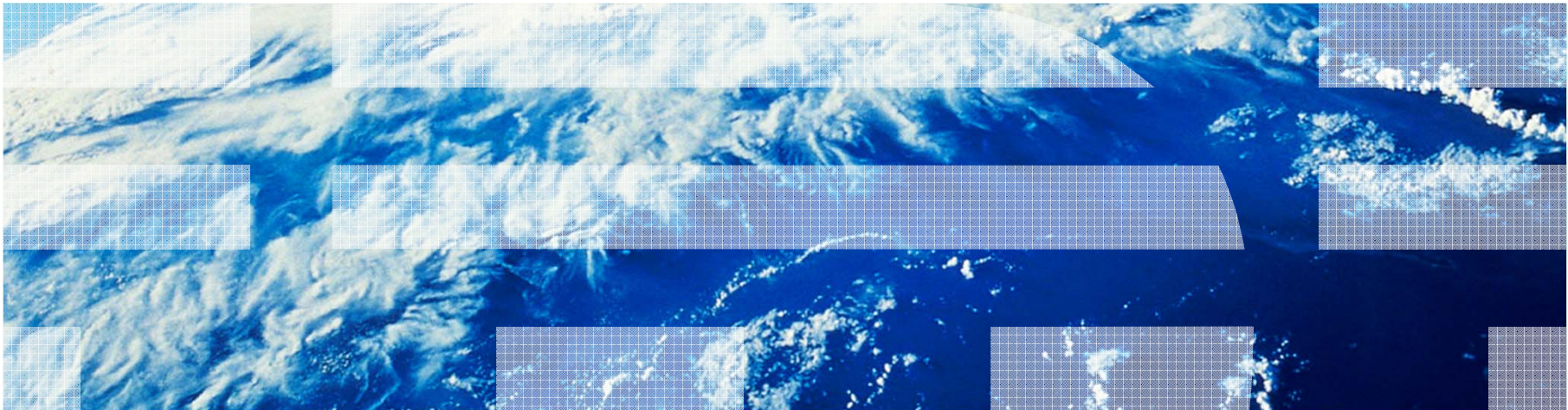


# WebSphere Application Server Update and Direction

**Ian Robinson, WebSphere Foundation Chief Architect**





## Trademark Statement

- IBM and the IBM logo are trademarks of International Business Machines Corporation, registered in many jurisdictions. Other marks may be trademarks or registered trademarks of their respective owners.
- Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
- Other company, product and service names may be trademarks, registered marks or service marks of their respective owners.
- References in this publication to IBM products and services do not imply that IBM intends to make them available in all countries in which IBM operates.

# Important Disclaimer



- THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY.
- WHILST EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.
- IN ADDITION, THIS INFORMATION IS BASED ON IBM’S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE.
- IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION.
- NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, OR SHALL HAVE THE EFFECT OF:
  - CREATING ANY WARRANTY OR REPRESENTATION FROM IBM (OR ITS AFFILIATES OR ITS OR THEIR SUPPLIERS AND/OR LICENSORS); OR
  - ALTERING THE TERMS AND CONDITIONS OF THE APPLICABLE LICENSE AGREEMENT GOVERNING THE USE OF IBM SOFTWARE.



# Application Infrastructure Trends

**Rapid application development and delivery** driving simplified, integrated and automated development and operations lifecycles

**Explosion of mobile, social and cloud applications** driving new demands on middleware infrastructures

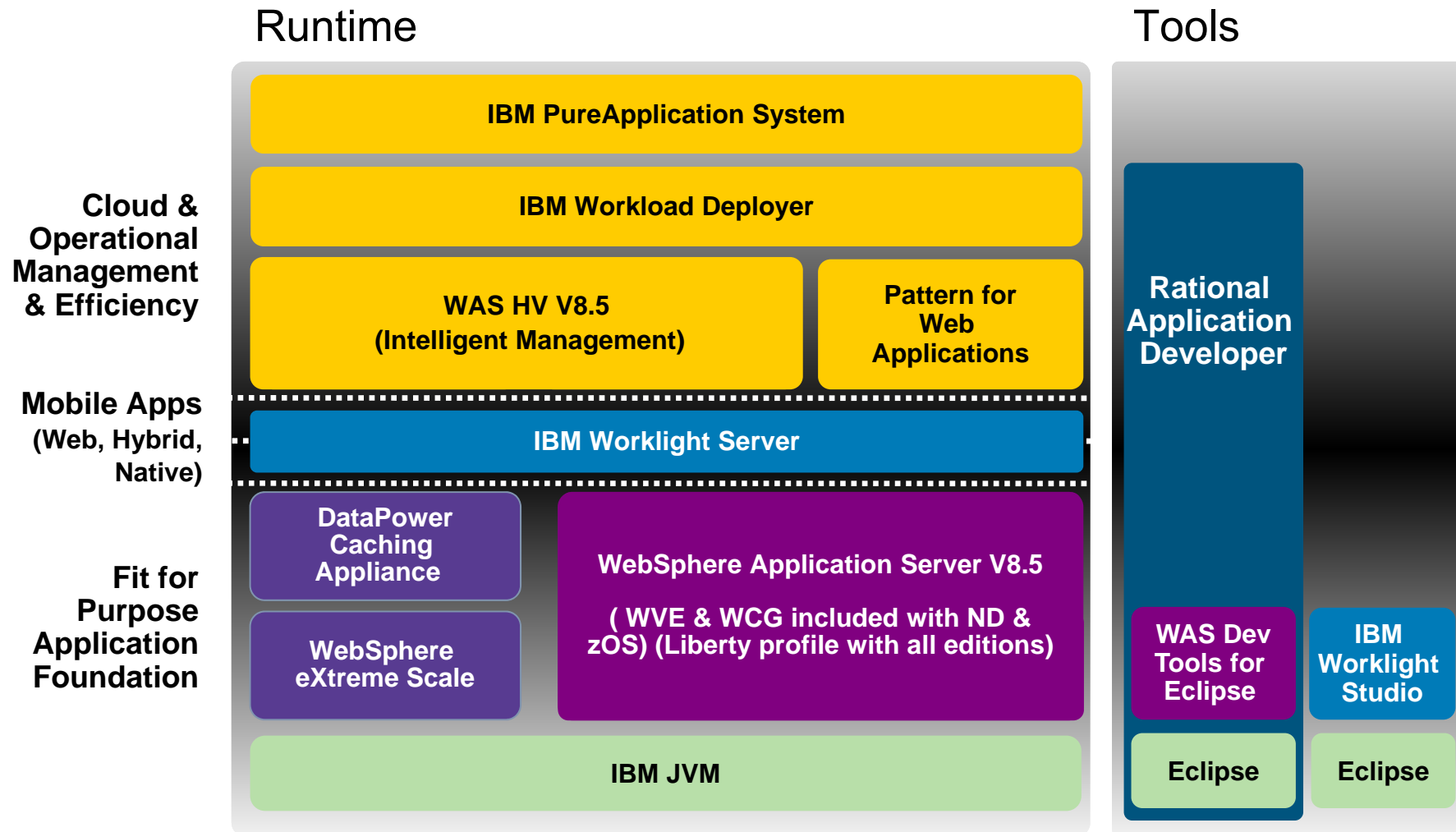
The combination of huge transaction volumes against massive amounts of data with little tolerance for delays is driving the need for **elastic caching** technologies

Use of **cloud delivery models** to provide elasticity, scale, multi-tenancy and context across different form factors and access methods



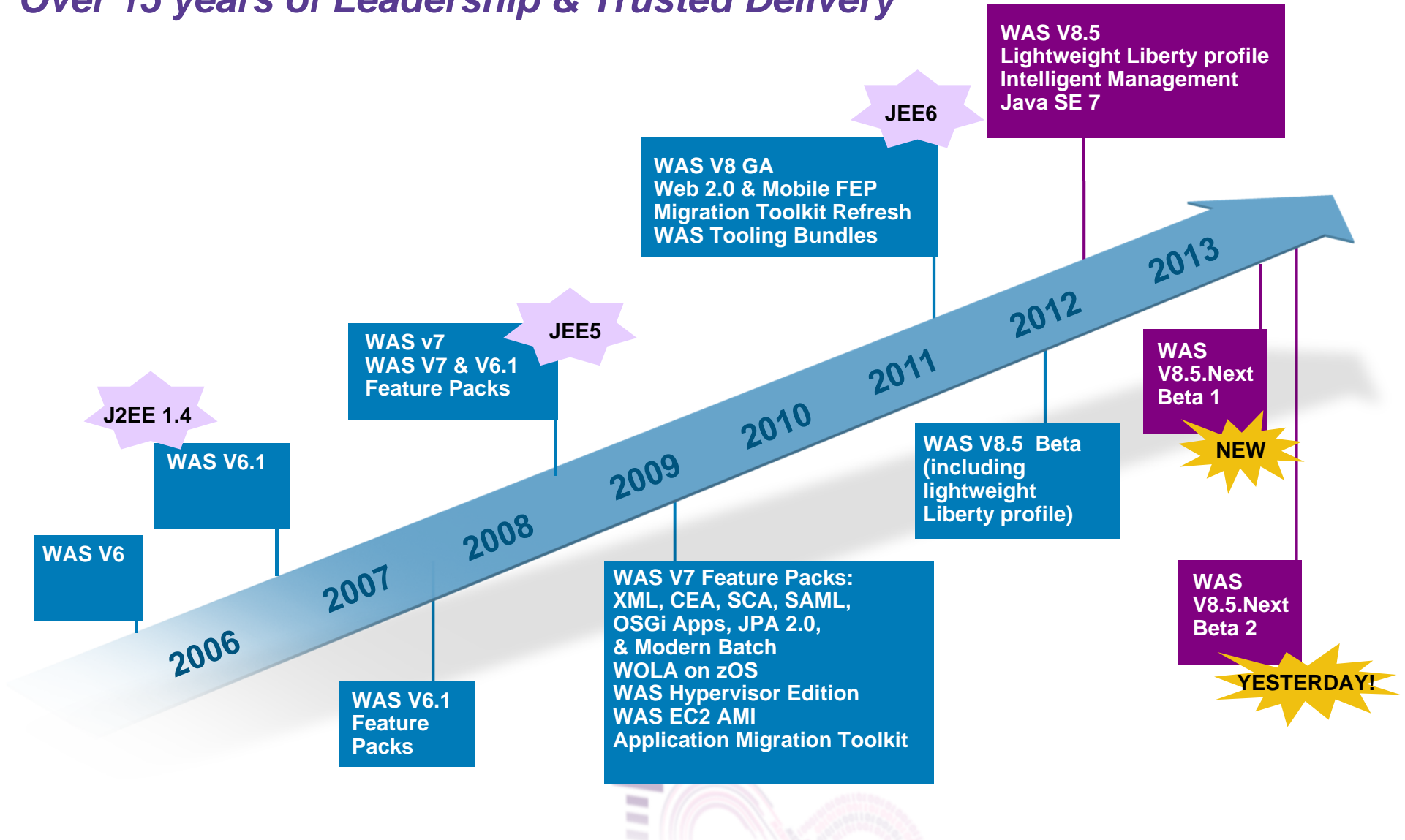
# WebSphere Application Infrastructure

## What's Changed Over the Last Year





# WebSphere Application Server: Over 15 years of Leadership & Trusted Delivery





# 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





# WAS v8.5 with the Liberty Profile and Intelligent Mgmt

<h3>WAS for Developers</h3> <p><b>Tools Edition</b> <b>+Liberty Profile</b></p> <p>Enables efficient development of innovative apps that will run on WAS in production</p> <p>Available as a no-charge edition for the developer desktop and includes Eclipse adapters</p>	<h3>WAS Hypervisor Edition</h3> <p><b>+Liberty Profile</b> <b>+Intelligent Mgmt</b></p> <p>Optimized to instantly run in VMware and other server virtualization environments</p>	<h3>WAS ND</h3> <p><b>Tools Edition</b> <b>+Liberty Profile</b> <b>+Intelligent Mgmt</b></p> <p>Delivers near-continuous availability, with advanced performance and mgmt capabilities, for mission-critical apps</p>	<h3>WAS for z/OS</h3> <p><b>+Liberty Profile</b> <b>+Intelligent Mgmt</b></p> <p>Takes full advantage of the z/OS Sysplex to deliver a highly secure, reliable, and resource efficient server experience</p>
<h3>WAS</h3> <p><b>Tools Edition</b> <b>+Liberty Profile</b></p> <p>Provides secure, high performance transaction engine for moderately sized configurations with web tier clustering and failover across up to five application server profiles</p>			
<h3>WAS Express</h3> <p><b>+Liberty Profile</b></p> <p>A lower-cost, ready-to-go solution to build dynamic Web sites &amp; apps</p>		<h3>WAS CE</h3> <p>An open source-based, small footprint foundation with no up-front acquisition costs</p>	

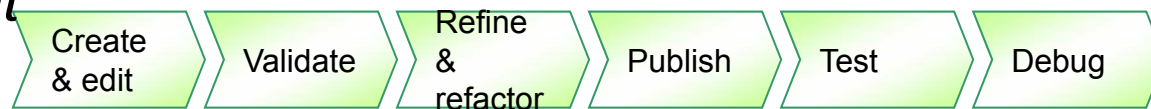
 Built on a common code base



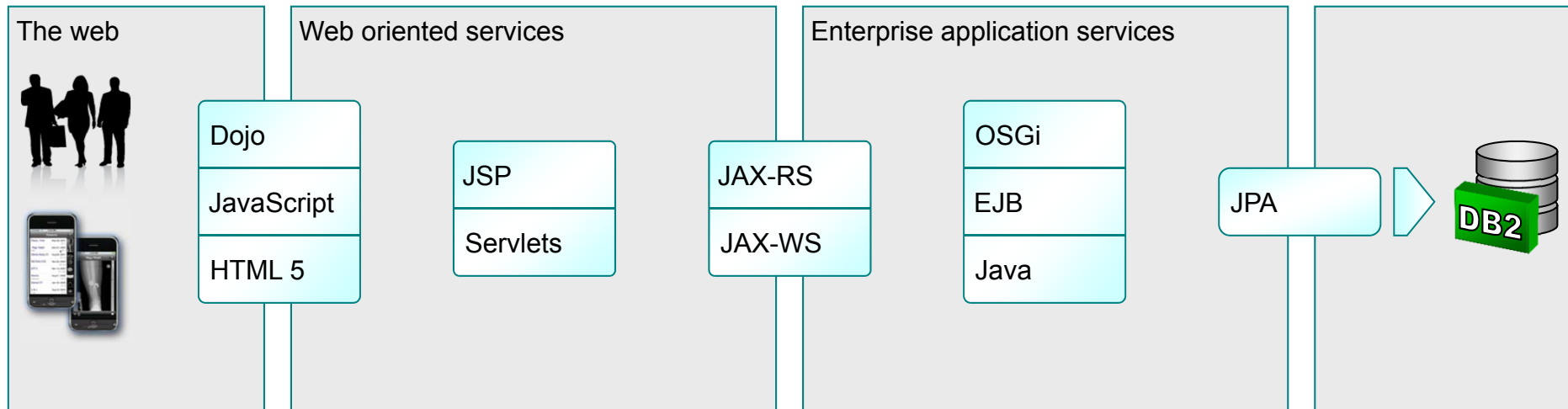


# Unleash innovation with WDT

*A cost effective IDE focused on web application and modern service development*



*Design web, Java EE and OSGi applications with persistence*



*Available from the Eclipse Marketplace*

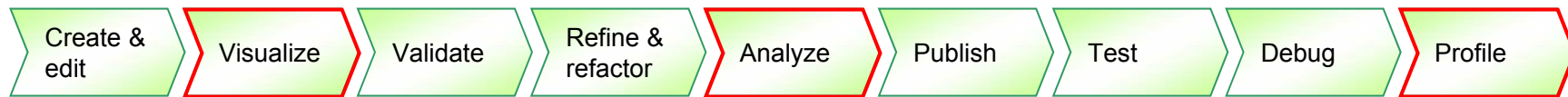
WebSphere foundation

# Enterprise Application Development with RAD

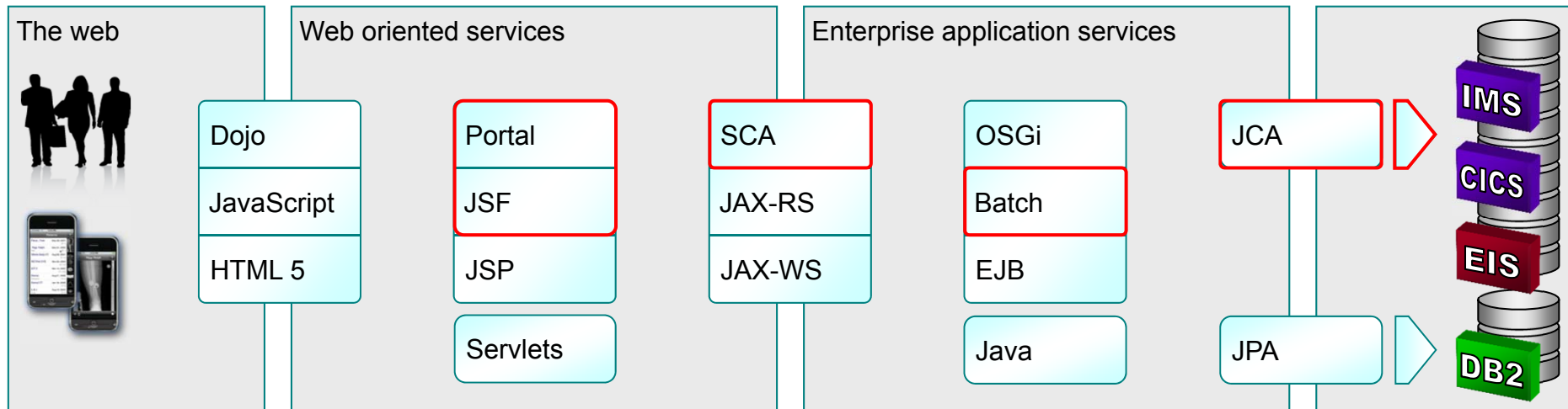


2012 Evans Data  
IDE Users Choice Award  
6 years in a row!

*Optimize team development of robust, high quality, enterprise integrated applications*



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



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





# WAS v8.5 introduced the Lightweight “Liberty” Profile – For Web and Mobile Apps

*A highly composable, dynamic Server profile*

## Dynamic Server Profile

Not static like Web Profile – configured by app at a fine-grained level

## Developer First Focus

Simplified, shareable server config (like a dev. artifact). One XML file or several to simplify sharing & reuse of config. Adds MAC OS for development

## Start fast, run efficiently

Starts in <3s  
Memory footprint: <50MB  
(TradeLite benchmark)

## Small Download:

50MB for Web Profile features



## Integrated Tooling

Powerful tools in WDT Eclipse feature. Maven plugin goals for build, test and deploy

Enhanced!

## WAS v8.5 Liberty Profile & WebSphere Developer Tools for Eclipse (WDT)

## Dynamically Extensible

Install new features from a repository (local or remote) without server restart

Enhanced!

## Unzip Install & Deploy

IM or unzip to install. Option to deploy “server package” of app + config + required subset of server runtime for highest density deployment.

Enhanced!

New!

## New features

CDI, JMS, JAX-WS, local EJB, OAuth, federated user repository, MongoDB, clustering, WebCache

## Lightweight cluster management

Liberty servers can join a lightweight cluster for workload balancing and high availability

New!

## Fidelity to full profile WAS

Same reliable containers & QOS. Develop on Liberty profile and deploy to Liberty or full-profile WAS



# Simplified Configuration

```
<server>  
  <featureManager>  
    <feature>jsp-2.2</feature>  
    <feature>jdbc-4.0</feature>  
  </featureManager>
```

Features control which capabilities (bundles) are installed in the server

'singleton' configurations specify properties for a runtime service like logging

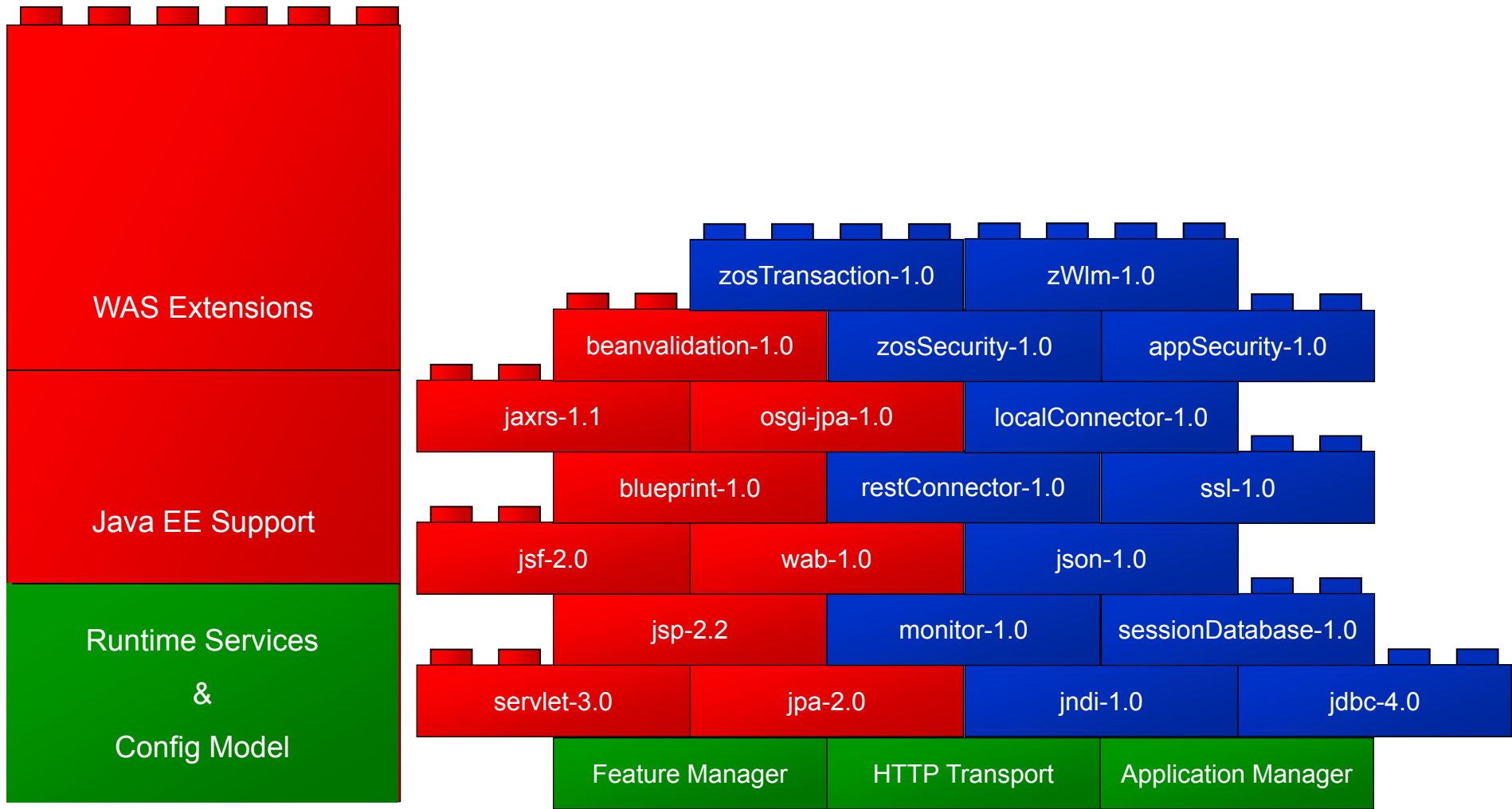
```
<logging traceSpecification="webcontainer=all=enabled:*=info=enabled"/>
```

'instance' configurations specify multiple resources like applications and datasource definitions

```
<application name="tradelite" location="tradelite.war"/>  
  
<dataSource jndiName="jdbc/TradeDataSource">  
  <properties.derby.embedded databaseName="${server.config.dir}/tradedb"/>  
</dataSource>  
</server>
```

Any of this configuration could be put into a separate xml file and 'included' in this 'master' configuration file

# Highly Composable Runtime Based on 'Features'



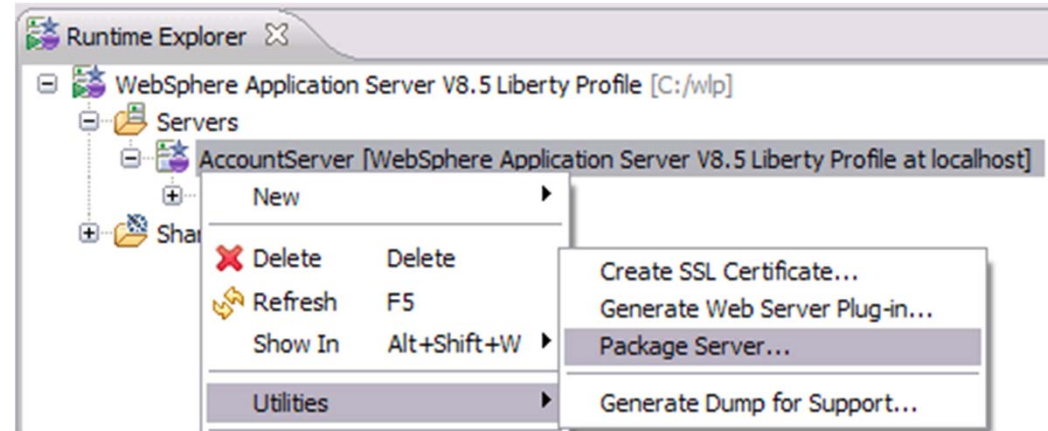
**Full WAS Profile**

**WAS v8.5 Liberty Profile**



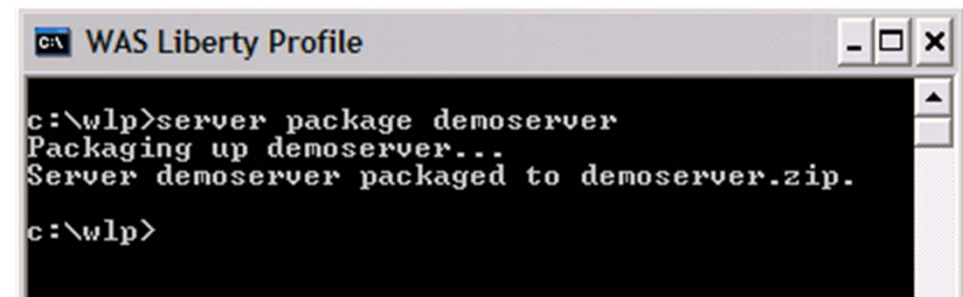
# New Deployment Options

- 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.



# Maven Integration

- Maven WebSphere Repository:  
<http://public.dhe.ibm.com/ibmdl/export/pub/software/websphere/wasdev/maven/repository/>
- Maven plugins:
  - liberty:install-server
  - liberty:start-server
  - liberty:stop-server
  - liberty:package-server
  - liberty:create-server
- Example:
  - `mvn liberty:start-server -DserverHome=/path/to/server_home -DserverName=[server_name]`



# 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

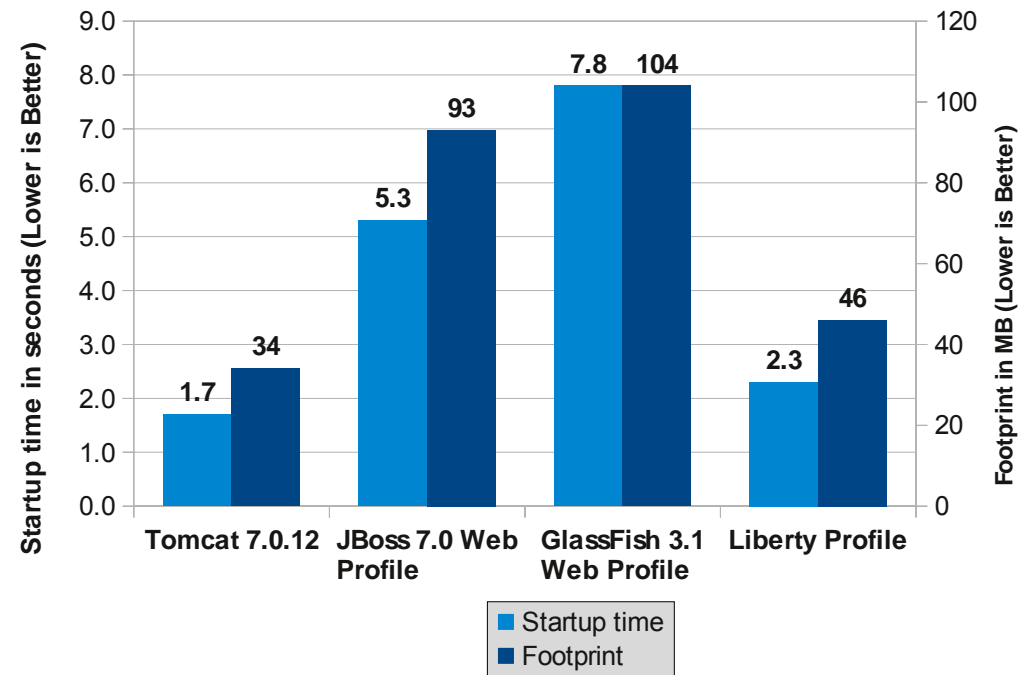
JBoss Community Edition 7.0 Web Profile server

GlassFish Server 3.1 Open Source Edition Web Profile

WAS V8.Next Liberty Profile

(All servers had the TradeLite benchmark application installed)

**Startup & Footprint Comparison of various lightweight servers**



*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

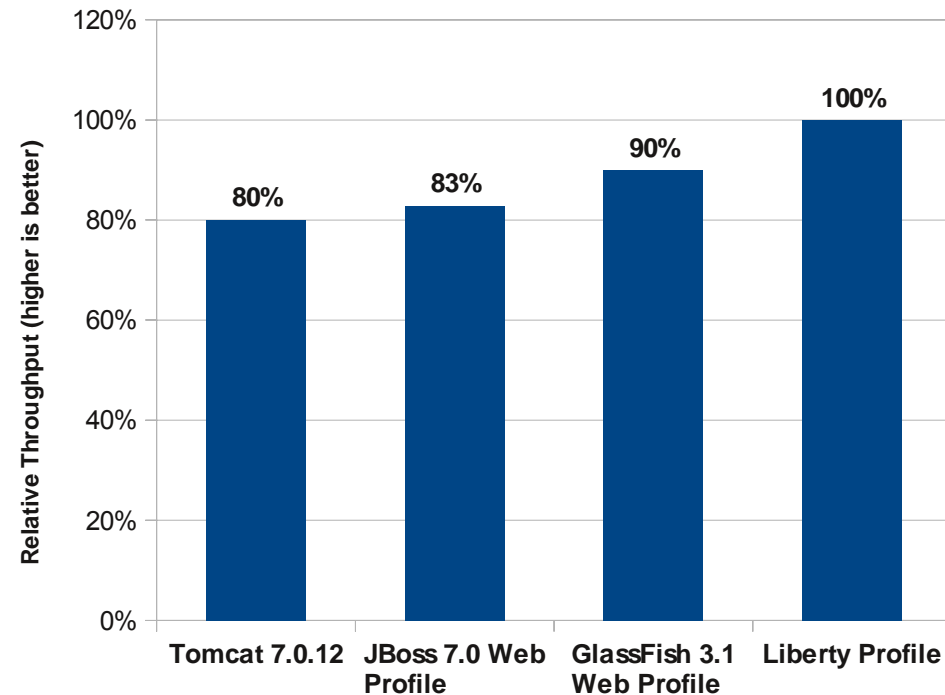
JBoss Community Edition 7.0 Web Profile server

GlassFish Server 3.1 Open Source Edition Web Profile

WAS V8.Next Liberty Profile

(All servers had the TradeLite benchmark application installed)

Throughput Comparison of various lightweight servers



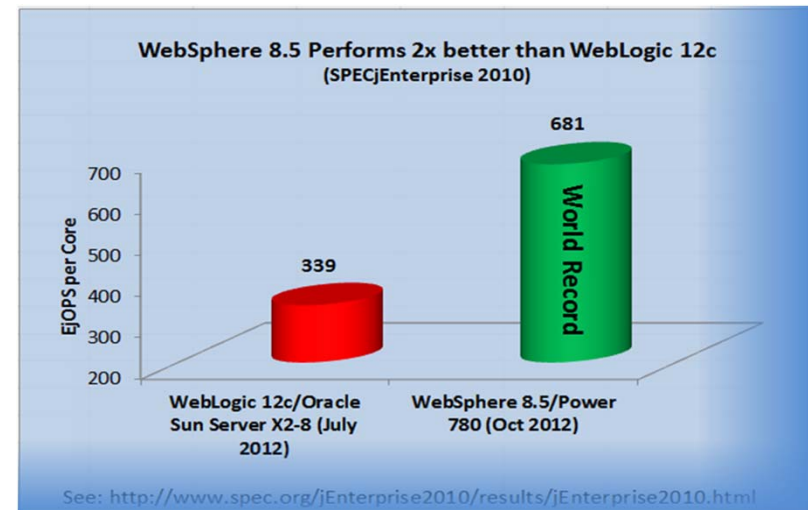
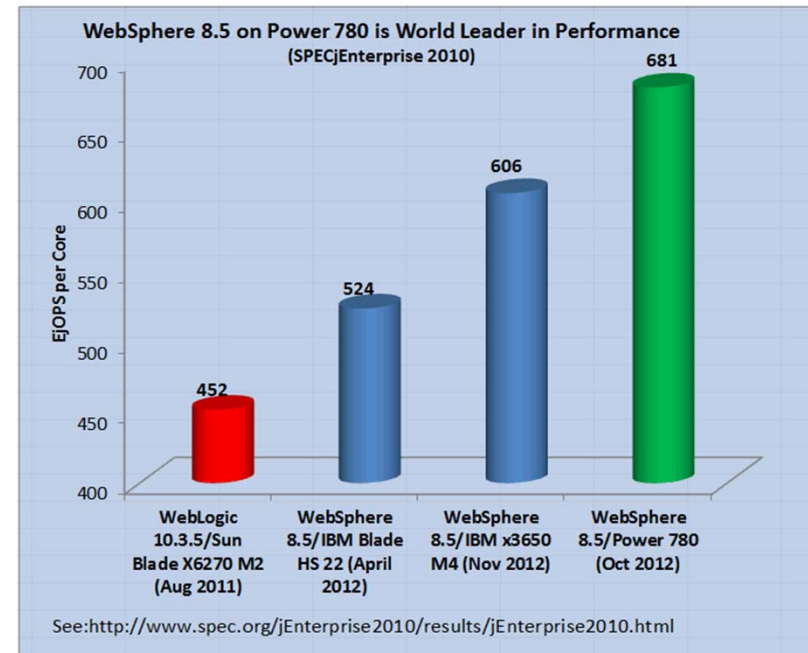
*Note: Tomcat , JBoss, and GlassFish were measured with the HotSpot JDK, while Liberty was measured with the IBM JDK.*



# World Record Performance

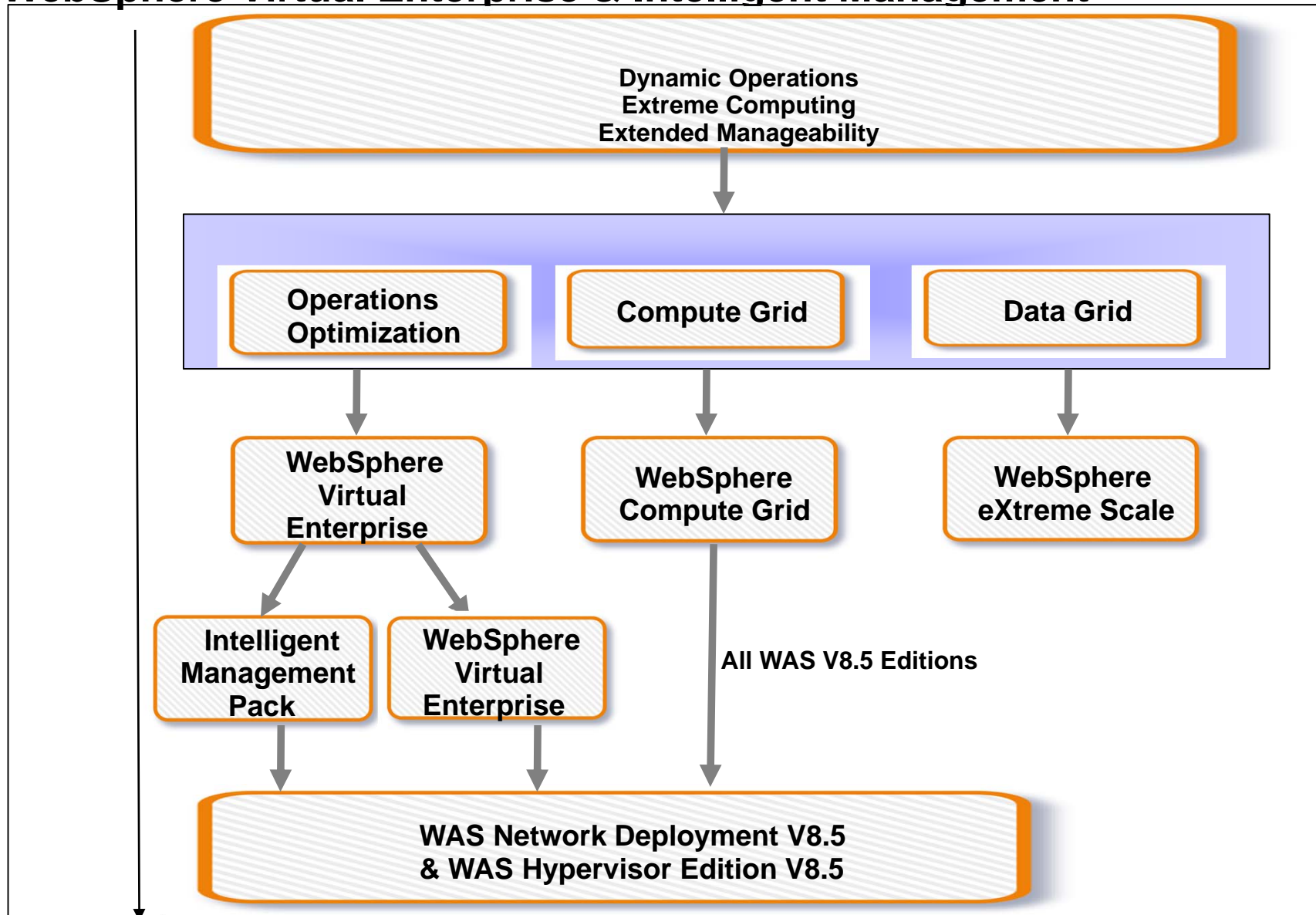
## IBM WebSphere Application Server v8.5 Performance on POWER7+

- Power 7+ leads the industry in performance
- WAS 8.5 performs almost 2x better than Oracle WebLogic 12c Server
- DB2 performs significantly better than Oracle DB on a per core basis
- Power 7+ performs 2.2x better than SPARC T4-4
- WAS 8.5/Power 7+ now delivers about 2.6x higher per core performance compared to WAS 7/Power 7
- Excellent collaboration between WAS, DB2, JDK, and Power teams which made this possible
- Teams are now geared to repeat the same success with Power 8 and WAS v.Next





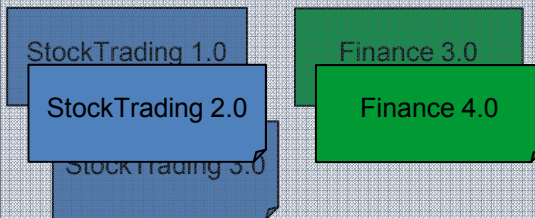
# WAS ND V8.5: Resiliency enhancements – WebSphere Virtual Enterprise & Intelligent Management



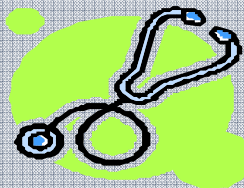
# Intelligent Management

*Extending QoS through autonomic computing*

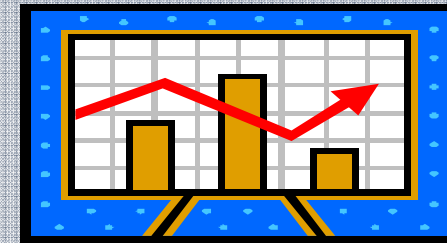
**Application  
Edition  
Management**  
*Self-Managing*



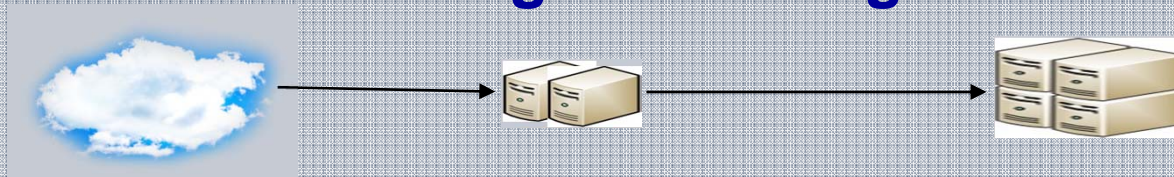
**Health  
Management**  
*Self-Protecting  
Self-Healing*



**Dynamic  
Clustering**  
*Self-Optimizing*



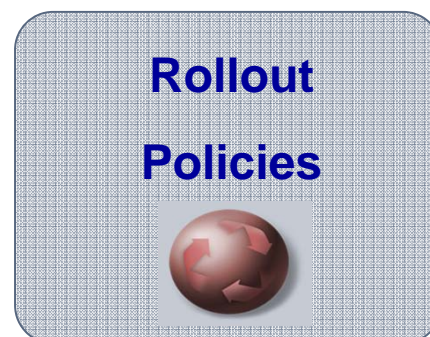
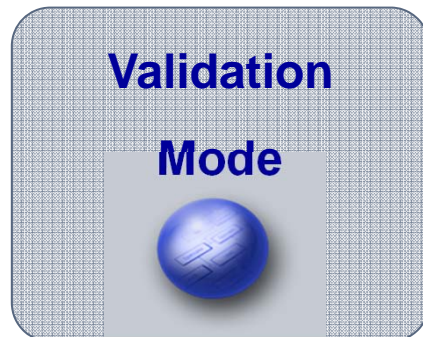
**Intelligent Routing**



# 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**

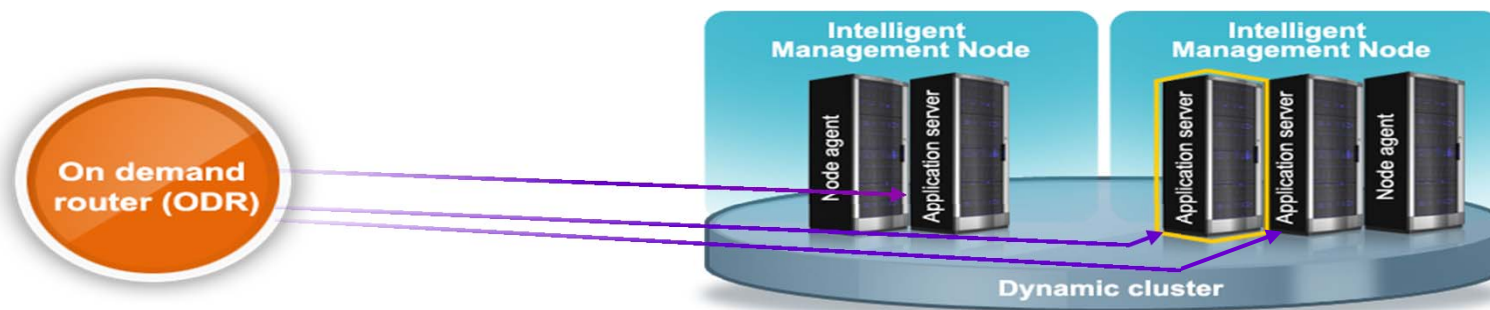




# Dynamic Clustering

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

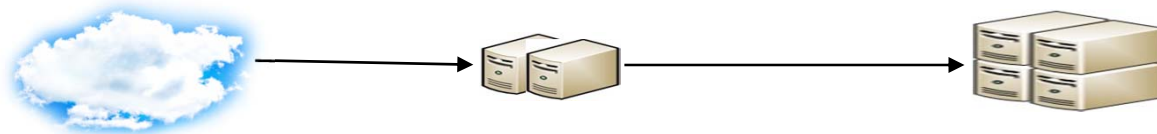
- Associate service policies with your applications
  - Let WebSphere manage to the service goals
- Programmatically respond to spikes in demand
  - Add or reduce application server instances as appropriate
- Automatically recover from infrastructure problems
- Includes automatic start and stop of cluster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



# Intelligent Routing

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

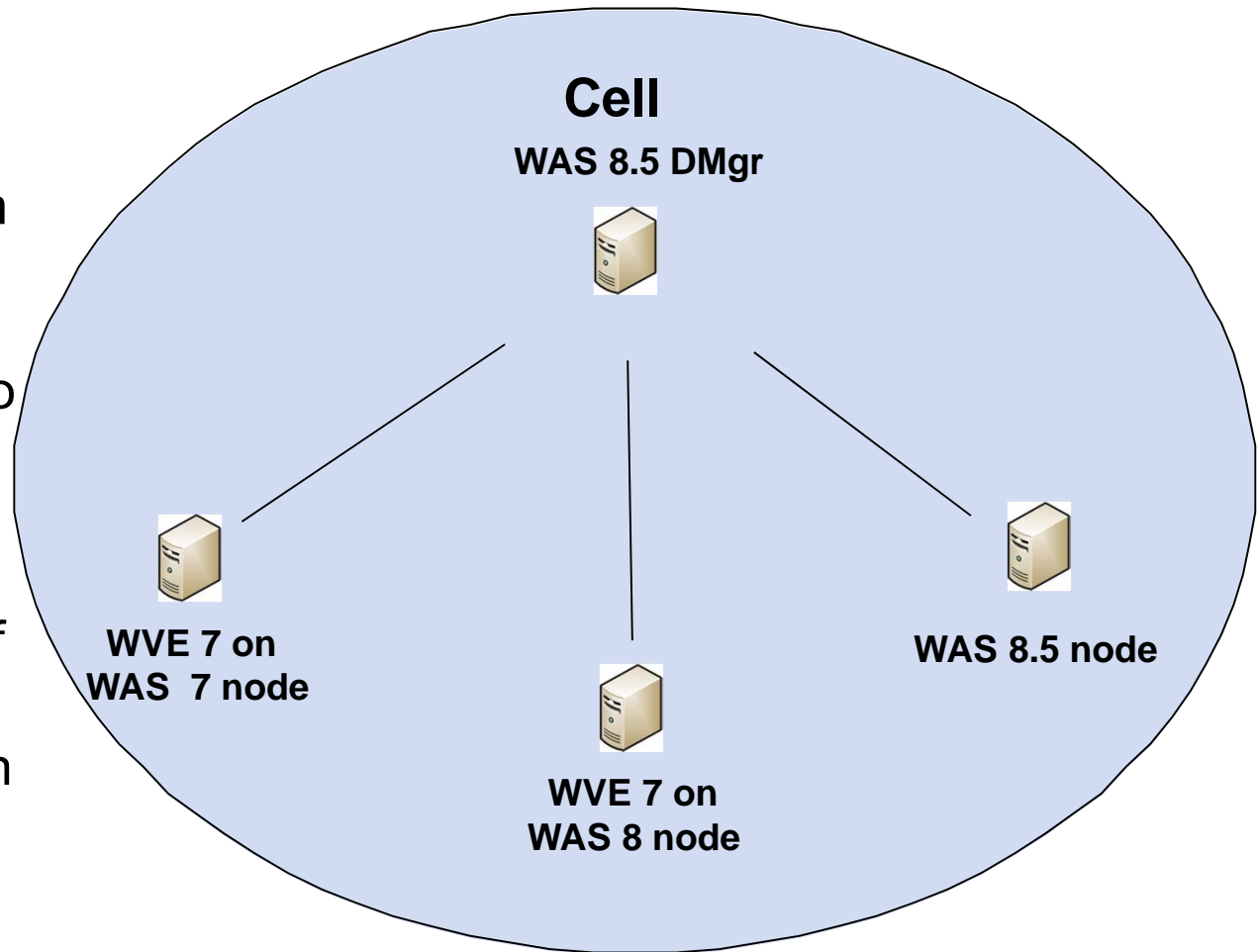
- Requests are prioritized and routed based upon administrator defined rules
  - Flexible policy-based routing and control
- On Demand Router (ODR) is the focal point for Intelligent Routing
- A routing tier that's aware of what's happening on the application server tier
  - Application server utilization, 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



# Migrating WebSphere Virtual Enterprise to WebSphere Application Server ND V8.5



- Migrate WVE configuration data from an existing WVE V7 installation (on either WAS V7 or WAS V8) to a WAS ND V8.5 installation
- Support coexistence of WVE V7 nodes (on WAS V7 or WAS V8) in a partially migrated WAS ND V8.5 cell





# WebSphere Batch – Yesterday & Today

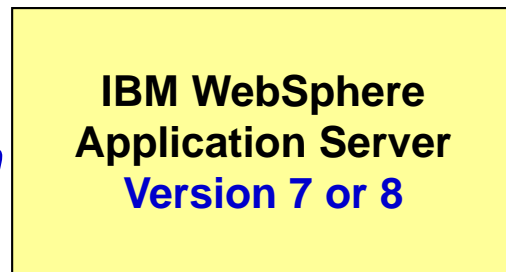
The WebSphere Java Batch function is provided in two ways today:

**Java  
Batch  
Function**



↓ Add the function ("Augment")

**Java  
Execution  
Runtime**



Operating Systems Supported:  
AIX, IBM i, Linux, Windows, HP-UX,  
Solaris, Linux for System z, z/OS

*Compute Grid V8 function  
incorporated into WAS V8.5*



Operating Systems Supported:  
AIX, IBM i, Linux, Windows, HP-UX,  
Solaris, Linux for System z, z/OS

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



# Migration from V7 to V8.x: Considerations

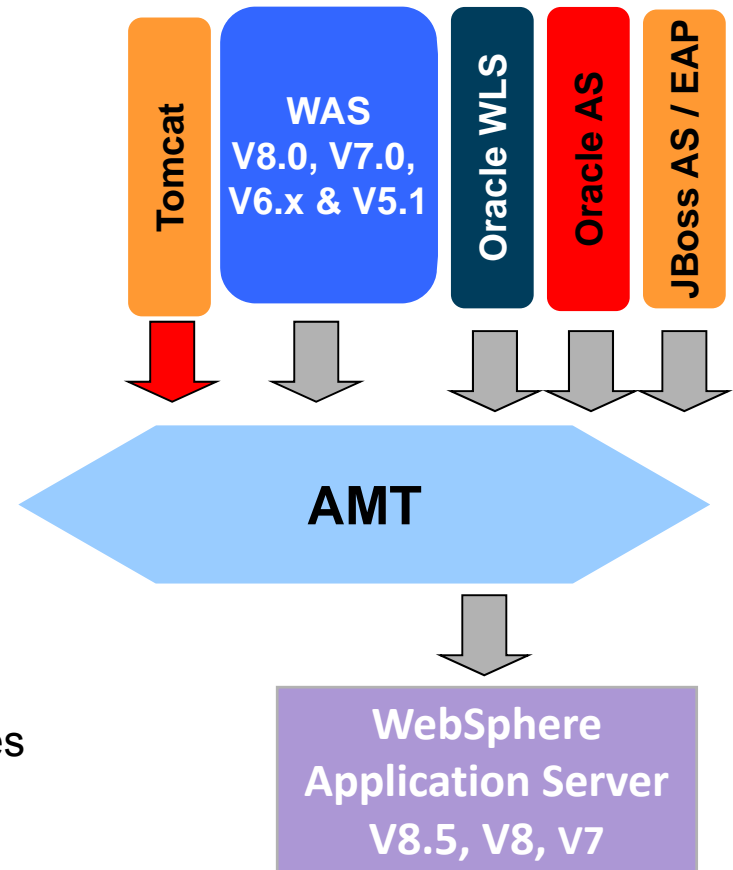
- WAS V8.0 and WAS V8.5 have the same Java EE 6 and same default Java SE 6
  - WAS V8.5 additionally supports Java SE 7
- Same Installation Manager, administrative infrastructure and security defaults.
- WAS V8.5 has significant additive features including:
  - Liberty Profile
  - Intelligent Management
- 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



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

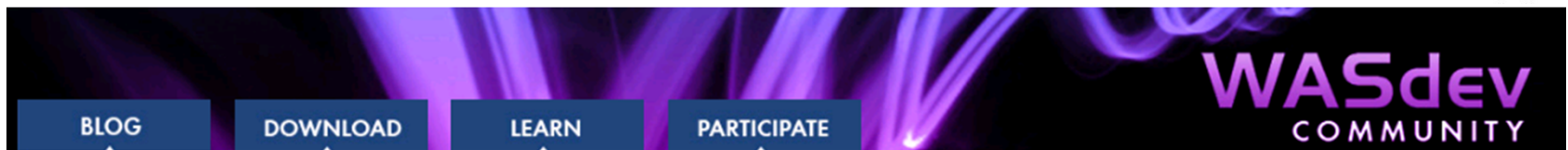
- Migrate apps from older vers to WAS V8.5, V8, V7
- Migrate from Oracle, TomCat, JBoss faster & easier
  - Migrate applications up to 2x as fast
  - Migrate web services up to 3x as fast
- Application Migration Tool
  - Analyzes source code to find potential migration problems:
    - Removed & deprecated 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>

# Looking Ahead: WAS V8.5.Next

## More Power to the Lightweight Profile

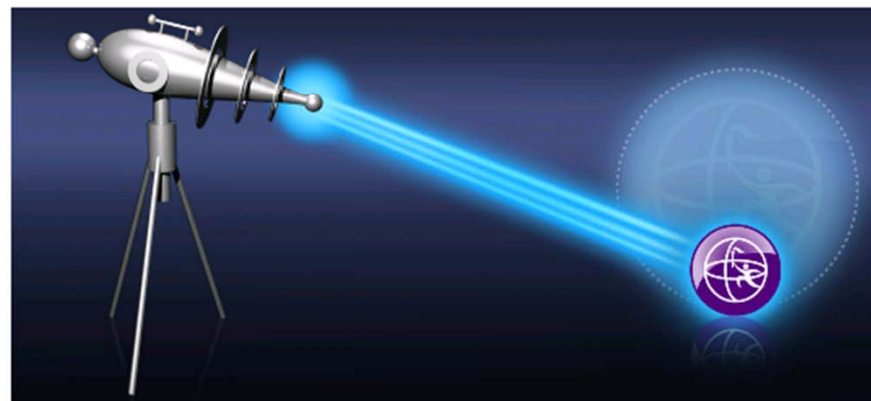


### Welcome to WASdev

The WASdev community is the place for developers to discuss, share and learn about developing applications that will run on WebSphere Application Server.

★ 14

**Roll up, roll up! Come see the amazing shrinking server - Available with the new Betas for WebSphere Application Server, Rational Application Developer and WAS Developer Tools now!!!**



#### Featured - New in March 2013

- Refresh of the Betas for WebSphere Application Server, Developer Tools, and RAD!
- New and Noteworthy in the new Beta refreshes
- Updates to the Liberty Profile Plugin for IBM Workload Deployer!

#### Participate

Connect.  
WASdev forum

- Join our developers as they demonstrate new programming support features in the Beta. Click for details.

Following on from WAS V8.5, shipped in June 2012, the WAS V8.5.Next Beta looks ahead at what's currently in development.

Get the details and the downloads from <http://wasdev.net>





# WAS 8.5.Next Beta



- New Liberty capabilities (all editions)
  - Java EE 6 Web Profile compliant in 50MB!
  - New “Minify” option on server **package** command to reduce to the specific set of features used for a server configuration
  - Extend runtime with third-party feature SPI
  - Custom User Registry
  - Install through Installation Manager or “archive install” (same as WAS V8.5.0)
  - Additional Liberty features include:
    - JMS
      - Embedded Liberty JMS Messaging Provider (SIB) or MQ JMS
    - JAX-WS
      - Including WS-Security
    - MongoDB NoSQL DB
- WAS ND
  - Lightweight “Collective” management of Liberty clusters



**Development  
features**

**Operations  
& Control**

# Lightweight Web Profile Completion

Liberty



- Proper subset of Java EE specification for web application on Liberty profile
- Supported by all WAS editions
- Web profile specification: **<feature>webProfile-6.0</feature> enables all**
  - Servlet 3.0
  - JavaServer Pages (JSP) 2.2
  - Expression Language (EL) 2.2
  - Debugging Support for Other Languages (JSR-45) 1.0
  - Standard Tag Library for JavaServer Pages (JSTL) 1.2
  - JavaServer Faces (JSF) 2.0
  - Common Annotations for Java Platform (JSR-250) 1.1
  - Java Transaction API (JTA) 1.1
  - Java Persistence API (JPA) 2.0
  - Bean Validation 1.0
  - **Enterprise JavaBeans (EJB) 3.1 Lite + Interceptors 1.1:**
    - **<feature>ejbLite-3.1</feature>**
  - **Managed Beans 1.0: <feature>managedBeans-1.0</feature>**
  - **Context Dependency Injection: JSR-299 1.0 and JSR-330 1.0:**
    - **<feature>cdi-1.0</feature>**

V8.5.Next

V8.5.Next

V8.5.Next



# EJB Lite vs Full EJB

	EJB 3.1 Lite	full EJB 3.1 API
Session Beans (stateful, stateless, singleton)	✓	✓
message driven beans		✓
2.x/1.x CMP/BMP Entity		✓
Java Persistence 2.0	✓	✓
Local/No Interface	✓	✓
3.0 Remote		✓
2.x Remote Home/Component		✓
JAX-WS web services		✓
JAX-RPC web services		✓
EJB Timer Service		✓
Async Bean Invocation		✓
Interceptors	✓	✓
RMI-IIOP interop		✓
Contained managed transactions/Bean managed transactions	✓	✓
Declarative Programmatic Security	✓	✓
Embeddable API	✓	✓ (only required to support EJB Lite)

# Messaging in WAS Liberty Profile

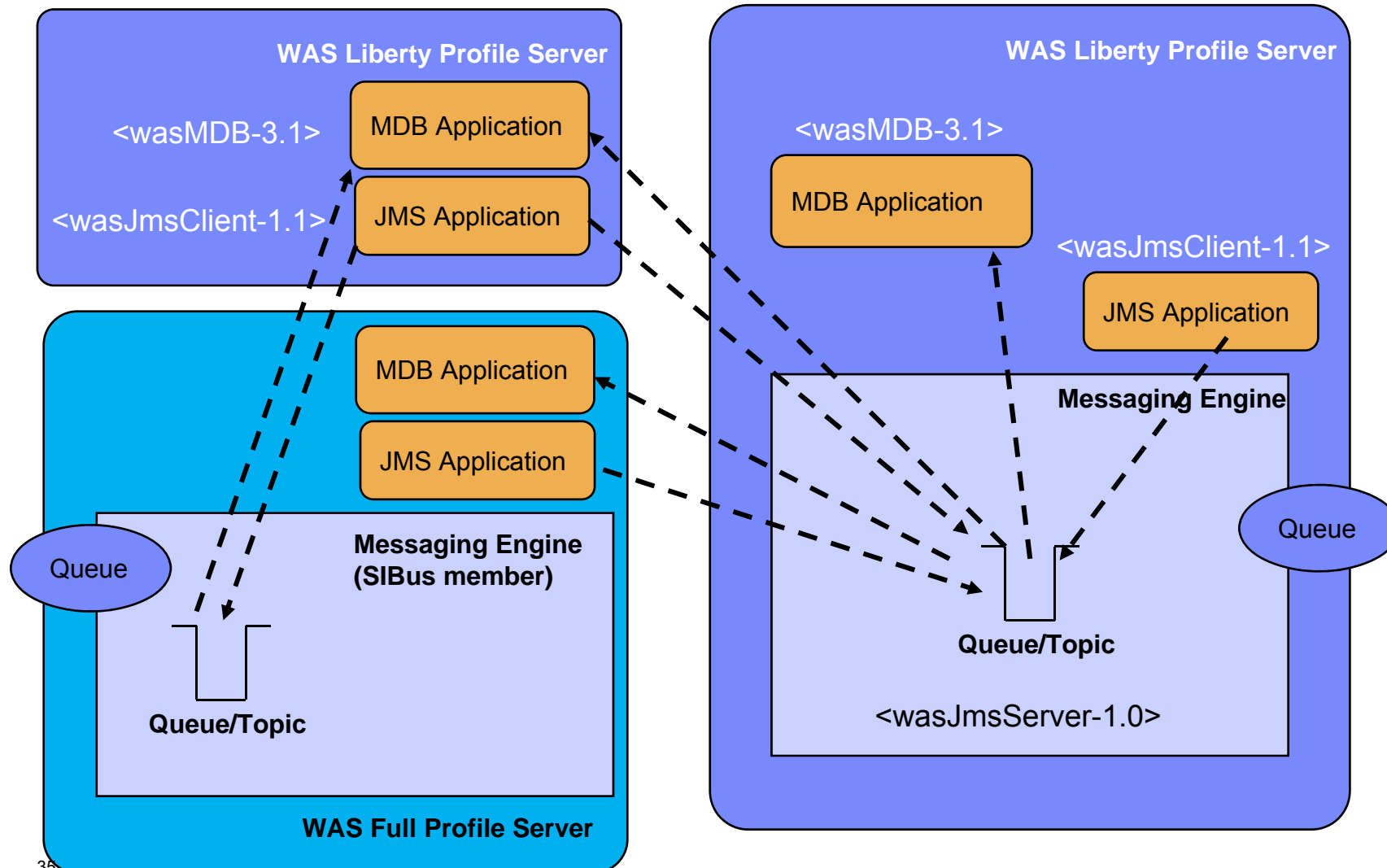
Liberty



- JMS and MDB programming models features:
  - `<feature>jmsMDB-3.1</feature>`
  - `<feature>wasJmsClient-1.1</feature>`
- Lightweight messaging engine localizes messaging destinations (queues and topics)
  - Integrates with transactions and security services
  - `<feature>wasJmsServer-1.0</feature>`
  - `<feature>wasJmsSecurity-1.0</feature>`
- Inter-operates with WAS full profile SIBus
  - Liberty client to full WAS SIBus
  - Full WAS JMS client to Liberty
- MQ Client:
  - `<feature>wmqJmsClient-1.1</feature>`

# Liberty JMS Features

Liberty WAS JMS messaging: all messages are sent and received from a localized ME.  
For Messaging Provider HA connect to WMQ or WAS full profile SIBus.



# Web Services in Liberty

Liberty



- `<feature>jaxws-2.2</feature>`
- Supported APIs
  - JAX-WS 2.2
  - JAXB 2.2
  - Web Services for EE 1.3
  - Both POJO and EJB based Web services
  - JDK bundled APIs also available: SAAJ, JAXP, StAX
- Protocols
  - SOAP/HTTP
  - WS-Security
- Simplified development and deployment
  - Eclipse WDT wizard supports top down WSDL to Java generation
  - Eclipse WDT plugin supports rapid edit-compile-debug cycles



# Security and Web services

Liberty



- Without WS-Security
  - BASIC AUTH
  - Web application transport constraints
  - Roles allowed
- WS-Security
  - `<feature>wsSecurity-1.1</feature>`
  - Uses default keystore and truststore
  - WS-Security policies
    - Username Token Profile 1.1
    - X.509 Token Profile 1.1
    - WS-I Basic Security Profile 1.1
  - Simplified development and deployment
    - Policy embedded in WSDL
    - Eclipse WDT wizard simplifies patterns for adding policies to WSDL

# MongoDB



- MongoDB is an open source highly scalable noSQL database
  - Document-centric unstructured data
  - Simple API for inserting and retrieving documents.
- Configure Liberty server to point to MongoDB Server
  - `<feature>mongodb-2.0</feature>`
- Programming model supports:
  - J2SE style
  - Lookup through JNDI
  - Resource Injection

# Asynchronous Work Management

Liberty



- Extends Java SE `java.util.concurrent.ExecutorService` to allow applications to perform work asynchronously, while retaining Java EE context of the calling thread.
  - Evolution of existing WAS full-profile support
  - New capabilities common to full WAS and Liberty profiles.
- Access `ExecutorService` through
  - JNDI
  - Resource injection
- Server configuration defines the JNDI name and the contexts to propagate, e.g., security, application env.
  - For Liberty: `<feature>concurrent-1.0</feature>`

```
<managedExecutorService jndiName="concurrent/execSvc1">
  <contextService>
    <!-- One or more of the following -->
    <jeeMetadataContext/>
    <classloaderContext/>
    <securityContext/>
    <zosWLMContext/>
  </contextService>
</managedExecutorService>
```



Development  
features

**Operations  
& Control**

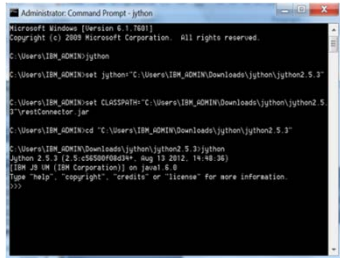
# Liberty “Collective” Administration



- New Liberty collective management infrastructure
  - Lightweight centralized management of Liberty profile servers
  - Agent-less administration
  - Deployment of topology of WAS Liberty profile servers
  - Static cluster management
  - Application and server deployment/update
  - JMX API through Java, Jython, and Jconsole clients
- Configuration:
  - Controller (requires WAS ND or WAS z/OS license):
    - `<feature>collectiveController-1.0</feature>`
    - Highly available
  - Managed server:
    - `<feature>collectiveMember-1.0</feature>` (All Editions)
    - `<feature>clusterMember-1.0</feature>` (ND+ only)
- Scales to 1000s of JVMs

# Liberty Collectives (WAS ND)

Standard, remote, examples



# Some Terminology



Name	Description
1. Liberty Collective	<ul style="list-style-type: none"><li>• Multi-server admin domain</li><li>• Loosely coupled, Highly available</li></ul>
2. Liberty Clusters	<ul style="list-style-type: none"><li>• Support for application server clusters</li><li>• Multiple cluster operations provided</li></ul>
3. MBean Routing	<ul style="list-style-type: none"><li>• Standards based admin operations</li><li>• Operate on any server standalone or collective</li></ul>
4. File Transfer	<ul style="list-style-type: none"><li>• Remote access to Liberty Profile files</li><li>• Supported for both Collective and Standalone</li></ul>
5. Jython Scripting	<ul style="list-style-type: none"><li>• Admin automation enablement, Mbean based</li><li>• Supported for both Collective and Standalone</li></ul>
6. JConsole Support	<ul style="list-style-type: none"><li>• Standards based console</li><li>• Supported for both Collective and Standalone</li></ul>



# MBeans for Collective-based Admin

Liberty



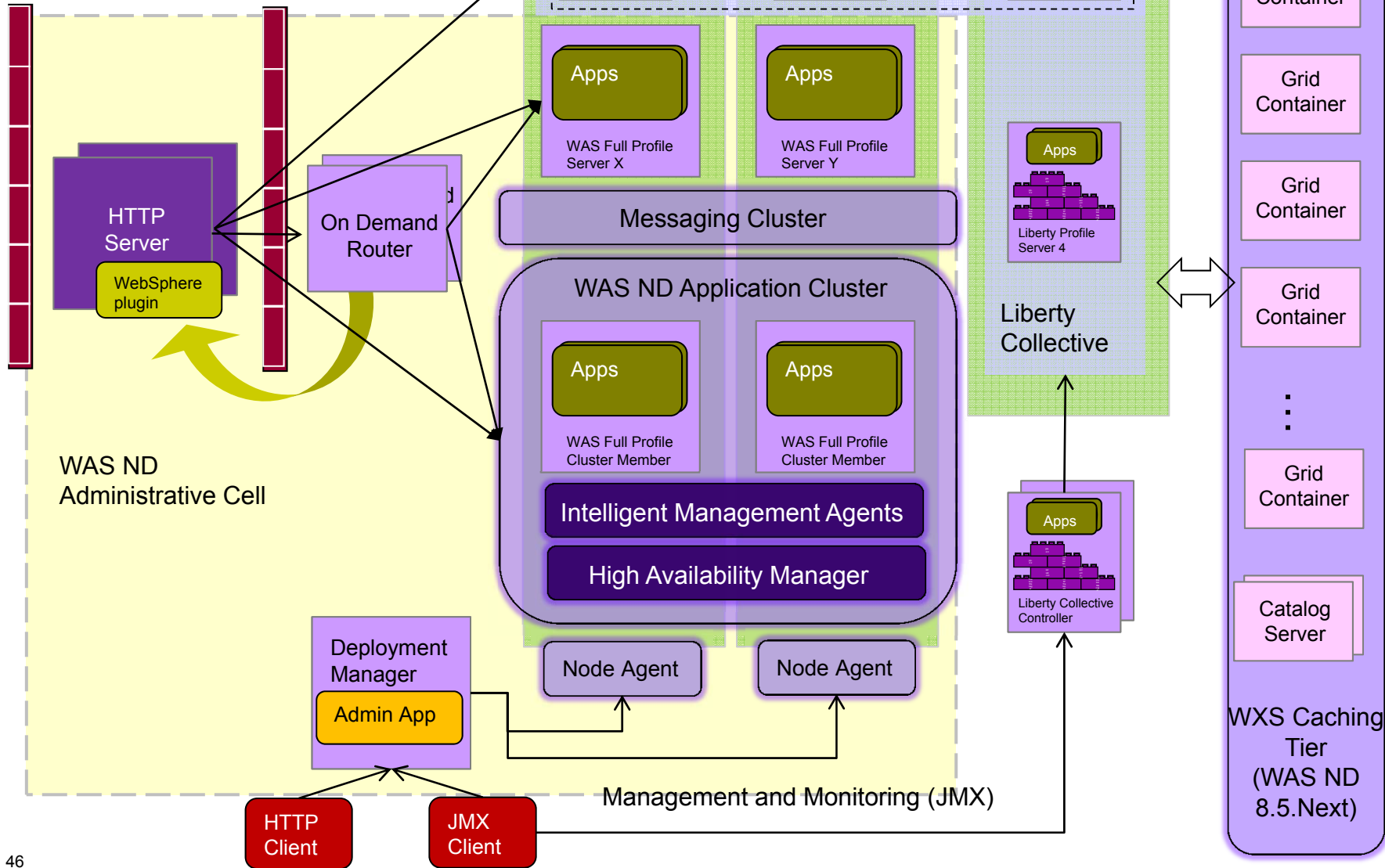
- **ServerCommand**
  - Start/stop server operation
  - Server started/stopped notifications
  - Runs in collective controller
- **ClusterManager**
  - Cluster ops: start,stop,listCluster,listMembers,generatePluginConfig
  - Runs in collective controller
- **RoutingContext**
  - Sets routing path for MBean operations routed through collective controller
- **FileTransfer**
  - Provides upload/download support for single server and server in collective when combined with RoutingContext Mbean.
- **FileService**
  - Provides operations for file manipulation, including reading meta data and archive operations (e.g. create, expand)



# Collective Command

- Use to build/modify collective
- wlp/bin/collective[.bat]
- Supports
  - [create] collective
  - [join] server to collective
  - [remove] server from collective
  - [replicate] add another controller to replica set
  - [registerHost] add a host to collective
  - [unregisterHost] remove host from collective
  - [updateHost] update host info in collective

# WAS V8.5.Next High Level Architecture



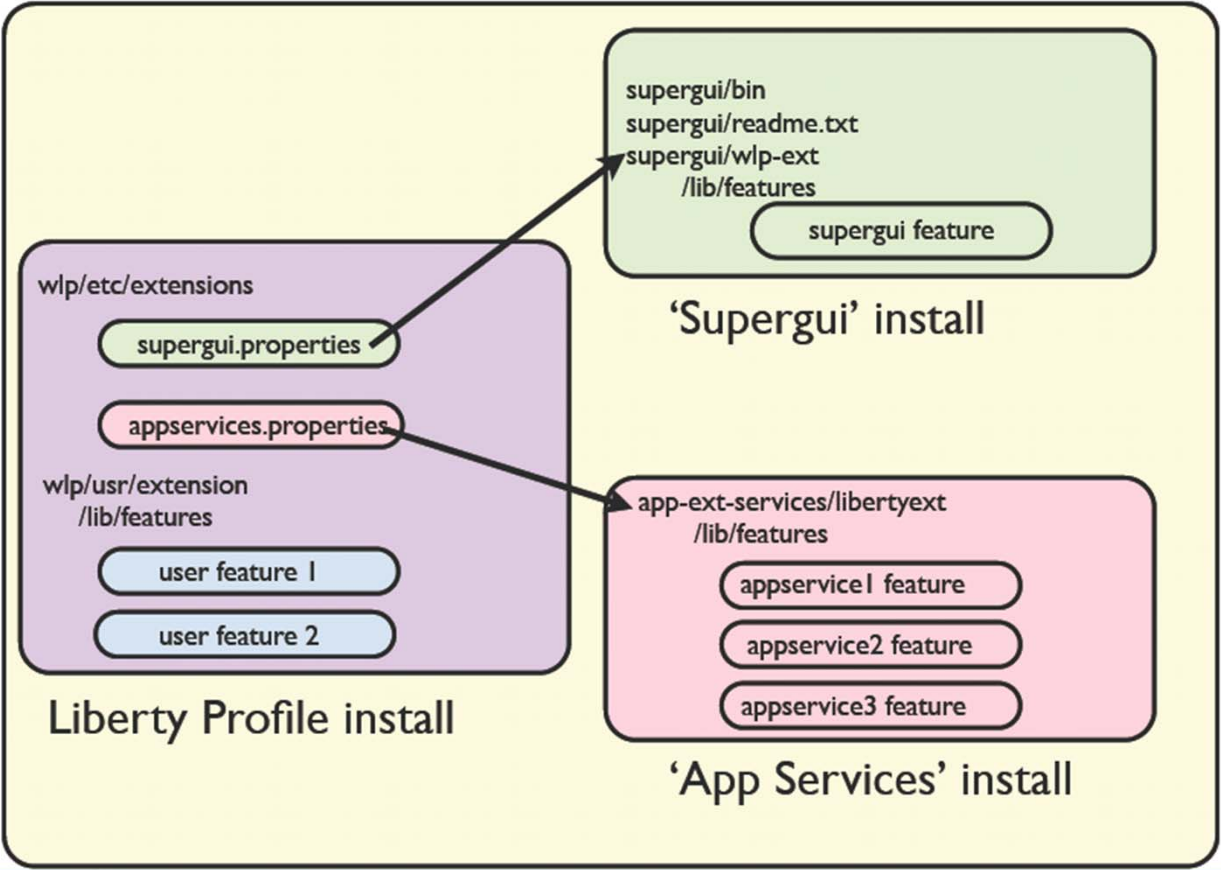
# Third Party Extensibility

Liberty



- Supports third party extension of the runtime
- 3rd party feature life cycle
  - ▶ Package and install
  - ▶ Extend configuration
  - ▶ Integrate with runtime
- WDT project type for feature development
- A product extension is a directory on disk structured like the wlp dir
- All content for a feature is relative to the extension location the feature is installed to
- Registered in wlp/etc/extensions
  - ▶ One file per extension

```
com.ibm.websphere.productId=<your product id>
com.ibm.websphere.productInstall=<absolute, or relative file path>
```





## SPIs to Extend the Runtime

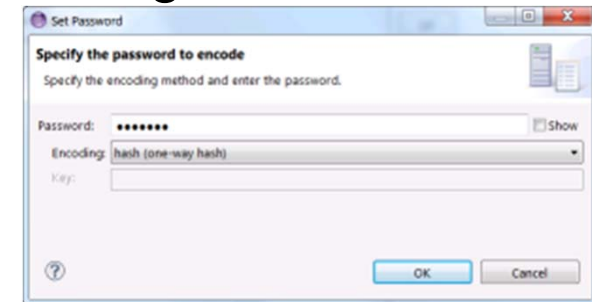
- Full life cycle: packaging, installation, and un-installation
- Service Programming Interface (SPI) for:
  - Adding your own configuration
  - Interaction with Liberty kernel
- Class visibility protected by class loader
- New command:  
**wlp/bin/featureManager** *action* [options]
  - ***action*** can take one of the following values:
    - install** featureLocation (can be filename or URL)
    - featureList** Generate an XML file containing a report detailing all the features that are installed.
  - **options** include
    - to=usr**
    - to=external\_extension\_location**

# Security enhancements for Liberty Servers

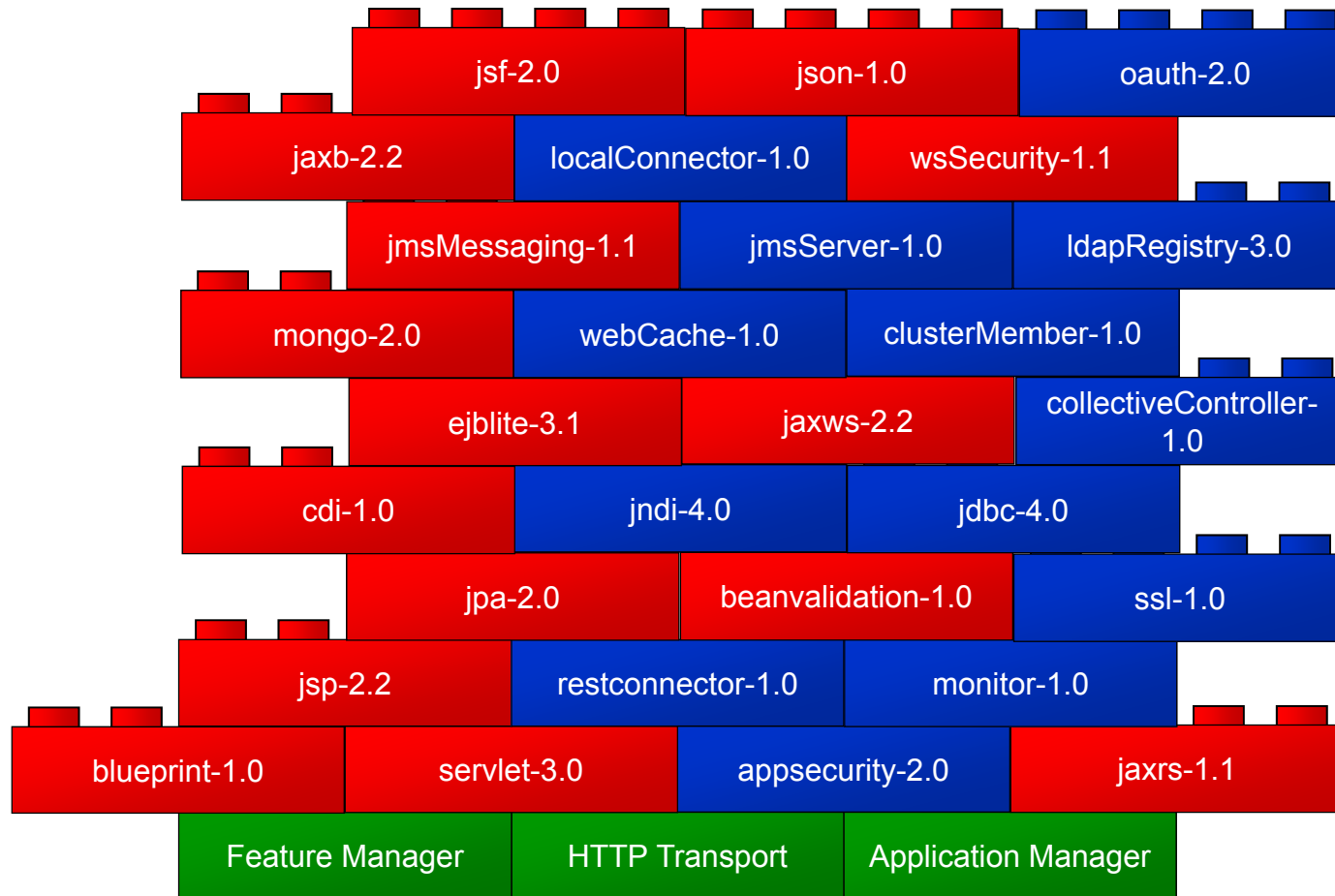
Liberty



- Federated LDAP Repositories
  - Supports federation of multiple LDAP repositories
- Custom User Registry
  - User provided alternate user registry
  - Installed as an extension to Liberty runtime
- Additional options for Password encryption in configuration files (cmd line util and WDT)
  - Advanced Encryption Standard (AES) encryption added
  - XOR hashing (since 8.5.0)
- OAuth 2.0 (V8.5.0.2)
  - OAuth Service Provider (SP) to handle OAuth 2.0 protocol requests.
  - <oauth-2.0> feature starts built-in OAuth SP web app.
  - Customizable form login to get access to resources
  - OAuthProvider clients defined in server.xml

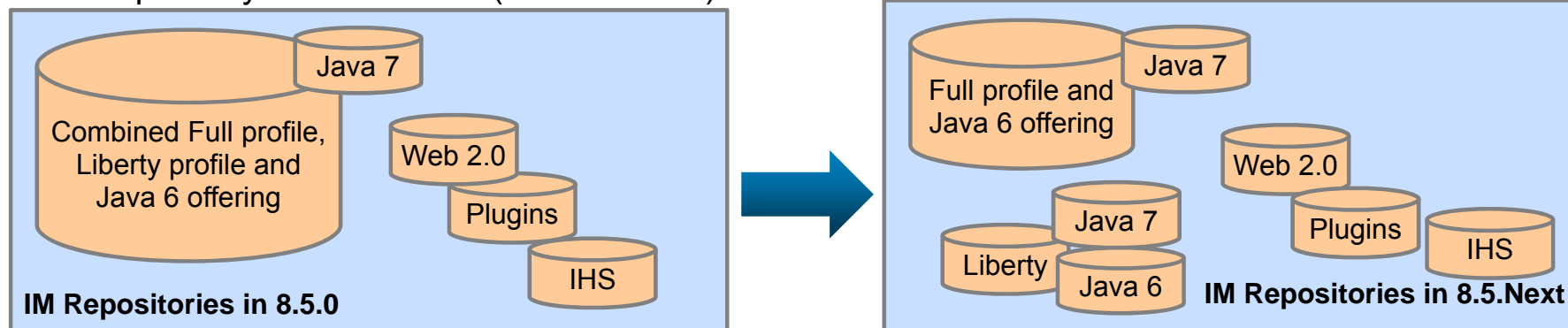


# Dynamically composable runtime features



# Larger Menu, Same Size Trousers

- The runtime is only as large as the set of features configured
- The product **install** is now more modular too:
  - Archive install is split into:  
Core (web profile) = 50MB  
Extended (JAX-WS, JMS and everything else) = 30MB
  - Installation Manager has same split and also separates Liberty profile repository apart from WAS full-profile repository.
- 8.5.Next IM Repository for offline IM install of Liberty is 105M compared to combined repository size of 1.5GB (~12x smaller).

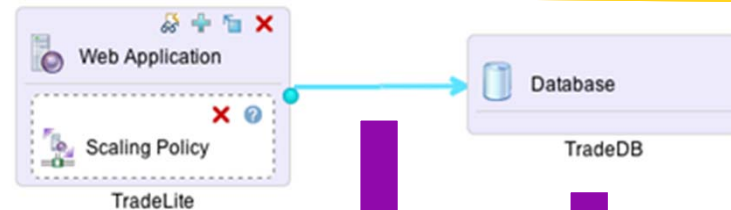


- And remember: `wlp/bin server -package` produces as “packaged server”?
  - Now with new **--include=minify**  
To automatically skinny-down a packaged server by stripping out all the runtime features not referenced in the configuration.



# Tech Preview: High Density IWD Web Application Pattern

Virtual Web App  
Pattern Type

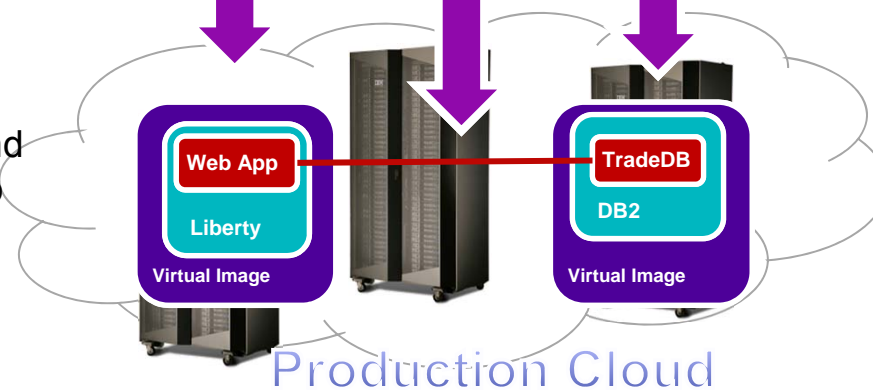


Same web application that runs on WAS Liberty Profile...deployed on IWD/PAS PaaS abstraction of the middleware

Plugins used by each  
element of the Web  
Application Pattern Type

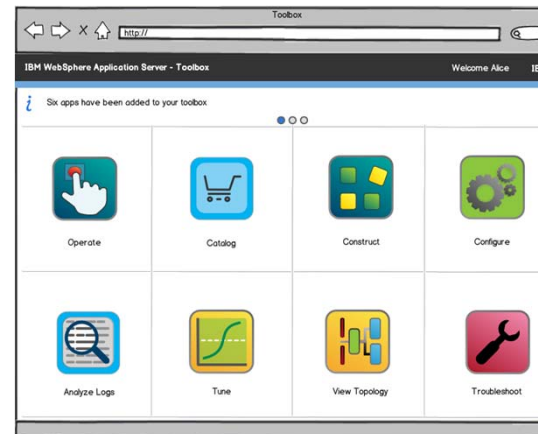
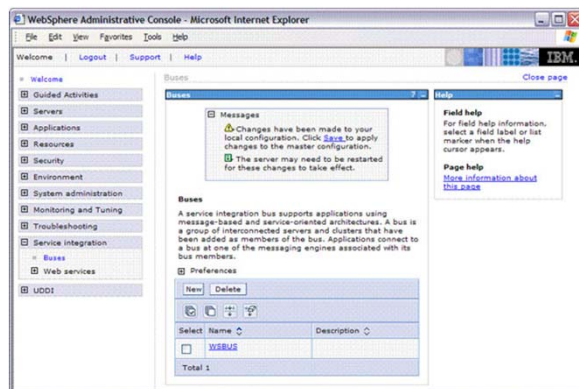


Virtual Images set up and  
products installed (unzip  
+ install operations)



# Liberty Profile Admin UI Tech Preview

- Tech preview of NextGen console
  - Early feedback to guide GUI direction
- Evolve from one-size-fits-all console
- Lightweight, task-oriented applications / app store approach with customizable toolbox for “right-sized” UI per user / device
- GUI function
  - view/start/stop liberty applications





# Get The Betas!

- ...from <http://WASdev.net>
- Open program – no sign up required but feedback welcome (see the “Participate” page).
- Optionally join the Feature Focus weeks to learn more.
  - These sessions require registration in the WebSphere Customer Experience Program (CEP). If you are not already enrolled, please email us at [cep@us.ibm.com](mailto:cep@us.ibm.com) requesting to join the program as soon as possible.
- Past and present sessions:

Upcoming Beta Sessions	
Date	Topic
Feb. 27	WebSphere Application Server and Tooling Beta Overview
Mar. 13	Programming Language: Whats new in the Beta
Mar. 27	Liberty Administration and Security



**Thank  
You**