z/OS

MVS Capacity Provisioning User's Guide

Version 2 Release 2

SC34-2661-01

Note
Before using this information and the product it supports, read the information in "Notices" on page 161.

This edition applies to Version 2 Release 2 of z/OS (5650-ZOS) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Figures

Tables

About this publication

This manual supports z/OS (5650-ZOS). This document contains information to help you use Capacity Provisioning and to make the most of the product.

Who should use this document

This document is intended for system administrators who want to use MVS™ Capacity Provisioning to manage Capacity on Demand or Defined Capacity, and for system programmers, system analysts, and systems engineers who are responsible for implementing MVS Capacity Provisioning.

Where to find more information

Where necessary, this document references information in other documents, by using shortened versions of the document title. For complete titles and order numbers of the documents for all products that are part of z/OS, see *z/OS V2R2 Information Roadmap*.

To view, search, and print z/OS publications, go to the z/OS Internet Library at http://www.ibm.com/systems/z/os/zos/bkserv/. Softcopy documentation is available as online collection kit that is available in compressed format for download from the IBM publication center.

Softcopy Title	Title	Order Number
z/OS Common Information Model User's Guide	*z/OS Common Information Model User's Guide*	SC34-2671
z/OS MVS Initialization and Tuning Guide	*z/OS MVS Initialization and Tuning Guide*	SA23-1379
z/OS MVS Initialization and Tuning Reference	*z/OS MVS Initialization and Tuning Reference*	SA23-1380
z/OS MVS JCL User's Guide	*z/OS MVS JCL User's Guide*	SA23-1386
z/OS MVS JCL Reference	*z/OS MVS JCL Reference*	SA23-1385
z/OS Planning for Installation	*z/OS Planning for Installation*	GA32-0890
z/OS MVS Planning: Workload Management	*z/OS MVS Planning: Workload Management*	SC34-2662
z/OS MVS Programming: Workload Management Services	*z/OS MVS Programming: Workload Management Services*	SC34-2663
z/OS MVS Programming: Callable Services for High-Level Languages	*z/OS MVS Programming: Callable Services for High-Level Languages*	SA23-1377
z/OS MVS Setting Up a Sysplex	*z/OS MVS Setting Up a Sysplex*	SA23-1399
z/OS MVS System Commands	*z/OS MVS System Commands*	SA38-0666
z/OS MVS System Messages, Vol 4 (CBD-DMO)	*z/OS MVS System Messages, Vol 4 (CBD-DMO)*	SA23-1389
System z10™ Enterprise Class Capacity on Demand User's Guide	*System z10 Enterprise Class Capacity on Demand User's Guide*	SC28-6871
IBM z Systems Capacity on Demand User's Guide	*IBM z Systems Capacity on Demand User's Guide*	SC28-6943
IBM z/OS Management Facility Configuration Guide	*IBM z/OS Management Facility Configuration Guide*	SA38-0657
z/OS Migration	*z/OS Migration*	GA32-0889

Other Referenced Documents

Softcopy Title	Title	Order Number
z/OS® Security Server RACF® Security Administrators Guide	*z/OS Security Server RACF Security Administrators Guide*	*SA23-2289*
z/OS Security Server RACF Security Callable Services	*z/OS Security Server RACF Security Callable Services*	*SA23-2293*
z/OS RMF Report Analysis	*z/OS RMF Report Analysis*	SC34-2665
z/OS UNIX System Services Programming Tools	*z/OS UNIX System Services Programming Tools*	SA23-2282
z/OS UNIX System Services Planning	*z/OS UNIX System Services Planning*	GA32-0884
PR/SM Planning Guide	*PR/SM Planning Guide*	Varies by CPC
Support Element Operations Guide	*Support Element Operations Guide*	SC28-6831
IBM z Systems Application Programming Interfaces	*IBM z Systems Application Programming Interfaces*	SB10-7164

The Capacity Provisioning Web Page

For the latest information about Capacity Provisioning, see the Capacity Provisioning web page at:

http://www.ibm.com/systems/z/os/zos/features/cpm/index.html

The z/OSMF Web Page

For more information about z/OSMF, visit the z/OSMF home page available at

http://www.ibm.com/systems/z/os/zos/zosmf/

How to read syntax diagrams

This section describes how to read syntax diagrams. It defines syntax diagram symbols, items that may be contained within the diagrams (keywords, variables, delimiters, operators, fragment references, operands) and provides syntax examples that contain these items.

Syntax diagrams pictorially display the order and parts (options and arguments) that comprise a command statement. They are read from left to right and from top to bottom, following the main path of the horizontal line.

For users accessing the Information Center using a screen reader, syntax diagrams are provided in dotted decimal format.

Symbols

The following symbols may be displayed in syntax diagrams:

Symbol
Definition

►►— Indicates the beginning of the syntax diagram.

—► Indicates that the syntax diagram is continued to the next line.

►— Indicates that the syntax is continued from the previous line.

—►◄ Indicates the end of the syntax diagram.

Syntax items

Syntax diagrams contain many different items. Syntax items include:

- Keywords - a command name or any other literal information.
- Variables - variables are italicized, appear in lowercase, and represent the name of values you can supply.
- Delimiters - delimiters indicate the start or end of keywords, variables, or operators. For example, a left parenthesis is a delimiter.
- Operators - operators include add (+), subtract (-), multiply (*), divide (/), equal (=), and other mathematical operations that may need to be performed.
- Fragment references - a part of a syntax diagram, separated from the diagram to show greater detail.
- Separators - a separator separates keywords, variables or operators. For example, a comma (,) is a separator.

Note: If a syntax diagram shows a character that is not alphanumeric (for example, parentheses, periods, commas, equal signs, a blank space), enter the character as part of the syntax.

Keywords, variables, and operators may be displayed as required, optional, or default. Fragments, separators, and delimiters may be displayed as required or optional.

Item type
Definition

Required
Required items are displayed on the main path of the horizontal line.

Optional
Optional items are displayed below the main path of the horizontal line.

Default
Default items are displayed above the main path of the horizontal line.

Syntax examples

The following table provides syntax examples.

Table 1. Syntax examples

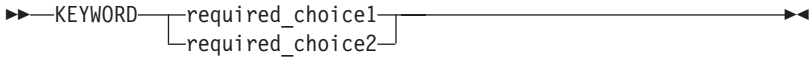
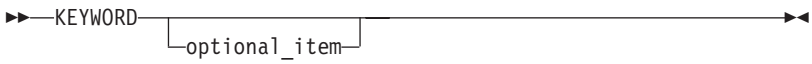
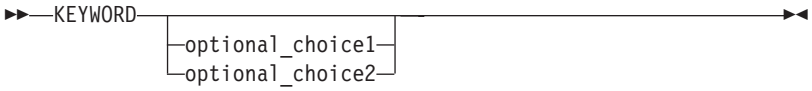
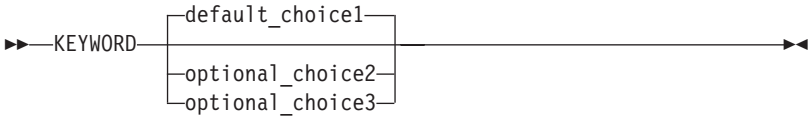
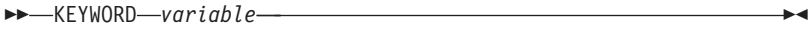
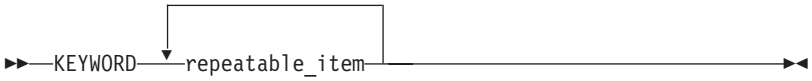
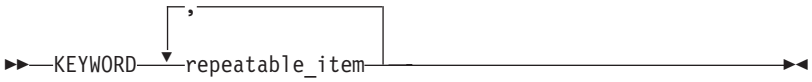
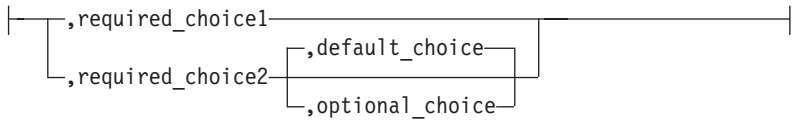
Item	Syntax example
Required item. Required items appear on the main path of the horizontal line. You must specify these items.	
Required choice. A required choice (two or more items) appears in a vertical stack on the main path of the horizontal line. You must choose one of the items in the stack.	
Optional item. Optional items appear below the main path of the horizontal line.	

Table 1. Syntax examples (continued)

Item	Syntax example
Optional choice. An optional choice (two or more items) appears in a vertical stack below the main path of the horizontal line. You may choose one of the items in the stack.	
Default. Default items appear above the main path of the horizontal line. The remaining items (required or optional) appear on (required) or below (optional) the main path of the horizontal line. The following example displays a default with optional items.	
Variable. Variables appear in lowercase italics. They represent names or values.	
Repeatable item. An arrow returning to the left above the main path of the horizontal line indicates an item that can be repeated. A character within the arrow means you must separate repeated items with that character. An arrow returning to the left above a group of repeatable items indicates that one of the items can be selected,or a single item can be repeated.	 
Fragment. The fragment symbol indicates that a labelled group is described below the main syntax diagram. Syntax is occasionally broken into fragments if the inclusion of the fragment would overly complicate the main syntax diagram.	 **fragment:** 

How to send your comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or provide any other feedback that you have.

Use one of the following methods to send your comments:

1. Send an email to mhvrcfs@us.ibm.com.
2. Send an email from the "Contact us" web page for z/OS (http://www.ibm.com/systems/z/os/zos/webqs.html).

Include the following information:

- Your name and address.
- Your email address.
- Your telephone or fax number.
- The publication title and order number:
  z/OS V2R2 MVS Capacity Provisioning User's Guide
  SC34-2661-01
- The topic and page number that is related to your comment.
- The text of your comment.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute the comments in any way appropriate without incurring any obligation to you.

IBM or any other organizations use the personal information that you supply to contact you only about the issues that you submit.

If you have a technical problem

Do not use the feedback methods that are listed for sending comments. Instead, take one of the following actions:

- Contact your IBM service representative.
- Call IBM technical support.
- Visit the IBM Support Portal at z/OS Support Portal (http://www-947.ibm.com/systems/support/z/zos/).

If you have a technical problem

Do not use the feedback methods listed in “How to send your comments to IBM.” Instead, do one of the following:

- Contact your IBM® service representative.
- Call IBM technical support.
- Visit the z/OS Support Portal (http://www-947.ibm.com/systems/support/z/zos/).

Summary of changes

Summary of changes for z/OS Version 2 Release 2 (V2R2)

The following changes are made for z/OS Version 2 Release 2 (V2R2).

New

- Support for Java Version 7.1 as a run-time environment for the Provisioning Manager.
- New examples of how z/OS Capacity Provisioning can help have been added. See "What Capacity Provisioning can do for you" on page 4 for more information.
- The Provisioning Manager provides a new function that warns when manual changes of defined capacity interfere with management of Capacity Provisioning. See "Processing Defined Capacity and Group Capacity" on page 42.
- New function to allow for provisioning of additional capacity based on the current CPC utilization. Utilization conditions are specified for the CPC wide physical processor utilization, broken down by CPU types such as CP, zIIP and zAAP. See the following sections for more information on the utilization condition:
  - "Capacity Provisioning Policy" on page 13
  - "Capacity Provisioning Policy" on page 16
  - "Utilization conditions" on page 33
  - "The policy report" on page 78
  - "The utilization report" on page 86
  - "REPORT UTILIZATION" on page 141.

Summary of changes for z/OS Version 2 Release 1

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

- *z/OS V2R2 Migration*
- *z/OS Planning for Installation*
- *z/OS Summary of Message and Interface Changes*
- *z/OS V2R2 Introduction and Release Guide*

Part 1. Getting started

Provides an overview of Capacity Provisioning (CP) and how to implement it.

Chapter 1. Introducing z/OS Capacity Provisioning

Performance and capacity management on z Systems must ensure that work is processed according to the service level agreements that are in place. Guaranteeing service levels continues to be a relatively static task as long as the workloads being considered are sufficiently stable. However, in many environments, workloads fluctuate considerably over time. Because the total workload or mixture of workloads varies, it can become increasingly difficult to guarantee service levels. With z/OS Workload Management (WLM), incoming work is classified with a performance goal and a business priority. WLM tries to accommodate the goals of all the work in the system. However, even with an ideal WLM service definition, it might not be possible to achieve all specified goals when the total workload increases. In this case, trade-offs must be made. WLM decides which goals can be compromised first, based on the assigned importance level. Discretionary work is displaced first, followed by low importance work.

At some point, however, this solution might not be acceptable, either because the displaced work is relevant from a business perspective, or because it interacts with resources that are required by more important work. There can be different reasons for this situation to occur with different ways how they can be resolved.

1. The CPC's available processor capacity has been exhausted.

   If the processor capacity of your zEnterprise server is insufficient the capacity may need to be increased to accommodate the increased workload. This can require a permanent capacity increase for planned growth, or a temporary capacity increase for seasonal or unpredictable peak periods. IBM z Systems can quickly and non-disruptively activate additional processor capacity. That capability is built directly into z Systems servers. It is provided by IBM Capacity Upgrade on Demand (CUoD) for a permanent increase in processor capability, and IBM On/Off Capacity on Demand (On/Off CoD) for a temporary capacity increase.

   On/Off Capacity on Demand allows the configuration, for example, of general purpose processor (CP) capacity and specialty processors, such as zAAPs, zIIPs, IFLs, ICFs, or SAPs. Several models of z Systems servers are subcapacity models. On such models, additional general purpose processor capacity can be provided by a different capacity level, additional processors, or a combination of the two.

   For example, for the z10 EC server, the capacity levels for the CP engine are 7, 6, 5, and 4. Full capacity CP engine is indicated by 7; subcapacity CP engines are indicated by 6, 5, and 4. For the z10 BC server, the capacity levels for the CP engine are A-Z. A full capacity CP engine is indicated by the letter Z. The capacity setting is derived from both the capacity level and the number of CPs. For the z10 EC server, the capacity settings are 4xx, 5xx, 6xx, 7xx. For the z10 BC server, the capacity levels are Axx - Zxx.

2. When Defined Capacity or Group Capacity has been exceeded.

   Such a situation can occur because the available processor capacity has been exhausted or due to WLM soft capping when Workload License Charges (WLC) are in effect and the Defined Capacity or Group Capacity limit has been exceeded. In this situation, your zEnterprise server might still have enough processor capacity but WLM caps the capacity available to one or more images for cost reasons. To overcome this situation you can temporarily increase Defined Capacity or Group Capacity.

3. Logical processor resources are insufficient.

   This situation may arise while your zEnterprise Server still has sufficient processor capacity but the number of logical processors of a partition is insufficient to contain the workload. The available physical capacity cannot be used by the partition that needs it. Since there is still sufficient processor capacity on the server, this capacity can be directed to the workload by configuring additional logical processors online.

In the context of Capacity Provisioning, the capacity of a z/OS system means processor capacity. A z/OS system can consume different types of processor capacity:

- General purpose capacity

  Processor capacity of general purpose processors (CPs), usually measured in million service units (MSU) per hour, or just MSU
- Application assist processor capacity

  Processor capacity of System z Application Assist Processors (zAAP), measured in number of processors
- Integrated information processor capacity

  Processor capacity of System z Integrated Information Processors (zIIP), measured in number of processors

What Capacity Provisioning can do for you

Different constraints may influence what capacity a z/OS system can consume. There are different ways how to relieve the constrains:

- If the overall processing capacity of your zEnterprise server (CPC) get exhausted, in order to relief the bottleneck additional physical capacity needs to be activated via the On/Off Capacity on Demand (On/Off CoD) feature.
- An installation may use Variable Workload License Charge (VWLC) and has configured defined capacity. Defined capacity comes in two flavors:
  - a capacity limit for a single logical partition (LPAR). This is referred to as Defined Capacity by Capacity Provisioning. In other documentation, this may also be referred to as LPAR defined capacity, or soft capping.
  - a capacity limit for a group of logical partitions. Such a capacity group is a named configuration element that is defined on a Support Element of a CPC. This is referred to as Group Capacity by Capacity Provisioning. In other documentation, it may also be referred to as group defined capacity.

  For a workload, Defined Capacity, Group Capacity, or both types of defined capacity can be in effect. The capacity limit for defined capacity is measured in million service units (MSU) per hour, our just MSU for short. When your workload is constrained by defined capacity, the capacity limit needs to be increased to resolve the bottleneck.
- A system may be constrained by the number of online logical processors, and more logical processors need to be configured online to resolve the constraint.

z/OS Capacity Provisioning can help you identifying and resolving capacity bottlenecks, or constraints, that may impact important workloads. Because both, the analysis of configuration and performance data, and eventual actions, as defined in your Capacity Provisioning policies, are automated, problems may be addressed more timely than through human action.

For example, you can:

- Activate and deactivate capacity through operator commands.
- Activate and deactivate capacity based on a defined schedule, without considering workload performance.
- Have the Provisioning Manager suggest changes to the capacity of the System server based on the observation of workloads or the processor utilization that you define.
- Have the Provisioning Manager automatically implement changes to the capacity of the z Systems server based on the observation of workloads or the processor utilization that you define.

Table 2. Capacity Provisioning Features

Type of Resource	**Capacity Provisioning can monitor for capacity shortage based on your policy**	**Capacity Provisioning can alert you to address capacity shortage based on your policy**	**Capacity Provisioning can resolve capacity shortage based on your policy (with or without additional confirmation)**	**Commands for manual control**
CPC physical general purpose processor capacity	Yes	Yes	Yes	Yes - valid On/Off CoD record required
CPC zIIP and zAAP processor capacity	Yes	Yes	Yes	Yes - valid On/Off CoD record required
CPC IFL, ICF, and SAP processor capacity	No	No	No	Yes - valid On/Off CoD record required
Logical processor capacity	Yes	Yes	No	Via MVS CONFIG command
Defined Capacity, Group Capacity	Yes	Yes	Yes	Yes

Note: Capacity Provisioning supports Group Capacity on IBM zEnterprise z196 servers and later.

Defined Capacity and Group Capacity Management

Defined Capacity and Group Capacity are frequently used in conjunction with the Variable Workload License Charge (VWLC) pricing model. Sub-capacity Workload License Charges introduce the capability to pay software license fees based on the processor utilization of the LPAR on which the product is running, rather than on the total capacity of the system, as follows:

- An installation can specify a Defined Capacity expressed in millions of service units per hour (MSU) for an LPAR. The Defined Capacity sets the capacity limit of an individual LPAR when soft capping is in effect.
- WLM keeps a 4-hour rolling average of the CPU usage of the LPAR, and when the 4-hour average CPU consumption exceeds the defined capacity limit, WLM

dynamically activates soft capping. When the rolling 4-hour average returns below the Defined Capacity, soft capping is removed.

The Group Capacity allows the definition of a group of LPARs on the same CPC and a limit for the combined capacity usage by those LPARs. This allows the system to manage the group in such a way that the limit for Group Capacity in MSU per hour will not be exceeded. PR/SM and WLM work together to enforce the capacity defined for the group and also enforce the capacity optionally defined for each individual LPAR.

An LPAR can be restricted by Defined Capacity and in addition belong to a capacity group.

z/OS Capacity Provisioning can help you to monitor and manage the Defined Capacity and Group Capacity either through policy based automation or through commands from the z/OS Console.

For further details on the management of Defined Capacity and Group Capacity with z/OS Capacity Provisioning see the section "Processing Defined Capacity and Group Capacity" on page 42.

Considerations for configuring capacity

Configuring additional capacity via On/Off Capacity on Demand can result in additional IBM hardware and software license charges. For more information, see the *System z10 Enterprise Class Capacity on Demand User's Guide*, SC28-6871, *IBM z Systems Capacity On Demand User's Guide*, SC28-6943, or your IBM sale representative.

Increasing Defined Capacity or Group Capacity can result in additional license charges. For more information, refer to http://www.ibm.com/systems/z/resources/swprice/index.html, or your IBM sales representative.

An accurate management for Group Capacity depends on reliable data from the monitoring product. For this reason it is advisable that the LPAR of a system that can trigger provisioning of Group Capacity belongs to a capacity group for at least 4 hours.

There might be additional fees for non-IBM software. In addition, some non-IBM software packages might require new license keys to take advantage of the additional capacity. Check with your software vendor for details.

Components of Capacity Provisioning

z/OS Capacity Provisioning is delivered as part of the z/OS MVS Base Control Program (BCP) component. Capacity Provisioning includes the following:

- Capacity Provisioning Manager (Provisioning Manager)
- Capacity Provisioning Management Console, available in the IBM z/OS Management Facility
- Sample data sets and files

Capacity Provisioning configuration entities, such as policies and domain configurations, are defined through a web-based interface, the Capacity

Provisioning Management Console. The Provisioning Manager on the z/OS host observes workloads and interacts with the z/OS systems and servers defined as part of Capacity Provisioning.

The Capacity Provisioning Manager

The Provisioning Manager monitors the workload on a set of z/OS systems and organizes the provisioning of additional capacity to these systems when required. You define the systems to be observed in a domain configuration file. Details of additional capacity and the rules for its provisioning are stored in a policy file. These two files are created and maintained through the Capacity Provisioning Management Console.

The Capacity Provisioning Management Console

The Capacity Provisioning Management Console (CPMC) is a console which administrators use to work with provisioning policies and domain configurations and to monitor the status of a Provisioning Manager. The management console is implemented by the Capacity Provisioning task in the IBM z/OS Management Facility (z/OSMF). z/OSMF provides a framework for managing various aspects of a z/OS system through a web browser interface. You can use the Capacity Provisioning task to work with provisioning policies and domain configurations and to work with the Capacity Provisioning Manager on your z/OS system. You can set up a connection to the Provisioning Manager and use it to transfer provisioning policies and domain configurations to the Provisioning Manager, or to query various status reports. The CPMC is a web-based application that runs in a browser on the users workstation. It is not required to operate the Provisioning Manager.

Capacity Provisioning sample data sets and files

The Capacity Provisioning component includes several samples that simplify customization and speed-up the definition of your provisioning policies:

- Sample jobs for setting up and customizing the Capacity Provisioning component are placed in `SYS1.SAMPLIB`. The use of these sample members is described in Chapter 3, “Setting up a Capacity Provisioning domain,” on page 47.
- Samples for a Capacity Provisioning Domain Configuration and a Capacity Provisioning Policy are shipped in the UNIX file system at /usr/lpp/cpo/samples as CPSAMDOM.xml and CPSAMPOL.xml. You can transfer these samples to your workstation and import them to the Capacity Provisioning Management Console.

Prerequisites

z/OS Capacity Provisioning has certain hardware and software requirements. If your configuration does not meet all these requirements, you cannot use the full functionality of Capacity Provisioning. However, some functions can be used with fewer requirements.

z Systems hardware requirements

- One or more z10 zEnterprise or z13 servers.
- If temporary capacity is to be controlled by the Provisioning Manager in confirmation or autonomic mode, or if provisioning actions are to be performed through Provisioning Manager commands, temporary capacity must be available. To make temporary capacity available, On/Off CoD must be enabled

(feature code 9896), as well as a valid On/Off CoD record for temporary general purpose processor, zAAP, zIIP, IFL, ICF, or SAP capacity.

- For managing Group Capacity with Capacity Provisioning, the observed systems have to run on an IBM zEnterprise z196 server or later.
- Capacity Provisioning communicates with the hardware to get information about the permanent and temporary capacity of the server. The communication is done using z/OS BCP internal interface (BCPii), a z/OS built-in communication that does not require a network connection. BCPii communicates to the local SE, but if information of other central processor complexes (CPCs) is needed, a HMC is required. The available options are listed in Table 3.

Table 3. Support Element (SE) or Hardware Management Console (HMC) requirements for Capacity Provisioning configuration

Configuration	Communication through BCPii
All observed systems and hosting system on one processor complex	SE
All observed systems on one processor complex; hosting system on a different processor complex	HMC
Observed systems on multiple processor complexes	HMC

z/OS software requirements

- Observed systems must use z/OS Version 1 Release 13 or higher.
- Observed systems must use z/OS Version 2 Release 1 or higher when Defined Capacity or Group Capacity will be managed by Capacity Provisioning.
- z/OS Resource Measurement Facility™ (RMF™), an optional element of z/OS, must be enabled, or you can use an equivalent product.
- Your z/OS security product must support creation of PassTickets (R_GenSec) and evaluation through the SAF interfaces. If you are using a security product other than IBM Security Server (Resource Access Control Facility, or RACF), check with your vendor.

Supported LPAR and z/OS environments and restrictions

The IBM z Systems platform and z/OS provide for great flexibility. Capacity Provisioning supports a broad range of configurations, but certain configurations are not supported or are restricted. Restrictions include:

- Observed systems must be running z/OS Release 13 or higher. Other operating systems or the Coupling Facility Control Code (CFCC), can be active in other LPARs.
- Observed systems that are running as guests under z/VM® are not supported. Do not use a z/OS system running as a z/VM guest to run the Provisioning Manager.
- The version of the Capacity Provisioning Management Console must be compatible with the version of the z/OS system.
- An observed system can run in a shared or dedicated LPAR. An LPAR with dedicated processors, however, can generate demand only for higher general purpose processor capacity level. If the processor complex is not a subcapacity model but is already operating at its maximum capacity level, additional demand is not recognized. If the LPAR is dedicated, no demand for additional special purpose processors is recognized.

- Demand for additional physical processors, for shared general purpose (CP) processors as well as for zAAP, or zIIP processors can be recognized only if the current sum of logical processors is higher than the number of physical processors in the respective processor pool, or the Capacity Provisioning policy allows to monitor logical processors. Demand for an increased capacity level is recognized regardless of the number of logical processors.
- The additional physical capacity provided through Capacity Provisioning is distributed through PR/SM™ and the operating systems. In general, the additional capacity is available to all LPARs, but facilities such as Defined Capacity (soft capping) or initial capping (hard capping) can be used to control the use of capacity.
- Defined Capacity is managed only for those systems that are defined with positive values in the respective LPAR controls on the HMC or SE.
- Group Capacity is managed only for those capacity groups that are defined with positive values in the respective capacity group controls on the HMC or SE.
- Do not define provisioning conditions for service classes associated with resource groups for which a capacity maximum is in effect.
- Provisioning conditions for physical processor utilization require the availability of shared processors.
- If a system has IRD Vary CPU Management turned on, no recommendations for the number of logical processors are issued by Capacity Provisioning.
- Logical processors are only monitored for systems with shared processors.
- You are strongly advised not to use hard capping (*initial capping* (hard capping) or *absolute capping*) for observed systems. Capacity Provisioning cannot detect whether an increase of temporary capacity or defined capacity would help on such a system.

Workstation requirements

To work with z/OSMF, your workstation requires a supported operating system and web browser. For details on the requirements see "Preparing your workstation for z/OSMF" in the IBM z/OS Management Facility Configuration Guide.

Chapter 2. Capacity Provisioning basics

It is assumed that the user knows the main Capacity Provisioning concepts and the terms related to them. Capacity Provisioning basics is intended to provide basic knowledge of terms and concepts related to the topic. Capacity Provisioning operates in the following environment:

Field of operation
The scope of a z/OS Capacity Provisioning system is referred to as a Capacity Provisioning Domain, or simply a domain. The domain configuration describes the scope of management within a provisioning domain. The domain includes hardware and software elements. The hardware elements are one or more central processor complexes (CPCs) where capacity can be provisioned or deprovisioned by Capacity Provisioning. The software elements are z/OS operating systems that can run on one or more of these CPCs. These elements are monitored by Capacity Provisioning to determine the hardware requirements.

Rules of operation
Provisioning Management is controlled by a Capacity Provisioning Policy, or simply a policy. This policy defines the actions to be performed on the hardware and software elements in response to the demands of the observed software elements. A policy contains rules, which define capacity that can be provisioned, time periods during which the rule is applied and, optionally, workload conditions that trigger provisioning.

The domain is controlled by the Provisioning Manager. The Provisioning Manager runs on a z/OS system and controls the domain in real time. It observes the software elements, monitors workload demands and observes the physical utilization of the shared processor pool. It recommends hardware configuration changes to the system operator, or can activate or deactivate capacity to satisfy these demands.

.

The Capacity Provisioning Management Console (CPMC) is the graphical user interface for the Provisioning Manager. You use this interface to work with provisioning policies and domain configurations, and to view the status of the Provisioning Manager.

These concepts are explained in “Capacity Provisioning Domain” on page 12, “Capacity Provisioning Policy” on page 13, “Capacity Provisioning Manager” on page 14, and “Capacity Provisioning Management Console” on page 14.

More information about key points is provided in “Capacity Provisioning in detail” on page 15.

Overview

This information describes the following elements:

- The environment of Capacity Provisioning, in “Capacity Provisioning Domain” on page 12
- The processing rules, in “Capacity Provisioning Policy” on page 13

- The program used to run Capacity Provisioning, in “Capacity Provisioning Manager” on page 14
- The graphical user interface to manage Capacity Provisioning, in “Capacity Provisioning Management Console” on page 14

Capacity Provisioning Domain

The domain consists of:

- Central processor complexes (CPCs) on which the configuration can be changed, as described in “CPCs” on page 15 and
- Observed systems that can trigger provisioning, as described in “Observed systems” on page 15, and where the number of logical processors can be monitored.

Capacity Provisioning can maintain the configuration of more than one domain for different purposes. Each domain is identified by a unique name.

An example of a domain is:

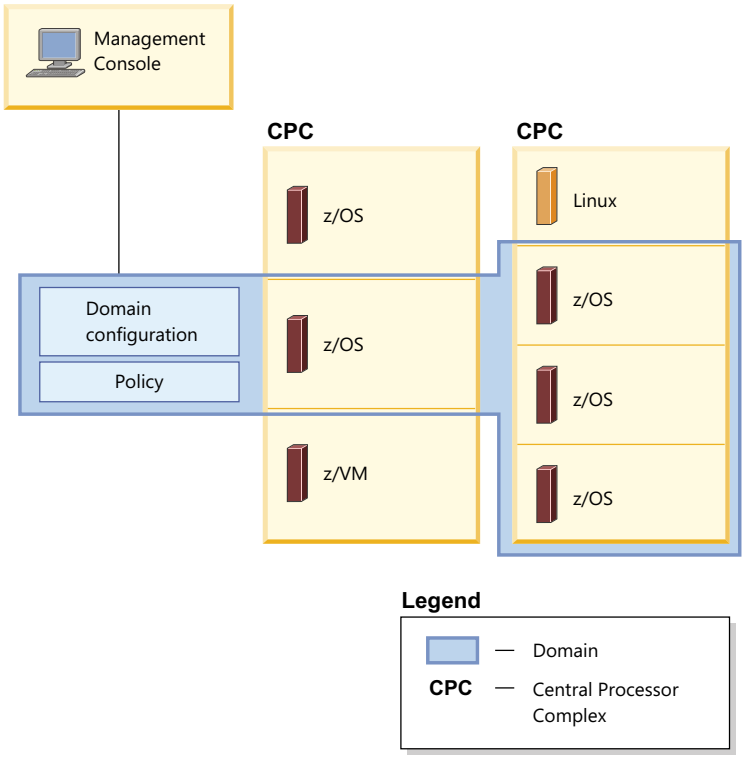


Figure 1. Capacity Provisioning domain

Figure 1 shows an example domain configuration with two CPCs being managed and four systems being observed. Capacity Provisioning can be performed for one or more CPCs. If a sysplex consists of multiple systems, all observed systems of this sysplex must belong to the same Capacity Provisioning domain. One CPC cannot belong to multiple domains. Each system in the domain configuration is identified by its z/OS system name and the name of the sysplex to which it belongs. Systems keep the same name even if they move across LPARs and CPCs.

Capacity Provisioning Policy

The policy describes the scope of management within a provisioning domain. The policy controls the provisioning of additional capacity. Different policies can be created for different circumstances, but only one of these policies can be used by the Provisioning Manager at a time. The policy defines:

- How much additional capacity can be provisioned
- When this additional capacity can be provisioned
- What triggers the provisioning of additional capacity

Each policy must specify at least one of the following scopes:

- A maximum processor scope that defines the total amount of temporary capacity that can be activated:
  - Maximum amount of general purpose capacity, in MSU
  - Maximum number of Application Assist Processors (zAAPs)
  - Maximum number of Integrated Information Processors (zIIPs)

  If you omit the maximum processor definition for an CPC, temporary capacity is not managed for this CPC.
- A logical processor scope that defines the z/OS systems where the number of logical processors is monitored. For each system you can either specify a maximum number of processors or specify that the limit of the LPAR definition applies. If you omit the logical processor definition for an observed system, logical processors are not monitored for this system. A policy specifying only a logical processor scope is not valid.
- A maximum defined capacity scope that defines the total amount of Defined Capacity that can be added. If you omit the capacity limit for an observed system, Defined Capacity is not managed for this system's LPAR.
- A maximum group capacity scope that defines the total amount of Group Capacity that can be added. If you omit the capacity limit for a capacity group on a managed CPC, capacity is not managed for this group.

A policy contains one or more provisioning rules. These rules define restrictions, so-called provisioning scopes, to the capacity that can be provisioned and they contain provisioning conditions that describe the situations in which the Provisioning Manager can provision additional capacity on behalf of the rule. These situations include time conditions that indicate periods in which provisioning is allowed, and can include workload conditions or utilization conditions that indicate demand, which in turn triggers activation. Rules without any workload conditions or utilization conditions are interpreted as mandatory requests for activation. For rules with workload conditions, additional capacity can be provisioned by z/OS Capacity Provisioning only when business critical work is suffering. This work is identified at the planning stage. Workload conditions are expressed in terms of the z/OS WLM service class model. For rules with utilization conditions, additional capacity can be provisioned by z/OS Capacity Provisioning only when a processor utilization is high. This also needs to be identified at the planning stage. Utilization conditions are expressed in terms of physical utilization of the shared processor pool.

Additional capacity can be provisioned by z/OS Capacity Provisioning only when business critical work is suffering. This work is identified at the planning stage and must be specified in the workload conditions of a policy.

For more information about rules, see “Rules” on page 22. For more information about conditions, see “Provisioning conditions” on page 23.

Capacity Provisioning Manager

The Provisioning Manager controls a domain. It monitors the observed systems and can provision or propose manual provisioning of capacity, based on the settings in your active domain configuration and policy.

The specifications of the CPCs to be managed and the systems to be observed are included in a domain configuration. The Provisioning Manager must be able to access these CPCs from every host system on which it runs. Information from available CPCs is obtained through a connection to the hardware console. This console can be a service element (SE) or a hardware management console (HMC).

Note: Unless otherwise indicated, the assumed console is HMC. At certain points, however, if it is possible to interchange the two consoles, the term HMC can be replaced with the term SE.

Management of the domain is controlled by a policy that specifies the time conditions when capacity can be provisioned, optionally specifies the workload conditions that can trigger provisioning, and finally specifies the capacity that can be provisioned under these conditions.

Domain configurations are part of a domain configuration repository , and policies are located in a policy repository. Because these repositories can contain more than one domain configuration or policy, each configuration or policy must be given a unique name. At any given time, only one configuration and one policy can be active in the domain. See “Defining the runtime data sets” on page 52 for more information about repository files.

The Provisioning Manager operates in any one of four processing modes, each with varying powers of autonomy. These modes are described in detail in “Processing modes” on page 38.

Only capacity that was provisioned by the Provisioning Manager is managed by the Provisioning Manager. Capacity that is activated manually, either by using Provisioning Manager commands or by using the interfaces available on the HMC, is not managed by the Provisioning Manager. Nevertheless, you have the option to pass manually provisioned capacity to the Provisioning Manager for deprovisioning of that capacity.

Capacity Provisioning Management Console

The Capacity Provisioning Management Console (CPMC) is a web browser interface that communicates with the z/OSMF application running on a z/OS host system. It holds information in a z/OSMF repository. You can prepare domain configurations and policies in the repository and transmit these from the CPMC to the Provisioning Manager if a connection for this is set up. You can also inspect the details of the Provisioning Manager status. When a connection is set up, the CPMC connects to the Common Information Model (CIM) server on the system where the Provisioning Manager is active, using the CIM HTTP or CIM HTTPS protocol.

For more information about the CPMC, see “Capacity Provisioning Management Console” on page 45, and for the repository, see “z/OSMF Repository” on page 45.

Capacity Provisioning in detail

This topic provides details about the Capacity Provisioning components.

Capacity Provisioning Domain Configuration

The CPCs to be managed and the systems to be observed are specified in the domain configuration. You can create and edit the domain configuration by using the Capacity Provisioning Management Console (CPMC). For more information about the CPMC see "Capacity Provisioning Management Console" on page 45. Before you can activate the domain configuration, you must add it to the domain configuration repository of the domain. You can store multiple domain configurations for different purposes in the repository of the domain, but only one domain configuration can be active in the domain.

Observed systems

To get information about the workload that is running on a system or to manage the number of processors on that system, the Provisioning Manager must be connected to the system. The information is provided by a CIM server on the system. The domain configuration includes attributes that describe how to connect to each system. One attribute is the host address of the system, another attribute is the protocol to be used, and a third specifies the port on which the CIM server is listening.

Each system has an enabled attribute that specifies whether the Provisioning Manager is allowed to connect to the system and retrieve the information. You can switch this attribute on or off at runtime by using the **ENABLE CONFIGURATION** and **DISABLE CONFIGURATION** Provisioning Manager commands, as described in "ENABLE CONFIGURATION" on page 125 and "DISABLE CONFIGURATION" on page 121.

CPCs

CPCs are the target for the management of temporary capacity, Defined Capacity, and Group Capacity by the Provisioning Manager.

Temporary capacity must be installed on a CPC before it can be activated. Installed capacity is described in a capacity record, as described in *System z10 Enterprise Class Capacity on Demand User's Guide* (SC28-6871) and *IBM z Systems Capacity On Demand User's Guide* (SC28-6943). The Provisioning Manager can activate the residual capacity only in this record within limits that are defined during the order process of the record. Some CPCs in a domain might not have temporary capacity. In this case, the Provisioning Manager still reports resource shortages on these CPCs.

The provisioning domain contains a set of logical partitions (LPARs). They can be part of a stand-alone system (a monoplex) or can be part of a sysplex. A z/OS system runs within each LPAR. When you define the domain configuration, you specify the set of z/OS systems to be observed. Each z/OS system is identified by name. If the system is part of a sysplex, the name also includes the name of the sysplex. A system can be observed and considered for capacity changes only if it runs on a CPC in the provisioning domain.

Each CPC in the domain configuration is identified by its logical name, which defines it on the support element (SE) of that processor complex. Each CPC has an enabled attribute that specifies whether the CPC is to be considered for capacity changes. If it is enabled, the Provisioning Manager can change the capacity of that CPC. If it is disabled, only manual capacity changes by using Provisioning

Manager commands are allowed. You can switch the enabled attribute on or off at runtime by using Provisioning Manager commands. For more information about the commands, see “ENABLE CONFIGURATION” on page 125 and “DISABLE CONFIGURATION” on page 121.

The temporary capacity on a CPC includes multiple capacity records that are identified with unique record IDs. Only one of these records can be used by the Provisioning Manager at any one time. You can specify the ID of the record to use in the domain configuration, or if there is only one record, you can let the Provisioning Manager find it.

Note: If you are using Capacity Provisioning solely to manage Defined Capacity and Group Capacity, specify `None` for the record ID of a CPC in the domain configuration.

Defined Capacity and Group Capacity can be managed by the Provisioning Manager when they are defined with positive values in the respective LPAR or LPAR Group controls on the HMC or SE. As soon as they are turned off (set to 0), the management of Defined Capacity is stopped by the Provisioning Manager.

For CPCs that support static power save mode, the Provisioning Manager does not add temporary capacity based on the active policy while static power save is enabled. However, already activated temporary capacity can be deactivated by the Provisioning Manager based on your policy. ACTIVATE RESOURCE and DEACTIVATE RESOURCE commands are not affected by power save mode. Defined Capacity and Group Capacity management are also not impacted by power save mode.

You can enable or disable static power save mode by using the **`ENABLE POWERSAVE`** or **`DISABLE POWERSAVE`** Provisioning Manager commands, as described in “ENABLE POWERSAVE” on page 127 and “DISABLE POWERSAVE” on page 123, or by using the SE or HMC.

Capacity Provisioning Policy

The management of additional capacity is based on a policy that contains rules for activation and deactivation and for capacity increases and decreases. You can create and edit a policy by using the Capacity Provisioning Management Console. When the policy is complete, you must install it into the policy repository of the domain before it can be activated. You can have multiple policies for different purposes in the policy repository of the domain, but at any time only one policy can be active in the domain. You can enable or disable elements of the policy, such as rules or conditions, even when the policy is already active.

Figure 2 on page 17 shows the basic structure of a policy.

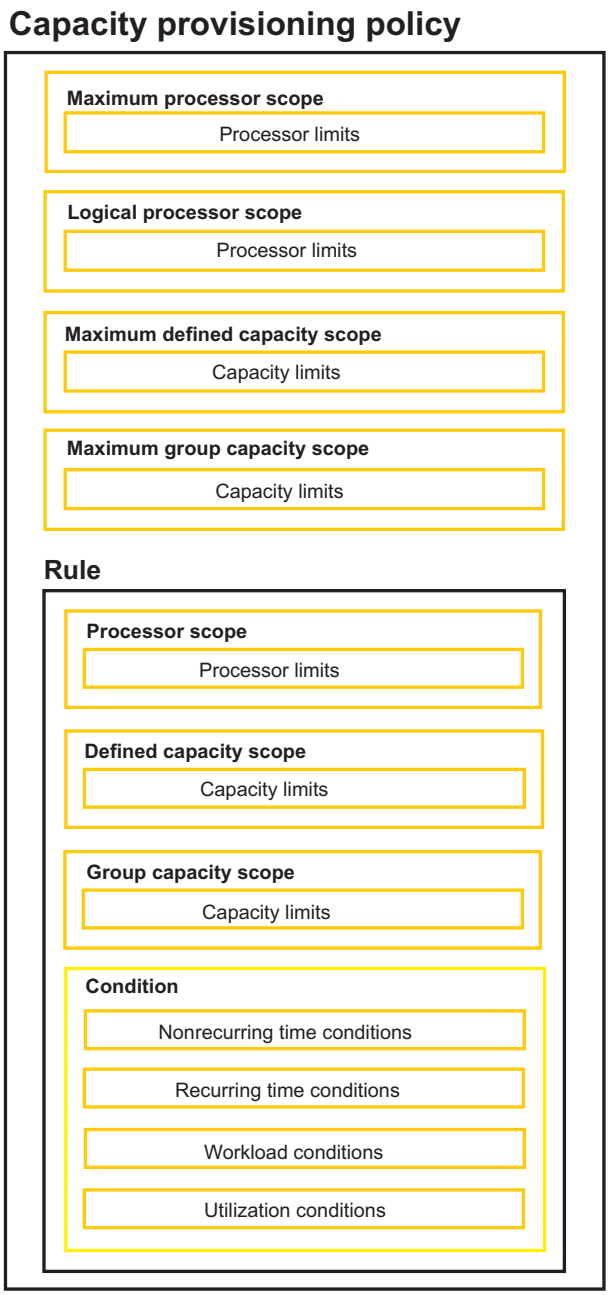


Figure 2. Capacity Provisioning policy structure

The policy contains scopes, which define provisioning limits for different types of capacity, and a set of provisioning rules. The scopes restrict the capacity that may be provisioned by the rules in the policy.

Maximum processor scope
Restricts the temporary processor resources which can be activated for CPCs

Logical processor scope
Defines the z/OS systems for which the number of logical processors is monitored

Maximum defined capacity scope
Defines the maximum amount of MSU by which the Defined Capacity for a z/OS system can be increased

Maximum group capacity scope
Defines the maximum amount of MSU by which the Group Capacity for a capacity group of a CPC can be increased

A provisioning rule contains a set of provisioning conditions and scopes which define provisioning limits for different types of capacity. The scopes restrict the capacity that may be provisioned by the conditions in the rule.

Each provisioning policy is identified by name. The names of rules, provisioning conditions, time conditions, and workload conditions must be unique within the policy. These names are used in commands to the Provisioning Manager (for example to enable or disable a rule or a provisioning condition), and in reports from the Provisioning Manager that reference these policy elements.

Maximum processor scope

A Capacity Provisioning policy includes a maximum processor scope which defines the total amount of temporary processor resources that can be activated for CPCs by all the rules contained in the policy. Each rule also contains a processor scope that restricts the temporary capacity that can be activated by that rule. If the processor scope of a rule includes restrictions on CPCs, they must also be included in the maximum processor scope, otherwise no additional processor capacity is activated for these CPCs.

Specifying a processor limit in the maximum processor scope is optional. It is necessary if the policy intends to allow the activation of temporary processor resources for the CPC.

For example, suppose you want to allow one additional zAAP to be activated when an online service class is impacted, and one or two additional zAAPs when a batch service class is impacted, but you do not want to have more than two additional zAAPs active at the same time. To model this scenario, you define a maximum processor scope of two zAAPs and two rules: one for the online service class with a processor scope of one zAAP and one for the batch service class with a processor scope of two zAAPs. With these rules, if two additional zAAPs are requested for a batch application and one additional zAAP is requested for an online application at the same time, one of the requests is unfulfilled. Either the rule for the batch service class activates only one additional zAAP, or the rule for the online service class activates no additional zAAPs. In total, no more than two additional zAAPs are active at the same time.

For each CPC, you can also specify the increments in which general purpose capacity is to be added. There is one increment for the first activation (primary activation) and one increment for all further activations (secondary activations). The capacity increments are specified in MSU and denote the minimum amount of capacity that can be added. The default increment is one MSU, meaning the software model with the next higher capacity. The capacity increments are used for workload-based or utilization-based capacity activation. Neither the primary nor the secondary activation amounts must exceed the maximum capacity specified as Max. activation (MSU). If you specify primary and secondary increments and the next activation would exceed the capacity of either the defined On/Off Capacity on Demand record for the CPC or the capacity allowed by the processor scope, the next increment is limited to activate the maximum allowed remaining capacity.

Table 4 shows an example of a maximum processor scope with limits for two CPCs.

Table 4. Maximum processor scope

CPC	Maximum activation (MSU)	Maximum zAAPs	Maximum zIIPs	Primary activation (MSU)	Secondary activations (MSU)
CPC0	200	0	4	1	1
CPC1	300	3	2	70	50

The first definition for CPC0 specifies that 200 MSU, four additional zIIPs, but no zAAPs can be activated on that CPC. With values for primary and secondary activations of 1 MSU the definition determines that each workload-based activation should choose the software model with the next higher capacity. The second definition specifies that a maximum of 300 MSU, two zIIPs, and three zAAPs can be activated on CPC1. The first workload-based or utilization-based activation on that CPC would activate the next suitable software model with at least 70 additional MSU, all following activations the next suitable software model with at least 50 additional MSU each.

Logical processor scope

A Capacity Provisioning policy includes a logical processor scope, which defines the systems on which the Provisioning Manager monitors the number of logical processors to issue provisioning recommendations. Specifying a logical processor limit is optional. If a limit is defined for a system the Provisioning Manager monitors the number of logical processors for this system and informs you when changes are required. If a console message is displayed, follow the recommendation and perform the activation and deactivation on the affected system yourself. When the specified limit is reached the Provisioning Manager stops to recommend on additional logical processors.

The number of logical processors on a system can only be increased if offline logical processors are available on this system. This is the case if processors were configured offline or if they are defined as reserved processors in the LPAR in which the system is running.

The logical processor scope is only observed if the system is running with WLM LPAR CPU management turned off and has shared processors.

The Provisioning Manager recommends to configure logical processors online if the number of logical processors restricts the consumption of physical capacity. This situation can occur due to a shortage of logical processors or in combination with a physical activation. The Provisioning Manager recommends to configure logical processors offline if the number of logical processors of a system prevents a change to the physical capacity of the CPC on which the system is running. In this case, the deactivation of physical resources is postponed until logical processors have been configured offline. The Provisioning Manager does not attempt to optimize the number of logical processors for the consumed capacity. However, the HiperDispatch function can be used for that purpose.

Recommendations to take additional logical processors online or offline are issued in confirmation and autonomic processing modes. These modes are described in detail in “Processing modes” on page 38.

Table 5 on page 20 shows an example of logical processor limits.

Table 5. Logical processor scope

System	Sysplex	Maximum CPs	Maximum zAAPs	Maximum zIIPs	Action
SYS0	PLEX0	7	5	3	Message on runtime system
SYS1	PLEX1	Max. possible	0	0	Message on managed system

The fields in Table 5 have the following meaning:

System
Name of the z/OS system

Sysplex
Name of the sysplex to which the system belongs

Maximum CPs
The maximum number of general purpose (CP) processors. When this limit is reached the Provisioning Manager stops to recommend on additional logical general purpose (CP) processors. Alternatively, `Max. possible` stands for as many logical processors as allowed by the z/OS and LPAR configuration.

Maximum zAAPs
The maximum number of zAAP processors. When this limit is reached the Provisioning Manager stops to recommend on additional logical zAAP processors. Alternatively, `Max. possible` stands for as many logical processors as allowed by the z/OS and LPAR configuration.

Maximum zIIPs
The maximum number of zIIP processors. When this limit is reached the Provisioning Manager stops to recommend on additional logical zIIP processors. Alternatively, `Max. possible` stands for as many logical processors as allowed by the z/OS and LPAR configuration.

Action
The action to be taken by the Provisioning Manager whenever more or fewer processors are required. You can choose between the following actions:

Message on runtime system
The Provisioning Manager issues a message on the system on which the Provisioning Manager runs. If this message is displayed, follow the recommendation and perform the activation and deactivations on the affected system yourself.

Message on managed system
The Provisioning Manager issues a message on the system on which the change needs to be performed. If this message is displayed, perform the activation and deactivation on the affected system yourself.

Maximum Defined Capacity Scope

A Capacity Provisioning policy includes a maximum defined capacity scope which specifies the total Defined Capacity that can be added by all the rules contained in the policy. A capacity limit in the Maximum Defined Capacity Scope is identified

by a system and sysplex name. The management of the system's Defined Capacity changes the limit of the LPAR in which the system is running.

Each rule also contains a defined capacity scope that restricts the capacity that can be increased by that rule. If the defined capacity scope of a rule includes restrictions for a system, a maximum defined capacity limit for the same system is needed, otherwise Defined Capacity is not increased for this system.

Specifying a capacity limit in the maximum defined capacity scope is optional. It is necessary if the policy intends to allow Defined Capacity increases for the system.

For each capacity limit in the maximum defined capacity scope, you must also specify the increments by which Defined Capacity is to be increased. Defined Capacity increments are used for workload-based capacity increases. You can define one value for the first increase (primary increment) and one value for all further increases (secondary increments). Defined Capacity increments are specified in MSU. The default increment value is one MSU. Neither the primary nor the secondary increments must exceed the maximum increase specified as Max. Increase. If the next workload-based increment would exceed the capacity allowed by the defined capacity scope, the next increment is limited to increase the allowed remaining capacity.

Table 6 shows an example of a maximum defined capacity scope with a limit related to system SYS0 in sysplex PLEX0.

Table 6. Maximum defined capacity scope

System	Sysplex	Max. Increase (MSU)	Primary Increment (MSU)	Secondary Increments (MSU)
SYS0	PLEX0	100	30	20

The limit specifies that Defined Capacity can be increased by up to 100 MSU. The first workload-based increment for that system would increase the Defined Capacity by 30 MSU, all following increments by 20 additional MSU each.

Maximum Group Capacity Scope

A Capacity Provisioning policy includes a maximum group capacity scope which specifies the total Group Capacity that can be added by all the rules contained in the policy. In the scope you define limits for individual capacity groups. A capacity group is identified by the name of the capacity group and the name of the CPC on which the capacity group is defined.

Each rule also contains a group capacity scope that restricts the capacity that can be increased by that rule. If the group capacity scope of a rule includes restrictions for a capacity group, a maximum group capacity limit for the same group is needed, otherwise Group Capacity is not increased for this group. Group capacity scopes only take effect if at least one system running in an LPAR belonging to that capacity group is observed by the Provisioning Manager.

Specifying a capacity limit in the maximum group capacity scope is optional. It is necessary if the policy intends to allow Group Capacity increases for that capacity group.

For each capacity limit in the maximum group capacity scope, you must also specify the increments by which Group Capacity is to be increased. Group capacity

increments are used for workload-based capacity increases. You can define one value for the first increase (primary increment) and one value for all further increases (secondary increments). Group capacity increments are specified in MSU. The default increment value is one MSU. Neither the primary nor the secondary increments must exceed the maximum increase specified as Max. Increase. If the next workload-based increment would exceed the capacity allowed by the group capacity scope, the next increment is limited to increase the allowed remaining capacity.

Table 7 shows an example of a maximum group capacity scope with a limit for capacity group GRP0 on CPC0.

Table 7. Maximum group capacity scope

Group	CPC	Max. Increase (MSU)	Primary Increment (MSU)	Secondary Increments (MSU)
GRP0	CPC0	250	30	20

The limit specifies that its Group Capacity can be increased by up to 250 MSU. The first workload-based increment for that group would increase the Group Capacity by 30 MSU, all following increments by 20 additional MSU each.

Rules

A provisioning rule contains a set of provisioning conditions and scopes which define provisioning limits for different types of capacity. The provisioning scopes restrict the capacity that can be provisioned by the conditions in the rule.

- The processor scope restricts the temporary processor resources which can be activated for CPCs.
- The defined capacity scope defines the maximum amount of MSU by which the Defined Capacity for a z/OS system can be increased.
- The group capacity scope defines the maximum amount of MSU by which the Group Capacity for a capacity group of a CPC can be increased.

Provisioning conditions describe the situations in which capacity changes are allowed. Two types of conditions are supported:

- *Time condition* specifies time periods during which additional capacity can be provisioned.
- *Workload condition* in addition identifies work which is eligible to cause provisioning of additional capacity, and the conditions under which provisioning can be triggered. Eligible work is specified according to the workload model of the z/OS Workload Manager (WLM).
- *Utilization condition* specifies processor utilization levels of specific CPCs which qualify for the provisioning of additional temporary capacity, and the processor types which should be managed. Utilization levels are specified in terms of shared physical processor pool busy-rates.

If only time conditions but no workload conditions or utilization conditions are defined, the Provisioning Manager executes a scheduled provisioning and deprovisioning of additional capacity up to the Max. Increase limit of the scope, ignoring increments.

A rule specifies the amount of additional capacity that can be added and the conditions under which an increase is allowed. In the provisioning scopes you can limit the additional capacity that can be provisioned for that rule. You can use time

conditions to select periods when you expect significant capacity shortages, and, optionally, you can identify triggers such as a WLM service class with an associated application or processor utilization limits of a certain CPC. A workload-based trigger will only activate additional capacity or increase capacity within periods specified by the time conditions and when at least one of the specified service classes is suffering by processor shortage. A utilization-based trigger requires that the indicated utilization for a given CPC is exceeded within the periods specified by the time conditions before it will activate additional temporary capacity. Rules should be defined for all applications for which additional capacity should be provisioned.

A rule can be enabled or disabled. Only enabled rules are considered by the Provisioning Manager. You can specify whether a rule is initially enabled or disabled. This status can be changed at runtime by using the **ENABLE POLICY** and **DISABLE POLICY** Provisioning Manager commands, as described in "ENABLE POLICY" on page 126 and "DISABLE POLICY" on page 122. Thus, you can specify different scenarios in the policy and activate only those scenarios that are relevant at a certain time, or you can temporarily disable provisioning, for example, if a maintenance period overlaps with a time condition in the policy.

Provisioning conditions

Optional workload conditions apply to all time conditions of the provisioning condition. For example, a workload condition can be defined for a service class SC1 associated with month-end jobs, and individual time conditions can be defined to cause provisioning on January 31st, February 28th, and so on. To consider workloads running on different sysplexes or systems, several workload conditions can be specified. For example, service class SC2 could be specified to trigger provisioning when running in sysplex PLEX1 and service class SC3 to trigger provisioning on system SYS2 only when running in sysplex PLEX2.

The same applies to optional utilization conditions: each of them can specify triggers for different CPCs and processor types and each of them is combined with every time condition of the same rule.

All capacity included in the provisioning scopes of a rule are shared by all conditions within that rule. If you want to provision a different set of capacity for a condition, create a new rule for this condition.

Provisioning conditions can be enabled or disabled in the same way as rules. Only enabled provisioning conditions are considered by the Provisioning Manager. You can specify in the policy whether a provisioning condition is initially enabled or disabled. This status can be changed at runtime, in the same way as rules, by using the **ENABLE POLICY** and **DISABLE POLICY** Provisioning Manager commands. In this way, you can specify different scenarios in the policy and activate only those scenarios that are relevant at a particular time. You can also temporarily disable part of the policy, for example, if a maintenance period overlaps with a time condition.

Time conditions

A time condition defines one or several periods during which the provisioning of additional capacity is allowed.

You can choose between two different types of time conditions:

- A nonrecurring time condition specifies a continuous period of time
- A recurring time condition describes weekly repeating periods

Nonrecurring time condition: A nonrecurring time condition specifies one period that has a clear start and end point in time. It is defined by the following parameters:

Name The time condition identifier within the policy. It must be unique across all nonrecurring and recurring time conditions.

Start time
The time at which the Provisioning Manager can start to provision additional capacity.

Deadline
The latest time when provisioning of additional capacity is allowed. Additional capacity that has already been provisioned can remain active until the end time or until the capacity is no longer needed.

End time
The time at which the Provisioning Manager starts to deprovision additional capacity.

Figure 3 describes two nonrecurring time conditions and shows how the Provisioning Manager interprets them in combination with a workload or a utilization condition.

Name	Start time	Deadline	End time
TC1	03/15/08 8:00 AM	03/18/08 10:00 AM	03/19/08 10:00 AM
TC2	10/28/08 8:00 AM	10/28/08 4:00 PM	10/30/08 11:59 AM

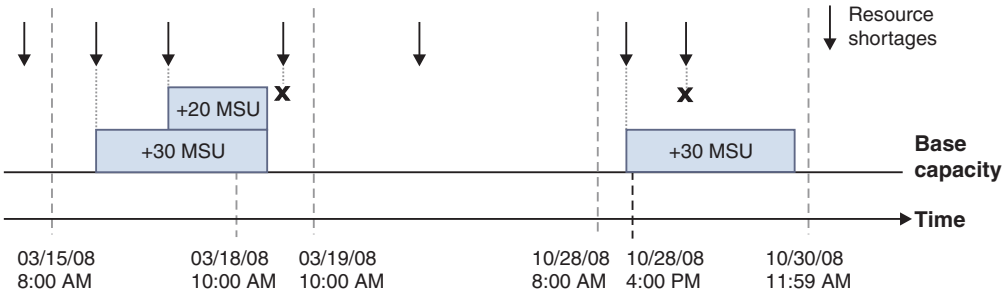


Figure 3. Nonrecurring time condition semantics

On the left, you see the effect of time condition TC1. Resource shortages are only considered between the start time and the deadline; resource shortages between the deadline and the end time cannot trigger activation of additional capacity. The boxes represent additionally provisioned general purpose capacity, with primary activations of 30 MSU and a secondary activation of 20 MSU, both set in the policy. On the right, you see the effect of time condition TC2. In this condition, the period between the start time and the deadline is very short compared to the period between the deadline and the end time. This means that additionally provisioned capacity can remain active for a longer period but cannot be increased after the deadline.

Recurring time condition: A recurring time condition describes weekly repeating periods. The periods describe the time of the day when provisioning of additional capacity is allowed and the days of the week to which these times apply.

A recurring time condition is defined by the following parameters:

Name The time condition identifier within the policy. It must be unique across all nonrecurring and recurring time conditions.

Start date
The date of the first day on which the Provisioning Manager can provision additional capacity.

End date
The date of the last day on which the Provisioning Manager can provision additional capacity.

Allowed days of the week
One or several days of the week that are eligible for provisioning of additional capacity by the Provisioning Manager.

Start time
The time on each selected day at which the Provisioning Manager can provision additional capacity.

Deadline time
The time on each selected day from which no further capacity is provisioned. Additional capacity that is already provisioned remains active until the end time or until the capacity is no longer needed.

End time
The time on each selected day at which the Provisioning Manager starts to deprovision additional capacity.

Deprovisioning of additional capacity assumes that it has been active for at least a minimum activation time, as specified in the Capacity Provisioning control parameters in Table 20 on page 54. If not, deprovisioning is delayed to fulfill that duration.

Name	Start date	End date	Start time	Deadline time	End time	Allowed days of the week
RTC1	03/15/08	03/17/08	8:00 AM	10:00 AM	11:00 AM	Saturday, Sunday

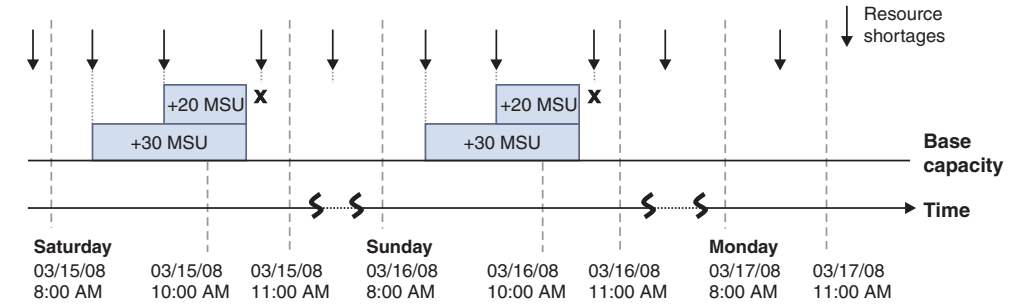


Figure 4. Recurring time condition semantics

Figure 4 describes a recurring time condition that covers the weekends: all Saturdays and Sundays from start date until end date are eligible for provisioning of additional capacity. Provisioning can occur on any of these days between the start time and the deadline time. The boxes represent additionally provisioned general purpose capacity, with a primary activation of 30 MSU and a secondary activation of 20 MSU, both set in the policy. Resource shortages that occur at a

different day of the week than the selected ones or outside the range that is specified by start date and end date do not trigger provisioning. Provisioned capacity is deprovisioned as soon as the end time is reached. The example assumes that minimum activation time for the resource type is short enough to allow deprovisioning not later than at end time.

Scheduled activations

If a provisioning condition contains time conditions but lacks of workload conditions and utilization conditions, the Provisioning Manager schedules provisioning and deprovisioning of additional capacity as specified by the provisioning scopes of the rule. The maximum additional allowed capacity in the provisioning scopes is provisioned either at start time, or if the Provisioning Manager has been started or the policy has been activated after start time, but not later than at time condition deadline. The additional capacity is deprovisioned at the specified end time.

Figure 5. Time scheduled activations

Name	Start Time	Deadline	End Time
TC1	03/15/08 8:00 AM	03/18/08 10:00 AM	03/19/08 10:00 AM
TC2	10/28/08 8:00 AM	10/28/08 4:00 PM	10/30/08 11:59 AM

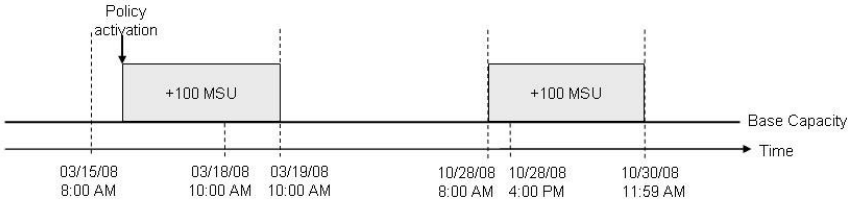


Figure 5 describes two nonrecurring time conditions and shows how the Provisioning Manager processes them, when they are not combined with workload conditions or utilization conditions, as scheduled activations. Time condition TC1 is handled as soon as the Provisioning Manager is ready to process it or as soon as the policy and its policy elements are active. In this example policy activation happens after TC1 start time, and only then the additional capacity of 100 MSU, which corresponds to the policy's Capacity limits for the suffering resource, is activated at once. At the end time of time condition TC1 the additional capacity is fully deprovisioned. Provisioning corresponding to time condition TC2 happens exactly at start time and deprovisioning happens at the end time of TC2.

Workload conditions

Use Capacity Provisioning to define business-critical work as being eligible for provisioning by defining workload conditions. The concept of a workload condition is based on the WLM service class model:

In a Workload Manager (WLM) service definition, work is assigned to WLM service classes, which are associated with goals, such as a response time.

A service class period describes how an amount of work is managed by WLM. It includes a duration, a goal, and an importance. A service class can be comprised of multiple periods. If the work is not completed within the specified duration, it falls through to the next period.

The importance of a service class period describes its business importance. If not all goals can be met, service classes with a lower importance are provided with less capacity to the benefit of service classes with higher importance.

Service classes are defined for the entire sysplex. The same or different service definitions can be active on different sysplexes.

For more information about WLM service classes, see *z/OS MVS Planning: Workload Management*.

A Capacity Provisioning workload condition specifies service class periods eligible for provisioning with several parameters:

Name The name of the workload condition uniquely identifies it within the policy and allows Provisioning Manager reports to reference it.

System
The z/OS system to which the workload condition applies.

Use **`Any in sysplex`** (*) to specify that the workload condition applies to all systems of the specified sysplex.

Sysplex
The sysplex to which the specified z/OS system belongs.

Use **`Any`** (*) to specify that the workload condition applies to all sysplexes observed by the Provisioning Manager.

Importance Filters
Eligible service class periods are ranked by their importance. They assign service class periods by importance level to sets of provisioning criteria; for more information, see "Provisioning criteria" on page 28. The Provisioning Manager checks for additional capacity for all service class periods with an importance level that is equal to or higher than the specified value. You can define separate provisioning criteria for each importance level.

The specification of the importance filter is optional; as an alternative, you can specify included service classes.

For more information about importance filters, see "Importance filter" on page 32.

Included Service Classes
This optional parameter specifies eligible service class periods by name and period. The Provisioning Manager monitors these periods and takes required provisioning action on behalf of them. Separate provisioning criteria can be defined for each service class period.

Included Service Classes are specified by service class period filters; see "Service class period filter" on page 32.

Excluded Service Classes Filter
This optional parameter specifies ineligible service class periods by name and period. They are excluded from the set of service class periods previously defined by an importance filter and are not considered by the Provisioning Manager.

Excluded Service Class Filters are specified by service class period filters; see "Service class period filter" on page 32.

At least one importance filter or included service class period must be defined for a workload condition to take effect.

The Provisioning Manager determines service class periods to be considered for provisioning as follows:

1. Service class periods with the specified or a higher importance on the specified system in the specified sysplex are chosen first.
2. This set is then extended with the periods contained in the included service classes.
3. Service class periods contained in the excluded service classes are then removed from the set.

All service class periods remaining in the set are monitored and can trigger provisioning. Whenever a new WLM service policy is activated, the set of observed service class periods is redetermined.

Make sure that service classes and classification rules are properly defined in WLM before using the service classes as provisioning triggers for Capacity Provisioning.

Provisioning criteria: The Provisioning Manager uses two major indications to detect when a service class period is impacted by insufficient processing capacity: the performance index (PI), and the resource demand of the service class period. The resource demand of a service class period can be detected automatically by the Provisioning Manager. You define PI thresholds in the provisioning policy as provisioning criteria that are associated with importance filters or included service classes.

The PI provided by WLM is calculated as the ratio of the specified goal of a service class period to the measured response time or velocity of the work running in this period. A PI of 1.0 indicates that the work is meeting the goal. A PI lower than 1.0 indicates that the work in the service class period exceeds the goal. A PI higher than 1.0 indicates that the goal is not fulfilled. In practice, a PI higher than 1.0 might be adequate for your installation, so you can assign a sufficient or appropriate target PI threshold to each service class period. Capacity Provisioning considers an action if the PI of a defined service class period is higher than the target PI for a specified amount of time. The PI that exceeds the limit for the specified amount of time indicates that WLM or, when active, Intelligent Resource Director (IRD), have not been able to resolve the bottleneck by shifting access to processor resources. The speed at which the Provisioning Manager is to react if a PI exceeds tolerance is also installation-dependent. Individual provisioning criteria can be defined for eligible service classes within the workload condition.

Importance filters and included service classes filters have the following parameters in common:

Table 8. Provisioning criteria

Provisioning criteria	Explanation
Provisioning PI	The value of the performance index, at or above which the Provisioning Manager considers the service class period to be suffering.

Table 8. Provisioning criteria (continued)

Provisioning criteria	Explanation
Provisioning duration	The duration (in minutes) that the performance index must exceed the specified provisioning PI before the Provisioning Manager considers the service class period to be suffering. The actual reaction time can be slightly longer than the specified duration. For example, when the performance monitor sampling interval is not aligned with the specified duration. If you choose a short provisioning duration of less than 5 minutes and require a rapid provisioning reaction from the Provisioning Manager, you might consider to downward adjust the Capacity Provisioning control parameter Analyzer.CycleTime, which is explained in Table 20 on page 54.
Deprovisioning PI	The value of the performance index which the Provisioning Manager considers the service class period to no longer be suffering. The deprovisioning PI must be lower than the provisioning PI.
Deprovisioning duration	The time (in minutes) that the performance index must be lower than the deprovisioning PI before the Provisioning Manager considers the service class period to be no longer suffering.
PI Scope	Indicates if the provisioning PI and deprovisioning PI values refer to the sysplex PI or the system PI of the service class period. **System** The system PI is the performance index of the service class period on each system in the sysplex. It is also referred to as the local performance index (local PI). This is the default setting. **Sysplex** The sysplex PI is the performance index of the service class period within the sysplex. A sysplex PI should only be chosen if all systems joining the sysplex are defined to the provisioning domain. If you choose PI scope "Sysplex", the Provisioning Manager monitors in addition to monitoring the sysplex PI also the system PI. Only if the monitored system is actually suffering, the Provisioning Manager starts to take actions. **Note:** Some system monitoring products can only display the sysplex PI. If you use z/OS RMF, you can find the local PI in the • RMF Monitor III Data Portal SYSINFO report • RMF Monitor III as metric 0x8D1020 • RMF WLMGL report if – The SYSTEMS option is in effect – Data from a single system only is provided

Provisioning PI and provisioning PI duration are used by the Provisioning Manager to detect whether observed service class periods are impacted by insufficient processing capacity. Before any actions are taken, the Provisioning Manager considers the resource demand of the service class period to ensure that the activation of additional processing capacity can improve the PI. Deprovisioning PI and deprovisioning PI duration are used by the Provisioning Manager to detect when a service class period no longer needs help.

For example, assume that a workload condition is specified including ONLINE service class. This condition is defined with one period of WLM service definition WLMSD, a provisioning PI of 1.8, and duration of 10 minutes, and a deprovisioning PI of 1.2 and duration of 10 minutes. If the PI of the service class period changes within a defined time condition, as shown in Figure 6, the Provisioning Manager detects three instances in which the provisioning PI criteria are fulfilled. For the first instance the Provisioning Manager provisions additional capacity by a primary activation amount as set in the policy, which is 20 MSU in this example. For the second instance the additional capacity corresponds to a secondary activation value of 30 MSU. The third instance is ignored, because it occurs after the deadline. The Provisioning Manager also detects an instance in which the deprovisioning PI criteria are fulfilled. The Provisioning Manager then decides that ONLINE service class no longer needs additional capacity and deprovisions it.

Provisioning criteria parameters
Name: PT1
Sysplex: PLEX1
System: SYSA
Importance Filter: Importance <= 2 with PI >= 1.8 for 10 min until PI <= 1.2 for 20 min
Included Service Class Periods: ONLINE.1 in WLMSD with PI >=1.8 for 10 min until PI <= 1.2 for 10 min
Excluded Service Class Periods: BACKUP.1 in WLMSD

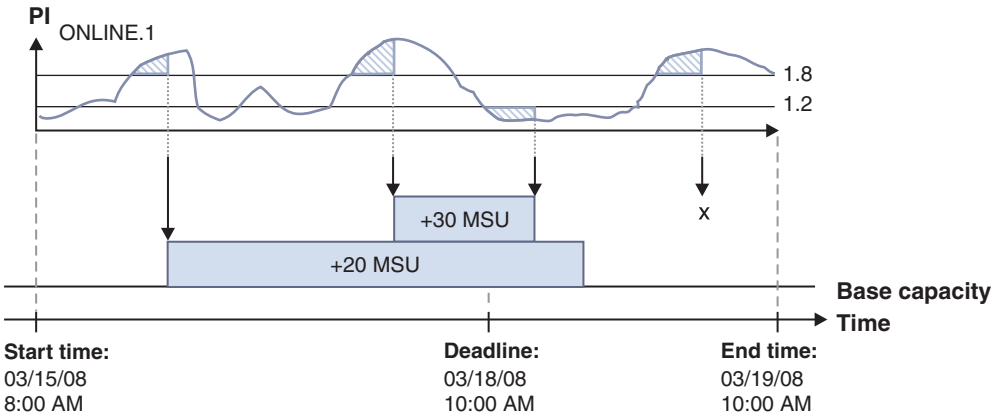


Figure 6. Provisioning criteria semantics

Moving average PI: The performance index of many workloads can change rapidly, for example because the amount of work varies, or because additional capacity becomes available in the system. If the provisioning duration includes several observation intervals, such as RMF MINTIME, it may be difficult to encounter a contiguous number of monitoring intervals where the PI exceeds the provisioning PI for the entire provisioning duration.

Managing a workload via moving average PI can account for that behavior. When the actual performance index of a workload decreases below the provisioning PI for a short time the moving average PI can still exceed that limit. Consequently, high PI values in the provisioning duration might be recognized more reliably.

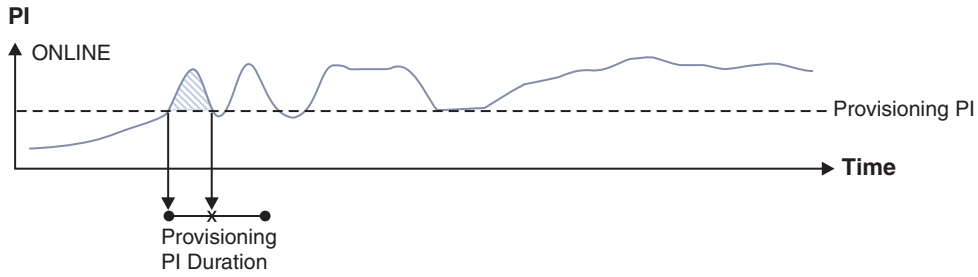


Figure 7. Fluctuating workload missing provisioning criteria

Figure 7 shows a fluctuating workload that does not meet the provisioning criteria because during the specified provisioning PI duration, the PI temporarily drops below the provisioning PI.

Optionally, the Provisioning Manager can average the PI provided by the monitoring component for a service class period. In this case the Provisioning Manager calculates a *moving average PI*. The actual PI pattern is averaged by using an exponentially weighted moving average (EWMA) function. The calculation considers current PI observations (interval t in the formula) as well as all preceding PI observations (as far as back to the first observed interval 0 in the formula) of a continuous time series, and weights the values with a user-specified smoothing factor as shown in Figure 8.

$$PI_{\underset{mov.avg.}{t}} = \omega \sum_{i=0}^{t-1} (1-\omega)^i PI_{t-i} + (1-\omega)^t PI_0$$

Figure 8. Exponentially weighted moving average (EWMA) PI formula

The graph that results from moving average smoothening is characterized by a more evenly value pattern; it might allow for activations in situations that would not be taken into account without the additional smoothening.

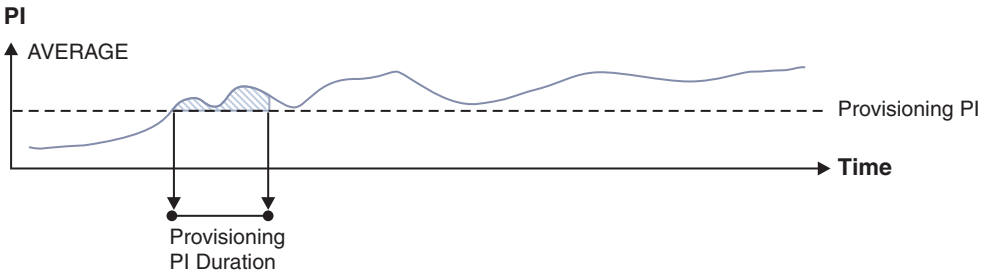


Figure 9. EWMA smoothened PI pattern with workload meeting provisioning criteria

Figure 9 shows the PI pattern, smoothened by the EWMA PI formula.

The smoothening algorithm delays the moments when the provisioning PI limit or the deprovisioning PI limit are being crossed; the smaller the smoothing factor, the more PI limit crossings are delayed. Hence, related capacity activations or deactivations are also delayed. Reducing provisioning or deprovisioning PI durations adequately can compensate for this delay.

The formula assumes that a contiguous series of PI values (time interval i=0 to t) is available, meaning that PI values are reported for every interval. For patterns with short gaps in workload, the formula disregards all PI observations that precede the last gap. The series of contiguous PI values can also be interrupted after subcapacity changes to the managed hardware, when the monitoring component, such as RMF, needs to recalculate the provided data.

To prevent high PI values from distorting the computed moving average PI pattern, a maximum capping PI value must be specified. This value limits the PI values that are considered to the maximum value specified by the capping PI. PI values that exceed that limit are replaced by the capping PI during computation of the moving average PI. That protects the resulting moving average PI graph from very high PI values that would have long-term effects. By default, the PI is capped at 5.5. If needed, it can be set to a different value, or moving average PI capping can be disabled.

Management on behalf of moving average PI and the capping value of the maximum capping PI is set globally. Hence, both values, the smoothing factor and the maximum capping PI, apply to all observed workloads.

For setting the moving average weight factor, see key **SystemObservation.MovingAveragePiWeight** in Table 20 on page 54.

For setting the moving average capping, see key **SystemObservation.MovingAveragePiCapping** in Table 20 on page 54.

Importance filter: An importance filter selects service class periods based on their importance. It includes the following parameter.

Importance
The relative importance of the service class periods. All service class periods with an importance value less than or equal to the specified match the filter unless another importance filter applies.

An importance filter also includes Provisioning Criteria PI values indicating when service class periods that match the importance filter are considered to be suffering. For more information about Provisioning Criteria PI values, see “Provisioning criteria” on page 28.

For example, if you specify importance value 3 in a filter, all service class periods with importance values 3, 2, and 1 match the filter and the specified provisioning criteria is applied to them. To define different provisioning criteria only for service class periods of importance value 1, you define another importance filter with the new criteria. The filter for importance value 3 then applies only to service class periods with importance values 3 and 2, and the filter for importance value 1 applies only to service class periods with importance value 1.

Service class period filter: Included and excluded service class periods are identified by service class period filters, which contain criteria that a service class period must match to be considered or ignored by the Provisioning Manager. These filters include the following parameters:

Service Definition
Name of the WLM service definition. The specified service class periods are only considered if this WLM service definition is installed. You can specify **Any service definition** (*) to include all WLM service definitions.

Service Policy
Name of the service policy within the WLM service definition. The specified service class periods are considered if a service policy with that name is activated. You can specify **Any service policy** (*) to include all service policies matching the other criteria.

Service Class
Name of the service class. You can specify **Any service class** (*) to include all service classes matching the other criteria.

Period
The period of the service class to be considered. In an included service class filter, this period and all periods with a lower period number are considered eligible to trigger provisioning. If a service class has fewer periods than this number, all periods are considered.

In an excluded service class filter, this period and all periods with a higher period number are excluded from provisioning.

Utilization conditions

Use Capacity Provisioning to allow for provisioning of additional capacity based on the current CPC utilization. Utilization conditions are specified for the CPC wide physical processor utilization, broken down by CPU types such as CP, zIIP and zAAP. You can specify a utilization limit above which additional temporary (On/Off CoD) capacity should be provisioned. Defined capacity or group capacity can not be managed by means of utilization conditions.

A Capacity Provisioning utilization condition is specified by several parameters:

Name The name of the utilization condition uniquely identifies it within the policy and allows Provisioning Manager reports to reference it.

CPC The name of the CPC of which the physical utilization needs to be monitored and where additional temporary capacity has to be provisioned. The Provisioning Manager will collect monitoring data about this CPC through an observed system. At least one system of the active Domain Configuration needs to belong to this CPC and needs to be observed by the Provisioning Manager.

Processor type
The type of the processor for which the physical utilization of the shared processor pool needs to be monitored and which needs to be provisioned in case of necessity.

Table 9. Provisioning criteria

Provisioning criteria	Explanation
Provisioning utilization	The value of the physical utilization in the shared processor pool, at or above which the Provisioning Manager assumes that the CPC is suffering from a processor bottleneck and provisioning of temporary capacity should occur.

Table 9. Provisioning criteria (continued)

Provisioning criteria	Explanation
Provisioning duration	The minimum duration (in minutes) that the physical utilization must be at or exceed the specified provisioning utilization value before the Provisioning Manager detects demand for additional temporary capacity. The actual reaction time can be slightly longer than the specified duration. This can happen when the performance monitor sampling interval is long or the Provisioning Manager configuration parameter Analyzer.CycleTime value is large.
Deprovisioning utilization	The value of the physical utilization of the shared processor pool, at or below which the Provisioning Manager assumes that the CPC is no longer suffering from a processor bottleneck and can start deprovisioning of temporary capacity.
Deprovisioning duration	The time (in minutes) that the physical utilization must be at or below the deprovisioning utilization before the Provisioning Manager detects that the provisioned temporary capacity is no longer needed.

In a utilization condition the CPC to be observed by the Provisioning Manager is defined. At least one z/OS system running in this CPC has to be defined in the active domain configuration. If more than one z/OS system for this CPC is defined in the active domain configuration, the Provisioning Manager will redundantly monitor all of these systems.

If time conditions are defined in a provisioning condition, but workload conditions and utilization conditions are not, the Provisioning Manager schedules provisioning and deprovisioning of additional capacity as specified by the provisioning scopes of the rule.

For example, assume a utilization condition which is specified for a CPC CPC1 and for processor type CP (general purpose processor). The condition is defined with provisioning utilization 90% and duration of 7 minutes, and a deprovisioning utilization of 65% and a duration of 15 minutes.

The Provisioning Manager will start to gradually activate additional temporary capacity as soon as the physical utilization of the shared CP processor pool on CPC CPC1 exceeds 90% for at least 7 minutes. The Provisioning Manager will disregard which workload has caused this processor shortage and will continue to provision more temporary capacity as long as the provisioning criteria are fulfilled.

Utilization conditions tend to provision capacity more aggressively than workload conditions. If we assume that the provisioning criteria of a workload condition and the provisioning criteria of a utilization condition would trigger at the same time, the utilization condition would immediately detect a demand, disregarding which system or workload could benefit from the additional capacity. A workload condition however would run an additional assessment where it checks other aspects of the suffering workload and system, to make sure that it really can absorb and benefit from the additional capacity. A a result a workload condition could delay or even prevent provisioning of additional capacity, whereas a utilization condition already might have provisioned.

Capacity Provisioning Manager

The Provisioning Manager program is located in `SYS1.SIEALNKE` and in the file system under `/usr/lpp/cpo`. It must be activated through a started task procedure.

It works with the resource definitions from the domain configuration and the workload, utilization and time conditions from the policy. These elements can be created by using the Capacity Provisioning Management Console.

The Provisioning Manager observes z/OS systems by connecting to the CIM servers on these systems, and uses these connections:

- To retrieve the required capacity and performance metrics
- To monitor the number of logical processors

To monitor and control resources, a connection to the HMC must be established. To use this connection, the BCPii component of z/OS must be set up and configured. For details, see "Defining the connection to the hardware" on page 66.

The processing mode of the Provisioning Manager (see "Processing modes" on page 38) is controlled by operator commands. The processing actions available are described in "Processing activations and deactivations" on page 40. For the method of entering these commands, see "Provisioning Manager commands" on page 74, and for the actual commands, see Chapter 8, "Provisioning Manager command reference," on page 103.

To ensure availability, you can install and set up the Provisioning Manager on more than one system. Run the Provisioning Manager on a system that is included in the domain. Only one instance of the Provisioning Manager can be active at a time. You can define a restart policy for situations when the Provisioning Manager needs to stop, or when you need to shut down the system on which the Provisioning Manager is currently running. You can use your automation product or the z/OS Automatic Restart Manager (ARM) for this purpose.

Processing the domain configuration

When connected to the HMC, the Provisioning Manager retrieves a list of available CPCs. This list is correlated to the list of CPCs that is defined in the active domain configuration. Any CPCs that are defined in the domain configuration that are not listed as available by the HMC are marked as "not correlated". The Provisioning Manager regularly checks for additional CPCs. When it finds them, it checks whether they can be correlated to CPCs that are defined in the domain configuration.

After the CPC lists have been correlated, the Provisioning Manager retrieves specific information about all CPCs that are common to both lists. When this information shows that a CPC has temporary capacity that can be managed by the Provisioning Manager, the CPC is considered valid for activation and deactivation requests of temporary capacity. This is indicated by message CPO3019I. In addition the Provisioning Manager retrieves Defined Capacity and Group Capacity information for all LPARs defined on a CPC. Once this information has been retrieved for all LPARs running z/OS, the according z/OS systems are considered valid for the provisioning of Defined Capacity or Group Capacity. This is indicated by message CPO3027I.

There can be multiple On/Off CoD records installed on a CPC. If there is just one On/Off CoD record, the Provisioning Manager can find the record. If there are multiple records, the identifier of the record to be used should be specified. Otherwise the Provisioning Manager selects an arbitrary one. If the Provisioning Manager manages only Defined Capacity or Group Capacity, no On/Off CoD record is required for a CPC.

If a CPC is enabled, the Provisioning Manager can automatically activate or deactivate temporary capacity and increase and decrease defined capacity on it. If the CPC is disabled, capacity on the CPC can only be changed manually by using Provisioning Manager commands.

When the Provisioning Manager is in the processing modes `analysis`, `confirmation`, or `autonomic`, and the observed system is enabled, the Provisioning Manager tries to connect to the CIM server on the observed system. A network connection to the observed system must be available, and all required services on this system must be configured and running. These services include the CIM server itself and the providers of workload or utilization metrics. If RMF supplies the CIM providers, RMF and the distributed data server (DDS) must be operating and the RMF CIM provider must have a connection to the DDS.

Once the connection to the observed system has been established, the Provisioning Manager retrieves configuration data about the CPC and LPAR on which the system runs. On behalf of this data the Provisioning Manager verifies that the responding z/OS system is the one specified in the domain configuration and that it is running on a CPC that is defined in the domain configuration. After this has been verified, information about Defined Capacity and Group Capacity is obtained for the LPAR that hosts the observed system.

When this information shows that Defined Capacity is turned on for the LPAR or that the LPAR belongs to a group for which Group Capacity is turned on, the system is considered valid for the provisioning of Defined Capacity and/or Group Capacity by the Provisioning Manager.

Performance information related to the CPC, the LPAR, and the WLM service classes is then retrieved for the workload or utilization condition that is defined in the policy. Because this information changes regularly, the rate at which the values are retrieved is synchronized with the change intervals. If RMF is used, this interval is the value of the RMF MINTIME option.

There are some situations when the metric data is not valid. In particular, changes to the WLM policy can cause disruption. Such changes are due to updating and reactivating the current WLM policy, or to installing a new WLM service definition. In these situations, the Provisioning Manager must reevaluate the workload or utilization situation. This might result in some history no longer being valid, and if capacity is currently provisioned by the Provisioning Manager it might be deprovisioned.

The state information about the domain configuration and observed systems and managed CPCs is reported by the Provisioning Manager in the domain configuration report. For more information about this report, see “The domain configuration report” on page 81. The enabled state of the system and the CPC domain configuration elements can be changed by using the **`ENABLE CONFIGURATION`** and **`DISABLE CONFIGURATION`** Provisioning Manager commands, as described in “ENABLE CONFIGURATION” on page 125 and “DISABLE CONFIGURATION” on page 121.

Processing policy time conditions

The provisioning condition that contains the time condition can also contain workload conditions or utilization conditions. Based on the provisioning PI durations of all associated workload conditions and based on the provisioning utilization durations of all associated utilization conditions, the Provisioning Manager calculates the time at which observation of the workload or utilization

must start in order to allow provisioning at the start time, if necessary. This calculated time is referred to as the observation start time.

Note: The Provisioning Manager can run at different times on different systems, and it normally observes several other systems. Because these systems can be running in different time zones, the Provisioning Manager calculates and reports all times based on a common time zone. This common time zone is the Coordinated Universal Time (UTC).

One of the parameters of a nonrecurring time condition is the start time, which defines when the provisioning of additional capacity can start. For a recurring time condition, start time is the combination of start time and a selected day of the week that occurs between the start date and the end date. Recurring time conditions are processed as a set of nonrecurring time conditions.

The state of time conditions at runtime is calculated based on the current time. A time condition can be in any of the states in the following list, depending on the current time, the enabled state of the time condition, and the enabled state of the policy. The enabled state of the time condition depends on the enabled states of the provisioning condition, rule, and policy that contain this time condition. If all of these policy elements are enabled, the time condition is also enabled.

Table 10. Time condition states

Time condition state	**Meaning**
Pending	The current time is before the start time of the condition and no system observation is necessary.
Observing and enabled	The current time is after the observation start time and before the start time of the condition. The time condition is enabled, and one or more workload conditions or utilization conditions are defined that require systems to be observed. Systems that are referenced by associated workload conditions or utilization conditions are contacted to get performance information for further processing.
Observing and disabled	The current time is after the observation start time and before the start time of the condition. The policy is enabled but the time condition is disabled.
Active and enabled	The current time is after the start time and before the deadline of the condition, and the time condition is enabled. The Provisioning Manager may change the managed capacity based on the provisioning condition that contains the time condition.
Active and disabled	The current time is after the start time and before the deadline of the condition. The policy is enabled but the time condition is disabled. The managed capacity cannot be changed by the Provisioning Manager based on the provisioning condition that contains the time condition.
Drained and enabled	The current time is after the deadline and before the end time of the condition; the time condition is enabled. The Provisioning Manager cannot change the managed capacity but can maintain the current managed capacity based on the provisioning condition that contains the time condition.
Drained and disabled	The current time is after the deadline and before the end time of the condition. The policy is enabled but the time condition is disabled. The managed capacity cannot be changed by the Provisioning Manager based on the provisioning condition that contains the time condition.

Table 10. Time condition states (continued)

Time condition state	Meaning
Inactive	The current time is after the end time of the condition; for recurring time conditions the current time is after the end time of the last selected day of the week between the start date and the end date. The managed capacity cannot be changed by the Provisioning Manager based on the provisioning condition that contains the time condition.

The states of time conditions in the active provisioning policy can be reported by the Provisioning Manager in the policy report. For more information about this report, see "The policy report" on page 78. The enabled or disabled states of the rule and provisioning condition policy elements can be changed by using Provisioning Manager commands. For more information about these commands, see "ENABLE POLICY" on page 126 and "DISABLE POLICY" on page 122. The state of the active policy itself depends on the processing mode of the Provisioning Manager; for more information, see "Processing modes." The policy is disabled in manual mode and enabled in all other processing modes.

Processing modes

Depending on its processing mode the Provisioning Manager collects different amounts of data. Possible processing modes are:

Table 11. Processing modes

Mode	Explanation
Manual	You can issue manual activation and deactivation commands to the Provisioning Manager, but the Provisioning Manager does not activate additional capacity or increase capacity by itself. Use this processing mode if you want to manage capacity by yourself, or if you want to quiesce the Provisioning Manager. The active policy is not processed. The Provisioning Manager will not collect or display Defined Capacity or Group Capacity information.
Analysis	You are informed through console messages if any additional capacity is required. The Provisioning Manager processes the active policy, analyzes the workload on all accessible systems or available physical utilization data, and issues messages. The messages contain details on the maximum amount of capacity that is allowed by the policy. Like in manual mode, the Provisioning Manager does not activate additional capacity or increase capacity by itself. **Note:** In this mode, the Provisioning Manager does not check whether the CPC on which the workload is running or the physical utilization of a CPC allows activation of temporary capacity. It might report the need for additional capacity for CPCs that in fact do not have any temporary capacity that can be activated by the Provisioning Manager.

Table 11. Processing modes (continued)

Mode	Explanation
Confirmation	You are informed of proposed capacity changes through console messages, and you are asked to confirm the changes. The Provisioning Manager processes the active policy and the active domain configuration, analyzes the current workload situation on the observed systems or the physical utilization of a CPC, calculates which capacity is needed, and displays proposals. You can either accept or deny the proposed action. If you accept the proposal, the Provisioning Manager performs the action and continues processing based on the new capacity level. If you deny the proposal, the Provisioning Manager does not consider the resource for additional changes for a period of time. After this time, the Provisioning Manager considers the resource again and proposes the same or other actions according to the new situation. While the answer to the request message is pending, the Provisioning Manager continues checking for any necessary changes. If the previous requirement no longer exists, the request is canceled. If a new requirement has arisen, a new proposal is issued. If you need more information to answer the proposal, you can request reports from the Provisioning Manager: The workload report shows the actual workload situation, the utilization report shows the actual physical utilization situation. The domain configuration report displays the current capacity level of the resource.
Autonomic	The Provisioning Manager autonomically adjusts the capacity settings of the resources as determined through analysis of the workload situation on the observed systems and the physical utilization levels on monitored CPCs. The Provisioning Manager processes the active policy and the active domain configuration. When any provisioning is performed, the Provisioning Manager issues a message to the console to inform you which activities have been performed.

You can switch between these processing modes while the Provisioning Manager is running. Information about capacity level changes that were performed based on the policy is preserved in all modes. If you switch to a mode that allows provisioning, the Provisioning Manager continues to manage the capacity that it has provisioned already.

When you start the Provisioning Manager, you can pass the initial processing mode as a parameter. You can change the processing mode at any time by using the Provisioning Manager **SET DOMAIN** command. For more information, see "SET DOMAIN" on page 148. The current processing mode is reported by the Provisioning Manager in the domain report, described in "The domain report" on page 77.

Table 12 shows the requirements for each mode and the functions available in each mode.

Table 12. Processing modes: requirements and functions supported

Requirement/Function	Manual	Analysis	Confirmation	Autonomic
Domain configuration required	Yes	Yes	Yes	Yes
Policy definitions required	No	Yes	Yes	Yes

Table 12. Processing modes: requirements and functions supported (continued)

Requirement/Function	Manual	Analysis	Confirmation	Autonomic
RMF DDS or equivalent, and CIM server required	No	Yes	Yes	Yes
Provisioning and deprovisioning through Provisioning Manager commands	Yes, if the CPC is defined in the domain configuration	Yes	Yes	Yes
Observation of defined workloads or physical utilization of CPCs	No	Yes	Yes	Yes
Provisioning and deprovisioning recommen-dations or actions	No	Recommen-dations through console messages.	Console messages (WTOR) that might be accepted, rejected, or ignored.	Actions are implemented immediately.
On/Off Capacity on Demand record required to be installed	No	No	Yes	Yes
Defined Capacity or Group Capacity information reporting	No	Yes	Yes	Yes

Processing activations and deactivations

The actions defined in a provisioning policy can be based exclusively on a time schedule or additionally on workload conditions or on utilization conditions.

Schedule-based provisioning actions are defined by provisioning conditions that do not have a workload condition or a utilization condition. All available associated capacity is provisioned at the start time of the time condition. If there is not enough capacity available for provisioning at the start time, capacity that becomes available later, for example by replenishing the record of the managed CPC or by deactivating capacity that is not managed by the Provisioning Manager, can be provisioned at any time up to the deadline. All provisioned capacity is deprovisioned at the end time of the condition.

Workload-based provisioning actions are defined by workload targets for the observed systems. If the workload is impacted due to insufficient capacity, the Provisioning Manager provisions additional capacity, one by one, until the situation is resolved or the supply of resources is exhausted. In its decision the Provisioning Manager considers the performance index of the workload, compares it to the limits set in the workload conditions, and assesses which resources are needed and how they would help the workload. The assessment includes the check whether the LPAR that contains the system has sufficient logical processors or whether the processing weight is high enough to absorb more capacity.

Utilization-based provisioning actions are defined by physical processor utilization targets for affected CPCs. If the specified shared processor pool utilization levels are exceeded, the Provisioning Manager immediately provisions additional temporary capacity, one by one, until the situation is resolved or the supply of resources exhausted.

The Provisioning Manager performs capacity changes based on the active policy and the workload situation on the observed systems or the utilization situation on the affected CPCs, and monitors the effect these changes have on the workload situation or the CPC utilization. There are multiple mechanisms that need to adjust to the new capacity situation after a capacity change, for example PR/SM on the hardware side and WLM and IRD on the software side. These mechanisms take time to distribute the available capacity, hence the Provisioning Manager blocks the provisioned resource for a period of time, referred to as the blocking time. During this period of time, the Provisioning Manager does not consider the resources for further capacity changes after any provisioning or deprovisioning.

Provisioned capacity is not deprovisioned by the Provisioning Manager until it has been active for a minimum activation time, even if the workload situation of the observed systems or the processor consumption of the affected CPCs no longer requires the capacity, or if rules in the provisioning policy become inactive. This solution is designed to ensure that short-term fluctuations in the workload situation do not cause too many changes in the capacity level.

Information about the provisioning actions that are performed by the Provisioning Manager based on the active policy is reported in the activity report. For more information about this report, see "The activity report" on page 87.

Note: Manual changes to the capacity level are not listed in the activity report.

Processing of manually activated temporary capacity

The Provisioning Manager manages the record describing the temporary capacity. If necessary, you can manually activate and deactivate temporary capacity contained in this record. If you manually activate resources, they are not managed by the Provisioning Manager and you must deactivate them manually as needed. If you manually deactivate resources that have been activated by the Provisioning Manager, it is detected and reported. The Provisioning Manager continues to manage from the new activation level of the CPC, and, if needed, the resources can be activated again. You can manually change the activation level of the record at the HMC or by using Provisioning Manager commands.

Sometimes you activate capacity manually, for example, if the authorized capacity in the policy is insufficient or you need to activate the capacity before the workload suffers or the physical processor consumption rises steeply. In these situations you can pass the manually activated capacity to the Provisioning Manager for further handling. The Provisioning Manager then deactivates this capacity according to your instructions in the workload conditions or utilization conditions. You provide a duration with this command, that describes the minimum time the resources should be held active. This time is used for the additional capacity to manage and for all Provisioning Manager activated capacity. Later policy based activations of resources may override this time with the minimum activation time starting at the time of that activation. In this period, the Provisioning Manager is only allowed to activate additional capacity, if it is needed according to the policy and the workload or utilization condition. After that time, policy based management of all Provisioning Manager owned capacity takes place. For example, if there is more capacity owned by the Provisioning Manager than allowed by the policy, then the Provisioning Manager deactivates the not allowed capacity immediately.

Passing manually activated capacity to the management of the Provisioning Manager is done using the **MANAGE RESOURCE** command. For details, see "MANAGE RESOURCE" on page 130.

Processing of logical processor changes

The logical processor scope is only considered if the observed system is running with WLM LPAR CPU management turned off.

The Provisioning Manager recommends to configure logical processors online if the number of logical processors restricts the consumption of physical capacity. This situation can occur due to a shortage of logical processors or in combination with a physical activation. The Provisioning Manager recommends to configure logical processors offline if the number of logical processors of a system prevents a change to the physical capacity of the CPC on which the system is running. In this case, physical resources deactivation is postponed until logical processors have been configured offline. The Provisioning Manager does not attempt to optimize the number of logical processors for the consumed capacity. However, the HiperDispatch function can be used for that purpose.

Recommendations to take additional logical processors online or offline are issued in confirmation and autonomic processing modes. For more information, see "Logical processor scope" on page 19.

Processing Defined Capacity and Group Capacity

The Provisioning Manager processes Defined Capacity and Group Capacity as described in "Processing activations and deactivations" on page 40.

The premise for managing Defined Capacity is that a Defined Capacity other than 0 has been assigned to the LPAR that hosts a system which is observed by the Provisioning Manager. Otherwise the Provisioning Manager considers Defined Capacity to be turned off for the LPAR and will not perform any Defined Capacity management nor even allow manual changes of Defined Capacity using Provisioning Manager commands. The same premise applies for Group Capacity. Only when a group containing an LPAR that hosts a system which is observed by the Provisioning Manager has been assigned a Group Capacity value other than 0, Group Capacity for this group will be managed by the Provisioning Manager.

Throughout the following text the term *defined capacity* is used as a synonym for Defined Capacity as well as Group Capacity, unless both terms are specified explicitly.

For processing Defined Capacity, the Provisioning Manager monitors the *4-hour rolling average* provided by WLM for each observed system, and the *projected time until capping* of the LPAR calculated by the monitoring product. When the projected time until a system's *4-hour rolling average* would reach the Defined Capacity limit for its LPAR falls below the lead time configured for the Provisioning Manager, the Provisioning Manager considers the system to be eligible for an increase of Defined Capacity based on the current policy.

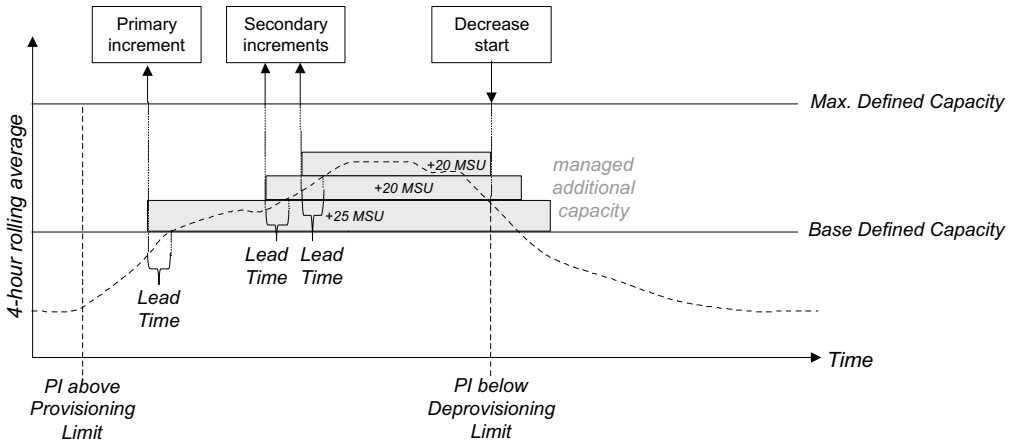


Figure 10. Processing Defined Capacity

For processing Group Capacity, the Provisioning Manager also monitors the *projected time until group capping* calculated by the monitoring product. A group is eligible for a capacity increase based on the current policy, when both the *projected time until capping* of the LPAR and the *projected time until group capping* fall below the lead time configured for the Provisioning Manager, and the Provisioning Manager has detected that an increase of Group Capacity would help a monitored system.

The lead time is configured through the Capacity Provisioning control parameter **`DefinedCapacity.LeadTime`**, which is explained in Table 20 on page 54.

The Provisioning Manager retrieves the assigned Defined Capacity values for observed systems during initialization and uses these values as the base for its management, referred to as the Management Base. Starting from the Management Base it increases Defined Capacity according to the scopes and conditions specified in the active policy. The amount of Defined Capacity that has been increased by the Provisioning Manager is also referred to as the Managed Capacity in the context of Capacity Provisioning. As such, only the Managed Capacity is also decreased by the Provisioning Manager. The amount of Defined Capacity that was assigned outside of the Provisioning Manager, including Defined Capacity that was manually increased using Provisioning Manager commands, will not be decreased by the Provisioning Manager. Nevertheless, the Provisioning Manager will always decrease the complete amount of Managed Capacity according to its policy. The following example illustrates this behavior:

Table 13. Example of Defined Capacity processing by the Provisioning Manager in autonomic mode

Action	Defined Capacity	Management Base	Managed Capacity
Initially assigned Defined Capacity in LPAR controls during Provisioning Manager initialization = 300 MSU	300 MSU	300 MSU	0 MSU
Provisioning Manager increases Defined Capacity by 100 MSU based on its policy	400 MSU	300 MSU	100 MSU

Table 13. Example of Defined Capacity processing by the Provisioning Manager in autonomic mode (continued)

Action	Defined Capacity	Management Base	Managed Capacity
Defined Capacity is manually increased by 50 MSU, for example using a Provisioning Manager command	450 MSU	350 MSU	100 MSU
Provisioning Manager increases Defined Capacity by 50 MSU based on its policy	500 MSU	350 MSU	150 MSU
Defined Capacity is manually decreased by 100 MSU	400 MSU	250 MSU	150 MSU
Provisioning Manager decreases Defined Capacity by 150 MSU based on its policy	250 MSU	250 MSU	0 MSU

As the Provisioning Manager remembers the Managed Capacity even across a restart, situations may occur where the Defined Capacity is manually decreased while the Provisioning Manager is down and when the Provisioning Manager comes up again, it will try to decrease the Defined Capacity to an unusual low value. To prevent this, the control parameters **DefinedCapacity.GlobalMinimumDefinedCapacity** and **DefinedCapacity.GlobalMinimumGroupCapacity** can be used to define the minimum Defined Capacity below which the Provisioning Manager must never decrease any Defined Capacity.

Decrease of Defined Capacity from outside of the Provisioning Manager, or by using Provisioning Manager commands, while the Provisioning Manager is running must be handled carefully. Because the Provisioning Manager will always try to decrease the complete amount of Managed Capacity according to its policy. If a change of Defined Capacity is significant below its previous value, the Provisioning Manager asks if he should set the Management Base to the new Defined Capacity value and set Managed Capacity to 0 or if he should adjust the Management Base according to the current Managed Capacity value. That will avoid a decrease to a very low Defined Capacity value. The following example illustrates this behavior:

Table 14. Example of Defined Capacity processing by the Provisioning Manager in autonomic mode

Action	Defined Capacity	Management Base	Managed Capacity
Initially assigned Defined Capacity in LPAR controls during Provisioning Manager initialization = 300 MSU	300 MSU	300 MSU	0 MSU
Provisioning Manager increases Defined Capacity by 200 MSU based on its policy	500 MSU	300 MSU	200 MSU
Defined Capacity is manually decreased by 90 MSU, for example using a Provisioning Manager command	410 MSU		200 MSU

Table 14. Example of Defined Capacity processing by the Provisioning Manager in autonomic mode (continued)

Action	Defined Capacity	Management Base	Managed Capacity
CPM message: New DC. Previous base was 300 MSU. Enter 1 to set base to 410 or 2 to set to 210 MSU. Answer 1 will set the Managed Capacity to 0 and the Management Base to the current Defined Capacity value. Answer 2 will continue the Defined Capacity Management with adapted Management Base.			
Answer 1	410 MSU	410 MSU	0 MSU
Provisioning Manager continues management but next action can be an increase only.			
Answer 2	410 MSU	210 MSU	200 MSU
Provisioning Manager continues management. For example: Decrease Defined Capacity by 200 MSU based on its policy	210 MSU	10 MSU	0 MSU

To define which change of Defined Capacity is considered to be significant the control parameter **DefinedCapacity.BaseToleration** can be set to any value between 0 – 100 Percentages. Its default value is 15%. That means that a Defined Capacity decrease of more than 15% is considered to be significant and that an adjustment of the Management Base and the Managed Capacity may be advisable. When this control parameter is set to 100% the Provisioning Manager will not ask for a Management Base adjustment.

Capacity Provisioning Management Console

The CPMC is a web based graphical user interface for Capacity Provisioning. You use this interface to work with provisioning policies and domain configurations. You can manage connections to the Provisioning Manager, and use them to transfer provisioning policies and domain configurations to the Provisioning Manager, or to query various status reports. The interface is described in Chapter 4, “Using the Capacity Provisioning Management Console,” on page 71.

z/OSMF Repository

The CPMC maintains a repository, that includes the Provisioning Manager connections, a set of domain configurations, and a set of provisioning policies. The repository is shared across all users authorized to access the Capacity Provisioning task; therefore, all users see the same list of domain configurations and the same list of provisioning policies.

Naming conventions

Policy names and configuration names must be unique within the repository where they are stored. The names of policy elements must be unique within that policy. If there is an element with the same name and type in the current policy an error is reported.

The length of names and the character set that can be used are restricted.

Named elements within a provisioning policy or a domain configuration, and elements referencing external entities such as a system name or the logical name of a CPC must conform to the rules defined in Table 15. In this table, the hyphen (-) indicates that any ASCII character within the range is valid.

Table 15. Naming restrictions

Name	Minimum length	Maximum length	Initial character	Subsequent characters
Domain	1	8	A - Z	A - Z, 0 - 9, #
Provisioning policy	1	8	A - Z	A - Z, 0 - 9, #
Provisioning rule	1	12	A - Z, a - z	A - Z, a - z, 0 - 9, #, _
Provisioning condition	1	12	A - Z, a - z	A - Z, a - z, 0 - 9, #, _
Time condition	1	12	A - Z, a - z	A - Z, a - z, 0 - 9, #, _
Workload condition	1	12	A - Z, a - z	A - Z, a - z, 0 - 9, #, _
Utilization condition	1	12	A-Z, a-z	A - Z, a - z, 0 - 9, #, _
WLM service definition	1	8	A - Z, a - z, 0 - 9, #, $, %, @	A - Z, a - z, 0 - 9, #, $, %, @, _
WLM service policy	1	8	A - Z, a - z, 0 - 9, #, $, %, @	A - Z, a - z, 0 - 9, #, $, %, @, _
WLM service class	1	8	A - Z, a - z, 0 - 9, #, $, %, @	A - Z, a - z, 0 - 9, #, $, %, @, _
Domain configuration	1	8	A - Z	A - Z, 0 - 9, #
System	1	8	A - Z, 0 - 9, #, $, @	A - Z, 0 - 9, #, $, @
Sysplex	1	8	A - Z, 0 - 9, #, $, @	A - Z, 0 - 9, #, $, @
CPC	1	8	A - Z, 0 - 9, #, $, @	A - Z, 0 - 9, #, $, @
CPC record ID	8	8	A - Z, 0 - 9	A - Z, 0 - 9
Description	0	128	A - Z, a - z, 0 - 9, #, $, %, @, _, blank, nl, period	A - Z, a - z, 0 - 9, #, $, %, @, _, blank, nl, period
Group	1	8	A - Z, 0 - 9, #, $, @	A - Z, 0 - 9, #, $, @

Chapter 3. Setting up a Capacity Provisioning domain

The scope of a z/OS Capacity Provisioning system is referred to as a Capacity Provisioning Domain, or simply a domain.

To set up a domain, follow these steps:

- Plan the domain set up: Define your configuration settings such as the name of the domain, prepare your security and collect information about your system.
- Define the Provisioning Manager: Prepare your z/OS system to start up a Provisioning Manager.
- Setup z/OSMF: The Capacity Provisioning task in z/OSMF provides a browser-based management console for working with the Provisioning Manager on a workstation.

Note: The second and third steps can be performed in either order.

Before customizing Capacity Provisioning, set up the prerequisite components:

- Customize your monitoring product, such as z/OS Resource Measurement Facility (RMF), including the Distributed Data Server (DDS). For more information, see the z/OS RMF User's Guide.
- Customize the z/OS Common Information Model (CIM) element, including the RMF CIM Monitoring providers or equivalent providers from other monitoring products.

When you set up the Provisioning Manager, you must create security definitions and CIM definitions. For more background information about the security definitions, see *z/OS RACF Security Administrators Guide*, and for CIM definitions, see *z/OS Common Information Model User's Guide*.

Planning the domain setup

Plan the domain setup before performing any other actions. It requires that you collect information, specify your goals, and revisit your corporate conventions.

To plan the domain setup consider the following points:

- What naming conventions to use
- Which z/OS system to run the Provisioning Manager on
- Where the prerequisites are located

Choosing names

Capacity Provisioning includes samples that you can use to set up a new domain. These samples contain default values for element names. If you are satisfied with these defaults, do not change them when you set up the domain. If you choose a different naming convention, replace all occurrences of the default names with the names you select during the actual definition steps. Table 16 on page 49 includes space for you to record the names you choose. The following paragraphs describe the element names, their default values, and their meaning.

Each domain is identified by a domain name. If you use multiple domains, each name must be unique. For more information about the rules for a valid domain name, see “Naming conventions” on page 45.

The Provisioning Manager runs as a started task by using an entry setup for this purpose in the procedure data set for started tasks used in your installation (normally `SYS1.PROCLIB`). A sample started task procedure, CPOSERV, is delivered in `SYS1.SAMPLIB`. Copy this procedure or an equivalent to your started task procedure data set. If you choose a name other than CPOSERV for the started task, you must name the member in this data set accordingly.

When the Provisioning Manager starts, a user is assigned to the started task. This user must be defined and authorized for all resources accessed by the Provisioning Manager. The Provisioning Manager user requires a unique UNIX identifier (UID). You also must define a default group for the user and an OMVS segment that specifies a home directory.

The Provisioning Manager connects to the observed systems to retrieve performance data. For this it uses a distinct user ID, the so called System Observation user. This user ID has to be defined on the runtime system as well as on the observed systems. Because the System Observation user is authenticated with PassTickets, it requires a unique UNIX identifier (UID), it cannot be defined as a protected user, and it must be assigned a password.

The default user ID for the System Observation user is CPOCIM. To change the default you have to update the configuration key `SystemObservation.UserName` in the Provisioning Managers PARM member. If you do not specify a user ID for this key, the Provisioning Manager started task user is used for system observation.

The Provisioning Manager contains status and configuration data in data sets. The names of these data sets use the same high-level qualifier, to make security definitions easier.

Provisioning policies and domain configurations are defined by using the CPMC. If a connection is set up between the CPMC and the Provisioning Manager, you can install and activate policies and domain configurations directly from the CPMC. In this case, the user of the CPMC must connect to the CIM server on the system where the Provisioning Manager runs. Different security definitions are needed according to the operations required:

- If a CPMC user only needs to query the Provisioning Manager, then the user must be a member of the Provisioning Manager query security group.
- If the CPMC user is allowed to modify the run time behavior of the Provisioning Manager, then the user must also be a member of the Provisioning Manager control security group.

The Provisioning Manager and the CPMC communicate by using the CIM protocol. The CIM server user is required to set up security definitions for the communication, see “Defining security for the Capacity Provisioning Management Console user” on page 62.

Table 16 on page 49 contains a list of names to be chosen and their default values. If you do not use the defaults, make a note of the names you use.

Table 16. Name information for a new domain

Name	Default	Your value
Domain name	DOMAIN1	
Started task procedure name	CPOSERV	
High-level qualifier for runtime data set	CPO	
Provisioning Manager user	CPOSRV	
System Observation user	CPOCIM	
Capacity Provisioning Management Console user[1]	ZMFUSR	
Provisioning Manager query security group	CPOQUERY	
Provisioning Manager control security group	CPOCTRL	
Default group for CIM user ID	CFZUSRGP	

Note:

1. The CPMC user can be any z/OS user that is authorized to interact between the CPMC and the Provisioning Manager. The CPMC user must be set up with a password (to authenticate through the CPMC).

Defining the runtime environment

Once you define your names, define the environment for the Provisioning Manager. To start with, identify the primary z/OS system on which you run the Provisioning Manager, and select alternative systems on which a backup of the Provisioning Manager can run if the primary system becomes unavailable. All of these systems need access to the Provisioning Manager data sets.

To allow a restart on an alternative system, the runtime information shown in Table 17 on page 50 and all paths listed in Table 18 on page 51 must be identical on the two systems. The term *runtime systems* refers to all systems on which the Provisioning Manager can run, but note that only one instance of the Provisioning Manager can run at any one time.

Note: All runtime systems must be on the same z/OS level. If the Provisioning Manager is able to restart on a system that is running with a previous z/OS release, configuration or status data might not be readable for the Provisioning Manager and the program might stop processing.

You can use z/OS Automatic Restart Manager (ARM) to restart the Provisioning Manager in the event of failure. To use ARM, define the ARM restart policy, element name, and element type. You can also use any other automation product that can restart the Provisioning Manager when needed.

The Provisioning Manager uses z/OS BCPii protocol to communicate with the hardware. BCPii is a method of communicating with the CPC support elements that does not require a network connection from the runtime system to the HMC. Instead, you must create security definitions on your runtime systems.

For these security definitions, you need the name of the community that BCPii uses to access the hardware console. Note that BCPii requires an uppercase community name. This community name must be authorized to issue read and write operations and to issue commands to change the temporary capacity, Defined Capacity or Group Capacity. You must set up the community name for each CPC that is managed by Capacity Provisioning and all CPCs on which the Provisioning Manager can run.

The access requirements for z/OS BCPii differ depending on whether you want to manage physical capacity (On/Off CoD), defined capacity, or both. Have in mind what you plan to manage when defining the security for hardware access.

The CPMC connects to the Provisioning Manager via the CIM server that runs on the same system as the Provisioning Manager. The connectivity between the CIM server and the Provisioning Manager domain(s) is configured in the CIM provider properties files for the Capacity Provisioning CIM provider. The default name and location for this file is `/etc/cpoprovider.properties`. For more information about the configuration of Capacity Provisioning CIM provider, see "Preparing the connection to the CIM server" on page 64.

To create service information, the Provisioning Manager can write trace and log data. Because the data is temporary, it is usually written to the `/tmp` directory on the runtime system. If you want this data to be written to another location, you can redirect it to another file system. The selected path must be available on the runtime system, and the Provisioning Manager user must be authorized to write to this location. Service data can be deleted after it is sent to IBM. Do not delete the data while the Provisioning Manager is still running.

Use the CPMC to define the policies and domain configurations for the Provisioning Manager. A web browser is required to run the CPMC program.

Table 17 contains a list of all necessary environment information, and the default values where applicable. Record any values you change for your domain in the table.

Table 17. Provisioning Manager runtime environment

Name	Default	Your value
Primary runtime system	n/a	
Alternative runtime systems	n/a	
ARM restart needed	no	
ARM element type	SYSCPM	
ARM element name	SYSCPO	
SNA (System Network Architecture) names of managed CPC	n/a	
Community name	n/a	
CIM provider configuration file	`/etc/cpoprovider.properties`	
Log data location	`/tmp`	
Trace data location	`/tmp`	

Table 17. Provisioning Manager runtime environment (continued)

Name	Default	Your value
Capacity Provisioning Management Console runtime system	n/a	

Collecting information about prerequisites

To set up the Provisioning Manager you need information about other components and products on your runtime system. The information required is:

- The installation directory for the Java™ product
- The installation directory for the CIM server
- The location of the CIM Client for Java
- The location of the System Authorization Facility (SAF) libraries
- The location of the SAF Java library for secured sign-on function

Table 18 contains a list of prerequisite components and products and their default locations. Record the locations that are defined on your runtime systems here.

Table 18. Prerequisites information

Product or Component	Default location	Your location
IBM 31-bit SDK for z/OS, Java Technology Edition, V7.1 (5655-W43)	`/usr/lpp/java/J7.1`	
CIM server	`/usr/lpp/wbem`	
CIM Client for Java Version 2	`/usr/lpp/wbem/jclient/sblim-cim-client2.jar`	
SAF library	`/usr/lib`	
SAF JAR file	`/usr/include/java_classes/IRRRacf.jar`	

Preparing the Provisioning Manager

The prerequisites must be satisfied before you can successfully start the Provisioning Manager. These prerequisites include the runtime system and the systems that are observed by the Provisioning Manager. The runtime system can also be one of the observed systems. On the runtime system, you must:

- Define data sets used for the runtime data.
- Set the configuration parameters to your chosen values.
- Create a started task procedure.
- Provide APF authorization.
- Define the security.
- Define a restart policy.

If you defined alternative runtime systems that share the same resources, for example a common RACF database, you create the definitions only once.

On the observed systems, security must allow the System Observation user to connect to the system and to query the information for management purposes.

Note: The following definitions and examples use the default values for all the data that you collected in the planning step. If you chose different values, adjust the examples accordingly. For more information about the different values and their defaults, see "Planning the domain setup" on page 47.

Defining the runtime data sets

The Provisioning Manager stores permanent and temporary data in data sets. You define these data sets only once for each domain. The data sets must be accessible on all runtime systems. Table 19 contains a list of the required data sets and their attributes.

Table 19. Provisioning Manager data sets

	Restart data set	Policy repository	Domain configuration repository	Provisioning Manager parameters
Data set	*prefix*.RESTART	*prefix*.POLICIES	*prefix*.DOMCFG	*prefix*.PARM
DSORG	PO	PO	PO	PO
DSNTYPE	LIBRARY (preferred) or PDS	LIBRARY (preferred) or PDS	LIBRARY (preferred) or PDS	LIBRARY (preferred) or PDS
RECFM	VB	VB	VB	VB
LRECL	16384	512	512	512
BLKSIZE	32760	32760	32760	32760
Directory blocks (if DSNTYPE=PDS)	20	20	20	20
Primary allocation	1 MB	1 MB	1 MB	300 KB
Secondary allocation	2 MB	2 MB	2 MB	600 KB

The *prefix* for the data set names is the high-level qualifier and the name of the domain defined in Table 16 on page 49. For example, with the default values, the restart data set name would be CPO.DOMAIN1.RESTART.

After you create the data sets, copy two sample files from the Capacity Provisioning installation file system to the data sets for Provisioning Manager parameters. The files are env and parm from directory /usr/lpp/cpo/samples. Copy them as members ENV and PARM.

Capacity Provisioning provides a sample job to define these data sets and copy the files. The sample job is available as member CPOMKDSN in library SYS1.SAMPLIB. Note that this job deletes any existing data sets that have the same name as the ones to be defined.

Adapting the Provisioning Manager parameters

Some parameters of the Provisioning Manager might need to be adapted to your environment. These parameters are part of the Provisioning Manager parameters data set, *prefix*.PARM, in the members ENV for the Provisioning Manager runtime environment data and PARM for the Provisioning Manager configuration information.

The ENV member contains information about the runtime processing environment for your Provisioning Manager. Modify the following paths to match your installation settings:

LIBPATH

This entry must contain:

- The path /usr/lib for SAF libraries
- The Java installation paths /usr/lpp/java/J7.1/bin and usr/lpp/java/J7.1/bin/classic
- The Capacity Provisioning installation path /usr/lpp/cpo/lib

CLASSPATH

This entry must contain the Capacity Provisioning JAR file cpom.jar from the installation directory /usr/lpp/cpo/classes and the SAF JAR file /usr/include/java_classes/IRRRacf.jar.

Optionally, if your Capacity Provisioning installation directory or your CIM installation directory are not at the default locations, you might need to add the CIM Client for Java, /usr/lpp/wbem/jclient/sblim-cim-client2.jar.

Add the CIM Client for Java Version 2 before adding the CIM Client for Java Version 1 if your Provisioning Manager is defined to be restarted on a system that has a lower release.

For example:

```
LIBPATH=/usr/lib:/usr/lpp/java/J7.1/bin:usr/lpp/java/J7.1/bin/classic:...
        /usr/lpp/cpo/lib
CLASSPATH=/usr/lpp/cpo/classes/cpom.jar:...
          /usr/include/java_classes/IRRRacf.jar
```

Note: The information for each path must be on one line, and there must be no blanks between the path values.

The PARM member contains configuration information for the Provisioning Manager. It has the structure of a Java property-file with keyword-value pairs. Both keywords and values are case-sensitive. Comment lines must start with a hash character (#). In the PARM member, you can enter keywords for:

Automatic Restart Manager setting

To use ARM to monitor availability of the Provisioning Manager, you must set the value of **ARM.Register** to Yes. You can do this simply by removing the comment symbol from this statement in the sample member provided. If this key is not specified, or is given any value other than Yes, the Provisioning Manager is not registered with ARM. The value is not case-sensitive.

If you use ARM to monitor the Provisioning Manager, you must define ARM policy. This policy specifies an ARM element type and an ARM element name. If you chose the default element type and name, **SYSCPM** and **SYSCPO**, no changes are needed. If you changed these values, you must replace the values of the keys **ARM.ElementType** and **ARM.ElementName** with the ones you have chosen. The defaults are:

```
# ARM settings
ARM.Register    = No
ARM.ElementType = SYSCPM
ARM.ElementName = SYSCPO
```

Security groups for Capacity Provisioning Management Console commands authorization

To allow the CPMC user to communicate with the Provisioning Manager, you must define the Provisioning Manager query security group and the Provisioning Manager control security group in the configuration keys **CIM.ReadGroup** and **CIM.ModifyGroup**. In this case, the defaults are:

```
# Command authorization definitions
CIM.ReadGroup=CPOQUERY
CIM.ModifyGroup=CPOCTRL
```

System Observation user

The Provisioning Manager can use a separate user ID for logon to the observed systems, the so called System Observation user. This user ID has to be defined on the runtime system as well as on the observed systems. To tell the Provisioning Manager which user ID it should use for system observation you have to specify the user ID in the configuration key **SystemObservation.UserName**. The default is:

```
SystemObservation.UserName=CPOCIM
```

If you do not specify a user ID for this key, the Provisioning Manager user is used for system observation.

Trace and Log data

If you chose directories other than the defaults for the trace and log data, you must set configuration keys **Trace.Path** and **Log.Path** accordingly. The directories must exist and the Provisioning Manager user must have write access to them. The default entries are:

```
# Service data location
Trace.Path = /tmp
Log.Path = /tmp
```

Additional parameters to control provisioning management

In addition to the configuration parameters described above, the PARM member can also contain optional directives that influence the operation of the Provisioning Manager. When these values are specified, they override the default values of the Provisioning Manager. Specify a value if you need to override the default.

Table 20. Additional control parameters

Key	Default value	Description
Planner.BlockingTime	15	Number of minutes the Provisioning Manager waits after detection of a CPC capacity change before any new capacity change action is initiated. This time allows the workload to be redistributed after a capacity change. If faster provisioning or deprovisioning is warranted, the value can be lowered. The recommended range is a value of 5 - 15 minutes.
Planner.ProvisioningRejectTime	120	Number of minutes a CPC is not considered for Provisioning Manager actions after a provisioning request proposed by the Provisioning Manager in confirmation mode has been rejected.
Planner.DeprovisioningRejectTime	120	Number of minutes a CPC is not considered for Provisioning Manager actions after a deprovisioning request proposed by the Provisioning Manager has been rejected in confirmation mode.

Table 20. Additional control parameters (continued)

Key	Default value	Description
Planner.MinimumActivationTime	240	Number of minutes temporary capacity must remain active before it is considered for deactivation. This time interval starts again with any new activation and applies to all active resources. When specified, the minimum activation time must be longer than the blocking time.
Analyzer.Threshold.TotalSharedPhysicalUtilCp	95	Lower limit of physical utilization on shared CPs that must be exceeded before workload condition triggered additional general purpose capacity is considered. In some environments, severe processor contention occurs at lower levels of processor utilization. In such environments you can specify a lower percentage value, such as 90.
Analyzer.Threshold.TotalSharedPhysicalUtilZaap	95	Lower limit of physical utilization on shared zAAPs that must be exceeded before workload condition triggered additional zAAP capacity is considered. In some environments, severe processor contention occurs at lower levels of processor utilization. In such environments you can specify a lower percentage value, such as 90.
Analyzer.Threshold.TotalSharedPhysicalUtilZiip	95	Lower limit of physical utilization on shared zIIPs that must be exceeded before workload condition triggered additional zIIP capacity is considered. In some environments, severe processor contention occurs at lower levels of processor utilization. In such environments you can specify a lower percentage value, such as 90.
Analyzer.Threshold.MvsUtilCp	95	Lower limit of MVS utilization on CPs that must be exceeded before workload condition triggered additional general purpose capacity is considered. In some environments, severe processor contention occurs at lower levels of MVS utilization. In such environments you can specify a lower percentage value, such as 90.
Analyzer.Threshold.MvsUtilZaap	95	Lower limit of MVS utilization on zAAPs that must be exceeded before workload condition triggered additional zAAP capacity is considered. In some environments, severe processor contention occurs at lower levels of MVS utilization. In such environments you can specify a lower percentage value, such as 90.
Analyzer.Threshold.MvsUtilZiip	95	Lower limit of MVS utilization on zIIPs that must be exceeded before workload condition triggered additional zIIP capacity is considered. In some environments, severe processor contention occurs at lower levels of MVS utilization. In such environments you can specify a lower percentage value, such as 90.
Analyzer.CycleTime	200	The Capacity Provisioning management cycle time in seconds. This time interval is used to analyze capacity metrics as to whether the capacity requirements have changed. It influences how quickly the Provisioning Manager detects changed situations. With some policies, for instance those with Utilization Conditions, it is advisable to specify a lower time, such as 60 or a value equivalent to the gathering interval length of your monitoring product.

Table 20. Additional control parameters (continued)

Key	Default value	Description
SystemObservation.MovingAveragePiWeight	100	Percentage from 1 to 100 that expresses the weight applied to the calculation of the moving average PI. The weight applies both to calculation of system and sysplex PI calculation. Smaller values increase smoothing by giving more weight to older PI data, while larger values discount older data in benefit of the latest PI data and produce a less smoothened moving average PI. Values below 100 enable the calculation of moving average PI for all observed workloads. The average is also the basis for the management process. The default value of 100 leaves moving average management disabled.
SystemObservation.MovingAveragePiCapping	5.5	Maximum PI limit that is considered for calculating the moving average PI. Prevents extremely high PI values from distorting computed moving average PI in the long term. Zero indicates that capping is disabled. Other allowed values are 1.3 - 55.
SystemObservation.PollingSpreadLimit	0	The default value of zero optimally times system observation requests. The value can be changed if it is important to distribute monitoring work more evenly over the observation cycle. If multiple systems are observed, the value helps limit the overflow of work eligible for specialty processor to general purpose processors. Allowed values are 0 - 100. The default is zero (polling spread disabled).
DefinedCapacity.BlockingTime	15	Number of minutes the Provisioning Manager waits after detection of a Defined Capacity or Group Capacity change for the system or capacity group before any new Defined Capacity or Group Capacity change action is initiated. This time allows the workload to be redistributed after a capacity change. The recommended range is a value of 5 - 15 minutes.
DefinedCapacity.ProvisioningRejectTime	120	Number of minutes a system or capacity group is not considered for Provisioning Manager defined capacity actions after a provisioning request proposed by the Provisioning Manager has been rejected in confirmation mode.
DefinedCapacity.DeprovisioningRejectTime	120	Number of minutes a system or capacity group is not considered for Provisioning Manager defined capacity actions after a deprovisioning request proposed by the Provisioning Manager has been rejected in confirmation mode.
DefinedCapacity.LeadTime	5	Lead time in minutes before Defined Capacity or Group Capacity is exhausted. The projected *time until capping* reported by the monitoring product must fall below this value before the Provisioning Manager will increase Defined Capacity or Group Capacity on behalf of suffering workload.
DefinedCapacity.MinimumActivationTime	15	Number of minutes a Defined Capacity or Group Capacity increase must remain active before it is considered for decrease by the Provisioning Manager. This time interval starts again with any new Defined Capacity or Group Capacity change. When specified, the minimum activation time must be longer than the blocking time.
DefinedCapacity.GlobalMinimumDefinedCapacity	5	Minimum allowed Defined Capacity for the Provisioning Manager. The Provisioning Manager will not decrease any Defined Capacity below this value.

Table 20. Additional control parameters (continued)

Key	Default value	Description
DefinedCapacity.GlobalMinimumGroupCapacity	5	Minimum allowed Group Capacity for the Provisioning Manager. The Provisioning Manager will not decrease any Group Capacity below this value.
DefinedCapacity.BaseToleration	15	Percentage from 0 to 100 that expresses which amount of Defined Capacity or Group Capacity decrease are considered to be significant. So that an adjustment of the Management Base and the Managed Capacity may be advisable.

Creating started task procedures

You can create started task procedures by copying member CPOSERV from data set SYS1.SAMPLIB to the started task procedure data set. The data set is normally SYS1.PROCLIB. If you have not chosen to use the default name for the started task, CPOSERV, you must rename the member appropriately. In the header section of the procedure, change the following values to the ones you chose:

HLQ High-level qualifier of the runtime data sets

DOMAIN
Name of the domain

CPODIR
Home directory of the Provisioning Manager user

OUTCLS
Suitable output class

RGNSIZE
Region size allowed by the server

The following example shows the supplied sample procedure supplied:

```
//*****************************************************
//* Licensed Materials - Property of IBM
//* 5694-A01
//* Copyright IBM Corp. 2007, 2010
//* Status = HPV7770
//* Change Activity:
//* $P1=OA31072 HPV7770 091118 3272HS: Increase default heap size
//*********************************************
//CPOSERV PROC PMODE='*',
// POLICY='*'
//*****************************************************
//* This section of variables may require customization.
//* Changing RGNSIZE may also require a change to the maximum Java heap
//* size (-Xmx) in the member allocated to STDENV below.
//HLQ SET HLQ=CPO HLQ of runtime datasets
//DOMAIN SET DOMAIN=DOMAIN1 provisioning domain name
//CPODIR SET CPODIR='/u/cposrv' home directory of cposrv
//OUTCLS SET OUTCLS=A output class
//RGNSIZE SET RGNSIZE=512M Server region.See note above. @P1C
//*
```

Providing APF authorization

First, ensure that data set SYS1.SIEALNKE is in the link list. The system automatically places this data set at the beginning of the link list, unless it is

overridden by a SYSLIB statement in PROG*xx*. The default IEASYS*xx* value LNKAUTH=LNKLST must also be in effect, or SYS1.SIEALNKE must be APF authorized.

The next steps in this process are optional. They are only required if you copied the Capacity Provisioning files or the Java files on your runtime system.

On any runtime system, the Provisioning Manager must run with APF authorization. In this case the code must be authorized; the main program is located in a program library data set (PDSE) and the libraries are located in the file system.

If you have copied the Capacity Provisioning library files from the UNIX file system, make sure that all Provisioning Manager libraries and the Java libraries are sufficiently authorized. The Provisioning Manager libraries are located in /usr/lpp/cpo/lib. You can check the authorization by entering the following command at a UNIX shell prompt:

```
extattr /usr/lpp/cpo/lib/*
```

The following libraries require APF authorized = YES:

- libcpoarm.so
- libcpoconsole.so
- libcposocket.so
- libcpostream.so
- libcpoii.so

The Java library is in /usr/lpp/java/J7.1/bin/classic and is named libjvm.so. If you have copied the Java SDK code, you must ensure that its APF authorization is the same as the one of the Provisioning Manager libraries. You can define APF authorization by entering command:

```
extattr +a <filespec>
```

Securing the runtime system

Set up security on the runtime system and on the observed systems. Because an observed system can also be a runtime system, you might have to perform both definitions on these systems. On the runtime system the steps to take are:

- Define the started task.
- Define ARM access.
- Define access for the Provisioning Manager user.
- Define the System Observation user.
- Define the secured sign-on function.
- Define access for the Capacity Provisioning Management Console user.
- Define access to the hardware.

The following examples assume that your external security manager is z/OS Security Server (RACF). If you have installed a different external security manager, modify the examples accordingly. These examples also assume that the Provisioning Manager user and the CPMC user are already defined to the security manager, and that an OMVS segment is defined for both users.

All RACF security definitions that are required for a runtime system that is also an observed system are contained in member CPOSEC1 in SYS1.SAMPLIB. You can copy and change this sample job to match your needs. All security definitions

required for additional observed systems that are not sharing the same security database are listed in sample member CPOSEC2.

The RACF security definitions listed in this document and contained in the members CPOSEC1 and CPOSEC2 assume a prior CIM setup. Detailed information about CIM setup can be found in *z/OS Common Information Model User's Guide*.

The samples CPOSEC1 and CPOSEC2 are adapted for the CIM security setup job CFZSEC.

Defining the started task

The Provisioning Manager started task on the runtime system must be assigned to the Provisioning Manager user. Here is an example definition for RACF:

```
RDEFINE STARTED CPOSERV.* STDATA(USER(CPOSRV))
SETROPTS RACLIST(STARTED) REFRESH
```

Defining ARM access

If Automatic Restart Manager (ARM) is used, a FACILITY class profile must be defined and the Provisioning Manager user must have an UPDATE access. If you change the default ARM element type, ARM element name or Provisioning Manager user you must replace the values SYSCPM, SYSCPO, or CPOSRV in the following example with the values you chose. An example definition is:

```
RDEFINE FACILITY IXCARM.SYSCPM.SYSCPO UACC(NONE)
PERMIT IXCARM.SYSCPM.SYSCPO CLASS(FACILITY) ID(CPOSRV) ACC(UPDATE)
SETROPTS RACLIST(FACILITY) REFRESH
```

Defining security for the Provisioning Manager user

The Provisioning Manager user requires access to local resources on your runtime system. The local resources include:

- Membership in the Provisioning Manager security groups CPOQUERY and CPOCTRL
- UPDATE access to the `CPO.DOMAIN1.*` Provisioning Manager data sets
- READ access to the `BPX.CONSOLE` profile in the FACILITY class
- CONTROL access to the `CPOSRV.*` Provisioning Manager user data sets

If the FACILITY class does not exist, you can create it using the `RDEF FACILITY BPX.CONSOLE UACC(NONE)` command.

You can define these access rights as follows:

```
ADDGROUP CPOQUERY OMVS(GID(...))
ADDGROUP CPOCTRL OMVS(GID(...))

CONNECT (CPOSRV) GROUP(CPOQUERY) AUTH(USE)
CONNECT (CPOSRV) GROUP(CPOCTRL) AUTH(USE)

ADDSD ('CPO.DOMAIN1.*') GENERIC UACC(NONE)
PERMIT 'CPO.DOMAIN1.*'  GENERIC ID(CPOSRV) ACCESS(UPDATE)
ADDSD ('CPOSRV.**')     GENERIC UACC(NONE)
PERMIT 'CPOSRV.**'      GENERIC ID(CPOSRV) ACCESS(CONTROL)
SETROPTS GENERIC(DATASET) REFRESH
```

Optionally, grant the Provisioning Manager access to `BPX.CONSOLE` in the FACILITY class. If this access is not granted, all messages issued by the Provisioning Manager are prefixed with a BPXM023I message.

```
RDEFINE FACILITY BPX.CONSOLE UACC(NONE)

PERMIT 'BPX.CONSOLE' CLASS(FACILITY) ID(CPOSRV) ACCESS(READ)
SETROPTS RACLIST(FACILITY) REFRESH
```

Note that you need to complete the OMVS information for the Provisioning Manager security groups before entering these definitions.

In addition, the Provisioning Manager user needs access to files and directories in the local file system of the runtime system. These access rights are usually sufficient by default. The following access is needed for the Provisioning Manager user:

- Read and execute access to the `/usr/lpp/cpo` Capacity Provisioning installation directory and all its subdirectories
- Read and execute access to the `/usr/lpp/java` Java installation directory and all its subdirectories
- Read access to the `/usr/lpp/wbem/jclient/sblim-cim-client2.jar` CIM Client for Java
- Read access to the `/usr/include/java_classes/IRRRacf.jar` SAF library
- Read and write access to trace and log data in the `/tmp` file system path
- Read and write access to the `/var` file system path
- Read and write access to the `/u/cposrv` home directory of the Provisioning Manager user

If the current access rights are insufficient, set the "other" read, write, and execute access permissions of the directories and files accordingly by using the UNIX **chmod** command.For example:

```
chmod -R o+rx /usr/lpp/cpo
```

These permissions allow all users to read and execute files and directories starting with `/usr/lpp/cpo`.

Note: This command must be issued by a user with appropriate access rights.

Defining the System Observation user

The Provisioning Manager can use a configured user ID to log on to the observed systems. The default for this user ID is `CPOCIM` and must be configured in the PARM member of the parameter data set *prefix*`.PARM` through key **SystemObservation.UserName.** This user requires an OMVS segment and unique non-zero UID. For example:

```
ADDUSER  (CPOCIM)
 ALTUSER  (CPOCIM) +
          NORESTRICTED +
          DFLTGRP(........) +
          OMVS(HOME('/u/cpocim') UID(......) ) +
          PASSWORD(......)
```

See also “Defining access to the CIM server” on page 64.

Defining the secured sign-on function on the runtime system

Communication between the runtime system and the observed systems requires authentication and authorization. These processes are implemented by using the secured sign-on function, which generates PassTickets when the Provisioning Manager logs on to an observed system. For the logon to the observed system either a separate System Observation user ID (default CPOCIM) is defined on the

runtime system and all observed systems, or the Provisioning Manager user (default CPOSRV) must be authorized appropriately on the runtime system and all observed systems.

The following definitions are needed to use the secured sign-on function for this user ID and for the Provisioning Manager to be able to generate PassTickets:

To activate the PTKTDATA class

Use the definitions:

```
SETROPTS CLASSACT(PTKTDATA)
SETROPTS RACLIST(PTKTDATA)
```

To use PassTickets

If it is not already implemented as part of the CIM server setup, define the profile CFZAPPL in the PTKTDATA class. The Provisioning Manager user must have at least READ access to this profile. If a common cryptographic architecture (CCA) product is installed on the systems with the secured sign-on function you can encrypt the secured sign-on application keys. If not, mask the secured sign-on application key by using the SSIGNON operand and a 64-bit KEYMASKED value. For example:

```
RDEFINE PTKTDATA CFZAPPL SSIGNON(KEYMASKED(XXXXXXXXXXXXXXXX))+
       APPLDATA('NO REPLAY PROTECTION')
SETROPTS RACLIST(PTKTDATA) REFRESH
```

To generate PassTickets

To enable the Provisioning Manager user to generate PassTickets for the System Observation user ID, create the security profile `IRRPTAUTH.CFZAPPL.CPOCIM` in the PTKTDATA class giving the Provisioning Manager user at least UPDATE access authority. For example:

```
RDEFINE PTKTDATA IRRPTAUTH.CFZAPPL.CPOCIM
PERMIT IRRPTAUTH.CFZAPPL.CPOCIM CLASS(PTKTDATA) ID(CPOSRV) ACCESS(UPDATE)
SETROPTS RACLIST(PTKTDATA) REFRESH
```

For more information about configuring RACF to use PassTicket services, see *z/OS Security Server RACF Security Administrators Guide*.

Defining security for hardware access

If you are using BCPii communication, the Common Event Adapter (CEA) must run in full function mode and the Provisioning Manager user must be authorized for to some Common Event Adapter (CEA) services and for the CPCs that need to be managed. Instructions for setting the CEA into full function mode are provided in *z/OS Planning for Installation*.

For the CEA services, the Provisioning Manager user needs READ authority to the following profiles in the SERVAUTH class:

- CEA.CONNECT
- CEA.SUBSCRIBE.ENF_0068*

If you previously defined access through more generic profiles, such as CEA.*, you might want to use those profiles also to permit the Provisioning Manager user.

A sample definition would look like the following:

```
SETROPTS CLASSACT(SERVAUTH)
RDEFINE SERVAUTH CEA.CONNECT UACC(NONE)
RDEFINE SERVAUTH CEA.SUBSCRIBE.ENF_0068* UACC(NONE)
PERMIT CEA.CONNECT CLASS(SERVAUTH) ID(CPOSRV) ACCESS(READ)
PERMIT CEA.SUBSCRIBE.ENF_0068*  CLASS(SERVAUTH) ID(CPOSRV) ACCESS(READ)
SETROPTS RACLIST(SERVAUTH)  REFRESH
```

The Provisioning Manager user needs the following authorizations to access information about the hardware, perform activation and deactivation requests for temporary capacity, and changing the defined capacity on a CPC:

- READ access to profile HWI.APPLNAME.HWISERV in the FACILITY class.
- CONTROL access to profile HWI.TARGET.*netid*.*name* in the FACILITY class. The net ID and name represent the SNA (System Network Architecture) name of the CPC as defined at the SE. The APPLDATA of the security definition must contain the uppercase community name for the Provisioning Manager as defined in Table 17 on page 50. The profiles must cover all CPCs to be managed by Capacity Provisioning and all CPCs on which the Provisioning Manager can run.
- Optionally, if you plan to manage physical (On/Off CoD) capacity, READ access to profile HWI.CAPREC.*netid*.*nau*.* in the FACILITY class. The net ID and name represent the SNA name of the CPC as defined at the support element. The profiles must cover all capacity records.
- Optionally, if you plan to manage defined capacity, UPDATE access to the profile HWI.TARGET.*netid*.*name*.*imagename* in the FACILITY class. The net ID and name represent the SNA name of the CPC as defined at the support element. The image name is the name of the LPAR that can be managed. The profiles must cover all LPARs that should be managed.

For example, if you have a CPC with SNA Name IBMNET.CPC1, and you plan to manage physical capacity and defined capacity for all LPARs, the definitions would look as follows:

```
RDEFINE FACILITY HWI.APPLNAME.HWISERV UACC(NONE)
RDEFINE FACILITY HWI.TARGET.IBMNET.CPC1 APPLDATA(...) UACC(NONE)
RDEFINE FACILITY HWI.CAPREC.IBMNET.CPC1.* UACC(NONE)
RDEFINE FACILITY HWI.TARGET.IBMNET.CPC1.* UACC(NONE)

PERMIT HWI.APPLNAME.HWISERV CLASS(FACILITY) ID(CPOSRV) ACCESS(READ)
PERMIT HWI.TARGET.IBMNET.CPC1 CLASS(FACILITY) ID(CPOSRV) ACCESS(CONTROL)
PERMIT HWI.CAPREC.IBMNET.CPC1.* CLASS(FACILITY) ID(CPOSRV) ACCESS(READ)
PERMIT HWI.TARGET.IBMNET.CPC1.* CLASS(FACILITY) ID(CPOSRV) ACCESS(UPDATE)

SETROPTS RACLIST(FACILITY)  REFRESH
```

For more information about BCPii setup, see *z/OS MVS Programming: Callable Services for High-Level Languages*.

Defining security for the Capacity Provisioning Management Console user

The CPMC user must be authorized to connect the CPMC to the Provisioning Manager. Define this user on the runtime system with an OMVS segment, and add the user to the appropriate Provisioning Manager security group, depending on which administration and operation commands the user is allowed.

Define the CPMC users with a non-zero z/OS UNIX identifier (UID). When using a UID of zero, the **enableRemotePrivileged UserAccess** configuration property must be enabled in the CIM server. All CPMC users need to have execute permission to the `/var` directory. You can grant all users execute permission via the command:

```
chmod o+x /var
```

At this time, the CIM server must be set up as described in *z/OS Common Information Model User's Guide*.

The following examples assume a default CPMC user of ZMFUSR. This ID has to be replaced with the distinct user ID or IDs for your environment.

The CPMC user must have UPDATE access to the CIMSERV profile in the WBEM class. If your system is set up to use z/OS UNIX level security (for example, if BPX.SERVER is defined), define the CIM server user as a surrogate of the CPMC user. To accomplish this, profile BPX.SRV.ZMFUSR in the SURROGAT class is required if a generic BPX.SRV.** is not defined. The current CIM recommendation for switching identity (SURROGAT) is explained in *z/OS Common Information Model User's Guide* during the CIM setup. The CIM server user requires access to this profile. The next step depends on the authorization to be given to the CPMC user.

If the CPMC user is only authorized to query information from the Provisioning Manager, add this user to the Provisioning Manager query security group CPOQUERY, and grant UPDATE access to the CIMSERV profile by connecting it to the CIM users ID default group CFZUSRGP, which was defined during the CIM setup. The following example assumes that the CPMC user has the user ID ZMFUSR:

```
CONNECT (ZMFUSR) GROUP(CPOQUERY) AUTH(USE)
CONNECT (ZMFUSR) GROUP(CFZUSRGP) AUTH(USE)
```

If instead the CPMC user is authorized to change the processing characteristics of the Provisioning Manager, add this user to both Provisioning Manager security groups, CPOQUERY and CPOCTRL, and grant UPDATE access to the CIMSERV profile by connecting it to the CIM users ID default group CFZUSRGP, which has been defined during the CIM setup:

```
CONNECT (ZMFUSR) GROUP(CPOQUERY) AUTH(USE)
CONNECT (ZMFUSR) GROUP(CPOCTRL) AUTH(USE)
CONNECT (ZMFUSR) GROUP(CFZUSRGP) AUTH(USE)
```

Securing the observed systems

When a system is observed, the Provisioning Manager connects to the CIM server on that system and retrieves configuration and performance information about the workload. For the logon to the observed system, either a separate System Observation user ID (default CPOCIM) can be defined on the runtime system and all observed systems, or the Provisioning Manager user (default CPOSRV) must be authorized appropriately on the runtime system and all observed systems. To enable this communication, establish a connection for the System Observation user that is authorized for PassTickets. The steps to take are:

- Define the System Observation user on the observed systems with the same password as on the runtime system.
- Enable the secured sign-on function.
- Authorize the System Observation user to access the CIM server.

The following definitions assume that the System Observation user is already defined and OMVS segment is assigned. These definitions must be effective on all observed systems.

The RACF security definitions listed here and contained in member CPOSEC2 assume a prior CIM setup. Detailed information about CIM setup can be found in *z/OS Common Information Model User's Guide*. The CPOSEC2 sample is adapted for the CIM security setup job CFZSEC.

Defining the secured sign-on function on the observed systems

This feature must be set up on the observed systems in the same way it was set up on the runtime system, which is described in "Defining the secured sign-on

function on the runtime system" on page 60. Note that the **KEYMASKED** value must match the runtime system, and PassTicket generation is only performed on the runtime systems.

Defining access to the CIM server

The System Observation user must be authorized to access the CIM server on all observed systems. This configuration assumes that the System Observation user is already defined and the CIM setup is complete.

- Give the System Observation user access to the CIMSERV profile in the WBEM class by connecting it to the default CIM users group CFZUSRGP, which was defined in the CIM setup:

```
CONNECT (CPOCIM) GROUP(CFZUSRGP) AUTH(USE)
```

Note: The example above assumes that you have defined a separate System Observation user (default CPOCIM). If not, you must authorize the Provisioning Manager user (default CPOSRV).

For more information, see *z/OS Common Information Model User's Guide*.

Setting up Automatic Restart Manager

This step is only necessary if you use Automatic Restart Manager (ARM) to restart the Provisioning Manager.

The Provisioning Manager requires the following conditions to use the ARM capability:

- The element name is `SYSCPO`, unless you chose another value in Table 17 on page 50.
- The element type is `SYSCPM`, unless you chose another value in Table 17 on page 50.
- The Provisioning Manager normally restarts with the policy and processing mode used the last time it ran, together with any modifications to the policy that are triggered by console commands. To achieve such a restart, the policy name and the processing mode on the restart command must be specified as '`*`'.
- The Provisioning Manager can be restarted on another system of the sysplex if that system has access to the runtime data sets and the required file systems used by the previous system.

An example setup is supplied in member CPOARMPO of `SYS1.SAMPLIB`. For more information about ARM, see *z/OS MVS Setting Up a Sysplex*.

Preparing the connection to the CIM server

The Provisioning Manager and the CPMC communicate with a CIM server. The CPMC uses a CIM connection to communicate with the Provisioning Manager. To obtain information about an observed system, the Provisioning Manager retrieves performance information via the CIM server on that system. To prepare the communication, you can establish connections to the server by using either the HTTP protocol or the HTTPS protocol, if the CIM server is configured to support the chosen protocol. For more information about configuring the CIM server, see the *z/OS Common Information Model User's Guide*. For the HTTPS protocol, you use an AT-TLS configuration. The Provisioning Manager and the CPMC do not support authentication based on SSL certificates.

Note: Ensure that the configured port of the CIM server matches the definitions that you make in your domain configuration and in the connections defined using the CPMC.

Required settings for the RMF Distributed Data Server

If you are using RMF to provide performance information on the observed system, then you should configure the GPMSRV*xx* member that is used by the RMF Distributed Data Server. This member must specify a higher value for **MAXSESSIONS_HTTP** than the default of 20. The recommended value is 60 or higher.

```
MAXSESSIONS_HTTP(60) /* MaxNo of concurrent HTTP requests */
```

Preparing the connection to the Provisioning Manager

Logon to a z/OS UNIX session as a CIM administrator user.

1. Copy the Capacity Provisioning CIM provider properties file to the /etc directory:

   ```
   cp /usr/lpp/cpo/provider/cpoprovider.properties /etc
   ```

   If you want to store the Capacity Provisioning CIM provider properties file (see also Table 17 on page 50) in a different location or use a different name, see "Customizing the location of the Capacity Provisioning CIM provider properties file."
2. If your domain name is not the default (DOMAIN1), edit the file you copied, for example by using `oedit /etc/cpoprovider.properties` to change the `DomainNames = DOMAIN1` line to specify the name of your domain.
3. Ensure that the file is readable:

   ```
   chmod a+r /etc/cpoprovider.properties
   ```
4. Verify that the program-controlled flag is set in the extended file attributes for the Capacity Provisioning CIM provider library:

   ```
   ls -E /usr/lpp/cpo/lib/libcpoprovider.so
   ```

   If the attribute is not set, use the following command to manually set the program-controlled flag:

   ```
   extattr +p /usr/lpp/cpo/lib/libcpoprovider.so
   ```
5. Verify that a link to the Capacity Provisioning CIM provider library is created in the CIM server provider directory:

   ```
   ls -l /usr/lpp/wbem/provider/libcpoprovider.so
   ```

   If the link does not exist, use the following command to manually create the link:

   ```
   ln -s /usr/lpp/cpo/lib/libcpoprovider.so
         /usr/lpp/wbem/provider/libcpoprovider.so
   ```
6. Restart the CIM server.

Customizing the location of the Capacity Provisioning CIM provider properties file

By default the Capacity Provisioning CIM provider properties file has to be copied to the /etc directory and named `cpoprovider.properties`. If you want to store the properties file in a different location or use a different name, you have to tell the CIM server about the new location by adding the environment variable **CPO_CIM_CONFIG_FILE** to the CIM server environment file. For example:

```
CPO_CIM_CONFIG_FILE=/etc/cpoprovider.SYSA.properties
```

The default name for the CIM Servers environment file is `/etc/wbem/cimserver.env` but this may differ, depending on the value of the STDENV DD statement in the CIM server started task procedure CFZCIM. Restart the CIM server to activate the new settings in the CIM server environment file.

Defining the connection to the hardware

For the BCPii communication, you must define the community name that you chose in your configuration of the Provisioning Manager; for details, see Table 17 on page 50. Note that the address and the network mask of the community name should specify the loopback device `127.0.0.1`, `255.255.255.255`, respectively. The community name must be authorized for read and write operations. The next step is to allow external programs remote operation using the API. You must perform the definitions on the SE of each CPC that will be managed by the Provisioning Manager and all CPCs on which the Provisioning Manager can run. For more information about the configuration of an SE for API programs, see *z/OS MVS Programming: Callable Services for High-Level Languages*.

Tuning and workload classification for Capacity Provisioning

Your Workload Management (WLM) service definition for the observed systems ensures that monitors run at a higher priority than the work being monitored. The Provisioning Manager, together with the infrastructure it uses, must be prioritized so that they perform reliably even in a capacity-constrained situation. On the runtime systems, ensure that the Provisioning Manager started task `CPOSERV` is classified appropriately.

Either classify the Provisioning Manager and the supporting subsystems into SYSSTC, or give them an aggressive single period velocity goal at an importance level that is higher than the work that is defined in your provisioning policy. In particular, the following subsystems and address spaces are required:

- On the runtime system:
  - The Provisioning Manager (CPOSERV)
  - The TCP/IP infrastructure
  - The CIM server, if the CPMC is used to control the Provisioning Manager
- On the observed systems:
  - The RMF address spaces
  - The RMF Distributed Data Server (GPMSERVE)
  - The TCP/IP infrastructure
  - The CIM server

Network latency in the CIM protocol can be minimized by specifying numeric IP addresses or by defining entries in the hosts file for the observed systems.

Other contention factors that could affect the ability of the Provisioning Manager to operate efficiently must be minimized. The Provisioning Manager and the CIM servers rely on the z/OS UNIX file system. Severe contention must be relieved by appropriate tuning measures to allow Capacity Provisioning to operate reliably.

Resource consumption considerations

The consumption of resources (processor or virtual storage) by the Provisioning Manager largely depends on the domain and policy definitions. The following factors increase resource consumption by the server:

- Many observed systems
- Many monitored service class periods

- Short gathering intervals of the monitoring product, such as a small RMF MINTIME
- Use of the HTTPS protocol for communication with the CIM server
- Activated traces

Most Provisioning Manager processing is zAAP-eligible. zAAPs are used automatically on systems configured with them and no additional definitions are required.

Part 2. Operating Capacity Provisioning

Describes the use of Provisioning Management.

Chapter 4. Using the Capacity Provisioning Management Console

IBM z/OS Management Facility (z/OSMF) provides a framework for managing various aspects of a z/OS system through a web browser interface. z/OSMF simplifies some areas of system management and reduces the level of expertise that is needed for managing a system.

Capacity Provisioning provides a plug-in for z/OSMF, the Capacity Provisioning Management Console (CPMC), that offers the following functionality:

Manage connections to your Provisioning Manager
Create, modify, and delete connections that you use to connect z/OSMF to your Provisioning Manager.

View reports
Request a domain status, active configuration, or active policy report.

Manage domain configurations
Create, modify, view, import, export, install, and activate domain configurations.

Manage provisioning policies
Create, modify, view, import, export, install, and activate provisioning policies.

Getting started with z/OSMF

After z/OSMF is installed and configured, you can log in with a web browser. You see a navigation area on the left and a working area on the right. Depending on the installed plug-ins, z/OSMF offers a number of traditional system programmer tasks for which a user is authorized. The tasks belong to different categories. The Capacity Provisioning task is in the Performance category. When you click the Capacity Provisioning task, the CPMC opens in the working area.

Working with the Capacity Provisioning Management Console

The CPMC provides a browser-based user interface for working with the Capacity Provisioning Manager on your z/OS system. With the CPMC, you can perform the following functions:

- Manage domain configurations and policies:
  - Manage, create, modify, and delete provisioning policies and domain configurations from a central shared repository.
  - Install and activate policies and domain configurations for the domain the Provisioning Manager controls.
  - Import and export domain configurations and policies.
- Manage connections to a Provisioning Manager and use them to transfer provisioning policies and domain configurations to the Provisioning Manager, or to query various status reports.
- View the status of a Provisioning Manager by displaying reports about the domain status, the active configuration and the active policy.

With the management functions, z/OSMF Capacity Provisioning has been extended to support all the functions that have previously been available in the Microsoft Windows-based Capacity Provisioning Control Center (CPCC).

The Capacity Provisioning Management Console time zone

The CPMC displays times in different contexts:

- In time conditions within a policy
- In status reports retrieved from the Provisioning Manager
- In the tables displaying the policies and the domain configurations in the z/OSMF repository, indicating when they were last modified

You can choose which time zone the CPMC uses to display all times. You have a choice between GMT and local time to be used for the display of date and time data. The local time zone is determined by your browser settings. This setting does not affect how times within time conditions of policies are stored in the z/OSMF repository or the file system; they are always stored in GMT. When you switch the CPMC to a different time zone, all dates and times are adjusted.

Policies contain time conditions that define time periods during which additional capacity can be provisioned. Recurring time conditions define weekly repeating periods. They describe the time of the day when provisioning of additional capacity is allowed and the days of the week to which these times apply.

The CPMC time zone is used to display the dates and times contained in the time conditions. If a time zone with daylight saving time is specified, the actual date and time values used might be one hour later or earlier than the values displayed.

If due to a time zone switch the start time of a recurring time condition moves to a different day, the checks for the selected days of the week are moved accordingly.

For more information

You can find more information about z/OSMF and the Capacity Provisioning task in *z/OS Management Facility Configuration Guide*.

Chapter 5. Controlling the Provisioning Manager

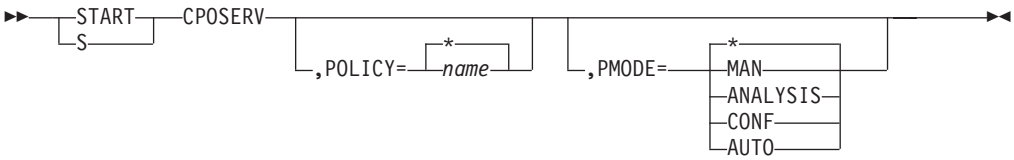
The Provisioning Manager is the server component in a Capacity Provisioning domain that manages the activation and deactivation of capacity on your CPCs. It can be accessed from the MVS console or from any console device.

It also can be accessed from the Capacity Provisioning Management Console.

Starting the Provisioning Manager

The Provisioning Manager runs as a z/OS started task. You use the MVS start command to start the program.

Syntax



Parameters

POLICY=*name*

The name of the policy to be activated when the Provisioning Manager starts. The specified policy must be available in the policy repository that is accessible to the Provisioning Manager. If the Provisioning Manager cannot activate the new policy, it stops.

You can specify an asterisk (*) instead of a policy name. In this case, the last recently active policy for the provisioning domain is used and no policy is retrieved from the policy repository. `POLICY=*` is the default for this parameter.

PMODE=*mode*

The initial processing mode of the Provisioning Manager. The following processing mode values are supported:

MAN
Manual mode, which disables policies

ANALYSIS
Analysis mode

CONF
Confirmation mode

AUTO
Autonomic mode

If the Provisioning Manager cannot activate the processing mode, it stops immediately.

You can specify an asterisk (*) instead of a processing mode. In this case, the last recently active processing mode for the provisioning domain is used. PMODE=* is the default for this parameter.

If you start the Provisioning Manager for the first time without specifying start parameters, it starts with the default policy and the default processing mode. The default policy is indicated by the policy name *none. This policy is an empty policy that contains no rules. The default processing mode is manual (MAN).

When you start the Provisioning Manager for the first time, the default domain configuration is activated. The default domain configuration is indicated by the name *none. This configuration is empty and contains no CPCs and no systems.

For more information about how to activate a new policy, a new domain configuration, or a new processing mode, see "SET DOMAIN" on page 148.

Provisioning Manager commands

When the Provisioning Manager is running, it responds to MVS **MODIFY** and **STOP** commands. You pass commands to the provisioning manager by using the **MODIFY** command. These commands are processed sequentially.

The syntax for issuing Provisioning Manager commands is:

For more information about the Provisioning Manager commands, see Chapter 8, "Provisioning Manager command reference," on page 103.

Stopping the Provisioning Manager

Two commands can be used to end Provisioning Manager processing:

- The Provisioning Manager **stop manager** command. For more information, see "STOP MANAGER" on page 152.
- The MVS **STOP** command. This command is equivalent to the Provisioning Manager **stop manager mode=normal** command. The format of the MVS command is:

After the Provisioning Manager stop command is issued, it does not accept any further commands.

While the Provisioning Manager runs, it writes status data to a restart data set, so that, when it is restarted, it continues to manage capacity from the point where it stopped. This function allows for short breaks, for example, if you need to transfer the Provisioning Manager to another system. If you expect a longer interruption to processing, end the program only when there is no additional capacity provisioned by the Provisioning Manager.

Note: After a restart, the Provisioning Manager needs time to collect all the data required to decide on actions related to workload conditions. The minimum time required for this is the provisioning duration of all the specified and active workload conditions. Until this data is gathered, the Provisioning Manager assumes that no additional capacity is needed for workload conditions.

Capacity that has been activated by the Provisioning Manager is not automatically deactivated when the Provisioning Manager stops. If you have to deactivate additional capacity, set the Provisioning Manager to `MANUAL` processing mode and do it manually before you issue the stop command. To find out which additional capacity is currently active, issue the **`REPORT CONFIGURATION`** command; see "REPORT CONFIGURATION" on page 134. After the Provisioning Manager stops, additional capacity can be deprovisioned only by using the hardware interface at the HMC.

Answering operator messages

In some situations, the Provisioning Manager issues operator messages that you can reply to. These situations can occur:

- When the Provisioning Manager runs in confirmation mode and asks whether the proposed actions are allowed
- When inconsistencies with the hardware are detected

If the Provisioning Manager is running in confirmation mode, it checks the policy and the workload on the systems that are defined in the domain configuration to see whether additional capacity must be provisioned or deprovisioned. The Provisioning Manager proposes a change to the activation level and issues a message that asks whether you want to allow the proposed action. If you allow the action, the Provisioning Manager attempts to apply it to the CPC. If you deny the action, the CPC is not considered again for activation or deactivation for a certain time period. For more information about how to configure this period, see Table 20 on page 54.

Until you respond, the Provisioning Manager continues to check the policy and workload to see whether the proposed action is still necessary. If the action is no longer necessary, the outstanding message is canceled. If another action is proposed instead, a new message is issued. While the Provisioning Manager is waiting for an answer, you can use report commands to help you decide how to respond.

In some situations, there might be inconsistencies between the status of the Provisioning Manager and the CPC, for example, if an activation is triggered and the expected change in the activation level of the On/Off CoD record did not complete. In such situations, the Provisioning Manager issues a request message to the console asking how to resolve this situation. If the situation is resolved by any new information from the hardware, the request message is canceled.

Chapter 6. Working with reports

You can request various reports about the status of the Provisioning Manager. The content varies depending on the type of report. The following reports can be requested:

- Domain report
- Policy report
- Domain configuration report
- Workload report
- Utilization report
- Activity report
- Record report
- Defined capacity report
- Group capacity report
- Trace report
- Log report

These reports can be written to the console or to a file in the file system by using Provisioning Manager commands. If the report is sent to the console, the command response message includes the type of report, date, and time in UTC, for example:

```
CPO1008I Domain report generated at 12/22/2012 08:45:10
```

If the report is written to a file, this information is included in the first line of the file, for example:

```
Domain report generated at 12/22/2007 08:45
```

In both cases, the report data follows this line. Sections described in the **Working with reports** chapter contain a description of the different reports and how to interpret the information.

Note: If the report is written to the console, the amount of information written is limited. If the data exceeds the limit, information is truncated or missing. In this case, direct the report to a file to get complete information.

Managing reports

The Provisioning Manager reports only the status information that is currently in memory, and does not keep a history. If you require a history, you must create it yourself by regularly requesting the reports you are interested in. Specify a unique file name for each report, because the Provisioning Manager overwrites the content of a file if it already exists.

The domain report

To create a domain report, issue a REPORT DOMAIN command. For the syntax of this command, see "REPORT DOMAIN" on page 137. The report contains information about the current setup of the domain managed by the Provisioning Manager. The information is listed in the following order, and includes:

- The name of the provisioning domain and the time when the Provisioning Manager was started
- The active processing mode and the time it was activated. The possible values for the processing mode are:
  - MANUAL
  - ANALYSIS
  - CONFIRMATION
  - AUTONOMIC
- The active domain configuration and the time it was activated. A domain configuration name of `*none` indicates that the default domain configuration is active, because you never activated your own domain configuration. In this case, the reported time is the time when the Provisioning Manager first ran.
- The name of the active provisioning policy and the time it was activated. A policy name of `*none` indicates that the default policy is active, because you never activated your own policy. In this case, the reported time is the time when the Provisioning Manager first ran.
- Service information about the code level of the running program.
- If `type=detailed` was requested, a list of all configuration information that was set in the `PARM` member of the domain. The information is displayed in the form *`key=value`*, where key is one of the defined parameters, and value the actual value currently active for the domain. For a list of supported configuration entities, see "Adapting the Provisioning Manager parameters" on page 52.

All reported times are in UTC.

Note: The activation times for the processing mode, domain configuration, and policy might have occurred in a previous run of the Provisioning Manager and might be earlier than the current start time of the Provisioning Manager.

An example of a domain report is:

```
CPO1008I Domain report generated at 12/22/2012 08:45:10
Provisioning Manager for domain DOMAIN1 started at 12/22/2012 08:40:53
Active processing mode is AUTONOMIC since 12/22/2012 08:40:54
Active domain configuration is DCSAMPLE since 12/22/2012 08:40:54
Active policy is EXAMPLE since 12/22/2012 08:40:54
Code level is 12201
Parm member entries:
 ARM.ElementName = SYSCPO
:
 SystemObservation.UserName = CPOCIM
:
 Topology.Protocol = INTERNAL
:
```

The policy report

To create a policy report, issue a REPORT POLICY command. For the syntax of this command, see "REPORT POLICY" on page 139. The report contains information about the active policy and its status. Indentation is used in the report to group related items together. The reported information indicates:

- The name of the active policy, and its enabled or disabled status. A policy is disabled if the Provisioning Manager is running in MANUAL processing mode. In all other processing modes, the policy is enabled.
- The logical processor scope of the policy. For each limit, the system and sysplex names, the processor limits, and the defined action are displayed. An asterisk (*) marks the limit as defined by the LPAR.

- The maximum processor scope of the policy. Two lines are displayed for each entry. The first line contains the CPC name and the maximum temporary capacity that can be activated by all provisioning rules. The second line denotes the primary and secondary capacity increments that are defined for the CPC.
- The maximum defined capacity scope of the policy. Two lines are displayed for each entry. The first line contains the system name, the sysplex name, and the amount of Defined Capacity that is allowed to be increased by all provisioning rules. The second line denotes the primary and secondary capacity increments that are defined for the system.
- The maximum group capacity scope of the policy. Two lines are displayed for each entry. The first line contains the group name, the CPC name, and the amount of Group Capacity that is allowed to be increased by all provisioning rules. The second line denotes the primary and secondary capacity increments that are defined for the group.
- The provisioning rules that are contained in the policy. The information displayed about each rule includes:
  - The rule name, the current enabled or disabled status, and the default status as defined in the policy when it was activated.
  - The processor scope for the rule. One line is displayed for each entry. It contains the CPC name and the temporary capacity that can be activated by the conditions in this rule.
  - The defined capacity scope for the rule. One line is displayed for each entry. It contains the system name, the sysplex name, and the amount of Defined Capacity that is allowed to be increased by the conditions in this rule.
  - The group capacity scope for the rule. One line is displayed for each entry. It contains the group name, the CPC name, and the amount of Group Capacity that is allowed to be increased by the conditions in this rule.
  - The provisioning conditions. For each provisioning condition, the following information is given:
    - The provisioning condition name, the current enabled or disabled status, and the default status as defined in the policy when it was activated.
    - The workload conditions, if any. For each workload condition, the following information is given:
      - The workload condition name and the name of the sysplex this condition is for.
      - The name of the system to which the workload condition applies.
      - The importance filters that are defined in the workload condition, if any. For each importance filter, the following information is given:
        - The importance (I), the provisioning performance index limit (PL), the provisioning performance index limit duration (PD), the deprovisioning performance index limit (DL), the deprovisioning performance index limit duration (DD), and the scope for the performance index (S).
      -

        The included service classes that are defined in the workload condition, if any. For each included service class, the following information is given:
        - The Workload Management (WLM) service definition name, policy name, service class name, and service class period number to be observed.
        - The provisioning performance index limit (PL), the provisioning performance index limit duration (PD), the deprovisioning

performance index limit (DL), the deprovisioning performance index limit duration (DD), and the scope for the performance index (S).

–

The excluded service classes that are defined in the workload condition, if any. For each excluded service class, the following information is given:

- The Workload Management (WLM) service definition name, policy name, service class name, and service class period number not to be observed.

- The utilization conditions, if any. For each utilization condition, the following information is given.
  - The utilization condition name and the name of the CPC this condition is for. The provisioning utilization limit (PU), the provisioning utilization duration (PD), the deprovisioning utilization limit (DU), the deprovisioning utilization duration (DD) and the processor type (PT) the utilization limits are for.
- The time conditions. All dates and times are reported in UTC. For nonrecurring and recurring time conditions, the information is different.
  - For nonrecurring time conditions, the following information is given:
    – The time condition name and status. The status might include:
      - Pending
      - Observing and enabled
      - Observing and disabled
      - Active and enabled
      - Active and disabled
      - Drained and enabled
      - Drained and disabled
      - Inactive

      For more information about the time condition status, see “Processing policy time conditions” on page 36. Because time conditions cannot be enabled or disabled, the status reflects the enabled or disabled status of all the elements it is contained in. A time condition is enabled if the policy, the rule, and the provisioning condition are all enabled. Otherwise it is disabled.
    – The start, deadline, and end time of the time condition.
  - For recurring time conditions, the following information is given:
    – The recurring time condition name and status. The status might include:
      - Pending
      - Observing and enabled
      - Observing and disabled
      - Active and enabled
      - Active and disabled
      - Drained and enabled
      - Drained and disabled
      - Inactive

      For more information about the time condition status, see “Processing policy time conditions” on page 36. Because time conditions cannot be enabled or disabled, the status reflects the enabled or disabled status of all the elements it is contained in. A time condition is enabled if the policy, the rule, and the provisioning condition are all enabled. Otherwise it is disabled.

– The start date, end date, and the selected days of the week. The days of the week are marked with X if the day of the week is allowed and with a dash (-) if the day of the week is not allowed. The days are reported from Monday to Sunday.
– The start time, deadline time, and end time of the recurring time condition.

An example of a policy report is:

```
CPO1005I Policy report generated at 01/16/2014 16:54:41
Policy EXAMPLE is enabled
Logical processor scope:
  System SYS1 sysplex PLEX1 scope is 4/7/3 Action: local message
       System SYS2 sysplex PLEX1 scope is */*/* Action: remote message
Maximum processor scope:
 Limit for CPC CPC1 is 150 MSU, 3 zAAPs, 3 zIIPs
  activation of 7/5 MSU
Maximum defined capacity scope:
 None
Maximum group capacity scope:
 None
Rule SampleRule is enabled (default enabled)
 Processor scope:
  Limit for CPC CPC1 is 30 MSU, 0 zAAPs, 0 zIIPs
 Defined capacity scope:
  None
 Group capacity scope:
  None
 Provisioning condition SampleCS is enabled (default enabled)
  Workload condition CPUMED for sysplex PLEX1
   System name "SYS1"
   I/PL/PD/DL/DD/S 1 1.5 10 1.2 10 System
   Included: SAMPLESD/SAMPLEP/SAMPSC.1
    PL/PD/DL/DD/S 1.4 5 1.2 5 Sysplex
   Excluded: SAMPLESD/SAMPLEP/SC2
  Utilization condition UtilCond1 for CPC CPC1
   PU/PD/DU/DD/PT 65.0 5 57.0 5 ZIIP
  Recurring time condition PeakTime is active and enabled
   s/e/w days : 01/06/2014 /    01/20/2014 / XXXXX--
   s/d/e times:      17:00 / 20:15 / 23:55
End of report
```

The domain configuration report

To create a domain configuration report, issue a REPORT CONFIGURATION command. For the syntax of this command, see “REPORT CONFIGURATION” on page 134. The report contains information about the active domain configuration and the status of its elements. Indentation is used in this report to group related items. The reported information indicates:

- The name of the active domain configuration and its enabled or disabled status. A domain configuration is disabled if the Provisioning Manager is running in MANUAL processing mode. In all other processing modes, the domain configuration is enabled.
- Information about the CPCs that are defined in the domain configuration. For each CPC, such information contains:
  – The name of the CPC, its record ID, its current enabled or disabled status, and the default status that was defined in the domain configuration when it was activated.
  – Runtime information about the CPC:
    - The status of the CPC at the support element (SE) or hardware management console (HMC):

not correlated
The CPC is not yet detected at the support element or hardware management console. It cannot be used to activate temporary capacity.

correlated
The CPC is detected at the support element or hardware management console, but detailed information about it is not yet available to the Provisioning Manager. It cannot be used to activate temporary capacity.

matched
The CPC is detected at the support element or hardware management console, and detailed information about it is available. Temporary capacity can be activated on it if a valid record exists.

- The CPC hardware type and model. If the CPC is not supported for capacity management, it is indicated at the end of the line. If it is supported, then detailed information about the CPC follows in separate lines:
  - The current CPC model, capacity in MSU, number of zAAPs, and number of zIIPs.
  - The current number of IFLs, ICFs, and SAPs.
  - If the current CPC model differs from the permanent model, the permanent model and capacity in MSU.
  - If a usable record was detected:
    - If the record ID is defined as `Any (*)` in the domain configuration, the record ID used by the Provisioning Manager for managing temporary capacity is displayed.
    - The remaining capacity: either the number of processors available that can still be activated as general purpose capacity, zAAP, zIIP, IFL, ICF, SAP, or the remaining capacity tokens followed by the number of available processors. If capacity tokens are available, they are reported in MSU days for general purpose capacity and in processor days for zAAPs, zIIPs, IFLs, ICFs, and SAPs.
    - The activation limits: the number of zAAP, zIIP, IFL, ICF, and SAP processors allowed to be active for this record. If there is no activation limit in the record, a value of `-1` is reported.
    - The active resources: how many resources are currently active for the record. The resources are displayed in MSU, followed by the number of additional processors, the number of increases of the capacity level, the number of zAAPs, the number of zIIPs, the number of IFLs, the number of ICFs, and the number of SAPs.
  - If no usable record was detected, a note that no record for capacity management is available.
  - If a record ID is specified in the domain configuration and the record is not found, a note that the record ID is not valid.
- If the CPC cannot be used for capacity management, the reason that capacity cannot be activated or deactivated. For example, the record has expired or the configuration of the CPC does not allow commands to perform the temporary capacity change.

– If the CPC supports static power save function, whether the function is enabled or disabled. In disabled state it is also reported whether it is allowed to enable static power save mode.

- Information about the systems in the domain configuration. For each system, it contains:
  - The system name, the name of the sysplex it belongs to, its enabled or disabled status, and the default status as defined in the domain configuration when it was activated
  - The primary host address
  - The alternate host address, if defined
  - The protocol and the port
  - Information about the system at the primary host address. The observation status can be:

    observed
    The Provisioning Manager retrieves information from the system at the reported host address. The system can be observed if the domain configuration is enabled and the system is enabled. If the alternate host address is not defined, the system at the primary host address is observed. If both host addresses are defined, at least one of the two systems is observed. This depends on the runtime status of the system at the other host address.

    not observed
    The Provisioning Manager does not retrieve information from the system at the reported host address. The system is not observed if the domain configuration is disabled or if the system is disabled. If the domain configuration is enabled and the system is enabled, and if both host addresses are defined, the system does not need to be observed. Whether one or more systems are observed depends on the runtime status of the system at the other host address.
- If the system is observed, the following lines contain runtime information about the system at the reported host address. The first piece of information is the connection status and the time when the connection changed into this status. The connection status can be one of the following:

  not connected
  The connection to the system has not yet been established successfully. The Provisioning Manager is trying to connect to the system at the related host address.

  available
  The connection to the system has been successfully established.

  temporarily unavailable
  The connection to the system is no longer available. The Provisioning Manager tries to reestablish the connection.

  unavailable
  The connection to the system has been broken. The Provisioning Manager tries to connect to the system again.
- If additional information about the system is available, it is reported as follows:
  - The date and time the system became available
  - The CPC where the system is running
  - The installed WLM service definition and active WLM policy
- If Defined Capacity or Group Capacity is turned on for the system's LPAR, the corresponding information is available and the processing mode is not MANUAL, the following is reported:
  - The name of the system's LPAR, and its current Defined Capacity

- The name of the capacity group the system's LPAR belongs to, and its current Group Capacity

- If the Provisioning Manager does not yet have all required information about the system at the related host address, the following messages are generated:
  - A note indicating that the information is unavailable.
  - A message indicating which initialization step has not yet finished or was not able to finish successfully. If the step was not able to finish successfully, a corresponding message is sent to the console the first time it occurs. This message is one of the following:

    The system is not identified
    Either the name of the system and the name of the sysplex or the name of the system or the name of the sysplex has not yet been retrieved.

    The system is not correlated
    The Provisioning Manager has not yet detected whether the name of the system and the name of the sysplex that the system belongs to match the names in the domain configuration. Both names are reported in the message.

    The version of this system is not available
    The version of the operating system has not yet been retrieved.

    The CPC serial number is not available
    The serial number of the CPC where the system is running has not yet been retrieved.

    The CPC serial number is not correlated
    The serial number of the CPC has not yet been correlated with the name of the CPC. The serial number of the CPC is reported in the message.

    The CPC is not correlated
    The CPC has not yet been correlated. The Provisioning Manager has not detected whether the CPC the system is running on is defined in the domain configuration.

    Insufficient information for retrieving metric values
    The information for retrieving metric values has not yet been retrieved, or could not be retrieved.

    The information about the WLM service definition is not available
    The name of the installed WLM service definition, the name of the active WLM policy, or the activation time of this policy have not yet been retrieved.

    The information about WLM service class periods is not available
    The service class periods defined in the active WLM policy have not yet been retrieved.

- Even if all data can be successfully initialized, other conditions might prevent further processing. If so, the condition that is not fulfilled is reported. This condition might be:

  The system is not the defined system
  Either the name of the system and the name of the sysplex or the name of the system or the name of the sysplex this system belongs to, does not match the definition of these names in the domain configuration. Both names are reported in the message.

The version of this system is not supported
The version of the system is not supported by the Provisioning Manager.

The CPC is not part of the domain
The CPC the system is running on is not defined in the domain configuration.

An example of a domain configuration report is:

```
CPO1010I Configuration report generated at 12/22/2012 08:45:10
Domain configuration DCSAMPLE for domain DOMAIN1 is enabled
CPC SAMPCPC with record * is enabled (default enabled)
 SAMPCPC is matched with serial 000020016F7A since 12/22/2012 08:41:54
 Hardware is of type 2827 with model H66
 Current model is 713 with 1822 MSU, 1 zAAPs, 1 zIIPs,
  51 IFLs, 0 ICFs, 12 SAPs
 Permanent model is 709 with 1350 MSU
 Active record ID is A0123456
 Residual capacity GP/zAAP/zIIP/IFL/ICF/SAP 100/50/50/250/0/75 spares=4
 Activation limits are 3 zAAPs, 1 zIIP, 5 IFLs, 0 ICFs, 5 SAPs
 Active resources GP/zAAP/zIIP 472(4/0)/0/0
                 IFL/ICF/SAP 0/0/0
 System SYSTEM1 in sysplex SVPLEX is enabled (default enabled)
  Primary host address: system1.ibm.com
  Alternate host address: 9.99.99.99
  Protocol: HTTP, port: 5988
  The system at primary host address is observed
   This system is available since 12/22/2008 08:41:56
   This system is running on CPC SAMPCPC
   WLM service definition: SAMPLESD, active policy: SAMPLEP
  LPAR SAMPLPAR with defined capacity of 550 MSU
  LPAR SAMPLPAR in capacity group GROUP1 with 750 MSU
  The system at alternate host address is not observed
```

The workload report

To create a workload report, issue a REPORT WORKLOAD command. For the syntax of this command, see “REPORT WORKLOAD” on page 143. The report contains information the workload that is currently being observed. The reported information indicates:

- The number of systems for which the workload can be observed
- For each system that is observed:
  - The name of the system, the name of the sysplex to which the system belongs, and the name of the CPC on which the system is running
  - For each WLM service class period that is observed on the system:
    - The name and number of a service class period, for example, SAMPSC.1, followed by the provisioning performance index limit (PL), the provisioning performance index limit duration (PD), the deprovisioning performance index limit (DL), the deprovisioning performance index limit duration (DD), and the scope for the performance index (S)
    - The last measured performance index and the time of this measurement. A dash (-) is displayed whenever the performance index is not available at the time of measurement. For example, this situation can occur when the workload that is running in the observed service class has been displaced. To show how long the performance index has been at this level, the line also indicates the last time the performance index crossed the provisioning or deprovisioning performance index limit. Both times are shown in UTC.

- If a detailed workload report is issued (`type=detailed`), it also lists the types of additional capacity that are needed for a service class that is suffering or, optionally, the reason why additional capacity is not needed by a suffering service class.

An example of a workload report is:

```
CPO1047I Workload report generated at 12/22/2012 08:55:10
Workload is analyzed for 1 system(s)
Workload for system SYSTEM1 of sysplex SYSPLEX on CPC SAMPCPC
 SAMPSC.1 PL/PD/DL/DD/S 1.4 10 1.2 10 Sysplex
  PI from 12/22/2012 08:54 is 1.6. Last limit crossing was 12/22/2012 08:45
   Demand for additional logical zIIPs not recognized
    Lack of spare logical zIIPs
   Demand for additional physical zAAPs not recognized
    Shared logical zAAPs not available on LPAR
   Demand for additional logical CPs recognized
   Demand for capacity level increase only recognized.
   Demand for additional physical CPs not recognized
    Amount of logical online CPs outnumbers amount of physical CPs
```

If management with moving average PI is enabled, the workload report lists the user settings that are being used for the moving average PI calculation, such as weight and capping. The report also lists the PI as it has been reported by WLM for each observed WLM service class period, as well as the alternate moving average PI that is calculated by Capacity Provisioning.

An example for this kind of workload report is:

```
CPO1047I Workload report generated at 12/22/2012 08:55:11
Workload is analyzed for 1 system(s)
average PI management enabled: weight 57%, capping PI 2.79
Workload for system SYSTEM1 of sysplex SYSPLEX on CPC SAMPCPC
 SAMPSC.1 PL/PD/DL/DD/S 1.4 10 1.2 10 Sysplex
  average PI from 12/22/2012 08:54 is 1.46. Last limit crossing was 12/22/2012 08:47
  PI from 12/22/2012 08:54 is 1.6. Last limit crossing was 12/22/2012 08:45
```

For more information about setting the moving average weight, see the **SystemObservation.MovingAveragePiWeight** key in Table 20 on page 54.

For more information about setting the moving average capping, see the **SystemObservation.MovingAveragePiCapping** key in Table 20 on page 54.

The utilization report

To create a utilization report, issue the **REPORT UTILIZATION** command. For the syntax of this command, see "REPORT UTILIZATION" on page 141. The report contains information about the physical utilization that is being observed on behalf of active utilization conditions of the current provisioning policy. The reported information indicates:

- The number of CPCs for which physical utilization can be observed.
- For each CPC that is observed:
  - The name of the CPC and the time of the last data collection.
  - The current readings of the physical utilization percentages for processor types CP (general purpose), zAAP and zIIP. If the data is not available or is not monitored, a dash (-) is displayed.
- Each active utilization condition that affects this CPC; for each utilization condition that is displayed:

- The Provisioning Utilization limit (PU), Provisioning Duration (PD), Deprovisioning Utilization limit (DU), Deprovisioning Duration (DD) and Processor Type (PT) settings.
- The last time the processor utilization of the given processor type crossed the provisioning or deprovisioning utilization limits. The time is shown in UTC.
- Optionally, if a demand has been recognized on behalf of this utilization condition, the demand type.

An example of a utilization report is:

```
CPO1022I CPC utilization report generated at 12/22/2012 08:55:10
Utilization is observed for 1 CPC(s)
SAMPCPC: Utilization from 12/22/2012 08:54:00
 CP 90%, zAAP -%, zIIP 65%
  Utilization Condition POLUC1
  PU/PD/DU/DD/PT 95% 5 92% 5 CP
  Last CPC utilization threshold crossing was 12/22/2012 08:48:00
  Demand for addition physical CPs recognized
  Demand for capacity level increase recognized
End of report
```

The activity report

To create an activity report, issue the **REPORT ACTIVITY** command. For the syntax of this command, see "REPORT ACTIVITY" on page 133. The report contains information about the activation and deactivation of additional capacity that was initiated by the Provisioning Manager based on the policy and workload status.

This activity is displayed in reverse chronological order, with the latest actions displayed at the top of the list, so that when the list is truncated, only the oldest data is not displayed. Activation and deactivation are processed asynchronously by the CPC. The activity report may also show whether failures to activations and deactivations have been detected and whether manually activated capacity has been passed to the Provisioning Manager management.

The reported information includes:

- A summary line that shows the time period reported and the number of activities performed by the Provisioning Manager.
- Optionally, if failures have been detected for the requested period, a summary line containing the number of failures in the report.
- For processor activities the following information is included:
  - The type of activity (activation or deactivation), the CPC affected and the time the activity took place.
  - The target activation level. This includes the target model for general purpose capacity and the numbers of temporary zAAP and zIIP processors.
  - The activation level of the CPC at the time the activity was initiated, in the same format as the target activation level.
  - If the activity was an activation of additional capacity, for each workload that triggered the activation:
    - The policy element that triggered the activation: the policy name, the provisioning rule, the provisioning condition and the time condition.
    - The system that was suffering and the sysplex it belongs to.
    - The service class period that was suffering: the active WLM service definition, the WLM policy name, the service class name and the period number.

  - If the activity was an activation of additional capacity for each utilization condition that triggered the activation:
    - The policy element that triggered the activation: the policy name, the provisioning rule, the provisioning condition and the time condition.
    - The utilization condition name.
- For Defined Capacity or Group Capacity activities, the following information is included:
  - The type of activity (capacity increase or decrease), the affected LPAR or capacity group, the affected CPC and the time the activity took place
  - The target capacity
  - The capacity before the activity was initiated
  - If the activity was a workload driven capacity increase, for each workload that triggered the activation:
    - The policy element that triggered the activation: the policy name, the provisioning rule, the provisioning condition, and the time condition
    - The system that suffered from the capacity shortage and the sysplex to which it belongs
    - The service class period that suffered from the capacity shortage: the active WLM service definition, the WLM policy name, the service class name, and the period number
- Information about command failures that have been asynchronously reported. Such information includes the time when the failure was detected, optionally the name of the affected LPAR or capacity group, the name of the affected CPC, and optionally the code of the cause of failure reported by the hardware.
- For processor capacity, information about manually activated capacity passed to the Provisioning Manager for deactivation. The entry contains information about the CPC and the new total managed capacity of the Provisioning Manager in MSU for general purpose capacity and number of processors for zAAP and zIIP capacity.

An example of an activity report is:

```
CPO1042I Activity report generated at 01/17/2013 08:45
Number of activities between 11/14/2012 and 01/17/2013 was 3
Deactivation for CPC SAMPCPC at 01/16/2013 21:26:29
    Deactivation to model 505, 0 zAAPs, 0 zIIPs
    Active resources before deactivation: model 506, 0 zAAPs, 0 zIIPs
Activation for CPC SAMPCPC at 01/16/2013 17:26:25
    Activation of model 506, 0 zAAPs and 0 zIIPs
    Active resources before activation: model 505, 0 zAAPs, 0 zIIPs
    Inducing policy element is policy EXAMPLE, rule SAMPRULE,
        provisioning condition SAMPCS, time condition SAMPTIME
    Inducing system is SYSTEM1 in sysplex SYSPLEX
    Inducing workload is WLM service definition SAMPLESD,
        policy SAMPLEP, service class period SAMPSC.1
Activation for LPAR IRD6 on CPC P35 at 01/16/2013 08:35:03
    Activation of 100 MSU
    Capacity after activation: 1000 MSU
    Capacity before activation: 900 MSU
    Inducing policy element is policy EXAMPLE, rule SAMPRULE,
            provisioning condition SAMPCS2, time condition SAMPTIME
    Inducing system is SYSTEM1 in sysplex SYSPLEX
    Inducing workload is WLM service definition SAMPLESD,
            policy SAMPLEP, service class period SAMPSC.1
```

The record report

To create a record report, issue the **REPORT RECORD** command. For the syntax of this command, see "REPORT RECORD" on page 140. The report contains information about the On/Off CoD record of a CPC that is managed by the Provisioning Manager. The displayed record either has been specified in the domain configuration or otherwise is chosen by the Provisioning Manager. The report helps you to understand which resources are currently active, which have been activated by the Provisioning Manager and which have been activated manually. It also shows residual capacity allowed by the record and its activation limits.

The reported information includes:

Record Id
The record ID of the currently managed On/Off CoD record. Displays that no record is available if no On/Off CoD record has been detected.

CPC name
The name of the CPC.

Management state
The management state of the Provisioning Manager for the management of the record. Optionally contains the time when the state will conclude. Displays that no additional information is available if the information of the record has not been read yet. Possible values are:

blocked
A capacity change of the CPC has been detected recently. The Provisioning Manager will not perform any capacity changes until the displayed time.

deprovisioning mismatch
The Provisioning Manager initiated the deactivation of temporary capacity and an expected notification about the capacity change has not been detected within a certain time. An outstanding operator message asking how to continue needs to be answered.

deprovisioning outstanding
The Provisioning Manager has deactivated additional capacity of the record and is waiting for a response from the CPC to confirm the deactivation.

deprovisioning rejected
The Provisioning Manager is in confirmation mode and a recommendation to deactivate additional capacity has been rejected by the operator. Recommendations for the deactivation of additional capacity will not be issued before the displayed time.

deprovisioning requested
The Provisioning Manager is in confirmation mode and currently has an outstanding message asking to confirm deactivation of additional capacity.

not provisioned
All information about the record is available. The Provisioning Manager currently has no activated resources of the record.

provisioning mismatch
The Provisioning Manager initiated the activation of temporary capacity and an expected notification about the capacity change

has not been detected within a certain time. An outstanding operator message asking how to continue needs to be answered.

provisioned, deprovisioning blocked
The Provisioning Manager has activated additional capacity of the record. No additional capacity is currently needed and the additional capacity will not be deactivated before the displayed time.

provisioning outstanding
The Provisioning Manager has activated additional capacity of the record and is waiting for a response from the CPC to confirm the activation.

provisioning rejected
The Provisioning Manager is in confirmation mode and a recommendation to activate additional capacity has been rejected by the operator. Recommendations for the activation of additional capacity will not be issued before the displayed time.

provisioning requested
The Provisioning Manager is in confirmation mode and currently has an outstanding message asking to confirm the activation of additional capacity.

provisioned
The Provisioning Manager has activated additional capacity of the record. Currently no further changes are needed.

Expiration date
The expiration date and time, indicating when the record will expire.

Active resources
The active resources, representing the additional capacity of the record that is currently active. Active resources of general purpose capacity are reported in MSU. Active zAAP, zIIP, IFL, ICF and SAP resources are reported in number of processors.

Managed resources
The managed resources, describing the part of the active resources that is managed by the Provisioning Manager. The resources are reported as number of general purpose processors (CPs), number of capacity level increments (CLIs), number of zAAP processors, and number of zIIP processors.

Activation time
The activation time, indicating when the record was activated the last time, as provided by the CPC.

Activation limits
The activation limits of the record report the maximum number of zAAP, zIIP, IFL, ICF and SAP processors that can be active at any time. Processor types without activation limits display the value -1.

Residual capacity
The capacity that can still be consumed on behalf of the record. General purpose capacity is reported in MSU days. ZAAP, zIIP, IFL, ICF and SAP capacity is reported in processor days. Processor types without activation limits display the value -1.

Allowed models
The allowed models for general purpose capacity. Displays a table that is

arranged by the capacity identifiers of models that are allowed by the On/Off CoD record. Each line displays the following details related to the respective model:

Model Capacity model identifier

CP Number of additional general purpose processors

CLI Number of additional capacity level increments

MSU original
Capacity, in MSU, that is billed to the record

MSU absolute
Total general purpose capacity in MSU

MSU relative
Change of capacity, in MSU, compared to the active model

MSU managed
Capacity, in MSU, that is managed by the Provisioning Manager if the Provisioning Manager would activate this model. This value is reported as -- if this model must not be activated by the Provisioning Manager.

Activation type
Indicates whether this model is managed by the Provisioning Manager or manually or is not active. Possible values are:

MAN The model is managed manually.

PM The model is managed by the Provisioning Manager.

-- The model is not active.

An example of an record report is:

```
CPO4430I Record report generated at 02/06/2014 09:12:15
  Record Id:              SAMPREC2 (On/Off CoD)
   CPC name:              SAMPCPC
   Management state:      provisioned
   Expiration date:       12/31/2014 23:59:59
   Active resources:      145 MSU, 0 zAAPs, 2 zIIPs
                          0 IFLs, 0 ICFs, 0 SAPs
   Managed resources:     1 CPs, 0 CLIs, 0 zAAPs, 0 zIIPs
   Activation time:       02/04/2014 15:21:49
   Activation limits:     3 zAAPs, 3 zIIPs
                          3 IFLs, 3 ICFs, 3 SAPs
   Residual capacity:     150 MSU days,4 zAAP days, 5 zIIP days
                          1 IFL days, 1 ICF days, 1 SAP days
   Allowed models:
    Model  CP   CLI     MSU      MSU      MSU      MSU    Activation
                      original absolute relative  managed    type
      505    0    0        0     240        -145     --        MAN
      506    1    0       39     279        -106     --        MAN
      605    0    1       52     292         -93     --        MAN
      507    2    0       77     317         -68     --        MAN
      606    1    1       99     339         -46      0        MAN
      705    0    2      123     363         -22     --        MAN
      607    2    1      145     385           0     46         PM
      706    1    2      182     422          37     --         --
      707    2    2      239     379          94     --         --
  End of report
```

The defined capacity report

To create a defined capacity report, issue the **REPORT DEFINEDCAPACITY** command. For the syntax of this command, see "REPORT DEFINEDCAPACITY" on page 135. The report contains information about the Defined Capacity of a system that is managed by the Provisioning Manager. The displayed Defined Capacity belongs to an LPAR which runs a system observed by the Provisioning Manager. The report helps you to understand how much Defined Capacity is currently active and which part of it has been provisioned by the Provisioning Manager.

The reported information indicates:

CPC.LPAR
The name of the CPC and LPAR where the Defined Capacity is located.

Sysplex.System
The sysplex and the system name of the system running in the LPAR which provides the management data.

Management state
The current state of the Provisioning Manager for the management of the Defined Capacity. Optionally contains the time when the state will conclude. Possible values are:

blocked
A Defined Capacity increase or decrease has been detected recently. The Provisioning Manager will not perform any Defined Capacity changes until the displayed time.

blocked after failure
The Provisioning Manager attempted to change the Defined Capacity but received a notification from the hardware interface that it failed. The Provisioning Manager will not perform any Defined Capacity changes until the displayed time.

blocked after restart
The Provisioning Manager has resumed processing and needs to readjust its management.

change outstanding
The Provisioning Manager initiated a Defined Capacity change that has not been confirmed yet by the observed system running in the affected LPAR.

connecting to hardware
The Provisioning Manager is trying to connect to the hardware interface.

connection lost
Connection to the observed system on the LPAR has been lost and is not available again yet.

continue management requested
A significant Defined Capacity decrease has been detected. The Provisioning Manager has issued a request message to the console asking if it should adjust the Defined Capacity Management Base. The user response is pending.

decrease blocked
The Provisioning Manager has increased Defined Capacity. No

Defined Capacity increase is currently needed and increased Defined Capacity that is no longer required will not be decreased before the displayed time.

decrease rejected
The Provisioning Manager is in confirmation mode and a recommendation to decrease Defined Capacity has been rejected by the operator. Recommendations for a Defined Capacity decrease will not be issued before the displayed time.

decrease requested
The Provisioning Manager is in confirmation mode and currently has an outstanding message asking to confirm the decrease of Defined Capacity.

decrease suggested
The Provisioning Manager is in analysis mode and has suggested to decrease Defined Capacity.

global minimum reached
The Provisioning Manager has detected a need for decreased Defined Capacity, but the global minimum limit disallows a further decrease. Defined Capacity will not be decreased under the current circumstances.

increased
The Provisioning Manager has increased Defined Capacity. Currently no further changes are needed.

increase rejected
The Provisioning Manager is in confirmation mode and a recommendation to increase Defined Capacity has been rejected by the operator. Recommendations for a Defined Capacity increase will not be issued before the displayed time.

increase requested
The Provisioning Manager is in confirmation mode and currently has an outstanding message asking to confirm the increase of Defined Capacity.

increase suggested
The Provisioning Manager is in analysis mode and has suggested to increased Defined Capacity.

initializing 1
Information from the observed system on the LPAR is not yet available.

initializing 2
Defined Capacity information about the LPAR is not yet available from the hardware interface.

not increased
Defined Capacity information of LPAR is available. Defined Capacity has not yet been increased.

not managed
in manual mode.

policy limit reached
The Provisioning Manager has detected a demand for increased

Defined Capacity, but the policy limits have been exhausted. Defined Capacity will not be further increased under the current circumstances.

reinitializing
Information from the observed system on the LPAR is not available after a connectivity problem or a domain configuration change.

For the management states `initializing 1`, `initializing 2`, `reinitializing`, `connection lost`, and `not managed`, the following values will be displayed as `not available`:

Policy limit
The policy limit indicates the sum of additional Defined Capacity in MSU of all rule limits whose rules currently trigger a demand for additional Defined Capacity.

Management base
The Defined Capacity management base indicates the Defined Capacity in MSU at which the Provisioning Manager takes up and eventually concludes management.

Managed capacity
The additional Defined Capacity in MSU which is currently managed by the Provisioning Manager. This is the amount MSU by which the Provisioning Manager has currently increased the Defined Capacity.

Current capacity
The current Defined Capacity in MSU.

Remaining time until capping
The remaining time in minutes until the reported LPAR is expected to be capped by Defined Capacity or Group Capacity.

4 hour rolling average consumption
The current four hour rolling average consumption of the reported LPAR.

An example of a defined capacity report is:

```
CPO1095I Defined capacity report generated at 12/22/2012 08:45:10
Defined capacity for LPAR SAMPLPAR on CPC SAMPCPC
 CPC.LPAR:                          SAMPCPC.SAMPLPAR
 Sysplex.System:                    SYSPLEX.SYSTEM1
 Management state:                  increased
 Policy limit:                      50 additional MSU
 Management base:                   100 MSU
 Managed capacity:                  30 additional MSU
 Current capacity:                  130 MSU
 Remaining time until capping:      3 minutes
 4 hour rolling average consumption: 122 MSU
End of report
```

The group capacity report

To create a group capacity report, issue the **REPORT GROUPCAPACITY** command. For the syntax of this command, see "REPORT GROUPCAPACITY" on page 136. The report contains information about the Group Capacity for a capacity group of a CPC that is managed by the Provisioning Manager. The displayed Group Capacity belongs to a capacity group that contains an LPAR which runs a system observed by the Provisioning Manager. The report helps you to understand how much Group Capacity is currently active and which part of it has been provisioned by the Provisioning Manager.

The reported information indicates:

Group name
Name of the capacity group.

CPC name
Name of the processor complex on which the capacity group is defined.

Management state
The current state of the Provisioning Manager for the management of the Group Capacity. Optionally contains the time when the state will conclude. Possible values are:

blocked
A Group Capacity increase or decrease has been detected recently. The Provisioning Manager will not perform any Group Capacity changes until the displayed time.

blocked after failure
The Provisioning Manager attempted to change the Group Capacity but received a notification from the hardware interface that it failed. The Provisioning Manager will not perform any Group Capacity changes until the displayed time.

blocked after restart
The Provisioning Manager has resumed processing and needs to readjust its management.

change outstanding
The Provisioning Manager initiated a Group Capacity change that has not been confirmed yet by an observed system running in an LPAR belonging to the capacity group.

connecting to hardware
The Provisioning Manager is trying to connect to the hardware interface.

connection lost
Connection to any observed system on an LPAR belonging to the capacity group has been lost and is not available again yet.

continue management requested
A significant Group Capacity decrease has been detected. The Provisioning Manager has issued a request message to the console asking if it should adjust the Group Capacity Management Base. The user response is pending.

decrease blocked
The Provisioning Manager has increased Group Capacity. No Group Capacity increase is currently needed and increased Group Capacity that is no longer required will not be decreased before the displayed time.

decrease rejected
The Provisioning Manager is in confirmation mode and a recommendation to decrease Group Capacity has been rejected by the operator. Recommendations for a Group Capacity decrease will not be issued before the displayed time.

decrease requested
The Provisioning Manager is in confirmation mode and currently has an outstanding message asking to confirm the decrease of Group Capacity.

decrease suggested
The Provisioning Manager is in analysis mode and has suggested to decrease Group Capacity.

global minimum reached
The Provisioning Manager has detected a need for decreased Group Capacity, but the global minimum limit disallows a further decrease. Group capacity will not be decreased under the current circumstances.

initializing 1
Information from an observed system on an LPAR belonging to the capacity group is not yet available.

initializing 2
Group capacity information about the capacity group is not yet available from the hardware interface.

increased
The Provisioning Manager has increased Group Capacity. Currently no further changes are needed.

increase suggested
The Provisioning Manager is in analysis mode and has suggested to increased Group Capacity.

increase requested
The Provisioning Manager is in confirmation mode and currently has an outstanding message asking to confirm the increase of Group Capacity.

increase rejected
The Provisioning Manager is in confirmation mode and a recommendation to increase Group Capacity has been rejected by the operator. Recommendations for a Group Capacity increase will not be issued before the displayed time.

not increased
Group capacity information is available. Group capacity has not yet been increased.

policy limit reached
The Provisioning Manager has detected a demand for increased Group Capacity, but the policy limits have been exhausted. Group capacity will not be further increased under the current circumstances.

reinitializing
Information from any observed system on an LPAR belonging to the capacity group is not available after a connectivity problem or a domain configuration change.

not managed
The Provisioning Manager is in manual mode.

For the management states `initializing 1`, `initializing 2`, `reinitializing`, `connection lost`, and `not managed`, the following values will be shown as `not available`:

Policy limit
The policy limit indicates the sum of additional Group Capacity in MSU of all rule limits whose rules currently trigger a demand for additional Group Capacity.

Management base
The Group Capacity management base indicates the Group Capacity in MSU at which the Provisioning Manager takes up and eventually concludes management.

Managed capacity
The additional Group Capacity in MSU which is currently managed by the Provisioning Manager. This is the amount MSU by which the Provisioning Manager has currently increased the Group Capacity.

Current capacity
The current Group Capacity in MSU.

An example of a group capacity report is:

```
CPO1096I Group capacity report generated at 12/22/2012 08:45:10
Group capacity for group GROUP1 on CPC SAMPCPC
  Group name:                       GROUP1
  CPC name:                         SAMPCPC
  Management state:                 increased, decrease blocked
                                    until 12/22/2012 08:55:05
  Policy limit:                     80 additional MSU
  Management base:                  250 MSU
  Managed capacity:                 45 additional MSU
  Current capacity:                 295 MSU
End of report
```

The trace report

To create a trace report, issue the **REPORT TRACE** command. For the syntax of this command, see "REPORT TRACE" on page 141. The report contains information about the current trace level settings and trace file settings:

Trace level settings
The information under this heading includes:

- The value for the default level
- The values for each of the trace components

Trace file settings
The information under this heading includes:

- The prefix of the trace file name. The full file name is constructed from this by adding a dot (`.`) and the sequence number of the trace, starting from zero. For example, the first trace file might be `/tmp/cpotrace_20080127134147469.log.0`.
- The maximum file size and number of trace files. When the file size limit is reached, a new file is started. When the maximum number is reached, the sequence number begins again at zero and the old files are reused and overwritten.

An example of a trace report is:

```
CPO1050I Trace report generated at 12/22/2012 08:45:10
Trace level settings
 Default level is set to INFO
 Level for component BASE is set to default
 Level for component UTIL is set to default
 Level for component POLICY is set to ALL
 Level for component CONFIGURATION is set to ALL
 Level for component ANALYZER is set to default
 Level for component PLANNER is set to default
 Level for component CONSOLESERVER is set to default
 Level for component CIMOBSERVER is set to default
 Level for component CIMCLIENT is set to OFF
 Level for component HMCCLIENT is set to OFF
Trace file settings
 Trace is written to directory /tmp
 Rotating set(s) of files, number of files is 2
 File name prefix is cpotrace_20122212084052.log
 Size is limited to 100 MB
```

The log report

To create a log report, issue the **REPORT LOG** command. For the syntax of this command, see "REPORT LOG" on page 138. The report contains information about the current log and log file settings, including:

- A header line that shows the configured destination for log files
- A line for each log that states if data writing is active for this log

An example of a log report is:

```
CPO1091I Log status report generated at 01/17/2014 08:16:56
Default destination is ./Logs/
Writing for log NotificationLog is inactive
Writing for log ErrorLog is active
Writing for log AnalyzerLog is inactive
Writing for log SystemObservationLog is inactive
Log is written to directory /user/x/Logs
End of report
```

Chapter 7. Considerations for defining policies and domains

You should consider several points when you are setting parameters in a Capacity Provisioning policy or domain configuration.

Choosing service class periods for workload conditions

z/OS Capacity Provisioning uses the performance index (PI) of service class periods as the primary trigger for provisioning and deprovisioning actions. For example, the Capacity Provisioning Manager considers a provisioning action only if the actual PI of any included service class period is worse than the defined PI. You must select appropriate service class periods for which the performance index is correlated to the performance of your business application.

For example, an important business application can consist of multiple service classes that are defined with importance 2 and 3 in the WLM service definition. Somewhat simplified, the goal attainment of those service classes primarily depends on other work that is classified as importance 1, SYSSTC, or SYSTEM. When system resources, such as processing capacity, become constrained, WLM attempts to help the most important work first and assign fewer resources to other work. In this example, resources are taken from the service class periods with an importance of 3 first, and they in turn show a higher (worse) PI.

It is best practice in WLM not to activate too many service class periods at any point in time. There must be a substantial service measurement in any period so that WLM has sufficient sample data. Having sufficient amounts of sample data is especially important for service class periods that are defined with average response times or response time with percentile goals. The same objective applies to Capacity Provisioning. The number of transaction endings should be high enough to allow your monitoring product, such as RMF, to compute a performance index. At least one transaction ending is required for each gathering interval, such as the RMF MINTIME, and preferably more endings to yield significant interval data.

You can specify service class periods with importance levels of 1–5 only. `SYSTEM`, `SYSSTC`, `SYSSTC1` to `SYSSTC5`, and `DISCRETIONARY` work cannot trigger a provisioning action and are ignored, if included. Do not include any service classes that are associated with a maximum capacity for a resource group. If a service class is capped, Capacity Provisioning avoids provisioning on behalf of it, if possible.

Choosing provisioning criteria for workload conditions

The PI limits that you specify in your provisioning criteria depend on what you would consider tolerable when capacity is constrained. If you define the goal for the service class period so that the goal can still be achieved in times of peak demand, and additional capacity does not need to be active, the provisioning PI could be set just above 1, down to an allowed value of 1.3. In other cases, with more aggressive goals in effect, a provisioning PI would need to be higher.

The provisioning duration determines how fast the Provisioning Manager activates additional capacity when the PI of that service class period is consistently above the limit. The duration must be viewed relatively to the gathering interval length of the monitoring product, such as the RMF MINTIME, whose default is 100

seconds. In general, the duration should be longer than 3 minutes, to prevent a short-term disturbance from triggering a provisioning action. It can also take some time for WLM to resolve a PI problem by reassigning resources.

The specified duration is not a guaranteed reaction time. Certain events cause the Provisioning Manager to block some time to allow WLM to readjust. Events that can trigger a block include, for example, the activation of a new WLM service definition or policy, or a capacity change in the observed CPC.

Utilization conditions versus workload conditions

In some configurations it may be difficult to identify specific WLM service class periods that indicate capacity bottlenecks timely enough. In such a situation you would first want to verify that the WLM policy specifies a correct importance and especially a correct goal for the service class period to be monitored. For example, a goal that is too relaxed can lead to a late but very rapid increase of the performance index as additional work needs to be handled.

Some few configurations may even suffer from processor contention at a relatively moderate physical processor utilization level with hardly any processor delays, making it difficult for the Provisioning Manager to detect the need for additional processor capacity.

If it is not possible to base Capacity Provisioning management on a set of one or more WLM service class periods or your configuration needs additional processor capacity in situations that lack of the characteristic signs of a processor contention, you still have the option to base Capacity Provisioning management on the physical utilization of the shared general purpose processor pool, by means of defining utilization conditions against the entire CPC.

Utilization conditions allow for a possibly more immediate response to high physical processor utilization, without waiting for delayed effects on WLM service class period performance which is needed for workload conditions.

On the other hand, utilization conditions do not discriminate between business important workload that would qualify for a provisioning action and less important workload which would not justify a provisioning action. In addition, utilization conditions will not ensure that systems running the business important workload will benefit from the provisioned additional capacity.

Part 3. Reference

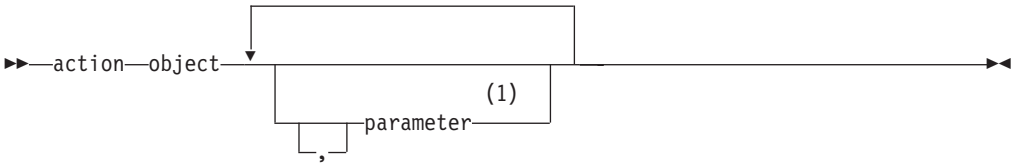
Provides an overview of the Provisioning Manager commands.

Chapter 8. Provisioning Manager command reference

Describes the Provisioning Manager commands and their syntax. For information about how to pass commands to the Provisioning Manager, see "Provisioning Manager commands" on page 74.

Introduction to Provisioning Manager commands

The Provisioning Manager command syntax is based on the z/OS command syntax shown in the following diagram. For information about these diagrams, see "How to read syntax diagrams" on page x.



Notes:

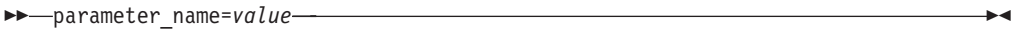
1 Parameters can be required, optional or nonexistent, depending on the specific action and object.

Commas are optional, and any number of blanks are allowed to separate action, object, and parameters.

A comment is allowed at any place where a blank is allowed. Comments have the following form:

```
/* comment */
```

Parameters have the following form:



You can specify the parameters in any order. A value can be any sequence of characters. If special characters are needed, such as spaces or commas, the parameter value must be enclosed in quotation marks (`'value'`). Within these quotation marks, any character is allowed. To specify an apostrophe (') within the parameter value, enclose it in quotation marks. For example, `parm1='It''s mine'` assigns the value `It's mine` to the **parm1** parameter. Actions, objects, and parameter names are not case-sensitive. Values can be case-sensitive if they are not keywords. In this case the value must be enclosed in quotation marks.

The commands fall into three types:

- *Control commands*, which alter the operation of the Provisioning Manager. These are listed in Table 21 on page 104.
- *Status commands*, which report the current situation. These are listed in Table 22 on page 104.
- *Debug commands*, which can be used to give diagnostic information to IBM service personnel. These are listed in Table 23 on page 104.

The supported commands, such as actions and objects, are listed in the tables together with a short form of the command and the page on which each command is described. The short forms can be used to save typing; note that the short forms are not abbreviations.

Table 21. Control commands

Command	Short form	See...
`ACTIVATE DEFINEDCAPACITY`	`a dc`	"ACTIVATE DEFINEDCAPACITY" on page 106
`ACTIVATE GROUPCAPACITY`	`a gc`	"ACTIVATE GROUPCAPACITY" on page 108
`ACTIVATE RESOURCE`	`a r`	"ACTIVATE RESOURCE" on page 112
`DEACTIVATE DEFINEDCAPACITY`	`i dc`	"DEACTIVATE DEFINEDCAPACITY" on page 114
`DEACTIVATE GROUPCAPACITY`	`i gc`	"DEACTIVATE GROUPCAPACITY" on page 116
`DEACTIVATE RESOURCE`	`i r`	"DEACTIVATE RESOURCE" on page 119
`DISABLE CONFIGURATION`	`d c`	"DISABLE CONFIGURATION" on page 121
`DISABLE POLICY`	`d p`	"DISABLE POLICY" on page 122
`DISABLE POWERSAVE`	`d ps`	"DISABLE POWERSAVE" on page 123
`ENABLE CONFIGURATION`	`e c`	"ENABLE CONFIGURATION" on page 125
`ENABLE POLICY`	`e p`	"ENABLE POLICY" on page 126
`ENABLE POWERSAVE`	`e ps`	"ENABLE POWERSAVE" on page 127
`MANAGE RESOURCE`	`m r`	"MANAGE RESOURCE" on page 130
`RESET CONFIGURATION`	`t c`	"RESET CONFIGURATION" on page 145
`RESET POLICY`	`t p`	"RESET POLICY" on page 146
`SET DOMAIN`	`s d`	"SET DOMAIN" on page 148
`STOP MANAGER`	`p m`	"STOP MANAGER" on page 152

Table 22. Status commands

Command	Short form	See...
`LIST CONFIGURATION`	`l c`	"LIST CONFIGURATION" on page 128
`LIST POLICY`	`l p`	"LIST POLICY" on page 129
`REPORT ACTIVITY`	`r a`	"REPORT ACTIVITY" on page 133
`REPORT CONFIGURATION`	`r c`	"REPORT CONFIGURATION" on page 134
`REPORT DEFINEDCAPACITY`	`r dc`	"REPORT DEFINEDCAPACITY" on page 135
`REPORT DOMAIN`	`r d`	"REPORT DOMAIN" on page 137
`REPORT GROUPCAPACITY`	`r gc`	"REPORT GROUPCAPACITY" on page 136
`REPORT LOG`	`r l`	"REPORT LOG" on page 138
`REPORT POLICY`	`r p`	"REPORT POLICY" on page 139
`REPORT RECORD`	`r r`	"REPORT RECORD" on page 140
`REPORT TRACE`	`r t`	"REPORT TRACE" on page 141
`REPORT UTILIZATION`	`r u`	"REPORT UTILIZATION" on page 141
`REPORT WORKLOAD`	`r w`	"REPORT WORKLOAD" on page 143

Table 23. Debug commands

Command	Short form	See...
`ACTIVATE LOG`	`a l`	"ACTIVATE LOG" on page 110
`DEACTIVATE LOG`	`i l`	"DEACTIVATE LOG" on page 118

Table 23. Debug commands (continued)

Command	Short form	See...
DUMP MANAGER	none	"DUMP MANAGER" on page 124
SET TRACE	**s t**	"SET TRACE" on page 150
RESET TRACE	**t t**	"RESET TRACE" on page 147
WRITE LOG	**w l**	"WRITE LOG" on page 153

Commands

ACTIVATE DEFINEDCAPACITY

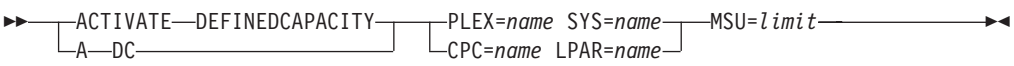
Use the **ACTIVATE DEFINEDCAPACITY** command to manually increase the Defined Capacity for an LPAR. You can identify the LPAR either by the name of the LPAR and the name of the CPC on which the LPAR is active, or by the name of the z/OS system and sysplex running in the LPAR. In the second case, the z/OS system needs to be defined in the active domain configuration. In every case the respective CPC has to be defined in the active domain configuration.

Note: The number that is supplied as the MSU parameter is the total limit to which the Defined Capacity should be increased, not the number of MSU by which you want to increase the limit.

Before performing the activation, the Provisioning Manager checks for the following conditions:

- that the specified LPAR is uniquely identified
- that at the HMC Defined Capacity is turned on for the LPAR
- that the target limit is an increase, meaning that the new Defined Capacity is higher than the current Defined Capacity.

Syntax



Parameters

The command has the following parameter:

PLEX=*name*
The name of the sysplex to which the specified system belongs.

SYS=*name*
The name of the system for which LPAR you want to increase Defined Capacity. The specified system must be currently observed.

CPC=*name*
The name of the CPC on which to find the specified LPAR.

LPAR=*name*
The name of an LPAR for which you want to increase the Defined Capacity.

MSU=*limit*
The new Defined Capacity for the specified LPAR. The limit is specified in MSU. The new value must be higher than the current Defined Capacity and Defined Capacity must be turned on for the LPAR at the HMC. A maximum value of 100000 MSU can be specified.

Example

To increase the Defined Capacity for system SYS1 in sysplex PLEX 2 to 1000 MSU, issue the following command:

```
MODIFY CPOSERV,APPL=ACTIVATE DEFINEDCAPACITY SYS=SYS1 PLEX=PLEX2 MSU=1000
```

or:

```
F CPOSERV,APPL=A DC SYS=SYS1 PLEX=PLEX2 MSU=1000
```

The response on the console is:

```
CPO1280I Defined capacity for system SYS1 in sysplex PLEX2 increased to 1000 MSU
```

ACTIVATE GROUPCAPACITY

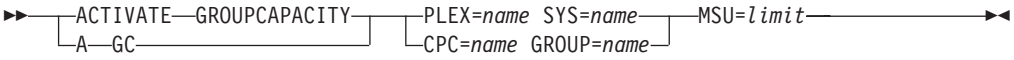
Use the **ACTIVATE GROUPCAPACITY** command to manually increase the Group Capacity for a capacity group. You can identify the group either by the name of the group and the name of the CPC on which the group is defined, or by the name of the z/OS system and sysplex running in an LPAR that is connected to the group. In the second case, the z/OS system needs to be defined in the active domain configuration. In every case the respective CPC has to be defined in the active domain configuration.

Note: The number that is supplied as the MSU parameter is the total limit to which the Group Capacity should be increased, not the number of MSU by which you want to increase the limit.

Before performing the activation, the Provisioning Manager checks for the following conditions:

- that the specified group is uniquely identified,
- that at the HMC Group Capacity is turned on for the group,
- that the target limit is an increase, meaning that the new Group Capacity is higher than the current Group Capacity.

Syntax



Parameters

The command has the following parameter:

PLEX=*name*
The name of the sysplex to which the specified system belongs.

SYS=*name*
The name of the system running in an LPAR belonging to the group for which you want to increase the Group Capacity.

CPC=*name*
The name of the CPC on which to find the specified capacity group.

GROUP=*name*
The name of a capacity group for which you want to increase the Group Capacity. An LPAR must belong to the specified group.

MSU=*limit*
The new Group Capacity for the specified capacity group. The limit is specified in MSU. The new value must be higher than the current Group Capacity and Group Capacity must be turned on for the group at the HMC. A maximum value of 100000 MSU can be specified.

Example

To increase the Group Capacity for group GROUP1 on CPC CPC1 to 1000 MSU, issue the following command:

```
MODIFY CPOSERV,APPL=ACTIVATE GROUPCAPACITY GROUP=GROUP1 CPC=CPC1 MSU=1000
```

or:

```
F CPOSERV,APPL=A GC GROUP=GROUP1 CPC=CPC1 MSU=1000
```

The response on the console is:

```
CPO1298I Group capacity for group GROUP1 on CPC CPC1 increased to 1000 MSU
```

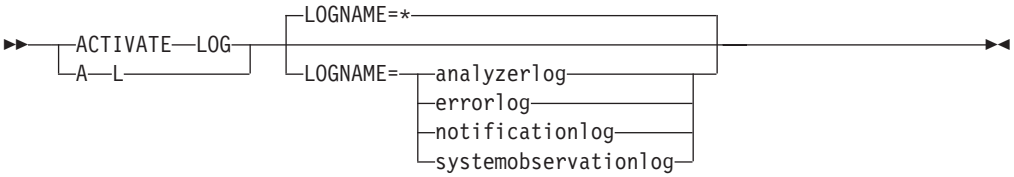
ACTIVATE LOG

Use the **ACTIVATE LOG** command to start recording log data to a file. Log data is always collected in memory. If the log is activated, whenever the log buffer is full, the buffer is written to a file. The default location for this file is the `/tmp` directory. This location can be overridden as described in "Planning the domain setup" on page 47. You must ensure that there is sufficient space in the active location.

The data is written until the log is deactivated or the Provisioning Manager terminates. See "DEACTIVATE LOG" on page 118. When the Provisioning Manager is started for the first time, all logs are in their default state. The default state for the error log is activated. For all other logs, it is deactivated.

Note: This command is to be used only if instructed to do so by IBM service personnel.

Syntax



Parameters

The command has the following parameter:

LOGNAME=*name*
The type of data to be recorded:

analyzerlog
Workload analysis results

errorlog
Unexpected errors detected during processing

notificationlog
Communications with the Provisioning Manager

systemobservationlog
Monitored metrics from the observed systems

You can also specify `LOGNAME=*` to indicate that writing is activated for all logs. It is the default if you omit the **LOGNAME** parameter; *name* is not case-sensitive.

Example

To record communications with the Provisioning Manager, issue the following command:

```
MODIFY CPOSERV,APPL=ACTIVATE LOG LOGNAME=NotificationLog
```

or:

```
F CPOSERV,APPL=A L LOGNAME=NotificationLog
```

The response on the console is:

```
CPO1031I Logging successfully activated for log NotificationLog
```

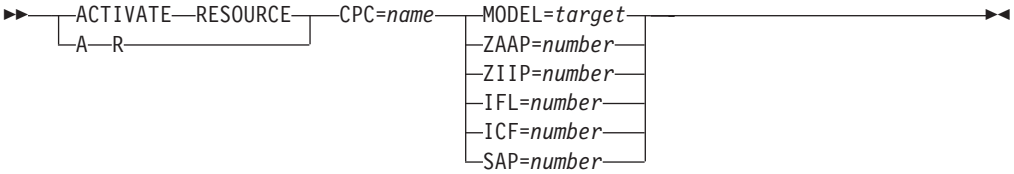
ACTIVATE RESOURCE

Use the **ACTIVATE RESOURCE** command to manually activate temporary capacity of a CPC in the provisioning domain. The additional capacity can be activated only for the On/Off CoD record that is identified in the domain configuration.

Note: The number that is supplied as a parameter is the total number of resources that you want to have active, not the number of resources that you want to add to the already active resources.

Before performing the activation, the Provisioning Manager checks that the specified resource level is an activation, that is it has more capacity, and that the target record allows the requested change. The resources are not managed by the Provisioning Manager and they are not deactivated automatically.

Syntax



Parameters

The command has the following parameters:

CPC=*name*
The name of the CPC on which you want to activate the additional capacity. The specified CPC must be part of the active domain configuration. The CPC must be at the supported hardware level, there must be an On/Off CoD record on the CPC that is managed by the Provisioning Manager, and this record must allow for the requested activation level.

MODEL=*target*
The model capacity identifier that you want to make active on the CPC. The target model must have more general purpose capacity measured in MSU than the current model. The On/Off CoD record managed by the Provisioning Manager must also allow for the capacity that is defined by the requested model.

ZAAP=*number*
The number of temporary zAAP processors that you want to have active. The target number must be higher than the number of active zAAP processors, and is limited by the number of spare processors on the machine, the maximum number of zAAP processors allowed by the On/Off CoD record, and the remaining capacity for zAAP processors of this record.

ZIIP=*number*
The number of temporary zIIP processors that you want to have active. The target number must be higher than the number of active zIIP processors, and is limited by the number of spare processors on the machine, the maximum number of zIIP processors allowed by the On/Off CoD record, and the remaining capacity for zIIP processors of this record.

IFL=*number*
The number of temporary IFL processors that you want to have active. The target number must be higher than the number of active IFL processors, and is limited by the number of spare processors on the machine, the maximum number of IFL processors allowed by the On/Off CoD record, and the remaining capacity for IFL processors of this record.

ICF=*number*
The number of temporary ICF processors that you want to have active. The target number must be higher than the number of active ICF processors, and is limited by the number of spare processors on the machine, the maximum number of ICF processors allowed by the On/Off CoD record, and the remaining capacity for ICF processors of this record.

SAP=*number*
The number of temporary SAP processors that you want to have active. The target number must be higher than the number of active SAP processors, and is limited by the number of spare processors on the machine, the maximum number of SAP processors allowed by the On/Off CoD record, and the remaining capacity for SAP processors of this record.

Example

To activate one zAAP for CPC G14, issue the following command:

```
MODIFY CPOSERV,APPL=ACTIVATE RESOURCE CPC=G14 ZAAP=1
```

or:

```
F CPOSERV,APPL=A R CPC=G14 ZAAP=1
```

The response on the console is:

```
CPO1026I Activation level change to 1 zAAPs successfully initiated for CPC G14
```

DEACTIVATE DEFINEDCAPACITY

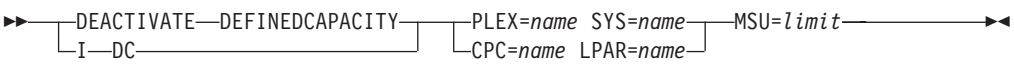
Use the **DEACTIVATE DEFINEDCAPACITY** command to manually decrease the Defined Capacity for an LPAR. You can identify the LPAR either by the name of the LPAR and the name of the CPC on which the LPAR is active, or by the name of the z/OS system and sysplex running in the LPAR. In the second case, the z/OS system needs to be defined in the active domain configuration. In every case the respective CPC has to be defined in the active domain configuration.

Note: The number that is supplied as the MSU parameter is the total limit to which the Defined Capacity should be decreased, not the number of MSU by which you want to decrease the limit.

Before performing the activation, the Provisioning Manager checks for the following conditions:

- That the specified LPAR is uniquely identified
- That at the HMC Defined Capacity is turned on for the LPAR
- That the target limit is a decrease, meaning that the new Defined Capacity is lower than the current Defined Capacity

Syntax



Parameters

The command has the following parameter:

PLEX=*name*
The name of the sysplex to which the specified system belongs.

SYS=*name*
The name of the system for which LPAR you want to decrease the Defined Capacity. The specified system must be currently observed.

CPC=*name*
The name of the CPC on which to find the specified LPAR.

LPAR=*name*
The name of an LPAR for which you want to decrease the Defined Capacity.

MSU=*limit*
The new Defined Capacity for the specified LPAR. The limit is specified in MSU. The new value must be lower than the current Defined Capacity and Defined Capacity must be turned on for the LPAR at the HMC. A value between 1 and 100000 MSU can be specified.

Example

To decrease the Defined Capacity for system SYS1 in sysplex PLEX2 to 500 MSU, issue the following command:

```
MODIFY CPOSERV,APPL=DEACTIVATE DEFINEDCAPACITY SYS=SYS1 PLEX=PLEX2 MSU=500
```

or:

```
F CPOSERV,APPL=I DC SYS=SYS1 PLEX=PLEX2 MSU=500
```

The response on the console is:

```
CPO1311I Defined capacity for system SYS1 in sysplex PLEX2 decreased to 500 MSU
```

DEACTIVATE GROUPCAPACITY

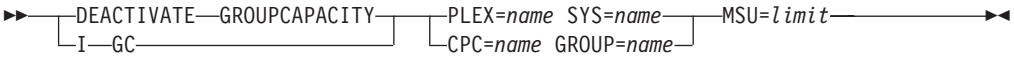
Use the **DEACTIVATE GROUPCAPACITY** command to manually decrease the Group Capacity for a capacity group. You can identify the group by the name of the group and the name of the CPC on which the group is defined, or by the name of the z/OS system and sysplex running in an LPAR that belongs to the group. In the second case, the z/OS system needs to be defined in the active domain configuration. In every case the respective CPC has to be defined in the active domain configuration.

Note: The number that is supplied as MSU parameter is the total limit to which the Group Capacity should be decreased, not the number of MSU by which you want to decrease the limit.

Before performing the deactivation, the Provisioning Manager checks for the following conditions:

- That the specified group is uniquely identified
- That at the HMC Group Capacity is turned on for the group
- That the target limit is a decrease, meaning that the new Group Capacity is lower than the current Group Capacity

Syntax



Parameters

The command has the following parameter:

PLEX=*name*
The name of the sysplex to which the specified system belongs.

SYS=*name*
The name of the system running in an LPAR belonging to the group for which you want to decrease the Group Capacity.

CPC=*name*
The name of the CPC on which to find the specified capacity group.

GROUP=*name*
The name of a capacity group for which you want to decrease the Group Capacity. An LPAR must belong to the specified group.

MSU=*limit*
The new value for the Group Capacity for the specified capacity group. The limit is specified in MSU. The new value must be lower than the current Group Capacity on the group and Group Capacity must be turned on for the group at the HMC. A value between 1 and 100000 MSU can be specified.

Example

To decrease the Group Capacity for group GROUP1 on CPC CPC1 to 500 MSU, issue the following command:

```
MODIFY CPOSERV,APPL=DEACTIVATE GROUPCAPACITY GROUP=GROUP1 CPC=CPC1 MSU=500
```

or:

```
F CPOSERV,APPL=I GC GROUP=GROUP1 CPC=CPC1 MSU=500
```

The response on the console is:

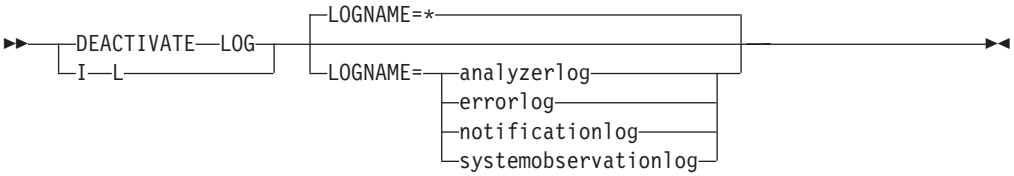
```
CPO1312I Group capacity for group GROUP1 on CPC CPC1 decreased to 500 MSU
```

DEACTIVATE LOG

Use the **DEACTIVATE LOG** command to stop recording log data to a file. Even though the data is no longer recorded in a file, it continues to be collected in memory. If the log was active, the current log buffer is written to the file. The default location for this file is the `/tmp` directory. This location can be overridden as described in "Planning the domain setup" on page 47.

Note: This command should be used only if instructed to do so by IBM service personnel.

Syntax



Parameters

The command has the following parameter:

LOGNAME=*name*
The type of data to stop recording:

analyzerlog
Contains workload analysis results

errorlog
Contains unexpected errors that are detected during processing

notificationlog
Contains communications with the Provisioning Manager

systemobservationlog
Contains monitored metrics from the observed systems

You can also specify `LOGNAME=*` to indicate that writing be deactivated for all logs. If you omit the **LOGNAME** parameter, the default is that writing is deactivated for all logs. *Name* is not case-sensitive.

Example

To stop recording monitored metrics for the observed systems, issue the following command:

```
MODIFY CPOSERV,APPL=DEACTIVATE LOG LOGNAME=NotificationLog
```

or:

```
F CPOSERV,APPL=I L LOGNAME=NotificationLog
```

The response on the console is:

```
CPO1031I Logging successfully deactivated for log NotificationLog
CPO2030I Log information written to file
                   /tmp/cpoNotificationLog_20071024131732469.log
```

DEACTIVATE RESOURCE

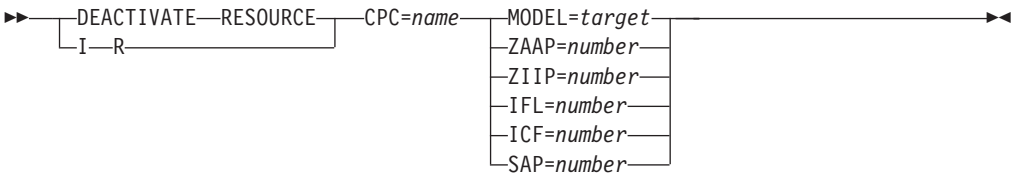
Use the **DEACTIVATE RESOURCE** command to manually deactivate temporary capacity of a CPC in the provisioning domain. The additional capacity can be deactivated only for the On/Off CoD record identified in the domain configuration.

Use this command to deactivate resources that you activated manually, for example, by using the **activate resource** command; see "ACTIVATE RESOURCE" on page 112).

Note: The number that is supplied as a parameter represents the total number of resources that you want to keep active, not the number of resources that you want to remove from resources that are already active.

Before deactivating, the Provisioning Manager checks that the specified resource level is below the currently active resources within the managed record.

Syntax



Parameters

The command has the following parameters:

CPC=*name*
The name of the CPC on which you want to deactivate the temporary capacity. The specified CPC must be part of the active domain configuration. The CPC must be at the supported hardware level and there must be an On/Off CoD record on the CPC that is managed by the Provisioning Manager.

MODEL=*target*
The model capacity identifier that you want to keep active on the CPC. The target model must have less general purpose capacity in terms of MSU than the current model. The On/Off CoD record managed by the Provisioning Manager must also allow for the requested model.

ZAAP=*number*
The number of temporary zAAP processors that you want to keep active. The target number must be less than the current number of active zAAP processors for this record.

ZIIP=*number*
The number of temporary zIIP processors that you want to keep active. The target number must be less than the current number of active zIIP processors for this record.

IFL=*number*
The number of temporary IFL processors that you want to keep active. The target number must be less than the current number of active IFL processors for this record.

ICF=*number*
the number of temporary ICF processors that you want to keep active. The target number must be less than the current number of active ICF processors for this record.

SAP=*number*
the number of temporary SAP processors that you want to keep active. The target number must be less than the current number of active SAP processors for this record.

Example

To deactivate all zAAPs for CPC G14, issue the following command:

```
MODIFY CPOSERV,APPL=DEACTIVATE RESOURCE CPC=G14 ZAAP=0
```

or:

```
F CPOSERV,APPL=I R CPC=G14 ZAAP=0
```

The response on the console is:

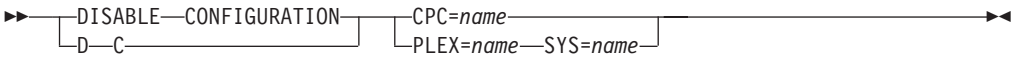
```
CPO1028I Activation level change to 0 zAAPs successfully initiated for CPC G14
```

DISABLE CONFIGURATION

Use the **DISABLE CONFIGURATION** command to disable a CPC or a system within the active domain configuration. If a system is disabled, the Provisioning Manager no longer observes the system. If a CPC is disabled, the Provisioning Manager no longer considers it for activation and deactivation of additional capacity.

If the command completes successfully, the selected CPC or system is listed as disabled in the domain configuration report.

Syntax



Parameters

The command has the following parameters:

CPC=*name*
The name of the CPC that you want to disable. The specified CPC must be part of the active domain configuration.

PLEX=*name*
The name of the sysplex to which the specified system belongs.

SYS=*name*
The name of the system that you want to disable. The specified system must be part of the active domain configuration.

Example

To disable CPC G14, issue the following command:

```
MODIFY CPOSERV,APPL=DISABLE CONFIGURATION CPC=G14
```

or:

```
F CPOSERV,APPL=D C CPC=G14
```

The response on the console is:

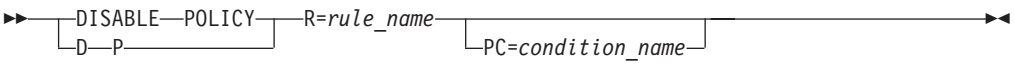
```
CPO1012I CPC G14 in current configuration successfully disabled
```

DISABLE POLICY

Use the **DISABLE POLICY** command to disable an entire provisioning rule or a specific provisioning condition within a provisioning rule in the active provisioning policy. If capacity is already provisioned because of affected time conditions, this capacity is deprovisioned.

If the command completes successfully, the selected rule or provisioning condition is listed as disabled in the policy report.

Syntax



Parameters

The command has the following parameters:

R=*rule_name*
The name of the rule that you want to disable. The specified rule must exist in the active policy. If the name contains lowercase characters, your console might require that the name be enclosed in quotation marks.

PC=*condition_name*
The name of the provisioning condition that you want to disable. A provisioning condition with the specified name must be part of the *rule_name* specified. If the name contains lowercase characters, your console might require that the name be enclosed in quotation marks.

Example

To disable rule CP160106R, issue the following command:

```
MODIFY CPOSERV,APPL=DISABLE POLICY R=CP160106R
```

or:

```
F CPOSERV,APPL=D P R=CP160106R
```

The response on the console is:

```
CPO1003I Rule CP160106R successfully disabled
```

DISABLE POWERSAVE

Use the **DISABLE POWERSAVE** command to initiate the transition of a CPC out of static power save mode. Only zEnterprise Systems and later support static power save mode. Disabling static power save resumes nominal capacity of the CPC model. The Provisioning Manager considers the CPC for activation of additional temporary capacity.

Transition into or out of power save mode takes some time. A subsequent Provisioning Manager message informs you when the operation is complete.

Syntax

Parameters

The command has the following parameters:

CPC=*name*
The name of the CPC on which you want to disable static power save. The specified CPC must be part of the active domain configuration.

Example

To disable the static power save function for CPC G14, issue the following command:

```
MODIFY CPOSERV,APPL=DISABLE POWERSAVE CPC=G14
```

or:

```
F CPOSERV,APPL=D PS CPC=G14
```

The response on the console is:

```
CPO1093I Static power save mode for CPC G14 successfully disabled
```

DUMP MANAGER

Use the **DUMP MANAGER** command to request dumps from the Provisioning Manager. Different types of dumps can be requested:

- Heap dumps
- Java dumps
- Java system dumps

The dumps are created in the home directory of the Provisioning Manager user.

Note: This command is to be used only if instructed to do so by IBM service personnel.

Syntax

Parameters

The command has the following parameter:

TYPE=*type*
The type of dump. The following types are supported:

HEAP
Requests a Java heap dump

JAVA
Requests a Java dump

SYSTEM
Requests a Java system dump

The type values are not case-sensitive.

Example

To perform a snap dump, issue the following command:

```
MODIFY CPOSERV,APPL=DUMP MANAGER TYPE=HEAP
```

The response on the console is:

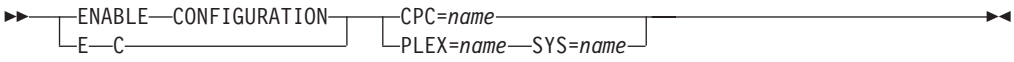
```
CPO1088I Dump manager command for dump type HEAP successfully performed
```

ENABLE CONFIGURATION

Use the **ENABLE CONFIGURATION** command to enable a CPC or a system within the active domain configuration. If a system is enabled, the Provisioning Manager starts to observe it and capacity is provisioned based on the workload situation of this system. If a CPC is enabled, the Provisioning Manager considers it for provisioning and deprovisioning of additional capacity.

If the command is successful, the selected CPC or system is listed as enabled in the domain configuration report.

Syntax



Parameters

The command has the following parameters:

CPC=*name*
The name of the CPC that you want to enable. The specified CPC must be part of the active domain configuration.

PLEX=*name*
The name of the sysplex to which the specified system belongs..

SYS=*name*
The name of the system that you want to enable. The specified system must be part of the active domain configuration.

Example

To enable CPC G14, issue the following command:

```
MODIFY CPOSERV,APPL=ENABLE CONFIGURATION CPC=G14
```

or:

```
F CPOSERV,APPL=E C CPC=G14
```

The response on the console is:

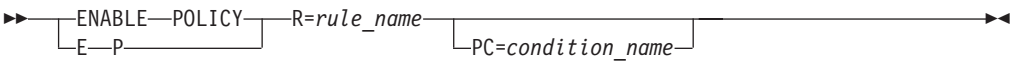
```
CPO1011I CPC G14 in current configuration successfully enabled
```

ENABLE POLICY

Use the **ENABLE POLICY** command to enable an entire provisioning rule or a specific provisioning condition within a provisioning rule in the active provisioning policy. If time conditions are enabled by this command, additional capacity can be provisioned.

If the command is successful, the selected rule or provisioning condition is listed as enabled in the policy report.

Syntax



Parameters

The command has the following parameters:

R=*rule_name*
The name of the rule that you want to enable. The specified rule must exist in the active policy. If the name contains lowercase characters, then your console might require that the name be enclosed in quotation marks.

PC=*condition_name*
The name of the provisioning condition that you want to enable. A provisioning condition with the specified name must be part of the *rule_name* specified. If the name contains lowercase characters, then your console might require that the name be enclosed in quotation marks.

Example

To enable rule CP160106R, issue the following command:

```
MODIFY CPOSERV,APPL=ENABLE POLICY R=CP160106R
```

or:

```
F CPOSERV,APPL=E P R=CP160106R
```

The response on the console is:

```
CPO1001I Rule CP160106R successfully enabled
```

ENABLE POWERSAVE

Use the **ENABLE POWERSAVE** command to initiate the transition of a CPC into static power save mode. Only zEnterprise Systems and later support static power save mode. Because static power save mode reduces the total capacity of a CPC, the Provisioning Manager no longer considers the CPC for activation of additional temporary capacity.

The CPC must be configured to allow for a certain command in order for it to complete successfully. For more information about how to set up a CPC for static power save mode, see the appropriate hardware documentation.

Transition into or out of power save mode takes some time. A Provisioning Manager message informs when the operation is complete.

Syntax

Parameters

The command has the following parameters:

CPC=*name*
The name of the CPC for which you want to enable static power save. A CPC of this name must be part of the active domain configuration.

Example

To enable static power save function for CPC G14, issue the following command:

```
MODIFY CPOSERV,APPL=ENABLE POWERSAVE CPC=G14
```

or:

```
F CPOSERV,APPL=E PS CPC=G14
```

The response on the console is:

```
CPO1092I Static power save mode for CPC G14 successfully enabled
```

LIST CONFIGURATION

Use the **LIST CONFIGURATION** command to list the entries in the domain configuration repository for the domain.

Syntax

Parameters

None.

Example

To list the domain configuration, issue the following command:

```
MODIFY CPOSERV,APPL=LIST CONFIGURATION
```

or:

```
F CPOSERV,APPL=L C
```

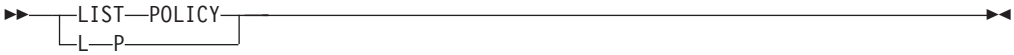
The response on the console might be:

```
CPO1049I Domain configuration list generated at 01/17/2006 02:14:08
  DS
```

LIST POLICY

Use the **LIST POLICY** command to list the entries in the policy repository of the domain.

Syntax



Parameters

None.

Example

To list the available policies, issue the following command:

```
MODIFY CPOSERV,APPL=LIST POLICY
```

or:

```
F CPOSERV,APPL=L P
```

The response on the console might be:

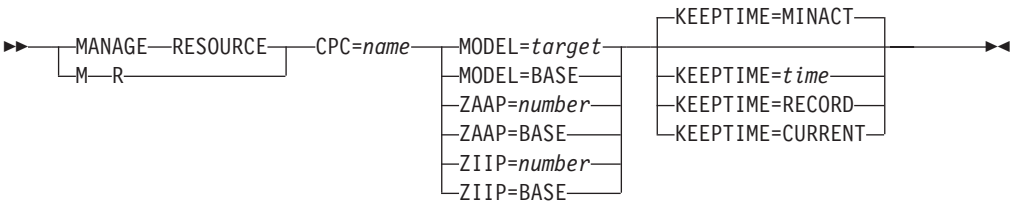
```
CPO1048I Policy list generated at 01/17/2006 02:38:39
  BERLINP
```

MANAGE RESOURCE

Use the **MANAGE RESOURCE** command to pass manually activated temporary capacity to the Provisioning Manager for deactivation according to the workload situation. Capacity may be activated proactively or because the capacity authorized by the policy is insufficient. Such capacity can be managed according to the policy by the Provisioning Manager. You need to specify the target up to which the Provisioning Manager is allowed to deactivate the capacity.

With the information about the resources to manage, you pass the information how long the current configuration should be at least held active. After that time the Provisioning Manager may start to deactivate the capacity if, according to the policy and workload situation, the capacity is no longer needed.

Syntax



Parameters

The command has the following parameters:

CPC
The name of the CPC for which you want to pass manually activated temporary capacity to the Provisioning Manager for deactivation. The specified CPC must be part of the active domain configuration. There should be temporary capacity active on the On/Off CoD record on the CPC that is not managed by the Provisioning Manager.

Model=
The model capacity identifier to which the Provisioning Manager can deactivate general purpose capacity on the specified CPC.

target
The model capacity identifier that you want to keep active on the CPC. The target model should have less capacity levels and number of general purpose processors than the base model used for the policy based management. The On/Off CoD record managed by the Provisioning Manager must also allow for the requested model.

BASE
All general purpose capacity of the On/Off CoD record that is currently active will be managed by the Provisioning Manager.

ZAAP=
The number of temporary zAAP processors to which the Provisioning Manager can deactivate zAAP capacity on the specified CPC.

number
The number of temporary zAAP processors that you want to keep active. The target number should be lower than the base number of zAAP processors used for the policy based management. If you specify a target

number of 0, then all temporary zAAP capacity of the defined On/Off CoD record will be managed by the Provisioning Manager.

BASE
All temporary zAAP capacity of the defined On/Off CoD record will be managed by the Provisioning Manager.

ZIIP=
The number of temporary zIIP processors to which the Provisioning Manager can deactivate zIIP capacity on the specified CPC.

number
The number of temporary zIIP processors that you want to keep active. The target number should be lower than the base number of zIIP processors used for the policy based management. If you specify a target number of 0, then all temporary zIIP capacity of the defined On/Off CoD record will be managed by the Provisioning Manager.

BASE
All temporary zIIP capacity of the defined On/Off CoD record will be managed by the Provisioning Manager.

KEEPTIME
The minimum time for which the passed capacity should be held active. You can specify a time in minutes or one of the following special values

MINACT
Use the minimum activation time as specified in the parameters for the Provisioning Manager. You set this value as shown in "Adapting the Provisioning Manager parameters" on page 52.

time
Use the specified time, which must be in the range of 1 to 1440 minutes.

CURRENT
If there are already resources managed by the Provisioning Manager, keep the current time until the resources are deactivated. In case that there are not yet any resources managed by the Provisioning Manager, the minimum activation time as specified in the parameters for the Provisioning Manager management of resources is used.

RECORD
Leave the capacity active until the current activation period of the On/Off CoD record ends. An activation period usually ends 24 hours after the activation of the currently active capacity. In contrast to all other values, the Provisioning Manager will deactivate all capacity at the end of that time if the workload situation allows it.

Note: The value is used for all capacity managed by the Provisioning Manager. It overrides the minimum activation time for all previously activated resources. Later policy based activations of resource may override this time with the minimum activation time starting at the time of that activation.

Example

To let the Provisioning Manager manage all general purpose capacity of CPC G14, issue the following command:

```
MODIFY CPOSERV,APPL=MANAGE RESOURCE CPC=G14 MODEL=BASE
```

or:

```
F CPOSERV, APPL=M R CPC=G14 MODEL=BASE
```

The response on the console is:

```
CPO4407I Management for CPC G14 to model 721 started.
         Managed resources remain active for at least 30 minutes
```

More examples

On CPC G14 you have 3 active zAAPs, 1 zAAP of it is managed by the Provisioning Manager and 2 were activated manually. To pass manually activated zAAP processors to the Provisioning Manager for deactivation, issue the following commands:

MODIFY CPOSERV, APPL=MANAGE RESOURCE CPC=G14 zAAP=0
: All manually activated zAAPs are passed to the provisioning manager. No zAAP is active after deprovisioning.

MODIFY CPOSERV, APPL=MANAGE RESOURCE CPC=G14 zAAP=1
: One manually activated zAAP is passed to the provisioning manager. 1 zAAP is active after deprovisioning.

MODIFY CPOSERV, APPL=MANAGE RESOURCE CPC=G14 zAAP=2
: An error message is issued.

MODIFY CPOSERV, APPL=MANAGE RESOURCE CPC=G14 zAAP=2 KEEPTIME= 120
: None of the manually activated zAAPs is passed to the Provisioning Manager. 2 zAAPs are active after deprovisioning. The zAAP managed by the Provisioning Manager is held active for 120 minutes starting from now.

MODIFY CPOSERV, APPL=MANAGE RESOURCE CPC=G14 zAAP=3
: An error message is issued.

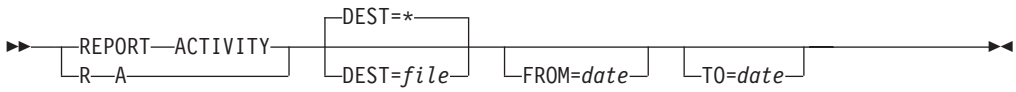
REPORT ACTIVITY

Use the **REPORT ACTIVITY** command to display the activation and deactivation operations that have been performed by the Provisioning Manager. Information about activations and deactivations is retained for approximately two months. You can select the time period to be reported.

For information about what is contained in the report, see "The activity report" on page 87.

Note: Manual activations and deactivations are not recorded by the Provisioning Manager in the activity log.

Syntax



Parameters

The command has the following parameters:

DEST=
The destination of the activity report. The following destinations are supported:

***** (Default) specifies that the report should be written to the console that is issuing the command.

file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute path or a relative path to the file. If this is not absolute, the path is relative to the home directory of the user who is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

FROM=*date*
The start date of the report. All activations and deactivations executed from the beginning of that day are reported. If the parameter is omitted, the start date of the log is used. The start date of the report must be before the end date of the report specified in the **TO** parameter.

The format for the start date is *mm/dd/yyyy*, where *mm* is the month, *dd* is the day, and *yyyy* is the year. For example April 20, 2006, is `4/20/2006`.

TO=*date*
The end date of the report. All activations and deactivations executed before the end of that day are reported. If the end date is omitted, the current date is used. The end date of the report must be after the start date of the report specified in the **FROM** parameter.

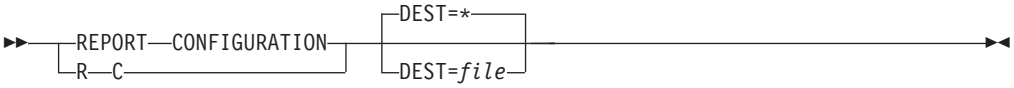
The format for the end date is the same as the one for the start date.

REPORT CONFIGURATION

Use the **REPORT CONFIGURATION** command to display the active domain configuration.

For information about what is contained in the report, see "The domain configuration report" on page 81.

Syntax



Parameters

The command has the following parameter:

DEST=
The destination of the domain configuration report. The following destinations are supported:

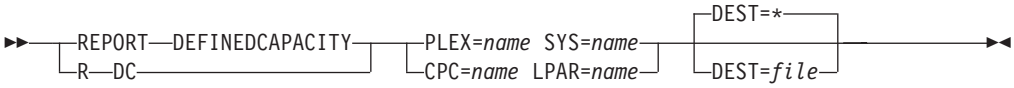
***** (Default) specifies that the report should be written to the console that is issuing the command.

file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute path or a relative path to the file. If this is not absolute, the path is relative to the home directory of the user who is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT DEFINEDCAPACITY

Use the **REPORT DEFINEDCAPACITY** command to display the current Defined Capacity management data for an LPAR. You can identify the LPAR either by the name of the z/OS system and sysplex running in the LPAR, or by the name of the LPAR and the name of the CPC on which the LPAR is active. The specified system must be part of the active domain configuration and the system must be currently observed. The specified CPC must be part of the active domain configuration. For information about what is contained in the report, see "The defined capacity report" on page 92.

Syntax



Parameters

The command has the following parameter:

PLEX=*name*
The name of the sysplex to which the specified system belongs.

SYS=*name*
The name of the system for which LPAR you want to display Defined Capacity management data. At the HMC Defined Capacity must be turned on for the LPAR.

CPC=*name*
The name of the CPC on which to find the specified LPAR.

LPAR=*name*
The name of an LPAR for which you want to display the Defined Capacity management data. At the HMC Defined Capacity must be turned on for the LPAR and the system running in this LPAR must be currently observed.

DEST=
The destination of the defined capacity report. The following destinations are supported:

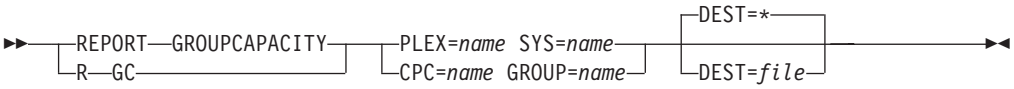
***** (Default) specifies that the report should be written to the console issuing the command.

file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user that is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT GROUPCAPACITY

Use the **REPORT GROUPCAPACITY** command to display the current Group Capacity management data for a capacity group. You can identify the capacity group either by the name of the z/OS system and sysplex running in an LPAR that belongs to the group, or by the name of the group and the name of the CPC on which the group is defined. The specified system must be part of the active domain configuration and the system must be currently observed. The specified CPC must be part of the active domain configuration. For information about what is contained in the report, see "The group capacity report" on page 94.

Syntax



Parameters

The command has the following parameter:

PLEX=*name*
The name of the sysplex to which the specified system belongs.

SYS=*name*
The name of the system running in an LPAR that belongs to the capacity group for which you want to display Group Capacity management data. Group Capacity must be turned on for the group at the HMC.

CPC=*name*
The name of the CPC on which to find the specified capacity group.

GROUP=*name*
The name of a capacity group for which you want to display the Group Capacity management data. Group Capacity must be turned on for the group at the HMC and at least one system running in an LPAR belonging to this capacity group must be currently observed.

DEST=
The destination of the group capacity report. The following destinations are supported:

***** (Default) specifies that the report should be written to the console issuing the command.

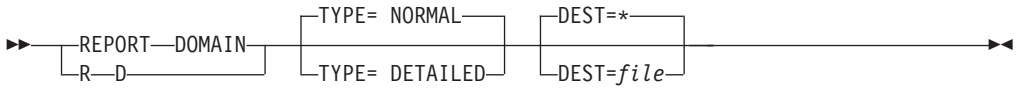
file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user that is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT DOMAIN

Use the **REPORT DOMAIN** command to display the status of the current domain.

For information about what is contained in the report, see "The domain report" on page 77.

Syntax



Parameters

The command has the following parameters:

TYPE=*type*
The type of report to display. The following types are supported:

NORMAL
Displays the domain report with main configuration information.

DETAILED
Displays the domain report including a list of active configuration parameters that was set in the PARM member.

DEST=
The destination of the domain report. The following destinations are supported:

***** (Default) specifies that the report should be written to the console issuing the command.

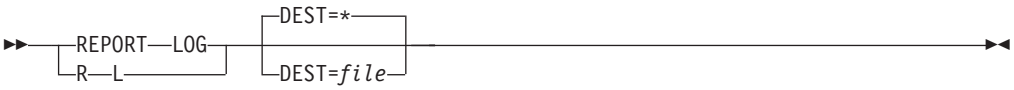
file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user who is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT LOG

Use the **REPORT LOG** command to display the status of the current log write settings.

For information about what is contained in the report, see "The log report" on page 98.

Syntax



Parameters

The command has the following parameter:

DEST=
The destination of the log report. The following destinations are supported:

***** (Default) specifies that the report should be written to the console issuing the command.

file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user who is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT POLICY

Use the **REPORT POLICY** command to display the active provisioning policy.

For information about what is contained in the report, see "The policy report" on page 78.

Syntax

Parameters

The command has the following parameter:

DEST=
The destination of the policy report. The following destinations are supported:

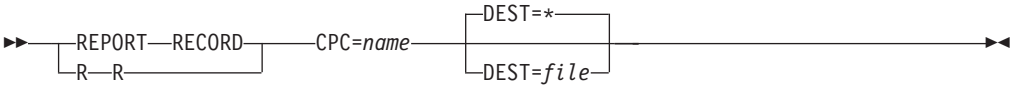
***** (Default) specifies that the report should be written to the console issuing the command.

file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user who is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT RECORD

Use the **REPORT RECORD** command to display information related to the On/Off CoD record of a CPC managed by the Provisioning Manager. For information about what is contained in the report, see "The record report" on page 89.

Syntax



Parameters

The command has the following parameter:

CPC=*name*
The name of the CPC for which you want to report the record information. The CPC of this name must be part of the active domain configuration.

DEST=
The destination of the record report. The following destinations are supported:

***** (Default) specifies that the report should be written to the console issuing the command.

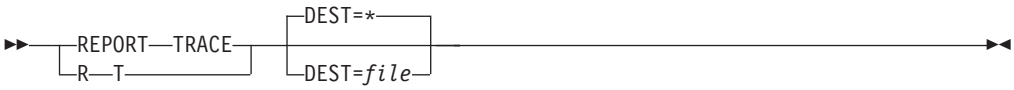
file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user that is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT TRACE

Use the **REPORT TRACE** command to display the active trace configuration.

For information about what is contained in the report, see "The trace report" on page 97.

Syntax



Parameters

The command has the following parameter:

DEST=
The destination of the trace report. The following destinations are supported:

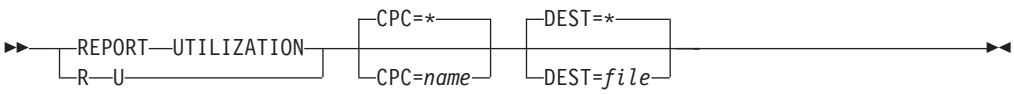
***** (Default) specifies that the report should be written to the console issuing the command.

file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user that is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT UTILIZATION

Use the **REPORT UTILIZATION** command to display the physical processor utilization of CPCs that are currently being monitored on observed systems. Only CPCs affected by currently active utilization conditions will be reported. For information about what is contained in the report, see "The utilization report" on page 86.

Syntax



Parameters

The command has the following parameter:

CPC=
The name of the CPC whose physical processor utilization you want to list exclusively. The specified CPC and at least one system running on the CPC must be part of the active domain configuration.

DEST=

The destination of the utilization report. The following destinations are supported:

***** (Default) specifies that the report should be written to the console issuing the command.

file

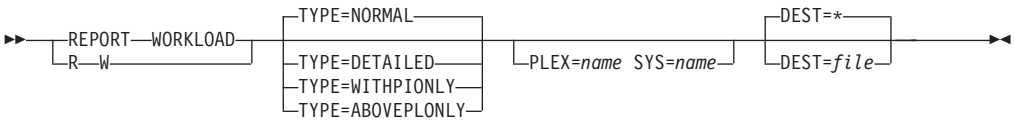
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user that is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

REPORT WORKLOAD

Use the **REPORT WORKLOAD** command to display the WLM service class periods that are currently being monitored on the observed systems.

For information about what is contained in the report, see "The workload report" on page 85.

Syntax



Parameters

The command has the following parameters:

TYPE=*type*
The type of report to display. The following types are supported:

NORMAL
Displays a short workload report

DETAILED
Displays a workload report that includes detailed information about WLM service class periods that are suffering. For each service class period, the report lists which types of additional capacity are needed, or why additional capacity is not needed.

WITHPIONLY
Displays a detailed workload report that lists workloads with a recent PI measurement only.

ABOVEPLONLY
Displays a detailed workload report that lists only workloads with a PI that is above the corresponding provisioning limit (PL), as specified in the provisioning policy.

PLEX=*name*
The name of the sysplex to which the specified system belongs.

SYS=*name*
The name of the system whose workload you want to list exclusively. The specified system must be part of the active domain configuration and it must also be currently observed.

DEST=
The destination of the workload report. The following destinations are supported:

***** (Default) specifies the report should be written to the console issuing the command.

file
Specifies a file where the information is to be stored. If the name includes lowercase characters this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute, path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user who is running the

Provisioning Manager. Make sure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location.

RESET CONFIGURATION

Use the **RESET CONFIGURATION** command to undo all configuration changes that have been issued since the domain configuration was activated.

Syntax

Parameters

None.

Example

To reset the configuration, issue the following command:

```
MODIFY CPOSERV,APPL=RESET CONFIGURATION
```

or:

```
F CPOSERV,APPL=T C
```

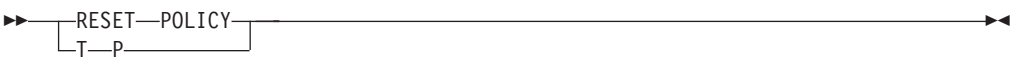
The response on the console might be:

```
CPO1019I Current configuration successfully reset
```

RESET POLICY

Use the **RESET POLICY** command to reset all elements in the active provisioning policy to their default states as defined in the policy at the time of activation.

Syntax



Parameters

None.

Example

To reset the current policy, issue the following command:

```
MODIFY CPOSERV,APPL=RESET POLICY
```

or:

```
F CPOSERV,APPL=T P
```

The response on the console might be:

```
CPO1018I Current policy successfully reset
```

RESET TRACE

Use the **RESET TRACE** command to reset the tracing parameters of the Provisioning Manager. The default trace level and the trace level for all trace components are reset to the defaults delivered with the product.

Note: This command is to be used only if instructed to do so by IBM service personnel.

Syntax

Parameters

None.

Example

To reset the current trace configuration issue the following command:

```
MODIFY CPOSERV,APPL=RESET TRACE
```

or:

```
F CPOSERV,APPL=T T
```

The response on the console might be:

```
CPO1070I The trace configuration has been reset
```

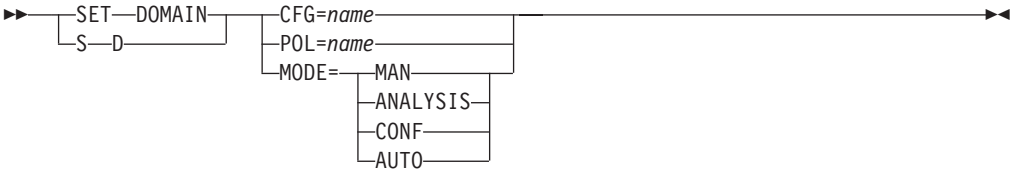
SET DOMAIN

Use the **SET DOMAIN** command to change the global processing information of the domain. This information includes the active provisioning policy, the active domain configuration, and the active processing mode.

The information supplied is validated before processing. If the command cannot be processed for any reason, the previous information remains active.

Note: If any additional capacity is active when you change a policy or configuration, this capacity might be deprovisioned.

Syntax



Parameters

The command has the following parameters:

CFG=*name*
The name of the domain configuration that you want to activate. A domain configuration with the specified name must be installed in the domain configuration repository of the Provisioning Manager. Before the domain configuration is activated, the content is validated.

POL=*name*
The name of the policy that you want to activate. A policy with the specified name must be installed in the policy repository of the Provisioning Manager. Before the policy is activated, the content is validated.

MODE=*mode*
The new processing mode for the Provisioning Manager. The following processing mode values are supported:

MAN
Manual mode (policies are disabled)

ANALYSIS
Analysis mode

CONF
Confirmation mode

AUTO
Autonomic mode

The processing mode values are not case-sensitive.

Example

To activate configuration DS, issue the following command:

```
MODIFY CPOSERV,APPL=SET DOMAIN CFG=DS
```

or:

```
F CPOSERV,APPL=S D CFG=DS
```

The response on the console might be:

```
CPO1041I Domain configuration DS successfully activated
```

To change the policy to BERLINP, issue the following command:

```
MODIFY CPOSERV,APPL=SET DOMAIN POL=BERLINP
```

or:

```
F CPOSERV,APPL=S D POL=BERLINP
```

The response on the console might be:

```
CPO1020I Policy successfully changed to BERLINP
```

To set the processing mode to autonomic, issue the following command:

```
MODIFY CPOSERV,APPL=SET DOMAIN MODE=AUTO
```

or:

```
F CPOSERV,APPL=S D MODE=AUTO
```

The response on the console might be:

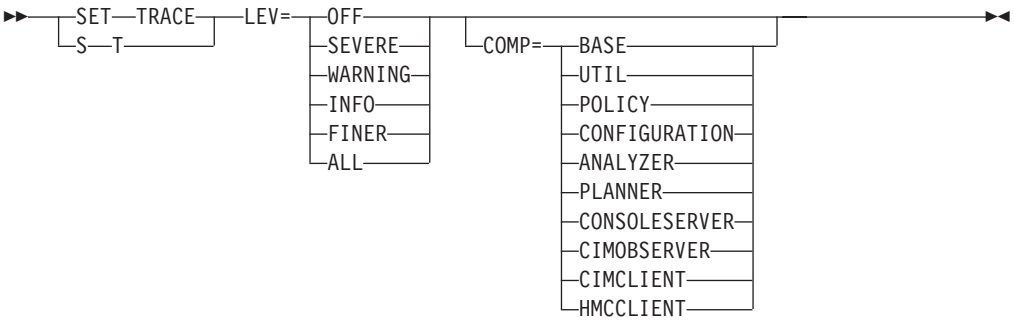
```
CPO1009I Processing mode successfully changed to AUTO
```

SET TRACE

Use the **SET TRACE** command to change the tracing parameters of the Provisioning Manager. You can set a default trace level and also separate trace levels for individual trace components. Setting the default trace level does not overwrite a trace level set for an individual component.

Note: This command is to be used only if instructed to do so by IBM service personnel.

Syntax



Parameters

The parameter names and trace values are not case-sensitive. This command has the following parameters:

LEV
Name of the trace level you want to activate. The following trace levels are supported:

OFF
Do not trace

SEVERE
Trace severe information

WARNING
Trace warnings or severe information

INFO
Trace general information

FINER
Trace at a finer information level

ALL
Trace all information

COMP
The name of a component of the Provisioning Manager for which you want to alter the trace level. The supported component names are:

BASE
Base processing

UTIL
Utility classes

POLICY
Policy-relevant processing

CONFIGURATION
Domain configuration-relevant processing

ANALYZER
Analysis of the metrics retrieved from the observed systems

PLANNER
Processing that is relevant to planning

CONSOLESERVER
Command processing

CIMOBSERVER
CIM requests processing

CIMCLIENT
CIM client API

HMCCLIENT
HMC or SE client API

If **COMP** is not specified, the overall trace level is changed.

Example

To set the global trace level to ALL, issue the following command:

```
MODIFY CPOSERV,APPL=SET TRACE LEV=ALL
```

or:

```
F CPOSERV,APPL=S T LEV=ALL
```

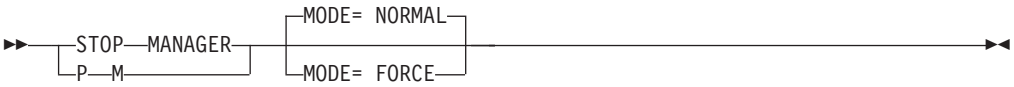
The response on the console might be:

```
CPO1060I The global trace level has been set to ALL
```

STOP MANAGER

Use the **STOP MANAGER** command to stop processing of the Provisioning Manager. After the Provisioning Manager is requested to stop, it does not accept any other commands. The state of resources does not change. Any additionally provisioned capacity remains active.

Syntax



Parameters

The command has the following parameter:

MODE
Mode in which you want to end the Provisioning Manager. There are two supported stop modes:

NORMAL
The default stop mode is to terminate all processing and to unregister the Provisioning Manager from ARM if applicable

FORCE
To stop processing immediately. If the Provisioning Manager is registered with ARM, the process restart.

The mode values are not case-sensitive.

Example

To stop the Provisioning Manager, issue the following command:

```
MODIFY CPOSERV,APPL=STOP MANAGER
```

or:

```
F CPOSERV,APPL=P M
```

The response on the console might be:

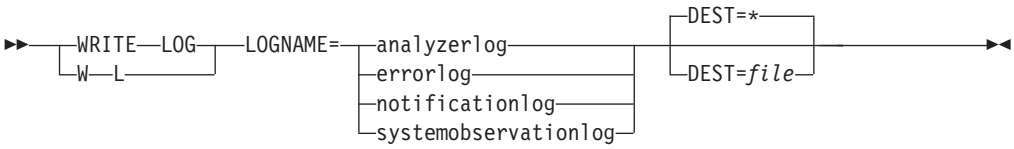
```
CPO1007I Stop command for the Provisioning Manager accepted
```

WRITE LOG

Use the **WRITE LOG** command to save the current data for a specific log to a file.

Note: This command is to be used only if instructed to do so by IBM service personnel.

Syntax



Parameters

The command has the following parameters:

LOGNAME=*name*
The type of data to be recorded might be:

analyzerlog
Workload analysis results

errorlog
Unexpected errors that are detected during processing

notificationlog
Communications within the Provisioning Manager

systemobservationlog
Monitored metrics from the observed systems

The log names are not case-sensitive.

DEST=
The destination of the log. The following destinations are supported:

***** (Default) specifies that the report be written to the standard log file in the log path specified during setup. This option can be used if logging is already active for the specified log and data is available in the log.

file
Specifies a file where the information is to be stored. If the name includes lowercase characters, this parameter must be enclosed in single quotation marks ('). The file must be in the hierarchical file system. You can specify either an absolute path or a relative path to the file. If this is not absolute the path is relative to the home directory of the user who is running the Provisioning Manager. You must ensure that there is sufficient space in the selected location. The Provisioning Manager user must be authorized to write to this location. This option can always be used and does not require that log writing be activated for the specified log.

Example

To write the notification log to file `nl.log`, issue the following command:

```
MODIFY CPOSERV,APPL=WRITE LOG LOGNAME=NotificationLog DEST='nl.log'
```

or:

```
F CPOSERV,APPL=W L LOGNAME=NotificationLog DEST='nl.log'
```

The response on the console might be:

```
CPO1040I Log NotificationLog successfully written to file nl.log
```

Part 4. Appendixes

Accessibility

Accessible publications for this product are offered through IBM Knowledge Center (http://www.ibm.com/support/knowledgecenter/SSLTBW/welcome).

If you experience difficulty with the accessibility of any z/OS information, send a detailed message to the "Contact us" web page for z/OS (http://www.ibm.com/systems/z/os/zos/webqs.html) or use the following mailing address.

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
United States

Accessibility features

Accessibility features help users who have physical disabilities such as restricted mobility or limited vision use software products successfully. The accessibility features in z/OS can help users do the following tasks:

- Run assistive technology such as screen readers and screen magnifier software.
- Operate specific or equivalent features by using the keyboard.
- Customize display attributes such as color, contrast, and font size.

Consult assistive technologies

Assistive technology products such as screen readers function with the user interfaces found in z/OS. Consult the product information for the specific assistive technology product that is used to access z/OS interfaces.

Keyboard navigation of the user interface

You can access z/OS user interfaces with TSO/E or ISPF. The following information describes how to use TSO/E and ISPF, including the use of keyboard shortcuts and function keys (PF keys). Each guide includes the default settings for the PF keys.

- *z/OS TSO/E Primer*
- *z/OS TSO/E User's Guide*
- *z/OS V2R2 ISPF User's Guide Vol I*

Dotted decimal syntax diagrams

Syntax diagrams are provided in dotted decimal format for users who access IBM Knowledge Center with a screen reader. In dotted decimal format, each syntax element is written on a separate line. If two or more syntax elements are always present together (or always absent together), they can appear on the same line because they are considered a single compound syntax element.

Each line starts with a dotted decimal number; for example, 3 or 3.1 or 3.1.1. To hear these numbers correctly, make sure that the screen reader is set to read out

punctuation. All the syntax elements that have the same dotted decimal number (for example, all the syntax elements that have the number 3.1) are mutually exclusive alternatives. If you hear the lines 3.1 USERID and 3.1 SYSTEMID, your syntax can include either USERID or SYSTEMID, but not both.

The dotted decimal numbering level denotes the level of nesting. For example, if a syntax element with dotted decimal number 3 is followed by a series of syntax elements with dotted decimal number 3.1, all the syntax elements numbered 3.1 are subordinate to the syntax element numbered 3.

Certain words and symbols are used next to the dotted decimal numbers to add information about the syntax elements. Occasionally, these words and symbols might occur at the beginning of the element itself. For ease of identification, if the word or symbol is a part of the syntax element, it is preceded by the backslash (\) character. The * symbol is placed next to a dotted decimal number to indicate that the syntax element repeats. For example, syntax element *FILE with dotted decimal number 3 is given the format 3 * FILE. Format 3* FILE indicates that syntax element FILE repeats. Format 3* * FILE indicates that syntax element * FILE repeats.

Characters such as commas, which are used to separate a string of syntax elements, are shown in the syntax just before the items they separate. These characters can appear on the same line as each item, or on a separate line with the same dotted decimal number as the relevant items. The line can also show another symbol to provide information about the syntax elements. For example, the lines 5.1*, 5.1 LASTRUN, and 5.1 DELETE mean that if you use more than one of the LASTRUN and DELETE syntax elements, the elements must be separated by a comma. If no separator is given, assume that you use a blank to separate each syntax element.

If a syntax element is preceded by the % symbol, it indicates a reference that is defined elsewhere. The string that follows the % symbol is the name of a syntax fragment rather than a literal. For example, the line 2.1 %OP1 means that you must refer to separate syntax fragment OP1.

The following symbols are used next to the dotted decimal numbers.

? indicates an optional syntax element
The question mark (?) symbol indicates an optional syntax element. A dotted decimal number followed by the question mark symbol (?) indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element, (for example 5? NOTIFY). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines 5 ?, 5 NOTIFY, and 5 UPDATE, you know that the syntax elements NOTIFY and UPDATE are optional. That is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.

! indicates a default syntax element
The exclamation mark (!) symbol indicates a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicate that the syntax element is the default option for all syntax elements that share the same dotted decimal number. Only one of the syntax elements that share the dotted decimal number can specify the ! symbol. For example, if you hear the lines 2? FILE, 2.1! (KEEP), and 2.1 (DELETE), you know that (KEEP) is the

default option for the FILE keyword. In the example, if you include the FILE keyword, but do not specify an option, the default option KEEP is applied. A default option also applies to the next higher dotted decimal number. In this example, if the FILE keyword is omitted, the default FILE(KEEP) is used. However, if you hear the lines 2? FILE, 2.1, 2.1.1! (KEEP), and 2.1.1 (DELETE), the default option KEEP applies only to the next higher dotted decimal number, 2.1 (which does not have an associated keyword), and does not apply to 2? FILE. Nothing is used if the keyword FILE is omitted.

*** indicates an optional syntax element that is repeatable**

The asterisk or glyph (*) symbol indicates a syntax element that can be repeated zero or more times. A dotted decimal number followed by the * symbol indicates that this syntax element can be used zero or more times; that is, it is optional and can be repeated. For example, if you hear the line 5.1* data area, you know that you can include one data area, more than one data area, or no data area. If you hear the lines 3* , 3 HOST, 3 STATE, you know that you can include HOST, STATE, both together, or nothing.

Notes:

1. If a dotted decimal number has an asterisk (*) next to it and there is only one item with that dotted decimal number, you can repeat that same item more than once.
2. If a dotted decimal number has an asterisk next to it and several items have that dotted decimal number, you can use more than one item from the list, but you cannot use the items more than once each. In the previous example, you can write HOST STATE, but you cannot write HOST HOST.
3. The * symbol is equivalent to a loopback line in a railroad syntax diagram.

+ indicates a syntax element that must be included

The plus (+) symbol indicates a syntax element that must be included at least once. A dotted decimal number followed by the + symbol indicates that the syntax element must be included one or more times. That is, it must be included at least once and can be repeated. For example, if you hear the line 6.1+ data area, you must include at least one data area. If you hear the lines 2+, 2 HOST, and 2 STATE, you know that you must include HOST, STATE, or both. Similar to the * symbol, the + symbol can repeat a particular item if it is the only item with that dotted decimal number. The + symbol, like the * symbol, is equivalent to a loopback line in a railroad syntax diagram.

Notices

This information was developed for products and services offered in the U.S.A. or elsewhere.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Site Counsel
IBM Corporation
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

COPYRIGHT LICENSE:

This information might contain sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Policy for unsupported hardware

Various z/OS elements, such as DFSMS, HCD, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted

for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: IBM Lifecycle Support for z/OS (http://www.ibm.com/software/support/systemsz/lifecycle/)
- For information about currently-supported IBM hardware, contact your IBM representative.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml .

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Microsoft, Windows, Windows NT, Windows Vista, Windows 7 and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, or service names may be trademarks or service marks of others.

Glossary for MVS Capacity Provisioning User's Guide

Application Assist Processor
See zAAP.

ARM See Automatic Restart Manager.

Automatic Restart Manager (ARM)
A z/OS recovery function that can automatically restart tasks after they or the system on which they are running end unexpectedly.

Base Control Program (BCP)
A program that provides essential services for the MVS and z/OS operating systems. The program includes functions that manage system resources. These functions include input or output, dispatch units of work, and the z/OS UNIX System Services kernel.

BCP See Base Control Program.

capacity
See temporary capacity, Defined Capacity, Group Capacity.

Capacity Provisioning Management Console
Management console which is implement by the Capacity Provisioning task in the IBM z/OS Management Facility (z/OSMF). z/OSMF provides a framework for managing various aspects of a z/OS system through a web browser interface. You can use the Capacity Provisioning task to work with provisioning policies and domain configurations and to work with the Capacity Provisioning Manager on your z/OS system.

Capacity Upgrade on Demand (CUoD)
The capability of permanently activating one or more inactive processors without having to restart the server or interrupt the data flow of the business, through the purchase of a permanent processor activation. This capability adds significant value by enabling a fast and economical way of adding capacity for new workloads, enabling the server to adapt to unexpected performance demands.

CDT See class descriptor table.

Central Processor Complex (CPC)
A physical collection of hardware that consists of main storage, one or more central processors, timers, and channels.

CFCC See Coupling Facility.

CIM See Common Information Model.

CIU See Customer Initiated Upgrade.

class descriptor table (CDT)
A table in RACF that contains class descriptors. The CDT contains descriptors with default class names for CICS® resources. Users can modify the supplied descriptors and add new ones.

Common Information Model (CIM)
An implementation-neutral, object-oriented schema for describing network management information. The Distributed Management Task Force (DMTF) develops and maintains CIM specifications.

condition
Provisioning conditions describe the situations in which capacity changes are allowed. Two types of conditions are supported: Time conditions and Workload conditions. See time condition, workload condition.

Coupling Facility
A special logical partition that provides high-speed caching, list processing, and locking functions in a sysplex.

CPC See Central Processor Complex.

CPC utilization based provisioning
Provisioning of physical capacity based on the criteria of a utilization condition.

CPC utilization Condition
A CPC utilization condition defines the physical processor utilization threshold that can trigger provisioning for a CPC.

CUoD See Capacity Upgrade on Demand.

Customer Initiated Upgrade (CIU)
A permanent upgrade ordered, downloaded, and installed by using the IBM CIU web-based application on Resource Link®.

deadline
The time within the duration of a time

condition after which provisioning of additional capacity is no longer allowed. Additional capacity that is provisioned between start time and deadline remains active until the end time or until the capacity is no longer needed. See also start time, end time.

Defined Capacity
A limit to the capacity of an LPAR, measured in MSU. WLM constraints the LPAR's workload if the its rolling 4-hour average utilization exceeds this limit.

defined capacity scope
The amount of MSU by which a rule can increase Defined Capacity. See also maximum defined capacity scope.

deprovisioning PI
The value of the performance index which the Provisioning Manager considers the service class period to no longer be suffering. The deprovisioning PI must be lower than the provisioning PI.

deprovisioning utilization of utilization condition
Processor utilization threshold for deprovisioning. If the deprovisioning utilization is equal to or lower than the specified percentage value, then the Provisioning Manager triggers deactivation of additional temporary capacity.

domain configuration
Defines the CPCs to be managed and the systems to be observed by the Provisioning Manager.

end time
The end of a time condition, after which the Provisioning Manager starts to deprovision any additional capacity provisioned within the duration of a time condition.

excluded service classes
A table listing ineligible service class periods by name and period. They are excluded from the set of service class periods previously defined by an importance filter and are not considered by the Provisioning Manager.

Group Capacity
An extension of the defined capacity concept. A common capacity limit, measured in MSU, is shared by multiple LPARs on the same CPC.

group capacity scope
The amount of MSU by which a rule can increase Group Capacity. See also maximum group capacity scope.

Hardware Management Console (HMC)
A system that controls managed systems, including the logical partitions and use of Capacity Upgrade on Demand. Using service applications, the HMC communicates with managed systems to detect, consolidate, and send information to IBM for analysis.

HMC See hardware management console.

ICF See internal coupling facility.

IFL See integrated facility for Linux.

IBM z/OS Management Facility (z/OSMF)
A framework for managing various aspects of z/OS systems. z/OSMF includes an intuitive graphical user interface (GUI) for performing various management tasks. Structurally, z/OSMF is comprised of a Web browser user interface and functions provided by z/OS components. z/OSMF is provided at no charge, as a separately licensed and entitled z/OS program product.

included service classes
A table that assigns certain service class periods to sets of provisioning criteria.

Integrated Facility for Linux (IFL)
A processor that can be used to run Linux, Linux guests on z/VM operating systems or IBM zAware. IFLs can be dedicated to a Linux, a z/VM, or a IBM zAware logical partition, or can be shared by multiple Linux guests, z/VM logical partitions or IBM zAware running on the same CPC. Only z/VM, Linux on z Systems operating systems, IBM zAware and designated software products can run on IFLs.

Internal Coupling Facility (ICF)
ICF processors allow multiple LPARs running z/OS to manage data and distribute workload in a Parallel Sysplex clustered system.

importance filter
A table that assigns service class importance levels to sets of provisioning criteria.

Integrated Information Processor
See zIIP.

Intelligent Resource Director (IRD)
A z/OS facility enhances the ability of an installation to dynamically move resources to the most important work.

IRD See Intelligent Resource Director.

logical processor
A processor that is defined in an LPAR profile. It can be either configured online, offline, or be reserved. Only online logical processors are dispatched on physical processors.

logical processor limit
A limit for the number of logical processors of a system. When this limit is reached the Provisioning Manager stops to recommend on additional logical processors.

logical processor scope
Limits for the number of logical processors for systems in the provisioning domain.

maximum defined capacity scope
The total amount of MSU by which a policy can increase defined capacity for systems in the provisioning domain. In addition specifies the increments for workload triggered increases. See also defined capacity scope.

maximum group capacity scope
The total amount of MSU by which a policy can increase Group Capacity for capacity groups in the provisioning domain. In addition specifies the increments for workload triggered increases. See also group capacity scope.

maximum processor scope
The total amount of CP, zAAP and zIIP capacity a policy can activate. In addition specifies the increments for workload triggered activations. See also processor scope.

maximum provisioning scope
The total amount of resources that can be additionally provisioned for a policy. See also provisioning scope. Comprises scopes for physical processors, logical processors, Defined Capacity and Group Capacity.

Million Service Units (MSU
Describes a per hour machine capacity. MSU are used to determine the software license charge for MLC software. Also the capacity for CPs is expressed in MSUs. In the context of defined capacity MSU units are used to set the capacity of an individual logical partition or a group of logical partitions when soft capping is selected. monoplex. A sysplex that consists of one system that

monoplex
A sysplex that consists of one system that uses a sysplex couple data set (CDS).

On/Off Capacity on Demand (On/Off CoD)
An addition to the CIU tool that enables you to configure, order, and download temporary upgrades for your processors. It is used to temporarily increase CPs, ICFs, IFLs, SAPs, zAAPs and zIIPs concurrently and non-disruptively. The increased capacity is billed on a 24-hour basis.

On/Off CoD
See On/Off Capacity on Demand.

PassTicket
A dynamically generated, random, one-time-use, password substitute in RACF secured sign-on that a workstation or other client can use to sign on to the host rather than sending a RACF password across the network.

performance index (PI)
A measure of the performance of a system, based on metrics such as transaction rates or response times.

PI See performance index.

policy A set of rules that controls the provisioning of additional capacity.

processor scope
The amount of CP, zAAP and zIIP capacity a rule can activate. CP capacity is measured in MSU, zAAP and zIIP capacity is measured in processors. See also maximum processor scope.

processor type of utilization condition
Processor Type for which the CPC utilization is monitored. Processor Types are CP, zAAP and zIIP.

provisioning
The process of configuring servers, software, networks, and storage resources.

provisioning condition
See time condition, workload condition.

provisioning criteria
A set of parameters that indicates the target performance index range that is expected of a service class period and the length of time it can go out of range before provisioning is required. See also provisioning PI, deprovisioning PI, performance index.

provisioning PI
The value of the performance index, at or above which the Provisioning Manager considers the service class period to be suffering.

provisioning scope
The amount of resources that can be additionally provisioned for a rule. See also maximum provisioning scope. Comprises scopes for physical processors, Defined Capacity and Group Capacity.

provisioning utilization of utilization condition
Processor utilization threshold for provisioning. If the provisioning utilization is equal or higher than the specified percentage value then the Provisioning Manager triggers activation of additional temporary capacity.

recurring time condition
A specification of regularly repeating time periods on a weekday basis, during which additional capacity can be provisioned or deprovisioned by a rule.

Resource Measurement Facility (RMF)
A feature of z/OS that measures selected areas of system activity and presents the data collected in the format of printed reports, System Management Facility (SMF) records, or display reports.

RMF See Resource Measurement Facility.

rule Links a processor scope to time conditions and can also link it to workload conditions.

SAF See System Authorization Facility.

SAP See System Assist Processor.

System Authorization Facility (SAF)
An MVS interface with which programs can communicate with an external security manager, such as RACF.

service class
A group of work items which have the same performance goals, resource requirements, or business importance. For workload management a service goal and optionally a resource group are assigned to a service class.

service class period
A group of work which has the same performance goals, and business importance. A service class period is managed by Workload Management as one entity.

specialty processor
See zAAP, zIIP.

start time
The start of a time condition, at which point the Provisioning Manager starts to provision additional capacity. See also deadline, end time.

sysplex
A set of z/OS systems that communicate with each other through certain multisystem hardware components and software services.

System Assist Processor
A processor that runs the channel subsystem licensed internal code (LIC) to control I/O operations. All SAPs perform I/O operations for all logical partitions.

temporary capacity
An option available on certain IBM Machines that can be enabled for each applicable TC Eligible Machine that you indicate on a Supplement. IBM can also see TC as "IBM eServer™ On/Off Capacity on Demand," "On/Off Capacity on Demand," or "On/Off CoD."

time condition
A specification of time periods during which additional capacity can be provisioned or deprovisioned by a rule. See also recurring time condition.

time period
A section of a time condition. The time periods are delimited by the start time, deadline, and end time.

utilization condition
See CPC utilization condition.

WLM See Workload Management.

workload condition
A definition of work that is eligible to cause activation of additional capacity, specifying additional constraints under which provisioning occurs.

Workload Management (WLM)
A component of z/OS that enables multiple workloads to run at the same time within one z/OS image or across multiple images.

zAAP z9® Application Assist Processor. A specialized processing unit that provides an economical Java execution environment for customers who want the traditional quality of service and integration advantages of the z platform.

zIIP z9 Integrated Information Processor. A specialty engine designed to improve resource optimization and lower the cost of eligible workloads, enhancing the role of the mainframe as the data hub of the enterprise.

z/OSMF
See IBM z/OS Management Facility.

Index

Special characters

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z

Product Number: 5650-ZOS

Printed in USA

SC34-2661-01