

Updates that apply to IBM® DB2® Analytics Accelerator Loader for z/OS® V2R1 User's Guide (SC27-6777-00)

Date of change: April 2017

Topic: Multiple

Change description: Documentation changes made in support of PTFs UI46554 and UI46555 APARs PI72331 and PI70677 – Enhancements for Accelerator Loader server

The following topics have been updated or introduced:

In chapter “Overview”:

Topic: “What’s new”

In chapter “Preparing to customize”:

Topic: “Worksheets: Gathering parameter values for Tools Customizer”

In chapter “Customizing DB2 Analytics Accelerator Loader”:

Topic: “Configuring access to data in IBM IMS databases”

Topic: “Configuring access to SMF data for IT Operational Analytics”

Topic: “Configuring access to System Management Facility (SMF) files”

Topic: “Configuring access to SYSLOG files”

Topic: “Configuring access to zFS files”

Topic: “Configuring access to distributed databases”

Topic: “Modifying the server configuration member”

Topic: “Configuring Server Event Facility rules for Microsoft SQL Server”

In chapter “Loading data from non-DB2, remote DB2, and remote system sources”:

Topic: “Creating virtual tables for IMS data”

Topic: “Creating virtual tables for zFS and HFS file system data”

Topic: “Accessing IT Operational Analytics data”

Topic: “Generating JCL”

In chapter “Syntax”:

Topic: “Example JCL”

Topic: “Syntax diagram: Load from a non-DB2, remote DB2, or remote system source”

Topic: “Syntax definitions: Load from a non-DB2, remote DB2, or remote system source”

In chapter “Administering the Accelerator Loader server”:

Topic: “Modifying IMS parameters”

Topic: “Configuring access to SMF files”

Topic: “IMS data access methods”

Topic: “Virtual table SAF security”

Topic: “Virtual Parallel Data”

Topic: “Configuring Virtual Parallel Data”

In chapter “Troubleshooting”:

Topic: “Messages and codes”

In chapter “Reference”:

Topic: “Accelerator Loader terminology”

Chapter “Overview”

Topic: “What’s new”

Add the following description:

- Accelerator Loader now supports accelerator groups, which allow users to load multiple accelerators by specifying a single accelerator group name. Users can specify either one group name or a list of accelerator names.
- Accelerator Loader now supports the new Virtual Parallel Data (VPD) feature, which lets you group multiple simultaneous requests against the same data source and run them in parallel, while performing the input and output only once. For example, using VPD, users can load several SMF record types from a virtualized data source with only a single read of the data set. Previously, the data set would have been read once for each record type.
- The default size of the global variable file that is used by the Accelerator Loader server has been increased.
- The following list highlights the enhancements to the Accelerator Loader server:
 - Distributed DRDA Data Servers (for example, DB2 LUW and DB2 Federation Servers) execute on servers that can support USERID values of various lengths. The Accelerator Loader server now supports Alternate Authentication USERID values up to 255 characters.
 - Through added MapReduce and parallelism support for accessing native IMS files, the Accelerator Loader server can now access IMS data directly (a feature named “IMS Direct”), as opposed to accessing the data through IMS DLI calls. This access method is similar to how the DB2 UNLOAD utility works and provides a significant increase in performance and reduced elapsed time.
 - The Accelerator Loader server can now call compression exits when reading IMS files with IMS Direct.
 - The Accelerator Loader server now supports SQL access to SMF stored in logstreams.
 - Support has been provided for a metadata repository in the server. This repository is used for MapReduce and parallelism exploitation of DRDA and IMS data sources by gathering metadata and persisting this information across server restarts. This support applies to all DRDA-backed data sources including those accessed using the IBM Federated Server, such as Terradata and Sybase, as well as data sources supported by direct DRDA support for the server, such as DB2 LUW and Oracle.
 - The Accelerator Loader server now implements SAF security in the SQL engine for virtual table access so multi-tenant environments are possible that limit both visibility and access to virtual tables between different tenants.
 - The Accelerator Loader server now provides enhanced distributed DRDA VRF support for Microsoft SQL Server using HIS 2016 DRDA AS.
 - The Accelerator Loader server now provides support for SQL access to zFS and HFS files.
 - The Accelerator Loader server now provides consistent uniqueness to parent and child keys across virtual table joins.

Chapter “Preparing to customize”

Topic: “Worksheets: Gathering parameter values for Tools Customizer”

In subtopic “Task: Create the server and the server components (required)”, the following rows have been added or updated in the table:

Step or parameter	Required?	Discovered?	Default value	Your value
IMS MODBLKS library Defines the IMS MODBLKS staging library that contains the control blocks to support online change of databases, programs, transactions, and MFS formats for the specified IMS subsystem. This data set enables the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	
IMS ACBLIB library Defines the IMS ACBLIB library that contains database and program descriptors for the specified IMS subsystem. This data set enables the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	
IMSDALIB library Defines the IMSDALIB library that contains the DFSMDA members that are used for dynamic allocation for the specified IMS subsystem. This data set enables the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	
RECON library Defines the first of two RECON libraries that contain system information for the specified IMS subsystem. This data set enables the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	
RECON2 library Defines the second of two RECON libraries that contain system information for the specified IMS subsystem. This data set enables the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	
RECON3 library Defines the spare RECON library for the specified IMS subsystem if the first two RECON files cannot be read. This data set enables the server to access IMS data directly.	No	No	No default	

Step or parameter	Required?	Discovered?	Default value	Your value
Global variable file primary allocation Specifies the primary allocation, in cylinders, for the global variable checkpoint data set. Approximately 1180 variables can fit in one cylinder.	Yes	No	25	
Global variable file secondary allocation Specifies the secondary allocation, in cylinders, for the global variable checkpoint data set. Approximately 1180 variables can fit in one cylinder.	Yes	No	5	

Chapter “Customizing DB2 Analytics Accelerator Loader”

Add the following topics, which are located at the end of this document:

Topic: “Configuring access to data in IBM IMS databases”
Topic: “Configuring access to SMF data for IT Operational Analytics”
Topic: “Configuring access to System Management Facility (SMF) files”
Topic: “Configuring access to SYSLOG files”
Topic: “Configuring access to zFS files”
Topic: “Configuring access to distributed databases”
Topic: “Modifying the server configuration member”
Topic: “Configuring Server Event Facility rules for Microsoft SQL Server”

Organize these topics as follows:

Configuring access to data in IBM IMS databases
Configuring access to SMF data for IT Operational Analytics
 Configuring access to System Management Facility (SMF) files
 Configuring access to SYSLOG files
Configuring access to zFS files
Configuring access to distributed databases
 Modifying the server configuration member
 Configuring Server Event Facility rules for Microsoft SQL Server

Chapter “Loading data from non-DB2, remote DB2, and remote system sources”

Update the topics as described below. Topics are located at the end of this document.

Replace the following topics with updated content:

Topic: “Creating virtual tables for IMS data”
Topic: “Generating JCL”

Add the following new topics:

Topic: “Creating virtual tables for zFS and HFS file system”
Topic: “Accessing IT Operational Analytics data”

Chapter “Syntax”

Under “Loading data from non-DB2, remote DB2, and remote system sources”, make following updates:

Topic: “Example JCL”

Add the following example:

Example: Load the accelerator with data using Virtual Parallel Data (VPD)

Virtual Parallel Data (VPD) allows you to group multiple simultaneous requests against the same data source and run them in parallel, while performing the input and output (I/O) only once. A separate Accelerator Loader job must be generated and submitted for each request, and these jobs must be run concurrently. When parallelism is used, each parallel thread joins the group separately and must join the group within a specified timeout value. Threads that do not appear within the timeout time are placed in a new group, resulting in an additional read of the data set.

To use Virtual Parallel Data (VPD) when loading data to the accelerator, use the following Accelerator Loader syntax options:

- ACCEL_HLV_VPD_GROUP
- ACCEL_HLV_VPD_MEMBERS
- ACCEL_HLV_VPD_TIMEOUT
- ACCEL_HLV_VPD_IOT

As an example, to process three different SMF record types in one pass through a data set, submit three Accelerator Loader jobs, one for each record type. The following sample control cards include the VPD syntax options for this example, which would need to be included in each of the jobs:

```
EXEC SQL DECLARE HLVC SR CURSOR FOR
      SELECT * FROM SMF_01400
ENDEXEC
```

```
LOAD DATA
  IDAA_ONLY ON DB9AAC1
  REPLACE
  LOG NO NOCOPYPEND
  ENFORCE NO
  ACCEL_CURSOR HLVC SR
  ACCEL_HLV_SSID HLVC
  ACCEL_HLV_VPD_GROUP TESTVPD
  ACCEL_HLV_VPD_MEMBERS 3
  ACCEL_HLV_VPD_TIMEOUT 300
  ACCEL_REMOVE_AND_ADD_TABLES
  ACCEL_ON_SUCCESS_ENABLE YES
  ACCEL_LOAD_TASKS 1
  INTO TABLE "USER1"."SMF_01400"
```

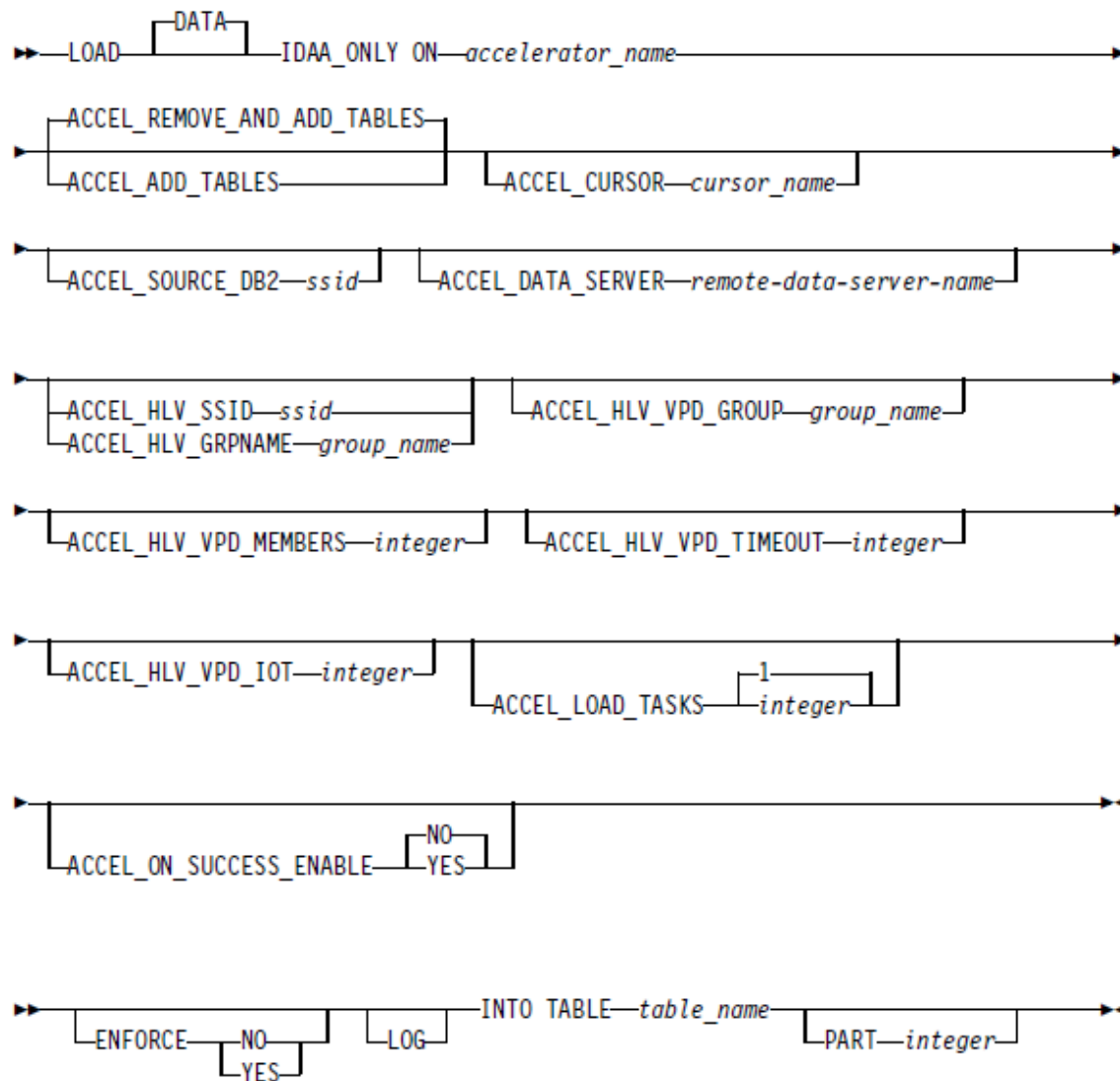
Note: ACCEL_HLV_VPD_IOT is an optional parameter. Because this parameter is not included in the example, the default value will be used.

Submit the jobs to run concurrently. If one of the jobs fails to join the group within the specified timeout value, the other two jobs would proceed and the third job would be placed in a new group, resulting in an additional read of the data set.

For more information about using VPD, see “Generating JCL” and “Virtual Parallel Data”.

Topic: “Syntax diagram: Load from a non-DB2, remote DB2, or remote system source”

Update diagram:



Topic: “Syntax definitions: Load from a non-DB2, remote DB2, or remote system source”

Add the following syntax elements:

ACCEL_HLV_VPD_GROUP *group_name*

Specifies the eight character VPD group name. This keyword is required to use the VPD feature.

ACCEL_HLV_VPD_MEMBERS *integer*

Specifies the number of members in the VPD group. Each Accelerator Loader job must be counted as a group member. This keyword is optional. If this value is not provided, the Accelerator Loader server will wait until the timeout expires before closing the group and finishing the request.

ACCEL_HLV_VPD_TIMEOUT *integer*

Specifies the amount of time, in seconds, that members have to join the group before it closes.

ACCEL_HLV_VPD_IOT *integer*

Specifies the number of I/O threads the Accelerator Loader server will create for reading the data set.

Chapter “Administering the Accelerator Loader server”

Update the topics as described below. Topics are located at the end of this document.

Remove the following topics in this chapter:

Topic: “Modifying IMS parameters”

Topic: “Configuring access to SMF files”

Add the following new topics:

Topic: “IMS data access methods”

Topic: “Virtual table SAF security”

Topic: “Virtual Parallel Data”

Topic: “Configuring Virtual Parallel Data”

Organize the new topics as follows:

IMS data access methods

Virtual table SAF security

Virtual parallel data

Configuring virtual parallel data

Chapter “Troubleshooting”

Topic: “Messages and codes”

Add the following messages.

HLV3786W IMS-Direct DBD refresh encountered %1 errors - See messages above

Explanation: One or more problems was encountered while refreshing the DBD information used for IMS-Direct map reduce processing. One or more sets of DBD information may not have been built or rebuilt correctly.

User response: No action is required.

HLV3787W IMS-Direct database(%1) dataset(%2) setup error: %

Explanation: One or more problems was encountered while collecting dataset information for use by IMS-Direct map reduce processing.

User response: No action is required.

HLV3788T IMS-Direct database(%1) thread count reduced from %2 to %3 due to dataset(%4) characteristics

Explanation: The parallel map reduce thread count (or default) to be used for IMS-Direct processing has been reduced due to dataset-versus-count sizing considerations. Fewer threads than requested will be scheduled. This will occur when a HIDAM primary index dataset s index set contains too few records to make beneficial use of more map reduce tasks.

User response: No action is required.

HLOU4067E: Function GET_ACCEL_GROUP failed. RC=<hex_returncode>, RSN=<hex_reason_code>

Explanation: An error has prevented the product from determining if the specified accelerator name is an accelerator group. This failure could be caused by an SQL error. Check the started task log for additional error messages.

User response: No action is required.

HLOU4068I: *<group_name>* is an accelerator group. The following members will be loaded:

Explanation: The specified accelerator group has been resolved to its member accelerators. All members in the group will be loaded. This message is followed by 1 or more HLOU4069I messages each of which lists one member of the group.

User response: No action is required.

HLOU4069I: *<member_accelerator_name>*

Explanation: This informational message lists a member of an accelerator group. This message is issued once for each member of the accelerator group. It is issued in conjunction with HLOU4068I to display all of the members of an accelerator group.

User response: No action is required.

Chapter “Reference”

Topic: “Accelerator Loader terminology”

Add the following terms:

IMS Direct

An Accelerator Loader server feature that enables access to IMS data directly as opposed to accessing the data through DLI calls.

Virtual Parallel Data (VPD)

A feature that enables simultaneous loading of various record types from a virtual source to the accelerator.

Configuring access to data in IBM IMS databases

Set up access to data in IBM IMS databases by configuring the Accelerator Loader server and verifying access to the data.

Before you begin

The Accelerator Loader server must already be installed. Use these instructions to configure the Accelerator Loader server.

About this task

To access an IMS database, the Accelerator Loader server started task and parameter file must be configured with information about the IMS databases to which you want to connect. Customizing these members is done using Tools Customizer. No configuration changes are necessary to IMS.

Procedure

1. Invoke Tools Customizer for z/OS.
2. Access the Product Parameters panel.
3. Under the task 'Create the server and the server components', select the steps **Create the server** and **Create the server parameters**, and provide values for the following fields:

Step or parameter	Required?	Discovered?	Default	Your value
IMS subsystem ID Specifies the IMS subsystem ID that the server uses as a data source. The IMS subsystem must be on the LPAR for which the product is being configured.	No	No	No default	
IMS SDFSRESL library Specifies the IMS load library that the server uses to connect to the IMS systems on the LPAR that you are configuring. If this value is defined, the server uses IMS™ as a data source. If this value is not defined, the server does not use IMS as a data source.	No	No	No default	
IMS MODBLKS library Defines the IMS MODBLKS staging library that contains the control blocks to support online change of databases, programs, transactions, and MFS formats for the specified IMS subsystem. This data set enables the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	

Step or parameter	Required?	Discovered?	Default	Your value
IMS ACBLIB library Defines the IMS ACBLIB library that contains database and program descriptors for the specified IMS subsystem. This data set enables the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	
IMSDALIB library Defines the IMSDALIB library that contains the DFSMDA members that are used for dynamic allocation for the specified IMS subsystem. This data set enables the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	
RECON library RECON2 library RECON3 library Define the RECON libraries that contain system information for the specified IMS subsystem. These data sets enable the server to access IMS data directly.	If an IMS subsystem is defined, you must specify a value.	No	No default	

4. Generate the customization jobs. The jobs are based on the templates HLOHLVS and HLOIN00. For more information, see Generating customization jobs.
5. Submit the customization jobs. For more information, see Submitting customization jobs.

Configuring access to SMF data for IT Operational Analytics

IT Operational Analytics (ITOA) allows you to retrieve, analyze, and report data for IT operations. System information can be logged using the IBM System Management Facility (SMF) and the native Accelerator Loader server logging feature. Logging allows you to collect various system and operations-related information.

Before you begin

IBM APAR OA49263 provides real-time SMF support and is a requirement for the configuration of real-time SMF. APAR OA48933 is required to address accessing logstreams. SMF logstream configuration is required for in-memory resource support.

About this task

Virtual tables for SMF are provided in the *hlq.SHLVMAP* data set.

There are three options to access the SMF data:

- Data set mode - SMF information is recorded in MANx data sets. When a data set gets full, the data is processed via IFASMFDP. When defining global variables for SMF in data set mode, the output of IFASMFDP is used.
- Logstream mode - SMF information is recorded in multiple logstreams and data can be read directly from the logstreams. Logstream recording is determined by the data set name beginning with IFASMF that is used in the VTB rule for SMF.
- In-memory (real-time) mode - SMF information is read directly from the system buffer. SMF information is read in real time.

When defining the global variables for SMF, the data set can be either a logstream or a SMF dump data set from IFASMFDP. The logstream data set is recommended for access to near real-time data.

Configuring access to System Management Facility (SMF) files

By default, access to System Management Facility (SMF) files is enabled in the Accelerator Loader server started task JCL and the server configuration member. To enable reading SMF data real-time using logstreams, you must have the **SMFPRMxx** member in the system PARMLIB data set configured to use both logstreams and in-memory resources. Follow the steps in this section to use SMF GDG data set names, or to use dynamic data set names.

About this task

SMF data set names are dynamic in local environments and require SEF rules enablement and optionally Global Variables set to specific values in order to provide data set names to the virtual tables and views when using SMF data set or logstream configurations.

You can choose either GDG data set name support or dynamic data set name support, or both, to quickly access your SMF data. These two options are provided

for your convenience to help you start accessing your SMF data. Custom rules may need to be developed to use your local naming convention to access your SMF files.

Procedure

1. To enable real-time access to SMF data, add the following statements to the *hlvidIN00* member after the GLOBAL PRODUCT OPTIONS statement:

```
IF DoThis
  THEN DO
    "DEFINE SMF NAME(IFASMF.INMEM)",
    "BUFSIZE(500)",
    "TIME(0)"
  END
```

Parameter	Description	Valid values
BUFSIZE	Indicates how much SMF data (megabytes) will be retained in memory for queries. If the buffer fills up, the oldest data will be discarded. In parallel, SMF is recording these records to a logstream.	1-10,000
TIME	Indicates how long (in minutes) to keep SMF data in memory. Older data will be discarded. Specifying 0 indicates no time limit and data will be retained until the buffer fills up.	0-999

2. Optional: Enable SMF using GDG data set names.
 - a. Enable the Server Event Facility rule SHLVSMAP in the VTB ruleset using ISPF panel option SEF Rule Management with the SMF dump data set, the logstream data set, or the in-memory resource name. You can choose to activate all three options by customizing the rule.

The VTB rule enables you to select from a GDG data set, any SMF dump data set, a logstream data set, or the in-memory stream.

 - 1) On the main menu, select **Server administration**.
 - 2) In the Administer Accelerator Loader Server menu, select option **3** for Manage Rules.
 - 3) Select option **2** for SEF Rule Management.
 - 4) Enter VTB for Display Only the Ruleset Named.
 - 5) Enable the rule by entering **E** and pressing Enter.
 - 6) Set the rule to Auto-enable by using **A** and pressing Enter.

Setting the rule to Auto-enable activates the rule automatically when the server is re-started.
 - b. Review the information in the SHLVSMAP rule for the instructions on setting Global Variables that will be used by the rule. Navigate one screen back on the ISPF panel, or start over by going to option **3** for Manage Rules and then option **1** for Global Variables. In the Global Variable display, select the following:
 - 1) Change Global Prefix to GLOBAL2.
 - 2) Select SMFTBL2 by entering **S** next to the SMFTBL2 data set.

3) Configure the SMF data access option:

- GDGBASE and DEFAULT should have corresponding SMF dump data set names if used.
- IM should have its corresponding SMF in-memory name if used.
- LOG should have its corresponding SMF logstream name if used.

Note:

VTB rules and global variables may be used to reference a GDG data set, any SMF dump data set, a logstream data set, or the in-memory stream. For example:

```
GLOBAL2.SMFGBL2.YESTERDAY = "YOUR.DATASET.SMFDUMP(-1)"
GLOBAL2.SMFGBL2.M2 = "YOUR.DATASET.SMFDUMP(-2)"
GLOBAL2.SMFGBL2.M3 = "YOUR.DATASET.SMFDUMP(-3)"
GLOBAL2.SMFGBL2.M4 = "YOUR.DATASET.SMFDUMP(-4)"
GLOBAL2.SMFGBL2.M5 = "YOUR.DATASET.SMFDUMP(-5)"
GLOBAL2.SMFGBL2.IM = "IFASMF.INMEM"
GLOBAL2.SMFGBL2.IM2 = "IFASMF.INMEM2"
GLOBAL2.SMFGBL2.LOG = "LOGSTREAM.dataset.name"
```

3. Optional: Enable SMF using dynamic data set names.

- a. Enable the virtual table rule HLVSMFT3 to support the ability to pass a dynamic data set name for SMF tables. The table name in the SQL statement must be of the form:

TableMapName__DataSetName

Where DataSetName is prefixed by two underscores '__' and the periods in the data set name are replaced with single underscores '_'.

For example, SELECT * FROM SMF_01400__DATA_SET_NAME would translate into an SQL query of SELECT * FROM SMF_14000 and access the data set DATA.SET.NAME.

- b. Enable the virtual view rules HLVSMFT4 and HLVSMFT5 to support the ability to pass a dynamic data set name for SMF virtual views. The virtual view name in the SQL statement must be of the form:

ViewMapName__DataSetName

Where DataSetName is prefixed by two underscores '__' and the periods in the data set name are replaced with single underscores '_'.

For example, SELECT * FROM SMFV_01400_IFGSMF__DATA_SET_NAME would translate into an SQL query of SELECT * FROM SMFV_01400 and access the data set DATA.SET.NAME.

Configuring access to SYSLOG files

The Accelerator Loader server is enabled to support access to SYSLOG files. Use these steps to enable the rule.

Procedure

1. On the main menu, select **Server administration**.
2. In the Administer Accelerator Loader Server menu, select option 3 for Manage Rules.
3. Select option 2 for SEF Rule Management.
4. Enter YSLG for Display Only the Ruleset Named.
5. Enable the rule by entering E and pressing Enter.
6. Set the rule to Auto-enable by using A and pressing Enter.

Setting the rule to Auto-enable activates the rule automatically when the server is re-started.

Configuring access to zFS files

The Accelerator Loader server is already configured to support zFS files. No modifications are needed to configure the SQL interface to access zFS files.

Configuring access to distributed databases

You can configure access to data on DB2 LUW (Linux, UNIX, and Windows), Microsoft SQL Server, Oracle, and QMF DRDA.

Before you begin

If you are connecting to a DB2 LUW database, then you must install and configure the IBM DB2 Federated Server. For additional information, refer to the documentation on the IBM website.

If you are connecting to an Oracle database, then you must install and configure the Oracle Database Provider for DRDA. For additional information, refer to the documentation on the Oracle website.

If you are connecting to a 2016 Microsoft SQL Server database, then you must install and configure the Host Integration Server for HIS DRDA Service. For additional information, refer to the documentation on the Microsoft website. The SYSIBM Views from Microsoft must be installed.

About this task

Configure access to distributed databases as follows.

Procedure

1. "Modifying the server configuration member."
2. Configure the Server Event Facility rules for the appropriate database.
 - Configuring Server Event Facility rules for Linux, UNIX, and Windows.
 - "Configuring Server Event Facility rules for Microsoft SQL Server" on page 12.
 - Configuring Server Event Facility rules for Oracle DRDA.
 - Configuring Server Event Facility rules for QMF DRDA Server.

Modifying the server configuration member

Configure the Accelerator Loader server to access data sources using Distributed Relational Database Architecture (DRDA).

About this task

The Accelerator Loader server is enabled for DRDA access. To access data sources using DRDA, modify the Accelerator Loader server parameter member *hlvidIN00* that was configured using Tools Customizer, and define those data sources to the configuration member.

The server configuration member *hlvidIN00* is in data set *hlq.SHLVEXEC*, where *hlvid* represents the name of the Accelerator Loader server started task that was customized using Tools Customizer.

Procedure

1. Verify that the Unicode translation of the Coded Character Set Identifier (CCSID) used in the DEFINE DATABASE statement and the CCSID used by the target RDBMS are defined for your z/OS environment.

- a. Identify the CCSID of the RDBMS.

For example, Oracle may use *ccsid1*. In your DEFINE DATABASE statement in the configuration member for the RDBMS you have *ccsid2*. For this example, where Oracle is using *ccsid1*, you need to verify that you have *ccsid1-ccsid2* and *ccsid2-ccsid1* defined in your Unicode translation table on z/OS using the command **D UNI,ALL**.

- b. If the entry is not present, add the entry to your Unicode translation table and refresh.

Refer to the IBM z/OS documentation on how to add the entry.

Note: As an alternative, the Unicode table can be appended within the Accelerator Loader server by using the following statement examples in the server configuration member:

```
"DEFINE CONV SOURCE(ccsid1) TARGET(ccsid2) TECH(RE) "  
"DEFINE CONV SOURCE(ccsid2) TARGET(ccsid1) TECH(RE) "
```

2. In the *hlvidIN00* member, locate the section that contains the comment "Enable DRDA access to DB2 database subsystems."
3. Define DRDA RDBMSs by entering a definition statement. Provide your local environment values for all the parameters.

```
"DEFINE DATABASE TYPE(type_selection) "      ,  
      "NAME(name) "                          ,  
      "LOCATION(location) "                   ,  
      "DDFSTATUS(ENABLE) "                   ,  
      "DOMAIN(your.domain.name) "           ,  
      "PORT(port) "                          ,  
      "IPADDR(1.1.1.1) "                     ,  
      "CCSID(37) "                           ,  
      "APPLNAME(DSN1LU) "                   ,  
      "IDLETIME(110) "                       ,
```

Where *type_selection* is either LUW, MSSQL, ORACLE, or QMFD RDA.

The following table lists the parameters for defining DDF endpoints:

Parameter	Description	Valid values
TYPE	<p>For distributed databases:</p> <p>LUW - DDF endpoint is a DB2 instance or group member for Linux, UNIX, or Windows.</p> <p>MSSQL - DDF endpoint is a DB2 instance or group member for Microsoft SQL Server.</p> <p>ORACLE - DDF endpoint is an Oracle instance. The parameter informs DRDA AR and supportive tooling that the remote server is an Oracle Database Provider which supports DRDA AS. The Oracle DRDA AS must be in z/OS simulation mode.</p> <p>QMFD RDA - DDF endpoint is a QMF DRDA AS Object Server instance.</p>	<p>For distributed databases:</p> <p>LUW</p> <p>MSSQL</p> <p>ORACLE</p> <p>QMFD RDA</p>
NAME	The database name as known to the server. <i>(Required)</i>	A valid value consists of 1 - 4 characters. Clients use this ID when they request access to a specific DB2 subsystem.
LOCATION	<p>For DB2: The DB2 location name.</p> <p>For LUW: The LUW database.</p> <p>For Oracle: The Oracle SSID as defined to the Oracle Database Provider (Gateway). <i>(Required)</i></p>	A valid value is a string 1 - 16 characters.
DDFSTATUS	The DDF activation status can be altered online by using the ISPF 4-DB2 dialog panels. <i>(Required)</i>	<p>ENABLE Make this DDF definition active.</p> <p>DISABLE DDF endpoint is not used.</p>
DOMAIN	The part of a network address that identifies it as belonging to a particular domain. Either DOMAIN or IPADDR is required, but not both.	No default value.
PORT	The TCP/IP port at which the server is listening. <i>(Required)</i>	A valid 1-5 numeric string.

Parameter	Description	Valid values
IPADDR	Specify the dot-notation IPV4 address of the DDF endpoint. Either DOMAIN or IPADDR is required, but not both.	If this parameter is not specified, the value 127.0.0.1 (local host) is the default. For group director definitions, use the DVIPA IP address of the group director.
CCSID	Specify the EBCDIC single-byte application CCSID (Coded Character Set Identifier) configured for this RDBMS subsystem on the RDBMS installation panel DSNTIPF, option 7. (<i>Optional</i>)	Refer to the RDBMS vendor documentation for a list of valid CCSIDs.
APPLNAME	Application name. The APPLNAME used by the target endpoint for passticket generations. (<i>Optional</i>)	A valid value is 1 - 8 characters. If APPLNAME is not specified in the definition statement, no default value is provided and passticket access is disabled. Note: APPLNAME is not required when connecting from the JDBC driver.
IDLETIME	If DB2 ZPARM parameter IDTHTOIN is set to a non-zero value set IDLETIME to a value slightly lower (10 secs.) than IDTHTOIN. This will also allow product DRDA threads to become inactive. (<i>DB2 for z/OS only</i>)	0-9999 seconds.
AUTHTYPE	Authentication type. This can be either DES for Diffie Hellman Encryption Standard or AES for Advanced Encryption Standard. When AUTHTYPE is not supplied, the default is DES. To force AES, the option must be added to the DEFINE DATABASE statement. Each server can be different in what is supported as to AES/DES.	DES Diffie Hellman Encryption Standard (default value) AES Advanced Encryption Standard.

Configuring Server Event Facility rules for Microsoft SQL Server

Configure Server Event Facility (SEF) rules to provide access to Microsoft SQL Server databases via the 2016 Host Integration Server for HIS DRDA Service.

Procedure

1. Auto-enable the SQL rule *hlq.SHLVXSQL*(HLVSMSSC) to allow Accelerator Loader studio discovery on SQL Server databases.
 - a. On the main menu, select **Server administration**.
 - b. On the Administer Accelerator Loader Server menu, select option 3 for Manage Rules.
 - c. Select option 2 for SEF Rule Management.
 - d. Enter * to display all rules, or SQL to display only SQL rules.
 - e. Set Auto-Enable for the HLVSMSSC rule member by entering A and pressing Enter.
2. Auto-enable the SEF ATH rule HLVMSSG to provide the logon credentials to each SQL Server instance. Global variables are used to define alternate authentication credential mapping for the SEF ATH rule.
 - a. On the main menu, select **Server administration**.
 - b. On the Administer Accelerator Loader Server menu, select option 3 for Manage Rules.
 - c. Select option 2 for SEF Rule Management.
 - d. Enter * to display all rules, or ATH to display only authentication rules.
 - e. Set Auto-Enable for the HLVMSSG rule member by entering A and pressing Enter.
3. Optional: Verify the HLV global variable setup for authentication rules:
 - a. On the main menu, select **Server administration**.
 - b. On the Administer Accelerator Loader Server menu, select option 3 for Manage Rules.
 - c. Select option 1 for Global Variables.
 - d. Enter GLOBAL2 and press Enter to display all GLOBAL2 Variables.
 - e. If subnode DRDA does not exist, enter S DRDA in the command line and press Enter.
 - f. If subnode ATH does not exist, enter S ATH in the command line and press Enter.
 - g. If subnode HLVMSSG does not exist, enter S HLVMSSG in the command line and press Enter.

Example

It is common for data centers to assign different user IDs for access to z/OS and for access to SQL Server. By default, the server will attempt to log on to SQL Server with the same user ID that was presented for logon to z/OS. A facility is needed in the server to optionally change a user's logon credentials when accessing DB2 on SQL Server.

A Server Event Facility (SEF) ATH rule can be used to set these two parameters for logon to SQL Server:

ATH.AUDROTUS	Microsoft SQL Server ID
ATH.AUDROTPW	Microsoft SQL Server Password

A SEF rule could be coded to insert hardcoded user IDs and passwords, even testing for different incoming IDs and specific SQL Server databases. However, as an alternative to hard-coding IDs and passwords in the SEF rule, *hlq.SHLVXATH*(HLVMSSG) is provided to resolve SQL Server credentials from

the Server Global Variables. Rule HLVMSSG should not be modified, except to toggle tracing of its execution. An example of how Server Global Variables are mapped is as follows:

```
GLOBAL2.DRDA.ATH.REXX.GLOBAL.DEFAULT
GLOBAL2.DRDA.ATH.REXX.GLOBAL.userid
GLOBAL2.DRDA.ATH.REXX.ssid.GLOBAL.DEFAULT
GLOBAL2.DRDA.ATH.REXX.ssid.userid
```

Where:

- *REXX* is the name of the active SEF ATH rule, in this case HLVMSSG.
- *ssid* is the target SQL Server subsystem name in the DEFINE DATABASE NAME(ssid) statement.
- *userid* is the incoming z/OS user ID.

The last node of each Global Variable sets the SQL Server user ID and password in the format:

userid:password;comment after the semicolon

For instance, using active SEF rule HLVMSSG, a default SQL Server user ID and password could be set to mssuser/msspswd by setting the subnode value of GLOBAL2.DRDA.ATH.HLVMSSG.GLOBAL.DEFAULT to:

mssuser:msspswd;Global Default for any user

Similarly, a default SQL Server user ID and password for subsystem MSS1 could be set to mssluser/msslpswd by setting the subnode value of GLOBAL2.DRDA.ATH.HLVMSSG.MSS1.GLOBAL.DEFAULT to:

mssluser:msslpswd;Default for MSS subsystem named MSS1

Each user can be assigned unique credentials for MSS1. Assuming ZOSUSER1 and ZOSUSER2 are the z/OS user IDs:

```
GLOBAL2.DRDA.ATH.HLVMSSG.MSS1.ZOSUSER1
subnode value
msslusera:pswda;MSS1 credentials for ZOSUSER1
GLOBAL2.DRDA.ATH.HLVMSSG.MSS1.ZOSUSER2
subnode value
mssluserb:pswdb;MSS1 credentials for ZOSUSER2
etc.
```

Each user can be assigned unique default credentials for all SQL Servers. Assuming ZOSUSER1 and ZOSUSER2 are the z/OS user IDs:

```
GLOBAL2.DRDA.ATH.HLVMSSG.GLOBAL.ZOSUSER1
subnode value
msslusera:pswda;MSS credentials for ZOSUSER1
GLOBAL2.DRDA.ATH.HLVMSSG.GLOBAL.ZOSUSER2
subnode value
mssluserb:pswdb;MSS credentials for ZOSUSER2
etc.
```

Searches to resolve the SQL Server credentials occur in the following order:

```
GLOBAL2.DRDA.ATH.HLVMSSG.ssid.userid
GLOBAL2.DRDA.ATH.HLVMSSG.GLOBAL.userid
GLOBAL2.DRDA.ATH.HLVMSSG.ssid.GLOBAL.DEFAULT
GLOBAL2.DRDA.ATH.HLVMSSG.GLOBAL.DEFAULT
```

The following is an example of SQL Server Global Variables for the ATH rule HLVMSSG.

----- Display Global Variables ----- Row 1 to 10 of 10

Command ==> Scroll ==> CSR
LCs: S Show Subnodes M Modify Value X Hex Browse
D Remove Node and Subnodes P Drop Node B Browse

Global Prefix: GLOBAL2.DRDA.ATH.HLVAMSSG

S Subnode Name Nodes Subnode Value

EXMP 0 TYPE(SQL Server) SQL Server DRDA Provider Example NAME(EXMP)
GLOBAL 2 TYPE(SQL Server) GLOBAL DEFAULT ATH USERID/PASSWORD

The following is an example of SQL Server Global Variables for SUBSYS
NAME(EXMP) to swap USERID to ALTUSER.

----- Display Global Variables ----- Row 1 to 1 of 1

Command ==> Scroll ==> CSR
LCs: S Show Subnodes M Modify Value X Hex Browse
D Remove Node and Subnodes P Drop Node B Browse

Global Prefix: GLOBAL2.DRDA.ATH.HLVAMSSG.EXMP

S Subnode Name Nodes Subnode Value

USERID 0 ALTUSER:PASSWORD;USERID SWAP TO ALTUSER

Creating virtual tables for IMS data

Create a virtual table that maps to the IMS data that you want to access, and from which the SQL used to access the data is generated and executed.

Before you begin

The Program Specification Block (PSB) and Database Definition (DBD) source members, and the copybooks for each segment must exist in the source library.

Procedure

1. Expand **SQL > Data** and expand the server on which you want to create the virtual table.
2. Right-click **Virtual Tables** and select **Create Virtual Table**.
3. Under **Wizards**, select the **IMS** wizard and click **Next**.
4. On the **New IMS Virtual Tables** page, complete the following data layout fields and click **Next**:

Field	Action
Choose IMS Definitions	<p>Before you can create an IMS virtual table, you must first extract both the DBD and PSB definitions that create the metadata on the Accelerator Loader server. To download new members from DBD files in your source libraries, click Extract DBD, or select a previously downloaded DBD definition from the DBD drop-down list.</p> <p>To download new members from PSB files in your source libraries, click Extract PSB, or select a previously downloaded PSB definition from the PSB drop-down list.</p> <p>Clicking Finish before specifying both the DBD and PSB to use produces the same results as clicking Cancel.</p>
Create Virtual Table	To create a virtual table for each IMS segment click Create Virtual Table .

5. On the **New IMS Virtual Tables** page, complete the following fields and click **Next**:

Field	Action
Name	Enter a unique name for the new virtual table.
Target	Shows the target data set that will be used to store the map information (for example: <i>hlq.USER.MAP</i>). Data sets are defined in the Accelerator Loader server configuration file.
Description	Enter an optional description.

6. On the **Source Download** page, complete the following fields and click **Next**:

Field	Action
Available Source Libraries	From the list of Available Source Libraries , select the source library that contains the data structure definition that you want the virtual table to use.
Source Library Members	Select the PDS members that represent the data structures to include and click Download to copy the members from the mainframe to your desktop.
Download Source Files	Select one or more previously downloaded members.

7. On the **Virtual Table Layout** page, complete the following fields and click **Next**:

Field	Action
Source	Expand the source file to verify that it correctly displays the source (member).
Start Field	Accept the default root start field, or expand the file and select a different start field. Generally, you want to map all the data. However, if the source file is a program listing from which you want to select specific information, select the start and end fields to only map to that information.
End Field	Accept the default root end field, or expand the file and select a different end field. By default, End Field is disabled.

8. On the **IMS Server Configuration** page, complete the following fields:

Field	Action
Server Configuration	<p>Select from the following IMS configuration options:</p> <ul style="list-style-type: none"> • Use IMS/DBCTL (read/write, transactional integrity) • Use IMS/Direct (read-only, fast image copy) <p>Choosing the default Use IMS/Direct (read-only, fast image copy) option enables IMS Direct on the virtual table. However, to use this feature, IMS Direct must also be enabled for the selected DBD and enabled in the Accelerator Loader server IN00 file.</p>

Field	Action
Advanced	When reading large volumes of data from tables, select the Advanced option to configure the MapReduce feature. The MapReduce feature enables you to divide the data into logical partitions and process those partitions in parallel using the Thread Count value. At runtime, the number of zIIP processors is verified and one thread is used for each zIIP processor; resulting in improved performance. The Thread Count value you specify overrides the default value (2) and the discovered value. To disable MapReduce , select the Disable MapReduce check box.

9. Click **Finish**.

What to do next

Generate the SQL query on the virtual table and execute the query to view the result set.

Creating virtual tables for zFS and HFS file system data

Create a virtual table that maps to file data that you want to access on a zFS or HFS file system and on which the SQL used to access the data is generated and executed.

Before you begin

Before creating the virtual table, verify that the PDS members that represent the data structures for the data you want to virtualize already exist in the source library.

Procedure

1. Expand **SQL > Data** and expand the server on which you want to create the virtual table.
2. Right-click **Virtual Tables** and select **Create Virtual Table**.
3. Under **Wizards**, select the **zFS** wizard and click **Next**.
4. On the **New Virtual Table Wizard** page, complete the following fields and click **Next**:

Field	Action
Name	Enter a unique name. The name can contain a maximum of 30 characters. Uppercase alphanumeric characters, the numbers 0-9, and the underscore (_) character are allowed. The initial character in the name must be an alphanumeric character.
Target	Shows the target data set that will be used to store the map information (for example: <i>hlq.USER.MAP</i>). Data sets are defined in the Accelerator Loader server configuration file
Description	Enter an optional description.
Convert VAR* fields to True VAR* fields	To convert existing VAR* fields to True VAR* fields, select the Convert VAR* fields to True VAR* fields check box.
Arrays Handling	Enable one of the following array management options: <ul style="list-style-type: none">• Flatten arrays into a single fixed table at runtime: This supports both OCCURS and OCCURS DEPENDING ON statements.• Return arrays into separate tables at runtime: This supports both OCCURS and OCCURS DEPENDING ON statements. A subtable is generated for each array. Subtables only support SQL read access.• Flatten arrays now: If you select this option, you cannot change array-handling after you save the virtual table.

5. On the **Source Download** page, complete the following fields and click **Next**:

Field	Action
Download Folder	Verify that the appropriate download folder is displayed.
Available Source Libraries	Select the source library that contains the data structure to use.
Source Library Members	Select the PDS members that represent the data structures to include and click Download to copy the members from the mainframe to your desktop.
Downloaded Source Files	Select one or more previously downloaded members. Selecting previously downloaded members is optional.

6. On the **Virtual Table Layout** page, complete the following fields and click **Next**:

Field	Action
Available Files	Accept the file that is displayed, or select a different file from the drop-down list.
Source	Expand the source file to verify that it correctly displays the source (member).
Start Field	Accept the default root start field, or expand the file and select a different start field. Generally, you want to map all the data. However, if the source file is a program listing from which you want to select specific information, select the start and end fields to only map to that information.
End Field	Accept the default root end field, or expand the file and select a different end field. By default, End Field is disabled.

7. On the **zFS Virtual Table Details** page, complete the following fields:

Field	Action
Pathname	<p>Enter the path name of the zFS file.</p> <p>If the absolute path name of the zFS file is less than 255 characters in length, you must include the root slash "/" in the path name. For example, /u/tsado/data/stuff.txt.</p> <p>If the absolute path name of the zFS file is greater than 255 characters in length, you must enter the relative path name. The relative path name starts with the name of the target system to indicate the top-level directory and does not include the leading root slash. For example, data/stuff.txt, where "data" is the name of the target system.</p>

Field	Action
Target System	<p>If you plan to map several zFS files under the same zFS directory location, specify a target system to use.</p> <p>You can click Create to add a new path name to use, or if a relative path name is already specified in the Pathname field, you must select an existing target system from the drop-down list.</p> <p>If you choose to create a new target system, complete the following fields and click Finish:</p> <p>Name – Enter the name for the new target system.</p> <p>CCSID – Enter the CCSID of the character set in which the zFS file data is encoded. The default setting is EBCDIC 1047.</p> <p>Base Pathname – Enter the absolute path name under which the zFS file resides. Typically, this is the path name of the zFS subdirectory that contains your zFS file. At runtime, the server will determine the location of the zFS file by concatenating the path name with the value specified in the virtual table Pathname field. The Accelerator Loader server does not insert additional slash (/) separators when concatenating the target system path name and the virtual table path name. If the target system path name represents a complete directory name, include the trailing slash (/tmp/).</p>
Advanced	<p>When reading large volumes of data from tables, select the Advanced option to configure the MapReduce feature. The MapReduce feature enables you to divide the data into logical partitions and process those partitions in parallel using the Thread Count value. At runtime, the number of zIIP processors is verified and one thread is used for each zIIP processor; resulting in improved performance. The Thread Count value you specify overrides the default value (2) and the discovered value. To disable MapReduce, select the Disable MapReduce check box.</p>

8. Click **Finish**.

What to do next

Generate the SQL query on the virtual table and execute the query to view the result set.

Accessing IT Operational Analytics data

To access, analyze, and report IT Operational Analytics (ITOA) data, generate the SQL from ITOA virtual tables.

The ITOA virtual tables are pre-defined data maps that define the following types of ITOA data:

- IBM System Management Facilities files (SMF)
- Operations Log files (OPERLOG)
- System Log files (SYSLOG)

System administrators can use these virtual tables to generate the SQL that is used to access ITOA data from these pre-defined data maps. Administrators can then use the data to analyze IT operations such as system performance. Because the Accelerator Loader server can use the ITOA pre-defined data maps, you can access the ITOA data.

Generating JCL

Generate the JCL that loads the data to the accelerator from a virtual table or a virtual view.

Before you begin

- Have the following information available before starting this task:
 - Target server name
 - Target DB2® subsystem name
 - DB2 load library names
 - Accelerator name
 - Product data set names
 - JCL library name
 - Table creator name
 - Table name
- To enable the **Load Resume** feature, you must have IBM DB2 Analytics Accelerator for z/OS V4 PTF5 installed on the accelerator.
- If the source data is not hosted on the same LPAR as the accelerator, two accelerator servers are required:
 - The *data server* is the Accelerator Loader server that hosts the virtual table or view.
 - The *target server* is the Accelerator Loader server that has access to the target DB2 subsystem and the accelerator. The target server's *hlvidIN00* file must contain a configuration entry for the data server. When you run the wizard to generate the JCL, the name of this configuration definition is referred to as the *Data Server Name*.

About this task

In the steps that follow, the information that you enter is only required the first time that you generate the JCL.

Procedure

1. On the **Server** tab, expand **SQL > Data**.
2. Expand **Virtual Views** or **Virtual Tables** to navigate to the virtual view or virtual table that represents the source data that you want to load.
3. Right-click the virtual view or table, and select **Generate JCL to Load Accelerator** to open the **Generate JCL to Load Accelerator** wizard.
4. On the **Source Information** page, review the source subsystem and the SQL query. If you choose to modify the SQL query, you must click **Validate** before you can proceed to the next page. If the query is not valid, an error message displays, showing the source of the problem. The SQL query must be corrected before you can continue.
5. Click **Next**.
6. On the **Target Information** page, specify the following information:

Field	Action
Target Server Selection	<ul style="list-style-type: none"> Choose Use Current Server if the server that is hosting the virtual table or virtual view resides on the same LPAR as the accelerator. This is the default setting. Choose Use Alternate Target Server if the server that is hosting the virtual table or virtual view does not reside on the same LPAR as the accelerator. If you choose this option, use Set Server to select the alternate server.
Target DB2 Subsystem	Accept the default DB2 subsystem name or choose a different subsystem to use from the drop-down list. This name must match the data server name that is configured in the target server <i>hlvid</i> IN00 file. The subsystem that you choose will display as the default setting the next time that you run the wizard.
Table Creator	Accept the Table Creator name that is displayed, or select a different name from the drop-down list.
Table Name	Accept the Table Name that is displayed, or select a different name from the drop-down list.
Available Accelerators	From the list of Available Accelerators , select the accelerators to use. If an accelerator is associated with a group, the Group label is displayed next to the accelerator name. You can select up to eight accelerators to load data simultaneously.
CREATE TABLE DDL Options	<p>Choose to include any of the following optional table DDL options:</p> <ul style="list-style-type: none"> Database Name – Enter the name of the database to use when loading data to the accelerator. Table Space Name – Enter the name of the table space to use when loading data to the accelerator. Accelerator Only Table – If this table is used to only load data to the accelerator and not to simultaneously load data to DB2 for z/OS, select Accelerator Only Table.
Enable Loader Parallelism	To use parallel processing (MapReduce), select Enable Loader Parallelism and either enter the number of parallel tasks to use, or click Fetch Existing Table Attributes if you are using an existing accelerated table name and you want to reuse the same parallelism settings. This setting is optional and is disabled by default.

Field	Action
Join Virtual Parallel Data Group (VPD)	<p>Select Join Virtual Parallel Data Group (VPD) to join an existing group of target servers, and specify the following:</p> <ul style="list-style-type: none"> • Group Name – Enter the name of the group to join. • Number of members – Enter the number of members (Accelerator Loader jobs) that are in the group. Although this setting is optional, it is recommended that you enter the number of Accelerator Loader jobs that will be used. • Group Timeout – Enter the timeout duration, in seconds. This controls the timing window starting from the point-in-time when the first VPD member query is made to the server, to the time that the VPD group is closed to further queries for this VPD group session. This setting is optional. • I/O Task Count – Enter the number of tasks to use for VPD. <p>This setting is optional and is disabled by default.</p>
DDL Preview	Click DDL Preview to preview the SQL CREATE statement that will be used to create the accelerated table. This step is optional.

7. Click **Next**.

8. On the **JCL Generation Details** page, specify the following information:

Field	Action
Target Server Group	Enter the name of the Target Server Group to use. This setting is optional.
Utility ID	Enter the target server Utility ID to use. This setting is optional.
STEPLIB DD Concatenation Libraries	<p>In the following fields, enter the data set names of the Accelerator Loader load libraries that you want to use.</p> <ul style="list-style-type: none"> • Accelerator Loader Library (1) • Accelerator Loader Library (2)
DB2 Load Libraries	To add a DB2 load library, click Add and enter the name of the DB2 load library. You can also choose an existing load library and click Modify or Delete to modify or delete the DB2 load library. This setting is optional.
DB2 Dynamic SQL Program (DSNTEP2)	<p>Enter the required details for running the DSNTEP2 sample DB2 program:</p> <ul style="list-style-type: none"> • DB2 Load Library (RUNLIB) • Plan Name (the default is DSNTEP2)

Field	Action
JCL Settings	Click JCL Settings to view or modify the JCL generation preferences in the JCL Preferences dialog and click OK . This setting is optional.

9. Click **Next**.

10. On the **Job Creation** page, specify the following information:

Field	Action
JCL Library	Select or enter the name of the JCL library to use. The JCL library that you choose is displayed the next time you open this page. If you do not plan to save the job on the host or if you do not know the location, leave this field blank. This setting is optional.
Member Name	Enter the name of the member to use to store your generated JCL.
Local File Name	Depending on the JCL library that you chose, you can choose to enter the name of a local file to use.
Job Name	Accept or modify the Job Name .
Job Step Details	<p>Select from the following job options:</p> <ul style="list-style-type: none"> • DROP TABLE – Drops the existing target table definition. Do not select this option if the accelerated table does not already exist. • CREATE TABLE – Creates a new target table definition. <p>From the LOAD REPLACE drop-down list, select a load option to use:</p> <ul style="list-style-type: none"> • LOAD REPLACE – Replaces the data in the table. • LOAD RESUME – Loads new data to an existing table. Selecting LOAD RESUME disables the DROP TABLE and CREATE TABLE options.
Additional Actions	<p>Choose from the following actions:</p> <ul style="list-style-type: none"> • Open in Editor – Select Open in Editor to review the JCL in the JCL editor. • Auto-run using JCL View – Select Auto-run using JCL View to automatically submit the JCL after it is generated. If the JCL library is specified in the JCL view, a copy of the JCL is saved in a PDS on the host. You can use this copy for future data loads. <p>These settings are optional.</p>

11. Click **Generate** to generate the JCL.

IMS data access methods

IMS data can be accessed by the Accelerator Loader server using different data access methods.

By default Accelerator Loader server will access IMS data directly using the underlying VSAM datasets. This access method, called "IMS Direct", provides both map reduce and parallelism support for accessing native IMS files. This support bypasses the requirement of having to use native IMS API calls by reading the IMS database files directly, similar to how an unload utility may work. This method provides a significant increase in performance and reduced elapsed time in processing analytical type queries.

When an IMS SQL query is run, the SQL engine for the server will determine if the request is best executed using IMS Direct (native file support) or if IMS APIs are required. The determination is based on database and file types supported as well as the size of the database. Virtual tables of the IMS segments are required.

The following types of IMS databases are currently supported:

- HIDAM/VSAM
- HDAM/VSAM
- PHDAM/VSAM
- PHIDAM/VSAM

When IMS Direct access is not available, the Accelerator Loader server will use DBCTL access using map reduce and parallelism support. Map reduce is an algorithm that enables the Accelerator Loader server to streamline how it accesses IMS data, thereby reducing the processing time required to virtualize IMS data. Statistics about the IMS database are collected and stored within a metadata repository from which the SQL engine optimizes the map reduce process.

In order to exploit the map reduce architecture for IMS using DBCTL as the access method, the Accelerator Loader server must collect information about the IMS database so that it can be used by the SQL engine optimizer. This information is stored within the Accelerator Loader server metadata repository for optimization and can be refreshed at regular intervals.

Metadata repository

The metadata repository for MapReduce stores statistics about virtual tables defined on IMS data sources that are used to enhance performance in conjunction with MapReduce. This support applies to IMS and all DRDA backend data sources, including those accessed via the IBM Federated Server (such as Terradata and Sybase), as well as data sources accessed via the Accelerator Loader server's direct DRDA support (DB2 LUW and Oracle).

This information can be collected by the following command query:

```
SELECT IMSRange('IMS database name')
```

The following sample batch job can be executed at regular intervals to populate the IMS metadata repository with fresh statistics:

```
//DSCLIENT EXEC PGM=HLVXMAPD,PARM='SSID=hlvid'  
//STEPLIB DD DISP=SHR,  
//          DSN=hlq.SHLVLOAD  
//OUT DD SYSOUT=*  
//IN DD *  
      SELECT IMSRANGE('abcd');
```

where:

- *hlvid* is the name of the Accelerator Loader server started task that was customized using Tools Customizer
- *hlq*.SHLVLOAD is the Accelerator Loader server load library
- *abcd* is the four-character IMS subsystem name.

No additional configuration or customization is required to take advantage of either of these access methods.

Virtual table SAF security

A single Accelerator Loader server environment can provide data virtualization to multiple independent tenants or application groups. The virtual table SAF (system authorization facility) security feature provides a SAF mechanism to secure virtual tables so that each tenant can only access tables authorized for members of the tenant group.

Activating this security feature will prevent using virtual table names in metadata queries (such as, **SQLENG.TABLES**, **SQLENG.COLUMNS**), as well as querying or updating application data mapped using unauthorized table names.

Server interface parameter

The SQLVTRESOURCETYPE parameter in the PRODSECURITY parameter group defines a security class name for virtual table resource checking. By default, this system parameter defaults to the value 'NON' indicating that security checking is disabled.

When activated with a class name, the SQLVTRESOURCETYPE parameter will enable SAF resource checking on metadata queries (such as, **SQLENG.TABLES**, **SQLENG.COLUMNS**) as well as virtual table queries using the resource name *resource_class.table_owner.table_name* where:

- *resource_class* is the class name define for the RESOURCETYPE parameter in the PRODSECURITY parameter group (for example: 'RSDB')
- *table_owner* is the SQL TABLE OWNER NAME (SQLTABLEOWNER) as defined in the PRODSQL parameter group (for example: 'DVSQ')
- *table_name* is the map (or virtual table) name as defined in the map data set

For improved performance in SAF calls, RACROUTE REQUEST=FASTAUTH provides general resource checking. A separate INTRNLONLY parameter named 'DISABLE FASTAUTH SECURITY CHECKS' disables use of FASTAUTH if security problems are encountered. Disabling FASTAUTH will switch to RACROUTE REQUEST=AUTH checking on all resource rules which can degrade query performance on metadata tables.

When securing metadata tables, READ access is required to query rows in the following tables.

- SQLENG.COLUMNS
- SQLENG.COLUMNPRIVS
- SQLENG.ERRORMSGs
- SQLENG.FOREIGNKEYS
- SQLENG.PRIMARYKEYS
- SQLENG.ROUTINES
- SQLENG.SPECIALCOLS
- SQLENG.STATISTICS
- SQLENG.TABLES
- SQLENG.TABLEPRIVS

Securing tables using the generic profile SQLENG.* is also an option if preferred.

Securing specific virtual tables is also required when activating this feature.

Securing virtual tables by specific or generic rules activates two security checks:

1. When querying metadata tables (SQLENG.*), users must minimally have READ access to the virtual tables in order for rows related to a table to be returned. In this case, there are no errors returned. Instead, the information about a specific table is omitted from the result set and the user has no indication that the table exists.
2. When querying virtual tables, the user must have READ access to each table in the SQL SELECT statement and UPDATE access to any table that is the target of an SQL INSERT, UPDATE, or DELETE statement.

Restrictions and Considerations

Virtual table authorization checking is built on general resource checking and is impacted by the following product parameter in the PRODSECURITY group:

- ALLOWUNPROT – The ALLOWUNPROT parameter allows access to unprotected resources. When set to YES, this parameter allows access to resource names that have no matching resource definition in the SAF database. ALLOWUNPROT should be set to NO to insure resource rules are correctly processed.

Note: ALLOWUNPROT=NO will automatically activate numerous resource checks unrelated to this feature.

The *table_owner.map_name* resource name is internally restricted to 44 bytes. While internal map names larger than 44 bytes are still allowed, resource checking will only pass the first 44 bytes of the *table_owner.map_name* string in the SAF call for validation. Generic resource rules will be necessary if map names exceed this limitation.

Because all maps are limited to a single table owner as defined in the SQLENGTABLEOWNER system parameter, users should consider a standard prefix for all map names they want to secure for application groups. This simple generic resource rules can be defined to protect these names. For example, if the SQLENGTABLEOWNER is configured as 'DVSQ' and an application group uses AG01 as a prefix on all table names, a generic resource 'DVSQ.AG01*' will control access to all tables starting with AG01 as a map name.

All SQL queries are automatically secured when this feature is activated. This means that resource rules must exist to allow READ access to the metadata tables SQLENG.*.

This feature is limited to SQL access to virtual tables. Users authorized to create tables can create tables which may not be accessible due to SQL access rules implemented using this feature.

Virtual Parallel Data

Virtual Parallel Data (VPD) allows you to group multiple simultaneous requests against the same data source and run them in parallel, while doing the input and output (I/O) only once. VPD also allows single or multiple requests to run with asymmetrical parallelism, separately tuning the number of I/O threads and the number of client or SQL engine threads.

To use this feature you must provide a VPD group name when submitting request(s). All requests submitted to the same Accelerator Loader server with the same group name within a time period will be placed into a VPD group. One or more I/O threads will be started to read the data source and write it to a wrapping buffer. Group members will share the data in the buffer(s), without having to read the data source directly.

A group is created when the first member request arrives. The group is closed either when all members (and all their parallel MRC threads) have joined, or when a timeout is expired. The I/O threads are started as soon as the group is created, and data begins to flow to the buffer. If the buffer fills before the group is closed, the I/O thread(s) will wait. Once the group is closed and active members begin consuming data, the buffer space is reclaimed and I/O continues.

VPD supports MapReduce Client (MRC), and group members can use different levels of MRC parallelism. For example, a single VPD group might have six members, three members using 5 MRC threads, and the other three using 9 MRC threads. The group will consist of six members and 42 client threads. The number of I/O threads is determined separately. VPD supports a group of a single member, thus supporting asymmetrical parallelism for single requests when using MRC.

Configuring Virtual Parallel Data

To configure Virtual Parallel Data, add required VPD parameters to your Accelerator Loader server configuration file. To use VPD when loading data, specify a group name and appropriate parameters when generating your load JCL.

Procedure

1. Configure the Accelerator Loader server by adding the following statements to your *hlvidIN00* member:

```
/-----/
/* Enable Virtual Parallel Data for asymmetrical parallelism */
/-----/
if DoThis then
do
  "MODIFY PARM NAME(VPDGROUPTIMEOUT) VALUE(60)"
  "MODIFY PARM NAME(VPDBUFFERSIZE) VALUE(40)"
  "MODIFY PARM NAME(VPDTRACEDB) VALUE(NO)"
```

The following table lists the VPD parameters:

Parameter	Description	Valid values
VPDBUFFERSIZE	Specifies the default buffer size, in megabytes above the bar, for a Virtual Parallel Data buffer.	Numeric value in megabytes. Default is 40.
VPDGROUPTIMEOUT	Specifies the maximum time, in seconds, from the time a group is formed until it is closed. Default: 60 seconds	Numeric value in seconds. Default is 60.
VPDTRACEDB	Controls whether Virtual Parallel Data processing will trace debugging messages.	NO Do not trace debugging messages (default). YES Trace debugging messages.
VPDTRACEREC	Causes Virtual Parallel Data to trace at the record level. <i>(Optional)</i> Note: Setting this to YES will produce a large amount of trace output.	NO Do not trace record level messages (default). YES Trace record level messages.

2. In the **Generate JCL to Load Accelerator** wizard in the Accelerator Loader studio, supply the group name.
3. Optional: Specify the number of members in the group. Although optional, this parameter is recommended. When this parameter is provided, the group is closed as soon as all members have joined. If the number is not provided, the group is not closed until the timeout expires. There is no default.
4. Optional: Specify a timeout value for the group formation. When the first group member request arrives at the Accelerator Loader server, the timer is started. If the group remains open when the request expires, it is closed. Any members/threads arriving after the timeout will be placed in a new group. The default is 60 seconds, and can be overridden in the *hlvidIN00* file.
5. Optional: Specify the number of I/O threads to use when reading the data source. If this value is not provided, the number of threads is determined as follows:
 - a. If the data source is a tape data set and the number of volumes can be determined, the same number of I/O threads will be started.
 - b. Otherwise, if a Map Reduce thread count is provided in the data map, that number is used.
 - c. Otherwise, if a value is configured for ACIMAPREDUCETASKS in the *hlvidIN00* configuration member, that number is used.
 - d. Otherwise, a single I/O thread will be started.