Systems and Technology Group

z/OS Capacity Provisioning Overview

Just-in-Time Capacity

Horst Sinram
System z Capacity Management & Workload Management
IBM Germany Research & Development
Email: sinram@de.ibm.com

August, 2009

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

APPN*
CICS*
DB2*
DB2 Connect
DirMaint
e-business logo*
ECKD
Enterprise Storage Server*
ESCON*
FICON*
GDPS*
Geographically Dispersed Parallel Sysplex
HiperSockets
HyperSwap
IBM*
IBM eServer
IBM e(logo)server*
IBM logo*
IMS
Language Environment*
MQSeries*
Multiprise*
NetView*
On demand business logo
OS/390*
Parallel Sysplex*
PR/SM
Processor Resource/Systems Manager
RACF*
Resource Link
RMF
S/390*
Sysplex Timer*
System z
System z9
System z10
TotalStorage*
VM/ESA*
VSE/ESA
VTAM*
WebSphere*
z/Architecture
z/OS*
z/VM*
z/VSE
zSeries*
z10
z9

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Microsoft, Windows and Windows XP are registered trademarks of Microsoft Corporation.
Red Hat, the Red Hat "Shadow Man" logo, and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc., in the United States and other countries.
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.
All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the 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.
Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

ENABLING BUSINESS.
A THROUGH Z.

Agenda

- Capacity Provisioning Overview
- Capacity Provisioning to simplify management of temporary capacity
  - Processing Modes
  - CPM Configuration
  - Reports, Logs, Audit Trails
  - Updates and enhancements
  - Documentation
- Special Topics

What offers IBM z/OS Capacity Provisioning?

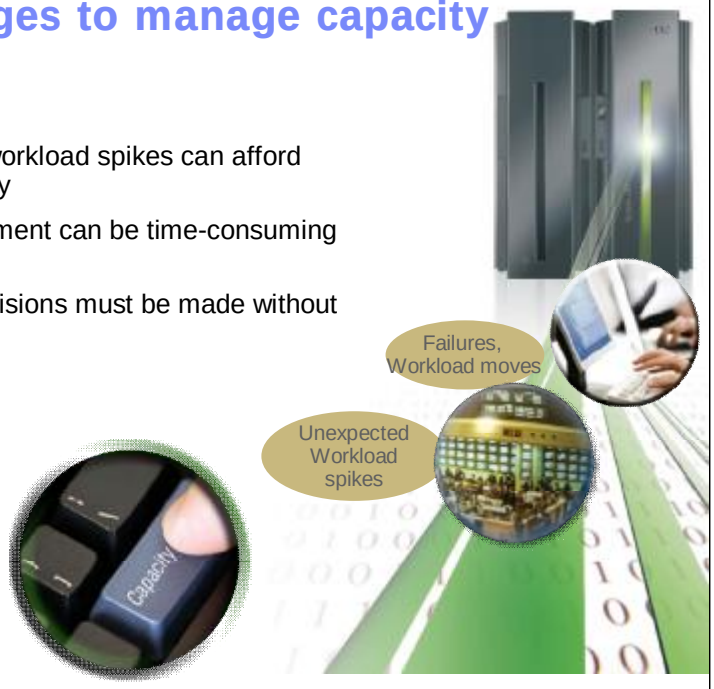
- Capacity Provisioning ensures that business has processing capacity it needs
- Capacity Provisioning allows managing processing capacity more reliably, more easily, and faster
- Capacity Provisioning provides new, flexible and automated way to control activation of On/Off Capacity on Demand
  - From manual to autonomic
- Capacity Provisioning Manager (CPM) helps to manage central processor, zAAP and zIIP capacity running z/OS System z10

Performance and capacity management on System z™ needs to ensure that the work is being processed according to the service level agreements that are in place. If for example workload increase requires that the processing capacity needs to be increased to accommodate the grown workload, capacity provisioning can do that either autonomic or in a simple manual way. The capacity change could be implemented via a permanent capacity increase or via a temporary capacity increase for seasonal or unpredictable peak periods. IBM System z provides the capability to quickly and nondisruptively activate additional processor capacity that is built directly into System z servers —IBM Capacity Upgrade on Demand (CUoD) for a permanent increase of processing capability, and IBM On/Off Capacity on Demand (On/Off CoD) for a temporary capacity increase that lets you revert to your previous processing level whenever you wish.

Capacity Provisioning is designed to simplify the management of temporary capacity. **The scope of z/OS Capacity Provisioning is to address capacity requirements for relatively short term workload fluctuations for which On/Off Capacity on Demand is applicable**. It is not a replacement for the Capacity Management process.

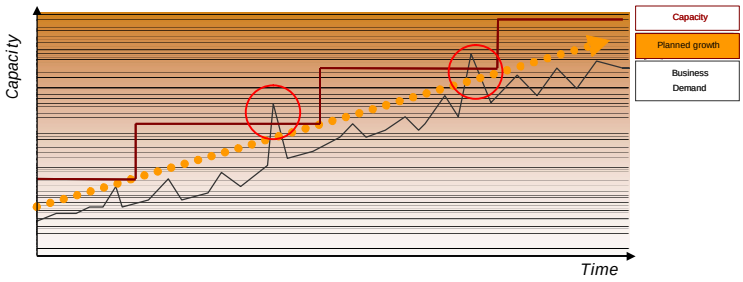
Today's challenges to manage capacity

- Unexpected events and workload spikes can afford higher processing capacity
- Manual capacity management can be time-consuming and error prone
- Capacity provisioning decisions must be made without sound data



IBM z/OS Capacity Provisioning Basics

- **Contained in z/OS base component free of charge**
  - Requires z/OS RMF feature
  - Runs on z/OS Release 9 and above
- **Based on System z10 (and follow-on) On/Off Capacity on Demand Feature**



Performance and capacity management on System z™ needs to ensure that the work is being processed according to the service level agreements that are in place. Guaranteeing service levels remains a relatively static task only as long as the workloads that need to be considered are sufficiently stable. However, in many environments workloads may fluctuate considerably over time. As the total workload or the mixture of workloads varies guaranteeing service levels may increasingly getting difficult. On z/OS, Workload Management (WLM) allows the incoming work to be classified with a performance goal and a priority that reflects the business priority of that work. WLM will try to accommodate the goals of all the work in the system.

However, even with an ideal workload management it may not be possible to achieve all specified goals when the total workload increases. In that case trade-offs need to be made. WLM decides which goals may be compromised first based on the assigned importance level. Discretionary, then low importance work will be displaced first. At some point, however, that may no longer be acceptable. In that case the processing capacity needs to be increased to accommodate the grown workload. The capacity change could be implemented via a permanent capacity increase or via a temporary capacity increase for seasonal or unpredictable peak periods. IBM System z provides the capability to quickly and nondisruptively activate additional processor capacity that is built directly into System z servers —IBM Capacity Upgrade on Demand (CUoD) for a permanent increase of processing capability, and IBM On/Off Capacity on Demand (On/Off CoD) for a temporary capacity increase that lets you revert to your previous processing level whenever you wish.

Capacity Provisioning is designed to simplify the management of temporary capacity. **The scope of z/OS Capacity Provisioning is to address capacity requirements for relatively short term workload fluctuations for which On/Off Capacity on Demand is applicable**. It is not a replacement for the Capacity Management process.

Capacity Provisioning Capabilities Overview

- The Capacity Provisioning Manager (CPM) can control temporary processor resources on IBM System z10
  - Number of zAAPs or zIIPs
  - General purpose capacity
    - Considers different capacity levels (i.e. effective processor speeds) for subcapacity processors (general purpose capacity)
  - Can advise on logical processors
  - Can control one or more IBM System z10 servers
    - Includes multiple Sysplexes

CPM differentiates between different types of provisioning requests:

- Manually at the z/OS console through Capacity Provisioning Manager commands
- Via user defined policy at specified schedules
- Via user defined policy by observing workload performance on z/OS

Capacity Provisioning can help you to manage processor capacity on IBM System z10 when a suitable On/Off Capacity on Demand record is available. Capacity provisioning allows you to change the activation level of that On/Off CoD record with respect to general purpose capacity, and the number of zAAP or zIIP processors. For general purpose capacity on subcapacity models it differentiate between „speed“ demand for higher capacity levels, and “unqualified” demand that could be satisfied by a capacity level increase as well as by additional processors.

CPM differentiates between different types of Provisioning Requests

- Manually through commands
  - SE/HMC actions still possible, of course
- Scheduled (time condition without workload condition)
- Conditional (based on workload condition)

Manual capacity upgrades – How it could look like

1.	Workload increases	0 Min
2.	Operator realizes bottleneck	5-10 Min
3.	Operator informs system programmers and manager	2 Min
4.	Discussion	10 Min
5.	Logon to HMC, activate record	5 Min

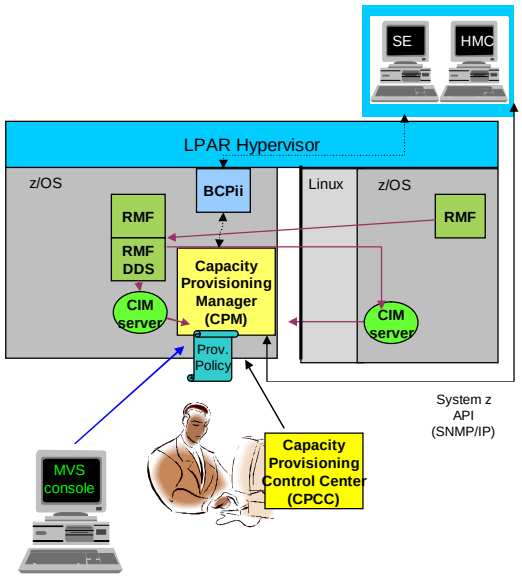
… meanwhile, so much workload has been queued that one additional processor is not sufficient to decrease the queued workload

à Two processors have to be added

CPM can react fast and reduce costs

Capacity Provisioning – Infrastructure in a Nutshell

- z/OS WLM manages workloads to goals and business importance
- WLM indicators available through z/OS Resource Measurement Facility (RMF)
  - One RMF gatherer per z/OS system
  - RMF Distributed Data Server (DDS) per Sysplex
- Capacity Provisioning Manager (CPM) retrieves critical metrics through CIM
- CPM communicates to support elements or HMC, via
  - System z API (SNMP via IP)
  - BCPii
- Capacity Provisioning Control Center is front end to *administer* Capacity Provisioning policies



On each z/OS system WLM manages the workload by goals and business importance

WLM metrics are available through existing interfaces and are reported through (e.g.) RMF Monitor III.

•One RMF gatherer per z/OS system

•Sysplex wide data aggregation & propagation occurs in the RMF distributed data server (DDS)

The RMF CIM providers (and associated CIM models) publish the RMF Monitor III performance data

The Capacity Provisioning Manager (CPM) retrieves critical metrics from one or more z/OS systems through CIM via http or https.

The CPM communicates to (local or remote) support elements and HMCs, respectively, via

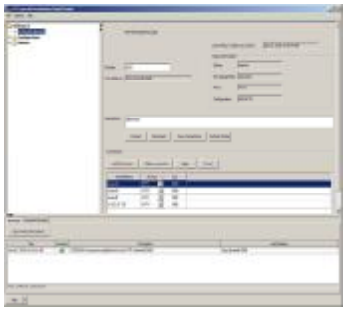
System z API (SNMP via IP)

The Capacity Provisioning Control Center is the user front end to *administer* Capacity Provisioning policies. It is installed on a Windows ™ workstation. It creates Capacity Provisioning policies in XML format.

(It is not required for regular CPM operation but only for policy administration).

Main Components of Capacity Provisioning

- **The Capacity Provisioning Manager** (CPM)
  - is the server program that monitors the defined systems and CPCs and takes actions as appropriate and authorized by the policies.
- **The Capacity Provisioning Control Center** (CPCC)
  - is the Graphical User Interface (GUI) component. It is the interface through which administrators work with provisioning policies and domain configurations.
  - Optionally, you can use the CPCC to transfer provisioning policies and domain configurations files to the CPM, or to query the Capacity Provisioning Manager status.
  - The CPCC is installed and used on a Windows™ workstation. It is not required for regular operation of the CPM.



This chart summarizes again the two main components of Capacity Provisioning:

1. **The Capacity Provisioning Manager** (**CPM**) which is a started task on z/OS and performs all tasks as defined into its policies.
   It is started via START CPOSERV. When running, it accepts operator commands through the MODIFY interface.

   E.g. F CPOSERV,APPL=<command>

2. **The Capacity Provisioning Control Center** (**CPCC**)

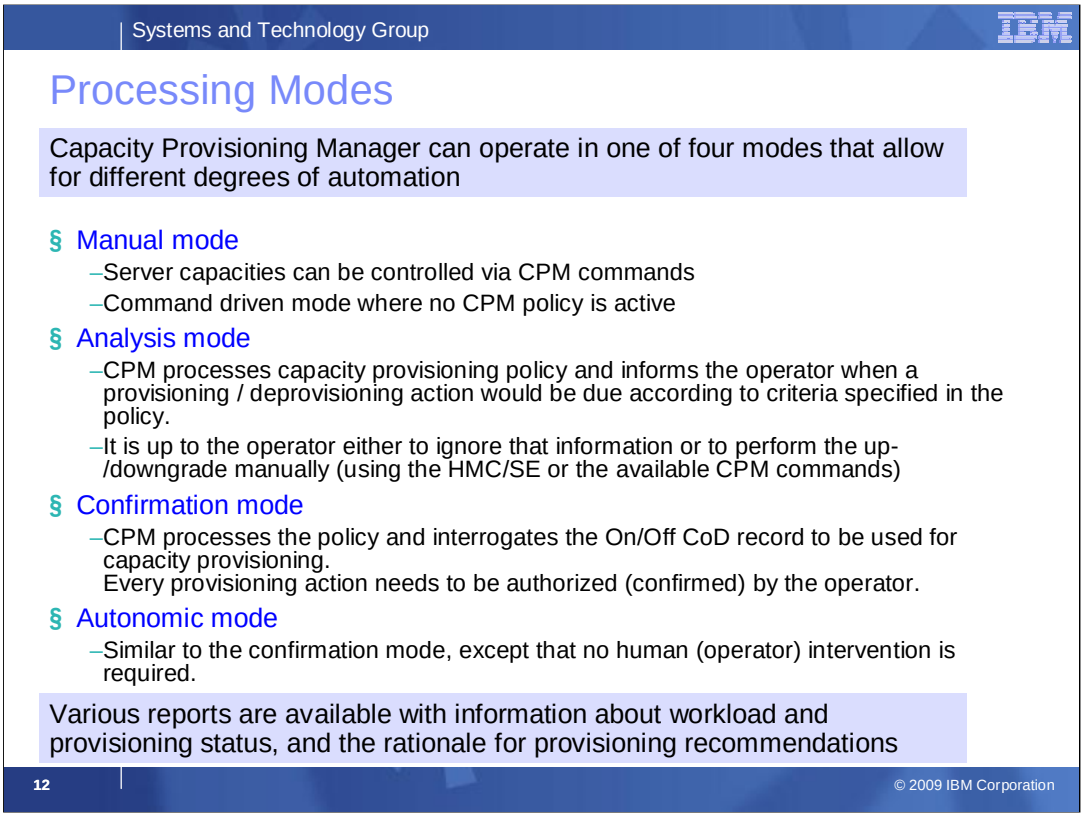
It is shipped with z/OS as /usr/lpp/cpo/pws/cpccsetup.exe. For installation it needs to be downloaded in binary format and executed which starts the install program.

After installation, the program is invoked via its icon, or through the Windows “start” menu.

The two types of policies that the Capacity Provisioning Manager observes are edited or viewed in the CPCC.

Agenda

- Capacity Provisioning Overview
- Capacity Provisioning to simplify management of temporary capacity
  - Processing Modes
  - CPM Configuration
  - Reports, Logs, Audit Trails
  - Updates and future enhancements
  - Documentation



The fully autonomic management of temporary capacity would usually only be implemented when an installation has verified that the CPM management is in line with the objectives.

Even then an installation may feel that it wants to control each single action suggested by the CPM.

In fact the CPM supports four different „processing modes“ that allow for different degrees of automation and serve different purposes:

1) The most basic mode is the „Manual“ mode. In this mode the CPM is not policy-aware. However, it does listen to console commands and does accept and execute commands. Also, policies may be installed through the CPCC.

2) In the analysis mode the CPM does process a policy. In fact this mode is primarily intended to develop a policy. So the CPM processes the policy, observes workload and indicates when additional or less capacity of different types would be suggested. However, in this mode the CPM does not observe the hardware capabilities and does not interrogate installed the On/Off CoD records. Consequently the capacity suggestions are not matched with the On/Off CoD record. In addition, the „manual“ mode capabilities are available.

The next two modes provide a similar, comprehensive functionality. The CPM is fully aware of the active policy, interrogates the On/Off CoD record and listens to hardware capacity changes.

In „Confirmation“ mode, workloads are observed, and when a capacity change is warranted the CPM message (WTOR) is issued to the z/OS console. That reply can either be approved in which cas it is implemented right away, or rejected. Also, the reply may be just left unanswered. When the workload situation or policy changes, the CPM reply may be withdrawn or updated. Only approved changes will be implemented.

In autonomic mode, no operator intervention will be required and the capacity setting of the server would be updated automatically, as authorized by policy and On/Off CoD record.

In all modes, various reports commands can be issued to report on the hardware, and system configuration and workloads as defined to, and observed by the CPM.

Agenda

- Capacity Provisioning Overview
- Capacity Provisioning to simplify management of temporary capacity
  - Processing Modes
  - CPM Configuration
  - Reports, Logs, Audit Trails
  - Updates and future enhancements
  - Documentation

CPM Policies and Processing Parameters

- CPM server uses three types of inputs:
  - Domain configuration defines the topology and connections, such as the CPCs and z/OS systems that are to be managed by the server
  - Policy contains the information as to
    - which work is provisioning eligible,
      - under which conditions and during which timeframes,
    - how much capacity may be activated when the work suffers due to insufficient processing capacity
  - PARM data set contains setup instructions such as UNIX environment variables, and various processing options that may be set by an installation.

The CPM processing is controlled by two definitions:

The **domain configuration** describes **WHAT** is to be managed. It defines the topology that the CPM controls, i.e.

- All the servers (CPCs) known to the CPM, and
- All the z/OS images observed by the CPM. Workloads can only be monitored on systems defined into the domain.

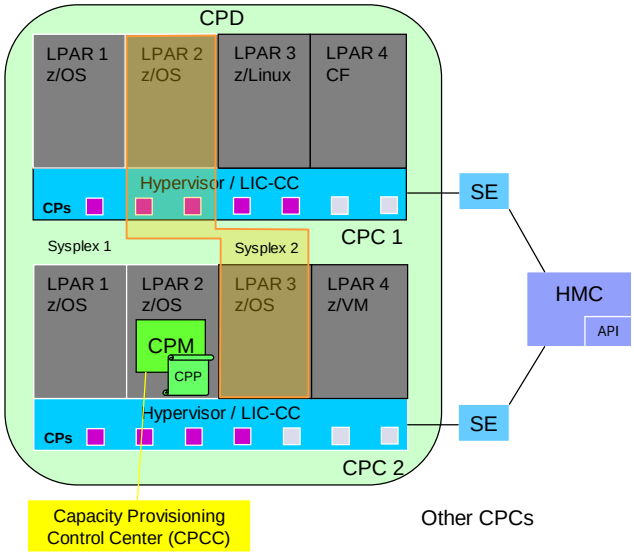
A domain configuration is required for all processing modes, including manual mode.

The **policy** describes **HOW** the domain is to be managed, i.e., the conditions and criteria governing the provisoning and deprovisioning of temporary capacity.

The **parm** member contains environmental setup and processing directives.

The domain configuration and policy are detailed on subsequent charts.

Domain Configuration



- Domain configuration defines the CPCs and z/OS systems that are controlled by an instance of the CPM
- One or more CPCs, sysplexes and z/OS systems can be defined into a domain
- Sysplexes and CPCs do not have to be completely contained in a domain but must not belong to more than one Capacity Provisioning domain
- One active Capacity Provisioning policy per domain
- Multiple Sysplexes and hence multiple WLM service definitions may be involved

Domain configuration defines the CPCs and z/OS systems that are controlled by an instance of the CPM.

Multiple CPCs, sysplexes and z/OS systems can be defined into a single domain.

Sysplexes and CPCs do not have to be completely contained in a domain but must not belong to more than one CP domain. In other words, no CPC and no Sysplex must be defined into different domains (and therefore controlled by different CPMs).

At CPM run time, CPCs and systems can be dynamically enabled or disabled for CPM processing via commands.

At any point in time only one Capacity Provisioning policy can be active per domain (or no policy at all).

Because multiple Sysplexes may be defined into a domain the policy must allow for referencing multiple WLM service definitions.

Policy Approach

Capacity Provisioning Policy defines the circumstances under which additional capacity may be provisioned:

- Three “dimensions” of criteria considered:
  - **When** is provisioning allowed
  - **Which** work qualifies for provisioning
  - **How much** additional capacity may be activated
- These criteria are specified as “rules” in the policy:

If
{ - in the specified time interval
- the specified work “suffers”
}
Then up to
{ - the defined additional capacity
may be activated
}

- The specified rules and conditions are named and may be activated or deactivated selectively by operator commands

Capacity Provisioning Policy defines the circumstances under which additional capacity may be provisioned.
Three “dimensions” of criteria are considered:

1. When is provisioning allowed
2. Which work qualifies for provisioning
3. How much additional capacity may be activated

These criteria are specified as “rules” in the policy:

„If

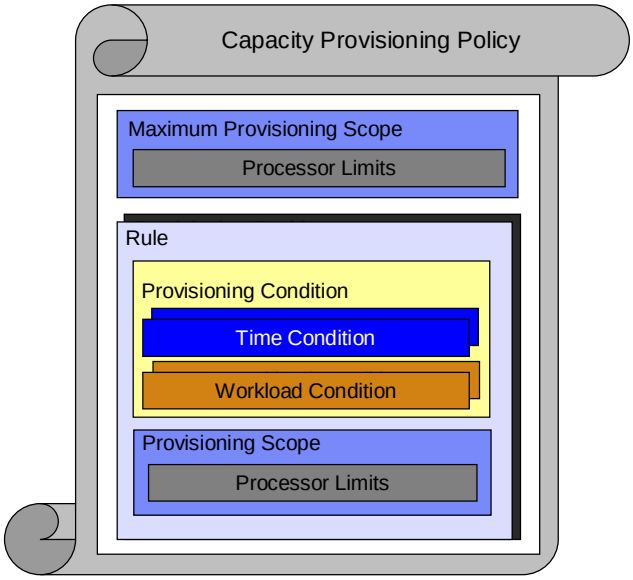
- in the specified time interval
- the specified work “suffers”

Then up to the defined additional capacity may be activated“

The specified rules and conditions are **named** constructs. At CPM run time, they can be dynamically enabled or disabled via commands without changing the installed policy.

.

Policy Overview



- The „Maximum Provisioning Scope“ defines the maximum additional capacity that may be activated, by *all* the contained rules
- „Provisioning Condition“ is simply a group of Time and Workload Conditions that can be referred to via its name
- “Provisioning Scope” defines the *maximum* capacity that may be activated based on the rule
  - Specified as number of zAAP/zIIP processors, or
  - MSU for general purpose capacity

Each Capacity Provisioning policy does usually consist of multiple parts that altogether allow for a comprehensive definition of the objectives.

The maximum provisioning scope specifies an upper limit to the capacity that may be activated by all contained rules at any point in time.

Then one or more rules can be defined into a policy.

Each rule consists of a „provisioning condition“ and the provisioning scope. The latter defines how much capacity may be activated by that specific rule **at most**. The provisioning scope is expressed in number of zAAP and zIIP processors, and MSU for general purpose processor capacity.

The provisioning condition consists of a time condition and a workload condition:

- The time conditions specify the time interval during which capacity may be added or kept online.
- Optionally, a workload condition specifies for how long a workload must have „suffered“ before adding capacity should be considered.

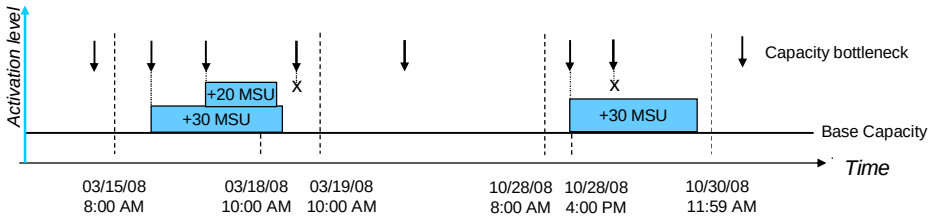
If no workload condition is specified in a rule then the full capacity as specified in the processor limits will be activated and deactivated at the times defined into the time condition. This is an unconditional –scheduled- activation and deactivation.

Note that the combination of scopes and maximum provisioning scope allows for a specification equivalent to e.g. „up to 3 processors for worklaod A, plus up to 2 processors for workload 2, but never more than 4 processors in total“.

Rules: Provisioning Conditions - Time

- Time condition defines *when* temporary capacity may be activated:
  - Start Time: provisioning of additional capacity allowed
  - Deadline: provisioning of additional capacity no longer allowed
  - End Time: deactivation of additional capacity should begin

Name	Start Time	Deadline	End Time
TC1	**03/15/08 08:00 AM**	**03/18/08 10:00 AM**	**03/19/08 10:00 AM**
TC2	**10/28/08 08:00 AM**	**10/28/08 04:00 PM**	**10/30/08 11:59 AM**

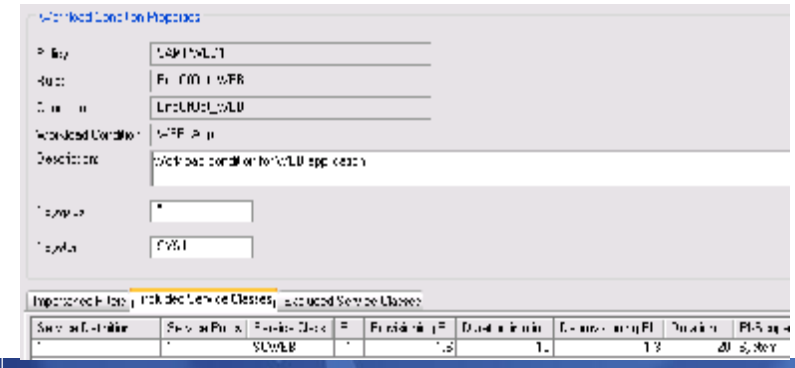


Time Conditions represent time periods during which additional capacity can be activated, up to the limit defined by the Provisioning Scope of the Rule.

The figure shows the definition of two Time Conditions and how CPM interprets them. On the left side of the figure, the effect of the first Time Condition is shown. Resource shortages are only considered between the specified start time and deadline. This means, additional resource shortages between the deadline and the end time are ignored. The boxes represent the capacity of additionally provisioned CP MSU capacity. On the right side of the figure, the effect of the second Time Condition is shown. Here, the period between start time and deadline is very small compared to the period between deadline and end time. This means, additionally provisioned capacity can remain active for a longer period, but it is not allowed to increase the provisioned capacity in that phase.

Rules: Provisioning Conditions - Workload

- Identifies the work that may trigger the activation of additional capacity,
  - when that work does not achieve its goal due to insufficient capacity *and* additional capacity would help.
  - expressed as one or more WLM service class periods
- Currently, CICS and IMS *transaction* service classes cannot be used to trigger provisioning actions
  - Service classes that the servers are managed to could be used, though.



A workload condition defines work that is eligible to cause activation of additional capacity, and the conditions under which that work can trigger activation. The specification of eligible work follows the workload model of the z/OS Workload Manager (WLM).

Workload Condition Parameters

- Parameters:
  - Sysplex/Systems: The z/OS systems that may run eligible work
  - Workload specification:
    - Importance Filter:
      Eligible service class periods, identified by *WLM importance*
    - Included Service Classes: Eligible service class periods
      - Extends the set of Service Class periods with qualified work (extends the default set of default eligible service classes) and may specify different PI criteria
    - Excluded Service Classes: Identifies service class periods, that should not be considered

The first part of a workload condition specifies the Sysplexes and systems to be monitored for the specific condition.

To apply the condition to all sysplexes in the domain configuration insert an asterisk (*) here. Otherwise only workload on the named Sysplex will be considered to trigger provisioning for this workload condition.

Similarly, you can enter the name of a z/OS system in the **System** field. To apply the condition to all systems in the named Sysplex or the domain configuration insert an asterisk (*). Otherwise only workload on the named system will be considered to trigger provisioning for this workload condition.

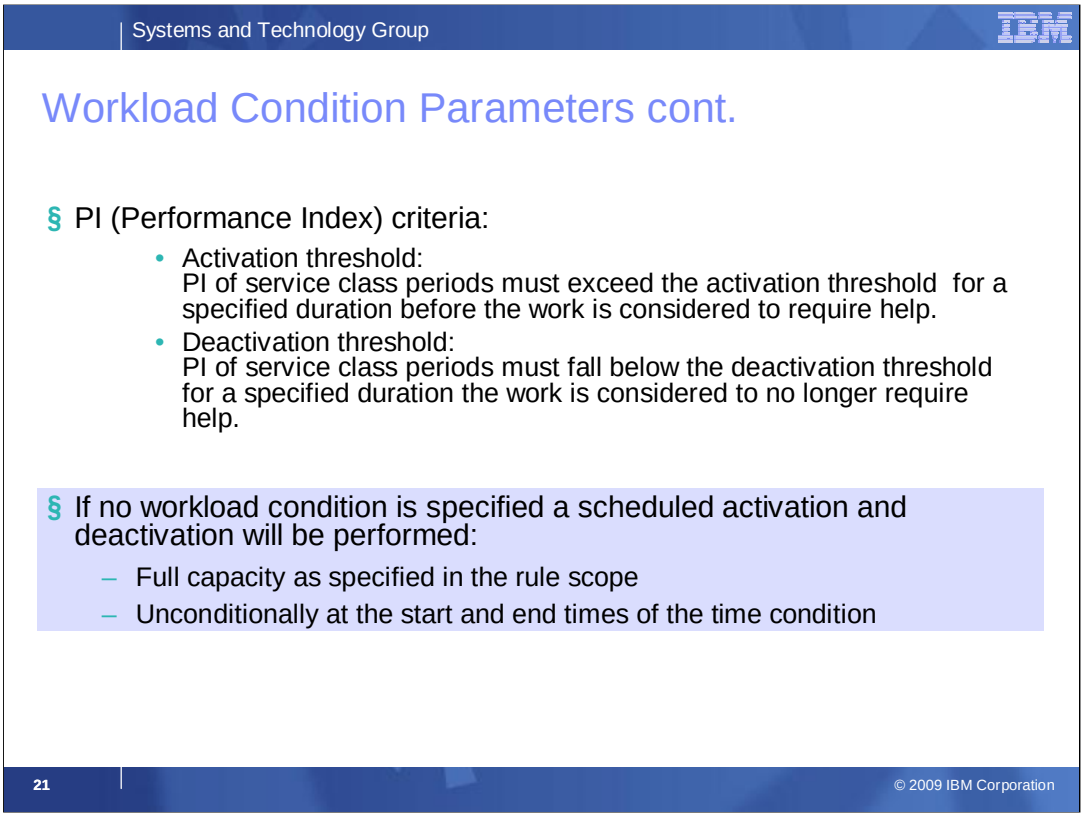
The workload to be observed can be defined by a combination of three tabs: Importance filters, includes service classes, and excluded service classes:

•**Importance Filters** allows you to specify service class periods to be monitored, based on their importance, and the performance index values and durations to trigger intervention.

•**Included Service Classes** allows you to specify service classes to be monitored in addition to any identified by importance filters, and the triggers for these.

•**Excluded Service Classes filter** allows you to specify service classes to be excluded from importance filters, or subsets

If specifications exist on multiple levels then the service class periods as derived from the importance filter are merged with the explicitly defined (included) service class period. Finally the excluded service class periods (if any) are removed from the previous set.



Provisioning PI: If the performance index of a service class period is equal or higher than the specified value the Provisioning Manager considers the service class period to be suffering.

Duration in min: This is the duration in minutes a service class period has to exceed the provisioning PI before the Provisioning Manager considers the service class period to be suffering.

Deprovisioning PI: If the performance index of a service class period is lower than the specified value it is not considered to be suffering. The deprovisioning PI must be at least 0.2 less than the provisioning PI limit, and must be not less than 1.1.

Duration in min : This is the duration in minutes the PI of the selected service classes must be lower than the specified deprovisioning PI for it to be considered no longer suffering. It must have a value between 5 and 1440 minutes.

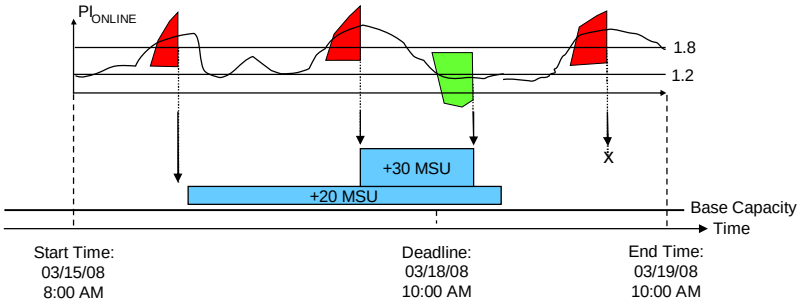
PI-Scope: Indicates which PI the other criteria apply to. The possible values are **System (known as local PI in WLM)** and **Sysplex**. The default and recommended value for most situations is **System**.

Sample Workload Condition

Sample definition

Name: *PT1*
Sysplex: *PLEX1*
System: *SYSA*
Included Service Class Periods:
ONLINE in WLMSD with PI >= 1.8 for 10 min until PI <= 1.2 for 10 min
Excluded Service Class Periods:
BACKUP in WLMSD

Monitor Service Class PI's:



Workload Conditions identify the work that is eligible for provisioning and the conditions under which that work can trigger the activation of additional capacity. The specification of eligible work incorporates the workload model of the z/OS Workload Manager (WLM), i.e. it refers to service classes defined in the WLM service definition.

This chart assumes a Workload Condition that includes service class ONLINE defined into WLM Service Definition WLMSD. The provisioning PI equals 1.8, the Provisioning PI Duration is 10 min. The Deprovisioning PI equals 1.2, and Deprovisioning PI Duration is 10 min. If the PI of the service class period evolves within a defined Time Condition as shown, CPM would detect three instances of the the provisioning PI criterion being fulfilled.

At the first two instants, CPM would activate additional capacity. The last instance does **not** lead to an additional activation because it is after the deadline. CPM would also detect an instance of the deprovisioning PI criterion being fulfilled. CPM decides here that service class ONLINE does no longer need help and deactivates the additional capacity.

Additional capacity would only be provisioned if demand for additional Capacity Provisioning/zAAp/zIIP is recognized. This analysis is based on (many) metrics on the CPC, system, and service class levels.

Rule: Provisioning Scope – Processor Limits

- CPC within provisioning domain for which activation of resources is allowed
- Max number of additional MSU/zAAPs/zIIPs that may be activated
  - Only the required delta capacity will be activated by the CPM
- Provisioning scope exists in two flavors:
  - Maximum provisioning scope defines an upper limit of resources that may be activated ***in total*** for all the contained rules at any point in time
  - Provisioning scope on the „rule“ level defines an upper limit of resources that may be activated for the single rule at any point in time
  - Allows for definitions like „*I authorize 300 MSU for workload 1 and 200 MSU for workload 2, but at no point in time more than 400 MSU*.“

CPC	Max MSU	Max zAAPs	Max zIIPs
CPC1	400	3	5
CPC2	800	0	0

A Provisioning Scope has CPC scope. It defines which capacity can additionally be activated on which CPC in the Provisioning Domain.

The table shows an example of a Provisioning Scope. Here, limits for two CPCs are defined. The first definition specifies that on CPC1 a maximum of 400 CP MSU, five additional zIIPs, and three additional zAAPs may be activated. The second definition specifies that on CPC2 800 MSU at a maximum, and no zIIPs or zAAPs may be activated.

Agenda

- Capacity Provisioning Overview
- Capacity Provisioning to simplify management of temporary capacity
  - Processing Modes
  - CPM Configuration
    - Domain Configuration
    - Policy
  - Reports, Logs, Audit Trails
  - Updates and future enhancements
  - Documentation

Agenda

- Capacity Provisioning Overview
- Capacity Provisioning to simplify management of temporary capacity
  - Processing Modes
  - CPM Configuration
  - Reports, Logs, Audit Trails
  - Updates and enhancements
  - Documentation
- Special Topics

Reports, Logs, Audit Trails

- History of actual workload and system activity available with CPM reports
  - Especially REPORT ACTIVITY, REPORT WORKLOAD
  - Reports can be directed to files and archived
- History of capacity changes available via CPM logging
  - Metrics, decisions and other data can be logged to file system
  - Binary format
- Other information available:
  - RMF Mon III data sets
  - Model and capacity changes recorded outside CPM
    - SMF22
    - RMF 70.1
    - Current capacity information also available via STSI instruction, and related MVS programming interfaces

Various CPM reports are available and all reports can be redirected to files from where they could be archived. Especially the ACTIVITY and WORKLOAD reports would be of interest, as well as the CONFIGURATION report.

The CPM can be configured dynamically (via command) to record observed metrics, decisions or exceptional situation. These logs are stored in binary format and since they are primarily intended for problem diagnosis the formatters are currently only available inside IBM.

For customer reporting purposes, it is possible to configure RMF Monitor III data sets.

Capacity changes are recorded via SMF. This occurs independently of CPM, e.g. in the type 22 and RMF type 70.1 records.

Also independently of CPM the current permanent, total, and total billable capacity can be queried. The STSI instruction, and the equivalent CSRSI service, and the SYSEVENT REQLPDAT provides that kind of information.

Agenda

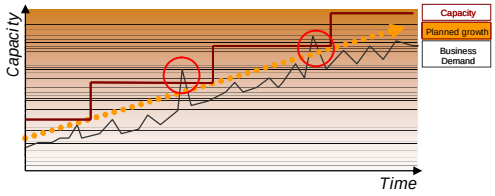
- Capacity Provisioning Overview
- Capacity Provisioning to simplify management of temporary capacity
  - Processing Modes
  - CPM Configuration
  - Reports, Logs, Audit Trails
  - Updates and enhancements
  - Documentation
- Special Topics

Capacity Provisioning R10+ Enhancements

- Additional z/OS R10 enhancements became available 6/2009
  - Capacity Provisioning exploitation of BCPii
    - Alternative option to configure SE/HMC communication
    - No need for IP connection to HMC and firewall settings
    - Access to resources can be granted via z/OS security definitions
  - Logical processor management
  - Enablement via APARs
    - OA24945 - CPM
    - OA25426 – BCPii
- z/OS R11 Enhancements
  - Capacity Provisioning CIM provider already pre-registered in CIM repository shipped with R11
    - Eliminates customization steps

Summary

- Capacity Provisioning allows for faster reaction to workload spikes
  - Replacing manual analysis with policy based monitoring of workloads
  - Customer defined criteria and objectives
  - Supports general purpose, zAAP, and zIIP capacity



- **Can be configured to different levels of automation**
  - z/OS system commands for capacity changes
    - Eliminates need to access HMC
  - Scheduled capacity changes
  - Provide capacity recommendations to staff
  - Optionally, full automation, eliminating human intervention
- **z/OS base component**
  - z/OS Release 9 and above
  - Utilizes z/OS Resource Monitoring Facility (RMF)
  - Uses open standards protocol
    - Common Information Model (CIM)
  - Mostly zAAP eligible
- **Hardware Pre-requisites**
  - IBM System z10 EC or BC
  - Based on System z10 On/Off Capacity on Demand

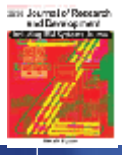
Capacity Provisioning is designed to simplify the management of temporary capacity. **The scope of z/OS Capacity Provisioning is to address capacity requirements for relatively short term workload fluctuations for which On/Off Capacity on Demand is applicable**. It is not a replacement for the Capacity Management process.

Agenda

- Capacity Provisioning Overview
- Capacity Provisioning to simplify management of temporary capacity
  - Processing Modes
  - CPM Configuration
  - Reports, Logs, Audit Trails
  - Updates and future enhancements
  - Documentation
- Special Topics

Documentation

- z/OS MVS Capacity Provisioning User's Guide,
  SC33-8299, at http://publibz.boulder.ibm.com/epubs/pdf/iea2u110.pdf
- IBM DEMOzone Demonstration of Capacity Provisioning
  http://www.demos.ibm.com/servers/Demo/IBM_Demo_IBM_z_OS_Capacity_Provisioning-Jan09.html
- Website under the WLM homepage
  http://www.ibm.com/servers/eserver/zseries/zos/wlm/cp
- ITSO Redbook:
  System z10 Enterprise Class Capacity on Demand, SG24-7504
  http://www.redbooks.ibm.com/abstracts/sg247504.html?Open



- Capacity on Demand advancements on the
  IBM System z10, IBM J. RES. & DEV. VOL. 53 NO. 1 PAPER 15 2009
  http://www.research.ibm.com/journal/rd/531/axnix.pdf

z/OS MVS Capacity Provisioning User's Guide, SC33-8299, is a new publication in the z/OS Release 9 library.

The ITSO is developing a redbook ITSO Redbook "System z10 Enterprise Class Capacity on Demand“, SG24-7504.

More current information will be hosted under the WLM homepage
Website to be created under the WLM/SRM homepage
http://www.ibm.com/servers/eserver/zseries/zos/wlm

Agenda

- Capacity Provisioning Overview
- Capacity Provisioning to simplify management of temporary capacity
  - Processing Modes
  - CPM Configuration
  - Reports, Logs, Audit Trails
  - Updates and future enhancements
  - Documentation
- Special Topics

Capacity Provisioning Service Data

- Capacity Provisioning is z/OS BCP FMID
  - z/OS R11: HPV7760
    z/OS R10: HPV7750
    z/OS R9: HPV7740
- Component ID (COMPID): 5752SCCAP
- Required service (all components, not just Capacity Provisioning) is identified via functional PSP bucket
  - Use "CAPPROV/K" keyword to search

- On z/OS Release 9 OA20824 enables the Capacity Provisioning function
- On z/OS Release 10 OA24945 enables the Capacity Provisioning usage of BCPii and Logical Processor Management
- Capacity Provisioning APARs that include a new level of the Capacity Provisioning Control Center (CPCC) specify a ++HOLD REASON(DOWNLD) action

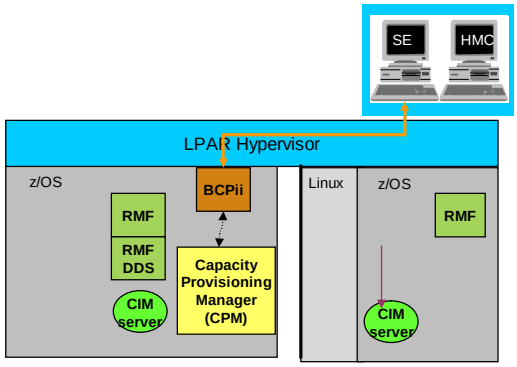
These APARs will be required for Capacity Provisioning on IBM z10 servers.

OA20824 is the APAR that enables the Capacity Provisioning function on z/OS Release 9.

OA26276 includes the support for z10 GA2 and z10 BC models.

OA24945: BCPii Support

- Alternate communication path to the hardware – substitutes SNMP (IP-based) communication
- Exploits new z/OS component Basic Control Program internal interface (BCPii) – SCLP based communication through Hypervisor
- New address space in z/OS (HWIBCPII) – started automatically after IPL
- Requires also CEA address space



System z API (SNMP/IP)

BCPii support requires minimum firmware level

- Review current PSP bucket 2098DEVICE or 2097DEVICE for required MCLs
  - As of 8/2009:
    - The PTF for APAR OA25426, PTF UA47493, along with underlying firmware support (identified below), now enables Base Control Program internal interface (BCPii). With BCPii, IBM provides an easy-to-use high-level programmable interface that can be utilized by any z/OS authorized application running in any address space to perform HMC-like functions, while eliminating potential security concerns by communicating directly to the HMC/SE through the operating system and not requiring the HMC network to be connected to any intranet/internet network.
    - Use of BCPii requires the following LIC levels be installed on a z10 GA1, or z10 GA2 system. - For z10 Systems at Driver-76: MCL129 in EC N10970 MCL046 in EC N10976
    - Note: Driver-73 should upgrade to Driver-76.

Customization Overview for BCPii (z/OS)

- z/OS “Common Event Adapter“ (CEA) address space must be set up to run in full function mode
  - See z/OS R10 “Planning for Installation”
- z/OS “BCP internal interface” (BCPii) must be set up
  - See z/OS R10 “OSMVS Programming: Callable Services for High-Level Languages” for details
- Security definitions must be made
  - BCPii needs be allowed to communicate with local SE
  - Allow Capacity Provisioning manager READ access to security profiles in SERVAUTH class
    - CEA.CONNECT
    - CEA.SUBSCRIBE.ENF_0068*
  - READ access to ESM profiles HWI.APPLNAME.HWISERV in FACILITY class
  - For each managed CPC
    - CONTROL access to ESM profile HWI.TARGET.*net ID.NAU* in FACILITY class
    - READ access to ESM profile HWI.CAPREC.*net ID.NAU.** in FACILITY class

For CEA:

Common event adapter (CEA) is a component of the BCP that provides the ability to deliver z/OS® events to C-language clients, such as the z/OS CIM server. A CEA address space is started automatically during initialization of every z/OS system.

CEA has two modes of operation:

Full function mode. In this mode, both internal z/OS components and clients (such as CIM providers) using the CEA application programming interface can use CEA functions.

Minimum mode. In this mode, only internal z/OS components can use CEA functions.

If you want CEA to start in full function mode, perform the following customization (or else CEA will start in minimum mode):

Define user ID CEA to the security product:

ADDUSER CEA DFLTGRP(SYS1) OMVS(UID(any) FILEPROCMAX(1024))

RDEFINE STARTED CEA.** STDATA(USER(CEA) GROUP(SYS1) TRACE(YES))

Give user ID CEA read access to the profile protecting SYS1.PARMLIB:

PERMIT 'SYS1.PARMLIB' GENERIC ACCESS(READ) ID(CEA)

For BCPii,

RALTER FACILITY HWI.TARGET.NET1.CPC001 APPLDATA(‘COMMUNITY_NAME’)
Note: A community name definition must be defined for at least the local CPC; otherwise, BCPii cannot continue with initialization of its address space and BCPii services are not available. This is accompanied by message HWI014I.

For CPM:

RDEFINE SERVAUTH CEA.CONNECT UACC(NONE)

RDEFINE SERVAUTH CEA.SUBSCRIBE.ENF_0068* UACC(NONE)

PERMIT CEA.CONNECT CLASS(SERVAUTH) ID(CPOSRV) ACCESS(READ)

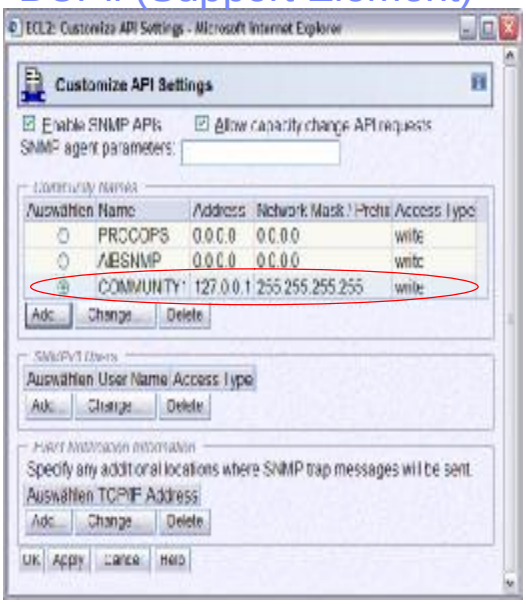
PERMIT CEA.SUBSCRIBE.ENF_006803010001 CLASS(SERVAUTH) ID(CPOSRV) ACCESS(READ)

SETROPTS RACLIST(SERVAUTH) REFRESH

Customization Overview for BCPii (Support Element)

On SE of all managed and the local CPC:

- Define community name as specified in ESM profile
- Allow for local host only
  - Address 127.0.0.1,
  - network mask 255.255.255.255
- Check „Enable SNMP API“
- Check „Allow capacity change API requests“



Activation of BCPii in Capacity Provisioning

- In the <hlq>.<domain>.PARM(PARM) change:
  - Topology.Protocol = INTERNAL
- All runtime systems must be using z/OS R10 with AO24945
- The following SNMP-specific key may be removed:
  - Topology.Protocol = SNMP
  - Topology.Address = *HMC_address*
  - Topology.Community = *community_name*

Configuring Capacity Provisioning Hardware access

The Provisioning Manager must access a hardware console to get information about the available CPCs and the temporary capacity of these, and to activate and deactivate the temporary capacity, if required. You must specify access information for the HMC. The primary information is the protocol. It is specified using the configuration key Topology.Address.

The value can be either SNMP for the SNMP protocol, or INTERNAL for BCPii. For BCPii the information would look like the following:

Topology settings

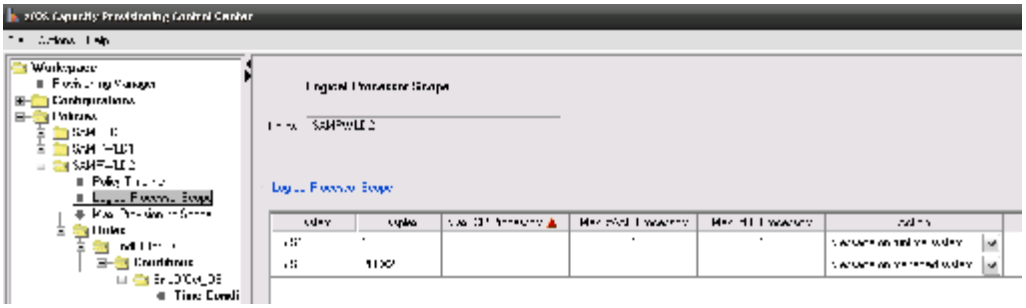
Topology.Protocol = INTERNAL

If the protocol is SNMP, you also need to specify the protocol. Additionally, you need to specify the host name or IP address of the HMC, and the community name under which all operations are performed. The HMC host address is specified using the configuration key Topology.Address.

The community name is specified using the key Topology.Community. The

Logical Processor Scope Definition in Policy

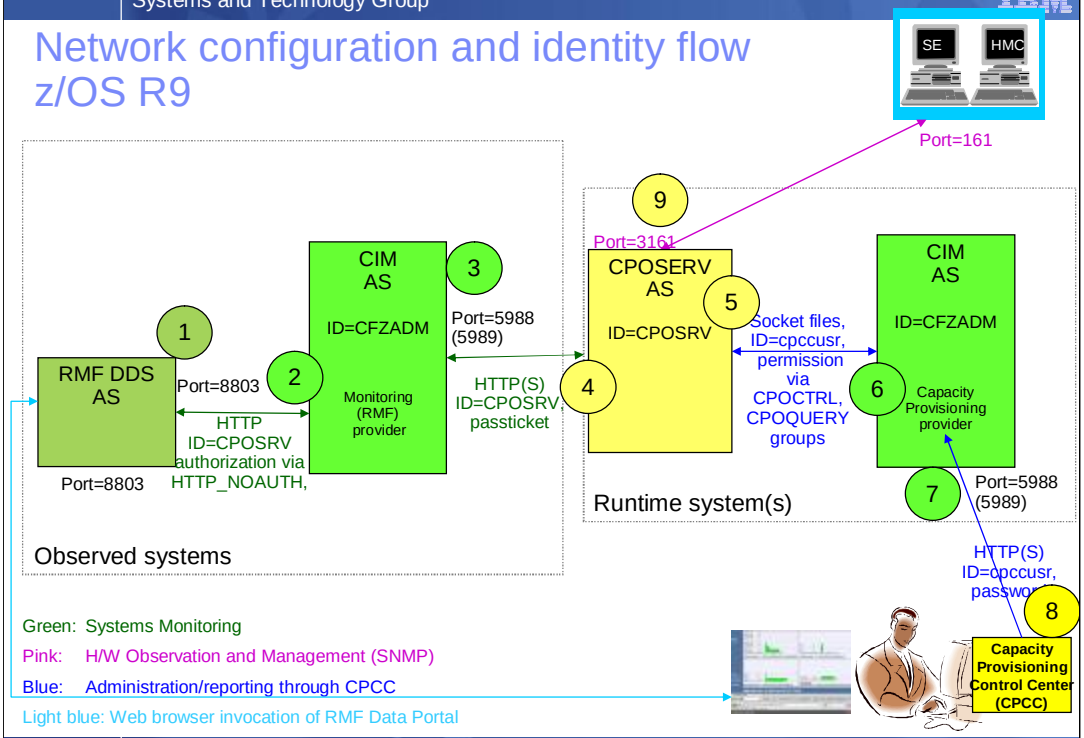
- Definition on same level as maximum provisioning scope
  - No “tight” management of logical processors.
    - LCPs configured on- or offline only when required to absorb physical capacity or to allow for deactivation
    - Use HiperDispatch or IRD Vary CPU management to optimize logical to physical ratio
    - Can only be used with shared LPARs



Coexistence Considerations: z/OS R9, R10, R11

- Initial functionality provided with OA20824 on z/OS R9
- z/OS R10 **GA level** fully compatible with R9
- z/OS R10 new function APAR OA24945 offers two new functions:
  1) Logical processor scope in policy
     - Must only be defined when all observed and runtime systems support this function
     - ++HOLD(DOWNLD): New CPCC must be downloaded and installed to define logical processor scopes
  2) BCPii setup (Topology.protocol=INTERNAL in <hlq>.<domain>.PARM(PARM) member
     - Must only be activated when all runtime systems support this function
- z/OS R11 **GA level** compatible with R10 plus OA24945
  - I.e. also compatible with z/OS R9 if those two new functions are not exploited
- For z/OS R9, R10, and R11 OA29173 allows for shorter provisioning durations
  - Must only be defined when all runtime systems support this function

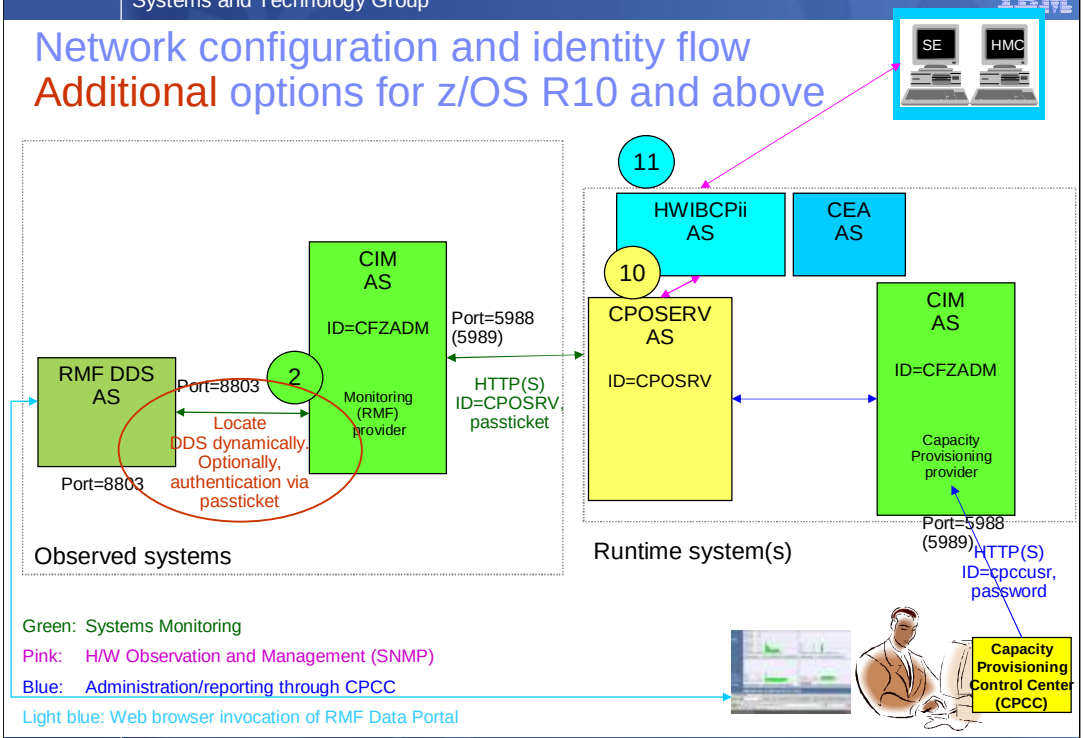
Network configuration and identity flow z/OS R9



Networking Configuration Controls – z/OS R9

1. GPMSRVxx: HTTP_PORT(), HTTP_NOAUTH()
2. *cimserver.env*: RMF_CIM_HOST, RMF_CIM_PORT
3. 7 *cimserver.conf*: httpPort
   Modify via *cimconfig* command or `MODIFY` console command
4. Host address and port in CPM domain configuration
5. CPM PARM member: CIM.ReadGroup, CIM.ModifyGroup
6. */etc/cpoprovider.properties*
8. Host address and port in CPCC “Provisioning Manager” panel
9. CPM PARM member: Topology.Address, Topology.Community

Network configuration and identity flow
Additional options for z/OS R10 and above



Green: Systems Monitoring

Pink: H/W Observation and Management (SNMP)

Blue: Administration/reporting through CPCC

Light blue: Web browser invocation of RMF Data Portal

Networking Configuration Controls
Additional options for z/OS R10 and above

2 — *cimserver.env*: RMF_CIM_HOST, RMF_CIM_PORT no longer required.

GPMSRVxx: HTTP_NOAUTH() no longer required. Use security profile GPMSERVE in PTKTDATA class and others to allow for passticket generation.

10 — CPM PARM member: Topology.Protocol = INTERNAL specifies that BCPii is to be used

11 — Security profile HWI.TARGET.net id.name in the FACILITY class. APPLDATA defines community name to be used.

APAR - II14385

CONSIDERATIONS FOR RMF DISTRIBUTED DATA SERVER (DDS) SETUP IN A CIM ENVIRONMENT

Beginning with z/OS R10 the CIM Monitoring providers can automatically locate an active RMF Distributed Data Server (DDS) in the Sysplex. When the DDS gets restarted on different systems through RMF DDS management, or through manual action, the CIM monitoring providers can connect to an active DDS without additional configuration. For more information on the RMF-managed DDS refer to "Starting the Distributed Data Server" in the z/OS RMF User's Guide.

.

Note: If you are running systems prior to z/OS R10 in your Sysplex you have to start the DDS manually on one particular system and use the environment variables RMF_CIM_HOST and RMF_CIM_PORT to specify the location of the DDS in order for the CIM Monitoring providers being able to connect. A more flexible solution, where the DDS can run on any system in the Sysplex, can be established by using Dynamic Virtual IP Address support (DVIPA). For more information on the setup of DVIPA refer to the z/OS Communications Server IP Configuration Guide.
Prior to z/OS R10 the CIM Monitoring providers cannot authenticate themselves to the DDS. This requires the use of the HTTP_NOAUTH option in the active GPMSRVxx parmlib member to disable authentication for a specific IP address, a group of IP addresses using wildcards, or entirely.
Example:

Overview on Observed Metrics

CPC Metrics

- Per processor type
  - Shared physical utilization
  - Total logical processors
  - Total weights
  - Physical processors

LPAR/System Metrics

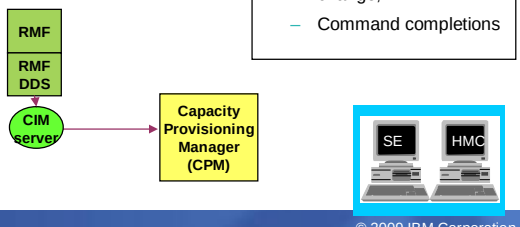
- WLM Service definition, policy, service classes Shared/dedicated
- Initial capping
- Defined capacity
- Capping
- 4h Rolling Average
- Dispatchable units (InR Queue)
- IRD weight & vary CPU mgmt
- Per processor type
  - Online CPs, zAAPs, zIIPs
  - Reserved processors
  - LPAR weight
  - MVS utilization
  - LPAR utilization

Service Class Period Metrics

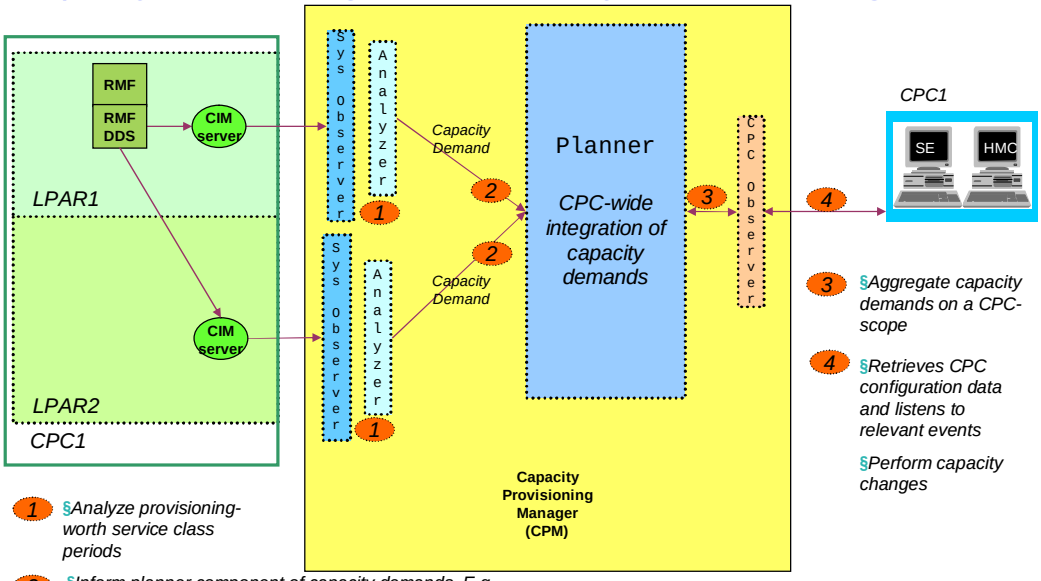
- Local PI
- Sysplex PI
- %Capped
- Delays
- Per processor type
  - Processor delays
  - TCB, SRB

Support Element

- H/W model
- Spare processors
- Installed OOCoD records
- Current S/W model
- Current number of processors
- For OOCoD record
  - Activation limits, validity,
  - Current activation level
- Event subscriptions
  - Capacity. accounting change, …
  - Command completions



Capacity Provisioning Workload Analysis and Planning



1 §Analyze provisioning-worth service class periods

2 §Inform planner component of capacity demands. E.g.

```
Pescp[Identifier[SVPLEX7/R76/basepol/BASEPOL/CPMCPU/1]]
registers Capacity Demand with Provisioning Types
[SpeedStep] and State CapacityDemandState.ACTIVE
```

3 §Aggregate capacity demands on a CPC-scope

4 §Retrieves CPC configuration data and listens to relevant events

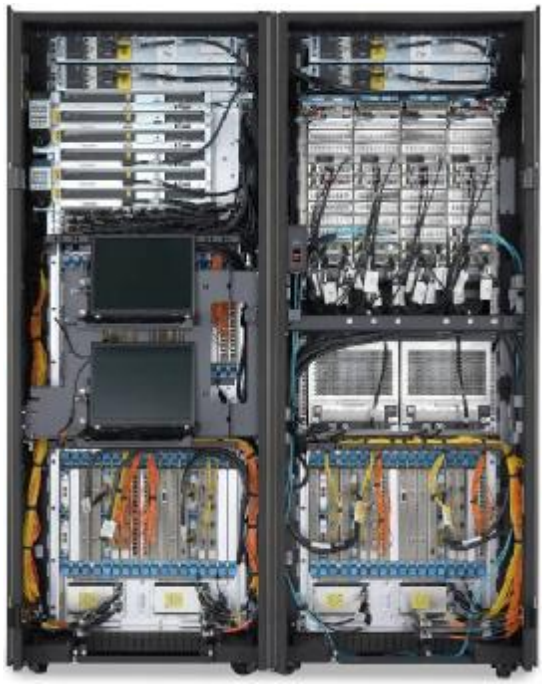
§Perform capacity changes

Interaction with Manual Activations

- Capacity Provisioning tolerates “manual” capacity changes
  - Initiated at HMC/SE, through CPM, or through other automation products (Tivoli System Automation, GDPS)
  - CBU activations are “mostly” ignored
    - CBU can be activated concurrently
    - If OOCoD resources are required to complete “Force” mode CBU activations then CPM prompts to release CBU resources (CPO3034W)
  - On/Off CoD: Only one OOCoD record may be active at a time
    - Because CPM tries to honor concurrent manual actions, such activations reduce the capacity navigation scope of CPM.
    - Messages inform about effects. E.g.
      - CPO4121I Some temporary resources were already active when starting managing the CPC ECL2. Only resources exceeding 2 CP, capacity level 0, 0 zAAP and 0 zIIP will be managed by the Provisioning Manager
      - CPO4101W Manual intervention detected for CPC H87. Continue managing model 404 (0/0) with 4 zAAPs and 0 zIIPs
      - CPO4105I A change of the manually activated resources has been detected for CPC H05. All resources of the defined On/Off CoD record are now managed by the Provisioning Manager

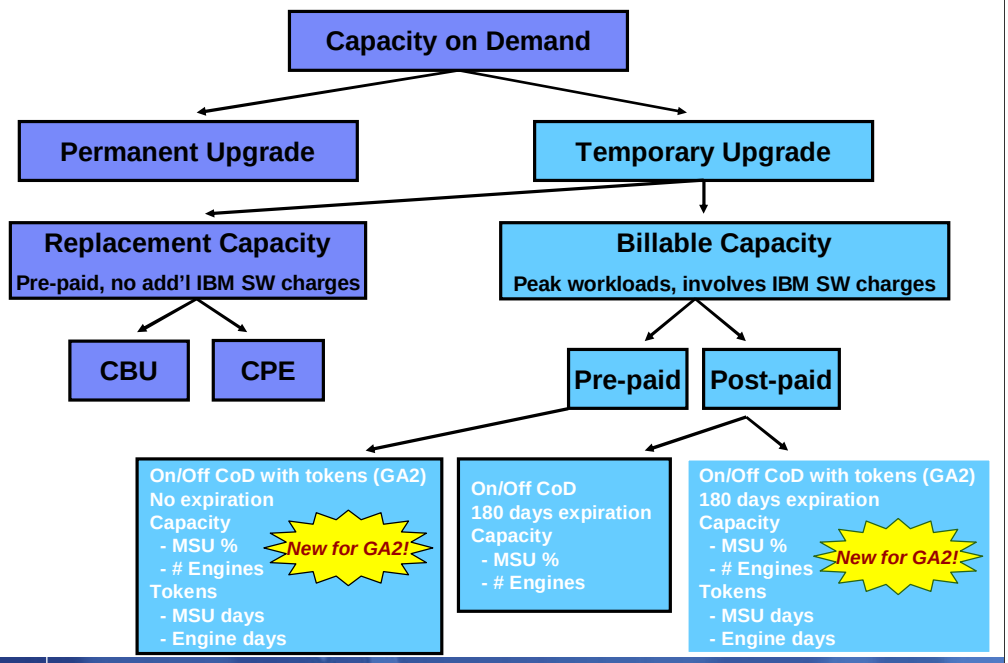
This chart provides more detail on the CPM treatment of capacity what has been activated manually.

Summary of IBM System z10 Temporary Capacity Enhancements



The following charts present a level set on the Capacity on Demand enhancements for IBM System z10, compared to earlier servers.

z10 – Basics of CoD



Here we see the different flavors of Capacity on Demand with focus on On/Off Capacity on Demand, including the enhancements for IBM System z10 EC (GA2) and z10 BC.

z10 On/Off CoD Overview

- Temporary CP capacity up to 100% of purchased capacity
- Number of temporary zIIPs or zAAPs <= total number of permanent + temporary CPs
- Number of temporary IFLs up to the total of purchased IFLs
- Number of temporary ICFs plus permanent ICFs <= 16

- Number of real activations = unlimited, with unlimited duration

- Post paid Offering
  - record expires after 180 days
  - Optional: set spending limits on the offering
- Pre-paid Offering
  - No expiration date

- Tokens to control pre-paid offering or to set spending limits on post-paid offering
  - 'telephone card'
  - Tokens are reduced based on highest usage within a 24 hour period
  - 'MSU day' tokens for CPs and 'processor day' tokens for specialty engines

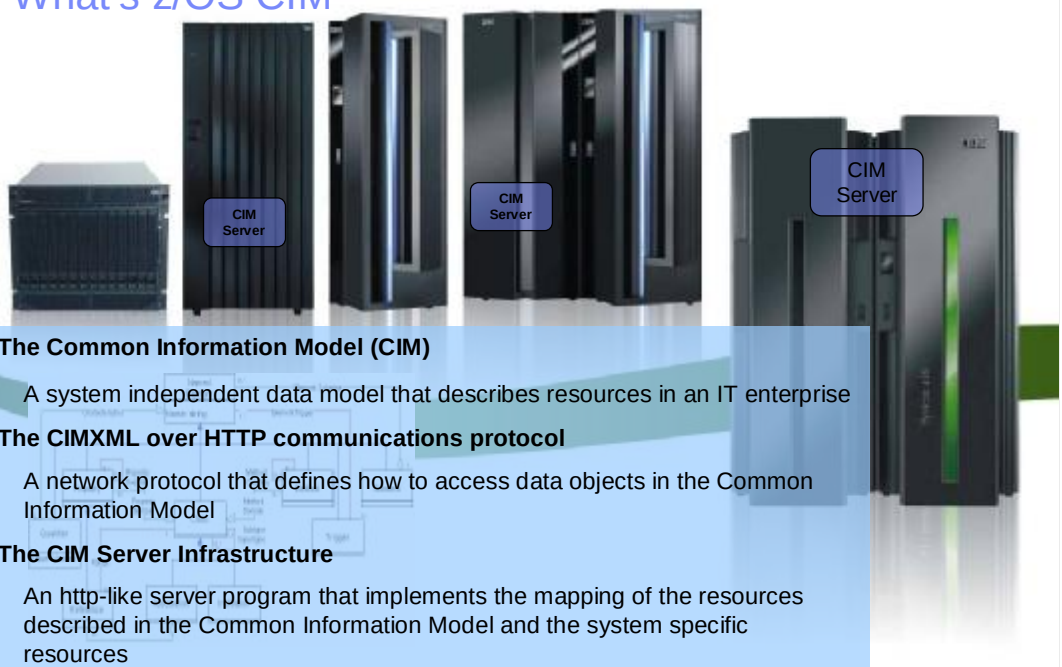
Temporary Capacity – a new approach on System z10

- All temporary capacity offering records are resident on machine
  - No connection or passwords required at time of activation
- Multiple records can be simultaneously active (eg CBU with OOCoD)
  - Each has independent controls and policy
  - Each can be activated / deactivated in any sequence
- Individual record can be used to temporarily reach multiple configurations
  - Resources can be activated in any amount up to defined limit
  - Customer can customize activation real-time, based on circumstances
  - Eliminates unique record to be managed for all possible permutations
  - (i.e. multiple use for a single On/Off CoD record, even during a permanent upgrade)
  - Dynamic changes in activation level without reloading records
- Various record limits can be dynamically updated / replenished
  - Changes possible even if record is currently active

Temporary Capacity – a new approach on System z10

- As records expire or are consumed, the resources will be deactivated
  - System will not reduce to sub-capacity when records expire
  - Will not deactivate if removing dedicated engines or last of that engine type
- Ability to perform permanent upgrades while temporary capacity is active
  - Allows quick conversion of temporary capacity to permanent
  - Modification of record entitlement performed dynamically and concurrently
- API enhancements to support use by Capacity Provisioning Manager
  - The z10 HMC/SE Application Programming Interfaces provide the following functions:
    - Commands to query the capacity records objects information (list of installed records and their status, list of possible target configurations)
    - Commands to add and remove temporary capacity for defined CPC objects
    - Event notifications for temporary capacity changes
  - Capacity Provisioning Manager provides policy based advice and automation

What's z/OS CIM



The Common Information Model (CIM)

A system independent data model that describes resources in an IT enterprise

The CIMXML over HTTP communications protocol

A network protocol that defines how to access data objects in the Common Information Model

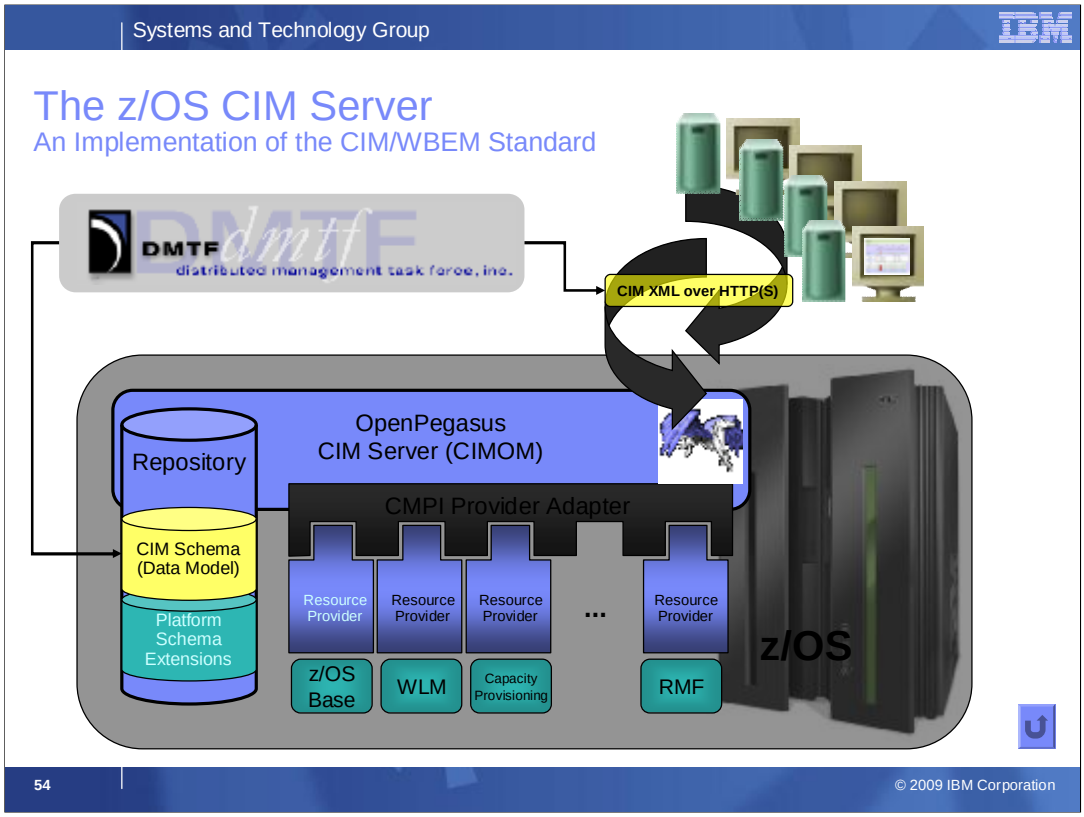
The CIM Server Infrastructure

An http-like server program that implements the mapping of the resources described in the Common Information Model and the system specific resources

For mastering the challenge to manage complex IT environments, that consist of software and hardware from many different vendors, each using its own set of technologies, a standard way is needed to describe the resources to be managed and to define operations that can be executed on these resources.

By implementing the Common Information Model (CIM), the z/OS Operating System on the mainframe supports an open standards based interface for management over the network. This enables z/OS for a new generation of management applications that provide modern, web based, user interfaces. Management applications that are specialized for the aspects of managing mainframes, as well as management applications that provide inventory and status information for distributed enterprise environments and business applications.

The development of such new management applications is accellerated by eliminating the need to talk multiple different protocols for different systems or even having to develop specific agents that run on the managed systems and facilitate remote access. Instead the management applications talk to the CIM Server that comes as part of z/OS as well as of most other popular operating systems, storage or network devices. For talking to a CIM Server it uses the Web Based Enterprise Management (WBEM) network protocol and the data model defined by the CIM Schema, which is the core of the Common Information Model.



The OpenPegasus CIM Server, also referred to as CIM Object Manager (CIMOM), is an implementation of the WBEM standards provided by DMTF. It is developed as an OpenSource project that runs on a large number of Operating Systems, including IBM's z/OS.

It comes with an XML based data repository where it stores the CIM Schema classes and all platform specific extensions of the Schema.

While the CIM Server has knowledge only about the CIM Schema Classes and extensions, that task for connecting these classes to actual IT resources is delegated to so called 'Dynamic Resource Providers' which can be registered for the CIM classes known by the CIM Server. When the CIM Server receives a request (CIM Operation) directed against a certain CIM class it looks up the list of applicable providers for that class and all of its subclasses and then forwards the request to each of the providers. A provider is a program library that implements a standard interface through which it plugs into the CIM Server on the one hand, and which implements the access to a resource that maps to a CIM class describing this resource.

The interface used for the providers on z/OS is called the Common Manageability Programming Interface (CMPI) and is also based on a standard publish by TheOpenGroup. It guarantees binary compatibility between the providers and the CIM Server and thus enables providers to be supplied from different vendors, independent of the CIM Server.

The CIM Server at its front end works like an HTTP server. By default it listens on the network ports 5988 (for plain HTTP) and 5989 (for secure HTTP) to receive requests from management applications over the CIM XML over HTTP protocol.