL — Part 1)

Entry:
CI1 or CO1

VIT option:
SSCP

Event:
Requests (Part 1)

VIT processing module:
ISTRACSC

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced
Byte (hex)

**Contents**

**00–02**

Record ID:

- C"CI1" for inbound processing
- C"CO1" for outbound processing

**03**

Flags

**Bit**

**Meaning**

0... .... Requests

1... .... Response

.0... .... No sense

.1... .... Sense

.... ..00 Format 0

.... ..01 Format 1

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

Control block ID (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)

**06**

CPCB flag (CPCBFL)

**07**

Return code (CPCBRC)

**08–0B**

Save area address

**0C–0F**

Save area ID (bytes 4, 5, 7, and 8 of the module name if available)

**10–13**

Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)

**14–17**

CPCB operation code (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)
CI2 or CO2 entry for SSCP (RUPE - Part 2)

Entry:
  CI2 or CO2

VIT option:
  SSCP

Event:
  Requests (Part 2)

VIT processing module:
  ISTRACSC

The CI2 and CO2 trace records are continuations of the CI1 and CO1 entries.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>C</td>
<td>C</td>
<td>2</td>
<td>C</td>
<td>B</td>
<td>I</td>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FIRST 24 BYTES OF RU

SENSE DATA
OR
SEQ NUM
OR
0

Byte (hex)

Contents

00 - 02
  Record ID:
    • C"CI2" for inbound processing
    • C"CO2" for outbound processing

03
  Control block ID=X'54'

04 - 1B
  First 24 bytes of RU

1C - 1F
  Sense data, or sequence number, or 0

CI2 or CO2 entry for SSCP (NCSPL — Part 2)

Entry:
  CI2 or CO2

VIT option:
  SSCP

Event:
  Requests (Part 2)

VIT processing module:
  ISTRACSC

This trace record is a continuation of the CI1 or CO1 trace record.
CI2 or CO2 entry for SSCP (not RUPE or NCSPL — Part 2)

Entry:
CI2 or CO2

VIT option:
SSCP

Event:
Requests (Part 2)

VIT processing module:
ISTRACSC

This trace record is a continuation of the CI1 or CO1 trace record.

Byte (hex)  Contents
00–02  Record ID:
• C"CI2" for inbound processing
• C"CO2" for outbound processing

03  Control block ID=X'60'

04–07  RDTE state (in control block RPRE). This value, a resource status code, is explained in z/OS Communications Server: IP and SNA Codes and in control block FSM. See z/OS Communications Server: SNA Data Areas Volume 1.

08–0B  CPCB WTD (first 4 bytes)

0C–1F  0

CTAM internal trace (VIT) record descriptions 319
03
Control block ID (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)

04 - 1F
0

CI3 or CO3 entry for SSCP (RUPE, NCSPL, or not RUPE or NCSPL — Part 3)

Entry:
CI3 or CO3

VIT option:
SSCP

Event:
Requests (Part 3)

VIT processing module:
ISTRACSC

Part 3 of the CIIn or COn record is for both format 0 and format 1 and contains the address of the ISTSIB for the session, the ISTPCID, and the PLU and SLU names.

Note: If this is the last CIIn or COn entry, these are the PLU and SLU names as known in this network. If this entry is followed by CI4 or CO4, these are just the PLU and SLU names. The CI4 or CO4 entry identifies the network.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CI3&quot; for inbound processing</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CO3&quot; for outbound processing</td>
</tr>
</tbody>
</table>

03
0

04–07
SIB address

08–0F
PCID for this request

10–17
PLU name

18–1F
SLU name

CI4 or CO4 entry for SSCP (RUPE, NCSPL, or not RUPE or NCSPL — Part 4)

Entry:
CI4 or CO4
VIT option:
SSCP

Event:
Requests (Part 4)

VIT processing module:
ISTRACSC

Part 4 of the CIn or COn entry is for format 1 only.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CI4&quot; for inbound processing</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CO4&quot; for outbound processing</td>
</tr>
<tr>
<td>03–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>PCID in previous request</td>
</tr>
<tr>
<td>10–17</td>
<td>PLU network ID</td>
</tr>
<tr>
<td>18–1F</td>
<td>SLU network ID</td>
</tr>
</tbody>
</table>

CMER entry for error in CMIP services (Part 1)

Entry:
CMER

VIT option:
None (Generated by CMIP services)

Event:
Error discovered in CMIP services (Part 1)

VIT processing module:
ISTITCCM

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced

This trace record is generated when an error is detected in CMIP services. For long CMIP strings, a maximum of seven CME2 trace records are generated. For longer CMIP strings, an additional CMER trace record with a maximum of seven CME2 trace records are generated.

For detailed descriptions of the fields, see z/OS Communications Server: CMIP Services and Topology Agent Guide.

This entry is always traced, regardless of the VIT options specified.
### Byte (hex) Contents

**00–03**  
Record ID: C"CMER"

**04**  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**  
0

**06–07**  
Error code indicating which error is detected

**08–0B**  
Place code indicating where the error is detected

**0C–1B**  
The first 16 bytes of data (padded on the right with zeros). 0 if there is no data. The data consists of one or more variables. Each variable is preceded by a 2-byte field indicating its length in bytes. The length of the data itself, not the number of bytes being traced, is used.

If the data is too long to fit in the VIT record, the length field does not match the number of bytes traced.

**1C–1F**  
Request parameter header (RPH) address

### CME2 entry for error in CMIP services (Part 2)

**Entry:**  
CME2

**VIT option:**  
None (Generated by CMIP services)

**Event:**  
Error discovered in CMIP services (Part 2)

**VIT processing module:**  
ISTITCCM

This trace record is a continuation of the CMER entry. It contains 28 more bytes of the CMIP error data. A maximum of seven CME2 trace records are generated. The number of records depends on the length of the data.
CNA entry for VCNSCMD issued by application

Entry:
CNA

VIT option:
VCNS

Event:
VCNSCMD issued by application

VIT processing module:
ISTRACNS

Control is returned to:
ISTAICVC

This CNA trace record shows VCNS command requests. The API routine ISTAICVC writes the entry and then queues the RPL to the correct PAB.

<table>
<thead>
<tr>
<th>CNA</th>
<th>ID</th>
<th>EXIT</th>
<th>DS</th>
<th>RPL ADDRESS</th>
<th>RPL3 ADDRESS</th>
<th>RPL AREA</th>
<th>VCNS FLAGS</th>
<th>OPERAND-DEPENDENT DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>23</td>
<td>04</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
</tr>
<tr>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>AB</td>
<td>C</td>
<td>DE</td>
<td>F</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td>1A</td>
<td>1B</td>
<td>1C</td>
<td>1D</td>
<td>1E</td>
<td>F0</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

00–03
Record ID: C"CME2"

04–1F
Up to 28 bytes of data (padded on the right with zeros)

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Exit definition (RPLEXTDS)

Bit

 Meaning

1..... Indicates RPL exit was scheduled

.1..... Indicates RPL exit not specified

..1.... Indicates RPL exit was specified

....x xx.. Indicates BRANCH=YES specified
Option code byte 1 (RPLOPT1)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>xxxx</td>
<td>0</td>
</tr>
<tr>
<td>....</td>
<td>1... Asynchronous request indicator</td>
</tr>
<tr>
<td>...x</td>
<td>0</td>
</tr>
<tr>
<td>....</td>
<td>1... External ECB indicator</td>
</tr>
</tbody>
</table>

VCNSCMD CONTROL= operand value

| 08  | INQUIRE                                      |
| 10  | LOGON                                        |
| 14  | TEST                                         |
| 18  | XID                                          |
| 20  | LOGOFF                                       |
| 40  | STATUS                                       |
| 50  | SET                                          |
| 80  | SEND                                         |
| C0  | CHECK                                        |
| E0  | SETCPARM                                     |
| F0  | RECEIVE                                      |

RPL address

RPL3 address (RPLAAREA)

Address of area into which data is to be read or from which data is to be written (RPLAREA), or 0 if no area is supplied

VCNS flag byte
Bit
Meaning
1.... Q-bit indicator
.1... M-bit indicator
..1. D-bit indicator
....1 Buffer list indicator
.... xxxx 0

15
VCNS flag byte

Bit
Meaning
00... Continue specific
01... Continue any
11... Continue same
..00 Data flow=on
..01 Data flow=off
..11 Data flow=same
.... 1... Receive any indicator
.... .xxx 0

16
VCNS flag byte

Bit
Meaning
1... STYPE=CONFIRM indicator
.xxx xxxx 0

17
0

18–1F
When CONTROL equals LOGON or CONTROL equals INQUIRE: symbolic name of the network access point

18–1B
When CONTROL does not equal LOGON and CONTROL does not equal INQUIRE and the request is an X.25 request: address into which expedited data is to be read or from which expedited data is to be written, or 0 if no area is supplied
When CONTROL does not equal LOGON and CONTROL does not equal INQUIRE: connection or resource identifiers (RID)

- RID is supplied when the VCNSCMD CONTROL value is:
  - LOGOFF
  - OPEN (STYPE=REQUEST)
  - READ
  - RECEIVE (when SMODE is CONNECTIONLESS)
  - REPLY
  - SEND (when the connection ID value is 0)
  - SET
  - SETCPARM (when the connection ID value is 0)
  - STATUS (when the RID value is not 0)
  - TEST
  - XID

- Connection identifier is supplied when the VCNSCMD CONTROL value is:
  - CLOSE
  - EXPEDITE
  - OPEN (STYPE=CONFIRM)
  - RECEIVE (Specific)
  - RESET
  - RESUME
  - SEND (when the connection ID value is not 0)
  - SETCPARM (when the connection ID value is not 0)
  - SUSPEND
  - STATUS (when the RID value is 0)

CNP1 or CNR1 entry for ECB posted or RPL exit dispatched (Part 1)

<table>
<thead>
<tr>
<th>Entry:</th>
<th>CNP1 or CNR1</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT option:</td>
<td>VCNS</td>
</tr>
<tr>
<td>Event:</td>
<td>ECB posted or RPL exit dispatched (Part 1)</td>
</tr>
<tr>
<td>VIT processing module:</td>
<td>ISTRACNS</td>
</tr>
<tr>
<td>Control is returned to:</td>
<td>ISTAICPT for CNP1. ISTAPCUE for CNR1</td>
</tr>
</tbody>
</table>

The CNP1 trace record is written when an ECB is posted. The CNR1 trace record is written when an RPL exit is dispatched. These trace records signal that execution of the VCNSCMD macroinstruction is complete and show that the data returned to the user application program.
Byte (hex)
Contents

00–03
Record ID:
  • C"CNP1" for ECB posting
  • C"CNR1" for RPL exit dispatching

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Global VTAM return code (RPLRTNCD)

06
Global VTAM feedback code (RPLFDB2)

07
VCNSCMD CONTROL= operand value

08
INQUIRE

10
LOGON

14
TEST

18
XID

20
LOGOFF

40
STATUS

50
SET

80
SEND

C0
CHECK

E0
SETCPARM

F0
RECEIVE

08–0B
RPL address

0C–0F
RPL3 address (RPLAAREA)
10–13
Address of area containing user data (RPLAREA), or 0 if no area is supplied

14
VCNS flag byte

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>Q-bit indicator</td>
</tr>
<tr>
<td>.1..</td>
<td>M-bit indicator</td>
</tr>
<tr>
<td>..1.</td>
<td>D-bit indicator</td>
</tr>
<tr>
<td>...1</td>
<td>Buffer list indicator</td>
</tr>
<tr>
<td>xxxx</td>
<td></td>
</tr>
</tbody>
</table>

15
VCNS flag byte

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00..</td>
<td>Continue specific</td>
</tr>
<tr>
<td>01..</td>
<td>Continue any</td>
</tr>
<tr>
<td>11..</td>
<td>Continue same</td>
</tr>
<tr>
<td>..00</td>
<td>Data flow=on</td>
</tr>
<tr>
<td>..01</td>
<td>Data flow=off</td>
</tr>
<tr>
<td>..11</td>
<td>Data flow=same</td>
</tr>
<tr>
<td>....</td>
<td>Receive any indicator</td>
</tr>
<tr>
<td>....</td>
<td></td>
</tr>
</tbody>
</table>

16
VCNS flag byte

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>STYPE=CONFIRM indicator</td>
</tr>
<tr>
<td>xxxx</td>
<td>xxxx</td>
</tr>
</tbody>
</table>

17
0

18–1B
Address into which expedited data is to be read or from which expedited data is to be written, or 0 if no area is supplied
1C–1F
Connection or resource identifier (RID)

- RID is returned when the VCNSCMD CONTROL value is:
  
  - LOGON
  - LOGOFF
  - READ
  - SET
  - STATUS (for network access point)
  - TEST
  - XID

- Connection ID is returned when the VCNSCMD CONTROL value is:
  
  - CLOSE
  - EXPEDITE
  - OPEN
  - RECEIVE
  - RESET
  - RESUME
  - SEND
  - SETCPARM
  - SUSPEND
  - STATUS (for connection)

- 0 is returned when the VCNSCMD CONTROL value is INQUIRE

CNP2 or CNR2 entry for ECB posted or RPL exit dispatched (Part 2)

Entry:
  
  CNP2 or CNR2

VIT option:
  
  VCNS

Event:
  
  ECB posted or RPL exit dispatched (Part 2)

VIT processing module:
  
  ISTRACNS

The CNP2 trace record is a continuation of the CNP1 trace record. The CNR2 trace record is a continuation of the CNR1 trace record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CNP2&quot; for continuation of CNP1</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CNR2&quot; for continuation of CNR1</td>
</tr>
</tbody>
</table>
CNP3 or CNR3 entry for ECB posted or RPL exit dispatched (Part 3)

Entry:
- CNP3 or CNR3

VIT option:
- VCNS

Event:
- ECB posted or RPL exit dispatched (Part 3)

VIT processing module:
- ISTRACNS

The CNP3 trace record is a continuation of the CNP2 trace record. The CNR3 trace record is a continuation of the CNR2 trace record. These records are written only when the VCNSCMD CONTROL operand value is RECEIVE or CLOSE, and RPLRLEN is greater than 0.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>04–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CNP3&quot; for continuation of CNP2</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CNR3&quot; for continuation of CNR2</td>
</tr>
<tr>
<td>04–1F</td>
<td>Twenty-eight bytes of user data (RPLAREA)</td>
</tr>
</tbody>
</table>
CNTP entry for buffer pool contraction

Entry:
  CNTP
VIT option:
  CSM
Event:
  Pool contraction
VIT processing module:
  ISTITCCS
Control is returned to:
  IVTSMCEX

This trace record is written when CSM pool contraction occurs.

<table>
<thead>
<tr>
<th>Byte</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C“CNTP”</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Pool address</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Extent address</td>
</tr>
<tr>
<td>10–13</td>
<td>Number of buffers in extent</td>
</tr>
<tr>
<td>14–17</td>
<td>Total number of buffers in the pool after contraction</td>
</tr>
<tr>
<td>18–1B</td>
<td>Number of free buffers in the pool after contraction</td>
</tr>
<tr>
<td>1C–1F</td>
<td>0</td>
</tr>
</tbody>
</table>
CONT entry for buffer pool contraction

Entry:
CONT

VIT option:
SMS

Event:
Buffer pool contraction

VIT processing module:
ISTRACSM

Control is returned to:
ISTORAPX

This trace record is generated whenever VTAM needs to contract a buffer pool. Buffer pool contraction will occur when VTAM determines that it has excess buffers that can be returned to the operating system.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CONT&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Address of buffer pool control block (BPCB) for which contraction occurred</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of pool expansion block (PXB) that is contracted</td>
</tr>
<tr>
<td>10–13</td>
<td>Extent address (PXBSTADR)</td>
</tr>
<tr>
<td>14–15</td>
<td>Number of queued RPHs waiting for expansion</td>
</tr>
<tr>
<td>16–17</td>
<td>0</td>
</tr>
<tr>
<td>18–1B</td>
<td>Total number of buffers in pool after this contraction (BPCBTOTL)</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Total number of available buffers in pool after this contraction</td>
</tr>
</tbody>
</table>

CPx entry for requests/responses processed by the CP (Part 1)

Entry:
CPI or CPO
VIT option:
SSCP

Event:
Requests/responses with a RUPE

VIT processing module:
ISTRACSC

Control is returned to:
Many modules possible

This trace record provides information about inbound and outbound requests and responses that are processed by the CP.

If this entry is associated with an event failure (that is, the sense data is nonzero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition, and therefore, is traced whenever the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CPI&quot; for inbound processing</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CPO&quot; for outbound processing</td>
</tr>
<tr>
<td>03</td>
<td>Flags</td>
</tr>
<tr>
<td></td>
<td>• X'80' = response RU</td>
</tr>
<tr>
<td></td>
<td>• X'40' = sense traced</td>
</tr>
<tr>
<td></td>
<td>• X'00' = request RU</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Control Block ID=X'54'</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Save Area Address</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Save Area ID</td>
</tr>
<tr>
<td>10–13</td>
<td>Address of the issuer of the APSEND macro</td>
</tr>
<tr>
<td>14–19</td>
<td>For sender, network address at origin</td>
</tr>
</tbody>
</table>
1A–1F
For sender, network address at destination

**CP2 entry for requests/responses processed by the CP (Part 2)**

**Entry:**
CP2

**VIT option:**
SSCP

**Event:**
Requests/responses with a RUPE (Part 2)

**VIT processing module:**
ISTRACSC

This trace record is a continuation of the CPI or CPO entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;CP2&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Control Block ID=X'54'</td>
</tr>
<tr>
<td>04–1B</td>
<td>First 24 bytes of RU</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Sense code or 0</td>
</tr>
</tbody>
</table>

**CPPG or CPPT entry for CPPURGE or CPPOST event (Part 1)**

**Entry:**
CPPG or CPPT

**VIT option:**
SSCP

**Event:**
CPPURGE or CPPOST Macro (Part 1)

**VIT processing module:**
ISTRACSC

**Control is returned to:**
ISTPVCWP

This trace record gives information about CPPURGE or CPPOST macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing.

CPPG and CPPT are the complements of CPWT. Each CPPT is preceded by a CPWT. But a CPPG can post several events at once, as it may be concluding several CPWT entries.
Byte (hex)

Contents

00–03
Record ID:
- C"CPPG" for CPPURGE processing
- C"CPPT" for CPPOST processing

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Return code set by CPPOST or CPPURGE

06
Type flags

Bit
Meaning

.... . .x
1 = PVI event

.... .x.
1 = search on EID
0 = search on URC

.... x...
For OPTIONS(POST) only, 1 indicates that return of a control block is optional. This bit corresponds to the CTLBLKOP keyword on the CPPOST macro.

.... x...
For OPTIONS(FIND) only, 1 indicates that the invoker intends to change the WRE's event ID. This bit corresponds to the CHGEID keyword on the CPPOST macro.

07
High-order 4 bits: Type of LQAB used for the search

Value
LQAB type

0000
Global LQAB

0001
EID-related LQAB

0010
DAF-related LQAB

0011
URC-related LQAB

Low-order 4 bits: Function requested through the OPTIONS keyword on the CPPOST macro, 0 for CPPG
Value

Keyword

0000
FIND

0001
DEQ

0010
POST

08–0B
LQAB group address

0C–0F
Data field in ISTWRE

10–13
Work element address

14–17
CPCB operation code that indicates the original request for this work element or 0 (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)

18–1B
For PVI events, caller's save area ID or 0. For non-PVI events, save area ID

1C–1F
For POST or PURGE, sense data or 0. For DEQ or FIND, 0

CPP2 entry for CPPURGE or CPPOST (Part 2)

Entry:
CPP2

VIT option:
SSCP

Event:
CPPURGE or CPPOST macro (Part 2)

VIT processing module:
ISTRACSC

This trace record contains part or all of the RU for CPPURGE or CPPOST macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing. If an RU is fewer than 24 bytes long, only the bytes in the actual RU will be moved to the corresponding fields in the trace records. The leftover bytes will be set to 0.

Note: This trace record is produced only for PVI events where a RUPE is available.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CPP2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Return Address</td>
</tr>
</tbody>
</table>
08–1F
First 24 bytes of RU or 0

CPP3 entry for CPPURGE or CPPOST (Part 3)

Entry:
CPP3

VIT option:
SSCP

Event:
CPPURGE or CPPOST macro (Part 3)

VIT processing module:
ISTRACSC

This trace record is a continuation of the CPPG or CPPT trace record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CPP3&quot;</td>
</tr>
<tr>
<td>04–1C</td>
<td>25 bytes of URC</td>
</tr>
<tr>
<td>1D–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

CPP4 entry for CPPURGE or CPPOST (Part 4)

Entry:
CPP4

VIT option:
SSCP

Event:
CPPURGE or CPPOST macro (Part 3)

VIT processing module:
ISTRACSC

This trace record contains part or all of the event ID (EID) for CPPURGE or CPPOST macroinstruction processing. If an EID is fewer than 24 bytes long, only the bytes in the actual EID will be moved to the corresponding fields in the trace records. The leftover bytes will be set to 0.

Note: This trace record is produced only when the WRE points to an EID.
Byte (hex)  
Contents  
00–03  
  Record ID: C"CPP4"
04–06  
  0
07  
  Length of EID
08–1F  
  First 24 bytes of EID

CPRC entry for CPRC (Part 1)  

Entry:  
CPRC  

VIT option:  
SSCP  

Event:  
CPRC macro (Part 1)

VIT processing module:  
ISTRACSC  

Control is returned to:  
Module invoking the INTRACE macro that caused the record to be produced  

This trace record gives information about CPRC macroinstruction processing. It is written when a VTAM module issues a CPRC macroinstruction to set a nonzero sense code in an RUPE.  

This event is treated as an exception condition and is always traced, whether the SSCP option is in effect or not.
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05–07
0

08–0B
Address of RUPE for this macro, or 0

0C–0F
CPCB operation code for RUPE that indicates the original request for this RUPE (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)

10–13
Sense code

14–19
Network address at origin (RUPE)

1A–1F
Network address at destination (RUPE)

CPR2 entry for CPRC (Part 2)

Entry:
CPR2

VIT option:
SSCP

Event:
CPRC macro (Part 2)

VIT processing module:
ISTRACSC

This trace record is a continuation of the CPRC entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CPR2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Address of the issuer of the CPRC macro</td>
</tr>
<tr>
<td>08–0C</td>
<td>Module name field from the register save area for the module that issued the CPRC macro</td>
</tr>
</tbody>
</table>

This field contains the abbreviated name (bytes 4, 5, 6, 7, 8) of the module that issued the CPRC macro. If the issuing module does not have its own save area, for example, modules ISTINCF1 and ISTPUCTI, this field contains the name field from the save area of the caller of the module that issued the CPRC macro.
0D
Instance of the CPRC macro in the issuer

0E
CPCBRC return code

Note: It is not required that users of the product know the meaning of this internal VTAM return code. When required, the product support organization may use it to assist in internal flow diagnosis.

0F–1F

CPR3 entry for CPRC (Part 3)

Entry:
CPR3

VIT option:
SSCP

Event:
CPRC macro (Part 3)

VIT processing module:
ISTRACSC

This trace record is a continuation of the CPRC entry.

<table>
<thead>
<tr>
<th>00–03</th>
<th>04–1C</th>
<th>1D–1F</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPR3</td>
<td>24 BYTES OF CPCBURC</td>
<td>0</td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

00–03
Record ID: C"CPR3"

04–1C
First 25 bytes of the user request correlator (CPCBURC) entry

CPWT entry for CPWAIT events (Part 1)

Entry:
CPWT

VIT option:
SSCP

Event:
CPWAIT macro (Part 1)

VIT processing module:
ISTRACSC

Control is returned to:
ISTCPCPW

This trace record gives information about CPWAIT macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing.
This entry is eventually followed by a CPPT entry (to resume processing) or a CPPG entry (to purge the waiting task). Use the EID field to match corresponding entries.

```
00–03
Record ID: C"CPWT"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Type flags
• B'00000001'=PVI event
• B'00000000'=non-PVI event

07
0

08–0B
LQAB address for this CPWAIT macro

0C–0F
Data field in ISTWRE when CPWAIT is performed

10–13
Work element address

14–17
CPCB operation code that indicates the original request for this work element or 0 (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)

18–1B
For PVI event, caller's save area ID or 0. For non-PVI event, save area ID

1C–1F
Sense code or 0

CPW2 entry for CPWAIT (Part 2)

Entry:
CPW2

VIT option:
SSCP

Event:
CPWAIT macro (Part 2)

VIT processing module:
ISTRACSC
This trace record is a continuation of the CPWT trace record. If an RU is fewer than 24 bytes long, only the bytes in the actual RU will be moved to the corresponding fields in the trace records. The leftover bytes will be set to 0.

**Note:** This trace record is produced only for PVI events where a RUPE is available.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CPW2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Return address</td>
</tr>
<tr>
<td>08–1F</td>
<td>First 24 bytes of RU</td>
</tr>
</tbody>
</table>

**CPW3 entry for CPWAIT (Part 3)**

Entry: CPW3

VIT option: SSCP

Event: CPWAIT macro (Part 3)

VIT processing module: ISTRACSC

This trace record is a continuation of the CPWAIT trace record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;CPW3&quot;</td>
</tr>
<tr>
<td>04–1C</td>
<td>RUPEURC or 0</td>
</tr>
<tr>
<td>1D–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

**CPW4 entry for CPWAIT (Part 4)**

Entry: CPW4
**VIT option:**
SSCP

**Event:**
CPWAIT macro (Part 4)

**VIT processing module:**
ISTRACSC

This trace record is a continuation of the CPW3 trace record. If an event ID (EID) is fewer than 24 bytes long, only the bytes in the actual EID will be moved to the corresponding fields in the trace records. The leftover bytes will be set to 0.

**Note:** This trace record is produced only when the WRE points to an EID event.

```
| 0 | 1 | 2 | 3 | 0 | 0 | 0 | 0 |
---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 2 | 3 |
---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 4 | 5 | 6 |
```

**Byte (hex)**

**Contents**

| 00–03 | Record ID: C”CPW4“ |
| 04–06 | 0 |
| 07 | Length of EID |
| 08–1F | First 24 bytes of EID |

**CPYB entry for COPY_DATA requests**

**Entry:**
CPYB

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=COPY_DATA

**VIT processing module:**
ISTITCCS

**Control is returned to:**
IVTSMCCD or IVTSM6CD

**Note:**
- IVTSMCCD processes COPY_DATA requests in 31-bit addressing mode.
- IVTSM6CD processes COPY_DATA requests in 64-bit addressing mode.

This trace record provides the status of an IVTCSM REQUEST=COPY_DATA macroinstruction.
### Byte (hex)

**Contents**

| 00–03 | Record ID: C"CPYB"
| 04    | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05    | 0
| 06    | Trace record flag:
|       | B'0... ....' Indicates that this is the last record for this event.
|       | B'1... ....' Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.
|       | B'..1. ....' Indicates that the buffer list is in 64-bit storage.
| 07    | Trace record number to correlate all the entries for this particular event
| 08–0B | 0
| 0C–0D | Return code
| 0E–0F | Reason code
| 10–13 | Address of utility routine caller or return address of the issuer of the IVTCSM macroinstruction
| 14–17 | Number of buffers containing the data to be copied
| 18–1B | Number of buffers to receive the copied data
| 1C–1F | THREAD value if specified or 0 if THREAD is not specified. The THREAD value is used only to correlate this trace record to a specific IVTCSM macroinstruction.

### CPY2 entry for COPY_DATA requests

**Entry:**

- **CPY2**

**VIT option:**

- **CSM**
Event:
IVTCSM REQUEST=COPY_DATA

VIT processing module:
ISTITCCS

This trace record is a continuation of the CPYB trace record. It is an exception record and is generated only when the return code of the macroinstruction is not 0.

### Byte (hex)

**Contents**

**00–03**
Record ID: C"CPY2"

**04–05**
0

**06**
Trace record flag:

- **B'0... ...'**
  Indicates that this is the last record for this event.

- **B'1... ...'**
  Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

**07**
Trace record number to correlate all the entries for this particular event

**08–0B**
Last source buffer list entry successfully processed for error return codes

**0C–0F**
Last target buffer list entry successfully processed for error return codes

**10–1F**
0

### CPY3 entry for COPY_DATA requests

**Entry:**
CPY3

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=COPY_DATA

**VIT processing module:**
ISTITCCS

This trace record is a continuation of the CPYB trace record. It contains information about an entry in the source buffer list. There is one CPY3 trace record for each entry in the source buffer list.
Byte (hex)

Contents

00–03
Record ID: C"CPY3"

04–05
0

06
Trace record flag:

B'0... ....'
Indicates that this is the last record for this event.

B'1... ....'
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

B'.1. ....'
Indicates that the buffer list is in 64-bit storage.

07
Trace record number to correlate all the entries for this particular event

08–13
Buffer token

14–17
ALET or first word of the 64-bit address of data being copied

18–1B
31-bit address of data being copied or second word of the 64-bit address of data being copied

1C–1F
Size of data being copied

**CPY4 entry for COPY_DATA requests**

Entry:
CPY4

VIT option:
CSM

Event:
IVTCSM REQUEST=COPY_DATA

VIT processing module:
ISTITCCS

This trace record is a continuation of the CPYB trace record. It contains information about an entry in the target buffer list. There is one CPY4 trace record for each entry in the target buffer list.
Byte (hex)

Contents

00–03
Record ID: C"CPY4"

04–05
0

06
Trace record flag:

B'0... ....'
Indicates that this is the last record for this event.

B'1... ....'
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

B'..1. ....'
Indicates that the buffer list is in 64-bit storage.

07
Trace record number to correlate all the entries for this particular event

08–13
Buffer token

14–17
ALET or first word of the 64-bit address of the storage where data is to be copied

18–1B
31-bit address of the storage where data is to be copied or second word of the 64-bit address of the storage where data is to be copied

1C–1F
Length of the storage where data is to be copied

CPY6 entry for COPY_DATA requests

Entry:
CPY6

VIT option:
CSM

Event:
IVTCSM REQUEST=COPY_DATA

VIT processing module:
ISTITCCS

This trace record is a continuation of the CPYB trace record. It contains the 64-bit buffer list entry address. This entry is present only when COPY_DATA request failed with the error return code in 64-bit addressing mode.
Byte (hex)

Contents

00–03
Record ID: C"CPY6"

04–05
0

06
Trace record flag:

B'0... ....'
Indicates that this is the last record for this event.

B'1... ....'
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

B'.1. ....'
Indicates that the buffer list is in 64-bit storage.

07
Trace record number to correlate all the entries for this particular event

08–0F
0

10–17
64-bit address of the last source buffer list entry

18–1F
64-bit address of the last target buffer list entry

CRx entry for ADD, DELETE, or FIND control block

Entry:
CRA, CRD, or CRF

VIT option:
SSCP

Event:
CRADD, CRDEL, CRFIND

VIT processing module:
ISTRACSC

The correlate search control block (CR) trace record contains information about a control block add, delete, or find operation in the VTAM control point (CP).
Byte (hex)

**00–02**

Record ID:

*C"CRA" (CRADD):*
- Add control block

*C"CRD" (CRDEL):*
- Delete control block

*C"CRF" (CRFIND):*
- Find control block

**03**

Type code

0
- Procedure-correlation identifier (PCID) only (6–F contain zeros)

1
- PCID and modifier list (6–F contain the modifier list)

2
- PCID and search number (6–7 contain the search number, 8–F contain zeros)

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

Return code if specified; some FIND calls to this macro do not include a return code but use CORCB ADDR as a return code.

**06–0F**

- If type code = 0, this field is 0.
- If type code = 1, this field represents the PCID modifier list and is padded on the right with zeros. If search number correlation is used, this field may be 0.
- If type code = 2, this field represents the search number and is padded on the right with zeros.

**10–13**

Calling module's return address

**14–1B**

PCID of the search being correlated

**1C–1F**

Address of an ISTCORCB; bytes 4–7 of the ISTCORCB contain the address of the control block being correlated (LCB, SITCB). This field is 0 if the entry type is "CRF" and the control block was not found.

**Note:**

1. A CRF entry is found soon after a DSP entry for LUSS or DSVC. If the CORCB ADDR is 0, the FIND failed.
2. Search number correlation is used with DS only.
CSx entry for resource state change

Entry:
CSC, CSD, or CSB

VIT option:
SSCP

Event:
Change in resource state

VIT processing module:
ISTRACSC

Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is written when the current state or required state, or both, of a resource for which tracing has been requested changes. If the network ID of the resource being traced differs from the network ID of the host, the trace record CSC2, CSD2, or CSB2 will follow this entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CSC&quot; for current state change</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CSD&quot; for required state change</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CSB&quot; for both current and required state change</td>
</tr>
<tr>
<td>03</td>
<td>Type of resource</td>
</tr>
<tr>
<td>04–05</td>
<td>Resource's current state</td>
</tr>
<tr>
<td>06–07</td>
<td>Resource's required state</td>
</tr>
<tr>
<td>08–0F</td>
<td>Resource name</td>
</tr>
<tr>
<td></td>
<td>Note: If the resource RDTE is not available, RDTEPTR is 0, bytes 04–0F will be 0.</td>
</tr>
<tr>
<td>10–13</td>
<td>Return address of the module that changed the resource's state</td>
</tr>
<tr>
<td>14–17</td>
<td>Address RDTE</td>
</tr>
<tr>
<td>18–1B</td>
<td>Name of the module that changed the resource's state</td>
</tr>
</tbody>
</table>
1C–1F
Request parameter header (RPH) address

CSC2, CSD2, or CSB2 entry for resource state change

Entry:
CSC2, CSD2, or CSB2

VIT option:
SSCP

Event:
Change in resource state of resource with network ID different from host’s network ID

VIT processing module:
ISTRACSC

This trace record is a continuation of the CSx entry and is written when the network ID of the resource being traced differs from the network ID of the host.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CSC2&quot; for current state change</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CSD2&quot; for required state change</td>
</tr>
<tr>
<td></td>
<td>• C&quot;CSB2&quot; for both current and required state change</td>
</tr>
<tr>
<td>04–0B</td>
<td>Network ID of the resource being traced</td>
</tr>
<tr>
<td>0C–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

DAPT entry for HPR out of sequence or received segments queue DAPTR

Entry:
DAPT

VIT option:
HPR

Event:
DAPTR alteration

VIT processing module:
ISTITCHP

This trace record shows most of the DAPTR fields. The DAPTR is used to chain inbound HPR data that arrives out of order or in order but is segmented into pieces. The DAPTR resides either on the RPN_OutOfSeq_Msg_Q or RPN_RCV_Segments_DaPtr queue of the RPNCB that is currently dispatched. The record shows the updated contents of the DAPTR.
### Byte (hex)

#### Contents

<table>
<thead>
<tr>
<th>Byte</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;DAPT&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>DAPTR type</td>
</tr>
<tr>
<td>0</td>
<td>Out of sequence queue</td>
</tr>
<tr>
<td>S</td>
<td>Received segments queue</td>
</tr>
<tr>
<td>06</td>
<td>NLP count (X'FF' if 255 or greater)</td>
</tr>
<tr>
<td>07</td>
<td>DAP_Flags:</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>1</td>
<td>DAP_End_Msg</td>
</tr>
<tr>
<td>.1</td>
<td>DAP_Beg_Msg</td>
</tr>
<tr>
<td>..1</td>
<td>DAP_Last_Msg</td>
</tr>
<tr>
<td>...1</td>
<td>DAP_COB_Ind</td>
</tr>
<tr>
<td>.....</td>
<td>xxxx</td>
</tr>
<tr>
<td>08–0B</td>
<td>DAP_First_Head_TSCB</td>
</tr>
<tr>
<td>0C–0F</td>
<td>DAP_Last_Head_TSCB</td>
</tr>
<tr>
<td>10–13</td>
<td>DAP_Begin_Seq</td>
</tr>
<tr>
<td>14–17</td>
<td>DAP_End_Seq</td>
</tr>
<tr>
<td>18–1B</td>
<td>RPNCB address</td>
</tr>
<tr>
<td>1C–1F</td>
<td>DAPTR address</td>
</tr>
</tbody>
</table>

352 z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
DBx entry for DBDELETE, DBQUERY, and DBUPDATE

Entry:
  DBD, DBQ, DBU

VIT option:
  SSCP

Event:
  DBDELETE, DBQUERY, DBUPDATE

VIT processing module:
  ISTRACSC

This traces an operation on the APPN directory database.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;DBD&quot;, C&quot;DBQ&quot;, C&quot;DBU&quot;</td>
</tr>
<tr>
<td>03</td>
<td>0</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Return code (RC)</td>
</tr>
<tr>
<td>06–07</td>
<td>Database information flags</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>1...</td>
<td>Registered entry-type indicator</td>
</tr>
<tr>
<td>.1...</td>
<td>Cache or dynamic entry-type indicator</td>
</tr>
<tr>
<td>..1...</td>
<td>Suggestion or defined entry-type indicator</td>
</tr>
<tr>
<td>...1...</td>
<td>Reserved for entry-type expansion</td>
</tr>
<tr>
<td>....1...</td>
<td>The VIT error was caused by an error in the cache data processing</td>
</tr>
<tr>
<td>....1...</td>
<td>The VIT error was caused by an error while loading the directory from storage</td>
</tr>
<tr>
<td>....1...</td>
<td>The returned network ID on DBQUERY differs from the one queried</td>
</tr>
<tr>
<td>.......x</td>
<td>Not used</td>
</tr>
</tbody>
</table>
Directed failed indicator copied from directory entry

Negative cache indicator copied from directory entry

Subarea LU indicator copied from directory entry

Surrogate owner indicator copied from directory entry

Dynamic subarea destination LU indicator copied from directory entry

Wildcard LU indicator copied from directory entry

Nonnative LU indicator copied from directory entry

Generic name indicator copied from directory entry

08–0F
Resource identification

10–13
Caller’s return address

14–17
Address of directory entry

18–1F
Network identification of resource

**DCON entry for discarded container**

**Entry:**
DCON

**VIT option:**
PIU

**Event:**
Discard Container

**VIT processing module:**
ISTRACOT

**Control is returned to:**
ISTTSCUA

This trace record is written when the VTAM Enterprise Extender utility discard routine, ISTTSCUA, disposes of a container. The reason code can be used to explain the reason for the discard.
Byte (hex)
  Contents
00–03
  Record ID: C'DCON'
04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05–07
  0
08–0B
  Container address
0C–0F
  Reason code

Code
  Meaning
X'0000'
  VTAM is not able to contact UDP.
X'0002'
  Data is sent successfully, but fragmented.
X'0004'
  Temporary error. Retry.
X'0008'
  Error. Local IP address is not valid.
X'000C'
  Error. Local IP address is not a valid VIPA address.
X'0010'
  Error. Port cannot be reserved.
X'0014'
  Error. Parameter is not valid.
X'0018'
  Error. State is not valid.
X'001C'
  Error. Destination for datagram is unreachable.
X'0020'
  Error. VTAM is not authorized.
X'0024'
  Error. Storage unavailable.
X'0028'
  Error. Data exceeds maximum.
X'002C'
  Error. Stack is not valid.
X'0040'
  Permanent error.
10–13
  Target IPv4 address or zeros. If zeros, target IPv6 address is reported in DCO2 trace record.
14–17
  Parameter list address
18–19
  Source port number
DCO2 entry for discarded container (Part 2)

Entry:
  DCO2
VIT option:
  PIU
Event:
  Discard container
VIT processing module:
  ISTRACOT

This trace record is a continuation of the DCON entry. This entry is present only when the discarded container contains an IPv6 address.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'DCO2'</td>
</tr>
<tr>
<td>04–0F</td>
<td>0</td>
</tr>
<tr>
<td>10–1F</td>
<td>IPv6 Address</td>
</tr>
</tbody>
</table>

DEVx entry for MPNCB device counter update

Entry:
  DEVD or DEVI
VIT option:
  CIA
Event:
  MPNCB Active device counter update
VIT processing module:
  ISTRACCI
Control is returned to:
  Modules invoking the INTRACE macro that caused the record to be produced.

This trace record is written when a module is about to update an active device counter in MPNCB. It records the current Read and Write device counters before the change takes place.

• DEVD is generated for a counter decrement.
• DEVI is generated for a counter increment.

See z/OS Communications Server: SNA Data Areas Volume 1 for a description of the NCB fields.
For record types with suffix I,X, or T the CIO events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>C&quot;DEVD&quot; for Device counter decrementation</td>
</tr>
<tr>
<td></td>
<td>C&quot;DEVI&quot; for Device counter incrementation</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>The module name that invokes the INTRACE macro.</td>
</tr>
<tr>
<td>0C–0F</td>
<td>The CPNCB address of the device that is about to be excluded (DEVD) or included (DEVI) from the MPNCB current active device counter.</td>
</tr>
<tr>
<td>10–13</td>
<td>Active Read device counter (MPNARTPH)</td>
</tr>
<tr>
<td>14–17</td>
<td>0</td>
</tr>
<tr>
<td>18–1B</td>
<td>Active Write device counter (MPNAWTPH)</td>
</tr>
<tr>
<td>1C–1F</td>
<td>RPH address</td>
</tr>
</tbody>
</table>

**DLT entry for directory services locate (Part 1)**

**Entry:**
- DLT

**VIT option:**
- SSCP

**Event:**
- Sending out a Locate Search

**VIT processing module:**
- ISTRACSC

**Control is returned to:**
- ISTDRSDL

This trace record is written when VTAM’s directory services component forwards a locate search to some adjacent node.

**Note:** This trace record has been reformatted because of an increase in the size of the task vector field.
Byte (hex)
Contents
00–02
Record ID: C"DLT"
03
0
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05–07
0
08–0F
The task vector bits indicate the possible tasks for this locate search. The hex values listed below are in the task vector field of this entry and the task vector results field of the DLT2 entry.
08
Vector
Description
X'80'
Null task
X'40'
Directory services management exit
X'20'
Directory services database query
X'10'
Topology and routing services database query
X'08'
Forward to network node server
X'04'
One hop if directory services database is found
X'02'
One hop if control vector X'0E' is received for request
X'01'
Nonverify attempt
09
Vector
Description
X'80'
Directed because of network node destination LU hierarchy received on a search request
*X'40'*
Directed if directory services database is found

*X'20'*
Directed if the topology and routing services valid route selection CV is returned

*X'10'*
Resource Discovery Search

*X'08'*
Directed to a directory server

*X'04'*
Directed to a higher function directory server

*X'02'*
Directed to a directory server retry

*X'01'*
Directed to a gateway node

0A

Vector
Description

*X'80'*
Sequential directed search to alternate directory servers

*X'40'*
Sequential directed search to interchange nodes

*X'20'*
Subarea system resolution table (SRT) cache search

*X'10'*
Subarea search after a positive cache search

*X'08'*
Subarea search after a positive directory services database query

*X'04'*
Subarea search after a negative or no cache search

*X'02'*
Domain broadcast search

*X'01'*
Originate network broadcast search

0B

Vector
Description

*X'80'*
Forward network broadcast not originated by this node

*X'40'*
One-hop search request because of end node destination LU hierarchy received on a search request

*X'20'*
A cross-subnetwork directed search because of information received on the original request

*X'10'*
A cross-network directed search because of information found in the directory services database

*X'08'*
A directed search because of information found in the topology and routing services database

*X'04'*
Sequential directed search with the intent of finding the resource cross-subnetwork
<table>
<thead>
<tr>
<th>Vector</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'02'</td>
<td>Generic cache search</td>
</tr>
<tr>
<td>X'01'</td>
<td>A directed search because of a SEARCH_RPY interprocess signal following a positive CACHE_SEARCH_RPY interprocess signal</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vector</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0C</td>
<td></td>
</tr>
<tr>
<td>Vector</td>
<td>Description</td>
</tr>
<tr>
<td>X'80'</td>
<td>A directed search because of a SEARCH_RPY interprocess signal following a positive directory services database query</td>
</tr>
<tr>
<td>X'40'</td>
<td>A subarea search because of a SESS_INIT_INFO_RPY interprocess signal</td>
</tr>
<tr>
<td>X'20'</td>
<td>Database query after an RDS</td>
</tr>
<tr>
<td>X'10'</td>
<td>Sequential directed search to other network nodes in the generic resource configuration</td>
</tr>
<tr>
<td>X'08'</td>
<td>Final subarea search after resource not found in APPN with SSEARCH = APPNFRST</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vector</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0D–0E</td>
<td></td>
</tr>
<tr>
<td>0F</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vector</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'01'</td>
<td>Post processing</td>
</tr>
</tbody>
</table>

10–13  
Caller's return address

14–15  
The node role for this search. More than 1 bit can be on.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'80'</td>
<td>CP originating LU</td>
</tr>
<tr>
<td>X'40'</td>
<td>CP destination LU</td>
</tr>
<tr>
<td>X'20'</td>
<td>NN originating LU</td>
</tr>
<tr>
<td>X'10'</td>
<td>NN destination LU</td>
</tr>
<tr>
<td>X'08'</td>
<td>Owning directory server</td>
</tr>
<tr>
<td>X'04'</td>
<td>Alternate directory server</td>
</tr>
<tr>
<td>X'02'</td>
<td>Intermediate network server</td>
</tr>
<tr>
<td>X'01'</td>
<td>Intermediate network node directed</td>
</tr>
</tbody>
</table>
Position  Description
X'80'  Intermediate network node broadcast

16  Return code from the directory services management exit

17  Terminating condition indicates why the search ended

Code  Description
X'00'  Processing can continue.
X'04'  Positive reply can be returned to parent.
X'08'  Gateway reply has been received.
X'0C'  Directory server reply has been received.
X'10'  An error was detected by a task called from the sequencer.
X'14'  Cleanup is pending after all replies are received from the search phase (CP session outage).
X'18'  A directed search was performed because of wildcard information, and a wildcard was returned. No further searching will be done.
X'20'  An alternate directory server had an unknown (neg cache) entry.
X'24'  Directory services management exit routine specified no search for this request.
X'28'  An error was detected during generic cache search task.
X'32'  This search kicked off an RDS which returned a negative reply. No more searching should be done for this search.
X'34'  IOPURGE occurred during verification after RDS found resource.
X'36'  Directed search failed after RDS found resource.

18–1B  Address of the locate control block for this search.

1C–1F  Sense code.

**DLT2 entry for directory services locate (Part 2)**

Entry:  
DLT2

VIT option:  
SSCP

Event:  
Sending out a Locate Search
VIT processing module:
ISTRACSC

The DLT2 trace record is a continuation of the DLT trace record.

<table>
<thead>
<tr>
<th>Byte</th>
<th>Hex</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>0 1 2 3 4 5 6 7 8 9 A B C D E F</td>
<td>Task Vector Results</td>
</tr>
<tr>
<td>04–07</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>0 1 1 1 1 1 1 1 1 1</td>
<td>Task Vector Results</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Byte (hex)
Contents

00–03
Record ID: C"DLT2"

04–07
0

08–0F
Task Vector Results
Use the flags in the task vector results field to determine whether the task indicated in bytes 08—0F in the DLT entry were invoked.

B'0'
Task was not invoked.

B'1'
Task was invoked.

10–1F
0

DRBx entry for invoking a RoCE doorbell operation

Entry:
DRBx

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) doorbell operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTRITCSH

This trace record is written upon completion of a RoCE doorbell operation.
### Byte (hex)

**Contents**

**00–03**
- Record ID:
  - C'DRBC' for CQ doorbell
  - C'DRBE' for EQ doorbell
  - C'DRBR' for Receive doorbell
  - C'DRBS' for Send doorbell

**04**
- ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
- 0

**06–07**
- Array index

**08–0F**
- Doorbell-related data

**10–17**
- Address of the PFCTE

**18–1F**
- Address of the control block that is associated with this doorbell activity

### DRP<sub>x</sub> entry for DLURRTP macroinstruction invocation (Part 1)

**Entry:**
- DRPA, DRPC, or DRPD

**VIT option:**
- HPR

**Event:**
- Invocation of DLURRTP macroinstruction

**VIT processing module:**
- ISTITCHR

**Control is returned to:**
- Module invoking the DLURRTP macroinstruction

This trace record is written when the DLURRTP macroinstruction is issued. It indicates the function being performed, the return code for that function, and the addresses of the parameters involved.
Byte (hex)

Contents

00–03

Record ID:

- C"DRPA" for DLURRTP ADD
- C"DRPC" for DLURRTP CHG
- C"DRPD" for DLURRTP DEL

04

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05

0

06–07

Use count is the number of sessions associated with this DLUR RTP entry.

08

Return code:

00
Success

04
Not found

08
Insufficient storage

16
Operation ended because of major error

09

Flags (as defined in DLURRTP control block)

Bit
Meaning

1... .
NetView notified of path switch for this RTP connection.

.1... .
RTP connection is in use by at least one session (has meaning during NLDLM start processing only).

..1... .
UNCOND specified (applicable to DLURRTP DEL only).

...1...
TCID or CQF information input to macroinstruction is not in a valid format.

.... 1...
RTP RSCV could not be saved.
Available.

0A–0B
0

0C–0F
Address of DLUR RTP control block

10–13
Address of issuer of the DLURRTP macroinstruction

14–1B
Data field of the transport connection identifier (TCID) control vector (X'4B')

1C–1F
Request parameter header (RPH) address

**DRP2 entry for DLURRTP macroinstruction invocation (Part 2)**

Entry:
DRP2

VIT option:
HPR

Event:
Invocation of DLURRTP macroinstruction

VIT processing module:
ISTITCHR

This trace record is a continuation of the DRP entry. It contains the four data fields from the network address control vector, left-aligned in fields of the maximum length allowed.

<table>
<thead>
<tr>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 1</td>
<td>0 0 0 2</td>
<td>0 0 0 3</td>
<td>0 0 0 4</td>
<td>0 0 0 5</td>
</tr>
<tr>
<td>0 0 0 6</td>
<td>0 0 0 7</td>
<td>0 0 0 8</td>
<td>0 0 0 9</td>
<td>0 0 A 0</td>
</tr>
<tr>
<td>0 C 0 D</td>
<td>0 E 0 F</td>
<td>1 1 1 1</td>
<td>1 1 1 1</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>DRP2</td>
<td>NETWORK ID</td>
<td>CP NAME</td>
<td>NCE ID</td>
<td>NCE INSTANCE</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03
Record ID: C"DRP2"

04–0B
Network ID of the DLUR

0C–13
CP name of the DLUR

14–1B
NCE identifier

1C–1F
NCE instance identifier
DRP3 entry for DLURRTP macroinstruction invocation (Part 3)

Entry:          
DRP3
VIT option:     
HPR
Event:          
Invocation of DLURRTP macroinstruction
VIT processing module:  
ISTITCHR

This trace record is a continuation of the DRPA and DRPC entries. It is not written for the DRPD entry.

| 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 |
| 0 1 2 3 | 4 5 6 7 | 8 9 A B | C D E F | 0 1 1 1 | 1 1 1 1 | 1 1 1 1 | 1 1 1 1 | 1 1 1 1 | 1 1 1 1 |
| DRP3 | 28 BYTES OF RSCV |

Byte (hex)  
Contents
00–03  
Record ID: C"DRP3"
04–1F  
28 bytes of RSCV representing the RTP connection

DSCx entry for discarded TSCBs or TIPAC (Part 1)

Entry:          
DSCD or DSCO
VIT option:     
PIU
Event:          
Discarded TSCB or TIPAC (Part 1)
VIT processing module:  
ISTRACOT
Control is returned to:  
ISTTSCUD

This trace record is written when VTAM's TSC utility discard routine, ISTTSCUD, disposes of transmission subsystem control blocks (TSCBs) or transport interface parameter access containers (TIPACs) because of an error condition. The reason code and module identifier can be used to explain the reason for the discard. This entry is treated as an exception condition and is always traced if the VIT is active, regardless of the VIT options specified. A DSC2 entry follows a DSCO or DSCD entry. The DSC2 entry includes the ID of the module that is discarding the storage.

| 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 |
| 0 1 2 3 | 4 5 6 7 | 8 9 A B | C D E F | 0 1 1 1 | 1 1 1 1 | 1 1 1 1 | 1 1 1 1 | 1 1 1 1 | 1 1 1 1 |
| DSCD | DSCO | I D | REASN CODE | TSCB ADDRESS | 20 BYTES OF TSCBs OR TIPACs |
**Byte (hex)**

**Contents**

**00–03**
Record ID:
- C"DSCD" for discarding from procedural modules
- C"DSCO" for discarding from object-oriented methods

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05–07**
Discard reason code

**08–0B**
Address of TSCB

**0C–1F**
First 20 characters of the TSCB or TIPAC.

### DSC2 entry for discarded TSCB or TIPAC (Part 2)

**Entry:**
- DSC2

**VIT option:**
- PIU

**Event:**
- Discarded TSCB or TIPAC (Part 2)

**VIT processing module:**
- ISTRACOT

This trace record is a continuation of the DSCx entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;DSC2&quot;</td>
</tr>
<tr>
<td>04–1B</td>
<td>24 more characters of the TSCB or TIPAC</td>
</tr>
<tr>
<td>1C–1F</td>
<td>ID of the module that is discarding the TSCB or TIPAC</td>
</tr>
</tbody>
</table>

### DSP entry for PAB dispatch

**Entry:**
- DSP

**VIT option:**
- PSS

**Event:**
- PAB dispatch
VIT processing module:
ISTRACPS

Control is returned to:
ISTAPCPD

The DSP entry marks the beginning of a PAB dispatch, which is the VTAM major unit of work. This unit exists until terminated by an EXIT entry. While processing, it might wait for other events to complete (WAIT). It resumes processing with a RESM entry. The unit might be interrupted by a higher-priority task and continue when that task is finished.

Each PAB is identified by a unique index number. VTAM uses this number to find the called module and then creates a VIT entry for it.

To correlate all events associated with this unit, look for the request parameter header (RPH) address, which is in many trace records, and match it to the request parameter header (RPH) address in the DSP trace record. This unit has exclusive use of that RPH until EXIT.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;DSP&quot;</td>
</tr>
<tr>
<td>03</td>
<td>0, or, for very extended PABs, PAB work element queue level dispatched</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Control block ID of work element (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)</td>
</tr>
<tr>
<td>06</td>
<td>PAB flag field (PABFLAGS)</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>1...</td>
<td>PAB is unconditionally scheduled.</td>
</tr>
<tr>
<td>..1...</td>
<td>PAB closedown is in progress.</td>
</tr>
<tr>
<td>...1...</td>
<td>PAB is synchronous.</td>
</tr>
<tr>
<td>...1...</td>
<td>PAB extension is present.</td>
</tr>
<tr>
<td>.... 1...</td>
<td>Do not dequeue work element.</td>
</tr>
<tr>
<td>.... .1...</td>
<td>Do not detach the request parameter header (RPH).</td>
</tr>
<tr>
<td>.... ...1.</td>
<td>Indicates a very extended PAB.</td>
</tr>
</tbody>
</table>
Indicates a slightly extended PAB.

**07**
PAB flag field (PABFLGS1)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>Switch the PST address of the major control block for this PAB to the new PST address contained in DYPNWPST.</td>
</tr>
<tr>
<td>.1..</td>
<td>This PAB has a data space extension.</td>
</tr>
<tr>
<td>..1.</td>
<td>This PAB's major control block is an FMCB.</td>
</tr>
<tr>
<td>....1.</td>
<td>PAB can be referenced in PSW disable mode.</td>
</tr>
<tr>
<td>....1.</td>
<td>PAB is persistent.</td>
</tr>
<tr>
<td>....1.</td>
<td>APSTERM/APSINIT FMCB during PAB dispatch.</td>
</tr>
<tr>
<td>...xx</td>
<td>Reserved.</td>
</tr>
</tbody>
</table>

**08–0B**
PST address

**0C–0F**
PAB address

**10–13**
Address of work element most recently queued to the PAB

**14–17**
Address of work element currently being dispatched

**18–1B**
Module name abbreviation (bytes 4, 5, 7, and 8 of the module name) or PAB DVT address (high-order bit of X'18' = 0). For an explanation of the module-naming convention, see “Using module names to isolate VTAM problems” on page 23. (The module name might be unavailable if the PAB being scheduled is associated with an address space different from the current one.)

**1C–1F**
Request parameter header (RPH) address

**DTSK entry for detach a subtask**

**Entry:**
DTSK

**VIT option:**
PSS

**Event:**
Detach a subtask.

**VIT processing module:**
ISTRACPS

**Control is returned to:**
The module that issued the DETACH
This trace record is written when a VTAM module detaches a VTAM subtask. Some subtasks are detached without generating this entry.

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00–03**
- Record ID: C"DTSK"

**04**
- ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
- Return code from the detach macro (ITKRC)

**06–07**
- 0

**08–0F**
- Name of the subtask being detached, or 0

**10–13**
- TCB address of the subtask being detached

**14–17**
- 0

**18–1B**
- Address of the issuer of the DETACH macro

**1C–1F**
- 0

**ENFx entry for ENF exit**

**Entry:**
- ENFF, ENFN, ENFP, or ENFR

**VIT option:**
- CIA

**Event:**
- ENF exit for APPN host-to-host channel dynamics

**VIT processing module:**
- ISTRACCI

**Control is returned to:**
- ISTTSCDY

This trace record is written when MVS schedules the VTAM ENF exit for APPN host-to-host channel dynamics for a subchannel device.
### Byte (hex)

#### Contents

**00–03**
- **Record ID:**
  - C"ENFF" for offline
  - C"ENFN" for online
  - C"ENFP" for pending offline
  - C"ENFR" for reaccessible

**04**
- ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
- Reserved

**06–07**
- 0

**08–0B**
- CUA from UCBCHAN or DACH_IORA_DEVN

**0C–0F**
- Address of SCL

**10**
- SCL ENF flags

**11**
- 0

**12–13**
- SCL states:
  - **12**
    - SCL channel state
  - **13**
    - SCL_SYS_state

**14–1B**
- 0

**1C–1F**
- Request parameter header (RPH) address

---

### ENR entry for APPN resource registration processing

**Entry:**

**ENR**
VIT option:
SSCP

Event:
An APPN resource registration function is being processed.

VIT processing module:
ISTRACSC

This trace record is written by end nodes when Configuration Services processes an APPN registration. The purpose of the ENR trace record is to record the update and delete FSM states before and after registration. This entry also provides information about which process invoked the registration function and what processing was done.

Note: This trace record is written at the conclusion of the registration process.

| 00–02 | Record ID: C"ENR"
| 03 | 0
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05–06 | 0
| 07 | RC is the return code on exit from ISTCSCRE.
| 08 | EUFSM is the UPDATE FSM state upon entry to ISTCSCRE. The FSM states are defined within ISTCSCRE.
| 09 | CUSFM is the changed UPDATE FSM's new value. The changed value may be the same as the entry value.
| 0A | EDFSM is the DELETE FSM state upon entry to ISTCSCRE. The FSM states are defined within ISTCSCRF.
| 0B | CDFSM is the changed DELETE FSM's value after processing by ISTCSCRF. The changed value may be the same as the entry value.
| 0C | EVENT is the registration event code that is being processed. These codes are defined in ISTREVNT.
| 0D–0E | 0
| 0F | Flags
Bit
11.. ....
    Server Availability Status
      00
      Not available
      01
      Available
      10
      Pending
..11 ....
    Request type
      01
      Network node server request
      10
      Central Resource Registration request
      11
      Delete request
      .... 1...
      Reply pending because server is not available
      .... .1..
      Processing USERVAR
10–13
    Module issuing the INTRACE macro
14–17
    Indicates which processing routine was invoked
18–1B
    Indicates which module invoked the registration process
1C–1F
    Request parameter header (RPH) address

** ERPx entry for error recovery procedures **

Entry:
ERPI, ERPL, ERPT, or ERPX

VIT option:
CIO

Event:
Error recovery

VIT processing module:
ISTRACCI

Control is returned to:
ISTZBM0J for LDNCB, or ISTZBM0K for ICNCB and RWNCP

This trace record is written during error recovery for channel I/O.

- ERPI is generated for communication controllers and local SNA cluster controllers.
- ERPL is generated for local non-SNA cluster controllers.
- ERPT is generated for IP over channel data link control connections.
**ERPI**, **ERPL**, **ERPX**, **ERPT**

For record types with suffix I, X, or T, the CIO events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>C&quot;ERPI&quot; for ICNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;ERPL&quot; for LDNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;ERPT&quot; for RWNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;ERPX&quot; for XCNCB</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>For ERPI, ERPL, and ERPT, link station state (NCBLNKST). For ERPX, station state (XCNSSFSM)</td>
</tr>
<tr>
<td>06</td>
<td>0</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Channel device name in EBCDIC (either a device address or device number)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>NCB address</td>
</tr>
<tr>
<td>10–13</td>
<td>Flag bytes (NCBFLAGS)</td>
</tr>
<tr>
<td>14</td>
<td>Flag byte as follows:</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>...1</td>
<td>Exception condition occurred (IOSEX flag is on)</td>
</tr>
<tr>
<td>...1</td>
<td>Error routine is in control (IOSERR flag is on)</td>
</tr>
<tr>
<td>15</td>
<td>I/O completion code (IOSCOD)</td>
</tr>
<tr>
<td>16–17</td>
<td>For ERPI, ERPL and ERPT, sense data (IOSSNS). For ERPX, 0.</td>
</tr>
</tbody>
</table>
**ESC entry for the TPESC macro**

Entry:
ESC

VIT option:
ESC

Event:
TPESC macro

VIT processing module:
ISTRACOT

Control is returned to:
ISTESC01

This trace record identifies the address of the next routine to get control in a destination vector table (DVT). The PAB address identifies the process. Using the request parameter header (RPH) address, you can correlate this trace record to the DSP entry to determine when the process was dispatched.

### Byte (hex)
**Contents**

**00–03**
Record ID: C"ESC"

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
0

**06–07**
PAB offset

**08–0B**
Dispatch address from DVT

**0C–0F**
PAB address

**10–13**
Address of the issuer of the TPESC macro

**14–17**
RPH work element

**18–1B**
One of the following items:
- Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the next module to get control)
- PAB DVT address (high-order bit of X'18' = 0)
ETSK entry for exit a subtask

Entry:
  ETSK

VIT option:
  PSS

Event:
  Exit a subtask

VIT processing module:
  ISTRACPS

Control is returned to:
  The subtask that is terminating

This trace record is written when a VTAM subtask is about to exit and return to the operating system supervisor.

This is not fully implemented for all subtask events in VTAM. There are a number of subtasks that terminate without this entry being generated.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ETSK&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>Name of the subtask about to end execution</td>
</tr>
<tr>
<td>10–13</td>
<td>TCB address of the current subtask</td>
</tr>
<tr>
<td>14–1B</td>
<td>0</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Contents of register 15</td>
</tr>
</tbody>
</table>

EXIT entry for the TPEXIT macro

Entry:
  EXIT

VIT option:
  PSS
**Event:**
TPEXIT macro

**VIT processing module:**
ISTRACPS

**Control is returned to:**
ISTAPCTX

This trace record identifies a VTAM process (PAB) that has finished executing. The PAB is rescheduled if more work elements are waiting to be processed.

In most cases, the request parameter header (RPH), whose address is in location 1C–1F, is freed or reused after this entry. One exception is the SMS dynamic expansion DYPAB (ATCPXPAB in module ISTORFPX), which has a dedicated RPH that is not used for anything else.

This entry is the complement of DSP. An EXIT should eventually follow every DSP. Use the RPH address to correlate the entries.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;EXIT&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>TPEXIT options</td>
</tr>
<tr>
<td>06–07</td>
<td>PAB offset, or 0 if no PAB</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>There is no PAB if the running process has already freed the major control block that contains the PAB.</td>
</tr>
<tr>
<td>08–0B</td>
<td>PST address</td>
</tr>
<tr>
<td>0C–0F</td>
<td>PAB address, or 0 if no PAB</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>There is no PAB if the running process has already freed the major control block that contains the PAB.</td>
</tr>
<tr>
<td>10–13</td>
<td>Address of the issuer of the TPEXIT macro</td>
</tr>
<tr>
<td>14–17</td>
<td>One of the following items:</td>
</tr>
<tr>
<td>• PAB work element queue</td>
<td></td>
</tr>
<tr>
<td>• Next dispatchable queue level for a very extended PAB (see corresponding DSP entry to find the PABFLAGS to determine the PAB type)</td>
<td></td>
</tr>
<tr>
<td>• 0, if no PAB</td>
<td></td>
</tr>
</tbody>
</table>
Note: There is no PAB if the running process has already freed the major control block that contains the PAB.

18–1B
Module name abbreviation (bytes 4, 5, 7, and 8 of the module name) or PAB DVT address (high-order bit of X'18' = 0). For an explanation of the module naming convention, see “Using module names to isolate VTAM problems” on page 23. (The module name might be unavailable if the PAB being scheduled is associated with an address space different from the current one.)

1C–1F
Request parameter header (RPH) address

Note: There is no PAB if the running process has already freed the major control block that contains the PAB.

**EXPN entry for buffer pool expansion**

**Entry:**
- **EXPN**

**VIT option:**
- **SMS**

**Event:**
- Buffer pool expansion

**VIT processing module:**
- **ISTRACSM**

**Control is returned to:**
- **ISTORAPX**

This trace record is generated whenever VTAM needs to expand a buffer pool. Buffer pool expansion may occur because of a scheduled request.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;EXPN&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Address of buffer pool control block (BPCB) for which expansion occurred</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of pool expansion block (PXB) that is expanded</td>
</tr>
<tr>
<td>10–13</td>
<td>Extent address (PXBSTADR)</td>
</tr>
</tbody>
</table>
14–15
Number of pages for an expansion in this pool. Number of pages for an expansion = (BPCBEXLN divided by 4096).

16
Expansion failure code if expansion failed

**Note:** VTAM issues codes 4 through 8 when a failure occurs during a deferred expansion.

4
Not enough CSA storage is available for the expansion.

5
VTAM cannot fix pages in storage because of insufficient page frames or some other page locking problem.

7
Storage unavailable. VTAM’s CSA limit is exceeded.

8
Expansion would cause the pool to exceed its xpanlim specification. See z/OS Communications Server: SNA Network Implementation Guide for additional information about xpanlim.

14
Not enough CSA storage is available for the expansion.

15
VTAM cannot fix pages in storage because of insufficient page frames or some other page locking problem.

17
Storage unavailable. VTAM’s CSA limit is exceeded.

18
Expansion would cause the pool to exceed its xpanlim specification. See z/OS Communications Server: SNA Network Implementation Guide for additional information about xpanlim.

See the explanation of IST154I in z/OS Communications Server: SNA Messages for more information about interpreting byte 16.

17
Flag byte

**Bit**

**Meaning**

x... ....
1=ISTORFBA caused the expansion.
0=ISTORAPX caused the expansion.

.x... ....
1=PXB was allocated by this expansion.
0=PXB existed from prior expansion.

..x. ....
1=Failure to obtain or fix storage.
0=Success in obtaining or fixing storage.

18–1B
Total number of buffers in pool after this expansion (BPCBTOTL)

1C–1F
Total number of available buffers in pool after this expansion

VTAM internal trace (VIT) record descriptions 379
EXPP entry for buffer pool expansion

Entry:
EXPP

VIT option:
CSM

Event:
Pool expansion

VIT processing module:
ISTITCCS

Control is returned to:
IVTSMCEX

This trace record is written when a CSM pool is expanded.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;EXPP&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Pool address</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Extent address</td>
</tr>
<tr>
<td>10–13</td>
<td>Number of buffers in extent</td>
</tr>
<tr>
<td>14–17</td>
<td>Total number of buffers in the pool after expansion</td>
</tr>
<tr>
<td>18–1B</td>
<td>Number of free buffers in the pool after expansion</td>
</tr>
<tr>
<td>1C–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

FBLK entry for FREEBLK macro (Part 1)

Entry:
FBLK

VIT option:
SMS
Event:
FREEBLK macro

VIT processing module:
ISTRACSM

Control is returned to:
ISTORCFB or ISTORCDF

This trace record shows the status of each FREEBLK request issued by VTAM components.

The FREEBLK macro is the complement of the GETBLK macro. FREEBLK must release the storage obtained by GETBLK. Each GBLK entry should eventually have a corresponding FBLK entry.

If the return code is nonzero, this entry is generated whether the SMS option is in effect or not. This event is treated as an exception condition and, therefore, is traced whenever the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;FBLK&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Return code</td>
</tr>
<tr>
<td>06</td>
<td>If FBLK is followed by FBL2:</td>
</tr>
<tr>
<td>00</td>
<td>AMUTSCBS</td>
</tr>
<tr>
<td>01</td>
<td>AMUTSCBL</td>
</tr>
<tr>
<td>02</td>
<td>CMPTAB</td>
</tr>
<tr>
<td>03</td>
<td>LNKDSTAB</td>
</tr>
<tr>
<td>04</td>
<td>MNPSRECV</td>
</tr>
<tr>
<td>05</td>
<td>ICV29CMP</td>
</tr>
</tbody>
</table>

If FBLK is not followed by FBL2, possible storage pool types (in hexadecimal format) follow (see z/OS Communications Server: SNA Network Implementation Guide for more information about storage pools):

<p>| 00 RUPEPRIV | 22 UTILCSAL | 44 WAR | 66 BFRTRFUL |
| 01 RUPECOMM | 23 AMU      | 45 UVRPL | 67 SILENT   |
| 02 SIB      | 24 HSIICB   | 46 DCCX | 68 DYPATH   |
| 03 SSCPFMCB | 25 LMTABLE  | 47 PLUSDATA | 69 PCDCA   |
| 04 NODAT    | 26 SAB      | 48 ADICP | 6A XNINFO   |
| 05 EPTDVT   | 27 RAB      | 49 ATGB  | 6B GRINS    |
| 06 CDRSC    | 28 PRIDBLK  | 4A TGP  | 6C BSBEXT   |
| 07 ACDEB    | 29 PRIDQAB  | 4B KEYTOKEN | 6D SOCKET |</p>
<table>
<thead>
<tr>
<th>Offset</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
<td>HSQH</td>
<td>2A AUTOLOGN</td>
</tr>
<tr>
<td>09</td>
<td>ERTF</td>
<td>2B CPWACSA</td>
</tr>
<tr>
<td>0A</td>
<td>WREEID</td>
<td>2C PGIOLBK</td>
</tr>
<tr>
<td>0B</td>
<td>FMCEXT</td>
<td>2D PRODE</td>
</tr>
<tr>
<td>0C</td>
<td>SIBEX</td>
<td>2E RIBRANT</td>
</tr>
<tr>
<td>0D</td>
<td>(Not used)</td>
<td>2F CANT</td>
</tr>
<tr>
<td>0E</td>
<td>UECB</td>
<td>30 CAB</td>
</tr>
<tr>
<td>0F</td>
<td>IOBLOCK</td>
<td>31 CNSFACUD</td>
</tr>
<tr>
<td>10</td>
<td>SRTE</td>
<td>32 BFRTRACE</td>
</tr>
<tr>
<td>11</td>
<td>ISTTRCEL</td>
<td>33 DMTSQ</td>
</tr>
<tr>
<td>12</td>
<td>UTILPVTS</td>
<td>34 FMCB</td>
</tr>
<tr>
<td>13</td>
<td>VRPL</td>
<td>35 PLUSFMCB</td>
</tr>
<tr>
<td>14</td>
<td>POWERPRV</td>
<td>36 PXBFIXED</td>
</tr>
<tr>
<td>15</td>
<td>POWECOMM</td>
<td>37 PXBPAGED</td>
</tr>
<tr>
<td>16</td>
<td>PULURDTE</td>
<td>38 PLUSC</td>
</tr>
<tr>
<td>17</td>
<td>PAQ</td>
<td>39 NSSCB</td>
</tr>
<tr>
<td>18</td>
<td>RAQ</td>
<td>3A (Not used)</td>
</tr>
<tr>
<td>19</td>
<td>CPWAPVT</td>
<td>3B (Not used)</td>
</tr>
<tr>
<td>1A</td>
<td>ERICPOOL</td>
<td>3C (Not used)</td>
</tr>
<tr>
<td>1B</td>
<td>SIBIX</td>
<td>3D FMH5</td>
</tr>
<tr>
<td>1C</td>
<td>CDAJSCP</td>
<td>3E OOBTSBC</td>
</tr>
<tr>
<td>1D</td>
<td>GWNAJSCP</td>
<td>3F SLD</td>
</tr>
<tr>
<td>1E</td>
<td>IOSIB</td>
<td>40 NSRUS</td>
</tr>
<tr>
<td>1F</td>
<td>DSSIB</td>
<td>41 NSRUL</td>
</tr>
<tr>
<td>20</td>
<td>UTILPVTL</td>
<td>42 RUCON</td>
</tr>
<tr>
<td>21</td>
<td>UTILCSSAS</td>
<td>43 STB</td>
</tr>
</tbody>
</table>

When set to 1, indicates that storage obtained through GETBLK request and converted to VTALLOC request is returned to system

**08–0B**
Address of block freed (or 0 if FREEBLK failed)

**0C–0F**
Address of storage pool anchor block (SPTAE) or, if FBLK is followed by FBL2, address of DSPSP

**10–13**
Address of the issuer of the FREEBLK macro

**14–17**
Length of storage freed not including the 8-byte header

**18–1B**
Caller of utility routine or 0. If the FREEBLK macro was issued from a utility routine, the address of the utility's caller is placed here. A 0 address indicates that the macro was issued directly by the caller (see return address) without a utility routine.

**1C–1F**
Request parameter header (RPH) address

**FBL2 entry for FREEBLK macro (Part 2)**

**Entry:**
FBL2

**VIT option:**
SMS

**Event:**
FREEBLK macro

**VIT processing module:**
ISTRACSM

This trace record is a continuation of the FBLK entry; it is generated only if the storage that is freed is in a data space.

FBL2 returns the address of the data space descriptor control block (DSDCB) and the data space name from the DSDCB. When using IPCS, the data space name is required to look at data in the data space.
Byte (hex)
Contents
00–03  
Record ID: C"FBL2"
04–07  
Address of DSDCB
08–0F  
Data space name from the DSDCB
10–1F  
0

FB64 entry for FREEB64 macro

Entry:
FB64
VIT option:
SMS
Event:
FreeB64 macro
VIT processing module:
ISTRACOT
Control is returned to:
ISTO64FB

This trace record shows the status of each FreeB64 request that VTAM components issue. The FreeB64 macro is the complement of the GetB64 macro. FreeB64 releases the storage that GetB64 obtains. Each GB64 entry should eventually have a corresponding FB64 entry. If the return code is not zero, this entry is generated whether the SMS option is in effect or not. This event is treated as an exception condition and, therefore, is traced whenever the VIT is active.

Byte (hex)
Contents
00–03  
Record ID: C'FB64'
04–05  
ID is the primary address space ID (ASID).
Return code

Possible storage pool types in hexadecimal format. For example, 86 SM3270. For more information about storage pools, see z/OS Communications Server: SNA Network Implementation Guide.

Address of block that is freed, or 0 if FreeB64 failed.

Address of the issuer of the FreeB64 macro.

FBFlags

Length of storage that is freed.

Address of storage pool anchor block (SPTAE).

Request parameter header (RPH) address.

---

**FIXB entry for FIX_BUFFER requests**

**Entry:**
- FIXB

**VIT option:**
- CSM

**Event:**
- IVTCSM REQUEST=FIX_BUFFER

**VIT processing module:**
- ISTITCCS

**Control is returned to:**
- IVTSMCBF or IVTSM6BF

**Notes:**
- IVTSMCBF processes FIX_BUFFER requests in 31-bit addressing mode.
- IVTSM6BF processes FIX_BUFFER requests in 64-bit addressing mode.

This trace record provides the status of an IVTCSM REQUEST=FIX_BUFFER macroinstruction.

---

### Byte (hex)

#### Contents

| 00-03 | Record ID: C"FIXB"
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

---

384 z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
Trace record flag:

- B'0... ....' Indicates that this is the last record for this event.
- B'1... ....' Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.
- B'.1. ....' Indicates that the buffer list is in 64-bit storage.

Trace record number to correlate all the entries for this particular event.

Return code

Reason code

Address of utility routine caller or return address of the issuer of the IVTCSM macroinstruction

Address of the buffer list entry that was being processed when the error was encountered.

When an error occurs in 64-bit addressing mode, the last buffer list entry here is X'FFFFFFFF'. The FIX6 entry follows with the 64-bit address of the last buffer list entry.

Number of buffers to be fixed

THREAD value if specified or 0 if THREAD is not specified. The THREAD value is used only to correlate this trace record to a specific IVTCSM macroinstruction.

**FIX2 entry for FIX_BUFFER requests**

Entry:

- FIX2

VIT option:

- CSM

Event:

- IVTCSM REQUEST=FIX_BUFFER

VIT processing module:

- ISTITCCS

This trace record is a continuation of the FIXB trace record. Each FIX2 record traces, at most, two buffers that were requested to be placed in a fixed state.
**Byte (hex)**

**Contents**

**00–03**
- Record ID: C"FIX2"

**04–05**
- 0

**06**
- Trace record flag:
  - B'0... ...'
    - Indicates that this is the last record for this event.
  - B'1... ...'
    - Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

**07**
- Trace record number to correlate all the entries for this particular event.

**08–13**
- Buffer token contained in input buffer list entry

**14–1F**
- Buffer token contained in input buffer list entry or 0

**FIX6 entry for FIX_BUFFER requests**

**Entry:**
- FIX6

**VIT option:**
- CSM

**Event:**
- IVTCSM REQUEST=FIX_BUFFER

**VIT processing module:**
- ISTITCCS

This trace record is a continuation of the FIXB trace record. It contains the 64-bit buffer list entry address. This entry is present only when the FIX_BUFFER request failed with the error return code in 64-bit addressing mode.
00-03
Record ID: C'FIX6'

04-05
0

06
Trace record flags:

B'0... ....'
Indicates that this is the last record for this event.

B'1... ....'
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

B'.1.... '
Indicates that the buffer list is in 64-bit storage.

07
Trace record number to correlate all the entries for this particular event.

08-F
0

10-17
64-bit address of the last buffer list entry.

18-1F
0

**FRBF entry for FREE_BUFFER requests**

Entry:
FRBF

VIT option:
CSM

Event:
IVTCSM REQUEST=FREE_BUFFER

VIT processing module:
ISTITCCS

Control is returned to:
IVTSMCFB or IVTSM6FB

Notes:
- IVTSMCFB processes FREE_BUFFER requests in 31-bit addressing mode.
- IVTSM6FB processes FREE_BUFFER requests in 64-bit addressing mode.

This trace record provides the status of an IVTCSM REQUEST=FREE_BUFFER macroinstruction.
Record ID: C"FRBF"

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Trace record flags:
- B'0... ....': Indicates that this is the last record for this event.
- B'1... ....': Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.
- B'.1. ....': Indicates that the buffer list is in 64-bit storage.

Trace record number to correlate all the entries for this particular event.

Return code

Reason code

Address of utility routine caller or return address of the issuer of the IVTCSM macroinstruction.

Address of the buffer list entry that was being processed when the error was encountered.

When an error occurs in 64-bit addressing mode, the last buffer list entry here is X'FFFFFFFF'. The FRB6 entry follows with the 64-bit address of the last buffer list entry.

Number of buffers requested to be freed.

THREAD value if specified or 0 if THREAD is not specified. The THREAD value is used only to correlate this trace record to a specific IVTCSM macroinstruction.

FRB2 entry for FREE_BUFFER requests

Entry:
FRB2

VIT option:
CSM

Event:
IVTCSM REQUEST=FREE_BUFFER

VIT processing module:
ISTITCCS

This trace record is a continuation of the FRBF trace record. It contains information about the buffers that were freed. Each FRB2 record traces, at most, two buffers that were freed.
Byte (hex)  
Contents  
00–03  
Record ID: C"FRB2"  
04–05  
0  
06  
Trace record flag:  
B'0... ....'  
Indicates that this is the last record for this event.  
B'1... ....'  
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.  
07  
Trace record number to correlate all the entries for this particular event  
08–13  
Buffer token contained in input buffer list entry  
14–1F  
Buffer token contained in input buffer list entry or 0  
FRB6 entry for FREE_BUFFER requests  
Entry:  
FRB6  
VIT option:  
CSM  
Event:  
IVTCSM REQUEST=FREE_BUFFER  
VIT processing module:  
ISTITCCS  
This trace record is a continuation of the FRBF trace record. It contains the 64-bit buffer list entry address. This entry is present only when the FREE_BUFFER request failed with the error return code in 64-bit addressing mode.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 0 0 1 2 3 4 5 6 7 8 9 A B C D E F 0 1 1 1 1 1</td>
<td>FRB6</td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) record descriptions 389
00-03
  Record ID: C'FRB6'

04-05
  0

06
  Trace record flags:
  B'0... ....'
  Indicates that this is the last record for this event.
  B'1... ....'
  Indicates that additional records exist for this event. Use the trace record number from this entry
  to locate corresponding continuation records.
  B'..1. ....'
  Indicates that the buffer list is in 64-bit storage.

07
  Trace record number to correlate all the entries for this particular event.

08-F
  0

10-17
  64-bit address of the last buffer list entry.

18-1F
  0

**FRES entry for FREESTOR macro**

**Entry:**
  FRES

**VIT option:**
  SMS

**Event:**
  Invocation of the FREESTOR macro

**VIT processing module:**
  ISTRACSM

**Control is returned to:**
  Module invoking the FREESTOR macro

This trace record shows the status of each FREESTOR request issued by VTAM components. The
FREESTOR macro releases the storage obtained by the GETSTOR macro. Each GETS entry should
eventually have a corresponding FRES entry.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FRES</td>
<td>ID</td>
<td>STORAGE ADDRESS</td>
<td>ADDRESS OF ORGSH</td>
<td>RETURN ADDRESS</td>
<td>ALLOCLEN</td>
<td>0</td>
<td>RETURN ADDRESS OF CALLER</td>
<td>RPH ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>AB</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**
Record ID: C"FRES"

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Return code

Pool Index (in hex)

<table>
<thead>
<tr>
<th>Pool Index</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Start</td>
</tr>
<tr>
<td>00</td>
<td>CFSACCP</td>
</tr>
<tr>
<td>01</td>
<td>CFSACCPD</td>
</tr>
<tr>
<td>02</td>
<td>CFSACCCS</td>
</tr>
<tr>
<td>03</td>
<td>CFSACCCD</td>
</tr>
<tr>
<td>04</td>
<td>RPMNPSPS</td>
</tr>
<tr>
<td>05</td>
<td>HIPOOLPS</td>
</tr>
<tr>
<td>06</td>
<td>MRPOOLPS</td>
</tr>
<tr>
<td>07</td>
<td>NLPDELPD</td>
</tr>
<tr>
<td>08</td>
<td>TCPIOCD</td>
</tr>
<tr>
<td>09</td>
<td>CSAPGVFE</td>
</tr>
<tr>
<td>0A</td>
<td>CSAPGVNE</td>
</tr>
<tr>
<td>0B</td>
<td>CSAPGJFE</td>
</tr>
<tr>
<td>0C</td>
<td>CSAPGJNE</td>
</tr>
<tr>
<td>0D</td>
<td>CSAXVFE</td>
</tr>
<tr>
<td>0E</td>
<td>CSAFXVNE</td>
</tr>
<tr>
<td>0F</td>
<td>CSAFXJFE</td>
</tr>
<tr>
<td>10</td>
<td>CSAPGVNE</td>
</tr>
<tr>
<td>11</td>
<td>PVTPGJFJ</td>
</tr>
<tr>
<td>12</td>
<td>PVTPGJFT</td>
</tr>
<tr>
<td>13</td>
<td>PVTPGJNT</td>
</tr>
<tr>
<td>14</td>
<td>PVTPGVFT</td>
</tr>
<tr>
<td>15</td>
<td>PVTPGVNT</td>
</tr>
<tr>
<td>16</td>
<td>CFSBUFPD</td>
</tr>
<tr>
<td>17</td>
<td>CFSBUFCS</td>
</tr>
<tr>
<td>18</td>
<td>CFSBUFPS</td>
</tr>
<tr>
<td>19</td>
<td>CFSBUFCD</td>
</tr>
<tr>
<td>1A</td>
<td>ALPHCD</td>
</tr>
<tr>
<td>1B</td>
<td>EEHNMPID</td>
</tr>
</tbody>
</table>

Address of storage freed

Address of GETSTOR header (ISTORGSH)

Address of the invoker of FREESTOR macro

Size of pool

Caller of utility routine or 0. If the FREESTOR macro was issued from a utility routine, the address of the utility's caller is placed here. A 0 address indicates that the macro was issued directly by the caller (see return address) without a utility routine.

Request parameter header (RPH) address

**FR64 entry for FRE64COMM or FRE64PVT requests**

**Entry:**

FR64

**VIT option:**

SMS

**Event:**

FRE64COMM or FRE64PVT request

**VIT processing module:**

ISTITCSH

**Control is returned to:**

Module invoking the INTRACE macro with type FRE64 that caused the record to be produced.
### Byte (hex)

**Contents**

<table>
<thead>
<tr>
<th>Hex</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03</td>
<td>Record ID: C’FR64’</td>
</tr>
<tr>
<td>04-05</td>
<td>ID is the primary address space ID (ASID).</td>
</tr>
<tr>
<td>06</td>
<td>Storage type</td>
</tr>
<tr>
<td>C</td>
<td>64-bit common storage</td>
</tr>
<tr>
<td>P</td>
<td>64-bit private storage</td>
</tr>
<tr>
<td>07</td>
<td>Flags</td>
</tr>
<tr>
<td>1</td>
<td>DREF</td>
</tr>
<tr>
<td>08-0F</td>
<td>64-bit buffer address</td>
</tr>
<tr>
<td>10-13</td>
<td>Address of issuer of the FRE64 trace request</td>
</tr>
<tr>
<td>14-15</td>
<td>Number of megabyte segments</td>
</tr>
<tr>
<td>16-1B</td>
<td>0</td>
</tr>
<tr>
<td>1C-1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>
GBLK entry for GETBLK macro (Part 1)

Entry:

GBLK

VIT option:

SMS

Event:

GETBLK macro

VIT processing module:

ISTRACSM

Control is returned to:

ISTORCDG or ISTORCGB

This trace record shows the status of each GETBLK request issued by VTAM components.

The FREEBLK macro is the complement of the GETBLK macro. FREEBLK must release the storage obtained by GETBLK. Each GBLK entry should eventually have a corresponding FBLK entry.

If the return code is nonzero, this entry is generated whether the SMS option is in effect or not. It is treated as an exception condition and is always traced if the VIT is active, regardless of the SMS option.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;GBLK&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Return code</td>
</tr>
<tr>
<td>06</td>
<td>If GBLK is followed by GBL2:</td>
</tr>
<tr>
<td>00</td>
<td>AMUTSCBS</td>
</tr>
<tr>
<td>01</td>
<td>AMUTSCBL</td>
</tr>
<tr>
<td>02</td>
<td>CMPTAB</td>
</tr>
<tr>
<td>03</td>
<td>LNKDSTAB</td>
</tr>
</tbody>
</table>
**MNPSRECV**

**ICV29CMP**

If **GBLK** is not followed by **GBL2**, possible storage pool types (in hexadecimal format) follow (see *z/OS Communications Server: SNA Network Implementation Guide* for more information about storage pools):

<table>
<thead>
<tr>
<th>Hex Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>RUPEPRIV</td>
</tr>
<tr>
<td>01</td>
<td>RUPECOMM</td>
</tr>
<tr>
<td>02</td>
<td>SIB</td>
</tr>
<tr>
<td>03</td>
<td>SSCPFMCB</td>
</tr>
<tr>
<td>04</td>
<td>NQDAT</td>
</tr>
<tr>
<td>05</td>
<td>EPTDVT</td>
</tr>
<tr>
<td>06</td>
<td>CDSC</td>
</tr>
<tr>
<td>07</td>
<td>ACDEB</td>
</tr>
<tr>
<td>08</td>
<td>HSQH</td>
</tr>
<tr>
<td>09</td>
<td>ERT</td>
</tr>
<tr>
<td>0A</td>
<td>WREEID</td>
</tr>
<tr>
<td>0B</td>
<td>FMCEBEXT</td>
</tr>
<tr>
<td>0C</td>
<td>SIBEXT</td>
</tr>
<tr>
<td>0D</td>
<td>(Not used)</td>
</tr>
<tr>
<td>0E</td>
<td>UECB</td>
</tr>
<tr>
<td>0F</td>
<td>IOBLOCK</td>
</tr>
<tr>
<td>10</td>
<td>SRTE</td>
</tr>
<tr>
<td>11</td>
<td>ISTTRCEL</td>
</tr>
<tr>
<td>12</td>
<td>UTILPVTS</td>
</tr>
<tr>
<td>13</td>
<td>VRPL</td>
</tr>
<tr>
<td>14</td>
<td>POWERPRIV</td>
</tr>
<tr>
<td>15</td>
<td>POWECOMM</td>
</tr>
<tr>
<td>16</td>
<td>PULURTE</td>
</tr>
<tr>
<td>17</td>
<td>PAQ</td>
</tr>
<tr>
<td>18</td>
<td>RAQ</td>
</tr>
<tr>
<td>19</td>
<td>CPWAPVT</td>
</tr>
<tr>
<td>1A</td>
<td>ERICPOOL</td>
</tr>
<tr>
<td>1B</td>
<td>SIBIX</td>
</tr>
<tr>
<td>1C</td>
<td>CDAJSCP</td>
</tr>
<tr>
<td>1D</td>
<td>GWNADSCP</td>
</tr>
<tr>
<td>1E</td>
<td>IOSIB</td>
</tr>
<tr>
<td>1F</td>
<td>DSSIB</td>
</tr>
<tr>
<td>20</td>
<td>UTILPVTL</td>
</tr>
<tr>
<td>21</td>
<td>UTICAS</td>
</tr>
</tbody>
</table>

**Flags:**

**Bit**

<table>
<thead>
<tr>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1... .....</td>
</tr>
<tr>
<td>.1... .....</td>
</tr>
<tr>
<td>...xx xxx.</td>
</tr>
<tr>
<td>00</td>
</tr>
</tbody>
</table>

**08–0B**

Address of block obtained (or 0 if GETBLK failed)

**0C–0F**

Address of storage pool anchor block (SPTAE) or, if GBLK is followed by GBL2, address of DSPSP

**10–13**

Address of the issuer of the GETBLK macro

**14–15**

If byte 07, bits 0 and 1 are 0, length of storage specified by the user. Otherwise, 0.
16-17
If byte 07, bits 0 and 1, are 0, length of storage obtained, rounded to the correct subpool length.
Otherwise, 0. This length does not include the 8-byte GETBLK header.

18-1B
 Caller of utility routine or 0. If the GETBLK macro was issued from a utility routine, the address of the
utility's caller is placed here. A 0 address indicates that the macro was issued directly by the caller
(see return address) without a utility routine.

1C-1F
Request parameter header (RPH) address

**GBL2 entry for GETBLK macro (Part 2)**

**Entry:**
GBL2

**VIT option:**
SMS

**Event:**
GETBLK macro

**VIT processing module:**
ISTRACSM

This trace record is a continuation of the GETBLK entry and provides the address of the data space
descriptor control block (DSDCB). This trace record is generated only if the storage that is acquired is in a
data space.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;GBL2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Address of DSDCB</td>
</tr>
</tbody>
</table>
| 08–0F      | Data space name from the DSDCB. When you are using IPCS, the data space name is required often to
            look at data in the data space. |
| 10–13      | If byte 07, bit 0 of GBLK record is 1, length of storage specified by the user. Otherwise, 0. |
| 14-17      | If byte 07, bit 0 of GBLK record is 1, length of storage obtained, rounded to the correct subpool length. 
            Otherwise, 0. This length does not include the 8-byte GETBLK header. |
| 18-1F      | 0 |

**GBL3 entry for GETBLK macro (Part 2)**

**Entry:**
GBL3
VIT option:
  SMS

Event:
  GETBLK macro

VIT processing module:
  ISTRACSM

This trace record is a continuation of the GETBLK entry and provides the length of storage requested and the length of storage obtained if either length is 65,536 or greater. This trace record is generated when needed to provide lengths of that magnitude, but only if the storage that is acquired is not in a data space.

```
<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0</th>
<th>0 0 0 0 0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>GBL3</td>
<td>0</td>
</tr>
<tr>
<td>LENGTH OF STORAGE REQUESTED</td>
<td>LENGTH OF STORAGE OBTAINED</td>
</tr>
</tbody>
</table>
```

**Byte (hex) Contents**

**00–03**  
Record ID: C"GBL3"

**04–0F**  
0

**10–13**  
Length of the storage specified by the user.

**14–17**  
Length of storage obtained, rounded to the correct subpool length. This length does not include the 8-byte GETBLK header.

**18–1F**  
0

---

**GB64 entry for GETB64 macro**

**Entry:**  
GETB64

**VIT option:**  
SMS

**Event:**  
GETB64 macro

**VIT processing module:**  
ISTRACOT

**Control is returned to:**  
ISTO64GB

This trace record shows the status of each GetB64 request that VTAM components issue. The FreeB64 macro is the complement of the GetB64 macro. FreeB64 releases the storage that GetB64 obtains. Each GB64 entry should eventually have a corresponding FB64 entry. If the return code is not zero, this entry is generated whether the SMS option is in effect or not. This event is treated as an exception condition and, therefore, is traced whenever the VIT is active.
Entry:

Entry: GCEL

VIT option:

VIT option: CIA

VIT processing module:

Processing module: ISTRACOU

Control is returned to:

Control is returned to: IUTLCCC (Module that issued the request)

This trace record shows the status of each (Get Cell) request for a CSM cell. It captures the outcome of the CELLCTL FUNC(GET) macro. This macro is used by DLC components.

There is no corresponding free cell event. The cells are freed directly to CSM using IVTCSM REQUEST=FREE_BUFFER (see “FRBF entry for FREE_BUFFER requests” on page 387) and are normally freed by non-VTAM components.
Byte (hex)

Contents

**01 - 03**

Record ID: C"GCEL"

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

0

**06 - 07**

Return code or length

**08 - 0B**

CSM buffer address. If this trace entry is for a 64 bit address, this field is X'FFFFFFFF'.

**0C - 0F**

Address of caller

**10 - 1B**

CSM buffer token

**1C - 1F**

RPH address

---

**GCEx entry for CSM storage movement**

**Entry:**

GCEA or GCER

**VIT option:**

CIA

**Event:**

CSM storage movement

**VIT processing module:**

ISTRACOU

**Control is returned to:**

The module that issued the INTRACE
This trace record shows the movement of CSM cells between different queues. This macro is used for QDIO and HiperSockets devices.

<table>
<thead>
<tr>
<th>GCEA</th>
<th>GCER</th>
<th>ID</th>
<th>CNT</th>
<th>ADDR OF CELL</th>
<th>ADDR OF CALLER</th>
<th>ADDR OF QUEUE Anchor</th>
<th>ANCHOR HEAD</th>
<th>MOD EYE</th>
<th>RPH OR THREAD</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>03</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00 - 03**

Record ID:

- C"GCEA" for adding CSM cell to queue
- C"GCER" for removing CSM cell from queue

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

0

**06 - 07**

Count of buffers available on queue

**08 - 0B**

CSM buffer address. If this trace entry is for a 64 bit address, this field is X'FFFFFFFF'.

**0C - 0F**

Address of caller

**10 - 13**

Starting address of the queue that the CSM cell is being added to or removed from. If this trace entry is for a 64 bit address mode, this field is X'FFFFFFFF'.

**14 - 17**

First 4 bytes of anchor

**18 - 19**

Module eye catcher of caller

**1A - 1B**

0

**1C - 1F**

RPH address or thread

**GCE6 entry for CSM storage movement**

The GCE6 trace record shows the movement of CSM cells between different queues.

**Entry:**

GCE6

**VIT option:**

CIA

**Event:**

CSM storage movement

**VIT processing module:**

ISTRACOU

**Control is returned to:**

The module that issued the INTRACE
This trace record is written after GCEL, GCEA and GCER when the trace being cut is for 64-bit addressing mode. This trace record shows the movement of CSM cells between different queues. This macro is used for QDIO and HiperSockets devices.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 - 03</td>
<td>Record ID: C&quot;GCE6&quot;</td>
</tr>
<tr>
<td>04 - 07</td>
<td>0</td>
</tr>
<tr>
<td>08 - 0F</td>
<td>CSM buffer address. This field is a 64-bit buffer address.</td>
</tr>
<tr>
<td>10 - 17</td>
<td>Starting address of the queue that the CSM cell is being added to or removed from. This field is a 64-bit address.</td>
</tr>
<tr>
<td>18 - 1B</td>
<td>First 4 bytes of anchor</td>
</tr>
<tr>
<td>1C - 1F</td>
<td>0</td>
</tr>
</tbody>
</table>

**GETS entry for GETSTOR macro**

**Entry:**
GETS

**VIT option:**
SMS

**Event:**
Invocation of the GETSTOR macro

**VIT processing module:**
ISTRACSM

**Control is returned to:**
Module invoking the GETSTOR macro

This trace record shows the status of each GETSTOR request issued by VTAM components.

A FREESTOR macro releases the storage obtained by GETSTOR. Each GETS entry should eventually have a corresponding FRES entry.
Byte (hex)  Contents
00–03  Record ID: C"GETS"
04  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05  0
06  Return code
07  Pool Index (in hex)

<table>
<thead>
<tr>
<th>Pool Index</th>
<th>Address in hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 CFSACCPS</td>
<td>07 NLPDELPD</td>
</tr>
<tr>
<td>01 CFSACCPD</td>
<td>08 TCPIOCD</td>
</tr>
<tr>
<td>02 CFSACCCS</td>
<td>09 CSAPGVFE</td>
</tr>
<tr>
<td>03 CFSACCDD</td>
<td>0A CSAPGVNE</td>
</tr>
<tr>
<td>04 RPMNPSPS</td>
<td>0B CSAFXJFE</td>
</tr>
<tr>
<td>05 HIPPOOLPS</td>
<td>0C CSAPGJFE</td>
</tr>
<tr>
<td>06 MRPOOLPS</td>
<td>0D CSAFXVFE</td>
</tr>
</tbody>
</table>

08–0B  Address of storage returned
0C–0F  Address of GETSTOR header (ISTORGSH)
10–13  Address of the invoker of GETSTOR macro
14–17  Allocated storage length
18–1B  Caller of utility routine. If the GETSTOR macro was issued from a utility routine, the address of the utility's caller is placed here. A 0 address indicates that the macro was issued directly by the caller (see return address) without a utility routine.
1C–1F  Request parameter header (RPH) address

**GNAM entry for GNAME macro invoked (Part 1)**

Entry:  
GNAM
VIT option:  
SSCP
Event:  
GNAME macro invocation
VIT processing module:  
ISTRACSC
Control is returned to:  
ISTCPCGM

This trace record is written when generic mapping information in the generic resource coupling facility structure is created, deleted, or changed using the GNAME macroinstruction.
Byte (hex)
Contents
00–03
Record ID: C"GNAM"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Return code from the GNAME macro
X'00'
Request completed successfully. This is an existing USERVAR return code.
X'01'
Request completed successfully and local data was returned. This is returned from GNAME only if SEARCH is CF&LOCAL.
X'04'
Resource specified could not be located. This is an existing USERVAR return code.
X'51'
Request failed because of APPC restrictions. This is an existing USERVAR return code.
X'52'
Request failed because of a parameter specification that was not valid. This is an existing USERVAR return code.
X'53'
The function requested could not be completed because of a storage allocation failure. This is an existing USERVAR return code.
X'54'
The real instance of user variable to be deleted could not be located. This is an existing USERVAR return code.
X'55'
A conflict was found because of NETIDs.
X'56'
A conflict was found because of CPNAMEs.
X'57'
VTAM is halting.
X'58'
ENDAFFINITY was issued for a session that was VTAM owned.
X'59'
RSCTYPE value conflicted between USERVAR and GR.
X'5A'
An attempt to change the generic name for a given application failed because SPTEs pertaining to the previous generic name still exist.
X'60'
A session pair could not be found in an SPT entry.
X'61'
An attempt to change the generic name for a given application failed because either (1) the existing generic name was never deleted, or (2) because SPTEs pertaining to the existing generic name still exist.

X'62'
An attempt to repopulate the generic resource coupling facility structure has failed because the local data being used for repopulation is back level compared to data already in the structure.

X'63'
An attempt to repopulate the generic resource coupling facility structure has succeeded but backlevel data has been identified in the structure and additional structure cleanup may be required.

X'64'
An attempt to increment the session count for an SPTE failed because the name type of the SPTE did not match what was expected.

X'65'
Request failed because of TSO mismatched.

X'66'
An attempt to register a generic resource with the work load manager failed because of the use of an STOKEN that is not valid.

X'A0'
A request made against the generic resources coupling facility structure will complete synchronously.

X'A1'
An attempt to update information in the generic resources coupling facility structure failed because that data had changed since it was last read. The data should be re-read and then modified again.

X'A2'
An attempt to access the generic resources coupling facility structure failed for an unexpected reason.

X'A3'
There is currently no connection to the generic resources coupling facility structure.

X'A4'
The buffers provided for reading data from the generic resources coupling facility structure were insufficient for buffering all the data associated with the list entry being read. No data (adjunct or element) is returned.

X'A5'
A read from the generic resources coupling facility structure failed because the requested data could not be found in that structure.

X'A6'
Data could not be added to the generic resources coupling facility structure because there is insufficient storage in the generic resources coupling facility structure to hold it.

06
Reason macro invoked:

X'01'
Find generic resource mapping.

X'02'
Find generic resource mapping or USERVAR.

X'03'
Add the application program network name to generic mapping.

X'04'
Delete the application program network name from generic mapping.
Write generic mapping to coupling facility structure.
Free local copy of generic mapping.
Find a generic resource name for an application program network name.
Find the generic number for a generic resource name.

07
Event flags:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>The addition or deletion of an application program network name is because of a change in CP-CP status.</td>
</tr>
<tr>
<td>.1...</td>
<td>The addition or deletion of an application program network name is because of a SETLOGON GNAMEADD or SETLOGON GNAMEDEL.</td>
</tr>
<tr>
<td>..1..</td>
<td>Partner is local, a local real instance will be given precedence used for resolution only. See byte 14.</td>
</tr>
<tr>
<td>...1...</td>
<td>Resolve to this node, the real instance must be on this node used for resolution only. See byte 14.</td>
</tr>
<tr>
<td>....1...</td>
<td>Update resolution count, the resolution count will be incremented or decremented used for resolution and termination only. See byte 14.</td>
</tr>
<tr>
<td>.... ..1..</td>
<td>Update session count, the session count will be incremented or decremented used for resolution and termination only. See byte 14.</td>
</tr>
<tr>
<td>.... ....x</td>
<td>Real instance is a subordinate resource.</td>
</tr>
</tbody>
</table>

08–0F
Generic name
10–13
Address of the invoker of the GNAME macro
14
Reason code (used for reason macros X'01' and X'05')
X'00'
Resolution
X'01'
Reserved
X'02'
Update session counts
X'03'
Termination
15–16
0
17
Resolution count or 0

18–1B
The generic number if available, otherwise 0

1C–1F
Request parameter header (RPH) address

**GNA2 entry for GNAME macro invoked (Part 2)**

**Entry:**
GNA2

**VIT option:**
SSCP

**Event:**
GNAME macro invocation

**VIT processing module:**
ISTRACSC

This trace record is a continuation of the GNAM entry.

<table>
<thead>
<tr>
<th>GNA2</th>
<th>SESS COUNT OR 0</th>
<th>PCID</th>
<th>NETID OF APPLICATION PROGRAM NETWORK NAME</th>
<th>APPLICATION PROGRAM NETWORK NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
</tr>
<tr>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
</tr>
<tr>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
</tr>
<tr>
<td>0E</td>
<td>0F</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>18</td>
<td>19</td>
<td>1A</td>
<td>1B</td>
<td>1C</td>
</tr>
<tr>
<td>1D</td>
<td>1E</td>
<td>1F</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03
Record ID: C’GNA2’

04–07
Session count or 0

08–0F
PCID if associated with a session, otherwise 0

10–17
Network ID of application program network name

18–1F
Application program network name

**GT64 entry for GET64COMM or GET64PVT requests**

**Entry:**
GT64

**VIT option:**
SMS

**Event:**
GET64COMM or GET64PVT request

**VIT processing module:**
ISTITCSH
Control is returned to:
Module invoking the INTRACE macro with type GET64 that caused the record to be produced.

<table>
<thead>
<tr>
<th>GT64</th>
<th>ID</th>
<th>STORAGE TYPE</th>
<th>64 - BIT BUFFER ADDRESS</th>
<th>RETURN ADDRESS</th>
<th>0</th>
<th>RET</th>
<th>R ASON CODE</th>
<th>RPH ADDRESS</th>
</tr>
</thead>
</table>

**Byte (hex)**

**Contents**

**00-03**
Record ID: C'GT64'

**04-05**
ID is the primary address space ID (ASID).

**06**
Storage type

C  
64-bit common storage

P  
64-bit private storage

**07**
Flags

**Bit**
**Meaning**

1... DREF
The memory object is referenced when running disabled. DREF attribute applies to the entire memory object. Pages are backed in real at first reference. They are never paged out to AUX.

**08-0F**
64-bit buffer address

**10-13**
Address of issuer of the GET64 trace request

**14-15**
Number of megabyte segments

**16-17**
0

**18-19**
Return code

**1A–1B**
Reason code
GTBF entry for GET_BUFFER requests

- **Entry:** GTBF
- **VIT option:** CSM
- **Event:** IVTCSM REQUEST=GET_BUFFER
- **VIT processing module:** ISTITCCS or ISTIT6CS
- **Control is returned to:** IVTSMCGB or IVTSM6GB

**Notes:**
- IVTSMCGB processes GET_BUFFER requests in 31-bit addressing mode.
- IVTSM6GB processes GET_BUFFER requests in 64-bit addressing mode.

This trace record provides the status of IVTCSM REQUEST=GET_BUFFER macroinstructions.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;GTBF&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace record flag:</td>
</tr>
<tr>
<td>B'0... ....'</td>
<td>Indicates that this is the last record for this event.</td>
</tr>
<tr>
<td>B'1... ....'</td>
<td>Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.</td>
</tr>
<tr>
<td>B'.1. ....'</td>
<td>Indicates that the buffer list is in 64-bit storage.</td>
</tr>
<tr>
<td>07</td>
<td>Trace record number to correlate all the entries for this particular event</td>
</tr>
<tr>
<td>08</td>
<td>Pool identifier, can be one of the following values:</td>
</tr>
</tbody>
</table>
X'10'
  4K ECSA pool
X'20'
  16K ECSA pool
X'30'
  32K ECSA pool
X'40'
  60K ECSA pool
X'60'
  180K ECSA pool
X'90'
  4K 31-bit backed DSPACE pool
X'98'
  4K 64-bit backed DSPACE pool
X'A0'
  16K 31-bit backed DSPACE pool
X'A8'
  16K 64-bit backed DSPACE pool
X'B0'
  32K 31-bit backed DSPACE pool
X'B8'
  32K 64-bit backed DSPACE pool
X'C0'
  60K 31-bit backed DSPACE pool
X'C8'
  60K 64-bit backed DSPACE pool
X'E0'
  180K 31-bit backed DSPACE pool
X'E8'
  180K 64-bit backed DSPACE pool
X'9C'
  4K buffer size 64-bit high virtual common (HVCOMM) storage pool
X'AC'
  16K buffer size 64-bit high virtual common (HVCOMM) storage pool
X'BC'
  32K buffer size 64-bit high virtual common (HVCOMM) storage pool
X'CC'
  60K buffer size 64-bit high virtual common (HVCOMM) storage pool
X'EC'
  180K buffer size 64-bit high virtual common (HVCOMM) storage pool

09–0B
  0

0C–0D
  Return code

0E–0F
  Reason code

10–13
  Address of utility routine caller or return address of the issuer of the IVTCSM macroinstruction
14–17
Address of the buffer list entry that was being processed when the error was encountered.

When an error occurs in 64-bit addressing mode, the last buffer list entry here is $'FFFFFFFF'$. The GTB6 entry follows with the 64-bit address of the last buffer list entry.

18–1B
0

1C–1F
THREAD value if specified or 0 if THREAD is not specified. The THREAD value is used only to correlate this trace record to a specific IVTCSM macroinstruction.

**GTB2 entry for GET_BUFFER requests**

**Entry:**
GTB2

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=GET_BUFFER

**VIT processing module:**
ISTITCCS or ISTIT6CS

This trace record is a continuation of the GTBF trace record. It contains additional information about the IVTCSM REQUEST=GET_BUFFER macroinstruction.

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Byte (hex)**

**Contents**

00–03
Record ID: C"GTB2"

04–05
0

06
Trace record flag:

B'0... ....'
Indicates that this is the last record for this event.

B'1... ....'
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

B'.1. ....'
Indicates that the buffer list is in 64-bit storage.

07
Trace record number to correlate all the entries for this particular event

08
Buffer type
GTB3 entry for GET_BUFFER requests

**Entry:**
GTB3

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=GET_BUFFER

**VIT processing module:**
ISTITCCS or ISTIT6CS

This record is written for a CSM buffer allocated as a result of the IVTCSM REQUEST=GET_BUFFER macroinstruction. This trace record corresponds to an entry in the buffer list returned by the IVTCSM macroinstruction.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;GTB3&quot;</td>
</tr>
</tbody>
</table>

**GTB3 entry for GET_BUFFER requests**

**Entry:**
GTB3

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=GET_BUFFER

**VIT processing module:**
ISTITCCS or ISTIT6CS

This record is written for a CSM buffer allocated as a result of the IVTCSM REQUEST=GET_BUFFER macroinstruction. This trace record corresponds to an entry in the buffer list returned by the IVTCSM macroinstruction.

**GTB3 entry for GET_BUFFER requests**

**Entry:**
GTB3

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=GET_BUFFER

**VIT processing module:**
ISTITCCS or ISTIT6CS

This record is written for a CSM buffer allocated as a result of the IVTCSM REQUEST=GET_BUFFER macroinstruction. This trace record corresponds to an entry in the buffer list returned by the IVTCSM macroinstruction.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;GTB3&quot;</td>
</tr>
</tbody>
</table>

**GTB3 entry for GET_BUFFER requests**

**Entry:**
GTB3

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=GET_BUFFER

**VIT processing module:**
ISTITCCS or ISTIT6CS

This record is written for a CSM buffer allocated as a result of the IVTCSM REQUEST=GET_BUFFER macroinstruction. This trace record corresponds to an entry in the buffer list returned by the IVTCSM macroinstruction.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;GTB3&quot;</td>
</tr>
</tbody>
</table>

**GTB3 entry for GET_BUFFER requests**

**Entry:**
GTB3

**VIT option:**
CSM

**Event:**
IVTCSM REQUEST=GET_BUFFER

**VIT processing module:**
ISTITCCS or ISTIT6CS

This record is written for a CSM buffer allocated as a result of the IVTCSM REQUEST=GET_BUFFER macroinstruction. This trace record corresponds to an entry in the buffer list returned by the IVTCSM macroinstruction.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;GTB3&quot;</td>
</tr>
</tbody>
</table>
04–05
0

06
Trace record flag:

B'0... ....'
Indicates that this is the last record for this event.

B'1... ....'
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

B'..1. ....'
Indicates that the buffer list is in 64-bit storage.

07
Trace record number to correlate all the entries for this particular event

08–13
Buffer token

14–17
ALET or first word of the 64-bit address of data being copied

18–1B
31-bit address of data being copied or second word of the 64-bit address of data being copied

1C–1F
Buffer size

GTB6 entry for GET_BUFFER requests

Entry:
GTB6

VIT option:
CSM

Event:
IVTCSM REQUEST=GET_BUFFER

VIT processing module:
ISTITCCS

This trace record is a continuation of the GTBF trace record. It contains the 64-bit buffer list entry address. The GTB6 entry is present only when the GET_BUFFER request failed with the error return code in 64-bit addressing mode.

<table>
<thead>
<tr>
<th>00-03 Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03 Record ID: C'GTB6'</td>
<td></td>
</tr>
</tbody>
</table>

04-05
0

06
Trace record flags:

B'0... ....'
Indicates that this is the last record for this event.
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.

Indicates that the buffer list is in 64-bit storage.

Trace record number to correlate all the entries for this particular event.

64-bit address of the last buffer list entry.

HCLK entry for HPR clock event

Entry: HCLK
VIT option: HPR
Event: HPR clock event
VIT processing module: ISTITCHP
Control is returned to: ISTRPCTM

This trace record is written when the clock state changes (approximately every second). This trace record is also written when the clock mode changes.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;HCLK&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Old HPR clock state</td>
</tr>
</tbody>
</table>
P  Pending stop
R  Running
S  Stopped

07
New HPR clock state
P  Pending stop
R  Running
S  Stopped

08–0F
Time stamp of HCLK entry

10–13
Total number of 25 millisecond timers currently on the HPR clock

14–17
Number of liveness timers on the HPR clock

18–1B
Number of timers on the late queue

1C–1F
Request parameter header (RPH) address

HCL2 entry for HPR clock event (Part 2)

Entry:
HCL2

VIT option:
HPR

Event:
HPR clock event

VIT processing module:
ISTITCHP

Control is returned to:
ISTRPCTM

This trace record is a continuation of the HCLK entry.

| HCL2 | 0 | 0 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 0 | 0 | 0 | 0 | A | B | 0 | 0 | 0 | 0 | D | E | F |
|      |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

Byte (hex)

Contents

VTAM internal trace (VIT) record descriptions  413
Record ID: C'HCL2'

06  Current HPR clock rate
    H    High mode
    S    Standard mode

07  Next HPR clock rate
    H    High mode
    S    Standard mode

08-0B  Total number of 1 millisecond timers currently on the HPR clock

0C-0F  Number of one-hop EE liveness timers (Liveness2) on the HPR clock

10-17  0

18-1B  Number of HPRPST path switch timers on the HPR clock

1C-1F  Number of new route path switch timers on the HPR clock

**HCQ entry for invoking a RoCE HCQ operation (Part 1)**

**Entry:**
HCQ

**VIT option:**
CIA

**Event:**
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCA Command Queue (HCQ) operation with a RoCE Express2 feature, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**
ISTITCSH

This trace record is written upon completion of an HCQ operation.
HCQ ADDRESS RPH

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C‘HCQ’</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.</td>
</tr>
<tr>
<td>05</td>
<td>HCQ operation ending status</td>
</tr>
<tr>
<td>06–07</td>
<td>Module identifier of the module that issued the INTRACE command</td>
</tr>
<tr>
<td>08–09</td>
<td>Return code</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Reason code</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Operation code modifier</td>
</tr>
<tr>
<td>10–17</td>
<td>Address of the PFCTE</td>
</tr>
<tr>
<td>18–1B</td>
<td>Hardware handle</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

**HCQ2 entry for invoking a RoCE HCQ operation (Part 2)**

**Entry:**
- HCQ2

**VIT option:**
- CIA

**Event:**
- Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCA Command Queue (HCQ) operation with a RoCE Express2 feature, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**
- ISTITCSH

This trace record is a continuation of the HCQ entry.
HCQ2 RETRIES
INPUT MAILBOX COUNT
OUTPUT MAILBOX COUNT
HCQ ELEMENT ADDRESS
0

<table>
<thead>
<tr>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>1 1 1 1</th>
<th>1 1 1 1</th>
<th>1 1 1 1</th>
<th>1 1 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 2 3</td>
<td>4 5 6 7</td>
<td>8 9 A B</td>
<td>C D E F</td>
<td>0 1 2 3</td>
<td>4 5 6 7</td>
<td>8 9 A B</td>
<td>C D E F</td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
Contents
00–03
Record ID: C'HCQ2'
04–07
Command retry counter
08–0B
Number of input mailboxes
0C–0F
Number of output mailboxes
10–17
Address of the HCA Command Queue Element
18–1F
0

HCQ3 entry for invoking a RoCE HCQ operation (Part 3)

Entry:
HCQ3
VIT option:
CIA
Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCA Command Queue (HCQ) operation with a RoCE Express2 feature, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTITCSH
This trace record is a continuation of the HCQ entry.

<table>
<thead>
<tr>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>1 1 1 1</th>
<th>1 1 1 1</th>
<th>1 1 1 1</th>
<th>1 1 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 2 3</td>
<td>4 5 6 7</td>
<td>8 9 A B</td>
<td>C D E F</td>
<td>0 1 2 3</td>
<td>4 5 6 7</td>
<td>8 9 A B</td>
<td>C D E F</td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
Contents
00–03
Record ID: C'HCQ3'
04–07
Length of command input
08–0F
Address of first input mailbox or zero if command input length is 16 or fewer bytes
The first 16 bytes of command input

**HCQ4 entry for invoking a RoCE HCQ operation (Part 4)**

**Entry:**
HCQ4

**VIT option:**
CIA

**Event:**
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCA Command Queue (HCQ) operation with a RoCE Express2 feature, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**
ISTITCSH

This trace record is a continuation of the HCQ entry, and is generated only when the HCQ operation requires input mailboxes. Multiple HCQ4 entries can be generated, depending on the length of the command input data.

| 00–03 | Record ID: C'HCQ4'
| 04–1F | Command input data

**Byte (hex)**

**Contents**

| 00–03 | Record ID: C'HCQ4'
| 04–1F | Command input data

| Byte (hex) |
| Contents |

| 00–03 | Record ID: C'HCQ4'
| 04–1F | Command input data

**HCQ5 entry for invoking a RoCE HCQ operation (Part 5)**

**Entry:**
HCQ5

**VIT option:**
CIA

**Event:**
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCA Command Queue (HCQ) operation with a RoCE Express2 feature, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**
ISTITCSH

This trace record is a continuation of the HCQ entry.

| Byte (hex) |
| Contents |

| Byte (hex) |
| Contents |

| 00–03 | Record ID: C'HCQ5'
| 04–1F | Command input data

**VTAM internal trace (VIT) record descriptions 417**
HCQ6 entry for invoking a RoCE HCQ operation (Part 6)

Entry: HCQ6
VIT option: CIA
Event: Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCA Command Queue (HCQ) operation with a RoCE Express2 feature, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module: ISTITCSH

This trace record is a continuation of the HCQ entry, and is generated only when the HCQ operation requires output mailboxes. Multiple HCQ6 entries can be generated, depending on the length of the command output data.

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 1 2 3 4 5 6 7 8 9 A B C D E F 0 1 2 3 4 5 6 7 8 9 A B C D E F
HCQ6 28 Bytes of Output Mailbox
```

HCR entry for invoking a RoCE HCR operation (Part 1)

Entry: HCR
VIT option: CIA
Event: Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCR operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing when the IBM 10 GbE RoCE Express feature operates in a dedicated environment.

VIT processing module: ISTITCSH
This trace record is written upon completion of an HCR operation.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C’HCR’</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.</td>
</tr>
<tr>
<td>05</td>
<td>Operation code</td>
</tr>
<tr>
<td>06–07</td>
<td>Module identifier of the module that issued the INTRACE command</td>
</tr>
<tr>
<td>08–09</td>
<td>Return code</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Reason code</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Operation code modifier</td>
</tr>
<tr>
<td>10–17</td>
<td>Address of the PFCTE</td>
</tr>
<tr>
<td>18–1B</td>
<td>Hardware handle</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

**HCR2 entry for invoking a RoCE HCR operation (Part 2)**

**Entry:**
- HCR2

**VIT option:**
- CIA

**Event:**
- Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCR operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**
- ISTITCSH

This trace record is a continuation of the HCR entry.
HCR3 entry for invoking a RoCE HCR operation (part 3)

Entry:
HCR3

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCR operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTITCSH

This trace record is a continuation of the HCR entry.
Byte (hex) Contents
00–03 Record ID: C’HCR3’
04–07 Command retry counter
08–0B Control operation data
0C–0F Control operation data
10–13 Control operation data
14–17 Control operation data
18–1B Control operation data
1C–1F Control operation data

HCR4 entry for invoking a RoCE HCR operation (part 4)

Entry: HCR4

VIT option: CIA

Event: Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCR operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module: ISTITCSH

This trace record is a continuation of the HCR entry, and is generated only when the HCR operation requires command input data. Multiple HCR4 entries can be generated, depending on the length of the command input data.
### Byte (hex)

<table>
<thead>
<tr>
<th>contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
</tr>
<tr>
<td>Record ID: C'HCR4'</td>
</tr>
<tr>
<td>04-1F</td>
</tr>
<tr>
<td>Command input data</td>
</tr>
</tbody>
</table>

**HCR5 entry for invoking a RoCE HCR operation (part 5)**

**Entry:**

HCR5

**VIT option:**

CIA

**Event:**

Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) HCR operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**

ISTITCSH

This trace record is a continuation of the HCR entry, and is generated only when the HCR operation returns command output data. Multiple HCR5 entries can be generated, depending on the length of the command output data returned.

<table>
<thead>
<tr>
<th>contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
</tr>
<tr>
<td>Record ID: C'HCR5'</td>
</tr>
<tr>
<td>04-1F</td>
</tr>
<tr>
<td>Command output data</td>
</tr>
</tbody>
</table>

---

### HIOx entry for Halt I/O

**Entry:**

HIOB, HIOD, HIOH, HIOI, HIOT, or HIOX
VIT option:
CIO

Event:
Halt I/O

VIT processing module:
ISTRACCI

Control is returned to:
ISTTSCLR

This trace record is written when a Halt I/O SVC is issued to end a currently executing channel program.

- HIOB is generated when an HIO SVC ends a channel program using a BSC line attached to the communication adapter.
- HIOD is generated when an HIO SVC ends a channel program for an OSA-Express QDIO or HiperSockets connection.
- HIOH is generated when an HIO SVC ends a channel program using an SDLC line attached to the communication adapter.
- HIOI is generated when a VM GENIO HALT or an MVS HIO SVC ends a channel program to a local SNA controller as a result of VARY INACT,FORCE processing.
- HIOT is generated when a Halt I/O SVC ends a channel program for a TCP/IP DLC connection.
- HIOX is generated when an HIO SVC ends communication between channel-to-channel-attached hosts.

An HIO entry should be followed shortly by an interrupt (INT) entry. Match the CUAs in the HIO and INT entries to be sure they are for the same device. See z/OS Communications Server: SNA Data Areas Volume 1 for a description of the NCB, ICNCB, and XCNCB fields.

For record types with suffix I, X, or T, the CIO events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIOB</td>
<td>HIOH</td>
<td>HIIO</td>
<td>HIOT</td>
<td>HIOX</td>
<td>I</td>
<td>D</td>
<td>S</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>E</td>
<td>M</td>
<td>O</td>
<td>D</td>
<td>I</td>
</tr>
<tr>
<td>CUA</td>
<td>DEVICE</td>
<td>NCB ADDRESS</td>
<td>FLAG BYTES</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00–03**

Record ID:

- C"HIOB"
- C"HIOD"
- C"HIOH"
- C"HIIO"
- C"HIOT"
- C"HIOX"

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

For HIOB, link station state (BSCLKFSM)
For HIOD, adapter state (DINCB_AFSM)
For HIOH, link station state (HALLFSM)
For HIOI, link state (NCBLNKST)
For HIOX, for a channel-to-channel adapter, station state (XCNSSFSM)
For HIOX, for a LAN channel station, link state (NCBLNKST)
For HIOT, link state (NCBLNKST)

06–07
Module ID (last 2 characters of TSC module name)

08–0B
Channel device name in EBCDIC (either a device address or device number)

0C–0F
NCB address

10–13
Flag bytes (NCBFLAGS)

14–1D
0

1E
For HIOB, HIOD, HIOH, HIOI, and HIOX: 0
For HIOT:
• For the CDLC DLC: C
• For the LCS DLC: L
• For the CLAW DLC: W
• For the CTC DLC: X
• For the HYPERchannel DLC: H

1F
0

**HLST entry for subtrace tree**

**Entry:**

\[HLST\]

**VIT option:**

SSCP

**Subtrace type:**

TREE

**Event:**

APPN route computation

**VIT processing module:**

ISTITCAB

**Control is returned to:**

ISTTRQWC

This record is generated when:

• Subtrace TREE under SSCP trace option is active.
• A candidate hop is processed by TRS in routing tree build or update processing.
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;HLST&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace entry instance</td>
</tr>
<tr>
<td>07</td>
<td>TG number for the current hop</td>
</tr>
<tr>
<td>08</td>
<td>TG weight</td>
</tr>
<tr>
<td>09</td>
<td>Node weight of the current hop</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Total path weight from the tree root to the current node</td>
</tr>
<tr>
<td>0C–13</td>
<td>CP name of the destination node on current hop</td>
</tr>
<tr>
<td>14–1B</td>
<td>CP name of the origin node on current hop</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

**HLS2 entry for subtrace tree (Part 2)**

**Entry:**
- HLS2

**VIT option:**
- SSCP

**Subtrace type:**
- TREE

**Event:**
- APPN route computation

**VIT processing module:**
- ISTITCAB

**Control is returned to:**
- ISTTRQWC
This record is a continuation of the HLST trace record. HLS2 is generated only if any of the CP names in the HLST record has a different network ID than the local node.

<table>
<thead>
<tr>
<th>0 0 0 0 0 4 0 0 0 0 0 0 0 0</th>
<th>0 1 2 3 4 6 7 8 9 A B</th>
<th>0 0 0 1 2 1 3 1 4 5 6 7 8 9 A B</th>
<th>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLS2</td>
<td>0</td>
<td>CURRENT NODE NETWORK ID</td>
<td>PARENT NODE NETWORK ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03  
Record ID: C"HLS2"

04–0B  
0

0C–13  
Network ID of the destination node on current hop

14–1B  
Network ID of the origin node on current hop

1C–1F  
0

**HPR entry for HPRCTL macroinstruction (Part 1)**

**Entry:**

HPR

**VIT option:**

HPR

**Event:**

Invocation of HPRCTL macroinstruction

**VIT processing module:**

ISTITCHP

**Control is returned to:**

Module invoking the HPRCTL macroinstruction

This trace record is written when the HPRCTL macroinstruction is issued. It indicates the type of lookup being performed, the return code for that operation, and the addresses of the parameters involved.
Byte (hex)

Contents

00–03
Record ID: C"HPR"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Entry type:
F
    FID5
L
    LU
M
    MRSU
P
    PMI
R
    RTP
S
    RSREC
T
    RTREC

07
Function:
A
    Add
C
    Change
D
    Delete
F
    Find
P
    Purge
S
    Associate
08  
Return code:

00  
Success

04  
Not found

08  
Insufficient storage

0C  
Length value not valid

10  
Function not supported

14  
Duplicate FID5

18  
Token not valid

09–0D  
Name of module issuing HPRCTL

0E  
HPRCTL instance within issuing module

0F  

10–13  
• If entry type is FID5, BSB address
• If entry type is LU, session count
• If entry type is PMI and function is Add or Find, CPNRB address
• If entry type is RSREC, RSREC address
• If entry type is RTP, RTP address
• If entry type is RTREC, RTREC address

14–1B  
• If entry type is FID5, FID5 address
• If entry type is LU, not applicable
• If entry type is MRSU, the MRSU address is located in bytes X'14' and X'15'
• If entry type is PMI, not applicable
• If entry type is RSREC, element address and element index
• If entry type is RTP, APPN CoS name. It is not written, however, for some cases of FIND FIRST and FIND NEXT.
• If entry type is RTREC, element address and element index

1C–1F  
Request parameter header (RPH) address

**HPR2 entry for HPRCTL macroinstruction (Part 2)**

Entry:

HPR2

VIT option:

HPR
Event:
Invocation of HPRCTL macroinstruction

VIT processing module:
ISTITCHP

This trace record is a continuation of the HPR entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;HPR2&quot;</td>
</tr>
<tr>
<td>04</td>
<td>0</td>
</tr>
<tr>
<td>05–06</td>
<td>HPRCTL flags:</td>
</tr>
<tr>
<td></td>
<td>Byte 1</td>
</tr>
<tr>
<td></td>
<td>Bit</td>
</tr>
<tr>
<td></td>
<td>Meaning</td>
</tr>
<tr>
<td>xx..</td>
<td>BSB lock level requested (entry type FID5):</td>
</tr>
<tr>
<td>00</td>
<td>No lock</td>
</tr>
<tr>
<td>01</td>
<td>Shared</td>
</tr>
<tr>
<td>10</td>
<td>Exclusive</td>
</tr>
<tr>
<td>..1.</td>
<td>Decrement LU session count (entry type LU)</td>
</tr>
<tr>
<td>0</td>
<td>FIND FIRST (entry type RTP or PMI)</td>
</tr>
<tr>
<td>...1.</td>
<td>Increment LU session count (entry type LU)</td>
</tr>
<tr>
<td>0</td>
<td>FIND NEXT (entry type RTP or PMI)</td>
</tr>
<tr>
<td>.... 1...</td>
<td>LU entry was deleted (entry type LU)</td>
</tr>
<tr>
<td>.... .1..</td>
<td>An unused LU entry was found and deleted (entry type LU)</td>
</tr>
<tr>
<td>.... ..1..</td>
<td>Expired entry request (entry type RSREC)</td>
</tr>
<tr>
<td>.... ...1.</td>
<td>APPN boundary function when RTP ends</td>
</tr>
<tr>
<td>Byte 2</td>
<td></td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) record descriptions 429
Bit
Meaning
xxx. ....

HPRCTL entry type

000
FID5

001
LU

010
RTP

011
RSREC

100
MRSU

101
PMI

110
RTREC

...x xxxx
Available

07–17
• If entry type is PMI, RSREC, or RTREC, CP name
• If entry type is RTP and HPRCTL flag is FIND FIRST or FIND NEXT, CP name

Byte
Contents

08–0F
Local NCE

10–17
0
• If entry type is LU, LU name
• If entry type is FID5 and function is associate (see HPR entry byte 07)

Byte
Contents

08–0B
RTP address

0C–17
0

18–1F
• If entry type is RSREC, PCID
• If entry type is RTREC, TCID
• If entry type is LU or RTP, remote NCE
• If entry type is FID5, not applicable
• If entry type is PMI, not applicable

HPR3 entry for HPRCTL macroinstruction (Part 3)

Entry:
HPR3
VIT option:
HPR

Event:
Invocation of HPRCTL macroinstruction

VIT processing module:
ISTITCHP

This trace record is a continuation of the HPR entry when entry type (byte 06) is RTP. It is not written, however, for some cases of FIND FIRST and FIND NEXT.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| HPR3 | 28 BYTES OF RSCV |

Byte (hex) Contents

00–03
Record ID: C"HPR3"

04–1F
28 bytes of RSCV

HPR4 entry for HPRCTL macroinstruction (Part 4)

Entry:
HPR4

VIT option:
HPR

Event:
Invocation of HPRCTL macroinstruction

VIT processing module:
ISTITCHP

This trace record is a continuation of the HPR entry and is written when the following conditions are met:

- The HPR entry type is RTP (HPR trace record byte 6 contains R)
- CP name and FIND FIRST or FIND NEXT is specified (HPR2 trace record byte 5 has either bit 2 or bit 3 on)
- NetID or CP name (HPR2 trace record bytes 07-14) contains at least one wildcard variable
- HPRCTL is successful (HPR trace record byte 08 contains 00)

The fully qualified CP name in this record matches the CP name specified with a wildcard value in the HPR2 record.

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| H P R | 4 | A 17-byte fully qualified CP name | Zeros |

Byte (hex) Contents
HPRT entry for HPR timer control macroinstruction

Entry:
    HPRT

VIT option:
    HPR

Event:
    Invocation of HPR timer control macroinstruction

VIT processing module:
    ISTITCHP

Control is returned to:
    Module invoking the HPRTIMER macroinstruction

This trace record is written when the HPR timer control macroinstruction is issued. It is used to set and cancel the liveness, burst, short request, and refifo timers associated with RTP processing.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;HPRT&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Timer type:</td>
</tr>
<tr>
<td>B</td>
<td>Burst</td>
</tr>
<tr>
<td>L</td>
<td>Liveness</td>
</tr>
<tr>
<td>N</td>
<td>New route</td>
</tr>
<tr>
<td>P</td>
<td>Path switch</td>
</tr>
</tbody>
</table>
R
Refifo
S
Short request

07
Function:
C
Cancel
S
Set

08–0B
Requested timer duration
• Burst, refifo, and short request timers, in milliseconds
• Liveness, new route, and path switch timers, in seconds

0C–0F
RPNCB address

10–13
Timer block address

14–15
Timer block flags:
Byte 1

Bit
Meaning
xxx. ....
Timer request
000. ....
Liveness timer
001. ....
Short request timer
010. ....
Burst timer
011. ....
REFIFO timer
100. ....
HPRPST path switch timer
101. ....
New route for path switch timer

...1 ....
Timer block is on a CLK slot.
.... 1...
Timer block is on the late queue.
.... .1..
Timer block is marked cancel.
.... ..1.
Timer block is on the live queue.
.... ...1
TPPOST of waiting RPH required for cancel function.

Byte 2
Bit
    Meaning
  1... ....
    Timer block is on HPR timer block queue.
  .1... ....
    Timer block is on the HPRPST queue.
  ..1. ....
    Timer block is on the new route queue.
  ...x xxxx
    Available.

16
    Instance of the trace in the issuing module.

17–1B
    Name of module issuing HPRTIMER.

1C–1F
    Request parameter header (RPH) address.

ICR entry for a control register operation

Entry:
    ICR

VIT option:
    CIA

Event:
    Internal shared memory (ISM) control register operation

This trace record is written when an ISM control register operation is performed to manage an ISM interface as part of Shared Memory Communications - Direct Memory Access (SMC-D) processing.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'ICR'</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06–07</td>
<td>Two-character identifier of the module that issued the control register operation</td>
</tr>
<tr>
<td>08–09</td>
<td>Return code</td>
</tr>
</tbody>
</table>
ICR2 entry for a control register operation (part 2)

Entry:
  ICR2

VIT option:
  CIA

Event:
  Internal shared memory (ISM) control register operation

This trace record is a continuation of the ICR record, and is generated only when the ISM control register operation requires command input data. Multiple ICR2 entries might be generated, depending on the length of the command input data.

<p>| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |</p>
<table>
<thead>
<tr>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>A B C D E F</th>
</tr>
</thead>
<tbody>
<tr>
<td>I C R 2</td>
<td>28 BYTES OF COMMAND INPUT DATA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

00–03
  Record ID: C’ICR2’

04–1F
  Command input data

ICR3 entry for a control register operation (part 3)

Entry:
  ICR3

VIT option:
  CIA

Event:
  Internal shared memory (ISM) control register operation

This trace record is a continuation of the ICR record, and is generated only when the ISM control register operation produces command output data. Multiple ICR3 entries might be generated, depending on the length of the command output data.

<p>| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |</p>
<table>
<thead>
<tr>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>A B C D E F</th>
</tr>
</thead>
<tbody>
<tr>
<td>I C R 3</td>
<td>28 BYTES OF COMMAND OUTPUT DATA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) record descriptions 435
IDXx entry for identification exchange

Entry: IDXI or IDXO
VIT option: CIA
Event: Identification Exchange (Part 1)
VIT processing module: ISTRACCI
Control is returned to: ISTLLCXI or ISTLLCXR

This trace record contains information about inbound or outbound identification exchange processing.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03 | Record ID: C'IDXI' for inbound IDX  
C'IDXO' for outbound IDX |
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'. |
| 05–07 | 0 |
| 08–0B | Channel Unit Address |
| 0C–0F | NCB address |
| 10–17 | Block header |
| 18–1B | 0 |
| 1C–1F | Request parameter header (RPH) address |
IDX2 entry for identification exchange (Part 2)

Entry:
IDX2

VIT option:
CIA

Event:
Identification Exchange (Part 2)

VIT processing module:
ISTRACCI

This trace record is a continuation of the IDXx entry.

| 00–03 | Record ID: C"IDX2"
| 04–0F | 28 bytes of additional IDX data

INOP entry for the INOPINFO macro

Entry:
INOP

VIT option:
None (Generated by VTAM)

Event:
Inoperative state

VIT processing module:
ISTRACOT

Control is returned to:
Module encountering inoperative

This trace record shows the invocation of the INOPINFO macro. It captures various INOP information related to a specific device.

This event is treated as an exception condition and is always traced, whether the CIO option is in effect or not.

For Local SNA Controller devices, TCP/IP DLC devices, or Channel-to-Channel devices, this INOP event is also captured within the NCB (pointed to by NCBCIOMV).
Byte (hex)

Contents

01–03
Record ID: C"INOP"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Reason code

07
Type field

X'41'
Connect INOP

X'42'
Signaling connection INOP

X'43'
Device INOP

X'44'
Soft INOP

X'45'
Hard INOP

X'46'
SAP INOP

08–0B
Channel device name in EBCDIC (either a CUA or device number) or blanks

0C–0F
NCB address

10–13
Flags (NCBFLAGS)

14
Product identifier. If the product identifier is S or U, then the product is VTAM. If it is V, the product is
Common Storage Manager.

15–19
Module name that detected INOP

1A–1B
Sense code

1C
Station state

1D–1F
0
INTx entry for channel interrupt

Entry:
   INTD, INTI, INTL, INTT, or INTX

VIT option:
   CIO

Event:
   Channel interrupt

VIT processing module:
   ISTRACCI

Control is returned to:
   ISTTSCLC

This trace record is written when a channel program interrupt occurs.

• INTD is generated for interrupts from OSA-Express QDIO or HiperSockets adapters.
• INTI is generated for interrupts between communication controllers and local SNA cluster controllers.
• INTL is generated for interrupts from local non-SNA cluster controllers.
• INTT is generated for interrupts between TCP/IP DLC connections.
• INTX is generated for interrupts between channel-to-channel-attached hosts.

Every INT entry has a previously issued SIO entry, although it may have been issued some time ago. Use the CUA device field to correlate SIO and INT entries. The SIO entry provides additional information about this completing channel program.

See for a description of the NCB fields.

For record types with suffix D, I, X, or T, the CIO events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>C&quot;INTD&quot; for DINCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;INTI&quot; for ICNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;INTL&quot; for LDNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;INTT&quot; for RWNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;INTX&quot; for XCNCB</td>
</tr>
</tbody>
</table>

| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'. |

| 05         | For INTD, adapter state (DINCB_AFSM) |
|           | For INTI, INTL, and INTT, link station state (NCBLNKST) |
|           | For INTX, station state (XCNSSFSM) |
06
For INTI, the ending operation code; otherwise, 0.

07
DLC type. For INTD, INTI, INTL, and INTX: 0
For INTT:
• For the CDLC DLC: C
• For the LCS DLC: L
• For the CLAW DLC: W
• For the CTC DLC: X
• For the HYPERchannel DLC: H

08–0B
Channel device name in EBCDIC (either a device address or device number)

0C–0F
NCB address

10–13
Flag bytes (NCBFLAGS)

14
Flag byte as follows:

Bit    Meaning
1.... Running in disabled interrupt exit (IOSDIE is not 0)
.1.... ERP work area address exists (IOSERP is not 0)
..1.. Exception condition (IOSEX flag is on)
...1 Error routine in control (IOSERR flag is on)

15
I/O completion code (IOSCOD)

16–17
Sense data in IOSSNS if this was a sense channel program; otherwise, 0

18–1F
Channel status word from IOSB

INT2 entry for channel interrupt (Part 2) for INTx

Entry:
INT2

VIT option:
CIO

Event:
Channel interrupt (Part 2)

VIT processing module:
ISTRACCI

This trace record is a continuation of the INTx entry.
<table>
<thead>
<tr>
<th>Byte (hex) Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
</tr>
<tr>
<td>Record ID: C&quot;INT2&quot;</td>
</tr>
<tr>
<td>04–05</td>
</tr>
<tr>
<td>CPNCB_Index_Word</td>
</tr>
<tr>
<td>06–07</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>08–17</td>
</tr>
<tr>
<td>First 16 bytes of transmit buffer, or 0, or device-dependent data</td>
</tr>
<tr>
<td>18–1F</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

### IO1 entry for TPIO request (Part 1)

**Entry:**
- IO1

**VIT option:**
- API

**Event:**
- Application program request (Part 1)

**VIT processing module:**
- ISTRACAP

**Control is returned to:**
- ISTAPCRS or ISTOCCSM

This trace record shows API requests. It also provides the RPL information supplied by a VTAM application program when it issues an RPL-based macroinstruction (such as SEND or OPNDST). The API routine ISTAICIR issues the TPIO SVC. ISTAPCTI gets control from the SVC, does validity checking, and queues the RPL to the PST request PAB. After the PST request PAB is dispatched, ISTAPCRS gets control and writes the entry, or calls ISTOCCSM to write the entry.

See z/OS Communications Server: SNA Programming for an explanation of the RPL fields.

The process that is started with this event will be completed by a post (UP), or an exit (RE or UE). To correlate the two entries, match the RPL addresses.
Byte (hex)  
Contents  
00–02  
  Record ID: C"IO1"

03  
  RPL request type (RPLREQ)  
    X'15'  
      SETLOGON  
    X'16'  
      SIMLOGON  
    X'17'  
      OPNDST  
    X'19'  
      CHANGE  
    X'1A'  
      INQUIRE  
    X'1B'  
      INTRPRET  
    X'1F'  
      CLSDST  
    X'22'  
      SEND  
    X'23'  
      RECEIVE  
    X'24'  
      RESETSR  
    X'25'  
      SESSIONC  
    X'27'  
      SENDCMD  
    X'28'  
      RCVCMD  
    X'29'  
      REQSESS  
    X'2A'  
      OPNSEC  
    X'2C'  
      TERMSESS

04  
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05  
  Exit definition (RPLEXTDS)

06  
  Option code byte 4 (RPLOPT4)

07  
  Option code byte 1 (RPLOPT1)

08–0B  
  RPL address
0C–0F
   NIB address or CID
10
   Third byte of RH (RPLRH3)
11
   Send/receive type (RPLSRTYP)
12
   VTAM flags (RPLVTFL1)
13
   Post/respond flags (RPLVTFL2)
14
   RU chain position (RPLCHN)
15–17
   RU control codes (RPLCNTRL)
18–1B
   VTAM options (RPLOPTC2)
1C–1F
   VTAM options (RPLOPTC3)

IO2 entry for TPIO request (Part 2)

Entry:
   IO2
VIT option:
   API
Event:
   Application program request (Part 2)
VIT processing module:
   ISTRACAP

This trace record is a continuation of the IO1 entry. It shows additional information about the request.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;IO2&quot;</td>
</tr>
<tr>
<td>03</td>
<td>0</td>
</tr>
<tr>
<td>04–07</td>
<td>Address of the issuer of the macro</td>
</tr>
</tbody>
</table>
| 08–0F      | For SIMLOGON, OPNDST, CLSDST, or REQSESS: Logon mode name (NIBLMODE) or 0 if the NIB is not available
           | For OPNSEC or TERMSESS: 0 |
10–17
Application LU name if available
18–1F
Partner LU name if available

**IO3 entry for TPIO request (Part 3)**

**Entry:**
IO3

**VIT option:**
API

**Event:**
Application program request (Part 3)

**VIT processing module:**
ISTRACAP

This trace record is a continuation of the IO2 entry. It shows additional information about the request.

<table>
<thead>
<tr>
<th>00–02</th>
<th>Record ID: C&quot;IO3&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>03</td>
<td></td>
</tr>
<tr>
<td>04–07</td>
<td>For SIMLOGON, CLSDST, or REQSESS: Address of user data (RPLAREA)</td>
</tr>
<tr>
<td></td>
<td>For OPNDST or OPNSEC: Address of BIND (NIBNDAR) or 0 if the NIB is not available</td>
</tr>
<tr>
<td>08–0F</td>
<td>Target LU network identifier (when for CLSDST PASS)</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

<table>
<thead>
<tr>
<th>00–02</th>
<th>Record ID: C&quot;IO3&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>03</td>
<td></td>
</tr>
</tbody>
</table>

| 04–07 | For SIMLOGON, CLSDST, or REQSESS: Address of user data (RPLAREA) |
|       | For OPNDST or OPNSEC: Address of BIND (NIBNDAR) or 0 if the NIB is not available |

| 08–0F | Target LU network identifier (when for CLSDST PASS) |

<table>
<thead>
<tr>
<th>08–09</th>
<th>Count of node initialization blocks (NIBs) in NIB list (when not for CLSDST PASS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0A–0B</td>
<td>Count of node initialization blocks (NIBs) in NIB list with NIBRPARM=0 (when not for CLSDST PASS)</td>
</tr>
</tbody>
</table>

| 0C–0F | 0 |

<table>
<thead>
<tr>
<th>10–17</th>
<th>Target logical unit name, if available, or 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>18–1F</td>
<td>Partner LU network identifier, if available</td>
</tr>
</tbody>
</table>
### IOSP entry for invoking a Peripheral Component Interconnect Express (PCIe) service (Part 1)

**Entry:**
IOSP

**VIT option:**
CIA

**Event:**
Invocation of a Peripheral Component Interconnect Express (PCIe) service, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing, or as part of Shared Memory Communications - Direct Memory Access (SMC-D) processing.

**VIT processing module:**
ISTITCSH

This trace record is written upon completion of a PCIe service.

<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</th>
<th>0 1 2 3 4 5 6 7 8 9 A B C D E F 0 1 2 3 4 5 6 7 8 9 A B C D E F</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOSP</td>
<td>ASID</td>
</tr>
<tr>
<td>COMMAND</td>
<td>MODULE</td>
</tr>
<tr>
<td>ASSOCIATED PARMLIST ADDR</td>
<td>PFCTE ADDRESS or SLNCB ADDRESS</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00–03**
Record ID: C"IOSP"

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
PCIe service identifier:
- 'A' for allocate service (IQP4ALL)
- 'C' for connect service (IQP4CON)
- 'D' for deallocate service (IQP4DEA)
- 'G' for get attribute service (IQP4GDI)
- 'L' for close service (IQP4CLO)
- 'M' for deregistration service (IQP4DMR)
- 'O' for open service (IQP4OPN)
- 'P' for get PFID attribute service (IQP4GPI)
- 'Q' for query system characteristics (IQP4QSC)
- 'R' for registration service (IQP4RMR)
- 'S' for search service (IQP4SRC)
Module identifier of the module that issued the INTRACE command

Input parameter list that is associated with the PCIe service

Address of the PFCTE or SLNCB

Return code

Reason code

Request parameter header (RPH) address

**IOS2 entry for invoking a Peripheral Component Interconnect Express (PCIe) service (Part 2)**

**Entry:**

IOS2

**VIT option:**

CIA

**Event:**

Invocation of a Peripheral Component Interconnect Express (PCIe) service, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing, or as part of Shared Memory Communications - Direct Memory Access (SMC-D) processing.

**VIT processing module:**

ISTITCSH

This trace record is a continuation of the IOSP record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03</td>
<td>Record ID: C'IOS2'</td>
</tr>
<tr>
<td>04-07</td>
<td>The Peripheral Component Interconnect Express (PCIe) function ID (PFID)</td>
</tr>
<tr>
<td>08-0F</td>
<td>Connection token that is associated with the PCIe service</td>
</tr>
<tr>
<td>10-1F</td>
<td>Operation handle that is associated with the PCIe service</td>
</tr>
</tbody>
</table>

**IOS3 entry for invoking a Peripheral Component Interconnect Express (PCIe) service (Part 3)**

**Entry:**

IOS3
VIT option: 
CIA

Event:  
Invocation of a Peripheral Component Interconnect Express (PCIe) service, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing, or as part of Shared Memory Communications - Direct Memory Access (SMC-D) processing.

VIT processing module:  
ISTITCSH

This trace record is a continuation of the IOSP trace record when the record represents a register (IQP4RMR) command.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'IOS3'</td>
</tr>
<tr>
<td>04</td>
<td>Storage key</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>Address of the memory region</td>
</tr>
<tr>
<td>10–17</td>
<td>Length of the memory region</td>
</tr>
<tr>
<td>18–1F</td>
<td>DMA address to be registered, or 0</td>
</tr>
</tbody>
</table>

**IPAD entry for Enterprise Extender Address Management**

Entry:  
IPAD

VIT option:  
TCP

Event:  
Enterprise Extender Address Management

VIT processing module:  
ISTRACIP

Control is returned to:  
The module that issued the INTRACE macroinstruction

This trace record is written when the AUNCB add, remove, find, find first, or find next function is performed.
Byte (hex)
Content
00–03
Record ID: C'IPAD'
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
0
06
Function:
A
Add
F
Find
N
Find next
R
Remove
S
Scan
T
Find first
07
Return code
X'00'
Function succeeded
X'08'
Function failed
X'OC'
Function not supported
08
Index 1
09
Index 2
0A
Local SAP value
0B
Remote SAP value
0C–0F
IPv4 address or 0. If 0, IPv6 address is reported in IPA2 trace record.
IPA2 entry for Enterprise Extender Address Management (Part 2)

Entry:
IPA2

VIT option:
TCP

Event:
Enterprise Extender Address Management

VIT processing module:
ISTRACIP

This trace record is a continuation of the IPAD entry. This entry is present only when the Enterprise Extender Address Management function was performed for an IPv6 address.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
<th>IPv6 ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C’IPA2’</td>
<td></td>
</tr>
<tr>
<td>04–0F</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10–1F</td>
<td>IPv6 address</td>
<td></td>
</tr>
</tbody>
</table>

IPGN entry for host name

Entry:
IPGN

VIT option:
TCP

Event:
Host name converted to an IP address or IP address converted to a host name

VIT processing module:
ISTRACIP

Control is returned to:
The module that issued the INTRACE macroinstruction

The IPGN record is discarded whenever an Enterprise Extender host name is converted to an IP address (either IPv4 or IPv6 address family). This record is generated whenever a host name is converted to an IP address for the origin or destination of an EE connection.
**Byte (hex)**

**Contents**

**00–03**

Record ID: C"IPGN"

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

0

**06 – 07**

IP hostname length

**08 – 0B**

IPv4 address or 0. If 0, IPv6 address is reported in IPG3 trace record.

**0C – 0F**

0

**10 – 13**

Return address of the module that tried to convert the host name to an IP address.

**14–17**

Return code

0

Success

1

Host not found

2

Try again

3

No recovery

4

No data or no address

**18–1B**

Identifier for the module that tried to convert the host name

**1C–1F**

RPH address

**IPG2 entry for host name (Part 2)**

**Entry:**

IPG2

**VIT option:**

TCP

**Event:**

Host name converted to an IP address or IP address converted to a host name

**VIT processing module:**

ISTRACIP
This trace record is a continuation of the IPGN entry. This entry may be repeated multiple times to display the entire host name.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IPG2&quot;</td>
</tr>
<tr>
<td>04 – 1F</td>
<td>IP host name</td>
</tr>
</tbody>
</table>

**IPG3 entry for host name (Part 3)**

**Entry:**
- IPG3

**VIT option:**
- TCP

**Event:**
- Host name converted to an IP address or IP address converted to a host name

**VIT processing module:**
- ISTRACIP

This trace record is a continuation of the IPGN entry. This entry appears after the last IPG2 record, and is present only when the hostname resolution returns an IPv6 address.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'IPG3'</td>
</tr>
<tr>
<td>04 – 0F</td>
<td>0</td>
</tr>
<tr>
<td>10 – 1F</td>
<td>IPv6 address</td>
</tr>
</tbody>
</table>

**IPLE entry for an internal shared memory (ISM) polling operation**

**Entry:**
- IPLE

**VIT option:**
- CIA
Event:
Invocation of internal shared memory (ISM) event queue polling

This trace record is written when VTAM polls the ISM interface event queue and passes the information to the owning TCP/IP stack as part of Shared Memory Communications - Direct Memory Access (SMC-D) processing.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1 1 1 1</td>
<td>1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td>1 1 1 1 1 1 1 1 1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
Contents
00–03
Record ID: C'IPLE'
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05–06
Last written index on the ISM event queue
07
Number of events reported to the TCP/IP stack
08–09
Return code
0A–0B
Reason code
0C–0F
Address of the SLNCB control block that represents the ISM interface
10–18
Address of the parameter list (PList) provided to the TCP/IP stack
19–1B
Return address of the module issuing the poll request
1C–1F
Request parameter header (RPH) address

**IPLA entry for an internal shared memory (ISM) polling operation (part 2)**

Entry:
IPLA

VIT option:
CIA

Event:
Invocation of internal shared memory (ISM) event queue polling

This trace record is a continuation of the IPLE record. Multiple IPLA entries can be generated, one for each array entry that contains data at the completion of the PollEQ operation. A single Poll operation can have up to 64 array entries with data.
### Byte (hex)

**Contents**

**00–03**
- Record ID: C'IPLA'

**04–07**
- Event Type:
  - 0: DMB Event
  - 1: GID Event
  - 2: Software Requested Event

**08–0B**
- Event code:
  - 1: GID in error state
  - 2: Owning function in error state
  - 3: Using function in error state
  - 4: DMB was unregistered
  - 5: VLAN mismatch with owner
  - 6: VLAN mismatch with user
  - 7: GID disabled
  - 8: Using function disabled

**0C–0D**
- 0

**0E–0F**
- Event Queue Element (EQE) number

**10–17**
- Event Token

**18–1F**
- Event Debug Information

<table>
<thead>
<tr>
<th>I P L A</th>
<th>EVENT TYPE</th>
<th>EVENT CODE</th>
<th>0</th>
<th>EVENT NUMBER</th>
<th>EVENT TOKEN</th>
<th>EVENT DEBUG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
IPOG entry for getibmopt call

Entry:
   IPOG

VIT option:
   TCP

Event:
   getibmopt is called

VIT processing module:
   ISTRACIP

Control is returned to:
   The module that issued INTRACE macroinstruction

This trace record is written when the getibmopt TCP/IP API function is called.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IPOG&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06–07</td>
<td>Number of TCP/IP jobs being reported</td>
</tr>
<tr>
<td>08–0F</td>
<td>TCP/IP job name</td>
</tr>
<tr>
<td>10–11</td>
<td>Status of the job (stack)</td>
</tr>
<tr>
<td>12–13</td>
<td>Version of the job (stack)</td>
</tr>
<tr>
<td>04–17</td>
<td>Return code (ERRNO)</td>
</tr>
<tr>
<td>18–1B</td>
<td>0</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

IPO2 entry for getibmopt call (Part 2)

Entry:
   IPO2
VIT option: TCP

Event: getibmopt is called (Part 2)

VIT processing module: ISTRACIP

This trace record is a continuation of the IPOG entry. It contains information about the additional TCP/IP jobs.

<table>
<thead>
<tr>
<th>Byte (hex) Content</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IPO2&quot;</td>
</tr>
<tr>
<td>04–0B</td>
<td>TCP/IP job name 1</td>
</tr>
<tr>
<td>0C–0D</td>
<td>Status of the job 1</td>
</tr>
<tr>
<td>0E–0F</td>
<td>Version of the job 1</td>
</tr>
<tr>
<td>10–17</td>
<td>TCP/IP job name 2</td>
</tr>
<tr>
<td>18–19</td>
<td>Status of the job 2</td>
</tr>
<tr>
<td>1A–1B</td>
<td>Version of the job 2</td>
</tr>
</tbody>
</table>

### IPTC entry for altering an Enterprise Extender timer FSM state

Entry:
- IPTC

VIT option: TCP

Event: Alteration of a timer FSM state

VIT processing module: ISTRACIP

Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is written when the current state of an Enterprise Extender timer changes.
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IPTC&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Old Enterprise Extender clock state</td>
</tr>
<tr>
<td>P</td>
<td>Pending stop</td>
</tr>
<tr>
<td>R</td>
<td>Running</td>
</tr>
<tr>
<td>S</td>
<td>Stopped</td>
</tr>
<tr>
<td>07</td>
<td>New Enterprise Extender clock state</td>
</tr>
<tr>
<td>P</td>
<td>Pending stop</td>
</tr>
<tr>
<td>R</td>
<td>Running</td>
</tr>
<tr>
<td>S</td>
<td>Stopped</td>
</tr>
<tr>
<td>08–0F</td>
<td>Time stamp of IPTC entry</td>
</tr>
<tr>
<td>10–13</td>
<td>Total number of timers currently on the Enterprise Extender clock</td>
</tr>
<tr>
<td>14–17</td>
<td>Number of liveness timers on the Enterprise Extender clock</td>
</tr>
<tr>
<td>18–1B</td>
<td>Number of timers on the late queue</td>
</tr>
<tr>
<td>1C–1D</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

**IPTM entry for Enterprise Extender timer operation**
VIT option: TCP

Event:
Timer set or canceled

VIT processing module: ISTRACIP

Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is written when an Enterprise Extender timer is set or canceled.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IPTM&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Timer type</td>
</tr>
<tr>
<td>T1</td>
<td>Short request</td>
</tr>
<tr>
<td>TL</td>
<td>Liveness</td>
</tr>
<tr>
<td>07</td>
<td>Function</td>
</tr>
<tr>
<td>S</td>
<td>Set</td>
</tr>
<tr>
<td>C</td>
<td>Cancel</td>
</tr>
<tr>
<td>08–0B</td>
<td>Request timer duration (seconds)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>NCB address</td>
</tr>
<tr>
<td>10–13</td>
<td>Timer block address</td>
</tr>
<tr>
<td>14</td>
<td>Timer block flags:</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>xx...</td>
<td>Timer type.</td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) record descriptions 457
Timer block is on a CLK slot.

Timer block on IP timer block queue.

Timer block is on the late queue.

Timer block is marked cancel.

Available.

15–16
0
17–1B
Name of module setting or canceling the timer
1C–1F
Request parameter header (RPH) address

**IRBD entry for IRB dispatch**

**Entry:**
IRBD

**VIT option:**
PSS

**Event:**
IRB dispatch

**VIT processing module:**
ISTRACPS

**Control is returned to:**
ISTAPCIE

This trace record provides information when an interrupt request block (IRB) is dispatched. See *z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures* to determine what document contains a map of the TCB.

**Note:** For this trace record to be created, in addition to specifying the PSS option you must specify IRB or BOTH on the PSSTRACE start option.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IRBD&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
</tbody>
</table>
**IRBX entry for IRB exit**

**Entry:**
IRBX

**VIT option:**
PSS

**Event:**
IRB exit

**VIT processing module:**
ISTRACPS

**Control is returned to:**
ISTAPCSD

This trace record provides information about an IRB exit.

**Note:** For this trace record to be created, in addition to specifying the PSS option you must specify IRB or BOTH on the PSSTRACE start option.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IRBX&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>PST address</td>
</tr>
<tr>
<td>0C–0F</td>
<td>New TCB address</td>
</tr>
</tbody>
</table>
ISPx entry for invoking an internal shared memory (ISM) Verb (part 1)

Entry:
ISPx

VIT option:
CIA

Event:
Invocation of an internal shared memory (ISM) verb as part of Shared Memory Communications - Direct Memory Access (SMC-D) processing.

VIT processing module:
ISTITCSH

Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is written when an ISM verb is invoked or when an ISM verb invocation is completed.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C'ISPI' for ISM parameter list information prior to the ISM verb invocation</td>
</tr>
<tr>
<td></td>
<td>• C'ISPO' for ISM parameter list information after the ISM verb invocation</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Instance identifier within the module</td>
</tr>
<tr>
<td>06–07</td>
<td>Length of the parameter list (PList) that is used for the ISM verb invocation</td>
</tr>
<tr>
<td>08–0B</td>
<td>A related control block for this particular ISM verb</td>
</tr>
</tbody>
</table>
ISP2 entry for invoking an internal shared memory (ISM) Verb (part 2)

Entry:
ISP2

VIT option:
CIA

Event:
Invocation of an internal shared memory (ISM) verb as part of Shared Memory Communications - Direct Memory Access (SMC-D) processing when the input area in the ISM verb parameter list (PList) has non-zero information.

VIT processing module:
ISTITCSH

This trace record is continuation of the ISPI and ISPO entry. Multiple ISP2 entries may be generated, depending on the length of the input area in the PList that is used for the ISM verb.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'ISP2'</td>
</tr>
<tr>
<td>04–1F</td>
<td>28 bytes of the input area in the PList</td>
</tr>
</tbody>
</table>

ISP3 entry for invoking an internal shared memory (ISM) Verb (part 3)

Entry:
ISP3

VIT option:
CIA

Event:
Invocation of an internal shared memory (ISM) verb as part of Shared Memory Communications - Direct Memory Access (SMC-D) processing when the output area in the ISM verb parameter list (PList) has non-zero information.
VIT processing module:
ISTITCSH

This trace record is continuation of the ISPO entry. Multiple ISP3 entries may be generated, depending on the length of the output area in the PList used for the ISM verb.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C’ISP3’</td>
</tr>
<tr>
<td>04–1F</td>
<td>28 bytes of the output area in the PList</td>
</tr>
</tbody>
</table>

IUTx entry for IUT processing (Part 1)

Entry:
- IUTC, IUTD, IUTF, IUTI, IUTM, IUTQ, IUTR, IUTS, or IUTX or IUT6 (for 64-bit trace addressing)

VIT option:
- CIA

Event:
- IUT processing (Part 1)

VIT processing module:
- ISTRACOU

Control is returned to:
- Module invoking the INTRACE macroinstruction that caused the record to be produced.

This trace record is written when an IUT process occurs. All CM-to-CM traffic is traced. On other sessions, only signaling traffic is traced.

IUTC, IUTI, IUTM, IUTQ, and IUTS mapping and field descriptions

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• C”IUTC“ for CONFIRM</td>
</tr>
<tr>
<td></td>
<td>• C”IUTI“ for INDICATE</td>
</tr>
</tbody>
</table>

Byte (hex): C”IUTC“ for CONFIRM
C”IUTI“ for INDICATE

IUTx entry for IUT processing (Part 1)
• C"IUTM" for IMMEDIATE
• C"IUTQ" for REQUEST
• C"IUTS" for RESPONSE

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
Primitive type and category
Bit
  Primitive Type
  B'0000 ....'
    REQUEST
  B'0001'
    IMMEDIATE
  B'1100 ....'
    CONFIRM
  B'0100 ....'
    INDICATE
  B'1000 ....'
    RESPONSE

Bit
  Category
  B'.... 0001'
    Connection Manager
  B'.... 0011'
    Data Manager

07
Primitive Function
• When category in byte 06 is Connection Manager:
  X'0D'
    Activate SAP
  X'0E'
    Deactivate SAP
  X'10'
    Call Setup
  X'11'
    Call Clear
  X'12'
    Call Connect
  X'14'
    Call Status
  X'16'
    Enable IC
  X'18'
    Disable IC
  X'1A'
    Call Control
• When category in byte 06 is Data Manager:

`X'60'`
  DMAct SAP

`X'63'`
  Message Unit Data

`X'67'`
  Message Data

**08–0B**
Provider ID (TOKEN)

**0C–0F**
If REQ or IND, address of PLIST list or 0. If CNF or RSP, status. See *z/OS Communications Server: IP and SNA Codes* for more information about DLC status codes.

**10–13**
Transaction ID

**14–17**
PList (Parameter List) address

**18–1B**
Return address of caller

**1C–1F**
Request parameter header (RPH) address

---

**IUTD mapping and field descriptions**

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IUTD&quot; for OSA-Express QDIO or HiperSockets</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–06</td>
<td>0</td>
</tr>
<tr>
<td>07</td>
<td>Priority</td>
</tr>
<tr>
<td>08–0B</td>
<td>Provider ID (TOKEN)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Status. See the <em>z/OS Communications Server: IP and SNA Codes</em> for more information about DLC status codes.</td>
</tr>
</tbody>
</table>
IUTF mapping and field descriptions

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IUTF&quot; for free PList</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Selector (see IUT2 record for selector values)</td>
</tr>
<tr>
<td>07</td>
<td>Protocol (see IUT2 record for protocol values)</td>
</tr>
<tr>
<td>08–0B</td>
<td>Provider ID (TOKEN)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Exit address</td>
</tr>
<tr>
<td>10–13</td>
<td>Container address (TIPAC)</td>
</tr>
<tr>
<td>14–17</td>
<td>PList (parameter list) address</td>
</tr>
<tr>
<td>18–1B</td>
<td>Caller of utility routine (caller of ISTTSCUD)</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>
## IUTR mapping and field descriptions

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IUTR&quot; for IUTIL-R call</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>X'00'</td>
</tr>
<tr>
<td>06–07</td>
<td>The low-order 2 bytes of the IUTIL Status field</td>
</tr>
<tr>
<td>08–0B</td>
<td>Provider ID (TOKEN)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>RUNCB address</td>
</tr>
<tr>
<td>10–17</td>
<td>Container address (PSAC)</td>
</tr>
<tr>
<td>18–1B</td>
<td>Caller of utility routine</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

## IUTX mapping and field descriptions

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>DQA ADDRESS or 0 or SMRQA ADDRESS</td>
</tr>
<tr>
<td>04</td>
<td>NCB ADDRESS</td>
</tr>
<tr>
<td>05</td>
<td>RETURN ADDRESS</td>
</tr>
<tr>
<td>06</td>
<td>ELEMENT COUNT</td>
</tr>
<tr>
<td>07</td>
<td>CALLED EXIT ADDRESS</td>
</tr>
<tr>
<td>08</td>
<td>THRAD ADDRESS or RPH ADDRESS</td>
</tr>
</tbody>
</table>
**Byte (hex)**

**Contents**

**00–03**
- Record ID: C"IUTX" for exit call

**04**
- ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
- DLC type:
  - C'Q' for QDIO
  - C'S' for Shared Memory Communications over Remote Direct Memory Access (SMC-R) and Shared Memory Communications - Direct Memory Access (SMC-D)

**06–07**
- Function:
  - C'WC' for write completion exit
  - C'RC' for read completion exit
  - C'SC' for SMC-D data completion exit
  - C'EC' for SMC-D event completion exit

**08–0B**
- Queue Data address that is associated with the event
  - If DLC type is 'Q', this field is a DQA address
  - If DLC type is 'S' and Function is 'WC', this field is a SMRQA address
  - If DLC type is 'S' and Function is 'SC' or 'EC', this field is a SMLQA address
  - Otherwise, this field is 0.

**0C–0F**
- NCB address that is associated with IUTIL processing:
  - If DLC type is 'Q', this is a DINCB address.
  - If DLC type is 'S' and Function is 'WC' or 'RC', this is an RUNCB address.
  - Otherwise, this is an SLNCB address.

**10–13**
- INTRACE invoker

**14–17**
- Number of elements on queue

**18–1B**
- Exit address

**1C–1F**
- Thread address or request parameter header (RPH) address:
  - If DLC type is 'Q', the thread value or 0
  - If DLC type is 'S', the RPH address
IUT6 mapping and field descriptions
IUT6 mapping and its detailed field descriptions, including different byte and contents are defined in this topic.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00 - 03    | Record ID: C"IUT6"
| 04 - 07    | 0        |
| 08 - 0F    | Container address or SBA address
  - If this record is immediately preceded by an IUTD record, this is a SPAC address.
  - If this record is immediately preceded by an IUTX record where DLC Type is 'S' and Function is 'WC' or 'RC', this is a PLAC address.
  - If this record is immediately preceded by an IUTX record where DLC Type is 'S' and Function is 'SC' or 'EC', this is an SBA address.
| 10 - 1F    | 0        |

IUT2 entry for IUT processing (Part 2)

Entry:
  IUT2

VIT option:
  CIA

Event:
  IUT process (Part 2)

VIT processing module:
  ISTRACOU

This trace record is a continuation of the IUTx trace record. It will not follow IUTD, IUTF, or IUTX records.
00–03
Record ID: C"IUT2"

04
Selector
X'00'
  RM
X'01'
  PTP
X'02'
  ATM
X'03'
  Samehost
X'81'
  CDLC
X'82'
  CLAW
X'83'
  CTC
X'84'
  LCS
X'85'
  Hyperchannel

05
Protocol
X'01'
  APPN
X'03'
  TCP/IP
X'04'
  UDP SAP support
X'05'
  X25
X'07'
  FastUdp
X'7E'
  Connection Manager

06–07
Interpreter ID (EBCDIC data)
ID
  Interpreter
SP
  IUTLLCSP
IO
  IUTLLCIO
II
  IUTLLCII
IS
  ISTALCIS
IUT3 entry for IUT processing (Part 3)

Entry:
  IUT3

VIT option:
  CIA

Event:
  IUT process (Part 3)

VIT processing module:
  ISTRACOU

Control is returned to:
  The module that issued the INTRACE macroinstruction

This trace record is a continuation of the IUTx trace record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;IUT3&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Length of data</td>
</tr>
<tr>
<td>08</td>
<td>Version of CSM buffer descriptor</td>
</tr>
</tbody>
</table>
LCSx entry for LAN channel station error (Part 1)

Entry:
LCSL, LCSP, LCSS, or LCSX

VIT option:
LCS

Event:
LAN channel station error (Part 1)

VIT processing module:
ISTRACLS

Control is returned to:
The module that issued the INTRACE

If the LCS trace option is specified, an LCSX trace record is created for every data frame VTAM receives from or sends to an IBM 3172 Interconnect Nways Controller.

If the VIT is active and VTAM receives a frame that is not valid from an IBM 3172 Interconnect Nways Controller, an LCSL, LCSP, or LCSS trace record is created depending on the following error conditions.

• The LCSL trace record is created when VTAM receives link-related data that is not valid.
• The LCSP trace record is created when VTAM receives adapter-related data that is not valid.
• The LCSS trace record is created when VTAM receives SAP-related data that is not valid.

Note: These records are created as an exception condition.
Byte (hex)
Contents

00–03
Record ID:
- C"LCSL" for information about a link
- C"LCSP" for information about an adapter
- C"LCSS" for information about an SAP
- C"LCSX" for information about a channel

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Direction
- Code
  Meaning
  C"I"
  Data received inbound
  C"O"
  Data sent outbound

06
Adapter number

07
Adapter type
- 0 = Control communication
- 2 = 802.5 (token-ring)
- 4 = CSMA/CD–802.3
- 5 = 802.4 (token-bus), MAP 3.0
- 6 = FDDI (fiber distributed data interface)

08–0B
Channel device name in EBCDIC (either a device address or device number)

0C–0F
Control block address
- For LCSL, address of LSNCB
- For LCSP, address of PRTCB
- For LCSS, address of SAPCB
- For LCSX, address of XCNCB

10–13
Requesting module abbreviated name

14–17
Data length

18–1B
Data address
FSM state

For LCSL:

00  Nonswitched/switched reset
01  Nonswitched add SAP User
02  Nonswitched delete SAP user
03  Nonswitched pending deallocation
04  Nonswitched link active
05  Nonswitched open station
06  Nonswitched test local
07  Nonswitched test global
08  Nonswitched negotiable
09  Nonswitched primary
0A  Nonswitched connect requested
0B  Nonswitched pending connect indication
0C  Nonswitched station active
0D  Nonswitched close station
81  Switched add SAP user
82  Switched delete SAP user
83  Switched pending deallocation
84  Switched link active
85  Switched inbound answer any
86  Switched inbound poll pending
87  Switched inbound XID pending
88  Switched inbound pending contact
89  Switched inbound open station

VTAM internal trace (VIT) record descriptions 473
8A  Switched outbound answer specific
8B  Switched outbound open station
8C  Switched outbound test local
8D  Switched outbound test global
8E  Switched outbound XID pending
8F  Switched outbound XID3 pending
90  Switched outbound pending contact
91  Switched pending abandon connection
92  Switched pending abandon connection out
93  Switched pending abandon connection, no station
94  Switched pending abandon connection out, no station
95  Switched close station
96  Switched answer off
97  Switched negotiable 1
98  Switched negotiable 2
99  Switched negotiable 3
9A  Switched primary
9B  Switched secondary
9C  Switched connect requested
9D  Switched pending connect indication
9E  Switched station active
9F  Switched inbound XID negotiation-preceding pending
For LCSP:
00  Reset
01  Add channel user pending
02  Adapter enable pending
03  Active
04  Adapter disable pending
05  Delete channel user pending
06  Inoperative
07  Blocked
For LCSS:
00  Reset
01  Add port user pending
02  Add SRM entry pending
03  Activate SAP pending
04  Active
05  Deactivate SAP pending
06  Delete SRM entry pending
07  Delete port user pending
08  Inoperative
09  Blocked
For LCSX:
00  Reset state
01  X-side I/O pending
02  Y-SIDE I/O pending
03  Contact wait
04  Contact wait DCM
05  XID7 1 pending
06  XID7 2 pending
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>07</td>
<td>XID bad pending</td>
</tr>
<tr>
<td>08</td>
<td>XID7 1 I/O pending</td>
</tr>
<tr>
<td>09</td>
<td>XID7 2 I/O pending</td>
</tr>
<tr>
<td>0A</td>
<td>XID bad I/O pending</td>
</tr>
<tr>
<td>0B</td>
<td>Ready</td>
</tr>
<tr>
<td>0C</td>
<td>Disconnect scheduled</td>
</tr>
<tr>
<td>0D</td>
<td>HDV pending</td>
</tr>
<tr>
<td>0E</td>
<td>HDV pending DCM</td>
</tr>
<tr>
<td>0F</td>
<td>Disconnect pending</td>
</tr>
<tr>
<td>1D</td>
<td>Reason code</td>
</tr>
<tr>
<td>00</td>
<td>Successful.</td>
</tr>
<tr>
<td>04</td>
<td>Exit PAB immediately</td>
</tr>
<tr>
<td>08</td>
<td>Entire primitive length is not valid.</td>
</tr>
<tr>
<td>0C</td>
<td>Buffer allocation failed.</td>
</tr>
<tr>
<td>10</td>
<td>Length of primitive header is not valid.</td>
</tr>
<tr>
<td>14</td>
<td>Length of data field is not valid.</td>
</tr>
<tr>
<td>18</td>
<td>Incorrect target layer identifier.</td>
</tr>
<tr>
<td>1C</td>
<td>Incorrect identifier type.</td>
</tr>
<tr>
<td>24</td>
<td>Correlator ID mismatch.</td>
</tr>
<tr>
<td>28</td>
<td>Incorrect parameter list version.</td>
</tr>
<tr>
<td>2C</td>
<td>Primitive was received in wrong state.</td>
</tr>
<tr>
<td>30</td>
<td>Unrecognized primitive code.</td>
</tr>
<tr>
<td>34</td>
<td>Incomplete primitive header.</td>
</tr>
<tr>
<td>38</td>
<td>User_sap_id provided is not valid.</td>
</tr>
</tbody>
</table>
Length of routing vector (if included) is not valid.

XID command/response value is not valid.

Poll/final indicator is not valid.

Unexpected XID type.

Negative confirm received.

User class is not valid.

Flow action value is not valid.

SAP is not net manager capable.

Not valid for net manager SAP.

Vector length field is not valid in adapter-specific overlay.

Length of adapter-specific overlay field is not valid.

Vector identifier field is not valid.

Destination MAC/Destination SAP reporting field is not valid.

Identifier (user_sap_id or user_cep_id) is inconsistent with identifier type.

Not valid for logon type.

Test/XID remote MAC/remote SAP does not match an entry in the SAP user matrix (SUM), and SUM has no "answer any" entries available.

Pacing fields are not 0.

Success count not X'FF'.

user_cep_id value not in SAP user matrix (SUM) range or SUM entry not in use or answer mode not specific for SUM entry.

Adapter is inactive.

No match in adapter routing table found or incorrect LAN type.

XCNCB 8-byte header error.

XCNCB byte count mismatch.

XCNCB bad primitive offset.
RVI

C"N" = route vector not included
C"Y" = route vector included

LCS2 entry for LAN channel station error (Part 2)

Entry:
LCS2
VIT option:
LCS
Event:
LAN channel station error (Part 2)
VIT processing module:
ISTRACLS

This trace record is a continuation of the LCSL, LCSP, LCSS, or LCSX entry.

The VTAM internal trace will generate as many as 31 LCS2 records for the LCSL, LCSP, LCSS, and LCSX entries.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;LCS2&quot;</td>
</tr>
<tr>
<td>04–1F</td>
<td>Up to 28 bytes of channel data</td>
</tr>
</tbody>
</table>

LCSM entry for MACaddress for DATA SEND/RECEIVE

Entry:
LCSM
VIT option:
LCS
Event:
MACaddress for DATA SEND/RECEIVE
VIT processing module:
ISTRACLS

If the LCS trace option is specified, an LCSM trace record is created for every data frame VTAM sends to and receives from an IBM 3172 Interconnect Nways Controller.

The LCSM trace record is created in association with an LCSX entry. However, because these trace entries are created by different component elements, the entries may be separated in the trace by other component traces. For outbound data, the LCSM entry will appear before the LCSX entry. For inbound data, the LCSM entry will appear following the LCSX entry.
Byte (hex)

Contents

00–03
Record ID: C"LCSM" for information about macaddress

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Direction

Code
Meaning
C"I"
Data received inbound
C"O"
Data sent outbound

06–0B
Destination macaddress

0C–0D
LAN Operation

Note: It is not required that users of the product know the meaning of this internal VTAM name. When required, the product support organization may use it to assist in internal flow diagnosis.

0E–13
Source MACaddress

14–17
Index 1

Note: It is not required that users of the product know the meaning of this internal VTAM name. When required, the product support organization may use it to assist in internal flow diagnosis.

18–1B
Index 2

Note: It is not required that users of the product know the meaning of this internal VTAM name. When required, the product support organization may use it to assist in internal flow diagnosis.

1C–1F
Request parameter header (RPH) address

**LDLC entry for invoking LDLC**

**Entry:**
LDLC

**VIT option:**
SSCP
Event:
Invocation of an LDLC module

VIT processing module:
ISTRACSC

Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is issued when an LDLC module is invoked.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;LDLC&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Data length control type: A ATM E Enterprise Extender</td>
</tr>
<tr>
<td>07</td>
<td>Destination</td>
</tr>
<tr>
<td>08</td>
<td>Code</td>
</tr>
<tr>
<td>09–0F</td>
<td>0</td>
</tr>
<tr>
<td>10–13</td>
<td>Return address of the module that invoked the LDLC</td>
</tr>
<tr>
<td>14–17</td>
<td>Work element address</td>
</tr>
<tr>
<td>18–1B</td>
<td>Name of the LDLC module</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Address of the ALCPL or AUCPL</td>
</tr>
</tbody>
</table>
LDL2 entry for invoking LDLC (Part 2)

Entry:
LDL2

VIT option:
SSCP

Event:
Invocation of an LDLC module Part 2

This trace record is a continuation of an LDLC module. It shows the variable data portion of the control block.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;LDL2&quot;</td>
</tr>
<tr>
<td>04–1B</td>
<td>Data</td>
</tr>
<tr>
<td>1C–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

**LKEX entry for TPLOCK exclusive**

Entry:
LKEX

VIT option:
LOCK

Event:
TPLOCK exclusive

VIT processing module:
ISTRACLK

Control is returned to:
ISTAPC36

This trace record identifies a request for exclusive control of a VTAM lock and the status of a lock. Exclusive locks can be held by only one VTAM process. If a WAIT entry immediately follows this entry, it could be because a wait-for-lock condition exists. Look at the lockword in the trace output to see if the lock is already held.

Locks obtained by LKEX are eventually released by UNLK or ULKA. See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for a list of VTAM locks.
Byte (hex)

Contents

00–03
Record ID: C"LKEX"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06–07
Level of lock to be obtained

08–0B
Address of lock to be obtained

0C–0F
CRA lock account word (CRALKACT)

10–13
Address of the issuer of the TPLOCK macro

14–1A
Lockword pointed to by address in bytes 08-0B

1B
The LOCK ID field. See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for a list of VTAM locks.

1C–1F
Request parameter header (RPH) address

LKSH entry for TPLOCK shared

Entry:

LKSH

VIT option:

LOCK

Event:

TPLOCK shared

VIT processing module:

ISTRACLK

Control is returned to:

ISTAPC35

This trace record identifies a request for a shared VTAM lock and the status of a lock. Shared locks can be held by more than one VTAM process at a time. If a WAIT entry immediately follows this entry, it may be because a wait-for-lock condition exists. Look at the lockword in the trace output to see if the lock is already held.
Locks obtained by LKEH are eventually released by UNLK or ULKA. See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for a list of VTAM locks.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;LKSH&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Indicates lock wait status:</td>
</tr>
<tr>
<td>X'00'</td>
<td>WAIT</td>
</tr>
<tr>
<td>X'80'</td>
<td>NOWAIT</td>
</tr>
<tr>
<td>06–07</td>
<td>Level of lock to be obtained</td>
</tr>
<tr>
<td>08–0B</td>
<td>Address of lock to be obtained</td>
</tr>
<tr>
<td>0C–0F</td>
<td>CRA lock account word (CRALKACT)</td>
</tr>
<tr>
<td>10–13</td>
<td>Address of the issuer of the TPLOCK macro</td>
</tr>
<tr>
<td>14–1A</td>
<td>Lockword pointed to by address in bytes 08-0B</td>
</tr>
<tr>
<td>1B</td>
<td>The LOCK ID field. See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for a list of VTAM locks.</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

**LNKx entry**

**Entry:**
- LNKA, LNKI, LNKL, LNKU, or LNKX

**VIT option:**
- CIA

**Event:**
- Change to link use count

**VIT processing module:**
- ISTRACCI
Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is written when the link use count in the node control block (NCB) is incremented or decremented.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;LNK&quot;</td>
</tr>
<tr>
<td>03</td>
<td>NCB type:</td>
</tr>
<tr>
<td></td>
<td>C&quot;LNKA&quot; for AHNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;LNKI&quot; for ICNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;LNKL&quot; for LDNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;LNKR&quot; for RPNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;LNKU&quot; for AUNCB</td>
</tr>
<tr>
<td></td>
<td>C&quot;LNKX&quot; for XCNCB</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Reason for change to use count (DNPFUNC):</td>
</tr>
<tr>
<td></td>
<td>X'01'</td>
</tr>
<tr>
<td></td>
<td>Increment for boundary SLU element address (BLB)</td>
</tr>
<tr>
<td></td>
<td>X'02'</td>
</tr>
<tr>
<td></td>
<td>Increment for boundary PLU element address (BPAT)</td>
</tr>
<tr>
<td></td>
<td>X'03'</td>
</tr>
<tr>
<td></td>
<td>Increment for boundary SNA half-session (BSB)</td>
</tr>
<tr>
<td></td>
<td>X'04'</td>
</tr>
<tr>
<td></td>
<td>Increment for boundary non-SNA half-session (LUST)</td>
</tr>
<tr>
<td></td>
<td>X'05'</td>
</tr>
<tr>
<td></td>
<td>Increment for application half-session (FMCB)</td>
</tr>
<tr>
<td></td>
<td>X'06'</td>
</tr>
<tr>
<td></td>
<td>Increment for rapid transport protocol (RTP) half-session queue</td>
</tr>
<tr>
<td></td>
<td>X'07'</td>
</tr>
<tr>
<td></td>
<td>Increment for rapid-transport protocol (RTP)</td>
</tr>
<tr>
<td></td>
<td>X'F1'</td>
</tr>
<tr>
<td></td>
<td>Increment for boundary SLU element address (BLB)</td>
</tr>
<tr>
<td></td>
<td>X'F2'</td>
</tr>
<tr>
<td></td>
<td>Increment for boundary PLU element address (BPAT)</td>
</tr>
<tr>
<td></td>
<td>X'F3'</td>
</tr>
<tr>
<td></td>
<td>Increment for boundary SNA half-session (BSB)</td>
</tr>
</tbody>
</table>
X'F4'
Decrement for boundary non-SNA half-session (LUST)

X'F5'
Decrement for application half-session (FMCB)

X'F6'
Decrement for line (NCB)

X'F7'
Decrement for rapid-transport protocol (RTP) half-session queue

X'F8'
Decrement for rapid-transport protocol (RTP)

06–07
Element address (for boundary element address reasons)

08–0B
Link use count after the increment or decrement

0C–0F
Node control block (NCB) address

10–13
Address of control block associated with the increment or decrement

14–15
0

16–17
Element index value

18–1B
Module address or address of the issuer of the TSCDN caller

1C–1F
NCB Dependent Data

For LNKU entries:
Represents the number of non-RS (route setup) RTP pipes that originate in this host and traverse this Enterprise Extender connection.

For all other entries:
0

**LOST entry for lost trace record**

**Entry:**
LOST

**VIT option:**
None

**Event:**
Lost trace record

**VIT processing module:**
ISTRACRR

**Control is returned to:**
The location shown in register 14

This trace record indicates that the internal trace table is incomplete. Information from events that were supposed to be traced was lost because there were not enough save areas available for the tracing routine to record the information.
LSNA entry for local SNA record (Part 1)

Entry:
- LSNA

VIT option:
- CIA

Event:
- Read/Write from APPN PU Channel

VIT processing module:
- ISTRACCI

Control is returned to:
- ISTTSC86

This trace record is written each time data is read or written across an APPN host-to-host channel.
Event Flags

Bit	Meaning
1... .... The data unit being traced (FID2 data or control data) is the last data unit in the multipath channel (MPC) transmit block header just received. This bit has meaning only for inbound data.
1... .... This data is control data. If off, this data is normal FID2 data traffic.
..1. .... This trace entry contains the beginning of a new PIU/control data. If off, this data unit is the middle or end portion of a PIU. For outbound data, this bit will always be on (only the first part of PIU/Control data outbound is traced).

A 1-byte field that indicates the direction of the data unit flow:

- Code

  Meaning
  C"I"
  Data received inbound
  C"O"
  Data sent outbound

TSCB address

Length of data being sent or received

This 18-byte field will map as follows:

- For control data
  - The 2-byte QLLC Header (indicating the type of the control data)
  - Up to 16 bytes of the remaining control data
- For FID2 data
  - The 6-byte FID2 TH header
  - A 3-byte RH if it exists
  - Up to 9 bytes of the remaining FID2 data (12 bytes if no RH)

LSN2 entry for local SNA record (Part 2)

Entry:
  LSN2
VIT option:
  CIA
Event:
  Read/Write from APPN Host PU Channel
VIT processing module:
  ISTRACCI

This trace record is a continuation of the LSNA entry. It contains 28 more bytes of data.
**Byte (hex)**

**Contents**

00–03

Record ID: C"LSN2"

04–1F

Up to 28 more bytes of data

---

### MCO1 entry for MIBConnect (Part 1)

**Entry:**

MCO1

**VIT option:**

CMIP

**Event:**

MIBConnect called by CMIP application program

**VIT processing module:**

ISTITCCM

**Control is returned to:**

Module invoking the INTRACE macro that caused the record to be produced

This trace record is generated when a CMIP application program calls the MIBConnect function. The trace record shows the information that is passed from the application program to the MIB controller and from the MIB controller to the application program.

For detailed descriptions of the fields, see *z/OS Communications Server: CMIP Services and Topology Agent Guide*.

If either the MIBConnect or OPEN ACB return code is not 0, this entry is always traced, regardless of the VIT options specified.

---

**Byte (hex)**

**Contents**

00–03

Record ID: C"MCO1"

04

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05

Return code from OPEN ACB
MIBConnect return code

Link identifier

Maximum number of outstanding invoke identifiers

Application name (padded on the right with blanks) or blanks. This field is blanks if the MIBConnect module cannot obtain storage for local variables.

0

Request parameter header (RPH) address

MCO2 entry for MIBConnect (Part 2)

Entry:
MCO2

VIT option:
CMIP

Event:
MIBConnect called by CMIP application program (Part 2)

VIT processing module:
ISTITCCM

This trace record is a continuation of the MCO1 entry and contains additional information that is passed from the application program to the MIB controller and from the MIB controller to the application program.

For detailed descriptions of the fields, see z/OS Communications Server: CMIP Services and Topology Agent Guide.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;MCO2&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>System management application entity (SMAE) name length on input. This field is X'FFFF' if the value is greater than 65 635.</td>
</tr>
<tr>
<td>06–07</td>
<td>System object name length on input. This field is X'FFFF' if the value is greater than 65 635.</td>
</tr>
<tr>
<td>08</td>
<td>API level. This field is X'FF' if the value is greater than 255.</td>
</tr>
<tr>
<td>09</td>
<td>Local identifier field. This field is X'FF' if the value is greater than 255.</td>
</tr>
</tbody>
</table>
Termination exit address or 0. If nonzero, this TPEND exit is driven and this field can be used to correlate with the UE1 and UE2 VIT entries.

Read-queue exit address or 0

User data

Data space name, or blanks (if no data space name is provided)

MDEL entry for MIBSendDeleteRegistration

Entry:
MDEL
VIT option:
CMIP
Event:
MIBSendDeleteRegistration called by CMIP application program
VIT processing module:
ISTITCCM
Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced

This trace record is generated when a CMIP application program calls the MIBSendDeleteRegistration function of the CMIP services API to delete the registration of an object. The trace record shows the information that is passed from the application program to the MIB controller and from the MIB controller to the application program.

For detailed descriptions of the fields, see z/OS Communications Server: CMIP Services and Topology Agent Guide.

If the return code is not 0, this entry is always traced, regardless of the VIT options specified.

| 00–03 | Record ID: C"MDEL"
| 04    | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05    | 0
| 06–07 | MIBSendDeleteRegistration return code

If the return code is not 0, this entry is always traced, regardless of the VIT options specified.
**MDIS entry for MIBDisconnect**

**Entry:**
- MDIS

**VIT option:**
- CMIP

**Event:**
- MIBDisconnect called by CMIP application program

**VIT processing module:**
- ISTITCCM

**Control is returned to:**
- Module invoking the INTRACE macro that caused the record to be produced

This trace record is generated when a CMIP application program calls the MIBDisconnect function of the CMIP services API. The trace record shows the information that is passed from the application program to the MIB controller and from the MIB controller to the application program.

**Note:** If the CMIP application program is internal, such as the VTAM topology agent, the MIBDisconnect function is not traced.

For detailed descriptions of the fields, see [z/OS Communications Server: CMIP Services and Topology Agent Guide](#).

If either the MIBDisconnect or CLOSE ACB return code is not 0, this entry is always traced, regardless of the VIT options specified.

---

**Byte (hex) Contents**

**00–03**
- Record ID: C"MDIS"

**04**
- ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
- Return code from CLOSE ACB
MIBDisconnect return code

Link identifier

0C–1F

0

MMG entry for memory management (Part 1)

Entry:

MMG

VIT option:

None

Event:

ISTORMMG detected an error condition.

VIT processing module:

ISTRACTR

Control is returned to:

ISTORMMG

This entry is written when ISTORMMG detects an overlay in the storage obtain or to be freed queue.

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
</tr>
<tr>
<td>MMG</td>
<td>FUNC</td>
<td>FREEPTR</td>
<td>TO_BE_FREED</td>
<td>HEADER_PTR</td>
<td>The contents of SMHDR pointed by HEADER_PTR for x’10’ bytes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

00–02

Record ID: C"MMG"

03

Function code:

1

Running through the single threaded storage obtained queue. The storage management header has invalid information pointed to by FREEPTR.

2

Running through the local TO_BE_FREED queue. The storage management header pointed to by FREEPTR contains invalid information. FREEPTR was copied from TO_BE_FREED.

3

Running through the local TO_BE_FREED queue. The backward pointer has invalid information in its storage management header.

4

Running through the local TO_BE_FREED queue. The forward pointer has invalid information in its storage management header.

04–07

FREEPTR (Address of the storage to be freed)

08–0B

Local TO_BE_FREED queue pointer
**0C-0F**
HEADER_PTR (The current storage management header address)

**10–1F**
The contents of SMHDR pointed to by HEADER_PTR

**MM2 entry for memory management (Part 2)**

**Entry:**
MM2

**VIT option:**
None

**Event:**
Continuation of MMG trace entry

**VIT processing module:**
ISTRACTR

<table>
<thead>
<tr>
<th>MM2</th>
<th>FUNC</th>
<th>0</th>
<th>FREEPTR</th>
<th>The contents of SMHDR pointed by FREEPTR for x'10' bytes OR ZERO</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>8</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>A</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>C</td>
<td>D</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>E</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>5</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>9</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>B</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>D</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>F</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00–02**
Record ID: C"MM2"

**03**
Function code:

1. Running through the single threaded storage obtained queue. The storage management header has invalid information pointed to by FREEPTR.

2. Running through the local TO_BE_FREED queue. The storage management header pointed to by FREEPTR contains invalid information. FREEPTR was copied from TO_BE_FREED.

3. Running through the local TO_BE_FREED queue. The backward pointer has invalid information in its storage management header.

4. Running through the local TO_BE_FREED queue. The forward pointer has invalid information in its storage management header.

**04–0B**
0

**0C-0F**
FREEPTR (Address of the storage to be freed)

**10–1F**
The contents of SMHDR pointed to by FREEPTR

**MNPS entry for MNPS macro**

**Entry:**
MNPS
VIT option:
CFS

Event:
Multinode persistent session coupling facility access

VIT processing module:
ISTRACCF

Control is returned to:
Module invoking the coupling facility access that caused the record to be produced

This entry is written when VTAM manipulates data in the MNPS coupling facility structure.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"MNPS"
| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05         | 0
| 06         | Function request by this invocation of MNPS:
| X'01'      | Read
| X'02'      | Update
| X'03'      | Delete
| 07         | Type of data being updated:
| X'01'      | Application status block (ASB)
| X'02'      | FMCB
| X'03'      | FMCB extension
| X'04'      | BSB
| X'05'      | Inbound CV29
Outbound CV29

RTP

NLP

LM Table

SAB

HSICB

08–0F
Name of the multinode persistent session application program

10–13
Address of the invoker of the MNPS macro

14–17
Return code from the MNPS macro

18–1B
CFS access token

1C–1F
Request parameter header (RPH) address

MPDU entry for MPC PDU processing

Entry:
MPDU

VIT option:
CIA

Event:
Inbound or outbound PDU over high performance data transfer

VIT processing module:
ISTRACCI

Control is returned to:
ISTTSCDP and IUTLLCMB for inbound processing or ISTTSC88 for outbound processing

The MPDU trace record shows outbound and inbound MPC PDU processing of CSM buffers.

<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0</th>
<th>0 0 0 0 0 0 0 0</th>
<th>0 0 0 0 0 0 0 0</th>
<th>0 0 0 0 0 0 0 0</th>
<th>0 0 0 0 0 0 0 0</th>
<th>0 0 0 0 0 0 0 0</th>
<th>0 0 0 0 0 0 0 0</th>
<th>0 0 0 0 0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPDU D ML TLN F</td>
<td>I ACT PDU SEQ XBUFLST PDU HEADER RPH</td>
<td>POINTER</td>
<td>POINTER</td>
<td>POINTER</td>
<td>POINTER</td>
<td>POINTER</td>
<td>POINTER</td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

00–03
Record ID: C"MPDU"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
<table>
<thead>
<tr>
<th>PDU moved data length</th>
<th>05 06–07</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDU total data length</td>
<td>08–0A</td>
</tr>
<tr>
<td>PDU header flag byte (See ISTPDHDR - PDHFLAGS)</td>
<td>0B</td>
</tr>
<tr>
<td>Protocol ID (See ISTPDHDR - PDHPID)</td>
<td>0C</td>
</tr>
<tr>
<td>PDU route action word</td>
<td>0D–0F</td>
</tr>
<tr>
<td>OSD</td>
<td>Outbound data</td>
</tr>
<tr>
<td>IRS</td>
<td>Outbound data</td>
</tr>
<tr>
<td>IRG</td>
<td>Outbound data</td>
</tr>
<tr>
<td>ICF</td>
<td>Outbound data</td>
</tr>
<tr>
<td>ICA</td>
<td>Outbound data</td>
</tr>
<tr>
<td>ICB</td>
<td>Outbound data</td>
</tr>
<tr>
<td>IEE</td>
<td>Outbound data</td>
</tr>
<tr>
<td>IQH</td>
<td>Outbound data</td>
</tr>
<tr>
<td>IQT</td>
<td>Outbound data</td>
</tr>
<tr>
<td>IQC</td>
<td>Outbound data</td>
</tr>
<tr>
<td>IDP</td>
<td>Outbound data</td>
</tr>
<tr>
<td>ICT</td>
<td>Outbound data</td>
</tr>
<tr>
<td>OEE</td>
<td>Outbound data</td>
</tr>
</tbody>
</table>

| 10–13 | PDU sequence number or X'00000000' for CNLS PDU |
| 14–17 | Address of the first ISTXBFLST |
| 18–1B | Address of the PDU header |
| 1C–1F | Address of VTAM RPH |
**MPD2 entry for MPC PDU processing**

**Entry:**
MPD2

**VIT option:**
CIA

**Event:**
Inbound or outbound PDU over high performance data transfer

**VIT processing module:**
ISTRACCI

This trace record is a continuation of the MPDU trace record. Up to three MPD2 trace records follow an MPDU trace record, each containing 28 bytes of data from the extended buffer list (XBUFLST) associated with this PDU.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;MPD2&quot;</td>
</tr>
<tr>
<td>04–1F</td>
<td>28 bytes of data from the extended buffer list (XBUFLST) associated with this PDU</td>
</tr>
</tbody>
</table>

---

**MQRQ or MQRS entry for MIBSendRequest or MIBSendResponse**

**Entry:**
MQRQ or MQRS

**VIT option:**
CMIP

**Event:**
MIBSendRequest or MIBSendResponse called by CMIP application program

**VIT processing module:**
ISTITCCM

**Control is returned to:**
Module invoking the INTRACE macro that caused the record to be produced

The MQRQ is generated when a CMIP application program calls the MIBSendRequest function of the CMIP services API to send a request. The MQRS is generated when a CMIP application program calls the MIBSendResponse function of the CMIP services API to send a response. The trace record shows the information that is passed from the application program to the MIB controller and from the MIB controller to the application program.

For detailed descriptions of the fields, see z/OS Communications Server: CMIP Services and Topology Agent Guide.

If the return code is not 0, this entry is always traced, regardless of the VIT options specified.

**Note:** Calls to MIBSendCmipRequest and MIBSendCmipResponse functions also generate these trace records, but there is not a one-to-one correspondence.
Byte (hex)  
Contents  
00–03  
Record ID:  
  C"MQRQ" for MIBSendRequest  
  C"MQRS" for MIBSendResponse  
04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.  
05  
06–07  
MIBSendRequest or MIBSendResponse return code  
08–0B  
Link identifier  
0C–0F  
Invoke identifier. The invoke identifier can be used to correlate this VIT entry with the CMIP message string in the buffer trace.  
10–17  
Local identifier (padded on the right with zeros) or 0  
18–1B  
Address of the message  
1C–1F  
Request for parameter header (RPH) address  

MREG entry for MIBSendRegister  

Entry:  
MREG  
VIT option:  
CMIP  
Event:  
MIBSendRegister called by CMIP application program  
VIT processing module:  
ISTITCCM  
Control is returned to:  
Module invoking the INTRACE macro that caused the record to be produced  
This trace record is generated when a CMIP application program calls the MIBSendRegister function of the CMIP services API to register an object. The trace record shows the information that is passed from the application program to the MIB controller and from the MIB controller to the application program.  
For detailed descriptions of the fields, see z/OS Communications Server: CMIP Services and Topology Agent Guide.  
If the return code is not 0, this entry is always traced, regardless of the VIT options specified.
Byte (hex)  
Contents  
00–03  
Record ID: C"MREG"

04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05  
Capability flags. This field is X'FF' if the value is greater than 255.

  Bit  
  Meaning  
  .... ....1.  
  Reserved  
  .... ....1  
  Subtree handler

06–07  
MIBSendRegister return code

08–0B  
Link identifier

0C–0F  
Invoke identifier. The invoke identifier can be used to correlate this VIT entry with the CMIP message string in the buffer trace.

10–17  
Local identifier (padded on the right with zeros)

18  
Name type. This field is X'FF' if the value is greater than 255.

19  
Allomorphs count. This field is X'FF' if the value is greater than 255.

1A  
Create handlers count. This field is X'FF' if the value is greater than 255.

1B  
0

1C–1F  
Request parameter header (RPH) address

**MRG entry for updates to the VTAM topology agent (Part 1)**

**Entry:**  
MRG

**VIT option:**  
CMIP
Event:
Resource updates sent to VTAM topology agent

VIT processing module:
ISTITCCM

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced

This trace record is generated when the VTAM topology agent receives updates from VTAM for changes in VTAM resources.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;MRG&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Object type. The type of resource causing the incoming update.</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Action performed because of the incoming update.</td>
</tr>
</tbody>
</table>

**Bit**
**Meaning**

1... ....
The incoming update is waiting to be merged with older updates before it is sent in a CMIP message.

.1... ....
The incoming update is merged with older updates for resources that own the resource that caused this incoming update.

..1. ....
The incoming update is merged with older updates for the same resource that caused the incoming update.

...1 ....
The resource causing the incoming update has a different vertex 1 from the older, existing update for the same resource. This bit has meaning only when bit 2 is on.

..... 1...
The vertex 1 or line of the older, existing update is the resource that caused the incoming update. This bit has meaning only when bit 3 is on.

..... .1...
An older, existing update is sent before the incoming update because the incoming update cannot be merged with the older update.
Event report or notification

SNAlocaltopology

LUcollection

SNAnetwork

Name of the resource causing the incoming update

Reason for the incoming update

Reason for the older, existing update, if there is one

Change in attribute value caused by the incoming update

Change in attribute value caused by the older, existing update, if there is one

Object flags indicated on the incoming update

Object flags indicated on the older, existing update, if there is one

The number of times resource updates for this particular resource were merged before the incoming update is processed. This number does not include the merges that occurred for resources that own this particular resource.

Workarea flags

Flags contained in the Agent Support Request Mapping control block

Request parameter header (RPH) address

---

**MRG2 entry for updates to the VTAM topology agent (Part 2)**

**Entry:**

MRG2

**VIT option:**

CMIP

**Event:**

Resource updates sent to VTAM topology agent

**VIT processing module:**

ISTITCCM

This trace record is a continuation of the MRG entry.
Byte (hex)  
Contents  
00–03  
Record ID: C"MRG2"  
04–07  
New state of the resource that caused the incoming update  
08–0B  
New state of the resource that caused the incoming update, as indicated on the older, existing update, if there is one  
0C–0F  
Oldest state of the resource causing the incoming update. If there is an older, existing update, this is the old state specified on the update that is waiting. Otherwise, this is the old state of the resource causing the incoming update.  
10–11  
Object count on the incoming update  
12–17  
0  
18–1F  
Value depends on type of update:  
Type of Update  
Value  
Event report or notification  
0  
SNA network  
Name of the vertex 1  
LU collection  
Name of the PU for which LU collection is requested  
SNA local topology  
Name of the vertex 1. If the incoming update is merged with an older, existing update for a resource that owns the resource causing the incoming update, this is the name of the owning resource.

MSG entry for message issued  
Entry:  
MSG  
VIT option:  
MSG
Event:
Message issued (Part 1)

VIT processing module:
ISTRACOT

Control is returned to:
ISTCFCTM

This trace record is written each time a message is issued by a VTAM module. You can use it to find the originator of a message and to see whether the message was solicited or unsolicited.

MSG trace entries are generated for all messages, even those that are suppressed by message-flooding prevention. MSG trace entries thus provide information that may be missing from the operator’s console.

See z/OS Communications Server: SNA Messages for information about the message flooding prevention.
For information about the MODIFY SUPP command, see z/OS Communications Server: SNA Operation. For information about the SUPP start option, see z/OS Communications Server: SNA Resource Definition Reference.

The VTAM Internal Trace (VIT) does not trace the following messages:

- Logon manager messages (beginning with ELM)
- TSO/VTAM messages (beginning with IKT)

---

**Byte (hex)**

| 00–03 | Record ID: C"MSG"
| 04    | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05    | 0
| 06    | ‘S’ if the message was solicited (issued in response to a VTAM command); ‘U’ if the message was unsolicited
| 07–0A | Message ID or ‘USS’. If USS, the message ID appears in the variable text field in the MSG2 entry. If message ID, the 4-digit message number (or 3-digit message number and 1-character type code) appears in this field.
| 0B–0F | Issuing module ID (4th, 5th, 6th, 7th, and 8th characters of the issuing module name). See “Using module names to isolate VTAM problems” on page 23 for information about determining the module name from the module ID.
| 10–13 | Save area address
| 14–17 | Address of the CPMSG macro
• If the high-order bit in the return address field is on, CPMSG has been issued from a utility module. The module name and return address are that of the caller of the utility module.
• If the high-order bit in the return address field is off, CPMSG has not been issued from a utility module. The module name and return address are that of the module that issued the CPMSG.

18–1F
Destination
• If byte 06 is S, this field contains one of the following items:
  SYSTEMxx
  xx is the ID of the system console.
  NLDM
  Message is destined for the session monitor component of the NetView program or NLDM.
  URC
  Message is to be sent to the system console using routing codes, instead of to one console in particular.
  POA
  Message is to be sent to a program operator application (POA) and the POA name is not available yet.
• If byte 06 is U, this field is 0

### MSGS entry for message sent (Part 1)

**Entry:**
- MSGS

**VIT option:**
- MSG

**Event:**
- Message sent (Part 1)

**VIT processing module:**
- ISTRACOT

**Control is returned to:**
- ISTLUCRC or ISTMSCCN

This trace record is written each time VTAM sends a message to its destination. It can be used to correlate the console log with the internal trace. The destination may indicate that the message was sent to a system console (by way of a WTO macro) or to a program operator application, such as the NetView program (by way of an RCVCMD macro).

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>1A</td>
<td>1B</td>
<td>1C</td>
<td>1D</td>
<td>1E</td>
<td>1F</td>
<td>11</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Message</th>
<th>ID</th>
<th>Flags</th>
<th>Destination</th>
<th>Message Header</th>
<th>First 12 Characters of Message Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSGS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03
- Record ID: C"MSGS"

04
- ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
Message suppression indicator

0  Message not suppressed.
1  Message was suppressed.

06–07

0  Destination. If SYSTEMxx, the message was sent to system console xx, by a WTO macro. If an application name, the message was sent to a program operator application, by an RCVCMD macro.

14–1F

First 12 characters of the message text. If the message text is longer than 12 characters, it will continue in the MSG2 entry.

MSG2 entry for message sent (Part 2)

Entry:

MSG2

VIT option:

MSG

Event:

Message issued or message sent (Part 2)

VIT processing module:

ISTRACOT

This trace record is a continuation of the MSG or MSGS trace record. It contains variable data for the MSG entry, or more message text for the MSGS entry.

Up to two MSG2 entries can follow the MSG entry; however, only one MSG2 entry can follow the MSGS entry.

| 00–03 | Record ID: C"MSG2"
| 04–1F | Variable data, or more message text

MT entry for module trace

Entry:

MT
VIT option: SSCP
Event: Module trace detects full data buffer
VIT processing module: ISTRACOT
Control is returned to: ISTITCTR

This trace record is written when the module trace detects a full data buffer.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–01</td>
<td>Record ID: C&quot;MT&quot;</td>
</tr>
<tr>
<td>02–1F</td>
<td>Data</td>
</tr>
</tbody>
</table>

**MU1 entry for LU 6.2 message unit (Part 1)**

Entry: MU1
VIT option: APPC
Event: LU 6.2 message unit (Part 1)
VIT processing module: ISTRACAC
Control is returned to: Module invoking the INTRACE macro that caused the record to be produced

This trace record shows information about an LU 6.2 message unit. It is generated whenever a message unit is queued to another component.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;MU1&quot;</td>
</tr>
</tbody>
</table>
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Message unit ID

X'010101' (END_CONVERSATION)
X'010102' (SESSION_FLOW_RESUMED)
X'010103' (FLUSH_DATA)
X'010104' (RESUME_PS_HS_FLOW)
X'010201' (PS_COPR_FMH5_RCVD)
X'010301' (SEND_EXPEDITED_DATA)
X'010302' (CONFIRMED)
X'010303' (REQUEST_TO_SEND)
X'010304' (SEND_DATA_RECORD)
X'010305' (SEND_ERROR)
X'010306' (SEND_PACING_RSP)
X'010307' (RSP_TO_EXPEDITED_DATA)
X'010501' (ALLOCATE_RCB)
X'010502' (DEALLOCATE_RCB)
X'010503' (GET_SESSION)
X'010504' (UNBIND_PROTOCOL_ERROR)
X'010505' (REJECT_SESSION)
X'010506' (SUSPEND_SESSION)
X'010507' (RESUME_SESSION)
X'010701' (ATTACH_TP)
X'010702' (END_XP)
X'020101' (CNOS_ABORT)
X'020101' (CNOS_CLEANUP)
X'020501' (CNOS_COMPLETE)
X'020502' (CHANGE_SESSIONS)
X'030101' (RECEIVE_EXPEDITED_DATA)
X'030102' (CONFIRMED)
X'030103' (REQUEST_TO_SEND)
X'030104' (RECEIVE_DATA)
X'030105' (RECEIVE_ERROR)
X'030106' (PACING_RSP_RCVD)
X'030107' (RSP_TO_REQUEST_TO_SEND)
X'030108' (INITIAL_PACING_COUNT)
X'030109' (DEALLOCATE_ABEND_REJECTED)
X'030401' (ABORT_HS)
X'030501' (ATTACH_HEADER)
X'030502' (FREE_SESSION)
X'030503' (BID)
X'030504' (BID_RSP)
X'030505' (BIS_RQ)
X'030506' (BIS_REPLY)
X'030507' (RTR_RQ)
X'030508' (RTR_RSP)
X'030509' (SECURITY_HEADER)
X'040101' (RESTOREgetSession)
X'040401' (LOAD_PROFILES)
X'040402' (RSP_LOAD_PROFILES)
X'040403' (DELETE_PROFILES)
X'040501' (SESSION_ACTIVATED)
X'040502' (SESSION_DEACTIVATED)
X'040503' (ACTIVATE_SESSION_RSP)
X'040504' (CTERM_DEACTIVATE_SESSION)
X'040505' (LNS_LRM_FREE_AMU)
08–0B
Message unit address

0C–0F
Half session ID

10–13
Address of correlator value

14
LU 6.2 message unit return code (AMURETCD)

15–16
0

17
Flag byte to indicate that the message unit contains an extended buffer list (bit 7)

18–1B
LU 6.2 message unit flags (AMUFLAGS)

1C–1F
LU 6.2 message unit sense data (AMUSENSE) or 0

**MU2 entry for LU 6.2 message unit (Part 2)**

**Entry:**
MU2

**VIT option:**
APPC

**Event:**
LU 6.2 message unit (Part 2)
VIT processing module:
ISTRACAC

This trace record is a continuation of the MU1 trace record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;MU2&quot;</td>
</tr>
<tr>
<td>04</td>
<td>0</td>
</tr>
<tr>
<td>05–07</td>
<td>Message unit ID shown in MU1 entry</td>
</tr>
<tr>
<td>08–0B</td>
<td>Data address or 0. If the message unit contains an extended buffer list, this field points to the address of the first extended buffer list entry that contains RU data.</td>
</tr>
<tr>
<td>0C–0F</td>
<td>AMUDATLN (data length) or 0. If the message unit contains an extended buffer list, this field contains the length of all RU data represented in that extended buffer list.</td>
</tr>
<tr>
<td>10–13</td>
<td>Resource allocation block address or 0</td>
</tr>
<tr>
<td>14–15</td>
<td>For MU ID X'020502': Session count. For all other MU IDs: 0.</td>
</tr>
<tr>
<td>16–17</td>
<td>For MU ID X'020502': Session limit. For all other MU IDs: 0.</td>
</tr>
<tr>
<td>18–1B</td>
<td>For MU IDs associated with an active conversation (X'010102', X'010201', X'010506', X'030108', X'050107', X'050109', X'050301'): Conversation ID. For all other MU IDs: 0.</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Variable overlay field</td>
</tr>
<tr>
<td>1E–1F</td>
<td>Change in the CNOS value</td>
</tr>
</tbody>
</table>

For all other MU IDs: 0
MU3 entry for LU 6.2 message unit (Part 3)

Entry:
MU3

VIT option:
APPC

Event:
LU 6.2 message unit (Part 3)

VIT processing module:
ISTRACAC

This trace record is a continuation of the MU2 trace record. It is generated for only those message units which contain the following MU IDs:

X'010102'
X'010201'
X'010501'
X'010505'
X'010506'
X'020502'
X'030509'
X'040401'
X'040402'
X'040403'
X'040501'
X'050106'
X'050107'
X'050108'
X'050109'
X'050309'
X'050401'
X'060201'
X'060202'
X'060203'
X'060204'
X'060205'
X'060206'
X'060401'
X'080401'

| 00–03 | Record ID: C“MU3” |
| 04    | 0                  |

510  z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
05–07
Message unit ID shown in the MU1 and MU2 entries

08–0F
Mode name for the following MU IDs:
- X'020502'
- X'040501'
- X'050401'
- X'060202'
- X'060203'

Mode name or 0 for the following MU IDs:
- X'010501'
- X'060201'
- X'060205'
- X'060206'

Session instance identifier for the following MU IDs:
- X'010102'
- X'010506'
- X'010507'
- X'050106'
- X'050107'
- X'050108'
- X'050109'

Session instance identifier or 0 for the following MU ID:
- X'010505'

0 for all other MU IDs

10–17
Local logical unit name

18–1F
Partner logical unit name or 0; for X'060204', 0

**MU4 entry for LU 6.2 message unit (Part 4)**

**Entry:**
- MU4

**VIT option:**
- APPC

**Event:**
- LU 6.2 message unit (Part 4)

**VIT processing module:**
- ISTRACAC

This trace record is a continuation of the MU3 trace record. It is generated for only those message units that contain the following MU IDs:
- X'010102'
- X'010201'
- X'010501'
- X'010505'
- X'010506'
- X'020502'
Byte (hex)

Contents

00–03

Record ID: C"MU4"

04–17

0

18–1F

Partner network identifier or 0; for X'060204', 0.

NIPx entry for IP characteristics list add, delete, or find

Entry:

NIPA, NIPD, NIPI, or NIPN

VIT option:

NRM

Event:

Call to module ISTNRCIP

VIT processing module:

ISTRACNR

Control is returned to:

ISTNRCIP

This trace record contains information about IP characteristics being added, deleted, or found.
Bit Meaning
1111 11.. 0
...... .1.
  LU is an APPL
...... 11
  LU is TN3270 client supporting definite response mode

07
Return code

08–0F
LU name

10–13
Address of the caller of ISTNRCIP

14–1B
Network ID of the LU

1C–1F
RPH Address

NIP2 entry for IP characteristics list add, delete, or find (Part 2)

Entry:
NIP2

VIT option:
NRM

Event:
Call to module ISTNRCIP

VIT processing module:
ISTRACNR
This trace record is a continuation of the NIPx entry.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**IP ADDRESS**

**PORT**

**IPADR ADDRESS**

**RDTE ADDRESS**

**Byte (hex)**

**Contents**

**00–03**

Record ID: C'NIP2'

**04–13**

IP address

**14–15**

Port number

**16–17**

0

**18–1B**

Address of IP address structure (ISTIPADR)

**1C–1F**

RDTE address

---

**NLPx entry for network layer packet (Part 1)**

**Entry:**

NLP

**VIT option:**

HPR

**Event:**

Network layer packet (NLP) sent or received

**VIT processing module:**

ISTITCHP

**Control is returned to:**

The module that issued the INTRACE macro

This trace record is written when a network layer packet (NLP) is sent or received at the DLC layer. The NLPI and NLPO entries provide, respectively, inbound and outbound information about the NLPs, including the FID5 PIU and information about the network layer header (control block ISTNLH) and the rapid transport protocol (RTP) transport header (control block ISTTHDR). The NLP record is 32 bytes in length, with up to 31 additional NLP2 continuation records, each 32 bytes in length.
### Byte (hex) Contents

#### 00–03
Record ID:
- C"NLPI" for inbound
- C"NLPO" for outbound

#### 04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

#### 05
0

#### 06–07
Length of data (TSCILNG)

#### 08–0B
TSCB address

#### 0C–0F
DLC NCB address

#### 10–13
RPNCB address

#### 14–17
Related TSCB address:
- Continuation TSCB for inbound (0 if no continuation TSCB exists).
- Data TSCB for outbound (first TSCB in chain with TSCNLP off, 0 if no data TSCB exists).

#### 18–1B
Name of the module that issued this trace entry

#### 1C–1F
RPH address

### NLP2 entry for network layer packet (Part 2)

**Entry:**
- NLP2

**VIT option:**
- HPR

**Event:**
- Network layer packet (NLP) sent or received

**VIT processing module:**
- ISTITCHP

This trace record is a continuation of the NLP entry.

| 0 | 0 | 0 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| NLP2 | 28 BYTES OF NLP |

### Byte (hex) Contents

#### 00–03
Record ID: C"NLP2"
NRSP entry for negative response to PIU request (Part 1)

Entry:
NRSP

VIT option:
PIU

Event:
Negative response to PIU request (Part 1)

VIT processing module:
ISTRACOT

Control is returned to:
ISTTSCGR

This trace record is written when VTAM generates a negative response to a PIU request. Because this entry is associated with an event failure, it is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;NRSP&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Control block ID</td>
</tr>
<tr>
<td>06</td>
<td>Flag field (TSCFLAG1)</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Address of TSCB</td>
</tr>
<tr>
<td>0C–1F</td>
<td>First 20 bytes of the PIU</td>
</tr>
</tbody>
</table>

NRS2 entry for negative response to PIU request (Part 2)

Entry:
NRS2

VIT option:
PIU
Event:
Negative response to PIU request (Part 2)

VIT processing module:
ISTRACOT

This trace record is a continuation of the NRSP entry.

<table>
<thead>
<tr>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>00–03</td>
<td>24 MORE CHARACTERS OF THE PIU</td>
<td>RETURN ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
Contents

00–03
Record ID: C"NRS2"

04–1B
24 more characters of the PIU

1C–1F
Return address of the issuer

NSD entry for NSIND exit

Entry:
NSD

VIT option:
VCNS

Event:
NSIND exit scheduled

VIT processing module:
ISTRACNS

Control is returned to:
ISTAPCUE

This trace record is written when the NSIND exit routine is scheduled. The exit is scheduled when VTAM receives a VCNSSCMD CONTROL=LOGON request from a VCNS user. This trace record shows information about the exit invocation.

<table>
<thead>
<tr>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
<th>0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>NSD</td>
<td>I</td>
<td>D</td>
<td>0</td>
<td>ACB ADDR</td>
<td>NETWORK ACCESS POINT</td>
<td>EXIT REASON</td>
<td>STORAGE ADDRESS</td>
</tr>
</tbody>
</table>

Byte (hex)
Contents

00–03
Record ID: C"NSD"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

VTAM internal trace (VIT) record descriptions 517
05–07
0
08–0B
Address of the ACB associated with the NSIND exit
0C–13
Symbolic name of the network access point, defined with USER=VCNS in the interconnect major node, representing a line to which the application program is logged on
14–17
Reason the exit was scheduled:
0
As a result of a network management-related event
4
As a result of a termination of the VCNS line
8
As a result of an error detected by VTAM that resulted in the termination of the LOGON request (for example, an abend)
18–1B
If exit reason in bytes 14–17 is 0, address of storage area containing specific error information about the exit event; otherwise, 0.
1C–1F
If bytes 18–1B contain a storage area address, first 4 bytes of specific error information from the storage area.

ODPK entry for OSA-Express QDIO or HiperSockets packets (Part 1)

Entry:
ODPK

VIT option:
CIA

Event:
Inbound or outbound data

VIT processing module:
ISTITCOD

Control is returned to:
ISTLLCHI, ISTLLCIE, ITSLLCWI, IUTLLCIZ

This trace record is written when packets are read from or written to an OSA-Express QDIO or HiperSockets adapter.
Byte (hex)
Contents
00 - 03
Record ID: C"ODPK"
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
Device type:
• If byte 6 is C"R":
  – C"I" for HiperSockets (iQDIO)
  – C"Q" for QDIO
• Else:
  – C"***" when device type is not applicable
06
Direction:
• C"D" for discarded
• C"I" for inbound
• C"O" for outbound
• C"R" for routed (accelerated), outbound
07
Write queue priority/Read queue identifier/0
• if byte 6 is C"D"
  – 0
• if byte 6 is C"I"
  – Read queue identifier
• if byte 6 is C"O" or C"R"
  – Write queue priority
08 - 0F
ShortPAC address. This is a 64-bit address.
10 - 13
XBUFLLST entry address
14
0
15
SBAL index
16
Start SBALE index (within SBAL)
17
End SBALE index (within SBAL)
18 - 19
Total length of packet
1A - 1B
Module identifier for the module that issued the INTRACE.
1C - 1F
Request parameter header (RPH) address.
ODP2 entry for OSA-Express QDIO or HiperSockets packets (Part 2)

Entry:
  ODP2

VIT option:
  CIA

Event:
  Inbound or outbound data

VIT processing module:
  ISTITCOD

This trace record is written after ODPK and may be written after another ODP2 depending on the amount of data to be traced.

Tip: The number of ODP2 records captured for each packet is limited. When running an IO trace, the number of ODP2 records will vary with the length specified on the MODIFY TRACE command. For a data path channel used for capturing OSA-Express network traffic analyzer traces, only one ODP2 record will be captured.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C“ODP2“</td>
</tr>
<tr>
<td>04–1F</td>
<td>If preceded by ODPK, first 28 bytes of packet.</td>
</tr>
<tr>
<td></td>
<td>If preceded by ODP2, next 28 bytes of packet.</td>
</tr>
</tbody>
</table>

ODTE entry for outbound DLC timer events

Entry:
  ODTE

VIT option:
  CIA

Event:
  OSA-Express QDIO, Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing, or HiperSockets timer

VIT processing module:
  ISTITCOD

Control is returned to
  ISTLLCWD, ISTLLCWT, ISTSRIWD, ISTSRIWT

This trace record is written when the OSA-Express QDIO, Shared Memory Communications over Remote Direct Memory Access (SMC-R), or HiperSockets write completion timer is started or restarted.
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ODTE&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
</tbody>
</table>
| 05         | Function:  
• C"D" for dispatch of the timer task  
• C"S" for start timer  
• C"R" for restart timer |
| 06–07      | STIMERM return code  
**Code value**  
**Meaning**  
X'0000'  
STIMERM service completed successfully.  
X'000C'  
Program error: Interval exceeds 2400 hours.  
X'0010'  
Program error: Parameters that are not valid are passed to STIMERM.  
X'001C'  
Program error: STIMERM SET limit for task exceeded.  
X'0024'  
Program error: STIMERM ID was not valid.  
X'0028'  
Program error: Interval plus TOD exceeds maximum clock comparator value. |
| 08–0F      | Current Timer of Day (in TOD format) |
| 10–17      | Interval or timer control information  
• When Function is 'S' or 'R', the timer interval (in TOD format)  
• When Function is 'D', the timer control fields for CDS processing |
| 18–1B      | Module identifier for the module that issued the INTRACE |
ONLP entry for orphaned Network Layer Packet

**Entry:**
- ONLP

**VIT option:**
- HPR

**Event:**
- Orphaned NLP

**VIT processing module:**
- ISTITCHP

**Control is returned to:**
- Module issuing INTRACE

An orphaned NLP is an NLP that appears to have been lost by the DLC.

This trace record is written when an orphaned NLP is detected, recovered, or freed. There are two types of orphaned NLPs, acknowledged and unacknowledged.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"ONLP"
| 04         | ID is the primary address spaced ID (ASID). This field is 0 if the ASID is greater than X'FF'. |
| 05         | 0 |
| 06         | ONLP function: |
| A          | Acknowledged orphaned NLP detected |
| F          | Free acknowledged orphaned NLP from Garbage_SNDs queue |
| R          | Recover unacknowledged orphaned NLP |
| S          | Segment recovered NLP |
| U          | Unacknowledged orphaned NLP detected |
| 07         | Recovery count |
**OON entry for module trace**

**Entry:**
- OON

**VIT option:**
- N/A

**Event:**
- Method has been entered.

**VIT processing module:**
- ISTRACOT

**Control is returned to:**
- Module invoking the INTRACE macro that caused the record to be produced

This trace record is written when a method is entered.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;OON&quot;</td>
</tr>
<tr>
<td>03</td>
<td>0</td>
</tr>
<tr>
<td>04–08</td>
<td>Name of the module containing the invoked method</td>
</tr>
<tr>
<td>09–1B</td>
<td>Method name</td>
</tr>
<tr>
<td>1C-1F</td>
<td>RPH address or register 1</td>
</tr>
</tbody>
</table>

**OON2 entry for module trace**

**Entry:**
- OON2
VIT option:
N/A

Event:
Method has been entered

VIT processing module:
ISTRACOT

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced

This trace record is a continuation record of the OON trace record. Multiple OON2 records are present if the OO method being traced has more than six input parameters.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;OON2&quot;</td>
</tr>
<tr>
<td>04–1B</td>
<td>Addresses of the parameters passed to the OO method or 0</td>
</tr>
<tr>
<td>1C-1F</td>
<td>Return or invocation address of the requester of the OO method or address of the parameter passed to the OO method or 0</td>
</tr>
</tbody>
</table>

**Note:** One word for each parameter passed to the OO method will be listed in the trace entry. If the parameter is not present, it will have the value of 0. The first OON2 trace entry has the structure address at offset 4 and has the return or the invocation address of the OO method at offset X'1C'. Additional OON2 trace entries will have a parameter address at offset X'1C', instead of the return or invocation address.

**OOSQ entry for HPR out of sequence queue sequence numbers**

Entry:
OOSQ

VIT option:
HPR

Event:
Out of sequence queue sequence numbers

VIT processing module:
ISTITCHP

This trace record shows the HPR sequence numbers in all the DAPTRs on the HPR out of sequence queue. Each sequence number pair represents the sequence numbers that have been received and queued. The first gap is between the LAST BYTE RCVD (see “RTP entry for RTP PAB dispatch” on page 596) and the begin sequence number of the first sequence number pair. Additional gaps exist between the end sequence number of one pair and the begin sequence number of the next pair. If there are more than 2
gaps, this record is followed by as many OOS2 records as required to show all the sequence numbers on
the RPNCB out of sequence queue.

The OOSQ record is written at the start of an RTP PAB dispatch when at least one DAPTR is on the out of
sequence queue. OOSQ is also written at the end of an RTP PAB dispatch when at least one of the DAPTRs
on the out of sequence queue has been altered during the dispatch.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;OOSQ&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>DAP_Flags for sequence number pair 1</td>
</tr>
<tr>
<td>07</td>
<td>DAP_Flags for sequence number pair 2 (zeros if only a single pair exists)</td>
</tr>
<tr>
<td>08–0F</td>
<td>First sequence number pair</td>
</tr>
<tr>
<td></td>
<td>• Word 1  Begin Sequence number</td>
</tr>
<tr>
<td></td>
<td>• Word 2  End Sequence number</td>
</tr>
<tr>
<td>10–17</td>
<td>Second sequence number pair (zeros if only a single pair exists)</td>
</tr>
<tr>
<td></td>
<td>• Word 1  Begin Sequence number</td>
</tr>
<tr>
<td></td>
<td>• Word 2  End Sequence number</td>
</tr>
<tr>
<td>18–1B</td>
<td>RPNCB address</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

**OOS2 entry for HPR out of sequence queue sequence numbers (Part 2)**

**Entry:**

OOS2

**VIT option:**

HPR

**Event:**

Out of sequence queue sequence numbers (Part 2)

**VIT processing module:**

ISTITCHP
Any number (including 0) of OOS2's may follow the OOSQ, depending on the number of DAPTRs on the HPR out of sequence queue.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: 'OOS2'</td>
</tr>
<tr>
<td>04</td>
<td>OOS2 record counter</td>
</tr>
<tr>
<td>05</td>
<td>DAP_Flags for sequence number pair n</td>
</tr>
<tr>
<td>06</td>
<td>DAP_Flags for sequence number pair n +1 (zeros if pair does not exist)</td>
</tr>
<tr>
<td>07</td>
<td>DAP_Flags for sequence number pair n +2 (zeros if pair does not exist)</td>
</tr>
<tr>
<td>08–0F</td>
<td>Next sequence number pair</td>
</tr>
<tr>
<td></td>
<td>• Word 1  Begin sequence number</td>
</tr>
<tr>
<td></td>
<td>• Word 2  End sequence numbers</td>
</tr>
<tr>
<td>10–17</td>
<td>Next sequence number pair (zeros if pair does not exist)</td>
</tr>
<tr>
<td></td>
<td>• Word 1  Begin sequence number</td>
</tr>
<tr>
<td></td>
<td>• Word 2  End sequence number</td>
</tr>
<tr>
<td>18–1F</td>
<td>Next sequence number pair (zeros if pair does not exist)</td>
</tr>
<tr>
<td></td>
<td>• Word 1  Begin sequence number</td>
</tr>
<tr>
<td></td>
<td>• Word 2  End sequence number</td>
</tr>
</tbody>
</table>

**OOX entry for module trace**

**Entry:**

OOX

**VIT option:**

N/A

**Event:**

Method has been exited.
VIT processing module:  
ISTRACOT

Control is returned to:  
Module invoking the INTRACE macro that caused the record to be produced

This trace record is written when a method is exited.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;OOX&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Return code</td>
</tr>
<tr>
<td>04–08</td>
<td>Name of the module containing the invoked method</td>
</tr>
<tr>
<td>09–1B</td>
<td>Method name</td>
</tr>
<tr>
<td>1C-1F</td>
<td>RPH address or register 1</td>
</tr>
</tbody>
</table>

**OPER entry for operator command (Part 1)**

Entry:  
OPER

VIT option:  
MSG

Event:  
Operator command (Part 1)

VIT processing module:  
ISTRACOT

Control is returned to:  
ISTLUCMD, ISTCFF3D, ISTINCF9

This trace record is written each time a VTAM command is issued at the operator console. This can help you match the console log to a surge of activity shown in the VIT. This trace record is also useful when the console log is unavailable. Additionally, it provides a record of POA commands that have been issued. These do not appear on the console log.

The variable data in bytes 08–1F is the command text, with minor modifications. If this data is longer than 24 characters, it will continue in the OPE2 entry. The command always appears in abbreviated form:

D  
DISPLAY command

F  
MODIFY command
VARY command

HALT command

Unknown command passed to VTAM

NET and proname are excluded from the command text. Because the VTAM internal trace is not active when VTAM is started, the START command is not traced.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;OPER&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Flag byte</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>1...</td>
<td>A POA command was issued.</td>
</tr>
<tr>
<td>0...</td>
<td>Not a POA command.</td>
</tr>
<tr>
<td>.1...</td>
<td>A POA needs the command complete message.</td>
</tr>
<tr>
<td>.0...</td>
<td>No command complete message for POA.</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–1F</td>
<td>First 24 characters of the operator command</td>
</tr>
</tbody>
</table>

OPE2 entry for operator command (Part 2)

Entry:

OPE2

VIT option:

MSG

Event:

Operator command (Part 2)

VIT processing module:

ISTRACOT

This trace record is a continuation of the OPER entry.
**ORMG entry for storage management**

Entry:
ORMG

VIT option:
SMS

Event:
ISTORMMG Processing

VIT processing module:
ISTRACSM

Control is returned to:
ISTORMMG

This trace record records various storage management values at critical points in ISTORMMG.

---

**Byte (hex)**

**Contents**

**00–03**
Record ID: C"ORMG"

**04–1F**
Up to 28 more characters of the operator command

---

**Byte (hex)**

**Contents**

**00–03**
Record ID: C"ORMG"

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
0

**06**
The instance of the trace record in the module

**07**
0
08–0B
BPDOBFOR–Double-threaded obtained storage queue forward pointer

0C–0F
BPDOBACK–Double-threaded obtained storage queue backward pointer

10–13
ATCOROBT–Single-threaded obtained storage queue pointer

14–17
ATCORTBF–To be freed queue pointer

18–1B
Free CSA bytes on the obtained queue

1C–1F
ATCCSAFI–Free CSA requests on the obtained queue

**P64Q entry for 64-bit PLO queue manager events (IUTP64QM macro) (Part 1)**

**Entry:**
P64Q

**VIT option:**
CIA

**Event:**
Queue management using IUTP64QM

**VIT processing module:**
ISTITCSH

**Control is returned to:**
IUTP64QM issuer

This trace record is written when the IUTP64QM macro is run. The record is primarily used to identify the addition or removal of elements to or from a queue header by using the perform locked operation (PLO) assembler instruction.

| 00–03 | Record ID: C’P64Q’ |
| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’. |
IUTP64QM return code

- Add_Chain return codes:

<table>
<thead>
<tr>
<th>Code value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'00'</td>
<td>Add_Chain successful and Synch_Byte mask operation was performed (if specified).</td>
</tr>
<tr>
<td>X'04'</td>
<td>Add_Chain successful but Synch_Byte mask operation was not performed (LE</td>
</tr>
</tbody>
</table>

- Remove_Element return codes:

<table>
<thead>
<tr>
<th>Code value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'00'</td>
<td>Remove_Element successful and Synch_Byte mask operation was performed (if specified).</td>
</tr>
<tr>
<td>X'04'</td>
<td>Remove_Element successful but Synch_Byte mask operation was not performed (LE</td>
</tr>
<tr>
<td>X'08'</td>
<td>Remove_Element unsuccessful but Synch_Byte mask operation was performed (if specified).</td>
</tr>
<tr>
<td>X'0C'</td>
<td>Remove_Element unsuccessful and Synch_Byte mask operation was not performed (LE</td>
</tr>
</tbody>
</table>

- Synch_Byte return codes:

<table>
<thead>
<tr>
<th>Code value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'00'</td>
<td>Synch_Byte mask operation was performed.</td>
</tr>
<tr>
<td>X'04'</td>
<td>Synch_Byte mask operation was not performed (LE</td>
</tr>
</tbody>
</table>

Function:

C'A'
  For Add_Chain

C'R'
  For Remove_Element

C'S'
  For Synch_byte_only

Address of caller (within IUTP64QM expansion)

Contents of synch byte before the PLO assembler instruction

Contents of synch byte after the PLO assembler instruction

Contents of count field after the PLO assembler instruction

Element address:
- For Add_Chain, address of first element in chain
- For Remove_Element, address of element removed
- For Synch_byte_only, 0

19–1F
PLO queue manager header address (IUTP64QH)

P642 entry for 64-bit PLO queue manager event (IUTP64QM macro) (Part 2)

Entry:
   P642
VIT option:
   CIA
Event:
   Queue management using IUTP64QM
VIT processing module:
   ISTITCSH
This trace record is a continuation of the P64Q entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C’P642’</td>
</tr>
<tr>
<td>04–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>Address of first element on the queue after the perform locked operation (PLO) assembler instruction</td>
</tr>
<tr>
<td>10–17</td>
<td>Address of last element on the queue after the PLO assembler instruction</td>
</tr>
<tr>
<td>18–1B</td>
<td>0</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

PAGB entry for PAGE_BUFFER requests

Entry:
   PAGB
VIT option:
  CSM

Event:
  IVTCSM REQUEST=PAGE_BUFFER

VIT processing module:
  ISTITCCS

Control is returned to:
  IVTSMCBF or IVTSM6BF

Notes:
  • IVTSMCBF processes PAGE_BUFFER requests in 31-bit addressing mode.
  • IVTSM6BF processes PAGE_BUFFER requests in 64-bit addressing mode.

This trace record provides the status of an IVTCSM REQUEST=PAGE_BUFFER macroinstruction.

<table>
<thead>
<tr>
<th>00–03</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03 | Record ID: C"PAGB"

| 04 | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'. |

| 05 | 0 |

| 06 | Trace record flag: |
|    | B'0... ....' |
|    | Indicates that this is the last record for this event. |

| 07 | Trace record number to correlate all the entries for this particular event. |

| 08 | Type or state of buffers allocated |
|    | X'80' |
|    | Fixed |
|    | X'20' |
|    | Eligible to be made pageable |

| 09–0B | 0 |
0C–0D
Return code
0E–0F
Reason code
10–13
Address of utility routine caller or return address of the issuer of the IVTCSM macroinstruction
14–17
Address of the buffer list entry that was being processed when the error was encountered.
When an error occurs in 64-bit addressing mode, the last buffer list entry here is X'FFFFFFFF'. The PAG6 entry follows with the 64-bit address of the last buffer list entry.
18–1B
Number of buffers
1C–1F
THREAD value if specified or 0 if THREAD is not specified. The THREAD value is used only to correlate this trace record to a specific IVTCSM macroinstruction.

PAG2 entry for PAGE_BUFFER requests

Entry:
PAG2

VIT option:
CSM

Event:
IVTCSM REQUEST=PAGE_BUFFER

VIT processing module:
ISTICCS

This trace record is a continuation of the PAGB trace record. Each PAG2 record traces, at most, two buffers that were requested to be placed in a pageable state.

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAG2</td>
<td>0</td>
<td>FLAG</td>
<td>RECNUM</td>
<td>BUFFER TOKEN FOR INPUT BUFFER LIST ENTRY</td>
<td>BUFFER TOKEN FOR INPUT BUFFER LIST ENTRY OR 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)
Contents
00–03
Record ID: C"PAG2"
04–05
0
06
Trace record flag:
B'0... ....'
Indicates that this is the last record for this event.
B'1... ....'
Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.
Trace record number to correlate all the entries for this particular event.

**Buffer token contained in input buffer list entry**

Buffer token contained in input buffer list entry or 0

**PAG6 entry for PAGE_BUFFER requests**

**Entry:**
- PAG6

**VIT option:**
- CSM

**Event:**
- IVTCSM REQUEST=PAGE_BUFFER

**VIT processing module:**
- ISTITCCS

This trace record is a continuation of the PAGB trace record. It contains the 64-bit buffer list entry address. The PAG6 entry is present only when the PAGE_BUFFER request failed with the error return code in 64-bit addressing mode.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03</td>
<td>Record ID: C’PAG6’</td>
</tr>
<tr>
<td>04-05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace record flags:</td>
</tr>
<tr>
<td></td>
<td>B’0... ....’</td>
</tr>
<tr>
<td></td>
<td>Indicates that this is the last record for this event.</td>
</tr>
<tr>
<td></td>
<td>B’1... ....’</td>
</tr>
<tr>
<td></td>
<td>Indicates that additional records exist for this event. Use the trace record number from this entry to locate corresponding continuation records.</td>
</tr>
<tr>
<td></td>
<td>B’.1 ....’</td>
</tr>
<tr>
<td></td>
<td>Indicates that the buffer list is in 64-bit storage.</td>
</tr>
<tr>
<td>07</td>
<td>Trace record number to correlate all the entries for this particular event.</td>
</tr>
<tr>
<td>08-F</td>
<td>0</td>
</tr>
<tr>
<td>10-17</td>
<td>64-bit address of the last buffer list entry.</td>
</tr>
<tr>
<td>18-1F</td>
<td>0</td>
</tr>
</tbody>
</table>
PCIx entry for program-controlled or suspend interrupt

Entry:
- PCID, PCII, PCIR, PCIT, or PCIX

VIT option:
- PCID, PCII, and PCIR: CIA
- PCIT and PCIX: CIO

Event:
- Program-controlled or suspend interrupt

VIT processing module:
- ISTITCOD

Control is returned to:
- ISTTSCIE, ISTLLCIE, ISTSRRIE, ISTSICIE

This trace record is written when a program-controlled interrupt occurs.

This interrupt occurs for a CLAW channel-to-channel attached host, for HPDT read and write devices, or for the OSA-Express QDIO, Shared Memory Communications over Remote Direct Memory Access (SMC-R), Shared Memory Communications - Direct Memory Access (SMC-D), or the HiperSockets adapter read queue.

The PCID entry is recorded when the OSA-Express QDIO or HiperSockets adapter has completed a read operation. The PCID entry may or may not be preceded by a SIGA (read) operation for the same device.

The PCIT and PCIX entries are correlated to the SIOx, RIOx, and INTx entries for the same device using the CUA field. The combination of the information provided by these entries describe the channel program management and I/O operations for the device.

The PCIR entry is recorded when the "RoCE Express" feature completes a read operation or encounters an error condition. The PCIR entry is followed by the RPLE, RPLP, and RPLA (optional) entries. These entries include information that describes the type and destination of the data received or the type of the error encountered.

The PCII entry is recorded when the internal shared memory (ISM) device completes a write operation or encounters an error condition.

See for a description of the NCB fields.

The PCIT and PCIX entries are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.

PCID mapping and field descriptions

| ID | 04 | 06 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |

Byte (hex)

<table>
<thead>
<tr>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
</tr>
<tr>
<td>Record ID: C&quot;PCID&quot; for DINCB</td>
</tr>
<tr>
<td>04</td>
</tr>
<tr>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
</tbody>
</table>
05–06
0

07
Function:
- C"R" for real PCI interrupt
- C"V" for virtual PCI interrupt

08–0B
Device address in EBCDIC (as specified in the TRL deck)

0C–0F
DINCB address

10
Content of DINCB_Q_Data_Index_Next (Next Empty Read)

11
Content of DINCB_Q_Data_Index_Last (Last Empty Read)

12–13
Start of Count/Status Pairs
Starting at SLSB(1), these fields represent the number of consecutive SLSBs containing the same status, and what the status value is. See SLS_SLSB_Status constants for status values.

14–15
Continuation Count/Status pairs (if applicable)

16–17
Continuation Count/Status pairs (if applicable)

18–19
Continuation Count/Status pairs (if applicable)

1A–1B
Continuation Count/Status pairs (if applicable)

1C–1D
Continuation Count/Status pairs (if applicable)

1E–1F
Continuation Count/Status pairs (if applicable)

<table>
<thead>
<tr>
<th>PCIR and PCII mapping and field descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte (hex) Contents</td>
</tr>
<tr>
<td>00–03</td>
</tr>
<tr>
<td>Record ID:</td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) record descriptions 537
• C'PCII' for SLNCB
• C'PCIR' for SRNCB

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05-06
0

07
Interrupt reason
• C'C' when the interrupt is driven as a result of a completed work request
• C'D' when the interrupt is driven as a result of device deallocation
• C'E' when the interrupt is driven as a result of a device error
• C'V' when the interrupt is driven as a result of the adapter interrupt monitoring function

08–0B
Peripheral Component Interconnect Express (PCIe) function ID (PFID) that is associated with the interrupt

0C
• Summary flags for PCII event
• 0 for PCIR event

0D–0E
0

0F
Return code from control block token validation processing

10–17
Address of the associated control block. For PCIR records, this is the PFCTE. For PCII records, this is the SLNCB.

18–1F
The time stamp that is taken when the interrupt occurs

PCIX and PCIT mapping and field descriptions

| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| PCIX PCIT | ID | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
| PCIX PCIT | ID | STATE | ST | PC | CD | DEVICE | NCB ADDRESS | FLAGS | CODE | CODE | CODE | CODE | CODE | CODE | CSW |

Byte (hex)
Contents

00–03
Record ID: C"PCI"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Reserved

06
Link station state (NCBLNKST)
07  Operation code
08–0B  Channel device name in EBCDIC (either a device address or device number)
0C–0F  NCB address
10–13  Flag bytes (NCBFLAGS)
14  IOS Bit 1 IOSSSPND
X’80’  IOS suspended
15  I/O completion code (IOSCOD)
16–17  CPNCB_Index_Word
18–1F  Channel status word from IOSB.

PIU entry (Part 1)

Entry:
  PIU
VIT option:
  PIU
Event:
  PIU record (Part 1)
VIT processing module:
  ISTRACOT
Control is returned to:
  ISTLSC6V ISTTSCPDR, ISTTSCLS, ISTTSCRI, ISTTSCSR, ISTTSCWS, ISTTSCPD, ISTTSCLE, ISTTSCXS (or ISTINCF1 and INTINCS1 for RUPE)

This trace record provides information about external and internal FID4 PIUs.

For most PIUs, only the first 48 bytes are traced (a 26-byte TH, a 3-byte RH, and the first 19 bytes of RU). However, certain classes of PIUs, such as session control RUs and VTAM RUs, are traced in their entirety.

If the control block is a RUPE, the TH is reconstructed by the internal trace using what information is available. For TSCB entries, the PIU is recorded as is.

If the PIU is a response with sense data, this entry is generated whether the PIU option is in effect or not. It is treated as an exception condition and is always traced if the VIT is active, regardless of the VIT options specified.
00–03
  Record ID: C“PIU“

04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
  Control block ID: X'54'=RUPE, X'99'=TSCB

06–07
  The control block identifier in byte 05 determines the meaning of this field.
  TSCB or RUPE information:
  • If byte 05 indicates a TSCB, use the following values:
    – Byte 06 contains flags (TSCFLAG1).
    – Byte 07 contains a unique instance identifier.
  • If byte 05 indicates a RUPE, use the following value:
    – Bytes 06-07 contain the index for a network element address.

08–0B
  TSCB or RUPE address (see byte 05)

0C–1F
  20 bytes of PIU

**PIU2 entry (Part 2)**

**Entry:**
  PIU2

**VIT option:**
  PIU

**Event:**
  PIU record (Part 2)

**VIT processing module:**
  ISTRACOT

This trace record is a continuation of the PIU entry and contains 28 more bytes of FID4 PIU.
The VTAM internal trace will generate as many as 31 PIU2 records in the following situations:
• For PIUs that are formatted and contain session control RUs
• For RUs to or from the SSCP or PUs
• When the VIT LCS option was specified and the PIU contains LCS data

The number of records generated depends on the data count field in the transmission header (TH).

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;PIU2&quot;</td>
</tr>
<tr>
<td>04–1F</td>
<td>Up to 28 more bytes of the FID4 PIU (padded at right with zeros)</td>
</tr>
</tbody>
</table>

**PKI or PKO entry for TCP/IP packet sent or received**

**Entry:**
PKI or PKO

**VIT option:**
CIA

**Event:**
TCP/IP packet sent or received

**VIT processing module:**
ISTRACCI

**Control is returned to:**
ISTTCCXD, ISTTCCXB, ISTTCCWD, ISTTCCWB, ISTTCCLB, ISTTCCLD, ISTTCCCB, ISTTCCCD

This trace entry provides information about packets received by or sent from TCP/IP data link control (DLC).

A PKI trace record is written whenever the buffer control channel program completes with packets received.

A PKO trace record is written whenever a packet is transferred to the telecommunications subsystem or communication adapter by inserting it into the buffer control channel program.
C"PKO" for outbound packet data

**04**
ID is the primary address space ID (ASID). This field is 0 if the ID is greater than X'FF'.

**05**
RESERVED

**06–07**
Total length of the packet

**08–0B**
TIPAC address

**0C–1F**
First 20 bytes of packet data

**PKI2 or PKO2 entry for TCP/IP packet sent or received**

**Entry:**
PKI2 or PKO2

**VIT option:**
CIA

**Event:**
TCP/IP packet sent or received

**VIT processing module:**
ISTRACCI

This trace entry is a continuation of PKI and PKO.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>C&quot;PKI2&quot; for inbound packet data</td>
</tr>
<tr>
<td></td>
<td>C&quot;PKO2&quot; for outbound packet data</td>
</tr>
</tbody>
</table>

**04–1E**
Next 28 bytes of packet data

**Note:** There is a maximum of seven PKI2 or PKO2 records.

**PLOQ entry for PLO queue manager event (IUTPLOQM macro)**

**Entry:**
PLOQ

**VIT option:**
CIA

**Event:**
Queue management using IUTPLOQM

**VIT processing module:**
ISTITCOD
Control is returned to
IUTPLOQM issuer

This trace record is written when the IUTPLOQM macro is run. It is primarily used to identify the addition or removal of elements to or from a queue header.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;PLOQ&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>IUTPLOQM return code</td>
</tr>
<tr>
<td>Add_Chain return codes:</td>
<td></td>
</tr>
<tr>
<td><strong>Code Value</strong></td>
<td><strong>Meaning</strong></td>
</tr>
<tr>
<td>X'00'</td>
<td>Add_Chain successful and Synch_Byte mask operation was performed (if specified).</td>
</tr>
<tr>
<td>X'04'</td>
<td>Add_Chain successful but Synch_Byte mask operation was not performed (LE</td>
</tr>
<tr>
<td>Remove_Element return codes:</td>
<td></td>
</tr>
<tr>
<td><strong>Code Value</strong></td>
<td><strong>Meaning</strong></td>
</tr>
<tr>
<td>X'00'</td>
<td>Remove_Element successful and Synch_Byte mask operation was performed (if specified.)</td>
</tr>
<tr>
<td>X'04'</td>
<td>Remove_Element successful but Synch_Byte mask operation was not performed (LE</td>
</tr>
<tr>
<td>X'08'</td>
<td>Remove_Element unsuccessful but Synch_Byte mask operation was performed (if specified).</td>
</tr>
<tr>
<td>X'0C'</td>
<td>Remove_Element unsuccessful and Synch_Byte mask operation was not performed (LE</td>
</tr>
</tbody>
</table>
X'00'
  Synch_Byte mask operation was performed.

X'04'
  Synch_Byte mask operation was not performed (LE|GE failed).

07
  Function:
  • C"A" for Add_Chain
  • C"R" for Remove_Element
  • C"S" for Synch_byte_only

08–0B
  Element address:
  • For Add_Chain, address of first element in chain
  • For Remove_Element, address of element removed
  • For Synch_byte_only, 0

0C
  Contents of synch byte before PLO

0D
  Contents of synch byte after PLO

0E–0F
  Contents of count field after PLO

10–13
  Address of caller (within IUTPLOQM expansion)

14–17
  Address of first element on queue after the PLO

18–1B
  Address of last element on queue after the PLO

1C–1F
  PLO Queue manager Header address (IUTPLOQH)

**POOF entry for freeing storage from GETBLK pool**

**Entry:**
  POOF

**VIT option:**
  SMS

**Event:**
  POOLFREE macro called to free all storage in a GETBLK pool

**VIT processing module:**
  ISTRACSM

**Control is returned to:**
  Module invoking the INTRACE macro that caused the record to be produced

This trace record is generated when the POOLFREE macro uses the VTFREE macro to free all storage that was allocated with GETBLK. This trace record helps the VIT analysis tool determine that storage is freed.
Byte (hex)
Contents
00–03
Record ID: C"POOF"
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
0
06
Pool type. The value X'7A' indicates the CMIPPVT storage pool type. For other possible storage pool
types, see “FBLK entry for FREEBLK macro (Part 1)” on page 380 or “GBLK entry for GETBLK macro
(Part 1)” on page 393. For more information about storage pools, see z/OS Communications Server:
SNA Network Implementation Guide.
07
Return code from the POOLFREE macro
08–0F
0
10–13
Address of the issuer of the POOLFREE macro
14–17
Length of storage freed, including the GETBLK headers. (Because the headers are not included in
GBLK length fields, the length of storage freed in the POOF VIT entry will be larger than the amount of
storage reported in the GBLK VIT entries.)
18–1B
0
1C–1F
Request parameter header (RPH) address

POST entry for post waiting event

Entry:
POST
VIT option:
PSS
Event:
Post waiting event
VIT processing module:
ISTRACPS
Control is returned to:
ISTAPCTP

This trace record identifies an RPH that is being posted (using the TPPOST macro) for restart after a
TPWAIT macro. The POST entry is followed sometime later by a RESM entry. However, when a TPPOST
precedes the TPWAIT, the TPWAIT returns to the caller immediately without waiting. Then POST is not followed by RESM.

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>0</th>
<th>0</th>
<th>8</th>
<th>9</th>
<th>0</th>
<th>D</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>1</td>
<td>0</td>
<td>D</td>
<td>P</td>
<td>A</td>
<td>B</td>
<td>O</td>
<td>F</td>
<td>PST ADDRESS</td>
<td>PAB ADDRESS</td>
<td>RETURN ADDRESS</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03

Record ID: C"POST"

04

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05

0

06–07

PAB offset

08–0B

PST address

0C–0F

PAB address

10–13

Address of the issuer of the TPPOST macro

14–17

Work element address (from RPHWEA)

18–1B

PAB DVT address

1C–1F

Address of the RPH being posted

### PROA or PROD entry for Profile Add or Delete

**Entry:**

PROA or PROD

**VIT option:**

NRM

**Event:**

Profile Add or Delete

**VIT processing module:**

ISTRACNR

**Control is returned to:**

ISTSDCPM

This trace record gives information about PROFILE macroinstruction processing. It is written when a VTAM module issues a PROFILE macroinstruction to add or delete an RDTE profile.
Byte (hex)

Contents

00–03
Record ID:
- C"PROA": Add a profile
- C"PROD": Delete a profile

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05–07
Key index into the RDTE profile table or 0. Note that an index value of 0 is valid and represents the first slot in the profile table. The index might also be 0 for PROA entries created when a profile is first associated with an RDTE during SYSDEF processing.

08–0B
Input profile address

0C–0F
Output profile address

10–13
Return address of the module calling the ISTSDCRP module or the address of the module issuing PROFILE macro

14–17
RDTE address

18–1F
Resource name

QAPL entry for OSA-Express QDIO or HiperSockets accelerator parameter list (Part 1)

Entry:
QAPL

VIT option:
CIA

Subtrace Type:
DIO

Event:
Parameter list upon return from accelerator exit (EZBIFQDR)

VIT processing module:
ISTITCOD

Control is returned to:
ISTLLCIE

This trace record is written to show partial contents of the accelerator list.
Byte (hex)
Contents
00 - 03
  Record ID: C"QAPL"
04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05 - 06
  0
07
  Inbound queue identifier
08 - 09
  Number of SPacs on the InSPac queue
0A - 0B
  Number of SPacs on the OutSPac queue
0C - 0F
  0
10 - 13
  0
14
  Last reason accelerator exit took slowpath
15 - 17
  0
18 - 1B
  Thread (interrupt) identifier
1C - 1F
  Request parameter header (RPH) address

QAP6 entry for OSA-Express QDIO or HiperSockets accelerator parameter list (Part 2)

Entry:
  QAP6
VIT option:
  CIA
Subtrace Type:
  DIO
Event:
  Parameter list upon return from accelerator exit (EZBIFQDR)
VIT processing module:
  ISTITCOD
Control is returned to:
ISTLLCIE

This trace record is a continuation of the QAPL entry and is written after QAPL. This trace record is only for 64-bit addressing.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Q | A | P | 6 | 0 | InSpac Head | OutSpac Head | 0 |

**Byte (hex)**

**Contents**

**00 - 03**  
Record ID: C"QAP6"

**04 - 07**  
0

**08 - 0F**  
Address of the first SPac on the InSPac queue

**10 - 17**  
Address of the first SPac on the OutSPac queue

**18 - 1F**  
0

---

**QDIP entry for QDIO performance statistics**

**Entry:**  
QDIP

**VIT option:**  
CIA

**Subtrace Type:**  
DIO

**Event:**  
QDIO performance statistics

**VIT processing module:**  
ISTITCOD

**Control is returned to:**  
IUTLLCDQ

This trace record is written during data transmission to and from OSA-Express adapters operating in QDIO mode. Its purpose is to report performance statistics about the adapter.

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>QDIP</td>
<td>I</td>
<td>D</td>
<td>0</td>
<td>D</td>
<td>E</td>
<td>R</td>
<td>E</td>
<td>C</td>
<td>T</td>
<td>I</td>
<td>O</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>DEVICE ADDRESS</td>
<td>NCB ADDRESS</td>
<td>INTERFACE</td>
<td>UNIT PERCENTAGE</td>
<td>MODE IDENTIFIER</td>
<td>RPH ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Byte (hex)  
Contents
00–03  
Record ID: C"QDIP"
04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05  
0
06  
Direction:
• C'O' for outbound
• C'I' for inbound
07  
0
08–0B  
Device address in EBCDIC (as specified in the TRL deck)
0C–0F  
DINCB address
10–11  
The number of iterations performed looking for data during initial inbound processing.
12–13  
The number of iterations performed looking for data after return from the ULP
14–19  
0
1A–1B  
Module identifier for the module that issued the INTRACE
1C–1F  
Request parameter header (RPH) address

**QREQ entry for queued REQSTORE**

Entry:
QREQ
VIT option:
SMS
Event:
Queued storage request
VIT processing module:
ISTRACSM
Control is returned to:
ISTORFBQ

This trace record identifies a REQSTORE request that was waiting for one or more buffers and is now satisfied.
```plaintext
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"QREQ"
| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05         | Control block ID index value
| 06–07      | 0
| 08–0B      | PST address
| 0C–0F      | Address of buffer obtained
| 10–13      | 0
| 14–15      | Number of buffers requested
| 16–1B      | 0
| 1C–1F      | Request parameter header (RPH) address

### QRE2 entry for queued REQSTORE

**Entry:**
- QRE2

**VIT option:**
- SMS

**Event:**
- Queued storage request

**VIT processing module:**
- ISTRACSM

This trace record is a continuation of QREQ entry. It is generated when a queued REQSTORE macro is processed for more than one buffer. The number of records generated depends on the number of buffers obtained by the REQSTORE request.

VTAM generates up to 31 QRE2 trace records. Each QRE2 entry contains addresses for up to seven buffers. If more than 218 buffers are requested, the last 4 bytes (1C–1F) in the last QRE2 entry are set to X'FFFFFF', indicating that not all of the requested buffers are traced.
QRE2 SEVEN WORDS OF BUFFER ADDRESSES

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;QRE2&quot;</td>
</tr>
<tr>
<td>04–1F</td>
<td>Up to seven more words of buffer addresses (padded on the right with zeros)</td>
</tr>
</tbody>
</table>

QRYL entry for query language

Entry:
QRYL

VIT option:
MSG

Event:
Query language error

VIT processing module:
ISTRACOT

Control is returned to:
ISTINCUL

This trace record is generated when a nonzero return code is received from the QRYLANG macro. When the return code and reason code indicate that the MVS message service is not active, no QRYL trace record is generated.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;QRYL&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Language query block address</td>
</tr>
</tbody>
</table>
0C–0F
Language query block length (LQBSIZE)

10–13
Return code

14–17
Reason code

18–1F
First 8 bytes of language name used in query (LQBINLNG)

**QSRB entry for Queue Service Request Block (SRB) events**

**Entry:**
QSRB

**VIT option:**
CIA

**Subtrace Type:**
DIO

**Event:**
Schedule, dispatch, return, or exit of an SRB that is associated with the OSA-Express QDIO, Shared Memory Communications over Remote Direct Memory Access (SMC-R), Shared Memory Communications - Direct Memory Access (SMC-D), or a HiperSockets read operation

**VIT processing module:**
ISTITCOD

**Control is returned to:**
IUTLLCIE, ISTLLCWC, IUTLLCDQ, ISTSRIDQ, ISTSICDQ

This trace record is written to show the scheduling, dispatching, returning, and exiting of inbound OSA-Express QDIO, SMC-R, SMC-D, or HiperSockets processing.
Byte (hex)  
Contents  
00–03  
Record ID: "QSRB"  
04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.  
05  
Function:  
• C"D" for Dispatch  
• C"E" for Exit  
• C"R" for Return to IUTLLCD  
• C"S" for Schedule  
06–07  
SRB instance field that is used to correlate QSRB events  
08–0B  
Service Request Block (SRB) address or FRR parameter list address  
| • If bytes 18–19 are 'SR' or 'SI', this is the FRR parameter list address for function Dispatch, and 0 for all other functions.  
| • Otherwise, this is the SRB address.  
0C–0F  
NCB address that is associated with this SRB event  
| • If bytes 18–19 are 'SR', this is an RUNCB address  
| • If bytes 18–19 are 'SI', this is an SLNCB address  
| • Otherwise, this is a DINCB address  
10–13  
If bytes 18–19 are not 'SR' or 'SI', this field contains processing flags. See the proc_flags definitions in the module that is identified by bytes 1A-1B.  
| • If bytes 18–19 are 'SR' or 'SI', this field contains the input token that is related to the SMC-R event.  
14–15  
Dedicated queue identifier, or zeros if bytes 18–19 are 'SR' or 'SI'.  
16–17  
Affinity queue identifier, or zeros if bytes 18–19 are 'SR' or 'SI'.  
18–1B  
If this event is associated with an SMC-R event, this field is a 4-character module identifier, where the first 2 characters are 'SR'.  
| • If this event is associated with an SMC-D event, this field is a 4-character module identifier, where the first 2 characters are 'SI'.  
| • Otherwise:  
| • Bytes 18–19 are the affinity queue element count, or zeros if bytes 16–17 are zeros.  
| • Bytes 1A-1B are a 2-character module identifier for the module that issued the INTRACE.  
1C–1F  
Request parameter header (RPH) address.
QSR2 entry for Queue Service Request Block (SRB) event (Part 2)

Entry:
   QSR2
VIT option:
   CIA
Subtrace Type:
   DIO
Event:
   Schedule, dispatch, return, or exit of an SRB that is associated with the OSA-Express QDIO, Shared Memory Communications over Remote Direct Memory Access (SMC-R), or a HiperSockets read operation

VIT processing module:
   ISTITCOD
Control is returned to:
   IUTLLCIE, ISTLLCWC, IUTLLCDQ

This trace record is written following the QSRB entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;QSR2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>Time stamp of QSRB event</td>
</tr>
<tr>
<td>10–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

QUE entry for work element queued to PAB

Entry:
   QUE
VIT option:
   PSS
Event:
   Work element queued to PAB

VIT processing module:
   ISTRACPS
Control is returned to:
   ISTAPCTQ

This trace record shows a work element queued to a PAB to allow another VTAM routine to do further processing with the work element. This entry may be followed shortly by a DSP entry, representing the PSS dispatch of this PAB. However, if the PAB is already running, it might not be dispatched again. In this case, no DSP entry follows the TPQUE.
Byte (hex)
Contents

00–02
Record ID: C"QUE"

03
Control block ID of work element (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Status
Bit
Meaning
xx... .
Type of scheduling request
Bit
Meaning
00
TPQUE none
01
TPQUE normal
10
TPQUE delay
...x... .
0
...1...
Queue is in last-in-first-out (LIFO) order.
...0...
Queue is in first-in-first-out (FIFO) order.
.... 1...
Registers are saved in the RPH control block.
.... .x...
0
.... .1.
PAB work element queue gate bit (PABWEQG).
.... .0
PAB change bit (PABCHNG). PAB is not scheduled to run.
.... .1
PAB is scheduled to run.

06
PAB flag field (PABFLAGS)
Bit

Meaning

1...  . . .
   PAB is unconditionally scheduled.

.1...  . . .
   PAB closedown is in progress.

..1.  . . .
   PAB is synchronous.

....1 . . .
   PAB extension is present.

.... 1...
   Do not dequeue work element.

.... .1..
   Do not detach the RPH.

.... ..1.
   Indicates a very extended PAB.

.... ...1
   Indicates a slightly extended PAB.

07
PAB flag field (PABFLGS1)

Bit

Meaning

1...  . . .
   Switch the PST address of this PAB’s major control block to the new PST address contained in DYPNWPST.

.1...  . . .
   This PAB has a data space extension.

..1.  . . .
   This PAB’s major control block is an FMCB.

....1 . . .
   PAB can be referenced in PSW disable mode.

.... 1...
   PAB is persistent.

.... .1..
   APSTERM/APSINIT FMCB during PAB dispatch.

.... ..xx
   Reserved

08–0B
PST address

0C–0F
PAB address

10–13
Address of the issuer of the TPQUE macro

14–17
Address of work element to be queued

18–1B
Module name abbreviation (bytes 4, 5, 7, and 8 of the module name) or PAB DVT address (high-order bit of X’18’ = 0). For an explanation of the module naming convention, see “Using module names to isolate VTAM problems” on page 23. (The module name might be unavailable if the PAB being scheduled is associated with an address space different from the current one.)
QUEN entry for work element queued to any control block

Entry:
QUEN

VIT option:
PSS

Event:
Work element queued to any control block

VIT processing module:
ISTRACPS

Control is returned to:
ISTAPCTQ

This trace record shows a work element queued to any control block (not just a PAB) to allow another VTAM routine to further process the work element.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;QUEN&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Control block ID of work element (See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.)</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Address of PST or 0</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of the control block field to which the work element is queued</td>
</tr>
<tr>
<td>10–13</td>
<td>Address of the issuer of the TPQUE NONE macro</td>
</tr>
<tr>
<td>14–17</td>
<td>Address of work element to be queued</td>
</tr>
<tr>
<td>18–1B</td>
<td>0</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address or 0</td>
</tr>
</tbody>
</table>
**RACR entry for LU 6.2 invocation of RACROUTE**

**Entry:**
RACR

**VIT option:**
APPC

**Event:**
Resource access control

**VIT processing module:**
ISTRACAC

**Control is returned to:**
ISTNSCSI

This trace record shows information about the completion of a RACROUTE macro. This trace record is written during security processing for session establishment.

| 00–03 | Record ID: C"RACR"
| 04    | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05    | RACROUTE request
  | X'01' | AUDIT
  | X'02' | EXTRACT
  | X'03' | LIST
  | X'04' | VERIFY
| 06    | RACROUTE type
  | For AUDIT: 0
  | For EXTRACT, LIST, or VERIFY:
  | X'01' | CREATE
  | X'02' | DELETE
  | X'03' | ENCRYPT

**Byte (hex)**
**Contents**
00–03  Record ID: C"RACR"
04      ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05      RACROUTE request
  | X'01' | AUDIT
  | X'02' | EXTRACT
  | X'03' | LIST
  | X'04' | VERIFY
06      RACROUTE type
  | For AUDIT: 0
  | For EXTRACT, LIST, or VERIFY:
  | X'01' | CREATE
  | X'02' | DELETE
  | X'03' | ENCRYPT
X'04' EXTRACT

07
0
08–0B
  Address control environment element (ACEE) address
0C–0F
  Half-session ID
10–13
0
14–17
  RACROUTE completion code
    X'00'
    Request successfully completed
    X'04'
    Request completed with nonzero return/reason code
    X'08'
    RACROUTE failure
18–1B
  Request return code from security management product. See the appropriate manual for your security management product for an explanation of the return/reason codes.
1C–1F
  Request reason code from security management product. See the appropriate manual for your security management product for an explanation of the return/reason codes.

**RAPB entry for invoking a RoCE anchor pool operation (Part 1)**

**Entry:**
  RAPB

**VIT option:**
  SMS

**Event:**
  Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) 64-bit anchor pool operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**
  ISTITCSH

This trace record is written upon completion of an anchor pool operation.
Byte (hex)
Contents
00–03  
Record ID: C'RAPB'
04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05  
Function identifier:
  • 'D' for Delete
  • 'F' for Free
  • 'G' for Get
  • 'I' for Init
  • 'T' for Traverse
  • 'X' for GetIndex
06–07  
Return code
08–0F  
Cell pool identifier or address:
  • If this entry represents a Delete function, this field is 0.
  • If this entry represents an Init function, this field is the cell pool identifier.
  • Otherwise, this field is the cell pool address.
10–17  
Anchor cell pool block address
18–1B  
When this entry represents a Traverse or GetIndex function, this field is the cell index number; otherwise, 0
1C–1D  
When this entry represents a GetIndex function, this field is the cell instance number; otherwise, 0.
1E–1F  
Module identifier of the module that issued INTRACE.

RAP2 entry for invoking a RoCE anchor pool operation (Part 2)

Entry:
  RAP2
VIT option:
  SMS
Event:
  Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) 64-bit anchor pool operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.
VIT processing module:
  ISTITCSH
This trace record is a continuation of the RAPB trace record when an Init function is processed for an anchor pool.
Byte (hex)
Contents
00–03
Record ID: C’RAP2’
04–07
Length of the anchor block
08–0B
Length of the anchor cell
0C–0F
The first 20 characters of the cell poll identification string

RCEEx entry for RCE macroinstruction

Entry:
RCEA, RCEC, RCED, or RCEF

VIT option:
NRM

Event:
RCE macroinstruction

VIT processing module:
ISTRACNR

Control is returned to:
The module that issued the RCE macroinstruction

This trace record contains information about an RCEADD, RCECHG, RCEDEL, or RCEFIND macroinstruction issued by a VTAM module.

Byte (hex)
Contents
00–03
Record ID: C”RCEA” (RCEADD), C”RCEC” (RCECHG), C”RCED” (RCEDEL), C”RCEF” (RCEFIND)
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.
05
Return code
Type of RCE entry

Mode of RCE entry (in hex)
- 01=Base entity mode
- 02=Reference entity mode

Key value (base entity name, base index, or reference index, depending on mode)

Address of the issuer of the macro

For RCEADD, RCEDEL, and RCEFIND: Value in RCEPOUT1
For RCECHG: First half of new table name

For RCEADD, RCEDEL, and RCEFIND: Value in RCEPOUT2
For RCECHG: Second half of new table name

Data address

RCM entry for RCM PAB dispatch

Entry:
RCM

VIT option:
HPR

Event:
Dispatch of RTP context manager (RCM) PAB

VIT processing module:
ISTITCHR

Control is returned to:
The module that issued INTRACE type (RCM)

This trace record is written by the RTP context manager (RCM) to summarize specific activities.

<table>
<thead>
<tr>
<th>00–03</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RCM&quot;</td>
</tr>
</tbody>
</table>

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

VTAM internal trace (VIT) record descriptions 563
RCPx entry for invoking a RoCE verb (Part 1)

Entry:
RCPx

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) verb, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTITCSH

Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is written upon invocation or completion of a RoCE verb.
• C’RCPI’ for the RoCE parameter list information before the call
• C’RCPO’ for the RoCE parameter list information after the call

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.

05
Instance identifier within the module

06–07
Length of the parameter list (PLIST) that is used for the RoCE verb invocation

08–0B
A related control block for this particular ROCE verb. This control block can be either a storage control block or a control block that represents the queue pair, VLAN, or “RoCE Express” interface that this verb affects.

0C–0F
Address of the associated RUNCB

10–13
Address of the associated SRNCB

14–17
Parameter list address. This is a 64-bit address, but only the lower 32 bits are shown in the trace record.

18–19
Last 2 characters of the module that issued the RoCE verb

1A–1B
Peripheral Component Interconnect Express (PCIe) function ID (PFID) defining the "RoCE Express" interface that was the target of the RoCE verb, in hexadecimal

1C–1F
Request parameter header (RPH) address

RCP2 entry for invoking a RoCE verb (Part 2)

Entry:
  RCP2

VIT option:
  CIA

Event:
  Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) verb when the input area in the RoCE verb parameter list (PList) contains nonzero information.

VIT processing module:
  ISTITCSH

This trace record is a continuation of the RCPx entries. Multiple RCP2 entries can be generated, depending on the length of the input area in the PList that is used for the RoCE verb.

<table>
<thead>
<tr>
<th>0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1</th>
<th>0 1 2 3 4 5 6 7 8 9 A B C D E F 0 1 2 3 4 5 6 7 8 9 A B C D E F</th>
</tr>
</thead>
<tbody>
<tr>
<td>R C P 2</td>
<td>28 BYTES OF PLIST INPUT AREA</td>
</tr>
</tbody>
</table>

Byte (hex)
  Contents

00–03
  Record ID: C’RCP2’
04–1F
28 bytes of the input area in the PList

RCP3 entry for invoking a RoCE verb (Part 3)

Entry:
  RCP3

VIT option:
  CIA

Event:
  Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) verb when the output area in the RoCE verb parameter list (PList) contains nonzero information.

VIT processing module:
  ISTITCSH

This trace record is a continuation of the RCPO entry. Multiple RCP3 entries can be generated, depending on the length of the output area in the PList that is used for the RoCE verb.

| 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 | 28 bytes of PLIST output area

Byte (hex)
  Contents

00–03
  Record ID: C'RCP3'

04–1F
  28 bytes of the output area in the PList

RCV entry for RCV PAB dispatch

Entry:
  RCV

VIT option:
  HPR

Event:
  Dispatch of MNPS recovery processing (RCV) PAB

VIT processing module:
  ISTITCHR

Control is returned to:
  ISTRVRR1

This trace record is written when a Recovery PAB (RPAB) is dispatched. The entry contains Recovery PAB control block information.
Byte (hex)

Contents

00–03
Record ID: C"RCV"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06–07
RCV status flags

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1... ....</td>
<td>Initial dispatch of this RPAB.</td>
</tr>
<tr>
<td>...1.....</td>
<td>Crypto sessions are being recovered.</td>
</tr>
<tr>
<td>....1....</td>
<td>Recovery PAB termination is underway.</td>
</tr>
<tr>
<td>.... ..1..</td>
<td>The RPAB has completed its recovery processing.</td>
</tr>
<tr>
<td>.... ...1</td>
<td>All data to be recovered has been received from the Recovery Manager PAB.</td>
</tr>
<tr>
<td>.... ....1</td>
<td>The RPAB is awaiting the status from RTP before beginning termination of the recovery.</td>
</tr>
</tbody>
</table>

08–0B
Address of the RTP NCB being recovered by this RPAB

0C
State information for internal recovery PAB processing

0D
State information for recovery PAB communication with Session Services

0E
State information for recovery PAB communication with RTP Context Manager

0F
Recovery APPC state

10
RCV input flags

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
</table>

This RPAB is still chained in the Recovery Manager session data queue.

No more data will be coming from the Recovery Manager.

The RPAB should terminate recovery when it is next dispatched.

Session Services was unable to send a response IPS to a prior MNPS_SESSINFO(Recover) request.

Session Services was unable to send a response IPS to a prior MNPS_SESSINFO(Terminate) request.

Number of sessions being recovered

Address of the Recovery Manager control block (RVM)

Request parameter header (RPH) address

RDSC entry for RUPE discard (Part 1)

Entry:

RDSC

VIT option:

PIU

Event:

Discarding a RUPE

VIT processing module:

ISTRACOT

Control is returned to:

ISTDLCDI

This trace record is written when VTAM’s dependent LU server receives an erroneous signal either from the VTAM configuration services component or from the dependent LU requester node across the CPSVRMGR session.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RDSC&quot;</td>
</tr>
</tbody>
</table>
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05–07
Reason code (decimal)

1
The dependent LU server abend recovery routine, ISTDLCRR, performs clean-up.

2
 CV X'60' is not found.

3
Dependent LU requester element is not found.

4
PU element is not found.

5
LU element is not found.

6
Storage shortage has occurred.

7
Dependent LU requester finite state machine is not valid.

8
Dependent LU requester finite state machine is pending inactive.

9
CPSVRMGR session pipe activation failure has occurred.

10
Unrecognized request unit processing element (RUPE) detected.

11
Start transaction program (STARTTP) failure has occurred.

12
CPSVRMGR session pipe has already been deactivated.

13
PU finite state machine is not valid.

14
SEND_ENCAP_TP process anchor block (PAB) does not exist.

08–0B
Address of the RUPE being discarded

0C–1F
First 20 bytes of PIU data (inbound) or RUPE's RU data (outbound) in the RUPE being discarded

RDS2 Entry for RUPE Discard (Part 2)

Entry:
RDS2

VIT option:
PIU

Event:
Discarding a RUPE

VIT processing module:
ISTRACOT

This trace record is a continuation of the RDSC entry.
Byte (hex)
Contents
00–03
  Record ID: C"RDS2"
04–17
  Bytes 21 through 40 of the PIU data (inbound) or RUPE's RU data (outbound) in the RUPE being discarded
18–1B
  Return address
1C–1F
  The module name that initiates the RUPE discard

**RDVX entry for channel exit redrive**

Entry:
  RDVX

VIT option:
  CIO

Event:
  Channel interrupt

VIT processing module:
  ISTRACCI

Control is returned to:
  ISTTSCLC

This trace record is written when a channel program interrupt occurs. RDVX is generated for interrupts between channel-to-channel-attached hosts.

The RDVX trace entry may follow the INTX entry. The RDVX entry is written when an abnormal I/O subsystem code has been presented back to VTAM for this CTC device. This entry documents the redrive of the CTC channel exit.

See for a description of the NCB fields.

The RDVX events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.
Record ID: C"RDVX"

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Station state (XCNSSFSM)

Macro starting instance

Channel device name in EBCDIC (either a device address or device number)

XCNCB address

Flag bytes (NCBFLAGS)

Flag byte as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>Running in disabled interrupt exit (IOSDIE is not zero)</td>
</tr>
<tr>
<td>.1...</td>
<td>ERP work area address exists (IOSERP is not zero)</td>
</tr>
<tr>
<td>..1...</td>
<td>Exception condition (IOSEX flag is on)</td>
</tr>
<tr>
<td>...1...</td>
<td>Error routine in control (IOSERR flag is on)</td>
</tr>
</tbody>
</table>

I/O completion code (IOSCOD)

Sense data in IOSSNS if this was a sense channel program; otherwise, zero

Channel status word from IOSB

RE entry for RPL exit

Entry:
RE
VIT option:
API
Event:
RPL exit
VIT processing module:
ISTRACAP
Control is returned to:
ISTAPCSX, ISTAICRX, or ISTAICPT

This trace record contains RPL information passed from VTAM to the application program. If the return code or feedback is nonzero, this entry is generated whether the API option is in effect or not. It is treated as an exception condition and is always traced if the VIT is active, regardless of the VIT options specified.

See z/OS Communications Server: SNA Programming for a description of RPL return codes.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–01</td>
<td>Record ID: C&quot;RE&quot;</td>
</tr>
<tr>
<td>02</td>
<td>RPL request type (RPLREQ)</td>
</tr>
<tr>
<td>X'15'</td>
<td>SETLOGON</td>
</tr>
<tr>
<td>X'16'</td>
<td>SIMLOGON</td>
</tr>
<tr>
<td>X'17'</td>
<td>OPNDST</td>
</tr>
<tr>
<td>X'19'</td>
<td>CHANGE</td>
</tr>
<tr>
<td>X'1A'</td>
<td>INQUIRE</td>
</tr>
<tr>
<td>X'1B'</td>
<td>INTRPRET</td>
</tr>
<tr>
<td>X'1C'</td>
<td>CLSDST</td>
</tr>
<tr>
<td>X'22'</td>
<td>SEND</td>
</tr>
<tr>
<td>X'23'</td>
<td>RECEIVE</td>
</tr>
<tr>
<td>X'24'</td>
<td>RESETSR</td>
</tr>
<tr>
<td>X'25'</td>
<td>SESSIONC</td>
</tr>
<tr>
<td>X'27'</td>
<td>SENDCMD</td>
</tr>
<tr>
<td>X'28'</td>
<td>RCVCMD</td>
</tr>
<tr>
<td>X'29'</td>
<td>REQSESS</td>
</tr>
</tbody>
</table>
X'2A'
   OPNSEC
X'2C'
   TERMSESS

03
   Exit definition (RPLEXTDS)

04
   ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
   Return code (RPLRTNCD) (See z/OS Communications Server: SNA Programming)

06
   Feedback code (RPLFDB2) (See z/OS Communications Server: SNA Programming)

07
   Feedback data flag (RPLFDB3) (See z/OS Communications Server: SNA Programming under INQUIRE macro with OPTCD=APPSTAT)

08–0B
   RPL address

0C–0F
   Exit address

10–13
   RPL data area pointer (RPLAREA)

14–17
   Record length (RPLRLEN)

18–1B
   CID from NIB or from RPL (RPLARG) or 0

1C–1F
   Second RPL feedback area - Sense code (RPLFDBK2)

RELSE entry for release storage

Entry:
   RELS

VIT option:
   SMS

Event:
   Release storage

VIT processing module:
   ISTRACSM

Control is returned to:
   ISTORMBD

This trace record provides the status of each RELSTORE request issued by VTAM components to release fixed-length buffers to one of the predefined buffer pools.

RELSTORE is the complement of REQSTORE. Storage obtained by REQSTORE must be released by RELSTORE. However, because more than one buffer can be obtained and released at a time, there may not be a RELS for every REQ, and there may not be a REQS for each RELS. Use the "next buffer address" in RELS and the "number of buffers" in REQS to help determine the correlation between buffers requested and buffers released.
### Byte (hex)

**Contents**

00–03  
Record ID: C"RELS"

04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05  
Control block ID index value or 0. A 0 is returned if the buffer is not allocated for a specific control block.

For the index values and the control block IDs they represent, see z/OS Communications Server: SNA Data Areas Volume 1.

06  
0

07  
Return code

08–0B  
PST address

0C–0F  
Address of buffer to be released

10–13  
Address of the issuer of the RELSTORE macro

14–17  
Pointer to the next buffer to be released if more than one buffer to be released; otherwise, 0

18–1B  
Register 1 (normally RPH address)

1C–1F  
Caller of utility routine, or 0

Address of the utility's caller if the RELSTORE macro was issued from a utility routine; otherwise, 0 if the macro was issued directly by the caller without a utility routine (See bytes 10–13 for the address of the issuer of the RELSTORE macro).

---

**REML entry for TPREMEL macro**

**Entry:**

REML

**VIT option:**

APPC, HPR

**Event:**

Element TPREMELed from a work queue; only for selective TPREMELs
**VIT processing module:**
ISTRACAC

**Control is returned to:**
Module invoking the INTRACE macro that caused the record to be produced

This trace record is written when an element is dequeued from a work queue and processing begins on the element.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;REML&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>REML instance within issuing module</td>
</tr>
<tr>
<td>07–0B</td>
<td>Issuing module name</td>
</tr>
<tr>
<td>0C–0E</td>
<td>0</td>
</tr>
<tr>
<td>0F</td>
<td>Control block ID</td>
</tr>
<tr>
<td>10–13</td>
<td>Information field or 0:</td>
</tr>
<tr>
<td></td>
<td>• AMU ID if work element is an AMU</td>
</tr>
<tr>
<td></td>
<td>• CONTROL and QUALIFY values if work element is an LU6.2 RPL</td>
</tr>
<tr>
<td></td>
<td>• Control point operation code if work element is a RUPE</td>
</tr>
<tr>
<td></td>
<td>• 0 if work element is a UECB or TQE.</td>
</tr>
<tr>
<td></td>
<td>• DAPT, IN, OUT, or TIMB if REML is traced for HPR</td>
</tr>
<tr>
<td>14–17</td>
<td>Work element address</td>
</tr>
<tr>
<td>18–1B</td>
<td>Address of the queue from which the work element is being dequeued</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

**REMQ entry for APPC**

**Entry:**
REMQ
**VIT option:**
APPC

**Event:**
Element TPDEQueued from a PAB; only for selective TPDEQs

**VIT processing module:**
ISTRACAC

**Control is returned to:**
Module invoking the INTRACE macro that caused the record to be produced

This trace record is written when an element is dequeued from a PAB and processing begins on the element.

![Trace Record Example]

**Byte (hex)**

**Contents**

00–03  
Record ID: C"REMQ"

04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05  
0

06  
REMQ instance within issuing module

07–0B  
Issuing module name

0C  
Conversation FSM state or 0

0D  
Router FSM state or 0

0E  
Error or failure FSM state or 0

0F  
Control block ID

10–13  
Information field or 0

- AMU ID if work element is an AMU
- CONTROL and QUALIFY values if work element is an LU6.2 RPL
- Control point operation code if work element is an RUPE
- 0 if work element is a UECB or TQE

14–17  
Work element address

18–1B  
Address of the PAB from which the work element is being dequeued

---

576  z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
1C–1F
Request parameter header (RPH) address

**REQS entry for request storage (Part 1)**

<table>
<thead>
<tr>
<th>Entry:</th>
<th>REQS</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT option:</td>
<td>SMS</td>
</tr>
<tr>
<td>Event:</td>
<td>Request storage (Part 1)</td>
</tr>
<tr>
<td>VIT processing module:</td>
<td>ISTRACSM</td>
</tr>
<tr>
<td>Control is returned to:</td>
<td>ISTORMBA</td>
</tr>
</tbody>
</table>

This trace record provides the status of each REQSTORE request issued by VTAM components to obtain fixed-length buffers from one of the predefined buffer pools. The storage obtained by REQSTORE is released by RELSTORE.

If this entry is associated with an event failure (that is, the return code is nonzero), this entry is generated whether the SMS option is in effect or not. It is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;REQS&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Control block ID index value or 0. A 0 is returned if the buffer is not allocated for a specific control block or if the buffer allocation failed. For the index values and the control block IDs they represent, see z/OS Communications Server: SNA Data Areas Volume 1.</td>
</tr>
<tr>
<td>08–0B</td>
<td>Pointer to the PST, or 0</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of buffer obtained, or 0 if buffer not obtained</td>
</tr>
<tr>
<td>10–13</td>
<td>Address of the issuer of the REQSTORE macro</td>
</tr>
<tr>
<td>14–15</td>
<td>Number of buffers requested</td>
</tr>
<tr>
<td>16–17</td>
<td>Return code from REQSTORE</td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) record descriptions 577
Successful.

Storage obtained, but slowdown threshold exceeded.

Storage request queued, but buffer pool has to be expanded before buffers can be allocated.

The number of buffers requested could not be allocated immediately. Depending on the conditions specified on the REQSTORE request and the state of the buffer pool that is requested, the request may or may not be queued for future processing.

The request is queued for future processing if one of the following conditions is met:

• The REQSTORE request specifies that it can be queued for future processing regardless of the state of the buffer pool from which buffers are requested.

• The REQSTORE request specifies that it can be queued for future processing if specific conditions are met. These conditions are as follows:
  - Number of buffers already allocated plus the number of buffers for which there are outstanding requests does not exceed the maximum size of the buffer pool.
  - Buffer pool is defined to be expandable.
  - Buffer pool expansion has never failed for the buffer pool or the buffer pool has contracted since it failed to expand.

The request is not queued for future processing if any of the following conditions apply:

• The REQSTORE request specifies that it is not to be queued under any circumstances.

• The REQSTORE request specifies that it is to be queued according to one or more of the conditions described previously in this topic and the specified conditions are not met.

This return code applies only to nonexpandable buffer pools. The number of buffers requested exceeds the maximum number which can be allocated in one request. This return code does not indicate that the number of buffers requested is not available; it indicates that the number of buffers exceeds the maximum allowed in one request.

Register 1 (normally RPH address)

Number of available buffers in pool after this request is processed.

**REQ2 entry for request storage (Part 2)**

**Entry:**

REQ2

**VIT option:**

SMS

**Event:**

Request storage (Part 2)

**VIT processing module:**

ISTRACSM

This trace record is a continuation of the REQS entry. It is generated when REQSTORE macro was issued for more than one buffer. The number of records generated depends on the number of buffers obtained by the REQSTORE request.

VTAM generates up to 31 REQ2 trace records. Each REQ2 entry contains addresses for up to seven buffers. If more than 218 buffers are requested, the last 4 bytes, 1C–1F, in the last REQ2 entry are set to X'FFFF' indicating that not all of the requested buffers are traced.
Byte (hex)
Contents
00–03
Record ID: C"REQ2"
04–1F
Up to seven more words of buffer addresses (padded on the right with zeros)

**RESM entry for resume event processing**

<table>
<thead>
<tr>
<th>Entry:</th>
<th>RESM</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT option:</td>
<td>PSS</td>
</tr>
<tr>
<td>Event:</td>
<td>Resume event processing</td>
</tr>
<tr>
<td>VIT processing module:</td>
<td>ISTRACPS</td>
</tr>
<tr>
<td>Control is returned to:</td>
<td>ISTAPCPD</td>
</tr>
</tbody>
</table>

This trace record gives information about which VTAM process is about to be redispached after a TPWAIT. A RESM entry follows a POST, although not every POST has a RESM. The POST entry is followed sometime later by a RESM entry. However, when a TPPOST precedes the TPWAIT, the TPWAIT returns to the caller immediately without waiting. Then POST is not followed by RESM. Use the RPH address to find the original DSP entry, which may have occurred earlier.
<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>PAB is unconditionally scheduled.</td>
</tr>
<tr>
<td>.1..</td>
<td>PAB closedown is in progress.</td>
</tr>
<tr>
<td>..1.</td>
<td>PAB is synchronous.</td>
</tr>
<tr>
<td>...1</td>
<td>PAB extension is present.</td>
</tr>
<tr>
<td>....1</td>
<td>Do not dequeue work element.</td>
</tr>
<tr>
<td>.....1</td>
<td>Do not detach the RPH.</td>
</tr>
<tr>
<td>.....1</td>
<td>Indicates a very extended PAB.</td>
</tr>
<tr>
<td>..... .1</td>
<td>Indicates a slightly extended PAB.</td>
</tr>
</tbody>
</table>

**07**

PAB flag field (PABFLGS1)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>Switch the PST address of this PAB’s major control block to the new PST address contained in DYPNW PST.</td>
</tr>
<tr>
<td>.1..</td>
<td>This PAB has a data space extension.</td>
</tr>
<tr>
<td>..1.</td>
<td>This PAB’s major control block is an FMCB.</td>
</tr>
<tr>
<td>...1</td>
<td>PAB can be referenced in PSW disable mode.</td>
</tr>
<tr>
<td>....1</td>
<td>PAB is persistent.</td>
</tr>
<tr>
<td>.....xx</td>
<td>APSTERM/APSINIT FMCB during PAB dispatch.</td>
</tr>
</tbody>
</table>

**08–0B**

PST address

**0C–0F**

PAB address

| 10–13 | Work element address or next dispatchable queue level for a very extended PAB |

| 14–17 | PAB work element queue |

| 18–1B | PAB DVT address or name |

| 1C–1F | Request parameter header (RPH) address |
RIOx entry for Resume I/O (Part 1)

Entry:
RIOx

VIT option:
CIO

Event:
Resume I/O

VIT processing module:
ISTRACCI

Control is returned to:
ISTLLCXR, ISTTSCIE, ISTTSCNY, ISTTSCXR, or ISTTSC8W

This trace record is written when the MVS Resume I/O service is invoked and for HPDT only.

Eventually, an INTx or PCIx entry follows the RIOx trace record. Use the CUA field to correlate the entries. See for a description of the NCB fields.

For record types with suffix I, X, or T, the CIO events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: &quot;RIOx&quot; for XCNCB</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>State</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Channel device name in EBCDIC (either a CUA or device number)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>NCB address</td>
</tr>
<tr>
<td>10–13</td>
<td>Flag bytes (NCBFLAGS)</td>
</tr>
<tr>
<td>14–17</td>
<td>Virtual channel address word (NCBCAW)</td>
</tr>
<tr>
<td>18–1F</td>
<td>Output control area (status information)</td>
</tr>
</tbody>
</table>
RIO2 entry for Resume I/O (Part 2)

Entry:
  RIO2

VIT option:
  CIO

Event:
  Resume I/O (Part 2)

VIT processing module:
  ISTRACCI

This trace record is a continuation of the RIOx entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RIO2&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>CPNCB_Index_Word</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–09</td>
<td>Module ID</td>
</tr>
<tr>
<td>0A–19</td>
<td>First 16 bytes of MPC transmit block data, or 0, or device-dependent data</td>
</tr>
<tr>
<td>1A–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

RPLx entry for invoking a RoCE Poll command (Part 1)

Entry:
  RPLx

VIT option:
  CIA

Event:
  Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) Poll command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
  ISTITCSH

Control is returned to:
  The module that issued the INTRACE macroinstruction

This trace record is written upon completion of the RoCE Poll operation.
Byte (hex)

Contents

00–03
Record ID:
  • C'RPLE' for Poll for event information
  • C'RPLR' for Poll for received data information
  • C'RPLS' for Poll for sent data information

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05–06
0

07
Number of array entries that are filled as a result of the Poll operation

08–09
Return code from the Poll operation

0A–0B
Reason code

0C–0F
RUNCB address or 0
  • For the RPLR and RPLS records, the address of the associated RUNCB
  • For RPLE records, 0

10–17
Parameter list address.

18–1B
Return address of the calling routine

1C–1F
Request parameter header (RPH) address

**RPLP entry for invoking a RoCE Poll command (Part 2)**

Entry:
RPLP

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) PollCQ or PollEQ command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.
VIT processing module:
   ISTITCSH

This trace record is continuation of the RPLx entry. A single RPLP entry is generated, which provides
information about the base portion of the PollCQ or PollEQ parameter list.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'RPLP'</td>
</tr>
<tr>
<td>04–07</td>
<td>Option and output flags:</td>
</tr>
<tr>
<td></td>
<td>• If this RPLP record follows an RPLE record, the option and output flags data</td>
</tr>
<tr>
<td></td>
<td>• Otherwise, the option flag data</td>
</tr>
<tr>
<td>08–0F</td>
<td>RoCE token</td>
</tr>
<tr>
<td>10–1F</td>
<td>Queue token</td>
</tr>
</tbody>
</table>

RPLA entry for invoking a RoCE Poll command (Part 3)

Entry:
   RPLA

VIT option:
   CIA

Event:
   Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) PollCQ or PollEQ command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
   ISTITCSH

This trace record is a continuation of the RPLx entry. Multiple RPLA entries can be generated, one for each
array entry that contains data at the completion of the PollCQ or PollEQ operation. A single Poll operation
can have up to 64 array entries with data.
Byte (hex)
Contents
00–03
  Record ID: C’RPLA’
04–07
  Flags:
  • If this RPLA record follows an RPLE record, 0
  • Otherwise, the output flags data for this array entry
08–0B
  Immediate data or event type:
  • If this RPLA record follows an RPLE record:
    – Bytes 08–09 are 0
    – Byte 0A is the event type
    – Byte 0B is the event subtype
  • Otherwise, the immediate data, if any, that is associated with this array entry
0C–0D
  Syndrome value:
  • If this RPLA record follows an RPLE record, 0
  • Otherwise, the syndrome value that is associated with this array entry
0E–0F
  Work Queue Element (WQE) number:
  • If this RPLA record follows an RPLE record, 0
  • Otherwise, the Work Queue Element (WQE) number that is associated with this array entry
10–1F
  User data:
  • If this RPLA record follows an RPLE record, the event user data field
  • If this RPLA record follows an RPLR record:
    – Bytes 10–17 are 0
    – Bytes 18–1F are the queue pair user data that is associated with this array entry
  • If this RPLA record follows an RPLS record:
    – Bytes 10–17 are 0
    – Bytes 18–1F are the sender user data that is associated with this array entry
RPLI entry for invoking a RoCE PollCQ command (Part 4)

Entry:
RPLI

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) PollCQ command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTITCSH

This trace record is continuation of the RPLA entry, and the RPLA entry is a continuation of the RPLx entry. One or two RPLI entries can be generated per corresponding RPLA entry, depending on the length of the inline data that is associated with the PollCQ array entry.

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

RPLI 28 BYTES OF ARRAY INLINE ENTRY DATA

Byte (hex)
Contents
00–03
Record ID: C’RPLI’
04–1F
28 bytes of inline data that is associated with the PollCQ array entry

RPST entry for invoking a RoCE Post command (Part 1)

Entry:
RPST

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) Post command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTITCSH

Control is returned to:
The module that issued the INTRACE macroinstruction

This trace record is written upon completion of a RoCE Post operation.
Entry:
RPSP

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) Post command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTITCSH

This trace record is continuation of the RPST record. A single RPSP entry is generated, and provides information about the base portion of the Post parameter list.

**RPSP entry for invoking a RoCE Post command (Part 2)**

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C’RPST’</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.</td>
</tr>
<tr>
<td>05–06</td>
<td>Option and output flags</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–09</td>
<td>Return code from the Post operation</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Reason code</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of the associated RUNCB</td>
</tr>
<tr>
<td>10–17</td>
<td>Parameter list address</td>
</tr>
<tr>
<td>18–1B</td>
<td>Return address of the calling routine</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>
### Byte (hex)

<table>
<thead>
<tr>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
</tr>
<tr>
<td>Record ID: C'RPSP'</td>
</tr>
<tr>
<td>04–05</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>06</td>
</tr>
<tr>
<td>Number of operations (array entries) provided as input on this Post command</td>
</tr>
<tr>
<td>07</td>
</tr>
<tr>
<td>Number of operations that completed successfully</td>
</tr>
<tr>
<td>08–0F</td>
</tr>
<tr>
<td>RoCE token</td>
</tr>
<tr>
<td>10–1F</td>
</tr>
<tr>
<td>Queue Pair (QP) token</td>
</tr>
</tbody>
</table>

#### RPSA entry for invoking a RoCE Post command (Part 3)

**Entry:**
- RPSA

**VIT option:**
- CIA

**Event:**
- Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) Post command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**
- ISTITCSH

This trace record is continuation of the RPST record. Multiple RPSA entries can be generated, one for each array entry that was provided as input on the Post command.
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'RPSA'</td>
</tr>
</tbody>
</table>
| 04        | Post operation type  
  • I for Send Immediate operation  
  • M for RDMA Write Immediate operation  
  • W for RDMA Write operation  
  • S for Send operation |
| 05        | 0 |
| 06–07     | Work Queue Element (WQE) number that is associated with this array entry |
| 08–0F     | UserData that is associated with this Post operation |
| 10–13     | Amount of data to be sent on this Post operation |
| 14–17     | Local Key (LKEY) of the source buffer or 0:  
  • If Post operation type is 'M' or 'W', the Local Key (LKEY) of the source buffer  
  • Otherwise, 0 |
| 18–1B     | 0 |
| 1C–1F     |  
  • If Post operation is 'I' or 'M', immediate data to be sent as part of this Post operation  
  • If Post operation is 'S', write completion timer control field  
  • Otherwise, 0 |

**RPS2 entry for invoking a RoCE Post command (Part 4)**

Entry:  
RPS2
VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) Post command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTITCSH

This trace record is a continuation of the RPSA entry, and the RPSA entry is a continuation of the RPST record. One RPS2 entry can be generated per corresponding RPSA entry, if the RPSA entry represents an RDMA Write operation.

<table>
<thead>
<tr>
<th>R</th>
<th>P</th>
<th>S</th>
<th>2</th>
<th>RKEY</th>
<th>REMOTE VIRTUAL ADDRESS</th>
<th>DATA SEGMENT ADDRESS</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>00–03</td>
<td>01 2 3 4 5 6 7 8</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>04–07</td>
<td>9 A B C D E F 0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>08–0F</td>
<td>01 2 3 4 5 6 7 8</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10–17</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>18–1F</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Byte (hex)

Contents

00–03
Record ID: C’RPS2’

04–07
Remote key (RKEY) of the destination buffer

08–0F
Address within the remote destination buffer where the data is to be stored

10–17
Source address for the data to be sent

18–1F
0

RPSI entry for invoking a RoCE Post command (Part 5)

Entry:
RPSI

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) Post command, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

VIT processing module:
ISTITCSH

This trace record is a continuation of the RPSA entry, and the RPSA entry is a continuation of the RPST record. One or two RPSI entries can be generated per corresponding RPSA entry, depending on the length
of the inline data that is associated with the Post array entry. If no inline data is associated with this Post operation, no RPSI record is created for this entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'RPSI'</td>
</tr>
<tr>
<td>04–1F</td>
<td>28 bytes of inline data that is associated with the Post array entry</td>
</tr>
</tbody>
</table>

### RQE entry for read queue exit

**Entry:**
- **RQE**

**VIT option:**
- **CMIP**

**Event:**
- Read-queue exit called by CMIP application program

**VIT processing module:**
- ISTITCCM

**Control is returned to:**
- Module invoking the INTRACE macro that caused the record to be produced

This trace record is generated when the read-queue exit of a CMIP application program returns from processing a message. The trace record shows the information that is passed from the MIB controller to the application program and the return code passed from the application program.

For detailed descriptions of the fields, see [z/OS Communications Server: CMIP Services and Topology Agent Guide](#).

If the return code is not 0, this entry is always traced, regardless of the VIT options specified.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RQE&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
</tbody>
</table>
05
0

06–07
Reason code indicating the reason the read-queue exit is called

08–0B
Link identifier

0C–0F
Invoke identifier or 0. The invoke identifier can be used to correlate this VIT entry with the CMIP message string in the buffer trace.

10–13
Return code from application program

14–17
User data

18–1B
Length of data being passed to the read-queue exit

1C–1F
Request parameter header (RPH) address

**RSCx entry for RSCVSCAN macroinstruction (Part 1)**

**Entry:**

RSC

**VIT option:**

HPR

**Event:**

Invocation of RSCVSCAN macroinstruction

**VIT processing module:**

ISTITCHP

**Control is returned to:**

Module invoking the RSCVSCAN macroinstruction

This trace record is written when the RSCVSCAN macroinstruction is issued. The record displays the input RSCV and the output RSCVs (the expanded, condensed, and essential RSCVs).

| 00  | 01  | 02  | 03  | 04  | 05  | 06  | 07  | 08  | 09  | 0A  | 0B  | 0C  | 0D  | 0E  | 0F  | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 1A  | 1B  | 1C  | 1D  | 1E  | 1F  |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 00  | 12  | 34  | 56  | 78  | 09  | 0A  | 0B  | 0C  | 0D  | 0E  | 0F  | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 1A  | 1B  | 1C  | 1D  | 1E  | 1F  | 00  |
| RSCI, RSCC, RSCE, OR RSCX | FUNCTION | ESCAN | RSCV ADDRESS | RETURN ADDRESS | RSCV |

**Byte (hex)**

**Contents**

00–03
Record ID:

C"RSCI" for tracing input RSCV
C"RSCC" for tracing condensed RSCV
C"RSCE" for tracing essential RSCV
C"RSCX" for tracing expanded RSCV

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Function:
1  Expand CV85 before building condensed and essential RSCVs
2  Expand CV85 only
3  Do not expand CV85 before building condensed and essential RSCVs
4  Build essential RSCV
5  Trace condensed RSCV
6  Trace essential RSCV

06
Expand Return code:
14  CV85 error
15  CV83 missing
16  CV46 missing

07
Scan Return code:
01  V46 length error
02  V2B length error
03  V80 length error
04  Unknown vector
05  Hop count not valid
07  V46 not valid
08  Subvector not valid
09  CPNAME breakdown error
12  CV2B prefix error
13  No RTP path exists
CV4680 missing
CPNAME breakdown error

08–0B
RSCV address

0C–0F
Return address

10–1F
16 bytes of RSCV

**RSC2 entry for RSCVSCAN macroinstruction (Part 2)**

**Entry:**
\[ RSC2 \]

**VIT option:**
\[ HPR \]

**Event:**
Invocation of RSCVSCAN macroinstruction

**VIT processing module:**
\[ ISTITCHP \]

This trace record is a continuation of the RSCx entry.

<table>
<thead>
<tr>
<th>00–03</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Record ID: C“RSC2”</td>
</tr>
</tbody>
</table>

| 04–1F | 28 bytes of RSCV                 |

**RSLK entry for invoking a RoCE shared lock operation**

**Entry:**
\[ RSLK \]

**VIT option:**
\[ CIA \]

**Subtrace Type:**
\[ DIO \]

**Event:**
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) 64-bit shared lock operation, as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing.

**VIT processing module:**
\[ ISTITCHP \]
This trace record is written upon completion of a shared lock operation.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'RSLK'</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Shared lock function identifier:</td>
</tr>
<tr>
<td></td>
<td>• 'C' for Recovery</td>
</tr>
<tr>
<td></td>
<td>• 'D' for Destroy</td>
</tr>
<tr>
<td></td>
<td>• 'I' for Init</td>
</tr>
<tr>
<td></td>
<td>• 'O' for Obtain</td>
</tr>
<tr>
<td></td>
<td>• 'Q' for Query</td>
</tr>
<tr>
<td></td>
<td>• 'R' for Release</td>
</tr>
<tr>
<td>06</td>
<td>Return code</td>
</tr>
<tr>
<td>07</td>
<td>Associated control block type:</td>
</tr>
<tr>
<td></td>
<td>• 'C' for connection queue control block</td>
</tr>
<tr>
<td></td>
<td>• 'E' for event queue control block</td>
</tr>
<tr>
<td></td>
<td>• 'Q' for queue pair control block</td>
</tr>
<tr>
<td>08–0F</td>
<td>Address of the control block that is associated with the shared lock</td>
</tr>
<tr>
<td>10–17</td>
<td>Shared lock address</td>
</tr>
<tr>
<td>18</td>
<td>Lock sequence number</td>
</tr>
<tr>
<td>19</td>
<td>Lock use count</td>
</tr>
<tr>
<td>1A–1B</td>
<td>Module identifier of the module that issued INTRACE</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>
RTP entry for RTP PAB dispatch

Entry:
RTP

VIT option:
HPR

Event:
Dispatch of a rapid transport protocol (RTP) PAB

VIT processing module:
ISTITCHR

Control is returned to:
ISTRPCRT

This trace record is written when an RTP PAB is dispatched. The entry contains RTP state and adaptive rate-base congestion control (ARB) information.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RTP&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Number of retries (RPN_NUM_RETRY)</td>
</tr>
<tr>
<td>06</td>
<td>RTP path switch fields:</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>1...</td>
<td>Path switch—Waiting for LU-LU session services or topology and routing services (RPN_WAIT_PSWCH)</td>
</tr>
<tr>
<td>.1...</td>
<td>Path switch—Waiting for partner (RPN_WAIT_PS_STAT)</td>
</tr>
<tr>
<td>..1...</td>
<td>Back pressure applied (RPN_BACK_PRESSURE)</td>
</tr>
<tr>
<td>...1...</td>
<td>Status pending (RPN_STATUS_PENDING)</td>
</tr>
<tr>
<td>.... 1...</td>
<td>Last message bit set in the NLP transport header (RPN_LAST_MSG)</td>
</tr>
<tr>
<td>.... .1...</td>
<td>Path switch in progress (RPN_PSWCH_STATE)</td>
</tr>
</tbody>
</table>
Modify RTP command in progress (RPN_MRT)

Allow reporting of lost data (RPN_REPORT_GAPS)

<table>
<thead>
<tr>
<th>ARB Flags:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bit</strong></td>
</tr>
<tr>
<td><strong>Meaning</strong></td>
</tr>
<tr>
<td>1... ...1</td>
</tr>
<tr>
<td>Echo pending (RPN_ECHO_PENDING)</td>
</tr>
<tr>
<td>.1... ...1</td>
</tr>
<tr>
<td>GAP pending (RPN_GAP_PENDING)</td>
</tr>
<tr>
<td>..xx ...1</td>
</tr>
<tr>
<td>Mode (ARB_MODE):</td>
</tr>
<tr>
<td>00</td>
</tr>
<tr>
<td>Green</td>
</tr>
<tr>
<td>01</td>
</tr>
<tr>
<td>Yellow</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>Red</td>
</tr>
<tr>
<td>.... xxxx</td>
</tr>
<tr>
<td>Unused; available</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Timers:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bit</strong></td>
</tr>
<tr>
<td><strong>Meaning</strong></td>
</tr>
<tr>
<td>1... ...1</td>
</tr>
<tr>
<td>Burst timer expired.</td>
</tr>
<tr>
<td>.1... ...1</td>
</tr>
<tr>
<td>Short request timer expired.</td>
</tr>
<tr>
<td>..1... ...1</td>
</tr>
<tr>
<td>Liveness timer expired.</td>
</tr>
<tr>
<td>.... 1... ...1</td>
</tr>
<tr>
<td>HPRPST path switch timer expired.</td>
</tr>
<tr>
<td>.... 1... ...1</td>
</tr>
<tr>
<td>Refifo timer expired.</td>
</tr>
<tr>
<td>.... .1... ...1</td>
</tr>
<tr>
<td>New route timer for path switch expired.</td>
</tr>
<tr>
<td>.... ..xx</td>
</tr>
<tr>
<td>Unused; available</td>
</tr>
</tbody>
</table>

| Number of retransmitted NLPs (RPN_REXMITTED_NLPS) |
14–17
Number of bytes to send in the current burst interval (RPN_BYTES_TO_SEND)

18–1B
RPNCB address

1C–1F
Request parameter header (RPH) address

**RTP2 entry for RTP PAB dispatch (Part 2)**

**Entry:**
- RTP2

**VIT option:**
- HPR

**Event:**
- Dispatch of a rapid transport protocol (RTP) PAB

**VIT processing module:**
- ISTITCHR

This trace record is a continuation of the RTP entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RTP2&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>RTP flags</td>
</tr>
<tr>
<td></td>
<td>Byte 1</td>
</tr>
<tr>
<td></td>
<td><strong>Bit</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Meaning</strong></td>
</tr>
<tr>
<td>1...</td>
<td>This RTP uses a CoS of CPSVCMG.</td>
</tr>
<tr>
<td>.1...</td>
<td>This RTP uses a CoS of RSETUP.</td>
</tr>
<tr>
<td>..1...</td>
<td>Remote node is a network node</td>
</tr>
<tr>
<td></td>
<td>(valid only for CPSVCMG RTPs).</td>
</tr>
<tr>
<td>...1...</td>
<td>Remote node is in this native subnet</td>
</tr>
<tr>
<td></td>
<td>(valid only for CPSVCMG RTPs).</td>
</tr>
<tr>
<td>.... 1...</td>
<td>Remote node is border node</td>
</tr>
<tr>
<td></td>
<td>(valid only for CPSVCMG RTPs).</td>
</tr>
<tr>
<td>.... .11.</td>
<td>Component initiating pipe termination.</td>
</tr>
<tr>
<td>.... .1...</td>
<td>Nonmobile endpoint is performing a</td>
</tr>
<tr>
<td></td>
<td>last ditch path switch attempt.</td>
</tr>
</tbody>
</table>
Byte 2

Bit  Meaning
1... ...  Waiting for Route_Setup reply during path switch state
.xx. ...  Adaptive Rate Based Algorithm Used
         00  ARB Mode Algorithm
         01  ARB Responsive Mode Algorithm
         10  ARB Progressive Mode Algorithm
...x x...  Unused; available
.... .xxx  Backpressure reason code (valid only when RPN_BACK_PRESSURE is on. See byte 6, bit 3 in the RTP entry)
B'001'  Path switch in progress
B'010'  Send queue backed up (congested)
B'011'  Storage shortage
B'100'  Stalled RTP pipe
B'101'  Wait-for-acknowledgement queue maximum reached

06–07  Number of slowdowns not honored
08–09  Last status number transmitted
0A–0B  Last status number received
0C–0D  Last status number echoed correctly by partner
0E–0F  Number of NLPs on out of sequence queue
10–11  Number of NLPs on inbound work queue
12–13  Number of NLPs on outbound work queue
14–15  Number of NLPs on waiting for acknowledgement queue
16–17  Number of NLPs on waiting to send queue
18–1B  Number of bytes on waiting for acknowledgement queue
RTP3 entry for RTP PAB dispatch (Part 3)

Entry: RTP3
VIT option: HPR
Event: Dispatch of a rapid transport protocol (RTP) PAB
VIT processing module: ISTITCHR

This trace record is a continuation of the RTP entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RTP3&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>Multinode persistent sessions (MNPS) flags:</td>
</tr>
<tr>
<td></td>
<td>Byte 1</td>
</tr>
<tr>
<td></td>
<td>Bit</td>
</tr>
<tr>
<td></td>
<td>Meaning</td>
</tr>
<tr>
<td>1...</td>
<td>This end of the RTP connection is associated with an MNPS application.</td>
</tr>
<tr>
<td>.1...</td>
<td>The partner endpoint is associated with an MNPS application.</td>
</tr>
<tr>
<td>.1...</td>
<td>MNPS endpoint path switch is in progress.</td>
</tr>
<tr>
<td>...1</td>
<td>Incoming data should be discarded until MNPS recovery is complete.</td>
</tr>
<tr>
<td>.... 1...</td>
<td>MNPS enhanced path switch processing is in progress.</td>
</tr>
<tr>
<td>.... .1...</td>
<td>This RTP connection maintains a real connection path that is different from its computed session path.</td>
</tr>
<tr>
<td>.... .1...</td>
<td>MNPS coupling facility structure data needs to be repopulated.</td>
</tr>
<tr>
<td>.... ....x</td>
<td>Unused; available</td>
</tr>
</tbody>
</table>

Byte 2

Number of NLPs sent
Bit

Meaning

1... ....
MNPS recovery is underway.

.x... ....
Unused; available

..1. ....
The recovery PAB has given permission to delete the RPNCB.

....1 ....
A request to delete the RPNCB has been queued to the recovery PAB by RTP context manager.

.... 1...
Recovery is being terminated, but RCM is waiting for completion of ALS processing before continuing.

.... .1..
This recovery PAB is chained on the ATCVT list of recovery PABs.

.... ..1.
PU termination processing is waiting for RTP connection cleanup.

.... ....1
Route setup processing is required during RTP recovery.

06–07
Unused, available

08–0B
Highest contiguous inbound NLP sequence number acknowledged by this endpoint.

0C–0F
Sequence number associated with the first NLP pending notification from TSC that the data can be acknowledged to a partner endpoint

10–13
0

14–17
0

18–1B
0

1C–1F
0

**RTPE entry for RTP error detection**

**Entry:**

RTPE

**VIT option:**

HPR

**Event:**

Error detected by rapid-transport protocol (RTP)

**VIT processing module:**

ISTITCHR

**Control is returned to:**

ISTRPCRC or ISTRPCRR

This trace record is written when rapid-transport protocol (RTP) detects an error condition, including protocol violations.
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RTPE&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Instance of the trace in the issuing module</td>
</tr>
<tr>
<td>07–0B</td>
<td>Name of the module that issued this trace entry</td>
</tr>
<tr>
<td>0C–0F</td>
<td>TSCB address containing network layer packet (NLP)</td>
</tr>
<tr>
<td>10–13</td>
<td>Sense code indicating error condition (MRPFA sense code)</td>
</tr>
<tr>
<td>14</td>
<td>Action:</td>
</tr>
<tr>
<td>01</td>
<td>RTP termination</td>
</tr>
<tr>
<td>02</td>
<td>Packet discard</td>
</tr>
<tr>
<td>03</td>
<td>Packet processed (error ignored)</td>
</tr>
</tbody>
</table>
| 15         | I—Inbound  
Partner error detected  
O—Outbound  
Origin node error detected |
| 16–17      | 0 |
| 18–1B      | RPNCB address |
| 1C–1F      | Request parameter header (RPH) address |
RTPP entry for path switch processing

Entry:
RTPP

VIT option:
HPR

Event:
Path switch

VIT processing module:
ISTITCHR

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced

This trace record is generated at various points during path switch processing.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RTPP&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>If nonzero, indicates one of the following path switch events: C - Path switch successfully completed, F - Path switch failed, O - Path switch overridden by partner, S - Path switch started</td>
</tr>
<tr>
<td>06</td>
<td>RTPP path switch 1 fields:</td>
</tr>
<tr>
<td></td>
<td>Bit</td>
</tr>
<tr>
<td></td>
<td>Meaning</td>
</tr>
<tr>
<td>1...</td>
<td>Path switch—Waiting for new route (RPN_WAIT_PS_APPN)</td>
</tr>
<tr>
<td>.1...</td>
<td>Path Switch—Waiting for partner (RPN_WAIT_PS_STAT)</td>
</tr>
<tr>
<td>..1...</td>
<td>Path switch needed (RPN_PS_NEEDED)</td>
</tr>
<tr>
<td>...1...</td>
<td>Path switch in progress (RPN_PSWCH_STATE)</td>
</tr>
<tr>
<td>.... 1...</td>
<td>Modify RTP command in progress (RPN_MRTP)</td>
</tr>
</tbody>
</table>
RTPP path switch 2 fields:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RPN_PS_TIMER</td>
</tr>
<tr>
<td>.1</td>
<td>RPN_NR_TIMER</td>
</tr>
<tr>
<td>.xx</td>
<td>RPN_PS_REASON</td>
</tr>
<tr>
<td>.xx</td>
<td>RPN_CONN_TYPE</td>
</tr>
</tbody>
</table>

RTPP path switch 3 fields:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RPN_LOCAL_MNPS</td>
</tr>
<tr>
<td>.1</td>
<td>RPN_REMOTE_MNPS</td>
</tr>
<tr>
<td>.1</td>
<td>RPN_TWOSTEP</td>
</tr>
<tr>
<td>.1</td>
<td>RPN_ENDPOINT_PS</td>
</tr>
<tr>
<td>.1</td>
<td>RPN_PSW_STARTED_MSG</td>
</tr>
<tr>
<td>.xxx</td>
<td>Unused; available</td>
</tr>
</tbody>
</table>

RPN_PS_FLAVOR
RPN_NUM_PS_RETRY
RPN_CONN_STATE
RPN_NEW_ROUTE_T (low half)
RPN_PATHSWITCH_T (low half)
Return address
RUPE opcode
18–1B
RPNCB address

1C–1F
Request parameter header (RPH) address

**RTSx entry for route setup (Part 1)**

**Entry:**
RTS

**VIT option:**
HPR

**Event:**
Route setup sent or received

**VIT processing module:**
ISTITCHR

**Control is returned to:**
ISTTSCAR, ISTTSCWS, or ISTTSC9S

This trace record is written when a FID2 route-setup record is sent or received at the DLC layer. The RTSI and RTSO entries provide, respectively, inbound and outbound information about the FID2 rapid transport protocol (RTP) route-setup signal when it is detected at the DLC layer.

---

**Byte (hex)**

**Contents**

**00–03**
Record ID:
- C"RTSI" for inbound
- C"RTSO" for outbound

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**
0

**06**
Control block ID

**07–0B**
Name of the module that issued this trace entry

**0C–0F**
TSCB address

**10–13**
0
14  TSCFLG1
15  TSCFLG2
16  TSCFLG3
17  TSCFLG4
18  TSCFLG5
19  TSCB FID type

1A–1B  
Length of data (TSCILNG)

1C–1F  
Request parameter header (RPH) address

**RTS2 entry for route setup (Part 2)**

**Entry:**

RTS2

**VIT option:**

HPR

**Event:**

Route setup sent or received

**VIT processing module:**

ISTITCHR

This trace record is a continuation of the RTS entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RTS2&quot;</td>
</tr>
<tr>
<td>04–1F</td>
<td>28 bytes of the FID2 route setup</td>
</tr>
</tbody>
</table>

**RVM entry for RVM PAB dispatch**

**Entry:**

RVM

**VIT option:**

HPR

**Event:**

Dispatch of recovery manager (RVM) PAB
VIT processing module:
ISTITCHR

Control is returned to:
ISTRVMRT

This trace record is written when a Recovery Manager PAB is dispatched. The entry contains Recovery Manager control block information.

| 00–03 | Record ID: C"RVM"
| 04    | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05    |
| 06    | Recovery Manager state
| 07    | Identifier of last FFST probe issued by this instance of Recovery Manager
| 08–0B | Recovery Manager status flags

### Bit Meaning

1... .... Another VTAM has conceded ownership of the ASB to this VTAM as a result of a takeover.

.1... .... This VTAM has claimed ownership of the ASB.

..1. ..... Recovery Manager is terminating.

...1 .... Recovery Manager has processed the request to begin termination that was sent by Configuration Services.

.... 1... Recovery Manager has incremented the ACDEB “pending recovery” session count.

.... .1... All the Recovery PABs managed by this Recovery Manager instance have completed.

.... ..1. Recovery Manager initialization is continuing.

.... ...1 The DSDCB for this Recovery Manager has been freed.
Bit

Meaning

1... ....
The takeover MNSPL for this Recovery Manager has been freed.

.1... ....
At least one set of session information blocks was incomplete.

.1... ....
No APPC LME data was available for this application.

...1 11...
ENCRTYPE encoding for this application.

.1... ....
The forced takeover request sent for this application has been accepted by the current owning node.

.1... x

0

Bit

Meaning

1... ....
Application is also a member of a generic resource.

.1... ....
Need to send an XCF takeover request.

.1... ....
Takeover reply has been received.

...1 ....
APPC list reading is required.

.... 1...
This takeover request is a forced takeover.

.... .xxx

0

0B

0

0C

Planned takeover status

0D–0F

0

10

Recovery Manager interface flags

Bit

Meaning

1... ....
Recovery Manager is still accepting work to do from other components.

.1... ....
One or more Recovery PABs have completed work.

.1... ....
Recovery Manager should terminate recovery processing at the next dispatch.

...1 ....
Initial dispatch of PAB.
11–13
Number of work elements queued to Recovery Manager

14–15
Number of Recovery PABs (RPABs) that are in the process of recovery

16–17
Number of Recovery PABs (RPABs) that have completed their recovery processing

18–1B
0

1C–1F
Request parameter header (RPH) address

**RXMT entry for adding PDU to RPN CB queue**

**Entry:**
RXMT

**VIT option:**
HPR

**Event:**
A PDU is transmitted.

**VIT processing module:**
ISTITCHR

**Control is returned to:**
ISTRPRCX

This trace record is written when a PDU is added to the RPN CB queue for data waiting to be retransmitted.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;RXMT&quot;</td>
</tr>
<tr>
<td>04</td>
<td>Primary address space ID</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Number of acknowledged bytes</td>
</tr>
<tr>
<td>07</td>
<td>Flags</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) record descriptions 609
Available

Outbound CV29 processed

Last message indicator

First transmission

Retransmit

Retransmitted

TSOP failed to get storage for token.

Need to free RPN_OUTBOUND_CV29_Q entry.

Send was built because of an MNPS recovery.

08–0B
ISTSND address

0C–0F
Segmentation address or 0

10–13
PIU length

14–17
Sequence number

18–1B
NCB address

1C–1F
RPH address

SBAL entry for Storage Block Address Lists (Part 1)

Entry:
SBAL

VIT option:
CIA

Event:
Capture SBAL

VIT processing module:
ISTITCOD

Control is returned to
ISTLLCHI, ISTLLCIE, ISTLLCWC, ISTLLCWI

This trace record is written when the OSA-Express QDIO or HiperSockets Storage Block Address List is altered. Specifically, SBAL records will follow or be preceded by an SLSB record. The SLSB shows the contents of all 128 status bytes while the SBAL record shows the contents of a single SBAL.
Byte (hex)

Contents

00–03
Record ID: C"SBAL"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
SBAL index

06
Direction:
• C"I" for inbound (read queue)
• C"O" for outbound (write queue)

07
Write queue priority/Read queue identifier
• if byte 6 is C"I"
  – Read queue identifier
• if byte 6 is C"O"
  – Write queue priority

08–0B
SBALE_SBALF_0, SBALE_SBALF_1, SBALE_SBALF_14, and SBALE_SBALF_15 respectively.
Note: Nonzero SBALE_SBALF_14 indicates SBAL error.

0C–0F
Contents of first SBALE:
Format:

Bit 0
SBALE_Last_Entry

Bit 1
SBALE_Next_Contig

Bits 2–3
SBALE–FT (fragment type)
"00"B
Not fragmented
"01"B
First fragment
"10"B
Middle fragment

"11"B
Last fragment

Bits 4–19
SBALE_length

Bits 20–31
SBALE_Addr page offset

10–13
Contents of continuation SBALEs (if applicable).

14–17
Contents of continuation SBALEs (if applicable).

18–1B
Contents of continuation SBALEs (if applicable).

1C–1F
Request parameter header (RPH) address.

SBA2 entry for Storage Block Address Lists (Part 2)

Entry:
SBA2

VIT option:
CIA

Event:
Capture SBAL

VIT processing module:
ISTITCOD

This trace record is written subsequent to SBAL or SBA2 entries when the previous entry did not contain enough room to capture up to and including the last active SBALE.

| Byte (hex) Content |
|-------------------|-----------------|
| 00–03             | Record ID: C"SBA2" |
| 04–07             | Contents of continuation SBALEs (see SBAL record for SBALE format) |
| 08–0B             | Contents of continuation SBALEs (if applicable) |
| 0C–0F             | Contents of continuation SBALEs (if applicable) |
SCHD entry for schedule of PAB dispatch

Entry:
  SCHD
VIT option:
  PSS
Event:
  Schedule of PAB dispatch
VIT processing module:
  ISTRACPS
Control is returned to:
  ISTAPCTS

This trace record shows the schedule of a PAB to be dispatched and the last work element, if any, that was queued to the PAB. This entry should be followed shortly by a DSP entry, representing the PSS dispatch of this PAB.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;SCHD&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Status</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>xx..</td>
<td>. . . .</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td>..1.</td>
<td>. . . .</td>
</tr>
<tr>
<td></td>
<td>TPSCHED unconditional</td>
</tr>
<tr>
<td>..x</td>
<td>. . . .</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td>.... 1...</td>
<td>. . . .</td>
</tr>
<tr>
<td></td>
<td>Registers are saved in the RPH control block.</td>
</tr>
</tbody>
</table>
PAB work element queue gate bit (PABWEQG).

PAB change bit (PABCHNG) = 0 if the PAB is not scheduled to run.

PABCHNG=1 if the PAB is scheduled to run.

06

PAB flag field (PABFLAGS)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>PAB is unconditionally scheduled.</td>
</tr>
<tr>
<td>.1..</td>
<td>PAB closedown is in progress.</td>
</tr>
<tr>
<td>..1.</td>
<td>PAB is synchronous.</td>
</tr>
<tr>
<td>...1</td>
<td>PAB extension is present.</td>
</tr>
<tr>
<td>....1</td>
<td>Do not dequeue work element.</td>
</tr>
<tr>
<td>....1</td>
<td>Do not detach the RPH.</td>
</tr>
<tr>
<td>.... ..1.</td>
<td>Indicates a very extended PAB.</td>
</tr>
<tr>
<td>.... ...1</td>
<td>Indicates a slightly extended PAB.</td>
</tr>
</tbody>
</table>

07

PAB flag field (PABFLGS1)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>Switch the PST address of this PAB’s major control block to the new PST address contained in DYPNW PST.</td>
</tr>
<tr>
<td>.1..</td>
<td>This PAB has a data space extension.</td>
</tr>
<tr>
<td>..1.</td>
<td>This PAB’s major control block is an FMCB.</td>
</tr>
<tr>
<td>...1</td>
<td>PAB can be referenced in PSW disable mode.</td>
</tr>
<tr>
<td>....1</td>
<td>PAB is persistent.</td>
</tr>
<tr>
<td>.... ...1.</td>
<td>APSTERM/APSINIT FMCB during PAB dispatch.</td>
</tr>
<tr>
<td>.... ...xx</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

08–0B

PST address associated with PAB
**SIGA entry for Signal Adapter event**

**Entry:**
- **SIGA**

**VIT option:**
- **CIA**

**Event:**
- Signal Adapter instruction

**VIT processing module:**
- ISTITCOD

**Control is returned to**
- ISTLCHI, ISTLCLIE, ISTLCLRB, ISTLCLCW, ISTLCLWI

This trace record is written immediately subsequent to the issuance of the Signal Adapter instruction. The Signal Adapter instruction is primarily used to notify the OSA-Express QDIO or HiperSockets adapter that the status of one or more SBALs has changed (for example: data is available for a write operation). SIGA is similar in function to SIO/SSCH. SIGA is also used to communicate with the VM operating system when z/OS is executing in a specific type of virtual machine.

| 00–03 | Record ID: C“SIGA” |
| 04     | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'. |
SIGA function:
• C"M" for SIGA-mw (SIGA multiple write)
• C"R" for SIGA-r (SIGA read)
• C"S" for SIGA-s (SIGA synchronize)
• C"W" for SIGA-w (SIGA write)

Left nibble: Condition code from the SIGA instruction.
Right nibble: Program mask.

Device address in EBCDIC (as specified in the TRL deck)

DINCB address

Subsystem Identification Word: X'0001' followed by the Subchannel Number that corresponds to the device address.

Queue indices bit mask (1):
• SIGA multiple write: Specifies write queues for which the Storage List Status Bytes have been updated
• SIGA read: Specifies read queues for which the Storage List Status Bytes are to be refreshed.
• SIGA sync: Specifies write queues for which the Storage List Status Bytes are to be refreshed.
• SIGA write: Specifies write queues for which the Storage List Status Bytes have been updated.

Queue indices bit mask (2):
• SIGA multiple write: X'0000'
• SIGA read: X'0000'
• SIGA sync: Specifies read queues for which the Storage List Status Bytes are to be refreshed.
• SIGA write: X'0000'

Module identifier for the module that issued the INTRACE

Request parameter header (RPH) address

SIOx entry for Start I/O

Entry:
SIOD, SIOI, SIOL, SIOT, or SIOX

VIT option:
CIO

Event:
Start I/O
VIT processing module:
ISTRACCI

Control is returned to:
ISTTSCIO or ISTTSCLC

This trace record is written when a Start I/O SVC is issued.

- SIOD is generated for OSA-Express QDIO or HiperSockets adapters.
- SIOI is generated for communication controllers and local SNA cluster controllers.
- SIOL is generated for local non-SNA cluster controllers.
- SIOT is generated for a TCP/IP DLC connection.
- SIOX is generated for channel-to-channel-attached hosts and channel-attached IBM 3172 Interconnect Controllers.

Eventually, an INTx or PCIx entry follows the SIO trace record. Use the CUA field to correlate the entries.

See z/OS Communications Server: SNA Data Areas Volume 1 for a description of the NCB fields.

For record types with suffix D, I, X, or T, the CIO events are also captured within the NCB (pointed to by NCBCIOMV). The NCB trace table is mapped by NCBCIOAR.

### SIOD and SIOT mapping and field descriptions

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIOT</td>
<td>I</td>
<td>D</td>
<td>S</td>
<td>T</td>
<td>A</td>
<td>E</td>
<td>CUA</td>
<td>DEVICE</td>
<td>NCB</td>
<td>ADDRESS</td>
<td>FLAG</td>
<td>BYTES</td>
<td>CAW</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00–03**
Record ID:

"SIOD" for DINCB
"SIOT" for RWN CB

**04**
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than 'X'FF'.

**05**
State

**06**
0

**07**
For SIOD, 0
For SIOT, type:

- For the CDLC DLC: C
- For the HYPERchannel DLC: H
- For the LCS DLC: L
- For the CLAW DLC: W
- For the CTC DLC: X
- For the Hyperchannel DLC: H
08–0B
Channel device name in EBCDIC (either a CUA or device number)

0C–0F
NCB address

10–13
Flag bytes (NCBFLAGS)

14–17
Virtual channel address word (NCBCAW)

18–1F
First channel command word (CCW)

**SIOI mapping and field descriptions**

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>• &quot;SIOI&quot; for ICNCB</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Link station state</td>
</tr>
<tr>
<td>06</td>
<td>Write start operation code</td>
</tr>
<tr>
<td>07</td>
<td>Write start operation code</td>
</tr>
<tr>
<td>08–0B</td>
<td>Channel device name in EBCDIC (either a CUA or device number)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>NCB address</td>
</tr>
<tr>
<td>10–13</td>
<td>Flag bytes (NCBFLAGS)</td>
</tr>
<tr>
<td>14–17</td>
<td>Virtual channel address word (NCBCAW)</td>
</tr>
</tbody>
</table>
| 18–1B      | If link is active, first WRITE CCW and READ START operation code (06 will be WRITE START operation code), or 0  
|            | If link is not active, first channel command word (CCW) |
| 1C         | Read start operation code |
SIOL mapping and field descriptions

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: “SIOL” for LDNCB</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>State</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Channel device name in EBCDIC (either a CUA or device number)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>NCB address</td>
</tr>
<tr>
<td>10–13</td>
<td>Flag bytes (NCBFLAGS)</td>
</tr>
<tr>
<td>14–17</td>
<td>Virtual channel address word (NCBCAW)</td>
</tr>
<tr>
<td>18–1B</td>
<td>If link is active and select operation; CCW and operation code, or 0. If link is not active, first channel command word (CCW)</td>
</tr>
<tr>
<td>1C</td>
<td>Code</td>
</tr>
<tr>
<td>1D–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

SIOX mapping and field descriptions

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: “SIOL” for LDNCB</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>State</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Channel device name in EBCDIC (either a CUA or device number)</td>
</tr>
<tr>
<td>0C–0F</td>
<td>NCB address</td>
</tr>
<tr>
<td>10–13</td>
<td>Flag bytes (NCBFLAGS)</td>
</tr>
<tr>
<td>14–17</td>
<td>Virtual channel address word (NCBCAW)</td>
</tr>
<tr>
<td>18–1B</td>
<td>If link is active and select operation; CCW and operation code, or 0. If link is not active, first channel command word (CCW)</td>
</tr>
<tr>
<td>1C</td>
<td>Code</td>
</tr>
<tr>
<td>1D–1F</td>
<td>0</td>
</tr>
</tbody>
</table>
**Byte (hex)**

**Contents**

00–03

Record ID: "SIOX" for XCNCB

04

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05

State

06–07

0

08–0B

Channel device name in EBCDIC (either a CUA or device number)

0C–0F

NCB address

10–13

Flag bytes (NCBFLAGS)

14–17

Virtual channel address word (NCBCAW)

18–1F

Output control area (status information)

**SIO2 entry for Start I/O (Part 2)**

Entry:

SIO2

VIT option:

CIO

Event:

Start I/O (Part 2)

VIT processing module:

ISTRACCI

This trace record is a continuation of the SIOI, SIOL, SIOT, and SIOX entries.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;SIO2&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>CPNCB_Index_Word</td>
</tr>
<tr>
<td>06–07</td>
<td>0</td>
</tr>
<tr>
<td>08–09</td>
<td>Module ID</td>
</tr>
</tbody>
</table>
SLSB entry for OSA-Express QDIO or HiperSockets Storage List Status Bytes (Part 1)

Entry:
SLSB

VIT option:
CIA

Event:
Capture SLSBs

VIT processing module:
ISTITCOD

Control is returned to
ISTLLCHI, ISTLLCIE, ISTLLCWC, ISTLLCWI, ISTITCOD

This trace record is written when the OSA-Express QDIO or HiperSockets Storage List Status Bytes are altered. Specifically, this record is written:

- Upon completion of the construction of write structures for a given write queue and immediately preceding the issuance of the Signal Adapter instruction.
- Upon detection of a write completion during OSA-Express QDIO or HiperSockets timer processing.
- Upon read completion (PCI interrupt processing).
  - Prior to read completion processing

  Note: The PCID record is also used to capture the SLSBs. If the PCID record captures all 128 status bytes, the SLSB record is skipped. If the PCID record does not capture all 128 status bytes, the SLSB record does not start where PCID left off but instead replicates the count/status pairs in the PCID.

- Upon a priority queue entering a congested state.

| 00–03 | Record ID: C“SLSB” |

0A–19
First 16 bytes of MPC transmit block data, or 0, or device-dependent data

1A–1F
0
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Congestion state; the value X'04' indicates that a write priority queue has entered a congested state.

Direction:
• C"I" for inbound (read queue)
• C"O" for outbound (write queue)

Write queue priority/Read queue identifier
• if byte 6 is C"I"
  – Read queue identifier
• if byte 6 is C"O"
  – Write queue priority

Device address in EBCDIC (as specified in the TRL deck).

DINCB address

Contents of DINCB_Q_Data_Index_Next
• Next Empty Read for read SLSBs
• Next Write to Complete for write SLSBs

Contents of DINCB_Q_Data_Index_Last
• Last Empty Read for read SLSBs
• Last Write to Complete for write SLSBs

First of up to 128 count/status pairs
Starting at SLSB(1), these fields represent the number of consecutive SLSBs containing the same status, and what the status value is. See SLS_SLSB_Status constants for status values.

Continuation count/status pairs (if applicable)

Continuation count/status pairs (if applicable)

Continuation count/status pairs (if applicable)

Continuation count/status pairs (if applicable)

Continuation count/status pairs (if applicable)

Request parameter header (RPH) address

**SLS2 entry for OSA-Express QDIO or HiperSockets Storage List Status Bytes (Part 2)**

Entry:

VIT option:

CIA
Event:
Capture SLSBs

VIT processing module:
ISTITCOD

This trace record is written when the previous SLSB or SLS2 entry did not contain enough room for all the count/status pairs needed to represent the entire SLSB.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;SLS2&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>Continuation count/status pairs</td>
</tr>
<tr>
<td>06–07</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>08–09</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>0A–0B</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>0C–0D</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>0E–0F</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>10–11</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>12–13</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>14–15</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>16–17</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>18–19</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>1A–1B</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>1C–1D</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
<tr>
<td>1E–1F</td>
<td>Continuation count/status pairs (if applicable)</td>
</tr>
</tbody>
</table>
SPT entry for SPT macro invoked (Part 1)

Entry:
SPT

VIT option:
SSCP

Event:
SPT macro invocation

VIT processing module:
ISTRACSC

Control is returned to:
ISTCPMSP

This trace record is written when partner LU mapping information in the generic resource coupling facility structure is created, deleted, or changed using the SPT macroinstruction.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;SPT&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Return code from the SPT macro</td>
</tr>
<tr>
<td>X'00'</td>
<td>Request completed successfully. This is an existing USERVAR return code.</td>
</tr>
<tr>
<td>X'01'</td>
<td>Request completed successfully and local data was returned. This is returned from SPT if SEARCH is CF&amp;LOCAL only.</td>
</tr>
<tr>
<td>X'04'</td>
<td>Resource specified could not be located. This is an existing USERVAR return code.</td>
</tr>
<tr>
<td>X'51'</td>
<td>Request failed because of APPC restrictions. This is an existing USERVAR return code.</td>
</tr>
<tr>
<td>X'52'</td>
<td>Request failed because of a parameter specification that was not valid. This is an existing USERVAR return code.</td>
</tr>
<tr>
<td>X'53'</td>
<td>The function requested could not be completed because of a storage allocation failure. This is an existing USERVAR return code.</td>
</tr>
<tr>
<td>X'54'</td>
<td>The real instance of user variable to be deleted could not be located. This is an existing USERVAR return code.</td>
</tr>
<tr>
<td>X'55'</td>
<td>A conflict was found because of NETIDs.</td>
</tr>
</tbody>
</table>
A conflict was found because of CPNAMEs.

VTAM is halting.

ENDAFFINITY was issued for a session that was VTAM owned.

RSCTYPE value conflicted between USERVAR and GR.

An attempt to change the generic name for a given application failed because SPTEs pertaining to the previous generic name still exist.

A session pair could not be found in an SPT entry.

An attempt to change the generic name for a given application failed because either (1) the existing generic name was never deleted, or (2) because SPTEs pertaining to the existing generic name still exist.

An attempt to repopulate the generic resource coupling facility structure has failed because the local data being used for repopulation is backlevel compared to data already in the structure.

An attempt to repopulate the generic resource coupling facility structure has succeeded but backlevel data has been identified in the structure and additional structure cleanup may be required.

An attempt to increment the session count for an SPTE failed because the name type of the SPTE did not match what was expected.

Request failed because of TSO mismatch.

An attempt to register a generic resource with the workload manager failed because a STOKEN that was not valid was used.

A request made against the generic resources coupling facility structure will complete synchronously.

An attempt to update information in the generic resources coupling facility structure failed because that data had changed since it was last read. The data should be reread and then modified again.

An attempt to access the generic resources coupling facility structure failed for an unexpected reason.

There is currently no connection to the generic resources coupling facility structure.

The buffers provided for reading data from the generic resources coupling facility structure were insufficient for buffering all the data associated with the list entry being read. No data (adjunct or element) is returned.

A read from the generic resources coupling facility structure failed because the requested data could not be found in that structure.
X'A6'  
Data could not be added to the generic resources coupling facility structure because there is insufficient storage in the generic resources coupling facility structure to hold it.

06  
Reason macro invoked:

X'01'  
Find an SPT entry.

X'02'  
Add a session pair to the SPT entry.

X'03'  
Decrement the session count for a session pair.

X'04'  
Increment the session count for a session pair.

X'05'  
End the affinity between the generic resource and its partner LU.

X'06'  
Update an affinity between the generic resource and its partner LU.

X'07'  
Delete a session pair in an SPT entry.

X'08'  
Free local SPT entry that was read from the coupling facility structure.

X'09'  
Create a local copy of an SPT entry.

07  
Flags:

X... X...  
Name type

B"0"  
Real names only

B"1"  
Generic names allowed

.1..  
This SPT will persist until the application issues the CHANGE OPTCD=ENDAFFIN macroinstruction. See z/OS Communications Server: SNA Programming for additional information.

.X. X...  
Ownership indicator

B"0"  
VTAM-owned

B"1"  
Application-owned

.X. X...  
Multinode persistent session indicator

B"0"  
SPT entry is not associated with a multinode persistent session application program

B"1"  
SPT entry is associated with a multinode persistent session application program

... X...  
Affinity creator.
The affinity was created during a previous request.

VTAM selected the instance of the generic resource.

The MVS WLM selected the instance of the generic resource.

The Generic Resources Exit selected the instance of the generic resource.

This affinity is for an LU 6.2 session that is not a user sync point or a limited resource.

Generic resource name

Address of the issuer of the SPT macro

Application program network name

Request parameter header (RPH) address

SPT2 entry for SPT macro invoked (Part 2)

Entry:
SPT2

VIT option:
SSCP

Event:
SPT macro invocation

VIT processing module:
ISTRACSC

This trace record is a continuation of the SPT entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;SPT2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Number of session information blocks (SIBs) that have been created for this SPT entry</td>
</tr>
<tr>
<td>08–0F</td>
<td>PCID if associated with a session, otherwise 0</td>
</tr>
<tr>
<td>10–17</td>
<td>Partner NETID</td>
</tr>
</tbody>
</table>

**VTAM internal trace (VIT) record descriptions 627**
SRBD entry for SRB dispatch

Entry:
SRBD

VIT option:
PSS

Event:
SRB dispatch

VIT processing module:
ISTRACPS

Control is returned to:
ISTAPCAD

This trace record provides information when PSS is dispatched under an SRB.

Note: For this trace record to be created, in addition to specifying the PSS option, you must specify SRB or BOTH on the PSS TRACE start option.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;SRBD&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>PST address</td>
</tr>
<tr>
<td>0C–0F</td>
<td>PST LIFO asynchronous dispatchable queue (PSTADSP)</td>
</tr>
<tr>
<td>10–13</td>
<td>PST FIFO asynchronous dispatchable queue (PSTALIST)</td>
</tr>
<tr>
<td>14–17</td>
<td>PST asynchronous nondispatchable queue (PSTANDSP)</td>
</tr>
<tr>
<td>18–1F</td>
<td>TCB address (PSTTCBA)</td>
</tr>
</tbody>
</table>
SRBX entry for SRB exit

Entry:
SRBX

VIT option:
PSS

Event:
SRB exit

VIT processing module:
ISTRACPS

Control is returned to:
ISTAPCSX, ISTAPCTX, or ISTAPCAD

This trace record provides information when PSS exits an SRB. For this trace record to be created, you must specify the PSS VIT option, and you must specify SRB or BOTH on the PSSTRACE start option.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 00–03      | Record ID: C"SRBX"
| 04         | ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
| 05         | Reason code for SRB exit
| 06–07      | 0
| 08–0B      | PST address
| 0C–0F      | PST LIFO asynchronous dispatchable queue (PSTADSP)
| 10–13      | PST FIFO asynchronous dispatchable queue (PSTLIST)
| 14–17      | New ASCB address
| 18–1B      | Old ASCB address
| 1C–1F      | Address of the module issuing the SRBX trace event
SRTx entry for SRTADD, SRTCHG, SRTDEL, SRTFIND macros

**Entry:**
SRTA, SRTC, SRTD, or SRTF

**VIT option:**
NRM

**Event:**
SRTADD, SRTCHG, SRTDEL, SRTFIND macros

**VIT processing module:**
ISTRACNR

**Control is returned to:**
ISTNRCSA for SRTADD, ISTNRCSD for SRTCHG, SRTDEL, and SRTFIND

This trace record contains information about an SRTADD, SRTCHG, SRTDEL, or SRTFIND macroinstruction issued by a VTAM module.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;SRTA&quot; (SRTADD), C&quot;SRTC&quot;(SRTCHG), C&quot;SRTD&quot; (SRTDEL), or C&quot;SRTF&quot; (SRTFIND)</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Return code. See the information about the SRT control block in z/OS Communications Server: SNA Data Areas Volume 1.</td>
</tr>
<tr>
<td>06</td>
<td>0 or SRTUSELN from the SRTE. This is a 4-byte field in the SRTE and will be set to X'FF' in the SRTx if SRTUSELN is greater than X'FF'. This field will be 0 if the return code is nonzero, or if the SRT type does not use SRTUSELN as a use count.</td>
</tr>
<tr>
<td>07</td>
<td>Type of SRT entry. For the types, see the information about the SRT control block in z/OS Communications Server: SNA Data Areas Volume 1.</td>
</tr>
<tr>
<td>08–0F</td>
<td>Network name or network address (padded on left with zeros) passed to the hashing algorithm</td>
</tr>
<tr>
<td>10–13</td>
<td>Address of the issuer of the SRTADD, SRTCHG, SRTDEL, or SRTFIND macroinstruction</td>
</tr>
<tr>
<td>14–17</td>
<td>Address of SRT entry for the resource</td>
</tr>
<tr>
<td>18–1F</td>
<td>Network ID of the resource</td>
</tr>
</tbody>
</table>
SRT2 entry for the SRTADD macro

Entry:
  SRT2
VIT option:
  NRM
Event:
  SRTADD
VIT processing module:
  ISTRACNR

This trace record is a continuation of the SRTADD entry. It is written after an SRTADD failure if the
duplicate SRTE is contained in or points to an RDTE.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;SRT2&quot;</td>
</tr>
<tr>
<td>04</td>
<td>SRTUSELN from the duplicate SRTE; this is a 4-byte field in the SRTE and will be set to X'FF' in the SRT2 if SRTUSELN is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>The first bit is SRTSPECE; the other seven are not used.</td>
</tr>
<tr>
<td>06</td>
<td>RPRENTRY of the duplicate RDTE</td>
</tr>
<tr>
<td>07</td>
<td>SRTTYP from the duplicate SRTE</td>
</tr>
<tr>
<td>X'00'</td>
<td>RDTE</td>
</tr>
<tr>
<td>X'03'</td>
<td>Shadow</td>
</tr>
<tr>
<td>X'09'</td>
<td>Alias name</td>
</tr>
<tr>
<td>X'0A'</td>
<td>Network address</td>
</tr>
<tr>
<td>X'FF'</td>
<td>SRTE</td>
</tr>
<tr>
<td>08–0F</td>
<td>RPRNAME of the duplicate RDTE</td>
</tr>
<tr>
<td>10–13</td>
<td>SRTDATA from the duplicate SRTE or 0</td>
</tr>
</tbody>
</table>
14–17
Not used

18–1F
Network ID of the duplicate RDTE. If RPRENTRY is RPRENTRC, it is RCDNETID; otherwise, it is ATCNETID.

TGM entry for APPN TG management (Part 1)

Entry:
TGM

VIT option:
SSCP

Event:
APPN TG management event

VIT processing module:
ISTRACSC

Control is returned to:
ISTATCTR

This record is generated when one of the following APPN TG management events has occurred:

• Add an ATGB to an ATGB chain.
• Remove an ATGB from an ATGB chain.
• Find an ATGB.
• Activate a TG number.
• Deactivate a TG number.
• Negotiate a TG number.

If TGFIND fails to locate an ATGB, a trace entry will still be recorded.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;TGM&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Type field</td>
</tr>
</tbody>
</table>

Code

Meaning

C
Activate transmission group number (TGN) or APPN transmission group block (ATGB)

D
Deactivate TGN or ATGB

A
Add an ATGB
R
Remove an ATGB
F
Find an ATGB
N
Negotiate a TGN

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
APPN TG Management function values
• Network node specification
  0: Not a network node
  1: Network node
• RU being processed
  0: Request Contact
  1: Contacted
• TG number coded on macroinstruction
  0: Not specified
  1: Specified
• Display CV47
• APPN connection pending, not reported to topology (LIMBO) specification
  0: No
  1: Yes
• Vector format
  0: Key-length format
  1: Length-type format
• Assigned chain specification
  0: Assigned not specified
  1: Assigned specified
• Unassigned chain specification
  0: Unassigned not specified
  1: Unassigned specified

06
The calling module's return code. 0 if no return address is provided.

07
Unused available

08–0B
Adjacent control point (ADJCP) address

0C
Dynamic indicators received in XID3 exchange (ADJDNFL)

0D
Number of active connections controlled (ADJCONCT), last digit

0E
Connections controlled by the ADJCP (ADJCPCT), last digit

0F
Transmission group number state map (ADJSTMP), indexed by TGN
10–17
CPNAME of the adjacent node

18–1B
Calling module name

1C–1F
Return address

TGM2 entry for APPN TG management (Part 2)

Entry:
TGM2

VIT option:
SSCP

Event:
APPN TG management event (Part 2)

VIT processing module:
ISTRACSC

This trace record is a continuation of the TGM trace record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–07</td>
<td>Overlay 1</td>
</tr>
</tbody>
</table>
|            | • 00-03 Record ID: C"TGM2"
|            | • 04-07 PU of current ATGB |
|            | Overlay 2 |
|            | • 00-03 Record ID: C"TGM2"
|            | • 04-06: Unused, available |
|            | • 07: Received TGN |
| 08–0B      | ATGB address |
| 0C         | ATGTGNUM |
| 0D         | ATGTOPR |
| 0E         | ATGDYNFL |
| 0F         | ATGPOS |
| 10–13      | ATGPUPTR: Top ATGB in the chain |
14–17
   ATGNXTPT: Top ATGB in the chain

18–1B
   ATGPUPTR: Next ATGB in the chain

1C–1F
   ATGNXTPT: Next ATGB in the chain

TGVC entry for Transmission Group Vector (Part 1)

Entry:
   TGVC

VIT option:
   SSCP

Subtrace Type:
   TGVC

Event:
   Request Route, Recompute Route, Request TG Vectors, or Cache Data message

VIT processing module:
   ISTITCTG

Control is returned to:
   Module invoking the INTRACE TYPE(TGVC) macro

This record is generated when the subtrace TGVC under SSCP trace option is active and a TRS Request Route, Recompute Route, Request TG Vectors, or SSL Cache Data message is requested.

Note: Because of the potentially large amounts of data contained in the TG Vectors, it is not recommended that this function be turned on, except for problem diagnosis. If this function is used, it should be turned on only for the time required to generate the necessary documentation and then turned off.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;TGVC&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ASID is the primary address space ID</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
</tbody>
</table>
06
Trace type
- C"P" PLU
- C"S" SLU
- C"U" Unknown

07
Continuation trace
- C"F" First set of entries
- C"C" Continuation set of entries

08–0F
PCID

10–1F
TG Vector data

TGV2 entry for Transmission Group Vector (Part 2)

Entry:
TGV2

VIT option:
SSCP

Subtrace Type:
TGVC

Event:
Request Route, Recompute Route, Request TG Vectors, or Cache Data message

VIT processing module:
ISTITCTG

This record is a continuation of the TGVC trace record.

Note: If the complete CV46/CV47 pair does not fit in the remaining TGV2 records, another TGVC/TGV2 set is generated.

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

TGV2

TG VECTOR DATA

Byte (hex)
Contents

00–03
Record ID: C"TGV2"

04–1F
TG Vector Data

TOD entry for time of day

Entry:
TOD

VIT option:
None
Event:
Time-of-day snapshot at or near the top of every other VIT page

VIT processing module:
ISTRACOT

Control is returned to:
Caller of ISTRACOT

This entry is not associated with any VIT options. It is recorded as close as practical to the top of every other page of the internal VIT.

<table>
<thead>
<tr>
<th>Bytes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: ‘C’TOD’</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X’FF’.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06-07</td>
<td>The CPU ID of the host processor in which the trace is run</td>
</tr>
<tr>
<td>08-0F</td>
<td>Time-of-day clock value (TOD format) when this trace record is written.</td>
</tr>
<tr>
<td>10-17</td>
<td>0</td>
</tr>
<tr>
<td>18-1F</td>
<td>VFSCA_LOST_RECORD_COUNT</td>
</tr>
</tbody>
</table>

TOKx entry for token collection

Entry:
TOKA or TOKD

VIT option:
CIA

Event:
Addition or deletion to token collection

VIT processing module:
ISTRACCI

Control is returned to:
Module invoking the INTRACE macroinstruction that caused the record to be produced

The TOKA trace record is written when a token is added to one of the token collections that are maintained by Registration Manager. The TOKD trace record is written when a token is deleted from one of the token collections that are maintained by Registration Manager.
Byte (hex)

Contents

00–03
Record ID:

• C"TOKA" for addition of token
• C"TOKD" for deletion of token

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
0

06
0

07–0B
Module eyecatcher

0C–0F
PTOKEN (Provider = D)

10–13
0

14–17
Token of object being added or deleted (filter object)

18–1B
Address of caller

1C–1F
Request parameter header (RPH) address.

**TOPN entry for node modified (Part 1)**

Entry:
TOPN

VIT option:
SSCP

Event:
Topology node entry modified

VIT processing module:
ISTITCST

Control is returned to:
Module invoking the INTRACE TYPE(TOPN) macro

This record is generated when a node entry in the topology database is created or modified.
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;TOPN&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Action performed on Topology Database:</td>
</tr>
<tr>
<td></td>
<td><strong>Code</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Meaning</strong></td>
</tr>
<tr>
<td>C&quot;A&quot;</td>
<td>Add node record</td>
</tr>
<tr>
<td>C&quot;D&quot;</td>
<td>Delete node record</td>
</tr>
<tr>
<td>C&quot;U&quot;</td>
<td>Update node record</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>CP name of the node represented by this node record</td>
</tr>
<tr>
<td>10–13</td>
<td>Return address of the module that changed the Topology Database</td>
</tr>
<tr>
<td>14–17</td>
<td>Node header pointer</td>
</tr>
<tr>
<td>18–1A</td>
<td>0</td>
</tr>
<tr>
<td>1B</td>
<td>Flag bytes as follows:</td>
</tr>
<tr>
<td></td>
<td><strong>Bit</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Meaning</strong></td>
</tr>
<tr>
<td>11</td>
<td>Node type</td>
</tr>
<tr>
<td>B&quot;00&quot;</td>
<td>End node</td>
</tr>
<tr>
<td>B&quot;01&quot;</td>
<td>Network node</td>
</tr>
</tbody>
</table>
Virtual routing node

Indicate whether this node is nonnative

Not used

Request parameter header (RPH) address

TPN2 entry for node modified (Part 2)

Entry:
   TPN2

VIT option:
   SSCP

Event:
   Topology node entry modified

VIT processing module:
   ISTITCST

This record is a continuation of the TOPN trace record.

| 00 01 02 03 04 05 06 07 08 09 A B 0 C D E 0 F 10 11 12 13 14 15 16 17 18 19 A B 1 C D E F |
| TPN2 | NODE RECORD POINTER | RESOURCE SEQUENCE NUMBER | V4580 FLAGS | REASON | 0 | NETWORK ID | TIME | 0 |

Byte (hex)

Contents

00–03
   Record ID: C"TPN2"

04–07
   Node record pointer

08–0B
   Current resource sequence number for the node

0C–0E
   Node characteristics control vector flags (V4580_FLAGS)

Bit

Meaning

1... ....
   Node congested

..1... ....
   Intermediate routing resources depleted

...1... ....
   Endpoint resources depleted
TOPT entry for TG modified (Part 1)

Entry:
   TOPT

VIT option:
   SSCP

Event:
   Topology TG entry modified

VIT processing module:
   ISTITCST

Control is returned to:
   Module invoking the INTRACE TYPE(TOPT) macro.

This record is generated when a TG entry in the topology database is created or modified.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;TOPT&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
</tbody>
</table>
Action performed on Topology Database:

<table>
<thead>
<tr>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>C&quot;A&quot;</td>
<td>Add TG record</td>
</tr>
<tr>
<td>C&quot;D&quot;</td>
<td>Delete TG record</td>
</tr>
<tr>
<td>C&quot;U&quot;</td>
<td>Update TG record</td>
</tr>
</tbody>
</table>

TG number

CP name of the TG origin node

Return address of the module that changed the Topology Database

CP name of the TG destination node

Request parameter header (RPH) address

TPT2 entry for TG modified (Part 2)

Entry:
TPT2

VIT option:
SSCP

Event:
Topology node entry modified

VIT processing module:
ISTITCST

This record is a continuation of the TOPN trace record.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Record ID: C&quot;TPT2&quot;</td>
</tr>
</tbody>
</table>
04–07
TG record pointer

08–0B
Current resource sequence number for the TG

0C
TG characteristics status flags (V47STAT)

0D
TG descriptor flags (V4680FLAGS)

0E
TG information flags

Bit  Meaning
-----  -------
111. . . .  TG type
  B"000"  Endpoint TG
  B"001"  Intermediate routing TG
  B"010"  Interchange TG
  B"011"  Intersubnet TG
  B"100"  Branch extender TG
...1 . . .  Indicate whether HPR control flows supported
.... xxxx  Not used

0F
Garbage collection indicator

C"G"  This entry is generated by Garbage Collection processing.

C"M"  This entry is generated by Routing Tree Marking processing.

10–13
Node record pointer of the TG origin node

14–17
Node record pointer of the TG destination node

18–1B
TG record pointer of the reverse TG

1C
Number of days left before this TG is garbage collected

1D–1F
0

TPT3 entry for TG modified (Part 3)

Entry:
  TPT3
VIT option:
SSCP

Event:
Topology node entry modified

VIT processing module:
ISTITCST

Control is returned to:
Module invoking the INTRACE TYPE(TOPT) macro.

This record is a continuation of the TPT2 trace record. This record will be present only if the TG origin node has a different Network ID than the TG destination node.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;TPT3&quot;</td>
</tr>
<tr>
<td>04–0B</td>
<td>Network ID of the TG origin node</td>
</tr>
<tr>
<td>0C–13</td>
<td>Network ID of the TG destination node</td>
</tr>
<tr>
<td>14–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

**TREx entry for routing tree Build or Update**

Entry:
TREM, TRED, TREI, or TREB

VIT option:
SSCP

Event:
Routing tree Build or Update

VIT processing module:
ISTRACSC

Control is returned to:
Module invoking the INTRACE macro that caused the record to be produced

A routing tree operations (TRE) VIT entry is created by topology and routing services (TRS) whenever one of the following events is encountered:

- TRS accesses or builds a routing tree to satisfy a request for an APPN route.
- An existing routing tree is marked for modification because of a topological change in the APPN network.
- TRS learns of the availability or loss of a central directory server or interchange node.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>ID</th>
<th>MARKERS</th>
<th>TREE MARKS</th>
<th>PATH WEIGHT OR RES SEQ NUMBER OR TOPO DB NODE POINTER</th>
<th>TREE HEADER POINTER OR TOPO DB RES POINTER</th>
<th>RETURN ADDRESS</th>
<th>ORIGIN TREE RECORD POINTER OR COS POINTER</th>
<th>DEST TREE RECORD POINTER OR CALLING MODULE NAME OR BAD TREE RECORD POINTER</th>
<th>TREE BUILD EXEC TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>I D</td>
<td>MARKS</td>
<td>TREES</td>
<td>PATH WEIGHT OR RES SEQ NUMBER OR TOPO DB NODE POINTER</td>
<td>TREE HEADER POINTER OR TOPO DB RES POINTER</td>
<td>RETURN ADDRESS</td>
<td>ORIGIN TREE RECORD POINTER OR COS POINTER</td>
<td>DEST TREE RECORD POINTER OR CALLING MODULE NAME OR BAD TREE RECORD POINTER</td>
<td>TREE BUILD EXEC TIME</td>
</tr>
</tbody>
</table>

**Byte (hex)**

00–03

Record ID:

*C'TREM'*

Mark tree.

*C'TRED'*

Add or change directory server.

*C'TREI'*

Add or change interchange node.

*C'TREB'*

Build tree.

04

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05

Tree marking flags

**Bit**

**Meaning**

x... ....

1 = Operable resource state

0 = Inoperable resource state

.xxx ....

0

.... xxxxx

Reason

0000 = Transmission group update

0001 = Topology database update

0010 = Garbage collection

0011 = Overuse

0100 = Modify topology

0101 = Database reset

0110 = Uncache tree
0111 = Unreachable partner change
1000 = Unacceptable hop detected

06
Indicates that a directory server or interchange node is added ("A") or deleted ("D")

07
Tree building flags

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>x...</td>
<td>Indicates whether existing tree is used</td>
</tr>
<tr>
<td>.x..</td>
<td>Indicates whether clean path is used.</td>
</tr>
<tr>
<td>..x.</td>
<td>Indicates number of destinations</td>
</tr>
</tbody>
</table>

08–0B
Path weight, resource sequence number, or topology database node pointer, depending on entry type and flags:

<table>
<thead>
<tr>
<th>Entry type</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>Path weight</td>
</tr>
<tr>
<td>M</td>
<td>• If tree marking flags are (&quot;1000&quot;), topology database node pointer associated with the unacceptable tree record</td>
</tr>
<tr>
<td></td>
<td>• Otherwise, resource sequence number</td>
</tr>
</tbody>
</table>

Note: If entry type is TREB and the destination tree record pointer indicates multiple destinations, the path weight field is not valid.

0C–0F
Tree header pointer or topology database resource pointer, depending on entry type and flags:

<table>
<thead>
<tr>
<th>Entry type</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>Tree header pointer</td>
</tr>
<tr>
<td>M</td>
<td>• If tree marking flags are (&quot;1000&quot;), tree header pointer</td>
</tr>
<tr>
<td></td>
<td>• Otherwise, topology database resource pointer</td>
</tr>
</tbody>
</table>

10–13
Return address of caller

14–17
Origin tree record pointer or CoS pointer, depending on entry type and flags:
### Entry type
**Meaning**

**B**  
Origin tree record pointer

**M**  
- If tree marking flags are ('1000'), origin tree record pointer
- If tree marking flags are ('0110'), CoS pointer

### 18–1B
 Destination tree record pointer, unacceptable tree record pointer, or calling module, depending on entry type and flags:

**Entry type**
**Meaning**

**B**  
Destination tree record pointer

**M**  
- If tree marking flags are ('1000'), unacceptable tree record pointer
- If tree marking flags are ('0110'), last four characters of calling module name

### 1C–1F
 Tree building execution time (microseconds), if entry type is B

---

### TRMR entry for mark record (Part 1)

**Entry:**  
TRMR

**VIT option:**  
SSCP

**Event:**  
Routing tree marked for future update

**VIT processing module:**  
ISTITCAB

**Control is returned to:**  
ISTTRRUM

This record is generated when TRS marks a tree record because of changes on the network topology.

<table>
<thead>
<tr>
<th>00–03</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record ID: C&quot;TRMR&quot;</td>
<td></td>
</tr>
</tbody>
</table>
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Trace entry instance

New status of the tree record being marked

<table>
<thead>
<tr>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;C&quot;</td>
<td>Tree record is clean, no changes on this tree record.</td>
</tr>
<tr>
<td>&quot;I&quot;</td>
<td>Tree record is incomplete, a change was detected on a descendant of this tree record but the tree has not been updated.</td>
</tr>
<tr>
<td>&quot;D&quot;</td>
<td>Tree record is dirty, a change was detected on this tree record but the tree has not been updated.</td>
</tr>
</tbody>
</table>

Tree record pointer

Tree header pointer

Node record pointer

Node header pointer

Tree record pointer for the root of the routing tree

Request parameter header (RPH) address

TRM2 entry for mark record (Part 2)

Entry:

TRM2

VIT option:

SSCP

Event:

Routing tree marked for future update

VIT processing module:

ISTITCAB

This record is a continuation of the TRM2 trace record.
Byte (hex)
Contents
00–03
Record ID: C"TRM2"
04
Action on the topology entry that caused the tree marking
Code
Meaning
C"I"
Incomplete
C"U"
Update
C"A"
Add
C"D"
Delete
05
Type of topology entry
Code
Meaning
C"T"
TG
C"N"
Node
06–07
0
08–0F
CP name of the node on tree root
10–17
CP name of the tree record being marked
18–1F
CoS name

**TRM3 entry for mark record (Part 3)**

Entry:
TRM3
VIT option:
SSCP
Event:
Routing tree marked for future update
VIT processing module:
ISTITCAB

This record is a continuation of the TRM2 trace record. TRM3 is generated only if any of the CP names in the TRM2 record has a different network ID than the local node.
### TRNM entry for translate message (Part 1)

**Entry:**
- TRNM

**VIT option:**
- MSG

**Event:**
- Translate message error (Part 1)

**VIT processing module:**
- ISTRACOT

**Control is returned to:**
- ISTUSCMS or IKTMSIFR

This trace record is generated when a return code greater than the value 4 is received from the TRANMSG macro. When the return code and reason code indicate that the MVS message service is not active, no TRNM trace record is generated. A single message is translated each time the TRANMSG macro is invoked.

---

### Table: TRNM Entry

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;TRM3&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>Network ID of the node on tree root</td>
</tr>
<tr>
<td>10–17</td>
<td>Network ID of the tree record being marked</td>
</tr>
<tr>
<td>18–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

---

### Table: Byte (hex) Contents

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;TRM3&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>Network ID of the node on tree root</td>
</tr>
<tr>
<td>10–17</td>
<td>Network ID of the tree record being marked</td>
</tr>
<tr>
<td>18–1F</td>
<td>0</td>
</tr>
</tbody>
</table>
Constraint: C"TRNM"

**ID**
- ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**Language Code**
- 3-character language code or blanks if language is not supported

<table>
<thead>
<tr>
<th>Hex Value</th>
<th>Language Code</th>
<th>Language Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'02'</td>
<td>ARA</td>
<td>Arabic</td>
</tr>
<tr>
<td>X'03'</td>
<td>CHT</td>
<td>Traditional Chinese</td>
</tr>
<tr>
<td>X'04'</td>
<td>CHS</td>
<td>Simplified Chinese</td>
</tr>
<tr>
<td>X'05'</td>
<td>DAN</td>
<td>Danish</td>
</tr>
<tr>
<td>X'06'</td>
<td>DEU</td>
<td>German</td>
</tr>
<tr>
<td>X'07'</td>
<td>DES</td>
<td>Swiss German</td>
</tr>
<tr>
<td>X'08'</td>
<td>ELL</td>
<td>Greek</td>
</tr>
<tr>
<td>X'09'</td>
<td>ENG</td>
<td>UK English</td>
</tr>
<tr>
<td>X'00'</td>
<td></td>
<td>US English (default)</td>
</tr>
<tr>
<td>X'01'</td>
<td>ENU</td>
<td>US English (specified)</td>
</tr>
<tr>
<td>X'0A'</td>
<td>ESP</td>
<td>Spanish</td>
</tr>
<tr>
<td>X'0B'</td>
<td>FIN</td>
<td>Finnish</td>
</tr>
<tr>
<td>X'0C'</td>
<td>FRA</td>
<td>French</td>
</tr>
<tr>
<td>X'0D'</td>
<td>FRB</td>
<td>Belgian French</td>
</tr>
<tr>
<td>X'0E'</td>
<td>FRC</td>
<td>Canadian French</td>
</tr>
<tr>
<td>X'0F'</td>
<td>FRS</td>
<td>Swiss French</td>
</tr>
<tr>
<td>X'10'</td>
<td>HEB</td>
<td>Hebrew</td>
</tr>
<tr>
<td>X'12'</td>
<td>ISL</td>
<td>Icelandic</td>
</tr>
<tr>
<td>X'13'</td>
<td>ITA</td>
<td>Italian</td>
</tr>
<tr>
<td>X'14'</td>
<td>ITS</td>
<td>Swiss Italian</td>
</tr>
<tr>
<td>X'11'</td>
<td>JPN</td>
<td>Japanese</td>
</tr>
<tr>
<td>X'15'</td>
<td>KOR</td>
<td>Korean</td>
</tr>
<tr>
<td>X'16'</td>
<td>NLD</td>
<td>Dutch</td>
</tr>
<tr>
<td>X'17'</td>
<td>NLB</td>
<td>Belgian Dutch</td>
</tr>
<tr>
<td>X'18'</td>
<td>NOR</td>
<td>Norwegian</td>
</tr>
<tr>
<td>X'19'</td>
<td>PTG</td>
<td>Portuguese</td>
</tr>
<tr>
<td>X'1A'</td>
<td>PTB</td>
<td>Brazil Portuguese</td>
</tr>
<tr>
<td>X'1B'</td>
<td>RMS</td>
<td>Rhaeto-Romanic</td>
</tr>
<tr>
<td>X'1C'</td>
<td>RUS</td>
<td>Russian</td>
</tr>
<tr>
<td>X'1D'</td>
<td>SVE</td>
<td>Swedish</td>
</tr>
<tr>
<td>X'1E'</td>
<td>THA</td>
<td>Thai</td>
</tr>
<tr>
<td>X'1F'</td>
<td>TRK</td>
<td>Turkish</td>
</tr>
<tr>
<td>Hex Value</td>
<td>Language Code</td>
<td>Language Name</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------</td>
<td>---------------</td>
</tr>
<tr>
<td>X'3F'</td>
<td>Unknown language code</td>
<td></td>
</tr>
</tbody>
</table>

08–0B  
Message input/output block address

0C–0F  
Message input/output block length

10–13  
Return code

14–17  
Reason code

18–1B  
Input message parameter block address (MIOINPTP). If a message text block or text is supplied instead, 18–1B is the address of the text block or text.

1C–1F  
Output message text block address (MIOBUFFP)

**TRN2 entry for translate message (Part 2)**

**Entry:**
TRN2

**VIT option:**
MSG

**Event:**
Translate message error (Part 2)

**VIT processing module:**
ISTRACOT

This trace record is a continuation of the TRNM entry. It is generated when a TRNM entry is produced and the input to the TRANMSG macro is the address of a message parameter block.

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRN2</td>
<td>IST, IKT, OR USS MESSAGE NUMBER</td>
<td>FIRST MESSAGE VARIABLE OR 0</td>
<td>SECOND MESSAGE VARIABLE OR 0</td>
<td>THIRD MESSAGE VARIABLE OR 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03  
Record ID: C"TRN2"

04–07  
Last four digits of the message ID (for example, 0200 for message IKT0200I, or SG02 for USSMSG02)

08–0F  
First 8 bytes of the first message variable, if present; otherwise, 0

10–17  
First 8 bytes of the second message variable, if present; otherwise, 0

18–1F  
First 8 bytes of the third message variable, if present; otherwise, 0
TRRT entry for subtrace tree

TRRT entry for subtrace tree (Part 1)

Entry:
TRRT

VIT option:
SSCP

Subtrace type:
TREE

Event:
APPN route computation

VIT processing module:
ISTITCAB

Control is returned to:
Module invoking the INTRACE TYPE(TRRT) macro.

This record is generated when:
• Subtrace TREE under SSCP trace option is active
• TRS accesses a routing tree to satisfy a request for an APPN route

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C“TRRT”</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X‘FF’.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace entry instance</td>
</tr>
<tr>
<td>07</td>
<td>Status on tree record for current node:</td>
</tr>
<tr>
<td>Code</td>
<td>Meaning</td>
</tr>
<tr>
<td>C“C”</td>
<td>Tree record is clean, no changes on this tree record.</td>
</tr>
<tr>
<td>C“I”</td>
<td>Tree record is incomplete, a change was detected on a descendant of this tree record but the tree has not been updated.</td>
</tr>
</tbody>
</table>
Tree record is dirty, a change was detected on this tree record but the tree has not been updated.

Node record pointer of the node described by the current tree record
Node record pointer of the node described by the child tree record
Node record pointer of the node described by the sibling tree record
Node record pointer of the node described by the parent tree record
Weight of the TG from parent node to this node
Weight of the current node
Request parameter header (RPH) address

Note: An APPN route is comprised of TGs and network nodes. To compute the optimal route, TRS assigns a weight to each TG and network node by comparing the resource's characteristics against the characteristics required to set up the route. By adding up all weights, a route weight can be calculated and the optimal route with the least weight can be chosen.

TRRT entry for subtrace tree (Part 2)
Entry: TRR2
VIT option: SSCP
Subtrace type: TREE
Event: APPN route computation
VIT processing module: ISTITCAB
This record is a continuation of the TRRT trace record.

<table>
<thead>
<tr>
<th>00 00 00 03 00 05 08</th>
<th>00 00 00 00 00 00 00 00</th>
<th>10 11 12 13 11 11 11 11</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRR2</td>
<td>T W I N E</td>
<td>T R R 2</td>
</tr>
<tr>
<td>O T A G M E N T</td>
<td>O</td>
<td>N E D E</td>
</tr>
<tr>
<td></td>
<td>CP</td>
<td>CURRENT</td>
</tr>
<tr>
<td></td>
<td>NAME</td>
<td>CP</td>
</tr>
</tbody>
</table>

Byte (hex)
Contents
00–03
Record ID: C"TRR2"
04–05
Total path weight from the root of routing tree to the current node
06
Trace entry instance
07
0
08–0F
CP name of the current node
10–17
CP name of the child node
18–1F
CP name of the sibling node

TRRT entry for subtrace tree (Part 3)

Entry:
TRR3
VIT option:
SSCP
Subtrace type:
TREE
Event:
APPN route computation
VIT processing module:
ISTITCAB

This record is a continuation of the TRR2 trace record. TRR3 is generated only if any CP names in the TRR2 record have a different network ID than the local node.

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TRR3</th>
<th>0</th>
<th>CURRENT NODE</th>
<th>CHILD NODE</th>
<th>SIBLING NODE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NETWORK ID</td>
<td>NETWORK ID</td>
<td>NETWORK ID</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03
Record ID: C"TRR3"

04–07
0

08–0F
Current node network ID

10–17
Parent node network ID

18–1F
Sibling node network ID
TRRT entry for subtrace tree (Part 4)

Entry:
- TRR4

VIT option:
- SSCP

Subtrace type:
- TREE

Event:
- APPN route computation

VIT processing module:
- ISTITCAB

This record is generated if the current routing tree hop (tree record) described by the TRRR and TRR2 pair contains any TGs. If the current hop contains multiple equal-weights TGs, TRR4 is generated for each TG on the hop.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;TRR4&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>Trace entry instance</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>CP name of the TG destination node</td>
</tr>
<tr>
<td>10–13</td>
<td>Resource sequence number</td>
</tr>
<tr>
<td>14–15</td>
<td>RSCV counter</td>
</tr>
<tr>
<td>16–17</td>
<td>Resource usage counter</td>
</tr>
<tr>
<td>18</td>
<td>TG number</td>
</tr>
<tr>
<td>19</td>
<td>TG descriptor flags</td>
</tr>
<tr>
<td>1A</td>
<td>TG status flags</td>
</tr>
</tbody>
</table>
TRRT entry for subtrace tree (Part 5)

Entry: TRR5
VIT option: SSCP
Subtrace type: TREE
Event: APPN route computation
VIT processing module: ISTITCAB

This record is a continuation of the TRR4 trace record. TRR5 is generated only if the destination CP name in the TRR4 record has a different network ID than the local node.

<table>
<thead>
<tr>
<th>TRR5</th>
<th>0</th>
<th>DESTINATION NODE NETWORK ID</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
</table>

Byte (hex)
Contents
00–03  Record ID: C"TRR5"
04–07  0
08–0F  Destination node network ID
10–1F  0

Note:
1. The group TRRT, TRR2, TRR3, TRR4, and TRR5 describes a single hop (represented by a tree record) on a routing tree. If the hop contains multiple equal-weight TGs, multiple TRR4 trace records are generated (one for each TG on that hop).
2. When a routing tree is accessed to satisfy a route request and the TREE subtrace is active, the whole routing tree will be traced. Each hop (tree record) on the tree will generate the group TRRT, TRR2, TRR3, TRR4, and TRR5. The processing module ISTITCAB is called when the INTRACE TYPE (TRRT) is issued. ISTITCAB will traverse the routing tree to trace every tree record on the tree.
3. After every tree record on the tree has been traced, control will be returned to the module that invoked the INTRACE macro.
**TSNS entry to trace sense codes**

**Entry:**
- TSNS

**VIT option:**
- PIU

**Event:**
- TSNS record

**VIT processing module:**
- ISTRACOT

**Control is returned to:**
- Module invoking the INTRACE macro that caused the record to be produced.

This trace record allows a given TSC module to trace a sense code at the time it is generated. The TSNS event is treated as an exception condition and is always traced, whether the PIU option is in effect.

<table>
<thead>
<tr>
<th>0–3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C“TSNS”</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–06</td>
<td>0</td>
</tr>
</tbody>
</table>
07
Instance identifier in invoking module

08–0F
0

10–13
Address of issuer of the TSNS macro

14–17
Control block address or 0

18–1B
Sense code

1C–1F
Request parameter header (RPH) address

**UE entry for USER exit (DFASY or RESP)**

**Entry:**
UE

**VIT option:**
API

**Event:**
User exit (DFASY or RESP)

**VIT processing module:**
ISTRACAP

**Control is returned to:**
ISTAICUE

This trace record identifies the exit dispatched and data passed back to the user application program.

If this entry is associated with an event failure (that is, the return code is nonzero), this entry is generated whether the API option is in effect. It is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

For DFASY and RESP exits (type codes = 07 and 08), the user exit trace record has the following format:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–01</td>
<td>Record ID: C&quot;UE&quot;</td>
</tr>
<tr>
<td>02</td>
<td>Exit type code (hex 7=DFASY, hex 8=RESP)</td>
</tr>
<tr>
<td>03</td>
<td>Exit definition (RPLEXTDS)</td>
</tr>
</tbody>
</table>
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Return code (RPLRTNCD) (For more information, see z/OS Communications Server: SNA Programming)

06
Feedback code (RPLFDB2) (For more information, see z/OS Communications Server: SNA Programming)

07
Feedback code (RPLFDB3) (For more information, see z/OS Communications Server: SNA Programming)

08–0B
RPL address

0C–0F
Exit address

10–13
RPL data area pointer (RPLAREA) or 0

14–17
Record length (RPLRLEN)

18–1B
CID from NIB or RPL (RPLARG) or 0

1C–1F
Second RPL feedback area, sense code (RPLFDBK2)

**UE1 entry for USER exit (not DFASY or RESP, Part 1)**

**Entry:**
UE1

**VIT option:**
API

**Event:**
User exit (not DFASY or RESP, Part 1)

**VIT processing module:**
ISTRACAP

**Control is returned to:**
ISTAICUE

The user exit trace record (not DFASY or RESP, Part 1) has the following format:

| 00 | 02 | 00 | 03 | 00 | 05 | 00 | 07 | 00 | 0A | 00 | 0C | 00 | 0F | 01 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 11 |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| **UE1** | **ID** | **CODE** | 0 | **EXIT-TYPE** | **DEPENDENT INFORMATION** | **PRIMARY (APPL) LU NAME** | **SECONDARY LU NAME OR 0** |

**Byte (hex)**
**Contents**

00–03
Record ID: C"UE1"
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Exit type code (see Table 15 on page 661)

08–0F
Exit-type-dependent information (see Table 15 on page 661)

Primary (application program) logical unit name

Secondary logical unit name, or 0, if this is a TPEND exit

<table>
<thead>
<tr>
<th>Exit type code</th>
<th>Exit type</th>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>05</td>
<td>SCIP (BIND)</td>
<td>08–09</td>
<td>RPL control flags (RPLCNTDC, RPLCNTSC)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0A–0F</td>
<td>Session parameters</td>
</tr>
<tr>
<td>05</td>
<td>SCIP (STSN)</td>
<td>08–09</td>
<td>RPL control flags (RPLCNTDC, RPLCNTSC)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0A–0B</td>
<td>RPLOBSQV</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0C–0D</td>
<td>RPLIBSQV</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0E</td>
<td>RPLOBSQ</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0F</td>
<td>RPLIBSQ</td>
</tr>
<tr>
<td>05</td>
<td>SCIP (other)</td>
<td>08–09</td>
<td>RPL control flags (RPLCNTDC, RPLCNTSC)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0C–0F</td>
<td>CID</td>
</tr>
<tr>
<td>06</td>
<td>LOGON</td>
<td>08–0B</td>
<td>Logon data length</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0C–0F</td>
<td>CID</td>
</tr>
<tr>
<td>09</td>
<td>LOSTERM</td>
<td>08–0B</td>
<td>Reason code</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0C–0F</td>
<td>CID</td>
</tr>
<tr>
<td>0A</td>
<td>RELREQ</td>
<td>08–0F</td>
<td>Unused</td>
</tr>
<tr>
<td>0D</td>
<td>TPEND</td>
<td>08–0B</td>
<td>Reason code</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0C–0F</td>
<td>Unused</td>
</tr>
<tr>
<td>0E</td>
<td>NSEXIT</td>
<td>08–0F</td>
<td>NS RU bytes 00–07</td>
</tr>
</tbody>
</table>

**UE2 entry for USER exit (Not DFASY or RESP, part 2)**

**Entry:**

UE2

**VIT option:**

API

**Event:**

User exit (not DFASY or RESP, Part 2)
VIT processing module:
ISTRACAP

This trace record is a continuation of the UE1 entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;UE2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>Exit address</td>
</tr>
<tr>
<td>08–17</td>
<td>0</td>
</tr>
<tr>
<td>18–1F</td>
<td>Network identifier, or 0 if this is a TPEND exit</td>
</tr>
</tbody>
</table>

ULKA entry for release all locks

Entry:
ULKA
VIT option:
LOCK
Event:
Release all locks
VIT processing module:
ISTRACLK
Control is returned to:
ISTAPC39

This trace record contains information about a routine releasing all locks it currently holds.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;ULKA&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
</tbody>
</table>
06–07
0
08–0B
PST address
0C–0F
CRA lock account word (CRALKACT)
10–13
Address of the issuer of the TPUNLOCK macro
14–1B
0
1C–1F
Request parameter header (RPH) address

**UNLK entry for release a lock**

**Entry:**
UNLK

**VIT option:**
LOCK

**Event:**
Release a lock

**VIT processing module:**
ISTRACLK

**Control is returned to:**
ISTAPC33

This trace record contains information for a routine releasing a lock. It is the complement of an LKEX or LKSH entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;UNLK&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06–07</td>
<td>Level of lock to be released. See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for a listing of VTAM locks.</td>
</tr>
<tr>
<td>08–0B</td>
<td>Address of lock to be released</td>
</tr>
</tbody>
</table>
0C–0F
CRA lock account word (CRALKACT)

10–13
Address of the issuer of the TPUNLOCK macro

14–1A
Lockword pointed to by address in bytes 08–0B

1B
The LOCK ID field. See z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures for a list of VTAM locks.

1C–1F
Request parameter header (RPH) address

**UP entry for user application program post**

**Entry:**
UP

**VIT option:**
API

**Event:**
User application program post

**VIT processing module:**
ISTRACAP

**Control is returned to:**
ISTAICPT or ISTAPCSX

This trace record provides the RPL information that VTAM passes to the application program. The RPL information and the API TPIO request record (AIn or IOn) show VTAM activity for a given API request.

See z/OS Communications Server: SNA Programming for a description of the RPL return codes.

If this entry is associated with an event failure (that is, the return code is nonzero), this entry is generated whether the API option is in effect. It is treated as an exception condition, and is always traced if the VIT is active, regardless of the VIT options specified.

| Command | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 | Byte 8 | Byte 9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| UP | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| VTI | E | X | I | T | A | C | P | T | O | M | T | E | N | C | D | B |
| RPL ADDRESS | F | D | B | 2 | F | D | B | 3 | RPL ADDRESS | EXIT OR ECB ADDRESS | RPL AREA | RPL RLEN | CID OR 0 | RPL FDBK2 |

**Byte (hex)**

**Contents**

00–01
Record ID: C"UP"

02
RPL request type (RPLREQ)

X'15'
SETLOGON

X'16'
SIMLOGON
Exit definition (RPLEXTDS)

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Return code (RPLRTNCD) (See z/OS Communications Server: SNA Programming.)

Feedback code (RPLFDB2) (See z/OS Communications Server: SNA Programming.)

Feedback data flag (RPLFDB3) (See z/OS Communications Server: SNA Programming under INQUIRE macro with OPTCD=APPSTAT.)

RPL address

Exit address or ECB address

RPL data area pointer (RPLAREA)

Record length (RPLRLEN)

CID from NIB or from RPL (RPLARG), or 0
USI or USO entry for APPC commands

Entry:
- USI or USO

VIT option:
- APPC

Event:
- APPCCMD

VIT processing module:
- ISTRACAC

Control is returned to:
- Calling module

The USI trace record contains what RPLAREA points to when the following APPC commands are issued:

APPCCMD
Data
- CONTROL=ALLOC,QUALIFY=*
  - FMH-5
- CONTROL=SENDFMH5
  - FMH-5
- CONTROL=OPRCNTL,QUALIFY=ACTSESS
  - Session parameters
- CONTROL=OPRCNTL,QUALIFY=CNOS
  - CNOS structure
- CONTROL=OPRCNTL,QUALIFY=DEFINE
  - DEFINE/DISPLAY structure

Note: * indicates all the possible values of QUALIFY when CONTROL equals ALLOC.

The USO trace record contains what RPLAREA points to when the following APPC commands complete:

APPCCMD
Data
- CONTROL=OPRCNTL, QUALIFY=CNOS
  - CNOS structure
- CONTROL=OPRCNTL, QUALIFY=DISPLAY
  - DEFINE/DISPLAY structure
- CONTROL=OPRCNTL, QUALIFY=RESTORE
  - RESTORE structure
- CONTROL=RCVFMH5
  - FMH-5

A maximum of 212 bytes of user data will be traced. The FMH-5 will be traced only up to but not including the security access parameters.

Note: If the address of the CNOS session limits structure is not specified on the AREA keyword of the APPCCMD, the USI and USO records are not cut for the APPCCMD.
### Byte (hex)
**Contents**

<table>
<thead>
<tr>
<th>00–03</th>
<th>Record ID: C&quot;USI&quot; or C&quot;USO&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>APPCCMD CONTROL operand value:</td>
</tr>
<tr>
<td></td>
<td>X'10' ALLOC</td>
</tr>
<tr>
<td></td>
<td>X'12' SENDFMH5</td>
</tr>
<tr>
<td></td>
<td>X'40' OPRCNTL</td>
</tr>
<tr>
<td></td>
<td>X'60' RCVFMH5</td>
</tr>
<tr>
<td>06</td>
<td>APPCCMD QUALIFY operand value:</td>
</tr>
<tr>
<td></td>
<td>X'06' CNOS</td>
</tr>
<tr>
<td></td>
<td>X'0C' DEFINE</td>
</tr>
<tr>
<td></td>
<td>X'0D' DISPLAY</td>
</tr>
<tr>
<td></td>
<td>X'12' ACTSESS</td>
</tr>
<tr>
<td></td>
<td>X'14' ALLOCD</td>
</tr>
<tr>
<td></td>
<td>X'15' IMMED</td>
</tr>
<tr>
<td></td>
<td>X'16' CONWIN</td>
</tr>
<tr>
<td></td>
<td>X'1B' RESTORE</td>
</tr>
<tr>
<td></td>
<td>X'1E' CONVGRP</td>
</tr>
<tr>
<td></td>
<td>X'1F' WHENFREE</td>
</tr>
</tbody>
</table>

---

#### Table

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;USI&quot; or C&quot;USO&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>APPCCMD CONTROL operand value:</td>
</tr>
<tr>
<td></td>
<td>X'10' ALLOC</td>
</tr>
<tr>
<td></td>
<td>X'12' SENDFMH5</td>
</tr>
<tr>
<td></td>
<td>X'40' OPRCNTL</td>
</tr>
<tr>
<td></td>
<td>X'60' RCVFMH5</td>
</tr>
<tr>
<td>06</td>
<td>APPCCMD QUALIFY operand value:</td>
</tr>
<tr>
<td></td>
<td>X'06' CNOS</td>
</tr>
<tr>
<td></td>
<td>X'0C' DEFINE</td>
</tr>
<tr>
<td></td>
<td>X'0D' DISPLAY</td>
</tr>
<tr>
<td></td>
<td>X'12' ACTSESS</td>
</tr>
<tr>
<td></td>
<td>X'14' ALLOCD</td>
</tr>
<tr>
<td></td>
<td>X'15' IMMED</td>
</tr>
<tr>
<td></td>
<td>X'16' CONWIN</td>
</tr>
<tr>
<td></td>
<td>X'1B' RESTORE</td>
</tr>
<tr>
<td></td>
<td>X'1E' CONVGRP</td>
</tr>
<tr>
<td></td>
<td>X'1F' WHENFREE</td>
</tr>
</tbody>
</table>

---

VTAM internal trace (VIT) record descriptions 667
USI2 or USO2 entry for APPC commands

Entry:
USI2 or USO2

VIT option:
APPC

Event:
APPCCMD

VIT processing module:
ISTRACAC

This trace record is a continuation of the USI or USO entries. A maximum of 212 bytes of data is traced.

<table>
<thead>
<tr>
<th>00–03</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Record ID: C&quot;USI2&quot; or C&quot;USO2&quot;</td>
</tr>
<tr>
<td>04–1F</td>
<td>User data</td>
</tr>
</tbody>
</table>

UVO or UVI entry for vector information

Entry:
UVO or UVI

VIT option:
APPC

Event:
VTRINA and VTRINL, or VTROUTA and VTROUTL specified on an APPCCMD macroinstruction

VIT processing module:
ISTRACAC

Control is returned to:
APPCVL

This trace record contains vector information for APPCCMD macroinstructions that specify a vector area. UVI traces any input vectors (application-to-VTAM) when an APPCCMD macroinstruction is issued. UVO traces any output vectors when an APPCCMD macroinstruction is completed. The user data field of the USI and USO entries (0C–1B) is used to trace the total vector length (including the length of the length field) and vectors up to a maximum of eight trace records. The UVI vectors are traced from the application’s storage. The UVO vectors are traced from VTAM’s storage.
Byte (hex)
Contents
00–03
  Record ID:
  • C"UVI" for input vector tracing
  • C"UVO" for output vector tracing
04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
  APPCCMD CONTROL operand value:
  X'10'
    ALLOC
  X'11'
    PREALLOC
  X'12'
    SENDFMH5
  X'20'
    RESETRCV
  X'30'
    DEALLOC
  X'31'
    DEALLOCQ
  X'40'
    OPRCNTL
  X'50'
    PREPRCV
  X'60'
    RCVFMH5
  X'70'
    RECEIVE
  X'71'
    RCVEXPD
  X'80'
    REJECT
  X'90'
    SEND
  X'91'
    SENDEXPD
  X'92'
    SENDRCV
APPCCMD QUALIFY operand value:

- X'00' NULL
- X'01' ABNDPROG
- X'02' ABNDUSER
- X'03' ABNDTIME
- X'04' ABNDUSER
- X'05' ANY
- X'06' CNOS
- X'07' CONFIRM
- X'08' CONFRMD
- X'09' DATA
- X'0A' DATACON
- X'0B' DATAFLU
- X'0C' DEFINE
- X'0D' DISPLAY
- X'0E' ERROR
- X'0F' FLUSH
- X'10' RQSEND
- X'11' SPEC
- X'12' ACTSESS
- X'13' DACTSESS
- X'14' ALLOCD
X'15'  IMMED
X'16'  CONWIN
X'17'  SESSION
X'18'  CONV
X'19'  SUSPEND
X'1A'  RESUME
X'1B'  RESTORE
X'1C'  SYNCBEG
X'1D'  SYNCEND
X'1E'  CONVGRP
X'1F'  WHENFREE
X'20'  IANY
X'21'  ISPEC
X'22'  ALL
X'23'  IALL
X'24'  QUEUE
X'25'  DATAQUE

07
0
08–0B  RPL address

0C–1B  Contents of the RPL6VAIA for UVO trace records. Contents of the RPL6VAOA for UVI trace records. No more than eight trace records of vector data is traced.

1C–1F  Request parameter header (RPH) address

**UVO2 or UVI2 entry for vector information**

**Entry:**
UVO2 or UVI2

**VIT option:**
APPC
Event:
VTRINA and VTRINL, or VTROUTA and VTROUTL specified on an APPCCMD macroinstruction

VIT processing module:
ISTRACAC

This trace record is a continuation of the UVO entry.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID:</td>
</tr>
<tr>
<td></td>
<td>- C&quot;UVI2&quot; for continuation of UVI</td>
</tr>
<tr>
<td></td>
<td>- C&quot;UVO2&quot; for continuation of UVO</td>
</tr>
<tr>
<td>04–1F</td>
<td>Vector data:</td>
</tr>
<tr>
<td></td>
<td>- Contents of the RPL6VAOA for UVI trace records.</td>
</tr>
<tr>
<td></td>
<td>- Contents of the RPL6VAIA for UVO trace records.</td>
</tr>
</tbody>
</table>

**VCC1 entry for CNCB queued (Part 1)**

Entry:
VCC1

VIT option:
VCNS

Event:
CNCB queued (Part 1)

VIT processing module:
ISTRACNS

The VCC1 trace record is written when a CNS protocol control block (CNCB) is queued between CNS and LSA components. The VCC2 trace record displays data contained in the CNCB.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;VCC1&quot;</td>
</tr>
</tbody>
</table>
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

Control block identifier
- CBID=X'0B' (ISTVLNCB)
  - X'00' (Reset)
  - X'01' (Pending active)
  - X'02' (Active without application logged on to line)
  - X'03' (Active with application logged on to line)
  - X'04' (Pending inactive phase 1)
  - X'05' (Pending inactive phase 2)
  - X'06' (Pending inactive phase 3)
  - X'07' (Blocked)
- CBID=X'7B' (ISTRIB)
  - X'00' (Idle)
  - X'01' (Logon in progress)
  - X'02' (Active)
  - X'03' (Logoff in progress)
  - X'04' (DACTLINK in progress)
  - X'05' (Forced termination in progress)

CNCB primitive code
- X'01' (CNS_INQUIRE_REQUEST)
- X'02' (CNS_INQUIRE_CONFIRM)
- X'03' (CNS_LOGON_REQUEST)
X'04'
   CNS_LOGON_CONFIRM
X'05'
   CNS_LOGOFF_REQUEST
X'06'
   CNS_LOGOFF_CONFIRM
X'07'
   CNS_LOGOFF_RESPONSE
X'08'
   CNS_LOGOFF_INDICATION
X'09'
   CNS_CLOSEACB_REQUEST
X'0A'
   CNS_CLOSEACB_CONFIRM
X'0B'
   CNS_CLOSEACB_COMPLETION_INDICATION

08–0B
   Address of the major control block
0C–0F
   CNCB address
10–13
   Requesting module abbreviated name
14–1F
   First 12 bytes of CNCB data

VCC2 entry for CNCB queued (Part 2)

Entry:
   VCC2
VIT option:
   VCNS
Event:
   CNCB queued (Part 2)
VIT processing module:
   ISTRACNS

This trace record is a continuation of the VCC1 trace record. Up to two VCC2 trace records might follow a VCC1 trace record, depending on the length of the primitive that is traced.

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
</tbody>
</table>

VCC2
   UP TO 28 BYTES OF ADDITIONAL CNCB DATA

Byte (hex)
   Contents
00–03
   Record ID: "VCC2"
04–1F
   Up to 28 bytes of additional CNCB data
**VCDQ entry for work element dequeued**

**Entry:**

- VCDQ

**VIT option:**

- VCNS

**Event:**

- Work element dequeued

**VIT processing module:**

- ISTRACNS

**Control is returned to:**

- Module invoking the INTRACE macro that caused the record to be produced

This trace record is written whenever a work element is dequeued for processing. The VCDQ trace record identifies the work element being dequeued.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;VCDQ&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>Process state of the associated connection or of the control block processing the dequeued work element:</td>
</tr>
<tr>
<td></td>
<td>• States when CBID=X'0B' (ISTVLNCB)</td>
</tr>
<tr>
<td></td>
<td>X'00' Reset state</td>
</tr>
<tr>
<td></td>
<td>X'10' ACTLINK pending</td>
</tr>
<tr>
<td></td>
<td>X'20' Active state</td>
</tr>
<tr>
<td></td>
<td>X'30' DACTLINK pending</td>
</tr>
<tr>
<td></td>
<td>• States when CBID=X'0C' (ISTPCLCB)</td>
</tr>
<tr>
<td></td>
<td>X'00' Reset</td>
</tr>
<tr>
<td></td>
<td>X'01' Logon complete</td>
</tr>
<tr>
<td></td>
<td>X'02' Logon pending</td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) record descriptions 675
X'03' Logoff pending
X'04' Adapter termination pending
• States when CBID=X'25' (ISTVCCB)
X'01' P1 Ready
X'02' P2 DTE waiting
X'03' P3 DCE waiting
X'04' P4 Data transfer
X'05' P5 Call collision
X'06' P6 DTE clear request
X'07' P7 DCE clear indication
X'11' D1 Flow control ready
X'12' D2 DTE reset request
X'13' D3 DCE reset indication
X'21' I1 No interrupt pending
X'22' I2 DTE interrupt pending
X'31' E1 No error reset
X'32' ER CNS error reset
X'33' CF CNS cancel forward
• States when CBID=X'66' (ISTCAB)
X'00' Idle
X'01' Connect requested
X'02' Connect indicated
X'03' Flow control ready
X'04' Error reset
X'05' Cancel hold
X'06'
  hold purge
X'07'
  reset requested
X'08'
  reset indicated
X'09'
  error disconnect
X'0A'
  cleanup
X'0B'
  disconnect requested
X'0C'
  disconnect indicated
X'0D'
  pending error disconnect

- States when CBID=X'6B' (ISTCAR)
  No states are traced when ISTCAR is the major control block.
- States when CBID=X'7B' (ISTRIB)

X'00'
  idle
X'01'
  logon in progress
X'02'
  active
X'03'
  logoff in progress
X'04'
  DACTLINK in progress
X'05'
  forced termination in progress

06
control block identifier

X'0B'
  ISTVLNCB
X'0C'
  ISTPCLCB
X'25'
  ISTVCCB
X'66'
  ISTCAB
X'6B'
  ISTCAR
X'7B'
  ISTRIB

07
if work element is an ‘RPL’ (see bytes 08–0B)
X'08'
  CONTROL=INQUIRE
X'10'
CONTROL=LOGON
X'14'
CONTROL=TEST
X'18'
CONTROL=XID
X'20'
CONTROL=LOGOFF
X'30'
CONTROL=READ
X'40'
CONTROL=STATUS
X'50'
CONTROL=SET
X'60'
CONTROL=OPEN
X'70'
CONTROL=CLOSE
X'80'
CONTROL=SEND
X'90'
CONTROL=SUSPEND
X'A0'
CONTROL=RESUME
X'B0'
CONTROL=EXPEDITE
X'C0'
CONTROL=CHECK
X'D0'
CONTROL=RESET
X'E0'
CONTROL=SETCPARM
X'F0'
CONTROL=RECEIVE
X'FC'
CONTROL=REPLY

If work element is a ‘PICB’ (see bytes 08–0B)
X'01'
LOGON_REQUEST
X'02'
LOGON_CONFIRM
X'03'
LOGOFF_REQUEST
X'05'
CNS_LOGON_REQUEST
X'06'
CNS_LOGON_CONFIRM
X'07'
  CNS_LOGOFF_REQUEST
X'08'
  CNS_LOGOFF_CONFIRM
X'09'
  CNS_LOGOFF_RESPONSE
X'0A'
  CNS_LOGOFF_INDICATION
X'0B'
  CNS_CLOSEACB_REQUEST
X'0C'
  CNS_CLOSEACB_CONFIRM
X'0D'
  CNS_CLOSEACB_COMPLETION_INDICATION
X'11'
  CONNECT_REQUEST
X'12'
  CONNECT_CONFIRM
X'13'
  CONNECT_ACCEPT
X'14'
  CONNECT_INDICAT
X'15'
  DISCONNECT_REQUEST
X'16'
  DISCONNECT_INDICAT
X'17'
  DISCONNECT_CONFIRM
X'18'
  ERROR_DISCONNECT_INDICAT
X'19'
  RESTART_INDICAT
X'1A'
  CNS_CLEAR
X'1B'
  ERROR_DISCONNECT_INDICATION (TIMER)
X'21'
  RESET_REQUEST
X'22'
  RESET_CONFIRM
X'23'
  RESET_ACCEPT
X'24'
  RESET_INDICAT
X'25'
  ERROR_RESET_INDICAT
X'31'
  DATA_OUT (LEVEL1)
X'32'
  DATA_IN (LEVEL1)
X'33'
  DATA_OUT (LEVEL2)
X'34'
  DATA_IN (LEVEL2)
X'35'
  INTERRUPT_OUT
X'36'
  INTERRUPT_CONFIRM
X'37'
  INTERRUPT_INDICATION
X'42'
  STATUS_INDICAT_FINAL_TERMINAT
X'53'
  SUSPEND_REQUEST
X'54'
  RESUME_REQUEST
X'55'
  RNR_IN
X'56'
  RR_IN
X'57'
  ROTATE
X'62'
  TIMER_EVENT
X'63'
  RESTART_VC_TRIGGER
X'71'
  RESET_PENDING_TRIGGER
X'72'
  RESET_COMPLETE_TRIGGER
X'73'
  CLEAR_PENDING_TRIGGER
X'74'
  CLEAR_COMPLETE_TRIGGER
X'75'
  CALL_PENDING_TRIGGER
X'76'
  CALL_COMPLETE_TRIGGER
X'77'
  INTERRUPT_PENDING_TRIGGER
X'78'
  ERROR_RESET_TRIGGER
X'79'
  ERROR_DISCONNECT_TRIGGER
X'82'
  PACKET_IN
X'91'
  TERMINATE_PORT
X'E0'
  CNS_NMVT
X'F0'
  READ_REQ
X'F1'
  READCONFIRM
X'F2'
  STATUS_PHYSICAL_REQUEST
X'F3'
  STATUS_PHYSICAL_CONFIRM
X'F4'
  STATUS_VIRTUAL_REQ
X'F5'
  STATUS_VIRTUAL_CONFIRM

If work element is a ‘CNCB’ (see bytes 08–0B)
X'01'
  CNS_INQUIRE_REQUEST
X'02'
  CNS_INQUIRE_CONFIRM
X'03'
  CNS_LOGON_REQUEST
X'04'
  CNS_LOGON_CONFIRM
X'05'
  CNS_LOGOFF_REQUEST
X'06'
  CNS_LOGOFF_CONFIRM
X'07'
  CNS_LOGOFF_RESPONSE
X'08'
  CNS_LOGOFF_INDICATION
X'09'
  CNS_CLOSEACB_REQUEST
X'0A'
  CNS_CLOSEACB_CONFIRM
X'0B'
  CNS_CLOSEACB_COMPLETION_INDICATION

08–0B
Work element type:
C"CNCB"
  CNCB
C"RPL"
  RPL
C"PICB"
  PICB
C"RUPE"
  RUPE
C"VCCB"
  VCCB
C"CAB"
  CAB
VHCR entry for invoking a RoCE VHCR operation (part 1)

Entry:
VHCR

VIT option:
CIA

Event:
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) VHCR operation as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing when the IBM 10 GbE RoCE Express feature operates in a shared RoCE environment.

This trace record is written upon completion of the Issue_HcrOp function. The VHCR entry is used to invoke a RoCE VHCR operation.
**Byte (hex)**

**Contents**

**00-03**

Record ID: C"VHCR"

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

Operation code

**06 - 07**

Identifier of the module that issued the INTRACE command

**08 - 09**

Return code

**0A - 0B**

Reason code

**0C - 0F**

Operation code modifier

**10 - 17**

Address of the PFCTE

**18 - 1B**

Hardware handle

**1C - 1F**

Request parameter header (RPH) address

---

**VHC2 entry for invoking a RoCE VHCR operation (part 2)**

**Entry:**

VHC2

**VIT option:**

CIA

**Event:**

Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) VHCR operation as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing when the IBM 10 GbE RoCE Express feature operates in a shared RoCE environment.

This trace record is a continuation of the VHCR entry.

---

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03</td>
<td>Record ID: C&quot;VHC2&quot;</td>
</tr>
<tr>
<td>04-07</td>
<td>Input modifier</td>
</tr>
</tbody>
</table>
VHC3 entry for invoking a RoCE VHCR operation (part 3)

**Entry:**
VHC3

**VIT option:**
CIA

**Event:**
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) VHCR operation as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing when the IBM 10 GbE RoCE Express feature operates in a shared RoCE environment.

This trace record is a continuation of the VHCR entry.

**Byte (hex)**

**Contents**

| 00-03 | Record ID: C"VHC3" |
| 04-07 | Operation code and operation retries values |
| Bit | Meaning |
| 0 - 11 | Represents the operation code performed |
| 12 - 31 | Represents the number of retries before the command completes |
| 08-0F | The value of the communication channel before the command is attempted |
| 10-17 | The value stored by z/OS Communication Server into the communication channel to execute the command |
| 18-1F | The value of the communication channel after the command completes |

VHC4 entry for invoking a RoCE VHCR operation (part 4)

**Entry:**
VHC4
**VIT option:**
- CIA

**Event:**
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) VHCR operation as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing when the IBM 10 GbE RoCE Express feature operates in a shared RoCE environment.

This trace record is a continuation of the VHCR entry, and is generated only when the VHCR operation requires command input data. Multiple VHC4 entries might be generated, depending on the length of the command input data.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03</td>
<td>Record ID: C&quot;VHC4&quot;</td>
</tr>
<tr>
<td>04-1F</td>
<td>Command input data</td>
</tr>
</tbody>
</table>

### VHC5 entry for invoking a RoCE VHCR operation (part 5)

**Entry:**
- VHC5

**VIT option:**
- CIA

**Event:**
Invocation of a Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) VHCR operation as part of Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing when the IBM 10 GbE RoCE Express feature operates in a shared RoCE environment.

This trace record is a continuation of the VHCR entry, and is generated only when the VHCR operation returns command output data. Multiple VHC5 entries might be generated, depending on the length of the command output data returned.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-03</td>
<td>Record ID: C&quot;VHC5&quot;</td>
</tr>
<tr>
<td>04-1F</td>
<td>Command output data</td>
</tr>
</tbody>
</table>

**VPST entry for VPOST macro**

**Entry:**
- VPST

VTAM internal trace (VIT) record descriptions 685
VIT option:
  PSS

Event:
  VPOST macro

VIT processing module:
  ISTRACPS

Control is returned to:
  Module invoking the INTRACE macro that caused the record to be produced

This trace record is written when a VTAM module issues a VPOST macro. The VPOST macro generates a system POST event.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;VPST&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Address of the ASCB or 0</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of the ECB being posted</td>
</tr>
<tr>
<td>10–17</td>
<td>Subtask name</td>
</tr>
<tr>
<td>18–1B</td>
<td>Address of the issuer of the VPOST macro</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Completion code or 0</td>
</tr>
</tbody>
</table>

**VRSM entry for resume after VWAIT**

Entry:
  VRSM

VIT option:
  PSS

event:
  VWAIT macro

VIT processing module:
  ISTRACPS

This trace record is written when VTAM resumes control after a system WAIT event generated by the VTAM VWAIT macro.
VTAM internal trace (VIT) record descriptions

VTAL entry for allocate storage

Entry:
VTAL
VIT option:
SMS
Event:
Allocate storage
VIT processing module:
ISTRACSM
Control is returned to:
ISTORMVA

This trace record provides the status of each VTALLOC request issued by VTAM components.

If this entry is associated with an event failure (that is, the return code is nonzero), this entry is generated whether the SMS option is in effect. It is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

Byte (hex)
Contents

00–03
Record ID: C"VRSM"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05–0B
Address of the ECB that was posted

10–17
Subtask name

18–1B
Address of the code that is resuming execution

1C–1F
0
00–03
Record ID: C"VTAL"

04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
Flags

Bit
Meaning
0... ....
FREED(EXPLICIT) storage request
1... ....
FREED(TASK) or FREED(JSTASK) storage request

06–07
0

08–0B
Address of area allocated

0C–0F
Subpool number of the area allocated

10–13
Address of the issuer of the VTALLOC macro

14–17
Length of area allocated

18–1B
If the VTALLOC macro is issued by a utility routine, this location contains the address of the issuer of the utility routine.
If the VTALLOC macro is not issued by a utility routine, this location contains 0.

1C–1F
VTALLOC return code

VTFR entry for free storage

Entry:
VTFR

VIT option:
SMS

Event:
Free storage

VIT processing module:
ISTRACSM

Control is returned to:
ISTORMVF

This trace record provides the status of each VTFREE request issued by VTAM components.
VTFR | ID | STORAGE ADDRESS | SUBPOOL NUMBER OR POOL ID | RETURN ADDRESS | LENGTH OF STORAGE FREED | CALLER OF UTILITY OR 0 | 0
---|---|----------------|--------------------------|----------------|------------------------|------------------------|---
00 | 12 | 04 | 08 | 11 | 14 | 17 | 0

**Byte (hex)**

**Contents**

**00–03**

Record ID: C"VTFR"

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05–07**

0

**08–0B**

Address of area to be freed

**0C–0F**

Subpool number of the area allocated

**10–13**

Address of the issuer of the VTFREE macro

**14–17**

Length of area to be freed

**18–1B**

If the VTFREE macro is issued by a utility routine, this location contains the address of the issuer of the utility routine.

If the VTFREE macro is not issued by a utility routine, this location contains 0.

**1C–1F**

0

---

**VWAI entry for VWAIT macro**

**Entry:**

VWAI

**VIT option:**

PSS

**Event:**

VWAIT macro

**VIT processing module:**

ISTRACPS

**Control is returned to:**

Module invoking the INTRACE macro that caused the record to be produced.

This trace record is written when a VTAM module issues a VWAIT macro. The VWAIT macro generates a system WAIT event.

If the "last address in ECB list" field is not 0, then this is a wait on an ECB list. Note that an ECB list is a list of the addresses of ECBs, not a list of ECBs.
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;VWAI&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0B</td>
<td>Address of the last pointer in an ECB list, or 0. If this field is 0, it implies that the previous field points to an ECB instead of an ECB list.</td>
</tr>
<tr>
<td>0C–0F</td>
<td>Address of the ECB that is being waited on, or (if the next field is nonzero) the address of an ECB list</td>
</tr>
<tr>
<td>10–17</td>
<td>Subtask name</td>
</tr>
<tr>
<td>18–1B</td>
<td>Address of the issuer of the VWAIT macro</td>
</tr>
<tr>
<td>1C–1F</td>
<td>0</td>
</tr>
</tbody>
</table>

**WAIT entry for TPWAIT macro**

- **Entry:**
  - WAIT
- **VIT option:**
  - PSS
- **Event:**
  - TPWAIT macro
- **VIT processing module:**
  - ISTRACPS
- **Control is returned to:**
  - ISTAPCTW

This trace record identifies an RPH that has been suspended to wait for some VTAM resource. This entry might be used with the entry immediately preceding it in the table to determine whether the wait is for storage or for a lock. If the preceding entry is for a REQSTORE, a waiting-for-storage condition may exist. If the preceding entry is for a lock request, the wait can be caused by that request. However, VTAM frequently uses a TPWAIT/TPPOST to synchronize intercomponent functions. Therefore, the TPWAIT entry does not necessarily identify an unusual condition.
Byte (hex) Contents
00–03
  Record ID: C"WAIT"

04
  ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05
  0

06
  PAB flag field (PABFLAGS)
    Bit Meaning
    1...  .... PAB is unconditionally scheduled.
    .1...  .... PAB closedown is in progress.
    ..1...  .... PAB is synchronous.
    ....1...  .... PAB extension is present.
    .......  .... Do not dequeue work element.
    ......1...  .... Do not detach the RPH.
    .......1.  .... Indicates a very extended PAB.
    ...... ...1  .... Indicates a slightly extended PAB.

07
  PAB flag field (PABFLGS1)
    Bit Meaning
    1...  .... Switch the PST address of this PAB's major control block to the new PST address contained in DYPNWPST.
    .1...  .... This PAB has a data space extension.
    ..1...  .... This PAB's major control block is an FMCB.
PAB can be referenced in PSW disable mode.

PAB is persistent.

APTERM/APSINIT FMCB during PAB dispatch.

Reserved.

**08–0B**

PST address

**0C–0F**

PAB address

**10–13**

Address of the issuer of the TPWAIT macro

**14–17**

PAB work element queue or next dispatchable queue level for a very extended PAB

**18–1B**

Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the next module to get control), PAB DVT address (high-order bit of X'18' = 0), function, or process.

**1C–1F**

Request parameter header (RPH) address

---

**XBA1 entry for extended buffer list (Part 1)**

**Entry:**

**XBA1**

**VIT option:**

**XBUF**

**Event:**

OPTCD=XBUFLST request on APPCCMD macroinstruction

**VIT processing module:**

ISTITCXB

**Control is returned to:**

ISTPSCFM, ISTPSCFR, ISTPSCKS, or ISTPSCMU

The XBA1 record is created by any of the following conditions:

- An HPDT send is issued (send request with OPTCD=XBUFLST specified).
- An HPDT send is completed and RPLXSRV is not set.
- An HPDT receive (APPCCMD CONTROL=RECEIVE with OPTCD=XBUFLST) is completed.

The entry contains information required to correlate this set of extended buffer list entries to an appropriate APPC entry (either ACA1 or ACP1/ACR1).

The number of trace records required to represent an application request buffer list is variable based on the number of entries present in the buffer list.

The first record in a request is XBA1. This record is followed with as many XBA2 and XBA3 record pairs as is necessary to record the entire application buffer list contents.

There is no limit to the number of trace entries that can be produced. Although VTAM groups records together as a single entity in terms of recording in the trace recording media, there is a finite limit to how many records can be written externally or recorded internally at one time. Because of multiprocessing, other VIT records from unrelated operations can be interleaved in the trace output between the XBUF
entries for a given operation. To provide correlation between groups of XBUF entries, the RPL address is traced in each entry.

|          | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| XBA1     | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|          | J  | D  | 0  |   | RPL| ADDRESS| 0  |   | RPH| ADDRESS|

**Byte (hex)**

**Contents**

**00–03**

Record ID: C"XBA1"

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05–07**

0

**08–0B**

RPL address. This field corresponds to the RPL address of the ACA1, ACP1, or ACR1 entry.

**0C–1B**

0

**1C–1F**

RPH address

**XBA2 entry for extended buffer list (Part 2)**

**Entry:**

XBA2

**VIT option:**

XBUF

**Event:**

OPTCD=XBUFLST request on APPCCMD macroinstruction

**VIT processing module:**

ISTITCXB

This trace record is a continuation of the XBA1 entry. It contains information about an XBUFLST entry.

|          | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| XBA2     | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |     |     |     |     |     |     |     |     |     |     |     |     |     |
| RPL ADDRESS | 0  | S  | O  | Y  | P  | C  | E  | F  | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |     |     |     |     |
| CSMLICENSE | CSM BUFFER TOKEN | CSM DATA SPACE ALET | ADDRESS OF DATA

**Byte (hex)**

**Contents**

**00–03**

Record ID: C"XBA2"

**04–07**

RPL address
08
 0
09
 Buffer source
 80
 ECSA
 40
 Data space
 20
 User data space
 10
 User storage area other than a data space

0A
 Buffer type
 80
 Fixed
 40
 Pageable
 20
 Eligible to be made pageable

0B
 BLXEN_FLAGS flag byte

0C–17
 CSM buffer token

18–1B
 CSM data space ALET

1C–1F
 Address of data

**XBA3 entry for extended buffer list (Part 3)**

**Entry:**

XBA3

**VIT option:**

XBUF

**Event:**

OPTCD=XBUFLST request on APPCCMD macroinstruction

**VIT processing module:**

ISTITCXB

This trace record is a continuation of the XBA2 entry.

<p>| | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>XBA3</td>
<td>RPL ADDRESS</td>
<td>APPL SUPPLIED LENGTH</td>
<td>VTAM ACCEPTED LENGTH</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00 00 00 00 00 00 00 00 00 00 00 00</td>
<td>11 11 11 11 11 11 11 11 11 11 11 11</td>
<td>00 00 00 00 00 00 00 00 00 00 00 00</td>
<td>11 11 11 11 11 11 11 11 11 11 11 11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**
This record is produced when an extended buffer list APPC message unit (AMU) or transmission subsystem control block (TSCB) is built by VTAM.

The number of trace records required to represent a VTAM internal extended buffer list is variable-based on the number of entries in the structure.

The first record in a request is XBI1. This record is followed with as many XBI2 and XBI3 record pairs as is necessary to record the entire VTAM internal extended buffer list contents. The first XBI2 and XBI3 record pair contains information relating to the extended buffer list control entry. Subsequent XBI2 and XBI3 record pairs contain information relating to extended buffer list data entries. In addition, if the internal extended buffer list spans more than one AMU or TSCB, then a new XBI1 record is recorded at the point at which a new AMU or TSCB is used. In this case, the XBI1 record contains the previous AMU or TSCB address so that the relationship between this record and the previous set of records can be maintained.

There is no limit to the number of trace entries that can be produced. Although VTAM groups records together as a single entity in terms of recording in the trace recording media, there is a finite limit to how many records can be written externally or recorded internally at one time. Because of multiprocessing, other VIT records from unrelated operations can be interleaved in the trace output between the XBUF entries for a given operation. To provide correlation between groups of XBUF entries, the AMU or TSCB address is traced in each entry.
00–03  
Record ID: C"XBI1"

04  
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05–07  
0

08–0B  
TSCB or AMU address

0C–0F  
Address of the previous TSCB or AMU address, or zeros if this is the first TSCB or AMU.

10–13  
Return address

14–1B  
0

1C–1F  
RPH address

**XBI2 entry for extended buffer list AMU or TSCB**

**Entry:**  
XBI2

**VIT option:**  
XBUF

**Event:**  
Extended buffer list AMU or TSCB

**VIT processing module:**  
ISTITCXB

This trace record is a continuation of the XBI1 entry. This record contains information about an extended buffer list entry contained in an AMU or TSCB.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;XBI2&quot;</td>
</tr>
<tr>
<td>04–07</td>
<td>XBUFLST TSCB or AMU address</td>
</tr>
<tr>
<td>08</td>
<td>Version of CSM buffer descriptor</td>
</tr>
<tr>
<td>09</td>
<td>Buffer source</td>
</tr>
<tr>
<td>80</td>
<td>ECSA</td>
</tr>
</tbody>
</table>
40  Data space
20  User data space
10  User storage area other than a data space

0A  Buffer type
80  Fixed
40  Pageable
20  Eligible to be made pageable

0B  0
0C–17  Buffer token or zeros
18–1B  CSM data space ALET
1C–1F  Address of data

**XBI3 entry for extended buffer list AMU or TSCB**

Entry:
XBI3

VIT option:
XBUF

Event:
Extended buffer list AMU or TSCB

VIT processing module:
ISTITCXB

This trace record is a continuation of the XBI1 entry. This record contains additional information about an extended buffer list entry contained in an AMU or TSCB.

| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| XBI3 | TSCB OR AMU ADDRESS | LENGTH OF DATA | VTAM IO BUFFER ADDRESS OR 0 | STORAGE RELEASE ROUTINE | X | B | F | L | G | S | 0 |

**Byte (hex)**

**Contents**

**00–03**

Record ID: C"XBI3"
04–07  
XBUFLST TSCB or AMU address

08–0B  
Length of data pointed to by this entry

0C–0F  
VTAM I/O buffer address if entry represents data contained in I/O buffer or zeros

10–13  
Storage release routine

14  
XBFLFLGS byte

15–1F  
0

**XB61 entry for extended buffer list SPAC**

<table>
<thead>
<tr>
<th>Entry:</th>
<th>XB61</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIT option:</td>
<td>XBUF</td>
</tr>
<tr>
<td>Event:</td>
<td>Extended buffer list SPAC</td>
</tr>
<tr>
<td>VIT processing module:</td>
<td>ISTITCXB</td>
</tr>
<tr>
<td>Control is returned to:</td>
<td>INTRACE invoker</td>
</tr>
</tbody>
</table>

This record is produced when VTAM builds an extended buffer list ShortPAC (SPAC).

The number of trace records required to represent a VTAM internal extended buffer list is based on the number of entries in the structure. The first record in a request is XB61. This record is followed with as many XB62 and XB63 record pairs as is necessary to record the entire VTAM internal extended buffer list contents. The first XB62 and XB63 record pair contains information relating to the extended buffer list control entry. Subsequent XB62 and XB63 record pairs contain information relating to extended buffer list data entries. In addition, if the internal extended buffer list spans more than one SPAC, a new XB61 entry is recorded whenever a new SPAC is used. In this case, the XB61 record contains the previous SPAC address so that the relationship between this record and the previous set of records can be maintained.

The number of trace entries that can be produced has no limit. Although VTAM groups records as a single entity in terms of recording in the trace recording media, there is a finite limit to the number of records that can be written externally or recorded internally at one time. Because of multiprocessing, other VIT records from unrelated operations can be interleaved in the trace output between the XBUF entries for a given operation. To provide correlation between groups of XBUF entries, the SPAC address is traced in each entry.

<table>
<thead>
<tr>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
<th>00</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>1A</td>
<td>1B</td>
<td>1C</td>
</tr>
<tr>
<td><strong>X B 6 1</strong></td>
<td><strong>A S I D</strong></td>
<td>0</td>
<td>SPAC ADDRESS</td>
<td>ADDRESS OF PREV SPAC</td>
<td>RETURN ADDRESS</td>
<td>RPH ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

**00 - 03**

Record ID: C"XB61"
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

05 - 07
0

08 - 0F
SPAC address

10 - 17
Address of the previous SPAC address, or zeros if this is the first SPAC.

18 - 1B
Return Address

1C - 1F
RPH address

### XB62 entry for extended buffer list SPAC

The trace record, as a continuation of the XB61 entry, contains information about an extended buffer list entry that is contained in a SPAC.

**Entry:**
XB62

**VIT option:**
XBUF

**Event:**
Extended buffer list SPAC

**VIT processing module:**
ISTITCXB

This trace record is a continuation of the XB61 entry. This record contains information about an extended buffer list entry that is contained in a SPAC.

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>0A</th>
<th>0B</th>
<th>0C</th>
<th>0D</th>
<th>0E</th>
<th>0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
<td>09</td>
<td>0A</td>
<td>0B</td>
<td>0C</td>
<td>0D</td>
<td>0E</td>
<td>0F</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X</th>
<th>B</th>
<th>6</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSMTOKEN</td>
<td>SPAC ADDRESS</td>
<td>CSM STORAGE ADDRESS</td>
<td></td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

<table>
<thead>
<tr>
<th>00 - 03</th>
</tr>
</thead>
</table>
| Record ID: C"XB62"

<table>
<thead>
<tr>
<th>04 - 0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSM TOKEN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10 - 17</th>
</tr>
</thead>
<tbody>
<tr>
<td>XBUFLST SPAC address</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>18 - 1F</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSM storage address version of the CSM buffer descriptor</td>
</tr>
</tbody>
</table>

### XB63 entry for extended buffer list SPAC

The XB63 trace record, as a continuation of the XB62 entry, contains additional information about an extended buffer list entry that is contained in a SPAC.

**Entry:**
XB63
**VIT option:**
XBUF

**Event:**
Extended buffer list SPAC

**VIT processing module:**
ISTITCXB

This trace record is a continuation of the XB62 entry. This record contains additional information about an extended buffer list entry that is contained in a SPAC.

---

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 - 03</td>
<td>Record ID: C&quot;XB63&quot;</td>
</tr>
<tr>
<td>04 - 07</td>
<td>Storage release routine XBUFLST SPAC address</td>
</tr>
<tr>
<td>08 - 0F</td>
<td>XBUFLST SPAC address</td>
</tr>
<tr>
<td>10</td>
<td>XBUFLFLGS byte</td>
</tr>
<tr>
<td>11</td>
<td>Version of the CSM buffer descriptor</td>
</tr>
<tr>
<td>12</td>
<td>Buffer source</td>
</tr>
<tr>
<td>80</td>
<td>ECSA</td>
</tr>
<tr>
<td>40</td>
<td>Data space</td>
</tr>
<tr>
<td>20</td>
<td>User data space</td>
</tr>
<tr>
<td>10</td>
<td>User storage area other than a data space</td>
</tr>
<tr>
<td>08</td>
<td>HVCOMM</td>
</tr>
<tr>
<td>13</td>
<td>Buffer type</td>
</tr>
<tr>
<td>80</td>
<td>Fixed</td>
</tr>
<tr>
<td>40</td>
<td>Pageable</td>
</tr>
<tr>
<td>20</td>
<td>Eligible to be made pageable</td>
</tr>
</tbody>
</table>
**XCFC entry for save message (Part 1)**

**Entry:**
- XCFC

**VIT option:**
- XCF

**Event:**
- A message from another XCF member is saved or a saved message is processed.

**VIT processing module:**
- ISTITCXF

**Control is returned to:**
- ISTTSCMX or ISTTSCDX

The XCFC trace record is written when VTAM cannot immediately receive a message from another VTAM because of a resource shortage or when VTAM is processing a previously saved message.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;XCFC&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–06</td>
<td>0</td>
</tr>
<tr>
<td>07</td>
<td>Request code:</td>
</tr>
<tr>
<td>C&quot;D&quot;</td>
<td>VTAM has discarded the received message.</td>
</tr>
<tr>
<td>C&quot;Q&quot;</td>
<td>VTAM has sent a query to determine how many saved messages exist.</td>
</tr>
<tr>
<td>C&quot;R&quot;</td>
<td>VTAM is redelivering a saved message to the message exit.</td>
</tr>
<tr>
<td>C&quot;S&quot;</td>
<td>VTAM’s first attempt to receive a message failed, so the message has been saved.</td>
</tr>
<tr>
<td>08–17</td>
<td>Original message token value</td>
</tr>
<tr>
<td>18–19</td>
<td>0</td>
</tr>
</tbody>
</table>
XCC2 entry for save message (Part 2)

Entry:
XCC2

VIT option:
XCF

Event:
A message from another XCF member is saved

VIT processing module:
ISTITCXF

This trace record is a continuation of the XCFC entry. It shows additional information about a saved message.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;XCC2&quot;</td>
</tr>
<tr>
<td>04–06</td>
<td>0</td>
</tr>
<tr>
<td>07</td>
<td>Request code</td>
</tr>
<tr>
<td>08–17</td>
<td>Saved message token value</td>
</tr>
<tr>
<td>18–1B</td>
<td>Reason code from an MVS IXCMSGC macroinstruction</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Return code from an MVS IXCMSGC macroinstruction</td>
</tr>
</tbody>
</table>

See z/OS MVS Programming: Sysplex Services Reference for a description of the return codes and reason codes.

XCFJ entry for join XCF group (Part 1)

Entry:
XCFJ

VIT option:
XCF

Event:
VTAM joins an XCF group
VIT processing module:
ISTITCXF

Control is returned to:
ISTTSCIX for XCF connectivity
ISTFSUXJ for multiple node persistent session or coupling facility structure support.

The XCFJ trace record is written when VTAM joins an XCF group. It shows information about the XCF connection established by an MVS IXCJOIN macroinstruction, placing an XCF member in an active state and associating it with an XCF group.

<table>
<thead>
<tr>
<th>00–03</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;XCFJ&quot;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>04</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>05–07</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>08–0F</td>
<td>XCF token value</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10–13</th>
<th>Major control block address of:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• ISTXCF CB for XCF connectivity.</td>
<td></td>
</tr>
<tr>
<td>• 0 for coupling facility structure support.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>14–1B</th>
<th>XCF group name:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• ISTXCFvv for XCF connectivity, where vv is the group ID specified on the XCFGRPID start option. If an XCFGRPID value was not specified, the group name is ISTXCF.</td>
<td></td>
</tr>
<tr>
<td>• ISTCFSvv for coupling facility structure support, where vv is the group ID specified on the XCFGRPID start option. If an XCFGRPID value was not specified, the group name is ISTCFS01.</td>
<td></td>
</tr>
</tbody>
</table>

| 1C–1F | Request parameter header (RPH) address |

**XCFJ2 entry for join XCF group (Part 2)**

**Entry:**
XCFJ2

**VIT option:**
XCF

**Event:**
VTAM joins an XCF group

**VIT processing module:**
ISTITCXF
This trace record is a continuation of the XCFJ entry. It shows additional information about the XCF connection established by an MVS IXCJOIN macroinstruction.

| 00–03 | Record ID: C"XCJ2"
| 04–13 | Group member name comprises:
  • Control point name
  • Network identifier
| 14–15 | MVS &SYSCLONE value for XCF connectivity
  0 for multiple node persistent session support
  The MVS &SYSCLONE value is defined in MVS. For complete information about how the symbol is defined, see z/OS MVS Setting Up a Sysplex and z/OS MVS Initialization and Tuning Reference.
| 16–17 | 0
| 18–1B | Reason code from an MVS IXCJOIN macroinstruction
| 1C–1F | Return code from an MVS IXCJOIN macroinstruction
See z/OS MVS Programming: Sysplex Services Reference for a description of the return codes and reason codes.

**XCFL entry for leave XCF group (Part 1)**

**Entry:**
XCFL

**VIT option:**
XCF

**Event:**
VTAM leaves an XCF group

**VIT processing module:**
ISTITCXF

**Control is returned to:**
ISTDECDH
The XCFL trace record is written when VTAM leaves an XCF group. It shows information about the XCF connection terminated by an MVS IXCLEAVE macroinstruction, placing an XCF member in an undefined state and disassociating it from its XCF group.

<table>
<thead>
<tr>
<th>byte (hex)</th>
<th>contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;XCFL&quot;</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05–07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>XCF token value</td>
</tr>
<tr>
<td>10–13</td>
<td>XCF control block address (ISTXFCB)</td>
</tr>
<tr>
<td>14–1B</td>
<td>XCF group name</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Request parameter header (RPH) address</td>
</tr>
</tbody>
</table>

**XCL2 entry for leave XCF group (Part 2)**

**Entry:**
XCL2

**VIT option:**
XCF

**Event:**
VTAM leaves an XCF group

**VIT processing module:**
ISTITCXF

This trace record is a continuation of the XCFL entry. It shows additional information about the XCF connection established or terminated by an MVS IXCLEAVE macroinstruction.
00–03
Record ID: C"XCFM"

04–13
Group member name comprises:
• Control point name
• Network identifier

14–15
MVS &SYSCONFIG2 value
The MVS &SYSCONFIG2 value is defined in MVS. For complete information about how the symbol is defined, see z/OS MVS Setting Up a Sysplex and z/OS MVS Initialization and Tuning Reference.

16–17
0

18–1B
Reason code from an MVS IXCLEAVE macroinstruction

1C–1F
Return code from an MVS IXCLEAVE macroinstruction
See z/OS MVS Programming: Sysplex Services Reference for a description of the return codes and reason codes.

XCFM entry for XCF group member (Part 1)

Entry:
XCFM

VIT option:
XCF

Event:
A member joins or leaves an XCF group

VIT processing module:
ISTITCXF

Control is returned to:
ISTTSCG for XCF connectivity. ISTFSXGE for multiple node persistent session support

The XCFM trace record is written when an adjacent VTAM joins or leaves an XCF group, or when an initial message is received from an adjacent VTAM after this VTAM joins the group.

Byte (hex) Contents
00–03 Record ID: C"XCFM"
04 ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
Old event state when a member joins or leaves the group

0 when an initial message is received

The event states are defined in the group exit parameter list (IXCYGEPL). See z/OS MVS Data Areas at z/OS Internet Library for an explanation of the event states.

New event state when a member joins or leaves the group

0 when an initial message is received

The event states are defined in the group exit parameter list (IXCYGEPL). See z/OS MVS Data Areas at z/OS Internet Library for an explanation of the event states.

XCF token value

AMRU address for XCF connectivity. 0 for coupling facility support

XCF group name:

- ISTXCFvv for XCF connectivity, where vv is the group ID supplied on the XCFGRPID start option. If XCFGRPID was not specified, the group name is ISTXCF.
- ISTCFSvv for coupling facility structure support, where vv is the group ID supplied on the XCFGRPID start option. If XCFGRPID was not specified, the group name is ISTCFS01.

Request parameter header (RPH) address

XCM2 entry for XCF group member (Part 2)

Entry: XCM2

VIT option: XCF

Event: A member joins or leaves an XCF group

VIT processing module: ISTITCXF

This trace record is a continuation of the XCFM entry. It shows additional information about the XCF member when an adjacent VTAM joins or leaves an XCF group, or when an initial message is received from an adjacent VTAM after this VTAM joins the group.
Byte (hex)  
Contents  
00–03    
Record ID: C"XCM2"  
04–13    
Group member names comprise:  
• Control point name  
• Network identifier  
14–15    
MVS &SYSCLONE value for XCF connectivity  
0 for coupling facility structure support  
The MVS &SYSCLONE value is defined in MVS. For complete information about how the symbol is defined, see z/OS MVS Setting Up a Sysplex and z/OS MVS Initialization and Tuning Reference.  
16    
Event type when a member joins or leaves the group  
0 when an initial message is received  
The event type is defined by the group exit parameter list (IXCYGEPL). See z/OS MVS Data Areas at z/OS Internet Library for an explanation of the event types.  
17–1F    
0  
XCFR entry for receive message (Part 1)  
Entry:  
XCFR  
VIT option:  
XCF  
Event:  
A message is received from another XCF group member.  
VIT processing module:  
ISTITCXF  
Control is returned to:  
ISTTSCMX, ISTTSCBX, or ISTFSXME  
The XCFR trace record is written when VTAM receives a message from another active member of an XCF group. It shows information about the message when an MVS IXCMSGI macroinstruction is invoked.  
Byte (hex)  
Contents  
00–03    
Record ID: C"XCFR"  
04    
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.  
05    
0  
06    
Flag field from MSGCNTRL area  
07    
State of the link if user type=C'X'; otherwise, 0.
08–0F
XCF token value
10
0
11
User type:
C'X'
VTAM/XCF
C'M'
MNPS
12–13
Length of message
14–17
XFNFCB address if user type is C'X'; otherwise, 0.
18–1B
Data element address
1C–1F
Request parameter header (RPH) address

**XCR2 entry for receive message (Part 2)**

**Entry:**
XCR2

**VIT option:**
XCF

**Event:**
A message is received from another XCF group member.

**VIT processing module:**
ISTITCXF

This trace record is a continuation of the XCFR entry. It is written when an MVS IXCMSGI macroinstruction is invoked. It shows the return code and reason code from the macroinstruction.

See [z/OS MVS Programming: Sysplex Services Reference](https://www.ibm.com/support/knowledgecenter/SSDZZU_2.2.0/com.ibm.zos.bks.2.2.0book.c) for a description of the return codes and reason codes.

**Mapping for Usertype=C'M'**

When Usertype (byte 6) is C'M', the mapping for XCR2 is as follows:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>XCR2</td>
</tr>
<tr>
<td>0</td>
<td>TYPE</td>
</tr>
<tr>
<td>0</td>
<td>STATUS</td>
</tr>
<tr>
<td>0</td>
<td>APPLNAME</td>
</tr>
<tr>
<td>0</td>
<td>REASON</td>
</tr>
<tr>
<td>0</td>
<td>RETURN</td>
</tr>
</tbody>
</table>

**VTAM internal trace (VIT) record descriptions** 709
00–03
  Record ID: C"XCR2"

04–05
  0

06
  User type:
    C'M'
    MNPS

07
  Message reply status

08–0F
  Application name

10–17
  0

18–1B
  Reason code from an MVS IXCMSGI macroinstruction

1C–1F
  Return code from an MVS IXCMSGI macroinstruction

**Mapping for Usertype=C'X'**

When Usertype (byte 6) is C'X', the mapping for XCR2 is as follows:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;XCR2&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>User type:</td>
</tr>
<tr>
<td></td>
<td>C'X'</td>
</tr>
<tr>
<td></td>
<td>VTAM/XCF</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–17</td>
<td>Message token</td>
</tr>
<tr>
<td>18–1B</td>
<td>Reason code from an MVS IXCMSGI macroinstruction</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Return code from an MVS IXCMSGI macroinstruction</td>
</tr>
</tbody>
</table>
**XCFS entry for send message (Part 1)**

**Entry:**

XCFS

**VIT option:**

XCF

**Event:**

A message is sent from this VTAM to another XCF group member

**VIT processing module:**

ISTITCXF

**Control is returned to:**

ISTTSCOX or ISTFSPFN

The XCFS trace record is written when VTAM sends a message to another active member of an XCF group. It shows information about the message when an MVS IXCMSGO macroinstruction is invoked.

![Trace record format](image)

**Byte (hex)**

**Contents**

**00–03**

Record ID: C"XCFS"

**04**

ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

**05**

0

**06**

Flag field from MSGCNTRL area

**07**

State of the link if user type=C'X'; otherwise, 0.

**Note:** If the XFNCB address is 0, the link state is also 0.

**08–0F**

XCF token value

**10**

0

**11**

User type:

C'X'

VTAM/XCF

C'M'

MNPS

**12–13**

Length of message
XCS2 entry for send message (Part 2)

Entry:
XCS2

VIT option:
XCF

Event:
A message is sent from this VTAM to another XCF group member

VIT processing module:
ISTITCXF

This trace record is a continuation of the XCFS entry. It is written when an MVS IXCMSGO macroinstruction is invoked and either the return code is greater than 0 or the message token was obtained from XCF by VTAM request on the macroinstruction. It shows the return code and reason code from the macroinstruction, and the message token if obtained by VTAM.

See z/OS MVS Programming: Sysplex Services Reference for a description of the return codes and reason codes.

Mapping for Usertype=C'M'

When Usertype (byte 6) is C'M', the mapping for XCS2 is as follows:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;XCS2&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>User type:</td>
</tr>
<tr>
<td></td>
<td>C'M'</td>
</tr>
<tr>
<td></td>
<td>MNPS</td>
</tr>
<tr>
<td>07</td>
<td>Message reply status if usertype=C'M'.</td>
</tr>
<tr>
<td>08–0F</td>
<td>Application name</td>
</tr>
</tbody>
</table>

See z/OS Communications Server: SNA Diagnosis Volume 2: FFST Dumps and the VIT
10–17
  0
18–1B
  Reason code from an MVS IXCMSGO macroinstruction
1C–1F
  Return code from an MVS IXCMSGO macroinstruction

Mapping for Usertype=C’X’

When Usertype (byte 6) is C’X’, the mapping for XCS2 is as follows:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C&quot;XCS2&quot;</td>
</tr>
<tr>
<td>04–05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>User type:</td>
</tr>
<tr>
<td></td>
<td>C’X’</td>
</tr>
<tr>
<td></td>
<td>VTAM/XCF</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–17</td>
<td>Message token or 0</td>
</tr>
<tr>
<td>18–1B</td>
<td>Reason code from an MVS IXCMSGO macroinstruction</td>
</tr>
<tr>
<td>1C–1F</td>
<td>Return code from an MVS IXCMSGO macroinstruction</td>
</tr>
</tbody>
</table>

XCFX entry for VTAM XCF message exit or notify exit driven

Entry:
  XCFX
VIT option:
  XCF
Event:
  VTAM XCF message exit or notify exit entered
VIT processing module:
  ISTITCXF
Control is returned to:
  ISTTSCMX (message exit), ISTTSCFX (notify exit), or ISTFSXME

The XCFX trace record is written when either the VTAM XCF message exit or notify exit is entered.
Byte (hex)
Contents
01–03
Record ID: C'XCFX'
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.
05
0
06
Flag Field (Message exit)
Bit
  Meaning
  B'1... ....'
  Message exit was solicited by the user.
  B'.1.. ....'
  Sender requested that XCF manage the response to this message.
  B'..1. ....'
  This message is a response being managed by XCF.
  B'...1 ....'
  Some portion of the message was delivered by the message-in service.
  B'.... 1...' 
  Message was saved with message control SAVEMSG service.
  B'..... 1.'
  Sender requested ordered message delivery.
  B'..... ...X.'
  0
  B'..... ....1'
  Additional data presented to the message exit in message exit extension.
Flag Field (Notify exit)
Bit
  Meaning
  B'1... ....'
  Sender specified SENDTO(GROUP) on the IXCMSGO invocation.
  B'.1.. ....'
  Sender requested that XCF manage the response to this message.
  B'..1. ....'
  This message is a response being managed by XCF.
  B'...1 ....'
  required sends not initiated by the message-out service.
  B'.... 1...' 
  Expected response not received.
Message considered to be complete.

Message did not complete within the timeout period.

Message was canceled before normal completion occurred.

**Function Code**

C'C' - Message exit for CFS
C'M' - Message exit for VTAM/XCF
C'N' - Notify exit for VTAM/XCF

**Original Message token**

**Source message code**

---

**XID entry for exchange ID (Part 1)**

**Entry:**
- XID

**VIT option:**
- CIA

**Event:**
- Exchange ID (Part 1)

**VIT processing module:**
- ISTRACCI

**Control is returned to:**
- ISTTSCCU or ISTTSC8X

This trace record contains information about inbound or outbound exchange ID processing.

<table>
<thead>
<tr>
<th>00–03</th>
<th>04–07</th>
<th>08–0B</th>
<th>0C–1F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record ID:</td>
<td>NCB address</td>
<td>Boundary physical unit block (BPB) address or 0</td>
<td>20 bytes of XID data</td>
</tr>
</tbody>
</table>

**Byte (hex)**

**Contents**

00–03
- Record ID:
  - "XIDI" for inbound XID
  - "XIDO" for outbound XID

04–07
- NCB address

08–0B
- Boundary physical unit block (BPB) address or 0

0C–1F
- 20 bytes of XID data
XID2 entry for exchange ID (Part 2)

Entry: XID2
VIT option: CIA
Event: Exchange ID (Part 2)
VIT processing module: ISTRACCI

This trace record is a continuation of the XID trace record.

| 00 03 04 05 06 07 08 09 A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 | 05 06 07 08 09 A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 |
|-------------------------------------------------------------------------------------------------------------------------------------|
| XID2                                                                                                                                  |
| ADDITIONAL XID DATA                                                                                                                   |

Byte (hex)
Contents

00–03
Record ID: C"XID2"

04–1F
28 bytes of additional XID data

XPST entry for TPSWPST macro

Entry: XPST
VIT option: PSS
Event: TPSWPST macro
VIT processing module: ISTRACPS
Control is returned to: ISTAPCPS

This trace record identifies a VTAM process (PAB) that is switching PSTs during termination of a task. Switching PSTs causes the PAB to be scheduled for dispatch under the new PST.

| 00 01 02 03 04 05 06 07 08 09 A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 | 01 02 03 04 05 06 07 08 09 A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 |
|-------------------------------------------------------------------------------------------------------------------------------------|
| XPST                                                                                                                                  |
| I D F L A G S NEW PST ADDRESS PAB ADDRESS ISSR WEQ MOD NAME OR DVT ADDRESS RPH ADDRESS OR 0 |

Byte (hex)
Contents
04
ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.

06
PAB flag field (PABFLAGS)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>PAB is unconditionally scheduled.</td>
</tr>
<tr>
<td>.1..</td>
<td>PAB closedown is in progress.</td>
</tr>
<tr>
<td>..1.</td>
<td>PAB is synchronous.</td>
</tr>
<tr>
<td>...1</td>
<td>PAB extension is present.</td>
</tr>
<tr>
<td>.... 1</td>
<td>Do not dequeue work element.</td>
</tr>
<tr>
<td>.... .1</td>
<td>Do not detach the RPH.</td>
</tr>
<tr>
<td>.... ..1</td>
<td>Indicates a very extended PAB.</td>
</tr>
<tr>
<td>.... ...1</td>
<td>Indicates a slightly extended PAB.</td>
</tr>
</tbody>
</table>

07
PAB flag field (PABFLGS1)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1...</td>
<td>Switch the PST address of this PAB’s major control block to the new PST address contained in DYPNWPST.</td>
</tr>
<tr>
<td>.1..</td>
<td>This PAB has a data space extension.</td>
</tr>
<tr>
<td>..1.</td>
<td>This PAB’s major control block is an FMCB.</td>
</tr>
<tr>
<td>...1</td>
<td>PAB can be referenced in PSW disable mode.</td>
</tr>
<tr>
<td>.... 1</td>
<td>PAB is persistent.</td>
</tr>
<tr>
<td>.... .1</td>
<td>APSTERM/APSINIT FMCB during PAB dispatch.</td>
</tr>
<tr>
<td>.... ..xx</td>
<td></td>
</tr>
</tbody>
</table>

08–0B
New PST address under which this PAB will now be dispatched

0C–0F
PAB address
3270 entry for 3270 Intrusion Detection Services

Entry:
3270

VIT option:
PIU

Event:
Internal processing during the analysis of a 3270 data stream buffer

VIT processing module:
ISTITC32

This record and the 3271 record are written during the analysis of the 3270 data steam buffer.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C'3270'</td>
</tr>
<tr>
<td>04</td>
<td>ID is the primary address space ID (ASID). This field is 0 if the ASID is greater than X'FF'.</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>ERC is an error code.</td>
</tr>
<tr>
<td>07</td>
<td>0</td>
</tr>
<tr>
<td>08–0F</td>
<td>64-bit address of the session control block (ISTS3270)</td>
</tr>
<tr>
<td>10–17</td>
<td>64-bit address of a PIU buffer (ISTB3270)</td>
</tr>
<tr>
<td>18–1B</td>
<td>Incident identifier</td>
</tr>
<tr>
<td>1A–1B</td>
<td>RU sequence number field</td>
</tr>
</tbody>
</table>
3271 entry for 3270 Intrusion Detection Services

Entry: 3271
VIT option: PIU
Event: Records when the 3270 entry is written
VIT processing module: ISTITC32

This trace record is a continuation of the 3270 trace record. It contains the in-flight recording of events.

```
<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>Record ID: C’3271’</td>
</tr>
<tr>
<td>04</td>
<td>The index into the in-flight recording area</td>
</tr>
<tr>
<td>05–07</td>
<td>Bits 28 to 39 of the STCK time stamp when this record was created</td>
</tr>
<tr>
<td>08–09</td>
<td>A code-data pair that identifies an in-flight trace event for the session</td>
</tr>
<tr>
<td>0A–1F</td>
<td>Additional code-data pairs</td>
</tr>
</tbody>
</table>
```
Appendix C. Internal topology traces

The internal topology traces provide a record of the creation, update, and deletion of TRS (topology and routing services) topology records. Trace data for the internal topology traces is always automatically recorded.

There are three locations where topology tracing is done:

- In an NDREC (node record) trace table following the NDREC control block, where the creation and update of a node record is recorded.
- In a TGREC (TG record) trace table following the TGREC control block, where the creation and update of a TG record is recorded.
- In a common TRS trace table, where the deletion of NDRECs and TGRECs are recorded.

The NDREC trace table

The NDREC trace table contains entries describing the creation and update of a node record. It is located after the CV45 in an NDREC. It is pointed to by a trace pointer in the NDREC. The first 8 bytes of the NDREC trace table is the header, which contains the following information:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>The current RSN (resource sequence number) of the last update to this NDREC. The current RSN is used when recording TDUs (topology database updates) related to the NDREC. Because duplicate TDU information can be received from multiple adjacent nodes, the RSN in the TDU is compared to the current RSN in the NDREC trace. The TDU will be recorded only in the NDREC trace if the RSN in the TDU is greater than the current RSN in the NDREC trace, indicating that the information in the TDU is more recent than that in the NDREC.</td>
</tr>
<tr>
<td>04–07</td>
<td>Address of the current NDREC trace table entry. This is the last trace entry where data was recorded.</td>
</tr>
</tbody>
</table>

Three NDREC trace table entries are displayed after the header. The information about the creation of the NDREC is recorded in the first NDREC trace entry. This entry remains unchanged for the life of the NDREC. Recording of trace events will continue in the remaining two trace table entries, with the trace wrapping back to the second entry after the third entry is completed.

NDREC trace record descriptions

This topic contains the NDREC trace record descriptions. The entries are listed alphabetically by entry name.

NDREC CPC entry for CP-CP session changes

Entry: CPC

Event: CP-CP session changes

Modules Generating Entry: ISTTRPCS

These trace records are issued when the status of a CP-CP session with this node changes.
Byte (hex) Contents
00–02 Record ID: C"CPC"
03 Action
  • C"U" Update CP-CP session status
04–06 Flag bytes from the CV4580
07 Number of days left before NDREC is garbage collected
08 Flag byte
 Bit Meaning
11... Node type from NDREC
   'B'00'
   End node
   'B'01'
   Network node
   'B'10'
   Virtual node
   ... Garbage collection indicator from NDREC
   ... Adjacent node indicator from NDREC
   .... DLUR end node indicator from NDREC
09–17 0
18–1B Resource sequence number from CV4580
1C–1F First word of timestamp for trace entry
NDREC HST entry for host node create

Entry:
HST

Event:
Host node record creation

Modules Generating Entry:
ISTTRCIT

This trace record is issued when the node record for the host node is created.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;HST&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Action</td>
</tr>
<tr>
<td>00–06</td>
<td>Flag bytes from the CV4580</td>
</tr>
<tr>
<td>07</td>
<td>Number of days left before NDREC is garbage collected</td>
</tr>
<tr>
<td>08</td>
<td>Flag byte</td>
</tr>
<tr>
<td>Bit</td>
<td>Meaning</td>
</tr>
<tr>
<td>11.. .</td>
<td>Node type from NDREC</td>
</tr>
<tr>
<td>B'00'</td>
<td>End node</td>
</tr>
<tr>
<td>B'01'</td>
<td>Network node</td>
</tr>
<tr>
<td>B'10'</td>
<td>Virtual node</td>
</tr>
<tr>
<td>09–17</td>
<td>0</td>
</tr>
<tr>
<td>18–1B</td>
<td>Resource sequence number from CV4580</td>
</tr>
<tr>
<td>1C–1F</td>
<td>First word of timestamp for trace entry</td>
</tr>
</tbody>
</table>
NDREC MOD entry for MODIFY TOPO command

Entry: MOD

Event: MODIFY TOPO operator command

Modules Generating Entry: ISTTROMT

This trace record is issued when the MODIFY TOPO command with the FUNCTION=QUIESCE or FUNCTION=NORMAL operand is entered by an operator.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;MOD&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Action</td>
</tr>
<tr>
<td>• C&quot;Q&quot; FUNCTION=QUIESCE</td>
<td></td>
</tr>
<tr>
<td>• C&quot;N&quot; FUNCTION=NORMAL</td>
<td></td>
</tr>
<tr>
<td>04–06</td>
<td>Flag bytes from the CV4580</td>
</tr>
<tr>
<td>07</td>
<td>Number of days left before NDREC is garbage collected</td>
</tr>
<tr>
<td>08</td>
<td>Flag byte</td>
</tr>
</tbody>
</table>

Bit Meaning

11 . . . . . . Node type from NDREC
B’00’ End node
B’01’ Network node
B’10’ Virtual node
..1 . . . . Garbage collection indicator from NDREC
...1 . . . Adjacent node indicator from NDREC
### NDREC TDU entry for topology database update

**Entry:**
TDU

**Event:**
Topology database update for a node

**Modules Generating Entry:**
ISTTRINP

This trace record is issued when a node record is created or updated when a TDU is received for a node and the information in the TDU about that node is more recent than the information already in the NDREC.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C“TDU”</td>
</tr>
<tr>
<td>03</td>
<td>Action</td>
</tr>
<tr>
<td></td>
<td>• C“C” Create node record</td>
</tr>
<tr>
<td></td>
<td>• C“U” Update node characteristics</td>
</tr>
<tr>
<td></td>
<td>• C“R” Node has changed node roles</td>
</tr>
<tr>
<td>04–06</td>
<td>Flag bytes from the CV4580</td>
</tr>
<tr>
<td>07</td>
<td>Flag byte</td>
</tr>
</tbody>
</table>

---

**Byte (hex) Contents**

<table>
<thead>
<tr>
<th>00–02</th>
<th>TDU ACTION</th>
<th>V4580 FLAGS</th>
<th>ADJACENT NODE SENDING TDU</th>
<th>CURRENT FRSN</th>
<th>LAST FRSN</th>
<th>V4580 RSN</th>
<th>TIME-Stamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00</td>
<td>00</td>
<td>01</td>
<td>0</td>
<td>3</td>
<td>03</td>
<td>00</td>
</tr>
<tr>
<td>01</td>
<td>02</td>
<td>03</td>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
<td>08</td>
</tr>
<tr>
<td>08</td>
<td>09</td>
<td>AB</td>
<td>CD</td>
<td>DE</td>
<td>0F</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td>19</td>
<td>1A</td>
<td>1B</td>
<td>1C</td>
<td>1D</td>
<td>1E</td>
<td>1F</td>
<td>20</td>
</tr>
</tbody>
</table>
Bit
Meaning

11....
   Node type from NDREC
B'00'
   End node
B'01'
   Network node
B'10'
   Virtual node

1....
   Garbage collection indicator from NDREC

1....
   Adjacent node indicator form NDREC

1...1...
   DLUR end node indicator from NDREC

08–0F
   CPNAME of the adjacent node sending the TDU. The CPNAME is not network qualified. This is the node
   that forwarded the TDU and is not necessarily the originator of the TDU information.

10–13
   Current FRSN (flow reduction sequence number)

14–17
   Last FRSN previously sent from the adjacent node that forwarded this TDU

18–1B
   Resource sequence number from CV4580

1C–1F
   First word of timestamp for trace entry

**NDREC TGU entry for TG update**

**Entry:**
   TGU

**Event:**
   Node created or updated because of a TG update signal

**Modules Generating Entry:**
   ISTTRTLT, ISTTRTGG

This trace record is issued when a node record is created or node information is changed because of a TG
update signal.
Byte (hex)
Contents
00–02
  Record ID: C"TGU"
03
  Action
  • C"C" Create node record
  • C"U" Update node characteristics
  • C"R" Node has changed node roles
04–06
  Flag bytes from the CV4580
07
  Number of days left before NDREC is garbage collected
08
  Flag byte
  Bit
  Meaning
  11.. ....
  Node type from NDREC
  B'00' End node
  B'01' Network node
  B'10' Virtual node
  ...1 ....
  Garbage collection indicator from NDREC
  ....1 ....
  Adjacent node indicator from NDREC
  .... 1...
  DLUR end node indicator from NDREC
09–17
  0
18–1B
  Resource sequence number from CV4580
1C–1F
  First word of timestamp for trace entry

NDREC TOP entry for topology data set create

Entry:
  TOP
Event:
  Node created from a checkpointed topology data set
Modules Generating Entry:
  ISTTRHND

This trace record is issued when a node record is created from a checkpointed topology data set.
Byte (hex)

Contents

00–02
  Record ID: C'TOP'

03
  Action
    • C'C' Create node record

04–06
  Flag bytes from the CV4580

07
  Number of days left before NDREC is garbage collected

08
  Flag byte

Bit
  Meaning

11.. . . . .
  Node type from TOPO data set
    B'00'
      End node
    B'01'
      Network node
    B'10'
      Virtual node

...1 . . . .
  Adjacent node indicator from the TOPO data set

...1 ....
  Indicator of adjacent node’s unknown control vector support, from the TOPO data set

09–0B
  0

0C–0F
  Current flow reduction sequence number (FRSN), from the TOPO data set, for topology database updates (TDUs) that include unknown topology control vectors (any topology control vectors other than CV44 and CV45)

10–13
  Current FRSN, from the TOPO data set, for TDUs that do not include unknown topology control vectors (any topology control vectors other than CV44 and CV45)

14–17
  Last FRSN, from the TOPO data set, that is received from this adjacent node
The TGREC trace table

The TGREC trace table contains entries describing the creation and update of a TG record. It is located following the CV46 in a TGREC. A trace pointer in the TGREC points to it. The first 8 bytes of the TGREC trace table is the header, which contains the following information:

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–03</td>
<td>The current RSN (resource sequence number) of the last update to this TGREC. The current RSN is used when recording TDUs (topology database updates) related to the TGREC. Because duplicate TDU information can be received from multiple adjacent nodes, the RSN in the TDU is compared to the current RSN in the TGREC trace. The TDU will be recorded only in the TGREC trace if the RSN in the TDU is greater than the current RSN in the TGREC trace, indicating that the information in the TDU is more recent than that in the TGREC.</td>
</tr>
<tr>
<td>04–07</td>
<td>Address of the current TGREC trace table entry. This is the last trace entry where data was recorded.</td>
</tr>
</tbody>
</table>

Five TGREC trace table entries are displayed after the header. The information about the creation of the TGREC is recorded in the first TGREC trace entry. This entry remains unchanged for the life of the TGREC. Recording of trace events will continue in the remaining four trace table entries, with the trace wrapping back to the second entry after the fifth entry is completed.

TGREC trace record descriptions

This topic contains the TGREC trace record descriptions. The entries are listed alphabetically by entry name.

TGREC CPC entry for CP-CP session changes for a TG

Entry: CPC
Event: CP-CP session status changes in a TG record
Modules Generating Entry: ISTTRPCS

This trace record is issued for a TG record when the status of a CP-CP session using a local TG has changed.
### Byte (hex)

**Contents**

**00–02**  
Record ID: C"CPC"

**03**  
Action  
- C"U" Update TG characteristics

**04**  
Status byte from the CV47

**05**  
Flag byte from the CV4680

**06**  
Flag byte

**Bit**  
**Meaning**

| 111. . . . . | TG type from TGREC |
| B'000' | Endpoint TG |
| B'001' | Intermediate routing TG |
| B'010' | Interchange TG |
| B'011' | Intersubnet TG |
| B'100' | Branch Extender TG |

| . . . 1 . . . | Garbage collection indicator from TGREC |

| . . . 1 . . . | Adjacent node’s HPRTT value from TGREC |

| . . . . 11. | Local node’s HPR/HPRTT values from TGREC |

| . . . . . 1 | TG carries conwinner CP-CP session indicator from TGREC |

**07**  
Number of days left before TGREC is garbage collected
Flag2 byte

Bit Meaning

11... CP-CP session status from CP status IPS
   B’01’ Pending active
   B’11’ Active
   B’00’ Inactive

11... CP-CP session type from CP status IPS
   B’01’ Conwinner
   B’00’ Conloser
   B’10’ Both conwinner and conloser

1... Adjacent node type from CP status IPS
   B’0’ End node
   B’1’ Network node

1... Adjacent node’s non-native indicator from CP status IPS
   B’0’ Native
   B’1’ Non-native (can still have the same NETID)

0... Adjacent node’s border node indicator from CP status IPS
   B’0’ Not a BN
   B’1’ BN

001... Adjacent node’s border node supported indicator from CP status IPS
   B’0’ BN not supported
   B’1’ BN supported

09 TG number from CP status IPS (from RTP only)

0A-0B 0
**0C-0F**  
Conwinner CGID from CP status IPS

**10-13**  
Conloser CGID from CP status IPS

**14-17**  
Last FRSN received by adjacent node (from this host node) from CP status IPS

**18-1B**  
Resource sequence number from CV47

**1C-1F**  
First word of timestamp for trace entry

---

**TGREC MOD entry for MODIFY TOPO command**

**Entry:**  
MOD

**Event:**  
MODIFY TOPO operator command

**Modules Generating Entry:**  
ISTTROMT

This trace record is issued when the MODIFY TOPO command with the FUNCTION=QUIESCE or FUNCTION=NORMAL operand is entered by an operator.

---

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00-02</td>
<td>Record ID: C&quot;MOD&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Action</td>
</tr>
<tr>
<td></td>
<td>• C&quot;Q&quot; FUNCTION=QUIESCE</td>
</tr>
<tr>
<td></td>
<td>• C&quot;N&quot; FUNCTION=NORMAL</td>
</tr>
<tr>
<td>04</td>
<td>Status byte from the CV47</td>
</tr>
<tr>
<td>05</td>
<td>Flags byte from the CV4680</td>
</tr>
<tr>
<td>06</td>
<td>Flag byte</td>
</tr>
</tbody>
</table>

**Bit**  
**Meaning**
TG type from TGREC

'B'000'
   Endpoint TG

'B'001'
   Intermediate routing TG

'B'010'
   Interchange TG

'B'011'
   Intersubnet TG

'B'100'
   Branch extender TG

Garbage collection indicator from TGREC

Adjacent node's HPRTT value from TGREC

Local node's HPR/HPRTT values from TGREC

Number of days left before TGREC is garbage collected

Scope of the MODIFY TOPO command

• C"L" SCOPE = LOCAL
• C"N" SCOPE = NETWORK

Resource sequence number from CV47

First word of timestamp for trace entry

TGREC TDU entry for topology database update

Entry:
   TDU

Event:
   Topology database update for a TG

Modules Generating Entry:
   ISTTRIAT

This trace record is issued when a TG record is created or updated when a TDU is received for a TG and the information in the TDU about that TG is more recent than the information already in the TGREC.
### Byte (hex)

#### Contents

<table>
<thead>
<tr>
<th>byte range</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;TDU&quot;</td>
</tr>
<tr>
<td>03</td>
<td>Action</td>
</tr>
<tr>
<td></td>
<td>• C&quot;C&quot; Create TG record</td>
</tr>
<tr>
<td></td>
<td>• C&quot;I&quot; TG became inoperative</td>
</tr>
<tr>
<td></td>
<td>• C&quot;O&quot; TG became operational</td>
</tr>
<tr>
<td>04</td>
<td>Status byte from the CV47</td>
</tr>
<tr>
<td>05</td>
<td>Flag byte from the CV4680</td>
</tr>
<tr>
<td>06</td>
<td>Flag byte</td>
</tr>
</tbody>
</table>

#### Bit

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>111.</td>
<td>TG type from TGREC</td>
</tr>
<tr>
<td>B'000'</td>
<td>Endpoint TG</td>
</tr>
<tr>
<td>B'001'</td>
<td>Intermediate routing TG</td>
</tr>
<tr>
<td>B'010'</td>
<td>Interchange TG</td>
</tr>
<tr>
<td>B'011'</td>
<td>Intersubnet TG</td>
</tr>
<tr>
<td>B'100'</td>
<td>Branch extender TG</td>
</tr>
<tr>
<td>....1</td>
<td>Garbage collection indicator from TGREC</td>
</tr>
<tr>
<td>....1</td>
<td>Adjacent node’s HPRTT value from TGREC</td>
</tr>
<tr>
<td>....11</td>
<td>Local node’s HPR/HPRTT values from TGREC</td>
</tr>
</tbody>
</table>

### Number of days left before TGREC is garbage collected
**08–0F**  
CPNAME of the adjacent node sending the TDU. The CPNAME is not network qualified. This is the node that forwarded the TDU and is not necessarily the originator of the TDU information.

**10–13**  
Current FRSN (flow reduction sequence number)

**14–17**  
Last FRSN previously sent from the adjacent node that forwarded this TDU

**18–1B**  
Resource sequence number from CV47

**1C–1F**  
First word of timestamp for trace entry

### TGREC TGU entry for TG update

**Entry:**  
TGU

**Event:**  
TG created or updated because of a TG update signal

**Modules Generating Entry:**  
ISTTRTG

This trace record is issued when a TG record is created or the TG status is changed because of a TG update signal.

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;TGU&quot;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>03</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• C&quot;C&quot; Create TG record</td>
</tr>
<tr>
<td></td>
<td>• C&quot;I&quot; TG became inoperative</td>
</tr>
<tr>
<td></td>
<td>• C&quot;O&quot; TG became operational</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>04</th>
<th>Status byte from the CV47</th>
</tr>
</thead>
<tbody>
<tr>
<td>05</td>
<td>Flag byte from the CV4680</td>
</tr>
<tr>
<td>06</td>
<td>Flag byte</td>
</tr>
</tbody>
</table>

---

**Table: TGREC TGU entry**

<table>
<thead>
<tr>
<th>Byte (hex)</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>00–02</td>
<td>Record ID: C&quot;TGU&quot;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>03</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• C&quot;C&quot; Create TG record</td>
</tr>
<tr>
<td></td>
<td>• C&quot;I&quot; TG became inoperative</td>
</tr>
<tr>
<td></td>
<td>• C&quot;O&quot; TG became operational</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>04</th>
<th>Status byte from the CV47</th>
</tr>
</thead>
<tbody>
<tr>
<td>05</td>
<td>Flag byte from the CV4680</td>
</tr>
<tr>
<td>06</td>
<td>Flag byte</td>
</tr>
</tbody>
</table>
Bit
Meaning
111. ....
   TG type from TGREC
B'000'
   Endpoint TG
B'001'
   Intermediate routing TG
B'010'
   Interchange TG
B'011'
   Intersubnet TG
B'100'
   Branch extender TG
.... 1 ....
   Garbage collection indicator from TGREC
.... 1 ....
   Adjacent node's HPRTT value from TGREC
.... .11.
   Local node's HPR/HPRTT values from TGREC
07
   Number of days left before TGREC is garbage collected
08–13
   0
14–17
   Information from the TG update signal
14
   TG type
X'01'
   Endpoint TG
X'02'
   Interchange TG
X'03'
   Intermediate routing TG
X'04'
   Intersubnet TG
X'05'
   Branch Extender TG
15
   TG status
X'01'
   Active
X'02'
   Inactive
16
   Partner node type
X'01'
   Network node
X'02'  
Virtual node

X'03'  
End node

17  
Direction of OP/INOP status

X'01'  
IN

X'02'  
OUT

X'03'  
IN/OUT

18–1B  
Resource sequence number from CV47

1C–1F  
First word of timestamp for trace entry

**TGREC TOP entry for topology data set create**

**Entry:**

**TOP**

**Event:**

TG created from a checkpointed topology data set

**Modules Generating Entry:**

ISTTRHTG

This trace record is issued when a TG record is created from a checkpointed topology data set.

| 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |   |

| 0002 | Action: C'TOP'

03  
Action
  • C'C' Create TG record

04  
Status byte from the CV47

05  
Flags byte from the CV4680

06  
Flag byte
Bit

Meaning

111. . . .
   TG type from TGREC

B’000’
   Endpoint TG

B’001’
   Intermediate routing TG

B’010’
   Interchange TG

B’011’
   Intersubnet TG

B’100’
   Branch extender TG

...1 ....
   Garbage collection indicator from TGREC

.... 1...
   Adjacent node’s HPRTT value from TGREC

.... .11.
   Local node’s HPR/HPRTT values from TGREC

07
   Number of days left before TGREC is garbage collected

08–0B
   0

0C–OF
   Current flow reduction sequence number (FRSN), from the TOPO data set, for topology database
   updates (TDUs) that include unknown topology control vectors (any topology control vectors other
   than CV46 and CV47)

10–13
   Current FRSN, from the TOPO data set, for TDUs that do not include unknown topology control vectors
   (any topology control vectors other than CV46 and CV47)

14–17
   Last FRSN received from the TOPO data set

18–1B
   Resource sequence number (RSN) from CV47 in the TOPO data set

1C–1F
   First word of timestamp for trace entry

The TRS common topology trace table

In the TRS (topology and routing services component of VTAM) common topology trace table, the deletion
of NDRECs (node records) and TGRECs (TG records) are recorded.

The TRS common topology trace table is located in extended private storage. At TRS initialization, a 40-K
buffer of storage is allocated for the TRS topology trace. When that buffer is filled with trace entries,
another buffer is allocated to continue the trace. Buffer allocation continues until a maximum of 30
buffers is allocated for the TRS topology trace. When all buffers of the TRS topology trace are full of trace
entries, the trace wraps back to the first entry on the first buffer of the trace table.

The TRS topology trace table is pointed to from the TRDAT. The following information about the TRS
topology trace table is included in the TRDAT:
• A pointer to the first buffer of the TRS topology trace
• A pointer to the last buffer of the TRS topology trace
• A pointer to the current buffer of the TRS topology trace
• A pointer to the current TRS topology trace entry
• The number of trace records that have been used in the current buffer of the TRS topology trace
• The number of times the TRS topology trace table has wrapped
• The number of TRS topology trace table buffers allocated

Figure 3 on page 740 shows the TRS topology trace table format:
Figure 3. TRS common topology trace table with two buffers allocated
The first X'18' bytes of the allocated buffer of TRS topology trace table is the header, which contains the following information:

**Byte (hex) Contents**

**00–03**
TRS topology trace control block ID – C"TRTR"

**04–07**
0

**08–0B**
Address of the previous buffer of the TRS topology trace. The previous buffer pointer is 0 for the first buffer of the trace.

**0C–0F**
Address of the next buffer of the TRS topology trace. The next buffer pointer is 0 for the last buffer of the trace.

**10–13**
Timestamp (first word) when the trace most recently wrapped back to the beginning. The timestamp is present only on the first buffer of the TRS topology trace.

**14–17**
Prior timestamp (first word) when the trace wrapped back to the beginning. The timestamp is present only on the first buffer of the TRS topology trace.

---

**TRS common topology trace record descriptions**

This topic contains the TRS topology trace record descriptions. Each TRS topology trace entry is X'40' bytes long. The record ID in the first two characters of the record indicates the following information:

- Byte 0 Reason for the record deletion or TDU entry. See the individual record descriptions for reasons.
- Byte 1 Type of entry
  - C"N": Node
  - C"T": TG

**TRS DELETE entry for NDREC deletion**

**Event:**
Node record deletion

This trace record is issued when a node record is deleted from the topology database.

<table>
<thead>
<tr>
<th>DELETE</th>
<th>V4580 FLAGS</th>
<th>NODE CPNAME</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>001234</td>
<td>0000000000111111111111</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Byte (hex)</td>
<td>Contents</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00–01</td>
<td><strong>Record ID</strong></td>
<td><strong>Calling module</strong></td>
<td><strong>Reason for node deletion</strong></td>
</tr>
<tr>
<td>C&quot;GN&quot;</td>
<td>ISTTRGTM</td>
<td>Garbage collection</td>
<td></td>
</tr>
<tr>
<td>C&quot;MN&quot;</td>
<td>ISTTROMT</td>
<td>MODIFY TOPO,DELETE,SCOPE=LOCAL</td>
<td></td>
</tr>
<tr>
<td>C&quot;SN&quot;</td>
<td>ISTTRGTM</td>
<td>Garbage collection because of MODIFY TOPO,DELETE,SCOPE=NETWORK</td>
<td></td>
</tr>
<tr>
<td>C&quot;TN&quot;</td>
<td>ISTTRGTM</td>
<td>Garbage collection because of receipt of a TDU with the GCI bit on</td>
<td></td>
</tr>
<tr>
<td>02–04</td>
<td>Flag bytes from the CV4580</td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>Flag byte</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Bit Meaning**

- **11. . . . . .**
  - Node type from NDREC
  - B'00' End node
  - B'01' Network node
  - B'10' Virtual node
- **. . . . . 1.**
  - Adjacent node indicator from NDREC
- **. . . . . . 1.**
  - Garbage collection indicator from NDREC
- **. . . . . . . 1.**
  - DLUR end node indicator from NDREC
- **. . . . . . . . 1.**
  - Scope of MODIFY TOPO
  - B'0' Local
  - B'1' Network
06–16
Network qualified CPNAME of the node being deleted

17–27
0

28–2A
Source of the node record creation
- C"TDU" Created because of TDU received from an adjacent node
- C"TGU" Created as the result of a TG update signal
- C"TOP" Created from a check pointed data set at VTAM initialization with INITDB=TOPO

2B
0

2C–2F
Resource sequence number at the time the node record was created

30–33
Resource sequence number at the time the node record was deleted

34–37
First word of timestamp when the node record was created

38–3B
First word of timestamp when the node record was deleted

3C–3F
Address of the node record being deleted

**TRS DELETE entry for TGREC deletion**

**Event:**
TG record deletion

This trace record is issued when a TG record is deleted from the topology database.
### Byte (hex)

#### Contents

<table>
<thead>
<tr>
<th>00–01</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Record ID</strong></td>
</tr>
<tr>
<td>C&quot;BT&quot;</td>
</tr>
<tr>
<td>C&quot;CT&quot;</td>
</tr>
<tr>
<td>C&quot;DT&quot;</td>
</tr>
<tr>
<td>C&quot;FT&quot;</td>
</tr>
<tr>
<td>C&quot;GT&quot;</td>
</tr>
<tr>
<td>C&quot;HT&quot;</td>
</tr>
<tr>
<td>C&quot;MT&quot;</td>
</tr>
<tr>
<td>C&quot;NT&quot;</td>
</tr>
<tr>
<td>C&quot;RT&quot;</td>
</tr>
<tr>
<td>C&quot;ST&quot;</td>
</tr>
<tr>
<td>C&quot;TT&quot;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>02</th>
</tr>
</thead>
<tbody>
<tr>
<td>TG number from the CV4680</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>03</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status byte from the CV47</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>04</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flag byte from the CV4680</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>05</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flag byte</td>
</tr>
</tbody>
</table>

#### Bit Meaning

<table>
<thead>
<tr>
<th>111. . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>TG type from TGREC</td>
</tr>
</tbody>
</table>

**B'000'**
- Endpoint TG

**B'001'**
- Intermediate routing TG

**B'010'**
- Interchange TG

**B'011'**
- Intersubnet TG

**B'100'**
- Branch extender TG

**.1 . . .**
- Garbage collection indicator from TGREC

**. . .1 . .**
- Scope of MODIFY TOPO

**B'0'**
- Local
Network qualified CPNAME of the origin node of the TG being deleted

Network qualified CPNAME of the destination node of the TG being deleted

Source of the TG record creation

- C"TDU" Created because of TDU received from an adjacent node
- C"TGU" Created as the result of a TG update signal
- C"TOP" Created from a checkpointed data set at VTAM initialization with INITDB=TOPO

Resource sequence number at the time the TG record was created

Resource sequence number at the time the TG record was deleted

First word of timestamp when the TG record was created

First word of timestamp when the TG record was deleted

Address of the TG record being deleted
Appendix D. First Failure Support Technology (FFST) probes

This appendix contains the following topics:

- “FFST probe index” on page 747
- “FFST probe information” on page 748
- “FFST probe naming conventions” on page 748
- “FFST probe descriptions” on page 748

See “First Failure Support Technology (FFST) for VTAM” on page 1 for additional information.

FFST probe index

The following table provides an index of FFST probes in alphanumerical order by probe name:

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Component</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTALCxx</td>
<td>APPN Over Logical Link Control</td>
<td>“APPN over logical link control probes” on page 749</td>
</tr>
<tr>
<td>ISTATCxx</td>
<td>APPN TG Management</td>
<td>“APPN TG management” on page 750</td>
</tr>
<tr>
<td>ISTAUCxx</td>
<td>Enterprise Extender</td>
<td>“Enterprise Extender probes” on page 767</td>
</tr>
<tr>
<td>ISTCMCx</td>
<td>Common Management Information Protocol (CMIP) Services</td>
<td>“Common Management Information Protocol (CMIP) services probes” on page 750</td>
</tr>
<tr>
<td>ISTCOCxx</td>
<td>Control Operator</td>
<td>“Control operator probes” on page 752</td>
</tr>
<tr>
<td>ISTCSCxx</td>
<td>Configuration Services</td>
<td>“Configuration services probes” on page 750</td>
</tr>
<tr>
<td>ISTDLCxx</td>
<td>Dependent LU Server</td>
<td>“Dependent LU server probes” on page 759</td>
</tr>
<tr>
<td>ISTDRCxx</td>
<td>Directory Services</td>
<td>“Directory services probes” on page 760</td>
</tr>
<tr>
<td>ISTFSCxx</td>
<td>Coupling Facility Services</td>
<td>“Coupling facility services probes” on page 753</td>
</tr>
<tr>
<td>ISTITUxx</td>
<td>Logical Link Control</td>
<td>“Logical link control” on page 768</td>
</tr>
<tr>
<td>ISTMTCxx</td>
<td>Management Services Transport</td>
<td>“Management services transport probes” on page 769</td>
</tr>
<tr>
<td>ISTNACxx</td>
<td>Network Resource Management</td>
<td>“Network resource management probes” on page 770</td>
</tr>
<tr>
<td>ISTNSCxx</td>
<td>LU Network Services</td>
<td>“LU network services probes” on page 769</td>
</tr>
<tr>
<td>ISTORCxx</td>
<td>Storage Management</td>
<td>“Storage management probes” on page 781</td>
</tr>
<tr>
<td>ISTOPSCxx</td>
<td>Presentation Services</td>
<td>“Presentation services probes” on page 770</td>
</tr>
<tr>
<td>ISTRACxx</td>
<td>Trace Services</td>
<td>“Trace services probes” on page 782</td>
</tr>
<tr>
<td>ISTRMCxx</td>
<td>LU Resource Manager</td>
<td>“LU resource manager probes” on page 769</td>
</tr>
<tr>
<td>ISTRVMxx</td>
<td>Recovery Manager</td>
<td>“Recovery manager (RV) probes” on page 771</td>
</tr>
<tr>
<td>ISTSCCxx</td>
<td>Session Services CP-CP</td>
<td>“Session services CP-CP probes” on page 772</td>
</tr>
</tbody>
</table>
Table 16. FFST probe index (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Component</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTSLCxx</td>
<td>Session Services LU-LU</td>
<td>“Session services LU-LU probes” on page 774</td>
</tr>
<tr>
<td>ISTSSCxx</td>
<td>Session Services</td>
<td>“Session services probes” on page 772</td>
</tr>
<tr>
<td>ISTTACxx</td>
<td>VTAM Topology Agent</td>
<td>“VTAM topology agent probes” on page 784</td>
</tr>
<tr>
<td>ISTTCCxx</td>
<td>TC-DLC (Transmission Control-Data Link Control)</td>
<td>“Transmission control — Data link control (TC-DLC) probes” on page 782</td>
</tr>
<tr>
<td>ISTTRCxx</td>
<td>Topology and Routing Services</td>
<td>“Topology and routing services probes” on page 781</td>
</tr>
<tr>
<td>ISTTSCxx</td>
<td>Transmission Subsystem</td>
<td>“Transmission subsystem probes” on page 783</td>
</tr>
</tbody>
</table>

FFST probe information

When a VTAM FFST probe is triggered, an unexpected condition has occurred in the network. The process that received the condition might not complete normally. The VTAM program will attempt to recover from the unexpected condition and will continue processing subsequent requests. Recovery might not be possible for some system conditions, and subsequent requests might fail, terminals might hang, and other abnormal conditions might occur.

Dump data is collected to assist in identifying the source of the problem. The processing element is freed and processing continues. If the probe triggers multiple times you might need to halt and restart VTAM.

Contact the appropriate IBM Support Center and supply the service representative with the console listing that is written at the time of the error and the dump data produced by the probe.

For information about using FFST dumps, see “First Failure Support Technology (FFST) for VTAM” on page 1.

FFST probe naming conventions

The following table describes the naming convention for the FFST probe name. ISTRAC01 is used as an example for VTAM.

Table 17. FFST probe naming conventions

<table>
<thead>
<tr>
<th>Characters</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,2,3</td>
<td>IST</td>
<td>These characters represent the product identifier. For VTAM probes, these characters are IST.</td>
</tr>
<tr>
<td>4, 5</td>
<td>RA</td>
<td>These characters represent the VTAM component identifier. In this example, RA is the component identifier for the Trace Services component.</td>
</tr>
<tr>
<td>6</td>
<td>C</td>
<td>For VTAM probes, this character is C.</td>
</tr>
<tr>
<td>7, 8</td>
<td>01</td>
<td>These characters represent the probe identification number. This number is not duplicated.</td>
</tr>
</tbody>
</table>

FFST probe descriptions

This information includes a table for each component that contains FFST probe instructions. The components are in alphabetical order, and the probes for each component are in alphanumerical order by probe name. “FFST probe index” on page 747 provides an index of FFST probes in alphanumerical order by
probe name. Each table in this information shows the probe name, the module that issues it, and whether
the probe creates a full or minidump when triggered.

“FFST probe index” on page 747 provides an index of FFST probes in alphanumeric order by probe name.

For information about using FFST dumps, see “First Failure Support Technology (FFST) for VTAM” on
page 1.

APPN over logical link control probes

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTALC01</td>
<td>ISTALCAL</td>
<td>The APPN over logical link control (ALLC) line index control block OSLIN indicates that ALNCB line entries are available, but none can be found. The maximum number of ATM native ALNCB line entries is 65535.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC02</td>
<td>ISTALCPE</td>
<td>An unusable primitive was received by ISTALCPE.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC03</td>
<td>ISTALCPF</td>
<td>An unusable work element was queued to the ALLC port finite state machine (FSM) processor ISTALCPF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC04</td>
<td>ISTALCPF</td>
<td>The work in progress field (ALPOR_WIP) in the ALLC port control block ALPOR has not been set and should have been set during DACTLINK processing.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC05</td>
<td>ISTALCPF</td>
<td>Disable confirmation was received and the filter has never gone inactive.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC06</td>
<td>ISTALCPF</td>
<td>The work element address (WEA) field in the request processing header control block RPH does not point to an ALLC parameter list (ALCPL).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC07</td>
<td>ISTALCLF</td>
<td>An unusable work element was received by ISTALCLF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC08</td>
<td>ISTALCIF</td>
<td>An unusable work element was received by ISTALCIF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC09</td>
<td>ISTALCOF</td>
<td>An unusable work element was received by ISTALCOF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC10</td>
<td>ISTALCCF</td>
<td>An unusable work element was received by ISTALCCF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC11</td>
<td>ISTALCOC</td>
<td>An usable AAL signal was received by ISTALCOC from the AAL FSM handler ISTALCAF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC12</td>
<td>ISTALCIC</td>
<td>An usable AAL signal was received by ISTALCIC from the AAL FSM handler ISTALCAF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC13</td>
<td>ISTALCOL</td>
<td>An usable XID signal was received by ISTALCOL from the XID FSM handler ISTALCXF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTALC14</td>
<td>ISTALCIL</td>
<td>An usable XID signal was received by ISTALCIL from the XID FSM handler ISTALCXF.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
### APPN TG management

**Table 19. FFST probes for APPN TG management**

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTATC01</td>
<td>ISTATCUP</td>
<td>Total connection count corrupted</td>
<td>FULL</td>
</tr>
</tbody>
</table>

### Common Management Information Protocol (CMIP) services probes

**Table 20. FFST probes for CMIP services (ISTCMCxx)**

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTCMC01</td>
<td>ISTCMAFR</td>
<td>CMIP services was dispatched with a work element type that it does not support or does not support in the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCMC02</td>
<td>ISTCMCRT</td>
<td>CMIP services was dispatched with a work element type that it does not support or does not support in the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCMC03</td>
<td>ISTCMDCR</td>
<td>CMIP services was dispatched with a work element type that it does not support or does not support in the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCMC04</td>
<td>ISTCMIOS</td>
<td>CMIP services was dispatched with a work element type that it does not support or does not support in the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCMC05</td>
<td>ISTCMMBR</td>
<td>CMIP services was dispatched with a work element type that it does not support or does not support in the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCMC06</td>
<td>ISTCMRPR</td>
<td>CMIP services was dispatched with a work element type that it does not support or does not support in the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCMC07</td>
<td>ISTCMSSR</td>
<td>CMIP services was dispatched with a work element type that it does not support or does not support in the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCMC08</td>
<td>ACYCMLNC</td>
<td>CMIP services attempted to send an internal CMIP services signal to a destination that is not valid.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCMC09</td>
<td>ACYCMPTC</td>
<td>CMIP services called the PTO_OPEN function and sent parameters that are not valid.</td>
<td>Full</td>
</tr>
</tbody>
</table>

### Configuration services probes

**Table 21. FFST probes for configuration services (ISTCSCxx)**

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTCSC01</td>
<td>ISTACCQ3</td>
<td>This probe detects control vector (or subvector) lengths that are not valid on request contact (REQCONT) RUs. It produces information about the failing request unit processing element (RUPE), request/response unit (RU), and variable work area (VWA). If available, the resource definition table entry (RDTE) for the LINE and PU is also reported.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTCSC02</td>
<td>ISTCSCRC</td>
<td>Resource registration failure was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTCSC03</td>
<td>ISTCSCRE</td>
<td>Resource registration failure was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTCSC04</td>
<td>ISTCSCRF</td>
<td>Resource registration failure was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTCSC05</td>
<td>ISTCSCRJ</td>
<td>Resource registration failure was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTCSC06</td>
<td>ISTCSCRK</td>
<td>Resource registration failure was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTCSC07</td>
<td>ISTCSCRH</td>
<td>Resource registration failure was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTCSC09</td>
<td>ISTDECQE</td>
<td>The resource definition table entry (RDTE) for the dependent LU requester CDRSC was not found.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCSC10</td>
<td>ISTDECQD</td>
<td>The resource definition table entry (RDTE) for the dependent LU requester CDRSC was not found.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCSC11</td>
<td>ISTDECP3</td>
<td>The resource definition table entry (RDTE) for the dependent LU requester CDRSC was not found.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCSC12</td>
<td>ISTDECP4</td>
<td>The resource definition table entry (RDTE) for the dependent LU requester CDRSC was not found.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCSC13</td>
<td>ISTDECQ4</td>
<td>A protocol violation was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCSC14</td>
<td>ISTDECQ4</td>
<td>A topology database update (TDU) error was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCSC15</td>
<td>ISTACCQE</td>
<td>An attempt to add the resource definition table entry (RDTE) for the DLUS-supported LU failed because the network address is already in use or was not properly freed by a previous user. Related pointer values are provided in the secondary symptom string.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCSC16</td>
<td>ISTDECLL</td>
<td>The GETRDTE for the higher-level resource failed.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTCSC18</td>
<td>ISTACCQ3</td>
<td>REQCONT failure RU is not valid.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
## Control operator probes

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
</table>
| ISTCOC01   | ISTCOCDL  | A suspended log mode is not found. Check the mode the application program specifies on the APPCCMD CONTROL=OPRCNTRL, QUALIFY=RESTORE macroinstruction. You can use the following VTAM dump functions to display advanced program-to-program communication (APPC)-related information:  
• APPLCONV  
• APPLMODE  
• APPMODAL  
• PARTNRLU  
For information about these functions, see the information about using VTAM dump analysis tools in z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures. | Full      |
| ISTCOC02   | ISTCOCSC  | A suspended log mode is not found. Check the mode the application program specifies on the APPCCMD CONTROL=OPRCNTRL, QUALIFY=RESTORE macroinstruction. You can use the following VTAM dump functions to display advanced program-to-program communication (APPC)-related information:  
• APPLCONV  
• APPLMODE  
• APPMODAL  
• PARTNRLU  
For information about these functions, see the information about using VTAM dump analysis tools in z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures. | Full      |
Table 22. FFST probes for control operator (ISTCOCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTCOC03</td>
<td>ISTCOCVR</td>
<td>A suspended log mode is not found. Check the mode the application program specifies on the APPCCMD CONTROL=OPRCNTRL, QUALIFY=RESTORE macroinstruction. You can use the following VTAM dump functions to display advanced program-to-program communication (APPC)-related information: • APPLCONV • APPLMODE • APPMODAL • PARTNRLU For information about these functions, see the information about using VTAM dump analysis tools in z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures.</td>
<td>Full</td>
</tr>
</tbody>
</table>

**Coupling facility services probes**

Table 23. FFST probes for coupling facility services (ISTFSCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTFSC00</td>
<td>ISTFSUMT</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for IXLLIST when VTAM attempted to move an entry to the end of a list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC01</td>
<td>ISTFSNRT</td>
<td>A work element that was not recognized by coupling facility services (CFS) was queued to the CFS connection services PAB.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC02</td>
<td>ISTFSNRT</td>
<td>A work element was recognized by coupling facility services (CFS), but it contained incorrect information.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC03</td>
<td>ISTFSNCF</td>
<td>An unexpected input was received for the current state of the connection finite state machine.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC04</td>
<td>ISTFSNST</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for the IXLDISC macro.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC05</td>
<td>ISTFSURB</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for the IXLREBLD macro.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC06</td>
<td>ISTFSURS</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for the IXLEERSP macro.</td>
<td>Mini</td>
</tr>
</tbody>
</table>

First Failure Support Technology (FFST) probes 753
Table 23. FFST probes for coupling facility services (ISTFSCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTFSC07</td>
<td>ISTFSUUS</td>
<td>An unexpected return code <code>xxxx</code> and reason code <code>yyyy</code> were returned for the IXLUSYNC macro.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC08</td>
<td>ISTFSUFC</td>
<td>An unexpected return code <code>xxxx</code> and reason code <code>yyyy</code> were returned for the IXLFORCE macro.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC09</td>
<td>ISTFSNCN</td>
<td>An unexpected return code <code>xxxx</code> and reason code <code>yyyy</code> were returned for the IXLCONN macro.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC0A</td>
<td>ISTFSUDC</td>
<td>VTAM was connected to an unexpected structure type.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC0B</td>
<td>ISTFSUDC</td>
<td>The connection name returned in the IXLCONN answer area was not the name specified on IXLCONN.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC0C</td>
<td>ISTFSGLR</td>
<td>VTAM was not able to find a local generic mapping while rebuilding the coupling facility structure with local data.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC0D</td>
<td>ISTFSDRT</td>
<td>A work element that was not recognized was received by a Sysplex Wide Security Associations structure object PAB.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC0E</td>
<td>ISTFSGG1</td>
<td>VTAM made a counting error associated with a generic resource.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC0F</td>
<td>ISTFSUUS</td>
<td>Unexpected input was received for current state of the user sync point finite state machine (FSM).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC10</td>
<td>ISTFSCA3</td>
<td>VTAM was unable to invoke the IWMGRREG macroinstruction for registering a generic name to the work load manager.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC11</td>
<td>ISTFSUDC</td>
<td>VTAM has disconnected from the generic resources coupling facility structure because a timed process did not complete within the given time.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTFSC12</td>
<td>ISTFSUXJ</td>
<td>An unexpected return code <code>xxxx</code> and reason code <code>yyyy</code> were returned from IXCJOIN when VTAM attempted to join an XCF group.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC13</td>
<td>ISTFSUAL</td>
<td>An unexpected return code <code>xxxx</code> and reason code <code>yyyy</code> were returned from IXLALTER when VTAM attempted to alter coupling facility structure attributes.</td>
<td>Mini</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTFSC14</td>
<td>ISTFSLML</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned from IXLLIST when VTAM attempted to establish list monitoring.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC15</td>
<td>ISTFSNPG</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned from IXLPURGE when VTAM attempted to purge outstanding structure access request.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC16</td>
<td>ISTFSLRM</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned from IXLLIST when VTAM attempted to read multiple entries from a list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC17</td>
<td>ISTFSLDM</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned from IXLLIST when VTAM attempted to delete multiple entries.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC18</td>
<td>ISTFSLWC</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned from IXLLIST when VTAM attempted to update the list control area.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC19</td>
<td>ISTFSLVC</td>
<td>An unexpected return code <em>xxxx</em> was returned from IXLVECTR when VTAM attempted to modify or test the list notification vector.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC1A</td>
<td>ISTFSURL</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned for IXLLIST when VTAM attempted to read a list entry.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC1B</td>
<td>ISTFSUUL</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned for IXLLIST when VTAM attempted to update a list entry.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC1C</td>
<td>ISTFSUCL</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned for IXLLIST when VTAM attempted to create a list entry.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC1D</td>
<td>ISTFSUHT</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned for IXLLIST when VTAM attempted to read and move a list entry.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC1E</td>
<td>ISTFSKRT</td>
<td>An unexpected return code <em>xxxx</em> and reason code <em>yyyy</em> were returned for IXLLIST when VTAM attempted to set a lock entry.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 23. FFST probes for coupling facility services (ISTFSCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTFSC1F</td>
<td>ISTFSKRT</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for IXLLIST when VTAM attempted to release a lock entry.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC20</td>
<td>ISTFSKRT</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for IXLLIST when VTAM attempted to release a lock entry for a failed connector.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC21</td>
<td>ISTFSCFG</td>
<td>VTAM found a name defined as both a generic resource name and a USERVAR.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC22</td>
<td>ISTFSDRT</td>
<td>A work element was recognized by a Sysplex Wide Security Associations structure object PAB, but it contained incorrect data.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC23</td>
<td>ISTFSUDL</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for IXLLIST when VTAM attempted to delete a list entry.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC24</td>
<td>ISTFSUEN</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for IXLLIST when VTAM attempted to read list control information.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC25</td>
<td>ISTFSURH</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for IXLLIST when VTAM attempted to read a list entry from the head of a list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC26</td>
<td>ISTFSUWT</td>
<td>An unexpected return code xxxx and reason code yyyy were returned for IXLLIST when VTAM attempted to write a list entry to the end of a list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC27</td>
<td>ISTFSLDE</td>
<td>An unexpected return code xxxx and reason code yyyy were returned from IXLLIST when VTAM attempted to delete a list of entries.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC28</td>
<td>ISTFSUQS</td>
<td>An unexpected return code and reason code were returned from IXCQUERY when VTAM attempted to obtain sysplex-related information.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC29</td>
<td>ISTFSPDN</td>
<td>A pointer to a buffer containing NLP entry IDs for an IXLLIST DELETE_ENTRYLIST request that was being retried was zero. NLP entries may not be deleted from a multinode persistent session coupling facility structure.</td>
<td>Full</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTFSC2A</td>
<td></td>
<td>A coupling facility user deregistered but did not provide an area to pass back a work element when one existed. The storage for the work element is not freed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC2B</td>
<td></td>
<td>A coupling facility user attempted to unregister, but provided a user token that was not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC2C</td>
<td></td>
<td>A coupling facility user indicated that a process was complete, but provided a user token that was not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC2D</td>
<td>ISTFSGRO</td>
<td>When attempting to alter the entry-to-element ratio for the generic resource structure, the new ratio was not valid. Either the entry portion was zero, the element portion was zero, or the entry portion divided by the element portion was greater than the maximum number of data elements.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC2E</td>
<td>ISTFSLRO</td>
<td>When attempting to alter the entry-to-element ratio for a VTAM list structure, the new ratio was not valid. Either the entry portion was zero, the element portion was zero, or the entry portion divided by the element portion was greater than the maximum number of data elements.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC30</td>
<td>ISTFSPDN</td>
<td>An unexpected return code of X'0008' and reason code of X'0082B' were received for IXLLIST while attempting to delete NLP entries in the multinode persistent coupling facility structure. A start or stop index was given that was not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC31</td>
<td>ISTFSPDN</td>
<td>An unexpected return code xxxx and reason code yyyy were received for IXLLIST while attempting to delete NLP entries in the multinode persistent coupling facility structure. NLP entries may not be deleted.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC32</td>
<td>ISTFSPRT</td>
<td>A work element that was not recognized was received by a multinode persistent session structure object PAB.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC33</td>
<td>ISTFSPRT</td>
<td>A work element was recognized by a multinode persistent session structure object PAB, but it contained incorrect data.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 23. FFST probes for coupling facility services (ISTFSCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTFSC34</td>
<td>ISTFSPRV</td>
<td>An unexpected return code and reason code were returned from IXLLIST while attempting to read in a multinode persistent session (MNPS) application program's data during an MNPS recovery. Sessions will not be recovered.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC35</td>
<td>ISTFSLRL</td>
<td>An unexpected return code and reason code were returned from IXLLIST when attempting to read multiple entries from a list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC36</td>
<td>ISTFSPDD</td>
<td>An unexpected return code and reason code were returned from IXLLIST when attempting to delete data associated with a multinode persistent session application program. The state of the application program will remain in CLEANUP and may be cleaned up by another VTAM.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC37</td>
<td>ISTFSDDD</td>
<td>An unexpected return code and reason code were returned from IXLLIST when attempting to delete data associated with a TCP/IP stack using Sysplex Wide Security Associations.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC38</td>
<td>ISTFSDRD</td>
<td>An unexpected return code and reason code were returned from IXLLIST when attempting to read data associated with a TCP/IP stack using Sysplex Wide Security Associations.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC39</td>
<td>ISTFSVRT</td>
<td>A work element that was not recognized was received by a Sysplexports structure object PAB.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC3A</td>
<td>ISTFSVRT</td>
<td>A work element was recognized by a Sysplexports structure object PAB, but it contained incorrect data.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC3C</td>
<td>ISTFSLLO</td>
<td>An unexpected return code and reason code were returned from IXLLIST when attempting to lock data associated with a structure object.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC3D</td>
<td>ISTFSVDD</td>
<td>An unexpected return code and reason code were returned from IXLLIST when attempting to delete data associated with a TCP/IP stack using Sysplexports.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTFSC3E</td>
<td>ISTFSCFG</td>
<td>Data that was not valid was returned by IXLLIST when attempting to read the generic resource mapping list of the generic resource structure.</td>
<td>Full</td>
</tr>
</tbody>
</table>
## Dependent LU server probes

### Table 24. FFST probes for dependent LU server (ISTDLCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTDLC01</td>
<td>ISTDLCRD</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC02</td>
<td>ISTDLCRD</td>
<td>Element representing the PU could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC03</td>
<td>ISTDLCDE</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC04</td>
<td>ISTDLCDE</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC07</td>
<td>ISTDLCAL</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC08</td>
<td>ISTDLCAL</td>
<td>Element representing the PU could not be found.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDLC09</td>
<td>ISTDLCAS</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC10</td>
<td>ISTDLCRT</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC11</td>
<td>ISTDLCRT</td>
<td>Unrecognized signal; received signal is not supported by dependent LU server.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC13</td>
<td>ISTDLCSI</td>
<td>During processing, if a CP_SVR_SESS_STAT(inactive) signal is sent by the LRM component when CP-SVR pipe is deactivated, the combination of the contention winner and contention loser finite state machine states was found to be not valid. As a result, the final cleanup associated with the pipe deactivation could not be performed.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDLC20</td>
<td>ISTDLCRP</td>
<td>Element representing the PU could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC21</td>
<td>ISTDLCAP</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC24</td>
<td>ISTDLCTD</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC26</td>
<td>ISTDLCDE</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC27</td>
<td>ISTDLCDE</td>
<td>Element representing the PU could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC28</td>
<td>ISTDLCDE</td>
<td>Element representing DLUS-supported LU could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC29</td>
<td>ISTDLCSA</td>
<td>DLR FSM not found</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC30</td>
<td>ISTDLCQA</td>
<td>DLR PU FSM not expecting ACTPU REQ</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDLC31</td>
<td>ISTDLCQA</td>
<td>DLR element not freed after inactive</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDLC33</td>
<td>ISTDLCED</td>
<td>Element representing the dependent LU requester could not be found.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
## Directory services probes

Table 25. FFST probes for directory services (ISTDRCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTDRC00</td>
<td>ISTDRCRT</td>
<td>An unrecognized request element was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC01</td>
<td>ISTDRCRT</td>
<td>An unrecognized interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC02</td>
<td>ISTDROBO</td>
<td>An unrecognized function code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC04</td>
<td>ISTDROFO</td>
<td>A build of a found CV80 is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC05</td>
<td>ISTDROFO</td>
<td>A build of a found CV3C is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC06</td>
<td>ISTDROFO</td>
<td>A build of a found CV3C is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC07</td>
<td>ISTDROFO</td>
<td>A build of a found CV3D is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC08</td>
<td>ISTDROFO</td>
<td>A build of a found CV3E is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC09</td>
<td>ISTDROFO</td>
<td>A build of a found CV40 is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC0A</td>
<td>ISTDROLM</td>
<td>A build of a locate GDS variable is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC0B</td>
<td>ISTDROLM</td>
<td>A build of a locate GDS variable is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC0C</td>
<td>ISTDROLM</td>
<td>A build of a found GDS variable is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC0D</td>
<td>ISTDROLM</td>
<td>A build of a register GDS variable is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC0E</td>
<td>ISTDROLM</td>
<td>A build of a CDINIT GDS variable is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC0F</td>
<td>ISTDROLM</td>
<td>A build of an IOCD GDS variable is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC11</td>
<td>ISTDRO0E</td>
<td>A build of a CV0E failed because the requested name was not recognized.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC12</td>
<td>ISTDRO60</td>
<td>A build of a CV60 failed because the requested name was not recognized.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC13</td>
<td>ISTDROCG</td>
<td>The search type that was requested was not recognized.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC14</td>
<td>ISTDROLM</td>
<td>A build of a notify GDS variable is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC15</td>
<td>ISTRDUE</td>
<td>An unknown entry type was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC16</td>
<td>ISTDRDDD</td>
<td>Either an unknown resource or no resource was found for a resource delete request.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC17</td>
<td>ISTDRDDD</td>
<td>The associated network ID information was not found for a resource delete request.</td>
<td>Full</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTDRC18</td>
<td>ISTDRDDDD</td>
<td>A release of storage associated with a resource failed to complete.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC19</td>
<td>ISTDRDDDD</td>
<td>A release of storage associated with a resource failed to complete.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC1A</td>
<td>ISTDRDDDD</td>
<td>A release of storage associated with a resource failed to complete.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC1B</td>
<td>ISTDRDUD</td>
<td>An update directory request was received, but no list elements were found.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC1C</td>
<td>ISTDRDCD</td>
<td>A release of storage associated with a resource failed to complete.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC1D</td>
<td>ISTDRDCD</td>
<td>A release of storage associated with a CACHE_DATA interprocess signal failed to complete.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC1E</td>
<td>ISTDRDCD</td>
<td>A CACHE_DATA interprocess signal failed because the requested name was not recognized.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC1F</td>
<td>ISTDRAPC</td>
<td>A CP_STATUS interprocess signal failed because the requested name was not recognized.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC20</td>
<td>ISTDRAUA</td>
<td>Cannot remove an adjacent control point from the list.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC21</td>
<td>ISTDRAUA</td>
<td>Cannot add an adjacent control point to the list.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC22</td>
<td>ISTDRAUA</td>
<td>ISTDRAUA received an action code indicating that an unexpected condition has occurred.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC23</td>
<td>ISTDRAUA</td>
<td>ISTDRAUA was unable to process an unrecognized action code.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC24</td>
<td>ISTDRED5</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC25</td>
<td>ISTDRAFC</td>
<td>An unrecognized interprocess signal was received for a node role.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC26</td>
<td>ISTDRAFL</td>
<td>The sender of a request cannot be identified.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC27</td>
<td>ISTDRAFL</td>
<td>An unknown verify indicator value was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC28</td>
<td>ISTDRAFL</td>
<td>An unrecognized CV35 vector was received from a VTAM component.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC29</td>
<td>ISTDRAFL</td>
<td>VTAM detected a software error while processing a CV35 vector.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC2A</td>
<td>ISTDRAFL</td>
<td>A protocol error was detected while processing an interprocess signal from a VTAM component.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC2B</td>
<td>ISTDRAFL</td>
<td>An unrecognized name was received from a VTAM component.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC2C</td>
<td>ISTDRAFL</td>
<td>A CV35 vector was received while the keep indicator was set.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC2D</td>
<td>ISTDRAFL</td>
<td>An unrecognized network-qualified name was received.</td>
<td>Full</td>
</tr>
</tbody>
</table>
Table 25. FFST probes for directory services (ISTDRCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTDRC2E</td>
<td>ISTDREFPY</td>
<td>An unrecognized network-qualified name was</td>
<td>Full</td>
</tr>
<tr>
<td></td>
<td></td>
<td>received.</td>
<td></td>
</tr>
<tr>
<td>ISTDRC2F</td>
<td>ISTDROLO</td>
<td>A build of a Locate CV80 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC30</td>
<td>ISTDROLO</td>
<td>A build of a Locate CV60 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC31</td>
<td>ISTDROLO</td>
<td>A build of a Locate CV81 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC32</td>
<td>ISTDROLO</td>
<td>A build of a Locate CV2B is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC33</td>
<td>ISTDROLO</td>
<td>A build of a Locate CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC34</td>
<td>ISTDROLO</td>
<td>A build of a Locate CV35 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC35</td>
<td>ISTDROI</td>
<td>A build of a Find CV80 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC36</td>
<td>ISTDROI</td>
<td>A build of a Find CV3C is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC37</td>
<td>ISTDROI</td>
<td>A build of a Find CV3C is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC38</td>
<td>ISTDROI</td>
<td>A build of a Find CV3D is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC39</td>
<td>ISTDROI</td>
<td>A build of a Find CV3E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC3A</td>
<td>ISTDROI</td>
<td>A build of a Find CV81 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC3B</td>
<td>ISTDROI</td>
<td>A build of a Find CV81 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC3C</td>
<td>ISTDROI</td>
<td>A build of a Find CV82 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC3D</td>
<td>ISTDRMCH</td>
<td>An unrecognized input was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC3E</td>
<td>ISTDRMCS</td>
<td>An unrecognized interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC3F</td>
<td>ISTDRMCS</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC40</td>
<td>ISTDRMDO</td>
<td>An unrecognized action was requested.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC41</td>
<td>ISTDRMDO</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC42</td>
<td>ISTDRMDO</td>
<td>Unexpected elements were found in a locate search.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC43</td>
<td>ISTDRMDO</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC44</td>
<td>ISTDRMDO</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC45</td>
<td>ISTDRMDO</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>------------</td>
<td>------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTDRC46</td>
<td>ISTDRCMD</td>
<td>A list requiring at least one element was empty.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC47</td>
<td>ISTDRCMFD</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC48</td>
<td>ISTDRCMFD</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC49</td>
<td>ISTDRCMDV</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC4A</td>
<td>ISTDRCMDV</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC4B</td>
<td>ISTDRCMDV</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC4C</td>
<td>ISTDRCMDA</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC4D</td>
<td>ISTDRCMDA</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC4E</td>
<td>ISTDRCMTD</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC4F</td>
<td>ISTDRCMTD</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC50</td>
<td>ISTDRCRSD</td>
<td>Unrecognized inputs for state were detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC51</td>
<td>ISTDRCRSSR</td>
<td>Unrecognized inputs for state were detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC52</td>
<td>ISTDRCRTT</td>
<td>A work element was received when none was expected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC53</td>
<td>ISTDRCRSB</td>
<td>ISTDRCRSB was unable to add a locate control block to the list.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC54</td>
<td>ISTDRCNB</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC55</td>
<td>ISTDRCMCH</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC56</td>
<td>ISTDRCSDB</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC57</td>
<td>ISTDRCSDB</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC58</td>
<td>ISTDRCSDS</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC59</td>
<td>ISTDRCSDS</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC5A</td>
<td>ISTDRCSDS</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC5B</td>
<td>ISTDRCSDS</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC5C</td>
<td>ISTDRCSDS</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC5D</td>
<td>ISTDRCSDS</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC5E</td>
<td>ISTDRCSDS</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC5F</td>
<td>ISTDRCSPF</td>
<td>An unrecognized combination of node roles was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC60</td>
<td>ISTDRCSSL</td>
<td>An unrecognized state was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC61</td>
<td>ISTDRCSSL</td>
<td>An unrecognized status was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC62</td>
<td>ISTDRCSSL</td>
<td>An unrecognized search status for a called task was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC63</td>
<td>ISTDRCMDM</td>
<td>An unrecognized function parameter was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC64</td>
<td>ISTDRCMDV</td>
<td>An unrecognized function was specified.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC65</td>
<td>ISTDRCMDV</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTDRC66</td>
<td>ISTDROH</td>
<td>An unrecognized function code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC67</td>
<td>ISTDROH</td>
<td>An unrecognized combination of inputs was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC68</td>
<td>ISTDROH</td>
<td>An unrecognized combination of inputs was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC69</td>
<td>ISTDROH</td>
<td>An unexpected interprocess signal was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC6A</td>
<td>ISTDROH</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC6B</td>
<td>ISTDROH</td>
<td>An unrecognized function was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC6C</td>
<td>ISTDROH</td>
<td>An unrecognized search task was requested.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC6D</td>
<td>ISTDROH</td>
<td>VTAM was unable to remove a locate control block from the list.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC6E</td>
<td>ISTDROH</td>
<td>An unrecognized query function was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC6F</td>
<td>ISTDROH</td>
<td>An unrecognized entry type was specified.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC70</td>
<td>ISTDROH</td>
<td>A release of storage associated with a resource failed to complete.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC71</td>
<td>ISTDROH</td>
<td>A build of a Locate CV82 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC72</td>
<td>ISTDROH</td>
<td>A build of a Find CV40 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC73</td>
<td>ISTDROH</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC74</td>
<td>ISTDROH</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC75</td>
<td>ISTDROH</td>
<td>An unrecognized state was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC76</td>
<td>ISTDROH</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC77</td>
<td>ISTDROH</td>
<td>An unrecognized state was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC78</td>
<td>ISTDROH</td>
<td>A register GDS variable is required, but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC79</td>
<td>ISTDROH</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC7A</td>
<td>ISTDROH</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC7B</td>
<td>ISTDROH</td>
<td>An unrecognized state was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC7C</td>
<td>ISTDROH</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC7D</td>
<td>ISTDROH</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC7E</td>
<td>ISTDROH</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC7F</td>
<td>ISTDROH</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTDRC80</td>
<td>ISTDRRNF</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC81</td>
<td>ISTDRRNF</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC82</td>
<td>ISTDRRNF</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC83</td>
<td>ISTDRRNF</td>
<td>The input received is not recognized for the current state.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC84</td>
<td>ISTDRRNP</td>
<td>A locate control block is required but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC85</td>
<td>ISTDRRNP</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC86</td>
<td>ISTDRRNR</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC87</td>
<td>ISTDRRNR</td>
<td>The available length is not sufficient for a required GDS variable.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC88</td>
<td>ISTDRRNR</td>
<td>The available length is not sufficient for a required GDS variable.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC89</td>
<td>ISTDRRNR</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC8A</td>
<td>ISTDRRNR</td>
<td>The available length is not sufficient for a required GDS variable.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC8B</td>
<td>ISTDRSBC</td>
<td>A parsed locate element is required, but is not present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC8C</td>
<td>ISTDRSNB</td>
<td>A state that is not valid was detected upon entry to ISTDRSNB.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC8D</td>
<td>ISTDRSSD</td>
<td>An unknown task was loaded.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC8E</td>
<td>ISTDRFDS</td>
<td>A directory server notify was received with an empty list.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC8F</td>
<td>ISTDRFDS</td>
<td>An unexpected return code was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC90</td>
<td>ISTDRESHR</td>
<td>An unexpected input operation was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC91</td>
<td>ISTDRESHR</td>
<td>An unrecognized max hierarchy was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC92</td>
<td>ISTDRESHR</td>
<td>An unexpected resource CV combination was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC93</td>
<td>ISTDRESHR</td>
<td>An unexpected max hierarchy combination was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC94</td>
<td>ISTDRSNI</td>
<td>An unrecognized locate was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC95</td>
<td>ISTDRSNI</td>
<td>An unrecognized node role was identified in an LCB.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC96</td>
<td>ISTDREDUP</td>
<td>A DBUPDATE was attempted at a VTAM end node.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC97</td>
<td>ISTDREDQP</td>
<td>A DBQUERY was attempted at a VTAM end node.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC98</td>
<td>ISTDREDDP</td>
<td>A DBDELETE was attempted at a VTAM end node.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC99</td>
<td>ISTDRERT</td>
<td>The DS PAB router received an unrecognized IPS.</td>
<td>Full</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>--------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTDRC9A</td>
<td>ISTDROCG</td>
<td>Unrecognized parameter value was received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9B</td>
<td>ISTDRRNF</td>
<td>Input not valid for current finite state machine (FSM) state</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9C</td>
<td>ISTDRRNF</td>
<td>Input not valid for current finite state machine (FSM) state</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9D</td>
<td>ISTDRRNF</td>
<td>Input not valid for current finite state machine (FSM) state</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9E</td>
<td>ISTDROFI</td>
<td>A build of a FIND CV26 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9F</td>
<td>ISTDROFO</td>
<td>A build of a FOUND CV26 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCAC</td>
<td>ISTDREDC</td>
<td>DSME exit returned a central directory server (CDS) list that was not valid.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB0</td>
<td>ISTDROIE</td>
<td>A build of a CV0E failed because of an unrecognized name.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB3</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB4</td>
<td>ISTDROIE</td>
<td>A build of a FIND CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB5</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB6</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB7</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB8</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9B</td>
<td>ISTDRRNF</td>
<td>Input not valid for current finite state machine (FSM) state</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9C</td>
<td>ISTDRRNF</td>
<td>Input not valid for current finite state machine (FSM) state</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9D</td>
<td>ISTDRRNF</td>
<td>Input not valid for current finite state machine (FSM) state</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9E</td>
<td>ISTDROFI</td>
<td>A build of a FIND CV26 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC9F</td>
<td>ISTDROFO</td>
<td>A build of a FOUND CV26 is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCAC</td>
<td>ISTDREDC</td>
<td>DSME exit returned a central directory server (CDS) list that was not valid.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB0</td>
<td>ISTDROIE</td>
<td>A build of a CV0E failed because of an unrecognized name.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB3</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB4</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB5</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB6</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB7</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCB8</td>
<td>ISTDROIE</td>
<td>A build of a Find CV0E is required, but none is present.</td>
<td>Full</td>
</tr>
</tbody>
</table>

Table 25. FFST probes for directory services (ISTDRCxx) (continued)
Table 25. FFST probes for directory services (ISTDRCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTDRCB9</td>
<td>ISTDRCB9</td>
<td>List operation failed.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCBA</td>
<td>ISTDRCB9</td>
<td>List operation failed.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCBB</td>
<td>ISTDRCB9</td>
<td>List operation failed.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCBC</td>
<td>ISTDRCB9</td>
<td>List operation failed.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCBD</td>
<td>ISTDRCB9</td>
<td>Attempt to free storage failed.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRCBE</td>
<td>ISTDRCB9</td>
<td>An unrecognized state was detected.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC1</td>
<td>ISTDRCB9</td>
<td>An unrecognized state was detected on entry.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC2</td>
<td>ISTDRCB9</td>
<td>Build of a find CV4A required, but none were present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC3</td>
<td>ISTDRCB9</td>
<td>Build of a found CV4A required, but none were present.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC4</td>
<td>ISTDRCB9</td>
<td>RDS search request is hung.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTDRC5</td>
<td>ISTDRCB9</td>
<td>Attempted to add an OSCB to a list when it is already on a list.</td>
<td>Full</td>
</tr>
</tbody>
</table>

Half-session services probes

Table 26. FFST probes for half-session services (ISTHSCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTDHSC01</td>
<td>ISTDHSC01</td>
<td>A problem occurred trying to free a CSM buffer.</td>
<td>Full</td>
</tr>
</tbody>
</table>

Enterprise Extender probes

Table 27. FFST probes for Enterprise Extender (ISTAUCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTDUC01</td>
<td>ISTDUC01</td>
<td>Unusable work element was queued to Enterprise Extender port finite state machine (FSM) processor ISTDUC01.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDUC02</td>
<td>ISTDUC01</td>
<td>Work-in-progress field (IPNCB_WIP) in the Enterprise Extender port control block IPNCB has not been set and should have been set during DACTLINK processing.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDUC03</td>
<td>ISTDUC01</td>
<td>Work element address (WEA) field in the request parameter header control block (RPH) does not point to an AUDP parameter list (AUCPL).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDUC04</td>
<td>ISTDUC01</td>
<td>Unusable work element was queued to the Enterprise Extender line finite state machine (FSM) processor ISTDUC01.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTDUC05</td>
<td>ISTDUC01</td>
<td>Unusable work element was queued to the Enterprise Extender call-in finite state machine (FSM) processor ISTDUC01.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
### Logical link control

#### Table 27. FFST probes for Enterprise Extender (ISTAUCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTAUC06</td>
<td>ISTAUOF</td>
<td>Unusable work element was queued to the Enterprise Extender call-out finite state machine (FSM) processor ISTAUOF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTAUC07</td>
<td>ISTAUCCF</td>
<td>Unusable work element was queued to the Enterprise Extender connection processor ISTAUCCF.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTAUC08</td>
<td>ISTAUCOL</td>
<td>An XID signal that is not valid was received by the outbound signal processor ISTAUCOL.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTAUC09</td>
<td>ISTAUCIL</td>
<td>An XID signal that is not valid was received by the inbound signal processor ISTAUCIL.</td>
<td>Mini</td>
</tr>
</tbody>
</table>

#### Table 28. FFST probes for logical link control

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTIUT02</td>
<td>IUTLLCII</td>
<td>Primitive is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT03</td>
<td>IUTLLCIO</td>
<td>Primitive is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT04</td>
<td>IUTLLCRD</td>
<td>Unrecognized primitive.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT05</td>
<td>IUTLLCRD</td>
<td>Input is not a response.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT06</td>
<td>IUTALCII</td>
<td>Primitive is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT14</td>
<td>IUTLLC9D</td>
<td>Unrecognized primitive.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT15</td>
<td>IUTLLC9D</td>
<td>Input is not a valid request.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT38</td>
<td>CMLOC$AC</td>
<td>Act_SAP_Cnf - state error</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT42</td>
<td>CMLOC$EC</td>
<td>Enable_IC_Cnf - state error</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT44</td>
<td>CMLOC$EI</td>
<td>Enable_IC_Ind - state error</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT49</td>
<td>CMLOC$SC</td>
<td>Call_Setup_Cnf - state error</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT50</td>
<td>CMLOC$TC</td>
<td>Deactivate_SAP_Cnf - state error</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT51</td>
<td>CMLOC$SI</td>
<td>Call_Setup_Ind - state error</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT65</td>
<td>REGMGRSRK</td>
<td>Resolve_Token - object address is zero</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT66</td>
<td>REGMGRUK</td>
<td>Reuse_Token - token is not valid</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT67</td>
<td>REGMGRIK</td>
<td>Invalidate_Token - object address is zero</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT72</td>
<td>IUTLLCRM</td>
<td>Registration manager initialization FSM error</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT73</td>
<td>ISTLLCM8</td>
<td>Unusable input has been received by IDX GROUP CONTROL (LLCM8).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT74</td>
<td>ISTLLCM8</td>
<td>IDX GROUP CONTROL (LLCM8) was unable to obtain an RU processing element (RUPE) for a critical identification exchange (IDX) function.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTIUT75</td>
<td>ISTLLCM8</td>
<td>IDX GROUP CONTROL (LLCM8) received an unexpected RU processing element.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 28. FFST probes for logical link control (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTIUT76</td>
<td>IUTLLCTP</td>
<td>Lost PDUS - The READ side of the HPDT connection detected a lost packet.</td>
<td>Full</td>
</tr>
</tbody>
</table>

**LU network services probes**

Table 29. FFST probes for LU network services (ISTNSCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTNSC01</td>
<td>ISTNSCRS</td>
<td>A suspended log mode is not found.</td>
<td>Full</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check the mode the application program specifies on the APPCCMD CONTROL=OPRCNTRL, QUALIFY=RESTORE macroinstruction.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can use the following VTAM dump functions to display advanced program-to-program communication (APPC)-related information:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• APLCONV</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• APPLMODE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• APPMODAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PARTNRLU</td>
<td></td>
</tr>
</tbody>
</table>

**LU resource manager probes**

Table 30. FFST probes for LU resource manager (ISTRMCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTRMC01</td>
<td>ISTRMCLR</td>
<td>A QUERY_SNASVCMG response was received, but none was outstanding.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRMC02</td>
<td>ISTRMCLR</td>
<td>The mode name for the specified LU was not found in the LU 6.2 logmode table.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRMC03</td>
<td>ISTRMCTP</td>
<td>The mode name for the specified LU was not found in the LU 6.2 logmode table.</td>
<td>Full</td>
</tr>
</tbody>
</table>

**Management services transport probes**

Table 31. FFST probes for management services transport (ISTMTCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTMTC01</td>
<td>ISTMTCFF</td>
<td>The main router received an unrecognized request-unit processing element.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC02</td>
<td>ISTMTCFF</td>
<td>The main router received an unrecognized work element.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTMTC03</td>
<td>ISTMTCFF</td>
<td>The START_TP_REPLY signal that is received contains a return code that is incorrect or unrecognized for this leg.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 31. FFST probes for management services transport (ISTMTCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTMTC04</td>
<td>ISTMTCFF</td>
<td>An unexpected condition is detected when a query request is sent to the NetView application program's programmable peripheral interface (PPI) queue.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC05</td>
<td>ISTMTCFF</td>
<td>An unexpected condition is detected while defining the VTAM queue to the NetView application program's programmable peripheral interface (PPI).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC06</td>
<td>ISTMTCFF</td>
<td>An unexpected condition is detected while sending data from VTAM to the NetView processor across the programmable peripheral interface (PPI).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC07</td>
<td>ISTMTCFF</td>
<td>The main router received an unrecognized work element.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC08</td>
<td>ISTMTCFF</td>
<td>Unrecognized data is received from the NetView application program across the programmable peripheral interface (PPI).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC09</td>
<td>ISTMTCFF</td>
<td>Unrecognized data is received from the NetView application program.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC10</td>
<td>ISTMTCFF</td>
<td>The PPI queue received from the NetView application program contains unrecognized data.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC11</td>
<td>ISTMTCFF</td>
<td>VTAM could not create a process scheduling table for MI.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC12</td>
<td>ISTMTCFF</td>
<td>The MDS-MU received from the NetView application program is unrecognized.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC13</td>
<td>ISTMTCFF</td>
<td>An expected outstanding request unit is not found on the outstanding request unit list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC14</td>
<td>ISTMTCFF</td>
<td>An expected outstanding request unit is not found on the outstanding request unit list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTMTC15</td>
<td>ISTMTCFF</td>
<td>An expected session list entry is not found, but an ATL entry is found.</td>
<td>Mini</td>
</tr>
</tbody>
</table>

Network resource management probes

Table 32. FFST probes for network resource management (ISTNACxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTNAC01</td>
<td>ISTNACTT</td>
<td>VTAM cannot complete a normal CLOSE ACB and must force the CLOSE ACB.</td>
<td>Full</td>
</tr>
</tbody>
</table>

Presentation services probes

Table 33. FFST probes for presentation services (ISTPSCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTOPSC01</td>
<td>ISTOPSCMD</td>
<td>CSM pool control block corrupted.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC02</td>
<td>ISTOPSCMD</td>
<td>A pool token was specified that is not valid.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 33. FFST probes for presentation services (ISTPSCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTOPSC03</td>
<td>ISTOPCMD</td>
<td>CSM detected an MVS system error.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC04</td>
<td>ISTOPCMD</td>
<td>An unexpected CSM reason code was encountered.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC06</td>
<td>ISTOPCFR</td>
<td>A buffer token was specified that is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC07</td>
<td>ISTOPCFR</td>
<td>CSM detected an MVS system error.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC08</td>
<td>ISTOPCFR</td>
<td>An unexpected CSM reason code was encountered.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC09</td>
<td>ISTOPCFP</td>
<td>A buffer token was specified that is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC10</td>
<td>ISTOPCFP</td>
<td>CSM detected an MVS system error.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC11</td>
<td>ISTOPCFP</td>
<td>An unexpected CSM reason code was encountered.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC12</td>
<td>ISTOPCMD</td>
<td>Data resides in a fixed I/O buffer instead of a CSM buffer.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTOPSC13</td>
<td>ISTOPCBM</td>
<td>Unexpected CSM reason code.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTOPSC14</td>
<td>ISTOPCMF</td>
<td>Unexpected CSM reason code.</td>
<td>Full</td>
</tr>
</tbody>
</table>

Recovery manager (RV) probes

The field RVM_FFST_ID within the ISTRVM control block contains additional information about the location from which the FFST PROBE was triggered.

Table 34. FFST probes for recovery manager (ISTRVMxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTRVM00</td>
<td>ISTRVMRT</td>
<td>Unexpected condition</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM01</td>
<td>ISTRVMRT</td>
<td>Unexpected RUPE</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM02</td>
<td>ISTRVMRT</td>
<td>Unexpected Data Recovered signal</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM03</td>
<td>ISTRVMIN</td>
<td>Unexpected condition</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM04</td>
<td>ISTRVMOS</td>
<td>Unexpected Data Recovered signal</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM05</td>
<td>ISTRVMCR</td>
<td>Unexpected condition</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM06</td>
<td>ISTRVMDL</td>
<td>Unexpected condition</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM07</td>
<td>ISTRVMDL</td>
<td>Unexpected condition</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM08</td>
<td>ISTRVMDS</td>
<td>Unexpected condition</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM0A</td>
<td>ISTRVRCM</td>
<td>Unknown RUPE from RCM</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM0B</td>
<td>ISTRVRI</td>
<td>Incorrect state</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM0C</td>
<td>ISTRVRI</td>
<td>Received RTPALS in wrong FSM state</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM0D</td>
<td>ISTRVRI</td>
<td>Received COMPLETE in wrong FSM state</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM0E</td>
<td>ISTRVRPQ</td>
<td>Unknown RUPE from SS</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM0F</td>
<td>ISTRVRPQ</td>
<td>Unexpected CV on sessinfo rsp</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM10</td>
<td>ISTRVRRCM</td>
<td>Unexpected failure from TSCM5</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 34. FFST probes for recovery manager (ISTRVMxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTRVM11</td>
<td>ISTTSCM2</td>
<td>Unexpected control block in chain</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM12</td>
<td>ISTTSCM3</td>
<td>Unexpected control block in chain</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM13</td>
<td>ISTRVRCM</td>
<td>Unexpected failure from GETENTRY</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRVM14</td>
<td>ISTRVMXF</td>
<td>Unexpected failure</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRVM15</td>
<td>ISTTSM1D</td>
<td>TREE ADD failed</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRVM16</td>
<td>ISTTSM1E</td>
<td>TREE ADD failed</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRVM17</td>
<td>ISTRVMLM</td>
<td>LM table restore failure</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRVM18</td>
<td>ISTRVRPI</td>
<td>Received APPCINFO in wrong FSM</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRVM19</td>
<td>ISTTSM1A</td>
<td>Duplicate FID5 address in HIT</td>
<td>Full</td>
</tr>
</tbody>
</table>

Session services probes

Table 35. FFST probes for session services (ISTSSCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTSSC01</td>
<td>ISTSSCTM</td>
<td>RU parsing error.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSSC02</td>
<td>ISTSSCXM</td>
<td>Address conflict was detected during merge.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSSC03</td>
<td>ISTSSCXV</td>
<td>An SRT was missing for an autologon session. As a result, session setup failed.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSSC04</td>
<td>ISTSSCFC</td>
<td>FREEBLK failure</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSSC05</td>
<td>ISTSSCY</td>
<td>FREEBLK failure</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSSC06</td>
<td>ISTSSCKI</td>
<td>FREEBLK failure</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSSC07</td>
<td>ISTSXCQ1</td>
<td>No session control block could be located for the FQPCID supplied by the recovery PAB on the multinode persistent Session Started signal flow.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSSC08</td>
<td>ISTSXCU5</td>
<td>Session services could not locate the correct recovery PAB when attempting to send a response to the recovery PAB’s request to recover a multinode persistent session.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSSCZ1</td>
<td>ISTSSCYC</td>
<td>An attempt was made to dequeue an SIB from a PU’s SIB queue, but the SIB was not on the queue.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSSCZ2</td>
<td>ISTSSCZY</td>
<td>A loop was detected in ISTSSCZY.</td>
<td>Full</td>
</tr>
</tbody>
</table>

Session services CP-CP probes

Table 36. FFST probes for session services CP-CP (ISTSCCxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTSCC01</td>
<td>ISTSCCRT</td>
<td>A request processing element is received that contained an unrecognized request unit processing element (RUPE).</td>
<td>Mini</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>-------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTSCC02</td>
<td>ISTSCRCA</td>
<td>SSC attempted to add an adjacent node control block to the adjacent node control block list, but the add failed to complete normally.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC03</td>
<td>ISTSCCFM</td>
<td>The SSC coordinator finite state machine processor received null input in an unexpected state when one or more of the CP-CP sessions was in active or pending active status.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC04</td>
<td>ISTSCUAR</td>
<td>An internal SSC module passed an unrecognized session status to ISTSCUAR.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC05</td>
<td>ISTSCUAR</td>
<td>An internal SSC module passed an unrecognized session type to ISTSCUAR.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC06</td>
<td>ISTSCUBS</td>
<td>An internal SSC module passed an unrecognized session type to ISTSCUBS.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC07</td>
<td>ISTSCUBS</td>
<td>An internal SSC module passed an unrecognized session status to ISTSCUBS.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC08</td>
<td>ISTSCUDS</td>
<td>An internal SSC module passed an unrecognized session type to ISTSCUDS.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC09</td>
<td>ISTSCUNO</td>
<td>SSC entered an unexpected state on a contention-winner session.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC10</td>
<td>ISTSCUNO</td>
<td>SSC entered an unexpected state on a contention-loser session.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC11</td>
<td>ISTSCUNO</td>
<td>An internal SSC module passed an unrecognized session type to ISTSCUNO.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC15</td>
<td>ISTSCCFM</td>
<td>The SSC coordinator finite state machine processor received nonnull input in an unexpected state when one or more of the CP-CP sessions is in active or pending active status.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC16</td>
<td>ISTSCTRQ</td>
<td>The REQUEST_CP_CAPABILITIES_TP received as input a signal that is neither an ACT_CP_CP_SESSION request nor a CONTINUE_CW response.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC19</td>
<td>ISTSCTR</td>
<td>The SEND_REJECT_CONVGRP_TP received an input signal other than a DEACTIVATE_CP_CP_SESSION request.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC20</td>
<td>ISTSCFM</td>
<td>The SSC coordinator finite state machine processor encountered a should-not-occur condition and no CP-CP session cleanup is required.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSCC21</td>
<td>ISTSCCRT</td>
<td>An adjacent node control block is not found for the received signal.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSCC23</td>
<td>ISTSCCRT</td>
<td>A processing element is queued to session services CP-CP (SSC) but is not recognized to be either a request unit processing element (RUPE) or a timer queue element.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
**Session services LU-LU probes**

The following table contains the LU-LU session services (SSL) software probes. When an SSL software probe is triggered, an error has occurred while processing network session services for an LU-LU connection. The search, session initiation, or request causing the error is rejected. VTAM will attempt to recover from the error and continue processing subsequent requests. Recovery might not be possible for some types of errors and subsequent requests might fail, terminals might hang, and other types of errors might occur.

A system dump is taken to assist in identifying the source of the problem. A VABEND macroinstruction can also be processed if specified by the probe instruction. The processing element is freed and processing continues.

*Table 37. FFST probes for session services LU-LU (ISTSLCxx)*

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTSLC01</td>
<td>ISTSLCRT</td>
<td>An unknown interprocess signal was dispatched to the SSL component process anchor block (PAB).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC02</td>
<td>ISTSLRDP</td>
<td>Finite state machine information is missing when a SEARCH_RPY signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC03</td>
<td>ISTSLRDP</td>
<td>Finite state machine information is missing when a CDINIT SEARCH_RPY signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC04</td>
<td>ISTSLRDP</td>
<td>Finite state machine information is missing when an IOCD SEARCH_RPY signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC05</td>
<td>ISTSLRDP</td>
<td>Finite state machine information is missing when a NOTIFY SEARCH_RPY signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC06</td>
<td>ISTSLRDP</td>
<td>Finite state machine information is missing when a SEARCH_RPY signal is processed and a storage failure occurs.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC07</td>
<td>ISTSLRXS</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC08</td>
<td>ISTSLTP1</td>
<td>An unexpected combination of conditions was encountered during third-party initiated finite state machine processing.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC09</td>
<td>ISTSLBXN</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC10</td>
<td>ISTSLURR</td>
<td>Route Selection control vector (RSCV) information is missing from SSL's session initiation control block when performing a build request for a RECOMPUTE_ROUTE signal.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC11</td>
<td>ISTSLBXJ</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC12</td>
<td>ISTSLRXJ</td>
<td>Finite state machine information is missing when an INIT_OTHER_COMP signal is processed, and a storage failure occurred.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC13</td>
<td>ISTSLRXJ</td>
<td>Finite state machine information is missing when an INIT_OTHER_COMP signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTSLC14</td>
<td>ISTSLBXI</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC15</td>
<td>ISTSLUDS</td>
<td>An error occurred while the SSL component was removing a session initiation control block key from the list.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSLC16</td>
<td>ISTSLUDS</td>
<td>An error occurred while the SSL component was removing a session initiation control block from the list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC17</td>
<td>ISTSLUGT</td>
<td>The ACMDT's node type information for this node had an unknown value because it did not indicate whether this is a VTAM end node or a VTAM network node.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC18</td>
<td>ISTSLUBX</td>
<td>A build request was received for an unrecognized interprocess signal.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC19</td>
<td>ISTSLUBX</td>
<td>A build request was received for an unrecognized interprocess signal.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC20</td>
<td>ISTSLUBX</td>
<td>The SSL component could not remove a fully qualified procedure correlation identifier (FQPCID) from the list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC21</td>
<td>ISTSLUBX</td>
<td>The SSL component could not remove a FQPCID_MODIFIER from the list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC22</td>
<td>ISTSLRXI</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC23</td>
<td>ISTSLVPE</td>
<td>An unrecognized signal was received from directory services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC24</td>
<td>ISTSLUSC</td>
<td>The SSL component was unable to determine the type of signal to send to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC25</td>
<td>ISTSLURX</td>
<td>An unrecognized interprocess signal was received from session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC26</td>
<td>ISTSLURD</td>
<td>An unrecognized interprocess signal was received from directory services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC27</td>
<td>ISTSLSNO</td>
<td>The Route Selection control vector (RSCV) information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC28</td>
<td>ISTSLSNO</td>
<td>The Route Selection control vector (RSCV) information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC29</td>
<td>ISTSLSNO</td>
<td>An unrecognized session initiation status was received for the current system state.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 37. FFST probes for session services LU-LU (ISTSLCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTSLC30</td>
<td>ISTLSNO</td>
<td>An unexpected combination of conditions was encountered during finite state machine processing initiated by the secondary logical unit (SLU) at a network node (NN).</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSLC31</td>
<td>ISTSLUBD</td>
<td>A build request was received for an unrecognized interprocess signal.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC32</td>
<td>ISTSLSED</td>
<td>An unexpected combination of conditions was encountered during finite state machine processing initiated by the secondary logical unit (SLU) at an end node (EN) destination logical unit (DLU).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC33</td>
<td>ISTSLRXO</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC34</td>
<td>ISTSLRXQ</td>
<td>Finite state machine information is missing when a QUEUED signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC35</td>
<td>ISTSLRXC</td>
<td>Finite state machine information is missing when a SESS_REQ_COMP signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC36</td>
<td>ISTSLRXC</td>
<td>Finite state machine information is missing when a SESS_REQ_COMP signal is processed and storage problems occur.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC37</td>
<td>ISTSLBXV</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC38</td>
<td>ISTSLBXS</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC39</td>
<td>ISTSLRXD</td>
<td>Finite state machine information is missing when a DEQUEUE signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC40</td>
<td>ISTSLPND</td>
<td>An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at a network node.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC41</td>
<td>ISTSLRXV</td>
<td>Finite state machine information is missing when a PROV_SC signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC42</td>
<td>ISTSLPNO</td>
<td>The Route Selection control vector (RSCV) information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC43</td>
<td>ISTSLPNO</td>
<td>The RSCV information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC44</td>
<td>ISTSLPNO</td>
<td>An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at a network node.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSLC45</td>
<td>ISTSLPNO</td>
<td>The RSCV information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTSLC46</td>
<td>ISTSLBXQ</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC47</td>
<td>ISTSLBXC</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC48</td>
<td>ISTSLRTT</td>
<td>Finite state machine information is missing when a RECOMPUTE_ROUTE_RPY signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC49</td>
<td>ISTSLRTR</td>
<td>Finite state machine information is missing when a REQ_ROUTE_RPY signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC50</td>
<td>ISTSLRDR</td>
<td>Finite state machine information is missing when a SEARCH_REQ signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC51</td>
<td>ISTSLRTC</td>
<td>Class of service (COS) and transmission priority field (TPF) control vector information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC52</td>
<td>ISTSLRTC</td>
<td>Finite state machine information is missing when a REQ_COS_TPF_RPY signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC53</td>
<td>ISTSLRTV</td>
<td>Finite state machine information is missing when a REQ_TG_VECTORS_RPY signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC54</td>
<td>ISTSLRNT</td>
<td>The NOTIFY generalized data stream (GDS) variable being processed contained a NOTIFY_TYPE that was unknown or not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC55</td>
<td>ISTSLRNT</td>
<td>Notify control vector 80 (CV80) information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC56</td>
<td>ISTSLRNT</td>
<td>Notify CV81 information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC57</td>
<td>ISTSLRDN</td>
<td>The CDINIT GDS variable information is missing from a NON_VERIFY_REQ signal received from directory services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC58</td>
<td>ISTSLRDC</td>
<td>Finite state machine information is missing when a CHAIN_FLOW signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC59</td>
<td>ISTSLPXR</td>
<td>An unexpected combination of conditions was encountered during processing of the PLU-initiate exit border node preprocessor finite state machine (FSM).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC60</td>
<td>ISTSLPXT</td>
<td>An unexpected combination of conditions was encountered during processing of the PLU-initiate exit border node postprocessor finite state machine.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC61</td>
<td>ISTSLPYR</td>
<td>An unexpected combination of conditions was encountered during processing of the PLU-initiate entry border node preprocessor finite state machine.</td>
<td>Mini</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTSLC62</td>
<td>ISTSLPNS</td>
<td>RSCV information is missing from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC63</td>
<td>ISTSLPNS</td>
<td>An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at a network node.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC64</td>
<td>ISTSLPED</td>
<td>An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at an end node destination logical unit.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC65</td>
<td>ISTSLPEO</td>
<td>An unexpected combination of conditions was encountered during finite state machine processing initiated by the primary logical unit (PLU) at an end node origination logical unit.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC66</td>
<td>ISTSLBXT</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC67</td>
<td>ISTSLBXR</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC68</td>
<td>ISTSLBXP</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC69</td>
<td>ISTSLBZO</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC70</td>
<td>ISTSLBZE</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC71</td>
<td>ISTSLBXD</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC72</td>
<td>ISTSLBDL</td>
<td>A failure occurred while performing postprocessor information processing for a signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC73</td>
<td>ISTSLSND</td>
<td>An unrecognized session initiation status was received for the current system state in SLU-initiate finite state machine processing.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC74</td>
<td>ISTSLSND</td>
<td>An unexpected combination of conditions occurred during SLU-initiate finite state machine processing at a network node.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC75</td>
<td>ISTSLSEO</td>
<td>An unexpected combination of conditions occurred during SLU-initiate finite state machine processing at an end node origination logical unit.</td>
<td>Mini</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>-------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTSLC76</td>
<td>ISTSLUCS</td>
<td>The SSL component was unable to remove the key entry for the session initiation control block from the list.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSLC77</td>
<td>ISTSLRXN</td>
<td>Finite state machine information is missing when a PEND_SC signal is processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC78</td>
<td>ISTSLUSL</td>
<td>An unrecognized LU role was provided for a session initiation control block find request.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC79</td>
<td>ISTSLCRT</td>
<td>An unrecognized work element was dispatched to the SSL process anchor block (PAB).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC80</td>
<td>ISTSLR63</td>
<td>The SSL component was unable to remove a CV63 from the control vector list of the session initiation control block.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC81</td>
<td>ISTSLBLC</td>
<td>A CV82 control vector contains a network-qualified sender name that is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC82</td>
<td>ISTSLBLC</td>
<td>A CV82 control vector contains a network-qualified host receiver name that is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC84</td>
<td>ISTSLUNV</td>
<td>The original class of service (COS) associated with the SLU-initiate request was not found in the parent session initiation control block during a nonverify search sent in response to a SLU-initiate request.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC85</td>
<td>ISTSLRDI</td>
<td>The SSL component received a SESS_INIT_INFO_REQ signal which contained inconsistent data.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSLC86</td>
<td>ISTSLBXB</td>
<td>Failure occurred when performing front-end processing for a BN_SESS_RPY signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC87</td>
<td>ISTSLBXH</td>
<td>Failure occurred when performing front-end processing for a CACHE_BN_INFO signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC88</td>
<td>ISTSLBTM</td>
<td>The network-qualified adjacent nonnative CP name is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC89</td>
<td>ISTSLRXB</td>
<td>The network-qualified adjacent LU name is not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC90</td>
<td>ISTSLRXB</td>
<td>Failure occurred when performing front-end processing for a BN_SESS_RPY signal being sent to session services.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC91</td>
<td>ISTSLRTM</td>
<td>Finite state machine information was missing when a PROCESS_MAP_COS signal from topology routing services (TRS) was being processed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC92</td>
<td>ISTSLUBX</td>
<td>The RSCV that should have been in the session initiation control block when building a CACHE_BN_INFO to session services was missing.</td>
<td>Mini</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ISTSLC93</td>
<td>ISTSLRXB</td>
<td>The class of service (COS) was received on a BN_SESS_REQ, but the primary logical unit (PLU) COS from LOCATE time processing could not be found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC94</td>
<td>ISTSLPYT</td>
<td>An unexpected combination of conditions was encountered during processing of the PLU-initiate entry border node postprocessor finite state machine.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC95</td>
<td>ISTLSXR</td>
<td>An unexpected combination of conditions was encountered during processing of the SLU-initiate exit border node preprocessor finite state machine.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC96</td>
<td>ISTLSXT</td>
<td>An unexpected combination of conditions was encountered during processing of the SLU-initiate exit border node postprocessor finite state machine.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC97</td>
<td>ISTLSYR</td>
<td>An unexpected combination of conditions was encountered during processing of the SLU-initiate entry border node preprocessor finite state machine.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC98</td>
<td>ISTLSYT</td>
<td>An unexpected combination of conditions was encountered during processing of the SLU-initiate entry border node postprocessor finite state machine.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLC99</td>
<td>ISTSLUBR</td>
<td>An unexpected combination of conditions was encountered during border node request finite state machine processing.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLCA0</td>
<td>ISTSLUFR</td>
<td>The finite state machine router was called for a situation where a finite state machine (FSM) is not used.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLCA1</td>
<td>ISTLSNO</td>
<td>Route Selection Control Vector (RSCV) missing from the session initiation control block’s control vector list.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLCA2</td>
<td>ISTLUAS</td>
<td>An unexpected combination of conditions was encountered during dependent LU requester search finite state machine processing.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSLCA3</td>
<td>ISTSLR2B</td>
<td>Missing subvector on CV2B (RSCV).</td>
<td>Full</td>
</tr>
<tr>
<td>ISTSLCA4</td>
<td>ISTSLRDI</td>
<td>The DLU is nonnative, but the previously stored DLU information and the DLU information about the SESS_INIT_INFO_REQ IPS just received do not match.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLCA5</td>
<td>ISTSLRDP</td>
<td>The DLU is nonnative, but the previously stored DLU information and the DLU information about the SEARCH_RPY IPS just received do not match.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLCA7</td>
<td>ISTSLRDS</td>
<td>A call to ISTSXCXS failed, either unknown RU or insufficient storage.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTSLCA8</td>
<td>ISTLU85</td>
<td>SSLCNO action invoked in FSMe.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
### Storage management probes

**Table 38. FFST probes for storage management (ISTORCxx)**

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTORC01</td>
<td>ISTORCHB</td>
<td>Unexpected I/O condition was detected. The output contains the buffer pool control block (BPCB) and one page of storage from each buffer pool expansion block (PXB).</td>
<td>Full</td>
</tr>
<tr>
<td>ISTORCZ1</td>
<td>ISTORCGB</td>
<td>Issued when an overlay of a GETBLKed header is detected and before the element is dequeued from the FBQE.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTORCZ2</td>
<td>ISTORCFB</td>
<td>GETBLD header overlay</td>
<td>Full</td>
</tr>
</tbody>
</table>

### Topology and routing services probes

**Table 39. FFST probes for topology and routing services (ISTTRCxx)**

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTRC01</td>
<td>ISTTRCRT</td>
<td>An unknown work element is received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC02</td>
<td>ISTTRCRT</td>
<td>An unknown signal is contained with the work element.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC03</td>
<td>ISTTRQDN</td>
<td>Issued when attempting to clean up storage for a node information block that is not found.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTRC04</td>
<td>ISTTRIFP</td>
<td>While processing a topology database update (TDU), an unrecognized finite state machine state is encountered.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC05</td>
<td>ISTTRINP</td>
<td>An internal parsing error has occurred. An Input_TDU is encountered with a mismatch between the number of node vectors and the number of associated transmission group (TG) vectors.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC06</td>
<td>ISTTRPSH</td>
<td>An unrecognized transmission group (TG) type is received on a RQ_SINGLE_HOP_ROUTE signal.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC07</td>
<td>ISTTRQDT</td>
<td>Unable to delete a node from a tree because the node cannot be found on the tree.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTRC08</td>
<td>ISTTRTLT</td>
<td>An unrecognized partner node type is received on a TG_UPDATE signal.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC09</td>
<td>ISTTRTLT</td>
<td>The transmission group (TG) direction specified on a TG_UPDATE signal is not recognized.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC10</td>
<td>ISTTRRPP</td>
<td>A problem is encountered while attempting to uncache the session trees from the topology and routing services (TRS) database.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTRC11</td>
<td>ISTTRRPP</td>
<td>While scanning the topology and routing services (TRS) database an unidentified resource is found.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC12</td>
<td>ISTTRDGM</td>
<td>An unrecognized secondary LU (SLU) node identification is received on a REQ_MULTIPLE ROUTES signal.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
### Table 39. FFST probes for topology and routing services (ISTTRCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTRC13</td>
<td>ISTTRRR</td>
<td>A route to a gateway node is received on a REQUEST_ROUTE; however, the signal contained unrecognized information.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRC14</td>
<td>ISTTRATE</td>
<td>Unidentified resource is in database.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTRCZ1</td>
<td>ISTTRRGT</td>
<td>Class of service information or node information not found.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTRCZ2</td>
<td>ISTTRCLT</td>
<td>Element is not on list.</td>
<td>Full</td>
</tr>
</tbody>
</table>

### Trace services probes

**Table 40. FFST probes for trace services (ISTRACxx)**

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTRAC01</td>
<td>ISTRACZT</td>
<td>FFST dump from the VTAM trap module.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRAC02</td>
<td>ISTRACZT</td>
<td>FFST dump from the VTAM trap module consisting of the ATCVT and the VTAM internal trace (VIT).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTRACZ3</td>
<td>ISTITCAS</td>
<td>FFST full dump for a predetermined APPC sense code.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTRACZ4</td>
<td>ISTITCAS</td>
<td>FFST full dump for a predetermined RPL6 return code.</td>
<td>Full</td>
</tr>
</tbody>
</table>

### Transmission control — Data link control (TC-DLC) probes

**Table 41. FFST probes for TC-DLC (ISTTCCxx)**

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTCC01</td>
<td>ISTTCCTD</td>
<td>Multiple function keywords were encountered. TCP-DLC supports ACTPATH, DACTPATH, OPENPATH, and CLOSEPATH.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTCC02</td>
<td>ISTTCCRI</td>
<td>Registration Manager or the DLC sent an unrecognizable RUPE.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTCC03</td>
<td>IUTLLCID</td>
<td>Unrecognized primitive.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTCC04</td>
<td>ISTTCCRI</td>
<td>An unexpected RUPE was encountered when a RUPE was sent to Registration Manager.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTCC05</td>
<td>IUTLLCID</td>
<td>Input is not a RUPE response.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
### Transmission subsystem probes

**Table 42. FFST probes for transmission subsystem (ISTTSCxx)**

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTSC01</td>
<td>ISTTSCRI</td>
<td>Inbound path information unit (PIU) with an out-of-order sequence number was detected. The output includes the transmission subsystem control block (TSCB) containing the PIU with the unexpected sequence number and the virtual route control block (VRBLK) containing the expected sequence number.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC02</td>
<td>ISTTSC8S</td>
<td>Unrecognized multipath channel sweep work element was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC03</td>
<td>ISTTSC8E</td>
<td>Unexpected multipath channel sweep finite state machine state was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC04</td>
<td>ISTTSC8I</td>
<td>Zero active read subchannel count was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC05</td>
<td>ISTTSC8E</td>
<td>Zero active write subchannel count was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC06</td>
<td>ISTTSC8E</td>
<td>Zero read sweep count decremented.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC07</td>
<td>ISTTSC8E</td>
<td>Zero write sweep count decremented.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC08</td>
<td>ISTTSC9L</td>
<td>Unexpected input in current state was received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC09</td>
<td>ISTTSC9C</td>
<td>Unexpected input in current state was received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC10</td>
<td>ISTTSC8E</td>
<td>Multipath channel incorrect path information unit (PIU) was built.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC11</td>
<td>ISTTSC8D</td>
<td>Unexpected input was received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC12</td>
<td>ISTTSC8E</td>
<td>Multipath channel group lost the last read device.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC13</td>
<td>ISTTSC8E</td>
<td>An attempt to send a data element response failed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC14</td>
<td>ISTTSC8E</td>
<td>Multipath channel device busy was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC15</td>
<td>ISTTSCM8</td>
<td>Multipath channel PU services process anchor block (PUPAB) finite state machine error was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC16</td>
<td>ISTTSCM8</td>
<td>Unable to obtain storage for a request unit processing element (RUPE).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC17</td>
<td>ISTTSC8X</td>
<td>Multipath channel PUPAB finite state machine error was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC18</td>
<td>ISTTSC8X</td>
<td>Unable to obtain storage for a request unit processing element (RUPE).</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC19</td>
<td>ISTTSC8B</td>
<td>Empty node control block (NCB) pending queue was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC20</td>
<td>ISTTSC87</td>
<td>Unexpected value for request/response unit (RU) was received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC21</td>
<td>ISTTSC9X</td>
<td>Unexpected input in current state was received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC22</td>
<td>ISTTSC9P</td>
<td>Unexpected input was received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC23</td>
<td>ISTTSC9D</td>
<td>Unexpected input was received.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 42. FFST probes for transmission subsystem (ISTTSCxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTSC24</td>
<td>ISTTSC8I</td>
<td>Unexpected multipath channel sweep finite state machine state was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC25</td>
<td>ISTTSC8I</td>
<td>Zero active write subchannel count was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC26</td>
<td>ISTTSC8I</td>
<td>Zero read sweep count decremented.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC27</td>
<td>ISTTSC8I</td>
<td>Zero write sweep count decremented.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC28</td>
<td>ISTTSCM8</td>
<td>Unexpected input was received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC29</td>
<td>ISTTSC8X</td>
<td>Unexpected input was received.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC30</td>
<td>ISTTSC6F</td>
<td>ISTTSC6F fails to dequeue the TRGCB from ATCNCBQ during deactivation of a major node. This can result in an abend in ISTTSCWU.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC31</td>
<td>ISTTSCUD</td>
<td>A problem occurred trying to free a CSM buffer.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTSC32</td>
<td>ISTTSC8E</td>
<td>Zero active write subchannel count was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC33</td>
<td>ISTTSC8E</td>
<td>Multipath channel incorrect path information unit (PIU) was built.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC34</td>
<td>ISTTSC8E</td>
<td>An attempt to send a data element response failed.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC35</td>
<td>ISTTSC8E</td>
<td>Multipath channel device busy was detected.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTSC36</td>
<td>ISTTSCYD</td>
<td>Sense code 08150004 received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTSC37</td>
<td>ISTTSCYD</td>
<td>Sense code 08090033 received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTSC38</td>
<td>ISTBSCUB</td>
<td>Sense code 08150004 received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTSC39</td>
<td>ISTBSCUB</td>
<td>Sense code 08090033 received.</td>
<td>Full</td>
</tr>
<tr>
<td>ISTTSCZ1</td>
<td>ISTTSCON</td>
<td>Negative NCBUSECT.</td>
<td>Mini</td>
</tr>
</tbody>
</table>

VTAM topology agent probes

Table 43. FFST probes for the VTAM topology agent (ISTTACxx)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTAC01</td>
<td>ISTTACOS</td>
<td>The topology agent received a CMIP message string with a syntax error. The topology agent cannot process or respond to the message.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC03</td>
<td>ISTTAAIGP</td>
<td>The topology agent received a CMIP GET request with a syntax error. The topology agent cannot process or respond to the request.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC04</td>
<td>ISTTaidaN</td>
<td>The topology agent received a distinguished name with a syntax error in a CMIP request. The topology agent cannot process or respond to the request.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC05</td>
<td>ISTTACII</td>
<td>The topology agent received a request for initialization at a time when the topology agent was in an internal state other than inactive state.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC06</td>
<td>ISTTACII</td>
<td>The topology agent received a register-accept acknowledgment at a time when the topology agent was in an internal state other than initializing.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 43. FFST probes for the VTAM topology agent (ISTTACxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTAC07</td>
<td>ISTTACRT</td>
<td>The topology agent was dispatched with a work element type that the topology agent does not support, such as a work element other than RUPE or TQE.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC08</td>
<td>ISTTACRT</td>
<td>The topology agent was dispatched with a RUPE work element containing an OPCODE that is not supported by the topology agent.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC09</td>
<td>ISTTACRT</td>
<td>The topology agent received a message from CMIP services with an unknown API message type in the API header section of the message.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
| ISTTAC10   | ISTTACOS| The topology agent received a CMIP message that was not identified as type ROIVapdu. The CMIP message was not a request or linked reply. It was identified as one of the following items, which the topology agent does not support:  
  • RORSapdu  
  • ROERapdu  
  • RORJapdu                                                                                           | Mini      |
| ISTTAC11   | ISTTAIPA| The topology agent received a CMIP message with a syntax error in one of the following items:                                                                                                               | Mini      |
| ISTTAC12   | ISTTACOS| The topology agent received a message from configuration services or topology and routing services indicating that a snapshot was in initial data phase when that snapshot was in a phase other than initial data phase. | Mini      |
| ISTTAC13   | ISTTAISP| The topology agent received a CMIP CANCEL-GET message with a syntax error.                                                                                                                                   | Mini      |
| ISTTAC14   | ISTTACRT| The topology agent has written a SNAlocaltopology response string in an output buffer. The string is greater than the size of the output buffer and has overlaid other storage.                        | Mini      |
| ISTTAC16   | ISTTAOAN| The topology agent has written a SNAnetwork response string in an output buffer. The string is greater than the size of the output buffer and has overlaid other storage.                                    | Mini      |
| ISTTAC17   | ISTTAOSN| The topology agent has written a SNAnetwork response string in an output buffer. The string is greater than the size of the output buffer and has overlaid other storage.                                    | Mini      |
Table 43. FFST probes for the VTAM topology agent (ISTTACxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTAC18</td>
<td>ISTTAOLC</td>
<td>The topology agent has written an LUcollection response string in an output buffer. The string is greater than the size of the output buffer and has overlaid other storage.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC19</td>
<td>ISTTAOLI</td>
<td>The topology agent has written a logicalUnitIndex response string in an output buffer. The string is greater than the size of the output buffer and has overlaid other storage.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC20</td>
<td>ISTTAONO</td>
<td>The topology agent, in processing notifications, has written data beyond the storage allocated for it.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC21</td>
<td>ISTTAOLG</td>
<td>The topology agent has written a transmission group distinguished name string to a data area that is smaller than the string.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC22</td>
<td>ISTTAOLT</td>
<td>The topology agent received a SNAlocaltopology message from configuration services containing an object block specifying an unknown object type.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC23</td>
<td>ISTTAOLC</td>
<td>The topology agent received an LUcollection message from configuration services containing an object block specifying an unknown or inappropriate object type.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC24</td>
<td>ISTTAOLI</td>
<td>The topology agent received a logicalUnitIndex message from configuration services containing an object block specifying an unknown or inappropriate object type.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC25</td>
<td>ISTTACA2</td>
<td>The topology agent definitionGroupName attribute builder encountered an unknown type of major node.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC26</td>
<td>ISTTAOPU</td>
<td>The topology agent received a snapshot update data message from configuration services or topology and routing services containing an unrecognized return code.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC27</td>
<td>ISTTAOPI</td>
<td>The topology agent received a snapshot initial data message from configuration services or topology and routing services containing an unrecognized return code.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC28</td>
<td>ISTTAOSN</td>
<td>The topology agent received a SNAnetwork message from configuration services containing an object block specifying an unknown or inappropriate object type.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC29</td>
<td>ISTTAISE</td>
<td>The topology agent has written a SET response string in an output buffer. The string is greater than the size of the output buffer and has overlaid other storage.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC30</td>
<td>ISTTAOSS</td>
<td>The topology agent received a synchronous return code from the CMIP services API indicating a permanent error has been encountered.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Table 43. FFST probes for the VTAM topology agent (ISTTACxx) (continued)

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTTAC31</td>
<td>ISTTAIPS</td>
<td>The topology agent received a CMIP SET message with a syntax error.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC32</td>
<td>ISTTAOUQ</td>
<td>The topology agent received a snapshot update message from configuration services containing an object block specifying an unknown object type.</td>
<td>Mini</td>
</tr>
<tr>
<td>ISTTAC33</td>
<td>ISTTAOUQ</td>
<td>A message routed to the module for the topology agent merge queue controller contained an operation code that is not supported by that module.</td>
<td>Mini</td>
</tr>
</tbody>
</table>
Appendix E. Communications storage manager (CSM)  
FFST probes

The following table describes the naming conventions for the FFST probe name. **IVTSMC01** is used as an example.

<table>
<thead>
<tr>
<th>Characters</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,2,3</td>
<td>IVT</td>
<td>These characters represent the product identifier. For CSM probes, these characters are IVT.</td>
</tr>
<tr>
<td>4, 5</td>
<td>SM</td>
<td>These characters represent the CSM component identifier. In this example, SM is the component identifier for the Storage Manager component.</td>
</tr>
<tr>
<td>6</td>
<td>C</td>
<td>For CSM probes, this character is C.</td>
</tr>
<tr>
<td>7, 8</td>
<td>01</td>
<td>These characters represent the probe identification number. This number is not duplicated.</td>
</tr>
</tbody>
</table>

### CSM probes

<table>
<thead>
<tr>
<th>Probe name</th>
<th>Module</th>
<th>Description</th>
<th>Dump type</th>
</tr>
</thead>
<tbody>
<tr>
<td>IVTSMC03</td>
<td>IVTSMCCP</td>
<td>Buffer size (BUFSIZE) or buffer source (BUFSIZE) specification on a CREATE_POOL request was not valid.</td>
<td>Mini</td>
</tr>
<tr>
<td>IVTSMC04</td>
<td>IVTSMCCP</td>
<td>SRB schedule of IVTSMCEX failed for a CREATE_POOL request.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC05</td>
<td>IVTSMCGF</td>
<td>A storage error occurred.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC06</td>
<td>IVTSMCFB</td>
<td>SRB schedule of IVTSMCEX failed during pool contraction.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC07</td>
<td>IVTSMCEX</td>
<td>Cross-memory post failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC08</td>
<td>IVTSMCF</td>
<td>SRB schedule of IVTSMCEX failed during pool contraction.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC09</td>
<td>IVTSMCFX</td>
<td>SRB schedule of IVTSMCE failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC0A</td>
<td>IVTSMCBF</td>
<td>SRB schedule of IVTSMCSD failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC0B</td>
<td>IVTSMCFR</td>
<td>SRB schedule of IVTSMCSD failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC0C</td>
<td>IVTSMCWL</td>
<td>SRB schedule of IVTSMCSD failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC0D</td>
<td>IVTSMCFX</td>
<td>Post failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC0E</td>
<td>IVTSMCFR</td>
<td>Post failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC0F</td>
<td>IVTSMCF</td>
<td>SRB schedule of IVTSMCSD failed.</td>
<td>Full</td>
</tr>
<tr>
<td>Probe name</td>
<td>Module</td>
<td>Description</td>
<td>Dump type</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>IVTSMC10</td>
<td>IVTSMCGB</td>
<td>The schedule of the SRB for the side routine failed in IVTSMCFB.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC11</td>
<td>IVTSMCGB</td>
<td>SRB schedule of IVTSMCSD failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC12</td>
<td>IVTSM6BF</td>
<td>SRB schedule of IVTSMCSD failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC13</td>
<td>IVTSM6GB</td>
<td>SRB schedule of IVTSMCSD failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC14</td>
<td>IVTSM6FB</td>
<td>SRB schedule of IVTSMCSD failed.</td>
<td>Full</td>
</tr>
<tr>
<td>IVTSMC15</td>
<td>IVTSM6FB</td>
<td>SRB schedule of IVTSMCEX failed.</td>
<td>Full</td>
</tr>
</tbody>
</table>
Appendix F. Architectural specifications

This appendix lists documents that provide architectural specifications for the SNA Protocol. The APPN Implementers' Workshop (AIW) architecture documentation includes the following architectural specifications for SNA APPN and HPR:

• APPN Architecture Reference (SG30-3422-04)
• APPN Branch Extender Architecture Reference Version 1.1
• APPN Dependent LU Requester Architecture Reference Version 1.5
• APPN Extended Border Node Architecture Reference Version 1.0
• APPN High Performance Routing Architecture Reference Version 4.0
• SNA Formats (GA27-3136-20)
• SNA Technical Overview (GC30-3073-04)

For more information, see the AIW documentation page at http://www.ibm.com/support/docview.wss?rs=852&uid=swg27017843.

The following RFC also contains SNA architectural specifications:

• RFC 2353 APPN/HPR in IP Networks APPN Implementers' Workshop Closed Pages Document

Appendix G. Accessibility

Publications for this product are offered in Adobe Portable Document Format (PDF) and should be compliant with accessibility standards. If you experience difficulties when using PDF files, you can view the information through the z/OS Internet Library website http://www.ibm.com/systems/z/os/zos/library/bkser/ or IBM Knowledge Center http://www.ibm.com/support/knowledgecenter. If you continue to experience problems, send a message to Contact z/OS web page (www.ibm.com/systems/z/os/zos/webqs.html) or write to:

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS enable users to:

- Use assistive technologies such as screen readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size

Using assistive technologies

Assistive technology products, such as screen readers, function with the user interfaces found in z/OS. Consult the assistive technology documentation for specific information when using such products to access z/OS interfaces.

Keyboard navigation of the user interface

Users can access z/OS user interfaces using TSO/E or ISPF. See z/OS TSO/E Primer, z/OS TSO/E User's Guide, and z/OS ISPF User's Guide Vol I for information about accessing TSO/E and ISPF interfaces. These guides describe how to use TSO/E and ISPF, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.
This information was developed for products and services that are offered in the USA or elsewhere.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user’s responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-1785 United States of America

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan Ltd. 19-21, Nihonbashi-Hakozakicho, Chuo-ku Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

This information could include missing, incorrect, or broken hyperlinks. Hyperlinks are maintained in only the HTML plug-in output for the Knowledge Centers. Use of hyperlinks in other output formats of this information is at your own risk.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation Site Counsel 2455 South Road Poughkeepsie, NY 12601-5400 USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.
Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM’s future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.
IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user’s name, email address, phone number, or other personally identifiable information for purposes of enhanced user usability and single sign-on configuration. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM’s Privacy Policy at ibm.com/privacy and IBM’s Online Privacy Statement at ibm.com/privacy/details in the section entitled “Cookies, Web Beacons and Other Technologies,” and the “IBM Software Products and Software-as-a-Service Privacy Statement” at ibm.com/software/info/product-privacy.

Policy for unsupported hardware

Various z/OS elements, such as DFSMS, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: IBM Lifecycle Support for z/OS (www.ibm.com/software/support/systemsz/lifecycle)
- For information about currently-supported IBM hardware, contact your IBM representative.
Programming interface information

This publication documents information NOT intended to be used as Programming Interfaces of z/OS Communications Server.

Policy for unsupported hardware

Various z/OS elements, such as DFSMS, HCD, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.
Bibliography

This bibliography contains descriptions of the documents in the z/OS Communications Server library. z/OS Communications Server documentation is available online at the z/OS Internet Library web page at http://www.ibm.com/systems/z/os/zos/library/bkserv/.

z/OS Communications Server library updates
Updates to documents are also available on RETAIN and in information APARs (info APARs). Go to http://www.software.ibm.com/support to view information APARs.

- z/OS V2R1 Communications Server New Function APAR Summary
- z/OS V2R2 Communications Server New Function APAR Summary
- z/OS V2R3 Communications Server New Function APAR Summary

z/OS Communications Server information
z/OS Communications Server product information is grouped by task in the following tables.

### Planning

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: New Function Summary</td>
<td>GC27-3664</td>
<td>This document is intended to help you plan for new IP or SNA functions, whether you are migrating from a previous version or installing z/OS for the first time. It summarizes what is new in the release and identifies the suggested and required modifications needed to use the enhanced functions.</td>
</tr>
<tr>
<td>z/OS Communications Server: IPv6 Network and Application Design Guide</td>
<td>SC27-3663</td>
<td>This document is a high-level introduction to IPv6. It describes concepts of z/OS Communications Server's support of IPv6, coexistence with IPv4, and migration issues.</td>
</tr>
</tbody>
</table>

### Resource definition, configuration, and tuning

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP Configuration Guide</td>
<td>SC27-3650</td>
<td>This document describes the major concepts involved in understanding and configuring an IP network. Familiarity with the z/OS operating system, IP protocols, z/OS UNIX System Services, and IBM Time Sharing Option (TSO) is recommended. Use this document with the z/OS Communications Server: IP Configuration Reference.</td>
</tr>
<tr>
<td>Title</td>
<td>Number</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------------------</td>
<td>-----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| z/OS Communications Server: IP Configuration Reference     | SC27-3651 | This document presents information for people who want to administer and maintain IP. Use this document with the z/OS Communications Server: IP Configuration Guide. The information in this document includes:  
  • TCP/IP configuration data sets  
  • Configuration statements  
  • Translation tables  
  • Protocol number and port assignments               |
| z/OS Communications Server: SNA Network Implementation Guide | SC27-3672 | This document presents the major concepts involved in implementing an SNA network. Use this document with the z/OS Communications Server: SNA Resource Definition Reference. |
| z/OS Communications Server: SNA Resource Definition Reference | SC27-3675 | This document describes each SNA definition statement, start option, and macroinstruction for user tables. It also describes NCP definition statements that affect SNA. Use this document with the z/OS Communications Server: SNA Network Implementation Guide. |
| z/OS Communications Server: SNA Resource Definition Samples | SC27-3676 | This document contains sample definitions to help you implement SNA functions in your networks, and includes sample major node definitions. |
| z/OS Communications Server: IP Network Print Facility      | SC27-3658 | This document is for systems programmers and network administrators who need to prepare their network to route SNA, JES2, or JES3 printer output to remote printers using TCP/IP Services. |

**Operation**

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP User's Guide and Commands</td>
<td>SC27-3662</td>
<td>This document describes how to use TCP/IP applications. It contains requests with which a user can log on to a remote host using Telnet, transfer data sets using FTP, send electronic mail, print on remote printers, and authenticate network users.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP System Administrator's Commands</td>
<td>SC27-3661</td>
<td>This document describes the functions and commands helpful in configuring or monitoring your system. It contains system administrator's commands, such as TSO NETSTAT, PING, TRACERTE and their UNIX counterparts. It also includes TSO and MVS commands commonly used during the IP configuration process.</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Operation</td>
<td>SC27-3673</td>
<td>This document serves as a reference for programmers and operators requiring detailed information about specific operator commands.</td>
</tr>
<tr>
<td>z/OS Communications Server: Quick Reference</td>
<td>SC27-3665</td>
<td>This document contains essential information about SNA and IP commands.</td>
</tr>
<tr>
<td>Title</td>
<td>Number</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>z/OS Communications Server:</td>
<td>SC27-3666</td>
<td>This document enables you to customize SNA, and includes the following information:</td>
</tr>
<tr>
<td>SNA Customization</td>
<td></td>
<td>• Communication network management (CNM) routing table</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Logon-interpret routine requirements</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Logon manager installation-wide exit routine for the CLU search exit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TSO/SNA installation-wide exit routines</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SNA installation-wide exit routines</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server:</td>
<td>SC27-3660</td>
<td>This document describes the syntax and semantics of program source code necessary to write your own application programming interface (API) into TCP/IP. You can use this interface as the communication base for writing your own client or server application. You can also use this document to adapt your existing applications to communicate with each other using sockets over TCP/IP.</td>
</tr>
<tr>
<td>IP Sockets Application Programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interface Guide and Reference</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>z/OS Communications Server:</td>
</tr>
<tr>
<td></td>
<td>SC27-3649</td>
<td>This document is for programmers who want to set up, write application programs for, and diagnose problems with the socket interface for CICS using z/OS TCP/IP.</td>
</tr>
<tr>
<td>IP CICS Sockets Guide</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SC27-3653</td>
<td>This document is for programmers who want application programs that use the IMS TCP/IP application development services provided by the TCP/IP Services of IBM.</td>
</tr>
<tr>
<td></td>
<td>SC27-3659</td>
<td>This document describes the syntax and semantics of a set of high-level application functions that you can use to program your own applications in a TCP/IP environment. These functions provide support for application facilities, such as user authentication, distributed databases, distributed processing, network management, and device sharing. Familiarity with the z/OS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.</td>
</tr>
<tr>
<td></td>
<td>SC27-3674</td>
<td>This document describes how to use SNA macroinstructions to send data to and receive data from (1) a terminal in either the same or a different domain, or (2) another application program in either the same or a different domain.</td>
</tr>
<tr>
<td></td>
<td>SC27-3669</td>
<td>This document describes how to use the SNA LU 6.2 application programming interface for host application programs. This document applies to programs that use only LU 6.2 sessions or that use LU 6.2 sessions along with other session types. (Only LU 6.2 sessions are covered in this document.)</td>
</tr>
</tbody>
</table>
### Title

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: SNA Programmer's LU 6.2 Reference</td>
<td>SC27-3670</td>
<td>This document provides reference material for the SNA LU 6.2 programming interface for host application programs.</td>
</tr>
<tr>
<td>z/OS Communications Server: CSM Guide</td>
<td>SC27-3647</td>
<td>This document describes how applications use the communications storage manager.</td>
</tr>
</tbody>
</table>

### Diagnosis

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP Diagnosis Guide</td>
<td>GC27-3652</td>
<td>This document explains how to diagnose TCP/IP problems and how to determine whether a specific problem is in the TCP/IP product code. It explains how to gather information for and describe problems to the IBM Software Support Center.</td>
</tr>
<tr>
<td>z/OS Communications Server: ACF/TAP Trace Analysis Handbook</td>
<td>GC27-3645</td>
<td>This document explains how to gather the trace data that is collected and stored in the host processor. It also explains how to use the Advanced Communications Function/Trace Analysis Program (ACF/TAP) service aid to produce reports for analyzing the trace data information.</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures and z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT</td>
<td>GC27-3667  GC27-3668</td>
<td>These documents help you identify an SNA problem, classify it, and collect information about it before you call the IBM Support Center. The information collected includes traces, dumps, and other problem documentation.</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Data Areas Volume 1 and SNA Data Areas Volume 2</td>
<td>GC31-6852  GC31-6853</td>
<td>These documents describe SNA data areas and can be used to read an SNA dump. They are intended for IBM programming service representatives and customer personnel who are diagnosing problems with SNA.</td>
</tr>
</tbody>
</table>

### Messages and codes

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: SNA Messages</td>
<td>SC27-3671</td>
<td>This document describes the ELM, IKT, IST, IUT, IVT, and USS messages. Other information in this document includes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Command and RU types in SNA messages</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Node and ID types in SNA messages</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Supplemental message-related information</td>
</tr>
<tr>
<td>Title</td>
<td>Number</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>---------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Messages Volume 1 (EZA)</td>
<td>SC27-3654</td>
<td>This volume contains TCP/IP messages beginning with EZA.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Messages Volume 2 (EZB, EZD)</td>
<td>SC27-3655</td>
<td>This volume contains TCP/IP messages beginning with EZB or EZD.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Messages Volume 3 (EZY)</td>
<td>SC27-3656</td>
<td>This volume contains TCP/IP messages beginning with EZY.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM)</td>
<td>SC27-3657</td>
<td>This volume contains TCP/IP messages beginning with EZZ and SNM.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP and SNA Codes</td>
<td>SC27-3648</td>
<td>This document describes codes and other information that appear in z/OS Communications Server messages.</td>
</tr>
</tbody>
</table>
Index

Special Characters

**** (FFST) trace record 173

Numerics

3270 trace record 718
3271 trace record 719

A

a customized dump
FFST/MVS 54
a formatted dump 59
abend (abnormal end) trace records
abend RELSTORE 244
ABND SNAP 177
ABND trace record 177
ACA1 trace record 177
ACA2 trace record 181
ACA3 trace record 182
accessibility 793
ACI1 trace record 177
ACI2 trace record 181
ACI3 trace record 182
ACP1 trace record 182
ACP2 trace record 186
ACR1 trace record 182
ACR2 trace record 186
ACRC trace record 188
ACSN trace record 190
ACU1 trace record 191
ACU2 trace record 192
ADE trace record 192
AFSM trace record 196
AI1 trace record 198
AI2 trace record 200
AI3 trace record 201
ALS trace record 202
AP trace record 204
API option
summary 18
VIT trace records created
AI1 198
AI2 200
AI3 201
IO1 441
IO2 (MVS) 443
IO3 (MVS) 444
RE 571
UE 174, 658, 659
UE1 660
UE2 661
UP 664

APPC option (continued)
VIT trace records created (continued)
ACA1 177
ACA2 181
ACA3 182
ACI1 177
ACI2 181
ACI3 182
ACP1 182
ACP2 186
ACR1 182
ACR2 186
ACU1 191
ACU2 191
RACR 559
REML 574
REMQ 575
USI 666
USI2 668
USO 666
USO2 668
UVI 668
UVO 668

APPC sense code 5
ARB trace record 206
AREL trace record 244
ARP trace record 245
ARP2 trace record 247
ARQ trace record 248
ARQ2 trace record 251
ARS trace record 245
ARS2 trace record 247
ASN2 trace record 253
ASN6 trace record 254
ASN1 trace record 252
ATCASLIP 5
ATSK trace record 255
ATT (attention) trace record 256
attached device support
S/390 channel 158
authorized IO trace record 198
AXIT trace record 260

B

BSPx trace record 257
BTSK trace record 261
BUFF trace record 262
buffer pool expansion, EXPN trace 378

C

C642 trace record 264
C64Q trace record 263
CCI or CCO trace record
NCSPL 267
not RUPE or NCSPL 268
CCI or CCO trace record (continued)
RUPE 265
CDHx trace record 271
CDN trace record 273
CDN2 trace record 274
CFA2 trace record 277
CFAL trace record 276
CFC2 trace record 280
CFCE trace record 278
CFCN trace record 279
CFDS trace record 281, 282
CFEE trace record 283
CFER trace record 288
CFFC trace record 289
CFL2 trace record 291, 292, 294–296, 298
CFL3 trace record 293, 296, 297
CFLS trace record 289
CFNF trace record 299
CFP2 trace record 301
CFPG trace record 300
CFRB trace record 302
CFS option
summary 18
VIT trace records created
CFC 280
CFCE 278
CFCN 279
CFD 282
CFDS 281
CFEE 283
CFER 288
CFFC 289
CFL2 291, 292, 294–296, 298
CFL3 293, 296, 297
CFLS 289
CFNF 299
CFRB 302
CFUS 308
CFT2 trace record 306
CFTP trace record 303
CFTX trace record 306
CFUS trace record 308
CFVC trace record 309
CHG2 trace record 312
CHG6 trace record 312
CHGO trace record 310
CI1 trace record 314
CI2 trace record 318
CI3 trace record 320
CI4 trace record 320
CICS
dump global exit 154
CIO option
summary 18
VIT trace records created
ATT 256
ENFF 370
ENFN 370
ENFP 370
ENFR 370
ERPI 373
ERPL 373
ERPX 373
HIO 422
CIO option (continued)
VIT trace records created (continued)
INT 440
LNIKA 483
LNIKI 483
LNKL 483
LNKx 483
SIO 2 620
XID 715
XID 716
CME2 trace record 322
CMER trace record 321
CMIP option
summary 18
VIT trace records created
MCO 1 488
MCO 2 489
MDET 490
MSDS 491
MQRQ 497
MQR 497
MREG 498
MRG 499
MRG 2 501
RQE 591
CNTP trace record 331
C01 trace record 314
CO2 trace record 318
CO3 trace record 320
CO4 trace record 320
Communications Server for z/OS, online information xxv
console message
using 72
CONT trace record 332
Coupling facility services probes 753
CPC trace record 721, 729
CPMSG trace record 502
CPPT trace record 334
CPR2 trace record 339
CPR3 trace record 340
CPRC trace record 338
CPW2 trace record 341
CPW3 trace record 342
CPW4 trace record 342
CPWAIT macro 341, 343
CPWT trace record 340
CPY2 trace record 344
CPY3 trace record 345
CPY4 trace record 346
CPY6 trace record 347
CPYB trace record 343
CRA trace record 348
CRD trace record 348
CRF trace record 348
CSB trace record 350
CSB2 trace record 351
CSC trace record 350
CSC2 trace record 351
CSD trace record 350
CSD2 trace record 351
CSDUMP 168
customized dump
MVS 29
F

FBL2 trace record 382
FBLK trace record 380
FFST
  commands 31
  components 28
  controls 159
  function 30
  messages 80
  MVS transition code function 152
  operation 30
  operations 27
  output 53
  outputs 29
  overview 27
  probe control 159
  probe return codes 77
  starting 36
  stopping 52
  usage 30
  FFST commands
    a console 36
  FFST functions
    disabling 36
    enabling 40
  FFST MODIFY command 32
  FFST START command 31
  FFST STOP command 36
  FFST/MVS symptom records
    using 68
  First failure support technology (FFST)
    **** trace record 173
      console 4
      dumps 1
      generic alert 3
      minidump 1
      probes 747
      problems 6
      sample minidump 2
      trap module 4
      traps 5
  FIX2 trace record 385
  FIX6 trace record 386
  FIXB trace record 384
  FRB2 trace record 388
  FRB6 trace record 389
  FRBF trace record 387
  FREEBLK trace record 380
  FRES trace record 390

G

GBL2 trace record 395
GBL3 trace record 395
GBLK trace record 393
GCEA trace record 398
GCEL trace record 397
GCER trace record 398
GCEx trace record 398
  generic alert
    using 73
  GETBLK trace record 393
  GETS trace record 400
GNA2 trace record 405
GNAM trace record 401
GTB2 trace record 409
GTB3 trace record 410
GTB6 trace record 411
GTBF trace record 407

H
HALT I/O trace record 422
HCL2 trace record 413
HCLK trace record 412
HCR trace record 418
HCR2 trace record 419
HCR3 trace record 420
HCR4 trace record 421
HCR5 trace record 422
HIO trace record 422
HIOX trace record 422
HPR option, VIT trace records created
ARB 206
HCLK 412
HPR 426
HPR2 428
HPR3 430
HPR4 431
HPRT 432
NLP2 515
NLP1 514
NLPO 514
RCM 563
RTP 596
RTPE 601
RTS2 606
RTSI 605
RTSO 605
summary 18
HPR trace record 426
HPR2 trace record 428
HPR3 trace record 430
HPR4 trace record 431
HPRT trace record 432
HST trace record 723

I
IBM Software Support Center, contacting xxi
ICR trace record 434
ICR2 trace record 435
ICR3 trace record 435
IDX2 trace record 437
IDXI trace record 436
IDXO trace record 436
Information APARs xxiii
INOP trace record 437
INT2 trace records 440
internal trace table location, VTAM
in a dump 23
in storage 18
Internet, finding z/OS information online xxv
INTX trace record 439
IO1 trace record 441
IO2 trace record, MVS 443
IO3 trace record, MVS 444
IOS2 trace record 446
IOS3 trace record 446
IOSP trace record 445
IPA2 trace record 449
IPAD trace record 447
IPG2 trace record 450
IPG3 trace record 451
IPGN trace record 449
IPLA trace record 452
IPLE trace record 451
IPO2 trace record 454
IPOG trace record 454
IPTC trace record 455
IPTM trace record 456
IRBD trace record 455, 456, 458
IRBX trace record 459
ISP2 trace record 461
ISP3 trace record 461
ISPx trace record 460
ISTCMCxx FFST probes 750
ISTCOCxx FFST probes 752
ISTCSCxx FFST probes 750
ISTDLCxx FFST probes 759
ISTDRCxx FFST probes 760
ISTFSCxx FFST probes 753
ISTIPxx FFST probes 767
ISTMCxx FFST probes 769
ISTNACxx FFST probes 770
ISTNSCxx FFST probes 769
ISTORCxx FFST probes 781
ISTORFBA 171
ISTRACxx FFST probes 782
ISTRMCxx FFST probes 769
ISTSCCxx FFST probes 772
ISTSLCxx FFST probes 774
ISTSSCxx FFST probes 772
ISTTACxx FFST probes 784
ISTTRCxx FFST probes 781
ISTTSCxx FFST probes 783
IUT2 trace record 468, 528
IUT3 trace record 470
IUTC, IUTI, IUTM, IUTQ, IUTS trace records 462
IUTD trace record 464
IUTF trace record 465
IUTR trace record 466
IUTx trace record 462
IUTX trace record 466

K
keyboard 793

L
LCS option
summary 18
VIT trace records created
LCS2 478
LCSL 471
LCMS 478
LCSP 471
LCSS 471
LCS option (continued)
VIT trace records created (continued)
LCSX 471
LCS2 trace record 478
LCSL trace record 471
LCSM trace record 478
LCSP trace record 471
LCSS trace record 471
LCSX trace record 471
LDL2 trace record 481
LDLC trace record 479
license, patent, and copyright information 795
LKEX trace record 481
LKSH trace record 482
LNKA trace record 483
LNKI trace record 483
LNKL trace record 483
LNKx trace record 483
LOCK option
  summary 18
  VIT trace records created
    LKEK 481
    LKSH 482
    ULKA 662
    UNLK 663
LOST trace record 485
LSN2 trace record 487
LSNA trace record 486
LU 6.2 message unit 506, 509

M
mainframe
  education xxiii
MCO1 trace record 488
MCO2 trace record 489
MDEL trace record 490
MDIS trace record 491
message log 30
message logs
  clearing 45
Message Sent trace record 504
messages 29
MM2 trace record 493
MMG trace record 492
MNPS trace record 493
MOD trace record 724, 732
MODIFY CSDUMP 168
module name, finding in an internal trace (VIT) 23
MPD2 trace record 497
MPDU trace record 495
MQRQ trace record 497
MQRS trace record 497
MREG trace record 498
MRG trace record 499
MRG2 trace record 501
MSG option
  summary 18
  VIT trace records created
    MSG 502
    MSG2 505
    MSGS 504
    OPE2 528
    OPER 527
    MSG trace record 502
    MSG2 trace record 505
    MSGS trace record 504
    MT trace record 505
    MU1 trace record 506
    MU2 trace record 509
    MU3 trace record 510
    MU4 trace record 511
    MVS
dump frequency threshold support 156
  post dump exit 152

N
NIP2 513
NIPA 512
NIPO 512
NIPI 512
NIPN 512
NLP2 trace record 515
NLP1 trace record 514
NLPO trace record 514
NRM option
  summary 18
  VIT trace records created
    BSPx 257
    BSXX 259
    BSXx 260
    PROA 546
    PROD 546
    RCEx 562
    SRT2 631
    SRTA 630
    SRTC 630
    SRTD 630
    SRTF 630
    SRTx 630
  NRS2 trace record 516
  NRSP trace record 516
  NSD trace record 517

O
ODP2 trace record 520
ODPK trace record 518
ODTE trace record 520
ONLP trace record 522
OON trace record 523
OON2 trace record
  OON2 523
  OOS2 trace record 525
  OOSQ trace record 524
  OOX trace record 524–526
  OPE2 trace record 528
  OPER trace record 527
  ORMG trace record 529
OSA-Express, VIT trace records created
  HIOx 422
  INTx 439
  IUTD 464
  IUTF 465
  IUTR 466
  IUTX 466
OSA-Express, VIT trace records created (continued)

ODP2 520
ODPK 518
ODTE 520
PLOQ 542
SBA2 612
SBAL 610
SIGA 615
SIOx 616
SLS2 622
SLSB 621
XBI1 695
XBI2 696
XBI3 697

output destinations
changing 49
resetting 50

P

P642 trace record 532
P64Q trace record 530
PAB DISPATCH trace record 367
PAG2 trace record 534
PAG6 trace record 535
PAGB trace record 532
Parameter list control 159
Parameter list control logic 162
PCID trace record 536
PCIR trace record 537
PCIX trace record 536
PFFST 174
PIU option
summary 18
VIT trace records created
DSC2 367
DSCD 366
NRS2 516
NRSP 516
PIU 539
PIU2 540
PIU trace record 539
PIU2 trace record 540
PKI or PKO trace record 541
PKI2 or PKO2 trace record 542
POOF trace record 544
POST trace record 545
prerequisite information xxiii
PROA trace record 546
probe message log entry 72
PROD trace record 546
product identification information
Specifying 156
profile add 546
profile delete 546
PSS option
summary 18
VIT trace records created
ATSK 255
BTSK 261
DSP 367
DTSK 369
ETSK 376
EXIT 376

PSS option (continued)

VIT trace records created (continued)
IRBD 455, 456, 458
IRBX 459
POST 545
QUE 555
QUEN 558
RESM 579
SCHD 613
SRBD 628
SRBX 629
VPST 685
VRSN 686
VWAI 689
WAIT 690
XPST 716

Q

QAPL trace record 547
QDIP trace record 549
QRE2 trace record 551
QREQ trace record 550
QRYL trace record 552
QRSB trace record 553
QUE trace record 555
QUEN trace record 558
query language VIT entry 552
QUEUED REQSTORE trace record 577

R

RACR trace record 559
RAP2 trace record 561
RAPB trace record 560
RCM trace record 562
RCE trace record 563
RCR trace record 564
RPC2 trace record 565
RCP3 trace record 566
RCPx trace record 564
RCV trace record 566
RDS2 trace record 569
RDSC trace record 568
RE trace record 571
record descriptions 536
RELS trace record 573
REML trace record 574
REMQ trace record 575
REQ2 trace record 578
REQS trace record 577
RESM trace record 579
return codes, RPL6 5
RFC (request for comments)
accessing online xxv
RIO2 trace record 582
RIOX trace record 581
RPL EXIT trace record 571
RPL6 return code 5
RPLA trace record 584
RPLI trace record 586
RPLP trace record 583
RPLx trace record 582
RPS2 trace record 589
RPSA trace record 588
RPSI trace record 590
RPSP trace record 587
RPST trace record 586
RQE trace record 591
RSC2 trace record 594
RSCX trace record 592
RSLK trace record 594
RTP trace record 596
RTP2 trace record 598, 600
RTPE trace record 601
RTPP trace record 603
RTS2 trace record 606
RTSI trace record 605
RTSO trace record 605
RTVM trace record 606
RVM trace record 606
RXMT trace record 609

S

SCHD trace record 613
SDUMP trace record 167
sense codes, APPC 5
shortcut keys 793
SIO2 trace record 620
SIOD, SIOT trace records 617
SIOI trace record 618
SIOL trace record 619
SIOL trace record 616
SMS option
summary 18
VIT trace records created
AREL 244
EXPN 378
FBL2 382
FBLK 380
GBL2 395
GBLK 393
POOF 544
QREQ 550
RELS 573
REQ2 578
REQS 577
VTAL 687
VTFR 688

SNAP
abend 177
trace record, VIT 20
softcopy information xxiii
SPT trace record 624
SPT2 trace record 627
SRBD trace record 628
SRBX trace record 629
SRT2 trace record 631
SRTA trace record 630
SRTC trace record 630
SRTD trace record 630
SRTF trace record 630
SRTx trace record 630
SSCP option
summary 18
VIT trace records created

SSCP option (continued)
VIT trace records created (continued)
A2 for CP 206
ACRC 188
ACSN 190
ALS 202
AP for CP 204
CC2 for NCSPN 268
CC2 for RUPE 266
CCI for NCSPN 267
CCI for RUPE 265
CCI, not RUPE or NCSPN 268
CCO for NCSPN 267
CCO for RUPE 265
CCO, not RUPE or NCSPN 268
CI1 314
C12 318
C13 320
C14 320
C01 314
C02 318
C03 320
C04 320
CPP2 336
CPP3 337
CPP4 337
CPPG 334
CPPT 334
CPR2 339
CPR3 340
CPRC 338
CPW2 341
CPW3 342
CPW4 342
CPWT 340
CRA add 348
CRD delete 348
CRF find 348
CSB2 351
CSC2 351
CSD2 351
CSD 350
BDB delete 351, 353
DBQ query 351, 353
DBU update 351, 353
DLT 354, 356, 357
DLT2 361
MT 505
TGM 632
TGM2 634
TREB 644
TRED 644
TREI 644
TREM 644
startup command list 53
status 46
summary of changes xxix–xxxii
Summary of changes xxxiii
symptom record
using 68
symptom string
using 66
T

tasks
  activating an APPC sense code trap 5
  using the trap module 4
TCP option
  summary 18
TCP/IP
  online information xxv
TDU trace record 725, 733
Tech notes xxiii
TGM trace record 632
TGM2 trace record 634
TGREC trace record descriptions 729
TGU trace record 726, 735
TGV2 trace record 636
TGVC trace record 635
TOD trace record 636
TOKX trace record 635–637
TOP trace record 727, 737
TPESC trace record 375
TPEXIT trace record 376
TPIO trace record 441
TPLOCK EXCLUSIVE trace record 481
TPLOCK SHARED trace record 482
TPMSG trace record 502
TPPOST trace record 545
TPQUE NONE trace record 558
TPQUE trace record 555
TPSCHED trace record 613
TPSWPST trace record 716
TPUNLOCK ALL trace record 662
TPUNLOCK trace record 663
trace entries, LAN channel station control 471
trace records, error recovery procedures 373
traces
  internal (VIT) 7
  table header record (VIT) 19
trademark information 798
transition code parameter list 157
translate message VIT entry 650
trap module, using the 4
traps 5
TREB trace record 644
TRED trace record 644
TREI trace record 644
TREM trace record 644
TRN2 trace record 652
TRN3 trace record 650
TRS common topology trace record descriptions 741
TRS common topology trace table 738
TSNS trace record 658

U

UE trace record 174, 658, 659
UE1 trace record 660
UE2 trace record 661
ULKA trace record 662
UNLK trace record 663
UP trace record 664
USER EXIT trace record 174, 658, 659
USER POST trace record 664
USI trace record 666
USI2 trace record 668
USO trace record 666
USO2 trace record 668
UV1 trace record 668
UV12 trace record 671
UVO trace record 668
UVO2 trace record 671

V

VCC1 trace record 672
VCC2 trace record 674
VCDQ trace record 675
VCNS option
  summary 18
  VIT trace records created
    CNA 323
    CNP1 326
    CNP2 329
    CNP3 330
    CNR1 326
    CNR2 329
    CNR3 330
    NSD 517
    VCC1 672
    VCC2 674
    VCDQ 675
    VPST trace record 685
    VRSM trace record 686
    VTAL trace record 687
VTAM
  service aids 1, 6
  traces, internal 7
VTAM internal trace
  record descriptions
    DCO2 356
VTAM internal trace (VIT)
  activating 9
  deactivating 21–23
  module names in trace records 23
  options (OPTION operand) 11
  record descriptions
    3270 718
    3271 719
    A2 206
    ABND 177
    ACA1 177
    ACA2 181
    ACA3 182
    AC1 177
    AC2 181
    AC3 182
    ACP1 182
    ACP2 186
    ACR1 182
    ACR2 186
    ACRC 188
    ACSN 190
    ACU1 191
    ACU2 191
    ADE 192
    AFSM 196
    AI1 198
VTAM internal trace (VIT) (continued)
record descriptions (continued)

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A12</td>
<td>200</td>
<td>CNA</td>
<td>323</td>
</tr>
<tr>
<td>A13</td>
<td>201</td>
<td>CNP1</td>
<td>326</td>
</tr>
<tr>
<td>ALS</td>
<td>202</td>
<td>CNP2</td>
<td>329</td>
</tr>
<tr>
<td>AP</td>
<td>204</td>
<td>CNP3</td>
<td>330</td>
</tr>
<tr>
<td>ARB</td>
<td>206</td>
<td>CNR1</td>
<td>326</td>
</tr>
<tr>
<td>AREL</td>
<td>244</td>
<td>CNR2</td>
<td>329</td>
</tr>
<tr>
<td>ARP</td>
<td>245</td>
<td>CNR3</td>
<td>330</td>
</tr>
<tr>
<td>ARP2</td>
<td>247</td>
<td>CNTP</td>
<td>331</td>
</tr>
<tr>
<td>ARQ</td>
<td>248</td>
<td>CO1</td>
<td>314</td>
</tr>
<tr>
<td>ARQ2</td>
<td>251</td>
<td>CO2</td>
<td>318</td>
</tr>
<tr>
<td>ARS</td>
<td>245</td>
<td>CO3</td>
<td>320</td>
</tr>
<tr>
<td>ARS2</td>
<td>247</td>
<td>CO4</td>
<td>320</td>
</tr>
<tr>
<td>ASN</td>
<td>253</td>
<td>CONT</td>
<td>332</td>
</tr>
<tr>
<td>ASN6</td>
<td>254</td>
<td>CPP2</td>
<td>336</td>
</tr>
<tr>
<td>ASNB</td>
<td>252</td>
<td>CPP3</td>
<td>337</td>
</tr>
<tr>
<td>ATSK</td>
<td>255</td>
<td>CPP4</td>
<td>337</td>
</tr>
<tr>
<td>ATT</td>
<td>256</td>
<td>CPPG</td>
<td>334</td>
</tr>
<tr>
<td>BSPx</td>
<td>257</td>
<td>CPPT</td>
<td>334</td>
</tr>
<tr>
<td>BSS</td>
<td>259</td>
<td>CPR2</td>
<td>339</td>
</tr>
<tr>
<td>BSX</td>
<td>260</td>
<td>CPR3</td>
<td>340</td>
</tr>
<tr>
<td>BTKS</td>
<td>261</td>
<td>CPRC</td>
<td>338</td>
</tr>
<tr>
<td>BUFF</td>
<td>262</td>
<td>CPW2</td>
<td>341</td>
</tr>
<tr>
<td>C642</td>
<td>264</td>
<td>CPW3</td>
<td>342</td>
</tr>
<tr>
<td>C64Q</td>
<td>263</td>
<td>CPW4</td>
<td>342</td>
</tr>
<tr>
<td>CC2</td>
<td>for NCSPL 268</td>
<td>CPW</td>
<td>340</td>
</tr>
<tr>
<td>CC2</td>
<td>for RUPE 266</td>
<td>CY2</td>
<td>344</td>
</tr>
<tr>
<td>CCI</td>
<td>for NCSPL 267</td>
<td>CPY3</td>
<td>345</td>
</tr>
<tr>
<td>CCI</td>
<td>for neither RUPE nor NCSPL 268</td>
<td>CPY4</td>
<td>346</td>
</tr>
<tr>
<td>CCI</td>
<td>for RUPE 265</td>
<td>CPY6</td>
<td>347</td>
</tr>
<tr>
<td>CCO</td>
<td>for NCSPL 267</td>
<td>CPYB</td>
<td>343</td>
</tr>
<tr>
<td>CCO</td>
<td>for neither RUPE nor NCSPL 268</td>
<td>CRA</td>
<td>add 348</td>
</tr>
<tr>
<td>CCO</td>
<td>for RUPE 265</td>
<td>CRD</td>
<td>delete 348</td>
</tr>
<tr>
<td>CDHX</td>
<td>271</td>
<td>CRF</td>
<td>find 348</td>
</tr>
<tr>
<td>CDN</td>
<td>273</td>
<td>CSB</td>
<td>351</td>
</tr>
<tr>
<td>CFA2</td>
<td>277</td>
<td>CSC</td>
<td>351</td>
</tr>
<tr>
<td>CFAL</td>
<td>276</td>
<td>CSD</td>
<td>351</td>
</tr>
<tr>
<td>CFCC</td>
<td>280</td>
<td>CSx</td>
<td>350</td>
</tr>
<tr>
<td>CFCE</td>
<td>278</td>
<td>DAPT</td>
<td>351</td>
</tr>
<tr>
<td>CFCN</td>
<td>279</td>
<td>DBD</td>
<td>delete 351, 353</td>
</tr>
<tr>
<td>CFD2</td>
<td>282</td>
<td>DBQ</td>
<td>query 351, 353</td>
</tr>
<tr>
<td>CFD2</td>
<td>289</td>
<td>DBU</td>
<td>update 351, 353</td>
</tr>
<tr>
<td>CFE2</td>
<td>283</td>
<td>DEVX</td>
<td>356</td>
</tr>
<tr>
<td>CFER</td>
<td>288</td>
<td>DLT</td>
<td>354, 356, 357</td>
</tr>
<tr>
<td>CFFC</td>
<td>289</td>
<td>DLT2</td>
<td>361</td>
</tr>
<tr>
<td>CFL2</td>
<td>291, 292, 294–296, 298</td>
<td>DRB</td>
<td>362</td>
</tr>
<tr>
<td>CFL3</td>
<td>293, 296, 297</td>
<td>DRR</td>
<td>365</td>
</tr>
<tr>
<td>CFLS</td>
<td>289</td>
<td>DRP</td>
<td>366</td>
</tr>
<tr>
<td>CFNF</td>
<td>299</td>
<td>DRPA</td>
<td>363</td>
</tr>
<tr>
<td>CFP</td>
<td>301</td>
<td>DRPC</td>
<td>363</td>
</tr>
<tr>
<td>CFPG</td>
<td>300</td>
<td>DRPD</td>
<td>363</td>
</tr>
<tr>
<td>CFRB</td>
<td>302</td>
<td>DSC</td>
<td>367</td>
</tr>
<tr>
<td>CFUS</td>
<td>308</td>
<td>DSCD</td>
<td>366</td>
</tr>
<tr>
<td>CFVC</td>
<td>309</td>
<td>DSCX</td>
<td>366</td>
</tr>
<tr>
<td>CHG</td>
<td>312</td>
<td>DSP</td>
<td>367</td>
</tr>
<tr>
<td>CHG6</td>
<td>312</td>
<td>DTSK</td>
<td>369</td>
</tr>
<tr>
<td>CHGO</td>
<td>310</td>
<td>ENFF</td>
<td>370</td>
</tr>
<tr>
<td>CI1</td>
<td>314</td>
<td>ENFN</td>
<td>370</td>
</tr>
<tr>
<td>CI2</td>
<td>318</td>
<td>ENFP</td>
<td>370</td>
</tr>
<tr>
<td>CI3</td>
<td>320</td>
<td>ENFR</td>
<td>370</td>
</tr>
<tr>
<td>CI4</td>
<td>320</td>
<td>ERPI</td>
<td>373</td>
</tr>
<tr>
<td>CME2</td>
<td>322</td>
<td>ERP</td>
<td>373</td>
</tr>
<tr>
<td>CMER</td>
<td>321</td>
<td>ERPX</td>
<td>373</td>
</tr>
<tr>
<td>VTAM internal trace (VIT) (continued)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>record descriptions (continued)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| ESC 375 | ETSK 376 | EXIT 376 | EXPN 378 | EXPP 380 | FBL2 173, 382 | FBLK 380 | FFST (****) 173 |
| FIX2 385 | FIX6 386 | FIXB 384 | FRB2 388 | FRB6 389 | FRBF 388 | FRES 390 | GBL2 395 |
| GBLK 393 | GCEA 398 | GCEL 397 | GCER 398 | GCEx 398 | GETS 400 | GNA2 405 | GNAM 401 |
| GTB2 409 | GTB3 410 | GTB6 411 | GTBF 407 | HCLK 412 | HCR 418 | HCR2 419 | HCR3 420 |
| HCR4 421 | HCR5 422 | HIO 422 | HIOX 422 | HPR 426 | HPR2 428 | HPR3 430 | HPRT 432 |
| ICR 434 | ICR2 435 | ICR3 435 | IDX2 437 | IDXI 436 | IDXO 436 | INOP 437 | INT2 440 |
| INTX 439 | I01 441 | I02 for MVS 443 | I03 for MVS 444 | IOS2 446 | IOS3 446 | IOSP 445 | IPA2 449 |
| IPAD 447 | IPG2 450 | IPG3 451 | IPGN 449 | IPLA 452 | IPLE 451 | IPOST (****) 173 | IPOST (****) 173 |

<table>
<thead>
<tr>
<th>VTAM internal trace (VIT) (continued)</th>
</tr>
</thead>
<tbody>
<tr>
<td>record descriptions (continued)</td>
</tr>
</tbody>
</table>

| IP2 454 | IPOP 454 | IPTC 455 | IPTM 456 | IRBD 455, 456, 458 | IRBX 459 | ISP2 461 | ISP3 461 |
| ISPx 460 | IUT2 468, 528 | IUT3 470 | IUTC, IUTI, IUTM, IUTQ, IUTS 462 | IUTD 464 | IUTF 465 | IUTR 466 |
| IUTx 462 | IUTX 466 | LCS2 478 | LCSL 471 | LCSP 471 | LCSS 471 | LCSX 471 | LDL2 481 |
| LDLC 479 | LKEX 481 | LKSH 482 | LNKI 483 | LNKL 483 | LNS2 487 | LSN2 487 | LSNA 486 |
| MCO1 488 | MCO2 489 | MDIS 491 | MNPS 493 | MPRD 497 | MPRU 495 | MQRQ 497 | MQRS 497 |
| MREG 498 | MRG 499 | MRG2 501 | MSG 502 | MSG2 505 | MSGS 504 | MT 505 | MU1 506 |
| MU2 509 | MU3 510 | MU4 511 | NIP2 513 | NIPD 512 | NIPI 512 | NIPN 512 | NL2P 515 |
| NLPI 514 | NLPO 514 |
VTAM internal trace (VIT) (continued)

record descriptions (continued)

<table>
<thead>
<tr>
<th>Record Code</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>NRS2 516</td>
<td></td>
</tr>
<tr>
<td>NSD 517</td>
<td></td>
</tr>
<tr>
<td>ODP2 520</td>
<td></td>
</tr>
<tr>
<td>ODPK 518</td>
<td></td>
</tr>
<tr>
<td>ODTE 520</td>
<td></td>
</tr>
<tr>
<td>OON 523</td>
<td></td>
</tr>
<tr>
<td>OOS2 525</td>
<td></td>
</tr>
<tr>
<td>OOSQ 524</td>
<td></td>
</tr>
<tr>
<td>OOX 524–526</td>
<td></td>
</tr>
<tr>
<td>OPE2 528</td>
<td></td>
</tr>
<tr>
<td>OPER 527</td>
<td></td>
</tr>
<tr>
<td>ORMG 529</td>
<td></td>
</tr>
<tr>
<td>P642 532</td>
<td></td>
</tr>
<tr>
<td>P64Q 530</td>
<td></td>
</tr>
<tr>
<td>PAG2 534</td>
<td></td>
</tr>
<tr>
<td>PAG6 535</td>
<td></td>
</tr>
<tr>
<td>PAG8 532</td>
<td></td>
</tr>
<tr>
<td>PCIR 537</td>
<td></td>
</tr>
<tr>
<td>PCIX 536</td>
<td></td>
</tr>
<tr>
<td>PIU 539</td>
<td></td>
</tr>
<tr>
<td>PIU2 540</td>
<td></td>
</tr>
<tr>
<td>PIU2 540</td>
<td></td>
</tr>
<tr>
<td>PK1 or PKO 541</td>
<td></td>
</tr>
<tr>
<td>PK2 or PKO2 542</td>
<td></td>
</tr>
<tr>
<td>PLOQ 542</td>
<td></td>
</tr>
<tr>
<td>POOF 544</td>
<td></td>
</tr>
<tr>
<td>POST 545</td>
<td></td>
</tr>
<tr>
<td>PROA 546</td>
<td></td>
</tr>
<tr>
<td>PROD 546</td>
<td></td>
</tr>
<tr>
<td>QAPL 547</td>
<td></td>
</tr>
<tr>
<td>QDIP 549</td>
<td></td>
</tr>
<tr>
<td>QRE2 551</td>
<td></td>
</tr>
<tr>
<td>QREQ 550</td>
<td></td>
</tr>
<tr>
<td>QRYL 552</td>
<td></td>
</tr>
<tr>
<td>QSRB 553</td>
<td></td>
</tr>
<tr>
<td>QUE 555</td>
<td></td>
</tr>
<tr>
<td>QUEN 558</td>
<td></td>
</tr>
<tr>
<td>RACR 559</td>
<td></td>
</tr>
<tr>
<td>RAP2 561</td>
<td></td>
</tr>
<tr>
<td>RAPB 560</td>
<td></td>
</tr>
<tr>
<td>RCE 562</td>
<td></td>
</tr>
<tr>
<td>RCM 563</td>
<td></td>
</tr>
<tr>
<td>RCP2 565</td>
<td></td>
</tr>
<tr>
<td>RCP3 566</td>
<td></td>
</tr>
<tr>
<td>RCPx 564</td>
<td></td>
</tr>
<tr>
<td>RCV 566</td>
<td></td>
</tr>
<tr>
<td>RDS2 569</td>
<td></td>
</tr>
<tr>
<td>RDCS 568</td>
<td></td>
</tr>
<tr>
<td>RE 571</td>
<td></td>
</tr>
<tr>
<td>RELS 573</td>
<td></td>
</tr>
<tr>
<td>REML 574</td>
<td></td>
</tr>
<tr>
<td>REMQ 575</td>
<td></td>
</tr>
<tr>
<td>REQ2 578</td>
<td></td>
</tr>
<tr>
<td>REQS 577</td>
<td></td>
</tr>
<tr>
<td>RESM 579</td>
<td></td>
</tr>
<tr>
<td>RIO2 582</td>
<td></td>
</tr>
<tr>
<td>RIOX 581</td>
<td></td>
</tr>
<tr>
<td>RPLA 584</td>
<td></td>
</tr>
<tr>
<td>RPLI 586</td>
<td></td>
</tr>
<tr>
<td>RPLP 583</td>
<td></td>
</tr>
<tr>
<td>RPLx 582</td>
<td></td>
</tr>
<tr>
<td>RPS2 589</td>
<td></td>
</tr>
</tbody>
</table>

VTAM internal trace (VIT) (continued)

record descriptions (continued)

<table>
<thead>
<tr>
<th>Record Code</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPSA 588</td>
<td></td>
</tr>
<tr>
<td>RPSI 590</td>
<td></td>
</tr>
<tr>
<td>RPSL 587</td>
<td></td>
</tr>
<tr>
<td>RPSL 586</td>
<td></td>
</tr>
<tr>
<td>RQE 591</td>
<td></td>
</tr>
<tr>
<td>RSC2 594</td>
<td></td>
</tr>
<tr>
<td>RSCX 592</td>
<td></td>
</tr>
<tr>
<td>RSLK 594</td>
<td></td>
</tr>
<tr>
<td>RTP 596</td>
<td></td>
</tr>
<tr>
<td>RTP2 598, 600</td>
<td></td>
</tr>
<tr>
<td>RTPE 601</td>
<td></td>
</tr>
<tr>
<td>RTPP 603</td>
<td></td>
</tr>
<tr>
<td>RTS2 606</td>
<td></td>
</tr>
<tr>
<td>RTSI 605</td>
<td></td>
</tr>
<tr>
<td>RTSO 605</td>
<td></td>
</tr>
<tr>
<td>RVM 606</td>
<td></td>
</tr>
<tr>
<td>RXMT 609</td>
<td></td>
</tr>
<tr>
<td>SBA2 612</td>
<td></td>
</tr>
<tr>
<td>SBAL 610</td>
<td></td>
</tr>
<tr>
<td>SCHD 613</td>
<td></td>
</tr>
<tr>
<td>SIGA 615</td>
<td></td>
</tr>
<tr>
<td>SI02 620</td>
<td></td>
</tr>
<tr>
<td>SIOD, SIO2 617</td>
<td></td>
</tr>
<tr>
<td>SIOI 618</td>
<td></td>
</tr>
<tr>
<td>SIO1 619</td>
<td></td>
</tr>
<tr>
<td>SIOx 616</td>
<td></td>
</tr>
<tr>
<td>SLS2 622</td>
<td></td>
</tr>
<tr>
<td>SLSB 621</td>
<td></td>
</tr>
<tr>
<td>SPT 624</td>
<td></td>
</tr>
<tr>
<td>SPT2 627</td>
<td></td>
</tr>
<tr>
<td>SRBD 628</td>
<td></td>
</tr>
<tr>
<td>SRBX 629</td>
<td></td>
</tr>
<tr>
<td>SRT 631</td>
<td></td>
</tr>
<tr>
<td>SRTA 630</td>
<td></td>
</tr>
<tr>
<td>SRTC 630</td>
<td></td>
</tr>
<tr>
<td>SRTD 630</td>
<td></td>
</tr>
<tr>
<td>SRTF 630</td>
<td></td>
</tr>
<tr>
<td>SRTx 630</td>
<td></td>
</tr>
<tr>
<td>TGM 632</td>
<td></td>
</tr>
<tr>
<td>TGM2 634</td>
<td></td>
</tr>
<tr>
<td>TGV 636</td>
<td></td>
</tr>
<tr>
<td>TGV2 634</td>
<td></td>
</tr>
<tr>
<td>TGC 635</td>
<td></td>
</tr>
<tr>
<td>TOD 636</td>
<td></td>
</tr>
<tr>
<td>TOKX 635–637</td>
<td></td>
</tr>
<tr>
<td>TREB 644</td>
<td></td>
</tr>
<tr>
<td>TRED 644</td>
<td></td>
</tr>
<tr>
<td>TRE1 644</td>
<td></td>
</tr>
<tr>
<td>TREM 644</td>
<td></td>
</tr>
<tr>
<td>TRN 652</td>
<td></td>
</tr>
<tr>
<td>TRNM 650</td>
<td></td>
</tr>
<tr>
<td>TSNS 658</td>
<td></td>
</tr>
<tr>
<td>UE 174, 658, 659</td>
<td></td>
</tr>
<tr>
<td>UE1 660</td>
<td></td>
</tr>
<tr>
<td>UE2 661</td>
<td></td>
</tr>
<tr>
<td>ULLA 662</td>
<td></td>
</tr>
<tr>
<td>UNLK 663</td>
<td></td>
</tr>
<tr>
<td>UP 664</td>
<td></td>
</tr>
<tr>
<td>USI 666</td>
<td></td>
</tr>
<tr>
<td>USI2 668</td>
<td></td>
</tr>
<tr>
<td>USO 666</td>
<td></td>
</tr>
<tr>
<td>USO2 668</td>
<td></td>
</tr>
<tr>
<td>UVI 668</td>
<td></td>
</tr>
</tbody>
</table>
VTAM internal trace (VIT) (continued)
  record descriptions (continued)
  UVI2 671
  UVO 668
  UVO2 671
  VCC1 672
  VCC2 674
  VCDQ 675
  VPSN 685
  VRSM 686
  VTAL 687
  VTFR 688
  VWAI 689
  WAIT 690
  XBA1 692
  XBA2 693
  XBA3 694
  XBI1 695
  XBI2 696
  XBI3 697
  XCC2 702
  XCFJ 703
  XCFM 706
  XCFR 708
  XCFX 713
  XCL2 705
  XCM2 707
  XCR2 709
  XCS2 712
  XID 715
  XID2 716
  XPST 716

selecting internal or external recording
  MODE=EXT 20
  MODE=INT 18

SNAP trace record 20
trace table header record 19
using 7

VTAM, online information xxv
VTFR trace record 688
VWAI trace record 689

Z

z/OS Basic Skills Information Center xxiii
z/OS, documentation library listing 799

W

WAIT trace record 690

X

XBA1 trace record 692
XBA2 trace record 693
XBA3 trace record 694
XBI1 trace record 695
XBI2 trace record 696
XBI3 trace record 697
XCC2 trace record 702
XCFJ trace record 702
XCFM trace record 706
Communicating your comments to IBM

If you especially like or dislike anything about this document, you can send us comments electronically by using one of the following methods:

**Internet email:**
comsvrcf@us.ibm.com

**World Wide Web:**

If you would like a reply, be sure to include your name, address, and telephone number. Make sure to include the following information in your comment or note:

- Title and order number of this document
- Page number or topic related to your comment

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this document. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.