

System Recovery Boost



Contents

- Tables..... V**

- About the System Recovery Boost Content Solution..... vii**

- Chapter 1. Overview of System Recovery Boost..... 1**

- Chapter 2. Terminology.....3**
 - Speed Boost..... 3
 - zIIP Boost.....3
 - zIIP Boost with Reserved zIIPs..... 4
 - System Recovery Boost Upgrade Record..... 4
 - GDPS.....5

- Chapter 3. Planning and Setup..... 7**
 - Prerequisites..... 7
 - Turning on System Recovery Boost.....8
 - BOOST.....8

- Chapter 4. Using System Recovery Boost.....11**
 - Startup Boost..... 11
 - Shutdown Boost.....11
 - Recovery Process Boost..... 11
 - Automation Considerations..... 12
 - Performance Considerations..... 13

- Chapter 5. Configuring Processors Offline During a Boost Period..... 15**

- Chapter 6. WLM Changes..... 17**

- Chapter 7. SMF Changes..... 19**

- Chapter 8. Other System Changes..... 21**

- Chapter 9. Interaction of Shutdown Boost and PRESCPU..... 23**

- Chapter 10. Software License Charge and Chargeback Implications.....25**

- Chapter 11. Stand Alone Dump..... 27**

- Chapter 12. Messages.....29**
 - Messages.....29

- Chapter 13. MVS Data Areas..... 39**
 - CVT information..... 39
 - CVT mapping..... 39
 - ECVT information..... 39
 - ECVT mapping..... 39
 - IHAENF84 information..... 41

IHAENF84 programming interface information.....	41
IHAENF84 heading information.....	41
IHAENF84 mapping.....	42
IHAIPA information.....	44
IHAIPA mapping.....	44
IRALPDAT information.....	44
IRALPDAT mapping.....	44
IRAQVS information.....	44
IRAQVS mapping.....	44
IRARMCTZ information.....	44
IRARMCTZ mapping.....	44
IWMWRCAA information.....	45
IWMWRCAA mapping.....	45
Index.....	47

Tables

- 1. Structure CVTFIX.....39
- 2. Structure ECVT..... 39
- 3. Structure ENF84.....42
- 4. Cross Reference for IHAENF84..... 43
- 5. Structure IPA..... 44
- 6. Structure LPDATMAP..... 44
- 7. Structure QVS..... 44
- 8. Structure RMCTZ..... 44
- 9. Structure RCAEIHDR..... 45
- 10. Structure RCAAGDDE.....45

About the System Recovery Boost Content Solution

Purpose of this information

This is a collection of all of the information that you need to understand and utilize System Recovery Boost. Some of this information also appears elsewhere in the z/OS library.

Who should read this information

This collection of documentation is intended for the systems programmer, or persons responsible for planned system outages, and system recovery during an unplanned outage. This collection also provides information for someone who wants to minimize their system down time and decrease the time it takes to shutdown, restart, and process backlog.

Related information

For an interactive starting point, and access to a variety of resources related to System Recovery Boost, see [System Recovery Boost](#).

Chapter 1. Overview of System Recovery Boost

What is System Recovery Boost?

System Recovery Boost is a feature of the z15 that provides capabilities to reduce the time it takes to shutdown, restart, and process the backlog that accumulated during a system outage, by providing additional processor capacity and throughput for the boosted image.

How does System Recovery Boost work?

System Recovery Boost can deliver higher processor capacity for a limited time after an IPL, while shutting down a system, or during specific, short-term sysplex recovery events. The increased capacity can be provided by one or more of the following ways:

- In an LPAR on a sub-capacity machine, by utilizing the full speed for the general purpose processors of the boosted partition (speed boost)
- Dispatching some work on z Integrated Information Processors (zIIP), even if the work is not eligible for zIIP (zIIP boost)
- Configure online zIIP(s) that are in the standby state for the duration of the boost period (zIIP boost)

System Recovery Boost is a feature that is automatically enabled during startup, and can be manually enabled during shutdown once you have indicated that a shutdown is in progress.

The limited duration boost only applies to the images that are in a boost period, with the exception of System Recovery Boost upgrade record.

For related information on System Recovery Boost read the white paper: [System Recovery Boost for z15](#)

Chapter 2. Terminology

This chapter covers the terminology, concepts, and features you need to understand to best utilize the capabilities of System Recovery Boost.

What do you need to know?

An IBM Z CPC has a large number of processing units, or PUs, that can be configured to perform various tasks. For z/OS, there are two types of PUs that matter to System Recovery Boost, Central Processors, or CPs, that can run any type of work, and z Integrated Information Processors, or zIIPs, that run work that has been marked as zIIP eligible. The processing capacity of the CPC is denoted by its model number, which is of the form *XYZ*, such as 755. *X* denotes the speed of the CPs; model 7 denotes full speed CPs, while lower numbers denote a CP running at a reduced speed. CPCs with reduced speed CPs are known as subcapacity CPCs. The *YZ* portion of the model number is the number of entitled CPs available to the CPC, so in this case, 55. You can see your model number on the Product Information tab in the System Details task on the SE. The number of processors configured to the CPC are known as physical CPs or zIIPs.

The number of CPs and zIIPs that an image will use is configured in the Image Activation Profile task on the HMC. The CPs and zIIPs that you define here are known as logical CPs and logical zIIPs. You can have more logical CPs and zIIPs than you have physical CPs and zIIPs. The two fields that matter here are Initial zIIPs and Reserved zIIPs. Initial zIIPs are brought online during IPL; reserved zIIPs are defined to the image, but are not brought online at IPL time. There are two types of reserved zIIPs, ones that are backed by hardware, and therefore could be brought online, and ones that are defined to the image, but not backed by real hardware, and can't be brought online without acquiring additional physical zIIPs.

Speed Boost

What is Speed Boost?

Speed Boost is a capability of System Recovery Boost that improves the recovery time of exploiting operating systems when running on a subcapacity CPC.

How does Speed Boost work?

If you are running on a subcapacity CPC, then while System Recovery Boost is active, z/OS will request that the CPC firmware increase the speed of CPs being used by the image to full capacity model speed for the duration of the boost. After the boost ends, the CPs being used by the image will return to the subcapacity model speed. Since zIIPs always run at full speed, zIIP work will see no benefit from Speed Boost.

This increase applies only to the image being boosted; all other images not being boosted will run at subcapacity model speed, and the CPC itself will still remain a subcapacity CPC. If you're running on a subcapacity machine, IBM recommends activating Speed Boost, as all CP processes will benefit.

zIIP Boost

What is zIIP Boost?

zIIP Boost is a capability of System Recovery Boost that can improve z/OS recovery time, assuming zIIP capacity is available to the image.

How does zIIP Boost Work?

z/OS is the only operating system that can exploit the zIIP Boost capability, as it's the only OS that can natively exploit zIIPs. While zIIP Boost is active, z/OS will make non-zIIP eligible work zIIP eligible, thus allowing most work to run on zIIPs if there isn't sufficient CP capacity available. This provides additional capacity and parallelism to accelerate processing during the boost periods. IBM refers to this as blurring the CPs and zIIPs together.

There are variations of zIIP Boost, depending on how the CPC is configured. If you have no zIIPs, you will of course see no benefit from zIIP Boost. There's also the priced System Recovery Boost upgrade record feature that will allow you to take additional advantage of zIIP Boost.

The most basic type of zIIP Boost is when you have defined Initial zIIPs, and no Reserved zIIPs that could be backed by physical processors. In this case, the zIIP blurring will happen for the duration of the boost, on your defined, entitled zIIPs. At the end of the boost period, normally non-zIIP eligible work will stop running on zIIPs.

For the duration of the boost period, CP work will be eligible to run onto any online zIIP. This additional capacity for CP work provides additional processing capacity and parallelism.

zIIP Boost with Reserved zIIPs

In this case, you have some zIIPs that are defined as reserved, and some number of those can be backed by physical zIIPs. Those reserved zIIPs that can be backed by physical zIIP will be brought online by the system at the beginning of the boost period, and then taken offline when the boost period ends.

As above, any zIIPs brought online will provide even more capacity and parallelism during the boost period.

System Recovery Boost Upgrade Record

What is System Recovery Boost upgrade record?

This is a new priced feature for the z15 that requires you to have purchased an annual subscription for the Boost feature for your CPC that will allow you to bring up to 20 additional physical zIIPs online to the CPC for the duration of the capacity record activation, limited by the number of unused "dark core" processors you have in your CPC. The intention is that, when needed for recovery purposes, you can bring this additional zIIP capacity online to your CPC. These zIIPs will then behave just like any other physical zIIP, in that the processors will be shared across the CPC based on their LPAR weights, and additional reserved zIIPs will be able to be brought online at the beginning of a boost due to the additional physical capacity being made available. After the Temporary Capacity Record expires, or is deactivated, the additional zIIPs will be removed from the CPC's pool of physical processors. Any zIIP processors you want managed by z/OS as part of System Recovery Boost (ie, automatically brought on and offline) have to be available to the CPC when the Boost is started. Physical processors that are made available after the boost has started will not be automatically brought online by z/OS.

How does System Recovery Boost upgrade record work?

As an example, consider a CPC with 5 shared physical zIIPs and three LPARs, one with 0 initial zIIPs, one with 3 initial zIIPs and one with 5 initial zIIPs.

With 0 initial zIIPs, z/OS won't be able to take advantage of zIIP Boost unless the image has 1 or more reserved zIIPs defined. If there are one or more reserved zIIPs defined, during the boost period, up to 5 reserved zIIPs will be configured online by z/OS, and then configured offline once the boost period is over. If you have a temporary capacity record, then more zIIPs could be configured online during a boost period, up to the maximum allowed by the system definition or the physical number of zIIPs available.

With three initial zIIPs, z/OS will be able to take advantage of zIIP Boost. If one or more reserved zIIPs are defined, up to 2 reserved zIIPs will be configured online by z/OS at the beginning of the boost period, and then configured offline once the boost period is over. If you have a temporary capacity record, then more could be configured online during a boost period, up to the maximum allowed by the activation profile or the physical number of zIIPs available.

With five initial zIIPs, z/OS will be able to take advantage of zIIP Boost. However, unless you have a temporary capacity record, no more zIIPs will be able to be configured online by z/OS at the beginning of a boost period. If you have a temporary capacity record, then more could be configured online during a boost period, up to the maximum allowed by the activation profile or the physical number of zIIPs available.

What is GDPS Performance and Parallelism?

GDPS Performance and Parallelism, increases the speed with which GDPS drives hardware actions, along with enhancing the speed of the underlying hardware services.

The GDPS Enhancements are a combination of changes to GDPS processing/scripting itself, and GDPS taking advantage of new functionality being offered by the z15 Support Element. Various APIs used by GDPS on the SE are being streamlined and improved, while GDPS itself will be doing more in parallel when possible. In addition, GDPS will be enhancing its scripting language to allow multiple systems to be addressed in parallel by some commands, as opposed to sequentially as before, and making changes to some control flows to improve efficiency.

How do the GDPS enhancements work?

The GDPS enhancements are available in GDPS 4.2 and above, once running on a z15 CPC.

As part of GDPS 4.2 the system infrastructure management has been improved, allowing a reduction in the observed RTO when there is a need for the IT infrastructure to be reconfigured. This is the case in scenarios such as on-site system restart, planned site or region switch, or in case of a disaster recovery process. This has been achieved by optimizing GDPS & CPC interactions and by providing the possibility for customers to perform actions against a group of System/LPARs rather than against each one individually.

Before this change, all actions were done sequentially, repeating in each step the required handshaking between GDPS and the CPCs hosting the systems. Grouping the actions eliminates the duplicate handshaking processes and reduces substantially the overall time needed to manage multiple systems. Early testing of these changes has demonstrated an improvement of between 30% and 60%, depending on the numbers of CPCs, number of system/LPARs managed, and CPC models.

The following GDPS manuals describe how to implement these RTO improvements:

- *GDPS Metro Planning and Implementation Guide (ZG24-6755), section 4.7.2, SYSPLEX script statement.*
- *GDPS Global - GM Installation and Customization Guide (ZG24-6757), section 4.5.7, SYSPLEX script statement.*
- *GDPS Global - XRC Installation and Customization Guide (ZG24-6759), section 4.6.9, SYSPLEX script statement.*

Chapter 3. Planning and Setup

This chapter covers information on prerequisites and what you need to do for planning and setup of System Recovery Boost.

Prerequisites

System Recovery Boost requires a z15 processor, and is only supported on z/OS V2R3 and z/OS V2R4.

For zIIP Boost, you also need to have one or more zIIPs defined in the Image Activation Profile, either as initial or reserved processors, have physical zIIPs online in the CPC, have HiperDispatch enabled (the default), and be running with shared, not dedicated processors.

What do you need to do?

For the base support, you need a z15 and the appropriate PTFs. You need to make sure you aren't disabling the support in your IEASYSxx member. Without any additional work, you will have Startup Boost, and partial GDPS support.

For additional benefit, you may want to consider doing the following:

- If you use GDPS, you should look at your GDPS SYSPLEX script and use the new GDPS verbs that allow you to act against multiple images simultaneously.
- You may want to update your shutdown automation to automatically use the START IEASDBS system command during your shutdown process. If your shutdown takes 30 minutes or less, you should invoke it at the beginning; if it takes more than thirty minutes, you'll need to figure out at what point it will provide the most benefit to your shutdown. In general, the period with the highest CPU utilization will have the most benefit for activating shutdown boost.
- You may want to update your startup and shutdown automation in general, to take advantage of the additional parallelism or capacity that System Recovery Boost will provide. Note, the terms "startup" and "IPL" are used interchangeably when referring to boost processing.
- You may want to automate on the new System Recovery Boost Messages.
- You may want to define reserved zIIPs to some or all of your images to be able to access additional zIIP capacity for zIIP Boost, so they can be configured online during IPL and shutdown boosts.
- You may want to look at your zIIP weights, and see if they still match the usage by your images. If you find zIIPs are using significant Vertical Low capacity during IPL or shutdown boost, you may want to adjust zIIP weights during startup and shutdown to optimize performance. If significant expansion into Vertical Lows happens consistently independent of boost, consider making permanent zIIP weight changes. When significant expansion into Vertical Lows happens primarily during boost, consider transiently adjusting zIIP weights only during the boost period.
- In order to do CEC to CEC comparisons as part of the migration process, you may want to disable System Recovery Boost initially.
- You may want to copy the new procs, IEASDBS and IEABE from SYS1.PROCLIB to an appropriate proclib on your system. Please note that any ID with authority to run said procs will be able to start or end a boost; no special permission is required beyond that.

If you want to take advantage of System Recovery Boost upgrade record, you will need to enable two z15 Hardware Feature Codes, 6802 –System Recovery Boost Record, and 9930 –Boost Authorization. You may also want to consider doing the following:

- You may want to define more additional reserved zIIPs for images that are expected to use the additional recovery capacity.
- You may want to update your automation to activate the Boost Capacity Record activation and deactivation.

- You may want to automate changing zIIP weights during the period of time the Boost Capacity Record is active, in order to ensure the systems that are recovering have access to the additional capacity.
- You may want to review the tech doc, which explains how to configure a LPAR for best performance. <https://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TD106388>

Turning on System Recovery Boost

System Recovery Boost is activated by default, and can be controlled via the BOOST= parameter in your IEASYSxx parmlib member. The default, BOOST=SYSTEM, means that z/OS will take advantage of whatever boost(s) are available to it, given the system configuration. BOOST=ZIIP will only allow for zIIP Boosting, while BOOST=SPEED will only allow for Speed Boost. BOOST=NONE will disable all types of boosting. (See below for more information on the BOOST parameter.)

The system command, DISPLAY IPLINFO,BOOST can be used to view the value of the BOOST system parameter. The system command, DISPLAY IPLINFO,BOOST,STATE can be used to display information about the boosts. There are also indicators on the HMC that can tell you if an image is currently boosting or not; the Image Detail panel will tell you if an image is boosting, and there's a new column in the Systems Management list and Custom Groups list that will indicate if a boost is active, but not which type(s).

System Recovery Boost can be used twice during an IPL, for each image: once during IPL, and once while shutting down the system.

Startup Boost will automatically be activated during IPL, and last for 1 hour, but you start Shutdown Boost by using the START IEASDBS command. For more information on each of these topics, see [“Startup Boost” on page 11](#), and [“Shutdown Boost” on page 11](#).

BOOST

BOOST={SYSTEM| ZIIP | SPEED | NO[NE]}

Two boosts are possible: zIIP boost and speed boost. These boosts are available when running z/OS® on a machine with IBM z15™ capabilities only.

The zIIP boost configures online zIIPs in the standby state in accordance with the zIIP configuration of the LPARs. The zIIP boost will "blur" work between standard CPs and zIIPs. (That is, work that is normally eligible only for a standard CP may be run on a zIIP.) zIIP boost only applies to an LPAR that, at the time of boost activation, meets all of the following criteria:

- Has at least one online or standby zIIP
- Is not a dedicated partition
- Is running with HIPERDISPATCH=YES

In order for zIIPs to be configured online during boost processing, the zIIPs must be defined to the LPAR. They may be defined before IPL or they may be dynamically added to the configuration after IPL, depending on the LOADxx DYNCPADD parameter value in effect. zIIPs dynamically added after IPL do not apply to IPL boost but could be used by shutdown boost.

The speed boost makes a sub-capacity standard CP run at the speed of a (full-speed) zIIP. Speed boost only applies to an LPAR that, at the time of boost activation, does not have its standard CP(s) already running at the (full-capacity) speed of a zIIP.

Boost start is evaluated at three points:

- At IPL
- At the requested start of a recovery process boost
- At shutdown, as identified by using the command START IEASDBS

SYSTEM

Apply system rules. If the machine does not support boosts, treat as NONE. If the system does support boosts, activate all boosts that the machine supports.

ZIIP

Activate only the zIIP boost. Do not activate the other (speed) boost.

SPEED

Activate only the speed boost. Do not activate the other (zIIP) boost.

NONE | NO

Do not activate either of the boosts.

The specification may be displayed using the DISPLAY IPLINFO,BOOST command.

Value range: Not applicable

Default: BOOST=SYSTEM

Associated parmlib member: None

Chapter 4. Using System Recovery Boost

This chapter contains information on using System Recovery Boost.

Startup Boost

Startup Boost is controlled by BOOST= in IEASYSxx. If you haven't disabled System Recovery Boost in parmlib, Startup Boost will automatically be activated during IPL, and last for 1 hour. If for some reason you want to stop the startup boost prior to the 1 hour time limit, you can issue system command START IEABE (IEA Boost End) to do so. The invoked module will likely complete before the (asynchronous) stopping of the boost completes.

During Startup Boost, WLM Routing Services will specify the correct, boosted capacity for most of the duration of the boost. A few minutes before the scheduled end of the boost, WLM will start reducing the routing weight, so that by the end of the boost, the routing weight will match the regular capacity of the image. The interval of time this decrease in routing weight happens over can be influenced via the RTBELEADTIME option in IEAOPTxx.

For more information about the BOOST parameter of IEASYSxx, see [“Turning on System Recovery Boost” on page 8](#).

Shutdown Boost

Shutdown Boost is controlled by BOOST= in IEASYSxx, just as Startup Boost is. System Automation will start IEASDBS when it takes a system down. So if you have disabled one or more varieties of System Recovery Boost in IEASYSxx, you won't be able to use it for Shutdown Boost either. Shutdown Boost will last for 30 minutes, or until the system stops. So if the system stops 15 minutes after you start Shutdown Boost, the remaining 15 minutes doesn't carry over to the following IPL.

To use System Recovery Boost to speed up shutdown, you start the proc IEASDBS (IEA Shut Down Boost Start) to tell z/OS that a shutdown is in progress and to begin the shutdown boost; note that once you have indicated that a shutdown is starting, WLM will essentially stop directing new work to that image, and will stop performing weight balancing activity for the image. If for some reason you want to stop a shutdown boost prior to the 30 minute time limit, you can run proc IEABE to do so. However, stopping Shutdown Boost using the IEABE proc will not cause WLM to resume normal behavior, nor will it allow you to use any remaining boost time in the future. IEASDBS and IEABE will run very quickly, and will stop as soon as finished.

Recovery Process Boost

In addition to Startup boost and Shutdown boost, Recovery Process (RP) boost is a boost class that can help accelerate system recovery. Recovery Process boost provides short-term acceleration for specific recovery events in z/OS. The applicable recovery events for Recovery Process boosts are limited to:

1. **HyperSwap®**

Boost all systems participating in a Hyperswap process. HyperSwap processing is a coordinated, sysplex-wide recovery process that restores access to DASD devices following the failure of a storage controller. Its recovery time is sometimes limited by slow processing on one or more participating systems.

2. **Coupling Facility data-sharing member recovery**

Boost all systems participating in recovery from termination of a CF data-sharing member. When a data-sharing member fails, the other surviving members have to do a lot of recovery/cleanup processing to free up locks and other data-sharing resources held by the failed member.

3. **Coupling Facility structure recovery**

Boost all systems participating in CF structure recovery processing, such as: CF structure rebuild, duplexing failover, and re-duplexing. Recovering failed CF structures and their data can be a process that requires the participation of all systems that were using those CF structures, and can apply to many structures in cases like loss of a CF image.

4. Sysplex partitioning

Boost all surviving systems in the sysplex as they take on the additional workload of sysplex partitioning related recovery, after planned or unplanned removal of a system from the sysplex. When a system in the sysplex is removed, the surviving systems have to do more recovery processing to clean up after the failed system and free up resources that were held on the failed system.

Recovery Process boost applies only to the z15. LPAR firmware in z15 T01 and T02 provides the support for the short-duration Recovery Process boosts. Recovery Process boost periods are restricted to 5 minute durations and are limited to 30 total minutes per partition per 24 hour period.

Recovery Process boosts are managed internally by z/OS, with the operating system initiating the boosts as these recovery events take place, and only on the images that are actually participating in or affected by these events. Recovery Process boost periods are started and ended solely by z/OS operating system controlled events. When initiating the RP boost, z/OS initiates whichever boost types (speed boost, zIIP boost, or both) are identified by the BOOST= system parameter currently in effect for the image. During Recovery Process boost periods, any available program within the boosting partition may be executed on a zIIP engine as defined and controlled by IBM's technological measures, when zIIP boost is active.

Recovery events can "overlap" when a second Recovery Process boost occurs before a first one has used its entire boost period. If this happens, then the overlapping boosts are merged and z/OS will extend the boost period to allow the full boost period duration for the overlapping recovery process. During a Recovery Process boost period, WLM neither routes work away from the system (as it does during shutdown boost) nor towards the system (as it does during startup boost). Instead, WLM essentially ignores short-duration recovery boosts for workload routing purposes, because they are too short-term.

Additionally, there is "boost class" for Recovery Process boost that appears in various system messages, ENF signals, SMF fields, and other z/OS APIs. For more information please refer to:

["Messages" on page 29](#)

[Chapter 7, "SMF Changes," on page 19](#)

[Chapter 6, "WLM Changes," on page 17](#)

[Chapter 8, "Other System Changes," on page 21](#)

Automation Considerations

The following new messages support the Sysplex Recovery Boost enhancements:

- IEA682I Recovery process zIIP boost is active
- IEA683I Unable to complete activation of recovery process boosts
- IEA684I Recovery process boost(s) have ended due to error
- IEA685I Recovery process boost(s) not {started | extended}. Duration limit has been reached
- IEA686I Recovery process boost(s) have been extended
- IEA687I Recovery process boost requestor: <string>
- IEE257I Boost State Boost class: {IPL | Shutdown | Recovery process | N/A} [Requestor: <string>] zIIP boost: {active with n transient zIIP cores | inactive} speed boost: {active | inactive}

The following changed messages have updates specific to Sysplex Recovery Boost enhancements:

- IEA671I Could not fully configure online <m>of the <n> intended {IPL | shutdown | recovery process} zIIP boost cores
- IEA672I {IPL | Shutdown | Recovery process} {zIIP | Speed} boost is not activated –rsn
- IEA674I Could not configure offline <m> of the <n> {IPL | Shutdown | recoveryprocess} zIIP boost cores

- IEA675I {IPL | Shutdown | Recovery process} zIIP boost is active with <n> transient zIIP cor
- IEA678I All {IPL | Shutdown | Recovery process} boosts have ended
- IEA681I {IPL | Shutdown | Recovery process} speed boost is active
- IEE174I (form 35 of 35) hh.mm.ss DISPLAY M [idr] PROCESSOR STATUS text

Messages introduced with this functionality:

- IEA676I No IPL boosts will be activated
- IEA677I {IPL | Shutdown}{zIIP | speed} boost has ended due to error
- IEA680I {zIIP | speed} boost is not available
- IWM063I WLM POLICY WAS REFRESHED

IWM064I new reasons:

- IWM064I BOOST ACTIVATED
IPL or Shutdown Boost has been activated
- IWM064I BOOST ENDED
IPL or Shutdown Boost has ended

If you want to automate something after all IPL Boosts have ended, you can automate on the messages IEA676I and IEA678I, each of which indicates one of the two cases of “no boosts are still active”. Note that these might have been issued before the automation program has started, in which case the automation program would need to be configured to look for earlier messages.

Some types of automation you may wish to consider:

- Activating the Temporary Capacity Record prior to shutdown, and then deactivating it after the recovery is complete.
- Dynamically changing LPAR weights as required during a shutdown or startup boost, to better align guaranteed capacity and consumed capacity to mitigate boosted image(s) from expanding into white space capacity by using vertical lows.
- Add the starting of the IEASDBS proc to your existing shutdown automation.
- Changing the level of parallelism present in the workload at startup (e.g. starting middleware regions) and shutdown (e.g. doing orderly shutdown of middleware). Automation may pace/throttle these activities to varying degrees; with Boost, less pacing / more parallelism may be desirable.
- To automate on the new z/OS messages that are issued at start/end of boost periods to take whatever actions are appropriate

IBM System Automation or any ISV vendor automation product can be used to provide these automation capabilities.

Performance Considerations

For Speed Boost alone, the increase in processor speed is visible only to the LPAR being boosted, so there’s no direct impact to other images on the CPC. However, because the boosted image will be running more work than is usually possible during the boost period, relative to other images on the CPC, it is possible that the boosted image will impact the data caches more than it would during non-boosted periods, leading to performance decreases on non-boosted images it is sharing data caches with.

However, if you are taking advantage of zIIP Boost, either by itself or in conjunction with Speed Boost, you may need to think about the impact zIIP Boost will have on other LPARs in the CPC.

Chapter 5. Configuring Processors Offline During a Boost Period

The CONFIG command should not be used to configure a zIIP offline while the zIIP boost is active in order not to inadvertently configure offline processors that are part of the zIIP boost (at the start of zIIP boost the system configures online the standby zIIPs and at the end of zIIP boost the system configures offline the zIIPs that it had configured online).

Chapter 6. WLM Changes

While the boost is in effect, WLM can manage the system slightly differently than at regular non-boosted periods:

- Overachieving non-discretionary work will not be capped to allow for higher throughput of discretionary work; that is, during the boost periods the system behaves as if Deactivate Discretionary Goal Management had been defined as Yes in the service definition.
- Resource groups and tenant resource groups will not be capped on boosted systems. For resource groups with a sysplex-wide limit, the non-boosted systems will cap the resource group as if no consumption had taken place on the boosted systems.
- The system long-term average consumption is based on the general purpose processor consumption assuming standard processor rating. Consequently both zIIP boost and speed boost will not directly contribute to the four-hour rolling average MSU consumption.
- CPU protection is assigned to critical work. That is, any single-period service class of importance 1 or 2 is implicitly set to CPU Critical which ensures that less important work will have a lower dispatching priority. When the boost periods ends, the CPU Critical option is reset to the state as defined in the currently active WLM policy.

Any defined capacity and group capacity limits remain in effect:

- For IPL boost, the WLM Sysplex Routing services may start reducing the routing weight as the end of the IPL boost period approaches and before the system returns to its regular capacity. When required, the lead time can be influenced via the RTBELEADTIME parameter in the IEAOPTxx member. When shutdown boost begins, the boosted system may be eliminated from the routing recommendations, or a very small weight may be assigned. Recovery boost periods have no impact on any routing recommendations; they are too short-term.
- Intelligent Resource Director (IRD) will not take weight management actions for LPARs that are being boosted.

During a Recovery Process boost period, WLM neither routes work away from the system (as it does during shutdown boost) nor towards the system (as it does during startup boost). Instead, WLM essentially ignores short-duration recovery boosts for workload routing purposes, because they are too short-term.

Chapter 7. SMF Changes

A new SMF interval will be started whenever a boost starts or ends. SMF 90-40 is a new subtype, and will be generated when a boost event occurs (starting/stopping boost), and will contain information about the boost.

SMF types 30, 70-1, 72-3, 89-1, 89-2, and 99-1 are all getting a boost information byte that will tell you whether a boost is active, and if so, which type. During the boost, SMF records that contain capacity information, such as 30, 70-1, 72-3, and 89, will reflect the unboosted capacity of the image, not the effective boosted capacity.

For more information on SMF see:

Chapter 8. Other System Changes

There is a new ENF type, ENF 84, that will signal when a boost is beginning or ending, along with the boost type. ENF 37 is updated to signal that a new SMF interval is happening due to a boost starting or stopping.

Chapter 9. Interaction of Shutdown Boost and PRESCPU

By default, during IPL, z/OS will configure the logical processors to match what is in the activation profile, so active CPs/zIIPs will be online, while any reserved CPs/zIIPs will be offline.

PRESCPU is an IEASYSxx option that allows you to persist any logical processor changes you've made to an image across IPL boundaries. With PRESCPU enabled, z/OS won't make any changes to the online or offline CPs/zIIPs during IPL, so they will match whatever was configured when the system went down. However, this means there's a potential interaction with a zIIP shutdown boost. If you start a shutdown boost, have not disabled zIIP boost, and you have reserved zIIPs that have physical backing, those zIIPs will be brought online to assist with shutdown processing. If the system is then stopped prior to the end of the shutdown boost, those zIIPs will be still configured online.

So when the system is then IPLed, if PRESCPU is configured, those zIIPs will be online from the beginning, not brought online by zIIP Boost. Therefore, once the boost ends, they won't automatically be brought offline. So you may end up with more zIIPs online than you'd prefer.

If you are currently using PRESCPU, you should determine if it is something you wish to continue using in light of the above behavior. If you do decide to continue using it, you may wish to update your automation to end boosts by issuing the START IEABE system command to end the boosts before shutdown completes.

Chapter 10. Software License Charge and Chargeback Implications

System Recovery Boost is designed to cause no impact to your IBM software licensing charges. During the boost period, WLM and SRM will record the unboosted capacity of the image in all billing-related records. For clients who are using subcapacity licensing, the pricing records will not record an increase in the available capacity, and thus will not increase your 4-hour rolling average. For clients paying for MSU consumption pricing, consumption on GP processors accrues as normal, but there is no net increase in consumption, as any work being done during boost would have been done before, just more slowly. For clients paying for full capacity, the total recorded capacity of the CEC doesn't change, and thus the bill will remain the same.

Chapter 11. Stand Alone Dump

Stand Alone Dump is single-threaded, and unable to use zIIP engines. As such, it will only utilize the Speed Boost functionality. How much of an improvement you may see is hard to say; a lot of the work Stand Alone Dump does involves writing data to DASD or tape, and that isn't impacted by the speed of the CPU.

Note that in production, SAD will take advantage of Speed Boost by default, once PTF OA57849 is installed and SADMP is regenerated.

Chapter 12. Messages

This chapter describes the messages that are affected by System Recovery Boost.

Messages

IEA671I **Could not fully configure online *m* of the *n* intended {IPL | shutdown | recovery process} zIIP boost cores**

Explanation

The system intended to configure online *n* zIIP boost cores but was unable to do so. One or more cores either was not configured online at all or did not have the intended number of threads configured online.

System action

The system continues processing. The zIIP boost might not be as effective as intended.

Operator response

None.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2, 10

Descriptor code

None.

IEA672I **{IPL | Shutdown | Recovery process} {zIIP | SPEED} boost is not activated - *rsn***

Explanation

Via the BOOST=ZIIP or BOOST=SPEED system parameter, the customer asked for a specific boost, but the machine does not support this boost. Within the message text, *rsn* is one of the following:

unable to identify zIIPs

For the zIIP boost, the system was unable to determine which zIIPs to configure online for the zIIP boost.

on a dedicated logical partition

The zIIP boost requires a shared logical partition.

no zIIPs

For the zIIP boost, there are no zIIPs configured online or available to be configured online.

not HIPERDISPATCH=YES

The zIIP boost requires HIPERDISPATCH=YES (via the IEAOPTxx parmlib member).

unable to complete activation

The process used to activate the boost did not complete successfully.

processor is already full speed

For the speed boost, the standard CP is already full speed.

System action

The system continues, with the specified boost not active.

Operator response

None.

Source

Supervisor Control.

Module

IEAVNP09, IEAVEMMT

Routing code

2, 10

Descriptor code

12 (IEAVNP09)

IEA674I **Could not configure offline *m* of the *n* {IPL | shutdown | recovery process} zIIP boost cores**

Explanation

Could not configure offline *m* of the *n* {IPL | Shutdown} zIIP boost cores.

n

The system intended to configure offline *n* zIIP boost cores but was unable to do so. One or more cores was not configured offline.

System action

The system continues.

Operator response

None.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2, 10

Descriptor code

None.

IEA675I {IPL | shutdown | Recovery process} zIIP boost is active with *n* transient zIIP cores

Explanation

The zIIP boost has been activated. 0 or more zIIP cores (indicated by the fill-in *n*) were configured online as part of the boost processing and will be configured offline once the boost ends.

System action

The system continues.

Operator response

None needed. If the customer is using the PRESCPU option and their previous IPL ended while the boost was active, their next IPL would consider these cores to be part of the normal configuration. Thus, the customer might choose to configure offline this number of zIIPs at completion of the next IPL to get back to the desired configuration.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2

Descriptor code

None.

IEA676I No IPL boosts will be activated

Explanation

Either the machine does not have the capability to support a boost or the customer has requested via the BOOST=NONE system parameter not to activate any boost or no IPL boost could be activated.

System action

The system continues.

Operator response

None needed. This message, along with message IEA678I, can be automated upon if there is a need to delay processing until there cannot be any IPL boosts still active.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2

Descriptor code

None.

IEA677I {IPL | shutdown} {zIIP | speed} boost has ended due to error

Explanation

An error occurred and the system needed to end the identified boost.

System action

The boost ends and the system continues.

Operator response

None.

System programmer response

Report the problem to IBM.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2, 10

Descriptor code

None.

IEA678I All {IPL | shutdown | recovery process} boosts have ended
Explanation

All boosts that were active have ended.

System action

The system continues.

Operator response

None needed.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2

Descriptor code

None.

IEA680I {zIIP | Speed} boost is not available
Explanation

Via the BOOST=ZIIP or BOOST=SPEED system parameter, the customer asked for a specific boost, but the machine does not support this boost.

System action

The system prompts for a re-specification of the BOOST system parameter.

Operator response

None.

Source

Supervisor Control.

Module

IEAVNP09

Routing code

2, 10

Descriptor code

12

IEA681I {IPL | shutdown | Recovery process} speed boost is active
Explanation

The speed boost is now active.

System action

The system continues processing.

Operator response

None needed.

Source

Supervisor Control.

Module

IEAVNP09, IEAVEMMT

Routing code

2, 10

Descriptor code

None.

IEA682I Recovery process zIIP boost is active
Explanation

The recovery process zIIP boost is now active.

System action

The system continues processing.

Operator response

None needed.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2

Descriptor code

None.

System action

The system continues processing.

Operator response

None needed.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2, 10

Descriptor code

None.

**IEA683I Unable to complete activation of
 recovery process boosts**
Explanation

Could not activate recovery process boost cores.

System action

The system continues processing.

Operator response

None needed.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2, 10

Descriptor code

None.

**IEA685I Recovery process boost(s) not
 {started | extended}. Duration
 limit has been reached**
Explanation

The duration limit for recovery process boosts has been exceeded. No further recovery process boosts will be started or extended until the pool of recovery boost time is replenished.

System action

The system continues processing.

Operator response

None needed.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2, 10

Descriptor code

None.

**IEA684I Recovery process boost(s) have
 ended due to error**
Explanation

An error occurred and the system needed to end the identified boost(s).

**IEA686I Recovery process boost(s) have
 been extended**

Explanation

A new recovery process boost request was received while recovery process boost was already active. The boost end time is extended by an amount specific to the requestor.

System action

The system continues processing.

Operator response

None needed.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2, 10

Descriptor code

None.

IEA687I Recovery process boost requestor: <string>
Explanation

Displayed after recovery process boost start (IEA675I and/or IEA681I) and recovery process boost extend (IEA686I).

In the message text:

<string>

The requestor shown is the requestor associated with the “stop ETOD” for the boost event (so could be the requestor for “start” or the requestor for “extend”). The potential values for <string> are:

- Not identified
- Sysplex Partitioning
- CF Structure Recovery
- CF Data-sharing Member Recovery
- Hyperswap

System action

The system continues processing.

Operator response

None needed.

Source

Supervisor Control.

Module

IEAVEMMT

Routing code

2, 10

Descriptor code

None.

IEE174I (form 35 of 35) hh.mm.ss DISPLAY M [idr] PROCESSOR STATUS text
Explanation

Where text is:

```
CORE STATUS: HD=h MT=mm [MODE: CP=cn
zIIP=zin]
```

```
ID ST ID RANGE VP ISCM CPU THREAD
STATUS
cId wst tIdL-tIdH vp xxxx tt
cId wst tIdL-tIdH vp xxxx tt
cId wst tIdL-tIdH vp xxxx tt
CPC ND = tttttt.nnn.mmm.pp.ssssssssss
```

```
CPC SI =
ttt.nnnnnnnnnnnnnnnnnn.mmmmmmmmmmmmmmmmm.pppp.ssss
sssssssssss
```

```
Model: oooooooooooooooooo
```

```
CPC ID = ii
CPC NAME = cpc_name
CPC IS RUNNING AT dd% OF NORMAL EFFECTIVENESS
LP NAME = lp_na LP ID = jj
CSS ID = cc
MIF ID = m
```

```
+ ONLINE - OFFLINE W WLM-MANAGED
N NOT AVAILABLE / MIXED STATE
```

```
[A APPLICATION ASSIST PROCESSOR]
[I INTEGRATED INFORMATION PROCESSOR
(zIIP)]
[B BOOST (TRANSIENT) zIIP]
CPC ND CENTRAL PROCESSING COMPLEX NODE
DESCRIPTOR
CPC SI SYSTEM INFORMATION FROM STSI INSTRUCTION
CPC ID CENTRAL PROCESSING COMPLEX IDENTIFIER
CPC NAME CENTRAL PROCESSING COMPLEX NAME
LP NAME LOGICAL PARTITION NAME
LP ID LOGICAL PARTITION IDENTIFIER
CSS ID CHANNEL SUBSYSTEM IDENTIFIER
MIF ID MULTIPLE IMAGE FACILITY IMAGE IDENTIFIER
```

A reconfiguration display request (a DISPLAY M=CORE command or an internal request) asked the system to display the status of selected cores. When you issue the DISPLAY M=CORE command from a PR/SM partition, the system displays the status for the logical cores. The system information presented on the CPC SI line reflects any dynamic upgrades that have been performed. The information presented on the CPC ND

and CPC ID lines remain unchanged across any such upgrades.

In the message text:

h

Indicates whether the system is enabled for HIPERDISPATCH (Y) or not (N).

mm

The maximum MT mode that can be supported by both the hardware and this IPL of z/OS.

cn

The MT mode for standard CP cores. This can be modified by using the MT_CP_MODE parameter in the IEAOPTxx member. For more information, see IEAOPTxx MT_CP_MODE in .

zin

The MT mode for IBM Z Integrated Information Processors. This can be modified by using the MT_ZIIP_MODE parameter in the IEAOPTxx member. For more information, see IEAOPTxx MT_ZIIP_MODE in .

cId

The core identifier.

wst

The state of the core, with each letter representing a different state:

w

When the core is being managed by WLM, this will be "W", otherwise it will be blank.

s

The status of the core, which can be one of the following:

- + indicates the core is online.
- - indicates the core is offline.
- / indicates the core is in a mixed state, because the core is online but the threads currently online are unexpected with the MT mode for this core type. This state can occur in certain system abnormal conditions. You can generally get out of this state by reconfiguring the core to its desired state. For more information, see IEAOPTxx MT_xxxx_MODE in .
- N indicates the core is neither online nor able to be configured online.

t

The type of the core, which can be one of the following:

- (a blank) – the core is a standard CP.
- A – The core is an IBM Z Application Assist Processor.

- I – The core is an IBM Z Integrated Information Processor.
- B – This is a zIIP added due to boost processing.

tIdL

The lowest CPU address of a thread that belongs to this core.

tIdH

The highest CPU address of a thread that belongs to this core.

v

The HIPERDISPATCH vertical-ness of the core:

- H – The core is classified as a vertical high core.
- M – The core is classified as a vertical medium core.
- L – The core is classified as a vertical low core.

p

Indicates whether the core is parked (P) or unparked (blank).

xxxx

The interrupt subclass mask for the core.

t

The state of each thread that belongs to this core. There are n visible thread states, where $n = (tIdH - tIdL) + 1$ and is OS and model dependent. The CPU ID for the first thread is $tIdL$. The state can be one of the following:

- + indicates the thread is online due to the current MT mode for this core type.
- - indicates the thread is offline due to the current MT mode for this core type.
- N indicates the thread is offline and expected to be offline due to the current MT mode for this core type. For more information, see IEAOPTxx MT_xxxx_MODE in .

CPC ND = ttttt.nnn.mmm.pp.sssssssss

The central processing complex node descriptor, in EBCDIC, where:

ttttt

The type number.

nnn

The model number of the processor at the last IML/POR.

Note: If the processor is concurrently upgraded to a new model, this number will not reflect the new model until the next processor IML/POR.

mmm

The manufacturer.

pp

The plant of manufacture.

ssssssssss

The sequence number.

dd

The approximate percentage, in decimal, of the actual effectiveness compared to the normal effectiveness of the CPC.

Indicates that a reduction in the current capability of the central processing complex (CPC) has been detected.

CPC SI = text02

Where *text02* is:

```
tttt.nnnnnnnnnnnnnnnn.mmmmmmmmmmmmmmm.pppp.s
ssssssssssssss
Model: 00000000000000
```

This is the system information from the STSI instruction, in EBCDIC, where:

tttt

The type number.

nnnnnnnnnnnnnnnn

The model-capacity identifier.

Note: The model-capacity identifier is updated after a concurrent upgrade.

mmmmmmmmmmmmmmmm

The manufacturer.

pppp

The plant of manufacture.

ssssssssssssss

The sequence code.

00000000000000

The model number.

Note: If the core is concurrently upgraded to a new model, the model number in field 00000000000000 will reflect the new model without a core IML/POR.

Note: The "Model:" line is displayed only if the model number 00000000000000 differs from the model number/model-capacity identifier nnnnnnnnnnnnnnnn.

CPC ID = ii

The central processing complex node identifier, in binary.

CPC NAME = cpc_name

The central processing complex name, in EBCDIC.

LP NAME = lp_na

The logical partition name, in EBCDIC.

LP ID = jj

The logical partition identifier, in binary.

CSS ID = cc

The logical channel subsystem identifier, in binary.

MIF ID = m

The multiple image facility identifier, in binary. The MIF ID appears only if you are running in LPAR mode.

System action

None.

Operator response

None.

System programmer response

None.

Module

IEEMPDM

Routing code

-

Descriptor code

4

IEE257I

Boost State

Boost class: {IPL | Shutdown | Recovery process | N/A}
[Requestor: <string>
zIIP boost: {active with n transient zIIP cores | inactive}
speed boost: {active | inactive}

Explanation

In response to DISPLAY IPLINFO,BOOST,STATE, this shows the state of the boosts.

In the message text:

N/A for Not Applicable

This is shown for the boost class when no boost is active.

<string>

Requestor: <string> is displayed only for Recovery Process boost. The requestor shown is the requestor associated with the "stop ETOD" for the boost event (so could be the requestor for "start" or the requestor for "extend"). The potential values for <string> are:

- Not identified
- Sysplex Partitioning
- CF Structure Recovery

- CF Data-sharing Member Recovery
- Hyperswap

n

Within “with *n* transient zIIP cores”, *n* is the number of cores.

System action

The system continues processing.

Source

Master scheduler

Module

IEECB985

Routing code

*

Descriptor code

5

IRA867I HIPERDISPATCH=NO CURRENTLY NOT ALLOWED

Explanation:

HIPERDISPATCH=NO because HiperDispatch mode is not allowed while zIIP Boost is active.

System action:

The system switched out of the HiperDispatch mode or no longer continues to run in HiperDispatch mode because zIIP Boost is active.

Operator response:

None.

System programmer response:

None.

Source:

System resources manager (SRM)

Module:

IRABAINI

IWM063I WLM POLICY WAS REFRESHED.

Explanation

WLM refreshed the active policy because of processor speed change, MT mode change, or boost status change.

The change might be expected, as in the case of a dynamic CPU capacity change or MT mode change for a processor class, or unexpected in the case where a processor goes into cycle steering mode. Beginning with z196, message IWM063I is accompanied by

message IWM064I that explains the reason for the policy refresh. Reporting products may close their reporting interval.

System action

Processing continues. Performance monitors may end their current and start new reporting interval.

Operator response

None, if this is because of a capacity or MT mode change. When the change is unexpected, contact hardware support and refer to the HMC for additional information.

System programmer response

None.

Source

Workload Manager (WLM)

Module

IWMP2RAC

Routing code

2,10

Descriptor code

4

IWM064I text

Explanation

This message follows message IWM063I whenever SRM is notified of a processor speed change. Beginning with the IBM z13[®], this message also follows message IWM063I whenever SRM is notified of an MT mode change for a processor class. Message IWM064I explains the reason for the change reported by message IWM063I. Message IWM064I will also be issued when the system is IPLed at reduced speed on z196, or follow-on machines.

In the message text: *text* indicates the reason for the change. Possible reasons are:

- THERE WAS A CHANGE OF THE MT MODE FOR A PROCESSOR CLASS.

This message is preceded by message IWM066I, which shows the affected processor class and the change in MT mode.

- THE SYSTEM IS RUNNING AT NOMINAL CAPACITY.

For example, a cooling problem has been resolved and the system is running at nominal capacity again.

- THE SYSTEM IS RUNNING AT NOMINAL CAPACITY; MODEL CONVERSION OCCURRED.

For example, the user may have upgraded or downgraded the capacity and the system is now running at the new nominal capacity.

- AN MSU TABLE UPDATE OCCURRED.

For example, the system's MSU table was updated. In this case, a second IWM064I message follows to report the actual state.

- THE SYSTEM IS RUNNING WITH REDUCED CAPACITY BECAUSE OF A MANUAL CONTROL SETTING.

For example, the user may have initiated Power[®] Saving Mode.

- THE SYSTEM IS RUNNING WITH REDUCED CAPACITY BECAUSE OF A MACHINE EXCEPTION CONDITION.

For example, there may have been a cooling problem.

- THE SYSTEM IS RUNNING WITH REDUCED CAPACITY BECAUSE OF A NON-EXCEPTION MACHINE CONDITION.

For example, there may have been a firmware update.

- THE SYSTEM IS RUNNING WITH REDUCED CAPACITY BECAUSE OF AN EXCEPTION CONDITION EXTERNAL TO THE MACHINE.

For example, the ambient temperature may have exceeded the maximum value.

- BOOST ACTIVATED.

IPL, recovery process, or shutdown boost has been activated.

- BOOST ENDED.

IPL, recovery process, or shutdown boost has ended.

As long as the system is running at reduced capacity, IWM068I is repeated every hour. When IWM064I has been issued at IPL and the system is running at reduced capacity, the first IWM068I message follows after 15 minutes. This gives automation enough time to initialize and react to the message when IWM064I is issued during IPL.

System action

Processing continues.

Operator response

None.

System programmer response

None.

Source

Workload Manager (WLM)

Module

IWMP2RAC, IWMI2JST

Routing code

2,10

Descriptor code

4

Chapter 13. MVS Data Areas

This chapter contains information on the data areas: CVT, ECVT, IHAENF84, IHAIPA, IRALPDAT, IRAQVS, and IRARMCTZ. Note: Each of these areas only contains the relevant portions of each data area that are affected by the APAR, and the complete content for each of these data areas is much larger.

CVT information

CVT mapping

Table 1. Structure CVTFIX

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
1272	(4F8)	BITSTRING	1	CVTOSLV8	BYTE 8 OF CVTOSLV8
	1..		CVTB00ST	"X'04'" Support for B00ST system parameter is available

ECVT information

ECVT mapping

Table 2. Structure ECVT

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
992	(3E0)	BITSTRING	24	ECVT_BOOSTINFO(0)	Valid only when CVTBoost is on
992	(3E0)	BITSTRING	1	ECVT_BOOSTINFO_FLAGS0	
		1...		ECVT_ZIIPB00ST_ACTIVE	"X'80'"
		.1..		ECVT_SPEEDB00ST_ACTIVE	"X'40'"
		..1.		ECVT_IPLB00STS_ACTIVATED	"X'20'" All IPL boosts to be activated have been activated.
		...1		ECVT_SDB00STS_ACTIVATED	"X'10'" All Shutdown boosts to be activated have been activated.
	 1...		ECVT_RPB00STS_ACTIVATED	"X'08'" All RP boosts to be activated have been activated. This bit will be turned off once the boost ends.
	111		ECVT_BOOSTCLASS	"X'07'" See EcvT_BoostClass_xxx equates. Valid only when one of the boosts is active
993	(3E1)	BITSTRING	1	ECVT_BOOSTINFO_SYSPARM_FLAGS	
		1...		ECVT_SYSPARM_ZIIPB00ST	"X'80'" According to the availability and the B00ST system parameter, we want to do ZIIP boost. This gets turned off if it could never be right to activate, such as for the case of a non-dedicated partition
		.1..		ECVT_SYSPARM_SPEEDB00ST	"X'40'" According to the availability and the B00ST system parameter, we want to do speed boost.
994	(3E2)	BITSTRING	1	ECVT_BOOSTINFO_FLAGS1	
		1...		ECVT_IPLZIIPB00ST_ENDEDBYERROR	"X'80'"
		.1..		ECVT_IPLSPEEDB00ST_ENDEDBYERROR	"X'40'"
	 1...		ECVT_IPLB00STS_ENDEDBYTIMER	

Table 2. Structure ECVT (continued)

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
					"X'08'"
	1..		ECVT_IPLBOOSTS_ENDEDBYPGM	"X'04'"
	1.		ECVT_IPLBOOSTS_ENDEDBYSHUTDOWN	"X'02'"
	1		ECVT_IPLBOOSTS_ENDEDBYERROR	"X'01'"
995	(3E3)	BITSTRING	1	ECVT_BOOSTINFO_SD_FLAGS1	
		1...		ECVT_SDZIIBoost_ENDEDBYERROR	"X'80'"
		.1..		ECVT_SDSPEEDBoost_ENDEDBYERROR	"X'40'"
	 1...		ECVT_SDBOOSTS_ENDEDBYTIMER	"X'08'"
	1..		ECVT_SDBOOSTS_ENDEDBYPGM	"X'04'"
	1		ECVT_SDBOOSTS_ENDEDBYERROR	"X'01'"
996	(3E4)	SIGNED	2	ECVT_BOOSTINFO_TRANSIENTZIIPORES	Number of zIIP cores configured online for the zIIP boost. Those cores will be configured offline at the end of the zIIP boost. Valid only when the zIIP boost active bit is on.
998	(3E6)	BITSTRING	1	ECVTR3E6	
999	(3E7)	BITSTRING	1	ECVT_BOOSTLEVEL	0: initial deliverable >0: EcvT_BoostInfo_V1 may be examined
999	(3E7)	X'0'	0	ECVT_BOOSTLEVEL_V0	"0" Initial deliverable
999	(3E7)	X'1'	0	ECVT_BOOSTLEVEL_V1	"1" EcvT_BoostInfo_V1 may be examined
999	(3E7)	X'1'	0	ECVT_BOOSTLEVEL_MAX	"1" Maximum level of support. This may change in the future.
1000	(3E8)	CHARACTER	16	ECVT_BOOSTINFO_EXPECTED_ENDETOD	Time (STCKE format) when boost(s) will end. It is valid when the following is true: -- one of the "Boost Active" bits is on, and -- if the boost class is IPL, bit EcvT_IplBoosts_Activated is on, and -- if the boost class is shutdown, bit EcvT_SDBoosts_Activated is on, and -- if the boost class is RP, bit EcvT_RPBoosts_Activated is on The zIIP boost might end early due to unexpected error. Boosts might end early due to customer action.
1016	(3F8)	BITSTRING	32	ECVT_BOOSTINFO_V1(0)	Recovery process boost support. Valid when EcvT_BoostLevel is at least EcvT_BoostLevel_V1
1016	(3F8)	SIGNED	4	ECVT_RPBOOSTS_NUM	Number of recovery-process boost start requests received across the life of the IPL that were not during IPL boost or shutdown boost
1020	(3FC)	SIGNED	4	ECVT_RPBOOSTS_NUM_IGNORED	Number of recovery-process boost start requests received across the life of the IPL that were ignored because the duration limit had been exceeded
1024	(400)	BITSTRING	8	ECVT_RP_DURATION	Total duration for the life of the IPL of recovery process boosts. Updated when RP boost is extended or ends. In STCK format.
1032	(408)	BITSTRING	1	ECVT_RP_BOOSTINFO_FLAGS1	

Table 2. Structure ECVT (continued)

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
	 1...		ECVT_RPB00STS_LAST_ENDEDBYTIMER	"X'08'" The last RPBoost(s) ended by timer. Valid only when RPBoost inactive.
	1.		ECVT_RPB00STS_LAST_ENDEDBYSHUTDOWN	"X'02'" The last RPBoost(s) ended due to shutdown. Valid only when RPBoost inactive.
	1		ECVT_RPB00STS_LAST_ENDEDBYERROR	"X'01'" The last RPBoost(s) ended due to error. Valid only when RPBoost inactive.
1033	(409)	BITSTRING	1	ECVT_RPB00STS_REQUESTOR_ID	The requestor ID associated with the start or extend. Updated when the boost is extended. Valid only when RP boost(s) are active. See equates ECVT_RPBReq_xxx
1034	(40A)	BITSTRING	2	ECVTR40A	
1036	(40C)	ADDRESS	4	ECVT_RPB00STS_NUMBYREQUESTOR_ADDR	
Boost Class equates. "AND" the byte with this mask and compare the result to the equate if you want to check the class					
	111		ECVT_BOOSTCLASS_MASK	"X'07'"
	1		ECVT_BOOSTCLASS_IPL	"X'01'"
	1.		ECVT_BOOSTCLASS_SHUTDOWN	"X'02'"
	11		ECVT_BOOSTCLASS_RP	"X'03'"
Recovery Process Boost requestor equates					
1080	(438)	X'0'	0	ECVT_RPBREQ_NOT_IDENTIFIED	"0"
1080	(438)	X'1'	0	ECVT_RPBREQ_SYSPLEX_PARTITIONING	"1"
1080	(438)	X'2'	0	ECVT_RPBREQ_CF_STRUCTURE_RECOV	"2"
1080	(438)	X'3'	0	ECVT_RPBREQ_CF_DATASHARING_MEMBER_RECOV	"3"
1080	(438)	X'4'	0	ECVT_RPBREQ_HYPERSWAP	"4"

IHAENF84 information

IHAENF84 programming interface information

IHAENF84 is a programming interface.

IHAENF84 heading information

Common name: Boost Event ENF mapping (event code 84)
Macro ID: IHAENF84
DSECT name: ENF84
Owning component: Supervisor (SC1C5)

Eye-catcher ID: None

Storage attributes: Subpool: 247 for ENF signal
Key: 0
Residency: Above 16M

Size: ENF84 -- X'0008' bytes

Created by: Boost processing, provided to ENF listeners for event 084.

Pointed to by: R1 on entry to ENF listening routine

Serialization: None required

Function: Maps the data provided for ENF event 084.

IHAENF84 mapping

Table 3. Structure ENF84

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	ENF84	Boost event ENF signal parameter list
0	(0)	BITSTRING	1	ENF84_VERSION	Initial version=0
1	(1)	BITSTRING	1	ENF84_FLAGS	Flags
Bit definitions:					
		1...		ENF84_ZIIPBOOSTACTIVE	"X'80'" For the "boosts end" events, the boost was active. For the "boost start" event, the boost was activated
		.1...		ENF84_SPEEDBOOSTACTIVE	"X'40'" For the "boosts end" events, the boost was active. For the "boost start" event, the boost was activated
	111		ENF84_BOOSTCLASS	"X'07'" See ENF84_BoostClass_xxx
2	(2)	BITSTRING	1	ENF84_RPBOOSTS_REQUESTOR_ID	For the "RPBoosts Start" and "RPBoosts Extend" events only. See equates ENF84_RPBReq_xxx.
3	(3)	CHARACTER	1		Reserved
4	(4)	SIGNED	4	ENF84_EVENT	The boost event. See equates beginning ENF84_Event
4	(4)	X'1'	0	ENF84_EVENT_IPLBOOSTS_START	"1" Application code might not be able to be started early enough to listen for, and receive, this event. Such code should examine the boost information in the ECVT
4	(4)	X'2'	0	ENF84_EVENT_IPLBOOSTS_END	"2"
4	(4)	X'3'	0	ENF84_EVENT_SHUTDOWNBOOSTS_START	"3"
4	(4)	X'4'	0	ENF84_EVENT_SHUTDOWNBOOSTS_END	"4"
4	(4)	X'5'	0	ENF84_EVENT_RPBOOSTS_START	"5"
4	(4)	X'6'	0	ENF84_EVENT_RPBOOSTS_EXTEND	"6"
4	(4)	X'7'	0	ENF84_EVENT_RPBOOSTS_END	"7"
Boost Class equates. "AND" the byte with this mask and compare the result to the equate if you want to check the class					

Table 3. Structure ENF84 (continued)

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
	111		ENF84_BOOSTCLASS_MASK	"X'07'"
	1		ENF84_BOOSTCLASS_IPL	"B'00000001'"
	1.		ENF84_BOOSTCLASS_SHUTDOWN	"B'00000010'"
	11		ENF84_BOOSTCLASS_RP	"B'00000011'"
Recovery Process Boost requestor constants					
4	(4)	X'0'	0	ENF84_RPBREQ_NOT_IDENTIFIED	"0"
4	(4)	X'1'	0	ENF84_RPBREQ_SYSPLEX_PARTITIONING	"1"
4	(4)	X'2'	0	ENF84_RPBREQ_CF_STRUCTURE_RECOV	"2"
4	(4)	X'3'	0	ENF84_RPBREQ_CF_DATASHARING_MEMBER_RECOV	"3"
4	(4)	X'4'	0	ENF84_RPBREQ_HYPERSWAP	"4"
4	(4)	X'8'	0	ENF84_LEN	"*-ENF84"

Table 4. Cross Reference for IHAENF84

Name	Offset	Hex Tag
ENF84	0	
ENF84_BOOSTCLASS	1	7
ENF84_BOOSTCLASS_IPL	4	1
ENF84_BOOSTCLASS_MASK	4	7
ENF84_BOOSTCLASS_RP	4	3
ENF84_BOOSTCLASS_SHUTDOWN	4	2
ENF84_EVENT	4	
ENF84_EVENT_IPLBOOSTS_END	4	2
ENF84_EVENT_IPLBOOSTS_START	4	1
ENF84_EVENT_RPBBOOSTS_END	4	7
ENF84_EVENT_RPBBOOSTS_EXTEND	4	6
ENF84_EVENT_RPBBOOSTS_START	4	5
ENF84_EVENT_SHUTDOWNBOOSTS_END	4	4
ENF84_EVENT_SHUTDOWNBOOSTS_START	4	3
ENF84_FLAGS	1	
ENF84_LEN	4	8
ENF84_RPBBOOSTS_REQUESTOR_ID	2	
ENF84_RPBREQ_CF_DATASHARING_MEMBER_RECOV	4	3
ENF84_RPBREQ_CF_STRUCTURE_RECOV	4	2
ENF84_RPBREQ_HYPERSWAP	4	4
ENF84_RPBREQ_NOT_IDENTIFIED	4	0
ENF84_RPBREQ_SYSPLEX_PARTITIONING	4	1
ENF84_SPEEDBOOSTACTIVE	1	40
ENF84_VERSION	0	
ENF84_ZIIPBOOSTACTIVE	1	80

IHAIPA information

IHAIPA mapping

Table 5. Structure IPA

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
3032	(BD8)	CHARACTER	8	IPABOOST	
PDE for BOOST					

IRALPDAT information

IRALPDAT mapping

Table 6. Structure LPDATMAP

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
173	(AD)	BITSTRING	1	LPDATBOOSTINFO	IRALPDAT.409: same as RMCTZ_BoostInfo

IRAQVS information

IRAQVS mapping

Table 7. Structure QVS

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
6	(6)	BITSTRING	1	QVSBOOSTINFO	IRAQVS.265: same as RMCTZ_BoostInfo

IRARMCTZ information

IRARMCTZ mapping

Table 8. Structure RMCTZ

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
44	(2C)	SIGNED	4	RMCTZ_ADJC_CEC	CPU rate adjustment factor based on the number of physical CPUs available for the logical partitions to share, excluding speed boost
48	(30)	SIGNED	4	RMCTZ_CEC_CAPACITY	Total CEC capacity derivated from rmctz_adjc_cec service units per 10 seconds. The value is based on the number of physical CPUs available for the logical partitions to share and does not include boost capacity
53	(35)	CHARACTER	3		Reserved
1261	(4ED)	BITSTRING	1	RMCTZ_BOOSTINFO	WLM view of boost information
Bit definitions:					
		1... ..		RMCTZ_ZIIPBOOST_ACTIVE	"X'80'"
		.1.. ..		RMCTZ_SPEEDBOOST_ACTIVE	"X'40'"
		..1.		RMCTZ_IPLBOOST_ACTIVATED	"X'20'" IPL boost has been activated. Once set, this flag never gets reset

Table 8. Structure RMCTZ (continued)

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
		...1		RMCTZ_SDBOOST_ACTIVATED	"X'10'" Shutdown boost has been activated. Once set, this flag never gets reset
	 1...		RMCTZ_RPBOOST_ACTIVATED	"X'08'" Recovery Process Boost has been activated. This bit will be turned off once the boost ends
	111		RMCTZ_BOOSTCLASS	"X'07'" See RMCTZ_BoostClass_xxx equates. Valid only when one of the boosts is active
Boost Class equates. "AND" the byte with this mask and compare the result to the equate if you want to check the class					
	111		RMCTZ_BOOSTCLASS_MASK	"X'07' "
	1		RMCTZ_BOOSTCLASS_IPL	"X'01' "
	1.		RMCTZ_BOOSTCLASS_SHUTDOWN	"X'02' "
	11		RMCTZ_BOOSTCLASS_RP	"X'03' "

IWMWRCAA information

IWMWRCAA mapping

Table 9. Structure RCAEIHDR

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
72	(48)	BITSTRING	1	RCAEIFLG	Period flags.
		1...		RCAECCIMP	"X'80'" 1: Service class period implicitly designated CPU critical

Table 10. Structure RCAAGDDE

Offset Dec	Offset Hex	Type	Len	Name (Dim)	Description
6	(6)	X'19'	0	RCAEVRID	"25" Current version level

Index

B

BOOST parameter of IEASYSxx [8](#)

I

IEASYSxx
BOOST parameter [8](#)

