

Glenn Anderson, IBM Lab Services and Training



## Understanding z/OSMF for the Performance Management Sysprog



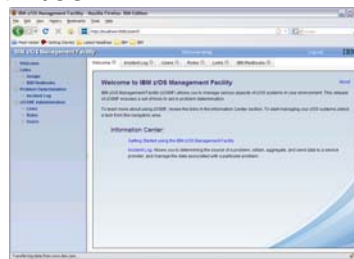
Winter SHARE  
March 2014  
Session 55220



## z/OSMF: the z/OS Management Facility



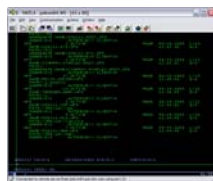
- **z/OSMF is a new product for z/OS customers and provides a modern browser based interface to managing the z/OS system.**
- **z/OSMF helps system programmers to more easily manage and administer a mainframe system by simplifying day to day operations and administration of a z/OS system.**
- **IBM z/OS Management facility (z/OSMF) delivers on IBM's strategy for mainframe simplification and modernization**
- **z/OSMF is a companion product to z/OS, offered at no additional charge**  
**z/OSMF 1.11 was the first release, delivered with z/OS 1.11**
- **z/OSMF has it's own product number**
  - ▶ Product ID for z/OSMF 2.1 is 5610-A01
  - ▶ Service & Subscription ID is 5655-S29
- **Both PIDs must be ordered**
- **It can be ordered in a serverpac with z/OS**
  - ▶ Or as its own product serverpac



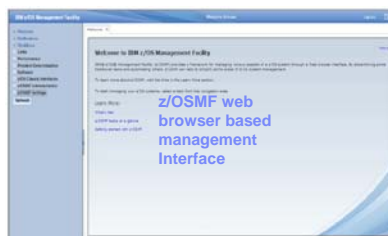
## Why z/OSMF?



- The IBM z/OS® Management Facility provides a Web-browser based management console for z/OS designed to improve productivity, quality and simplify management
- Helps the experienced and not so experienced system programmers more easily manage z/OS, by simplifying day to day operations and administration
- z/OS Management Facility helps automate management tasks
  - Can help reduce the learning curve and improve productivity
  - Helps guide users easily through tasks with embedded user assistance (such as wizards)
  - Helps accelerate productivity, making navigation and task steps more seamless.
  - Makes administration more intuitive



3270 green screen




**z/OSMF is a companion product to z/OS, offered at no additional charge**


## z/OS simplification focus areas

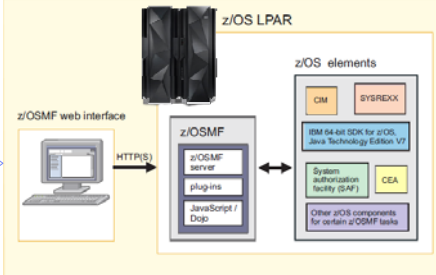


<p><b>Software management, Migration, and Maintenance</b>  <i>Planning, installing, and upgrading z/OS systems and products that run on z/OS</i></p>	<p><b>Configuration</b>  <i>Adding or changing z/OS system components; enabling new features; defining and updating policies that affect system behavior</i></p>	<p><b>Problem and Performance Management</b>  <i>Monitoring z/OS system health; identifying real and potential problems; Analyzing and resolving problems</i></p>
<p><b>Simplify and modernize the System Programmer user experience</b>  <i>Deliver solution in a task-oriented browser based user-interface, with end-to-end task simplification, eliminating opportunity for error</i></p>		

## IBM z/OS Management Facility structure








- **z/OS Management Facility is a Web 2.0 application**
- Uses industry standards (Java™, CIM, DOJO)
- Uses Java; some use z/OS CIM Server – workloads can run on zIIPs, zAAPs.
- New efficient code base: WebSphere Application Server Liberty profile

**Java code is zAAP eligible**

**z/OS CIM Server is eligible for zIIP**

## z/OSMF 2.1 functions



**IBM z/OS Management**

- » Welcome
- » Notifications
- » Workflows
- » **Configuration**
  - » Configuration Assistant
- » **Links**
  - » ShopSeries
  - » Support for z/OS
  - » System z Redbooks
  - » WSC Flashes & Techdocs
  - » z/OS Basics Information Center
  - » z/OS Home Page
  - » z/OS Internet Library
- » **Performance**
  - » Capacity Provisioning
  - » Resource Monitoring
  - » System Status
  - » Workload Management
- » **Problem Determination**
  - » Incident Log
- » **Software**
  - » Software Management
- » **z/OS Classic Interfaces**
  - » ISPF
- » **z/OSMF Administration**
  - » Application Linking Manager
  - » Links
- » **z/OSMF Settings**
  - » FTP Servers
  - » Systems
- » Refresh

- **Notifications and Workflow \*(R2.1)**
- **Configuration category**
  - **Configuration Assistant for z/OS Communication Server** application
  - Simplified configuration and setup of TCP/IP policy-based networking functions
- **Links category**
  - Links to resources - provides common launch point for accessing resources beyond z/OSMF
- **Performance category**
  - **Capacity Provisioning (updated)** manage connections to CPMs, view reports for domain status, active configuration and active policy.
  - **Resource Monitoring, System Status** - provide integrated performance monitoring of customer's enterprise
  - **Workload Manager Policy Editor** application
  - Facilitate the creation and editing of WLM service definitions, installation of WLM service policies, and activation of WLM service policies
- **Problem Determination category**
  - **Incident Log** : provide a consolidated list of SVC Dump related problems, along with details and diagnostic data captured with each incident; facilitate sending the data for further diagnostics.
- **Software category (updated)**
  - **Management**: deployment of installed software simpler and safer, manage service levels and product levels
- **z/OS classic Interface category**
- **ISPF Task** integrate existing ISPF into z/OSMF to enable tasks from single interface and ability to launch to ISPF functions directly
- **z/OSMF Administration category**
  - z/OSMF authorization services for administrator:- dynamically add links to non-z/OSMF resources; application linking manager(R13)
- **z/OSMF Settings category (New!)**
  - Manage FTP destinations and systems

## z/OSMF V2.1 implementation

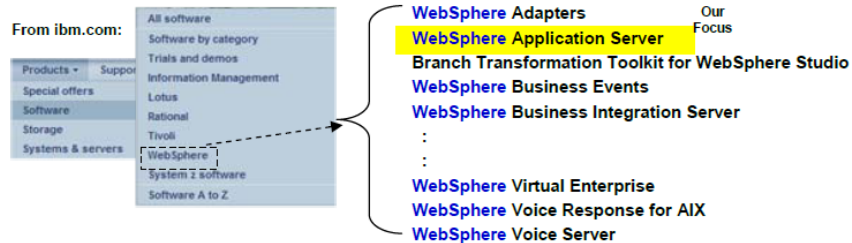


- z/OSMF is rebased on the WebSphere Application Server for z/OS V8.5 Liberty profile
  - ▶ This is expected to provide significant reductions in the resource requirements for z/OSMF
    - The WASOEM FMID is no longer required and the requirement for separate configuration of the runtime is eliminated.
    - Result is reduced footprint size, reduced memory requirement and reduced CPU requirement
  - ▶ z/OSMF setup is simplified
    - Reduced steps to configure z/OSMF
  - ▶ Applying service is easier
  - ▶ Faster startup of application

## WebSphere is a brand: WebSphere Application Server is a product



We'll start by clearing up a point of confusion about the term *WebSphere*.




Probably close to 100 products carry the "WebSphere" brand name.



For many, the term *WebSphere* means *WebSphere Application Server*. Sometimes the acronym *WAS* is also used informally.

## The J2EE application model IBM



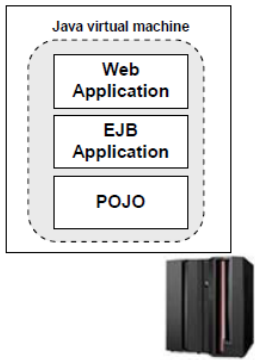
- Components
  - The key focus of application developers; these are the EJBs, Servlets, JSPs, and clients.
  - Many component behaviors can be specified at deployment time, rather than in program code.
- Containers
  - These provide services to components transparently, including transaction support and resource pooling.
  - Containers and connectors conceal complexity and promote portability.
- Connectors
  - These sit under the J2EE platform, defining portable service APIs to plug into existing enterprise vendor offerings.
  - Connectors promote flexibility by enabling a variety of implementations of specific services.

## Different kinds of Java programs IBM

**WebSphere Application Server can host (or “run” or “support”) several different kinds of applications, all written in Java:**

It is okay not to understand the details of these things. It is better at this point just to understand that different kind of programs exist and listen for these terms when others talk about the WebSphere environment.

**WebSphere Application Server**



**Web application**

An application that is accessed with a browser. This typically consists of static files (HTML, JPG/GIF), and Java programs that generate dynamic output:

- **Servlets:** Java program that contains logic to do things like perform calculations, access data, and format a reply
- **JSPs:** stands for Java Server Pages, it's a way to create a dynamic web page that can be populated with dynamic content

**EJB application**

Stands for “Enterprise Java Bean,” it's a more sophisticated application that's intended for high-end applications. Two flavors:

- **Session Beans:** meant to hold the logic of the application
- **Entity Beans:** meant to represent data as an “object”

Many EJB applications are made up of just session beans – easier.

**POJO**

Stands for “Plain Old Java Object.” It is the simplest form of a Java program and lately more people are returning to simplicity. (POJO commonly applies to the EJB 3.0 environment and Java Batch environment).

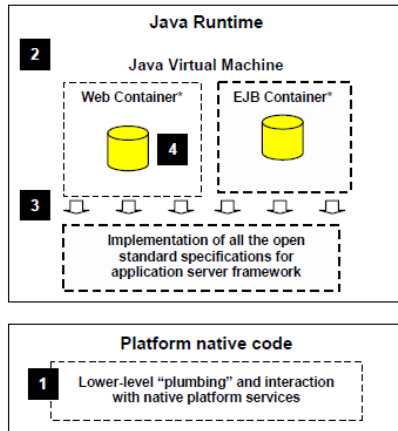
**Java EE**

Too simple a categorization, but okay for now

## Schematic diagram of WebSphere Application Server

Here is a semi-conceptual view of what WebSphere Application Server is:

The real product is of course *far more sophisticated* than this, but this gets the key points across



### 1. Server is started

- On z/OS that is done with a `START` command (more later).
- This native code is what establishes the lower-level "plumbing" and allows for the invocation of the Java environment.

### 2. Java runtime established, including JVM

- When the native base is ready, it establishes Java runtime environment and launches the JVM.

### 3. WebSphere Java components loaded into JVM

- With the JVM launched, WebSphere Application Server can now load the Java components that make up the Java EE environment.
- This is the "framework" we mentioned earlier.
- This is why WebSphere Application Server is more than just a JVM.

### 4. Your applications are loaded and started

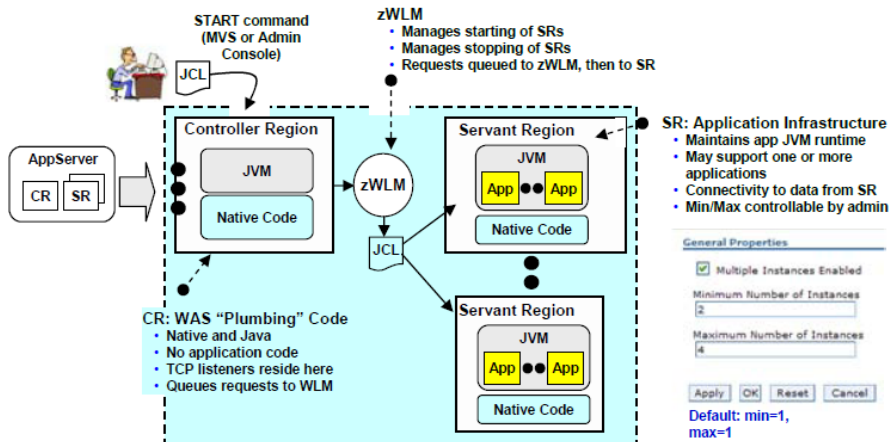
- If they are deployed in the server and configured to start automatically, WebSphere will do that for you.

Now we are ready to see how this is implemented on z/OS.

\* "Containers" are just logical software constructs inside the JVM that provide services specific to the type of application that runs in them. "Web Container" is for web applications; "EJB Container" is for EJBs.

## A peek inside the application server architecture

We see that inside our little curved-box picture of the Application Server resides two or more address spaces as well as integration with zWLM:



This is a built-in "vertical scaling" mechanism. It also allows for redundancy of application JVM to prevent single point of failure.

**IBM**

## Basics of accessing web applications

http://www.host.com/PolicyIVP/PolicyServlet

The diagram illustrates the flow of an HTTP request. On the left, a client labeled 'Was70lvp' is shown sending a request to a server. The URL 'http://www.host.com/PolicyIVP/PolicyServlet' is displayed above the client. The request travels through the network, labeled 'HTTP PROTOCOL', to the server. On the server side, the request is first intercepted by the 'HTTP Protocol "Catcher"' (labeled 1). This catcher then passes the request to the 'Servlet Engine' (labeled 2), which contains the 'Was70lvp' application (labeled 3). The server is represented by a rack of hardware.

- Three key steps:
  - A *protocol catcher* takes HTTP protocol off network and interprets its meaning.
  - Based on the format of the URL, the request must be directed to the correct application server to execute.
  - The web container must identify which web application and servlet to execute and then use the servlet engine to run the servlet.

**IBM**

## WAS z/OS V8.5 overview

With WAS z/OS V8.5 we now have two server models to choose from:

**Traditional Multi-JVM Model**  
"Application Server"

The diagram shows a 'Controller Region' box on the left, a yellow circle labeled 'WLM' in the center, and a stack of boxes labeled 'Servant Regions' on the right. All three are enclosed in a dashed-line box.

- Two or more JVMs make up an application server instance
- CR does the request handling, SR hosts the applications
- Full Java EE server runtime
- Administration through DMGR and Admin Console as seen in Unit 2
- Includes "Granular RAS" function which we'll explore in this unit

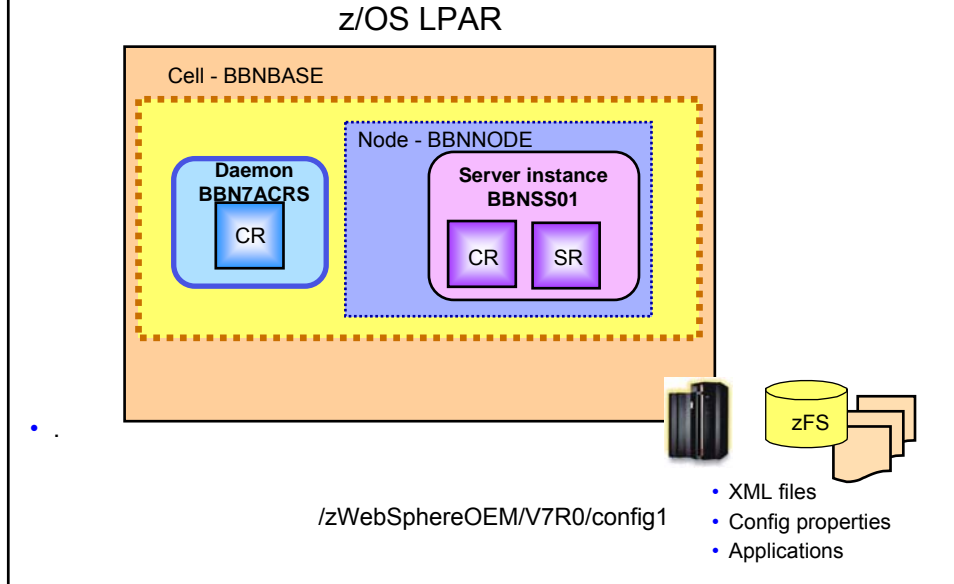
**Liberty Profile Model**  
"Application Server"

The diagram shows a single box labeled 'Liberty Profile Server Instance'.

- One JVM makes up an application server instance
- Lightweight, composable and dynamic updates
- Web applications at this time
- Simple configuration and administrative model
- Not part of the traditional WAS cell or administrative model



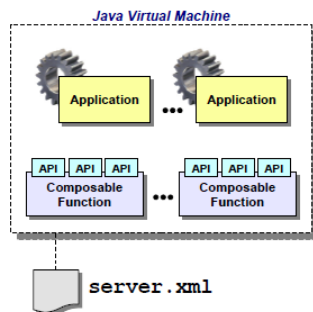
## WAS OEM stand-alone server node (z/OSMF 1.13)



## Overview of Liberty Profile



The Liberty Profile is designed to be a single-JVM server model that is lightweight, composable and dynamic:



- **Composable** -- you configure the function the application needs; you don't need to load up everything
- **Dynamic** -- changes to configuration or changes to applications detected and dynamically enabled
- **Subset of traditional WAS function**  
Liberty is not full Java EE, traditional WAS is
- **Upwards application compatibility**  
Apps that run in Liberty will run in traditional WAS ... but not necessarily the other way around since Liberty is subset of traditional WAS
- **Each server is one JVM**
- **Run from UNIX shell or as started task**
- **One required configuration file: server.xml**
- **Not part of traditional WAS administrative DMGR, federated node model**  
But there is an ability to manage via the "Job Manager" function of traditional WAS (advanced topic, we won't get into that here)



## z/OSMF V2.1 implementation

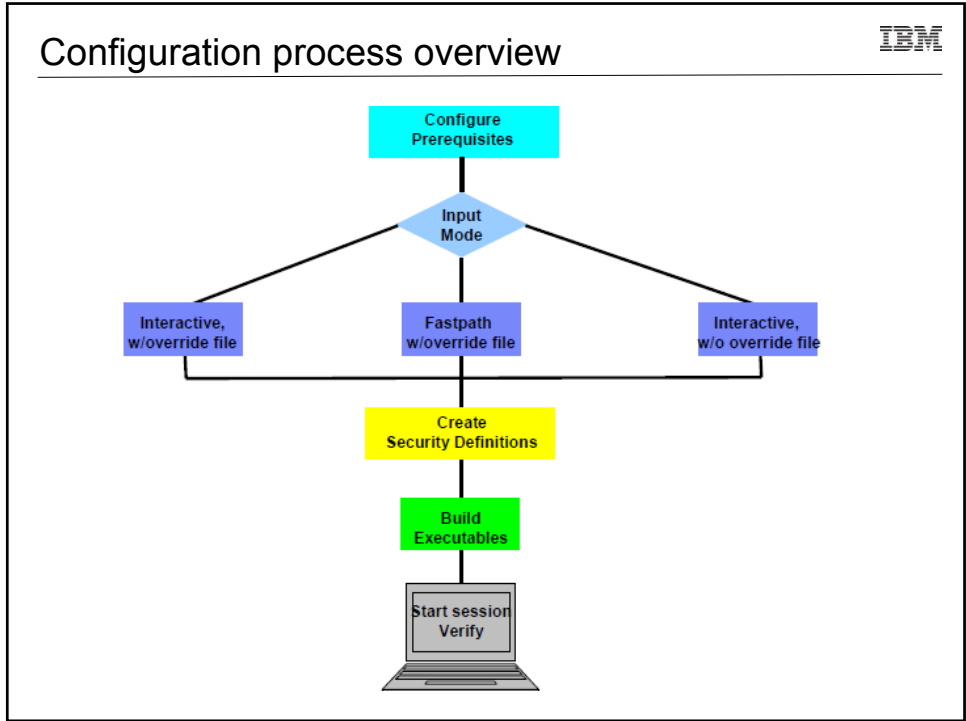


- **z/OSMF is rebased on the WebSphere Application Server for z/OS V8.5 Liberty profile**
  - ▶ This is expected to provide significant reductions in the resource requirements for z/OSMF
    - The WASOEM FMID is no longer required and the requirement for separate configuration of the runtime is eliminated.
    - Result is reduced footprint size, reduced memory requirement and reduced CPU requirement
  - ▶ z/OSMF setup is simplified
    - Reduced steps to configure z/OSMF
  - ▶ Applying service is easier
  - ▶ Faster startup of application

## z/OSMF V2.1 FMIDs



- **z/OSMF V2.1 consists of nine (9) FMIDs:**
  - HSMA210 - z/OS Management Facility core
  - HSMA211 - z/OSMF ISPF
  - HSMA212 - z/OSMF Resource Monitoring
  - HSMA213 - z/OSMF WLM
  - HSMA214 – z/OSMF Software Deployment (really Software Management)
  - HSMA215 - z/OSMF Incident Log
  - HSMA216 - z/OSMF Capacity Provisioning
  - HSMA217 – z/OSMF Workflow
  - HSMA21A - z/OSMF Configuration Assistant



## System Status and Resource Monitoring

- System Status is an RMF based workload reporting application that provides a performance status of sysplexes
- The status indicator informs you whether or not workloads are meeting WLM goals
- You can Monitor z/OS and Linux workloads and view workload resource metrics

The System Status task links to WLM (e.g. linking to service definitions and active WLM policies), and the WLM application links to the System Status task.

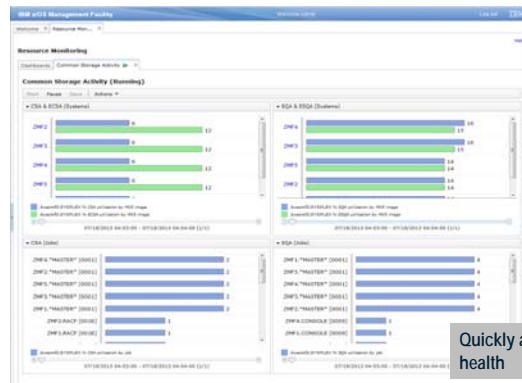
**System Status** screenshot showing a table of resources with columns: Resource, System Type, Connectivity, Performance Index Status, Related Service Definition File, and Active WLM Policy File. A callout points to the Performance Index Status column: "Performance Index Status column for a quick indicator of sysplex health".

**Resource Monitoring** screenshot showing a navigation menu on the left and a list of dashboards on the right. A callout points to the Resource Monitoring link: "Resource Monitoring can link to the Workload Management task. Easily view the corresponding WLM service definitions to manage performance".

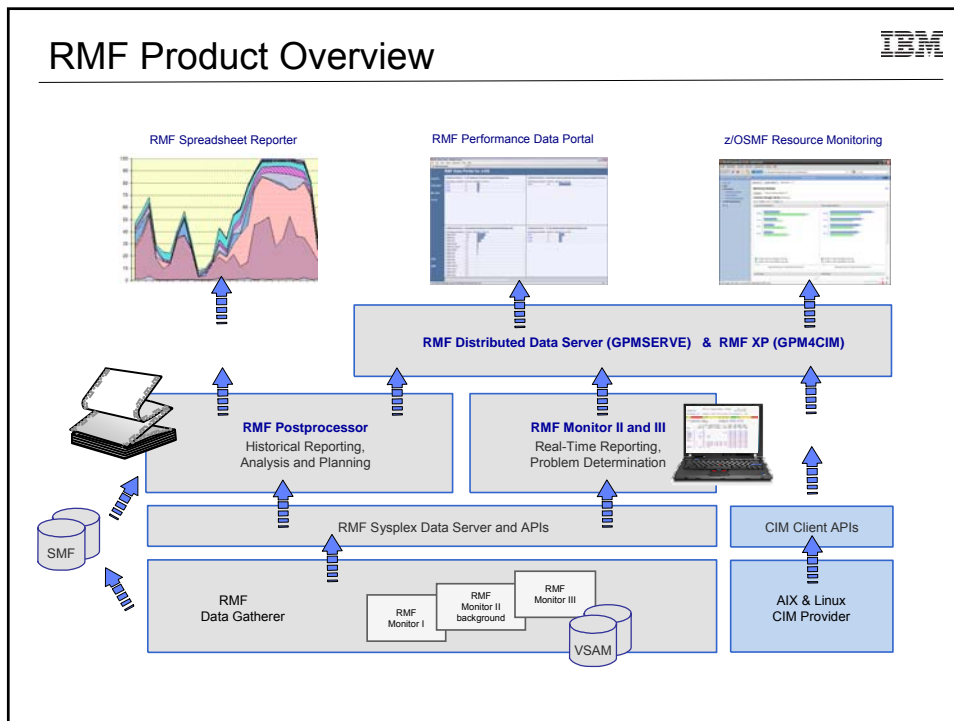
## Resource Monitoring



- Provide real time status on systems or sysplexes defined to system status task
- Monitor most RMF Monitor III metrics, create and save custom views, and display real-time performance data as bar charts.
- Use the RMF XP capabilities to allow you to monitor the zBX, combining metrics on a customizable dashboard to view the health of the zEnterprise Ensemble
- Define and customize monitoring dashboards to focus more precisely on specific workloads
- Link to view appropriate WLM Service definitions



## RMF Product Overview



## z/OSMF Resource Monitoring IBM

*Infrastructure*

- Browser connects to z/OSMF
- z/OSMF Resource Monitoring can connect to all systems where the RMF Distributed Data Server (DDS) is running
- DDS retrieves z/OS performance data from all images of the sysplex where the RMF Monitor III gatherer is active

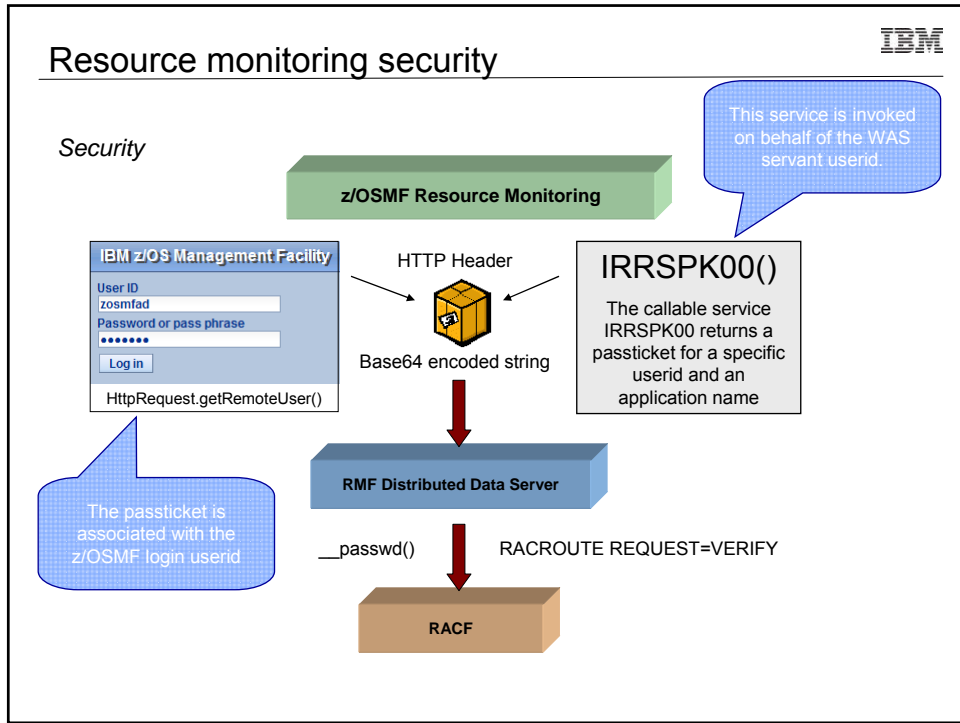
## z/OSMF Resource Monitoring IBM

*The Resource Model*

- The Sysplex is the top-level resource
- The resource identifier consists of three parts:
  - High level qualifier
  - Resource Name
  - Resource Type

```

SYSDPLEX,SYSPLEX
├── ,SYSF,MVS_IMAGE
│   ├── SYSF,*,I/O_SUBSYSTEM
│   ├── SYSF,*,ALL_SSIDS
│   ├── SYSF,*,ALL_LCUS
│   ├── SYSF,*,ALL_CHANNELS
│   ├── SYSF,*,ALL_VOLUMES
│   └── SYSF,*,ZFS
├── ,SYSF,*,PROCESSOR
│   ├── SYSF,*,STORAGE
│   ├── SYSF,*,ENQUEUE
│   ├── SYSF,*,OPERATOR
│   └── SYSF,*,SW_SUBSYSTEMS
├── ,SYSE,MVS_IMAGE
├── ,SYSD,MVS_IMAGE
├── ,CF01,COUPLING_FACILITY
├── ,CF02,COUPLING_FACILITY
└── ,70B82,CPC
    
```



## Resource Monitoring – Predefined Dashboards

**IBM z/OS Management Facility**

Welcome | Resource Mon... | Log out

**Resource Monitoring**

**Dashboards**

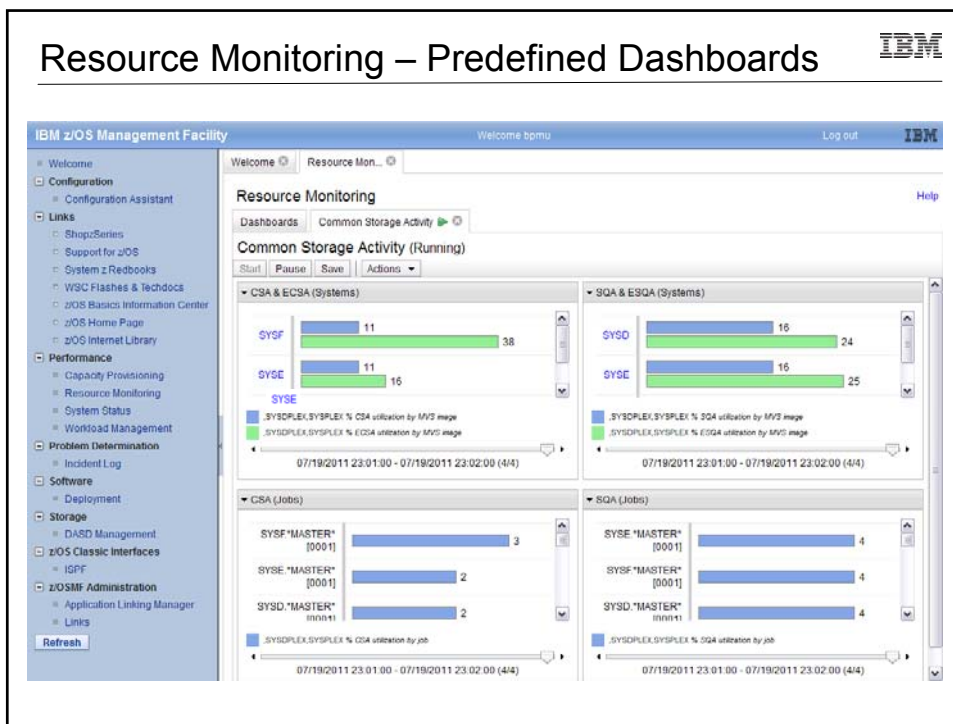
Actions

Name
<input type="radio"/> Common Storage Activity
<input type="radio"/> Coupling Facility Overview
<input type="radio"/> Execution Velocity
<input type="radio"/> General Activity
<input type="radio"/> Overall Image Activity
<input type="radio"/> Performance Index
<input type="radio"/> Response Time
<input type="radio"/> Using & Delays
<input type="radio"/> XCF Activity

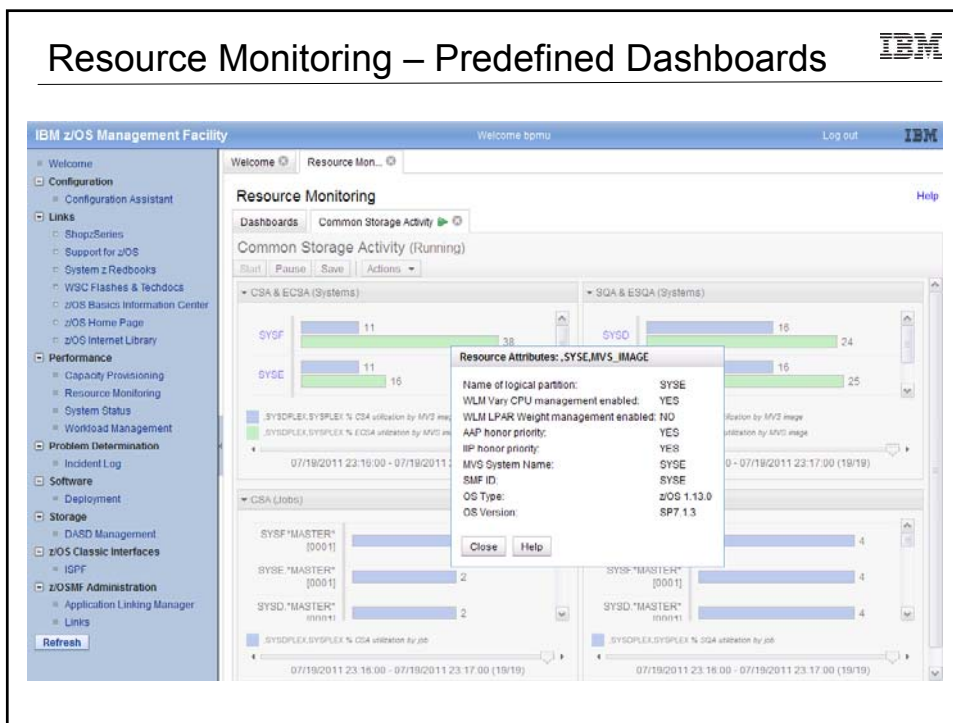
Total: 9

Refresh Last refresh: Jul 19, 2011 10:29:27 PM local time (Jul 19, 2011 8:29:27 PM GMT)

## Resource Monitoring – Predefined Dashboards



## Resource Monitoring – Predefined Dashboards



## Resource Monitoring – New Dashboard



**Resource Monitoring**

Available resources:

- LOCALPLEX.SYSPLEX
- SYSDEX.SYSPLEX
- SYSD.MVS.IMAGE
- SYSEMVS.IMAGE
- SYSE.MVS.IMAGE
- SYSE.\*.WO.SUBSYSTEM
  - SYSE.\*.PROCESSOR
  - SYSE.\*.STORAGE
  - SYSE.\*.ENQUELUE
  - SYSE.\*.OPERATOR
  - SYSE.\*.SW.SUBSYSTEMS
- CF01.COUPLING.FACILITY
- CF02.COUPLING.FACILITY
- .70B2.CPC
- SCLMPLEX.SYSPLEX

## Resource Monitoring – New Dashboard



**New Dashboard (Running)**

Resource	Frames Active
RMFCAT [0040]	48874
RMF [001F]	26349
GPM4CIMZ [00A0]	14189
DFS2FS [0028]	12336
ESCM [0097]	10119
GPMGRVLU [0095]	9005
GPMSERVE [00AA]	8953
GPM4CIMA [00A8]	7897



SYSE.\*.STORAGE # frames active by job (frames[SYSSTC.3])

07/20/2011 00:20:00 - 07/20/2011 00:21:00 (3/3)

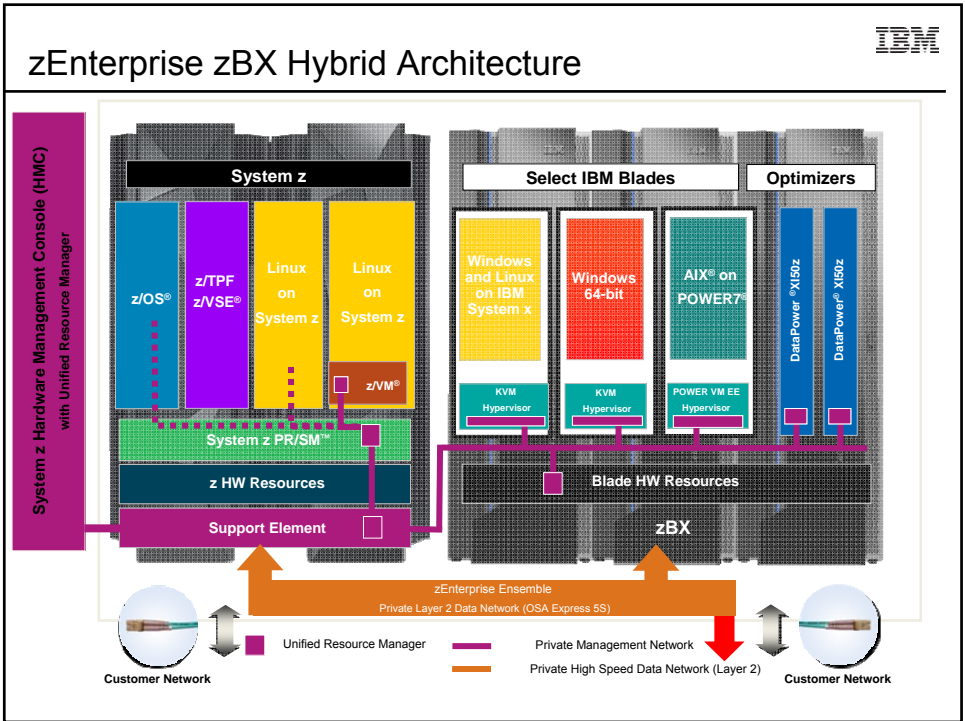


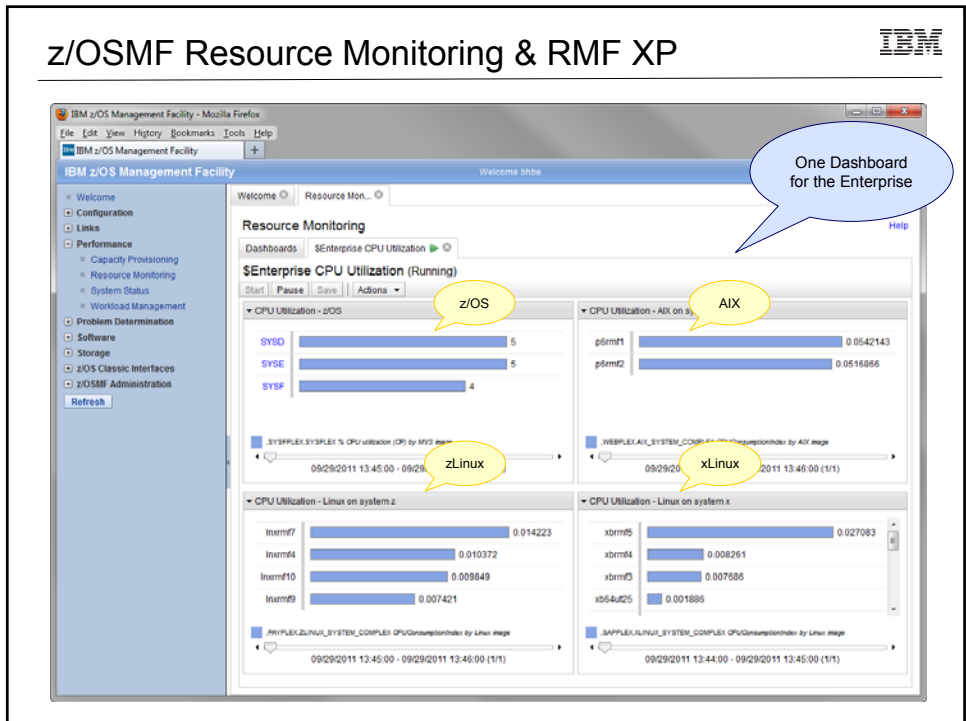
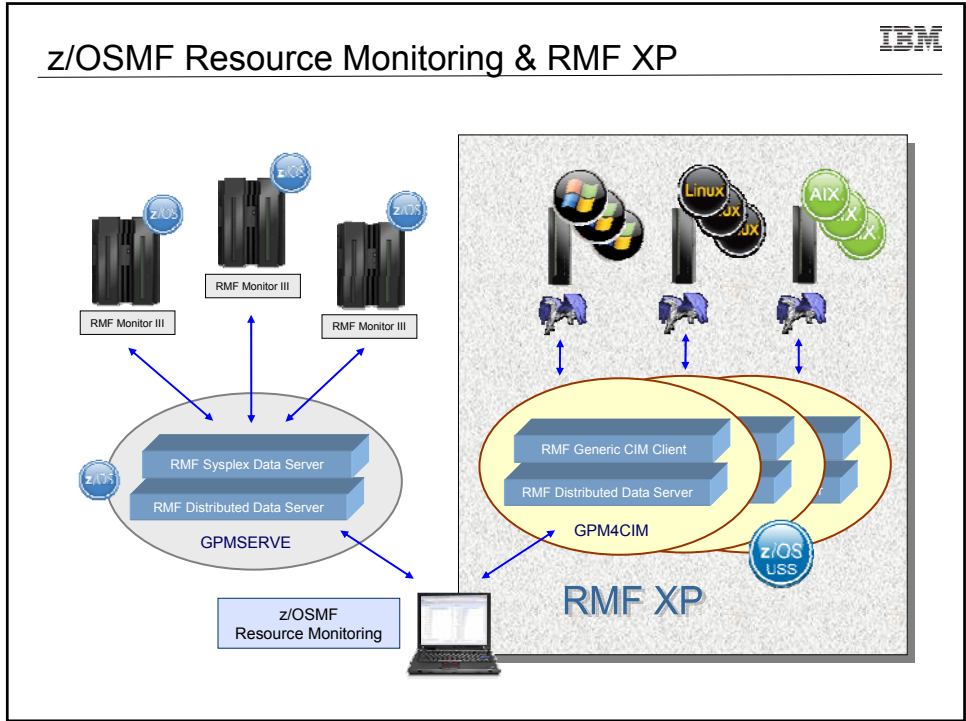
## z/OSMF Resource Monitoring & RMF XP IBM

- RMF XP is the solution for **C**ross **P**latform Performance Monitoring
- RMF XP supports the Operating Systems running on
  - x Blades
  - p Blades

- In addition RMF XP supports Linux on System z
  - LPAR Mode
  - VM Guest Mode

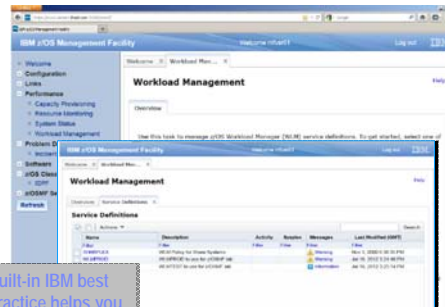




## z/OSMF V2.1 Workload Management



- Simplify creation, modification and review of z/OS Workload manager service definitions
- Accelerate the time to establish service definitions with built in best practices
- Easily install/extract service definitions and edit them
- Activate service policies and monitor WLM status
- View performance data for the currently active service classes, service policies, and service definition with the Resource Monitoring plug-in
- Link automatically to Resource Monitoring and System Status



Built-in IBM best practice helps you avoid errors

Use resource monitoring to monitor active policies

The Resource Monitoring application links to Workload Management in context (& vice versa)

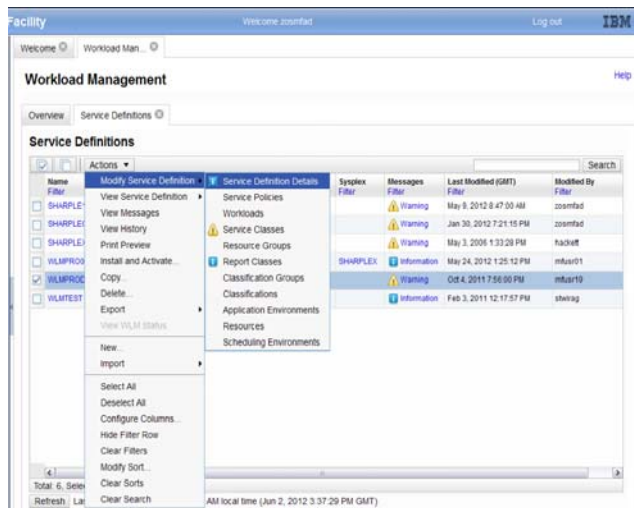
*Test show it is 10X Faster to Review and Update WLM Policies with z/OSMF than through typical manual means*

All performance information was determined in a controlled environment. Actual results may vary. Performance information is provided 'AS IS' and no warranties or guarantees are expressed or implied by IBM

## z/OSMF Workload Management



- Integrates repository to store service definitions
- Import and export of service definitions in XML format
- Printing of service definitions
- Creation, editing, reviewing of service definitions in tabular format
- Direct navigation between policy elements during editing/viewing of service definitions
- Best-practice checking for service definitions
- Supports the installation and the activation of service policies
- Displays WLM status of systems in sysplex
- Different authorization levels for viewing, modifying and installing service definition



z/OSMF Workload Management – Some Benefits		IBM
	Without WLM Policy Editor** in z/OSMF using WLM Administrative Application	With WLM Policy Editor** in z/OSMF
Optimization of a service definition based on best-practices	Read through WLM-related manuals and identify best-practices. Print out the service definition and investigate it with respect to proposed best-practices. If required, modify the policy elements correspondingly.  <b>Hours (or days when done initially)</b>	Check the <u>best-practice hints</u> the GUI displays for policy elements. If required, modify the policy elements correspondingly.  <b>Minutes (or hours when done initially)</b>
Review of service definitions for daily changes, migration, consolidation	To get an overview of a service definition you have to print it to a data set, download the data set, and print it out or feed it into the <i>Service Definition Formatter</i> tool to filter and sort policy elements.  <b>5-10 minutes until review can start</b>	Open a service definition from the service definition repository. Navigate through it using links. <u>Filter and sort</u> policy elements in the tables.  <b>Seconds until review can start</b>
Transfer policy elements from a test service definition to a production service definition	Print out the test service definition and update the production service definition by typing in the changes.  <b>Up to several minutes per policy element</b>	Open the test and production service definition simultaneously and copy over the changed policy elements <u>via copy&amp;paste operations</u> .  <b>Seconds per policy element</b>

\*\* Based on IBM laboratory results, your results may vary

## Service Definition Editing

- Simplified creation, modification and review of service definitions
  - Policy elements are presented in tables
  - Tables can be filtered and sorted
  - Direct editing of policy elements within tables
  - Best-practice hints are displayed automatically while specifying policy elements
  - Several service definitions can be opened simultaneously
  - Cut, Copy, Paste of policy elements between service definitions

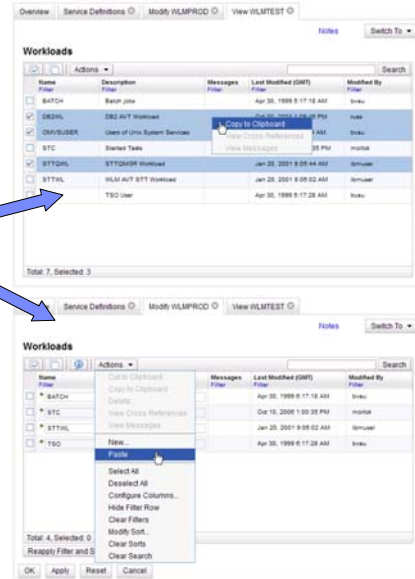
The screenshot shows the IBM z/OSMF Workload Management interface. A table titled 'Service Classes' is displayed with columns for Name, Priority, Period, Importance, Duration, Error, Goal Type, Response Time Goal, Percentage (Relative to RLB), Goal, CPU Control, and Resource. Annotations include:
 

- A blue box pointing to a row with the text: 'Best-practice hints help to optimize service definitions'.
- A blue box pointing to a context menu with the text: 'Click to copy element on clipboard for insertion into another service definition'.
- A blue box pointing to a row with the text: 'Click to check where the element is used'.

## Transfer of Service Definition Elements



- Multiple table elements can be copied from one service definition to another service definition with one copy&paste operation
  1. Open the source service definition(s) in *View* tab(s) and switch to the table with the elements that you want to copy over
  2. Open the target service definition in a *Modify* tab and switch to the same table
  3. Select the elements in the table in the *View* tab and select action *Copy to clipboard*
  4. Switch to the *Modify* tab and trigger table action *Paste*
- You can copy&paste multiple table elements within the same service definition
  - E.g. you can copy the Periods of a Service Class from the Service Classes table to a Service Class Overrides table of a Service Policy if you want to make only small goal changes in the Service Policy
  - E.g. you can duplicate Classification Rules and insert them under another parent Classification Rule

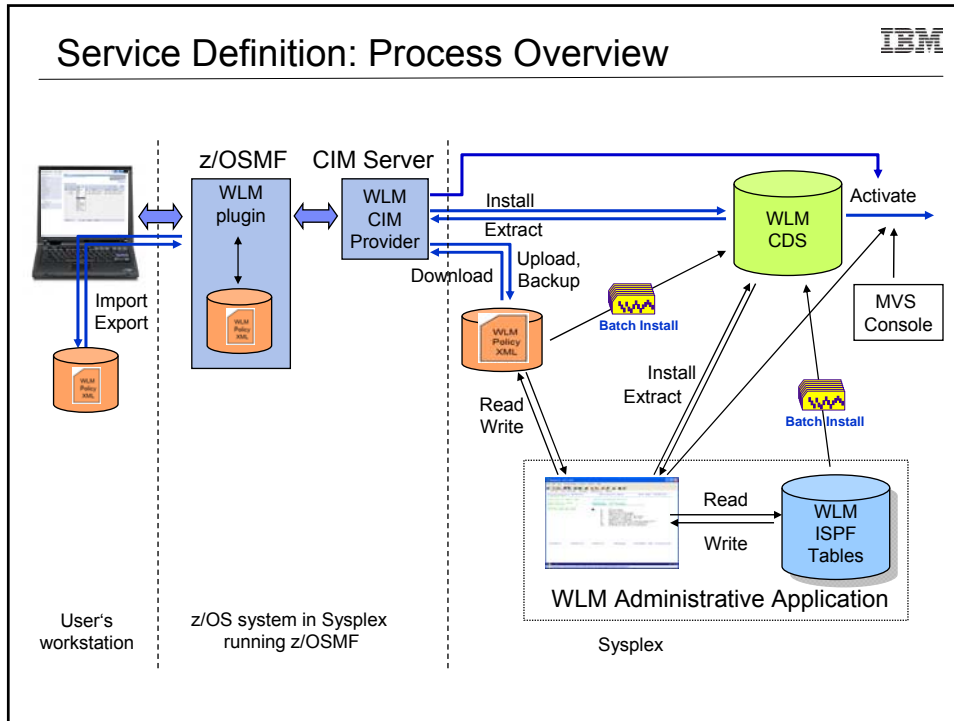


## Fine-grained Authorization (V1.13)



- Separate authorization levels for
  - Viewing of service definitions, service policies, and WLM status
  - Installation and activation of service policies
  - Modification of service definitions
- In repository authorization mode the WLM authorization of roles is controlled by three tasks on the Roles panel:
  - Workload Management
  - Workload Management Install
  - Workload Management Modify
- In SAF authorization mode the WLM authorization of roles is controlled via the SAF resource names:
  - ZOSMF.WORKLOAD\_MANAGEMENT.WORKLOAD\_MANAGEMENT.VIEW
  - ZOSMF.WORKLOAD\_MANAGEMENT.WORKLOAD\_MANAGEMENT.INSTALL
  - ZOSMF.WORKLOAD\_MANAGEMENT.WORKLOAD\_MANAGEMENT.MODIFY
- To enable a role to launch the Workload Management task it is not sufficient to provide authorization for 'installation' or 'modification'; in addition the role has to be authorized for 'viewing'.



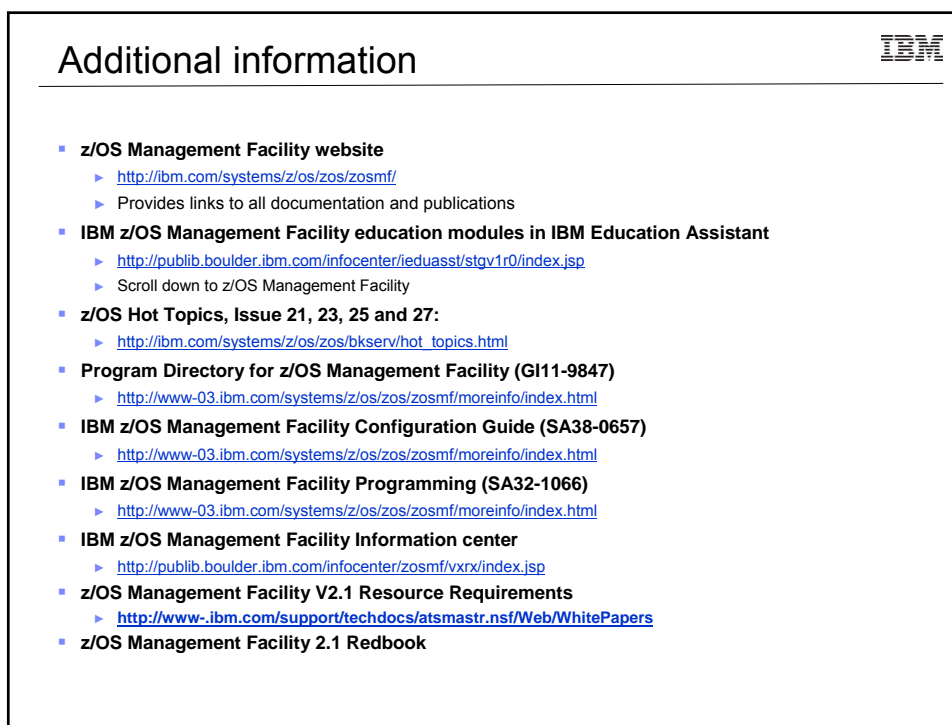
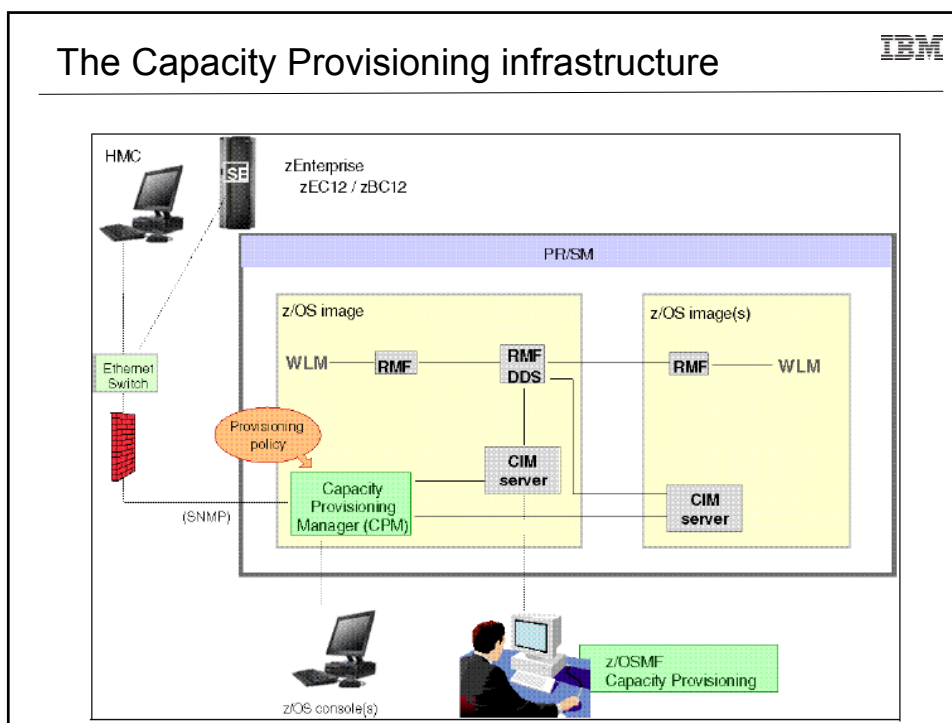


### Capacity Provisioning in z/OSMF V2.1 IBM

- The z/OS Capacity Provisioning Manager can help you to monitor your systems for short term capacity fluctuations
- Helps you manage the physical server capacity as well as defined capacity and group capacity limits in use.
- Based on On/Off Capacity on Demand (CoD), temporary capacity can be automatically activated / deactivated based on a user defined policy or on command
- Manage, install, import /export and also activate domain configurations and policies
- Manage connections to Provisioning Manager and transfer provisioning policies and domain configurations from a central shared repository.
- Display reports about domain status.

*z/OSMF V2.1 Capacity Provisioning supports all of the functions available in the Microsoft Windows-based Capacity Provisioning Control Center.\**

*\*Microsoft Windows based Capacity Provisioning Control Center is no longer available in z/OS V2.1*





Glenn Anderson, IBM Lab Services and Training



## Understanding z/OSMF for the Performance Management Sysprog

Thanks for Attending!



Winter SHARE  
March 2014  
Session 55220

