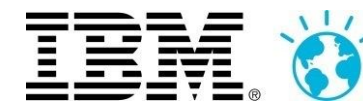




# **IBM Technical Summit 2013**

Démarquez-vous

17 octobre | IBM Client Center Paris



# IBM Technical Summit 2013

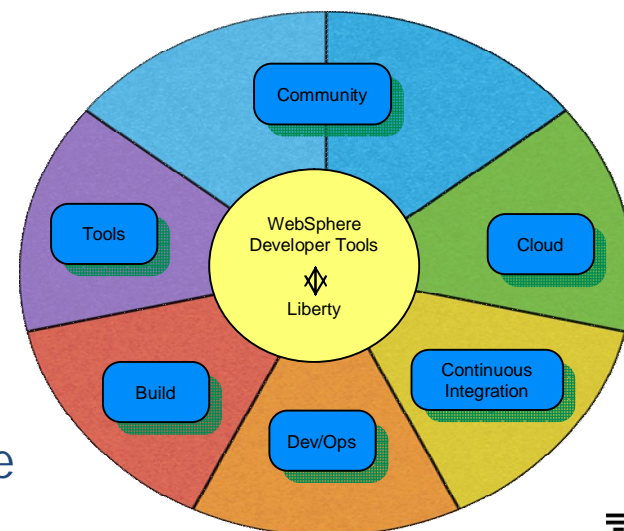
Démarquez-vous

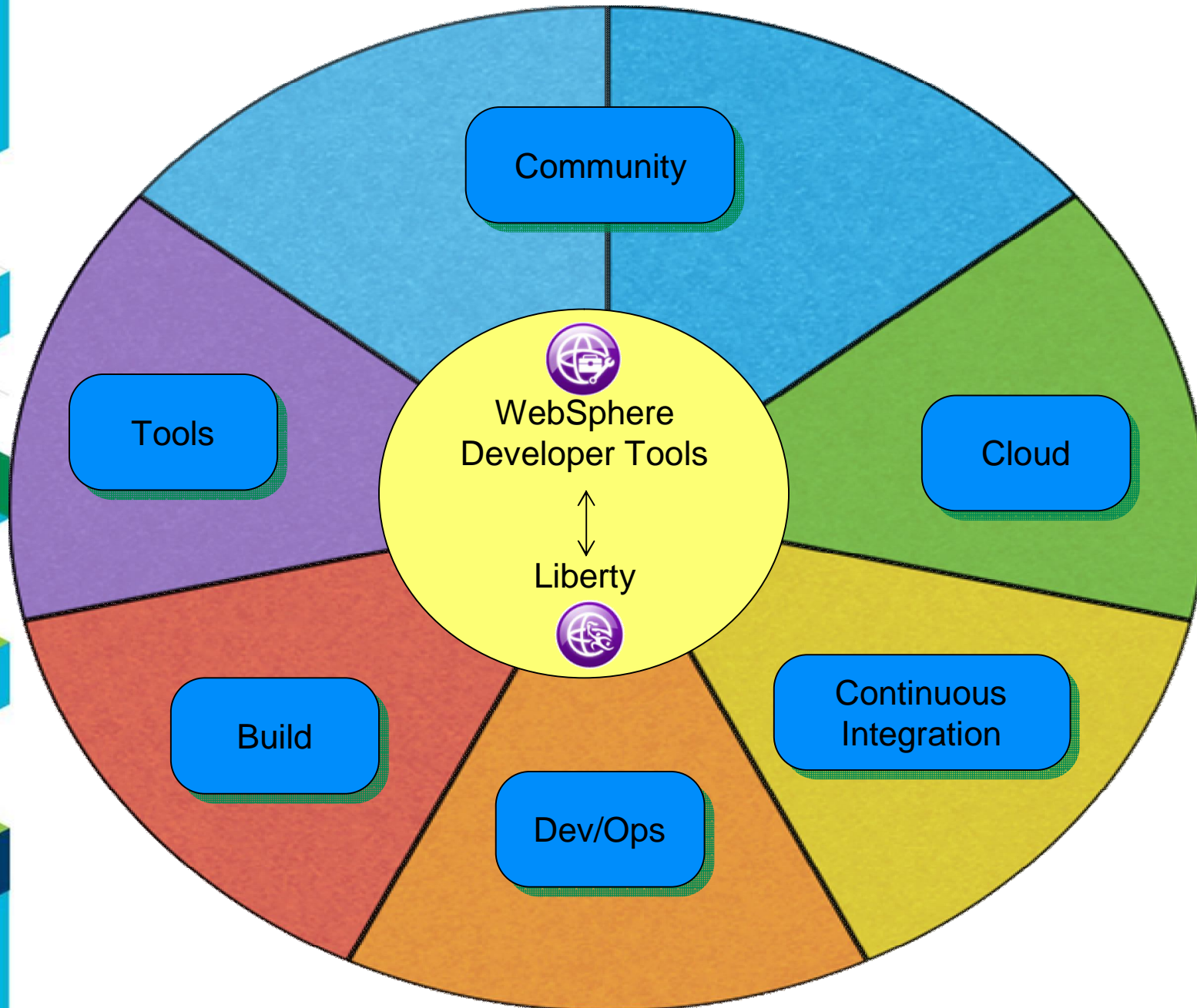
17 octobre | IBM Client Center Paris

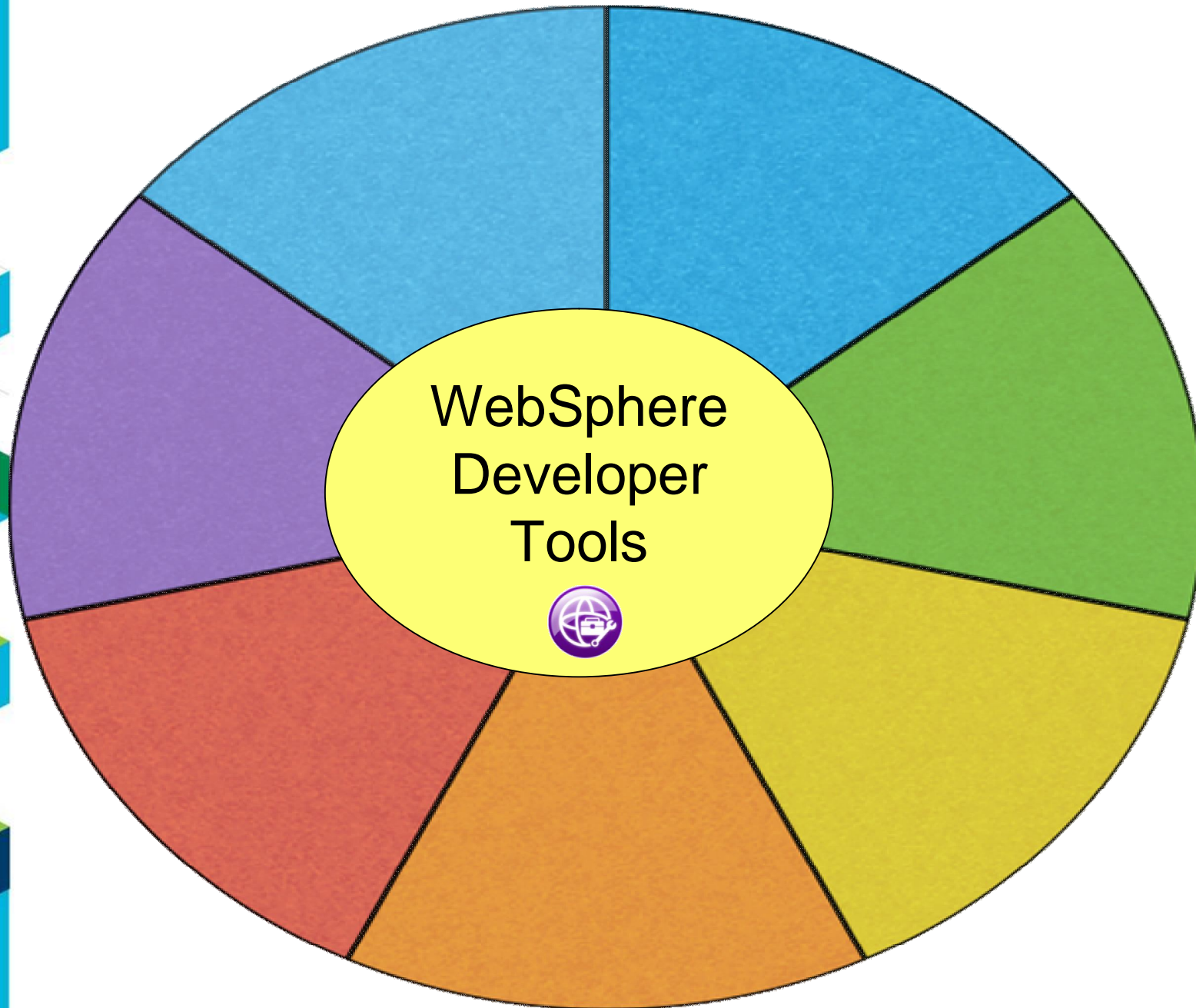
## The WebSphere Liberty Development Environment:

**WebSphere Application Server Developer Tools**  
**WebSphere Liberty Profile**  
**Liberty Ecosystem**

Iain Duncan,  
Software Engineer,  
WebSphere Application Server Liberty Profile







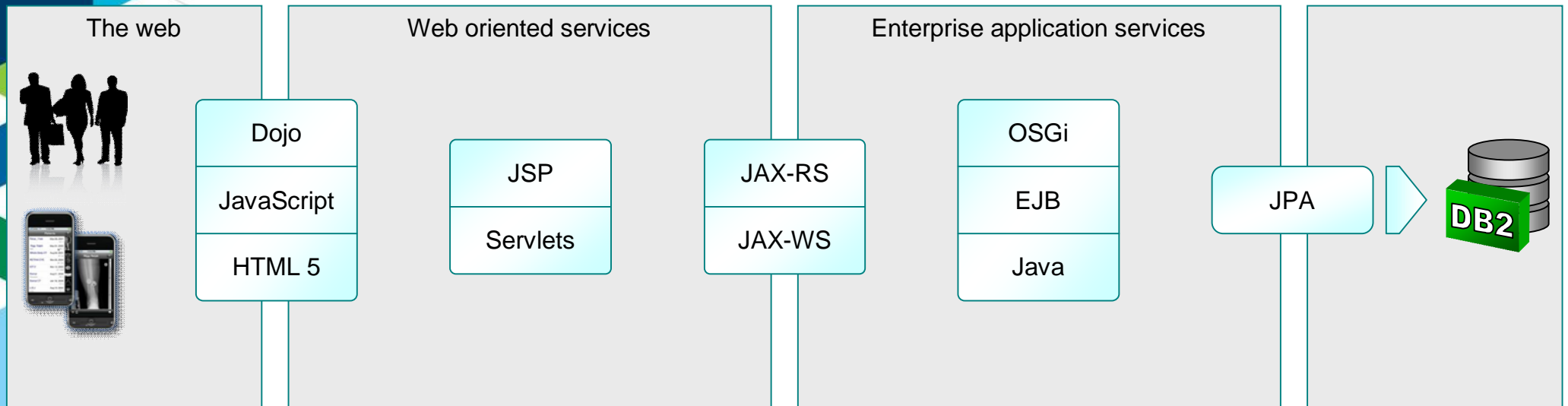


# Unleash innovation with WDT 8.5.5

*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*



# Tools for the Developer Desktop

## RAD

Install: IM

### WebSphere Integration

- Support for WAS v6.1
- Test Environments for WAS v7.0, v8.0, v8.5
- Portal Tools / Portal Svr supt
- Profile applications on WAS
- Deploy to WebSphere or Portal instances in IBM SBDT cloud
- IWD 3.1, 3.0

### Problem Determination

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

### Team Productivity

- RTC integration
- Collaborative debug
- Collaborative code analysis

### Extended Programming Model Support

- Advanced support for J2EE 1.4 and earlier:
  - EJB & Web Services deploy
  - DD editors
  - JAX-RPC
- Web:
  - Page designer
  - Struts, JSF support
  - iWidget support

### Enterprise Connectivity

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

### Programming Model Support

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

## WDT

Install: Eclipse update site

### 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, v8.5.5
- 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 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, possibly with exception of WebSphere Integration

# Introducing the new Rich Page Editor

The screenshot displays the Eclipse IDE with the following components:

- Enterprise Explorer:** Shows a project structure with folders like 'themes' and 'themeTesterImages', and files like 'themeTester.html'.
- Outline:** Shows a hierarchical view of the Dojo widget tree, including 'ItemFileReadStore', 'Menu', 'BorderContainer', and 'TabContainer'.
- Design View:** The main editor area showing a rendered page with a calendar, buttons, checkboxes, and radio buttons. A toolbar at the top of this view includes 'File', 'Edit', 'View', 'Themes', 'Dialogs', 'TextBox Padding', 'Help', 'Disabled', and 'Click me!'. A 'Device: Standard' dropdown is also present.
- Properties:** A vertical panel on the right showing the 'Properties' view for the selected widget, with sections for 'Class', 'Properties', 'Font', 'Format', 'Alignment', and 'Color'.
- Widget Palette:** A panel on the right containing various Dojo widgets such as 'Button', 'CheckBox', 'ComboBox', 'Form', and 'Radio Button'.
- Code Editor:** The bottom view showing the source code for the widgets, including HTML and JavaScript snippets.

High fidelity rendering using browser engine

Multiple browser preview

Precision drag-and-drop from palette

Simplified Dojo Outline view

Vertical Properties view for maximum editing area

# Mobile Browser Simulator

The screenshot displays the Mobile Browser Simulator application. At the top, the browser address bar shows the URL: `http://localhost:8080/_MobileBrowserSimulator/index.html?webpage=http://localhost:8080/apps/services/preview/dojo/iphone/1.0/d`. Below the browser, the simulator interface includes a control bar with options like "Add Device", "Scale All Devices: 100%", and "Enable Useragent Switching". On the left, a sidebar lists various device services such as "Accelerometer", "Battery", "Camera", and "Compass". The main area features two emulated devices: an "Apple iPhone 3GS" and an "Apple iPhone 4 and 4s". Both devices display a mobile web page with a navigation menu containing "IBM Innovate", "Mobile", and "Jazz" items. Three blue callout boxes are overlaid on the image, pointing to specific features of the simulator.

Test device services

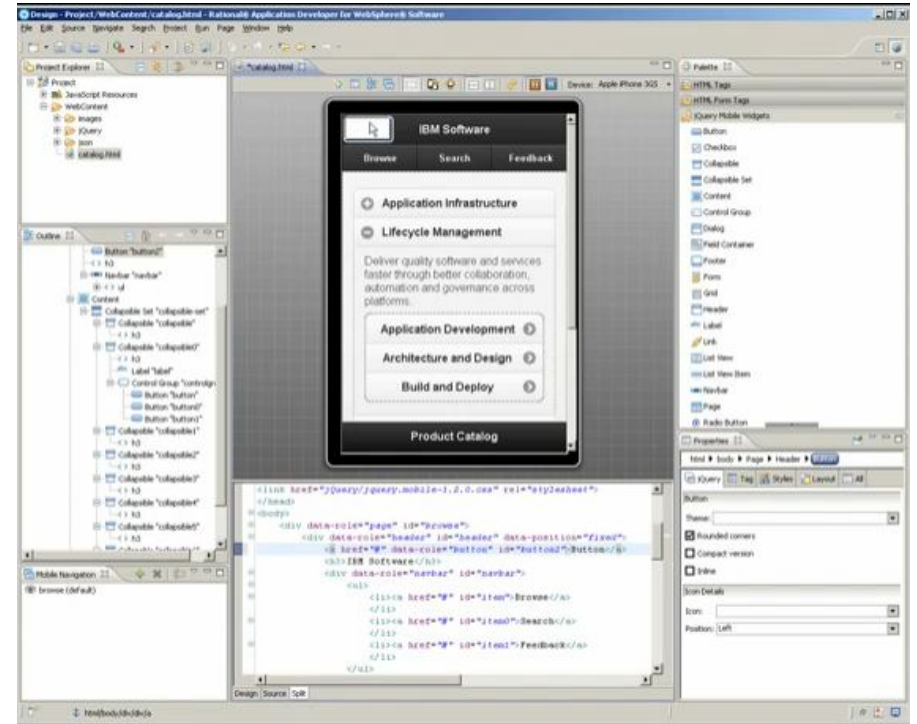
Hi-fi rendering of the device UI in the browser

Multiple device form-factors



# jQuery Mobile Development

- Unified editing experience through Rich Page Editor
- Instant graphical feedback for jQuery selectors using “jQuery Smart Highlights”
- Over 20 reusable & extensible mobile web patterns to speed up development
- Drag & drop and visualize mobile widgets from the palette
- Content assist for jQuery API in source editor
- New wizards for jQuery Plugins & widgets creation
- In-editor “device” rotation and scaling



Mobile web patterns



# Remote web code debugging *with desktop browser inside the workbench*

The screenshot displays the Eclipse IDE interface during a remote web code debugging session. The main window is titled "Debug - static/WebContent/dojo/dojo/dojo.js - Eclipse".

**Debug Console:** Shows the execution stack for a thread [xf0.3::1057864 - http://localhost:10546/static/index.html] (suspended). The stack includes:

- dojo.js contextRequire() line: 688
- dojo.js result() line: 741
- has.js (?)() line: 171
- dojo.js injectPlugin() line: 1231
- dojo.js (?)() line: 1053
- dojo.js forEach() line: 93
- dojo.js resolvePluginLoadQ() line: 1045
- dojo.js finishExec() line: 1087
- dojo.js execModule() line: 1147
- dojo.js checkComplete() line: 1160

**Variables Window:** Displays a table with columns "Name" and "Value".

Name	Value
------	-------

**Code Editor:** Shows the source code of `dojo.js` with a breakpoint set at line 688. The code snippet is:

```
}  
if(isArray(a1)){  
    // signature is (requestList [,callback])  
    if(!a1.length){  
        a2 && a2();  
    }else{  
        syntheticMid = "require*" + uid();  
  
        // resolve the request list with respect to the reference  
        for(var mid, deps = [], i = 0; i < a1.length;){  
            mid = a1[i++];  
        }  
    }  
}
```

**Console:** Shows the command prompt for the Debug Firebug Launch Configuration, with the path `C:\Program Files\Mozilla Firefox\firefox.exe`.

**Status Bar:** Displays "Writable", "Sma...ert", and "688 : 1".

# Java EE 6 Development

- Content assist and as you type validation
- Quick fixes for code and project configuration
- Advanced refactoring options to allow you to modify and maintain code in an iterative manner
- Annotation view to manage and modify annotation properties
- More flexible assembly of modules into EARs.
- Develop EJBs directly within Web projects (for WAS v8).

The screenshot displays an IDE interface with several key features highlighted:

- Code Editor:** Shows a Java file named `MessageBean.java` with the following code:

```
package com.ibm.test;
import javax.ejb.MessageDriven;

@MessageDriven
{
    JSR 220: EJB 3.0 Simplified API, 10.1.3: @MessageDriven.messageListenerInterface must be
    specified if the bean class does not implement its message listener interface.
}
```
- Quick Fix:** A context menu is open over the `@MessageDriven` annotation, offering options like "Add support for EJB 3.0", "Create annotation 'Stateless'", "Change to 'Stateful' (com.sun.xml.internal.ws.dev)", and "Rename in file (Ctrl+2, R direct access)".
- Rename Type Dialog:** A dialog box titled "Rename Type" is shown, listing changes to be performed:
  - Rename compilation unit 'TestBean.java' to 'TestBean2.java'
  - Maintain references to entity bean in JPA files and classes
    - Change references to bean in persistence file src\META-INF\persistence.xml
    - Change references to bean in mapping file src\META-INF\orm.xml
- JPA Details:** The "JPA Details" view for `JPACounterEntity` is shown, displaying:
  - Map As: Entity
  - Name: Default (JPACounterEntity)
  - Table: Name: EJB3COUNTERTABLE, Catalog: Default (), Schema: Default ()
  - Attribute Overrides
  - Secondary Tables
  - Inheritance

# 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

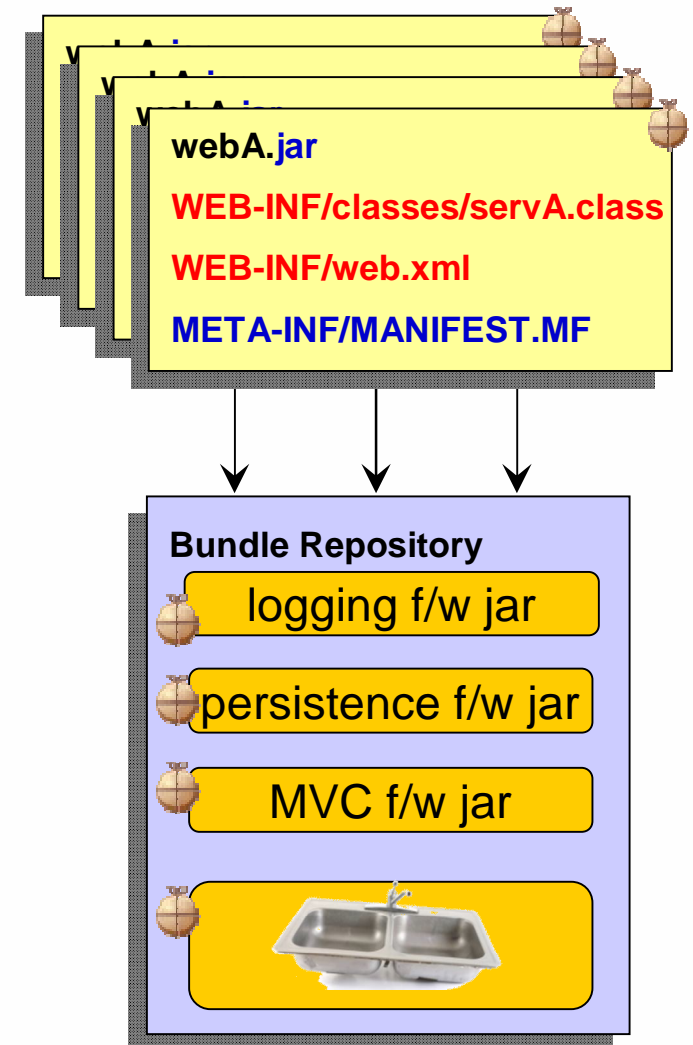
- **Standards Based DI Framework:** POJO development model, integrated with the server to control injection, activation & deactivation.

**Update applications modules without restarting the application**

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

## Development productivity tools with RAD:

- ✓ **Content assist, validation and wizards and editors** to define OSGi application manifests
- ✓ **Projects for OSGi applications bundles**, composite bundles and blueprint files.
- ✓ **Wizards and editors** to define OSGi application manifests, including validation.
- ✓ **WAS / WDTs** Supporting EJBs in OSGi bundles





# WebSphere Application Server Developer Tools

## Eclipse Marketplace



### IBM WebSphere Application Server V7.0 Developer Tools for Eclipse Juno V8.5.5 Member Company

An eclipse Juno SR2 IDE for building and deploying Java EE and Web 2.0 applications to WebSphere Application Server V7.0. For more information about installing this version of...

[Application Development Frameworks](#) | [Application Server](#) | [IDE](#) | [J2EE Development Platform](#) | [Team Development](#) | [Tools](#) | [Web](#) | [Web Services](#) | [XML](#)

Last Updated on 11 September 2013 by Eric D. Peters



### IBM WebSphere Application Server V8.0 Developer Tools for Eclipse Juno V8.5.5 Member Company

An eclipse Juno SR2 IDE for building and deploying Java EE, OSGi and Web 2.0 applications to WebSphere Application Server V8.0. For more information about installing this...

[Application Development Frameworks](#) | [Application Server](#) | [IDE](#) | [J2EE Development Platform](#) | [Mobile and Device Development](#) | [Team Development](#) | [Tools](#) | [Web](#) | [Web Services](#) | [Web, XML, Java EE and OSGi Enterprise Development](#)

Last Updated on 11 September 2013 by Eric D. Peters



### IBM WebSphere Application Server V8.5.5 Developer Tools for Eclipse Juno V8.5.5 Member Company

An eclipse Juno SR2 IDE for building and deploying Java EE, OSGi and Web 2.0 applications to WebSphere Application Server V8.5.5. For more information about installing this...

[Application Development Frameworks](#) | [Application Server](#) | [IDE](#) | [J2EE Development Platform](#) | [Mobile and Device Development](#) | [Team Development](#) | [Tools](#) | [Web](#) | [Web Services](#) | [Web, XML, Java EE and OSGi Enterprise Development](#)

Last Updated on 11 September 2013 by Eric D. Peters



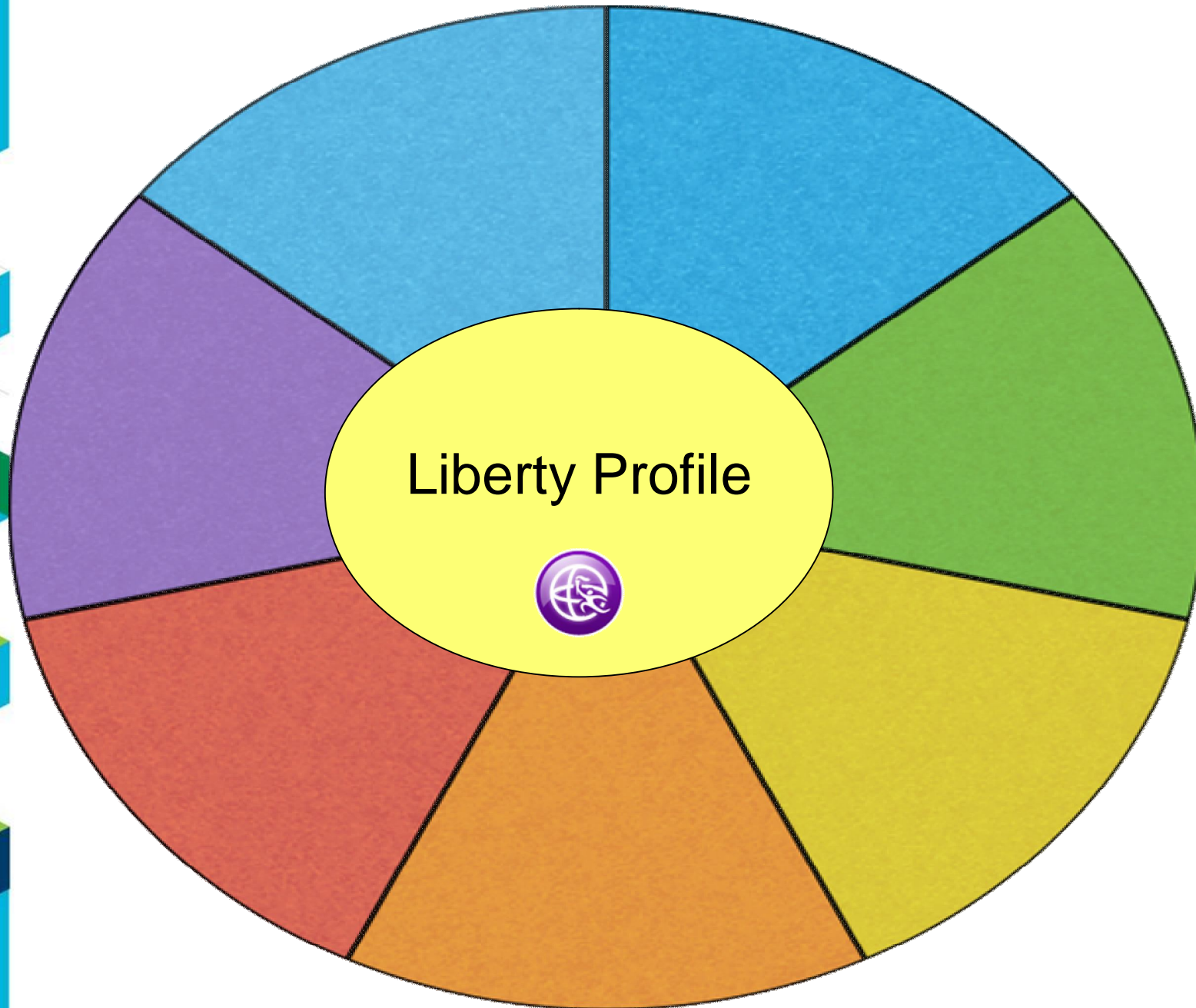
### IBM WebSphere Application Server V8.5.5 Liberty Profile Developer Tools for Eclipse Juno V8.5.5 Member Company

An eclipse Juno SR2 IDE for building and deploying Java EE, OSGi and Web 2.0 applications to WebSphere Application Server V8.5.5 Liberty Profile. For more information about...

[Application Development Frameworks](#) | [Application Server](#) | [IDE](#) | [J2EE Development Platform](#) | [Mobile and Device Development](#) | [Team Development](#) | [Tools](#) | [Web](#) | [Web Services](#) | [Web, XML, Java EE and OSGi Enterprise Development](#)

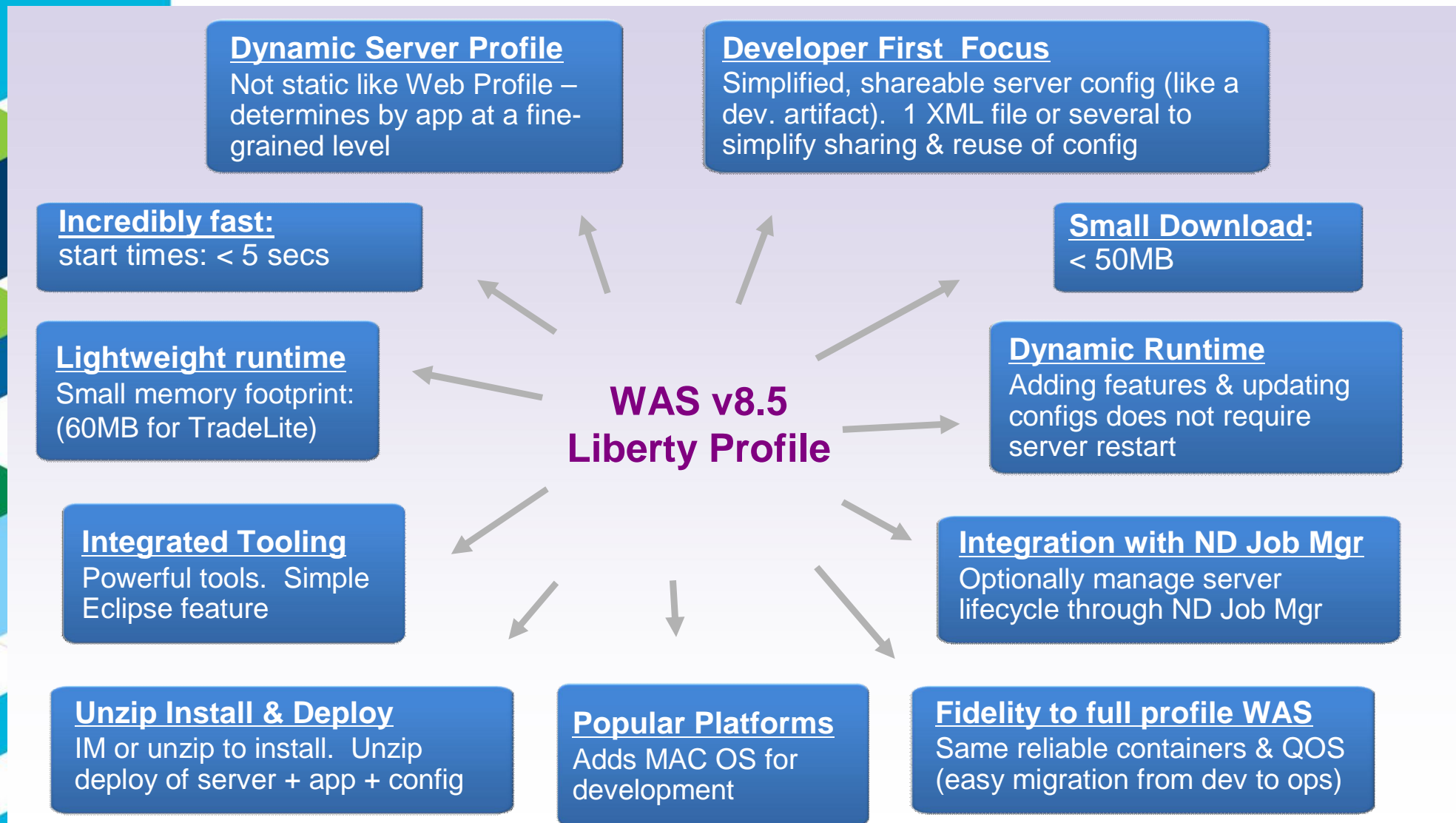
The tools are the same in each version except it includes version specific server adapters

WAS v7.0	WAS v8.0	WAS v8.5.5	WAS v8.5.5 Liberty Profile
<b>WebSphere Application Server Developer Tools</b>			



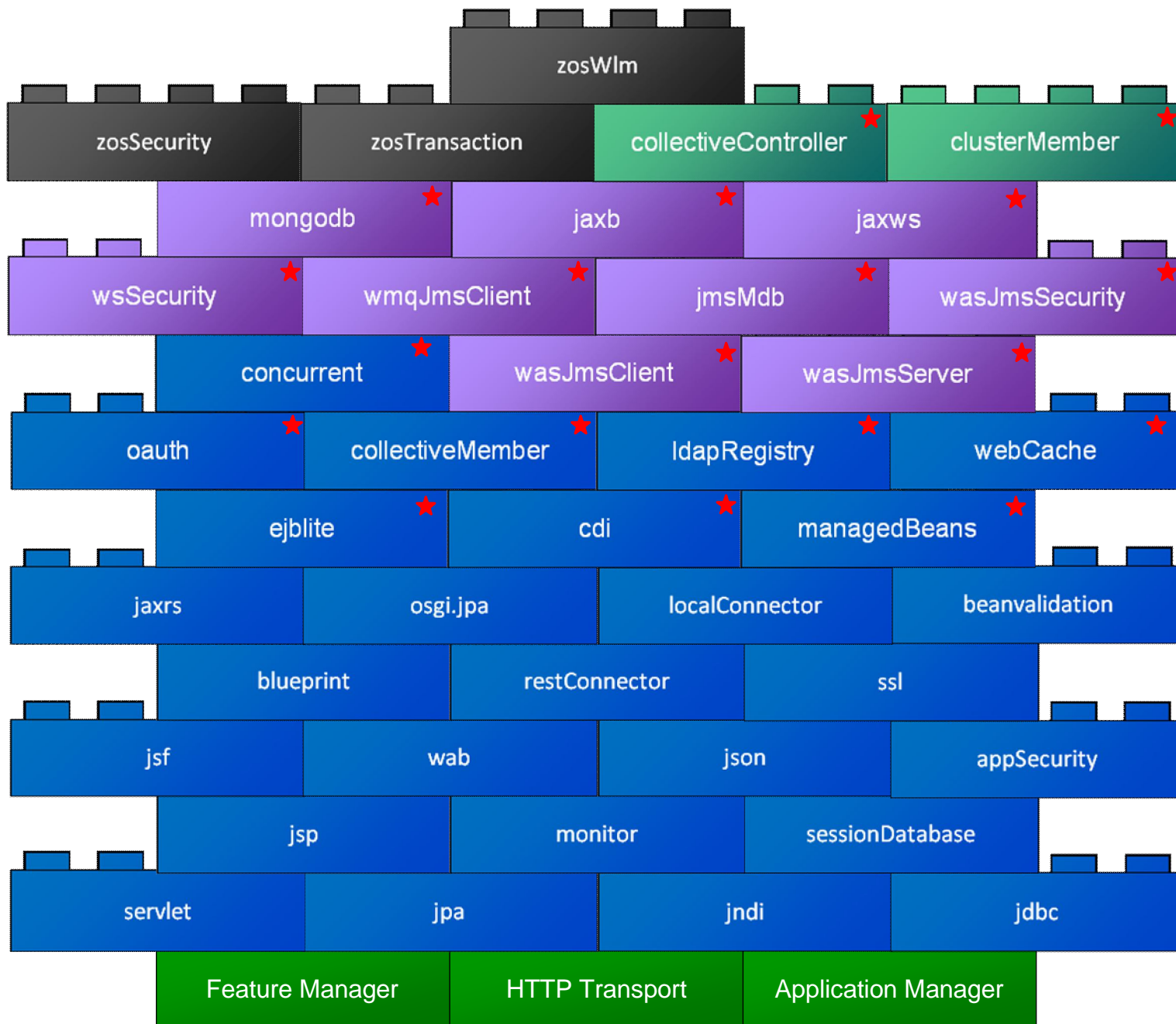
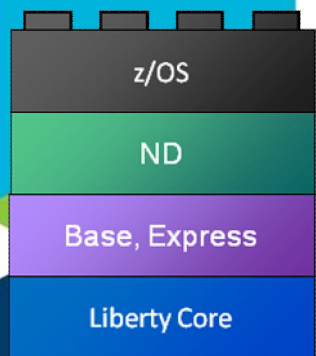
# WAS v8.5: Introduces the Lightweight “Liberty” Profile – For Web, OSGi and Mobile Apps

*A highly composable, dynamic Server profile*





# Liberty feature set – V8.5.5



WAS v8.5.5 Liberty Profile\*





# 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

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

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

```
<application name="tradelite" location="tradelite.war"/>
```

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

```
<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

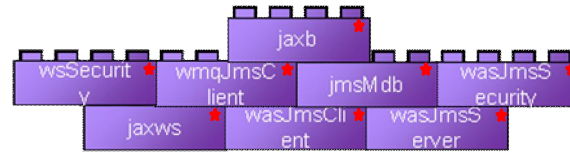
# Size Matters

- **Install** only what you need to **minimize disk footprint**
  - ▶ Modular “Archive Install” or
  - ▶ Standalone Installation Manager repository

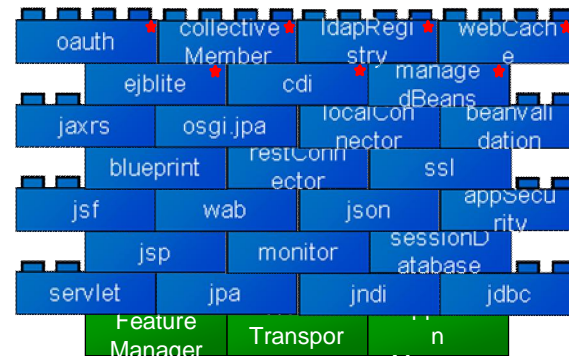
- **Start** only what you need – only configured features are started by the Liberty kernel to **minimize memory footprint and startup time**

- **Package** only what you need to **minimize a packaged server:**

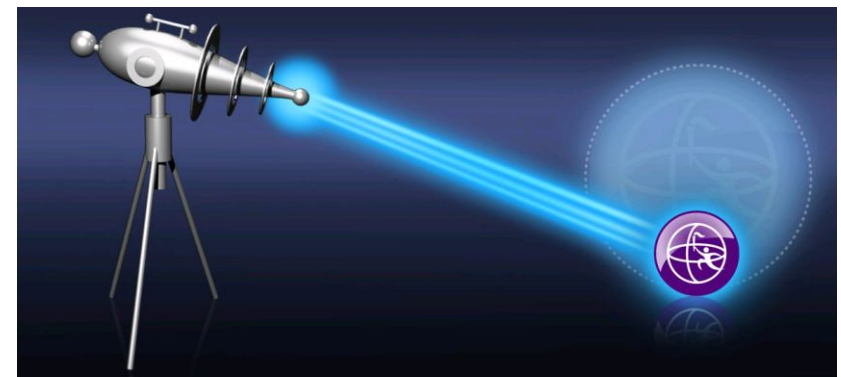
```
server package serverName --include=minify
```



Liberty **extended** archive 24MB

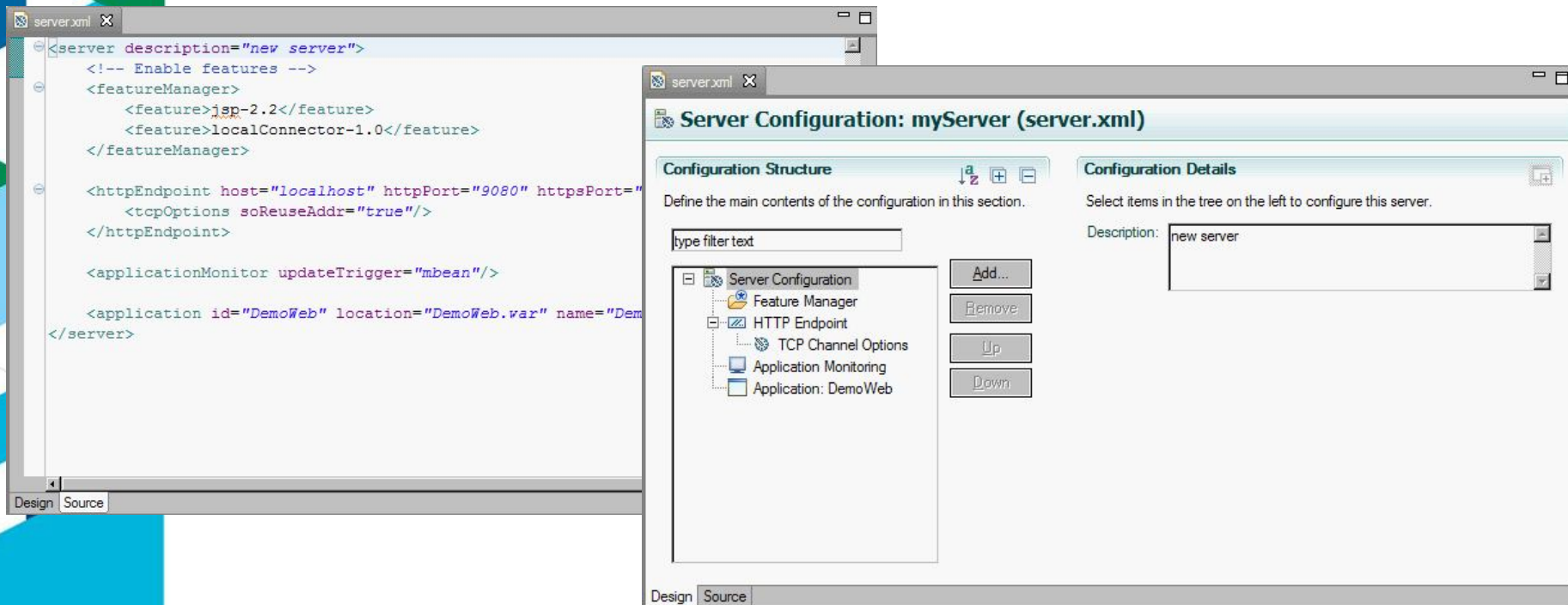


Liberty **runtime** archive 50MB (Java EE Web Profile)



# Working with Server Configuration

- Editing configuration is at the heart of working with any server
- Even with the simplicity of Liberty's configuration, a powerful configuration editor can be very helpful:
  - ▶ Source view with content-assist
  - ▶ Design view for advanced editing
  - ▶ Integrated support for features, variables, references, ...



# Liberty – Java EE Web Profile

The Java EE Web Profile provides a proper subset of the Java EE specification enabling capabilities focused on developing lightweight web applications

- Supported by all WAS editions

- Web profile specification: `<feature>webProfile-6.0</feature>` enables all

  - jpa-2.0

  - jsf-2.0

  - jsp-2.2

  - beanValidation-1.0

  - servlet-3.0

  - appSecurity-2.0

  - jndi-1.0

  - jdbc-4.0

## And (new in V8.5.5):

  - Enterprise JavaBeans (EJB) 3.1 Lite + Interceptors 1.1:

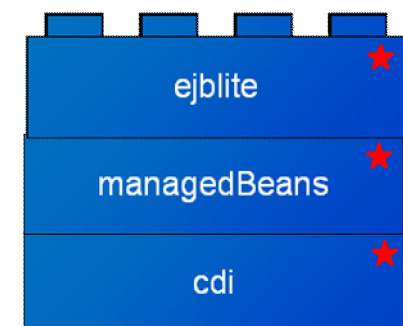
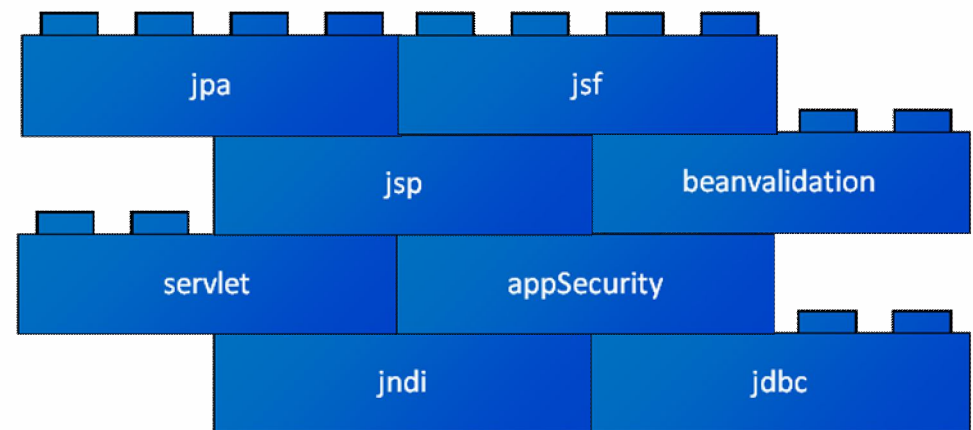
    - `<feature>ejbLite-3.1</feature>`

  - Managed Beans 1.0: `<feature>managedBeans-1.0</feature>`

    - `<feature>managedBeans-1.0</feature>`

  - Contexts and Dependency Injection: JSR-299 1.0 and JSR-330 1.0:

    - `<feature>cdi-1.0</feature>`

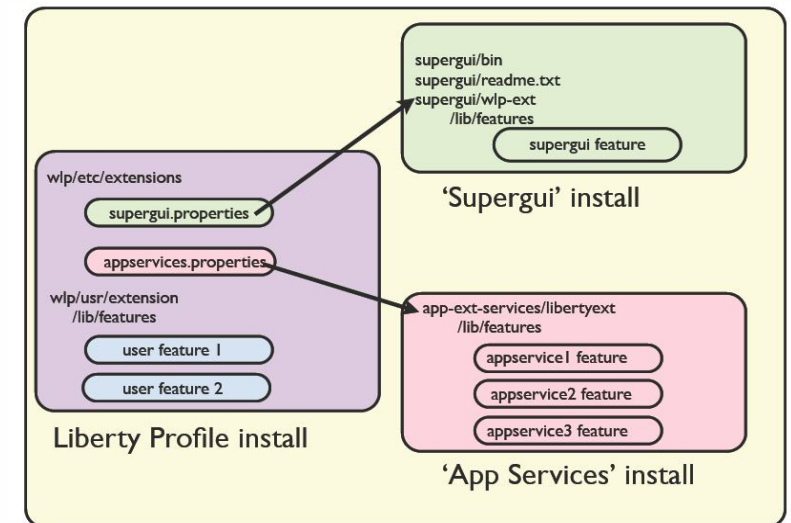




# Liberty – SPI for Liberty Profile Extensions

*The Liberty Extensions System Programming Interface (SPI) provides the ability to extend the Liberty profile with custom features, including full life cycle management*

- Add your own Liberty features via OSGi bundles
- Allows complete integration of third-party function into the Liberty runtime and development tools
- System Programming Interface (SPI) for:
  - ▶ Adding your own configuration
  - ▶ Interaction with Liberty kernel
- WebSphere Developer Tools (WDT) project type for feature development
- Full life cycle: packaging, installation, and un-installation
- Sample scenarios:
  - ▶ Extend OSGi applications programming model
  - ▶ Provide new infrastructure (i.e. Custom User Registry)



# EJB-lite Support

*The EJB-Lite support in WAS v8.5.5 provides a subset of the full EJB 3.1 API enabling a convenient and lightweight EJB implementation meeting the needs of many applications*

- EJB 3.1 Lite is a key component of the Java EE Web Profile spec
- Subset of EJB 3.1 spec focused on session beans and local interfaces
- Keeps EJB implementation small and lightweight
- Available in all WAS editions



	EJB 3.1 Lite (Liberty and Full Profile)	Full EJB 3.1 (Full Profile)
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		✓
Container managed transactions / Bean managed transactions	✓	✓
Declarative Programmatic Security	✓	✓
Embeddable API	✓	✓ (only required for EJB Lite)

# Web Services

WAS v8.5.5 Liberty provides both client and server functionality for SOAP based Web Services enabling easy development, integration and management of Web Services

## ■ Apache CXF implementation

## ■ Supported APIs

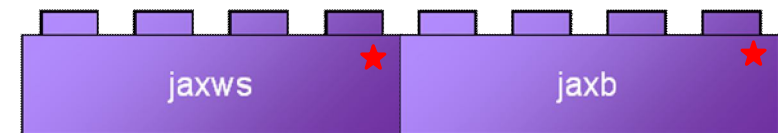
- ▶ JAX-WS 2.2
- ▶ JAXB 2.2
- ▶ Web Services for EE 1.3
- ▶ Both POJO and EJB based Web services
- ▶ JDK bundled APIs also available: SAAJ, JAXP, StAX

## ■ Protocols

- ▶ SOAP/HTTP
- ▶ WS-Security

## ■ Simplified development and deployment

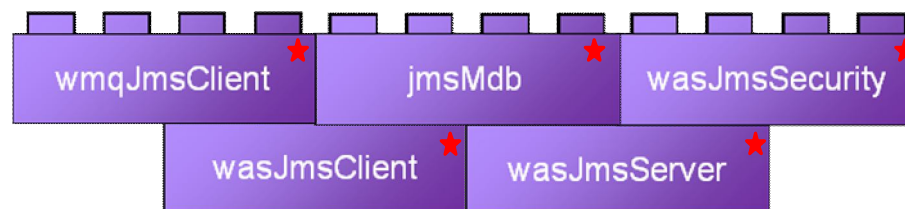
- ▶ Eclipse WDT wizard supports top down WSDL to Java generation, and bottom up POJO to web service
- ▶ Eclipse WDT plugin supports rapid edit-compile-debug cycles
- ▶ For EJB based implementation:
  - No router WAR generation required
  - No EAR wrapper required



# Java Messaging Support

*v8.5.5 provides Java Messaging Support for Liberty, including a lightweight single server message provider and connections to message providers such as WebSphere MQ*

- JMS and MDB programming models features
- Lightweight messaging engine localizes messaging destinations (queues and topics)
  - ▶ Integrates with transactions and security services
  - ▶ For development and testing
  - ▶ Supports security across process boundaries
- Inter-operates with WAS full profile SIBus
  - ▶ Liberty client to full WAS SIBus
  - ▶ Full WAS JMS client to Liberty
- WMQ Client:
  - ▶ Integration with WMQ for production messaging





# MongoDB

*Create applications for mobile and cloud environments using the Liberty profile and new integration with the MongoDB NoSQL database*

- MongoDB is an open source highly scalable noSQL database
  - ▶ Scalable, document-oriented, open source noSQL database
  - ▶ JSON data model with dynamic schemas
- Configure Liberty server to point to MongoDB Server
- Programming model supports:
  - ▶ Injection, lookup through JNDI, J2SE style
  - ▶ Application uses MongoDB Java API directly after obtaining reference
- Available in Liberty for WAS, WAS Express, WAS ND, and WAS z/OS
- Value: Easy for developers to create web applications and integrate with MongoDB as a document store (e.g. for user created data).  
Minimal impact to business as applications change in response to business changes.



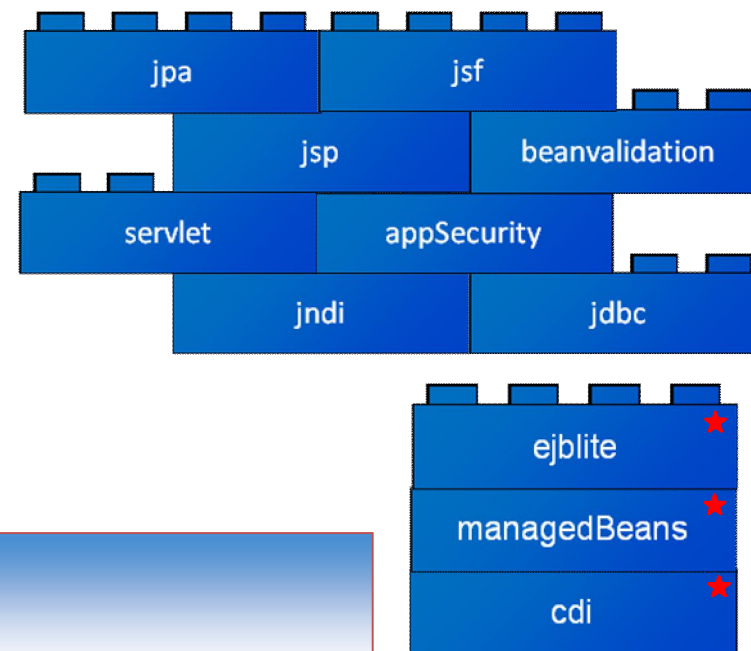
# Liberty – Java EE Web Profile Completion

*The Java EE Web Profile provides a proper subset of the Java EE specification enabling capabilities focused on developing lightweight web applications*

- Supported by all WAS editions
- Enables a lightweight and standard runtime as a best practice for building web applications
- 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:
- ▶ Managed Beans 1.0:
- ▶ Contexts and Dependency Injection: JSR-299 1.0 and JSR-330 1.0

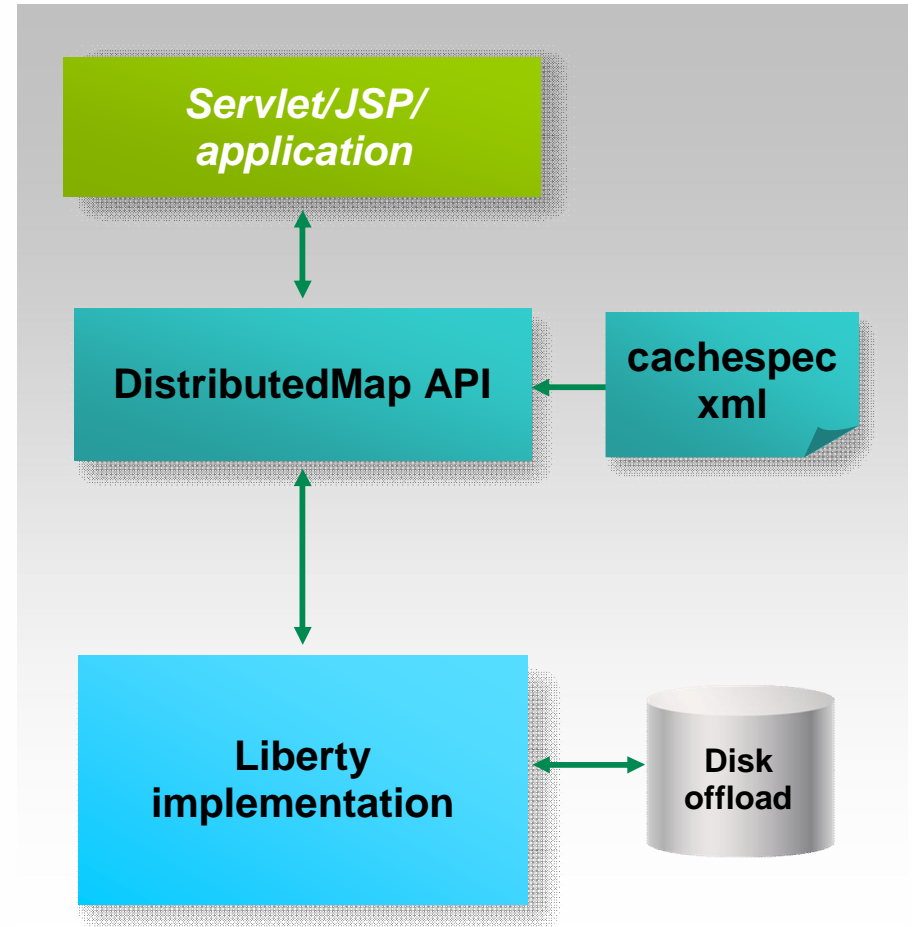


v8.5.5

# Support for WebSphere Web Cache

WAS v8.5.5 provides support for WebSphere Web Cache, a.k.a. DynaCache, enabling a local caching service for simplified development of applications

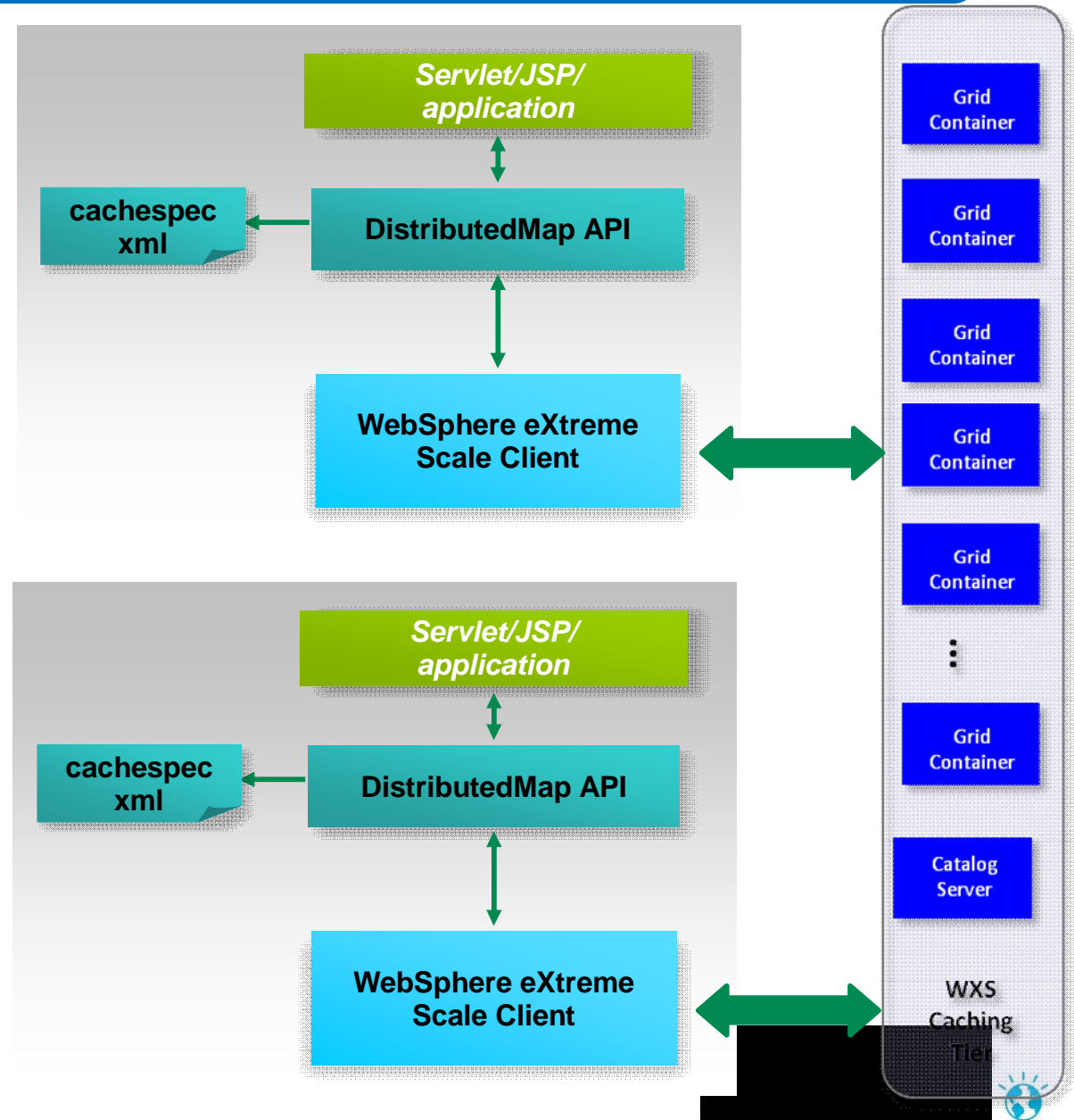
- Single server cache for dynamic web content
- Improve performance and responsiveness of web applications
- Same API as full profile WAS
- Enables developers to easily develop / test applications for single server caching solutions
- Run as single server in production or move applications unchanged to distributed caching via WebSphere eXtreme Scale



# Upgrade to WebSphere Extreme Scale for Web Caching

*Upgrade to WebSphere Extreme Scale for greater data elasticity and higher caching qualities of service*

- Massive data volume with linear scalability
- Elasticity: add more instances dynamically and runtime redistributes data automatically.
- Cache consistency
- Configurable number of replicas: balance between read and write operations.
  - ▶ No need to store cache at every application server.





# Asynchronous Work Management

*Provides multi-threaded programming model in the application server enabling concurrent processing*

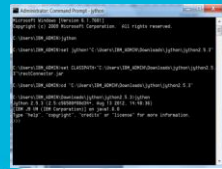
- New capability in V8.5.5 on both full WAS and Liberty profile enabling concurrent processing and improved performance
- Allows application to use `java.util.concurrent.ExecutorService` to perform work asynchronously, while retaining context of the calling thread
  - ▶ Fire and forget: `execute`
  - ▶ Wait for completion: `invokeAll`, `invokeAny`, `submit`
- Access `ExecutorService` through
  - ▶ JNDI
  - ▶ Resource injection
- Server configuration defines the JNDI name and the contexts to propagate, e.g., security, transaction.



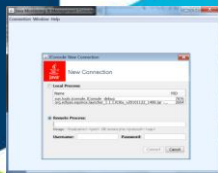
```
<managedExecutorService jndiName="concurrent/execSvc1">
  <contextService>
    <!-- One or more of the following -->
    <jeeMetadataContext/>
    <classloaderContext/>
    <securityContext/>
    <zosWLMContext/>
  </contextService>
</managedExecutorService>
```

# Liberty Server Administration

WAS v8.5.5 introduces a new administrative model enabling the grouping of Liberty servers into collectives for common management and static clustering (ND or z/OS)



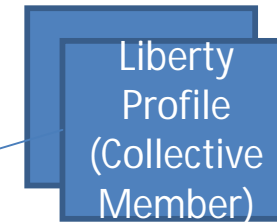
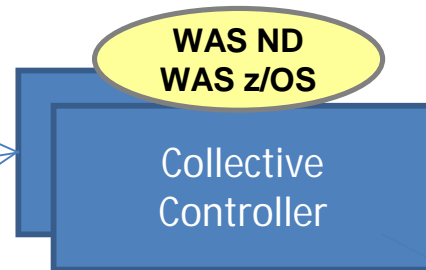
jython



jconsole

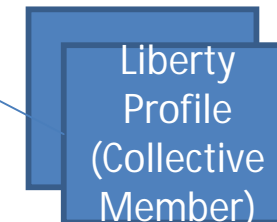


java



Collective  
Member

Any  
Liberty  
Edition



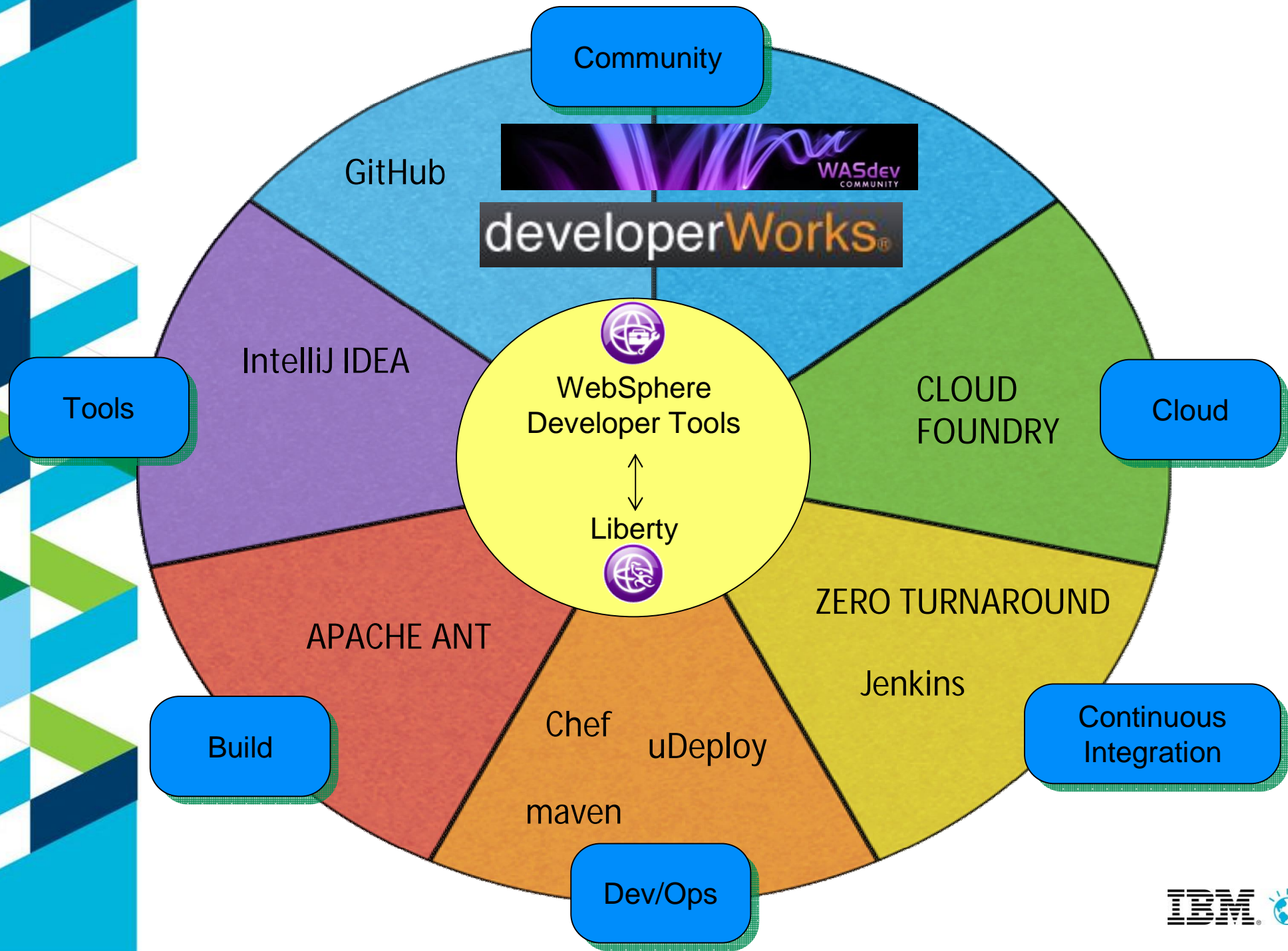
Application  
Cluster

WAS ND  
WAS z/OS

## New Liberty collective management infrastructure

- Agent-less administration
- Application and server deployment / updates
- Static cluster management
- All WAS editions can be Collective Members
- Collective Controller similar in role to Job Manager
- **WAS ND or z/OS required for Collective Controller**
- **Only WAS ND and z/OS can be Cluster Members**
- Highly scalable (3000 or more JVMs)

# Liberty Ecosystem - Current





BLOG

DOWNLOAD

LEARN

PARTICIPATE

# WASdev

COMMUNITY

## WASdev Home

0

Like

6  
tweets  
retweet

25

+1

★ 9

### FEATURED



**WebSphere Application Server V8.5 Announced**



**Liberty Profile running on a Raspberry Pi - now with MQTT to control a house!**



**New and Noteworthy in the Rational Application Developer V8.5 Beta March Refresh**

## What's new on WASdev

**New!**

### WebSphere Application Server V8.5 Announced

This is an exciting day in WebSphere Application Server (WAS) history. Today, IBM officially announced a new developer friendly light-weight web app server called WAS V8.5 - Liberty Profile and a set of WAS ND enhancements that reset the bar on app server resiliency for mission critical applications. [Read more...](#)

**New!**

### The Liberty Profile, MQTT and RSMB running on the Raspberry Pi to control a house 50 miles away!

In this video, we see The liberty profile running a web front end which interfaces with Really Small Message Broker (RSMB) to ultimately monitor and control devices that are 50 miles away on the Isle of Wight! [Read more...](#)

[more...](#)

### Three Steps to Liberty

Follow these three steps to run your first web application on IBM® WebSphere Application Server V8.5 Liberty Profile Beta. [Read more...](#)

### TRENDING TOPICS

[learn](#)

[article](#)

[blog](#)

[sample](#)

[video](#)

### LINKS

**Connect.**  
WASdev forum

### RELATED POSTS

The Liberty Profile, MQTT and RSMB running on the Raspberry Pi to control a house 50 miles away! Relationships: 1

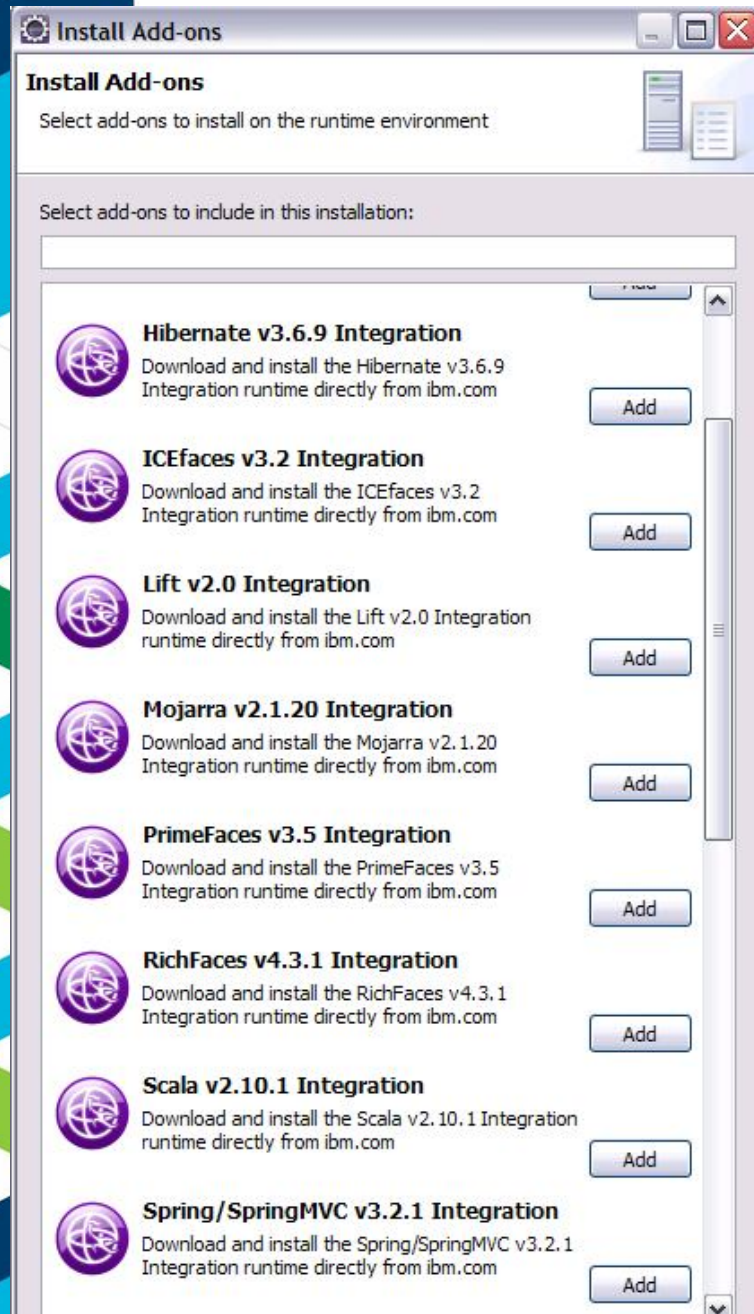
Three Steps to Liberty Relationships: 1

WebSphere Application Server running on the Raspberry Pi Relationships: 1



# Repository for Extending the Liberty Profile Runtime

Community



- An **online repository** to deliver **Liberty platform extensions**
- Content includes:
  - Open source project integration
  - Individual Liberty features
  - Samples
- Enables more **fine-grained**, early access content, simplest possible **integration with external open source**
- Install using Liberty command-line tools or WDT/RAD



# “WASdev” Open Source Community on GitHub

Community

- WASdev Open Source community extension to the WASdev developer community

- ▶ <http://wasdev.github.io>

- ▶ <https://github.com/wasdev>

- GitHub - 3.5M users, 6M repos (Jan 2013)

- IBM contributions at launch, community to follow

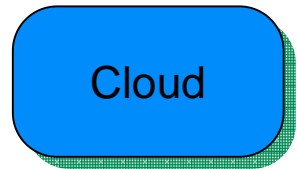
- Apache License, Version 2.0

- Resulting binaries hosted on Liberty Repository

- ▶ <https://www.ibmdev.net/wasdev/repo>

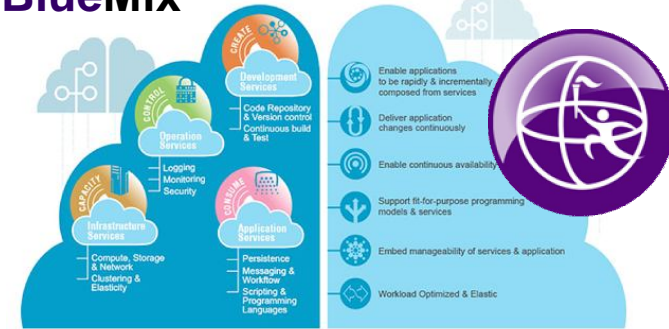


# IBM WebSphere Liberty Buildpack



Cloud Foundry

IBM BlueMix



The Buildpack for IBM WebSphere Liberty is *freely* available for developers

<https://github.com/cloudfoundry/ibm-websphere-liberty-buildpack>

- Push deploy applications to PaaS-provided WebSphere Liberty
- Hosted or on-premise
- Currently development-use only. But free!
- Includes IBM Java 7
- Uses Cloud Foundry extensible buildpack architecture
- The WebSphere Liberty buildpack is an open source contribution to the Cloud Foundry community and provisions developer-licensed instances of WAS Liberty Core V8.5.5.0



Contributed to open source Cloud Foundry community

Visit our developer community at <http://wasdev.net>

IBM BlueMix: <http://ibm.com/bluemix>





# DevOps Tool Chain

Dev/Ops

Release and Deploy

## IBM SmartCloud



Line of Business

Plan and Measure

uRelease

Application Release Management

uDeploy

Application Deployment Automation

SmartCloud Orchestrator  
IBM Pure Application System  
Openstack

Cloud Provisioning

Jenkins uBuild  
Rational Build Forge



Development



Build



Package Repo



TEST ENV.



STAGE ENV.



PROD ENV.

Develop and Test



SmartCloud Control Desk



SmartCloud Application Performance Management

Monitor and Optimize





# Environment flexibility

Dev/Ops

Plan and Measure

Release and Deploy



Line of Business

Application Release Management

Chef



SmartCloud Orchestrator  
IBM Pure Application System  
Openstack  
EC2

Cloud Provisioning

Jenkins

Application Deployment Automation

Maven



Development



Build



Package Repo



TEST ENV.



STAGE ENV.



PROD ENV.

Git / SVN etc  
Eclipse / WDT

Gerrit (code review)

Monitor and Optimize

Develop and Test

## Chef recipe to install and configure WebSphere Application Server Liberty Profile

GitHub

Search or type a command

Explore Features Enterprise Blog

Sign up

Sign in



WASdev

Joined on Jul 03, 2013

8

public repos

3

members

Repositories

Members

Find a repository...

Search

All Sources Forks Mirrors



**ci.chef.wlp.application**

Ruby ★ 0 🍴 0

Chef cookbook for deploying applications to the WebSphere Application Server Liberty Profile

Last updated 8 hours ago



**ci.chef.wlp**

Ruby ★ 2 🍴 0

Chef Recipes for Liberty profile

Last updated 8 hours ago



**ci.chef.wlp.samples**

Ruby ★ 0 🍴 0

Chef recipes to run Liberty profile samples

Last updated 6 days ago

# Integration with Maven

Dev/Ops

*The WebSphere tools and runtime integrate with Maven to allow its usage as part of your software development process*



## WDT/RAD Maven Tools

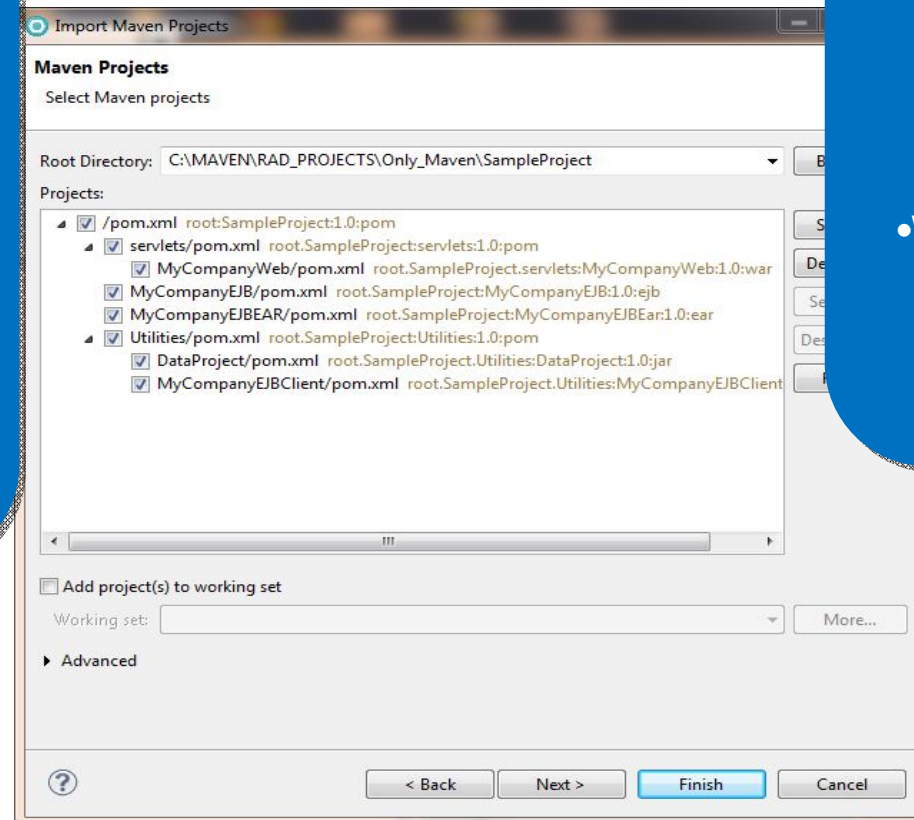
- “Mavenize” current tools projects
- Create new Maven projects
  - POM editing
- Ability to browse repository (Local & Maven central)
  - Maven Build (execute Maven commands)
  - Maven tools for OSGi bundles

Enhanced!

## IBM Maven Repository

- WebSphere Liberty plugin
  - Start, stop, install, deploy
- Archetypes
  - Java EE, OSGi, Portal
- WebSphere target POMs
  - Replace server target classpath

Enhanced!



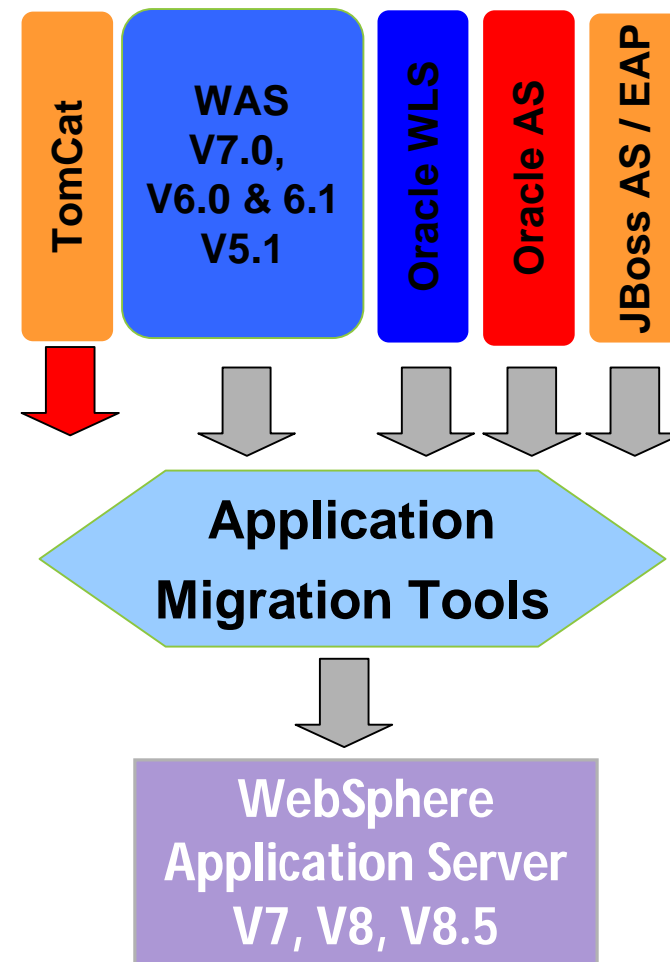
New!

# Application Migration Tools

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

## Support for Tomcat migration to WAS v8.5

- Migrate applications from older releases to WAS V8 or V7
- Migrate from Oracle or JBoss faster and easier to WAS V8.5 Beta, V8 or V7
  - ▶ Migrate applications up to 2x as fast
  - ▶ Migrate web services up to 3x as fast
- Application Migration Tool
  - ▶ Analyzes source code to find potential migration problems:
    - Removed & deprecated features
    - Behavior changes
    - JRE 5 & JRE 6 differences
    - Java EE specification changes or enforcements
  - ▶ Capable of making some application changes
  - ▶ Provides guidance on how to make required changes
  - ✓ Works with Eclipse, WebSphere Developer Tools (WDT) or Rational Application Developer (RAD)



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

✓ - Tools support available in RAD





IBM, the IBM logo, ibm.com, WebSphere and Rational are 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](http://www.ibm.com/legal/copytrade.shtml).

Apache Maven, Maven, Apache Ant and Ant are registered trademark of The Apache Software Foundation (ASF) in the United States, other countries, or both.

Github is a registered trademark of Github Inc. in the United States, other countries, or both.

Zeroturnaround is a registered trademark of Zeroturnaround in the United States, other countries, or both.

Pivotal, the Pivotal logo, Cloud Foundry are trademarks and/or registered trademarks of GoPivotal, Inc. in the United States and/or other countries.

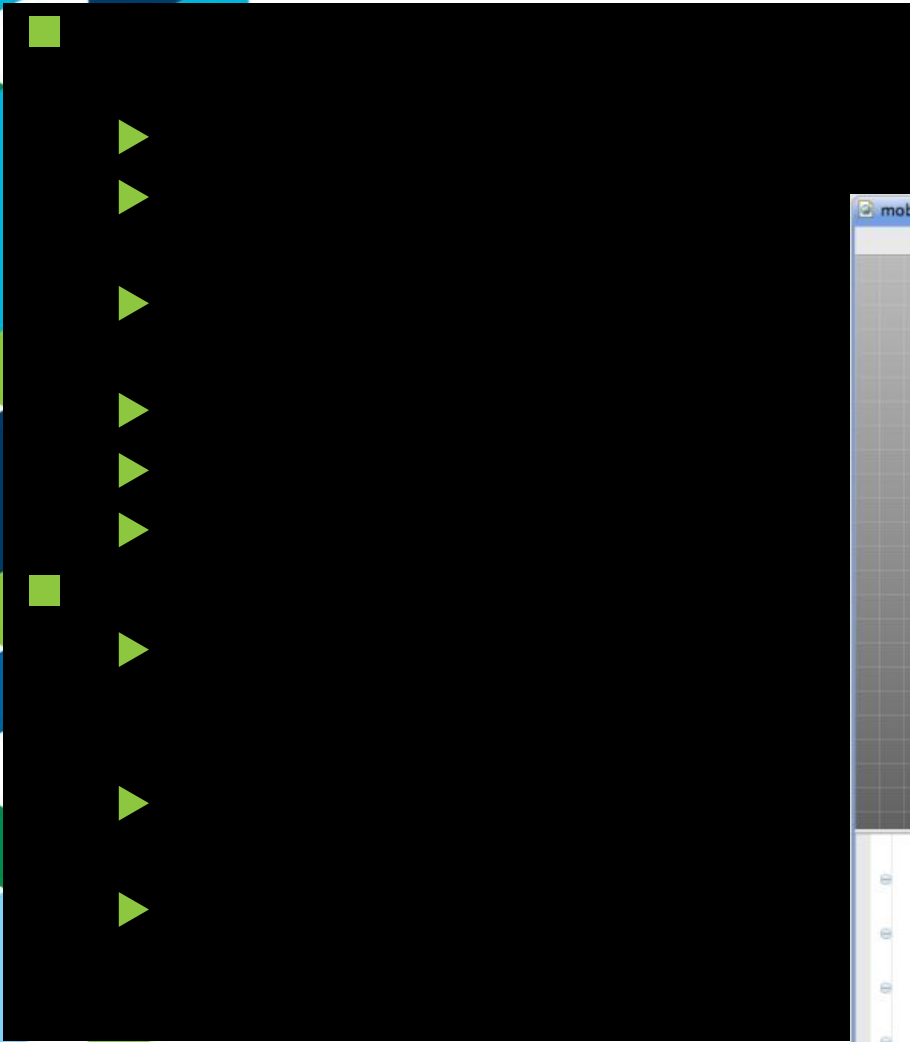
OpsCode and Chef are registered trademark of OpsCode in the United States, other countries, or both.

Jenkins is a registered trademark of Jenkins CI in the United States, other countries, or both.

IntelliJ is a registered trademark of JetBrains in the United States and other countries.

# Backup

# Web and Mobile development



```
style="color: gray; font-size: smaller">Re: Automated test
results</span></a></li>
<li><a href="#">Scott Paxton<br> <span
style="color: gray; font-size: smaller">RPE mobile device
frame - tweaks? </span></a></li>
<li><a href="#">Gabriel Indik<br> <span
style="color: gray; font-size: smaller">Outline view now
shows mobile widget icons</span></a></li>
<li><a href="#">Orlando Ezequiel Rincon Ferrera<br> <span
style="color: gray; font-size: smaller">jQuery widgets in
the Properties View</span></a></li>
<li><a href="#">Nick Sandonato<br> <span
style="color: gray; font-size: smaller">Try out the new
scaling feature</span></a></li>
<li data-role="list-divider">Unread</li>
<li><a href="#">IBM Corporate<br> <span
style="color: gray; font-size: smaller">Seriously you
guys... Synergy, FTW!</span></a></li>
</ul>
</div>
<div data-role="footer" data-position="fixed">
<h3>86 Unread messages</h3>
</div>
```

Design Source Split

# Source editing tools

	HTML5	JavaScript	Dojo	CSS3	JSON	jQuery
Code Assist		✓	✓	✓	✓	✓
Validation		✓	✓	✓	✓	✓
Outline		✓	✓	✓	✓	✓

The screenshot shows an IDE with a code editor and two floating panels. The code editor contains the following HTML and JavaScript code:

```

<script type="text/javascript">
  document.get
</script>
</head>
<body class="
<canvas dragg
</body>
</html>

```

The first floating panel shows a list of JavaScript methods for `document.get`:

- getElementById(String elementId)
- getElementsByName(String elementName)
- getElementsByTagName(String tagName)
- getElementsByTagNameNS(String namespaceURI, String tagName)

The second floating panel shows a CSS3 property list for `.AccountDetails {`:

- border
- border-bottom
- border-bottom-color
- border-bottom-left-radius
- border-bottom-right-radius
- border-bottom-style
- border-bottom-width
- border-color
- border-left

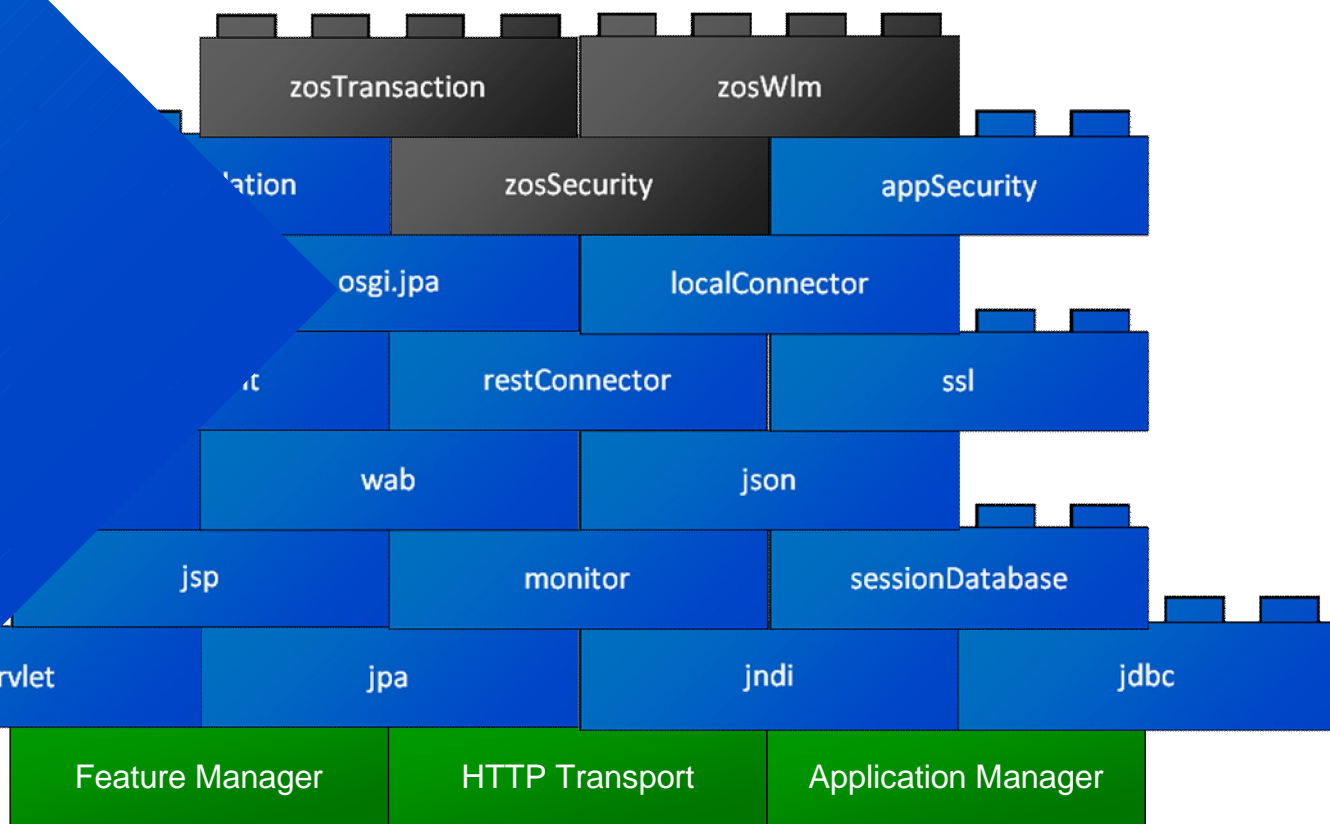
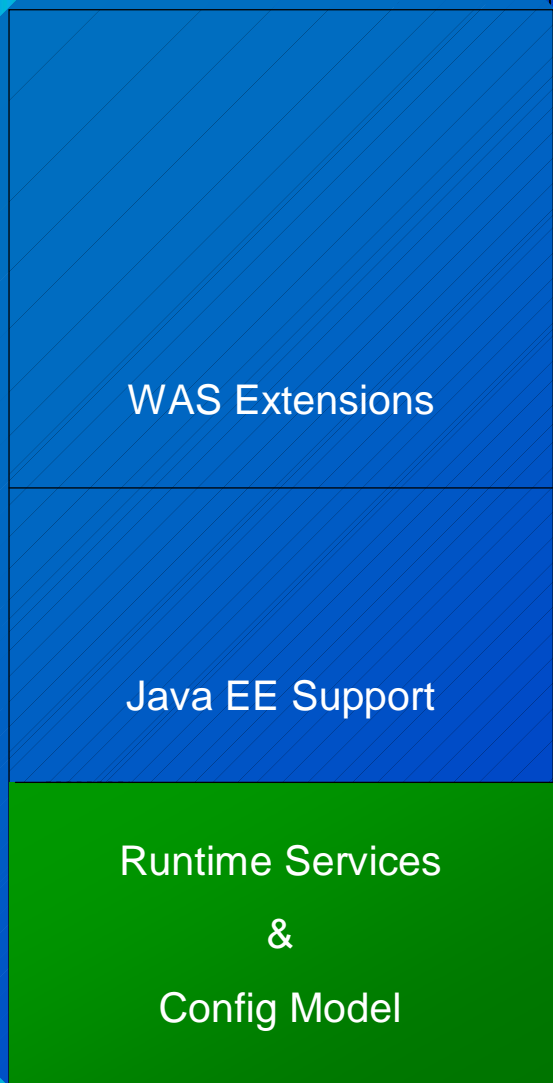
The rightmost panel shows the details for the selected `border-bottom-right-radius [CSS3]` property, including supported browsers: Chrome 40, Firefox 30, and Safari 50.



# Liberty feature set – V8.5

*delivers programming model and qualities of service enhancements to the Liberty profile, expanding the scope of Liberty based applications and deployment*

## Highly Composable Runtime Based on 'Features'



WAS v8.5 Liberty Profile



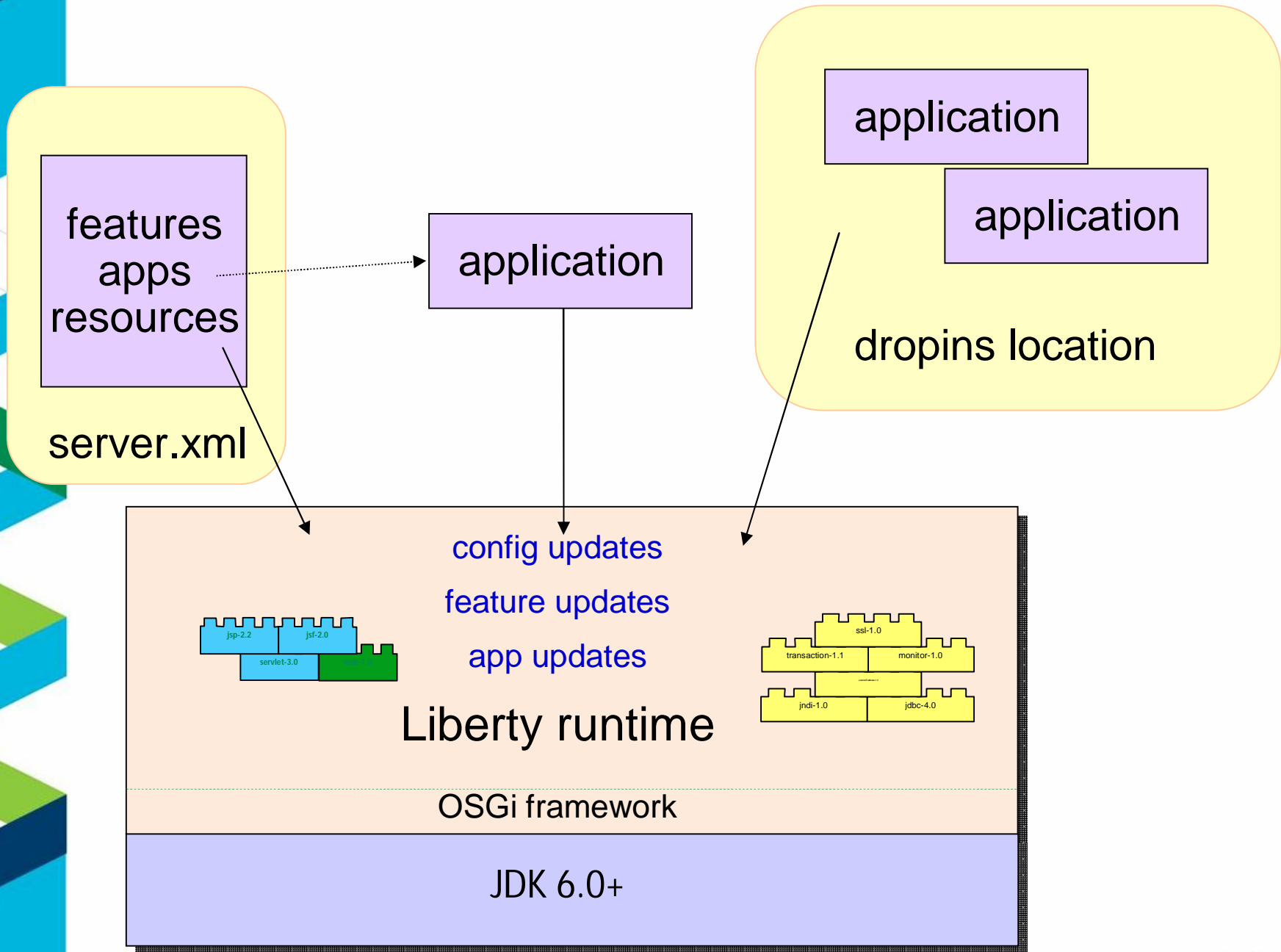
# Flexible Configuration

- Shareable config snippets:

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

- Config can be componentized at any level of granularity, from 1 file to many
- Visualization through the tools as a single logical view
- Team development: keep the application and configuration components together

# Dynamic Runtime Provisioning



# 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.

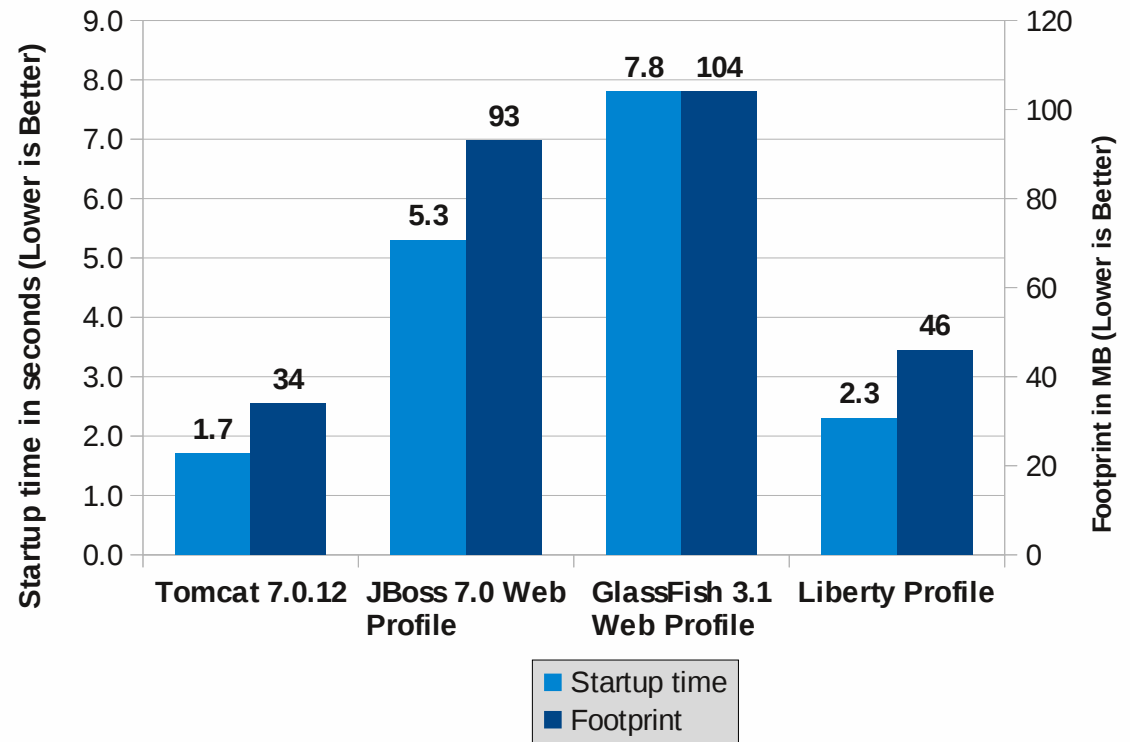
## System Info:

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

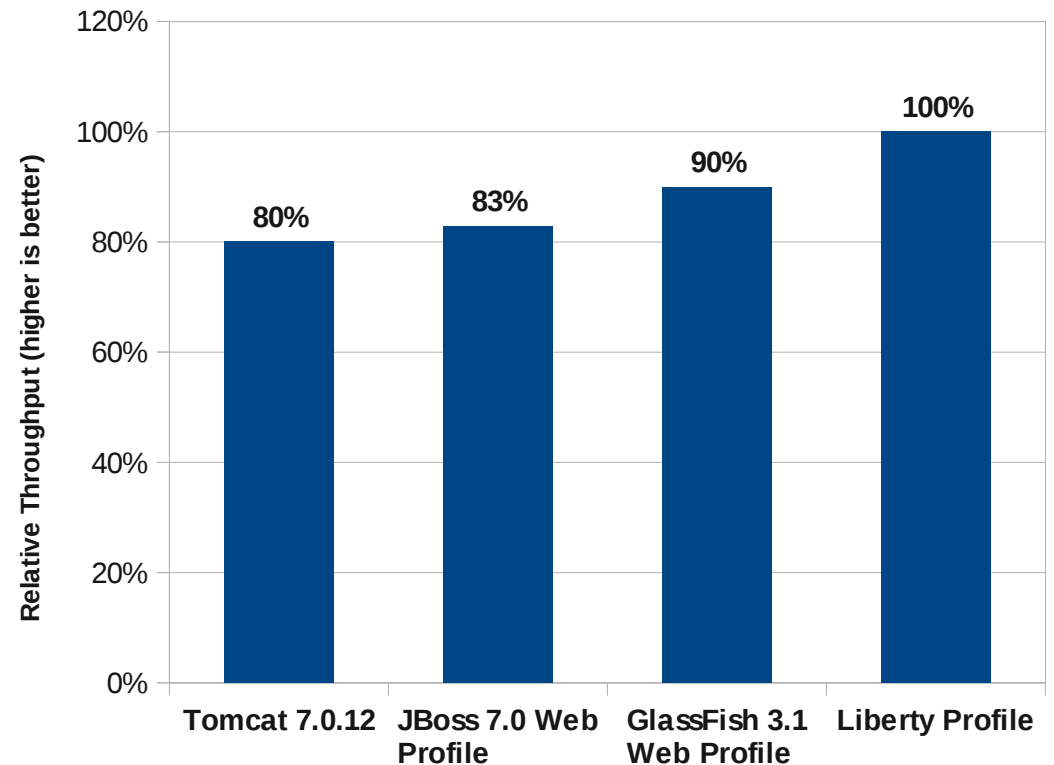
JBoss Community Edition 7.0 Web Profile server

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.*