Diego Cardalliaguet Gómez-Málaga

WebSphere for System z CTP
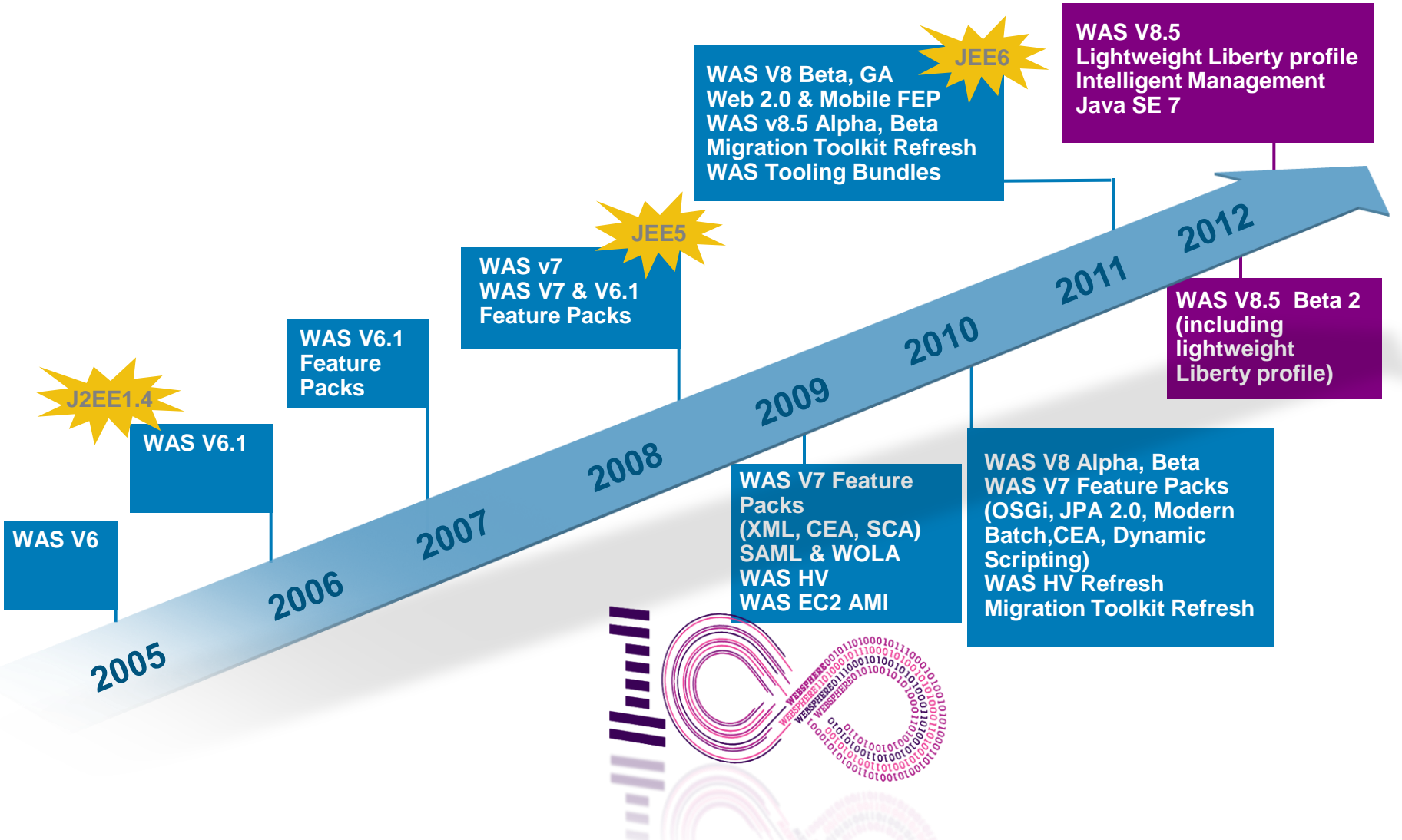
WebSphere Application Server V8.5 Technical Overview
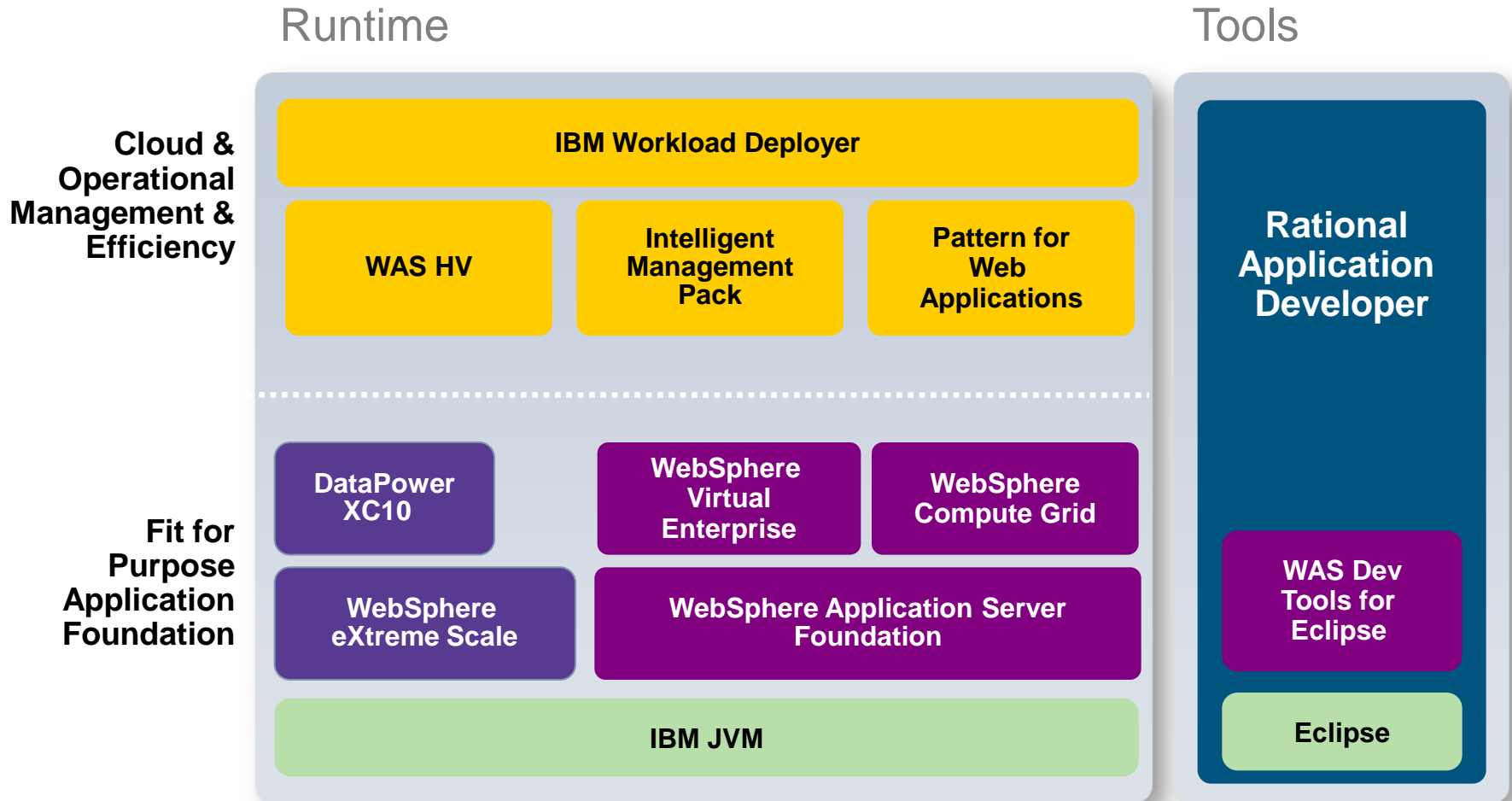
Disclaimers

- IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

- The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

- **Performance**. Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including 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 results similar to those stated here.

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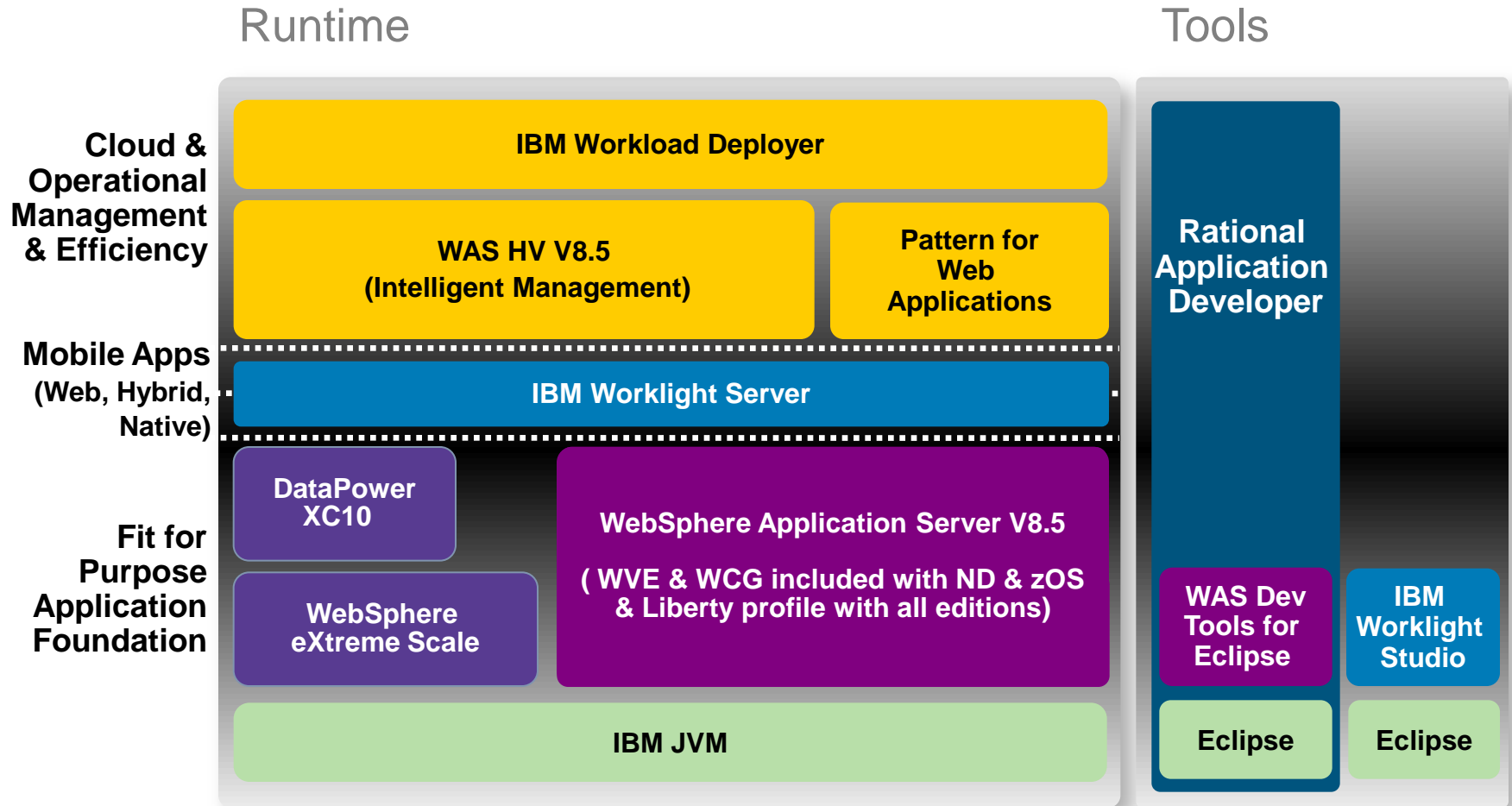*Over 14 years of Leadership & Trusted Delivery*

JEE6

WAS V8.5
Lightweight Liberty profile
Intelligent Management
Java SE 7

WAS V8 Beta, GA
Web 2.0 & Mobile FEP
WAS v8.5 Alpha, Beta
Migration Toolkit Refresh
WAS Tooling Bundles

JEE5

WAS v7
WAS V7 & V6.1
Feature Packs

WAS V6.1
Feature
Packs

J2EE1.4

WAS V8.5 Beta 2
(including
lightweight
Liberty profile)

WAS V6.1

WAS V6

WAS V7 Feature
Packs
(XML, CEA, SCA)
SAML & WOLA
WAS HV
WAS EC2 AMI

WAS V8 Alpha, Beta
WAS V7 Feature Packs
(OSGi, JPA 2.0, Modern
Batch,CEA, Dynamic
Scripting)
WAS HV Refresh
Migration Toolkit Refresh

2005
2006
2007
2008
2009
2010
2011
2012

3

WebSphere Application Infrastructure V8.5
What's Changed

Runtime

Tools

Cloud & Operational Management & Efficiency

IBM Workload Deployer

WAS HV V8.5 (Intelligent Management)	Pattern for Web Applications

Mobile Apps (Web, Hybrid, Native)

IBM Worklight Server

Fit for Purpose Application Foundation

DataPower XC10

WebSphere eXtreme Scale

WebSphere Application Server V8.5

(WVE & WCG included with ND & zOS & Liberty profile with all editions)

IBM JVM

Rational Application Developer

WAS Dev Tools for Eclipse	IBM Worklight Studio

Eclipse	Eclipse

WAS Hypervisor Edition (WAS HV)

- WAS shipped ready to run on a hypervisor based on OVF standard

- No installation required (just run and choose a profile)

- Single virtual image capable of supporting single servers or clusters

- WAS v6.1, v7, v8, v8.5 available with full support for FePs

- New images released on quarterly update intervals

- Maintenance, support, and fixes through IBM for both WAS and Operating System

- Self optimizing & autonomic clouds via newly announced Intelligent Management Pack, an optional add-on to WAS HV & that IBM Workload Deployer can leverage

WAS V6.1 HV

WAS V7 HV

WAS V8 HV

WAS V8.5 HV

WebSphere Application Server

Operating System
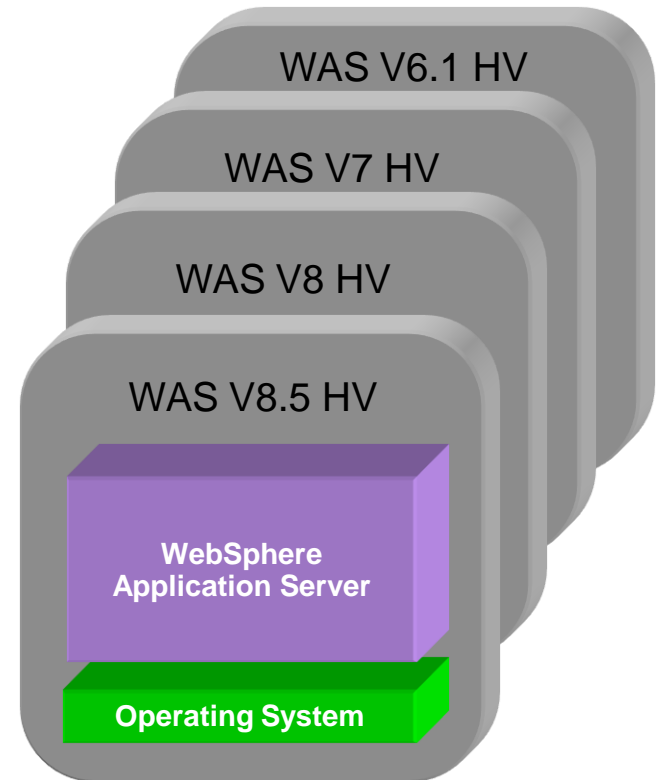
Speed the Development & Test Lifecycle Through Self Service Access to Repeatable Environments
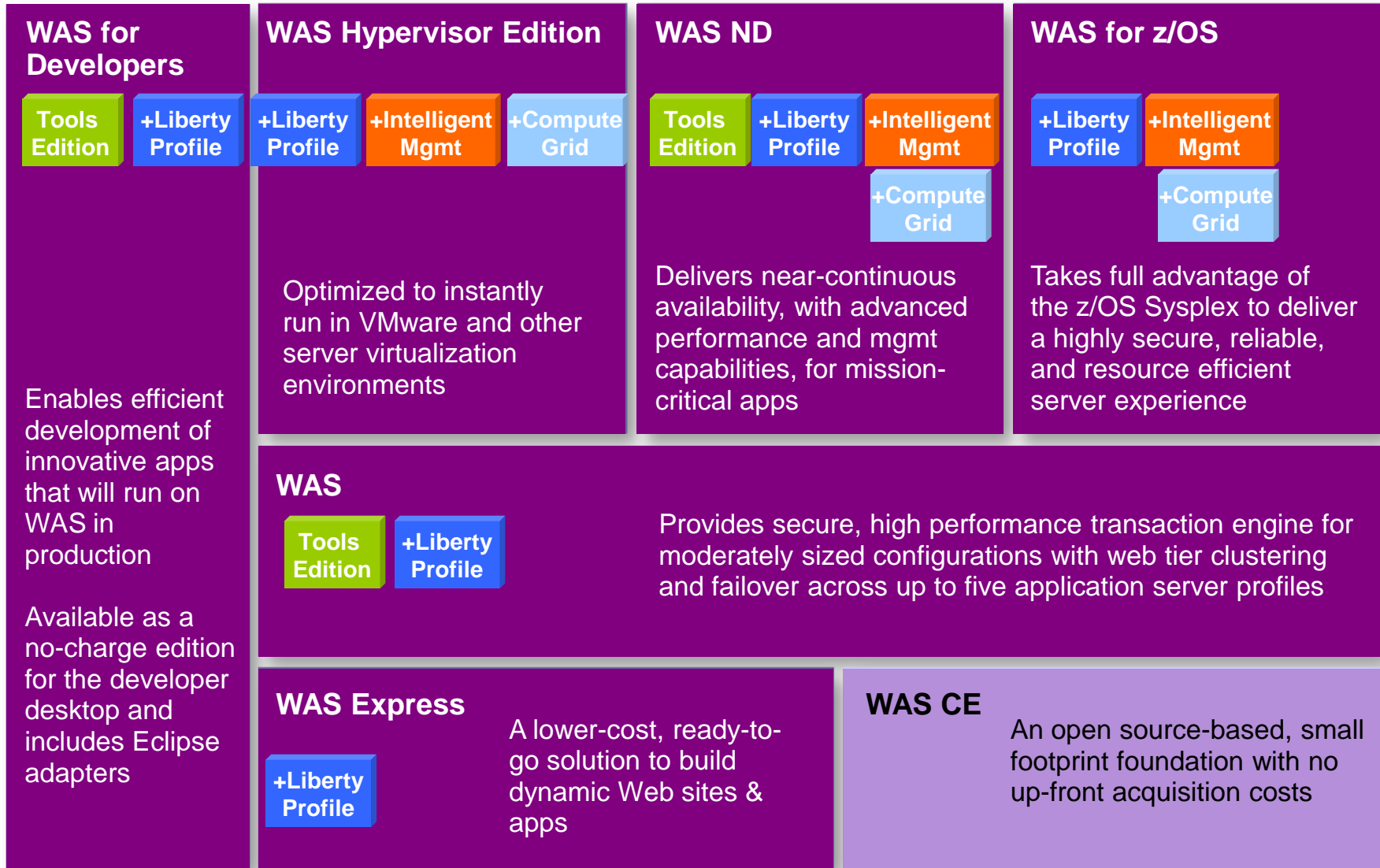
IBM Workload Deployer & WAS Hypervisor Edition

1 Self service request

Developer

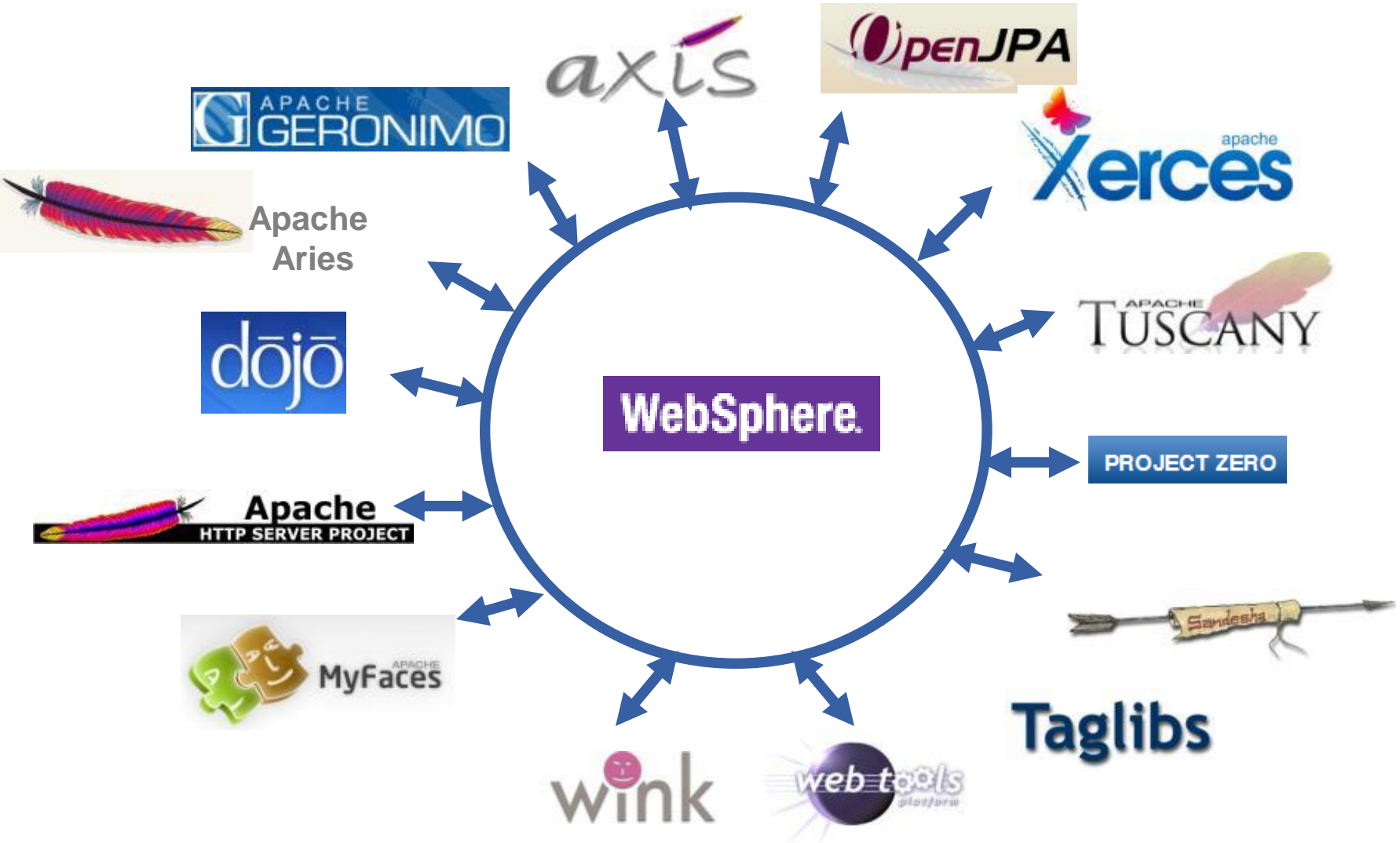**2** Rapidly access consistent & repeatable provisioned development & test environment
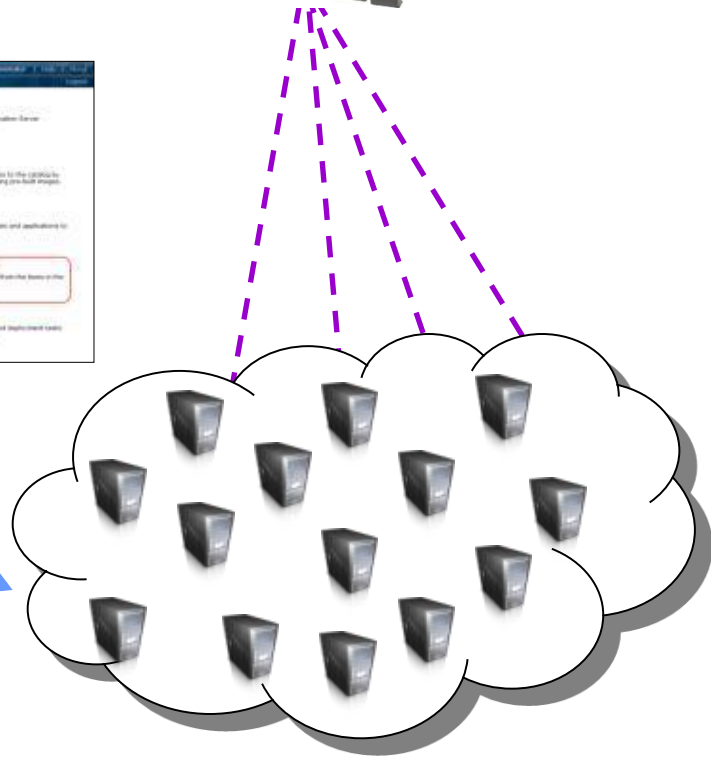
# WAS V8.5 With the Liberty Profile, Intelligent Management & CG

WAS for Developers

[Tools Edition] [+Liberty Profile]

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

[+Liberty Profile] [+Intelligent Mgmt] [+Compute Grid]

Optimized to instantly run in VMware and other server virtualization environments

WAS ND

[Tools Edition] [+Liberty Profile] [+Intelligent Mgmt] [+Compute Grid]

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

WAS for z/OS

[+Liberty Profile] [+Intelligent Mgmt] [+Compute Grid]

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

WAS

[Tools Edition] [+Liberty Profile]

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

[+Liberty Profile]

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

[ ] Built on a common code base

Enabling Developers to Start With Open Source/Community Software & Benefit from IBM Value Add in Production

Summary of System z and z/OS exploitation

	Passive Exploitation	Active Exploitation
Availability and Reliability	The System z hardware and the key operating systems (z/OS and z/VM with Linux) are mature and proven.	The active exploitation elements take direct advantage of Parallel Sysplex and the shared data clustering capabilities of the technology. Properly configured, no other platform offers the same degree of availability and reliability.
Manageability	The platform has a rich set of systems management tools that help maintain the platform and keep it running.	The active exploitation of system components such as WLM, SAF, RRS and SMF allow for management using proven and mature technologies.
Flexibility	For z/OS the shared-resource design is mature and allows for co-location with key isolation attributes. z/VM, Linux and zBX provides extensive virtualization capabilities.	The active exploitation of z/OS interfaces such as MODIFY provide dynamic operations at the OS level against the operating WAS servers.
Affordability	The value proposition of System z and z/OS is centered around efficient sharing of resources.	One element of contribution to the expense question is the zAAP specialty engine for Java offload.

Exploitation of zAAP Specialty Engines

zAAP engines are Java offload engines. They enhance the financial picture of the z/OS platform, and they free up GP for other key subsystem processing
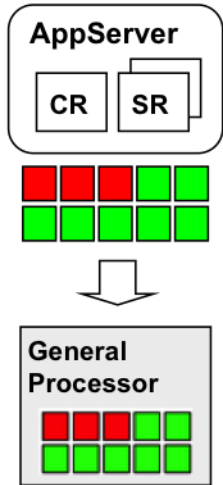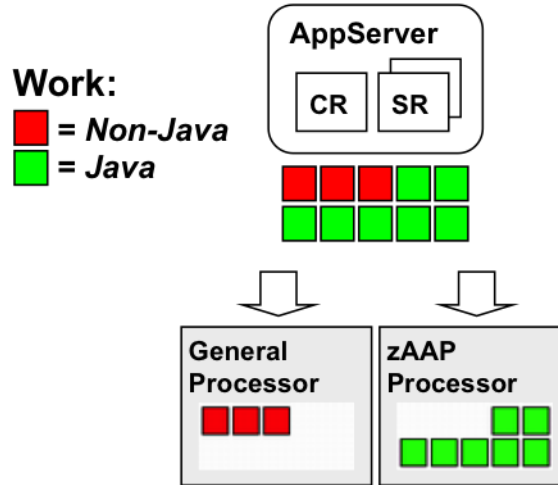

Keys to understanding value of zAAPs:
- **zAAP processors have a considerably lower acquisition cost compared to GPs**
- **Offloading Java to zAAP frequently allows growing non-Java work to live within existing GPs, thus avoiding capital acquisition**
- **Monthly license charges based on capacity of the system can be influenced by the presence of zAAPs, which do not count towards charges**

There are many technical details left unsaid here with respect to how they're configured, the rules for dispatching, when Java might go to GP, etc. Objective here was key points, not details.

This is really a function of the Java SDK and the dispatcher of z/OS. The zAAP-enabled Java SDK is packaged with WAS z/OS, so WAS automatically takes advantage of zAAPs if they're present and configured

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:**
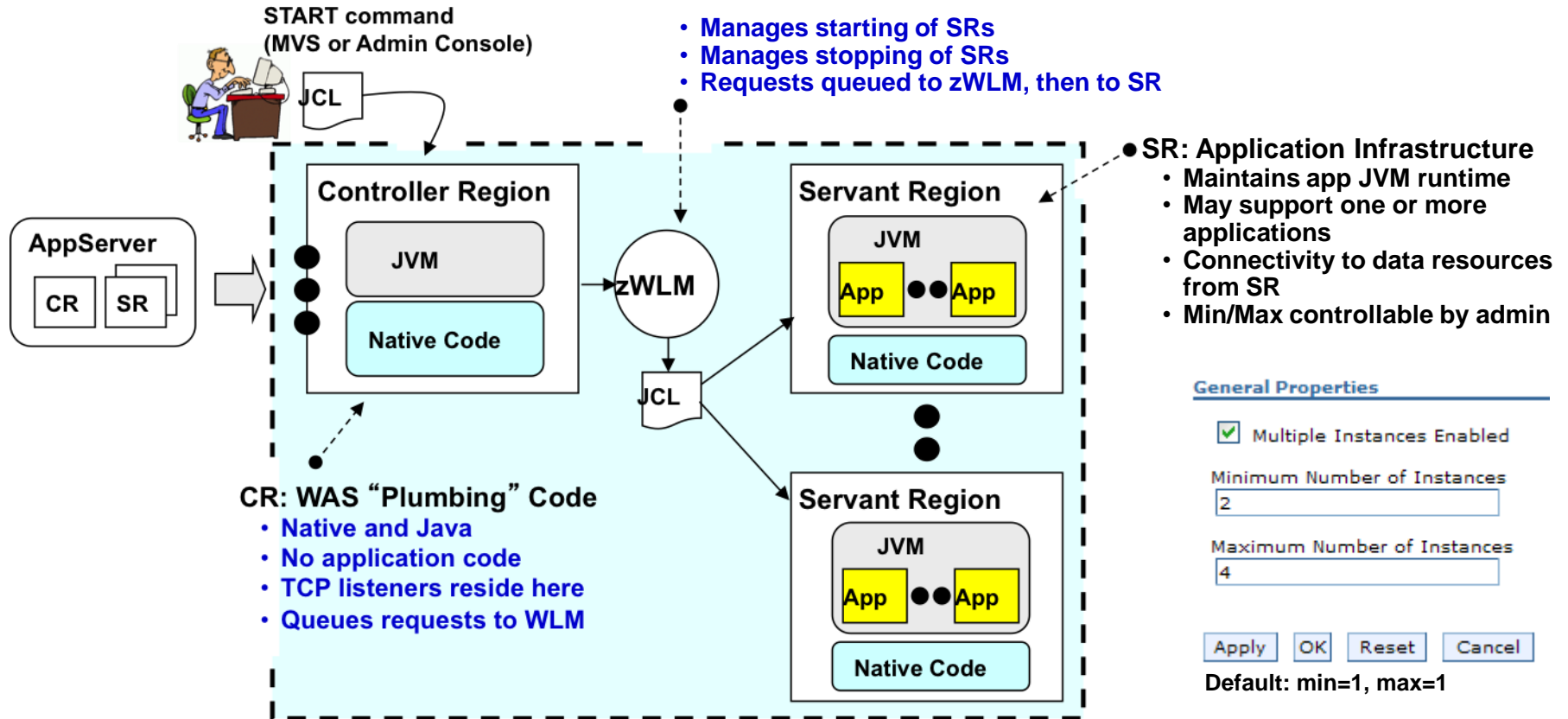
Let's now explore how this is accomplished …

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

© 2012 IBM Corporation

The Big Picture of WAS z/OS and Parallel Sysplex HA

It's all about redundancy *and* integration with platform HA function

6 DVIPA + Sysplex Distributor

VIPA   **VIPA**

2   **5**

LPAR A OSA ●● OSA   **LPAR B** OSA ●● OSA

Daemon
CR

DMGR
CR | SR

Daemon
CR

Mobile DMGR

Node Agent
CR

Node Agent
CR

1

AppServer
CR | SR   *Cluster*   Appl   **3**   Appl   AppServer
CR | SR

CICS | IMS | MQ | DB2   CICS | IMS | MQ | DB2

4 RRS - Res. Recovery Svcs.   RRS - Res. Recovery Svcs.

SAF, WLM, IRD, JES, etc.   SAF, WLM, IRD, JES, etc.

CF   **Shared Data**
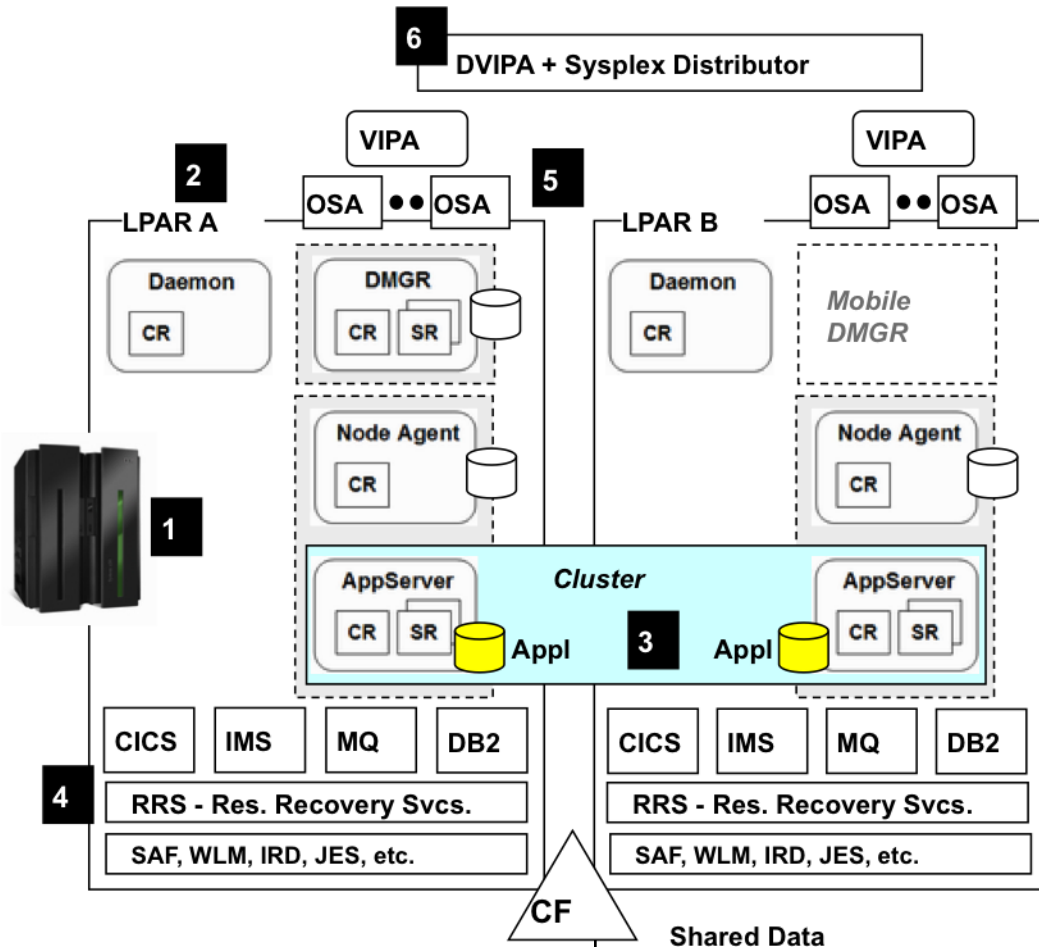
1. **Redundant and fault-tolerant hardware**
   System z hardware design has many layers of fault tolerance and redundancy.
2. **Redundant z/OS instances**
   Either through logical partitioning (LPAR) or separate physical machines.
3. **Clustered WebSphere z/OS servers**
   Multiple application servers grouped into a logical unit for application deployment and management
   z/OS exclusive: dynamic SR expansion (more coming up)
4. **Redundant data resource managers with Sysplex shared data**
   Multiple resource managers instances with shared data in CF and a global syncpoint manager (RRS)
5. **Redundant network adapters hidden behind Virtual IP address**
   On the front end, multiple network interfaces with a moveable virtual IP address protecting against outage
6. **Workload distribution hidden behind distributed virtual IP and Sysplex Distributor**
   Further abstraction of real IP addresses behind a virtual IP that can be swapped across images in a Sysplex, with Sysplex Distributor providing TCP connection distribution based on WLM

We show two operating system instances. That can be higher for greater availability and more manageable failover

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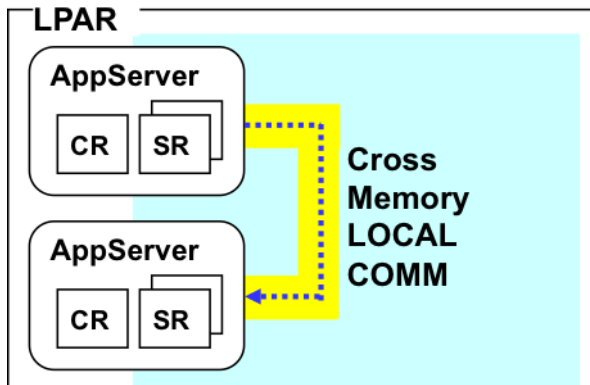
Data Access
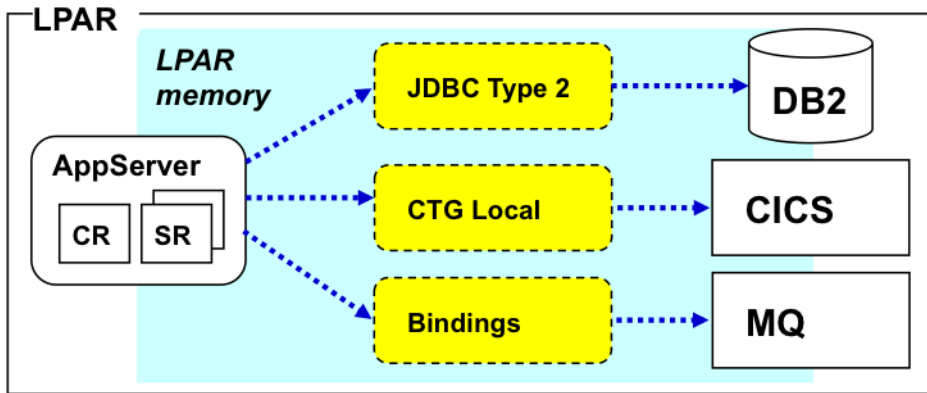
Benefits:
- **Cross memory speed**
- **Security ID propagation (no alias)**
- **Exploitation of RRS**
- **Avoid serialization of parameters**
- **Avoids SSL overhead**
- **Single thread of execution**

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)

IBM

New in V7.0.0.4, this new function allows external address spaces to participate in a cell's Local Comm communications.
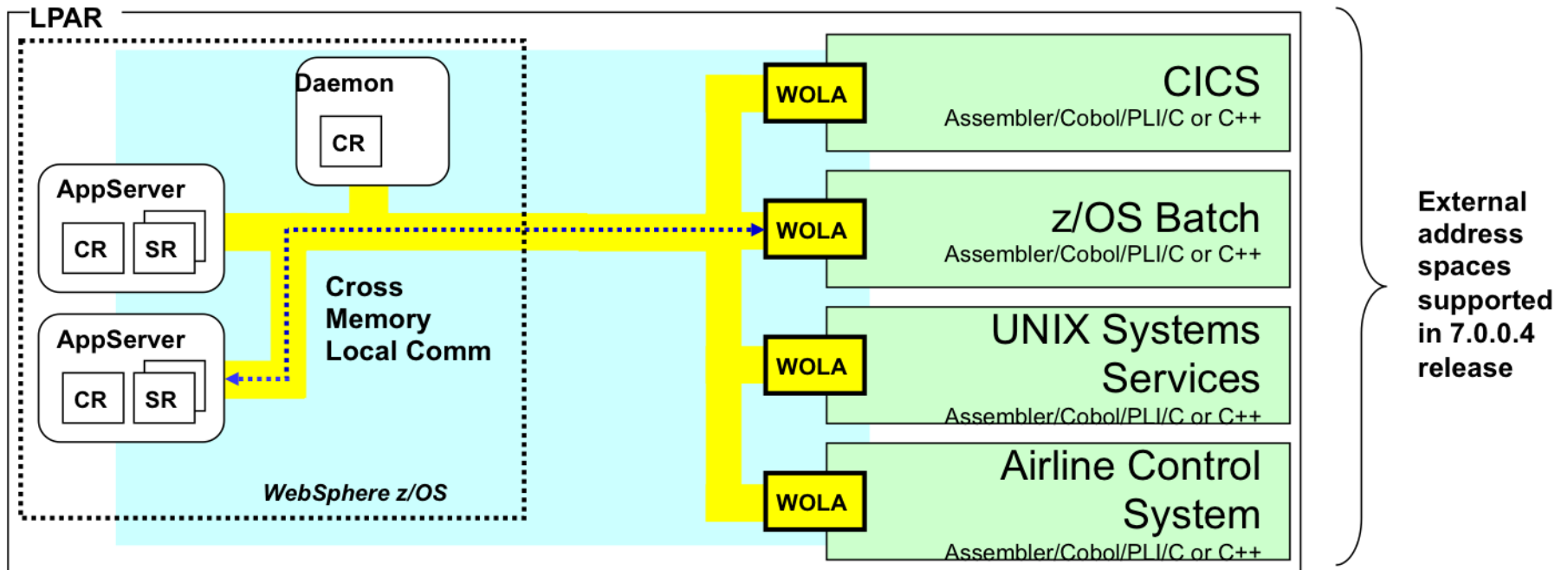
Benefits:
- **Based on Local Comm (z/OS exclusive)**
- **Bi-directional ... WAS outbound or inbound to WAS (WOLA exclusive)**
- **CICS Security and Transaction propagation (some restrictions apply)**
- **Faster than other local solutions**

WP101490 on
`ibm.com/support/techdocs`
for more

Unparalleled Application Development and Management Environment, Rich User Experiences…Faster

Developer Experience

Fast, flexible, and simplified application development

- Java 6 EE
- Liberty Profile
- Expanded Tooling and WAS Tooling Bundles
- Web 2.0 & Mobile Toolkit; IBM Worklight Integration
- JDK7 Support
- Migration toolkit
- OSGI programming model enhancements
- EJB support in OSGi apps
- SCA OASIS programming model

Application Resiliency

Intelligent Management & Enhanced Resiliency

- Improved Performance
- Application Edition Management
- Application Server Health Management
- Dynamic Clustering
- New Intelligent Routing capabilities
- Installation and Maintenance
- Messaging infrastructure resiliency
- Memory leak detection & protection in WAS

Operations and Control

Improved Operations, Security, Control & Integration

- Selectable JDK
- WebSphere Batch enhancements
- Configuration Change Tracking
- OSGi Blueprint security improvements
- Cross Component Trace (XCT)
- Enhanced IBM Support Assistant
- Better log and trace filtering

WAS V8.5 - "Liberty" Profile

- Not a single static profile: rather a dynamic, flexible profile of the runtime to load only what the application needs
  - Memory footprint (web feature): < 60 MB
  - Profile is dynamic - switch parts of the server on & off w/out restart

- Extremely lightweight
  - Incredibly fast (re)start times: <5 seconds

- Simplified configuration for quick time to productivity; one single config file or modular config (as desired)
  - Easy to share / diff / manage in version control
  - Easy to componentize config across larger development teams

- Easy access
  - smaller download, unzip install, eclipse plugin, Mac OS support, jdk flexibility

- Tools available as Eclipse features as well as in RAD…

Composability – Based on *features*

```
<server description="composabilityIsKey">

    <featureManager>
        <feature>appSecurity-1.0</feature>
        <feature>jsp-2.2</feature>
        <feature>restConnector-1.0</feature>
        <feature>jpa-2.0</feature>
    </featureManager>

</server>
```
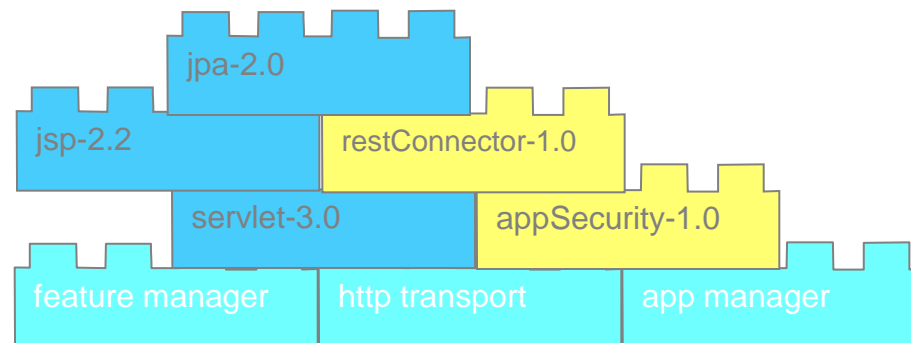
What is the WAS for z/OS Liberty profile?

- The WAS for z/OS Liberty profile is Liberty with *optional*, independently enabled *extensions* that exploit z/OS facilities
  - Only enable exploitation of z/OS features you need
  - Only configure the z/OS functions you use

- Focus of v8.5 is basic integration and exploitation

zosTransaction-1.0

zosWlm-1.0 zosSecurity-1.0

appSecurity-1.0

localConnector-1.0 restConnector-1.0

ssl-1.0

transaction-1.1 monitor-1.0

sessionDatabase-1.0

jndi-1.0 jdbc-4.0
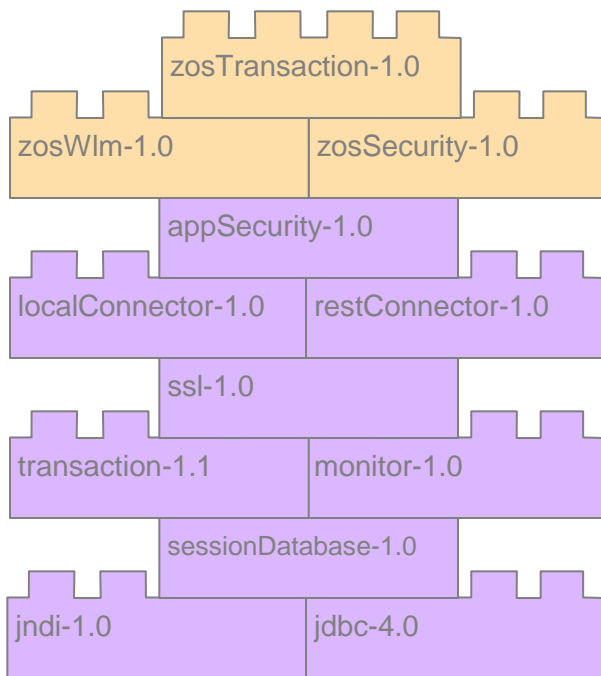
z/OS Feature Sets

Common Feature Sets

Under The Hood
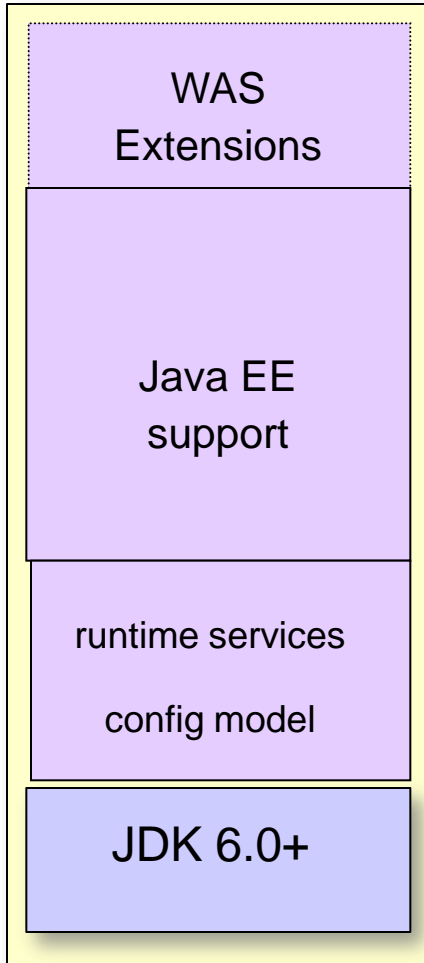
Traditional WAS profile

- WAS Extensions
- Java EE support
- runtime services
- config model
- JDK 6.0+
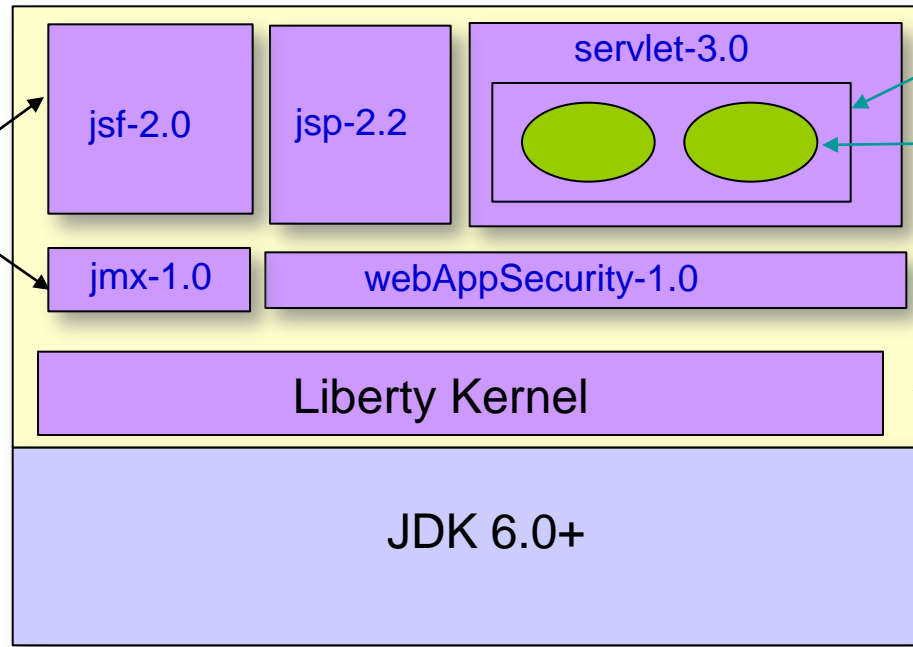
WebSphere Liberty profile

- jsf-2.0
- jsp-2.2
- servlet-3.0
- jmx-1.0
- webAppSecurity-1.0
- Liberty Kernel
- JDK 6.0+

WAS Container

Applications

OSGi framework (runtime)

Lightweight configuration

```xml
<server description="tradeLiteServer">
    <featureManager>
        <feature>jsp-2.2</feature>
        <feature>jdbc-4.0</feature>
    </featureManager>


    <logging consoleLogLevel="INFO" />



    <application type="war"
                 id="tradelite"
                 name="tradelite"
                 location="${shared.app.dir}/webcontainer/tradelite.war" />



    <include location="jdbc-drivers.xml" />
    <include location="${user.home}/custom.xml" optional="true" />



    <dataSource id="jdbc/DerbyTradeDataSource"
                jndiName="jdbc/TradeDataSource"
                jdbcDriverRef="DerbyEmbedded">
        <properties databaseName="${shared.resource.dir}/data/tradedb" />
    </dataSource>
</server>
```

Features control what's available in the runtime

Singleton configurations specify properties for runtime services for which there is only one instance

Instance configurations allow multiple instances of resources and applications to be declared

Includes can be used to implement an extensible configuration model

References can be used in multiple elements to point share a common definition

Liberty and traditional profile capabilities

There are functional differences between traditional WAS and the Liberty profile – Liberty provides a useful subset of traditional WAS

Liberty Profile

Bean validation
Blueprint
Java API for RESTful Web Services
Java Database Connectivity (JDBC)
Java Naming and Directory Interface (JNDI)
Java Persistence API (JPA)
Java Server Faces (JSF)
Java Server Pages (JSP)
JMX
Monitoring
OSGi JPA
Remote connector
Secure Sockets Layer (SSL)
Security
Servlet
Session Persistence
Transaction
Web application bundle (WAB)
z/OS Security (SAF)
z/OS Transactions (RRS)
z/OS Workload Management

Traditional WAS Profile

Everything Liberty has…

➕

Enterprise Java Beans (EJBs)
Messaging (JMS)
Web Services
Service Component Arch (SCA)
Java Connector Architecture (JCA)
Clustering
WebSphere Optimized Local Adapters
Administrative Console
WSADMIN scripting
Multi-JVM Server Model

And much more …

z/OS Operations – Choose your interface

- Run from a shell or as a started task with the provided launchers and PROCs

- Important messages routed as WTOs for automation

- Modify commands enable changes to trace specification or to request a diagnostic dump


```
                                      xa1
   - SY1  start bbgzangl
   - SY1  IRR812I PROFILE BBGZANGL.* (G) IN THE STARTED CLASS WAS USED
   -         TO START BBGZANGL WITH JOBNAME BBGZANGL.
     SY1  $HASP100 BBGZANGL ON STCINRDR
   - SY1  $HASP373 BBGZANGL STARTED
     SY1  CWWKB0056I INITIALIZATION COMPLETE FOR ANGEL
   - SY1  start bbgzsrv,parms='tradeLiteServer'
   - SY1  IRR812I PROFILE BBGZSRV.* (G) IN THE STARTED CLASS WAS USED
   -         TO START BBGZSRV WITH JOBNAME BBGZSRV.
     SY1  $HASP100 BBGZSRV  ON STCINRDR
   - SY1  $HASP373 BBGZSRV  STARTED
     SY1  +CWWKF0011I: The server tradeLiteServer is ready to run a smarter
      planet.
```

```
  sykesm@sykesm-MacBook:~/work/liberty-workspace/com.ibm.ws.security.authori...
   sykesm@sykesm.../wlp/lib/features ...    sykesm@sykesm...saf/build/classes
$ wlp/bin/server start tradeLiteServer
Server tradeLiteServer started with process ID 65682.
$ cat wlp/usr/servers/tradeLiteServer/logs/console.log
Launching tradeLiteServer (wlp-1.0.0.201203311104/websphere-kernel_1.0.0) on IBM J9 VM, vers
ion pmz6460_26sr1fp1-20120201_02 (SR1 FP1) (en_US)
[AUDIT   ] CWWKE0001I: The server tradeLiteServer has been launched.
Listening on port localhost/127.0.0.1:5678 ...
[AUDIT   ] J2CA8004I: The dataSource jdbc/TradeDataSource is available as jdbc/TradeDataSour
ce.
[AUDIT   ] J2CA8000I: The jdbcDriver DerbyEmbedded is available.
[AUDIT   ] CWWKZ0058I: Monitoring dropins for applications.
[AUDIT   ] CWWKT0016I: Web application available (default_host): http://flash226.pok.ibm.com
:9080/snoop/*
[AUDIT   ] CWWKZ0001I: Application snoop started in 0.12 seconds.
[AUDIT   ] CWWKT0016I: Web application available (default_host): http://flash226.pok.ibm.com
:9080/tradelite/*
[AUDIT   ] CWWKZ0001I: Application tradelite started in 0.59 seconds.
[AUDIT   ] CWWKF0011I: The server tradeLiteServer is ready to run a smarter planet.
$ 
```

```
IEE612I CN=C3E0SY1   DEVNUM=03E0 SYS=SY1

_

IEE163I MODE= RD
```

Wed 04 Apr 09:43

23

Why Liberty on z/OS?

Simplification

- Liberty environments don't need significant z/OS configuration and customization
    - RRS, WLM, and SAF exploitation and configuration is optional
    - No authorized code is ***required*** to host applications

- Liberty runs in a single process instead of 3+ started tasks
    - Significantly reduced resource consumption
    - No started task definitions are ***required***
    - No need to create users and groups for controllers, servants

- Server instances can be quickly created or cloned
    - `server` `create` *serverName* `[options]`
    - `server` `package` *serverName* `[options]`

Application portability and stack consistency

- Liberty behaves *exactly the same* on all platforms out of the box
  - z/OS specific behaviour must configured if desired

- Administration is the same for all platforms out of the box
  - Server operations are controlled by the same `server` script
  - Logs, trace, and configuration live in the hierarchical file system and are tagged with the appropriate code page for easy viewing and editing
  - Existing server configurations can be brought to z/OS from distributed without modification

- An extremely light-weight, single process runtime
  - Removes deployment and runtime complications introduced by the split process, multi-JVM runtime of traditional WAS for z/OS

Elastic Cache and Liberty

- **WebSphere eXtreme Scale and WebSphere DataPower XC10**

  - Integrates seamlessly with Liberty Runtime and its dynamic feature model

  - WXS Container servers can now run on Liberty profile

  - Standalone Liberty Servers can maintain HTTP Session failover and high availability by leveraging an WXS Grid

  - Simple for customer to make use of WXS Use cases!

- **For Developers, easy to develop WXS applications using Liberty within Eclipse**

  - Start up a WXS grid, start up a WXS client, & start up the Liberty server all within a single runtime environment!

- **Additional Tooling for WXS supported for Liberty**

  - Very easy for customers to develop & configure WXS applications right in the Eclipse tool!

Elastic Cache

1 DataPower XC10 for simple data oriented scenarios

2 WebSphere. software

eXtreme Scale for maximum flexibility

eclipse

Java EE 6

Simplify standards-based enterprise Java development for dept. to core business apps

Enhanced developer productivity, user experiences, performance & integration:

- **Enterprise JavaBeans (EJB) 3.1:** Enhanced developer productivity through simplification including testing outside of the application server, new timer support & asynch enhancements

- **Contexts and Dependency Injection for Java (CDI) 1.0:** Faster time to value through tighter and simpler integration between Web & business logic tiers

- **Java Persistence API (JPA) 2.0:** Enhanced developer ease of use & app performance through improved locking, mapping support & dynamic query construction

- **Java Servlet 3.0:** Enhanced time to value through annotations and ease of integrating third party presentation frameworks

- **Java API for RESTful Web Services (JAX-RS) 1.1:** Deliver better user experiences faster through integrated Web 2.0 programming model support

- **JavaServer Faces (JSF) 2.0:** Enhanced developer productivity & end user experience through annotations & Facelets support

- **Bean Validation 1.0:** Improved developer productivity through declarative means for describing validation constraints for data

- **Java Architecture for XML Binding (JAXB) 2.2:** Improved performance via new default marshalling optimizations

- **Enterprise Web Services 1.3:** Improved integration and reuse support

- **Java API for XML-Based Web Services (JAX-WS) 2.2**: Developer productivity and security enhancements

New In V8.0 !

IBM

Java EE 6 Highlights: A Deeper Look 1 of 4

- **WebSphere Application Server V8**
  - Provides high performance, reliable and scalable implementations of Java EE 6 specifications
  - Integration value add, such as Dynacache Servlet caching support for Servlet 3.0, JPA L2 cache performance and security integration

- **Enterprise JavaBeans (EJB) 3.1:**
  - Embeddable EJB container to unit test EJBs outside of the application server
  - Simple beans with no interfaces
  - Simpler packaging via EJBs packaged in WAR files
  - New asynch-method invocations for handling long running requests
  - Improvements to timers for calendar based events, non-persistent timers and automatically created timers
  - Singleton EJBs for easier creation and management of caches
- **Contexts and Dependency Injection for Java (CDI) 1.0:**
  - Better integration between Web (JSF) & business logic (EJB) tiers
  - Declare an EJB that should be used within the context of an HTTP session and used in a scriptlet with a few lines of annotations, removing much logic for handling the integration of web applications and EJB business logic.

© 2012 IBM Corporation

CDI Example
Integration and automated lifecycle management

JSF Engine

EJB Container

WAR file

JSF Page

Faces-config.xml

Account EJB

Web app logic in EJB

EL Resolver

Get CDI Request scoped bean

Resolve injections

Session bean create

Resolve injections

CDI Service

Annotation Scanning Service
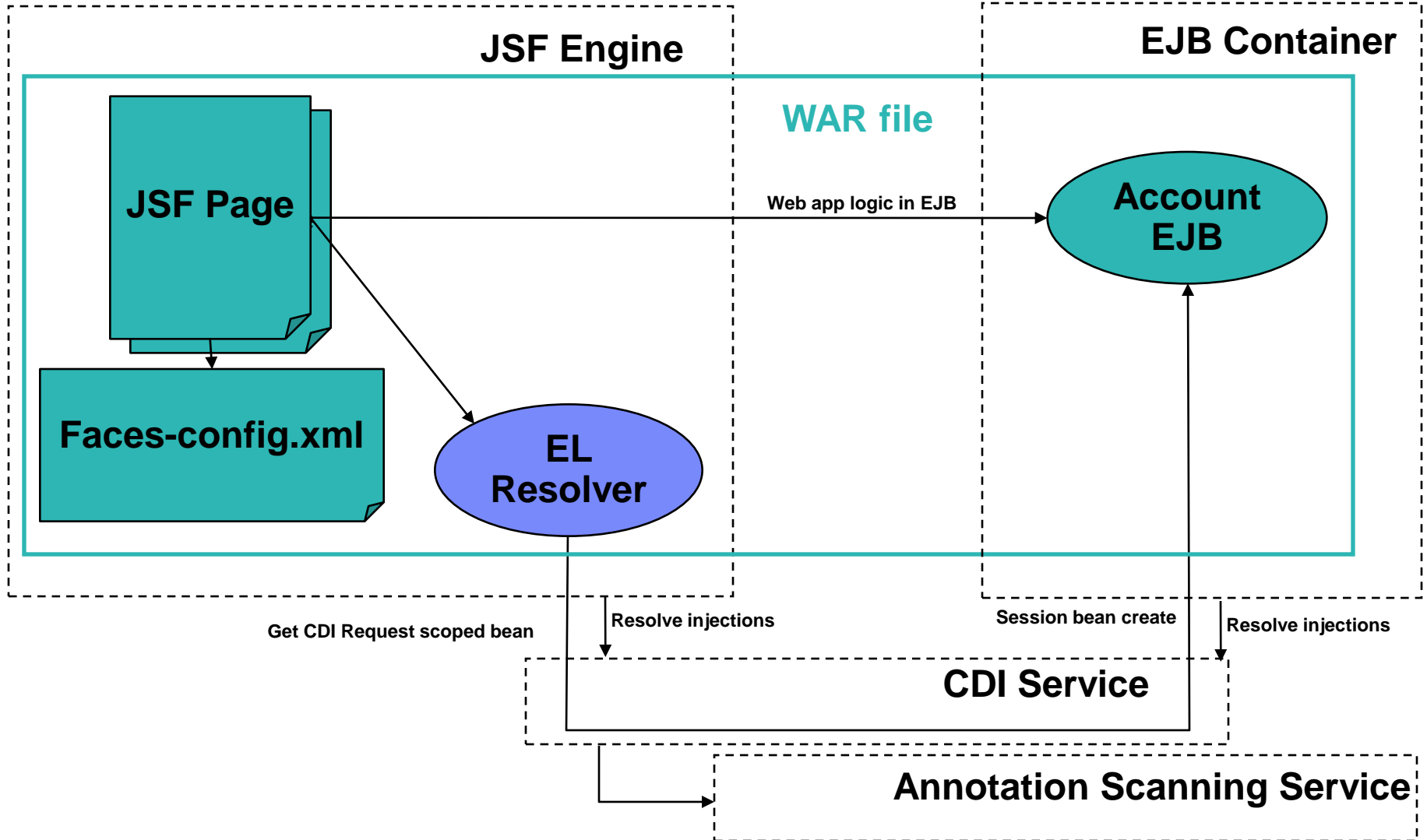
Java EE 6 Highlights: A Deeper Look 2 of 4

- **JavaServer Faces (JSF) 2.0:**
  - Extensive use of annotations for improved developer productivity.
  - Facelets support for tighter integrated page description format that improves performance and UI composition capabilities
  - Support for AJAX life cycles for better UI interaction, UI look and feel customization through skins

- **Java Servlet 3.0:**
  - Extensive use of annotations for improved developer productivity
  - Simpler & faster to integrate third party presentation frameworks through automatic metadata discovery and integration provided by the web fragment support
  - New asynchronous protocol support for SIP and COMET

Java EE 6 Highlights: A Deeper Look 3 of 4

- **Java Persistence API (JPA) 2.0:**
    - Improved mapping support to handle embedded collections and ordered lists
    - Standardized query hints
    - Pessimistic locking is now standardized to support write-mostly application performance
    - Standardized level 2 cache plugin and configuration
    - New dynamic Criteria API for dynamic construction of queries without an in-depth knowledge of SQL
    - Bean Validation integration

- **Bean Validation 1.0:**
    - Declarative means for describing validation constraints for data

        @Pattern(regexp="((\d\d\d\-)?\d\d\d\-\d\d\d\d)?")

        private String phoneNum;

    - Improved developer efficiency by not having to write and maintain validation logic multiple times in multiple places within the application

    - Integration
        - JSF to ensure form data is automatically validated to be correct,
        - JPA to ensure incorrect data is not persisted,
        - JCA to ensure Connectors are correctly configured.

IBM

Bean Validation Scenarios
Declarative constraints

JSF Engine

EJB Container

WAR file

JSF Page

Web app logic in EJB

Account EJB

Backing bean

Faces-config.xml

EL Resolver

Account Entity

Validate form data

Get CDI Request scoped bean

Session bean create

Validate and persist

CDI Service

JPA

Bean Validation Service

32

© 2012 IBM Corporation
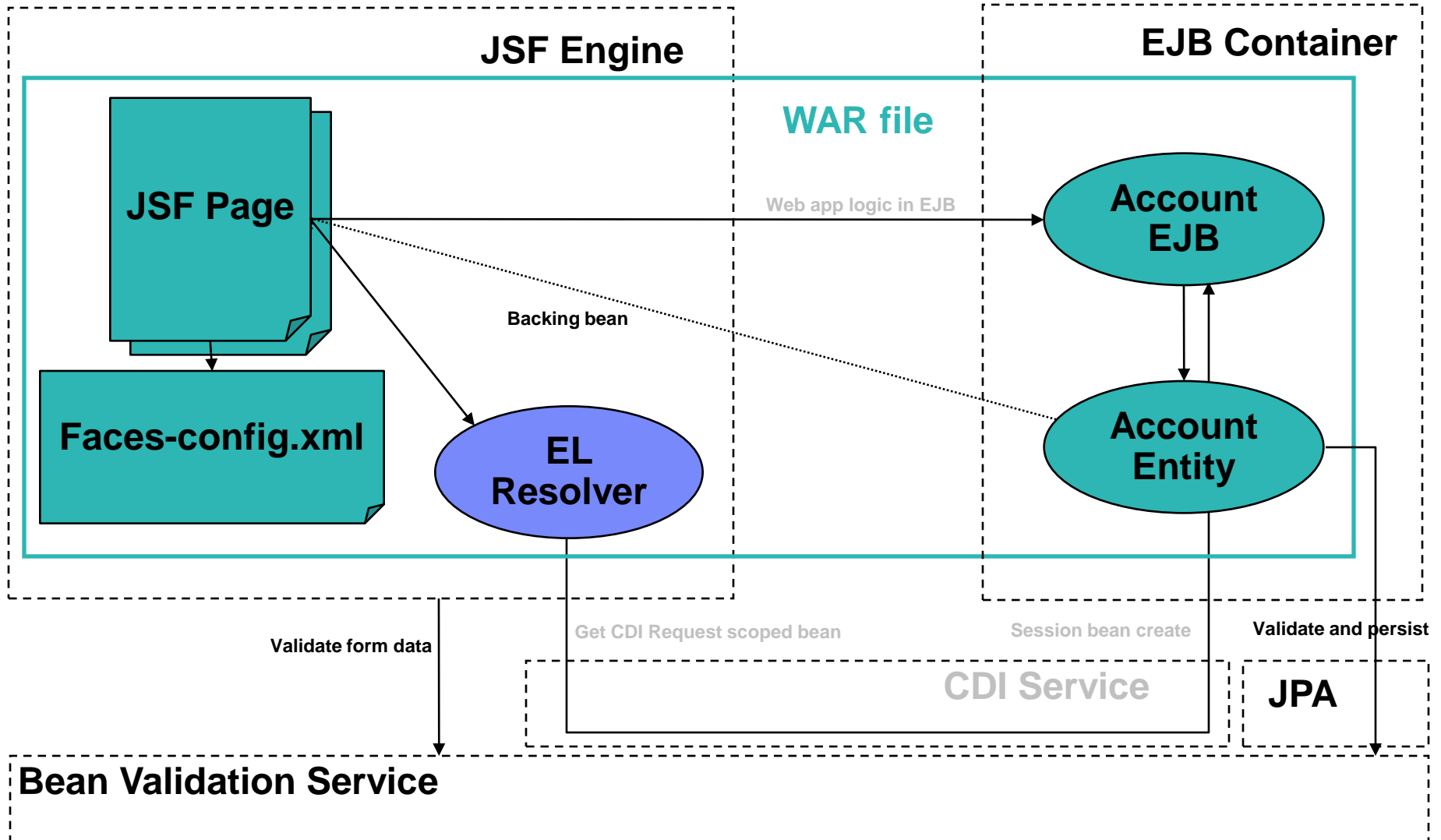
Java EE 6 Highlights: A Deeper Look 4 of 4

- **Java API for RESTful Web Services (JAX-RS) 1.1:**
  - Web 2.0 programming model support within JEE
  - Allows applications to easily and quickly expose resources to the web in a controlled fashion

- **Java API for XML-Based Web Services (JAX-WS) 2.2:**
  - Improved business agility and reuse through support for specifying policy sets & bindings for a service reference which are different from the policy set attachment for the service
  - Improved productivity through ability to specify message exchange patterns required by a Web service using new annotations
  - Enhanced security through support for fine-grained Transport Level Security configuration for policy acquisition from external registries
  - Faster time to value through enabling and configuring WS-Addressing support on a client or service by adding WS-Policy assertions into the WSDL document
  - Improved flexibility through enhanced support for custom properties

- **Enterprise Web Services 1.3 (JSR-109):**
  - Support for singleton session beans as endpoints
  - Support for CDI in JAX-WS handlers and Endpoints
  - Support for global, application, and module naming contexts

- **Java Architecture for XML Binding (JAXB) 2.2:**
  - Improved performance through marshalling optimizations enabled by default

Mobile Application Development

Worklight takes WAS mobile web applications to the next level

WAS

WAS + Web 2.0 and Mobile Toolkit

Worklight

Enterprise Web Applications

• Java EE programming

• Build, deploy and manage Enterprise applications and services

• Server-side & Client-side development

Mobile Web App development based on standard web technologies:

• Run application in mobile browser

• Based on HTML5, CSS3, JavaScript

• Native look and feel

• Advanced mobile UI components

Feature Pack for WAS v6.1/7/8

Toolkit in WAS v8.5

Application delivery in a variety of forms:

• Hybrid application

• Native

• Install through App Store

• Access to native services

Desktop Web Applications

Mobile Web Applications

Mobile Applications

OSGi Applications

Speed development, increase ease of use and reuse through the modularity, dynamism, and versioning capabilities of OSGi applied to web & enterprise applications

Key Features:

- **Modular deployment and management:** Separate common libraries from application archives; manage them centrally and across many versions, concurrently

- **Standards Based DI Framework:** POJO development model, with a container that manages injection of configuration, and controls activation & deactivation, integrated with the server

- **In-place update:** Update applications modules without restarting the application

- **Java Standards Layering:** Java standards such as transaction, security, & persistence can be mixed into the componentized apps as services

- **SCA Integration:** Components can be decorated as SCA components to provide coarse grain SOA services

webA.jar

WEB-INF/classes/servA.class

WEB-INF/web.xml

META-INF/MANIFEST.MF

Bundle Repository

logging f/w jar

persistence f/w jar

MVC f/w jar
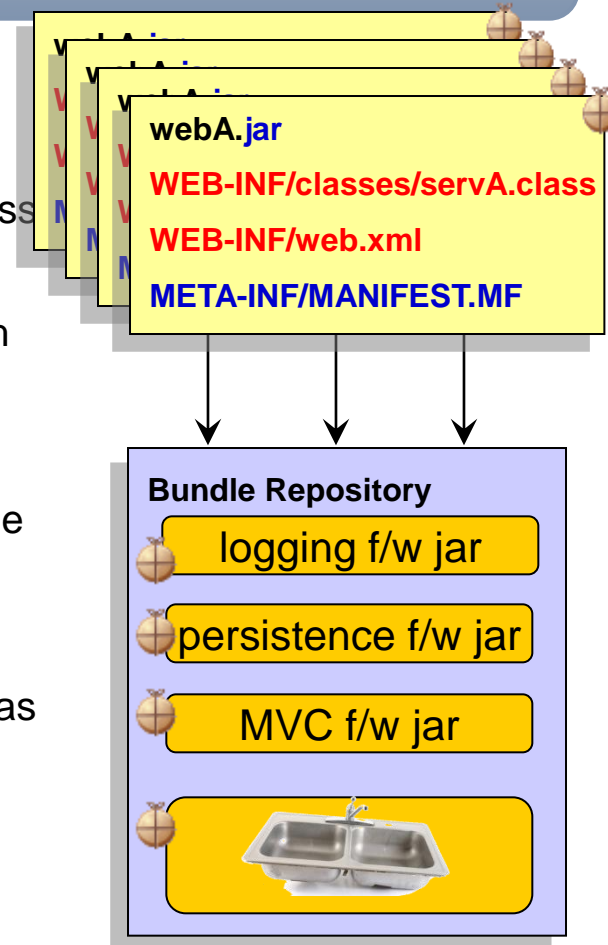
OSGi and SCA: the assembly food chain

SCA Composite

Component

POJO

EAR
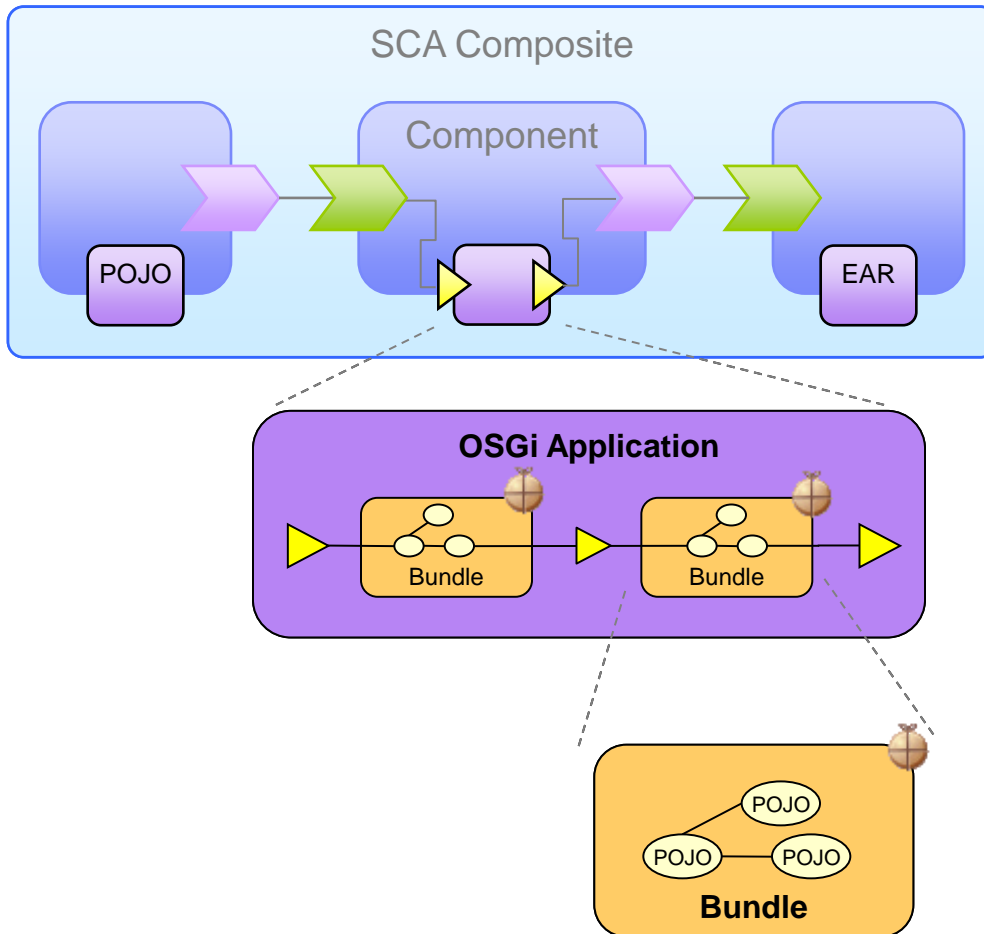
SCA Composite assembled from heterogeneous components including an **OSGi Application** component, and integrated through SCA services with configurable bindings (JMS, web services…).

OSGi Application

Bundle

Bundle

OSGi Bundles assembled in an **OSGi Application** and integrated through services in the OSGi service registry

POJO

POJO   POJO

Bundle

POJOs assembled using a Blueprint context and scoped by an **OSGi Bundle**.

V8.5 OSGi Application Enhancements – EJBs

Include EJB bundles in OSGi applications to simplify the development, deployment and administration of modular enterprise applications

- OSGi bundles can now contain version 3.x EJBs
  - Local & Remote
  - Asynchronous beans, Session beans & Singleton beans

- Stateless Session Beans can be exported as services to the OSGi Service Registry

- Provisioner and Internal Bundle Repository extended to understand EJB Bundles
  - Services provided by EJBs
  - Services required by EJBs

WebSphere Application Server

2. install

EJB Bundle

OSGi App

Admin Console

1. save

3. resolve

EJB Bundle

Internal Bundle Repository
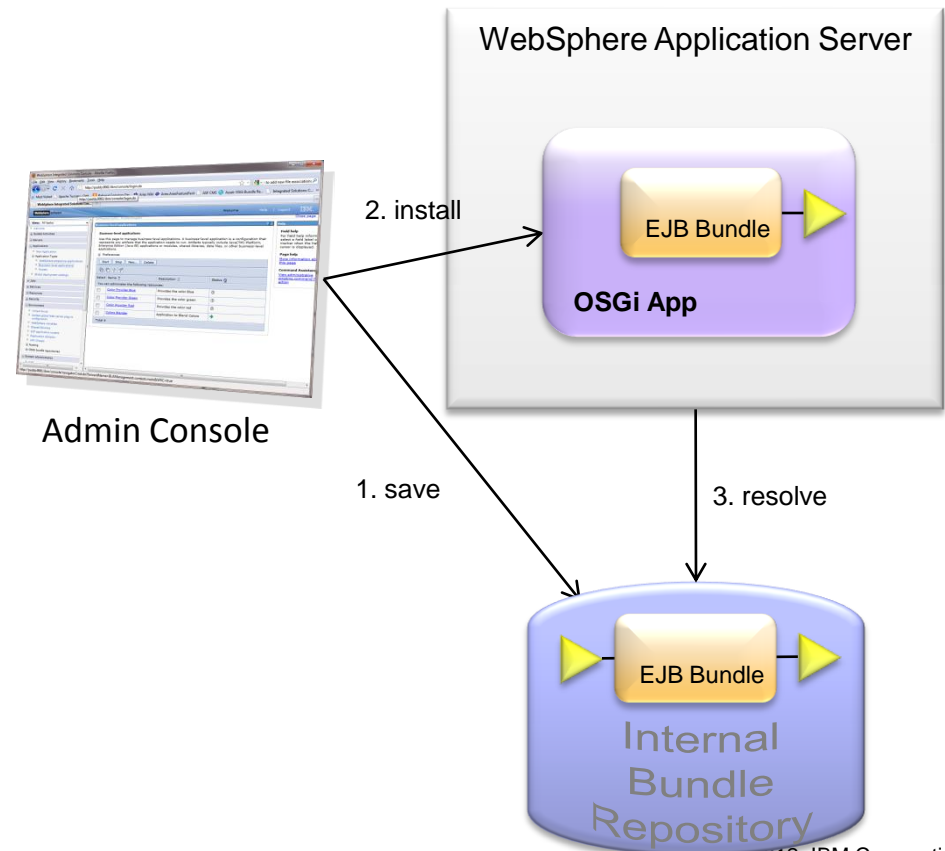
Service Component Architecture (SCA)

Speed SOA application delivery and customization by separating business logic from implementation considerations

Key Features:

- **Compose:** Create SCA service compositions using POJOs, EJB 3.1, 3.0 or 2.1 components, Java Servlets, OSGi bundles, & AJAX/JavaScript

- **Wire Services:** Bindings for Web Services, JMS, SCA and EJB 3.1, 3.0, 2.1 & 2.0

- **Spring Support:** Expose EJB 3.x to Spring components for composition & re-use

- **RIA/Web 2.0 Support:** Expose business logic to Web 2.0 apps via JSON-RPC & ATOM/RSS feeds

- **SCA Domains:** Services interoperability across WAS V8 & V7 over all supported bindings

- **Data Support:** Support for data as Java Architecture for XML Binding (JAXB) or SDO 2.1

- **Simplified Deployment:** Flexible service deployment as a JAR

SCA point of variability: are locations where decisions are implemented that are likely to change and thus should be externalized. Service components with built-in points-of-variability allow users to configure these services to meet different requirements.

http://www.ibm.com/developerworks/library/ar-cbspov1/

WebSphere Application Server

Composite Applications | Business Services

Trans. | QoS | Security

Service Oriented Architecture (SOA)

Existing Enterprise Services

Composite Applications

Web Services (SOAP)

EJB 2.1 3.0, 3.1

WAS 8.0 | WAS 7.0

SCA

Other Platforms
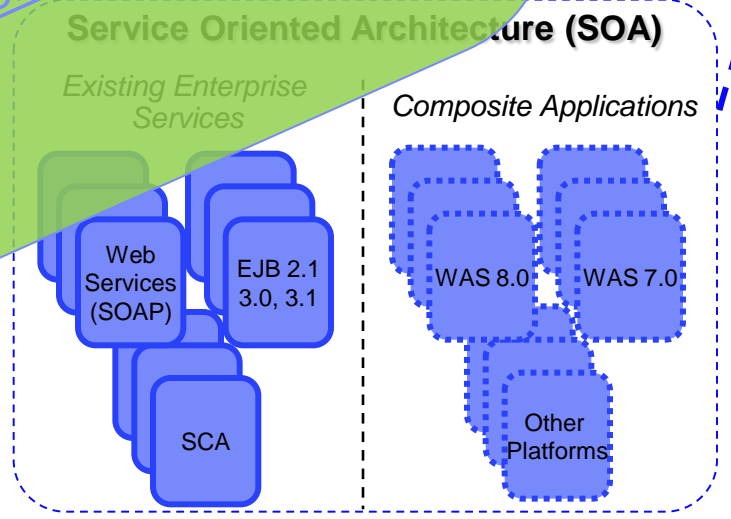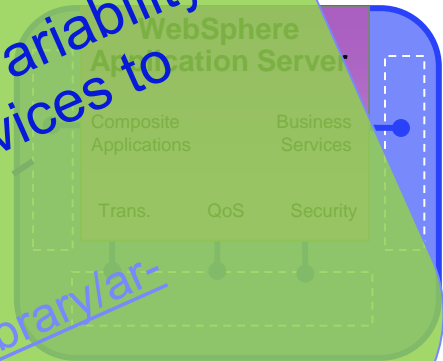
V8.5 Java Batch Enhancements

> Quickly develop and deploy batch applications and dramatically reduce infrastructure and operational costs

- **Parallel Job Manager**
  - Controls parallel job execution including splitting and merging of jobs.

- **Enterprise Scheduler Connectors**
  - Enables integration to external products for scheduling (i.e. Tivoli Workload Scheduler) and monitoring (i.e. ITCAM) batch workloads.

- **Advanced Operations Pack**
  - Provides enhanced operations support, including integration for goal-oriented SLA management, job classes, and usage accounting (including SMF on z/OS).

- **In Addition to V8.0**
  - **Batch Container**
    - Provides the batch execution environment, including services such as checkpoint/restart and job-logging.

  - **Batch Scheduler**
    - Job management control point for determining when/where jobs run.
    - Supports operational commands and provides a visual job console.

  - **Batch Toolkit**
    - Provides tooling for the creating, packaging, and testing batch jobs.

WebSphere Application Server AppServer — Web Container (Web Modules), EJB Container (EJB Modules), Batch Container (Batch Modules), WebSphere Application Server Foundation Services — Security, Transaction, Data Access, Logging ...
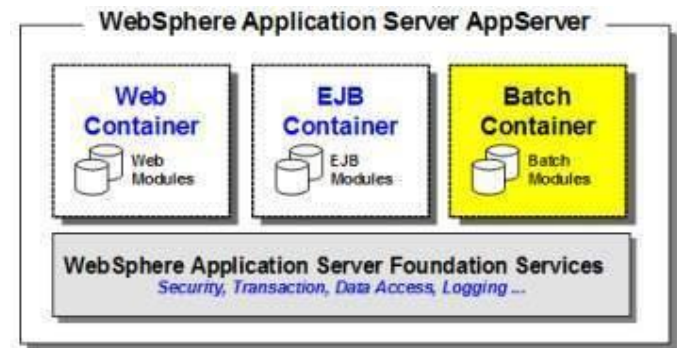
XML

Reuse Java skills & improve ease of use while developing applications to process structured data

Key Features:

- **Speed & Simplicity:** Work with structured data using high performance tools optimized for XML data processing and querying

- **Standards Based:** Support for the XPath 2.0, XSLT 2.0, and XQuery 1.0 W3C standards

- **Consistency:** XML runtime API that offers consistent execution and data navigation API while allowing access to existing Java logic

- **Enterprise grade:** Enterprise class multi-threaded scalability & serviceability with IBM support

- **Samples:** 40+ samples including 4 end to end scenarios

- **Ease of Access & Use:** Integrated with WAS V8

XML Thin Client

WAS Applications | Java Extensions

WAS 8.0

XML API

XPath 2.0 | XSLT 2.0 | XQuery 1.0 Runtime
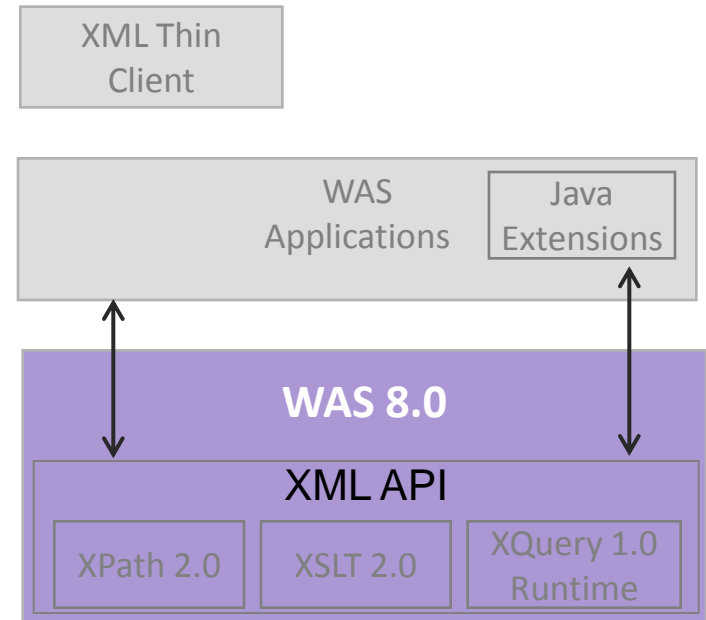
Session Initiation Protocol (SIP)

Develop, deliver and manage powerful large-scale mission-critical converged communications services and applications

Key Features:

- **Carrier Grade:** High availability, reliability, and scalability to meet the needs real time converged communications apps

- **Standards Based:** SIP Servlet 1.1 (JSR 289) including annotation support to reduce complexity & improve productivity

- **Converged Container:** HTTP, SIP and now with web services support to integrate Web services into a SIP-based applications

- **Ease of use:**
  – Simplified routing of SIP requests between multiple applications
  – Simplified use of back-to-back user agents (B2BUA) through new B2BUAHelper class

- **Security & Flexibility:**
  – Multihome support to send/receive requests over multiple NW interfaces for increased security without sacrificing productivity
  – Improved firewall support to simplify development & config. of SIP apps that consume/provide services through a firewall
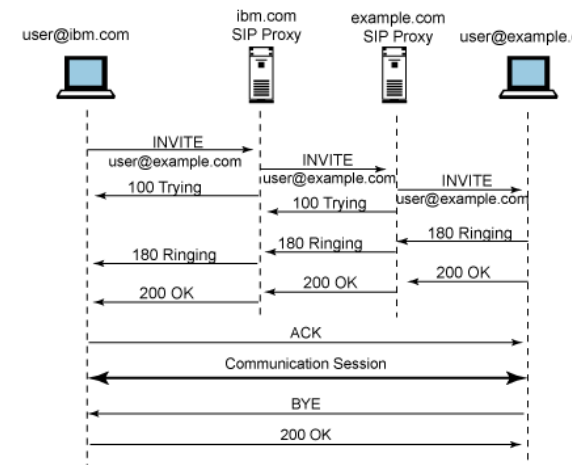
Communications Enabled Applications (CEA)

Simply and rapidly add communications capabilities, like Click to Call and Cobrowsing, to any Web application leveraging existing skills and an SOA approach

Key Features:

- **Simplicity:** 3 lines of code to add CEA into web app
- **Existing Skills:** Java & JavaScript
- **Mobile Browser Widgets:** Enable native look & feel
- **Telephony Access:** REST & Web service interfaces to Make call, disconnect call & incoming call notifications
- **Web 2.0 Widgets:** Customizable & extensible with iWidget support
  - Click to Call
  - Call Notifications
  - Collaboration Dialog
  - Contact Center Cobrowsing
  - Peer to Peer Cobrowsing
  - Two-way Synchronized Forms
- **PoC Friendly:** Unit test environment & pre-tested with Avaya, Cisco & Nortel unified communications products
- **Ease of Access & Use:** Integrated with WAS V8

Shopper's friend

Shopper
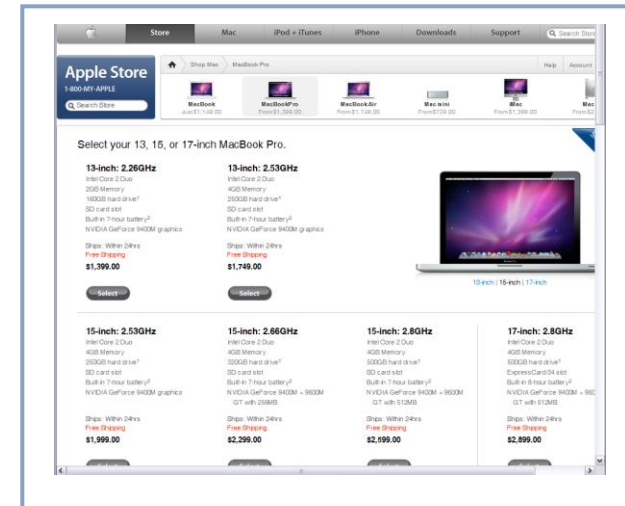
Contact Center Rep

- *Peer to Peer Cobrowsing*
- *Click to Call*
- *Contact Center Cobrowsing*
- *Two-way Synchronized Forms*

Dynamic Scripting

Leverage existing platform investment to rapidly address situational application requirements using PHP or Groovy

Key Features:

- **Time to Value:** Rapid development with PHP, Groovy, and a Web 2.0 oriented programming model based on WebSphere sMash

- **Reuse:** Develop and deploy application components supporting the iWidget specification that can be incorporated into WebSphere Portal and IBM Mashup Center-based applications

php **Groovy**

Web 2.0
REST, RSS / ATOM

http://www.projectzero.org/

Available as a Feature Pack supporting WAS v8, v7 & v6.1

New In V8.0 !

IBM

Monitored Directory Support

Accelerate edit-compile-debug tasks during the ***development*** lifecycle

- Enhanced ***developer*** productivity through new monitored directory-based application install, update and uninstall of Java EE applications

- Drag & drop and command line support

- Supported with WAS Express, Base, ND & z/OS

- Java 5 EE and Java 6 EE Supported file types:
  - EAR (Enterprise Archive)
  - WAR (Web Application Archive)
  - JAR (Java Archive)
  - SAR (SIP Application Resource)

© 2012 IBM Corporation

Lowering Barriers to Developer Adoption

- No charge WebSphere Developer Tools for Eclipse   Announced Q4/2011 !

- No charge WebSphere Application Server for Developers
  - For use on developer desktop at no charge

Download http://bit.ly/bq49yq

WebSphere Developer Tools and Rational Application Developer

RAD
Install: IM

WebSphere Integration
- Support for WAS v6.0, v6.1
- Test Environments for WAS v6.1, 7.0, v8.0
- Portal Tools & Portal Server support
- Profile applications on WAS
- Cloud: Deploy to IWD, or WebSphere/Portal instances on SCE

Problem Determination
- Code visualization - class, sequence and topic diagrams
- Static analysis (code review)
- Code coverage: optimize unit testing
- Profiling

Enterprise Connectivity
- J2C (EIS) tools
- CICS, and IMS Adapters
- Adapters for SAP, Siebel, JDE, Oracle, PeopleSoft

Team Productivity
- RTC integration
- Collaborative debug
- Collaborative code analysis

Programming Model Support
- SCA
- Java (WAS) Batch
- SIP/CEA
- XML (feature pack)

Extended Programming Model Support
- Advanced support for J2EE 1.4 and earlier:
  - EJB & Web Services deploy
  - DD editors
  - JAX-RPC
- Web:
  - Page & site designer
  - Web diagram Editor
  - Struts, JSF support
  - iWidget support

WDT
Install: Eclipse update site or IM

JEE Tools
- Advanced support for JEE 5+
- DD editors, enhanced project explorer, additional validation

Web Tools
- Advanced web development tools
- Rich page (WYSIWYG) editor for HTML, JSP
- Web 2.0 and Mobile support

WebSphere Integration
- Support for WAS v7.0, v8.0
- Publish, start/stop the server
- Debug Jython/wsadmin scripts

WAS Extensions Support
- Binding and extension editors
- Support for non-spec extensions

OSGi Tools
- Full creation and editing support
- Blueprint editor and validation
- Visual Bundle Explorer

Liberty Profile Integration
- Publish, start/stop the server
- Edit & manage server configuration

Eclipse (WTP, DTP)

Programming Model Support
- Basic creation, editing, and validation support for JEE applications:
  - Web, XML, JPA, EJB, EAR
- Database tools

WDT adds support for Mac for dev/test with WAS Liberty Profile

46

© 2012 IBM Corporation

WAS Tools Edition Bundles

Team

WAS ND – Tools Edition

- Like "WA4D – Tools Edition" but for WAS ND production use

WAS – Tools Edition

- Solution: Production WAS + unlimited tools (RAD or WAS Developer Tools)
- Terms (runtime): Production use
- Terms (tools): Unlimited use of tools for developing applications to be deployed on WAS included with this bundle.

WAS for Developers – Tools Edition for Eclipse

- Solution: WAS for Developers + WAS Developer Tools
- Terms: Single user. Development use only
- Freely available, supported for a fee
- Easily obtained for rapid development to WAS v7, v8, v8.5 and Liberty

Individual

IBM Assembly and Deploy Tools for WebSphere Administration (IADT)

Rapidly assemble & deploy applications to WebSphere Application Server environments

Key Capabilities:

- Import and validate applications

- Edit deployment descriptors and binding files

- Edit EAR-level configuration (Enhanced EAR)

- Create and debug Jython and wsadmin scripts

- Deploy EJB and web services

- Deploy applications to local or remote WAS v8.x servers

- Debug applications on WAS v8.x

- IADT tools replace the previously available IBM Rational Application Developer Assembly and Deploy function

- Restricted to assembly and deployment usage only

Application Adapters

Enhance reuse and extend application asset life

- IBM WebSphere Adapters 7.5 included with WAS V8:

    – SAP Software

    – Siebel Business Applications

    – Oracle E-Business Suite

    – JD Edwards EnterpriseOne

    – PeopleSoft Enterprise

- Supported for development & test with WebSphere Application Server as part of WAS V8 license

- Production usage requires separate WebSphere Adapters license
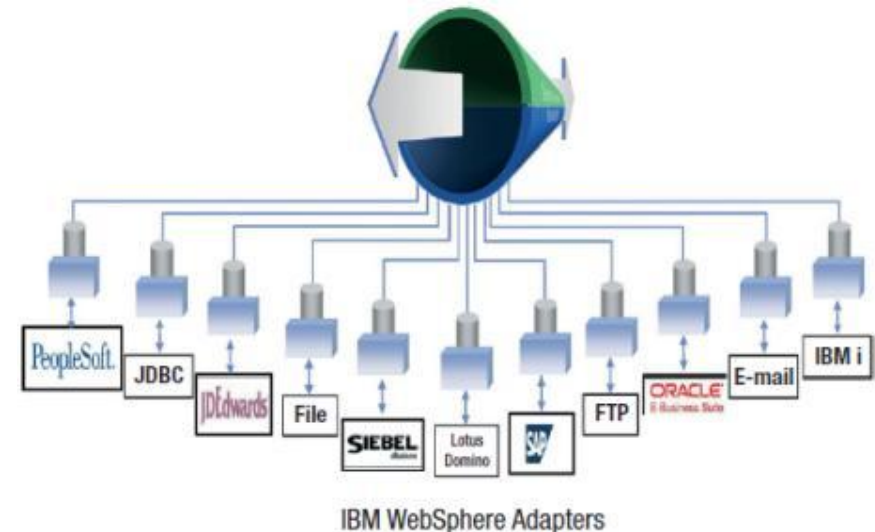
IBM WebSphere Adapters

Unparalleled Application Development and Management Environment, Rich User Experiences…Faster

Developer Experience

Fast, flexible, and simplified application development

- Java 6 EE
- Liberty Profile
- Expanded Tooling and WAS Tooling Bundles
- Web 2.0 & Mobile Toolkit; IBM Worklight Integration
- JDK7 Support
- Migration toolkit
- OSGI programming model enhancements
- EJB support in OSGi apps
- WebSphere Batch enhancements

Application Resiliency

Intelligent Management & Enhanced Resiliency

- Improved Performance
- Application Edition Management
- Application Server Health Management
- Dynamic Clustering
- New Intelligent Routing capabilities
- Installation and Maintenance
- Messaging infrastructure resiliency
- Memory leak detection & protection in WAS

Operations and Control
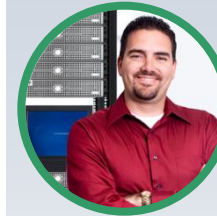
Improved Operations, Security, Control & Integration

- Selectable JDK
- Configuration Change Tracking
- OSGi Blueprint security improvements
- Cross Component Trace (XCT)
- Enhanced IBM Support Assistant
- Better log and trace filtering

WAS V8 Performance Enhancements

Reduce TCO through higher performance application foundations

- **Java 6**
  - JVM runtime enhancements
  - JIT optimizations

- **Application Performance Improvements vs. WAS v7**
  - DayTrader: Up to 23%
  - OSGi Applications: Up to 26%

- **End-to-end performance improvements vs. WAS v7 including**
  - Up to 15% faster server startup time for developers
  - Up to 92% faster application server creation in a large topology
  - Up to 58% faster application server cluster creation in a large topology
  - Up to 34% faster application deployments in a large topology
  - Up to 10% better vertical scaling on larger multi-core systems
  - JPA 2.0 optimizations with DynaCache and JPA Level 2 cache

Performance data is based on WebSphere Application Server in a distributed environment

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including 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 results similar to those stated here.

WebSphere V8.5 Middleware Performance Leadership

WAS V8.5 Performance Enhancements

- Significant Performance Improvements across many components through Engineering Efforts

- Performance Improvements can be seen in
    - JDK 7.0
    - JPA 2.0 Persistence Layer
    - WebContainer
    - JSP Engine
    - EJBContainer
    - J2C/RRA and Connection Management
    - Messaging Optimizations

WAS V8.5 Performance Improvements

SPECjEnterprise 2010
(Release-to-Release Performance)

Chart (EjOPS/core):
- WAS 7.0.0.5 (8-core Nehalem EP): 126.7
- WAS 7.0.0.9 (8-core Nehalem EP): 149.4
- WAS 7.0.0.9 + JPA 2.0 FP (8-core Nehalem EP): 226.7
- WAS 8.0 (8-core Nehalem EP): 292.6
- WAS 8.0 (12-core Westmere EP): 307.9
- WAS 8.5 (12-core Westmere EP): 524.6
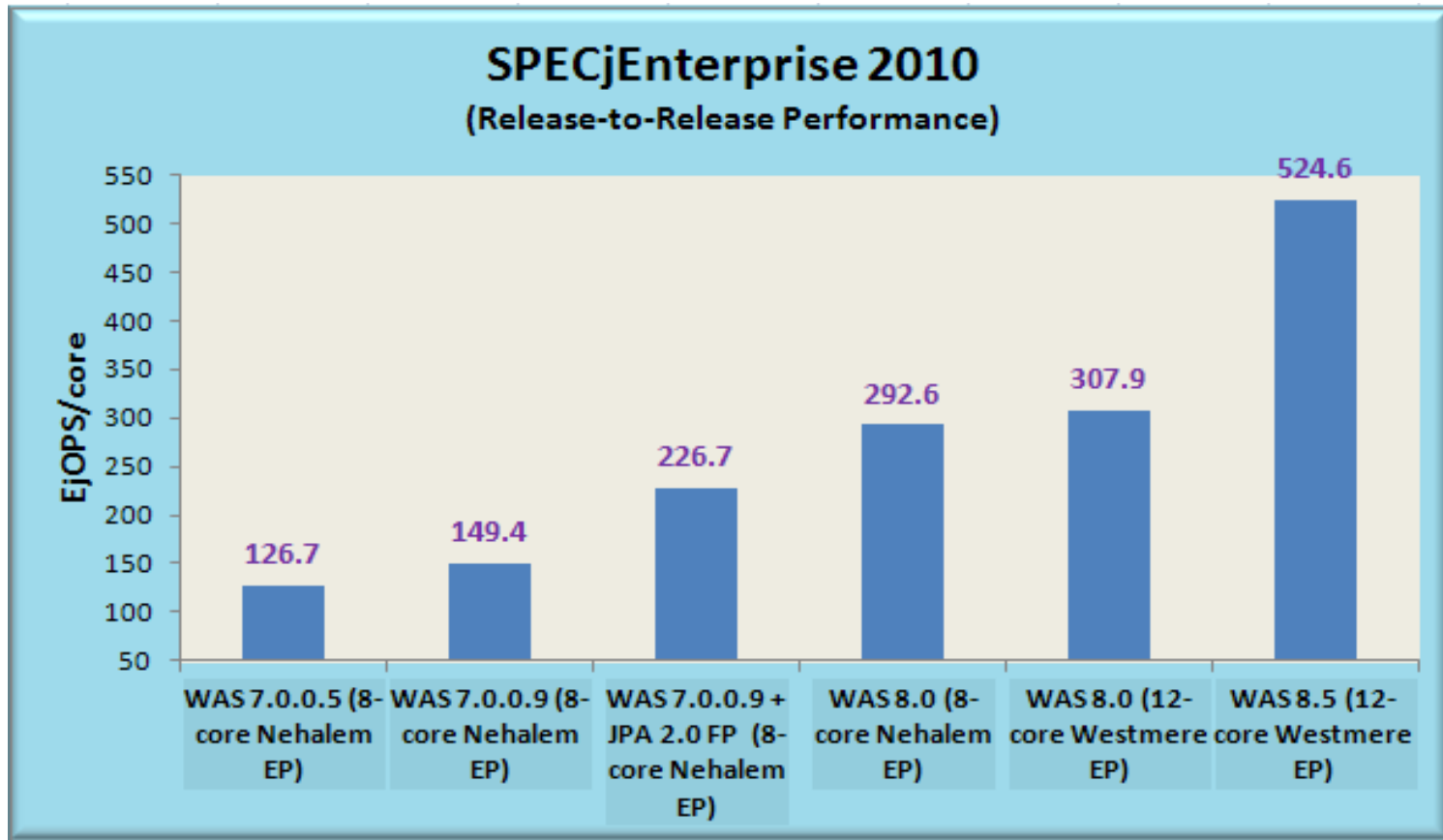
Consistent Performance gains across WAS Releases

As per SPEC Published Data as of 4/26/2012

http://www.spec.org/jEnterprise2010/results/jEnterprise2010.html

WAS for z/OS Performance & Utilization Enhancements

> Reduce TCO through higher performance application foundations

- **Performance improvement through z196 hardware exploitation**

- **Enhanced system utilization by providing granular control over application server configuration parameters for application consolidation**
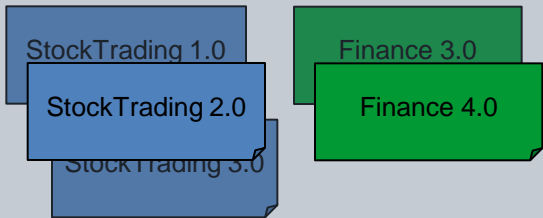
IBM System z

WAS V8.5 Intelligent Management
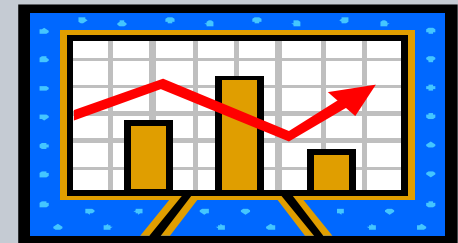
Extending QoS through autonomic computing

Application Edition Management
Self-Managing

StockTrading 1.0
StockTrading 2.0
StockTrading 3.0

Finance 3.0
Finance 4.0

Health Management
Self-Protecting
Self-Healing

Dynamic Clustering
Self-Optimizing

Intelligent Routing

Explanations in the WAS v8.5 for z/OS IC

Application Edition Management

> ## Applications can be upgraded without incurring outages

- Upgrade Applications without interruption to end users
- Concurrently run multiple editions of an application
  - Automatically route users to a specific application
- Multiple editions can be activated for extended periods of time
- Rollout policies to switch from one edition to another without service loss
- Easily update OS or WebSphere without incurring down time
- Easy-to-use edition control center in admin console, plus full scripting support

| Validation Mode | Rollout Policies | Concurrent Activation |

Health Management

Sense and respond to problems before end users suffer an outage

- Automatically detect and handle application health problems
  - Without requiring administrator time, expertise, or intervention
- Intelligently handle health issues in a way that will maintain continuous availability
- Each health policy consists of a condition, one or more actions, and a target set of processes
- Includes health policies for common application problems
- Customizable health conditions and health actions

Comprehensive Health Policies

Customizable Health Conditions

Customizable Health Actions

Intelligent Routing

Improves business results by ensuring priority is given to business critical applications

- Requests are prioritized and routed based upon administrator defined rules
  - Flexible policy-based routing and control
- On Demand Router (ODR) is the focal point for Intelligent Routing
- A routing tier that's aware of what's happening on the application server tier
  - Application server utilization, request performance, etc…
- Route work to the application server that can do it best
- Provide preference for higher priority requests
- Integrates with Health Management and Dynamic Clustering

Dynamic Clustering

Proactively provision and start or stop application servers based on workload demands to meet Service Level Agreements

- Associate service policies with your applications
  - Let WebSphere manage to the service goals
- Programmatically respond to spikes in demand
  - Add or reduce application server instances as appropriate
- Automatically recover from infrastructure problems
- Includes automatic start and stop of cluster members based on load for MQ-driven applications
- Decrease administrative overhead required to monitor and diagnose performance issues

Transactional Integrity

No transaction is ever lost or violated with WebSphere application infrastructure's built-in transaction integrity

- WebSphere Platform designed as a transactional server from Day One.
  - For all transaction protocols (XA, OTS, WS-AT)
  - On all platforms

- WebSphere transaction costs are "pay-as-you-go"
  - WebSphere fully optimized for 1PC and dynamically engages 2PC *only* after a 2nd resource is used
  - Distributed transaction contexts created dynamically *only* during the first remote request in a transaction
  - No change to application configuration as it engages 2PC or distributed transactions → the "simplest" configuration always works.

- Flexible configuration
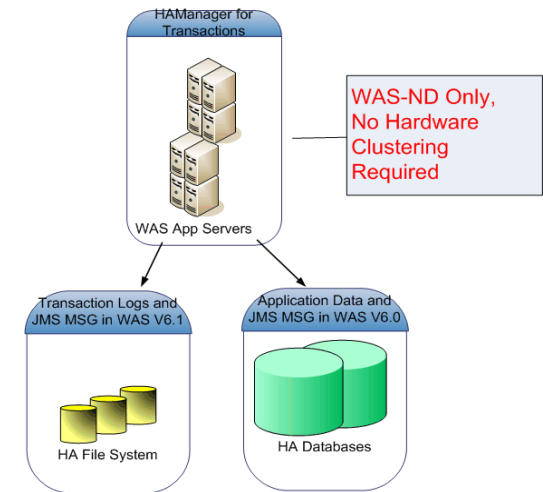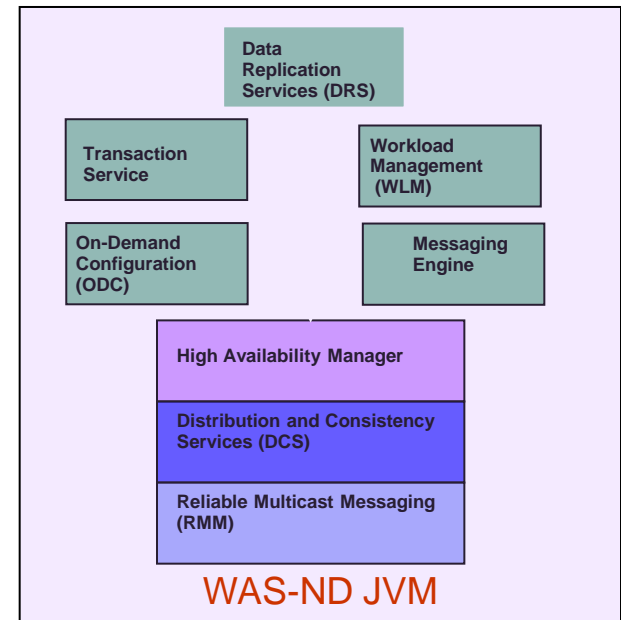 - e.g. Per-application resource commit-ordering for DB/JMS scenarios
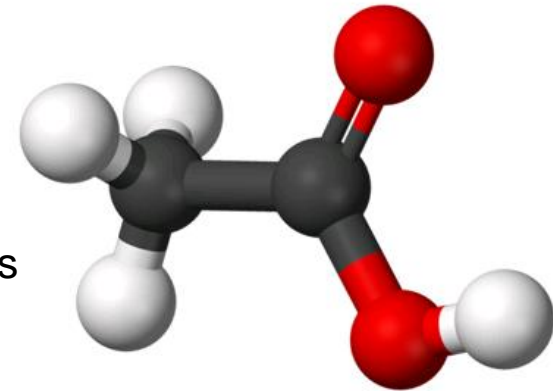
IBM has been doing this right for 40+ years

WebSphere Application Server: HA Architecture – A Brief Review

High Availability Architecture provides:

- Peer Recovery Model with Active Hot Standbys for persistent services
  - Transactions
  - Messaging

- If a JVM fails then any Singletons running in that JVM are restarted on a Peer once the Failure is detected

- Starting on an already running Peer eliminates the start up time of a new process which could take minutes

- Planned failover takes a few seconds

- This low failover time means WAS can tolerate many failures without exceeding the 5.5 minute yearly maximum outage dictated by 99.999%

- **Failure Scenario:**
  - Client calls EJB which updates database using 2PC transactions
  - Failure during in-process transaction (after the prepare statement)
  - Database record is locked until transaction is recovered (committed or rolled back)

- **WAS-ND HA**
  - HA Manager detects the failure
  - Failover to a peer server which recovers the transaction log (shared on a NAS) from the failed server
  - Recovery in a few seconds

Data Replication Services (DRS)

Transaction Service

Workload Management (WLM)

On-Demand Configuration (ODC)

Messaging Engine

High Availability Manager

Distribution and Consistency Services (DCS)

Reliable Multicast Messaging (RMM)

WAS-ND JVM

HAManager for Transactions

WAS App Servers

WAS-ND Only, No Hardware Clustering Required

Transaction Logs and JMS MSG in WAS V6.1

Application Data and JMS MSG in WAS V6.0

HA File System

HA Databases

High Availability Improvements

Reduce unexpected and expected operational down time

- Improved HA support for messaging applications
  - Reconnect to a standby gateway queue manager when an active queue manage fails or becomes available

- Resource failover and retry logic for relational data sources and JCA connection factories
  - Simplifies application development
    - Minimizes the application code required to handle failure of connections to relational databases and other JCA resources
    - Provides a common mechanism for applications to uniformly respond to planned or unplanned outages
  - Typically Employed with Database Replication (e.g DB2 HADR, Oracle RAC)
  - Administrator can tailor data sources and connection factory configuration based on application needs:
    - Alternate/failover resource reference on primary data source
    - Optionally
      - ❑ Number of connection retries
      - ❑ Pre-population of alternate/failover resource connection pool
      - ❑ Auto failback
  - Full control of functionality available to scripts and programs via management MBean

- Improved reliability & performance with DB2
  - Support for client affinity & client reroute for apps that use IBM DB2
  - New location transparency for EJBs using DB2 connections

- Improved transactional integrity
  - Support for shared DB locks between transaction branches and integration of new programming models with WAS proven transaction engine

WAS for z/OS High Availability Enhancements

Reduce unexpected and expected operational down time

- HA enhancements for WebSphere Optimized Local Adapters resource adapter failover scenarios

- Enhanced resource and request routing by matching failure notification codes with response actions

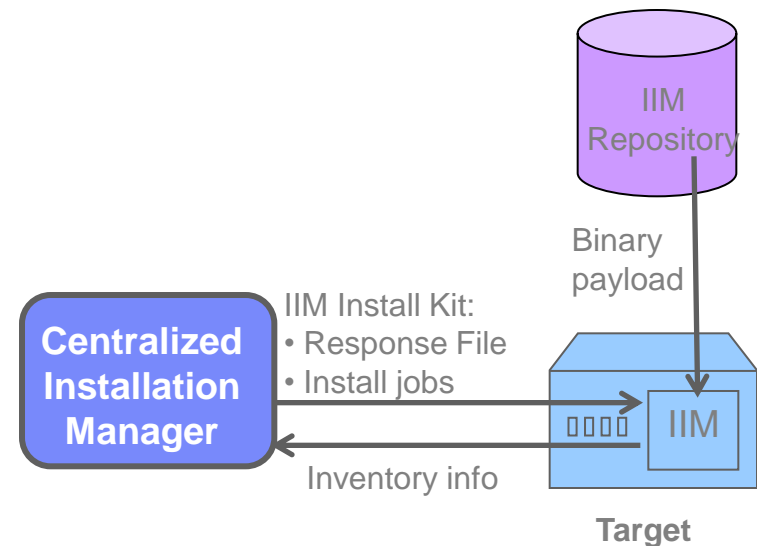**IBM System z**

IBM Installation Manager

Faster time to value & lower operational costs through new install & maintenance tech.

- Full local & centralized product lifecycle management:
  - Install/Uninstall
  - Update/Rollback (Fixpacks and iFixes)
  - Modify (Add/Remove features)

- Installs exactly the desired level of service in one pass
  - No need to install GA product first and then apply a fixpack and/or ifixes as a separate step

- Lays down binaries relevant to user selections and system environment

- GUI & response file modes of operation
  - GUI to perform individual operations
  - Response files can be recorded from the GUI or created by specifying the appropriate xml
  - Silent mode support for invoking multiple operations

- Single user experience across WAS, WAS components & various IBM products
  - A single instance of IIM can manage the product lifecycle for any IM based products, from WebSphere, Rational, etc.
  - Support for WAS, IHS, WCT, etc.

Centralized Installation Manager (CIM)
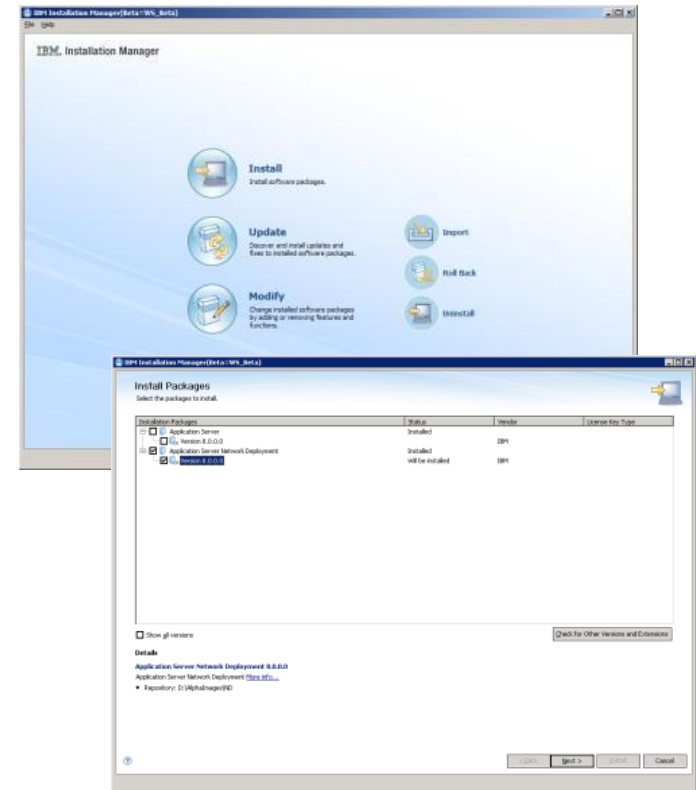
Faster time to value & lower operational costs through new install & maintenance tech.

- CIM V8 is available from Job Manager & DManager

  – Job Manager based solution spans the boundaries of the cell

  – Install targets are specified in agentless fashion

  – Install and config job scheduling is supported

- CIM V8 is able to remotely install WebSphere Application Server, IBM HTTP Server, Application Clients, DMZ Security Proxy Server, and Web Server Plug-ins

- Better scalability due to more distributed architecture

- Distributed & z/OS scenarios supported

- "CIM V7" function is still available with Deployment Manager along with new "CIM V8" function

IIM Repository

Binary payload

Centralized Installation Manager

IIM Install Kit:
• Response File
• Install jobs

IIM

Inventory info

Target

Separation between Job Manager, Target Hosts and IIM repositories

WebSphere Customization Toolbox (WCT)

Consolidates Multiple Stand-alone Tools

- **Replacement for Profile Management Tool**
  - Manages Profiles
  - Migration
  - Web Server Plug-in Configuration

67
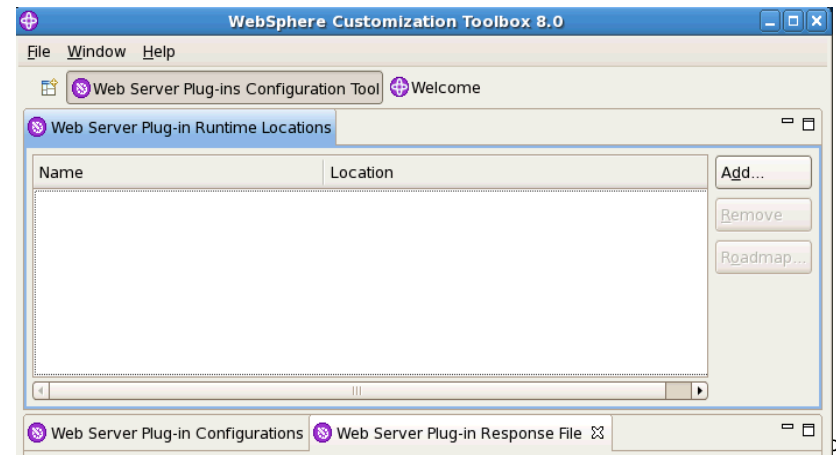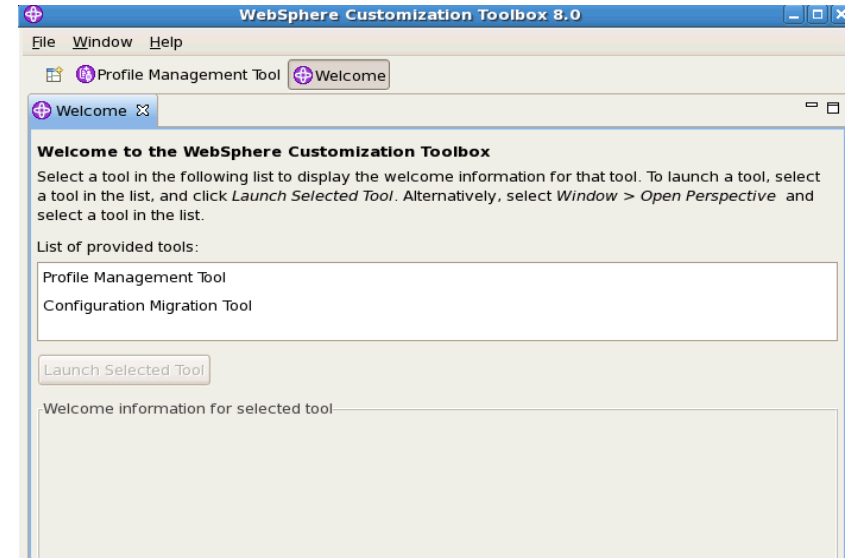
V8.5 Default Messaging Provider Infrastructure Improvements

- Restrict *long running* Database Locks
  - Active ME now holds only *short locks* on the SIBOWNER table while revalidating its ownership at regular intervals

- Ability for SIBus to detect a hang in the "*active*" ME and switch over to the "*standby*" ME
  - Adds ME Last Update Time to SIBOWNER Table
  - Backup ME Can Safely Take Ownership and avoid Split Brain

- ME able to *gracefully stop* from database failures instead of *killing* the entire JVM
  - Other Applications In JVM Hosting ME Continue to Run

- Automatically "*re-enable*" a ME if it enters a "*disabled*" state
  - In a Large Cluster It Can Be Difficult to Administratively Determine "disabled" ME

- Configure a new ME to *recover data* from a orphaned persistence store
  - Reads and Updated ME UUID from Persistent Records

- *Persist* JMS re-delivery count value
  - Avoids Reprocessing of Message That May Cause Outage

- Utilization of multi-cores for *quicker ME start-up* when large number of messages and destinations are present

Memory Leak Detection and Protection in WAS V8.5

Reduce possibilities
of memory leak in
your applications

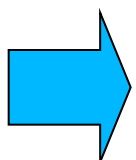Get enough info.
if leak is detected
to help fix my app

List stopped
apps that have
memory leaks

WebSphere Application Server V8.5:

- Ability to mitigate memory leak when stopping apps

- Ability to prevent leaks, receive leak warnings and get heap/system dumps

- MBean to list stopped apps that have memory leaks

- Java Heap not native

Platform & Environment Flexibility

Enhanced! **IBM**

Lower TCO through aligning business needs with platform/environment capabilities

Platform Specific Exploitation

**z/OS
Linux for System z**

AIX, i5/OS

**Linux
Windows**

**IBM System z**   **IBM System p, i**   **IBM System x**

Match deployments to preferred hardware: Platform specific exploitation on IBM systems along with broad hardware & OS support including HP Itanium, Oracle SPARC, AIX, IBM i, z/OS, Linux, Windows, HP-UX & Oracle Solaris

Common Management of Heterogeneous Systems

WAS	WAS	WAS	WAS	WAS	WAS
z/OS	Linux on System z	AIX	IBM i	Linux	Windows
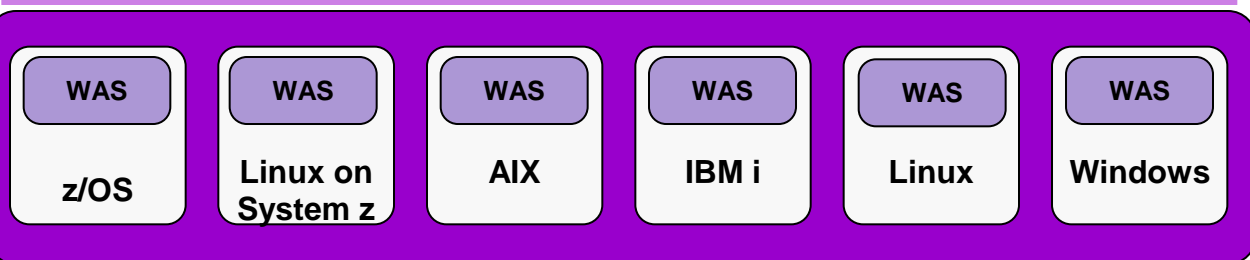
IBM zEnterprise System
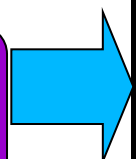
Match deployments to preferred OS on a unified zEnterprise System: Take advantage of zManager to manage multiple platforms under a logical management infrastructure

Unparalleled Application Development and Management Environment, Rich User Experiences…Faster

Developer Experience

Fast, flexible, and simplified application development

- Java 6 EE
- Liberty Profile
- Expanded Tooling and WAS Tooling Bundles
- Web 2.0 & Mobile Toolkit; IBM Worklight Integration
- JDK7 Support
- Migration toolkit
- OSGI programming model enhancements
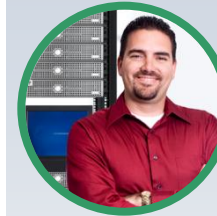- EJB support in OSGi apps
- SCA OASIS programming model

Application Resiliency

Intelligent Management & Enhanced Resiliency

- Improved Performance
- Application Edition Management
- Application Server Health Management
- Dynamic Clustering
- New Intelligent Routing capabilities
- Installation and Maintenance
- Messaging infrastructure resiliency
- Memory leak detection & protection in WAS
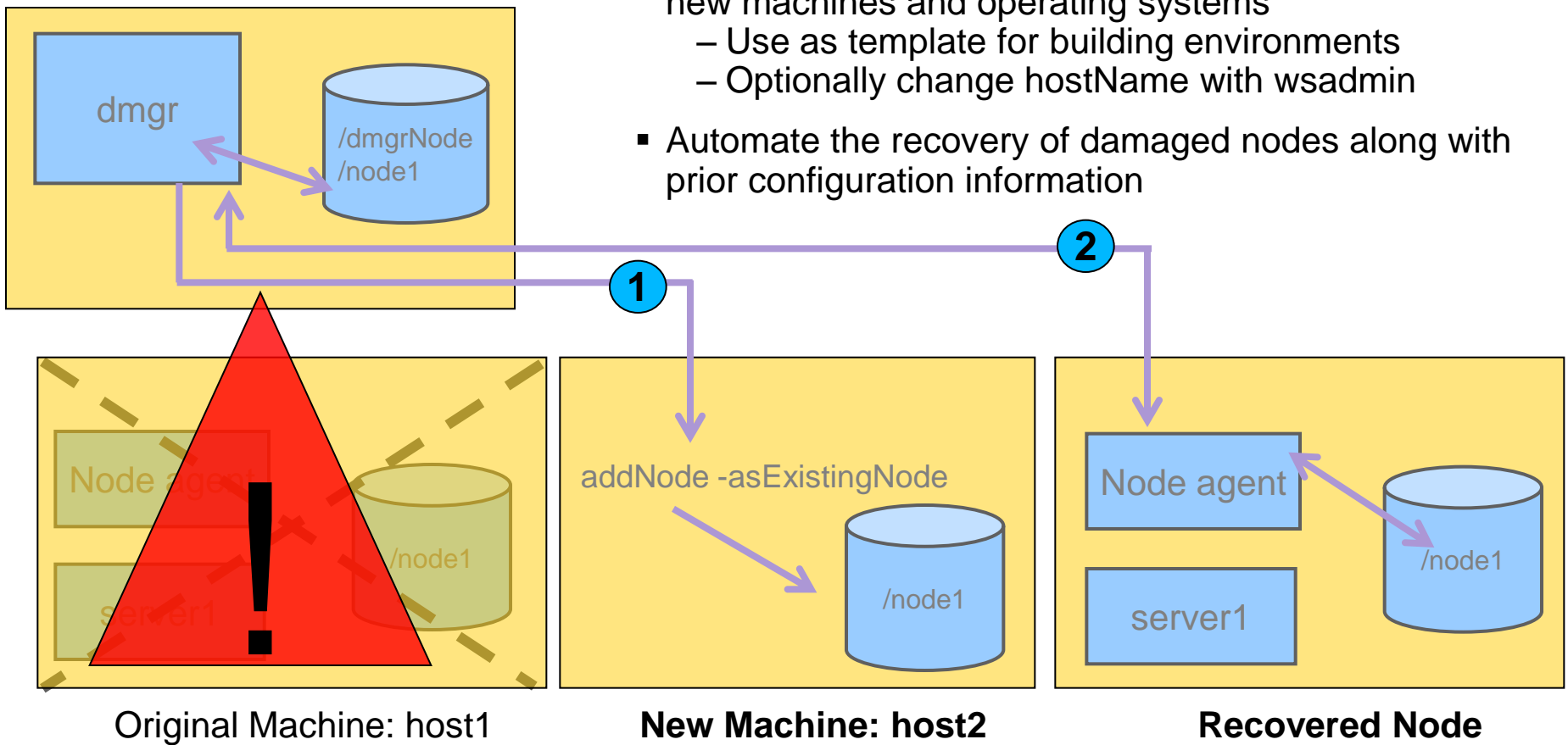
Operations and Control

Improved Operations, Security, Control & Integration

- Selectable JDK
- WebSphere Batch enhancements
- Configuration Change Tracking
- OSGi Blueprint security improvements
- Cross Component Trace (XCT)
- Enhanced IBM Support Assistant
- Better log and trace filtering

Move and Recover Nodes with Ease

Improve administrator productivity and minimize down time

- Automate the movement of existing deployments to new machines and operating systems
  - Use as template for building environments
  - Optionally change hostName with wsadmin

- Automate the recovery of damaged nodes along with prior configuration information

dmgr

/dmgrNode /node1

1

2

Node agent

server1

/node1

addNode -asExistingNode

/node1

Node agent

server1

/node1

Original Machine: host1

Damaged Node

New Machine: host2

Recovered Node

WAS V8 Additional Administrator Productivity Enhancements

Improve administrator productivity and minimize down time

- Job Manager enhancements
  - Simplify the creation, augmenting and deletion of profiles on remote nodes

- Enhanced portability of Properties File Based Configuration
  - Speed and standardize customizations across different cells

- Enhanced Properties File Based Configuration format
  - Easier editing of application deployment options

- Administrative option for all platforms to list all SDKs in use and select SDK to use from among supported Java SDKs
  - manageSDK command replaces enableJVM command
  - Typically used on z/OS and iSeries for 31/32 bit and 64 bit SDK selection
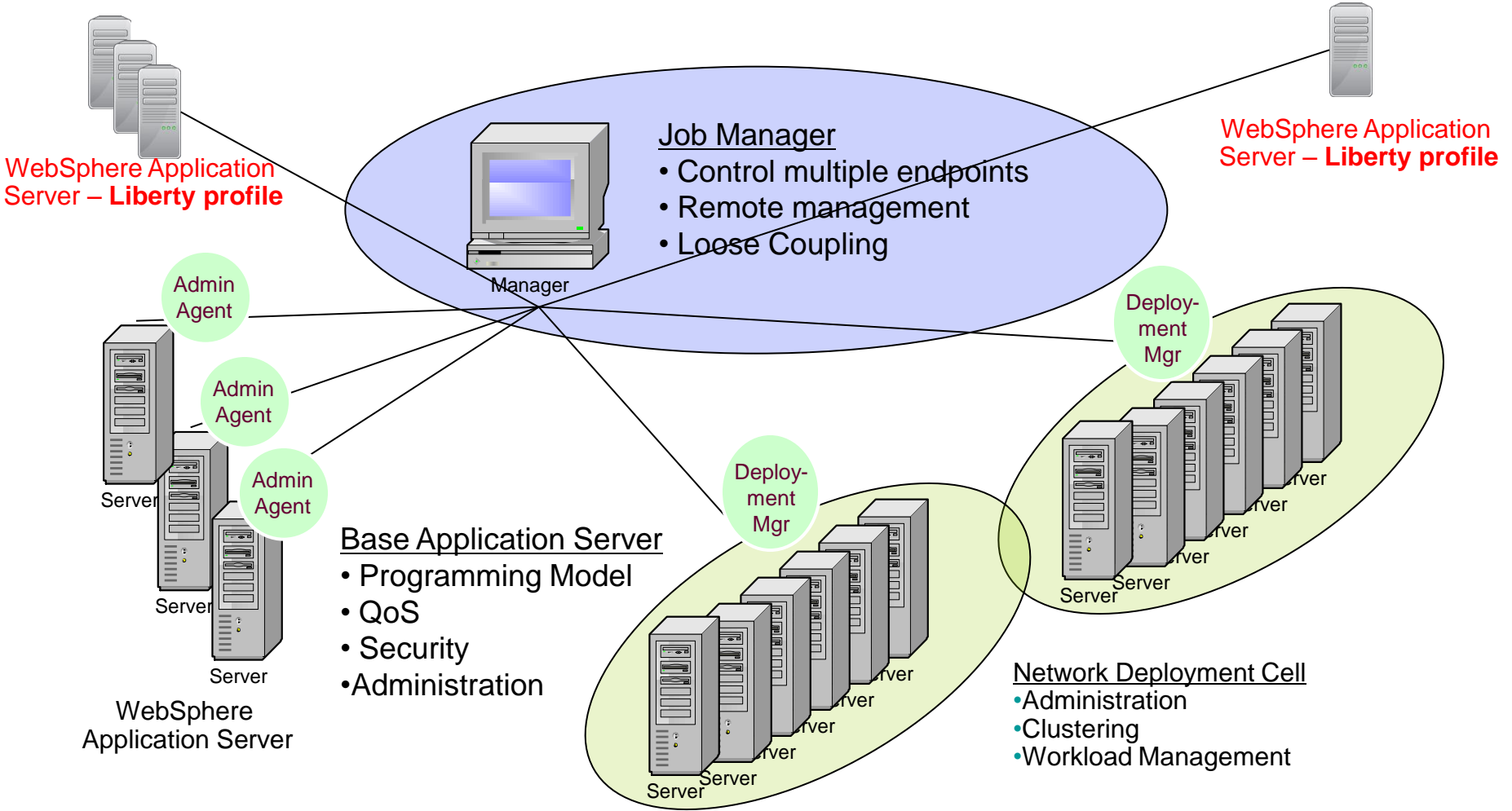
WAS V8.5 Selectable SDK

Allow development and production environments to select the most appropriate JDK for the situation (JDK 6 or 7)

- WAS v8.5 introduces "selectable" JDK:
  - Some of the environment can use Java 7 while the rest continues to use Java 6
  - Use Java 7 in a small subset of your topology & keep the rest on Java 6
  - Switch back and forth between Java 7 and Java 6 as necessary
- Install as feature extension to new or existing WAS v8.5 installation
  - Use with either full WAS profile or Liberty profile
- Create WAS admin profiles for developer use
  - Use managesdk to set WAS new profile and command defaults to Java 7
  - Create admin profile and start server
- Build and test Java 7 applications
  - Use ant or maven to build Java 7 applications
    - See PlantsByWebSphere sample docs for information
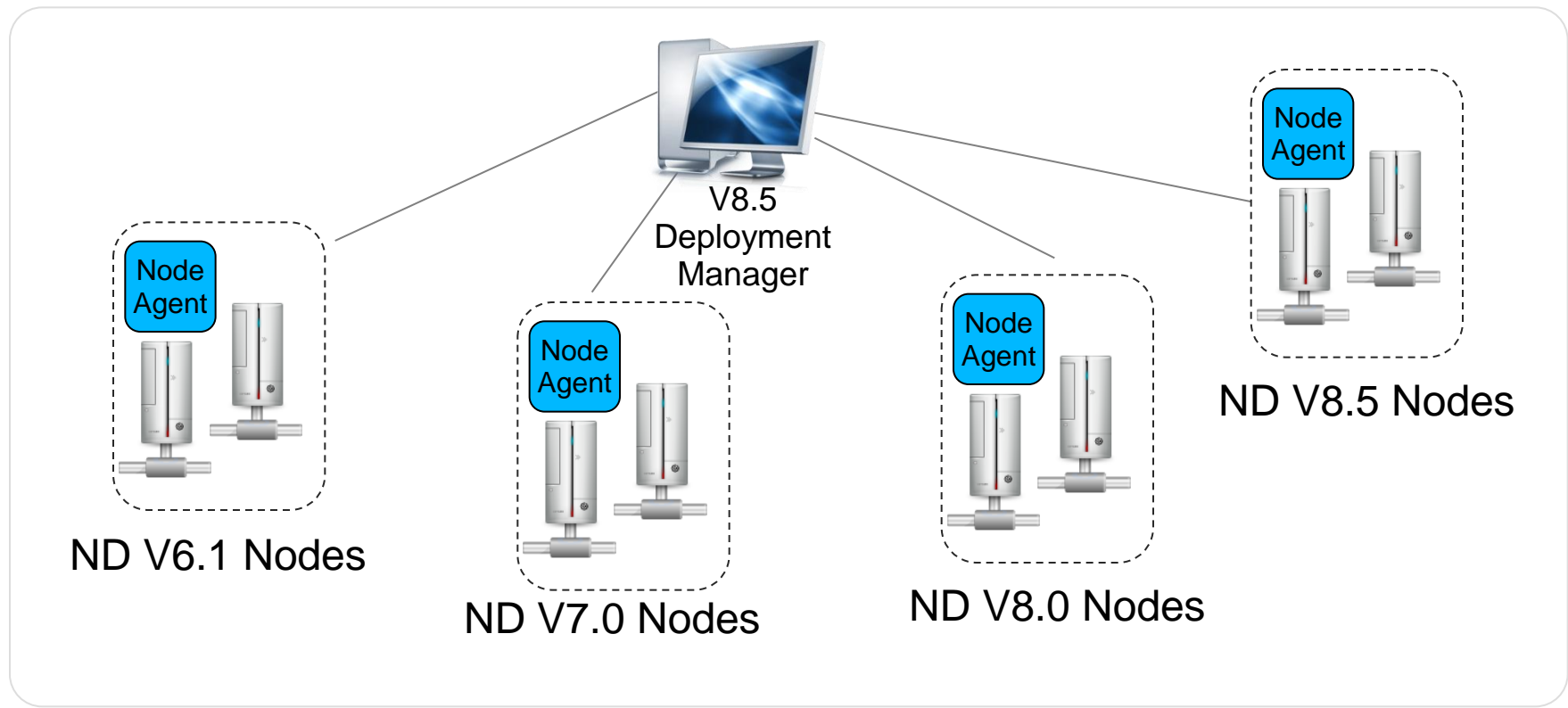 - Use RAD to develop, deploy, and test Java 7 applications

WAS V8.5 Job Manager Improvements

WebSphere Application Server – **Liberty profile**

WebSphere Application Server – **Liberty profile**

Admin Agent

Admin Agent

Admin Agent

Server

Job Manager
- Control multiple endpoints
- Remote management
- Loose Coupling

Manager

Deploy-ment Mgr

Deploy-ment Mgr

Base Application Server
- Programming Model
- QoS
- Security
- Administration

Server

Server

WebSphere
Application Server

Server
Server
Server
Server
Server

Server
Server
Server
Server
Server

Network Deployment Cell
- Administration
- Clustering
- Workload Management

Mixed Version Cell Support

Support for existing infrastructure in new V8.5 deployments to save time, money and reduce risk
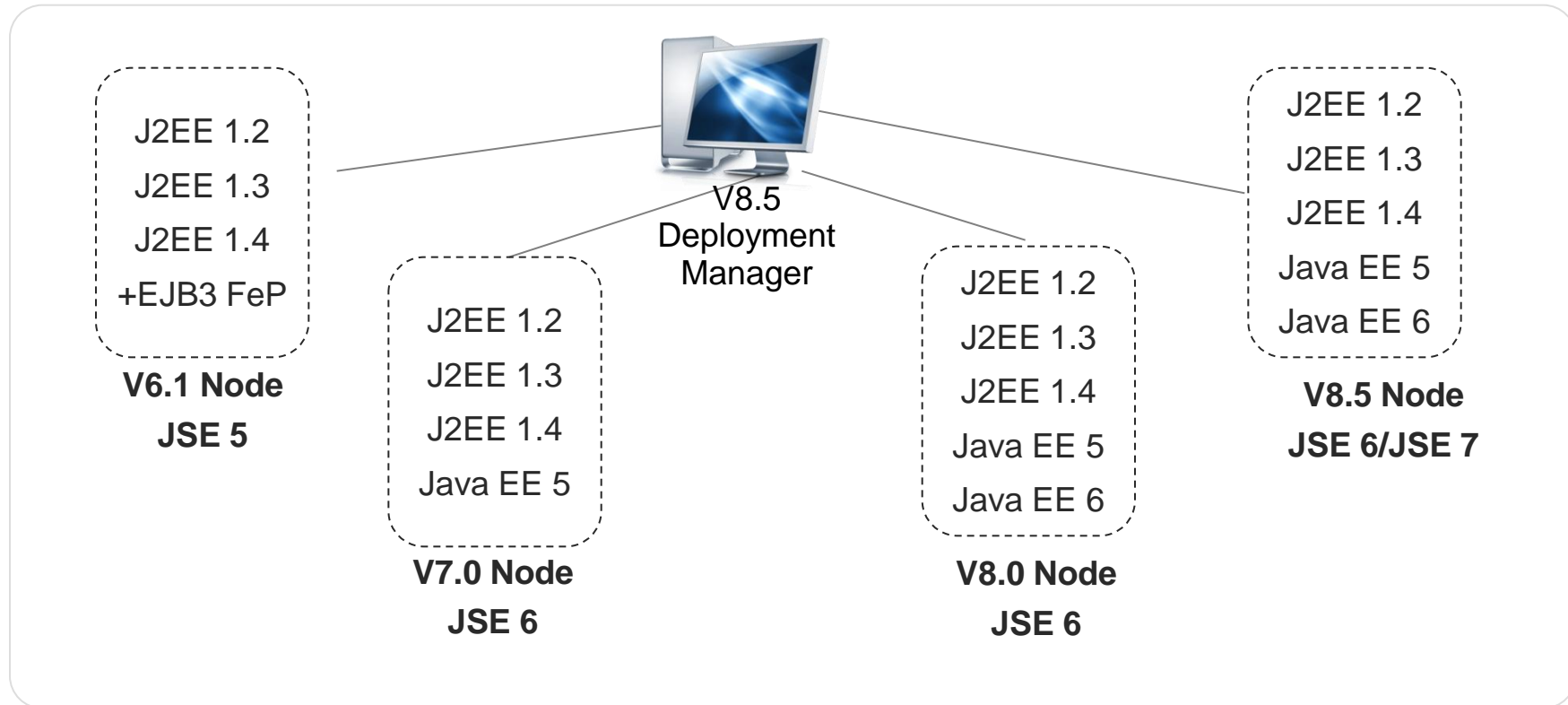
WAS Network Deployment V8.5 Cell

Node Agent

V8.5 Deployment Manager

Node Agent

ND V8.5 Nodes

Node Agent

ND V6.1 Nodes

Node Agent

ND V7.0 Nodes

Node Agent

ND V8.0 Nodes

V8 Cell can contain 6.1. 7.0 & 8.0 nodes

Support for Existing Applications

Support for existing Java EE applications in new V8.5 deployments to continue achieving value from existing investments
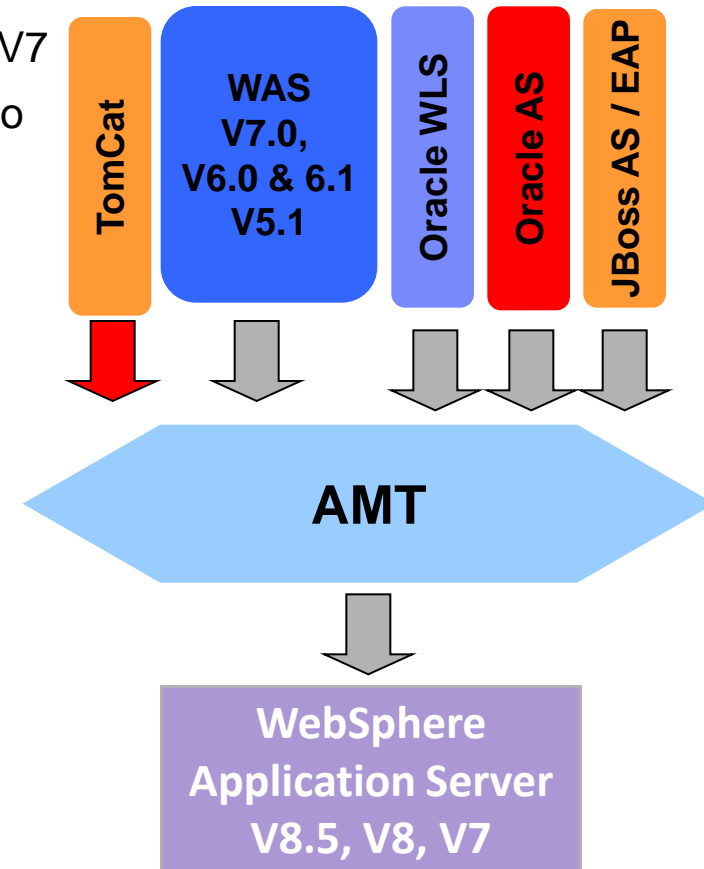
WAS Network Deployment V8.5 Cell

J2EE 1.2
J2EE 1.3
J2EE 1.4
+EJB3 FeP

V6.1 Node
JSE 5

J2EE 1.2
J2EE 1.3
J2EE 1.4
Java EE 5

V7.0 Node
JSE 6

V8.5
Deployment
Manager

J2EE 1.2
J2EE 1.3
J2EE 1.4
Java EE 5
Java EE 6

V8.0 Node
JSE 6

J2EE 1.2
J2EE 1.3
J2EE 1.4
Java EE 5
Java EE 6

V8.5 Node
JSE 6/JSE 7

V8.x Cell can contain 6.1. 7.0 & 8.x nodes

Enhanced! **IBM**

Application Migration Toolkit V3.5

Migrate applications from WebSphere & other Java EE application servers to WebSphere faster with minimized risk

- Migrate applications from older releases to WAS V8.x or V7

- Migrate from Tomcat, Oracle or JBoss faster and easier to WAS V8.x 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 and JDK 7differences
    - 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)

TomCat **WAS V7.0, V6.0 & 6.1 V5.1** **Oracle WLS** **Oracle AS** **JBoss AS / EAP**

AMT

WebSphere Application Server V8.5, V8, V7

Get the Tool at No Charge: http://ibm.co/hqfkdj   © 2012 IBM Corporation

ITCAM for WebSphere Application Server for WAS 8

Lower TCO and minimize down time through integrated monitoring tools
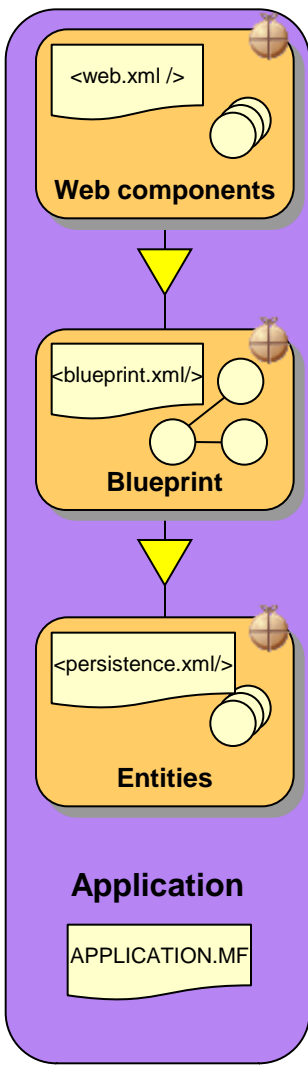
IBM Tivoli Composite Application Manager (ITCAM)

- Data Collector available in WebSphere Application Server v8.0 as an extension offering (optional install)

- V8.5 Download and Install
  - https://www-304.ibm.com/software/brandcatalog/ismlibrary/

- ITCAM for WebSphere Application Server provides additional request-based response time and CPU metrics
  - 90%CPUUsage
  - 90%ResponseTime
  - AverageCPUUsage
  - AverageResponseTime
  - LastMinuteAverageCPUUsage
  - LastMinuteAverageResponseTime
  - MaximumCPUUsage
  - MaximumResponseTime
  - MinimumCPUUsage
  - MinimumResponseTime
  - RequestCount

- Customer application code is not instrumented in any way

- Simple upgrade from ITCAM for WebSphere Application Server to ITCAM for Application Diagnostics – no rip and replace

- After upgrade ITCAM data still visible in Tivoli Performance Viewer as well

© 2012 IBM Corporation
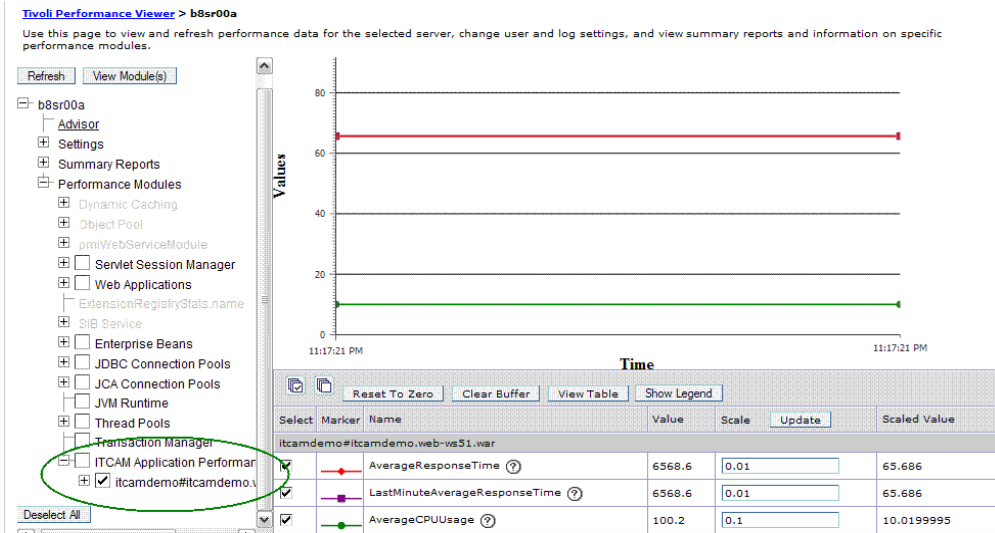
Dynamic Application Update of OSGi Apps

Rapidly extend applications to meet new business requirements with reduced down time

- Administratively preview new bundles before making updates
- In-place bundle update enables application to remain continuously available throughout the update process

Dynamic Application Extension of OSGi Apps

Rapidly extend applications to meet new business requirements with reduced down time

WebSphere. software — Welcome admin — H

Cell=paddyNode02Cell, Profile=AppSrv01

View: All tasks

- Welcome
- Guided Activities
- Servers
- Applications
  - New Application
  - Application Types
    - WebSphere enterprise applications
    - Business-level applications
    - Assets
  - Global deployment settings
- Services
- Resources

Business-level applications

Business-level applications > Trader > com.acme.trader_1.0_0001.eba > Preview

A preview of the result of the proposed changes to the bundle versions in this application.

Symbolic name	Deployed version	New version
com.acme.trade.manager	1.1.0	1.1.0
com.acme.trade.manager.api	1.0.0	1.0.0
com.paytrade.provider.extension	NOT_DEPLOYED	1.0.0

OK   Cancel

TradeAPI

TradeProvider1 (Extension)

TradeProvider2 (Extension)

TradeManager

TradeProvider3 (Extension)

Application

- Administratively add new functionality to deployed applications

- Well-designed extensions result in zero application down-time as extensions are added and removed

81

Federated Repository (VMM) Support for Multiple Security Domains

Enhanced security control and flexibility for improved business agility

- . Removed WAS7 restriction
  - In WAS7, only one VMM configuration per cell was allowed

- VMM Supported types of configurations
  - Global Federated Repository
    - New User Registry type
    - One configuration allowed per cell
    - Sharable across
      - Global Security configuration (Admin and default)
      - 1 or more Security Domain

  - Federated Repository
    - User Registry type that existed since WAS61
    - One or more configuration allowed per cell
    - Uniquely define at Global Security and/or at a Security Domain Level.

- WAS V8 Security Domain Scopes
  - Cell, Cluster, Application Server, SIBus

V8.0 Security Enhancements

Utilize a more secure environment out of the box

- **Security hardening**
  - Require SSL communication for RMI/IIOP communication by default
  - Enable session security by default
  - Enable cookie protection via HttpOnly attribute to reduce cross-site vulnerabilities

- **Support for Java EE 6 security standards**
  - Servlet 3.0 security
    - HTTPServletRequest object
  - Basic security for EJB Embeddable container
  - Support for Java Authentication SPI for containers (JASPI)
  - Web Services Security API (WSS API) and WS-Trust support in JAX-WS to enable customers building single sign on Web services-based applications
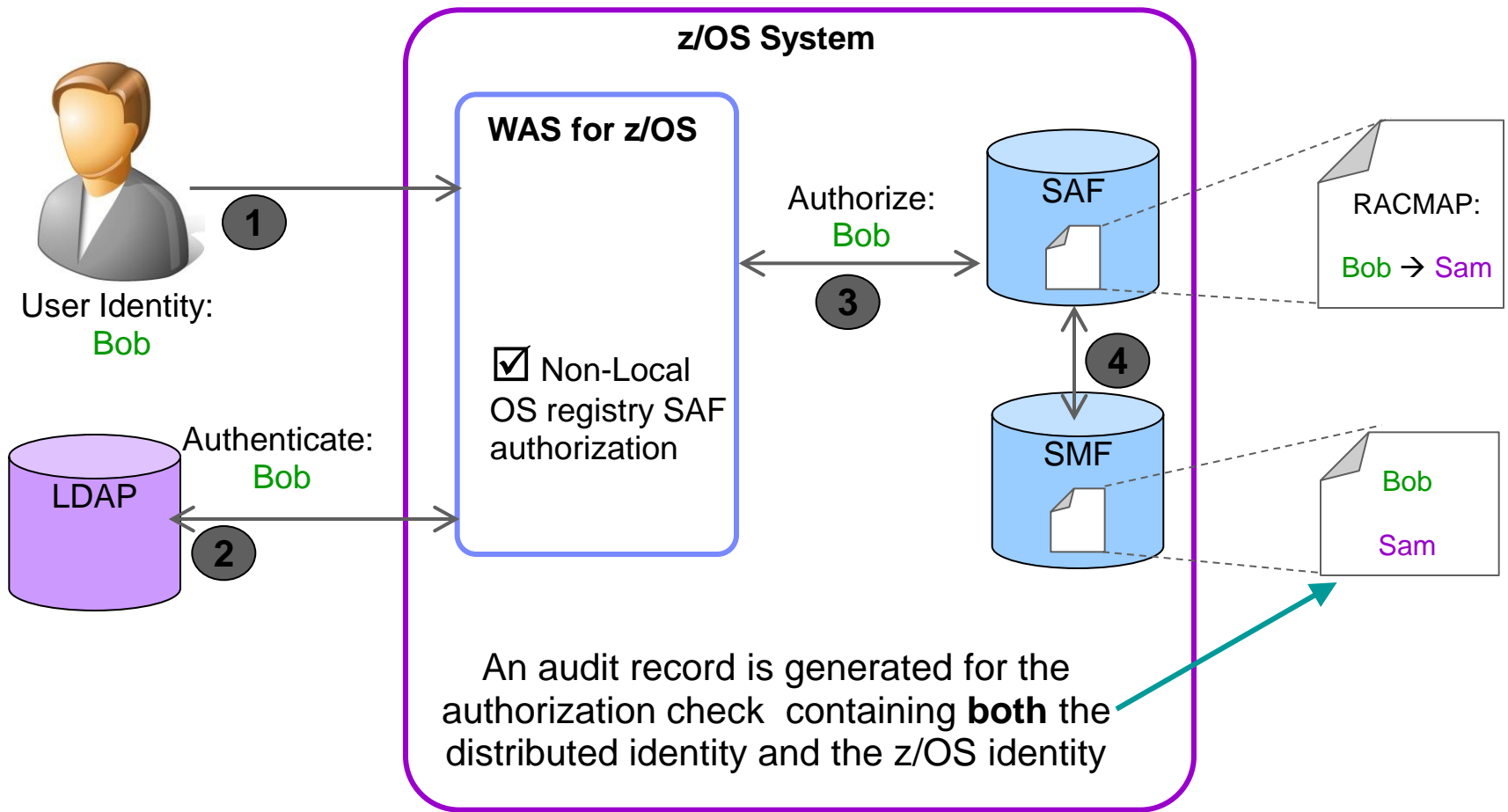 - Security enhancement for JAX-RS 1.1

V8.5 Security Enhancements

Ensure app server environment complies with OSGi security improvements and improve the ability to audit and track changes

- OSGi Blueprint security improvements:
  - Configure bean security in the Blueprint xml file
  - Configure bean-level security in OSGI apps
  - Configure method level security in OSGI apps

- Checkpoint Repository:
  - Audit and track any changes anybody makes to the WebSphere application Server configuration
    - Leverages Extended Repository from WVE

Distributed Identity Propagation for z/OS

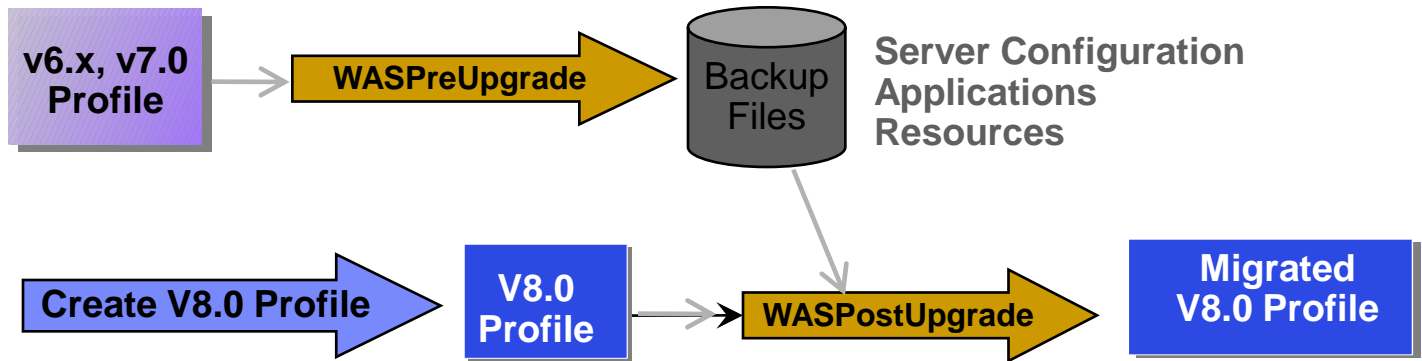Enhanced security and auditability for applications requiring distributed and z/OS system access

z/OS System

WAS for z/OS

User Identity:
Bob

☑ Non-Local OS registry SAF authorization

Authenticate:
Bob

LDAP

1

2

Authorize:
Bob

3

SAF

4

SMF

RACMAP:

Bob → Sam

Bob

Sam

An audit record is generated for the authorization check containing **both** the distributed identity and the z/OS identity

Configuration Migration Tooling

Migrate WebSphere environments faster with minimized risk

Assists administrators in moving their configuration when migrating

- Merges old configuration with new configuration

- Provides deep functionality, e.g. "Lights-on" WAS migration

- Especially useful for customers that have large topologies

  - Large telecom customer recently used the tool when migrating a 500+ JVM environment

Provides a framework for Stack product migration

- Already in use by Commerce, Portal, WPS and Virtual Enterprise

| v6.x, v7.0 Profile | → | **WASPreUpgrade** | → | Backup Files | **Server Configuration** **Applications** **Resources** |

| **Create V8.0 Profile** | → | **V8.0 Profile** | → | **WASPostUpgrade** | → | **Migrated V8.0 Profile** |

IBM Support Assistant Data Collector

> Resolve software issues as well as locate and collect key data in a timely manner

- Quickly collect diagnostic files or run traces that are predefined for WAS components
- View files and optionally send to IBM swiftly
- Shorten time to resolution

Customization

Automated data collections are specific to products and symptoms

Automated version of many MustGathers

Efficiency

Less time spent between a support analyst and the system operator

Reduces the effort to send the collected data to IBM support

Repeatability

Repeatable process with similar inputs without fear of human error

Simplicity

Executable by operators not familiar with the nature of the working product

V8.5 Cross Component Trace (XCT) for Problem Determination

Improve your ability to diagnose & debug SW problems in order to minimize and eliminate application downtime

- XCT log viewer - available for the IBM Support Assistant, can render log and trace content from multiple log / trace files grouped by request
  - View that detailed information on HTTP and JMS requests and responses to easily debug complicated application problems.
- High Performance Extensible Logging (HPEL) now has log / trace entry extensions
  - Filter entries by application name, by request ID or by other custom fields

Cross-Component Trace (XCT)

- XCT enables correlation of log and trace entries created by multiple threads and/or processes on behalf of the same request
  - XCT can augment log / trace entries with a requestID which you can view and filter using HPEL

```
[3/18/11 14:50:17:391 EDT] 00000018  W UOW= source=com.ibm.somelogger.QuickLogTest org= prod= component=
thread=[WebContainer : 1] requestID=AAP+k9s6JZ9-AAAAAAAAAA
            hello world
```

  - XCT can add records to your log / trace files so you can see how work related to each request branched between all involved threads / processes.

```
[3/23/12 14:01:40:615 CDT] 00000032 XCT              I   BEGIN AAP+k9s6JZ9-AAAAAAAAAA 00000000000-cccccccccc2
HTTPCF(InboundRequest /HelloWorld/ RequestContext(828937987))
[3/23/12 14:01:40:678 CDT] 00000032 XCT              I   END   AAP+k9s6JZ9-AAAAAAAAAA 00000000000-cccccccccc2
HTTPCF(InboundRequest RC=200 RequestContext(828937987))
[3/23/12 14:01:50:381 CDT] 00000032 XCT              I   BEGIN AAP+k9s6JZ9-AAAAAAAAAB 00000000000-cccccccccc2
HTTPCF(InboundRequest /HelloWorld/ RequestContext(435283455))
[3/23/12 14:01:50:443 CDT] 00000032 XCT              I   END   AAP+k9s6JZ9-AAAAAAAAAB 00000000000-cccccccccc2
HTTPCF(InboundRequest RC=200 RequestContext(435283455))
```

  - XCT can store data snapshots, for example the complete text of a JMS request, to facilitate in-depth problem determination

```
[1/17/12 16:58:46:765 EST] 0000001c XCT              I   BEGIN AAP+k9s6JZ9-AAAAAAAAAB 00000000000-cccccccccc2
MyAnnotation(Attachment(MyRequest.txt))
```

V8.0 High Performance Extensible Logging (HPEL)

Improve performance & ease of use of log & trace to improve problem determination

Key Features:

- Speeds up logging and tracing
  - Log primitive over 6x faster than WAS v7
  - Trace primitive 3.8x faster than WAS v7
- Provides more flexible access to log and trace data
  - Command-line access to filter and format
  - Administrative console GUI to filter and format local or remote logs and trace, even when the remote server is down
  - Programmatic access to filter, format, and merge local or remote logs and trace
- Works with existing application log and trace instrumentation
- Provides a common solution for z/OS and distributed platforms
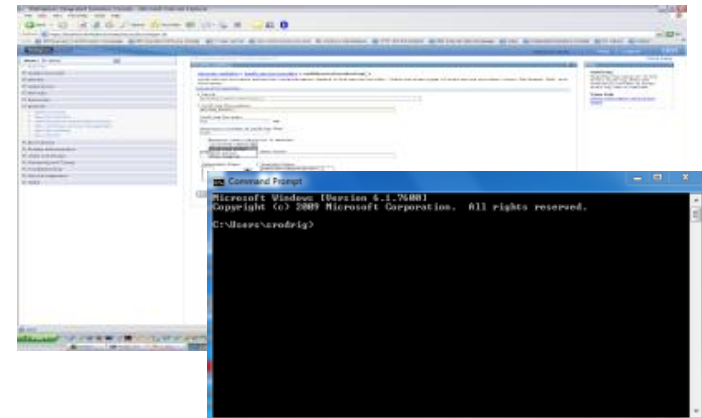
Examples:

View only warning and higher msgs for this one application:
   logViewer.sh -minLevel warning -includeLoggers "com.acme.app1.*"

View msgs from 07/11/2010 onward beginning with SEC on thread 0x0000000c:
   logViewer.sh -startDate 07/11/2010 -message "SEC*" -thread 0c

V8.5 HPEL Extensions and Improvements

- HPEL log / trace entries can now be extended with name value pair 'extensions'
  - JEE application name has been added as an extension (called 'appName') to all log / trace entries created on threads associated with an application
  - XCT requestID has been added as an extension (called 'requestID') to all log/trace entries known to be created on threads associated with a request

- When viewing HPEL log / trace, entries can be filtered by appName, requestID, or any other extension via the HPEL logViewer command

```
logViewer.sh -includeExtensions appName=ACMEShovels –format advanced
…
[12/10/11 10:52:01:500 EST] 000001c6  1 UOW= source=com.acme.SomeLogger thread=[WebContainer : 6] org= prod= component= appName=[ACMEShovels]
      This is a trace entry from the MyShovels application

. . .
```

- Developers can add their own extensions to HPEL log / trace entries via the new **LogRecordContext** API

Copyright and Trademarks

© IBM Corporation 2012. All Rights Reserved.

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Backup

Maximize the Value of your WebSphere Investment

The technology you need, when you need it
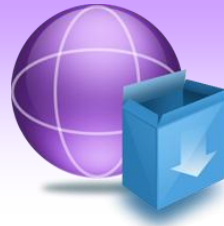
Access to WebSphere Feature Packs

Award-winning IBM Support Portal

Renewing your IBM Software Subscription and Support is the best way to ensure you get continuous and maximum value

▶ Protect your investment

▶ Leap ahead of competitors

▶ Get up and running faster

Download the latest enhancements of entitled WebSphere software

Enhance your business. Renew your software subscription today.

Flexible Delivery & Pricing Models for WebSphere Software

Extending Value and Function

- No-cost WAS for Developers
- No-cost WAS Feature Packs
- Rational Application Developer for WebSphere Standard Edition
- No-cost Version to Version Application Migration Tool

- No-cost Open Source WAS-CE with optional support
  - Socket based pricing for optional support
- Socket based pricing for WAS Base
- Simple Load Balancing (Web-tier) in WAS Base
- Flexible licensing

Traditional

- WebSphere Application Server
- WebSphere Virtual Enterprise
- WebSphere eXtreme Scale
- Sub-capacity licensing

Private clouds

- IBM Workload Deployer
  - WAS Hypervisor Editions
  - WebApp Workload Pattern
- Intelligent Management Pack
- Sub-capacity licensing

Public Clouds

- WAS on IBM Public Cloud
- WAS Amazon Machine Image (AMI)
- Pay per SW use or bring your SW
- Bring your license (on IBM Cloud)
- IBM Sandbox

IBM Cloud

amazon web services™

WebSphere Application Server Feature Packs

Access innovative standards and programming models faster on a stable foundation

With WAS V6.1 & 7.0

EJB 3.0	Web 2.0	Web Services

J2EE 1.4   WAS V6.1

→

| Web 2.0 | CEA | XML | SCA | OSGi Apps & JPA 2.0 | Java Batch |

| EJB 3.0 | Web Services | Java EE 5   WAS V7 |

With WAS V8

Web 2.0 & Mobile   Dynamic Scripting

| CEA | XML | SCA | OSGi Apps & JPA 2.0 | Java Batch | Java EE 6   **WAS V8** |
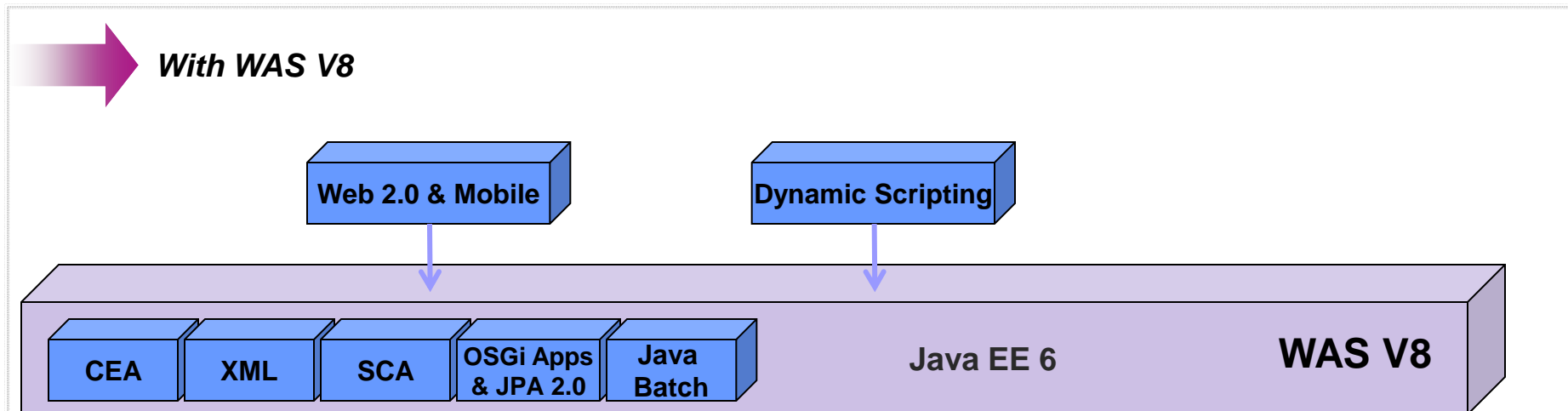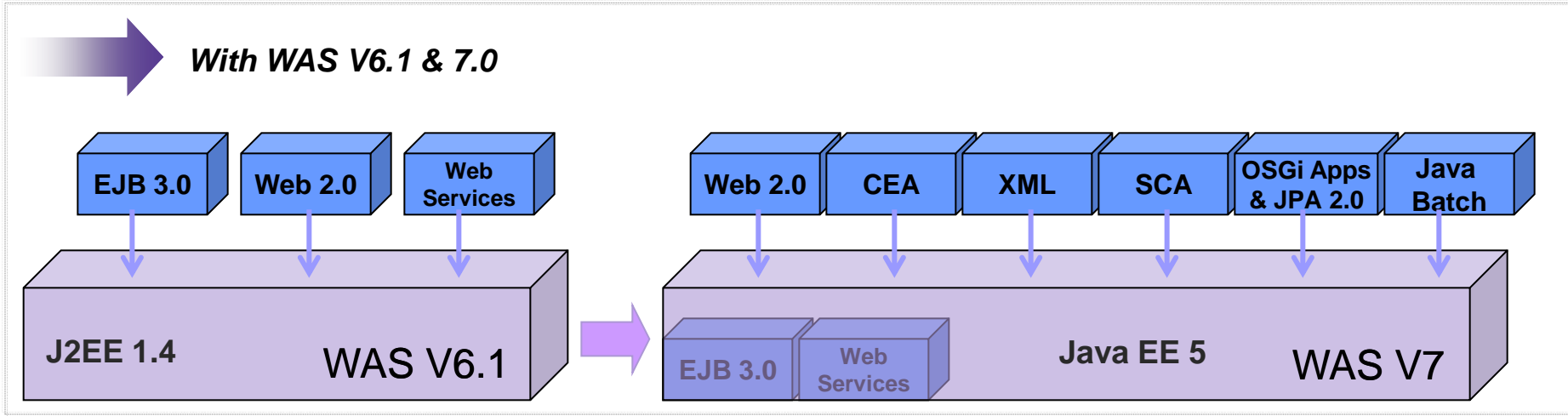
WebSphere Application Server support for OSGi projects

		Supported in WebSphere Application Server version 7.0	**Supported in WebSphere Application Server version 8.0**
OSGi Applications containing:	OSGi Bundles	✓	✓
	OSGi Fragments	✓	✓
	OSGi Composite bundles		✓
	Java EE WAR Modules	✓	✓
	PDE plug-ins	✓	✓
	PDE fragments	✓	✓
OSGi Composite bundles containing:	OSGi Bundles		✓
	OSGi Fragments		✓
	PDE plug-ins		✓
	PDE fragments		✓

WebSphere Business Level Applications (BLA)

> Simplify admin tasks and management of multi-component applications

- A composition model that extends the notion of "Application"
  - Manages JEE and non-JEE artifacts like SCA packages, libraries, proxy filters etc.
  - Performs dependency management by tracking relationships between application components
  - Supports Application Service Provider (ASP) scenarios by allowing single application binaries to be shared between multiple deployments

- Supports full lifecycle management of business level applications
  - create, start, stop, edit, delete

- Aligns WebSphere Applications better with business as opposed to IT configuration

Composition

BLA3   BLA1   BLA2

Configuration

EJB Module   Web Module   J2EE Enterprise App   Java Lib   Java Lib   J2EE Enterprise App

Business Logic

JAR   WAR   EAR   JAR   EAR
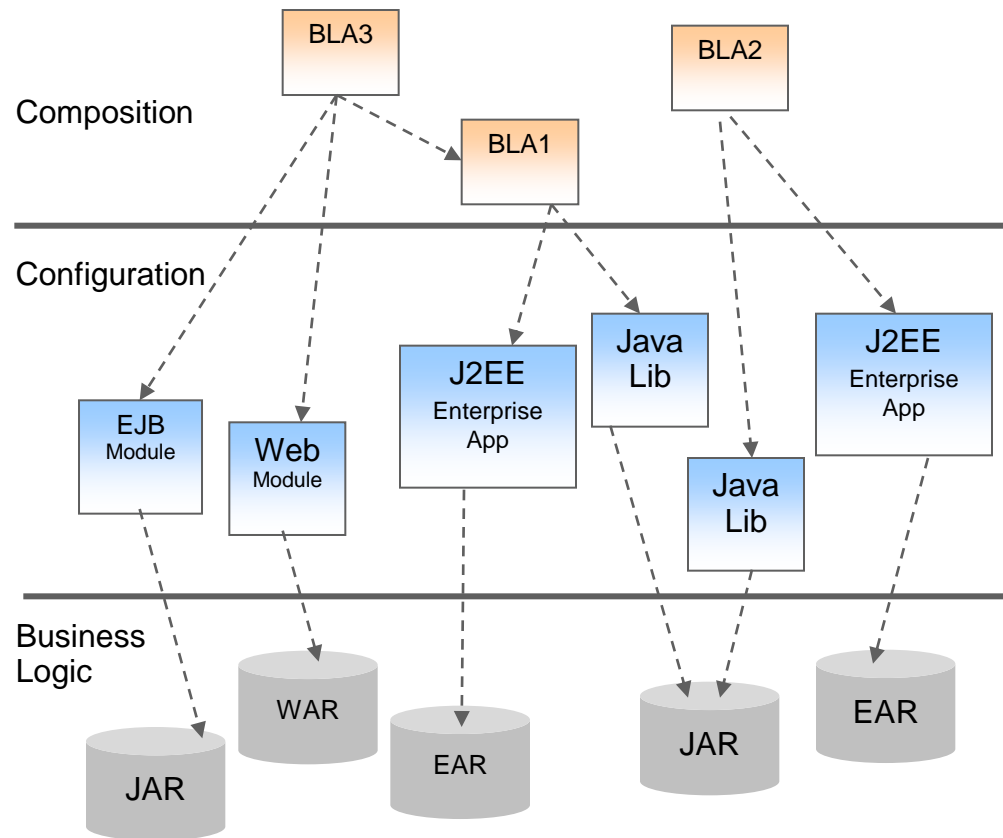
Multiple Security Domains

Separate applications, users and infrastructure to increase flexibility and control

Applications can have their own application security domain. Own user population

Cluster

Applications in a cluster share a common application security domain.
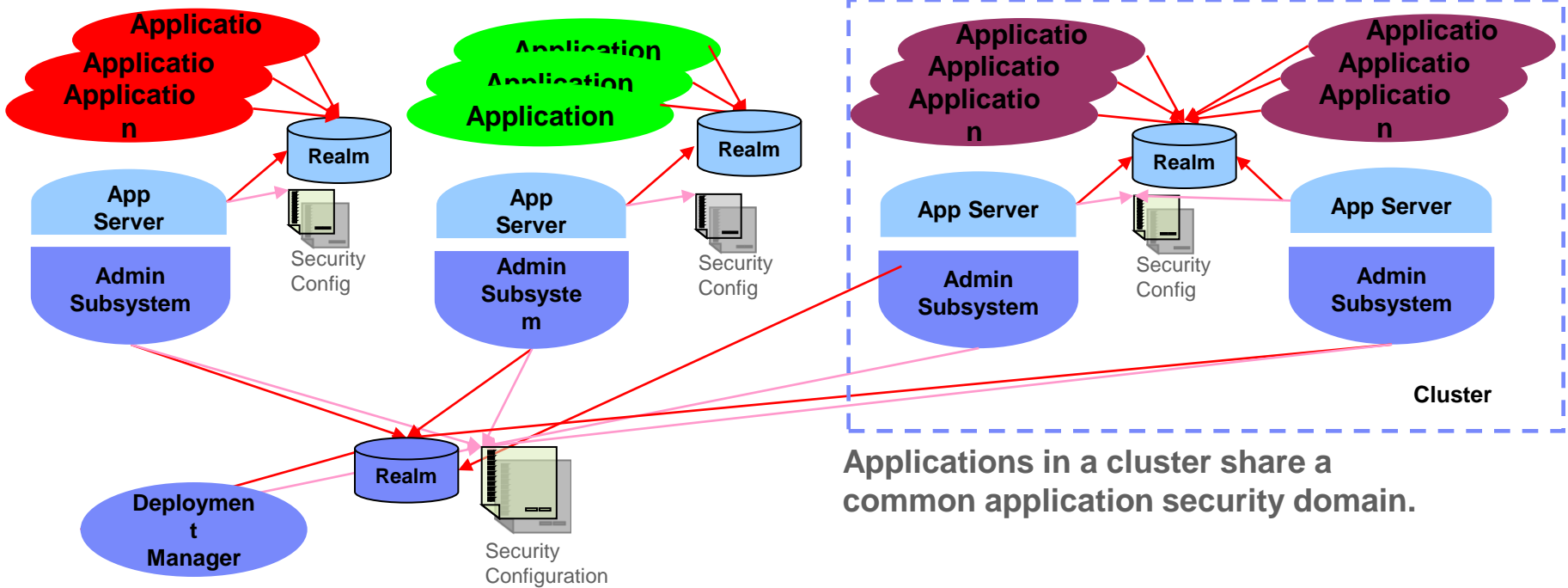
Deployment Manager, Node Agent, and the Admin Subsystem common administrative security domain.

- Multiple security domains provide flexible security configuration under centralized management
- Ability to separate User security domain from administrative security domain

![IBM]

Fine-grained Administrative Security

Isolate administrators from each other and according to access levels to improve security and governance

Key Features:

- Users can be defined with administrative roles on specific resources:
  - Cells, node groups, nodes, clusters, servers, and applications

- Administrative Console will be filtered by user's administrative role

- User cannot access any other resources outside assigned resources

Create Cells from a Template

Improve administrator productivity and repeatability and minimize errors
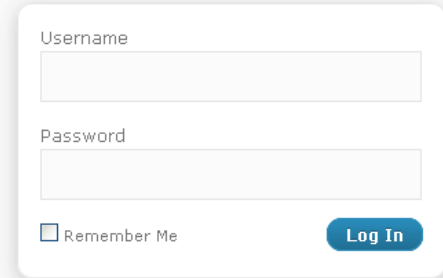
- Automate and improve repeatability of deploying consistent WebSphere Application Server environments

/dmgrNode /node1 → backupConfig → template.zip

restoreConfig

dmgr

dmgr

dmgr

addNode

addNode

addNode
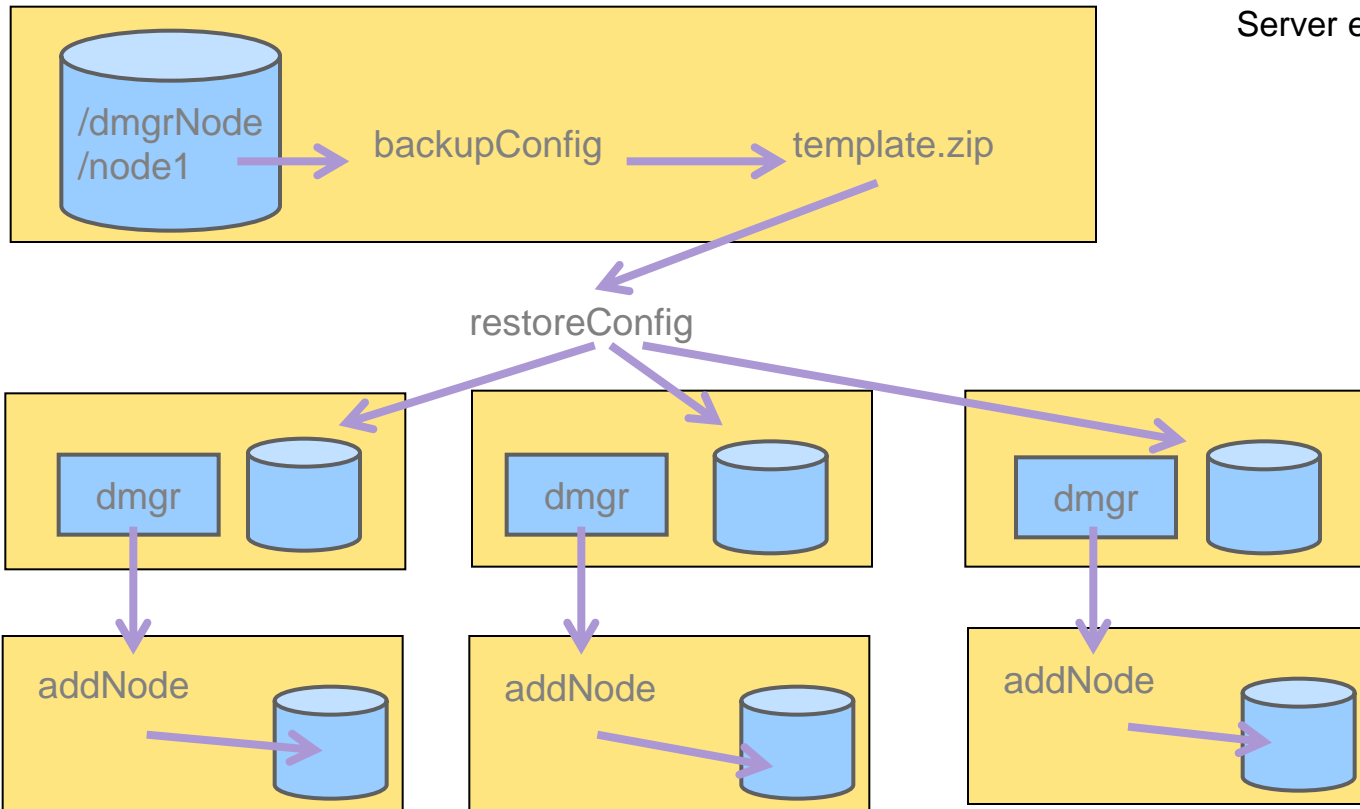
Single Sign On Improvements

Improve end user ease of use while maintaining security controls

- Enhanced security in SSO web applications to reduce cross-site scripting vulnerabilities

- Support for using SAML Token through WS-Security SAML Token Profile 1.1

- Generate SAML tokens, request SAML tokens from an external Security Token Service (STS) & propagate SAML tokens in SOAP messages using the Web Services Security application programming interfaces (WSS API)

- Generate and consume tokens using WS-Trust Issue and WS-Trust Validate requests for JAX-WS Web services that use Web Services Security
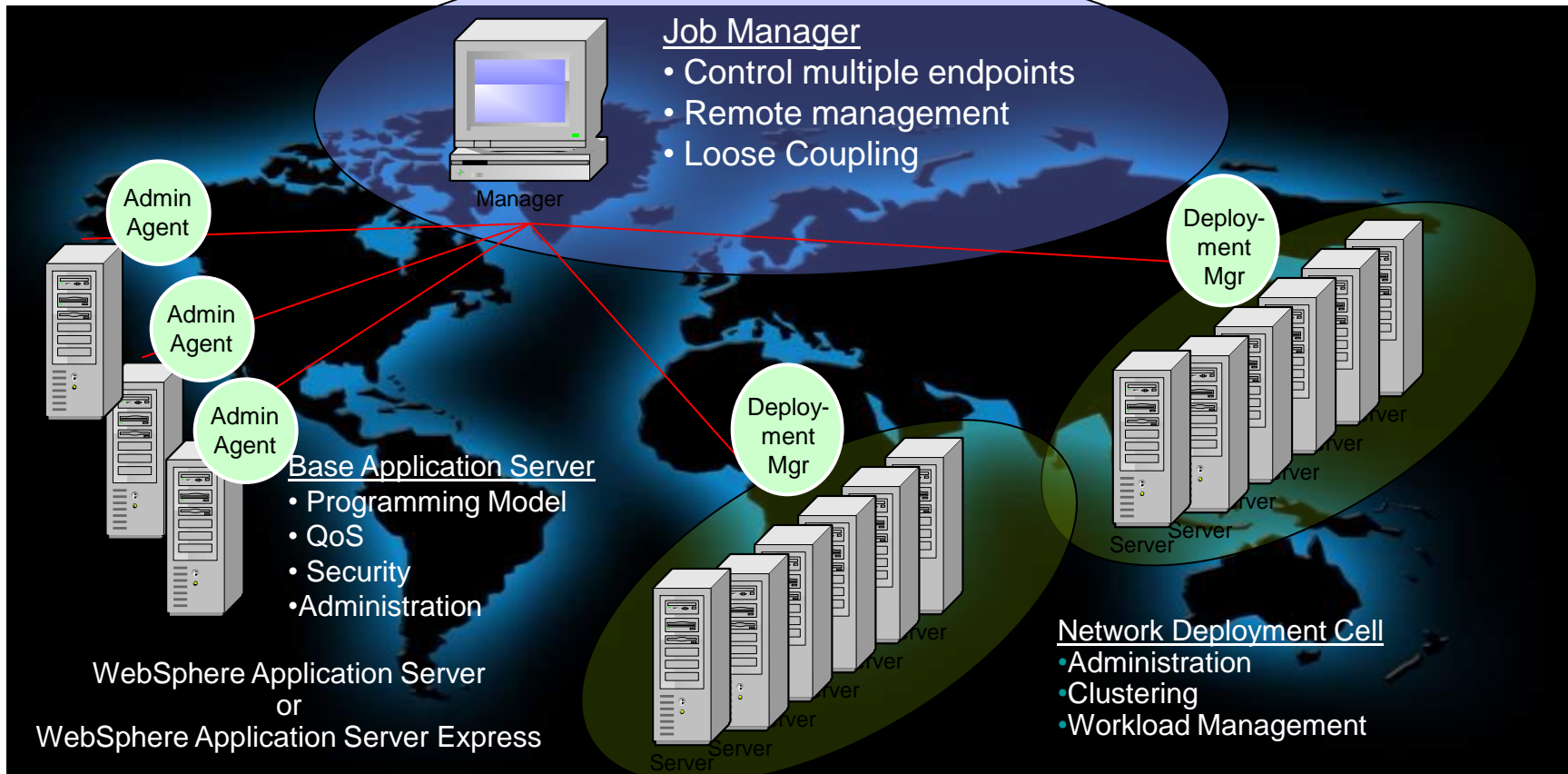
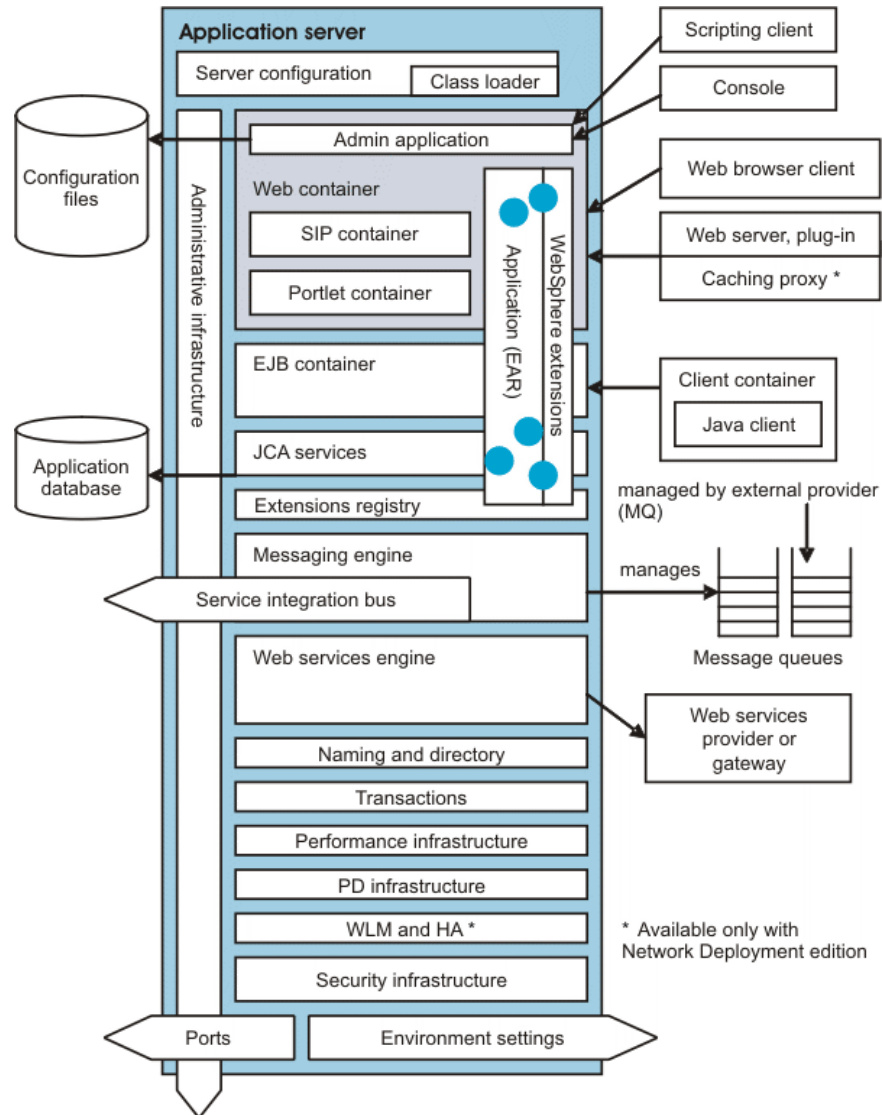Username

Password

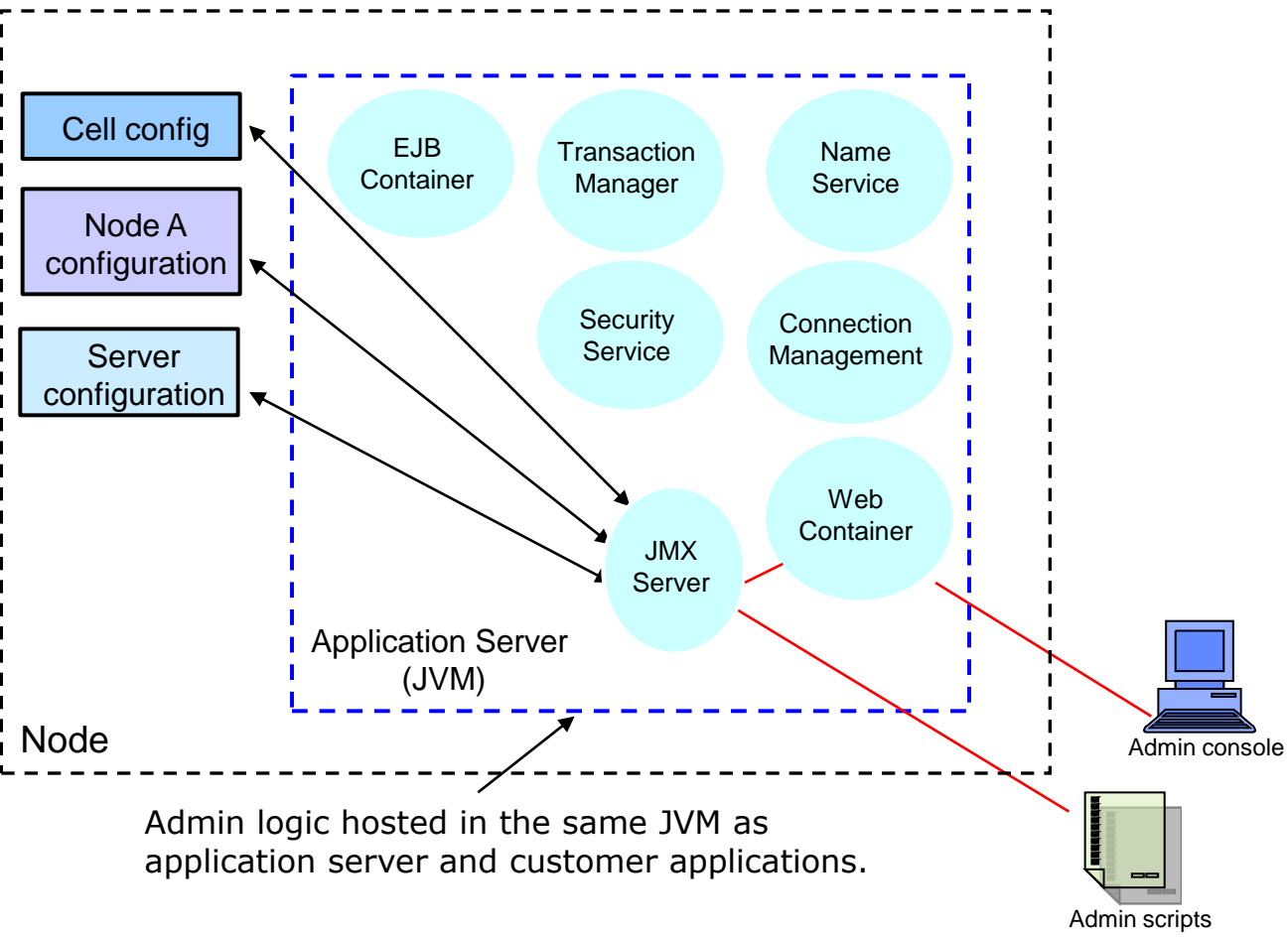☐ Remember Me   Log In

Lost your password?

![IBM]

Flexible Management

Utilize a flexible, scalable and asynchronous administrative topology for highly productive global administration and management

Job Manager
• Control multiple endpoints
• Remote management
• Loose Coupling

Manager

Admin Agent

Admin Agent

Admin Agent

Base Application Server
• Programming Model
• QoS
• Security
•Administration

WebSphere Application Server
or
WebSphere Application Server Express

Deploy-
ment
Mgr

Deploy-
ment
Mgr

Network Deployment Cell
•Administration
•Clustering
•Workload Management

WebSphere Application Server Components

WebSphere Application Server Architecture

Cell config

Node A configuration

Server configuration

EJB Container

Transaction Manager

Name Service

Security Service

Connection Management

JMX Server

Web Container

Application Server (JVM)

Node

Admin console

Admin scripts

Admin logic hosted in the same JVM as application server and customer applications.

WebSphere Application Server Network Deployment Architecture
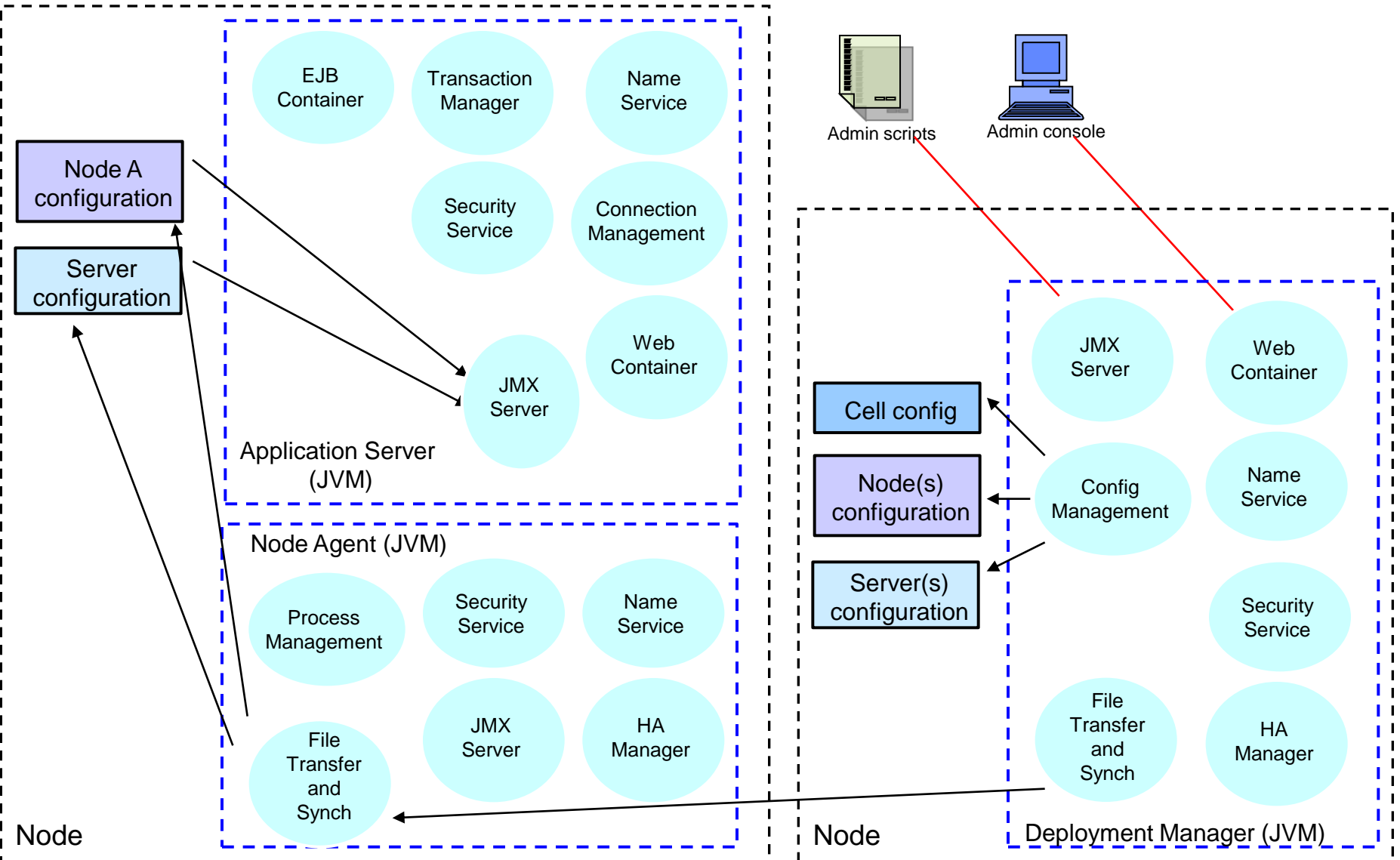
WebSphere Application Server Vocabulary

- **Application Server**
  - JVM hosting applications
    - Hosts administration in WAS

- **Node Agent**
  - WAS-ND JVM Providing Node Configuration and Process Management

- **Deployment Manager**
  - WAS-ND JVM Providing Centralized Administration and Monitoring

- **WAS Cell**
  - All Nodes Managed by Single Administrative Process
    - Multiple Nodes in WAS-ND Managed by Deployment Manger
    - Single Node/Application Server in WAS **

- **WAS-ND Cluster**
  - Application Servers running same application(s)
    - Provides Request Distribution and Scalability

- **WAS-ND Core Group**
  - HA Domain
    - One or more per cell

** Admin Agent optional in WAS