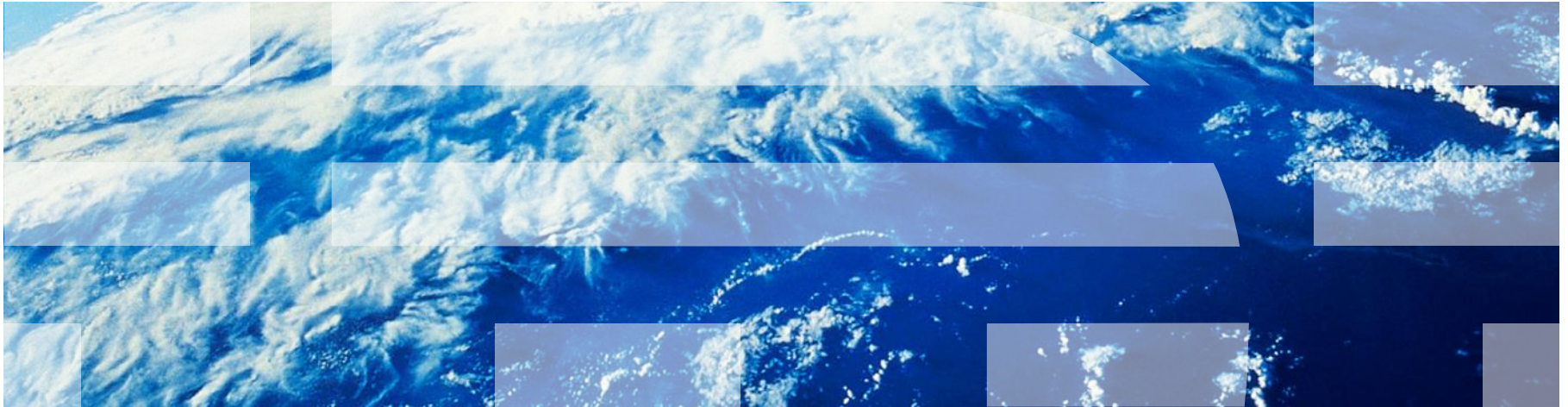


WebSphere Application Server Update and Direction

Ian Robinson, WebSphere Foundation Chief Architect



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.

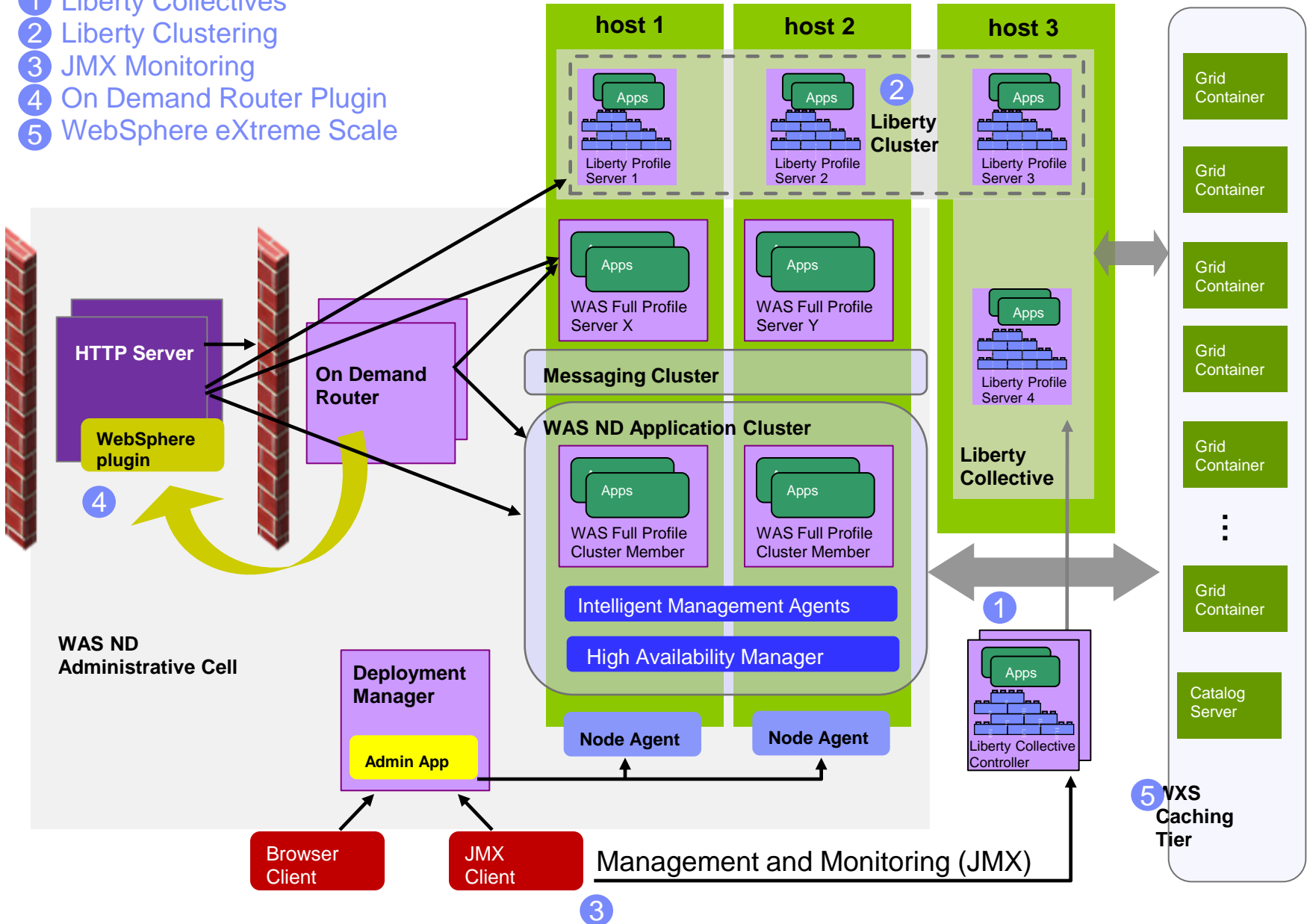
Agenda



- Current Application Infrastructure portfolio
- Enhanced WebSphere Application Server Family
- Overview of new release content
- Continued performance leadership
- Closing

WAS V8.5.5 High Level Architecture

- 1 Liberty Collectives
- 2 Liberty Clustering
- 3 JMX Monitoring
- 4 On Demand Router Plugin
- 5 WebSphere eXtreme Scale



WAS v8.5.5 Family



WAS for Developers



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

New!

Provide WAS and WDT editions as freely available for dev desktops and supported under production runtime licenses

WAS Hypervisor Edition



The WAS ND server optimized to instantly run in Pure Application System, VMware, PowerVM, zVM and other server virtualization environments.

WAS ND



Delivers near-continuous availability, with advanced performance and intelligent management capabilities, for mission-critical apps. Full entitlement to WXS.

WAS for z/OS



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

WAS



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

Includes entitlement to eXtreme Scale for HTTP session caching and DynaCache on the entitled WebSphere Application Server.

WAS Liberty Core

Liberty Profile
(Web Profile only)

A lightweight and low-cost Liberty profile based offering (not full-profile WAS), providing the capabilities to rapidly build and deliver web apps that do not require the full Java EE stack.

New!

WAS Express

Liberty
and Full
Profiles

A low-cost, ready-to-go solution to build dynamic Web sites & apps, including both Liberty and full-profile WAS. Restricted to a set amount of PVUs.

- ✓ **New WebSphere Application Server Liberty Core edition**
- ✓ **Entitlement to WebSphere eXtreme Scale (WXS) for some editions**
- ✓ **Developer install/support for WAS & WDT with active production server S&S**

Introducing WebSphere Application Server V8.5.5



Delivering value to enterprises, partners, and developers through fit for purpose application servers, flexible and extensible platforms, and ease of development

- **Developers** – **Easy and no-charge access; Lightweight developer friendly tools**
 - Fast and free download of WebSphere developer tools
 - Robust set of programming models, including several new for Liberty profile
 - Extensible runtime with ability to add custom features
 - Simple configuration and rapid server restart
 - Free and supported install of server runtime on developer machines with active server S&S

- **Enterprises** – **Fit for purpose application servers**
 - Breadth of application server types from lightweight to highly resilient
 - Develop, test, and run on one common application platform
 - Entitlement to WebSphere eXtreme Scale included with some WAS editions
 - Highly resilient application environments with enhancements to Intelligent Management

- **Partners** – **Easily develop, package and deploy custom applications**
 - Extend the Liberty Profile with custom features
 - Ability to package the server runtime, config, and application
 - Minimized server packages for maximum install density and rapid deploy
 - New lightweight Liberty only offering for Web Profile applications

WAS v8.5.5 Delivers



Fit for Purpose Servers enabling unmatched combination of Application Server runtime and development experience, from the highly resilient to the lightweight and nimble

Developer Experience

Fast, flexible, and simplified application development

- Lightweight Liberty Profile in all WAS Editions (since V8.5)
- New **WAS Liberty Core** edition
- Liberty Profile
 - Java EE 6 Web Profile
 - EJB Lite, CDI, Managed Beans
 - Web Services
 - JMS, MDB
 - NoSQL DB (MongoDB Client)
 - Support for WebSphere Web Cache (DynaCache)
 - Add custom and third party Liberty features
- Asynchronous work management
- Enhanced developer tools
- Supported WAS and WDT on developer desktops
- Enterprise Java Batch

Application Resiliency

Intelligent Management and Enhanced Resiliency

- WAS ND and WAS z/OS Intelligent Management (since v8.5)
 - App Edition Mgmt
 - App Server Health Mgmt
 - Dynamic Clustering
 - Intelligent Routing
 - Messaging resiliency
 - Memory leak protection
- Liberty Profile collective administration
- Liberty Profile clustering
- Intelligent management in WebSphere web server

Operational Excellence

Improved Operations, Security, Control & Integration

- Liberty profile
 - Security enhancements
 - Problem determination
 - Monitoring
- Service Mapping
- SIP improvements
- Improved Load Balancer for IPV4 and IPV6
- Liberty profile packaging and install enhancements
- WebSphere Extreme Scale integration
- Performance enhancements



WAS v8.5.5 Liberty Profile “Enhanced”

WAS v8.5.5 delivers significant enhancements to Liberty profile, including new programming models, administration, qualities of service and development tools

Dynamic Server Profile

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

“Developer First” Focus

Simplified, shareable XML server config. New integrated messaging server, DynaCache support, new prog. models, such as Web Services, JMS & EJB-Lite.

Enhanced!

Start fast, run efficiently

Starts in <3s; Mem footprint <50MB; (TradeLite benchmark)

Integrated tools

Powerful tools in WDT Eclipse feature. Enhanced for v8.5.5 prog models, Maven integration, ++

Enhanced!

Web Profile Certified

Create web apps for the Java EE Web Profile standard.

New!

Unzip install and deploy

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

Enhanced!

Liberty Extensions

Add custom features and integrate 3rd party components via Liberty extensions interface

New!

Small Download

50MB for Web Profile features

Dynamically Extensible

Install new features from repository (local or remote) with no svr restart

New!

Lightweight cluster Mgmt

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



WAS v8.5.5 Liberty Profile & WAS Developer Tools for Eclipse (WDT)

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>
```

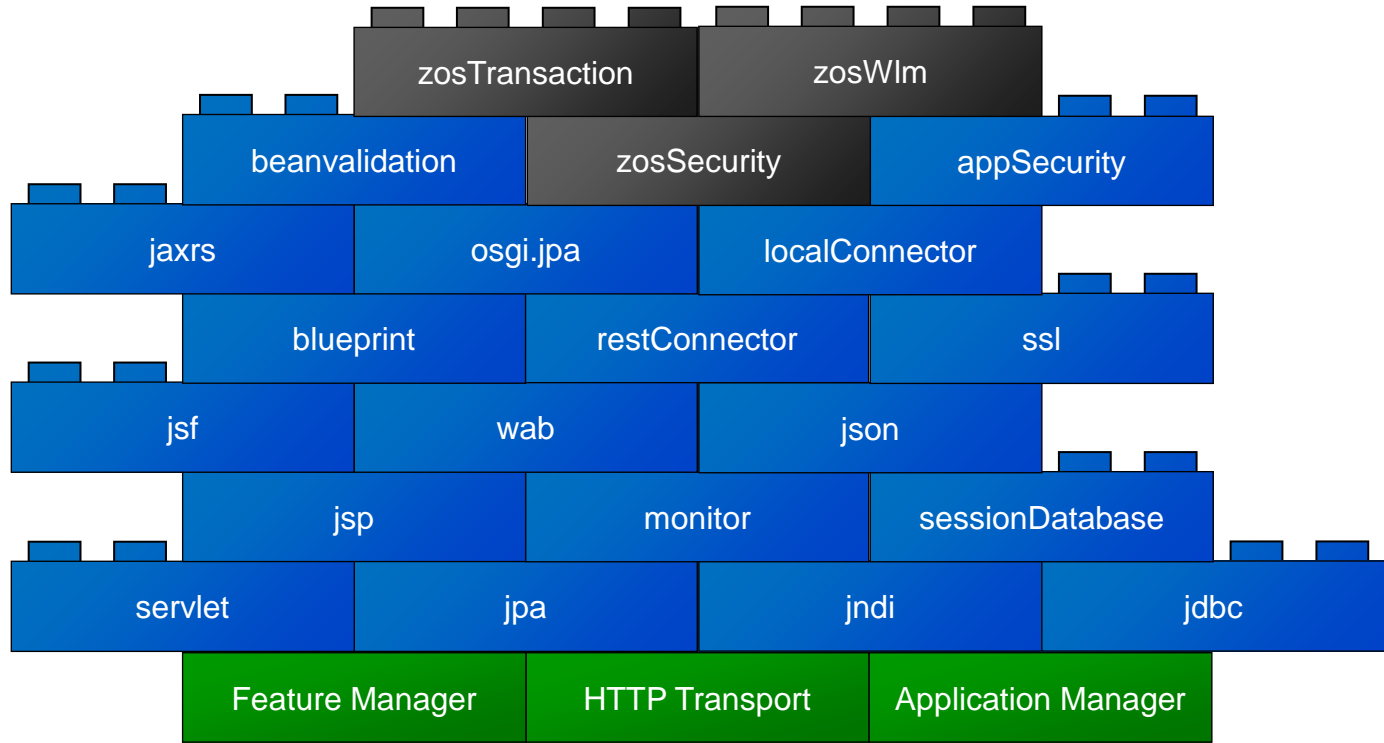
```
</server>
```

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'

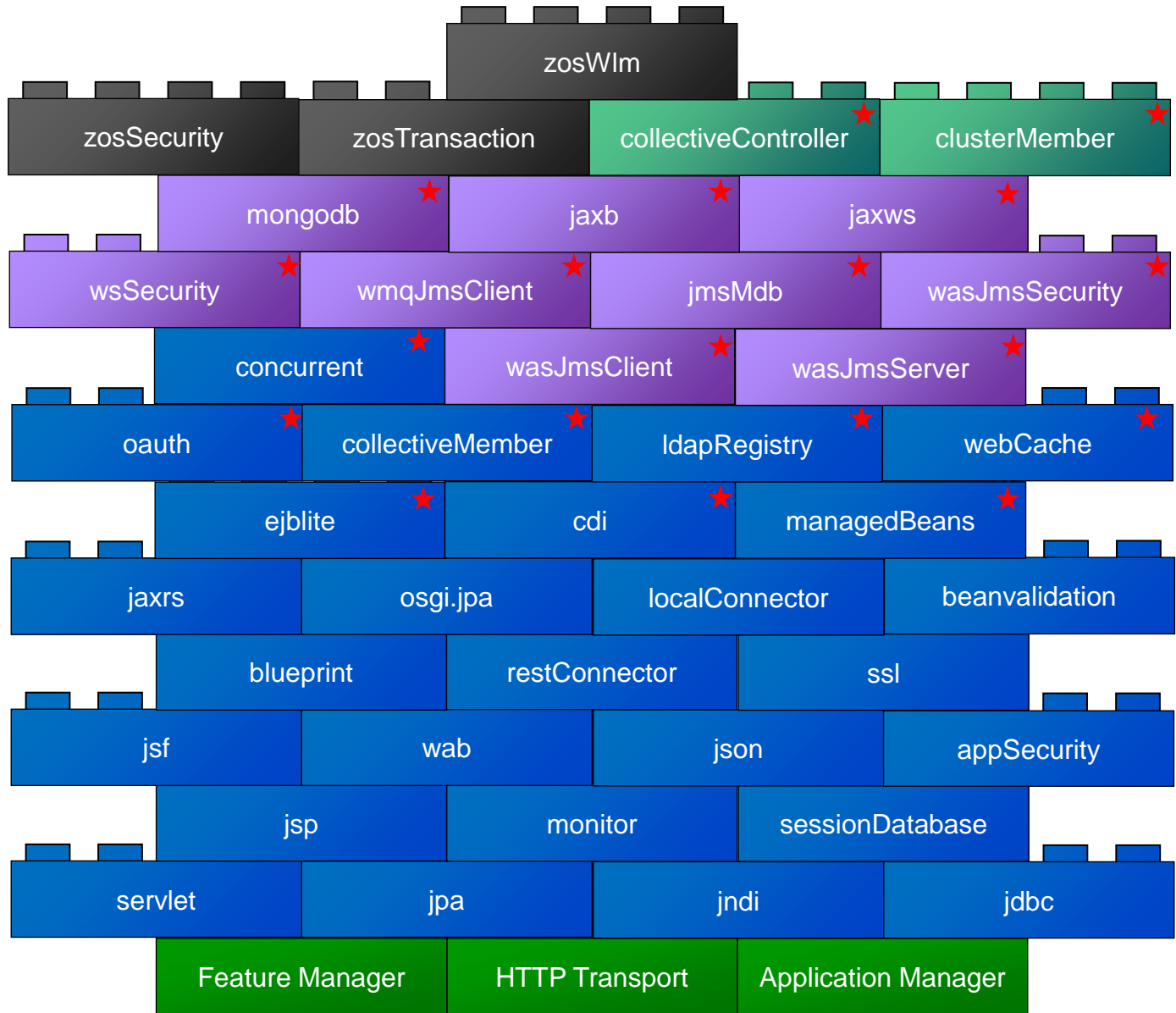
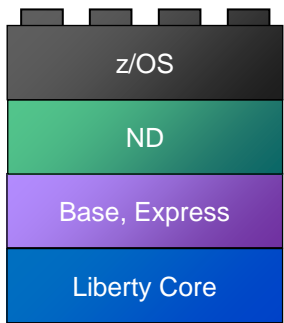


Full WAS Profile



WAS v8.5 Liberty Profile (V8.5.0)

Liberty feature set

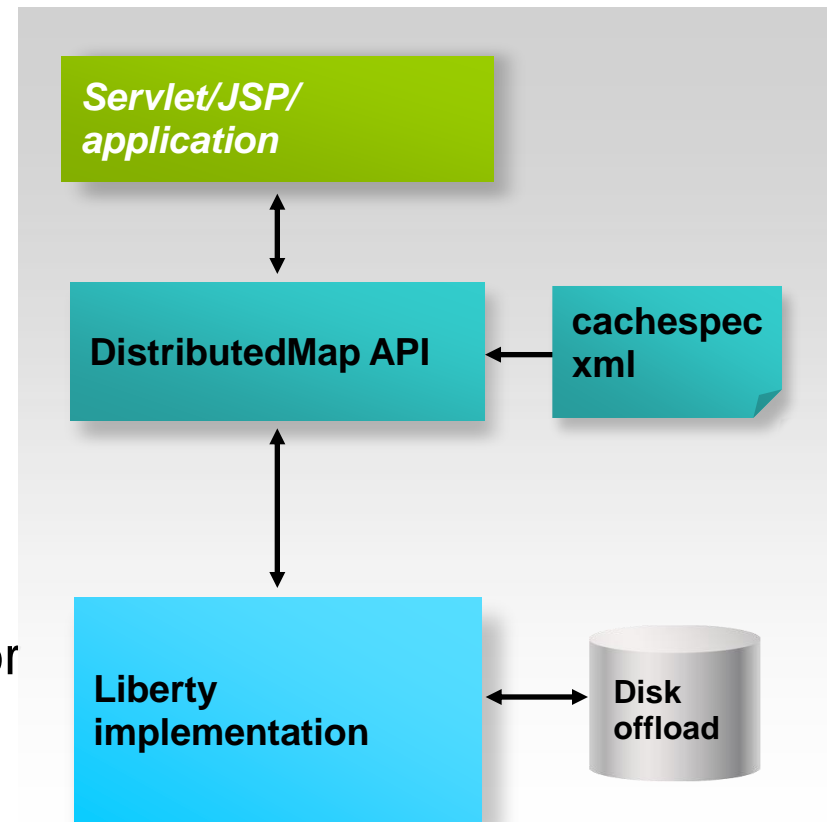
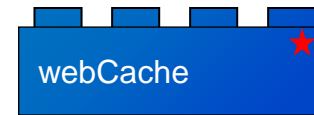


WAS v8.5 Liberty Profile (V8.5.5★)

Liberty – *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

- Local server cache for dynamic web content
- Enables developers to easily develop / test applications for local server caching solutions
- Same API as full WAS
- Optional disk offload
- LRU replacement policy
- Run as local server cache in production or move applications unchanged to distributed caching via WebSphere eXtreme Scale

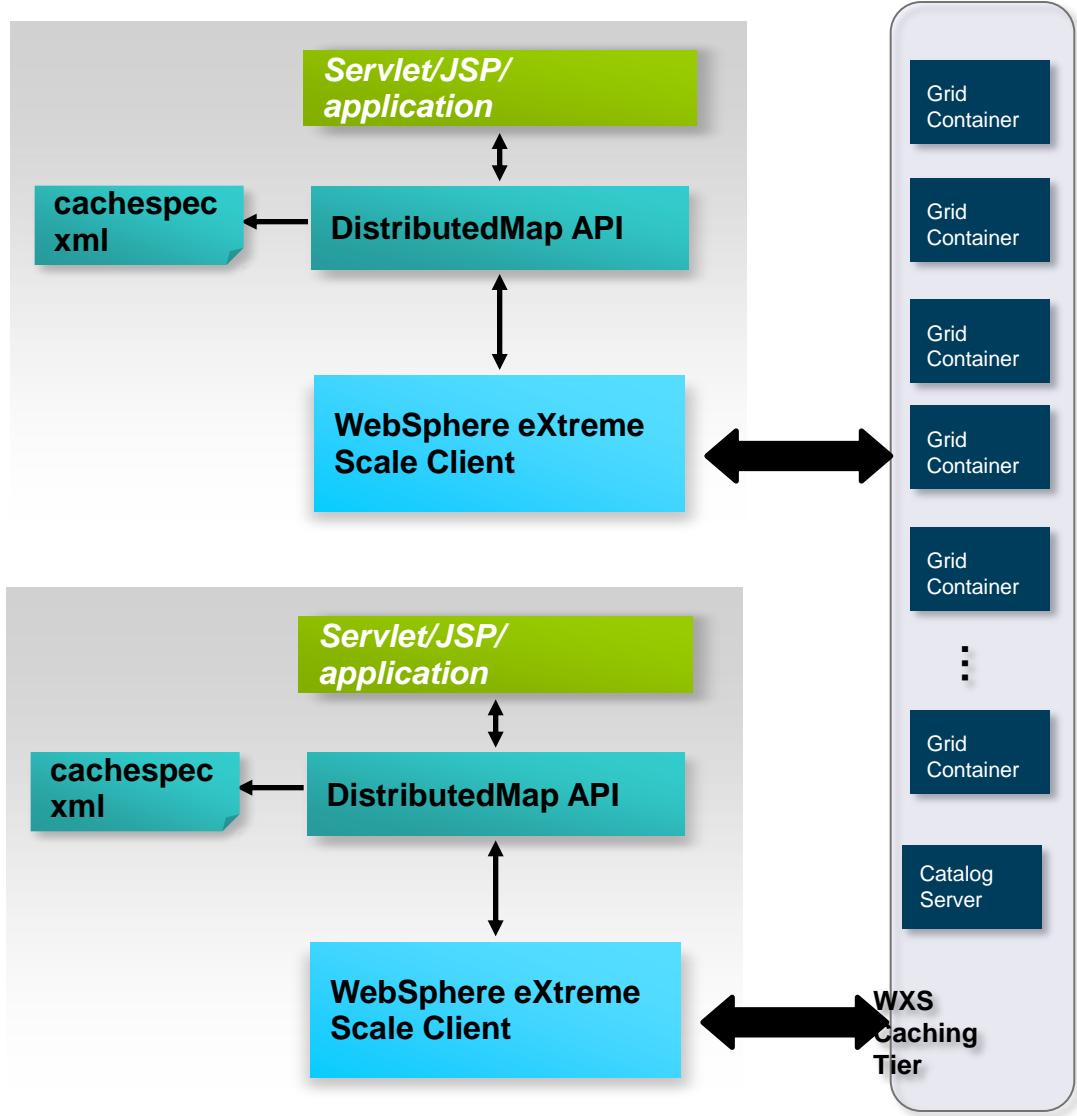


Upgrade to WebSphere Extreme Scale for Web Caching



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.



Platform Extensions – *SPI for Liberty Profile Extensions*



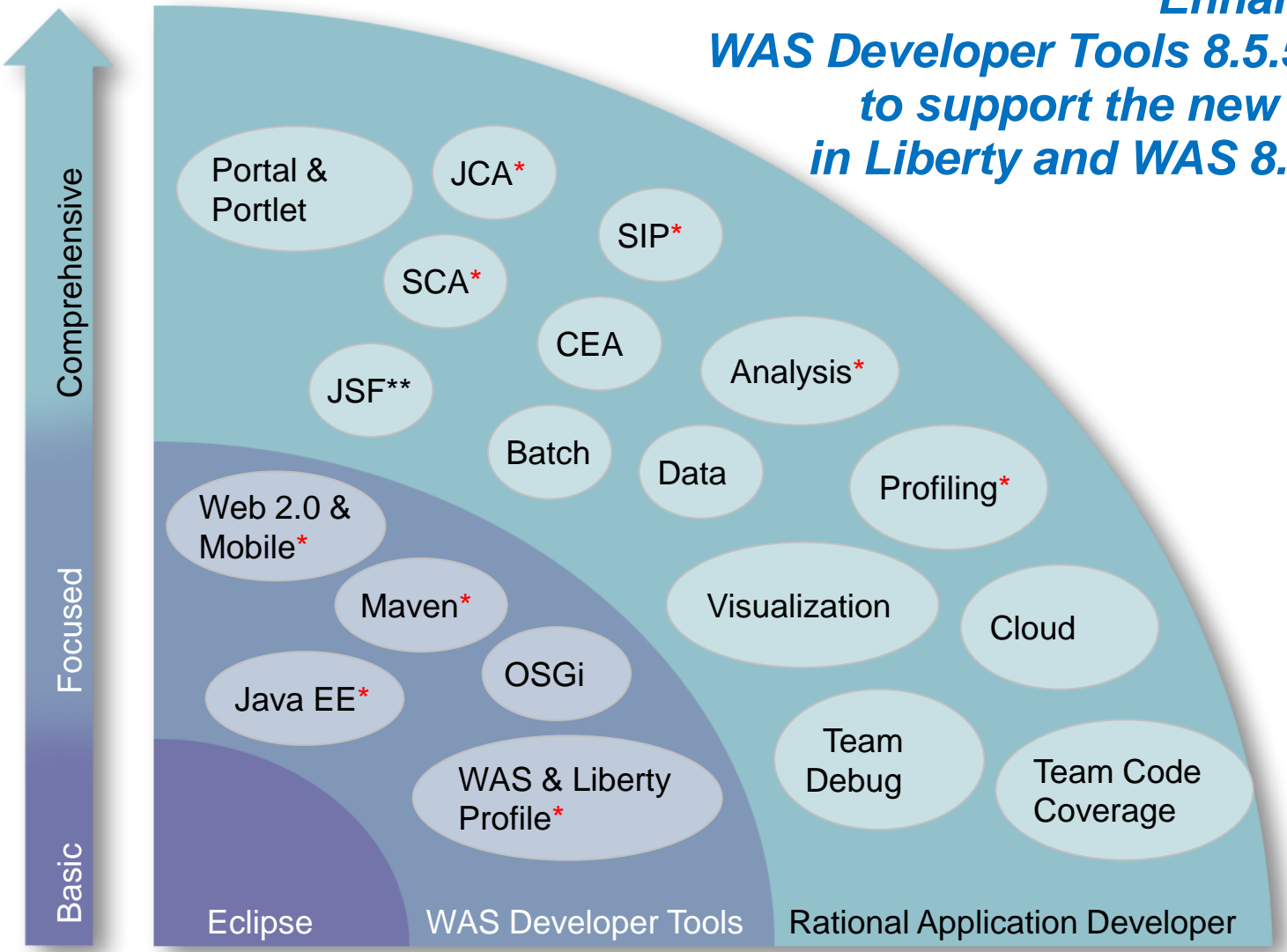
The Liberty Extensions System Programming Interface (SPI) provides the ability to extend the Liberty profile with custom features, including full life cycle management

- Add your own Liberty features via OSGi bundles
 - Web application bundles also supported
- Allows complete integration of third-party function into the Liberty runtime and development tools
- Full life cycle: packaging, installation, and un-installation
- System Programming Interface (SPI) for:
 - Adding your own configuration
 - Interaction with Liberty kernel
- Class visibility protected by class loader
- Sample scenarios:
 - Extend OSGi applications programming model
 - Provide new infrastructure (i.e. Custom User Registry)

Rational Application Developer and WAS Developer Tools



Enhancements in WAS Developer Tools 8.5.5 & RAD 9.0 to support the new capabilities in Liberty and WAS 8.5.5 editions



*Enhanced

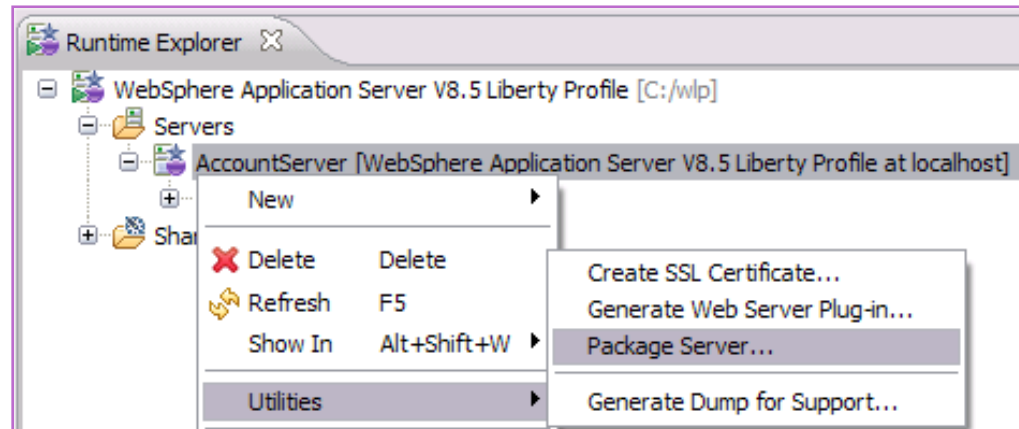


** JSF pgm model support in WDT. Enhanced tooling in RAD.

Liberty – “Packaged Server” Deployment Option



- 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.

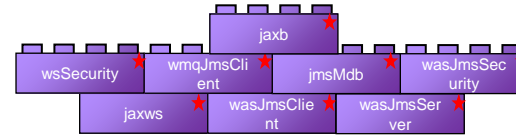


- For test automation outside the IDE, a command-line program to manage the lifecycle of server instances:
 - Create [serverName]
 - Start and stop [serverName]
 - Package [serverName]
 - Status [serverName]

```
c:\wlp>server package demoserver
Packaging up demoserver...
Server demoserver packaged to demoserver.zip.
c:\wlp>
```

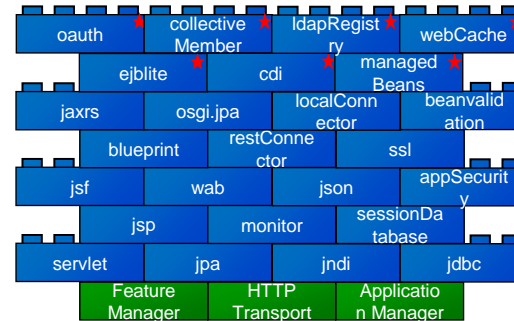
- Updates to configuration of running server are effective immediately.
- Add/remove apps dynamically by drag/drop to monitored directory.

Liberty – *Minimized Server Packages*



Liberty **extended**
archive 30MB

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

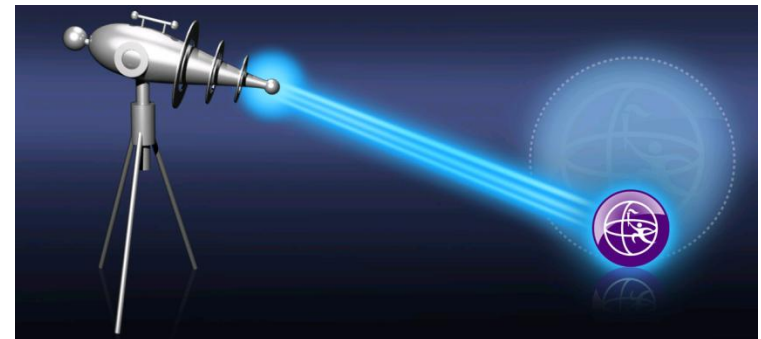


Liberty **runtime**
archive 50MB
(Java EE Web
Profile)

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

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

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

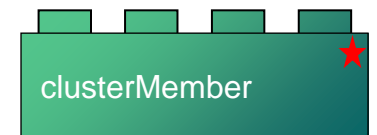
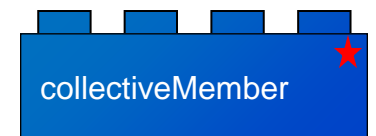


Liberty – *Collective Administration*



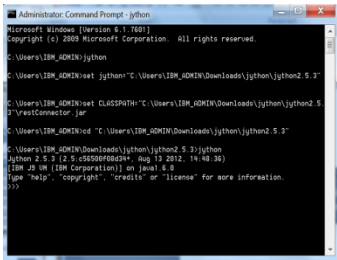
WAS v8.5.5 introduces a new administrative model enabling the grouping of Liberty servers into collectives for common management and static clustering

- New Liberty collective management infrastructure
 - No dependency on full WAS
 - Agent-less administration
 - Deployment of Liberty topology
 - 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
 - Managed server:
 - May be Liberty core, WAS Base, WAS ND, or WAS z/OS
 - Static cluster supported on WAS ND and WAS z/OS
- Highly available
- Highly scalable: Scales to 1000s of JVMs

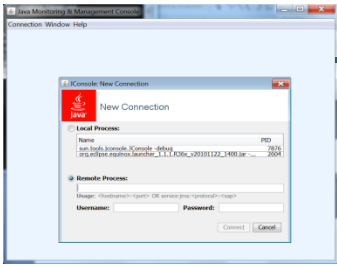




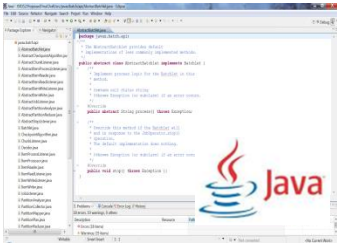
Liberty – Collectives and Clusters



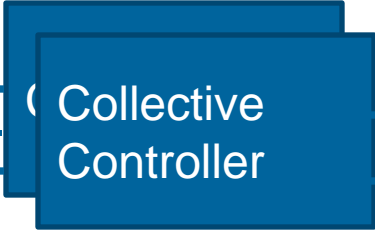
jython



jconsole



Java



- Only WAS ND and z/OS can be Collective Controller
- Provides Operational Registry
- Access all members through controller
- Highly Available
- Secure, Scalable
- Agentless
- No central config



Liberty Cluster A

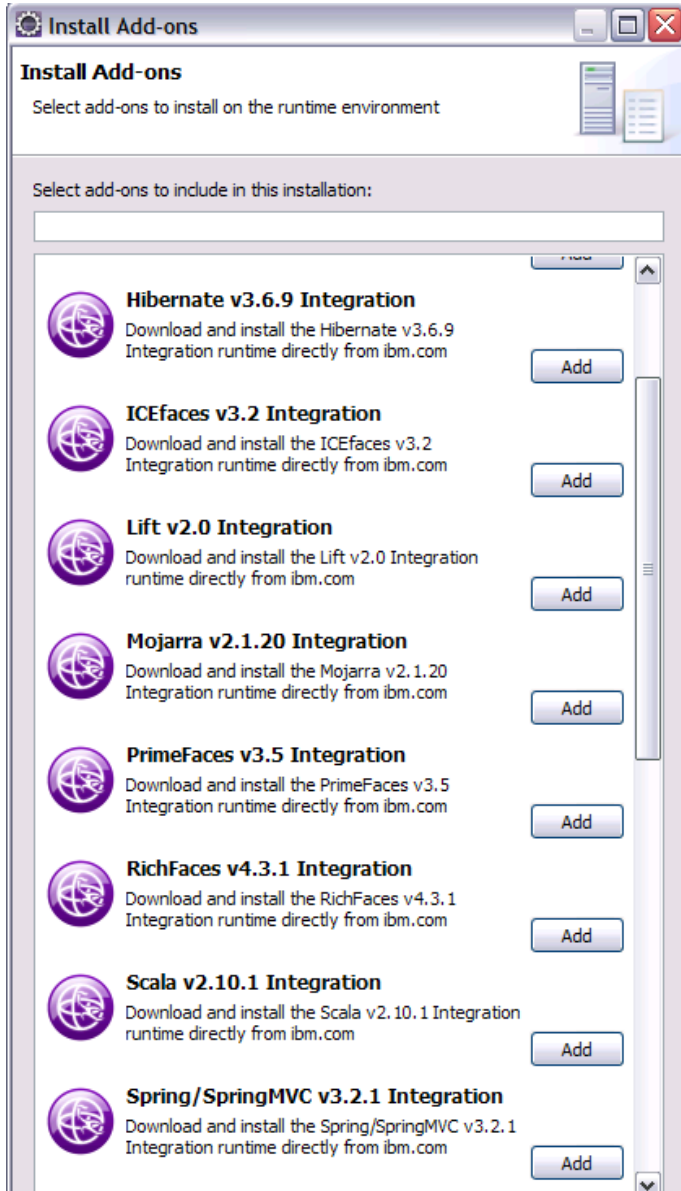


Liberty Cluster B

- All WAS editions eligible to join collectives as members
- Profiles self-assign to clusters (WAS ND and z/OS)
- Simple config updates
- Easily reversed, reconfigured

An Online Repository for Extending the Liberty Profile Runtime

- 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





WAS v8.5.5 Delivers

Fit for Purpose Servers enabling unmatched combination of Application Server runtime and development experience, from the highly resilient to the lightweight and nimble

Developer Experience

Fast, flexible, and simplified application development

- New lightweight **WebSphere Application Server Liberty Core** edition
- Liberty Profile
 - Java EE 6 Web Profile
 - EJB Lite, CDI, Managed Beans
 - Web Services
 - JMS, MDB
 - NoSQL DB (MongoDB Client)
 - Support for WebSphere Web Cache (DynaCache)
 - Add custom and third party Liberty features
- Asynchronous work management
- Enhanced developer tools
- Supported WAS and WDT on developer desktops

Application Resiliency

Intelligent Management and Enhanced Resiliency

- WAS ND and WAS z/OS full profile enhancements in v8.5 **
 - App Edition Mgmt
 - App Server Health Mgmt
 - Dynamic Clustering
 - Intelligent Routing
 - Messaging resiliency
 - Enterprise Java Batch
 - Memory leak protection
- Liberty Profile collective administration
- Liberty Profile clustering
- Intelligent management in WebSphere web server

Operational Excellence

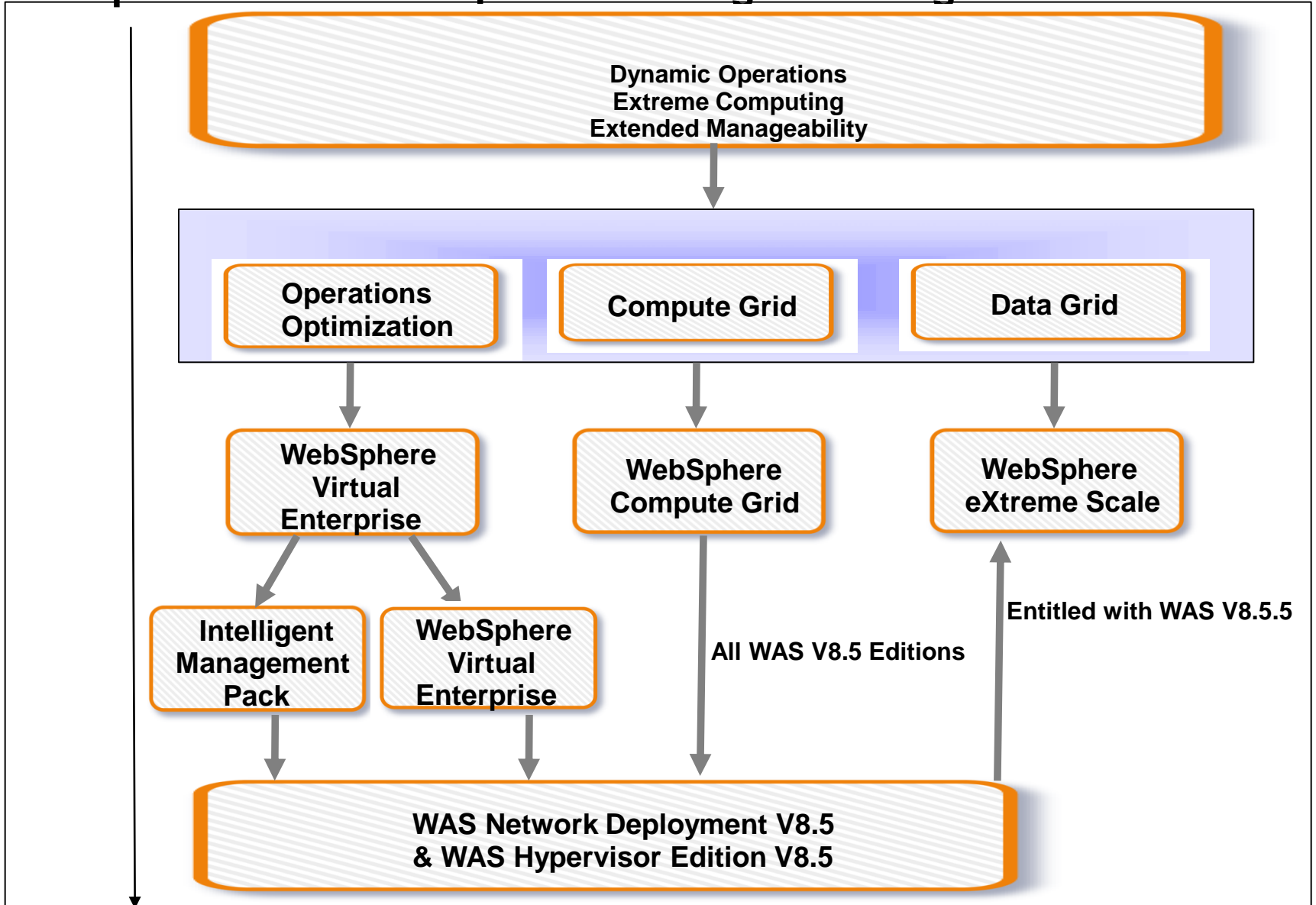
Improved Operations, Security, Control & Integration

- Liberty profile
 - Security enhancements
 - Problem determination
 - Monitoring
- Service Mapping
- SIP improvements
- Improved Load Balancer for IPV4 and IPV6
- Liberty profile packaging and install enhancements
- WebSphere Extreme Scale integration
- Performance enhancements

** Available since WAS 8.5



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

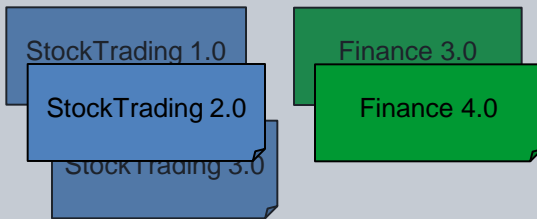




Intelligent Management

Extending QoS through autonomic computing

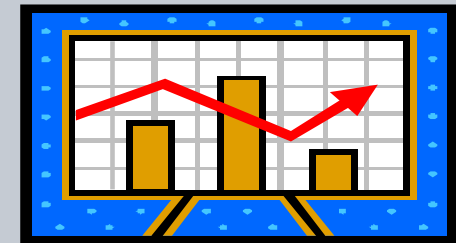
**Application
Edition
Management**
Self-Managing



**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

**Validation
Mode**



**Rollout
Policies**



**Concurrent
Activation**



Health Management

Sense and respond to problems before end users suffer an outage

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

**Comprehensive
Health Policies**



**Customizable
Health
Conditions**



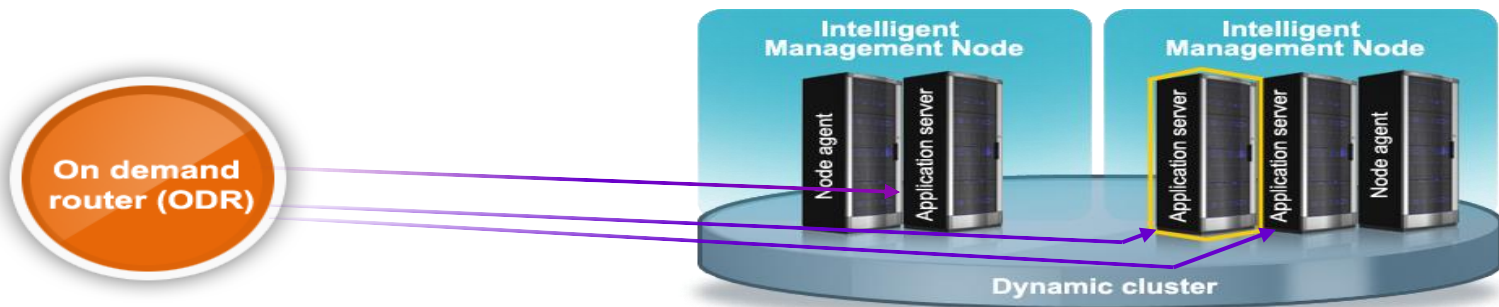
**Customizable
Health
Actions**



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

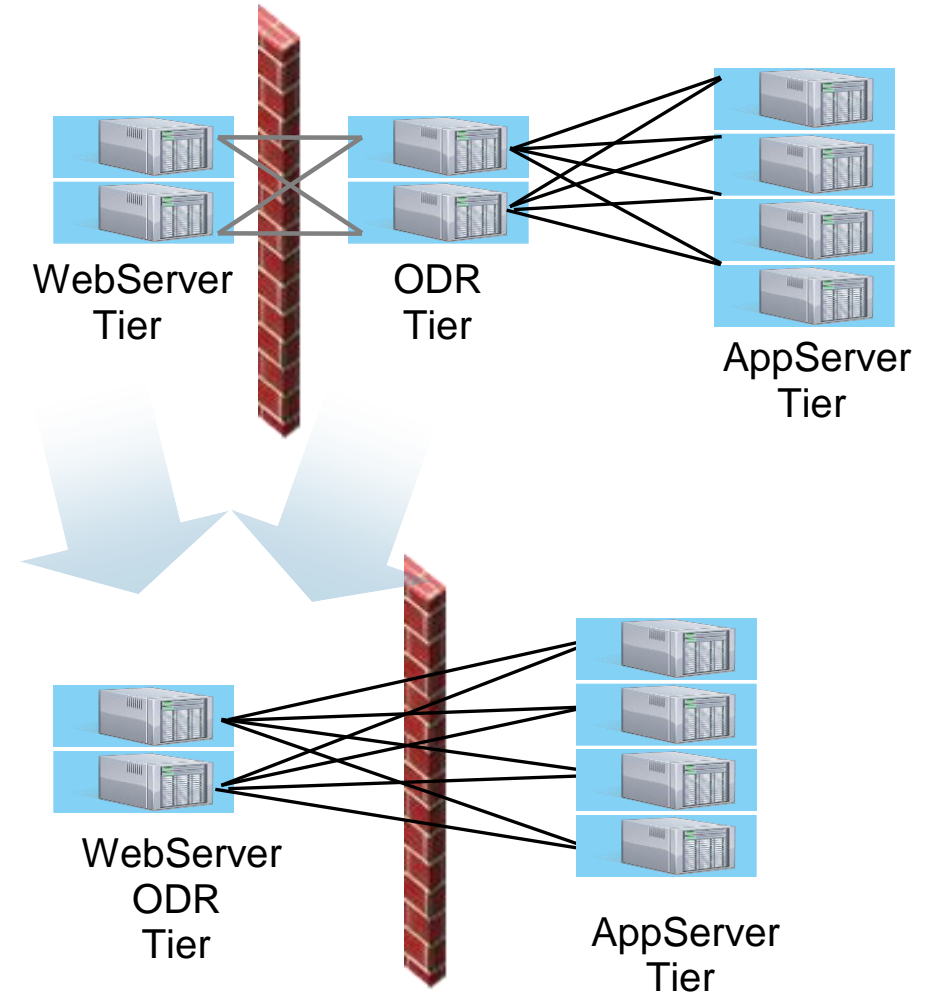
- Requests are prioritized and routed based upon administrator defined rules
 - Flexible policy-based routing and control
- On Demand Router (ODR) is the focal point for Intelligent Routing
- A routing tier that's aware of what's happening on the application server tier
 - Application server utilization,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



Intelligent Management in WebSphere WebServer

v8.5.5 enhances Intelligent Management through integration of the On Demand Router capability into the WebSphere Web Server plug-in, simplifying topologies and reducing TCO

- ODR features integrated into the WebSphere WebServer plugin in WAS ND
- Supported in IHS and Apache web servers
- Key Benefits
 - Topology consolidation and simplification
 - Increased resiliency
 - Fewer resources
 - Lower latency
 - Lower TCO (config/monitoring)
 - Faster problem determination





WAS v8.5.5 Delivers

Fit for Purpose Servers enabling unmatched combination of Application Server runtime and development experience, from the highly resilient to the lightweight and nimble

Developer Experience	Application Resiliency	Operational Excellence
<p><i>Fast, flexible, and simplified application development</i></p> <ul style="list-style-type: none"> ■ New lightweight WebSphere Application Server Liberty Core edition ■ Liberty Profile <ul style="list-style-type: none"> ■ Java EE 6 Web Profile <ul style="list-style-type: none"> ■ EJB Lite, CDI, Managed Beans ■ Web Services ■ JMS, MDB ■ NoSQL DB (MongoDB Client) ■ Support for WebSphere Web Cache (DynaCache) ■ Add custom and third party Liberty features ■ Asynchronous work management ■ Enhanced developer tools ■ Supported WAS and WDT on developer desktops 	<p><i>Intelligent Management and Enhanced Resiliency</i></p> <ul style="list-style-type: none"> ■ WAS ND and WAS z/OS full profile enhancements in v8.5 ** <ul style="list-style-type: none"> ■ App Edition Mgmt ■ App Server Health Mgmt ■ Dynamic Clustering ■ Intelligent Routing ■ Messaging resiliency ■ Enterprise Java Batch ■ Memory leak protection ■ Liberty Profile collective administration ■ Liberty Profile clustering ■ Intelligent management in WebSphere web server 	<p><i>Improved Operations, Security, Control & Integration</i></p> <ul style="list-style-type: none"> ■ Liberty profile <ul style="list-style-type: none"> ■ Security enhancements ■ Problem determination ■ Monitoring ■ Service Mapping ■ SIP improvements ■ Improved Load Balancer for IPV4 and IPV6 ■ Liberty profile packaging and install enhancements ■ WebSphere Extreme Scale integration ■ Performance enhancements

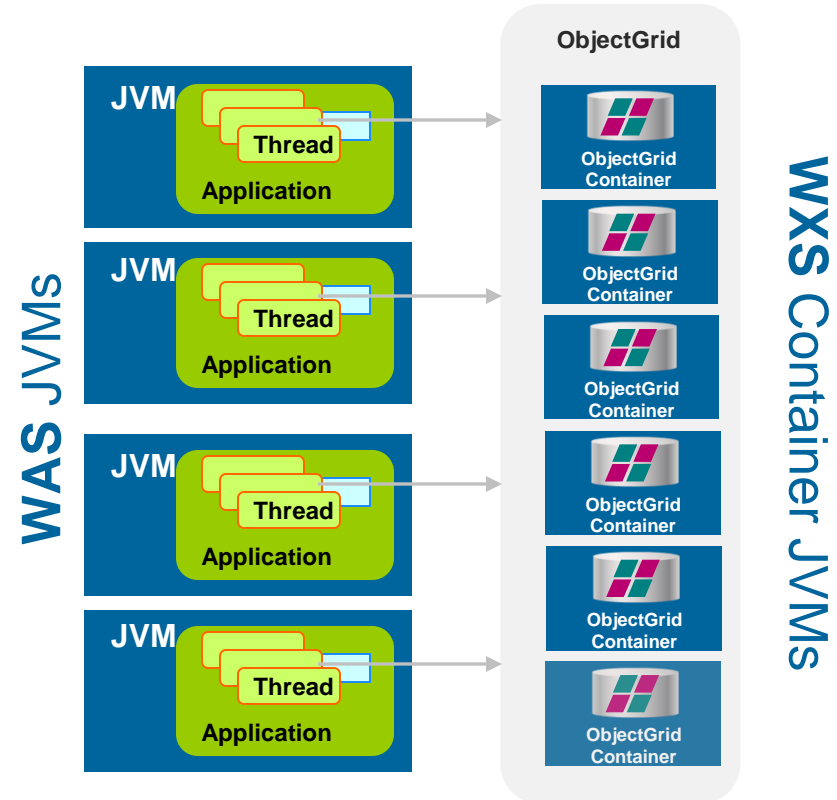
** Available since WAS 8.5



WebSphere eXtreme Scale Integration

Easily enable applications for caching of data to improve application performance, leveraging WebSphere eXtreme Scale packaging and entitlement with WAS v8.5.5

- WebSphere eXtreme Scale packaged with WAS v8.5.5 with entitlement by edition
- Enables installation of elastic caching with WXS on WAS server out of the box
- Gain the advantage of integrated caching
- Grow your environment to distributed caching servers as load increases

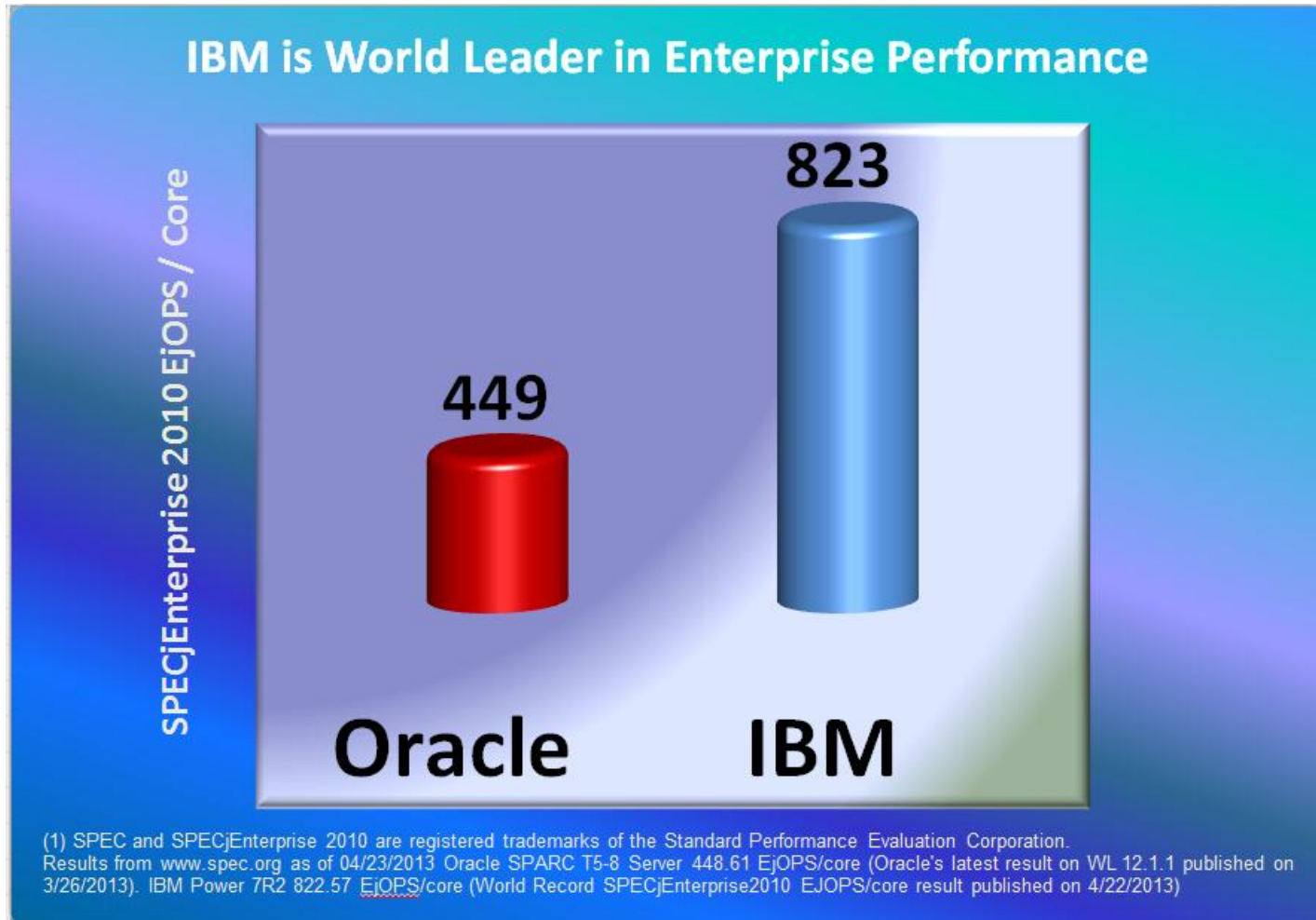


Entitlement by Edition

- ND = Unlimited
- Base = HTTP Session and DynaCache
- z/OS = WXS client on z/OS
- Express/Liberty Core = N/A



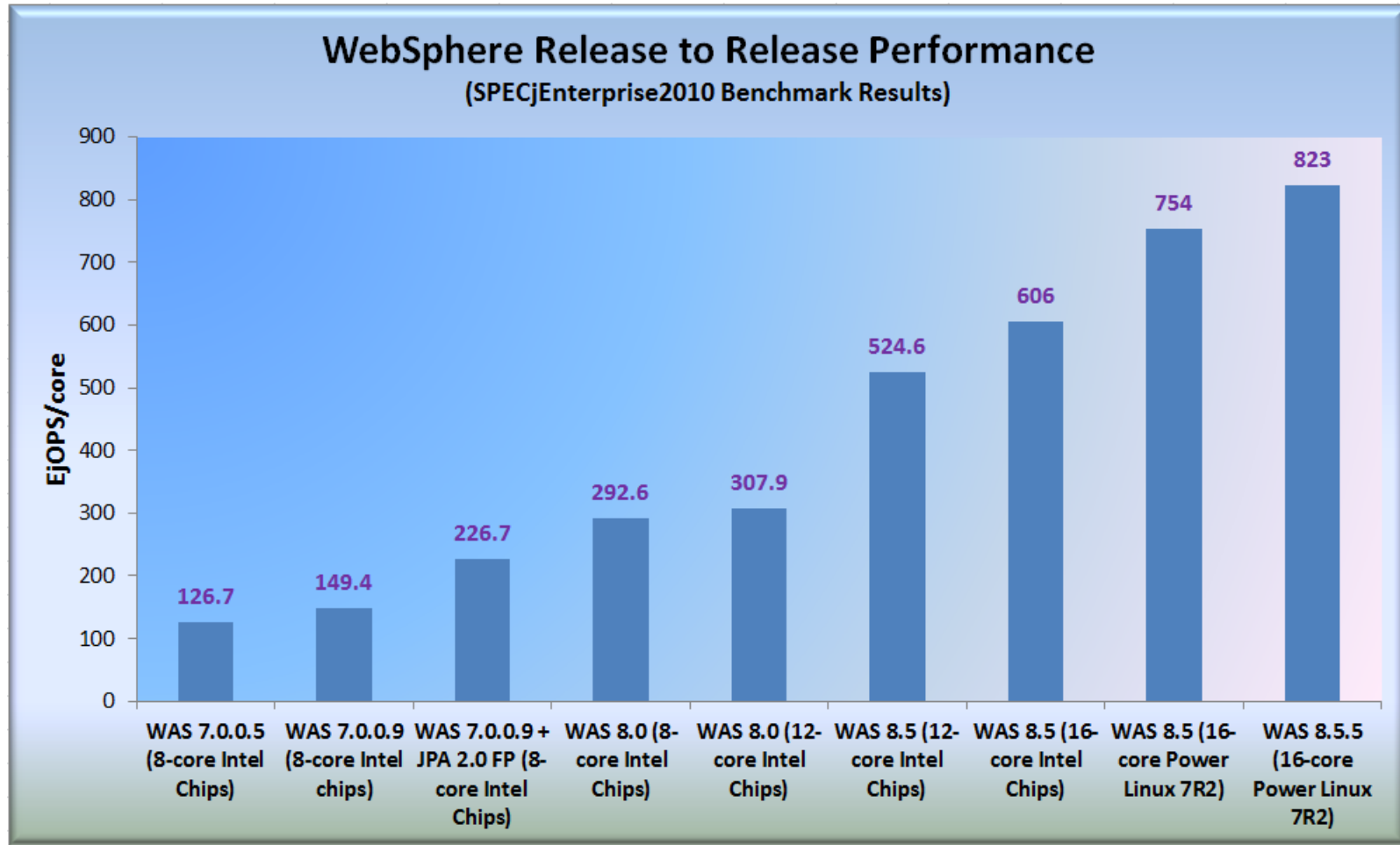
IBM is World Leader in Enterprise Performance



- IBM Middleware running on Power 7 beats Oracle Middleware running on SPARC T5 Servers by almost 2x on a per core level
- Comparison is based on latest WL 12.1.1 and WAS 8.5.5 SPECj Results



WebSphere Release to Release Performance



Consistent Performance gains across WAS Releases

As per SPEC Published Data as of 4/26/2013

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

Liberty Startup and Footprint (TradeLite)

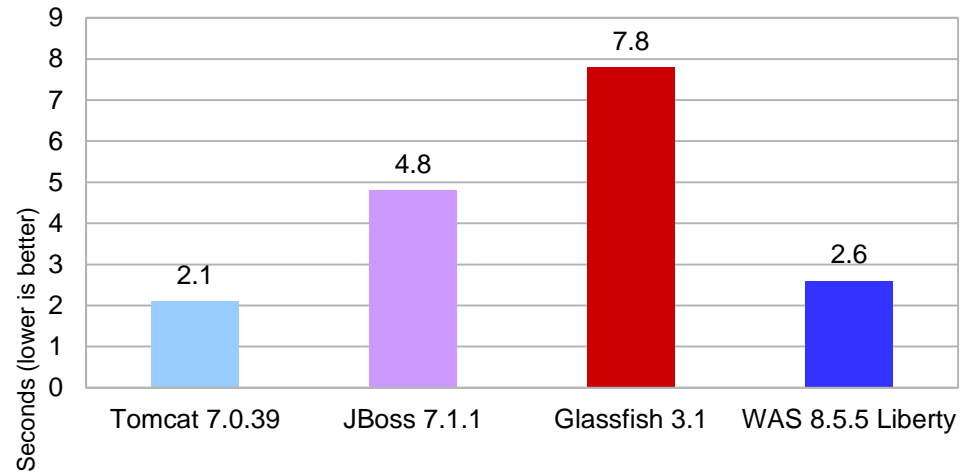


- ✓ WAS 8.5.5 Liberty Server startup and memory footprint is comparable to or faster than other lightweight App Servers
- ✓ TradeLite is a very simple benchmark...as the workload gets more complex the advantages with Liberty increase further...

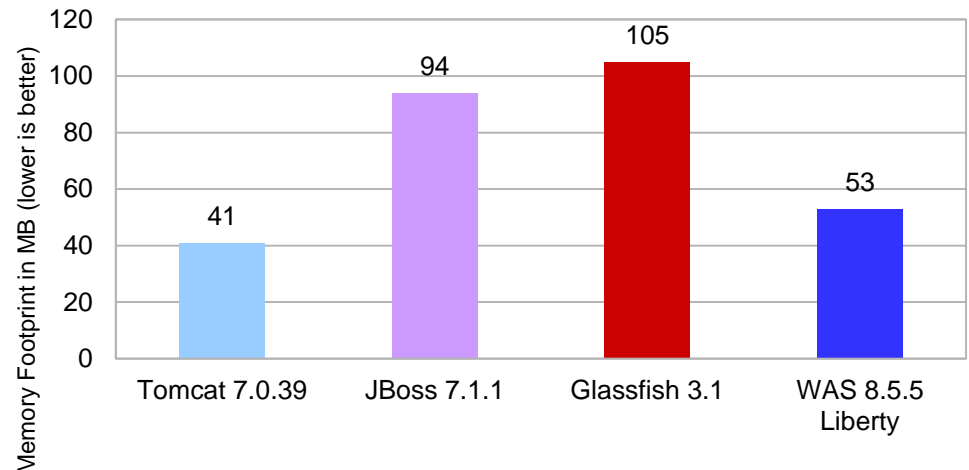
System Configuration:

SUT: T60p Thinkpad Windows XP 32-bit Intel Core 2 Duo CPU @ 2.16 GHz [2 cores] 2MB L2, 2GB RAM
IBM JDK 7 sr4fp1ifix-20130404 used with Liberty
Oracle JDK 7 su21 used for other products

Startup Time Comparison of Lightweight Servers (TradeLite)



Memory Footprint Comparison of Lightweight Servers (TradeLite)



Liberty Startup and Footprint (DayTrader3)



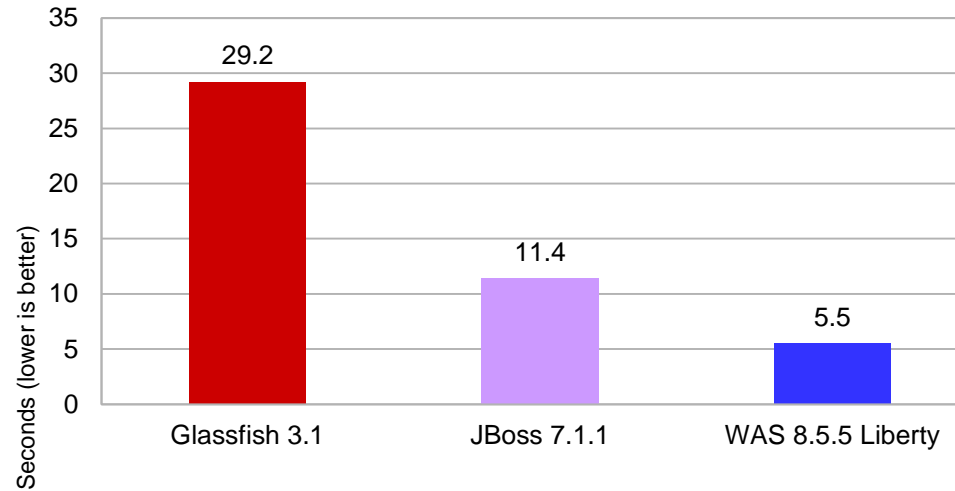
- ✓ DayTrader3 is an update of the Apache DayTrader benchmark for Java EE 6 using Web and EJB technologies, JPA and JMS.
- ✓ Liberty starts in half the time of JBoss and Glassfish is way behind.

Note: Note that Tomcat was excluded from this comparison because it does not have J2EE features needed to run the DayTrader3 application.

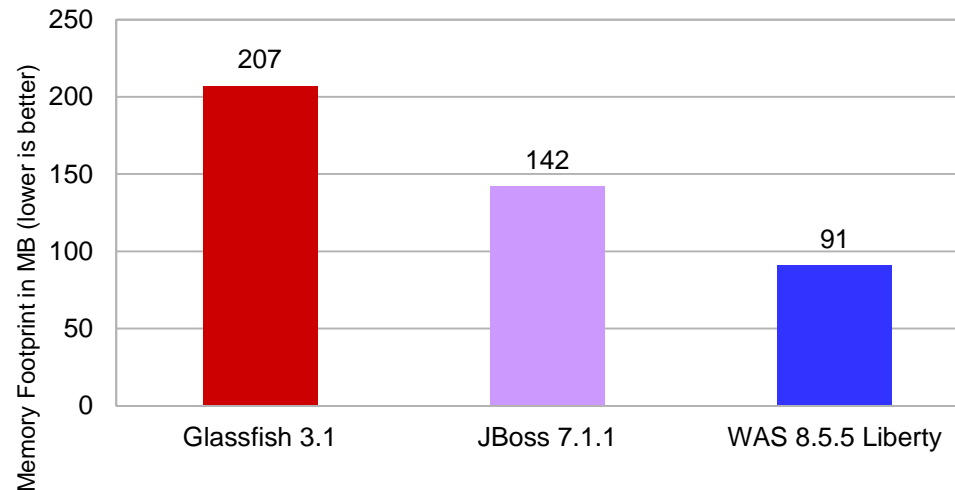
System Configuration:

SUT: T60p Thinkpad Windows XP 32-bit Intel Core 2 Duo CPU @ 2.16 GHz [2 cores] 2MB L2, 2GB RAM
IBM JDK 7 sr4fp1ifix-20130404 used with Liberty
Oracle JDK 7 su21 used for other products

Startup Time Comparison of Lightweight Servers (DayTrader3)



Memory Footprint Comparison of Lightweight Servers (DayTrader3)





Runtime Performance Comparison

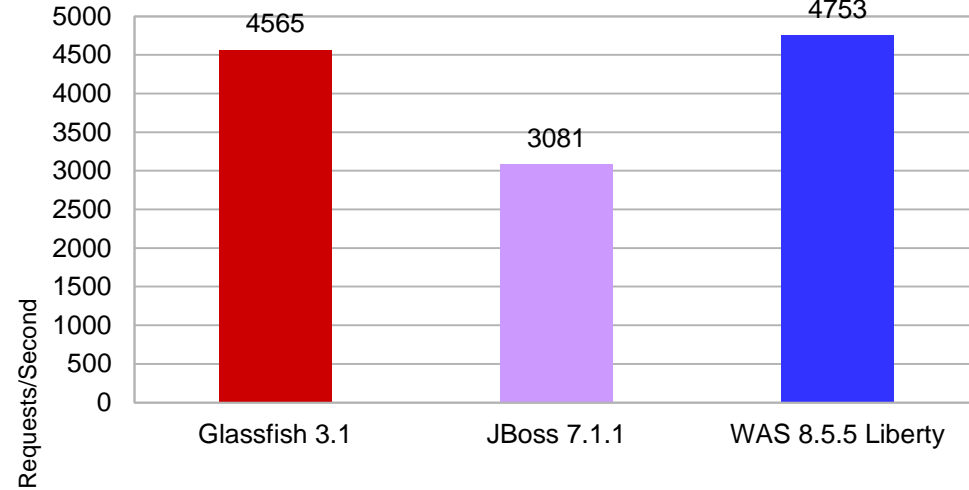
✓ Its not just about development environments – in production, WAS V8.5.5 Liberty outperforms the other servers too.

Note: Tomcat is compared using TradeLite since it does not provide the features to run the DayTrader3 workload.

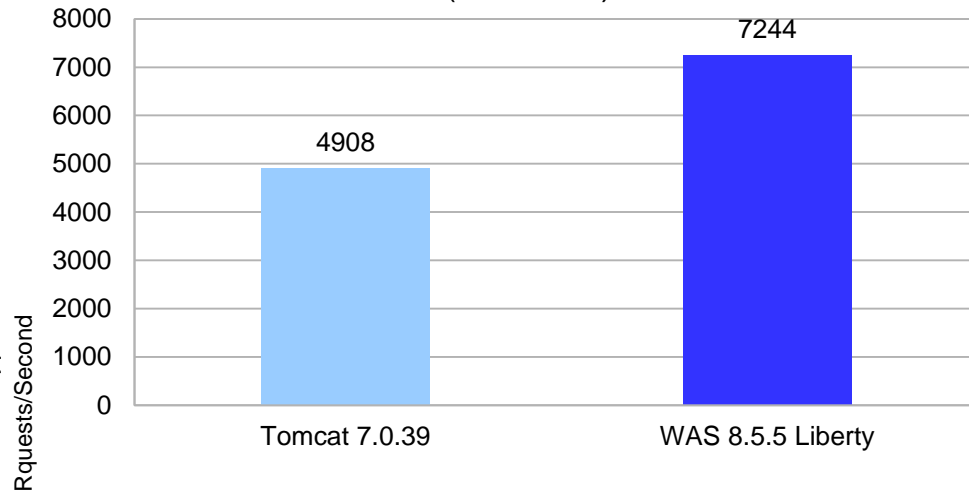
System Configuration:

SUT: Intel Westmere with 4 cores enabled, Linux 64-bit
IBM JDK 7 sr4 used with Liberty
Oracle JDK 7 su17 used for other products

Throughput Comparison of Lightweight Servers (DayTrader3)



Throughput Comparison of Lightweight Servers (TradeLite)



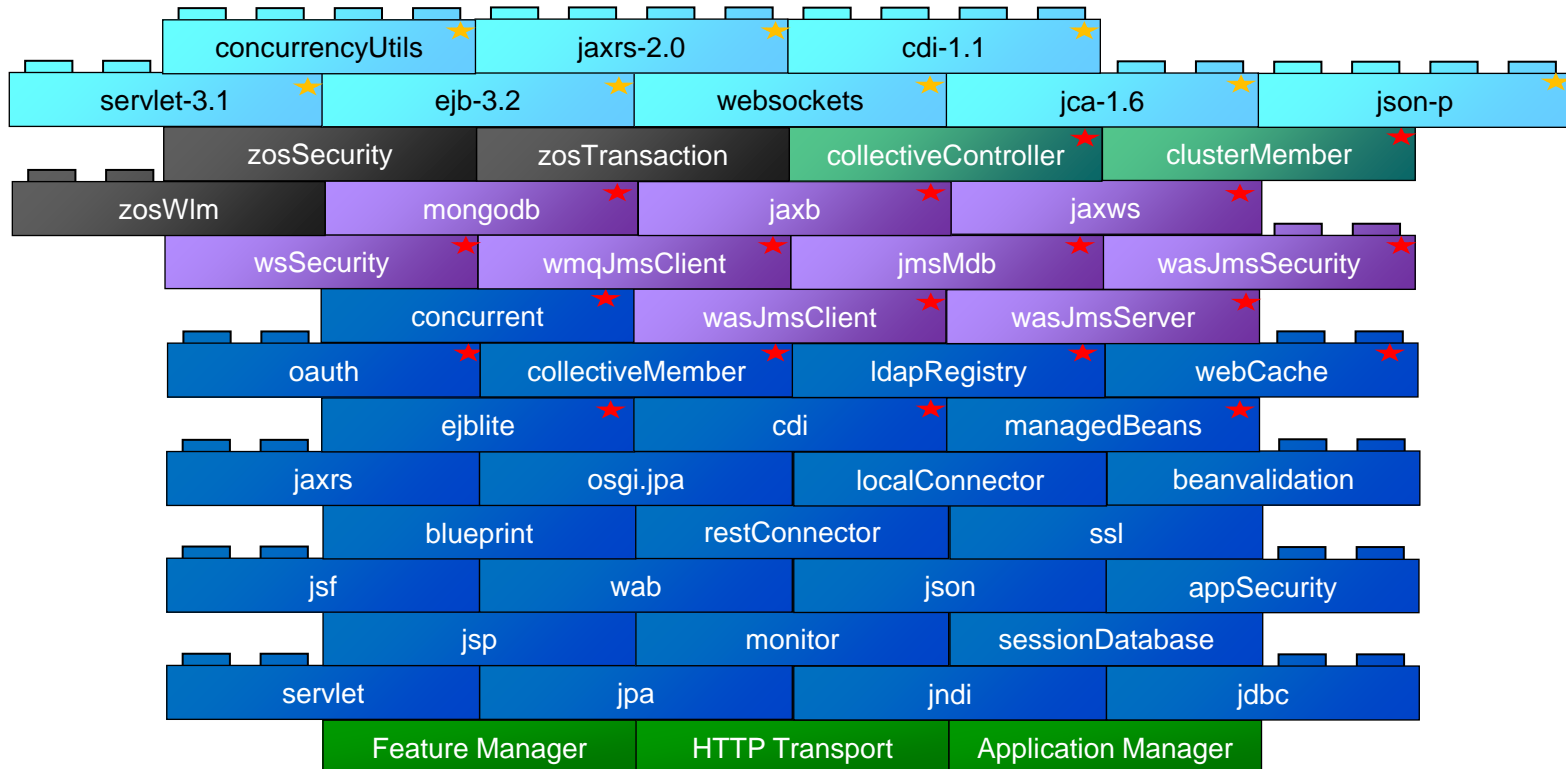


Looking Ahead



New features and programming models

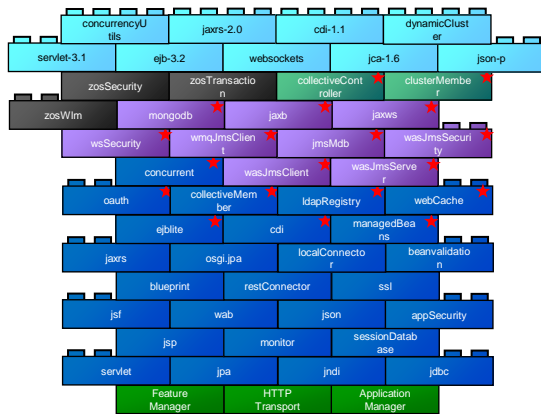
- Java EE 7 is now finalized.
- The composable WAS Liberty Profile enables us to target specific high-value JSRs.
 - Potential Liberty features:



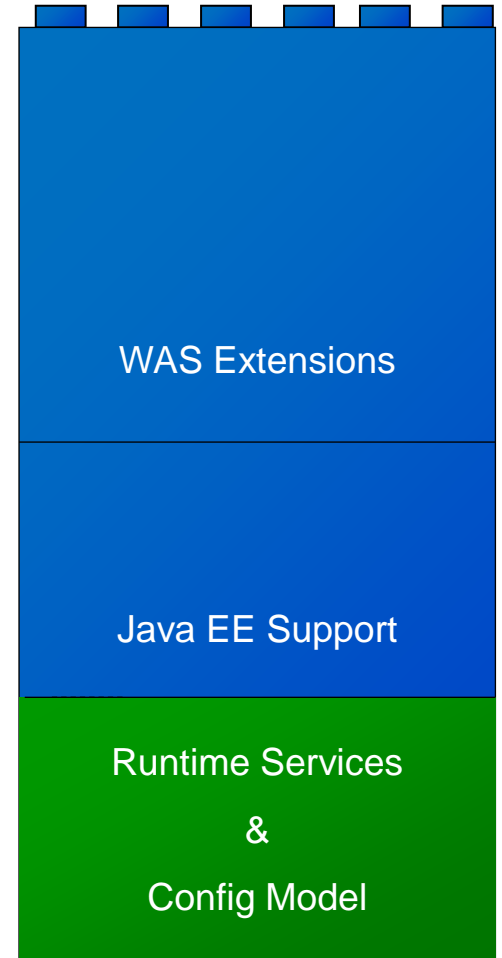


WAS Full Profile and Liberty Profile

- Programming models from WAS full profile that we will retain on the WAS full profile only:
 - Pre JEE6 technologies e.g.
 - JAX-RPC and WSIF
 - EJB 2.x and CMP
 - SDO/E
 - JAXR/UDDI
 - Startup Beans and Custom services
(Standard aspects of Liberty features)



WAS Liberty Profile



Full WAS Profile

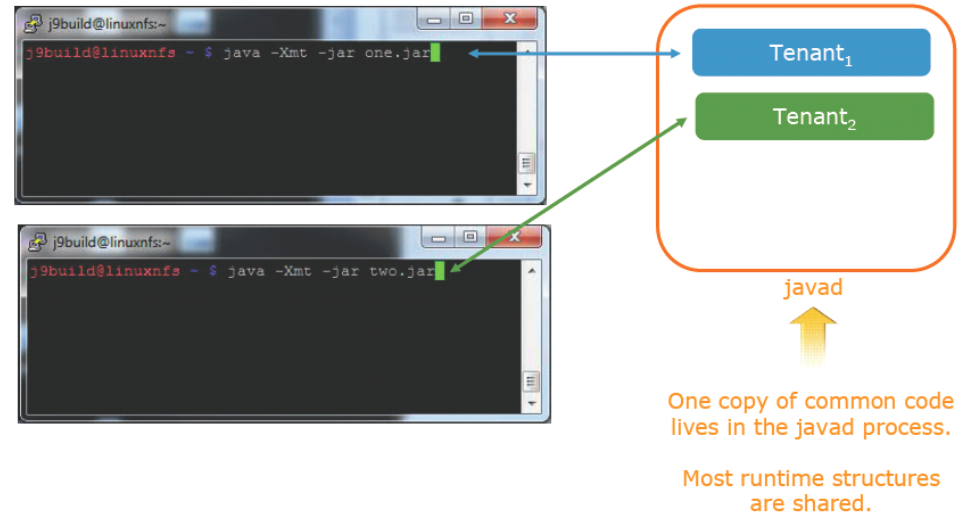
Multitenant JDK

- The IBM JDK intends to support multiple tenants in the same JVM
→ reducing resources consumption even further.
- Multiple instances of WAS Liberty profile servers in the same JVM
 - Even faster startup
 - Even smaller footprint

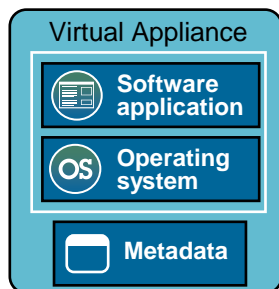
- -Xmt option to launch Tenant in shared **javad** daemon process
- Tenants resources can be limited to ranges:

-
Xlimit:resource_name=minLimit
it – maxLimit

- cpu
- threads
- diskIO
- netIO



Pattern-based Deployment

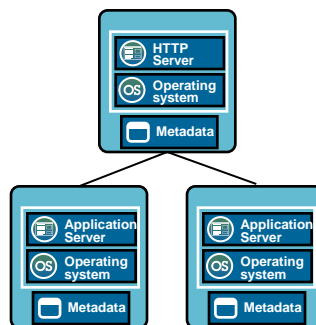


Virtual Appliances

- Standard software installation and configuration on OS
- Images created through extend/capture
- Traditional administration and management model
- Infrastructure driven elasticity

Virtual Appliances

Standard TCO
existing applications



Virtual System Patterns

- Automated deployment of middleware topology patterns
- Traditional administration and management model
- Application and infrastructure driven elasticity

Virtual System Patterns

Improved TCO
virtualized applications



Virtual Application Patterns

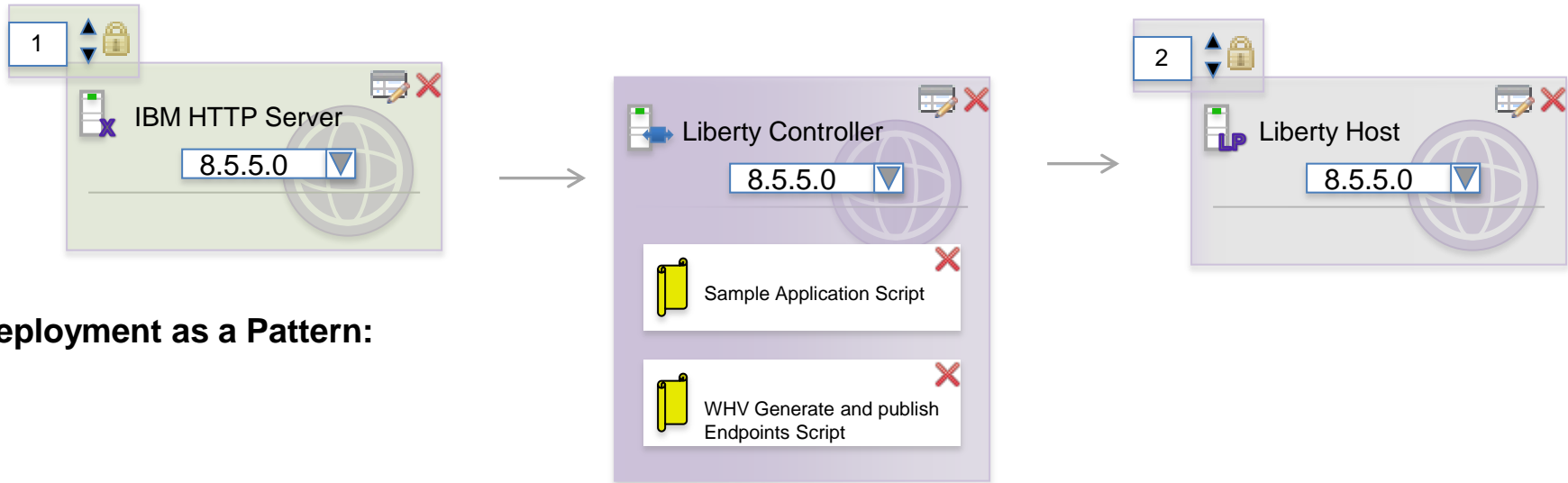
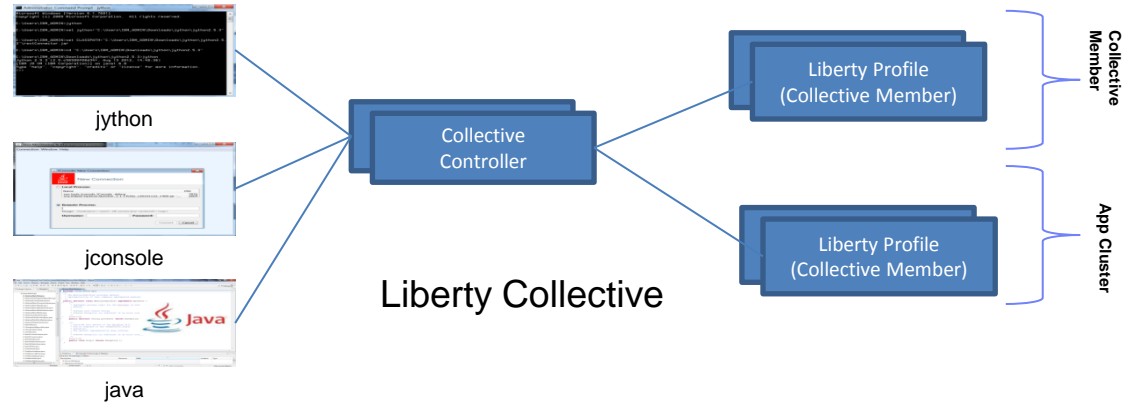
- Highly automated deployments using expert patterns
- Business policy driven elasticity
- Built for the cloud environment
- Leverages elastic workload management services

Virtual Application Patterns

Best TCO
cloud applications

High Density Virtual System Patterns

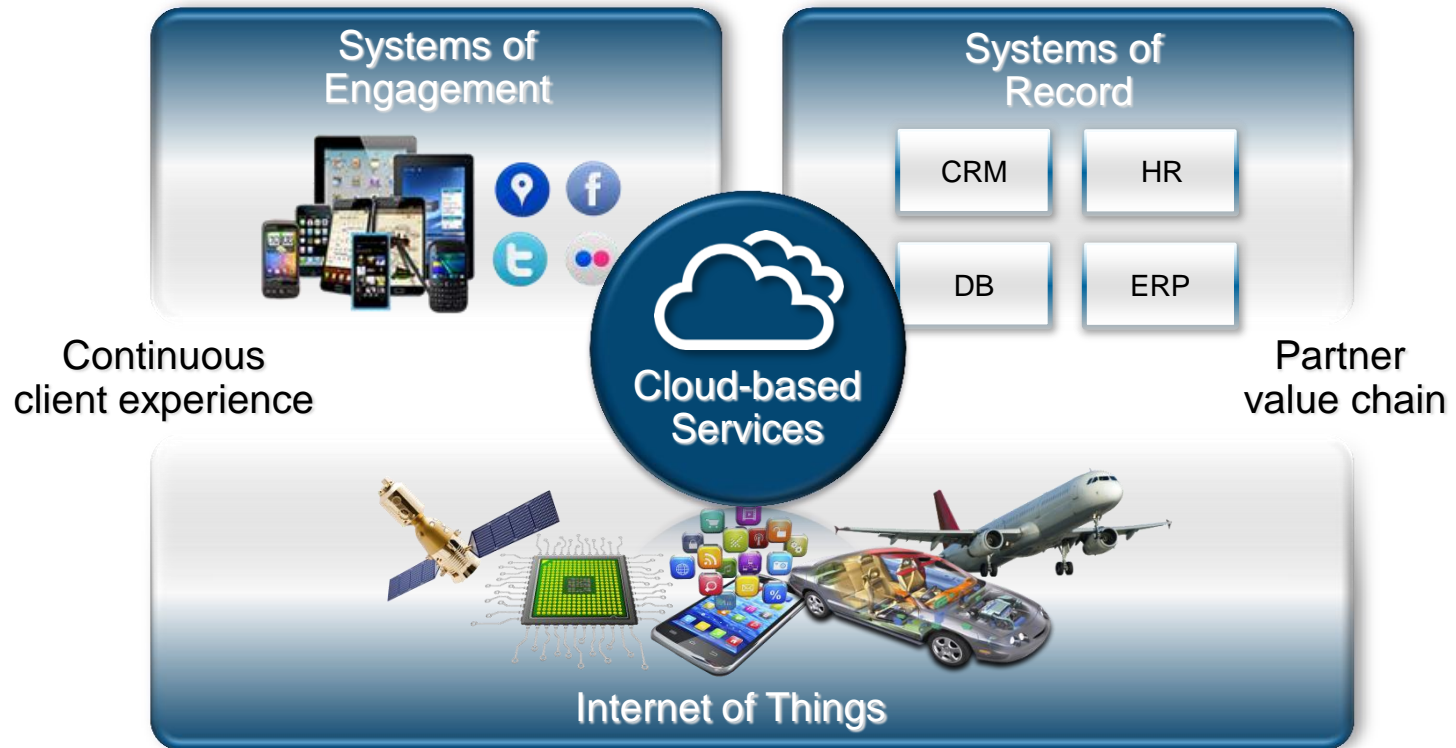
Remember this:



Deployment as a Pattern:

- Quickly create and deploy high-density virtual system patterns:
 - Liberty Controller
 - Scalable number of Liberty Hosts
- Can be tuned and managed in exactly the same way as a non-patterned deployment.

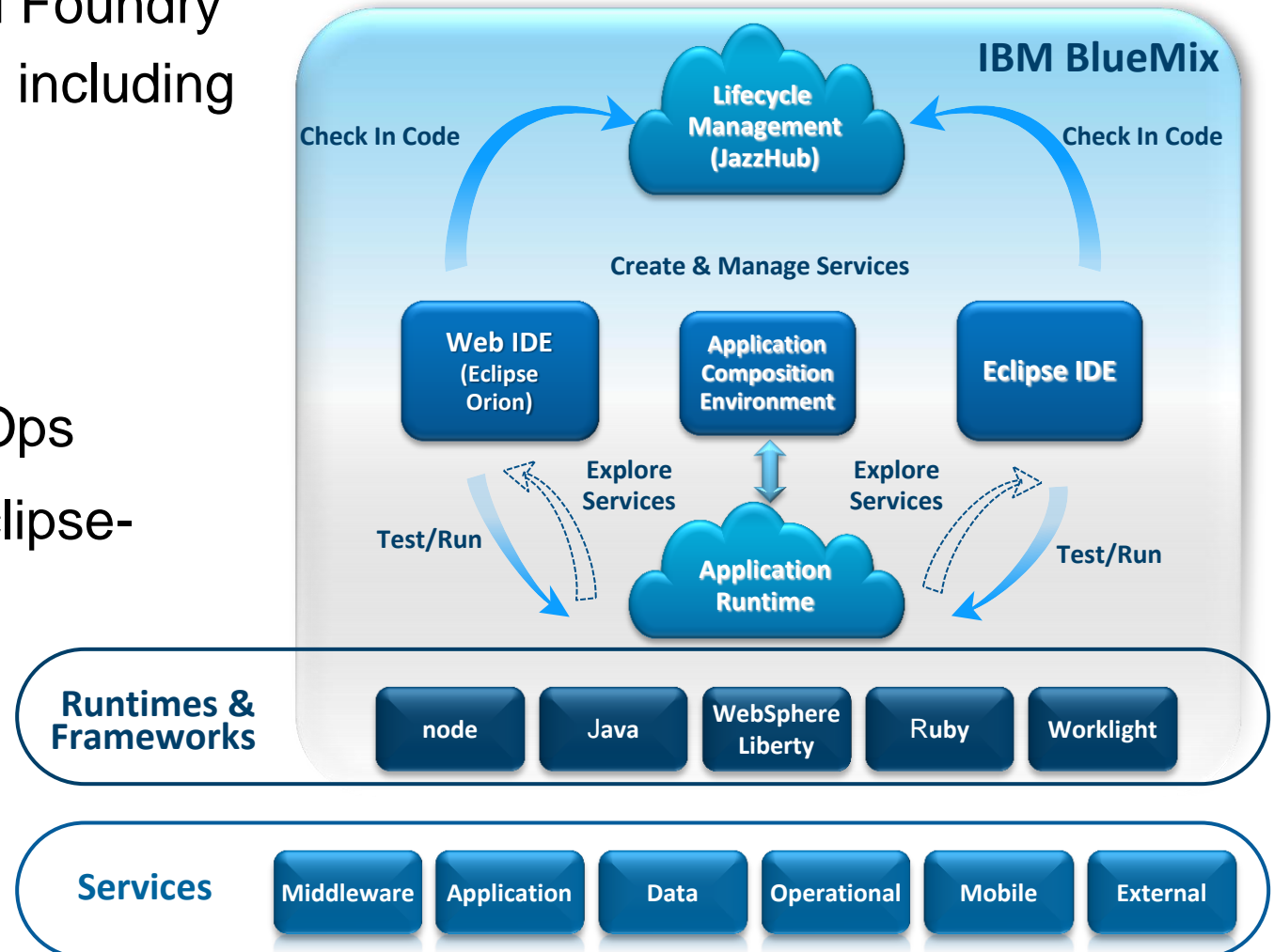
Systems of Interaction



IBM BlueMix

- Open Cloud PaaS
- Based on Cloud Foundry running on IaaS including OpenStack
- Runtimes and Frameworks
- Integrated DevOps
- Browser and Eclipse-based tools

Experiment @ <http://bluemix.net>

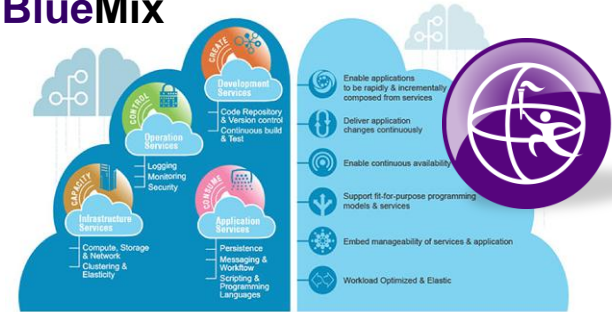


IBM WebSphere Liberty Buildpack

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>





**Thank
You**