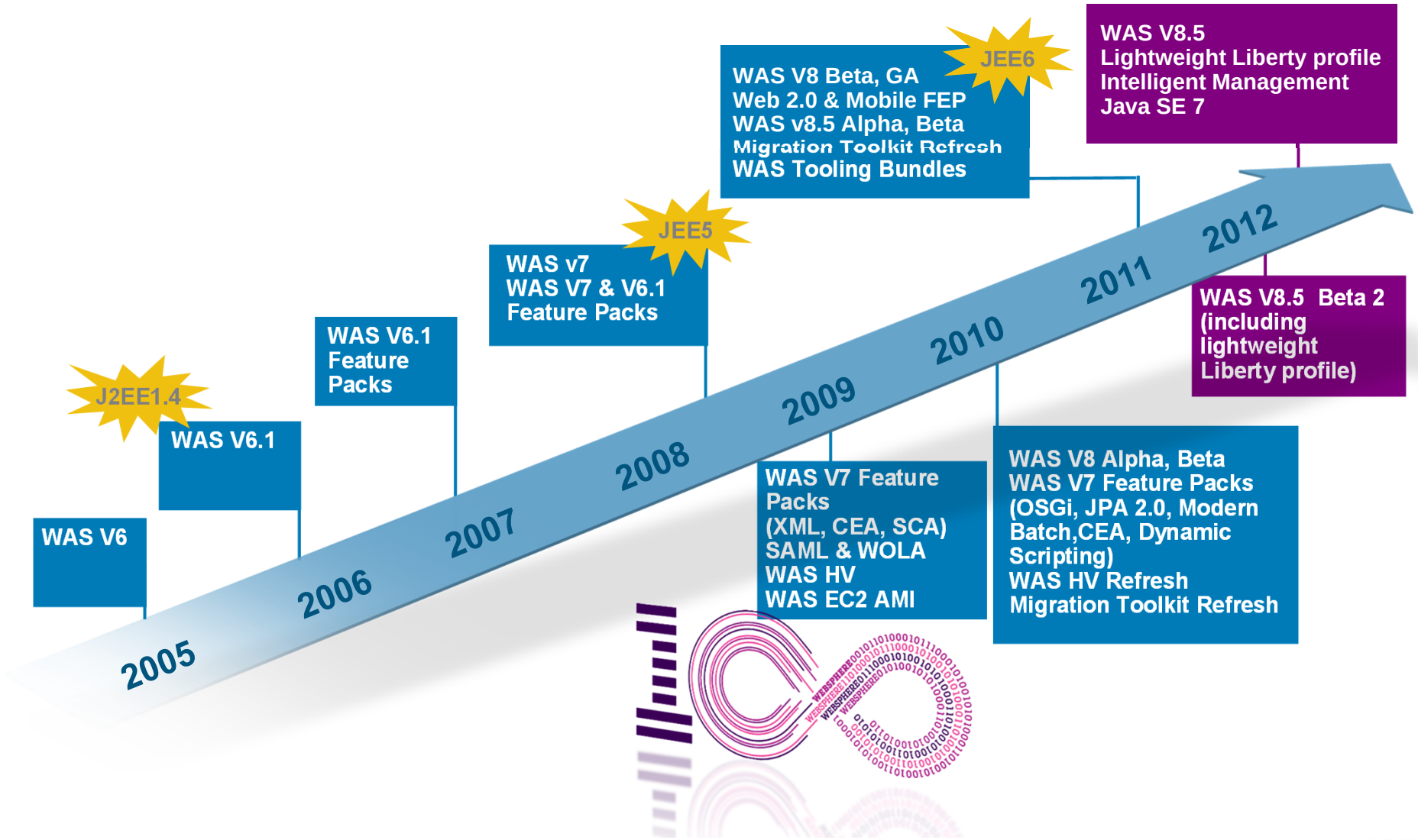
WebSphere Foundation Update and Direction

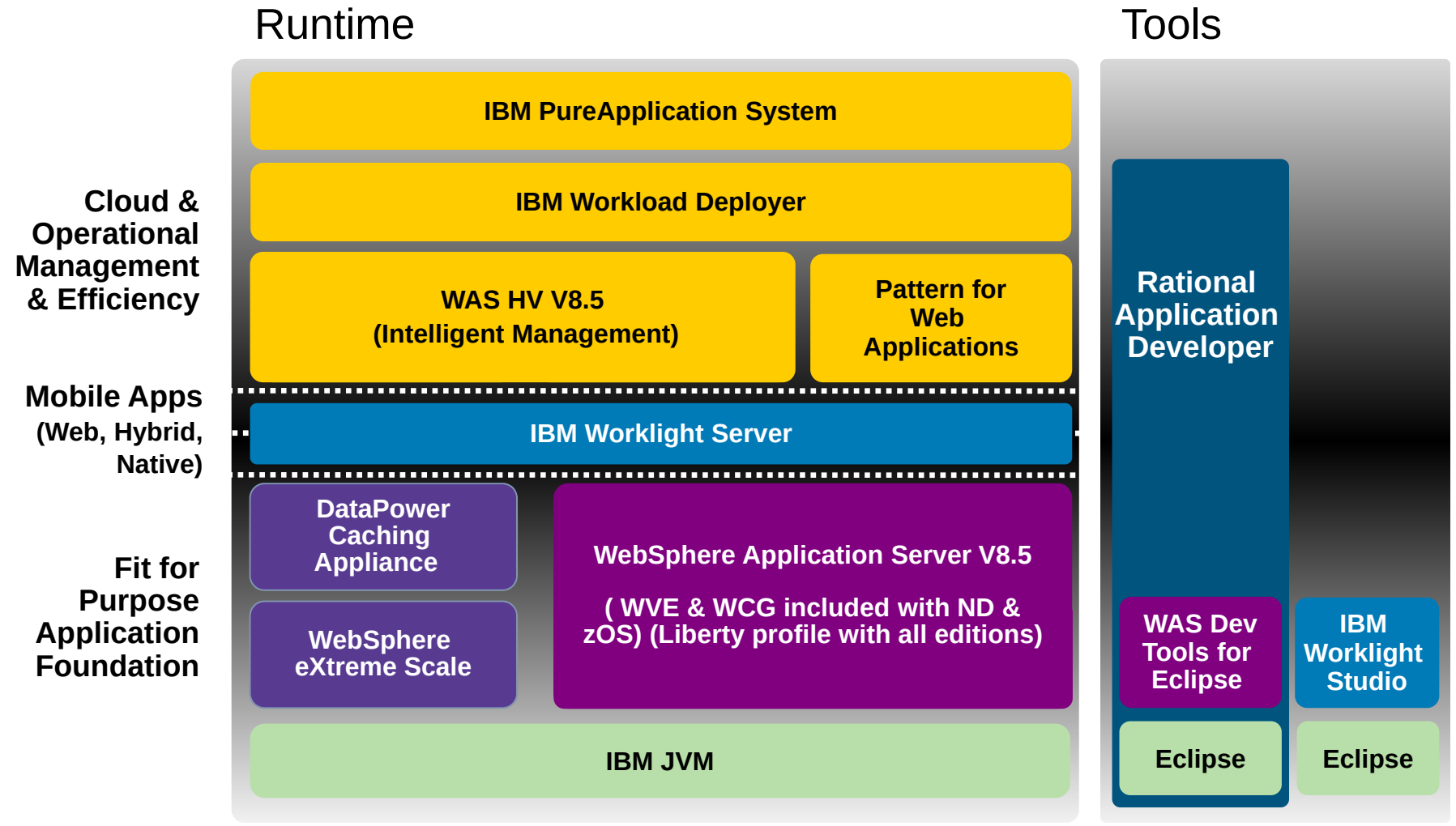
Ian Robinson, IBM Distinguished Engineer

WebSphere Foundation Chief Architect

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



WebSphere Application Infrastructure
Changes in V8.5



WAS V8.5 Delivers

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

Developer Experience

Fast, flexible, and simplified application development

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

Application Resiliency

Intelligent Management & Enhanced Resiliency

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

Operations and Control

Improved Operations, Security, Control & Integration

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

WebSphere Application Server Family

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

Optimized to instantly run in VMware and other server virtualization environments

WAS ND

Tools Edition | +Liberty Profile | +Intelligent Mgmt

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

WAS for z/OS

+Liberty Profile | +Intelligent Mgmt

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

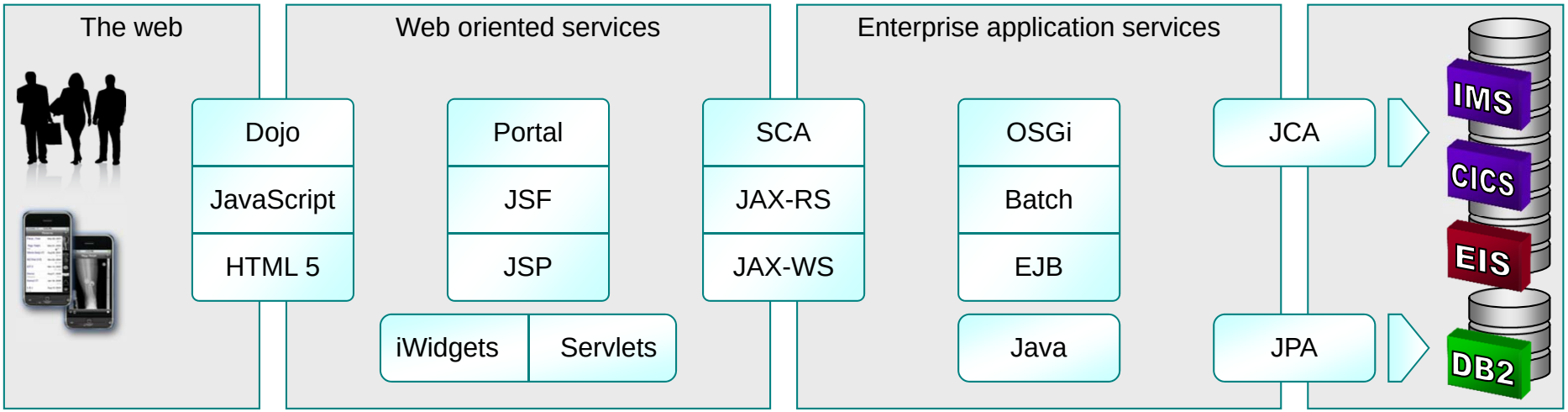
Built on a common code base

Enterprise Application Development with RAD

For enterprise developers needing advanced end to end capabilities to deliver robust, scalable applications integrating modern & legacy systems

Create & edit | Visualize | Validate | Refine & refactor | Analyze | Publish | Test | Debug | Profile

***Integrate** the enterprise to the social web using state of the art technology*

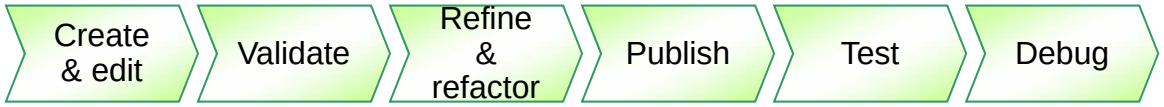


***Collaborate** with agility, exploiting modern infrastructure and the Cloud*

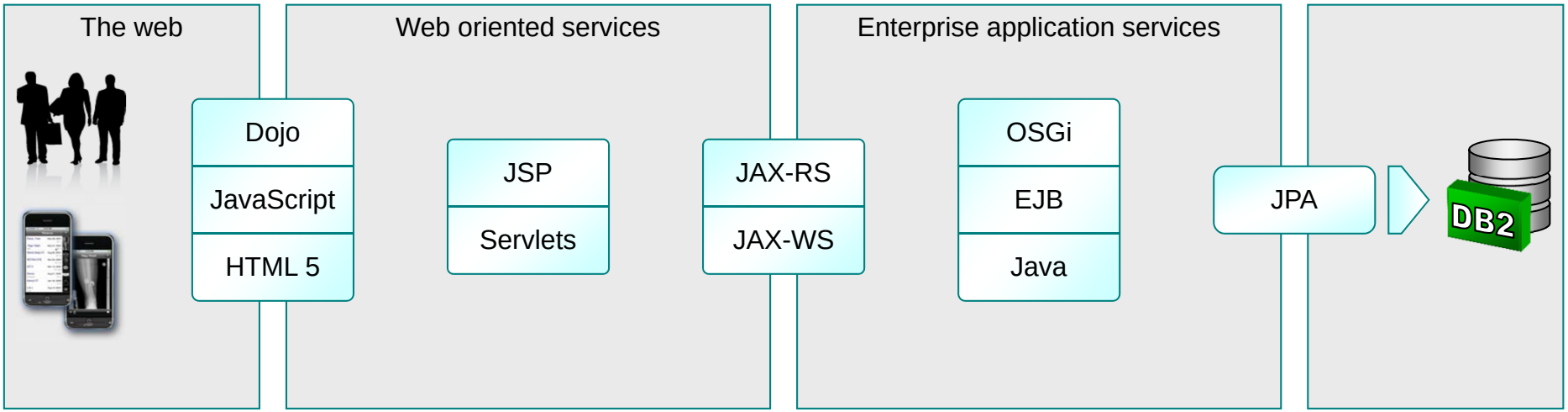
Collaborative Lifecycle Management | WebSphere foundation | Public and Private Clouds

Unleash innovation with WDT 8.5

For developers that need core capabilities to develop simple web applications with modern services.



Design web, Java EE and OSGi applications with persistence

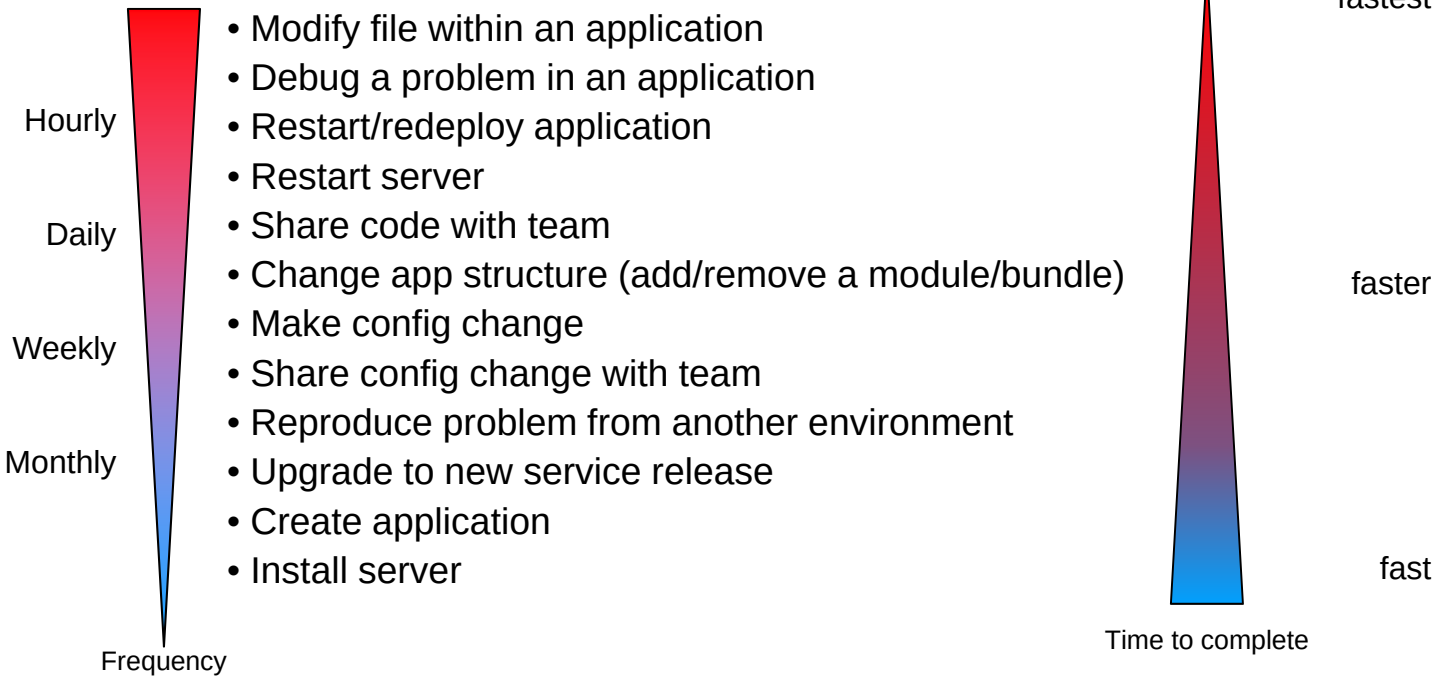


Made for WebSphere Application Server and the Liberty profile
Available from the Eclipse Marketplace

WebSphere foundation

What Do Developers Care About?

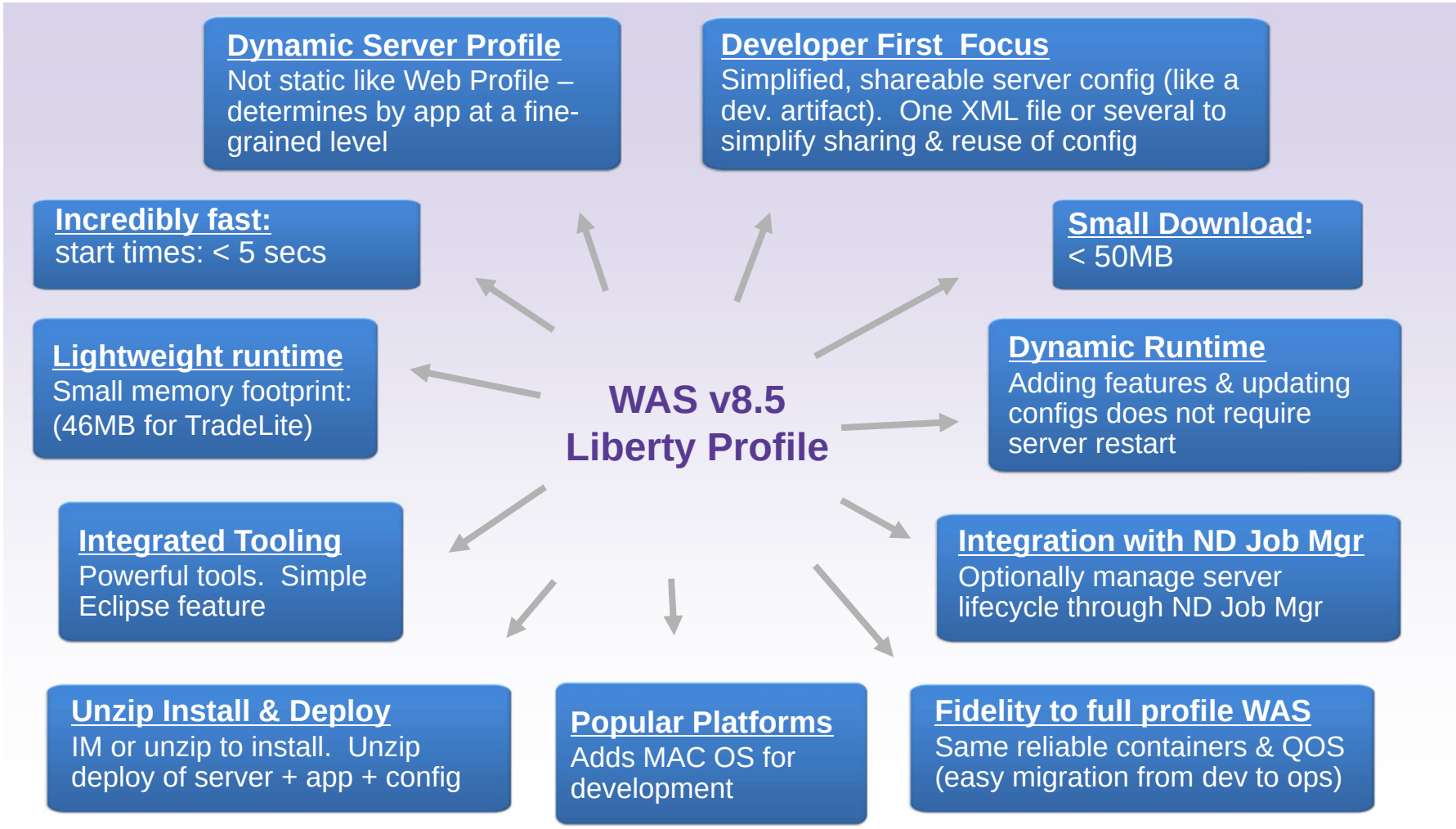
Common development tasks include:



- 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

WAS v8.5: Introduces the Lightweight "Liberty" Profile – For Web, OSGi and Mobile Apps

A highly composable, dynamic Server profile

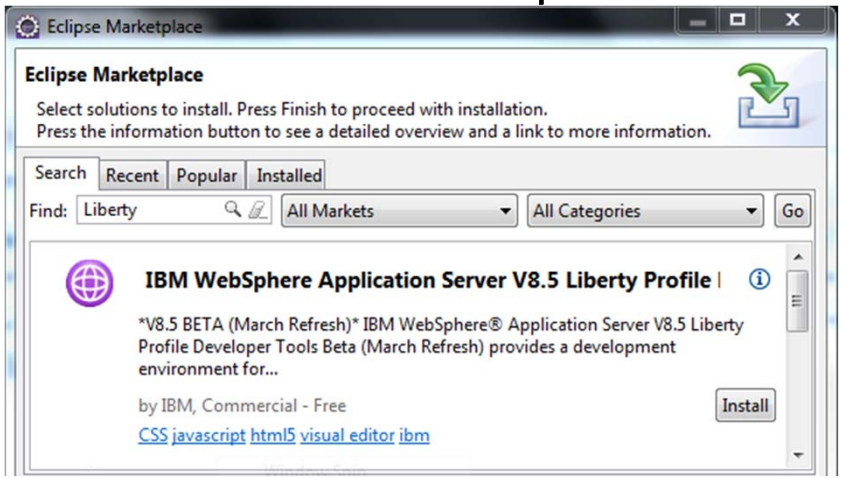


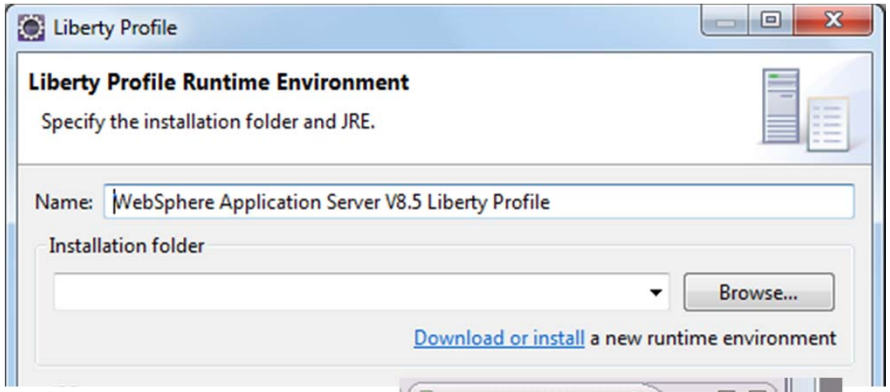
No Hurdles to Install

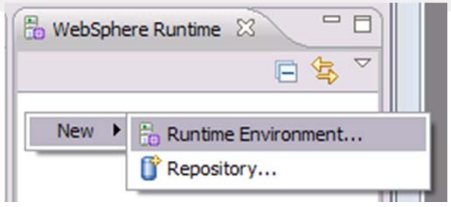
- Tools and runtime are **no-charge** for development. No time limit
- Eclipse feature install for tools; 40MB zip download for server profile.
  - Installation Manager also supported → same installed result.

2 minutes from "Nothing" to "Done":

1. Install WAS Developer Tools for Eclipse Feature



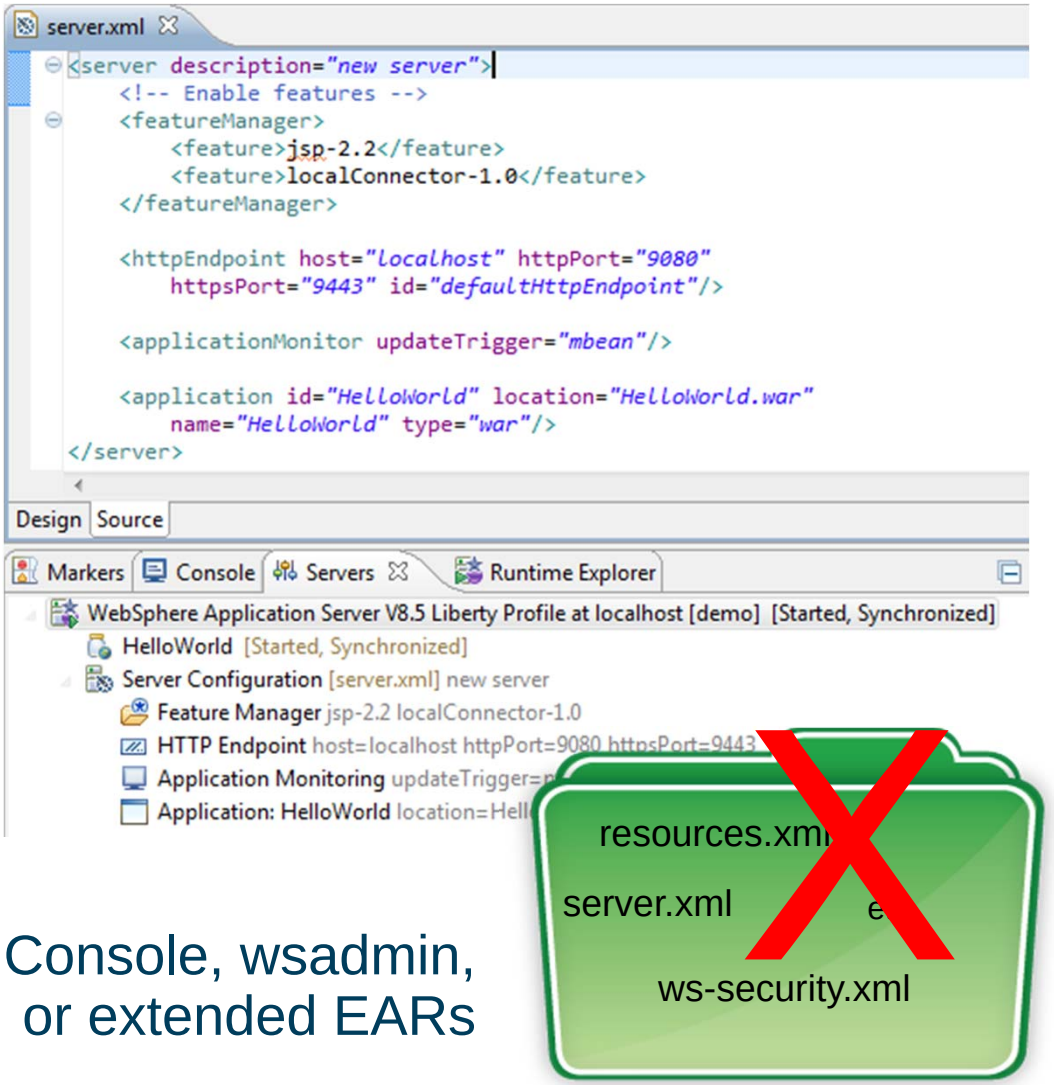




2. Use the Tools to download the WAS Liberty Profile or download 40MB zip from WASdev.net

Simplified Server Configuration

- Simplest case: One XML file for all server config
- Editable within the workspace
- Exportable, shareable, versionable



No need for Admin Console, wsadmin, or extended EARs

Flexible Configuration

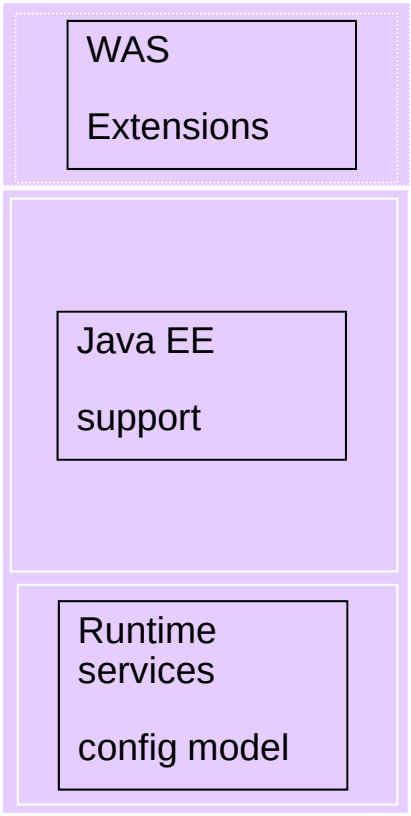
- Shareable config snippets

```
<server>
    ...
    <include location="http://cfgserver/global.xml" />
    <include location="${shared.config.dir}/datasource.xml" />
<server>
```

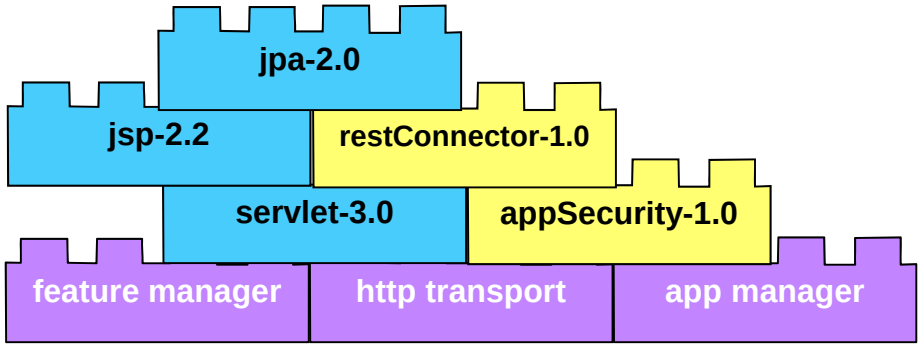
server.xml

- Config can be componentized at any level of granularity, from 1 file to many.
  - Can use WDT to associate config snippets with a server config.
- Visualization through WDT tools as a single logical view.
- Team development: keep the application and configuration components together.

Highly composable runtime based on ‘features’

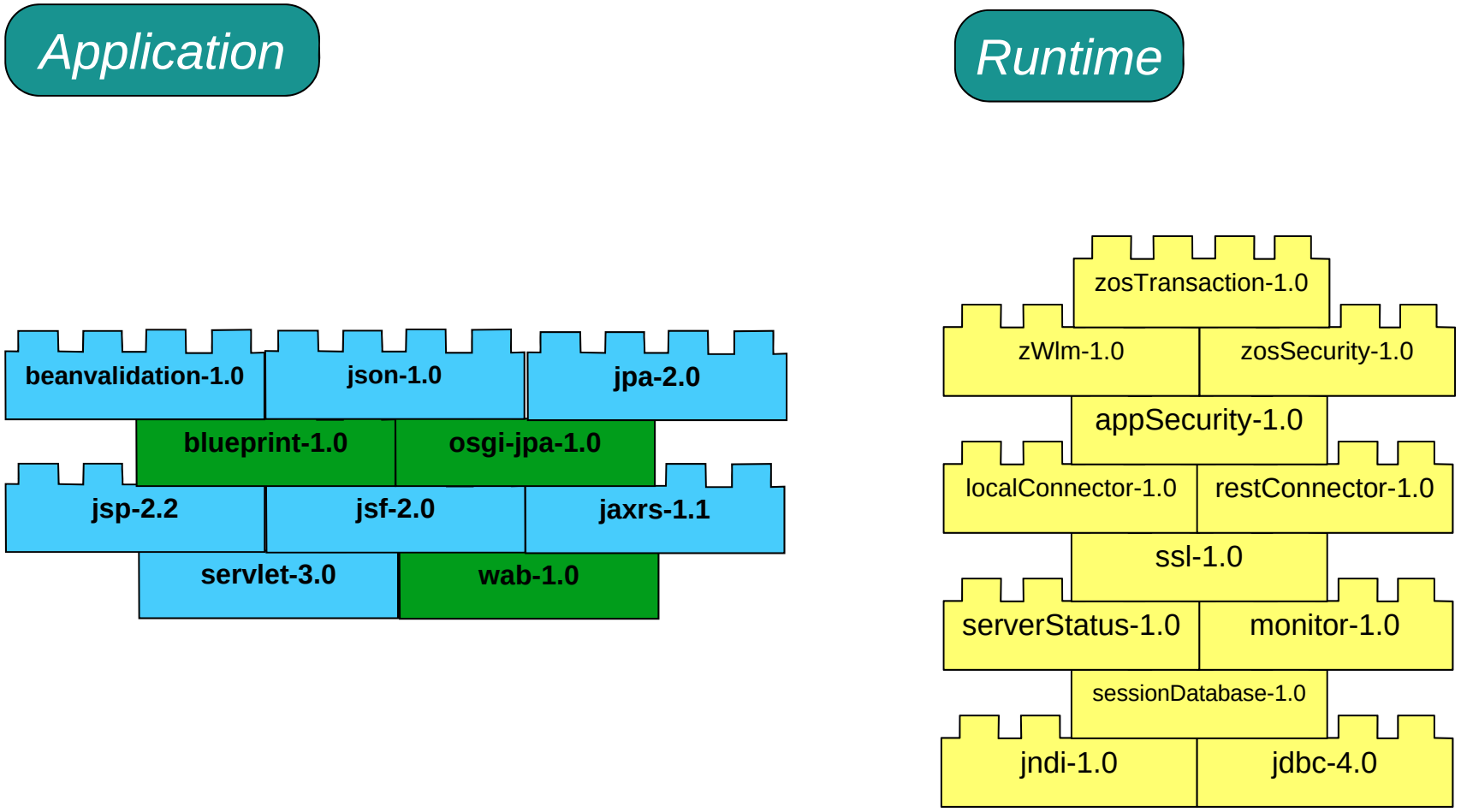


Full WAS Profile



WAS v8.5
Liberty Profile

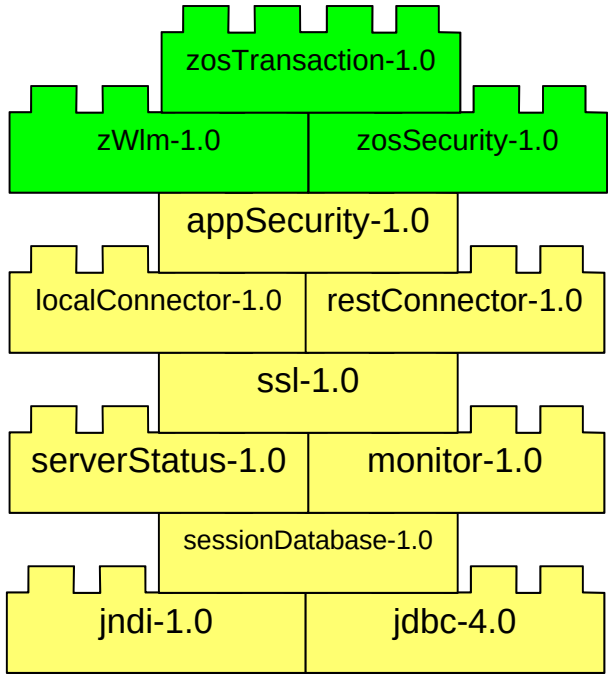
Dynamic enablement of feature set in application



WebSphere for z/OS – Liberty Profile

The **Liberty Profile** introduces patterns of usage that **accelerate application development and deployment**, while leveraging **z/OS qualities of service**

- The Liberty profile with *optional*, independently enabled extensions that exploit z/OS facilities
  - z/OS transaction management, workload management, and security
- Simplified and reduced configuration for both WebSphere and z/OS
- Fast startup and small footprint
- Extremely lightweight single process runtime
- Ability to package and distribute applications and associated configuration
- Easily test applications in a z/OS environment



Libraries

- Associated with applications
- Move common libraries out of your wars

```
<library id=”libs”>
   <fileset dir="${shared.resource.dir}/libs"
            includes="*.jar"/>
</library>
```

- Share classes between apps

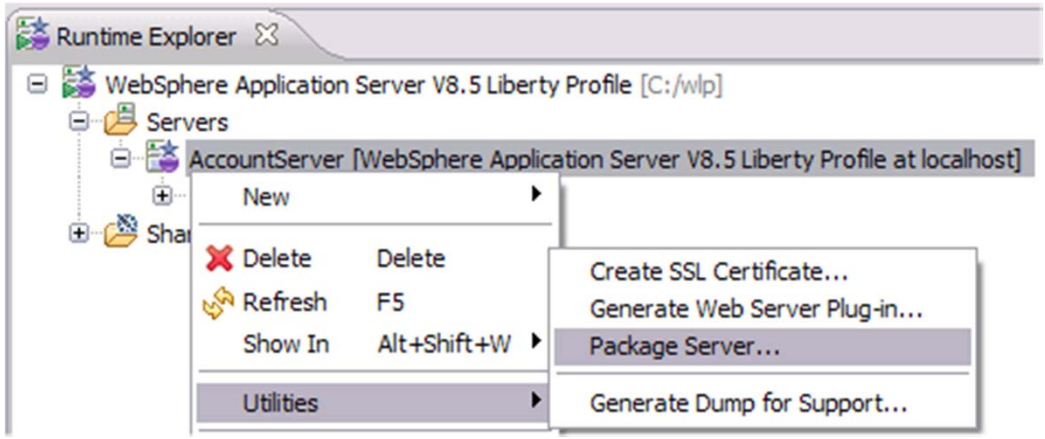
```
<application location="snoop.war">
    <classloader commonLibraryRef="libs" />
</application>
```

- Or have an instance per app

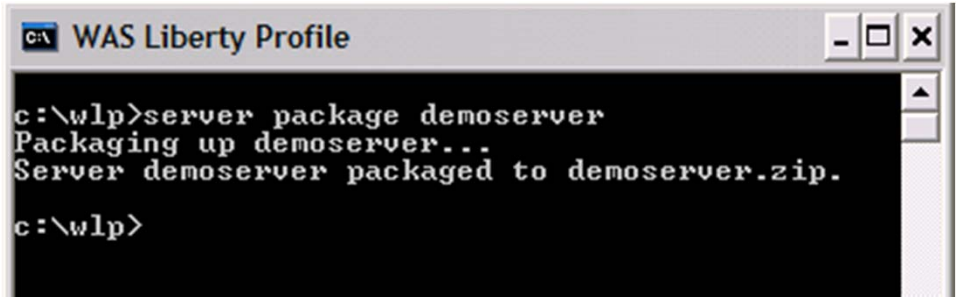
```
<application location="snoop.war">
    <classloader privateLibraryRef="libs" />
</application>
```

New Deployment Options

- Package up a compressed archive of a configured Liberty server type along with its applications
  - Directly from Eclipse environment
  - Resulting zip can be copied to integration or production environment and unzipped.



- For test automation outside the IDE, a command-line program to manage the lifecycle of server instances:
  - Create [serverName]
  - Start and stop [serverName]
  - Package [serverName]
  - Status [serverName]



- Updates to configuration of running server are effective immediately.
- Add/remove apps dynamically by drag/drop to monitored directory.

Creating a Production Image

Installation Manager,
Or zip download

Install

WLP

SDK

Manually,
Eclipse,
Or RAD

add

server

app

Manually,
Eclipse,
or RAD

Repackage everything,
or subset needed by server

WLP

SDK

server

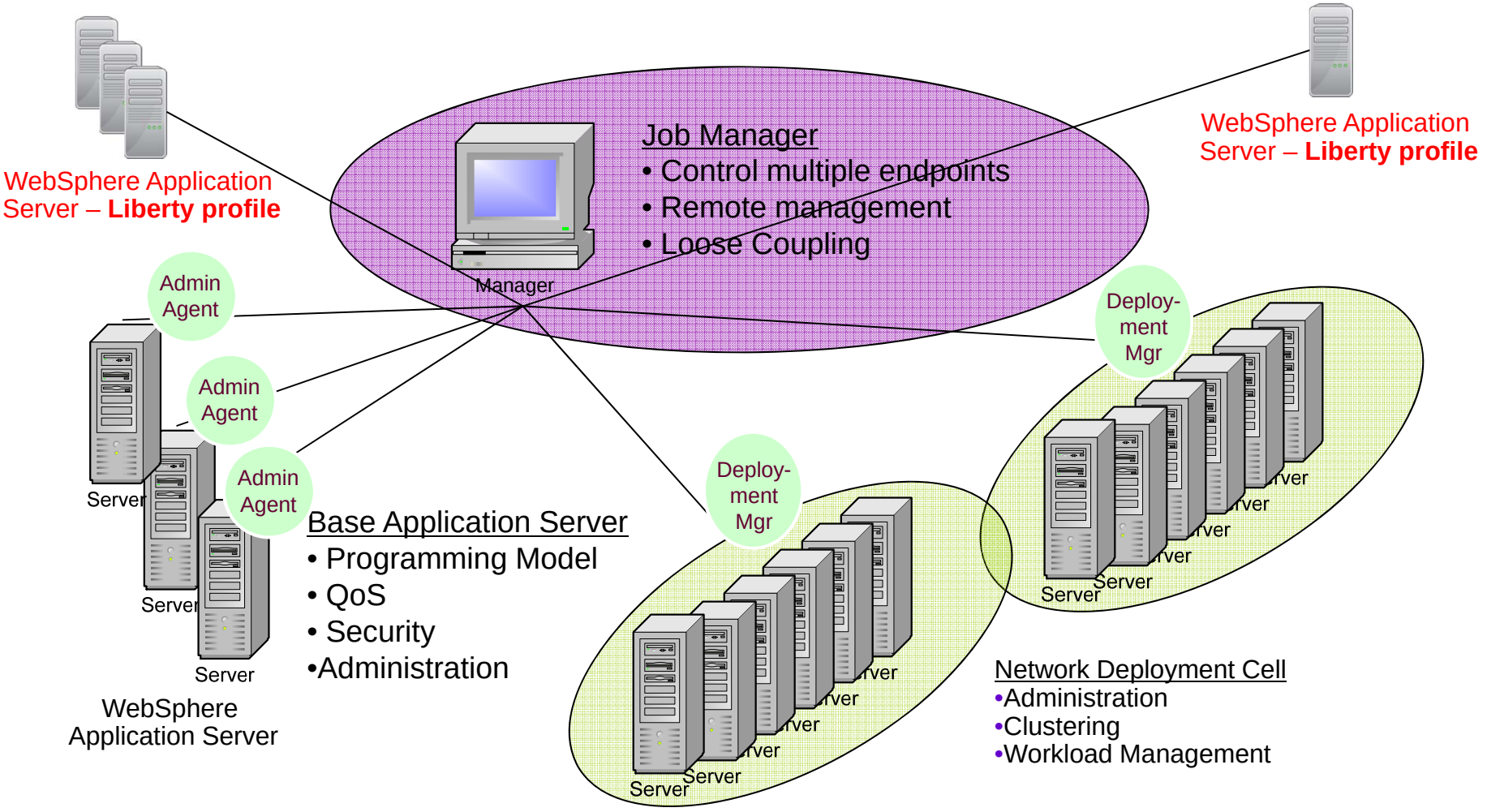
app

deploy

Manually, or
Job Manager

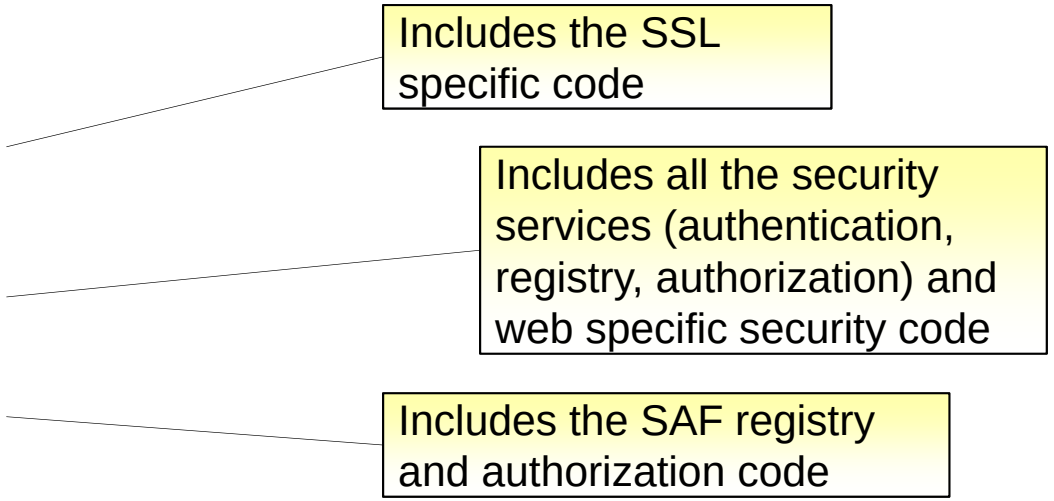
WLP = WAS Liberty Profile

Flexible Management



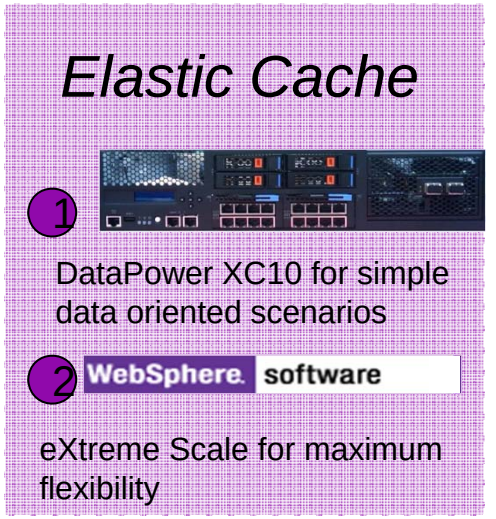
Security

- Secure out of the box
- All opened ports are local host only
- Exposes no remote management by default
- Seamless transition when enabling security
- Aim to make security painless
- Three key features
  - ssl-1.0
  - appSecurity-1.0
  - zosSecurity-1.0



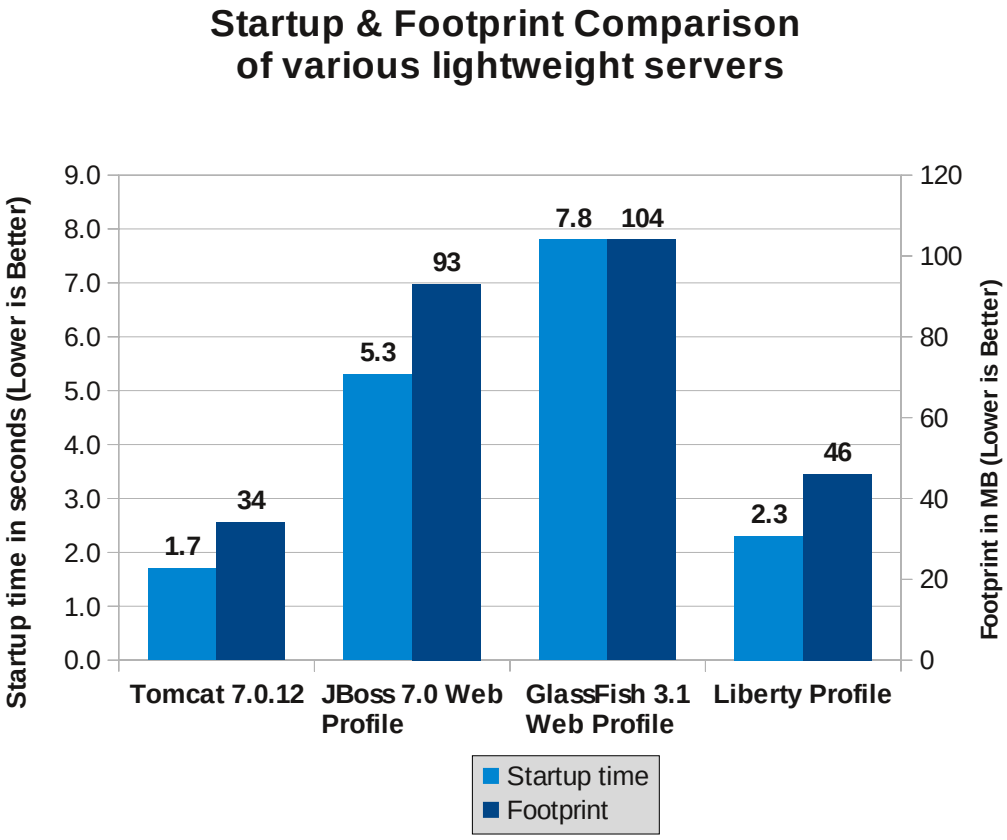
Elastic Caching with Liberty

- **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!



Liberty Profile – Startup & Footprint

- **The problem of a lightweight development environment in WebSphere has been solved!**
  - Liberty Profile startup & footprint are on par with Tomcat.
  - Liberty Profile starts up in less than half the time of JBoss Web profile.

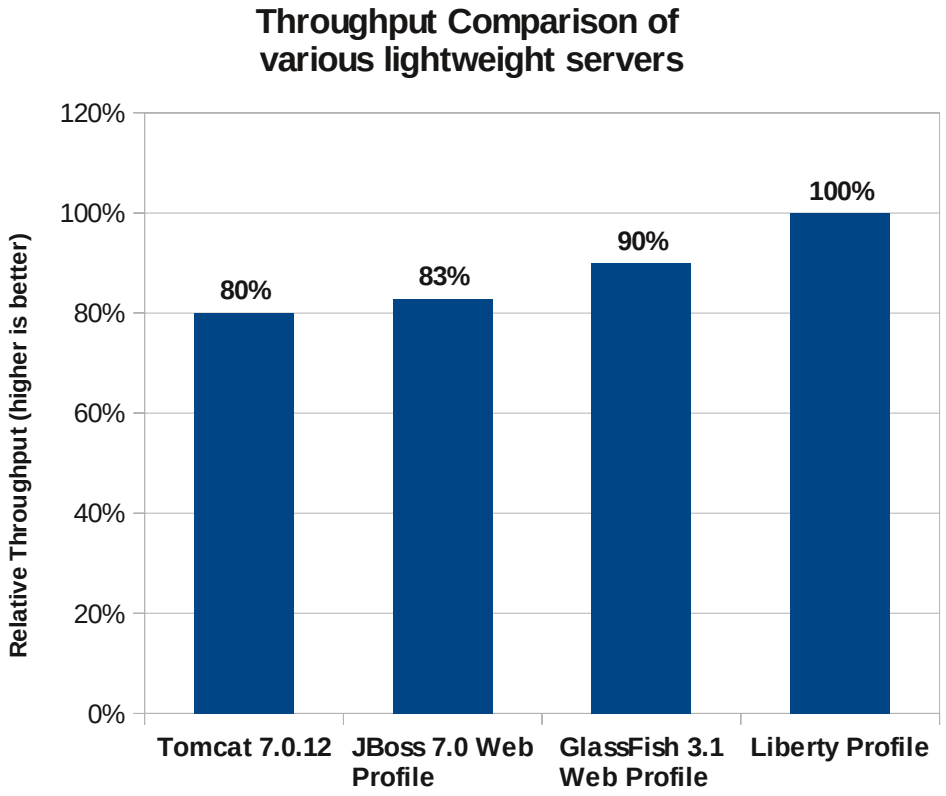


System Info:
Lenovo T60p - 2 x 2.16 GHz Intel Core Duo T2600
2GB RAM, Windows XP 32-bit
Apache Tomcat 7.0.12
JBoss Community Edition 7.0 Web Profile server
GlassFish Server 3.1 Open Source Edition Web Profile
WAS V8.5 Liberty Profile
(All servers had the TradeLite benchmark application installed)

Note: Tomcat , JBoss, and GlassFish were measured with the HotSpot JDK, while Liberty was measured with the IBM JDK.

Liberty Profile – Throughput

- **A lightweight server that can service requests with the speed of a full production server!**
  - Liberty Profile provides up to 20% better runtime performance than JBoss and 25% better than Tomcat.



System Info:
IBM x3550 – 4 x 1.86 GHz Intel Xeon E5320, 8 GB RAM
RedHat Linux 5.3 32-bit
Apache Tomcat 7.0.12
JBoss Community Edition 7.0 Web Profile server
GlassFish Server 3.1 Open Source Edition Web Profile
WAS V8.Next Liberty Profile
(All servers had the TradeLite benchmark application installed)

Note: Tomcat , JBoss, and GlassFish were measured with the HotSpot JDK, while Liberty was measured with the IBM JDK.

WAS V8.5 Delivers

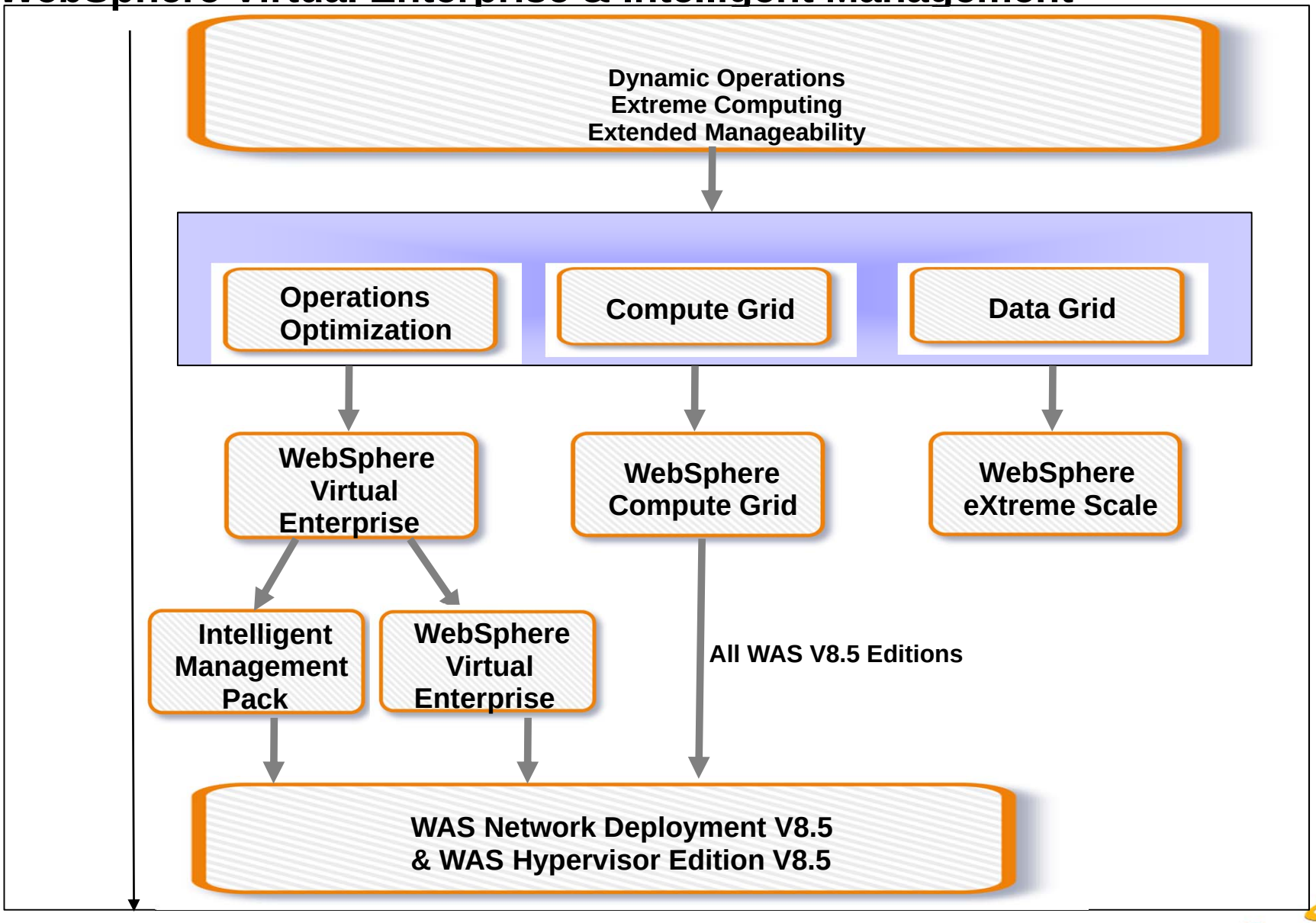
Unparalleled Application Development and Management Environment, Rich User Experiences…Faster

Application Resiliency

Intelligent Management & Enhanced Resiliency

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

WAS ND V8.5: Resiliency enhancements – WebSphere Virtual Enterprise & Intelligent Management

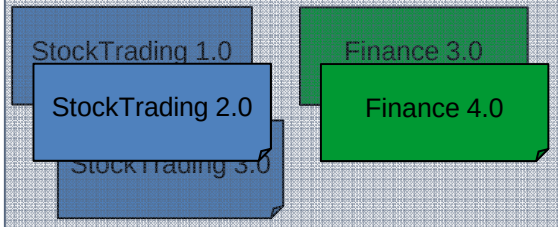


Intelligent Management

Extending QoS through autonomic computing

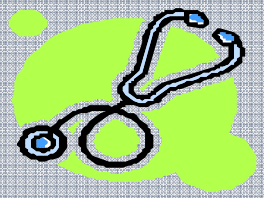
Application Edition Management

Self-Managing



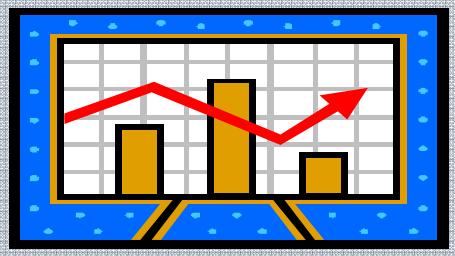
Health Management

Self-Protecting
Self-Healing

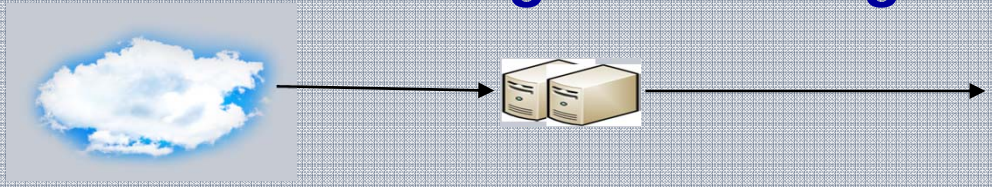


Dynamic Clustering

Self-Optimizing



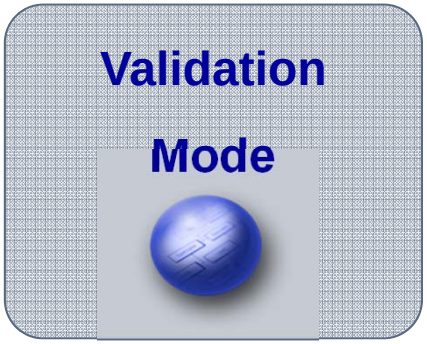
Intelligent Routing

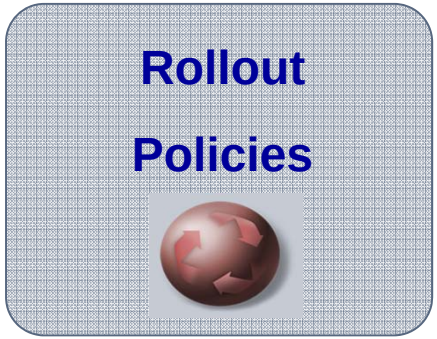


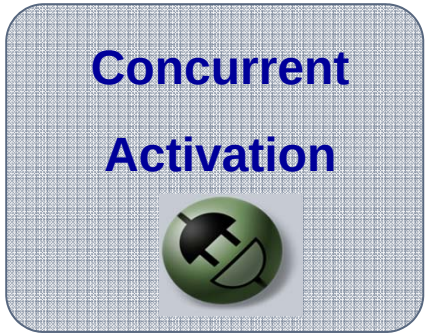
Application Edition Management

Applications can be upgraded without incurring outages

- Upgrade Applications without interruption to end users
- Concurrently run multiple editions of an applications
  - 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







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

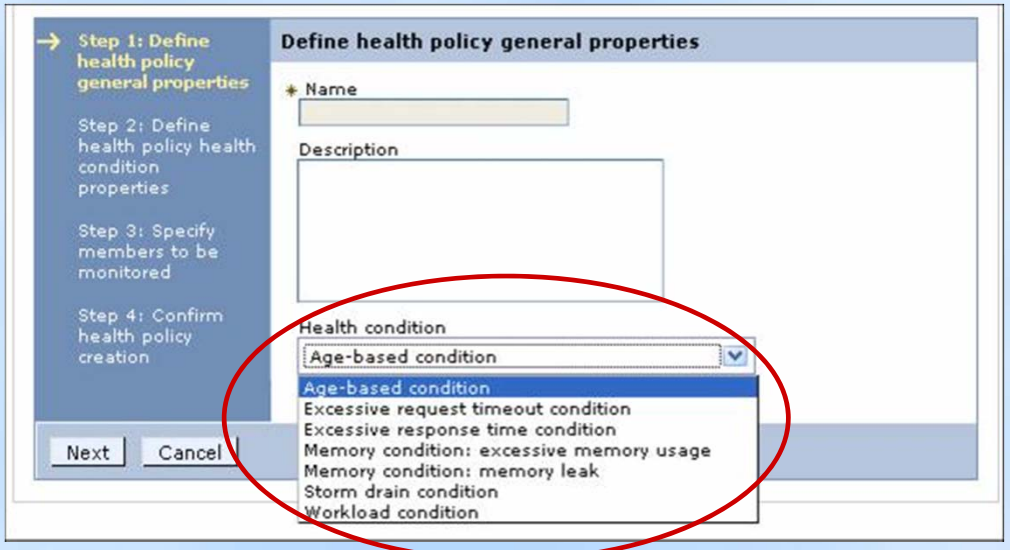
Customizable Health Conditions

Customizable Health Actions

Health Management – Health Policies

Helps mitigate common health problems before outages occur

- Health policies can be defined for common server health conditions
- When a health policy's condition is true, corrective action execute automatically or require approval
  - Notify administrator (send email or SNMP trap)
  - Capture diagnostics (generate heap dump, java core)
  - Restart server
- Excessive response time means you are monitoring what matters most: your customer's experience!
- Application server restarts are done in a way that prevent outages and service policy violations
- Each health policy can be in supervise or automatic mode. Supervise mode is like training wheels to allow you to verify that a health policy does what you want before making it automatic.



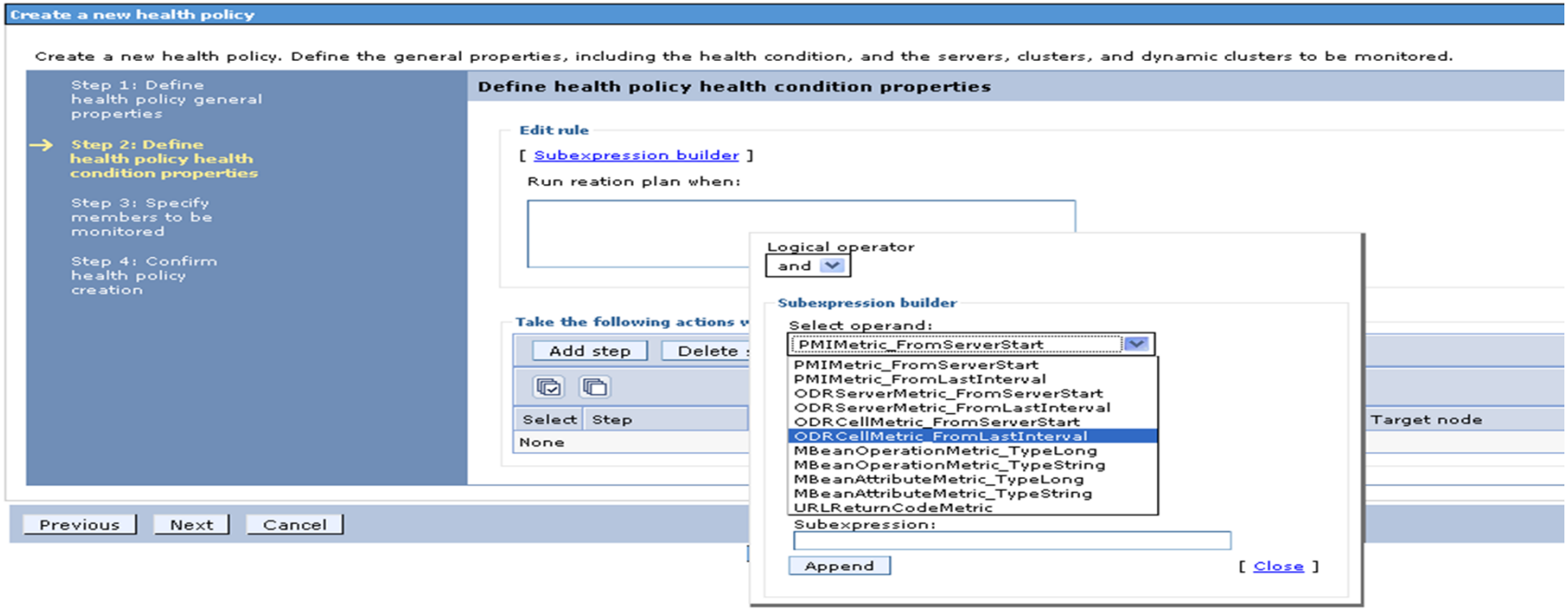
Health Conditions

- **Excessive request timeouts:** *% of timed out requests*
- **Excessive response time:** *average response time*
- **Excessive garbage collection:** *% of time spent in GCs*
- **Excessive memory:** *% of maximum JVM heap size*
- **Age-based:** *amount of time server has been running*
- **Memory leak:** *JVM heap size after garbage collection*
- **Storm drain:** *significant drop in response time*
- **Workload:** *total number of requests*

Health Management – Custom Health Conditions

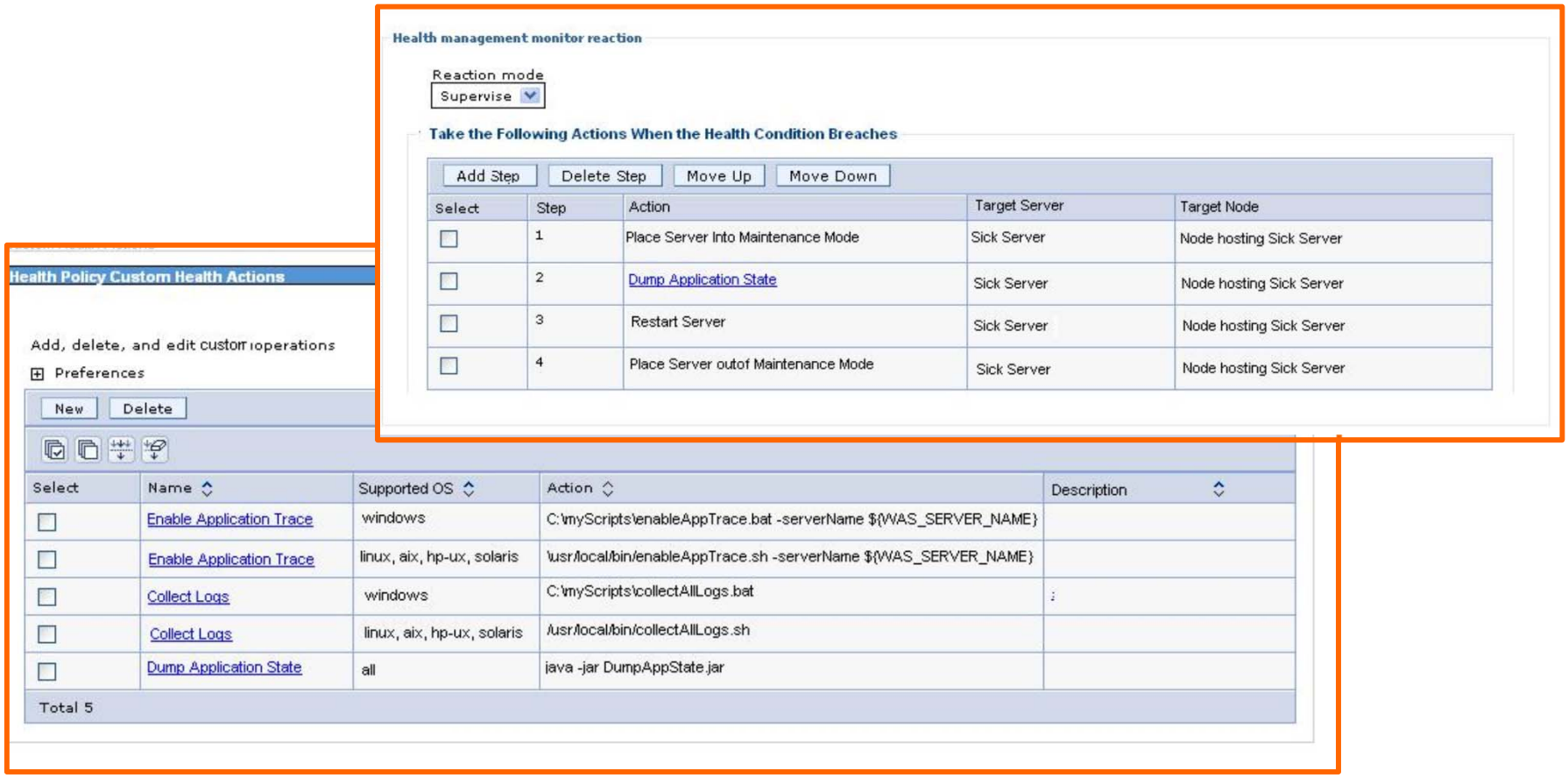
Flexibility to determine what an "unhealthy" condition is…

- Custom expressions can be built which use metrics from:
  - The On Demand Router, URI return codes
  - PMI metrics, MBean operations and attributes
  - Examples: hung thread detection, DB connection pool exhaustion or slow down
- Complex boolean expressions using a mix of operands is supported (AND, OR, NOT)



Health Management – Custom Health Actions

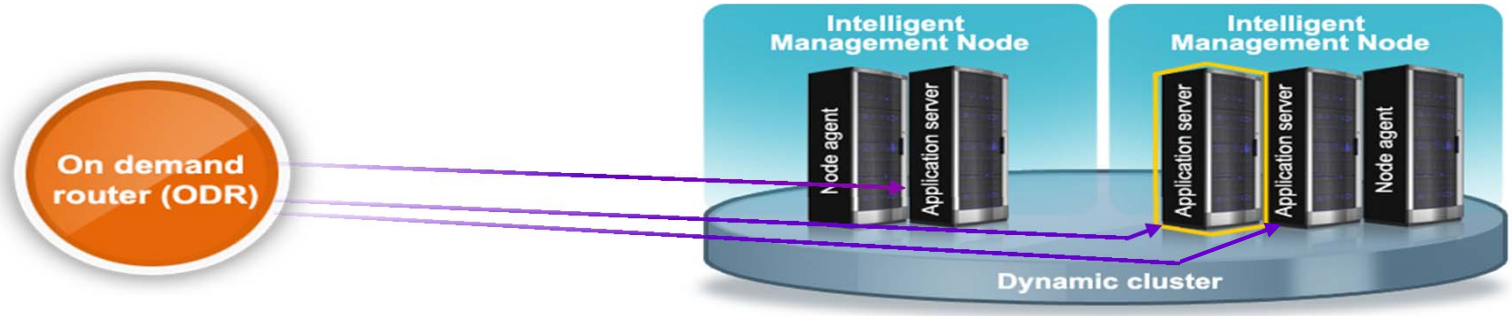
Provides flexibility by allowing the definition of custom actions allowing administrators to define an action plan to be carried out when the unhealthy situation detected.



Dynamic Clustering

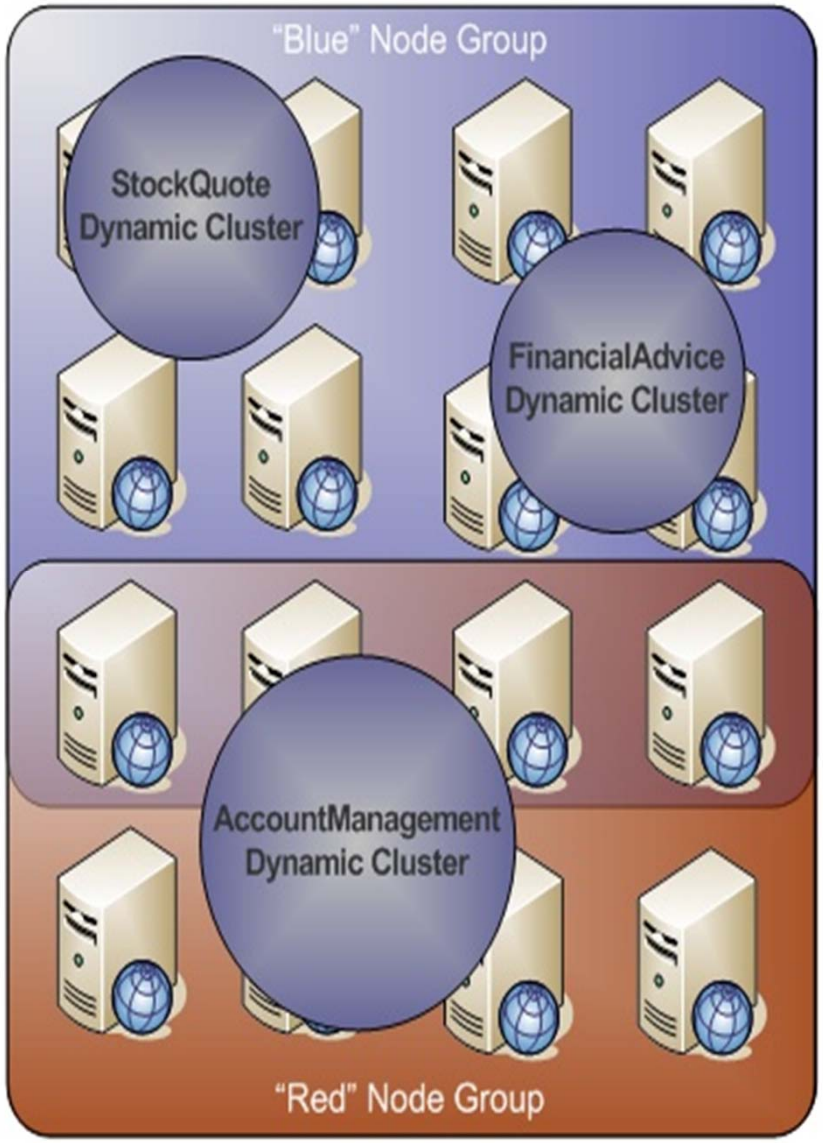
Proactively provision and start or stop application servers based on workload demands to meet Service Level Agreements

- 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



Dynamic Clustering

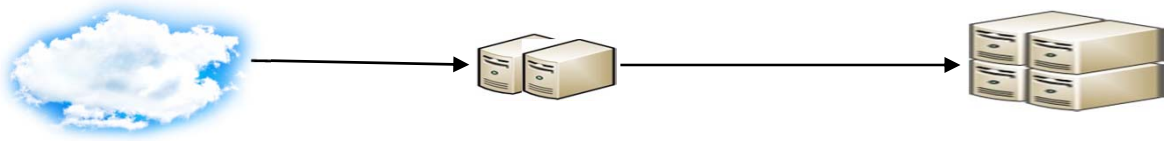
- A Dynamic Cluster is a virtual cluster of servers (JVMs) hosting the application that lives on group of nodes
- What is dynamic about a dynamic cluster?
  - App server definitions are dynamically created or deleted based upon the node membership policy (e.g. Servers are created/deleted if a node is added to /removed from a node group)
  - App server definitions are automatically updated when the server template associated with the dynamic cluster is updated
  - App servers are started / stopped based upon current application demand & service policies



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



WAS V8.5 Delivers

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

Operations and Control

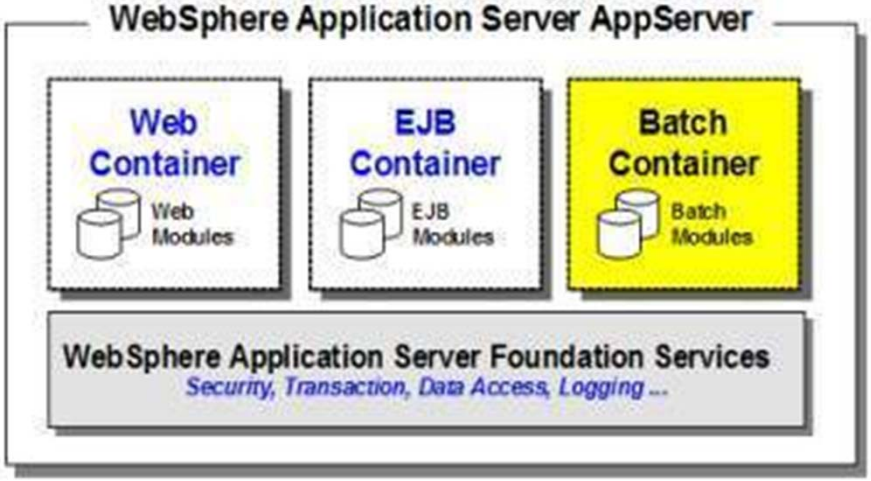
Improved Operations, Security, Control & Integration

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

WebSphere Batch

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

- **Lower TCO:** Concurrent execution of batch & online transaction processing (OLTP) workloads using shared business logic on a shared infrastructure; Higher throughput and lower resource consumption on z/OS when collocated with data subsystems
- **Enhanced Developer Productivity:** Pre-integrated application framework, Java batch programming model and tools
  - **Automation & Admin:** Container managed services for checkpoint and restart capabilities. Integrated administration of OLTP applications and batch jobs
  - **Packaging utility:** Utility to package batch application that can be deployed using JEE runtime

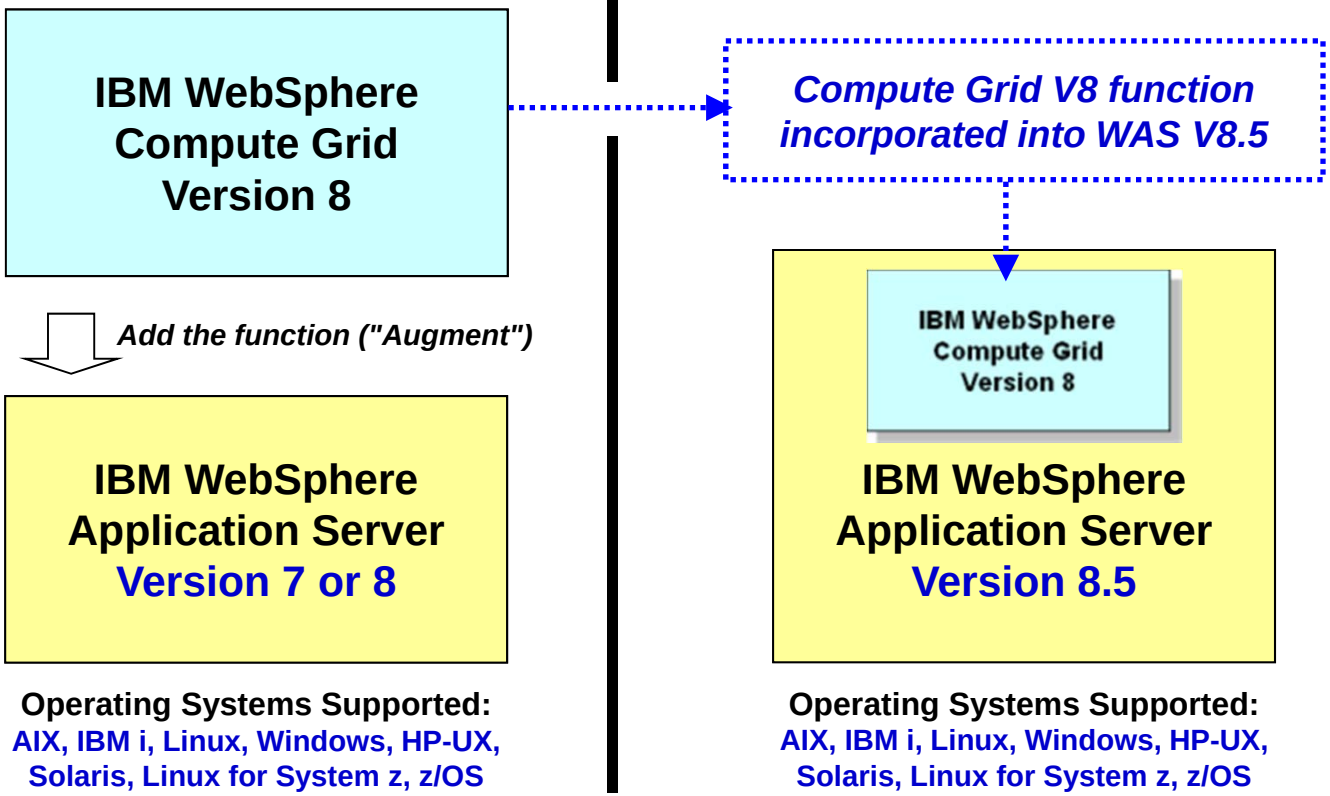


WebSphere Batch – Yesterday & Today

The WebSphere Java Batch function is provided in two ways today:

Java Batch Function

Java Execution Runtime



Function is identical between the two environments

Compute Grid V8 available for those who have not yet migrated their execution runtimes to WAS V8.5

WebSphere Batch Components

- **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.

- **Parallel Job Manager**
  - Controls parallel job execution including splitting and merging of jobs.
- **External Scheduler Integration**
  - Enables integration to external products for scheduling (i.e. Tivoli Workload Scheduler) and monitoring (i.e. ITCAM) batch workloads.
- **COBOL Container on z (new in WAS V8.5 and WCG V8)**
  - Call COBOL modules from Java on the same thread. COBOL and Java run in same transaction scope. WAS-Managed DB2 Connections shareable with COBOL.

Migration from V7 to V8.x: Considerations

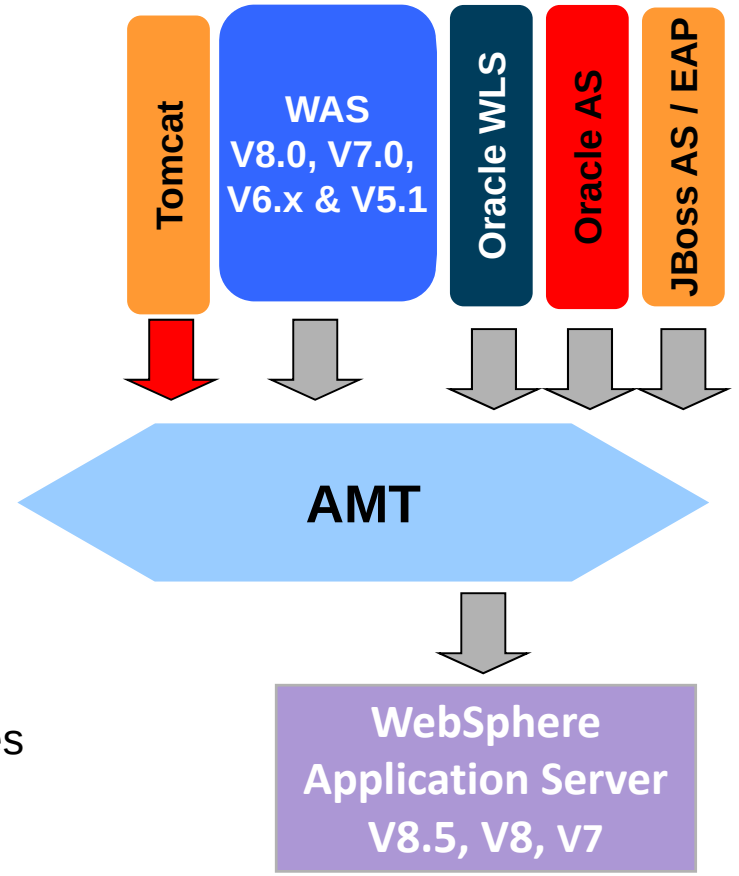
- WAS V8.0 and WAS V8.5 have the same Java EE 6 and same default Java SE 6
  - WAS V8.5 additionally supports Java SE 7
- Same Installation Manager, administrative infrastructure and security defaults.
- WAS V8.5 has significant additive features including:
  - Liberty Profile
  - Intelligent Management
- All applications developed using WAS V8.5 Liberty Profile will run unmodified on **WAS V8.0** and **WAS V8.5** full profile servers.
  - Using traditional WAS deployment methodology and server configuration for full-profile servers
  - Since Liberty is a profile sub-set, not all Apps can be moved in the other direction

Application Migration Toolkit v3.0

Enhanced!

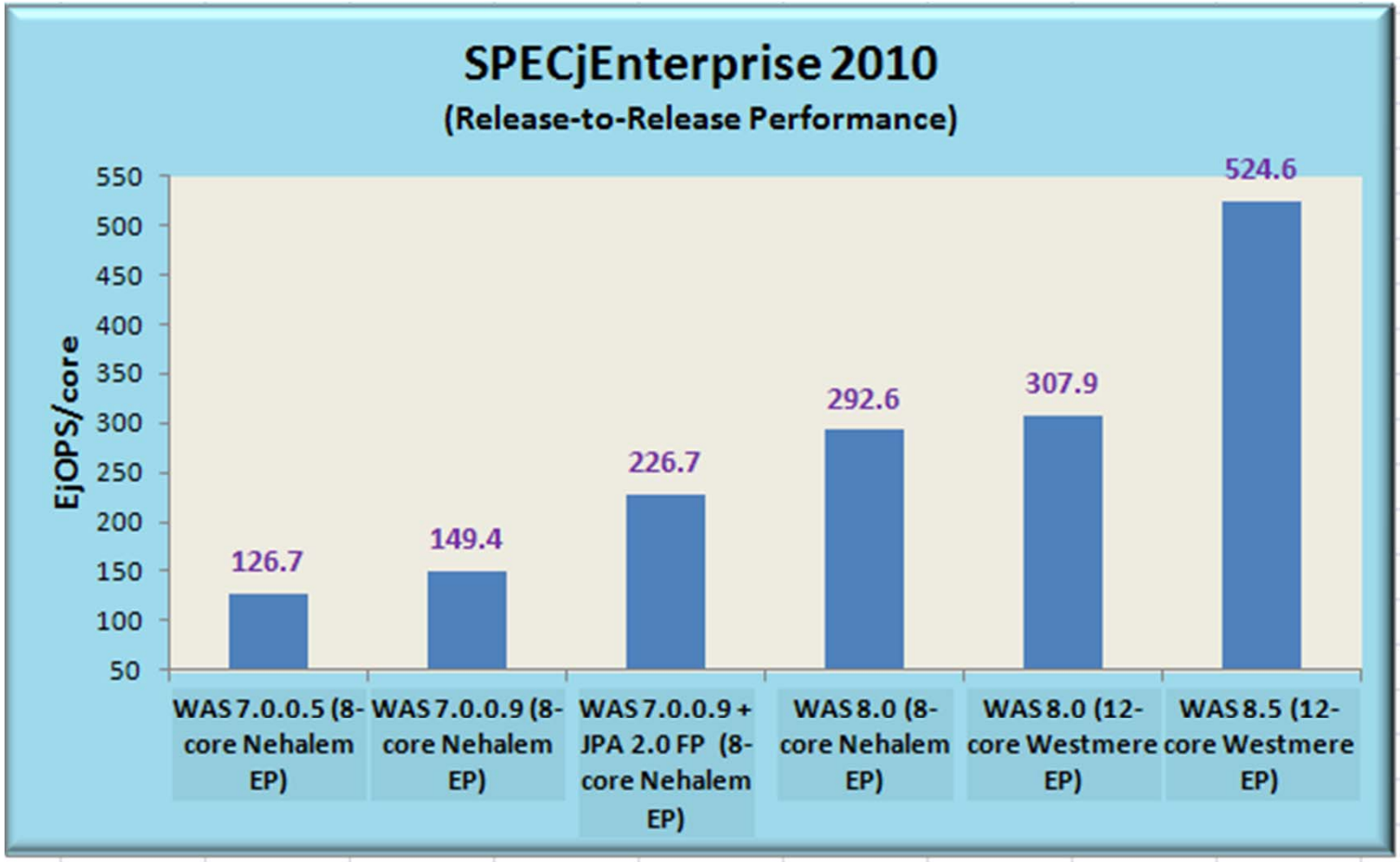
Migrate applications from WebSphere & other Java EE application servers to WebSphere faster with minimized risk

- Migrate apps from older vers to WAS V8.5, V8, V7
- Migrate from Oracle, TomCat, JBoss faster & easier
  - 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 & deprecated features
    - Behavior changes
    - JRE 5 & JRE 6 differences
    - Java EE spec changes or enforcements
  - Capable of making some application changes
  - Provides guidance on how to make required changes
  - Works with Eclipse or RAD (RAD)



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

WebSphere Release-to-Release Performance

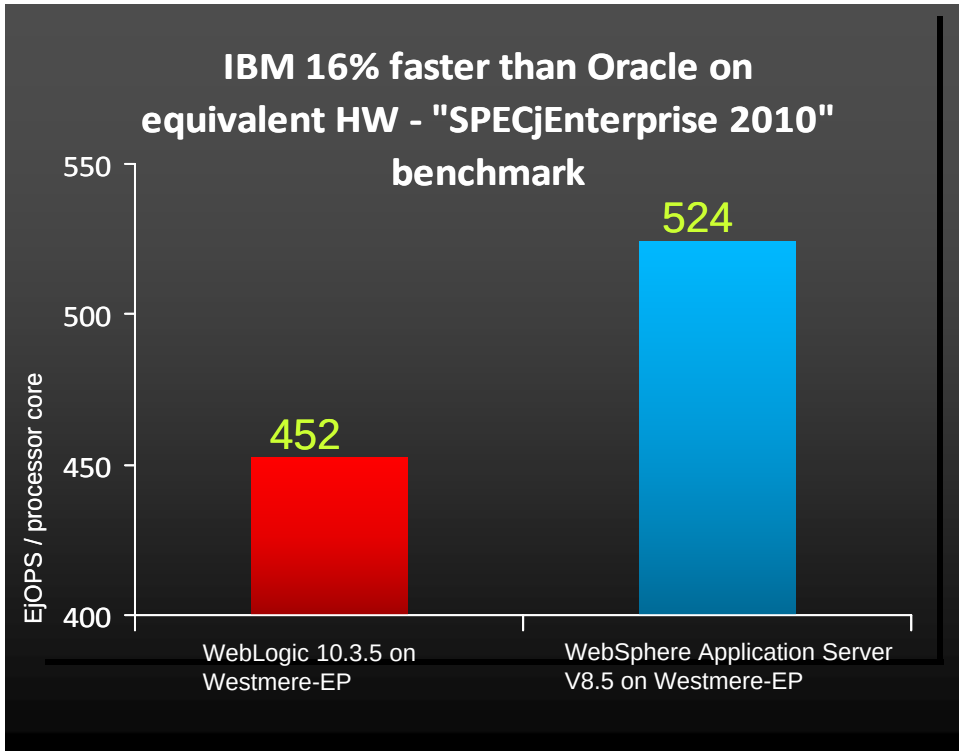


Consistent Performance gains across WAS Releases

As per SPEC Published Data as of 4/26/2012
http://www.spec.org/jEnterprise2010/results/jEnterprise2010.html

WebSphere outperforms Oracle WebLogic

IBM is the world leader in middleware performance

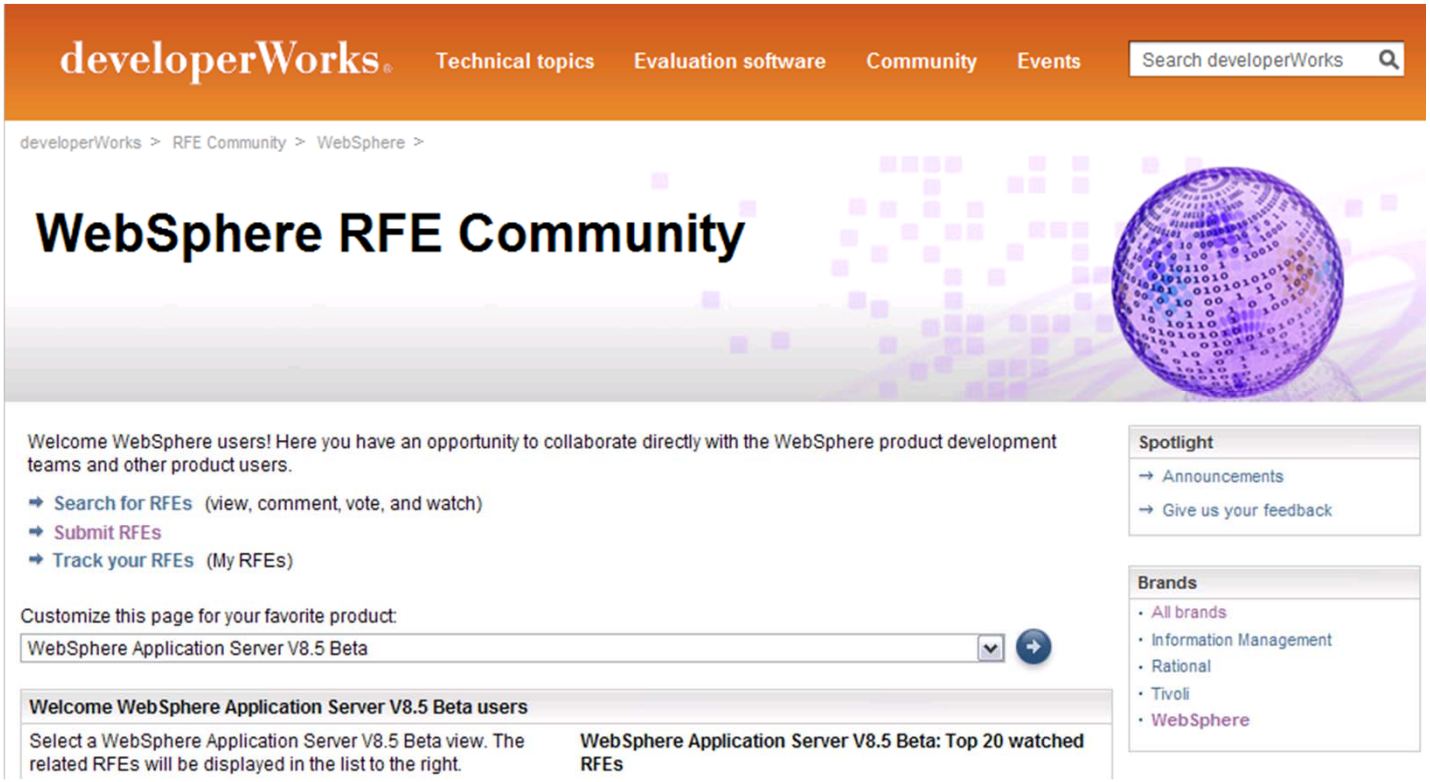


- IBM 16% better than Oracle on same HW
- Improve performance and efficiency leveraging current HW investments
- Improve transaction availability of your SOA infrastructure by getting more out of your Hardware
- IBM #1 even if Oracle uses latest HW
- IBM middleware makes the best use of all HW Platforms – Intel, Power and/or z
- Get the best bang for your buck – run the most transactions at the lowest cost

1) SPEC and SPECjEnterprise 2010 are registered trademarks of the Standard Performance Evaluation Corporation.
Results from www.spec.org as of 04/29/2012 Oracle SUN Blade Server X6270 M2 452.285 EjOPS/core SPECjEnterprise2010, Oracle Sun Fire X4170 M3 – 519.386 SPECjEnterprise2010 EjOPS (Oracle's best SPECjEnterprise2010 EjOPS/core result so far). IBM HS 22 Blade 524.621 EjOPS/core (World Record SPECjEnterprise2010 EJOPS/core result)

Request For Enhancements (RFE)

http://www.ibm.com/developerworks/rfe



- **Raise and track requirements using the new IBM RFE system:**
  - Introduced for WAS V8.5 Beta
- **Live for commercially available WAS since March 2012.**