

Hendrik Woerner

IT-Specialist for WebSphere on System z

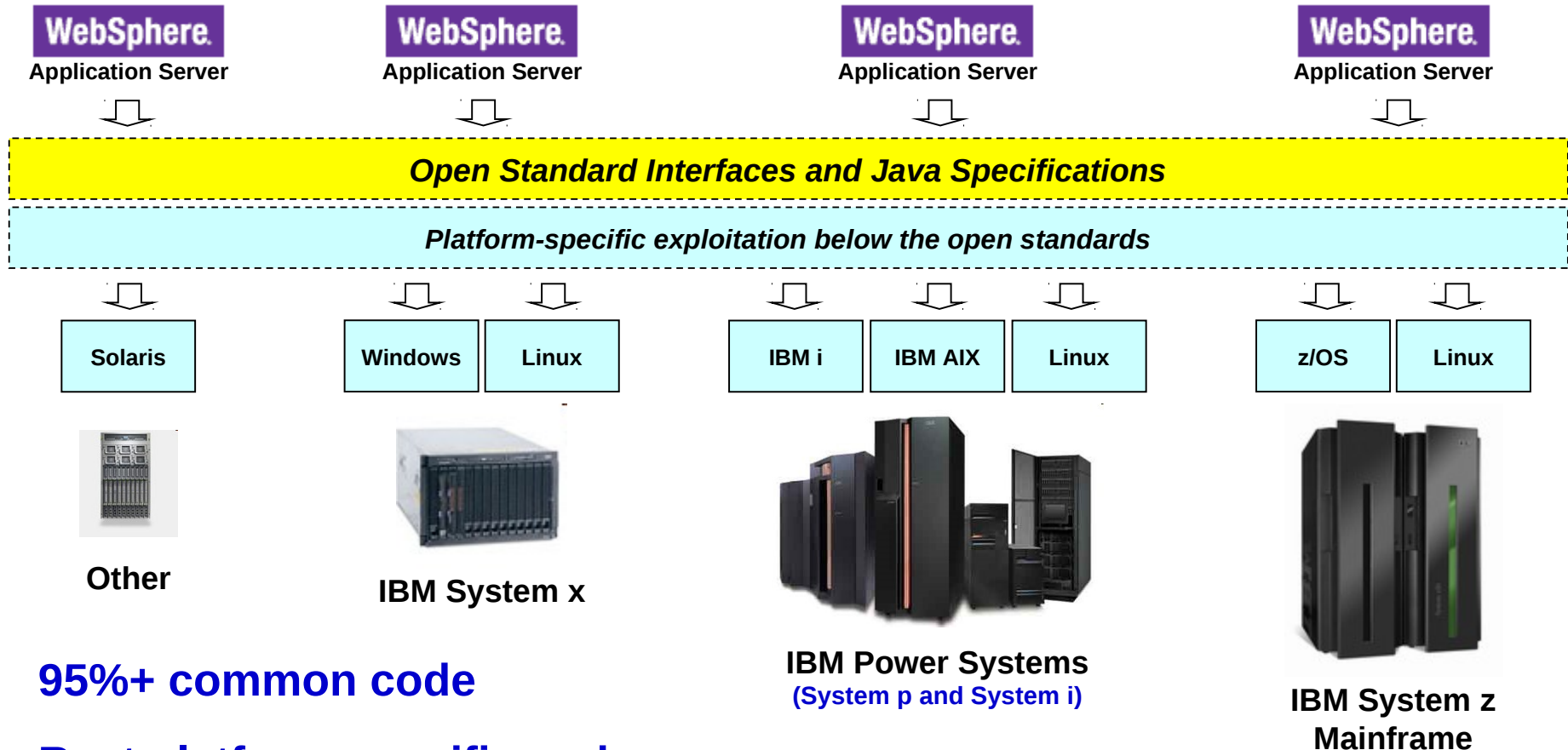
SWG Europe Team

hwoerner@de.ibm.com

WebSphere z/OS Architecture and Overview

WebSphere is WebSphere Above the Specification Layer

This is a key strategic statement from IBM. You will have aligned specifications and delivery schedules for WAS across the platforms



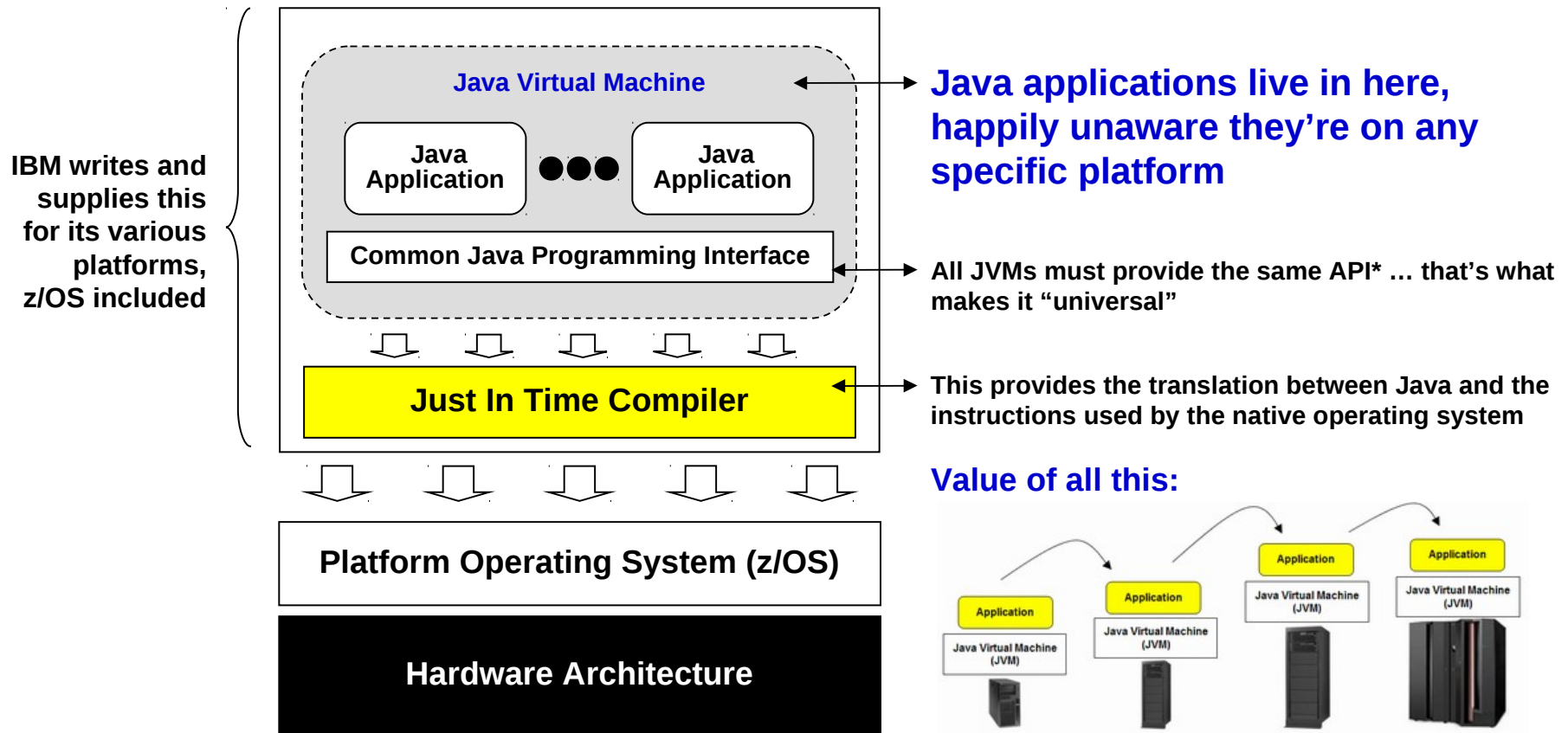
95%+ common code

Rest platform-specific code

Java and JVM ...

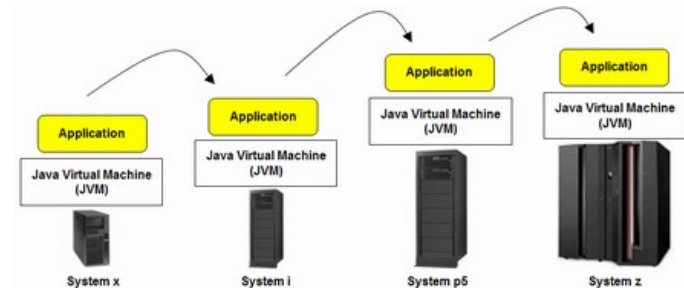
Java and the Java Virtual Machine (JVM)

Java is a programming language; the JVM is what the Java programs runs in. It's what shields the Java program from the specifics of the platform.



WebSphere Application Server contains a JVM
(Several, as a matter of fact)

Value of all this:



Platform neutrality

* For a given specification level of Java

Is Java on System z Different?

Is there anything special that needs to be considered when designing and writing Java code to run on System z or z/OS?

Answer:



No!

Clear enough? ☺

Java is Java

The point of an open standard application platform is to eliminate platform dependencies

Two points:

1. That wasn't always the case ... earlier we had ASCII / EBCDIC issues. No more.
2. There are poor coding practices that make bringing applications to z/OS problematic ...

What An “Application Server” Provides

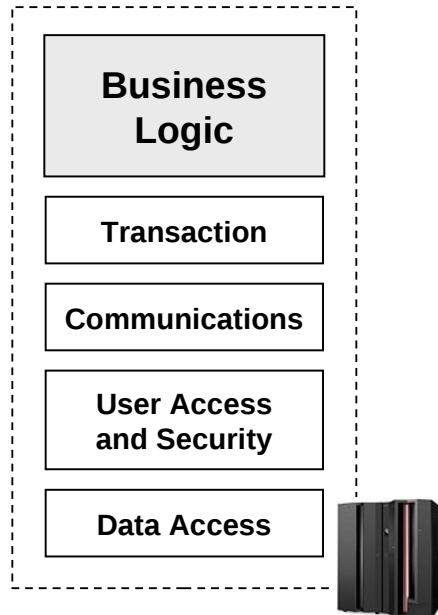
WebSphere Application Server is an “application server” ... but what is that?

In the “Old Days”



Developer

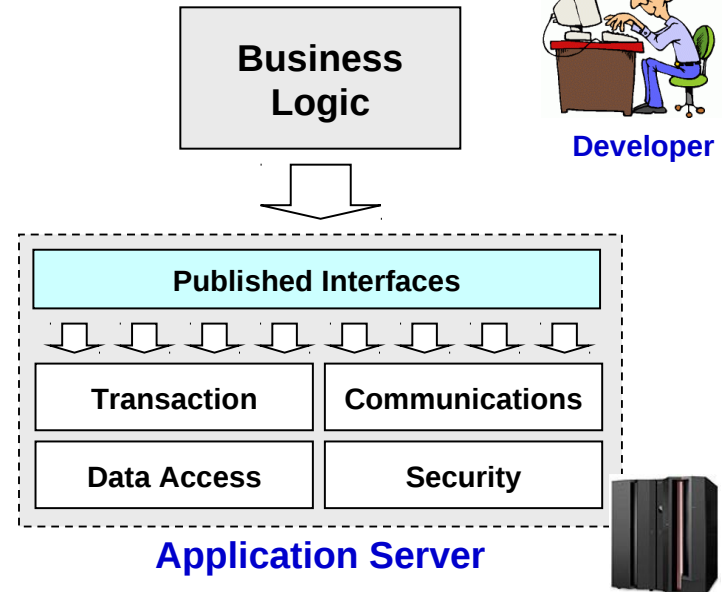
**Had to write it all
Everyone was
reinventing wheel
over and over
again**



Nowadays



Developer



Purpose is to provide pre-packaged application support stuff so developers can focus on the main business task. No more re-inventing the wheel.

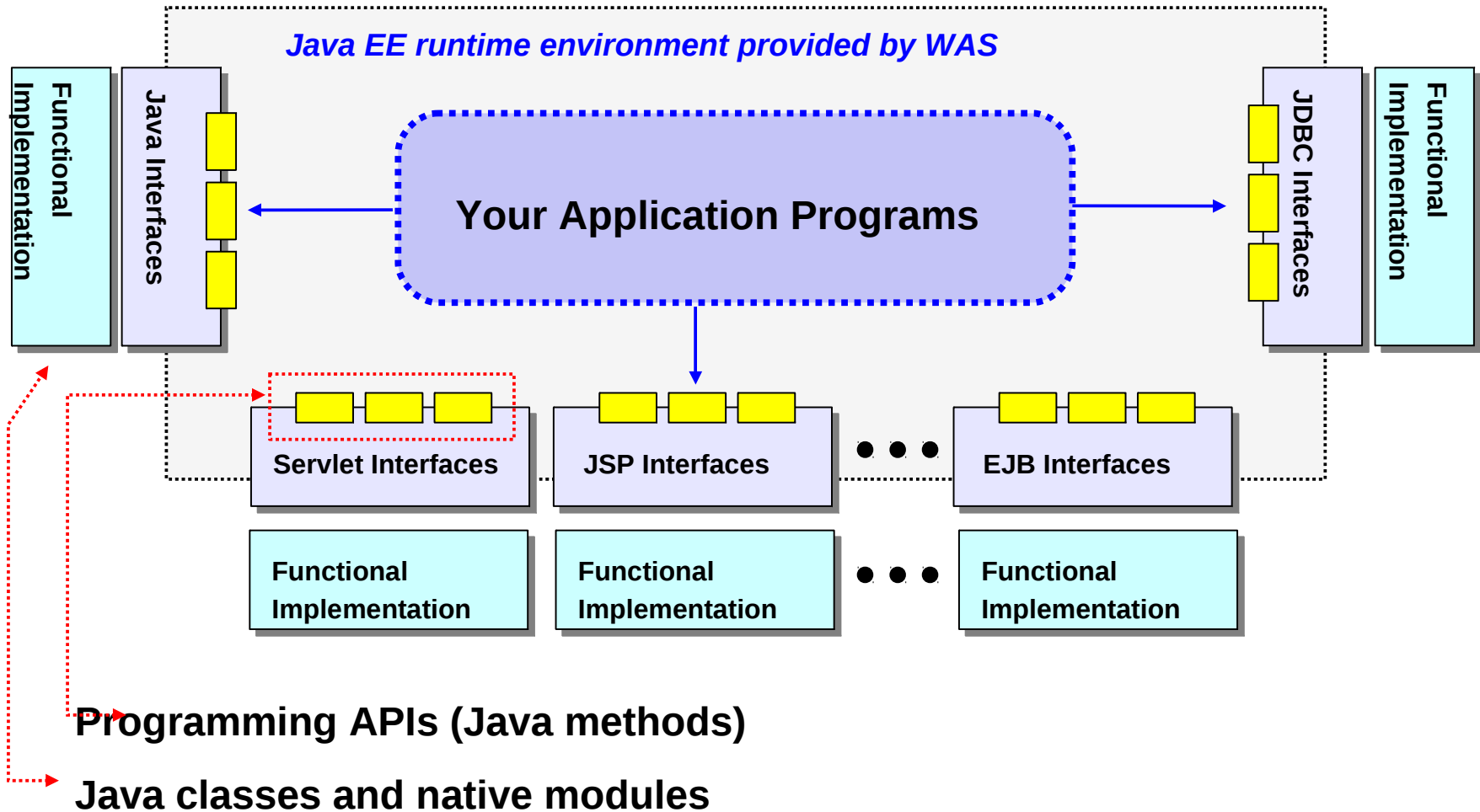
This is not new with WebSphere ... IBM had an application server back in 1968!*

So what’s the key difference between WebSphere and past application servers?

* CICS is an application server

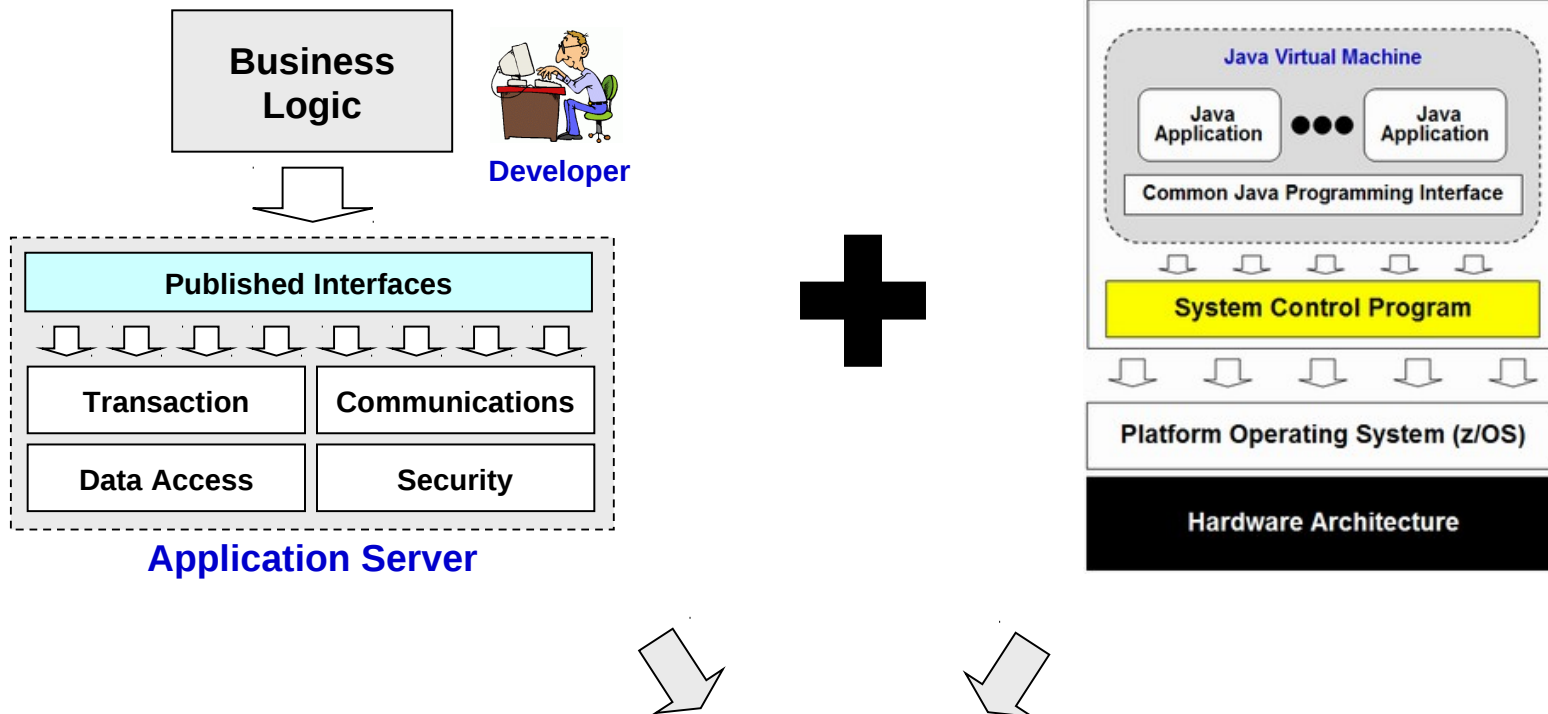
What An “Application Server” Provides

The open standard specifications are just words on paper. It isn't until they are implemented in WAS that they provide function to your application:



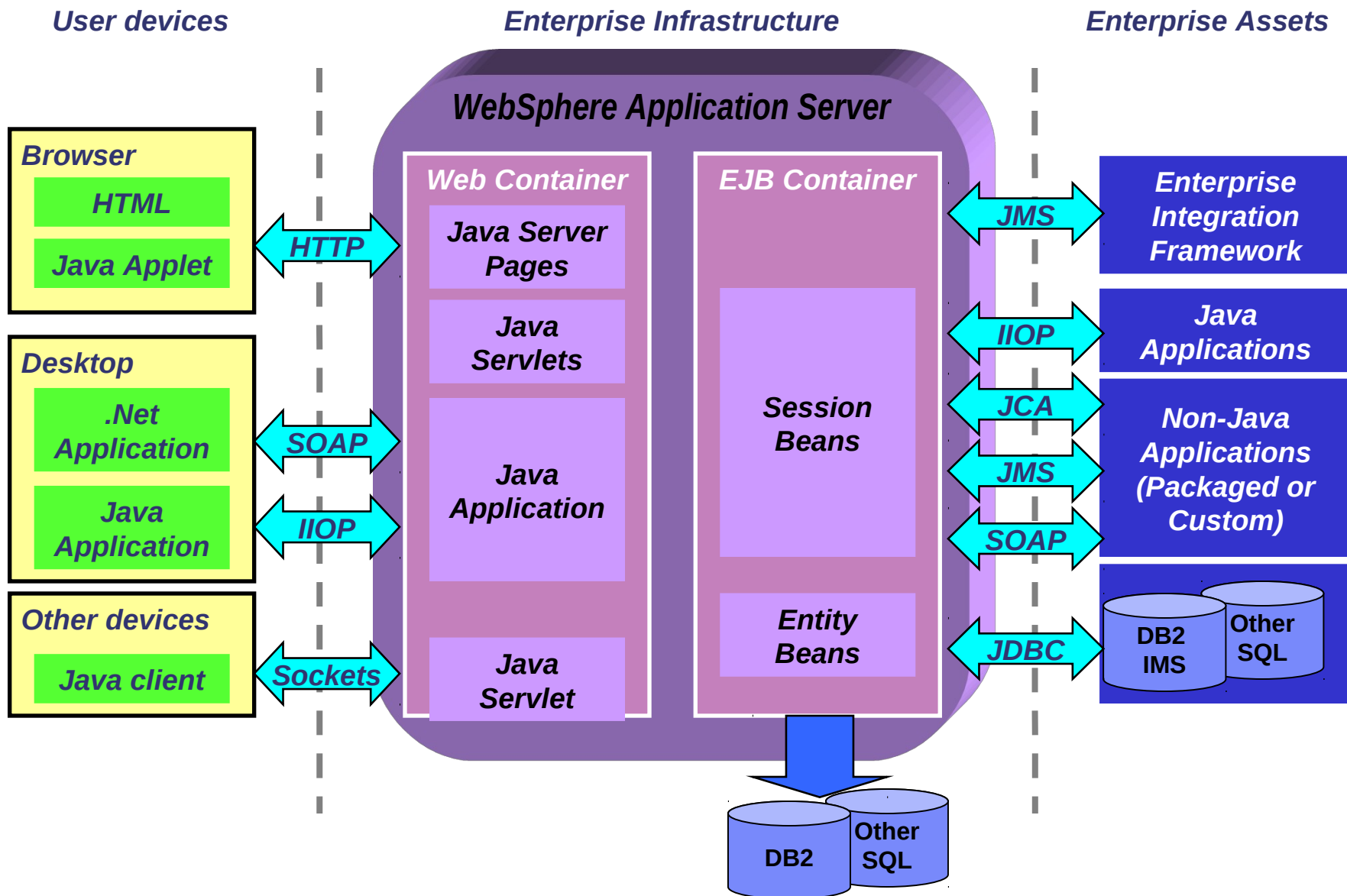
We Need to Marry Two Key Concepts Together

The idea of a framework that provides common services, and the notion of a Java Virtual Machine:



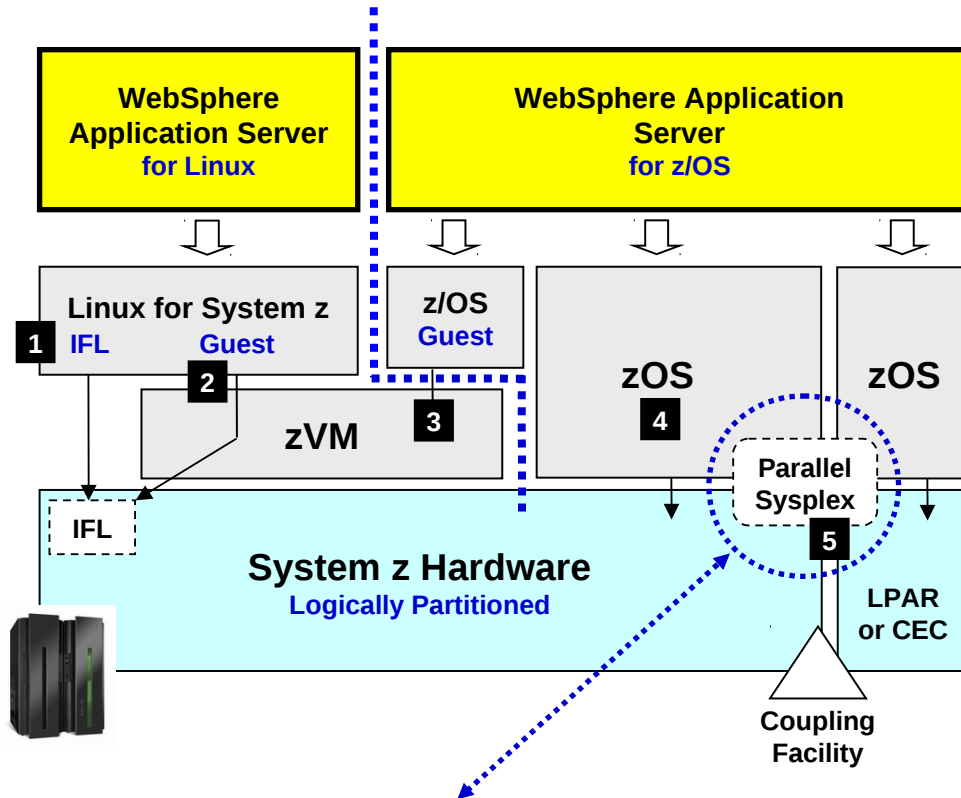
WebSphere Application Server

WebSphere Application Server Services



System z, z/OS, Linux, and WebSphere Application Server

Here's a mapping of how the two flavors of WebSphere Application Server can be hosted on System z hardware:



WebSphere z/OS design and implementation capitalizes on the Sysplex environment

Much more to follow

1. Linux for System z directly on IFL

Possible, but not very common. Solution where no zVM skills exist

1. Linux for System z as guest on zVM

Very common. This provided excellent virtualization (zVM) and Linux. Runs on the IFL.

1. z/OS as guest on zVM

Another example of zVM's virtualization capabilities. WAS z/OS as guest typically in a development or test environment.

1. z/OS in a non-Sysplex environment

WAS runs directly on z/OS with no zVM virtualization. No Sysplex more common in test environments or small production.

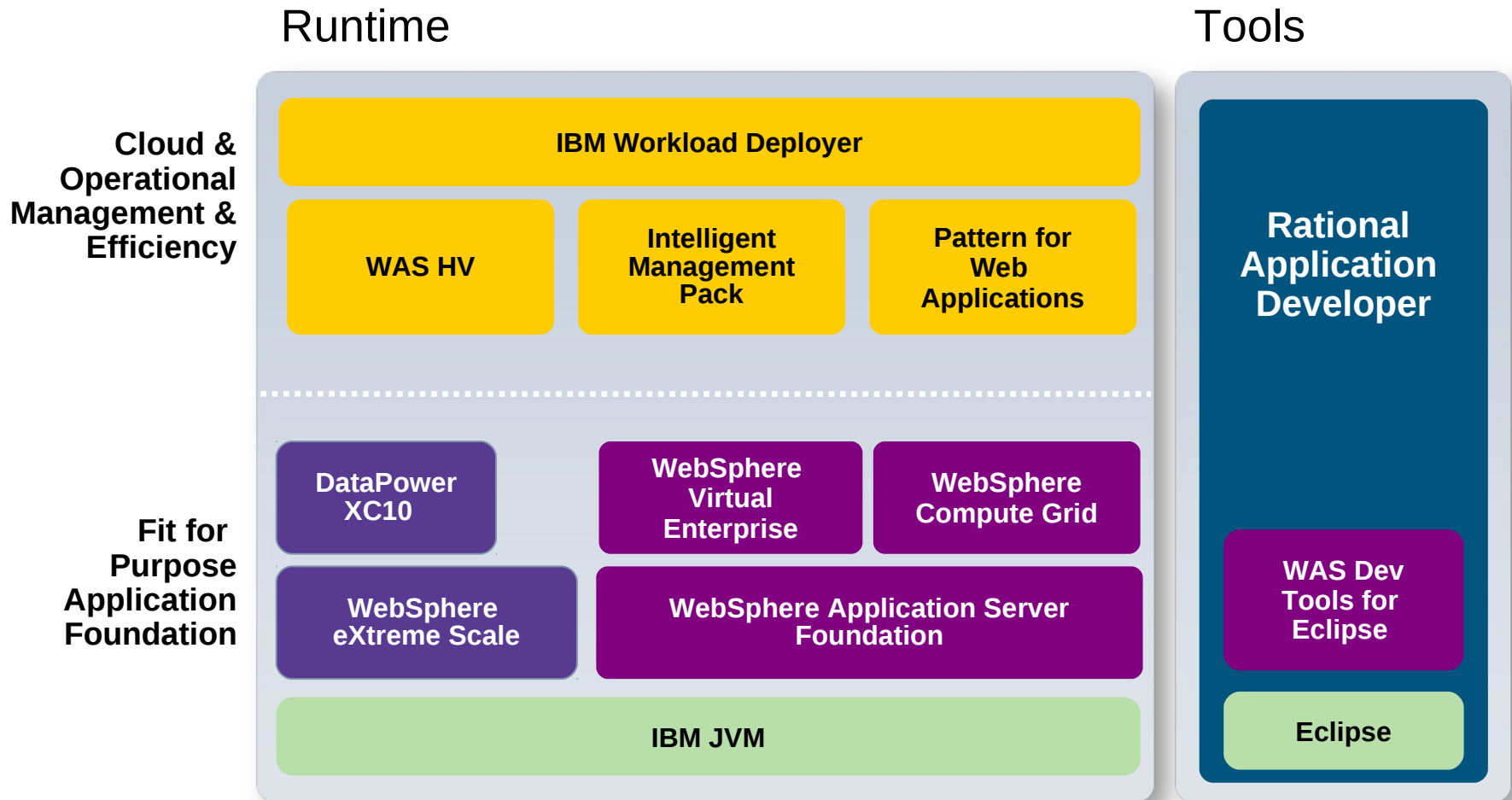
1. z/OS in a Parallel Sysplex environment

This is the flagship environment. This is where high availability, scalability and maximum platform exploitations takes place.

WebSphere Version 8.0 Product Portfolio

WebSphere Application Infrastructure

Current Offerings



WebSphere Application Server Family with WAS V8.0

WAS for Developers

Tools Edition

Enables efficient development of innovative apps that will run on WAS in production

Available as a no-charge edition for the developer desktop and includes Eclipse adapters

WAS Hypervisor Edition

Optimized to instantly run in VMware and other server virtualization environments

WAS ND

Tools Edition

Delivers near-continuous availability, with advanced performance and mgmt capabilities, for mission-critical apps

WAS for z/OS

Takes full advantage of the z/OS Sysplex to deliver a highly secure, reliable, and resource efficient server experience

WAS

Tools Edition

Provides secure, high performance transaction engine for moderately sized configurations with web tier clustering and failover across up to five application server profiles

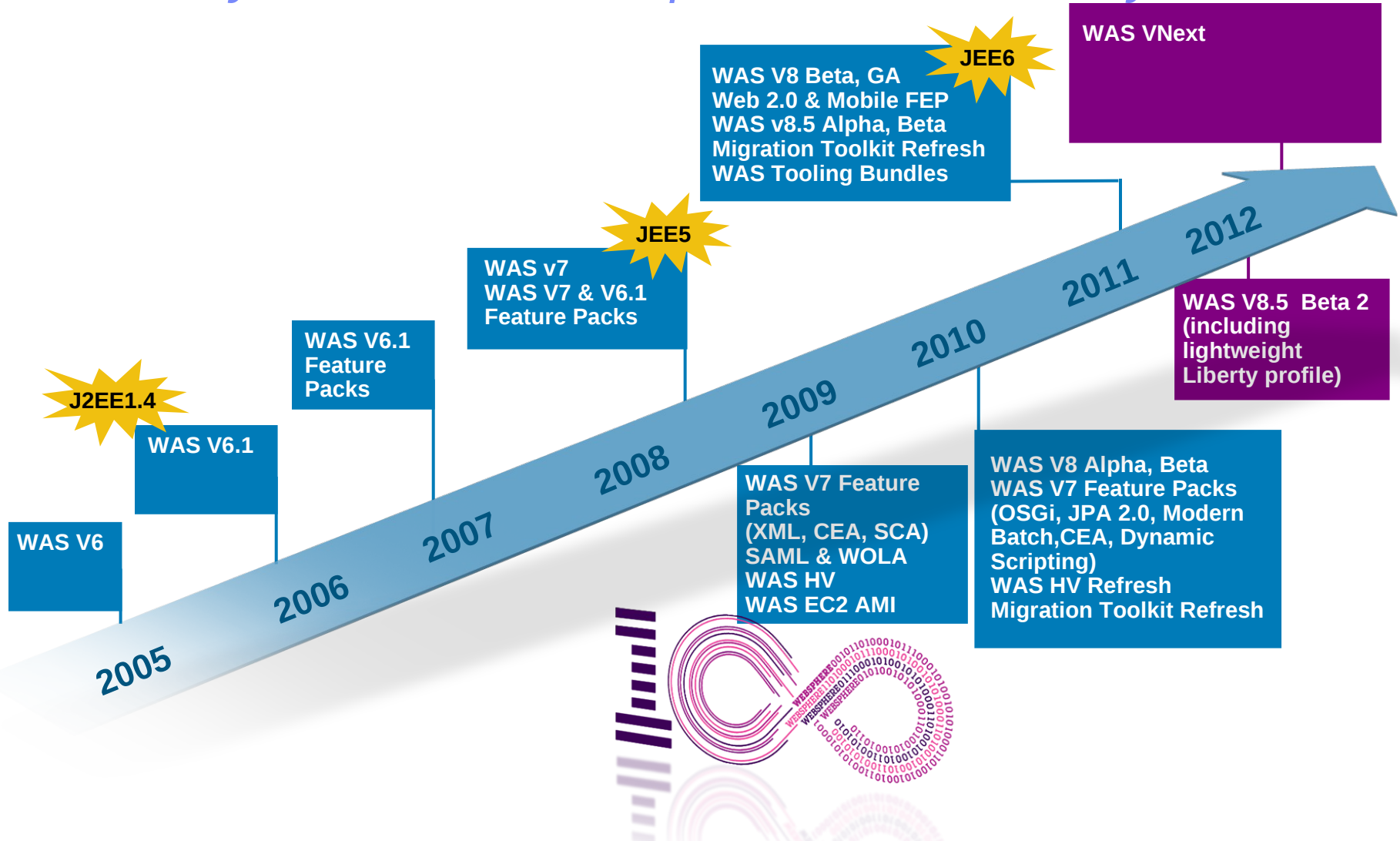
WAS Express

A lower-cost, ready-to-go solution to build dynamic Web sites & apps

WAS CE

An open source-based, small footprint foundation with no up-front acquisition costs

WebSphere Application Server: *Over 14 years of Leadership & Trusted Delivery*



WAS Tools Editions

New!

FREE for Developers- Unit Testing

WAS for Developers – Tools Edition for Eclipse



Easily & Quickly Obtain Eclipse Adapter for WAS Environments (V7.0 & V8.0)

- ✓ **WAS for Developers + Eclipse Adapter**
- ✓ Available unsupported (no charge) and supported (for fee)
- ✓ Eclipse adapter makes it easy to deploy applications from Eclipse development environments directly to WAS

Purchase these Bundles – RAD + WAS

WAS – Tools Edition

WAS Network Deployment– Tools Edition

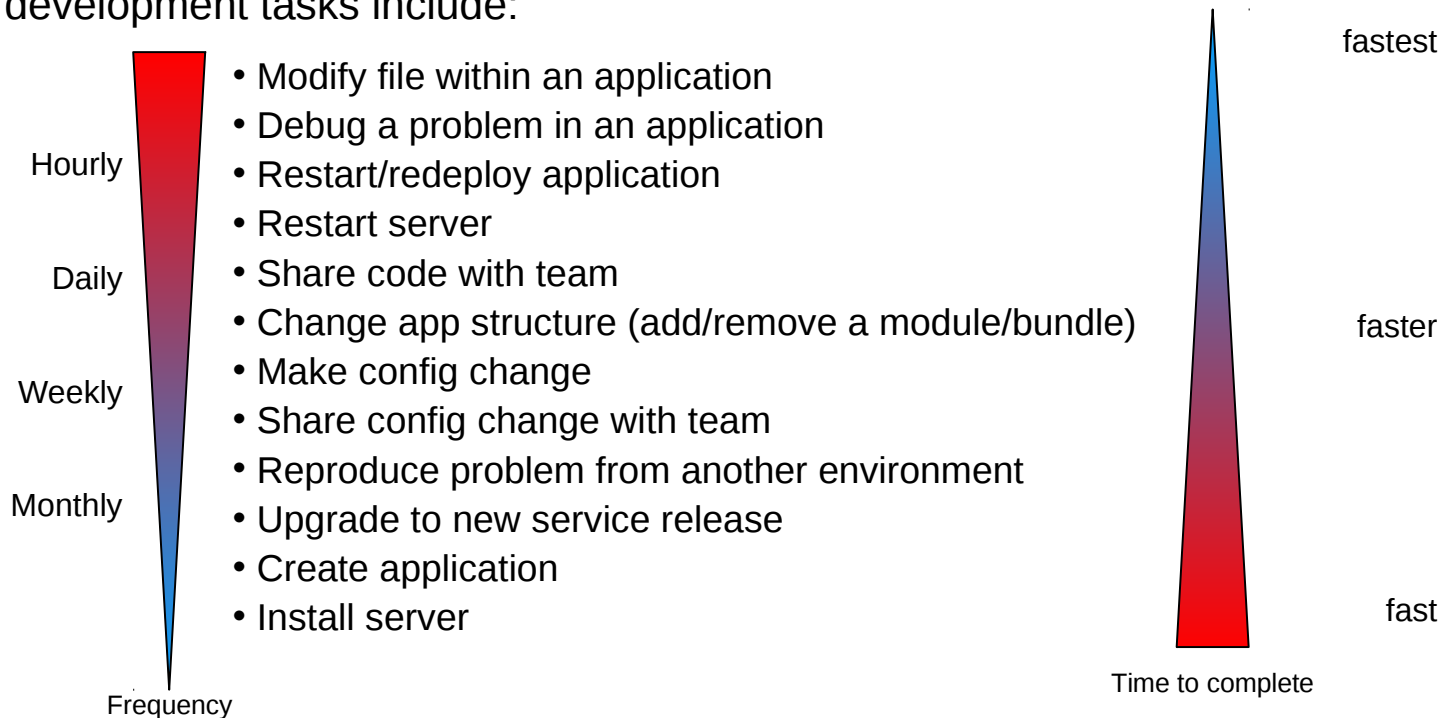


WebSphere Application Server V7.0 & V8.0 Bundled Solutions:

- ✓ **Unlimited use of RAD (Rational Application Developer) and/or Eclipse adapter w/ WAS or WAS ND for developing applications *for the production use WAS server in the bundle***
- ✓ **Production ready runtime and tools**

Think About What Developers Do (to understand what they *want*)

- Common development tasks include:



- All tasks should be as painless as possible, with special emphasis on the more frequent ones. If the time taken to accomplish these tasks is an impediment to the development, the cost of the fidelity of the test server runtime is challenged.
- These kind of capabilities reflect on the Application Server runtime as well as the Tools

WebSphere Application Server V8.5 Beta

Lightest weight tools and runtime

WAS Liberty Profile

- WAS V8.5 Beta

WAS Dev Tools for Eclipse v8.5 Beta

- Install as Eclipse feature
- Server Adapter tools (V7, V8, V8.5 Beta)
- Java EE, OSGi, web/mobile dev tools

Eclipse

- WAS 8.5 Beta introduces the lightweight WAS Liberty profile for developing and testing web apps that don't require the full JEE environment
- WAS Liberty profile will be part of all editions of WAS V8.Next
 - WAS Liberty Profile + WebSphere Developer Tools for Eclipse = free, lightweight tools and runtime for development
- Support for development/debug on Mac OS
- Operational support for “packaged” unzip deploy of application+server+config
- Integration with WAS ND Job Manager for centralized lifecycle management

Other Developer Highlights of WAS V8.5 Beta:

- Pluggable JDK with Java SE 7 support for latest Java standards
- Modular OSGi Application support (includes EJB bundles)

WAS V8.5 Beta:

<http://www14.software.ibm.com/iwm/web/cc/earlyprograms/websphere/wasv8na/>

WebSphere Application Server Family with WAS V8.0

WebSphere Application Server for Developers – Tools Edition for Eclipse

Optimizes the development of innovative apps that will run on WAS in production

Available as a no-charge edition for the developer desktop and includes Eclipse adapters

WAS Hypervisor Edition

Optimizes the delivery of services in the cloud
+ Lightweight profile

WAS ND

Tools Edition

Delivers a lightweight, high performance, mission-critical capabilities, for mission-critical applications

New bundle with RAD - WAS ND - Tools Edition

WAS for z/OS

Takes full advantage of the z/OS architecture to deliver a high performance, reliable, and resource-efficient server experience
+ Lightweight profile

WAS

Tools Edition

+ Lightweight profile

High performance transaction engine for modern, large scale configurations with web tier clustering and dynamic scaling

New bundle with RAD – WAS - Tools Edition

WAS Express

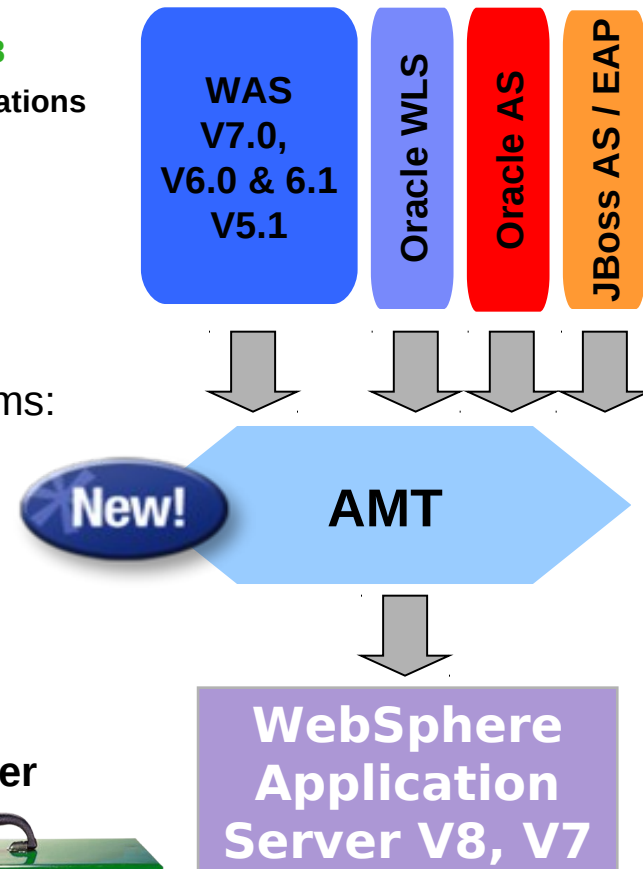
A lower cost solution to build dynamic Web sites & apps
+ Lightweight profile

WAS CE

An open source-based, small footprint foundation with no up-front acquisition costs

Migrate applications from previous versions of WAS & other Java EE application servers to WebSphere faster with minimized risk

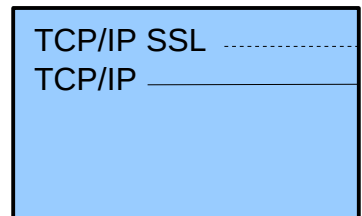
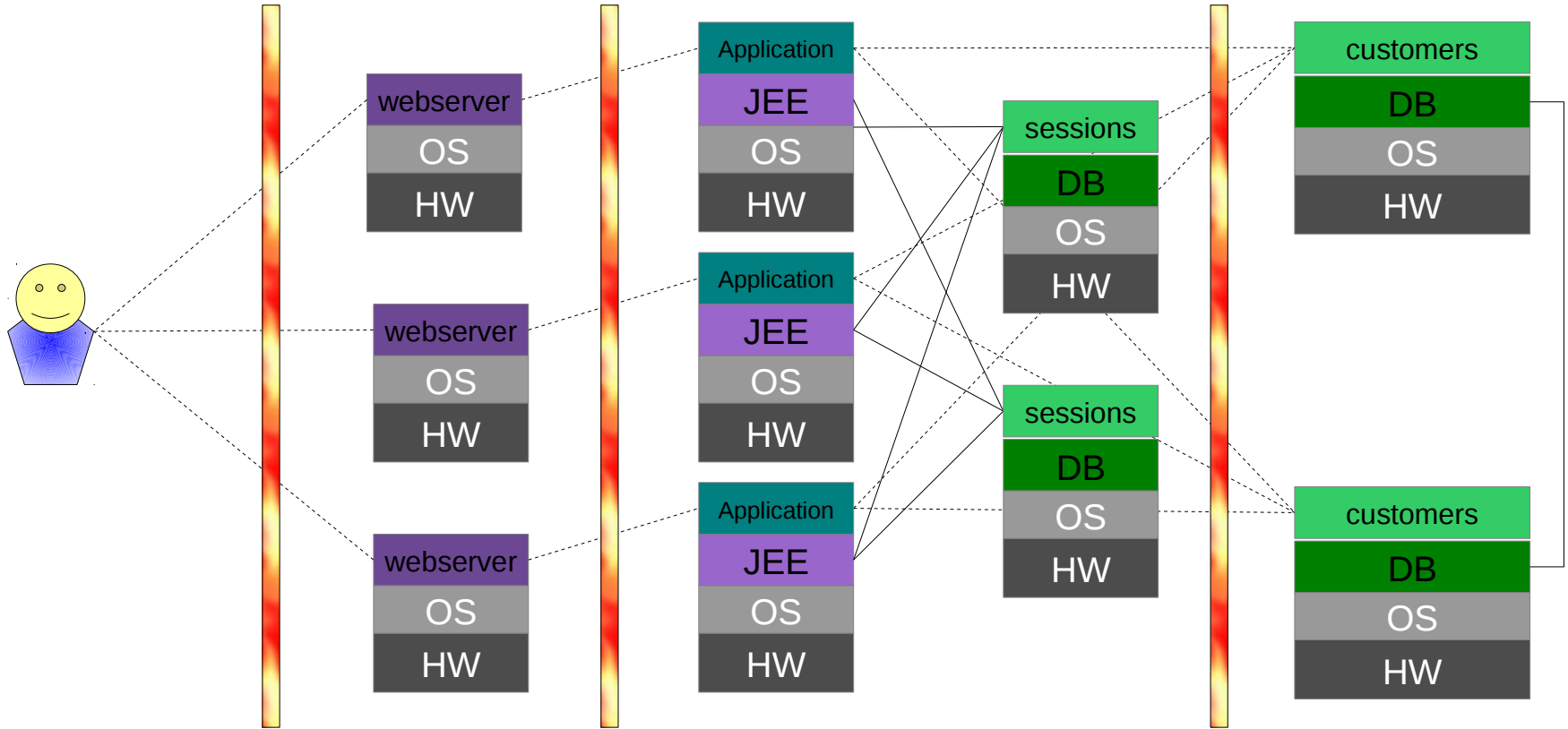
- Migrate applications from older **WAS 5.1, 6.0.2, and 6.1 to WAS v7 / v8**
 - **Get latest Feature Packs and latest WAS release....** Build new applications with Mobile support, CEA, Dynamic Scripting and more
- Migrate from Oracle or JBoss faster and easier to WAS V8 or V7
 - Migrate applications up to 2x as fast
 - Migrate web services up to 3x as fast
- **Application Migration Tool**
 - Analyzes source code to find potential migration problems:
 - Removed features
 - Deprecated features
 - Behavior changes
 - JRE 5 & JRE 6 differences
 - Java EE specification changes or enforcements
 - Capable of making some application changes
 - Provides guidance on how to make required changes
 - **Works with Eclipse or Rational Application Developer (RAD)**



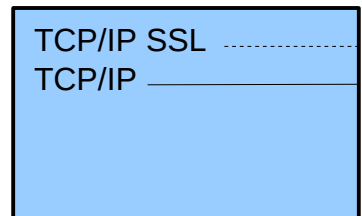
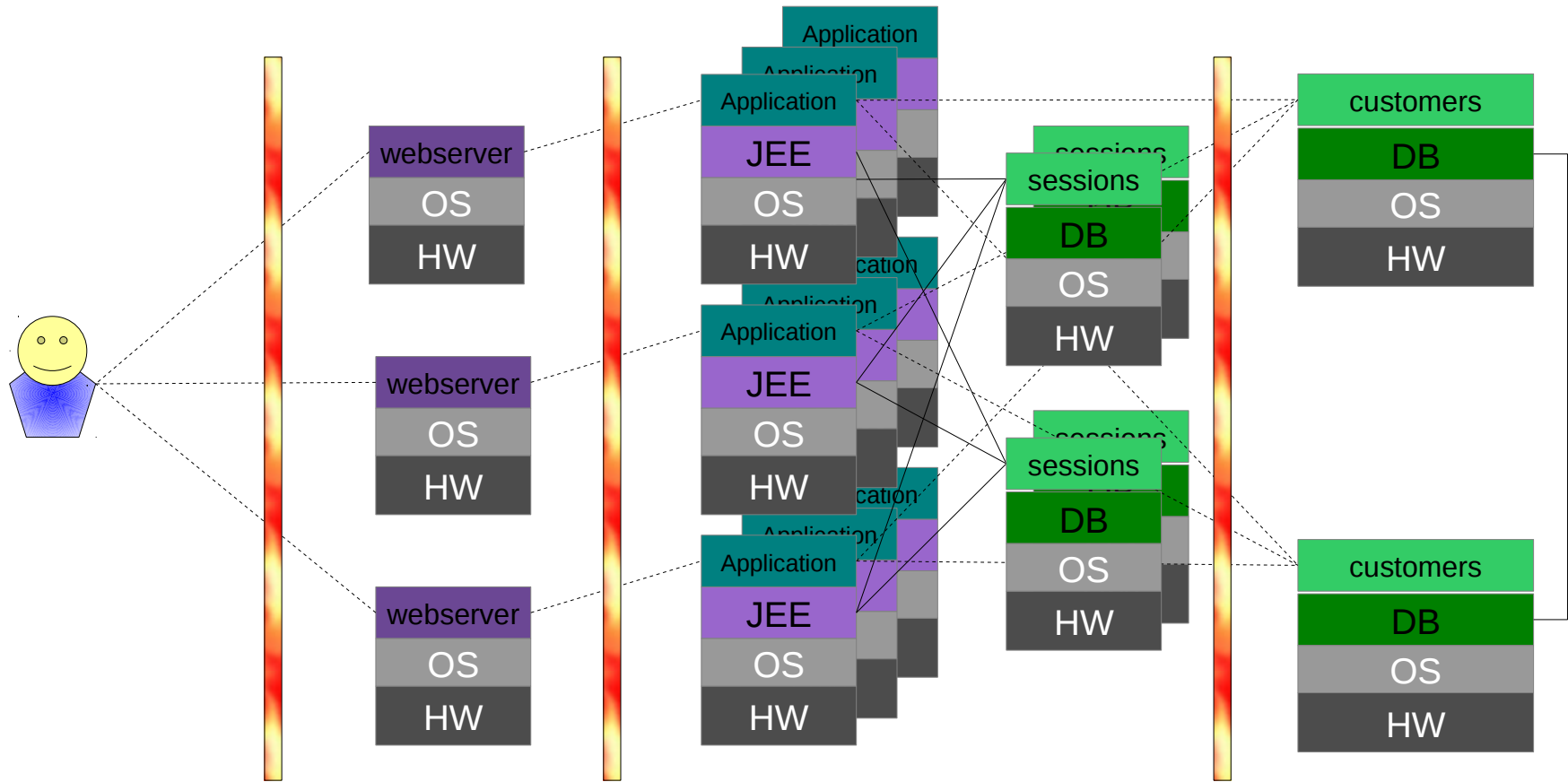
Get the Tool at No Charge: <http://ibm.co/hqfkdj>

The value add of WAS z/OS

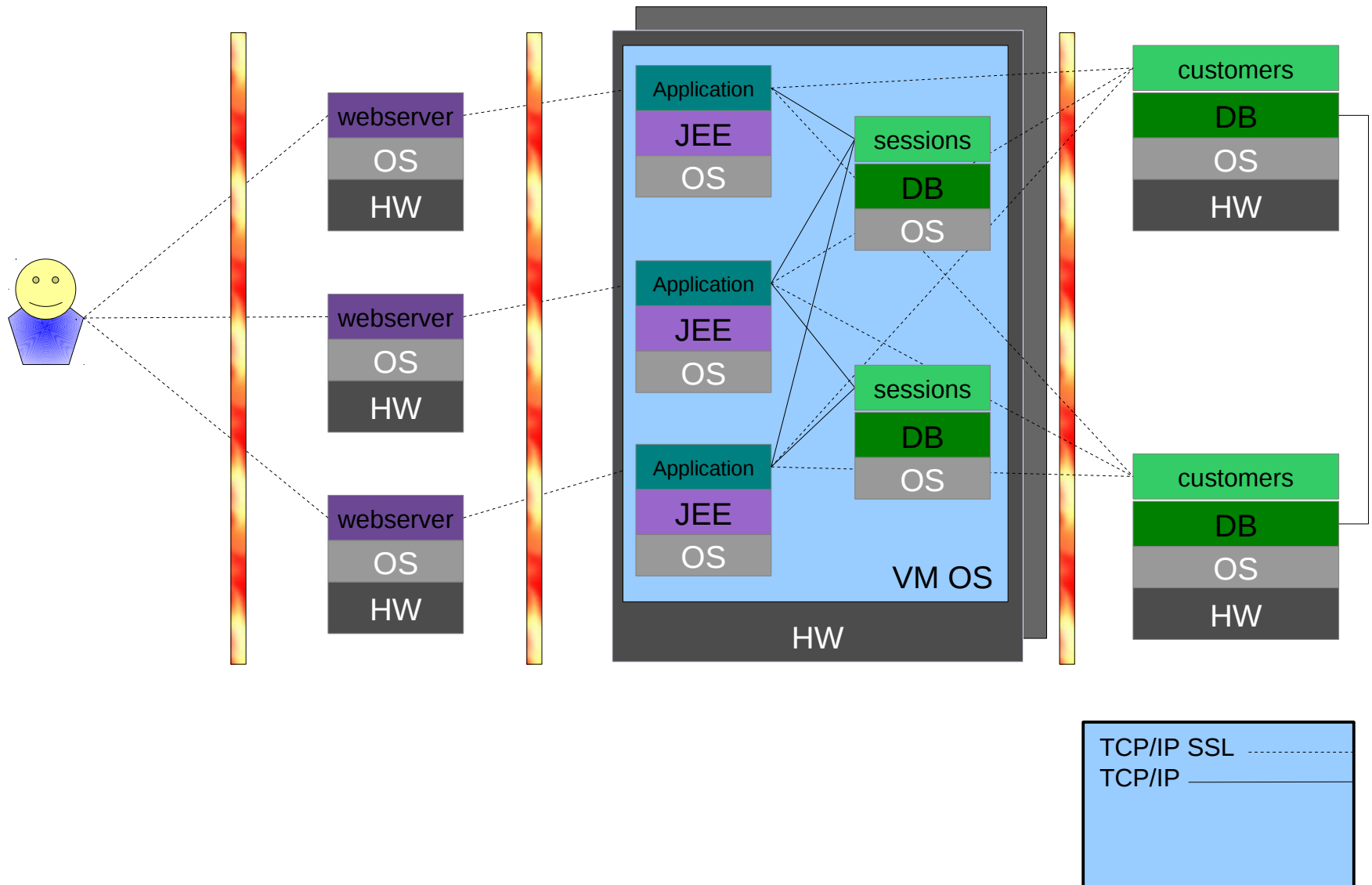
Distributed Infrastructure on non-shared Hardware



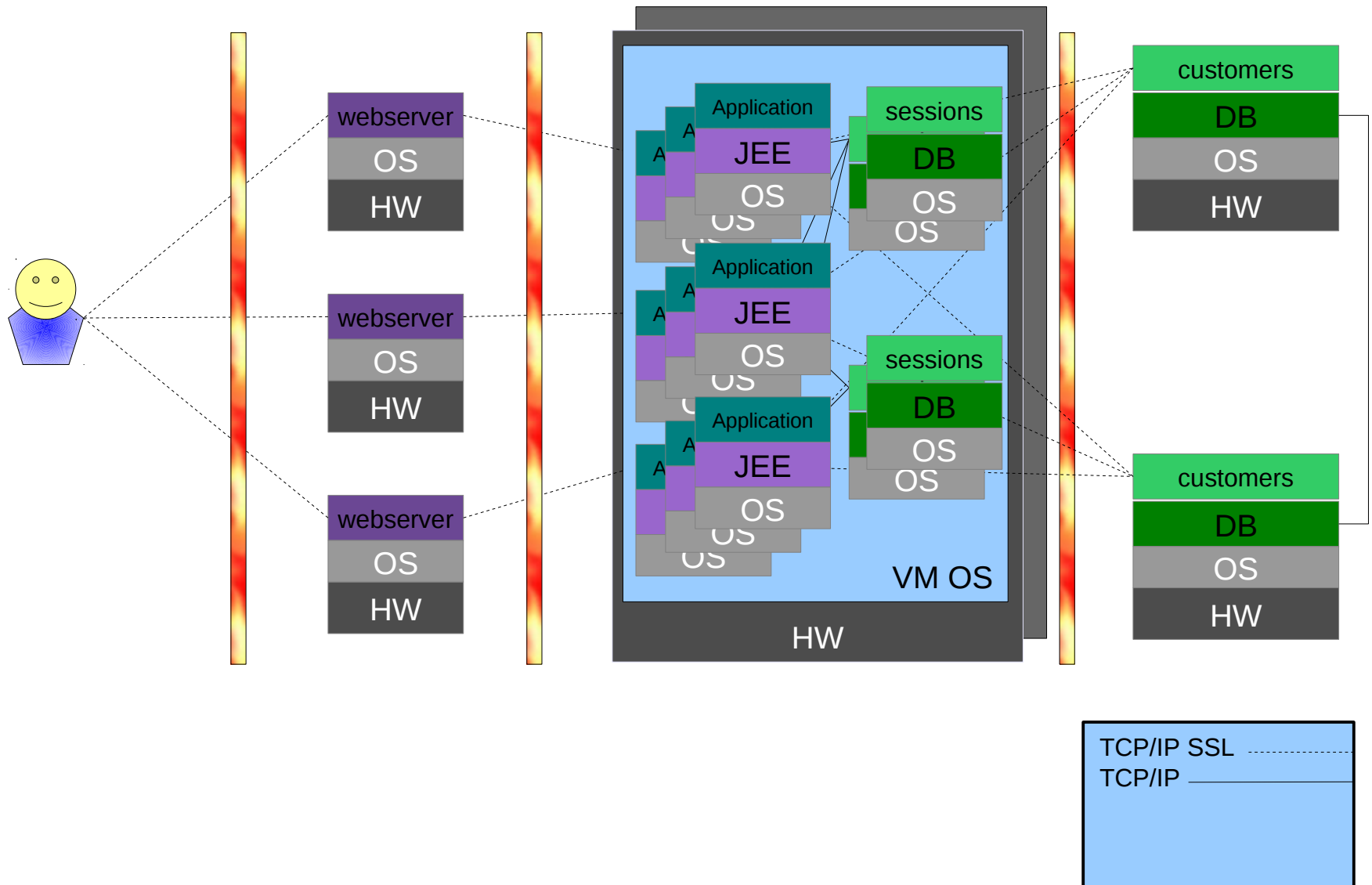
Distributed Infrastructure on non-shared Hardware – Scale Up



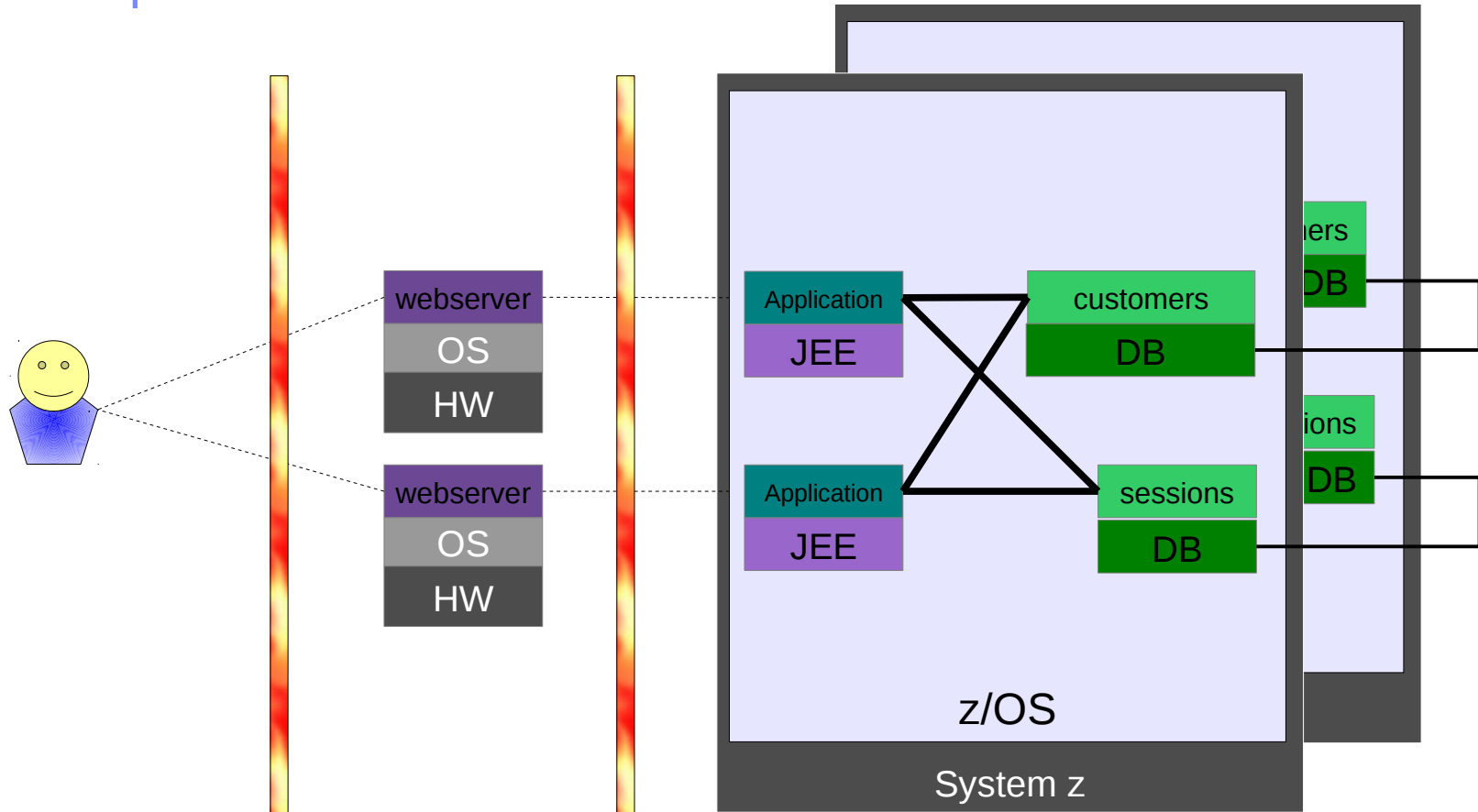
Distributed Infrastructure with virtual guests



Distributed Infrastructure with virtual guests – Scale Up

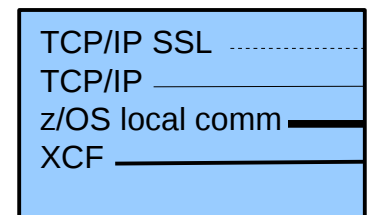
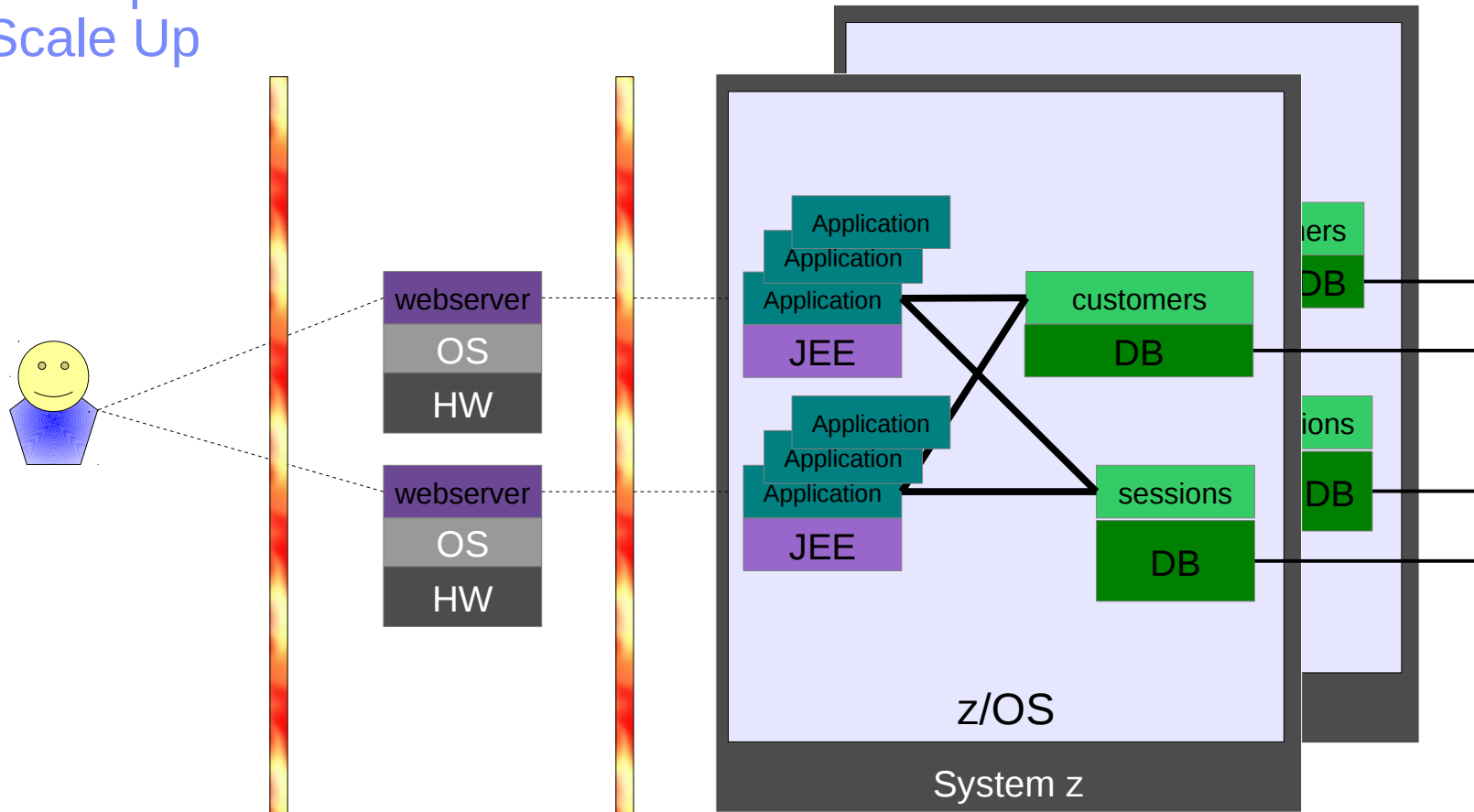


WebSphere on z/OS Infrastructure

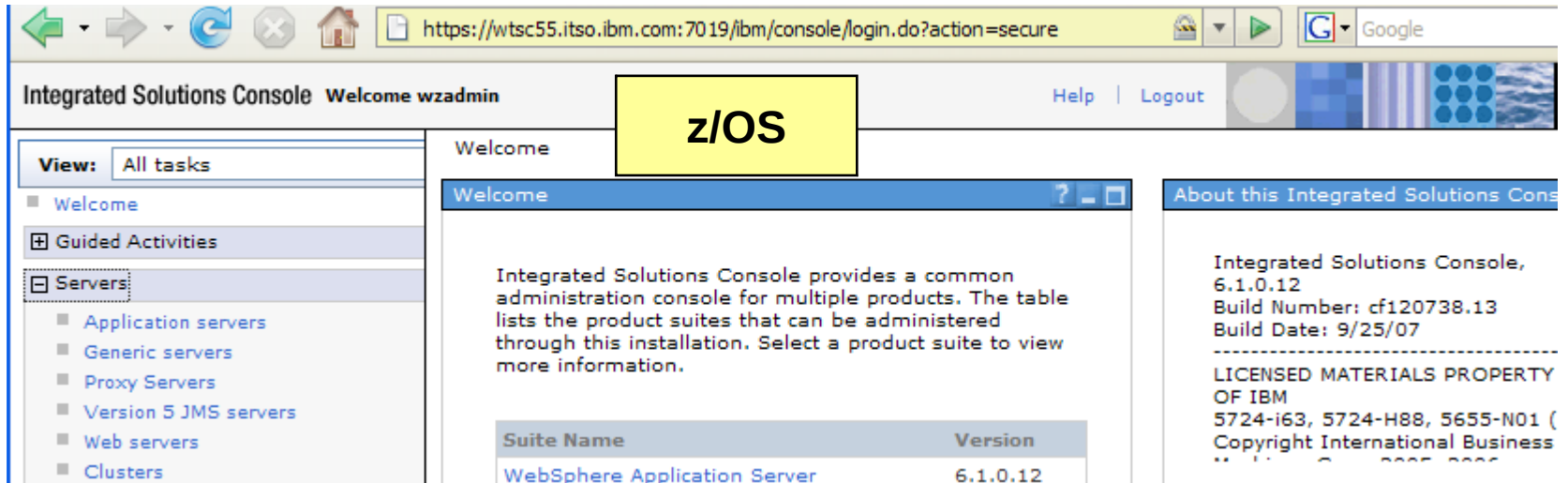


TCP/IP SSL
TCP/IP	———
z/OS local comm	————
XCF	————

WebSphere on z/OS Infrastructure Scale Up



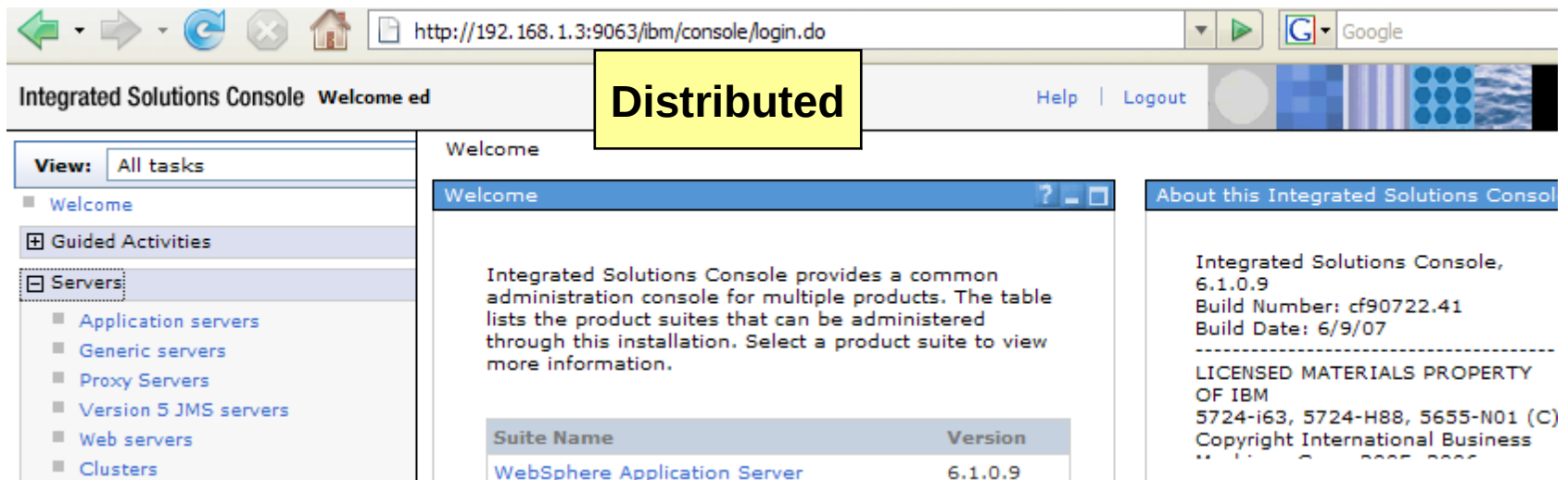
Which WebSphere admin console belongs to which platform?



The screenshot shows the Integrated Solutions Console for z/OS. The browser address bar displays <https://wtsc55.itso.ibm.com:7019/ibm/console/login.do?action=secure>. The page header includes "Integrated Solutions Console Welcome wzadmin" and a yellow box with the text "z/OS". The left sidebar shows a tree view with "Servers" selected, containing links for Application servers, Generic servers, Proxy Servers, Version 5 JMS servers, Web servers, and Clusters. The main content area has a "Welcome" message and a table of product suites.

Suite Name	Version
WebSphere Application Server	6.1.0.12

The right sidebar contains information about the Integrated Solutions Console, including version 6.1.0.12, build number cf120738.13, and build date 9/25/07.



The screenshot shows the Integrated Solutions Console for Distributed. The browser address bar displays <http://192.168.1.3:9063/ibm/console/login.do>. The page header includes "Integrated Solutions Console Welcome ed" and a yellow box with the text "Distributed". The left sidebar is identical to the z/OS screenshot. The main content area has a "Welcome" message and a table of product suites.

Suite Name	Version
WebSphere Application Server	6.1.0.9

The right sidebar contains information about the Integrated Solutions Console, including version 6.1.0.9, build number cf90722.41, and build date 6/9/07.

WSADMIN scripting on WAS z/OS vs. distributed

```
WZADMIN @ SC55:/WebSphereEd/wzcell/dmgr/DeploymentManager/profiles/default/bin>./wsadmin.sh –  
port 7010 –user wzadmin –password xyz –lang jython
```

```
WASX7209I: Connected to process "dmgr" on node wzdmnode using SOAP connector; The type of process  
is: DeploymentManager
```

```
WASX7031I: For help, enter: "print Help.help()"
```

```
wsadmin>AdminControl.completeObjectName("type=DeploymentManager,*")
```

```
'WebSphere:name=DeploymentManager,process=dmgr,platform=common,node=wzdmnode,diagnosticProvi  
der=true,version=6.1.0.12,type=DeploymentManager,mbeanIdentifier=DeploymentManager,cell=wzcell,spec  
=1.0'
```

```
C:\zProducts\was61\AppServer\profiles\Dmgr01\bin>wsadmin -lang jython
```

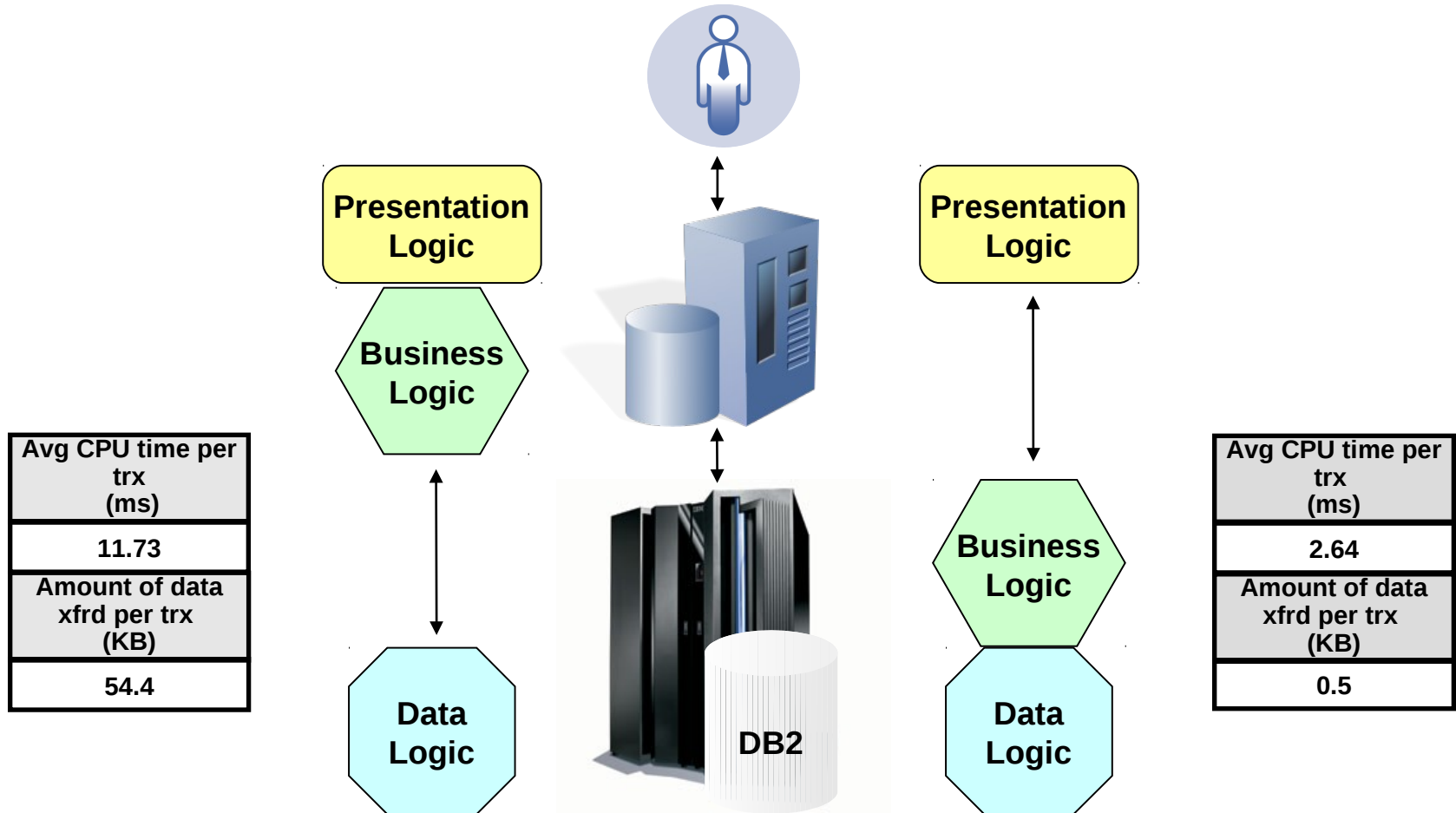
```
WASX7209I: Connected to process "dmgr" on node Dmgr01 using SOAP connector; The type of process  
is: DeploymentManager
```

```
WASX7031I: For help, enter: "print Help.help()"
```

```
wsadmin>AdminControl.completeObjectName("type=DeploymentManager,*")
```

```
'WebSphere:name=DeploymentManager,process=dmgr,platform=common,node=Dmgr01,diagnosticProvid  
er=true,version=6.1.0.9,type=DeploymentManager,mbeanIdentifier=DeploymentManager,cell=Dmgr01,spe  
c=1.0'
```

The value of proximity: transportation industry POC

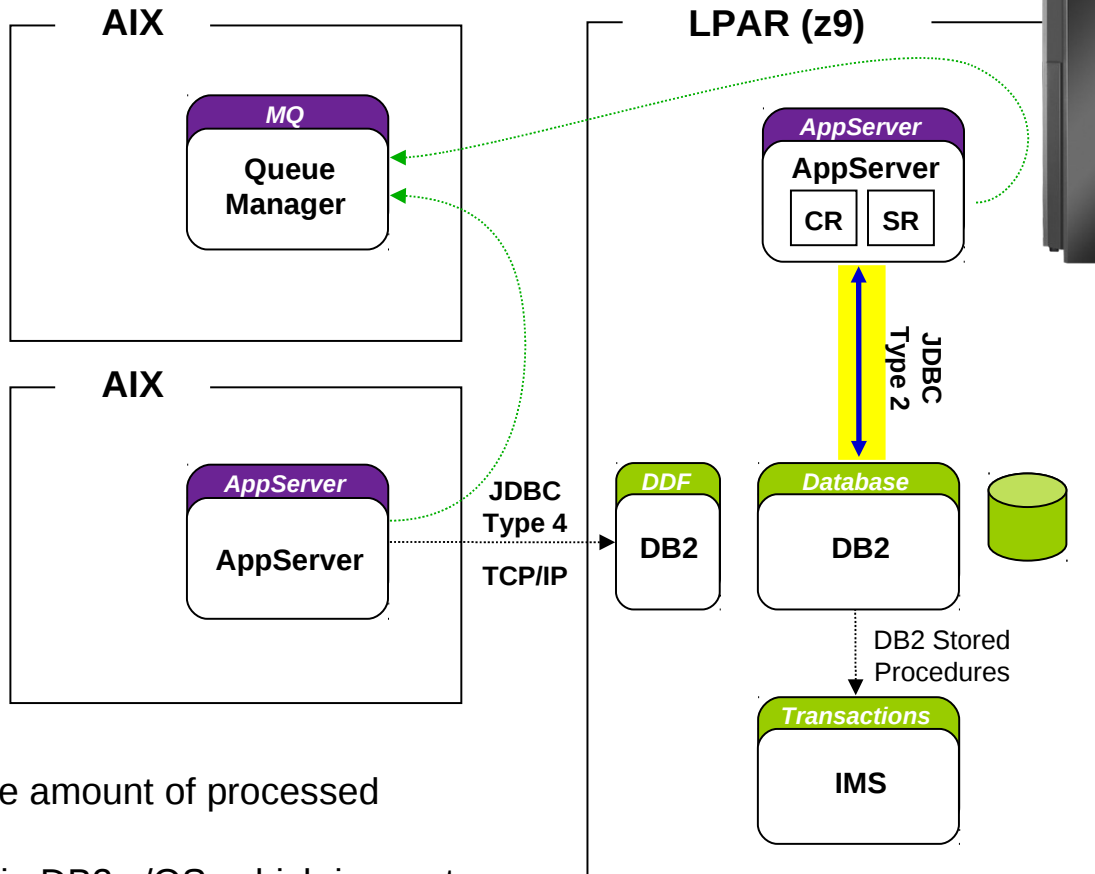


- **Effect of refactoring business logic to be co-resident with z/OS data:**
 - Average CPU time per EJB transaction was reduced by over 77%
 - Number of bytes of data transferred per EJB transaction was reduced by 99%

<http://www.ibm.com/support/techdocs>, Optimizing WebSphere Performance on DB2, WP100558

WAS z/OS Compute Grid PoC Architecture

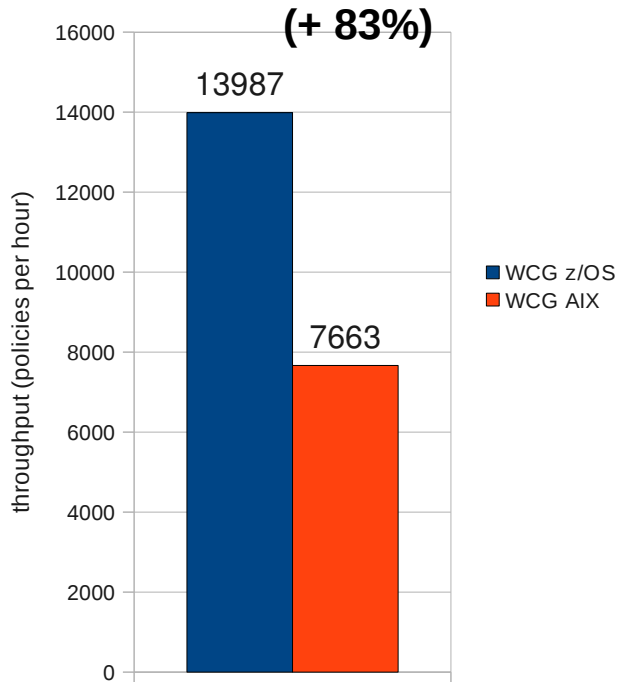
An insurance company has an existing WCG infrastructure on distributed AIX with data intensive JEE applications, which accesses DB2 z/OS via JDBC type 4. Within the scope of this PoC the Java batch job, which calculates the dynamic of the accident insurance is evaluated on WCG z/OS using JDBC type 2 cross memory adapter.



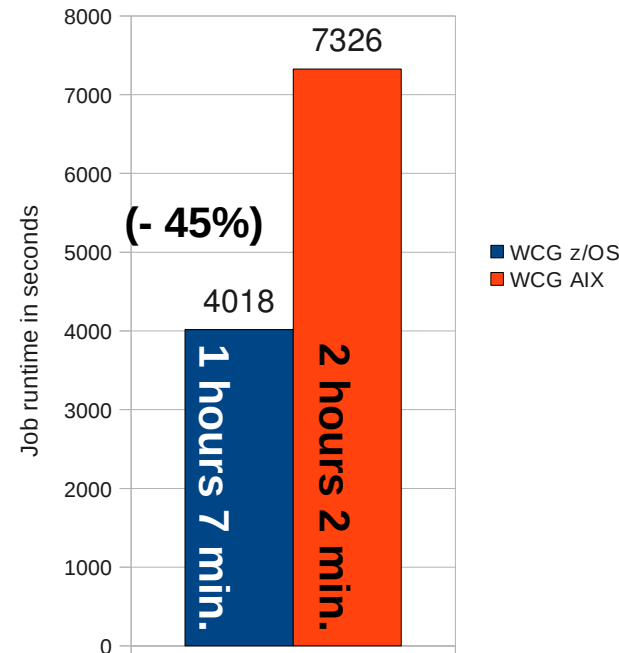
- ✓ Same job with same amount of processed insurance policies
- ✓ Same data basis in DB2 z/OS, which is reset after each run

WAS z/OS Compute Grid PoC Results

Throughput platform comparison
based on 15592 policies



Job runtime platform comparison
based on 15592 policies

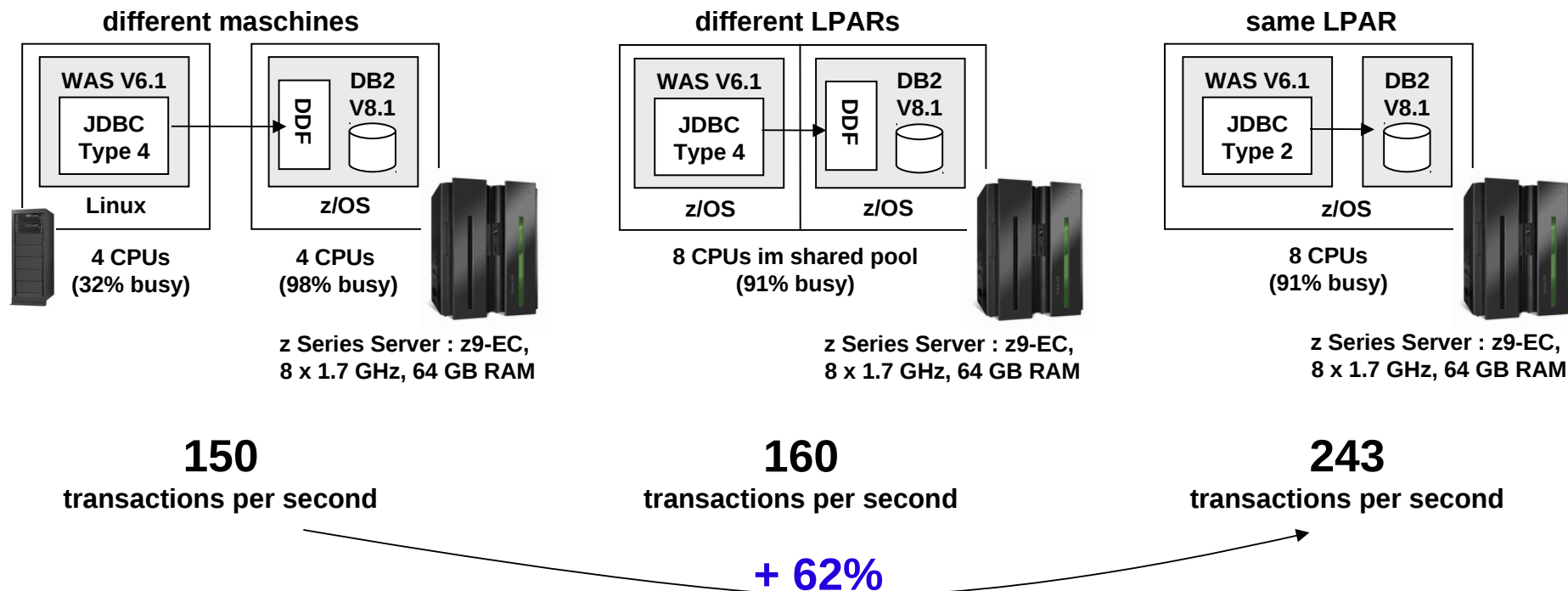


The same job runs on both platforms WCG AIX and WCG z/OS. Because of the proximity to data in DB2 z/O the job, which runs on **WCG z/OS** has **83.5% more throughput** and the **job runtime is shortened by 45%.**

Result of the WebSphere z/OS Co-location Benchmark (1)

If the data for an existing Java Enterprise applications are located in **DB2 z/OS**, it makes sense from a technical as well as financial point of view **to co-locate WAS z/OS and DB2 z/OS** in the same LPAR. In this **official benchmark the throughput has been increased by 62% by co-location**.

This throughput could be realized with **Cross-Memory Communication** (JDBC Type 2) from WAS z/OS to DB2 z/OS . The Overhead through network protocols will be eliminated. As a consequence the cpu consumption can be reduced dramatically.

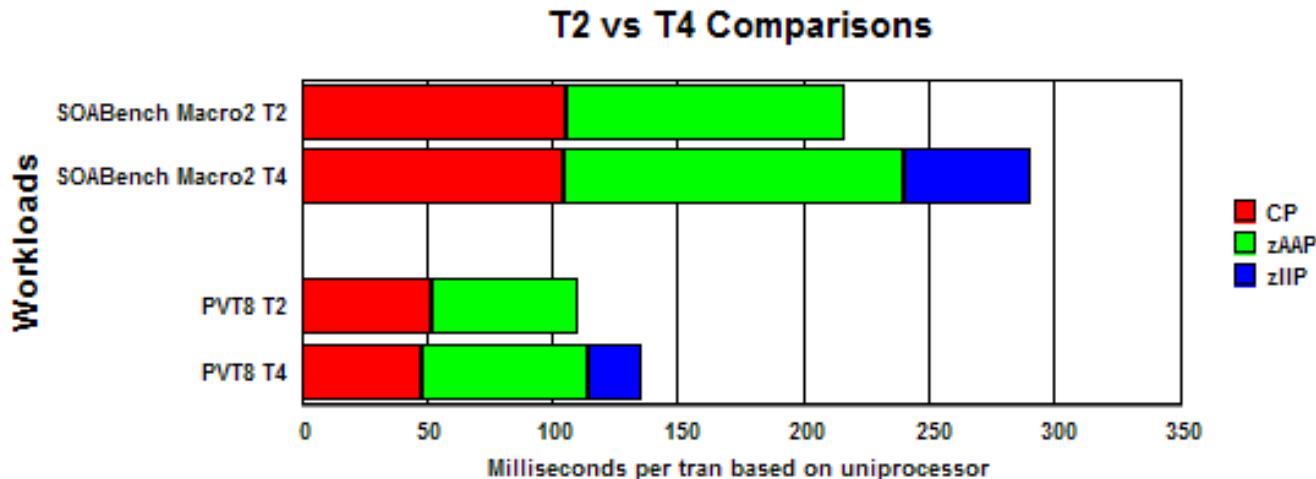


Quelle: WebSphere z/OS – The Value of Co-Location Benchmark

<http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101476>

Result of the WebSphere z/OS Co-location Benchmark (2)

Compared to a database connection over TCP/IP (JDBC Typ 4) Cross-Memory connections (JDBC Typ 2) could reduce the overall CPU consumption dramatically.



These values represent the utilization of the CPs, zAAPs and zIIPs on the system in total while the workload is processed with a constant transaction rate. This figure doesn't represent neither the enclave time for the application nor just the JDBC access to DB2.

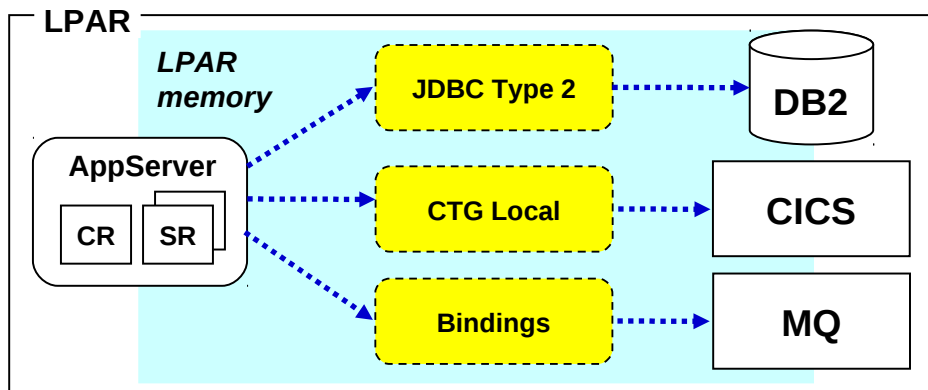
Quelle: WebSphere z/OS – The Value of Co-Location Benchmark

<http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101476>

Exploitation of Cross-Memory Communications

Any time client and target are in the same LPAR, there's an opportunity for cross-memory exploitation. Let's look at a few examples:

Data Access

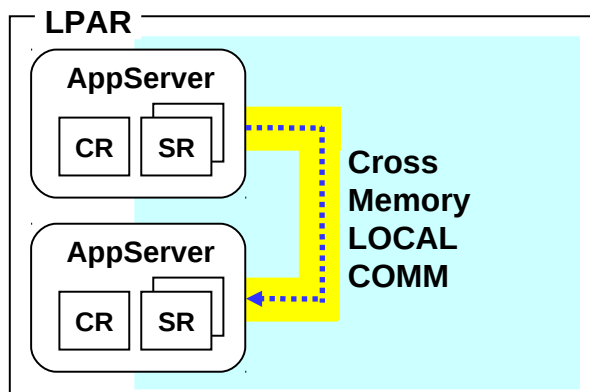


Benefits:

- Cross memory speed
- Security ID propagation (no alias)
- Exploitation of RRS
- Avoid serialization of parameters
- Avoids SSL overhead
- Single thread of execution

LOCAL COMM

Used for IIOP flows between servers on the same LPAR.



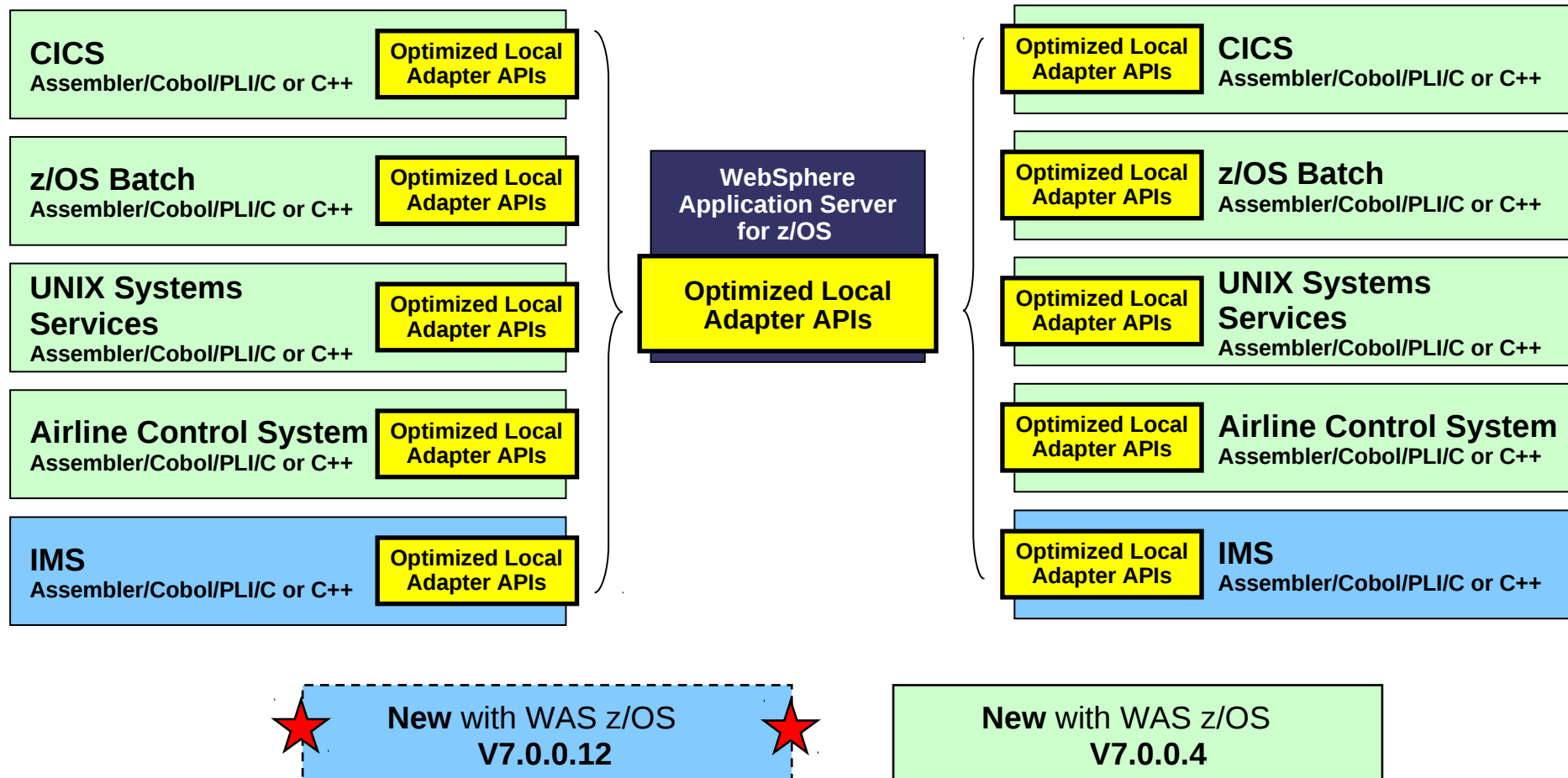
Benefits:

- Avoids IP stack entirely
- Avoids SSL overhead
- Very fast, very secure

**Extension to
Local Comm: new
Optimized Local
Adapters ...**

Cross-Memory: New Optimized Local Adapters (WOLA)

The new WOLA Adapters allow a bi-directional Cross-Memory communication between WAS and CICS, z/OS Batch and USS.



Cross-Memory: New Optimized Local Adapters (WOLA)

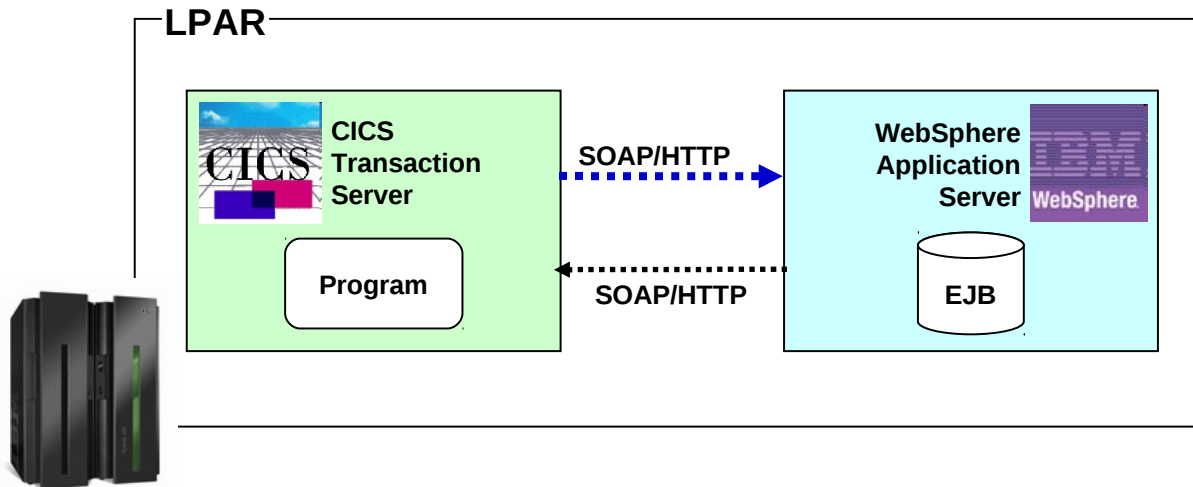
The new WOLA Adapters allow a bi-directional Cross-Memory communication between WAS and CICS, z/OS Batch and USS.

Advantages:

- Based on local Cross-Memory communication
- Based on bi-directional WAS or feasible incoming connections
- Significant efficient and faster then other access methods to Backend-Systems
- The CICS Transaction Gateway is no longer necessary for WAS z/OS

A Comparison of CICS Invoking WAS Web Services

A recent benchmark compared the case of web services into WAS against invoking the EJB using WOLA:

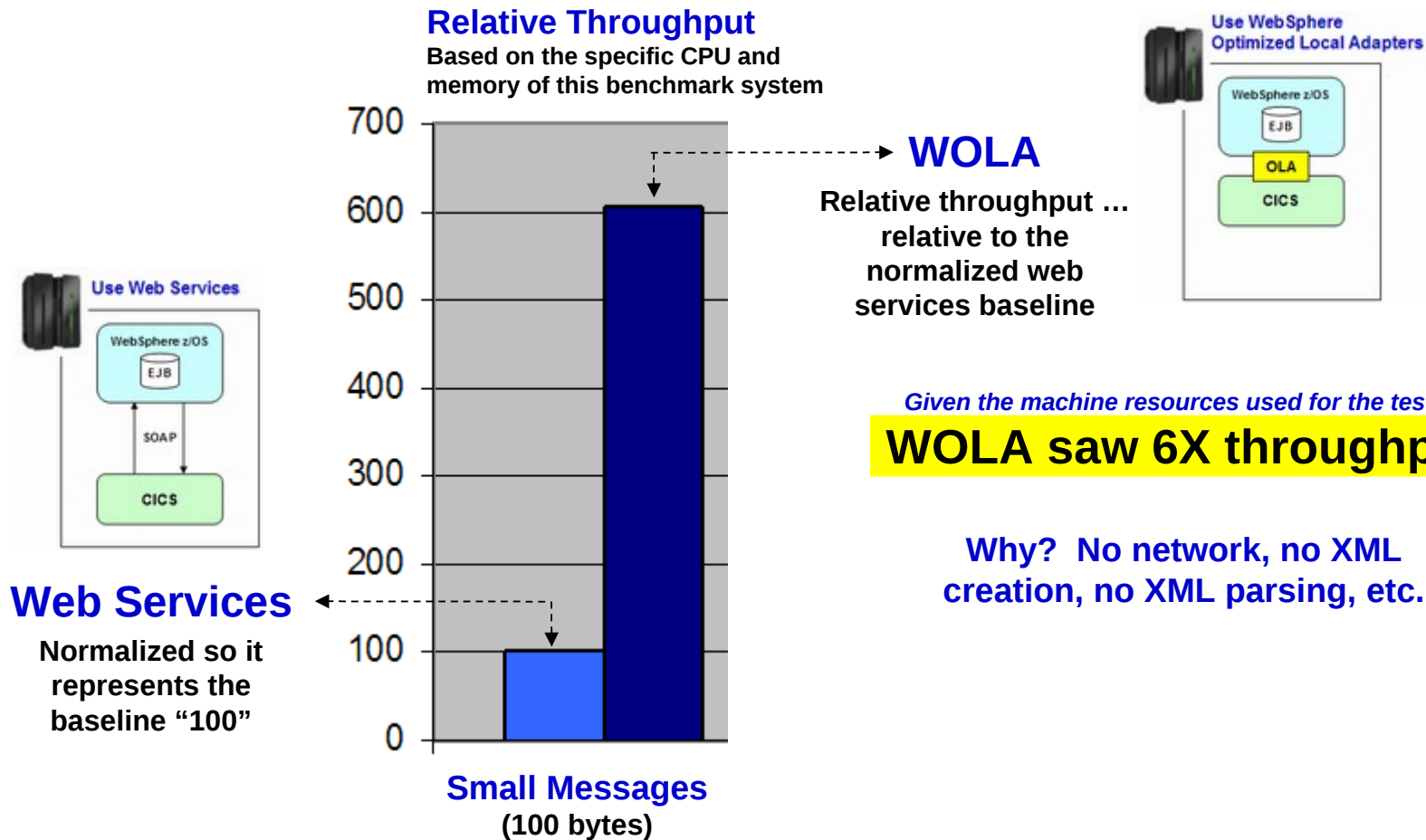


Some notes:

- We are not suggesting WOLA can or should replace web services in all cases. But in some cases it might well make sense
- In this test CICS and WAS on same LPAR, so TCP is optimized between the two
- The test was a relatively simple echo application

Relative Throughput for 100 Byte Message

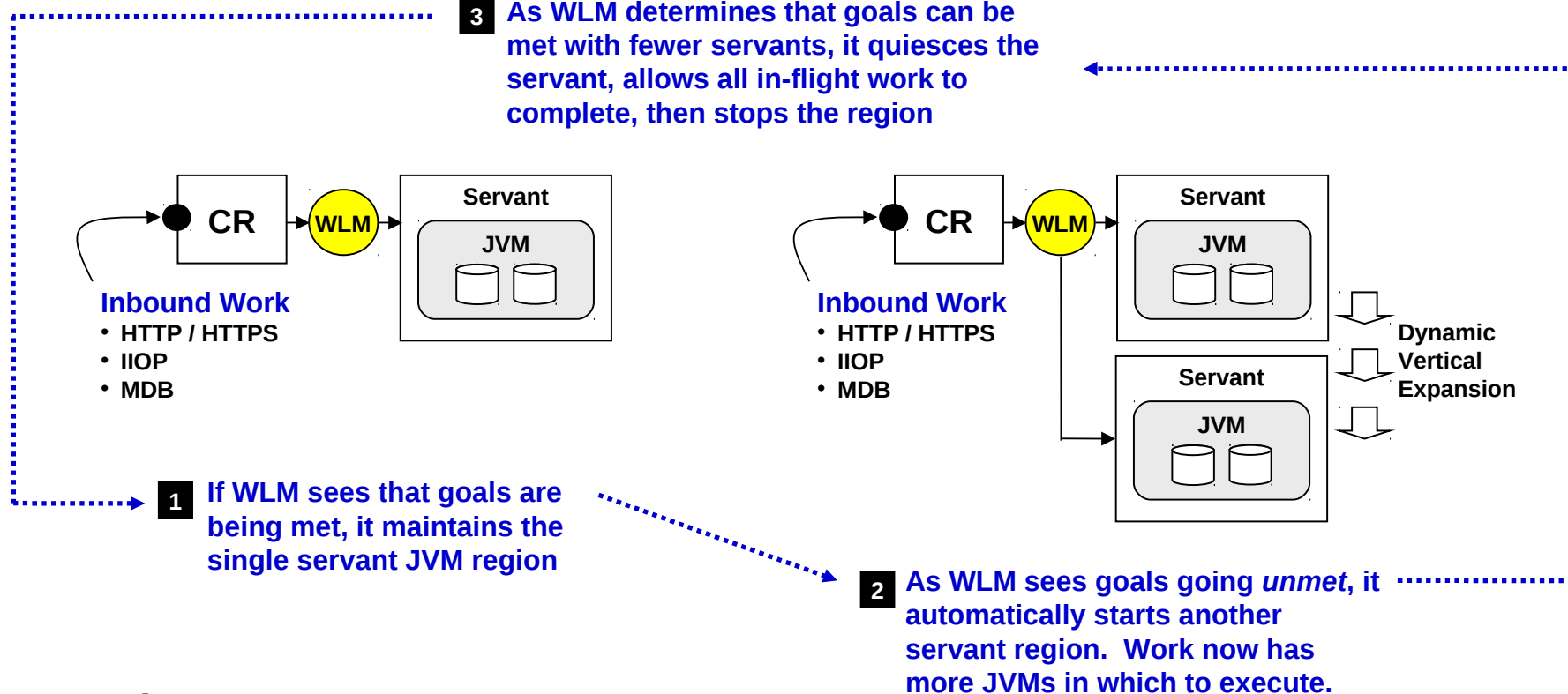
The following chart serves as our starting point ... a comparison of relative throughput based on a 100 byte message exchange:



Intelligent Dynamic Capacity Expansion

This is the “vertical scaling” capability of the multi-Servant structure. If allowed, WLM will start additional servant regions if it sees unmet goals:

3 As WLM determines that goals can be met with fewer servants, it quiesces the servant, allows all in-flight work to complete, then stops the region

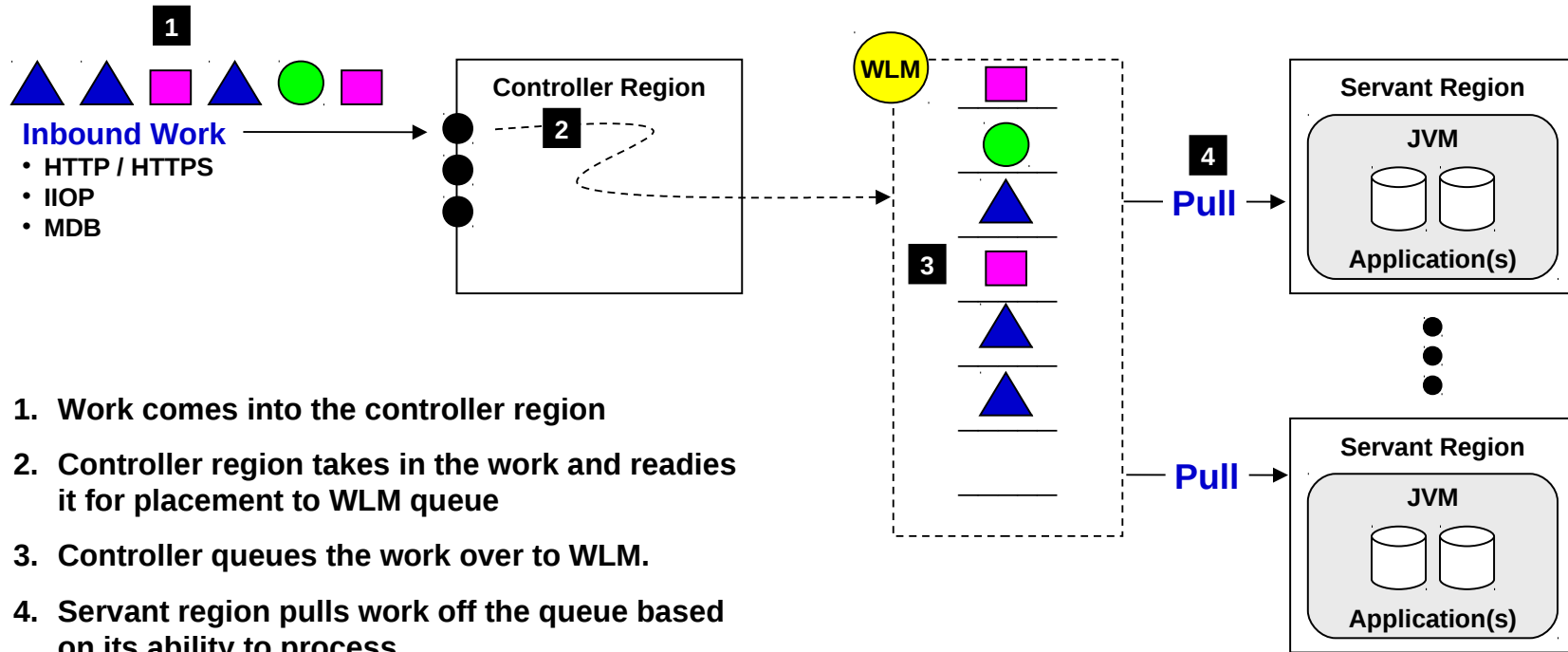


Key Points:

- The minimum and maximum number of servants is configurable. Default: Min=1, Max=1
- We see distributed WAS users trying to do something similar by configuring a “vertical cluster” to provide duplicate JVMs on a server box. Not quite the same -- no WLM assist of that

Intelligent Workload Flow Control

This is the WLM queueing mechanism that exists between the CR and the SR. It creates a “pull” model that prevents overwhelming an application JVM:



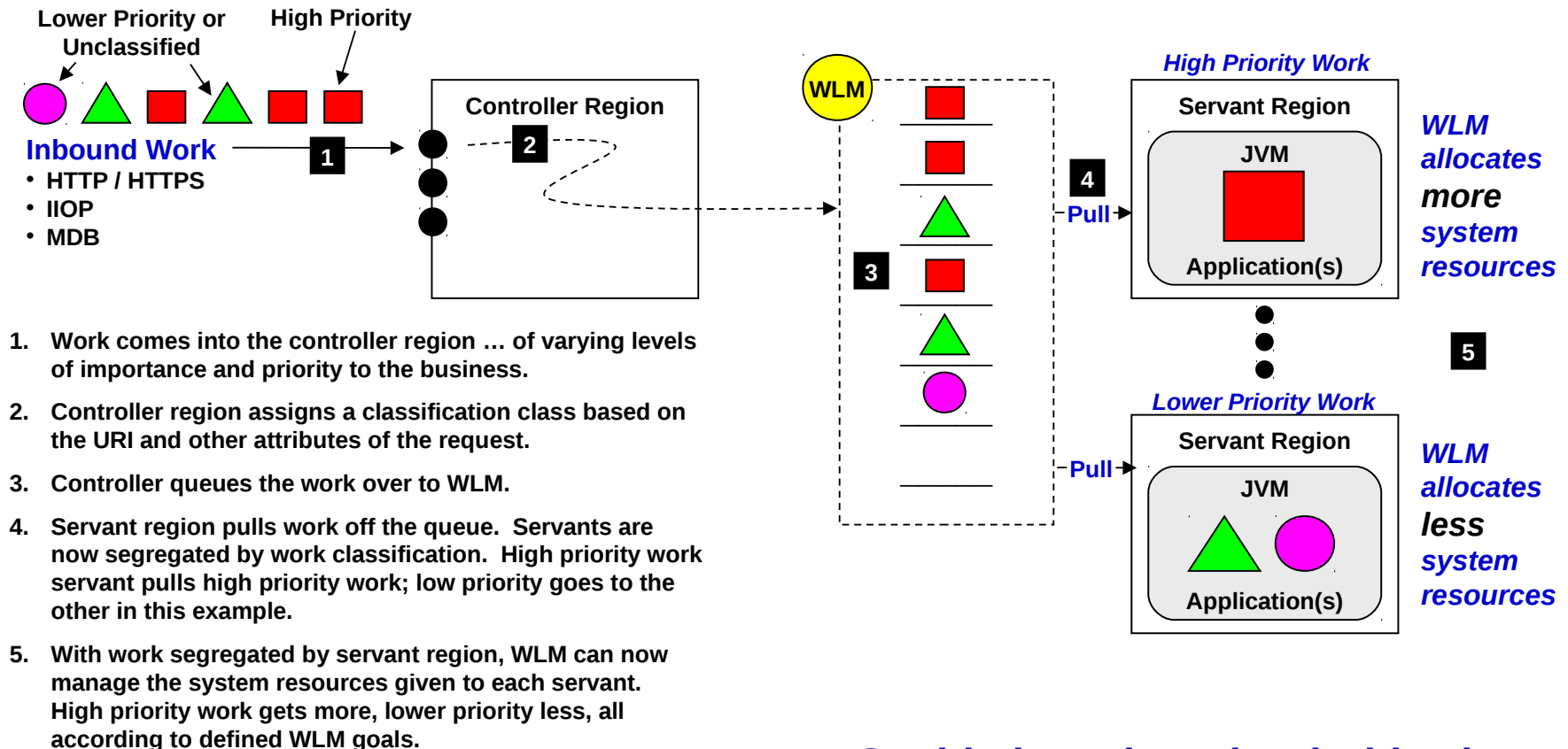
1. Work comes into the controller region
2. Controller region takes in the work and readies it for placement to WLM queue
3. Controller queues the work over to WLM.
4. Servant region pulls work off the queue based on its ability to process

Servant can't be overwhelmed

Servant only takes what it can. Controller will take in and queue up what can't be handled immediately.

Intelligent Management of Mixed Work in Server

This involves inbound work being given a “Transaction Classification.” With that, the CR can direct work to servants and WLM can manage:

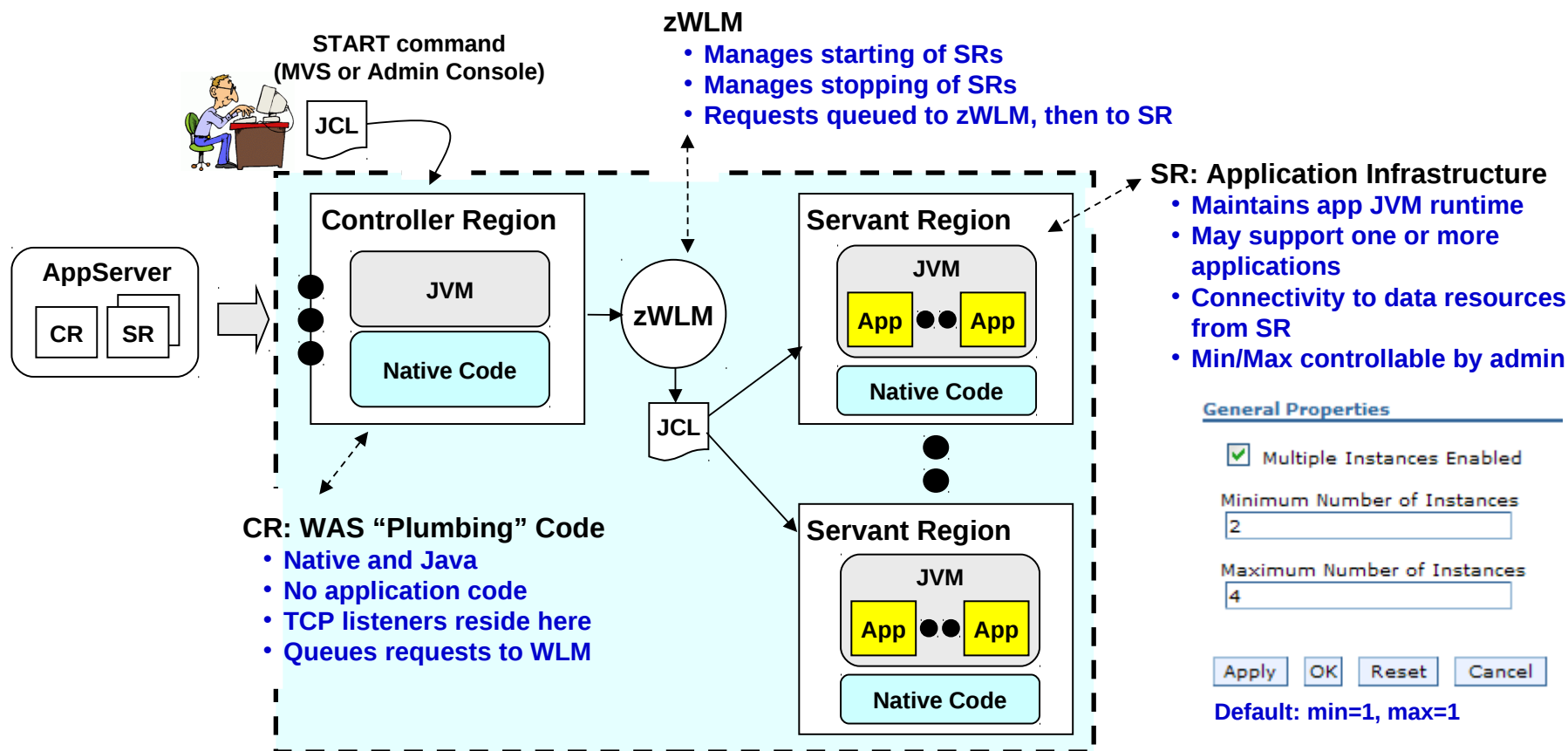


Sophisticated Work Prioritization

On other platforms this can only be done by allocating work to separate servers. No WLM there to manage at this level.

The Controller / Servant Architecture

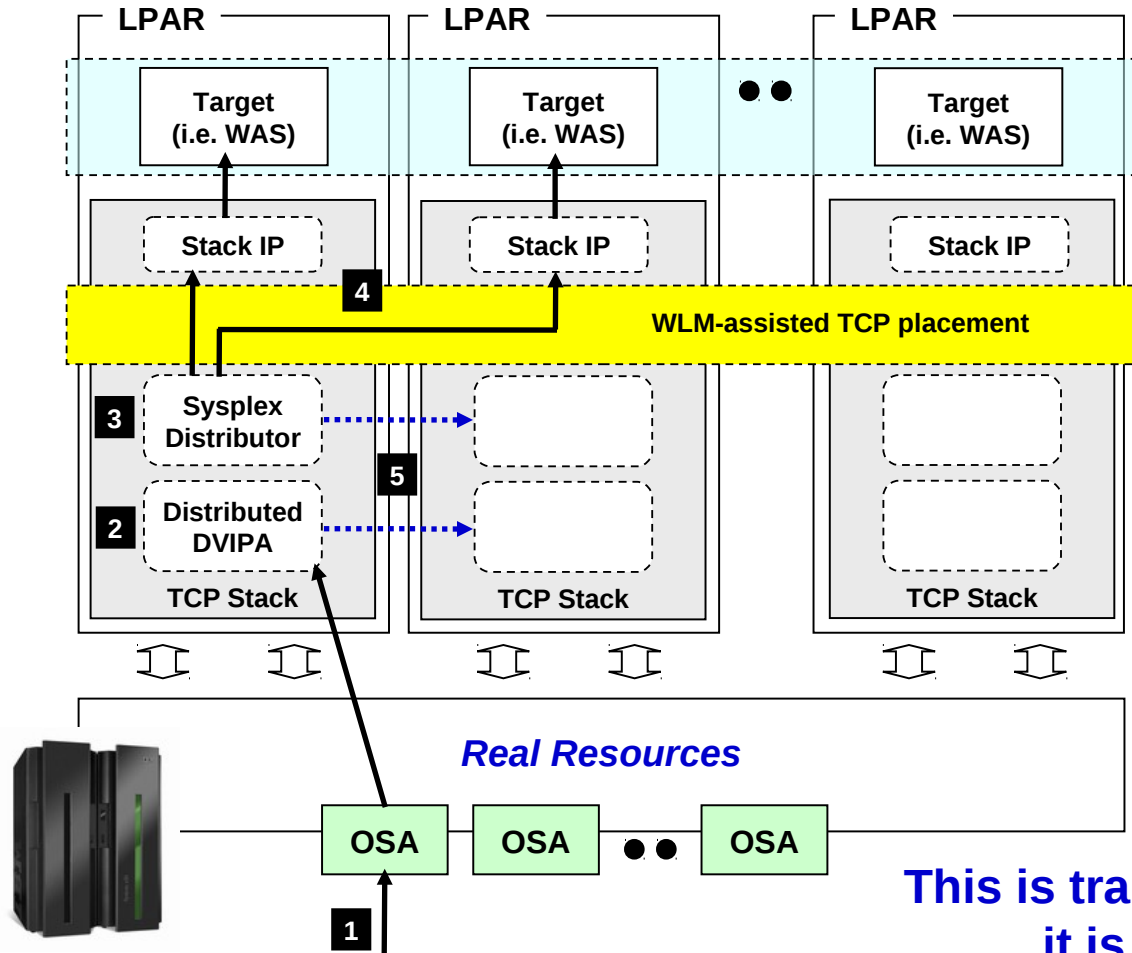
This is a unique architectural element to the WAS z/OS design. No other platform has this design because no other platform has WLM**:



** WebSphere on distributed uses the phrase "Workload Management" but it's not the same as zWLM

Virtual IP and Sysplex Distributor

Is a function of TCP on z/OS which allows you to “hide” duplicated resources behind a single IP address with WLM-assisted TCP connection placement



What's going on in this picture:

1. Clients in the world point themselves at a “generic” IP host name. Routers resolve that to one of the OSA adapters on the machine.

Note: there are ways to have redundant OSA adapters for availability

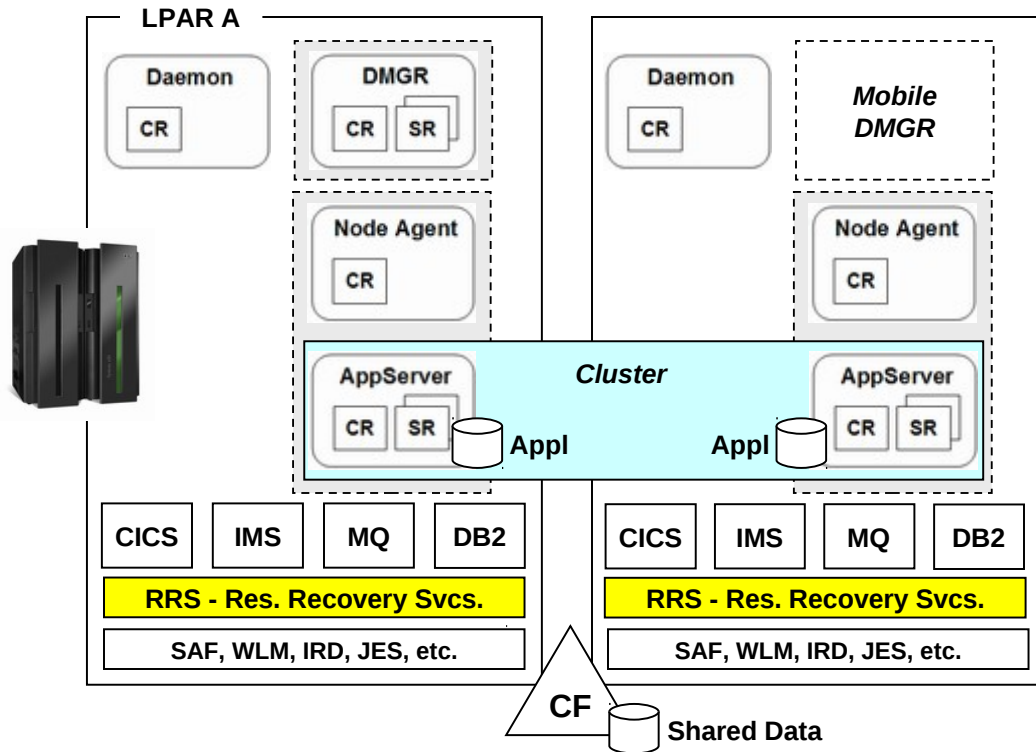
Note: it's not shown on this picture, but WLM can also advise some off-board Cisco routers.

1. Request is mapped to the TCP stack in the Sysplex that's hosting the Distributed Virtual IP (DVIPA) generic host.
2. Sysplex Distributor function determines which of the potential target LPARs is the best candidate to receive new work at that point in time.
3. TCP connection is made between client and the target
4. In the event of an outage of the hosting LPAR or TCP stack, the DVIPA and Sysplex Distributor functions automatically move to a defined “next in line” stack.

**This is transparent to the application ...
it is passive in this process**

Exploitation of Resource Recovery Services (RRS)

Two-phase commit processing involves coordination of participants to make sure all are ready to commit. RRS plays that role in Parallel Sysplex:

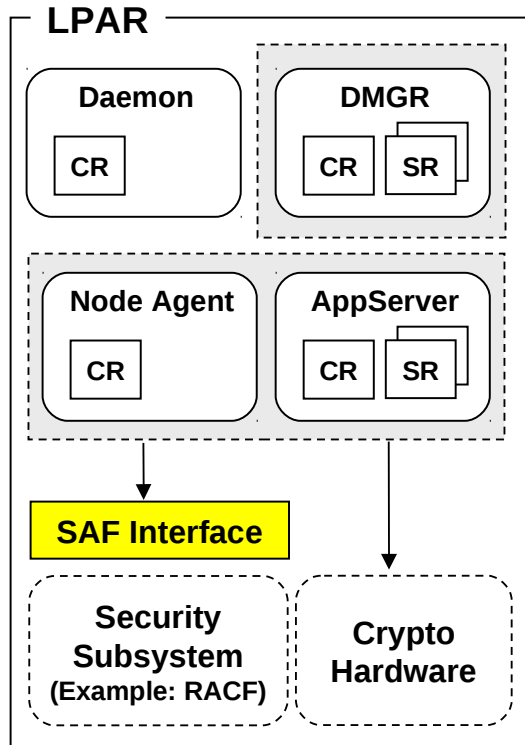


- We'll see this picture later when we discuss high availability
- WebSphere Application Server is a transaction manager ... it is able to initiate a transaction and have other resource managers (DB2, CICS, IMS) participate in the unit of work
- For two phase commit processing, *someone* has to play the role of syncpoint coordinator
- On z/OS and Parallel Sysplex that *someone* is RRS, which uses Coupling Facility data structures and patented recovery algorithms to provide very efficient failed transaction recovery
- WAS z/OS registers with RRS, as do resource managers. RRS handles the two-phase commit coordination

Another case of “below the specification line” exploitation of existing z/OS and Parallel Sysplex technology to perform a task in an optimized manner for the platform

Exploitation of SAF and Crypto

SAF is a security interface; Crypto is a hardware-assist processor for encryption and key storage on the System z and z/OS platform



SAF Security Subsystem

- Sysplex-wide integrated security repository
- Single location for security artifacts rather than scattered model
- IDs, groups, keyrings, certificates, EJB role enforcement
- Local access ... unlike LDAP, do not need to traverse network
- Extremely robust security model

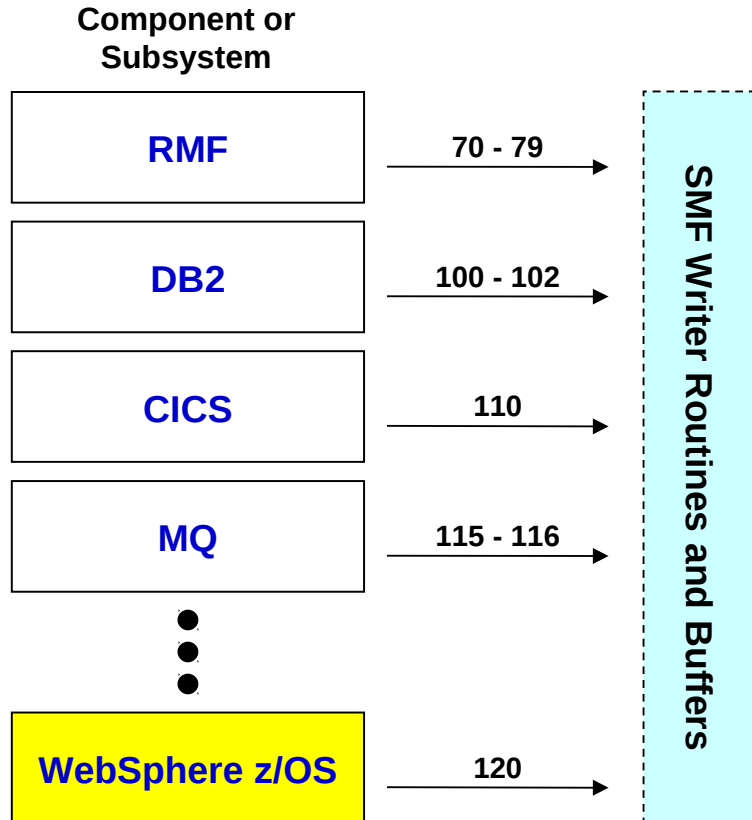
Crypto Hardware

- Hardware-assisted cryptographic encryption and de-encryption
- Extremely secure private key store management

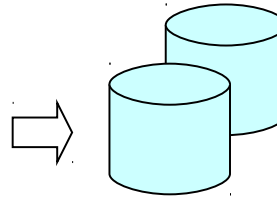
Properly configured, z/OS provides an extremely secure environment ... many say the most secure available

Exploitation of SMF

SMF is an activity recording facility of z/OS that allows subsystems to record key activity for analysis, management and accounting chargeback



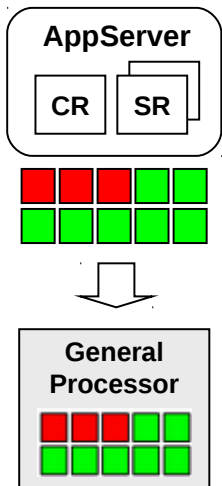
SMF Data Sets



- **WAS z/OS writes SMF 120 records**
- **With WAS z/OS V7, a new subtype was created: 9**
- **New SMF 120 subtype 9 provides better data with lower overhead cost**
- **New SMF 120 subtype 9 records complement and extend existing SMF from other subsystems, allowing a far better picture of what's going on**
- **Better data available for ...**
 - Activity analysis
 - Usage statistics
 - Accounting chargeback

Specialty engines

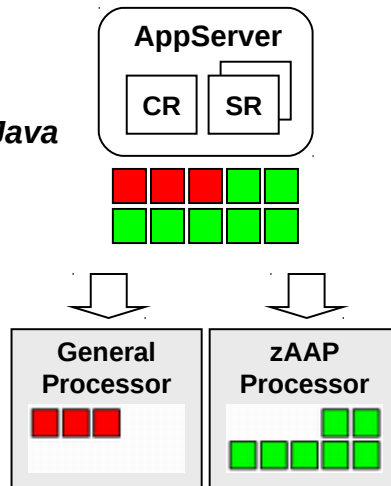
without zAAPs



Work:

■ = *Non-Java*
■ = *Java*

With zAAPs



zAAP on zIIP

WAS z/OS workload can run on zIIPs

- zAAP eligible workload runs on zIIPs
- z9, z10 or z196 Hardware
- z/OS V1.11 or z/OS V1.9 or V1.10 with the PTFs for APAR OA27495 installed
- This function is not available to z/OS LPARs if zAAPs are installed on that machine

<http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=an&subtype=ca&appname=gplateam&supplier=897&letternum=ENUS209-242>

JDBC Type 2 zAAP offload

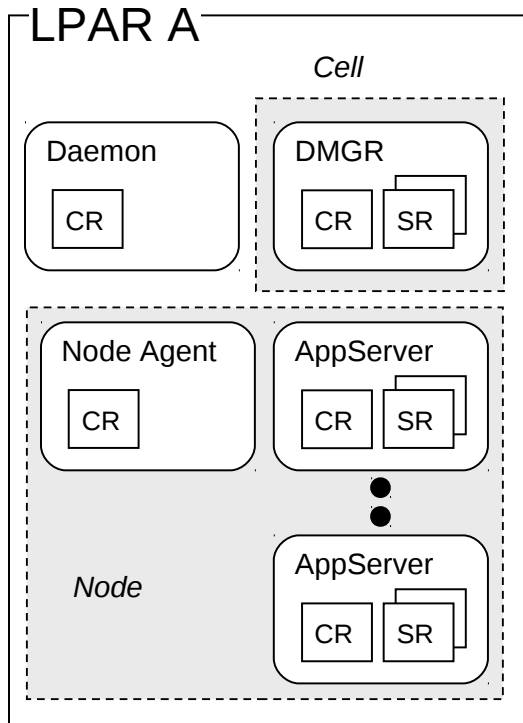
JDBC Type 2 workload can be offloaded to the zAAP processor

APAR (**PTF UA45546 for z/OS 1.9; PTF UA45547 for z/OS 1.10**) is what provides the updates to the z/OS dispatcher to allow offload of Type 2 processing to the zAAP.

The architecture of a WAS z/OS environment

How It's Implemented on z/OS

The developers of WebSphere on z/OS chose to implement the function of WebSphere Application Server as a series of z/OS started tasks:



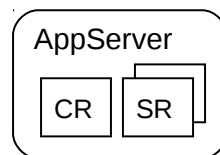
It provides an implementation very familiar to z/OS system administrators

It maps very well to z/OS utilities such as automation and monitoring

But what are those things in the picture?

- The small boxes inside the curved boxes
- CR and SR
- DMGR? Node Agent? Node? Cell?

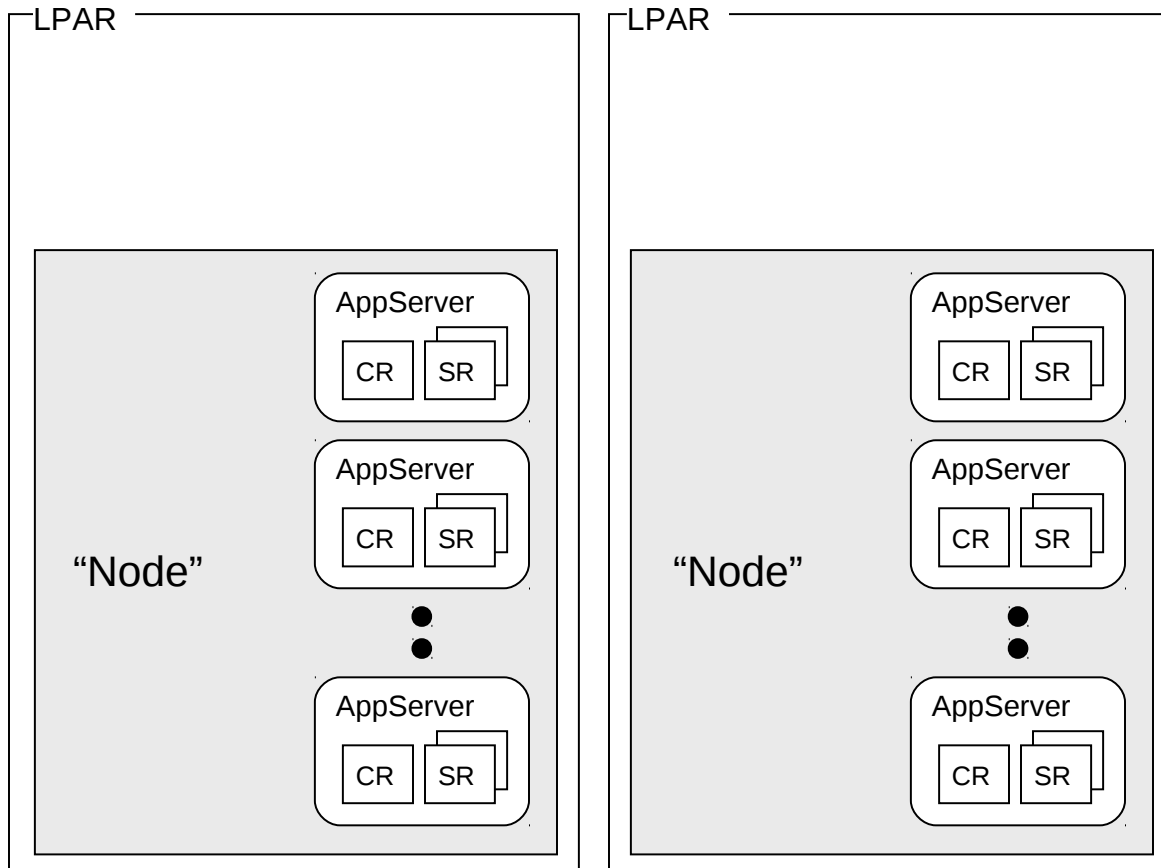
We'll start with the servers where your applications run:



This icon will be used throughout this presentation to represent the “application server,” which is where applications run. The small boxes inside are a design unique to z/OS

Multiple Application Servers and the Concept of a “Node”

There are many reasons* for creating multiple application servers. A “node” is simply the logical collection of application servers on an LPAR:



Key Points:

- Nodes are a logical thing ... it's not a started task
- They logically organize application servers on an LPAR
- No architectural limit to the number of application servers in a node; limited only by system resources
- Rule: node must stay on an LPAR; it can't span LPARs in a Sysplex

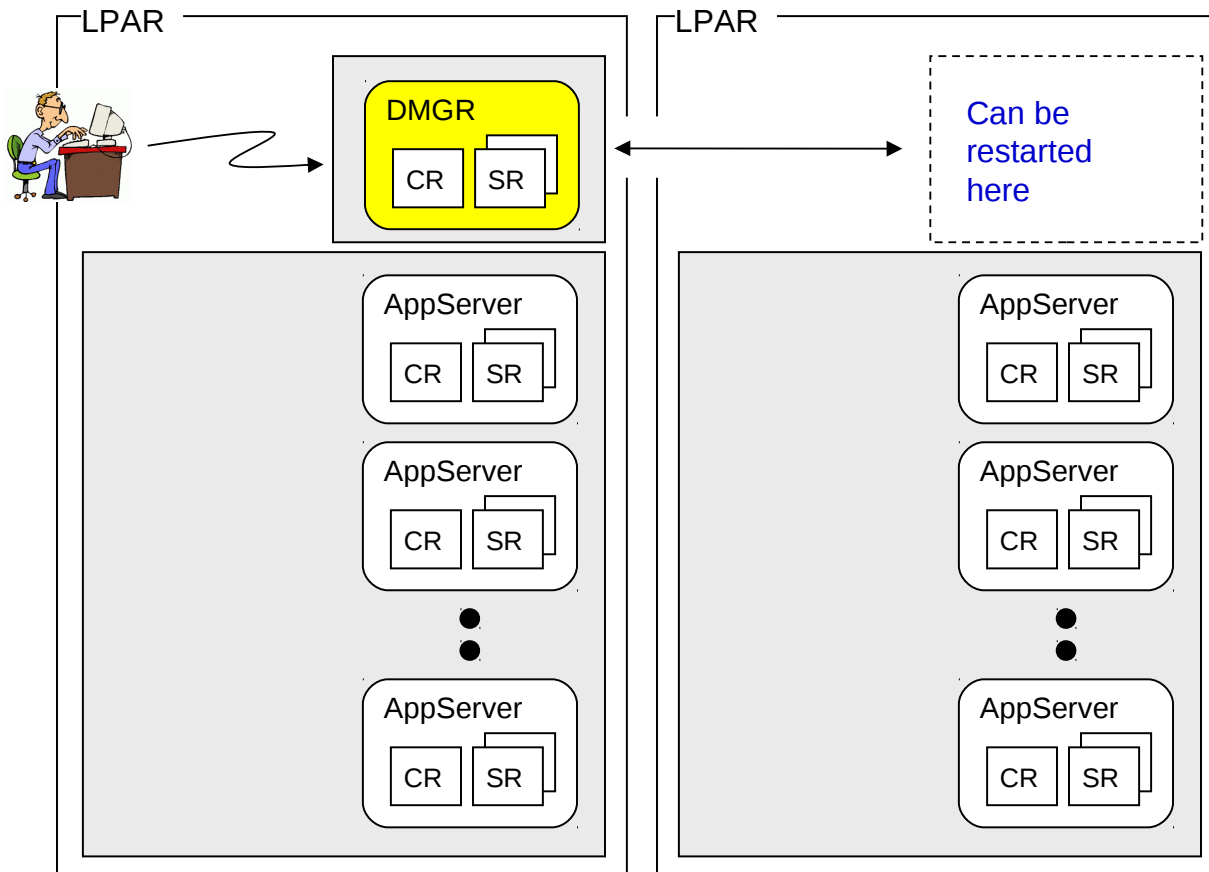
What's the point?

(We'll see in a moment)

* Requirement for separation of application. Applications have different custom JVM settings. Different performance requirements

First -- The Administrative Application Server

There is a special purpose server called the “Deployment Manager” that runs the Administrative Console:



Key Points:

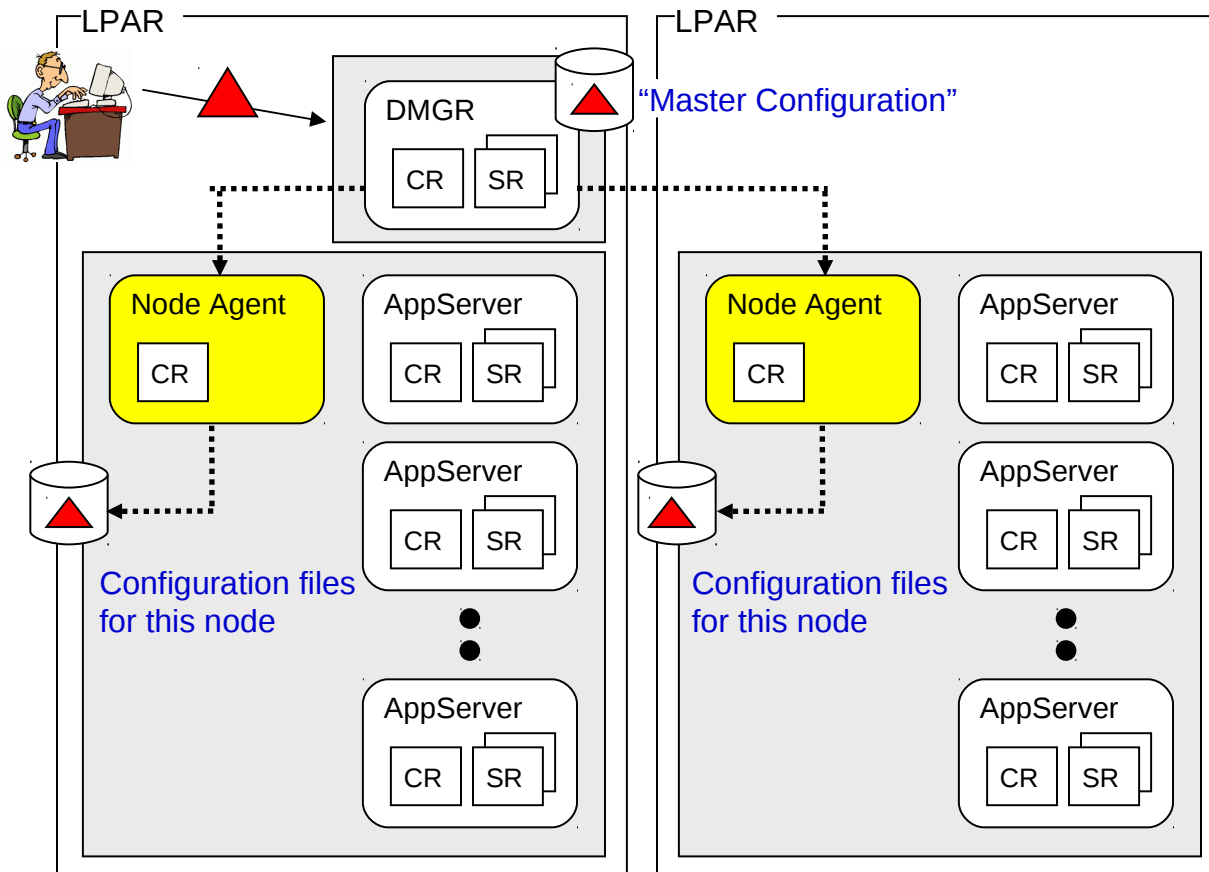
- DMGR structure like application server -- one CR and one or more SRs.
- Only the Administrative Console is allowed to run in this special purpose server.
- The Administrative Console is really just a very smart web app that knows how to translate your configuration mouse clicks into updates to XML configuration docs.
- Properly configured, the DMGR can be started on other LPARs
- Only one DMGR is allowed per “Cell” (which we’ll describe soon)

Something is missing ...

Node Agents ...

Node Agents -- Act on Behalf of DMGR in the Node

Node Agents are single-CR structures that update the node's configuration on behalf of the DMGR, which sends updates to the Node Agent:



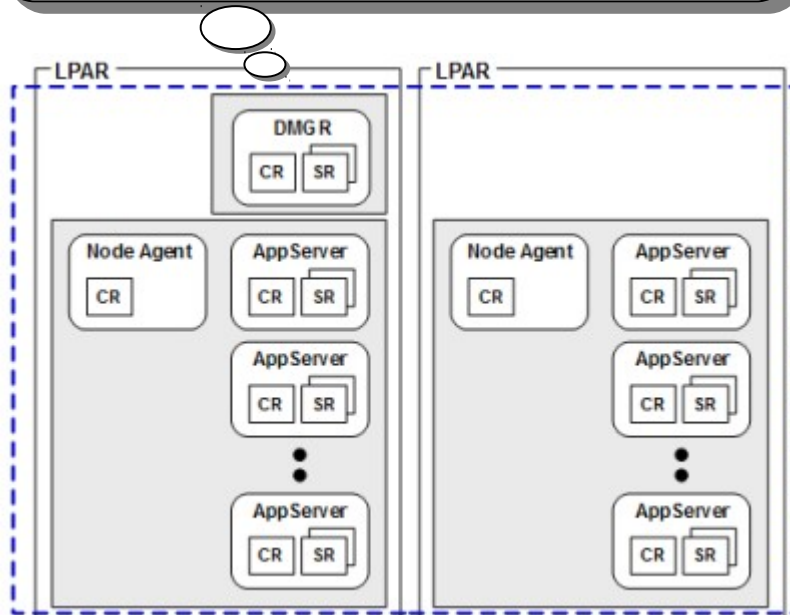
Key Points:

- WebSphere is a distributed architecture -- this allows the configuration to be on separate machines and still work.
- This design frees the DMGR from requiring write access to each node's configuration file system.
- Node Agents are just that -- agents that work on behalf of the DMGR to make the changes in the node.
- Act of copying down changes is called "synchronization"
- Trivia - DMGR maintains master copy of configuration, changes made there first, then copied out to the nodes.

Now We Can Introduce Concept of the “Cell”

The Cell is really nothing more than the extent of administrative control a DMGR has. In this example it controls two nodes on two LPARs ... that's the cell.

I am the administrative point of control. I am aware of two nodes I manage ... I see information about them in my XML files. They are aware of me as my information is in *their* XML files.
My "cell" includes all that I manage.



A logical construct ... not a started task

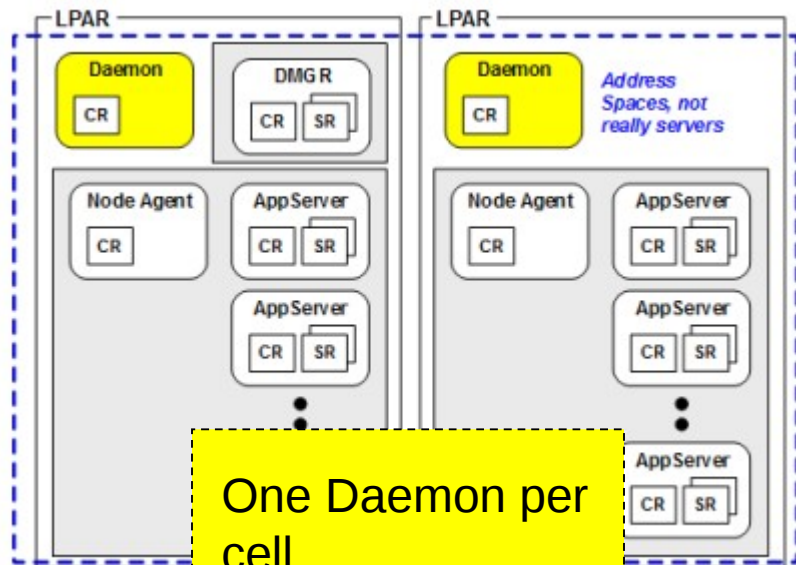
Is important because it is often used as the means of isolating at the security and purpose level -- test, QA, production

The cell is automatically created and automatically expanded as more nodes are built and added to it.

- ❑ Multiple cells allowed ... per LPAR and per Sysplex
- ❑ Cells *can* span to non-z/OS platforms. That introduces a bit of complexity, but it is possible.

Wrap-Up: Daemon Address Spaces

Daemons are not really "servers" -- no JVM, just an address space. They serve two main roles in a WAS z/OS ND Cell environment.

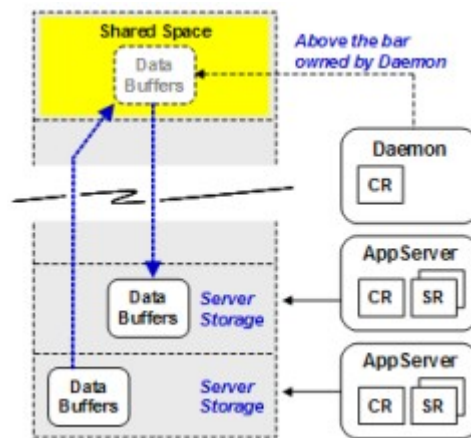


One Daemon per cell per z/OS image



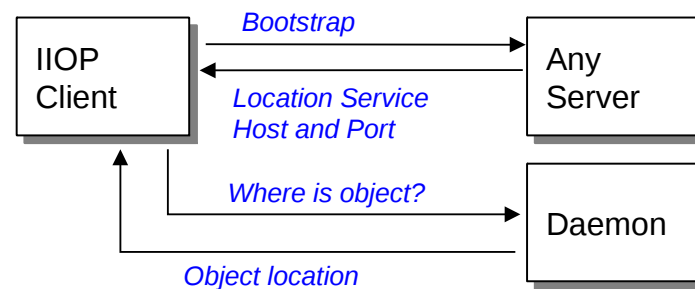
This is the rule of thumb to remember

Owens above the bar shared storage



Part of "Local Comm" function. We'll see more in zDiff section. Daemon owns the space, maintains control blocks, and performs the task switching between server address spaces

Hosts "Location Service" for cell

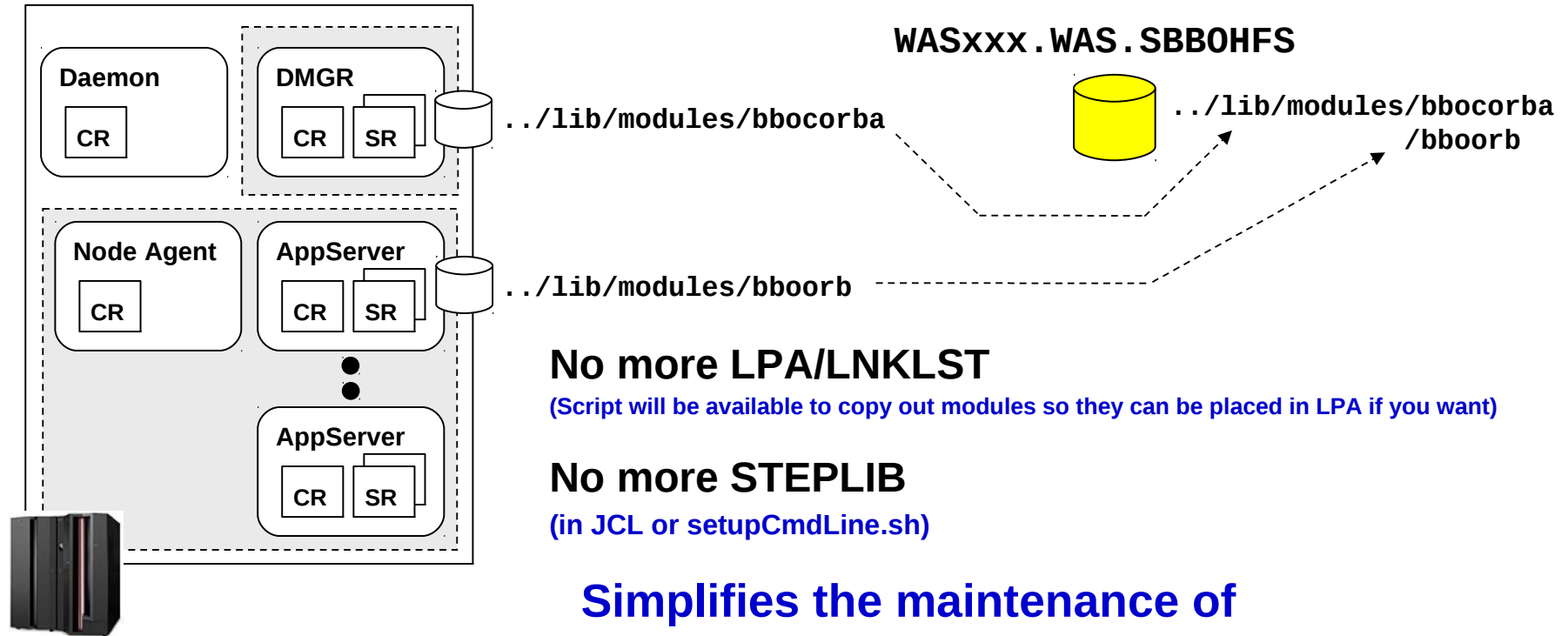


Installation and what's new ...

New features with WAS z/OS V7.0

No more Load module libraries

The load modules will now be included in the HFS under `/lib/modules`.



No more LPA/LNKLST

(Script will be available to copy out modules so they can be placed in LPA if you want)

No more STEPLIB

(in JCL or setupCmdLine.sh)

**Simplifies the maintenance of
WebSphere on z/OS.**

**Eliminates possibility of accidental mis-
match between PDSE and HFS**

★ Needs APAR OA25489 ★

Overview of zDiff Items

Background: through feedback at forums such as zBLC, customers have indicated they wish WebSphere on z/OS to exploit the platform strengths.

New SMF 120 Subtype 9 Record

Issue: Previous SMF 120 records were very costly and didn't provide sufficient data, thus rarely used

Solution: New subtype reduces overhead and provide meaningful information

Thread Hang Recovery

Issue: In the past a hung thread resulted in a servant abend, disrupting operations

Solution: New mechanism that attempts to "shake loose" hung threads

FRCA Caching in Controller

Issue: Caching of objects in WAS cache not as efficient as FRCA, which is handled at lower TCP level

Solution: Exploit FRCA API out of controller. FRCA now used as an external cache of DynaCache

DCS signalling over XCF

Issue: High Availability Manager (HAM) signalling over TCP incurs sizeable overhead. For large server topologies there's an "N-squared" problem -- signalling overwhelms system.

Solution: Exploit XCF with new "plugin" that allows DCS (HAM) signalling over Sysplex facility

Optimized Local Adapters

Issue: Desire for very fast and highly efficient local connectivity, particularly *inbound* to WAS

Solution: WebSphere Optimized Local Adapters (WOLA)

All of these are down at the "plumbing" layer and are intended to exploit the platform capabilities more directly. Higher level functionality common across platforms as it should be.

64 Bit Performance

64-bit JVMs provide relief from heap crowding, but initially came with a performance cost. Two new features match the 31-bit performance profile:


ibm.com/partnerworld/wps/whitepaper/systemz/java_websphere/performance

PartnerWorld > Products > Systems, servers, and storage > Technical >

Match 31-bit WebSphere Application Server performance with new features in 64-bit Java on System z

by Kishor Patil, Marcel Mitran, Jim Cunningham

Last updated: 2009-05-20

 [Download](#) the white paper (285 KB)



Compressed References

Function added to the z/OS Real Storage Manager (RSM) with APAR OA26294 provides a direct assembler interface which allow memory allocations in the 2GB (2^{31}) to 32GB (2^{35}) virtual address range. The JVM uses this API to allocate the heap in this virtual address range.

Large Page Support

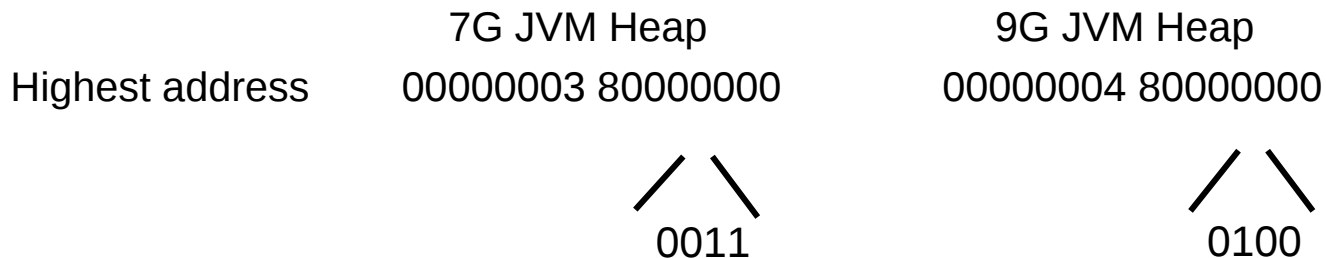
System z10 processors introduced support for 1MB pages (APAR OA20902 and OA25485 for z/OS 1.9). The 64-bit JVM can achieve performance gains by using large pages, which results in fewer Translation Look-aside Buffer (TLB) entries needed. Fewer TLBs needed for the data footprint of the JVM means more TLBs are available for the executable code. Fewer TLB misses in instruction fetches occurs, which enhances performance.

System z/10 with these two features allows a 64-bit JVM to operate with a larger heap and match the performance seen with the smaller heap 31-bit JVMs.

Compressed References – How do they work?

- Address space “hole” is range 2G to 32G (30G in size)
- “Compressed” JVM heap placed here
- Heap allocations aligned on double-word boundaries
 - ▶ Lowest 3 bits are always zero
 - ▶ Allows address to be shifted or compressed without losing data
 - ▶ Shift amount depends on the address

Examples:



New features with WAS z/OS V8.0

Summary

Skills



Existing Skills Carry Forward

Greater leveraging of skills across platforms

WebSphere Application Server z/OS Version 8.0

Updated
Standards
Support

Java 6.0.1

Improved
JVM

More
Granular
RAS

Installation
Manager

Data
Resource
Routing

HPEL

z/OS
Additional
Options

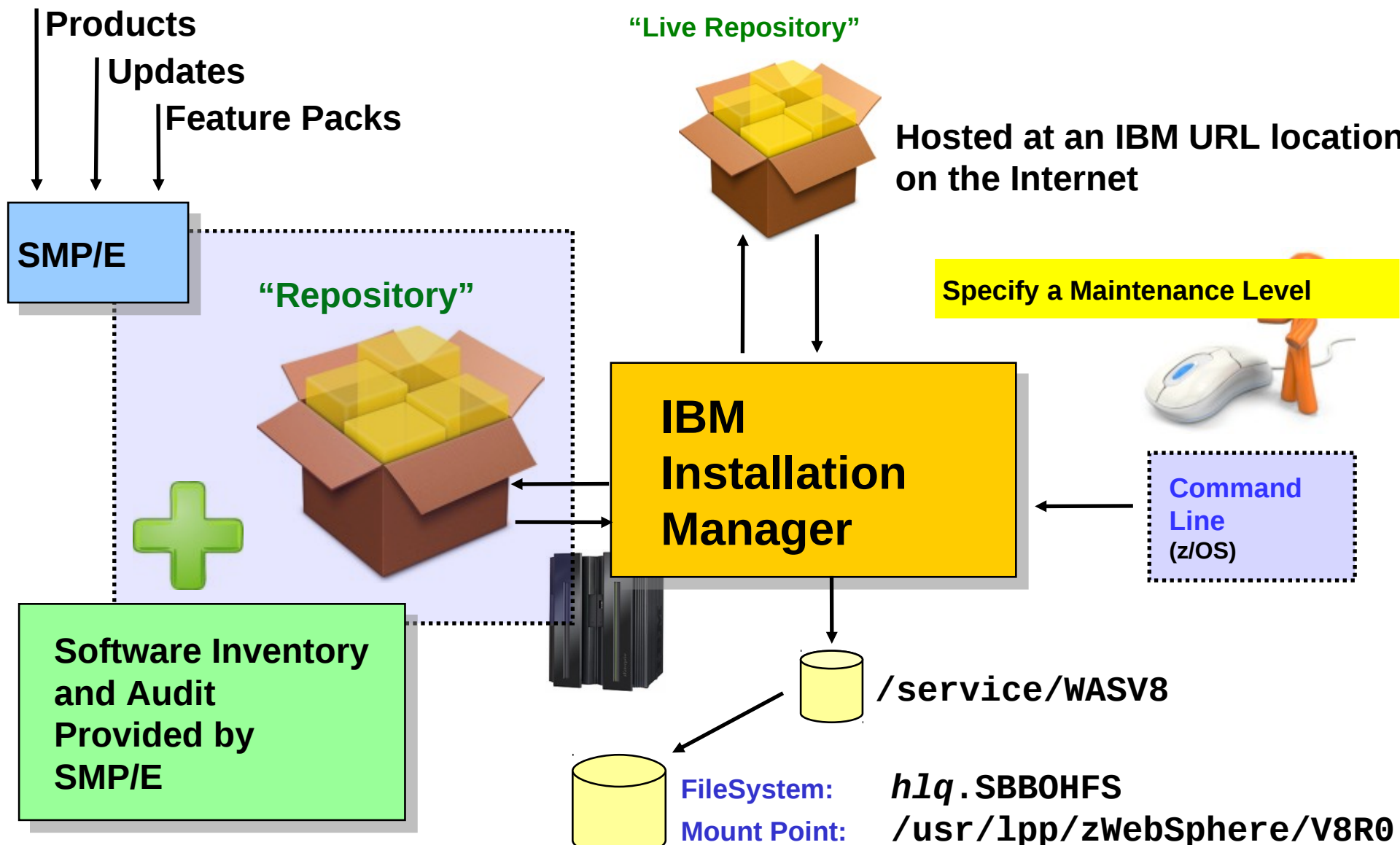
WOLA

A High Level Review of WebSphere Application Server for z/OS V8

The “Repository” and SMP/E

Here’s where SMP/E re-enters the picture **if you wish*** – it can be used to manage a local copy of the repository

*** Or you may opt to bypass SMP/E and a) use IBM hosted repository, or b) downloaded copy**

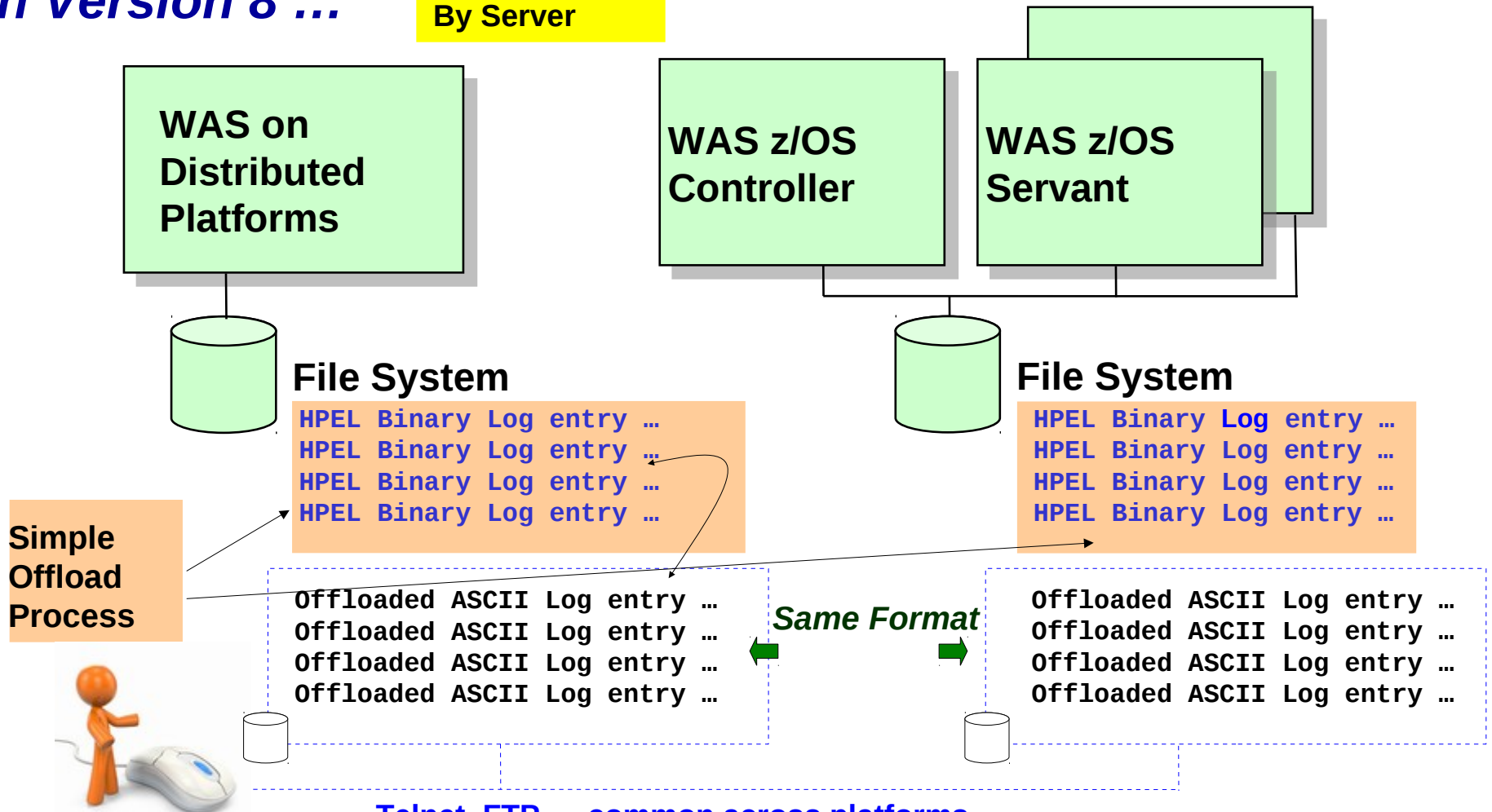


High Performance Extensible Logging

Writing output to logs can be resource intensive. HPEL makes this more efficient in a number of ways. Logs may now be common across platforms:

Configurable
By Server

In Version 8 ...



WAS
Administrators
and Developers

Telnet, FTP ... common across platforms

Allows better leveraging of Java developer and WAS administrator skills across platforms

New Level of Java Inside of WAS V8

The latest revision of the JVM inside of WAS z/OS. When discussing the performance benefits, it's important to keep a few things in mind:



System z z196

Performance benefits that result from the z196's **faster chip**, **larger and improved cache**, and **Out of Order Execution Pipeline**
regardless of the level of WAS

X



System z z10

Performance benefits that result from general improvements included in the **WAS V8 JVM** ...
regardless of the hardware

Y



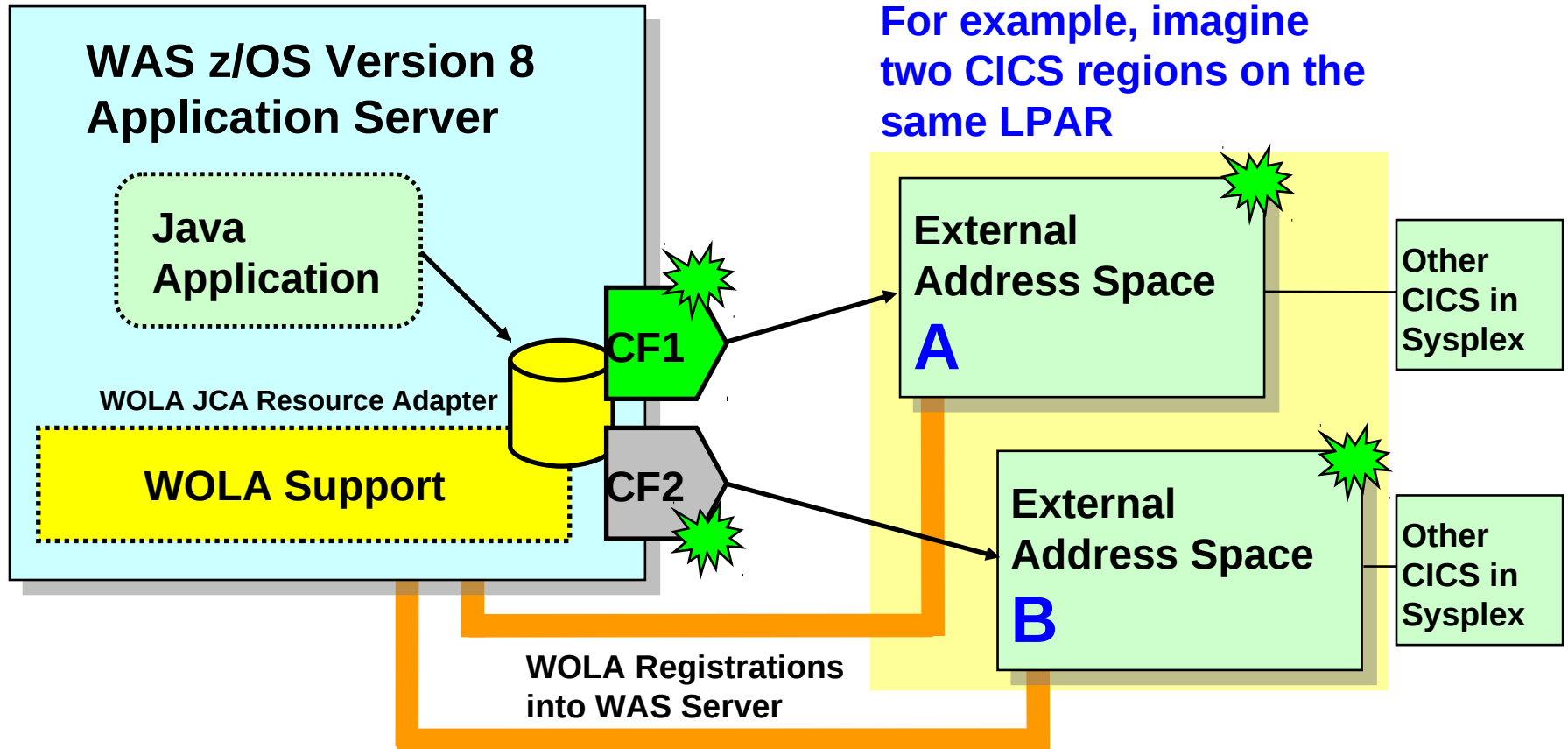
System z z196

Performance benefits that result from **taking advantage of the new instructions** in the z196 processor,
which requires WAS V8

Z

WOLA Enhanced Availability

WOLA is updated to exploit the same kind of connector failover we just saw. This will allow registration failover to alternate address space:



The same kind of detection of backend outage and automatic failover to backup connection factory as we just saw for JDBC

There are a few other WOLA enhancements as well ... see InfoCenter

*New features with the next WAS
Version – maybe:)*

The next WebSphere Application Server Version might include

- Liberty Profile fully supported on WAS z/OS with z/OS unique features right from the start
- Enhanced Focus on Mobile Application Development
- Changes in packaging of the product

You will get all the details later this day when the announcement was done:)

Link to announcement letter:

<http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=AN&subtype=CA&htmlfid=897/ENUS212-106&appname=USN>

***Ich mag den Mainframe, weil wir ein gemeinsames Talent haben:
Wir können viele verschiedene Dinge erledigen – gleichzeitig!***

Isabel Arnold, Technical Sales IBM System z

