What’s New with Rational Application Developer and Development with WebSphere Application Server

Chuck Bridgham – Architect, Rational Java EE Tools(cbridgha@us.ibm.com)

IBM Software
Innovate2012
The Premier Event for Software and Systems Innovation

Next NOW!

Please note

IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM’s sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

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

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Agenda

The business of developing software

Developing applications for people

Developing services for applications

Rapid development and testing

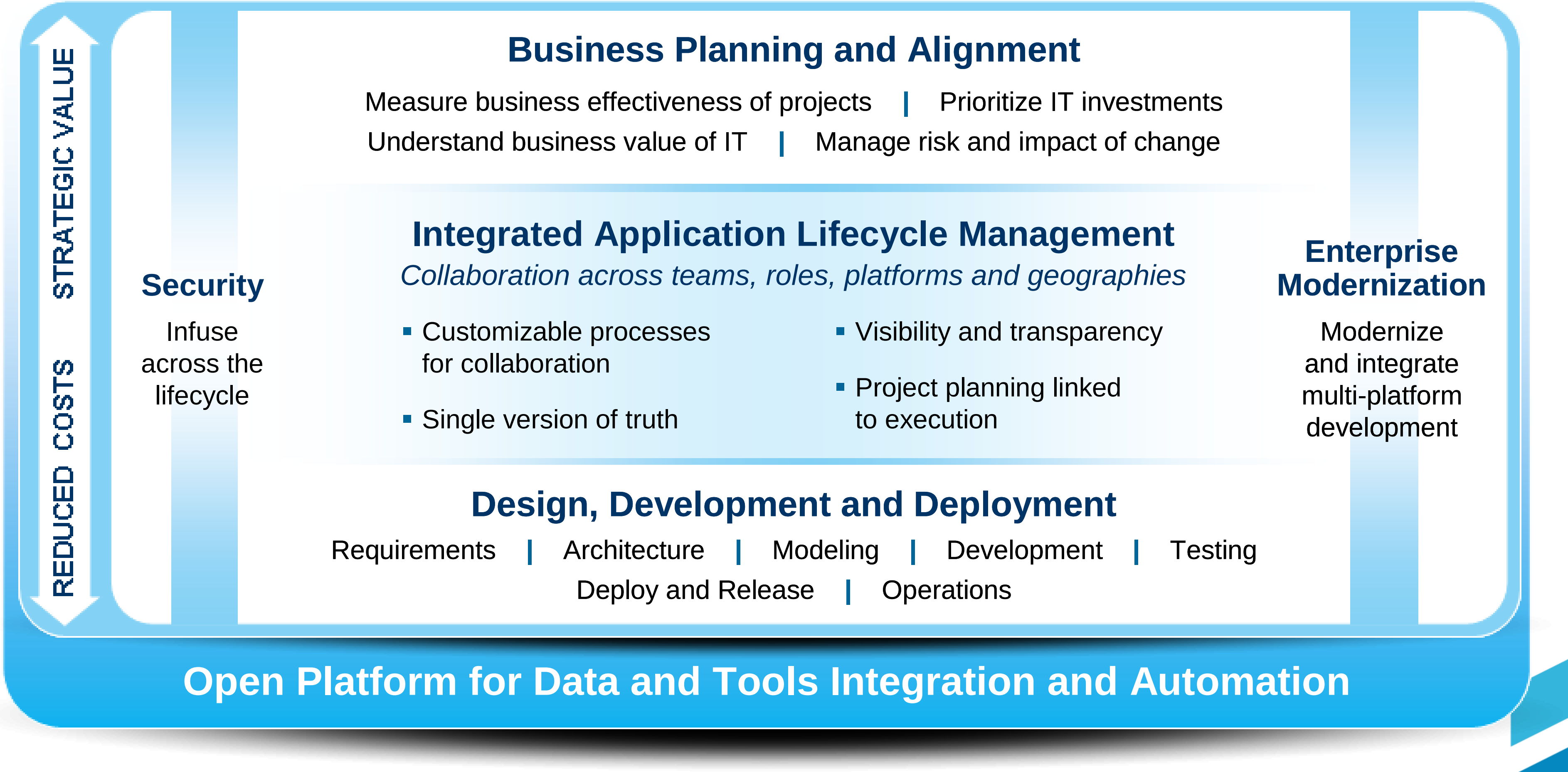
Collaboration and quality

Cloud computing

Epilogue

- *The challenges and opportunities of software development*

The grand scheme of things



STRATEGIC VALUE

REDUCED COSTS

Business Planning and Alignment

Measure business effectiveness of projects | Prioritize IT investments

Understand business value of IT | Manage risk and impact of change

Security

Infuse across the lifecycle

Integrated Application Lifecycle Management

Collaboration across teams, roles, platforms and geographies

- Customizable processes for collaboration
- Single version of truth
- Visibility and transparency
- Project planning linked to execution

Enterprise Modernization

Modernize and integrate multi-platform development

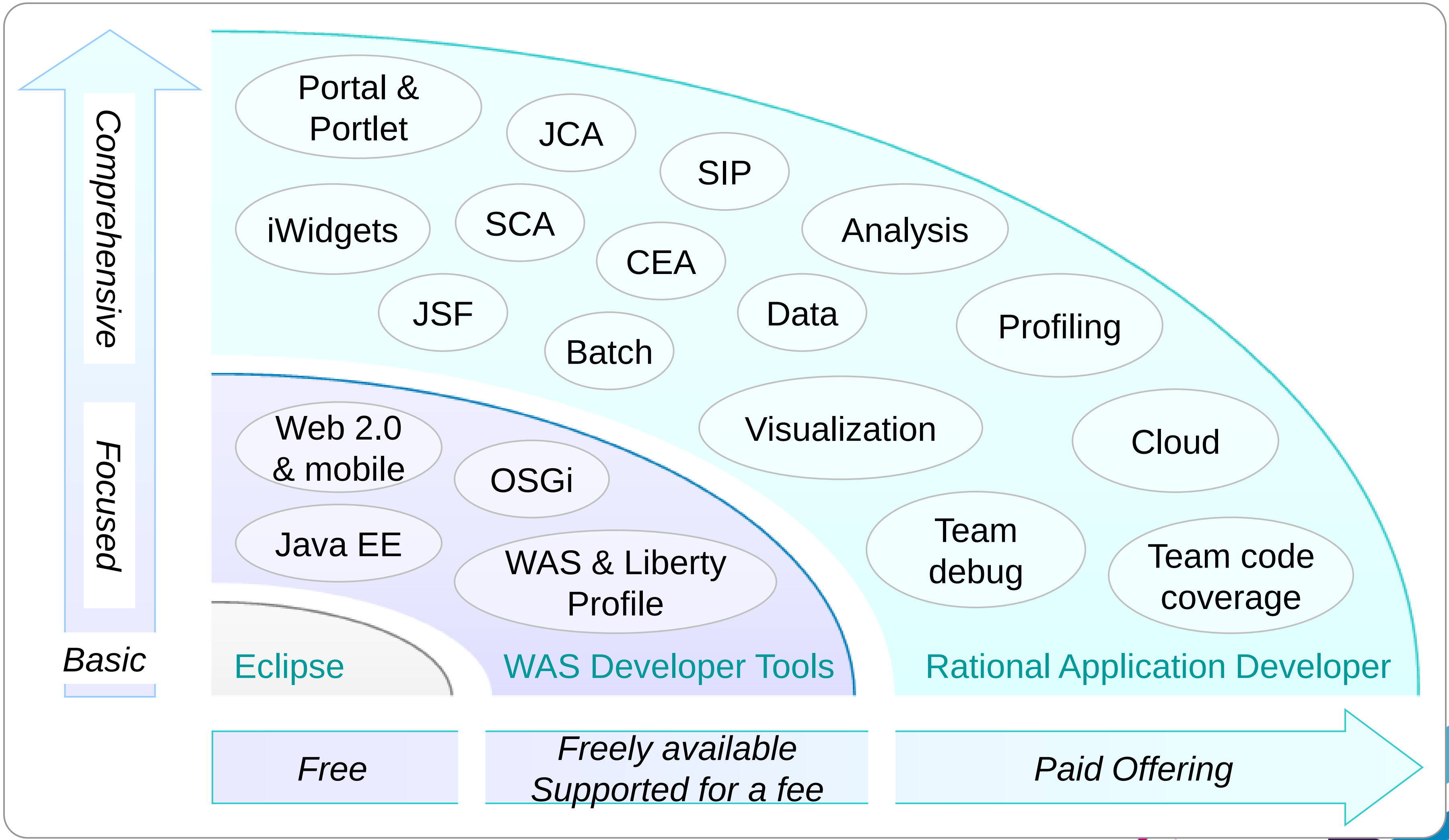
Design, Development and Deployment

Requirements | Architecture | Modeling | Development | Testing

Deploy and Release | Operations

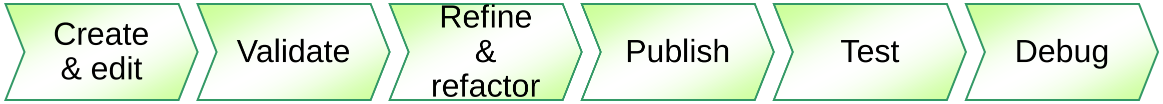
Open Platform for Data and Tools Integration and Automation

Rational Application Developer V8.5 and WAS Developer Tools v8.5



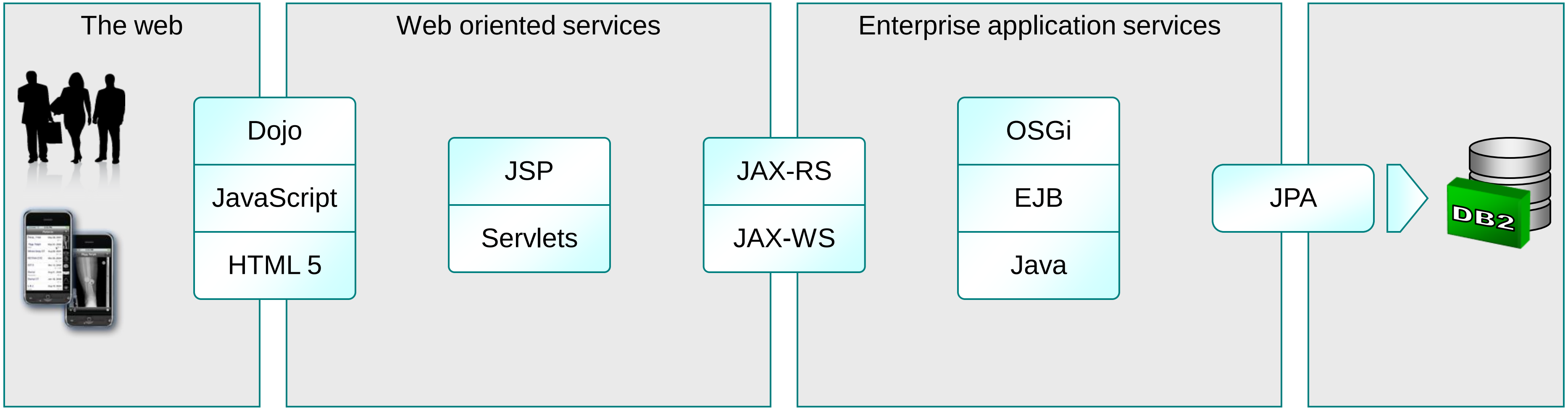
Unleash innovation with WDT 8.5

A cost effective IDE focussed on web application and modern service development



Create & edit → Validate → Refine & refactor → Publish → Test → Debug

Design web, Java EE and OSGi applications with persistence



The web

Dojo

JavaScript

HTML 5

Web oriented services

JSP

Servlets

JAX-RS

JAX-WS

Enterprise application services

OSGi

EJB

Java

JPA

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

WebSphere foundation

Enterprise Application Development with RAD

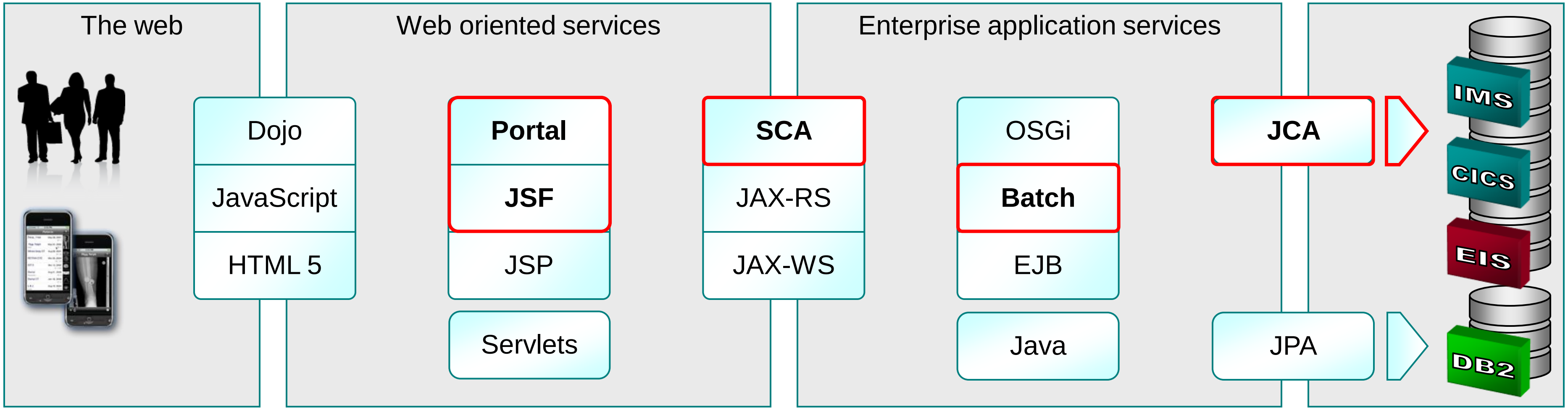
Evans Data Corporation EDC

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

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

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

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



Collaborate with agility, exploiting modern infrastructure and the Cloud

Collaborative Lifecycle Management

WebSphere foundation

Public and Private Clouds

Next NOW!

Introducing the WAS Tools Edition Bundles

WAS ND – Tools Edition

- Like “WAS – Tools Edition” but for WAS ND production use

WAS – Tools Edition

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

WAS for Developers – Tools Edition for Eclipse

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

Agenda

The business of developing software

Developing applications for people

- *The server side web*
- *Portals and portlets*
- *The client side web*
- *Mobile*

Developing services for applications

Rapid development and testing

Collaboration and quality

Cloud computing

Epilogue

Web and Mobile development made efficient

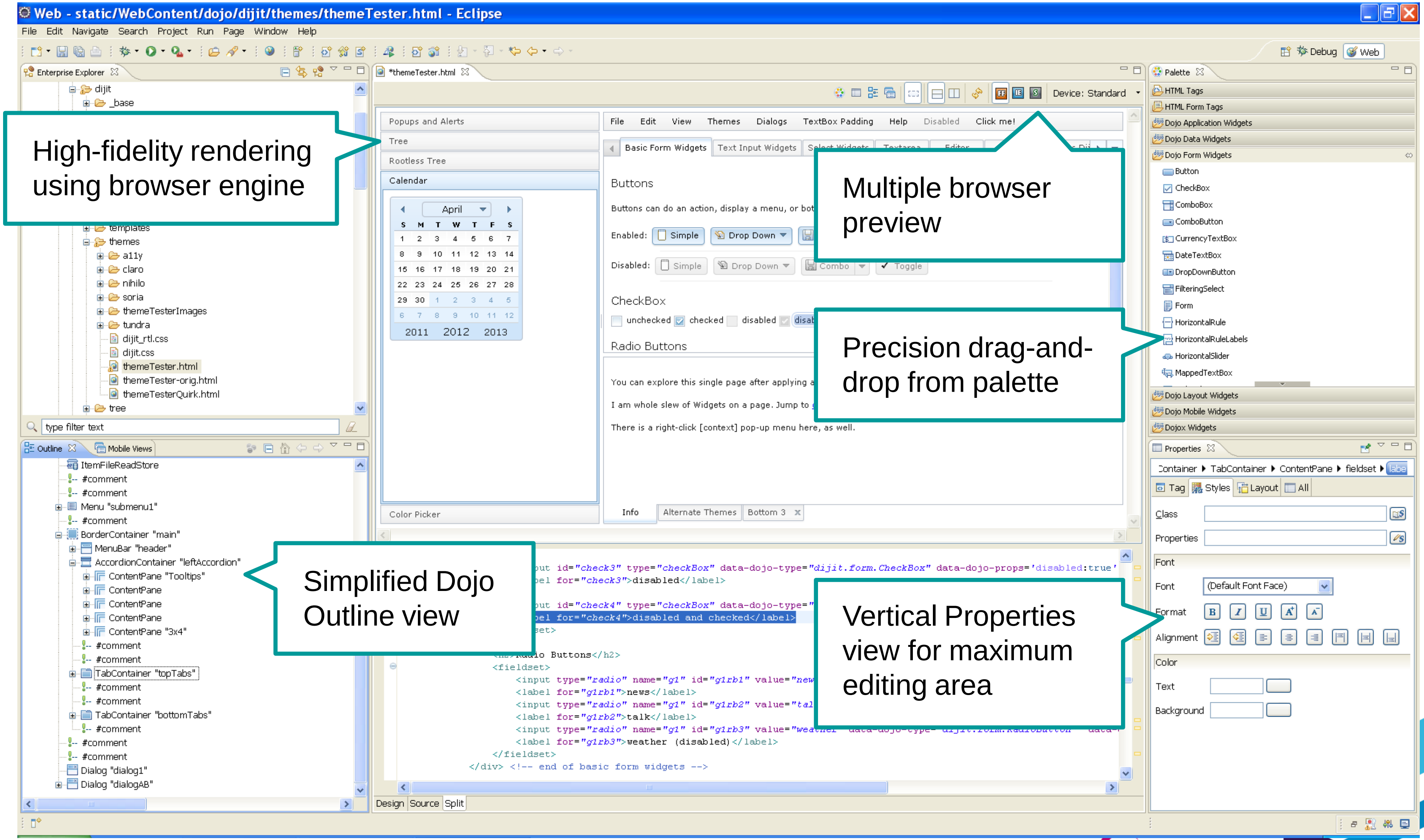
- Classic Web and Web 2.0 application development
  - Source editing
  - HTML5, CSS3, Dojo, JavaScript, JSON tools
  - New Rich Page Editor for client web development and JSF
  - Dojo tools, custom builds, unit testing
  - Web Preview Server
  - JavaScript debugging
  - JSF 2.0 tools
- Portal / portlet application development
  - With Web 2.0 technologies: Dojo, JAX-RS, RPC adaptors, Ajax Proxy, Active Site Analytics
  - Multi-channel development, targeting Desktop and Mobile
- Mobile web development (Worklight Studio)
  - Mobile Web applications with the Dojo mobile library
  - Rich Page Editor extended with target device form factors
  - Browser-based, simulated device application testing

Included in WDT

Included in RAD

New in V8.5

Introducing the new Rich Page Editor



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

```
put id="check3" type="checkBox" data-dojo-type="dijit.form.CheckBox" data-dojo-props='disabled:true'
bel for="check3">disabled</label>

put id="check4" type="checkBox" data-dojo-type="
bel for="check4">disabled and checked</label>
set>

<h2>Radio Buttons</h2>
<fieldset>
    <input type="radio" name="g1" id="g1rb1" value="new
    <label for="g1rb1">news</label>
    <input type="radio" name="g1" id="g1rb2" value="tal
    <label for="g1rb2">talk</label>
    <input type="radio" name="g1" id="g1rb3" value="weather" data-dojo-type="dijit.form.RadioButton" data-
    <label for="g1rb3">weather (disabled)</label>
</fieldset>
</div> <!-- end of basic form widgets -->
```

Next NOW!

Include
d in
WDT

Include
d
in RAD

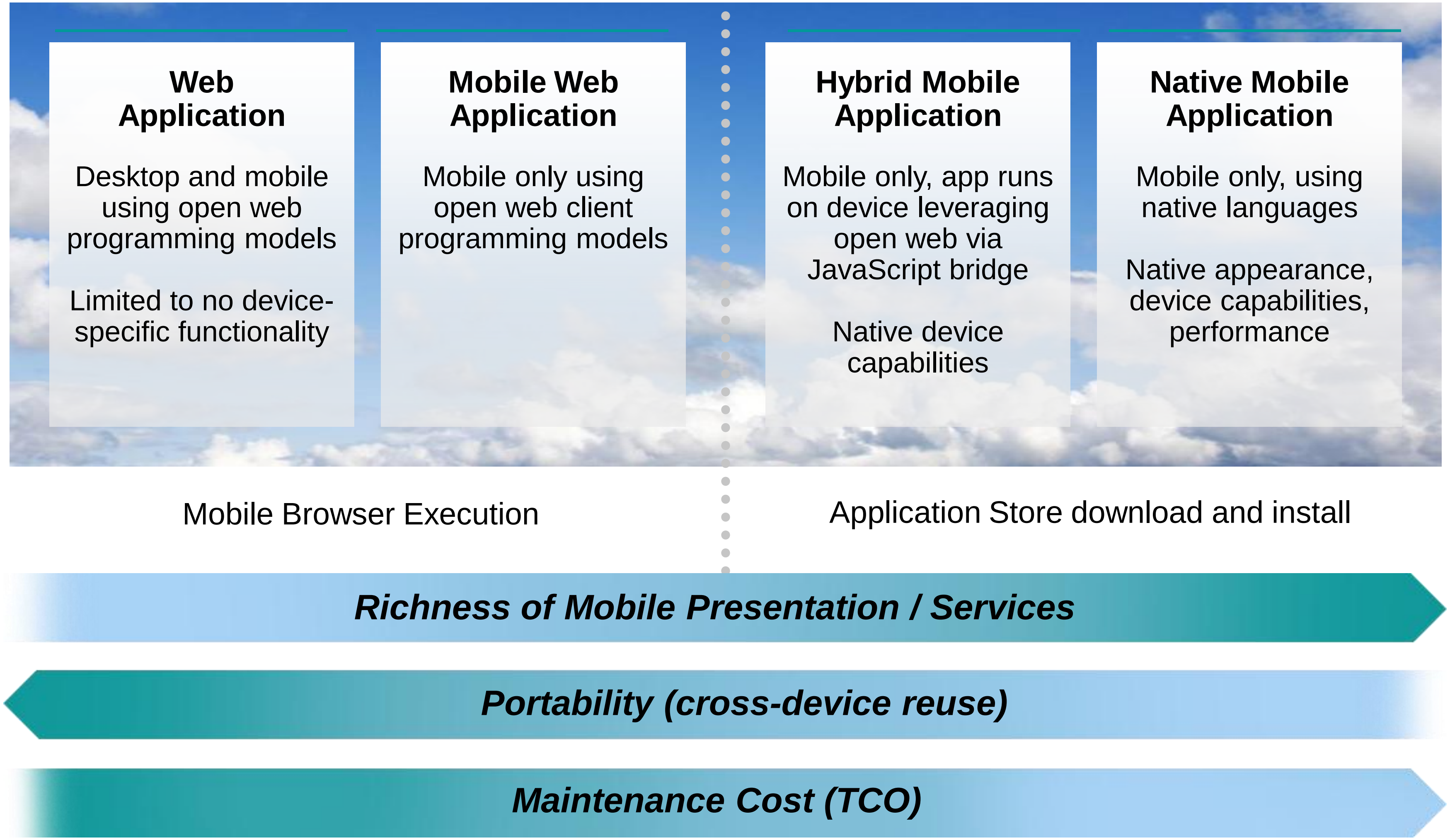
Improve
d in
V8.5

Building and deploying Dojo applications

- Where’s my Dojo?
  - In the project that needs it, or
  - In a shared project, or
  - Remote, on a CDN
- Dojo custom build tool
  - Based on the Dojo profile builder
  - Consolidates loose Dojo modules
  - Minifies JavaScript and CSS
- Dojo Objective Harness (DOH!)
  - “Junit for Dojo”
  - Part of Dojo Toolkit
  - Generate, run, and manage tests

- Web Preview Server
  - Fast and Lightweight!
    - Based on Liberty Profile
    - Starts up: < 5s
    - Deploys: < 5s
    - Stops: <1s
  - Static and dynamic web content
  - Built-in Ajax proxy
  - Built-in JAX-RS

The mobile application spectrum



Web Application

Desktop and mobile using open web programming models

Limited to no device-specific functionality

Mobile Web Application

Mobile only using open web client programming models

Hybrid Mobile Application

Mobile only, app runs on device leveraging open web via JavaScript bridge

Native device capabilities

Native Mobile Application

Mobile only, using native languages

Native appearance, device capabilities, performance

Mobile Browser Execution

Application Store download and install

Richness of Mobile Presentation / Services

Portability (cross-device reuse)

Maintenance Cost (TCO)

Included in WDT

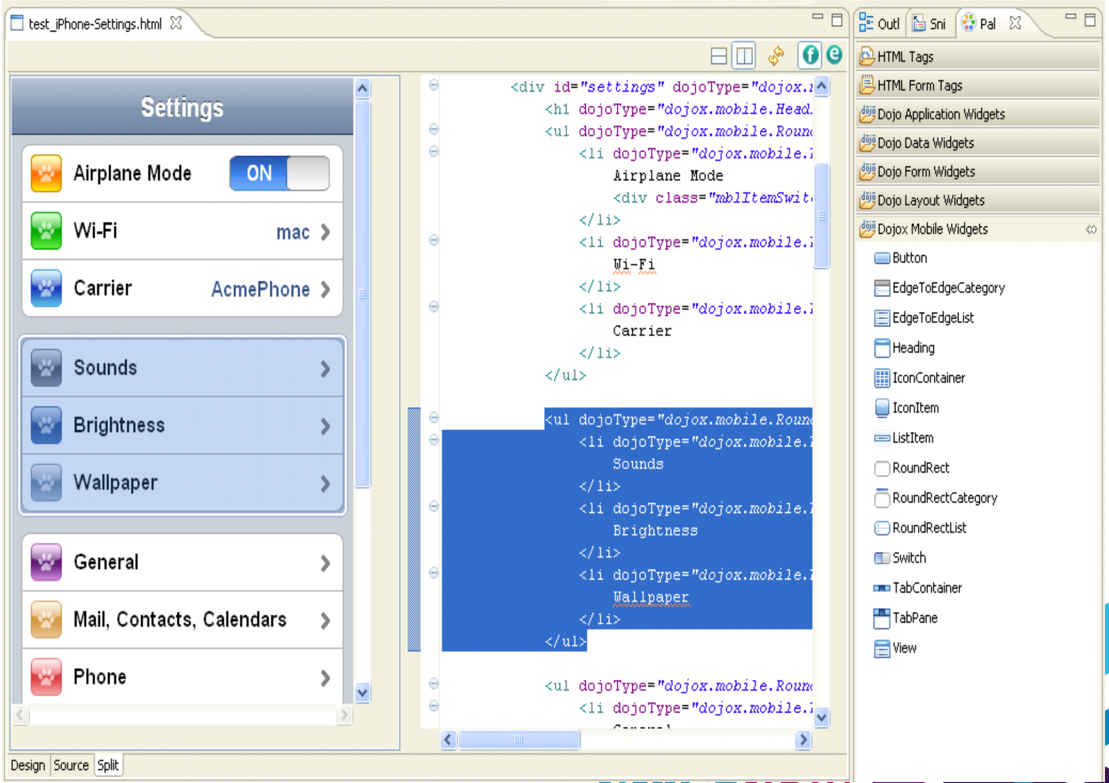
Included in RAD

Improved in V8.5

Taking your business on the road
Mobile web application development

- Competitive, compelling web access to your business from anywhere is becoming crucial. How do you develop and maintain an app once that runs well and looks good across desktops, tablets and smart phones?

- The RAD / WDT mobile web development tools
- A natural extension of the Web development tools
- Based on the Dojo mobile library (dojox.mobile)
- Rich Page Editor loaded with numerous target device form factors

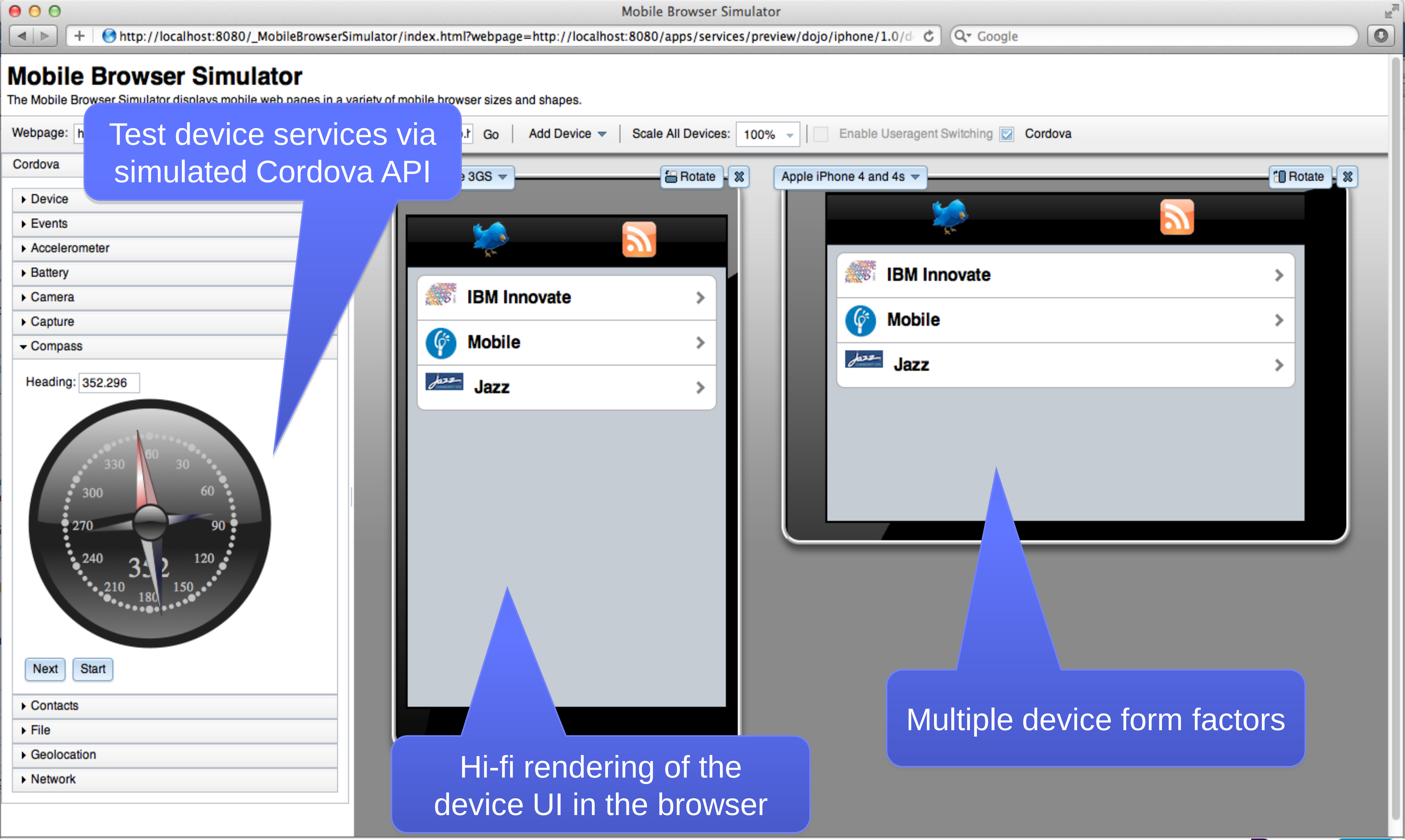


```
<div id="settings" dojoType="dojox.
    <h1 dojoType="dojox.mobile.Head
    <ul dojoType="dojox.mobile.Round
        <li dojoType="dojox.mobile.
            Airplane Mode
            <div class="mblItemSwit
        </li>
        <li dojoType="dojox.mobile.
            Wi-Fi
        </li>
        <li dojoType="dojox.mobile.
            Carrier
        </li>
    </ul>

    <ul dojoType="dojox.mobile.Round
        <li dojoType="dojox.mobile.
            Sounds
        </li>
        <li dojoType="dojox.mobile.
            Brightness
        </li>
        <li dojoType="dojox.mobile.
            Wallpaper
        </li>
    </ul>

    <ul dojoType="dojox.mobile.Round
        <li dojoType="dojox.mobile.
```

Previewing your app in the *Mobile Browser Simulator*



Mobile Browser Simulator

http://localhost:8080/_MobileBrowserSimulator/index.html?webpage=http://localhost:8080/apps/services/preview/dojo/iphone/1.0/d

Mobile Browser Simulator

The Mobile Browser Simulator displays mobile web pages in a variety of mobile browser sizes and shapes.

Webpage: h | Go | Add Device | Scale All Devices: 100% | ☐ Enable Useragent Switching | ☑ Cordova

Heading: 352.296

Test device services via simulated Cordova API

Hi-fi rendering of the device UI in the browser

Multiple device form factors

Agenda

The business of developing software

Developing applications for people

Developing services for applications

- *Service orienting your enterprise.*
- *Integration and reuse across lines of business*

Rapid development and testing

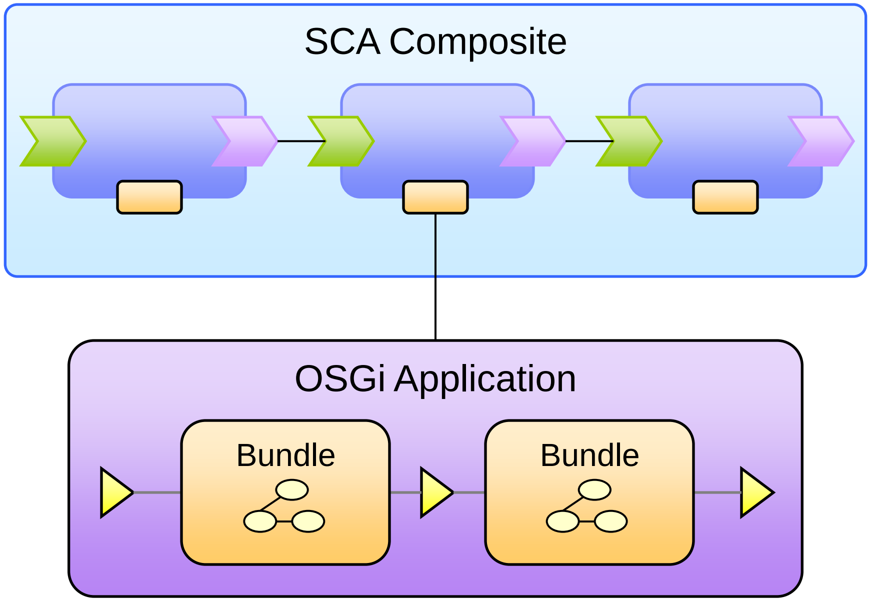
Collaboration and quality

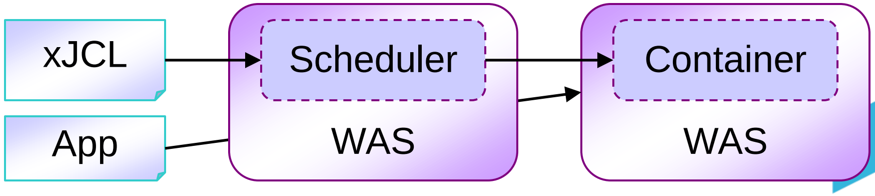
Cloud computing

Epilogue

The right framework for the right job

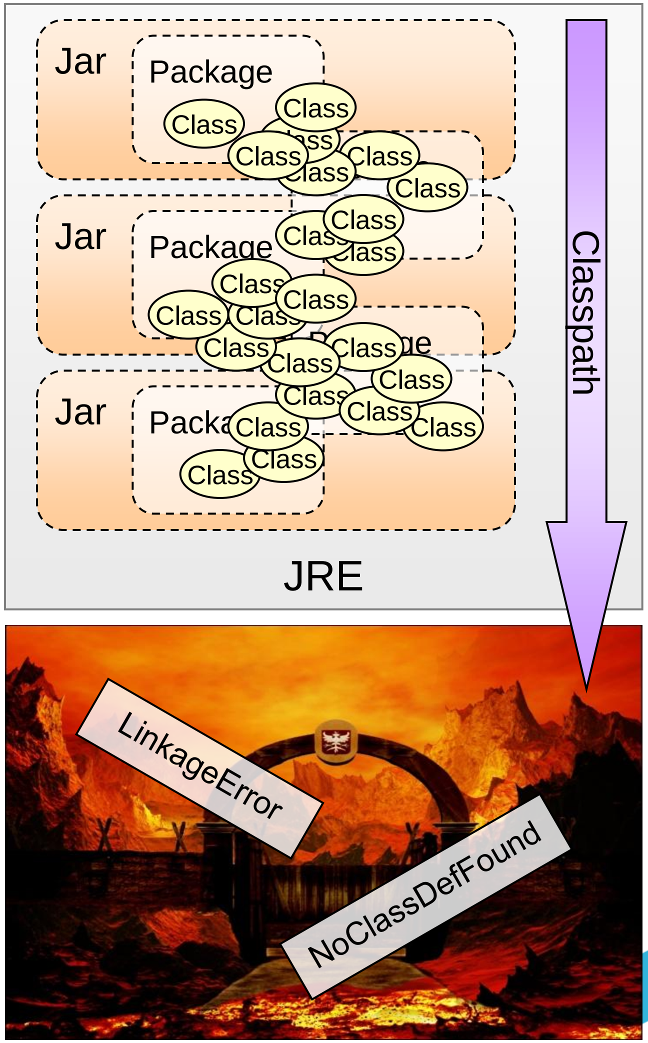
- SCA
  - Heterogeneous SOA programming model
  - Java, OSGi, Spring, EAR, Widget impls
  - WS, EJB, JMS, HTTP, Atom bindings
- OSGi
  - Bundles of Java, JPA, EJB, JSF, JAX-RS
  - Blueprint services for fine grained SOA
- Java EE
  - EJB – JPA – JAX-WS – JAX-RS – JCA
  - Refined deployment assembly management
- Modern Batch
  - For compute or data intensive work





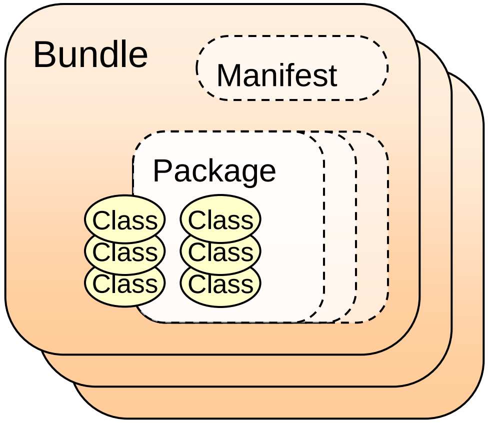
Java chaos and jar hell

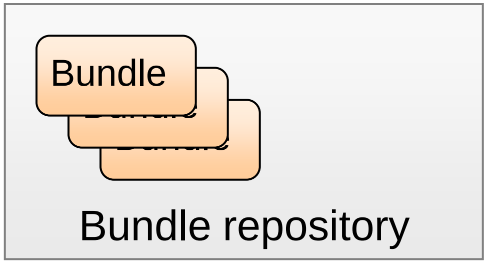
- In Java, modularity begins and ends with classes and class visibility
  - Packages manage naming, but no more
  - Jars deliver classes to the JRE, but they neither understand nor care what they contain, offer and depend upon
  - By run-time, all is reduced to a set of classes on a classpath. Java EE brings a little order to the classpath, but much of the Java chaos remains
- What’s missing?
  - Modularity. Isolation. Dependencies. API. Identity. Versioning. Lifecycle.



OSGi – Bringer of order & service orientation

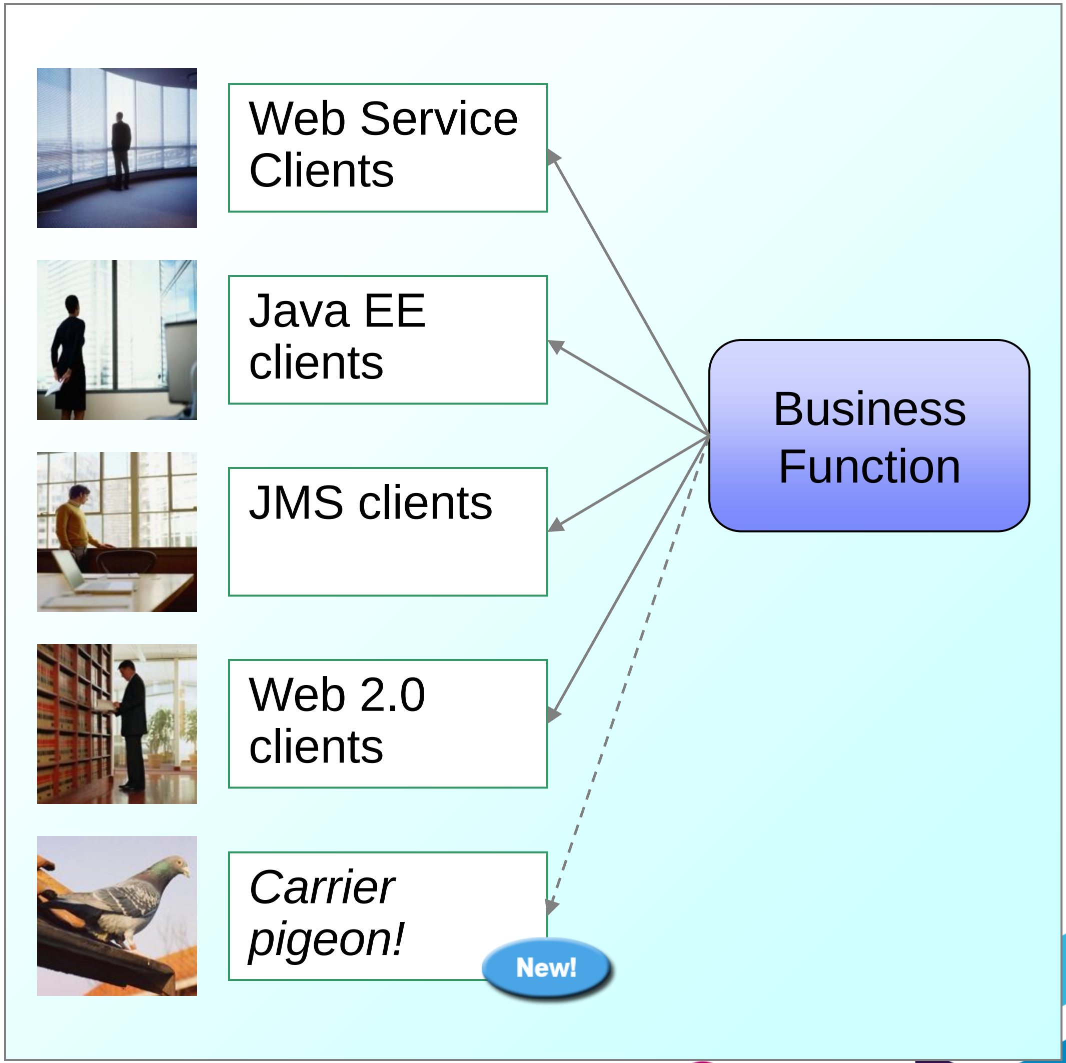
- Fosters organization of code into identified, versioned, reusable, centrally managed bundles with declared API and dependencies
- Establishes a fine-grained SOA programming model for Java
- Activates, injects dependencies, manages and deactivates POJOs
- Handles updates to modules without restarting the application
- Mixes Java standards (transactions, security, persistence) into the application as services
- Supports EJB bundles – *New in V8.5!*





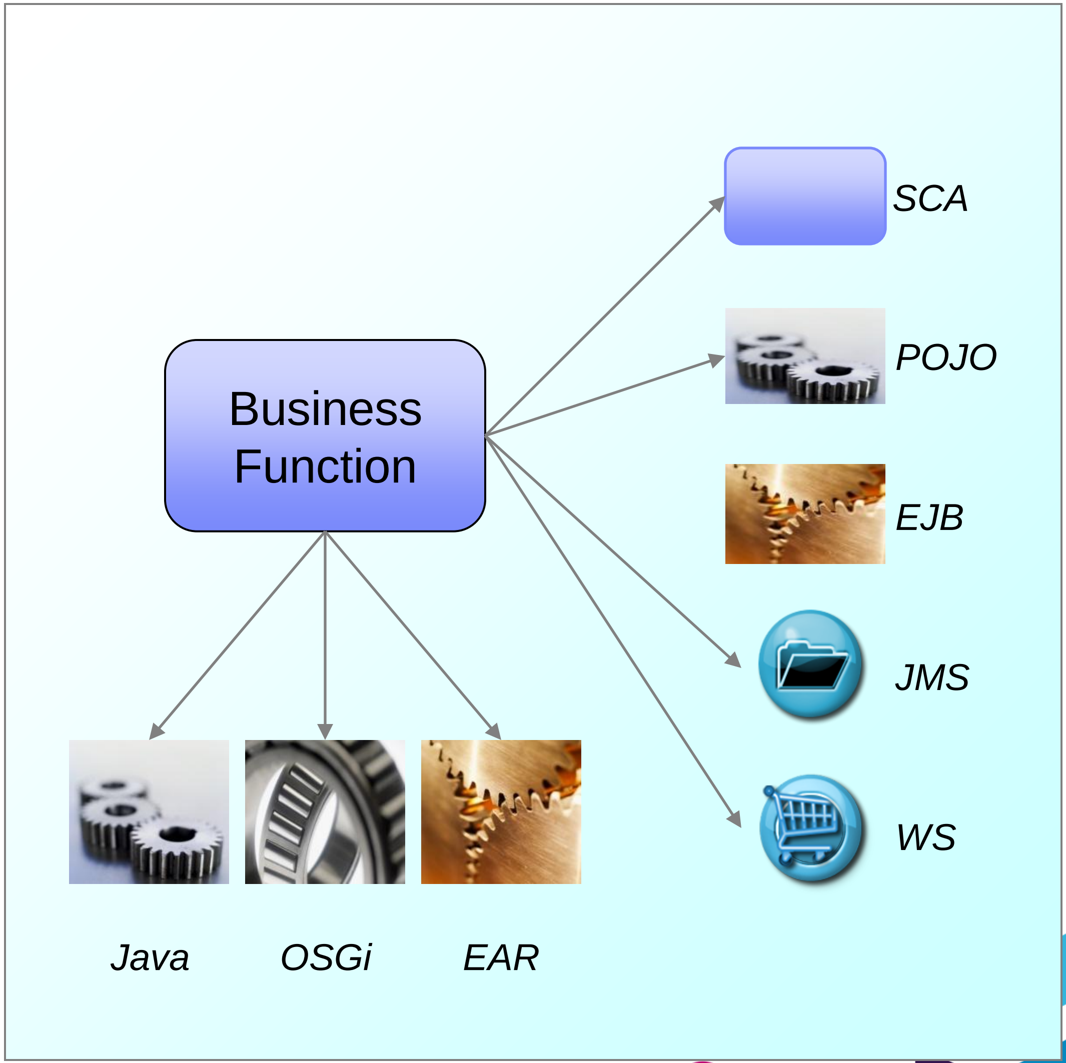
Challenges of service orientation

- How do I enable many lines of business with diverse skills to use my service, without mucking about with “glue code”?
- How do I future proof my services as new clients & technologies emerge?
- Consumers like providers who offer services on the consumers’ terms
- Consumers far outnumber producers



Challenges of service orientation

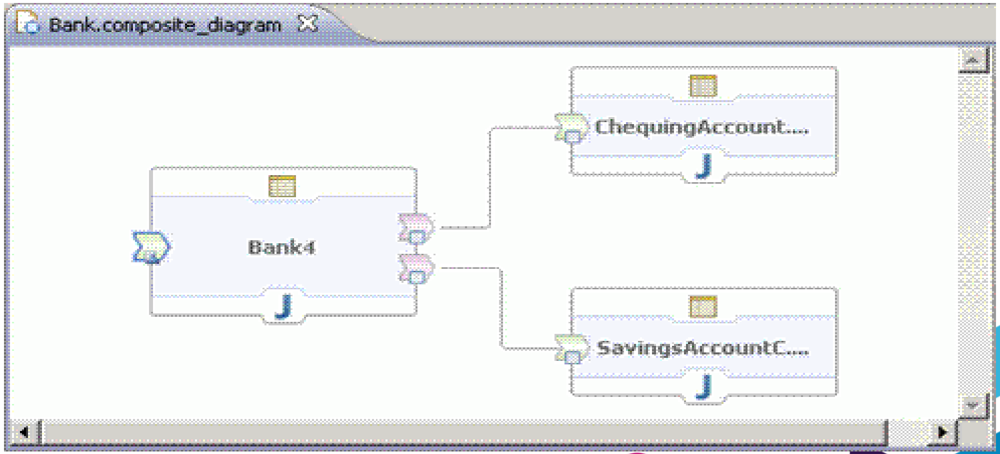
- How do I tap into the valuable IT collateral across my enterprise and begin to build common services?
- Reuse diverse assets by communicating with them on their terms
- Reuse diverse assets by using them as the machinery of service components



Breach the silos with SCA

- Express new or existing Java classes, enterprise apps, OSGi apps, Spring apps and more as reusable business services
- Compose simpler services into new, aggregate services
- Deliver and consume business services across multiple lines of business via SOAP, HTTP, ATOM, RMI, JMS
- Package, deploy, test and refine on WAS

- Welcome OASIS SCA 1.1, the open standard evolution of OSOA SCA 1.0.
- Inaugural tools and run-time in RAD v8.5 and WAS v8.5.

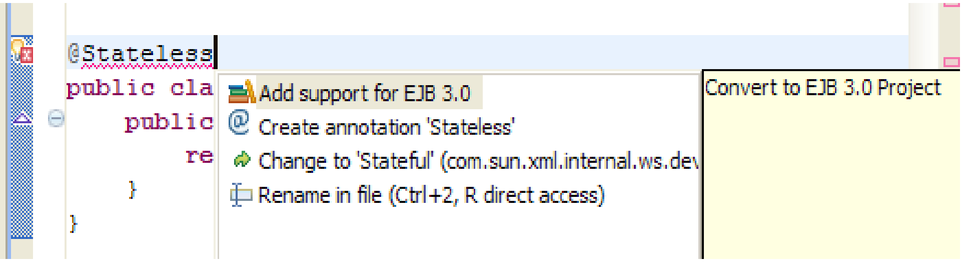


Features of Java EE 6 include…

- Enterprise JavaBeans (EJB) 3.1
  - Timer and async enhancements
  - Ability to test outside of WAS
- Java Persistence API (JPA) 2.0
  - Improved locking and mapping
  - Improved query construction
  - Dynamic type-safe query building
- Java Servlet 3.0
  - Annotations
  - Easier integration with third party presentation frameworks
- JavaServer Faces (JSF) 2.0
  - Annotations; Facelets; Ajax
- Java API for RESTful Web Services (JAX-RS) 1.1
  - The service side of web 2.0
- Java Architecture for XML Binding (JAXB) 2.2
  - Default marshalling optimizations
- Enterprise Web Services 1.3
  - Better integration and reuse
- Java API for XML-Based Web Services (JAX-WS) 2.2
  - Productivity and security enhancements

Java EE 6 development made intuitive

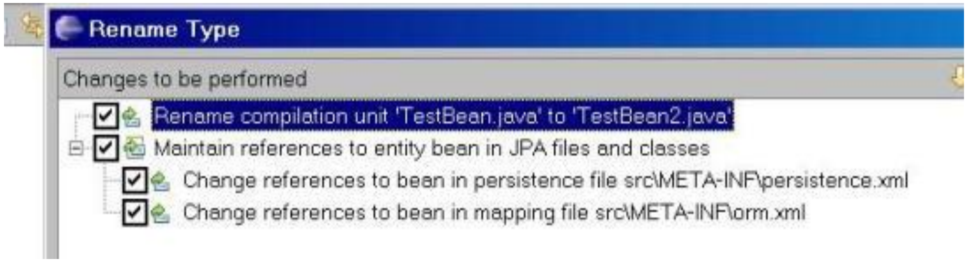
- Java EE navigation, content assist, validation and quick fixes keep you on track
- Keep cohesion throughout iterative development with the help of refactoring tools
- Use the annotations view to understand what all those annotations really do
- Organize enterprise modules with greater flexibility, leading to greater reuse and efficiency at run-time



```
@Stateless
public cla
    public
        re
    }
}
```

- Add support for EJB 3.0
- Create annotation 'Stateless'
- Change to 'Stateful' (com.sun.xml.internal.ws.dev
- Rename in file (Ctrl+2, R direct access)

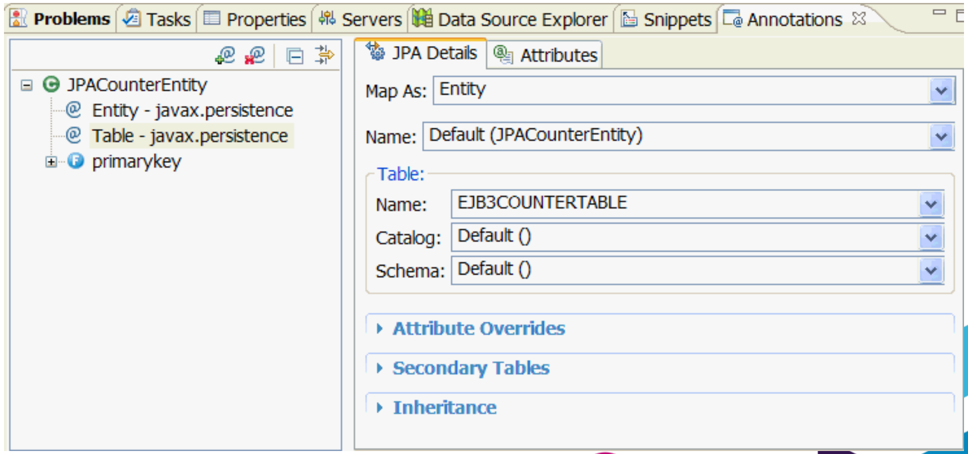
Convert to EJB 3.0 Project



Rename Type

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



- JPACounterEntity
  - Entity - javax.persistence
  - Table - javax.persistence
  - primarykey

JPA Details | Attributes

Map As: Entity

Name: Default (JPACounterEntity)

Table:

Name: EJB3COUNTERTABLE

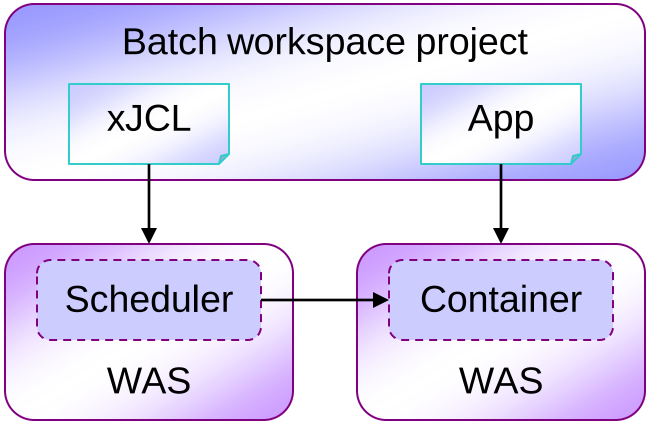
Catalog: Default ()

Schema: Default ()

WebSphere Batch

- Brings compute and data intensive batch processing to WAS, sharing the same business logic used by online systems.
- Framework and tools make writing, running and managing batch applications a snap.
- Java EE based packaging and deployment.

- Projects, wizards and editors streamline the development of xJCL and batch classes.
- Deploy and submit to WAS, with job progress integrated into the IDE’s console.
- Support for WAS Compute Grid Runtime



Agenda

The business of developing software

Developing applications for people

Developing services for applications

Rapid development and testing

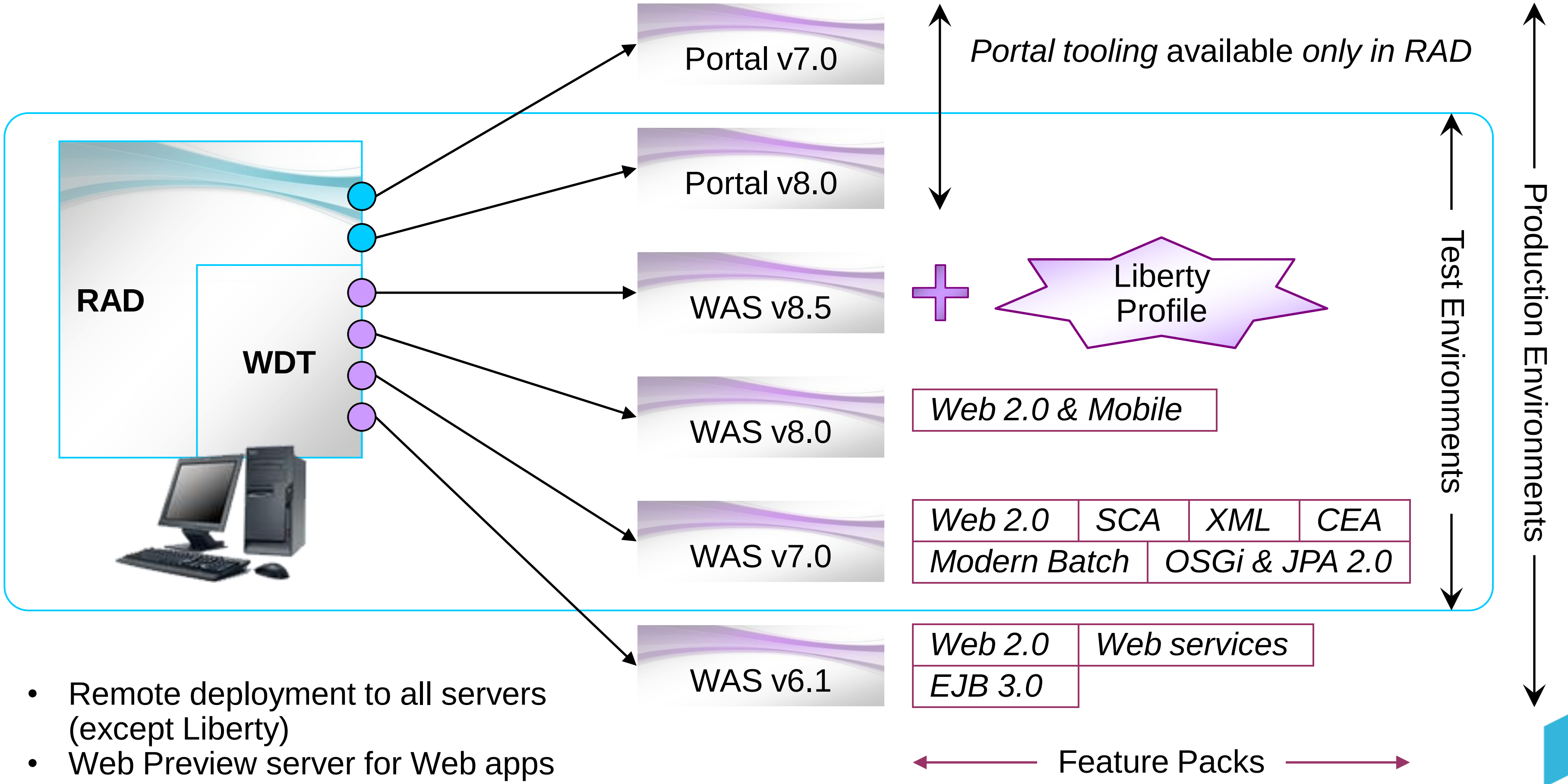
- *Continuous development and test*
- *Application and Portal servers*
- *Production and developer friendly servers*

Collaboration and quality

Cloud computing

Epilogue

WebSphere Application Server and WebSphere Portal Server



Portal tooling available *only in RAD*

Web 2.0 & Mobile

Web 2.0 | *SCA* | *XML* | *CEA*
Modern Batch | *OSGi & JPA 2.0*

Web 2.0 | *Web services*
EJB 3.0

- Remote deployment to all servers (except Liberty)
- Web Preview server for Web apps
- Universal Test Client

Next NOW!

Included in WDT

Included in RAD

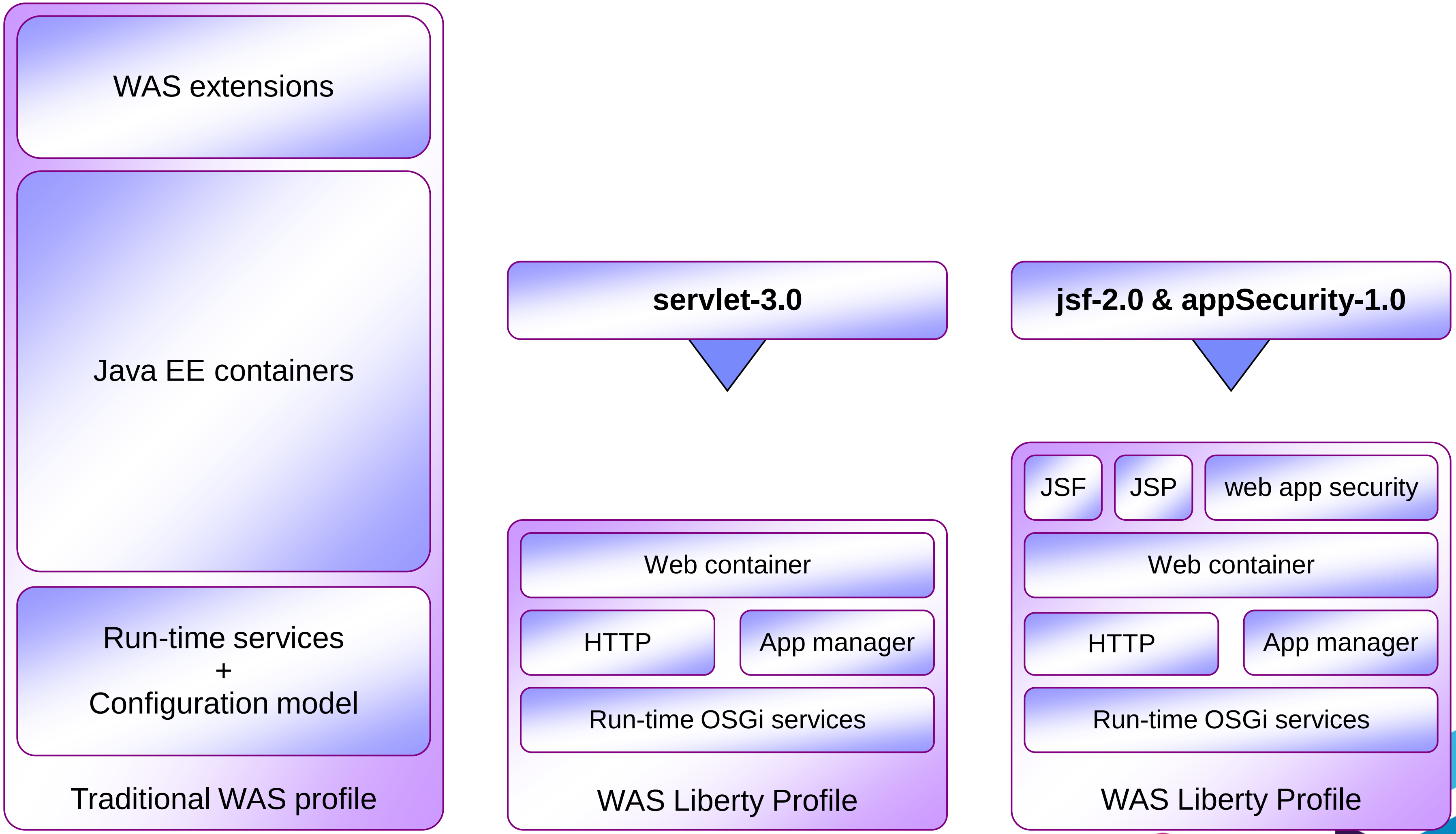
Improved in V8.5

By developers, for developers
Liberty at last

- A new dynamic profile of WAS aimed at the developer experience
  - Development and test server initially for web, mobile and OSGi
  - Footprint < 50 Mb
  - Fast startup < 5 seconds
  - Shareable, simplified “by exception” configuration
  - Fidelity with full profile WAS – same containers and QoS
  - Freely available
- Shorten development time to value
  - Develop and test in RAD with the WAS Liberty Profile
  - Deploy applications as-is to Liberty profile or full profile of WAS.

http://wasdev.net

Liberty Profile – A composable run-time that starts the features your application needs, and no more



Agenda

The business of developing software

Developing applications for people

Developing services for applications

Rapid development and testing

Collaboration and quality

- *Development is a team sport*
- *Achieving agility at scale*
- *Continuous integration*
- *Analysis and profiling*

Cloud computing

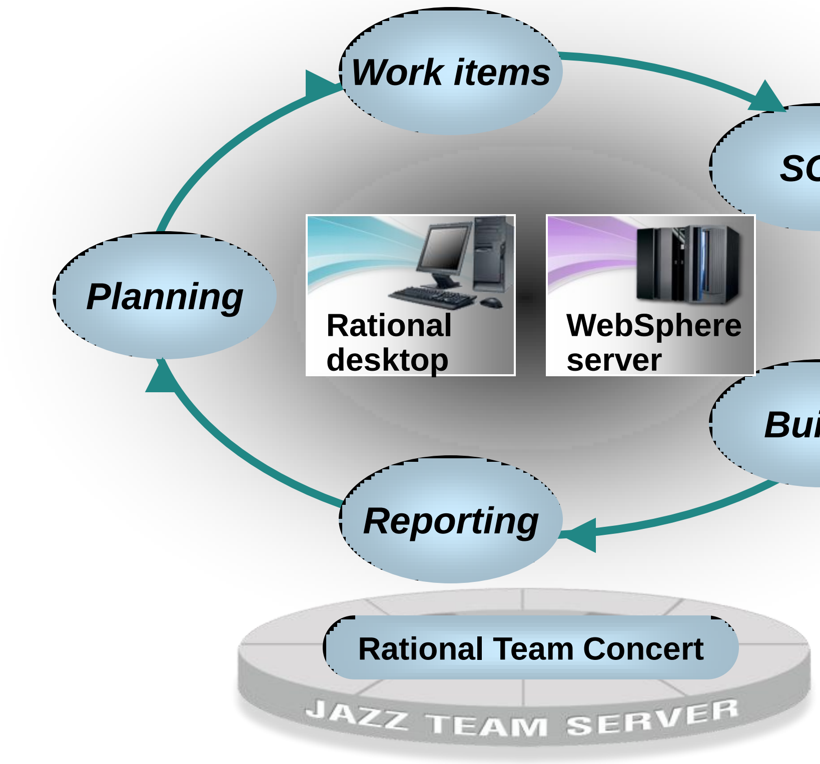
Epilogue

Agile Application Development for WebSphere

Rational Application Developer + Rational Team Concert

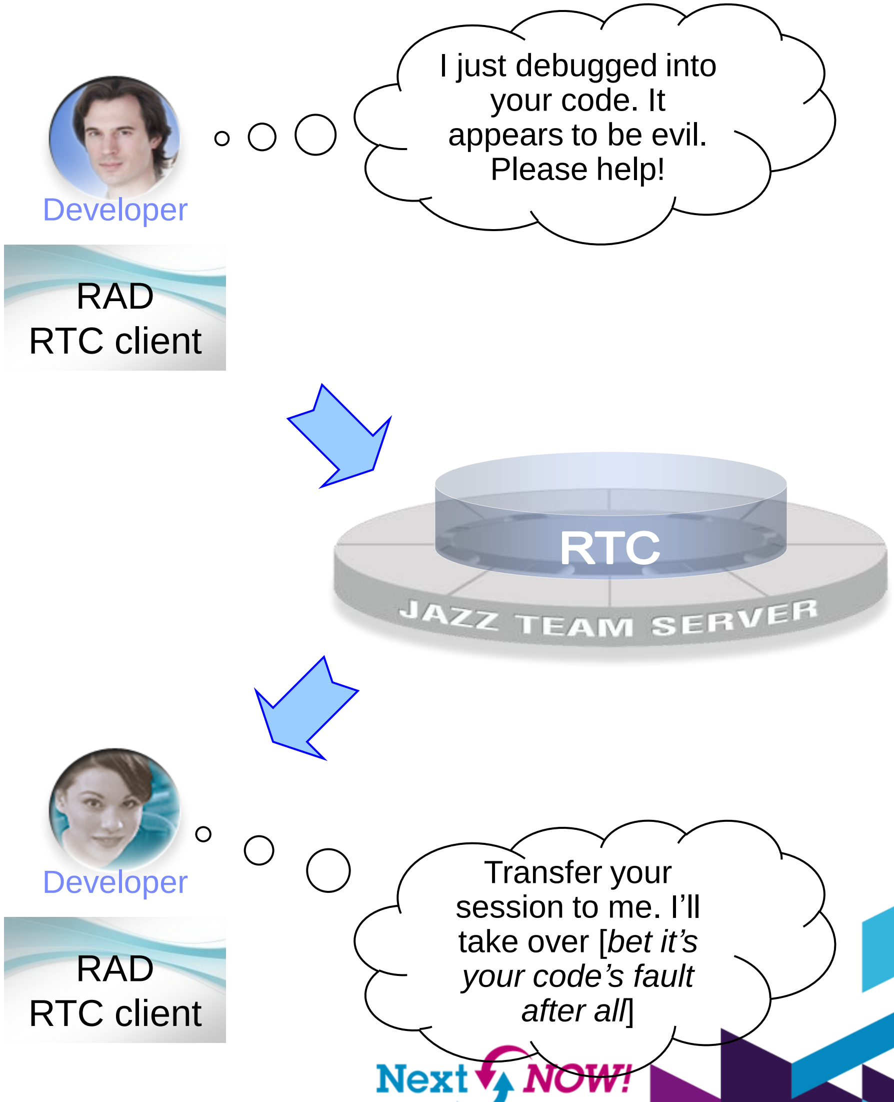
Team productivity thru integration of WebSphere development into the application lifecycle.

- Maximize product value with **In-Context Collaboration**
- Accelerate time to delivery with **Real-Time Planning**
- Improve quality with **Lifecycle Traceability**
- Achieve predictability with **Development Intelligence**
- Reduce costs with **Continuous Improvement**



Quality through teamwork – Debugging
Pest control is a global, 24/7 issue

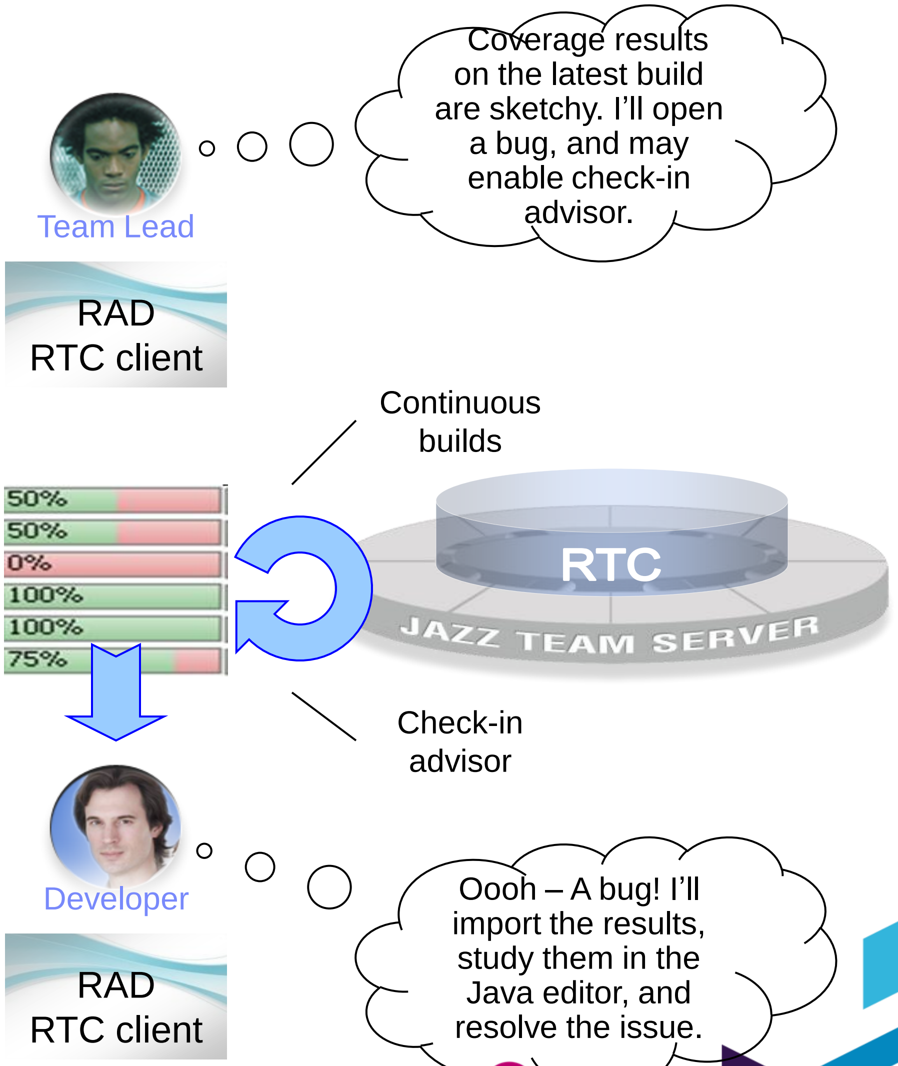
- Diagnosis may lead you into somebody else’s code
  - You could spend hours or days learning their code
  - They could spend hours or days reproducing the bug (or not)
- Don’t bring the bug to them, bring them to the bug
  - Transfer debug sessions in real time, along with the source
  - Park debug sessions for later


Developer
I just debugged into your code. It appears to be evil. Please help!
RAD RTC client
RTC
JAZZ TEAM SERVER
Developer
Transfer your session to me. I’ll take over [*bet it’s your code’s fault after all*]
RAD RTC client

Quality through teamwork – Code coverage

Infuse quality from day one

- Run on demand, UI or headless
- Supports WAS V6.1 to V8.5
- Supports Java 5, 6 and 7
- Integrate into RTC builds
- Import results to the workspace
- Find bad code or test suite gaps
- Dynamically reset coverage statistics at server launch
- Open work items directly from undesirable coverage results
- Prevent checking in of code with missing or poor results

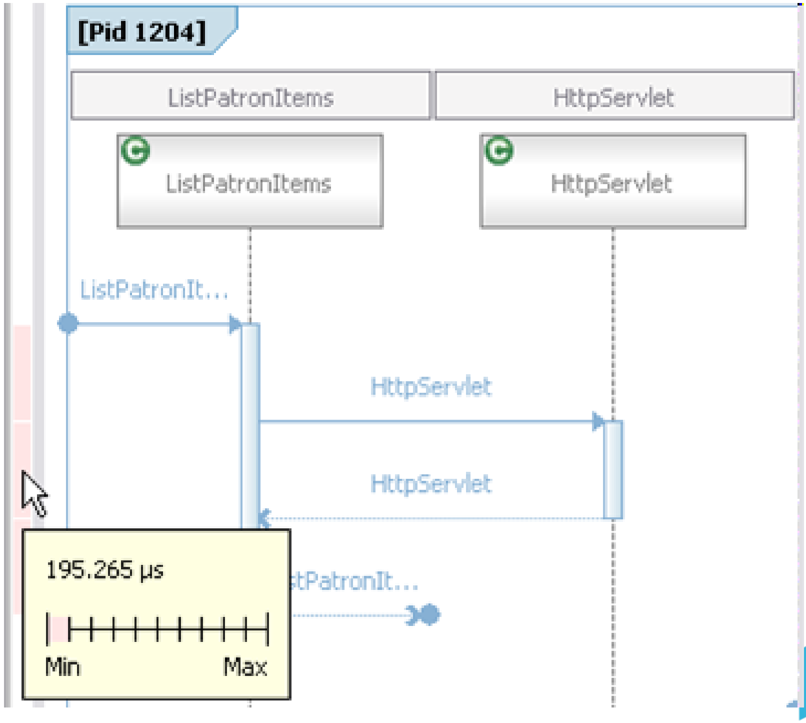


Coverage results on the latest build are sketchy. I'll open a bug, and may enable check-in advisor.

Oooh – A bug! I'll import the results, study them in the Java editor, and resolve the issue.

Quality through static and dynamic analysis

- The longer a bug lives the bigger and hairier it gets, and the more expensive it is to exterminate
- Static analysis
  - – Find anti-patterns and lame code
  - – Over 200 rules with quick fixes
- Dynamic profiling
  - – Find memory / object leaks
  - – Find performance root causes
  - – Find undesirable threads
  - – Supports WAS V6.1 to V8.5
  - – Supports Java 5, 6 and 7
- Probekit
  - – For precise, detailed profiling



Agenda

The business of developing software

Developing applications for people

Developing services for applications

Rapid development and testing

Collaboration and quality

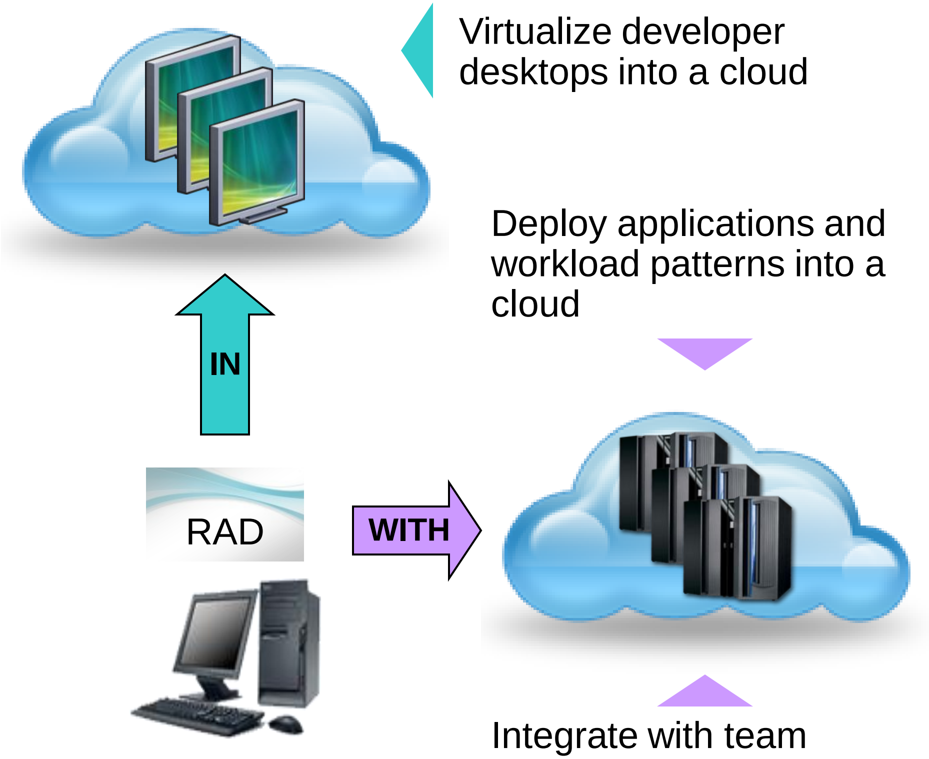
Cloud computing

- *Virtual development and test environments in the cloud*
- *Developing and testing applications for public and private clouds*

Epilogue

Two basic IDE relationships with the cloud *"in" and "with"*

- An IDE can run on the cloud
  - Virtual Desktop Infrastructure
- An IDE can integrate with the cloud...
  - Accessing cloud-hosted team collaboration services
  - Deploying applications under development to the cloud
- Some clouds are better suited to hosting desktop workloads, while others are better suited to team and middleware services



Virtualize developer desktops into a cloud

Deploy applications and workload patterns into a cloud

Integrate with team lifecycle services running in a cloud

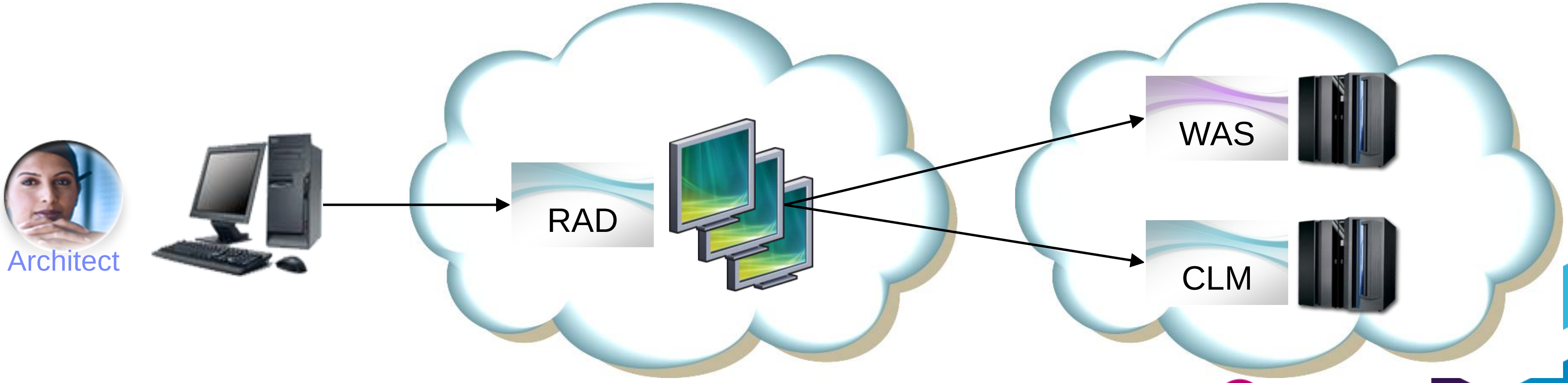
Virtualizing the RAD desktop

- Offload developer desktops from stressed PCs into the cloud
- Define and centrally manage desktops shaped to your standards
- Reduce capital and operational expense while improving mobility
- Rapidly onboard new people and teams, 24/7, across the globe

Traditional desktop – Light weight PCs with remote access software

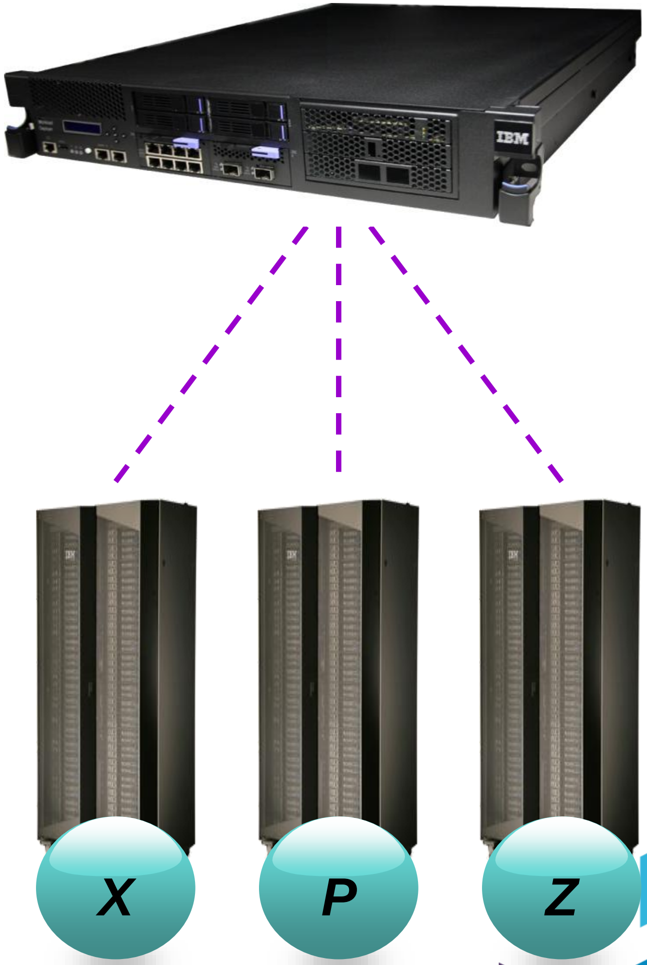
IBM Smart Business Desktop on the Cloud or IBM Virtual Desktop for Smart Business

IBM SmartCloud Enterprise, IBM Workload Deployer



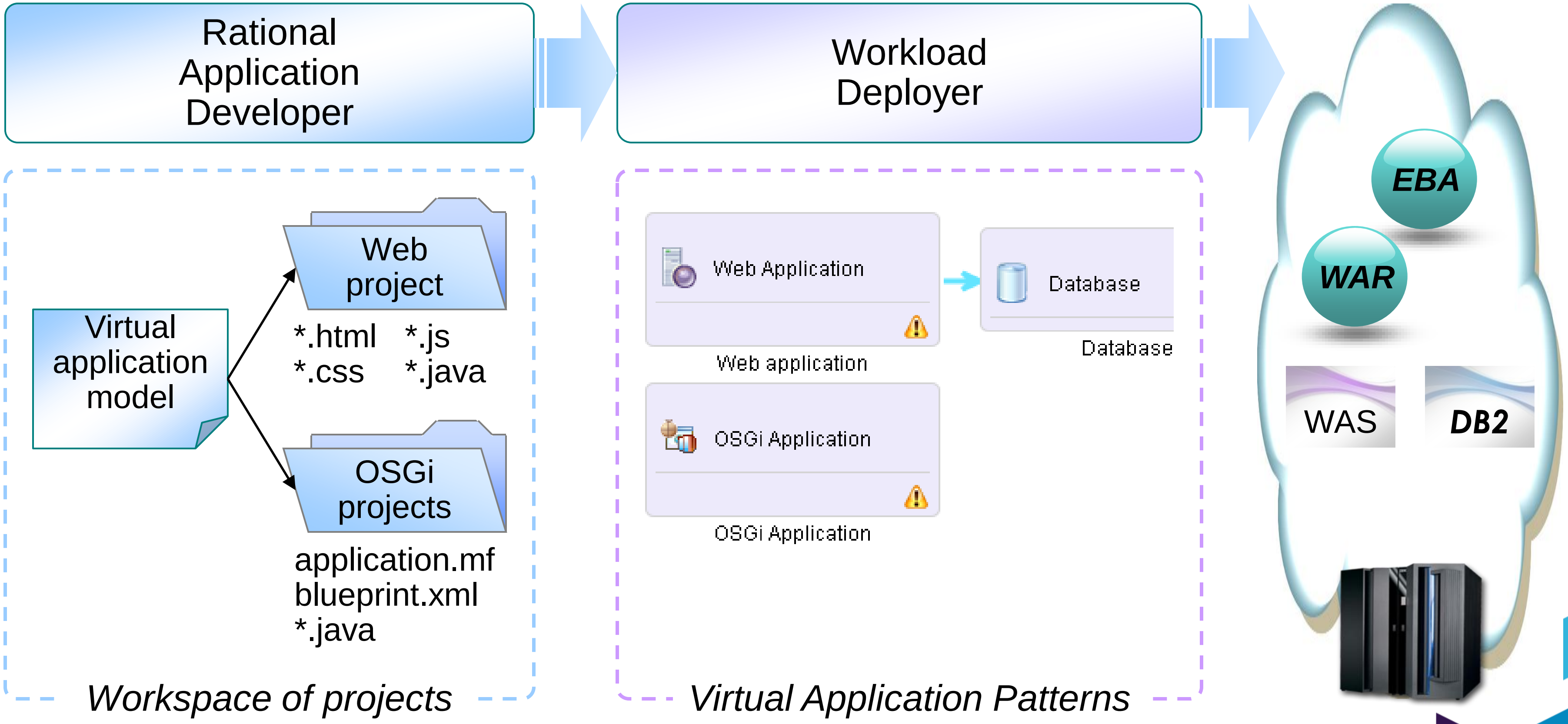
IBM Workload Deployer
Build your own private platform as a service cloud

- Secure, self-service cloud management hardware appliance
- Design and deploy consistent and repeatable middleware patterns into a cloud of virtualized hardware running a supported hypervisor
  - VMware ESX – z/VM – PowerVM
- Bring your own cloud to leverage your existing underutilized hardware
- Full lifecycle management for IBM middleware, limited lifecycle management for third part products



Integrating with IBM Workload Deployer

- Develop virtual application patterns in Workload Deployer, linked with normal projects & source code in the RAD workspace



Rational Application Developer

Workload Deployer

Virtual application model

Web project

*.html *.js
*.css *.java

OSGi projects

application.mf
blueprint.xml
*.java

Workspace of projects

Web Application

Web application

Database

Database

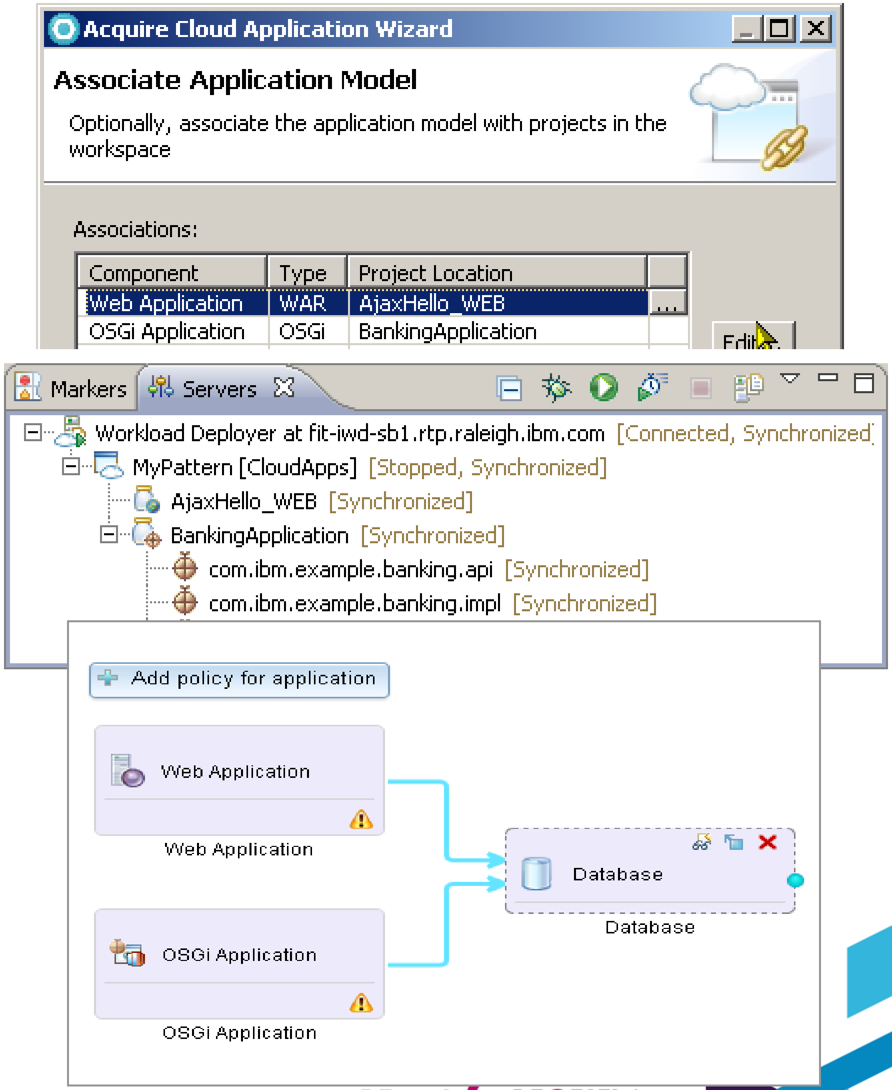
OSGi Application

OSGi Application

Virtual Application Patterns

Tools for IBM Workload Deployer

- Create and manage Workload Deployer server definitions
- Create basic virtual application patterns from existing projects
- Acquire virtual application patterns from Workload Deployer and associate projects to them
- Publish; Run; Update; Stop; Remove; Import; Export
- Publish changes to Workload Deployer
- Detect and reacquire changes from Workload Deployer.



Acquire Cloud Application Wizard

Associate Application Model

Optionally, associate the application model with projects in the workspace

Associations:

Component	Type	Project Location
Web Application	WAR	AjaxHello_WEB
OSGi Application	OSGi	BankingApplication

Workload Deployer at fit-iwd-sb1.rtp.raleigh.ibm.com [Connected, Synchronized]
MyPattern [CloudApps] [Stopped, Synchronized]
AjaxHello_WEB [Synchronized]
BankingApplication [Synchronized]
com.ibm.example.banking.api [Synchronized]
com.ibm.example.banking.impl [Synchronized]

Agenda

The business of developing software

Developing applications for people

Developing services for applications

Rapid development and testing

Collaboration and quality

Cloud computing

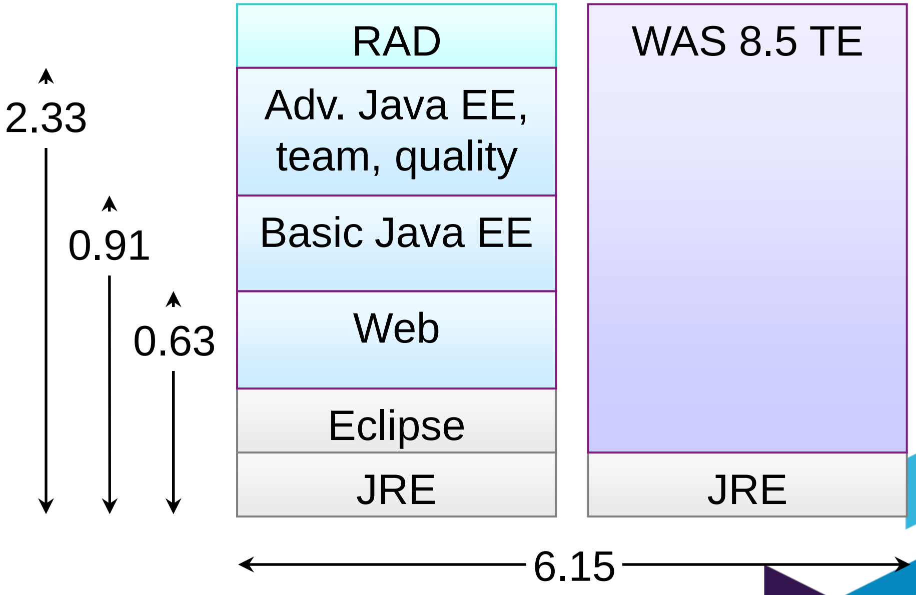
Epilogue

- *Installation, migration & help*
- *Handy resources*
- *Q&A*

Installation and licensing

- Eclipse
  - Included with installation, or
  - Install into existing Eclipse 3.6.2
- Installation
  - Installation Manager
    - Interactive, or automated using response files
    - Enterprise install capabilities for simplifies maintenance of standardized images
  - Eclipse Updater (WDT only)

- Licensing
  - RAD: perpetual, term, floating, token
  - WDT: perpetual, free/unsupported
- Memory: 3 GB recommended
- Storage (in GB)



Handy resources

- RAD for WebSphere

  Product info, viewlets, tutorials, Redbook

  http://ibm.com/software/awdtools/developer/application

- WASdev Community

  Downloads, blog, forum, Q&A

  http://wasdev.net

www.ibm.com/software/rational

Acknowledgements and disclaimers

Availability: References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While 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. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

If you have mentioned trademarks that are not from IBM, please update and add the following lines:

[Insert any special third-party trademark names/attributions here]

www.ibm.com/software/rational

Backup slides

Software development is hard, the rapid pace of change makes it difficult to keep up…

Developer and team productivity

- Adjusting to rapidly changing requirements and shorter project cycles
- Complexity in adopting new technology
- Developers work as individuals - skills are not leveraged across the team

Application quality

- Hard to diagnose problems that show up late
- Shorter project cycles puts quality at risk
- Need to infuse quality from the beginning

Product integration

- Need an end to end environment that is not brittle to changes of individual components

Cloud computing

- Migrating our application to the cloud is not so simple
- IDEs are expensive to manage, and too much downtime for developers

Standards and platforms

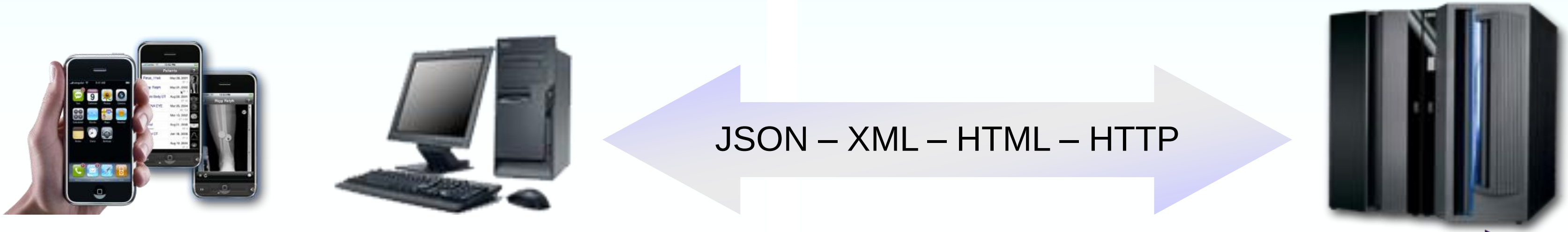
- Assurances that our investment is here to stay and has a healthy future.

Web application development

- Modern culture expects a web that is fast, slick and runs anywhere
- Good news: Many technologies exist for building rich web apps
- Bad news: Many technologies exist for building rich web apps

- The client side web
  - Dojo mobile
  - Dojo
  - JavaScript
  - HTML & CSS

- The server side web
  - Portal and Portlet
  - JSF, JSP and Servlets
  - JAX-RS for REST services



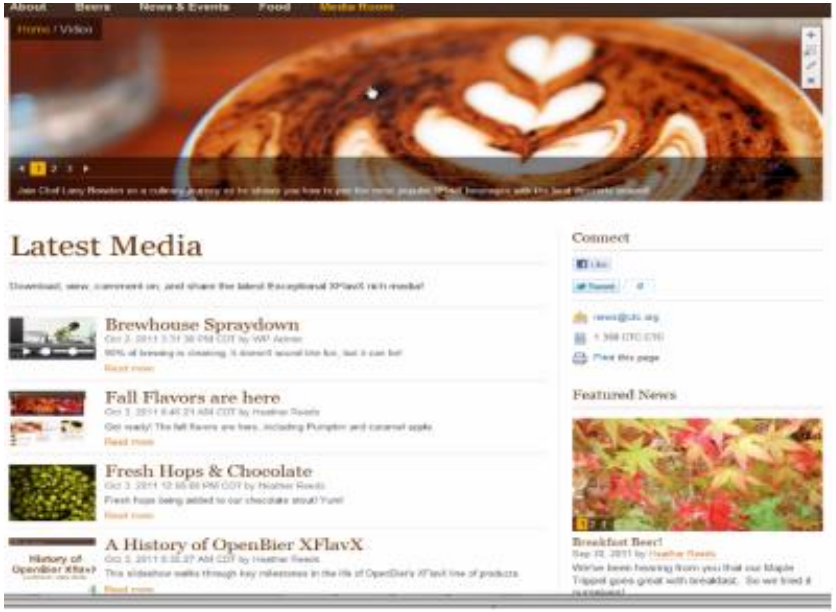
JSON – XML – HTML – HTTP

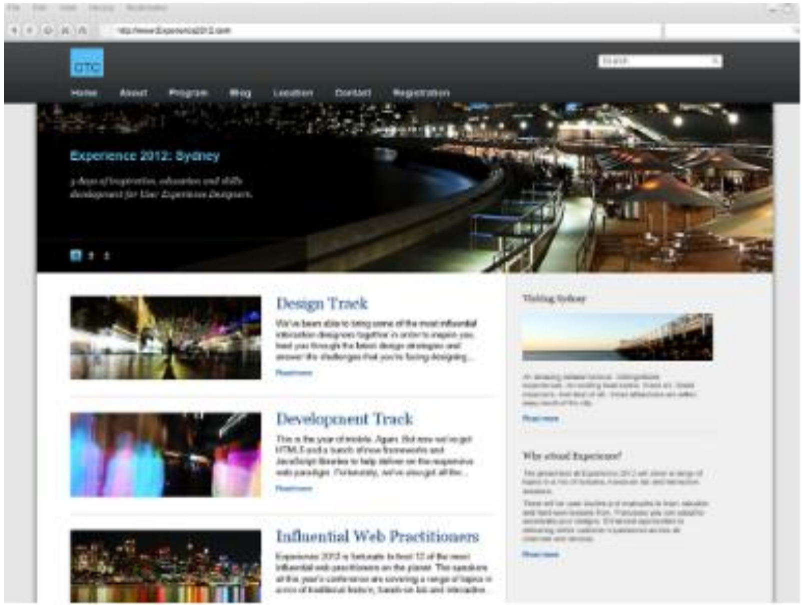
Only in RAD

New in V8.5

Portals and portlets

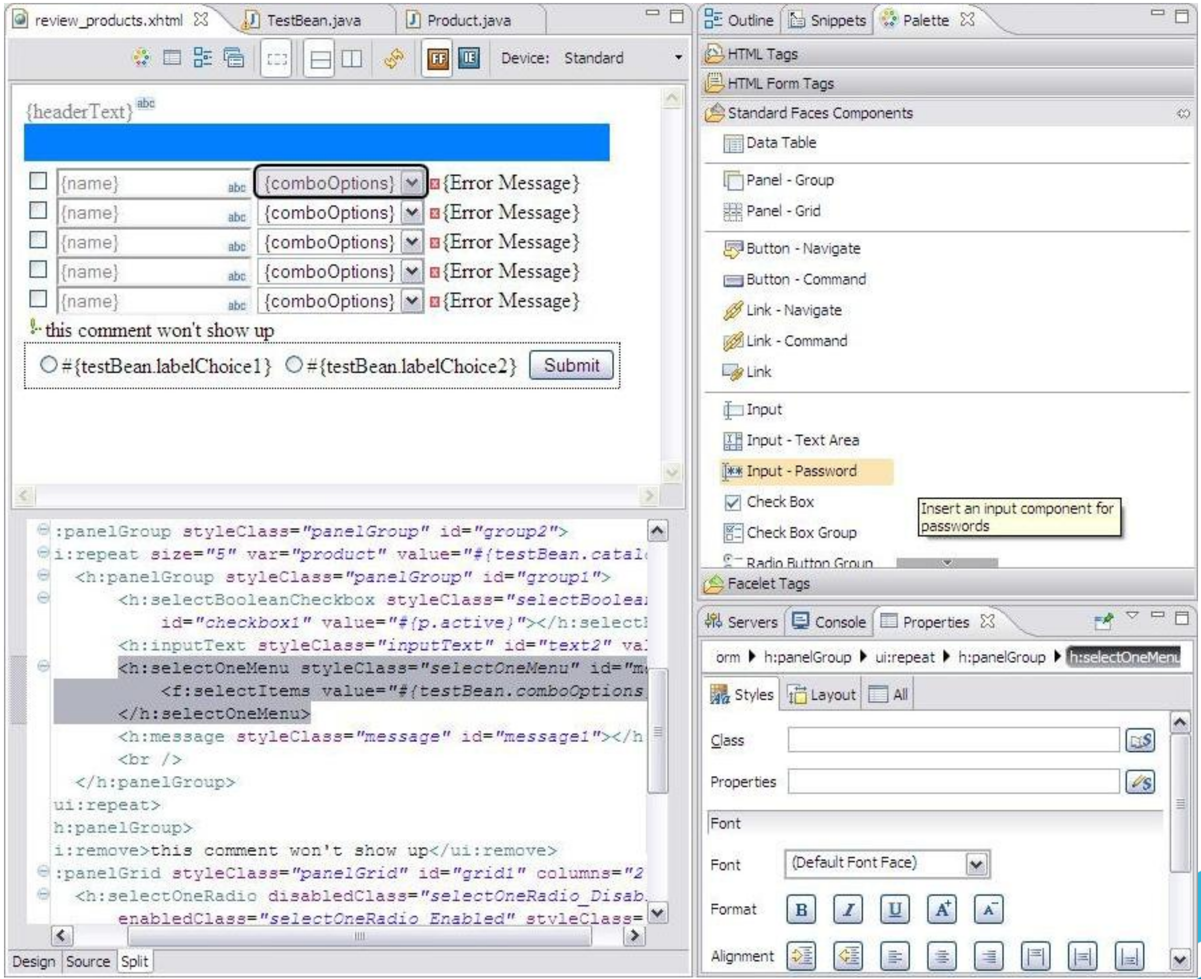
- New Tooling for building an exceptional web experience
- Enable Ajax, JAX-RS, Dojo, analytics and more at a single click
- High-fidelity editing with the Rich Page Editor
- Smooth and seamless integration of JSF 2.0 content
- Latest platform: WebSphere Portal V8.0





JSF - JavaServer Faces tools

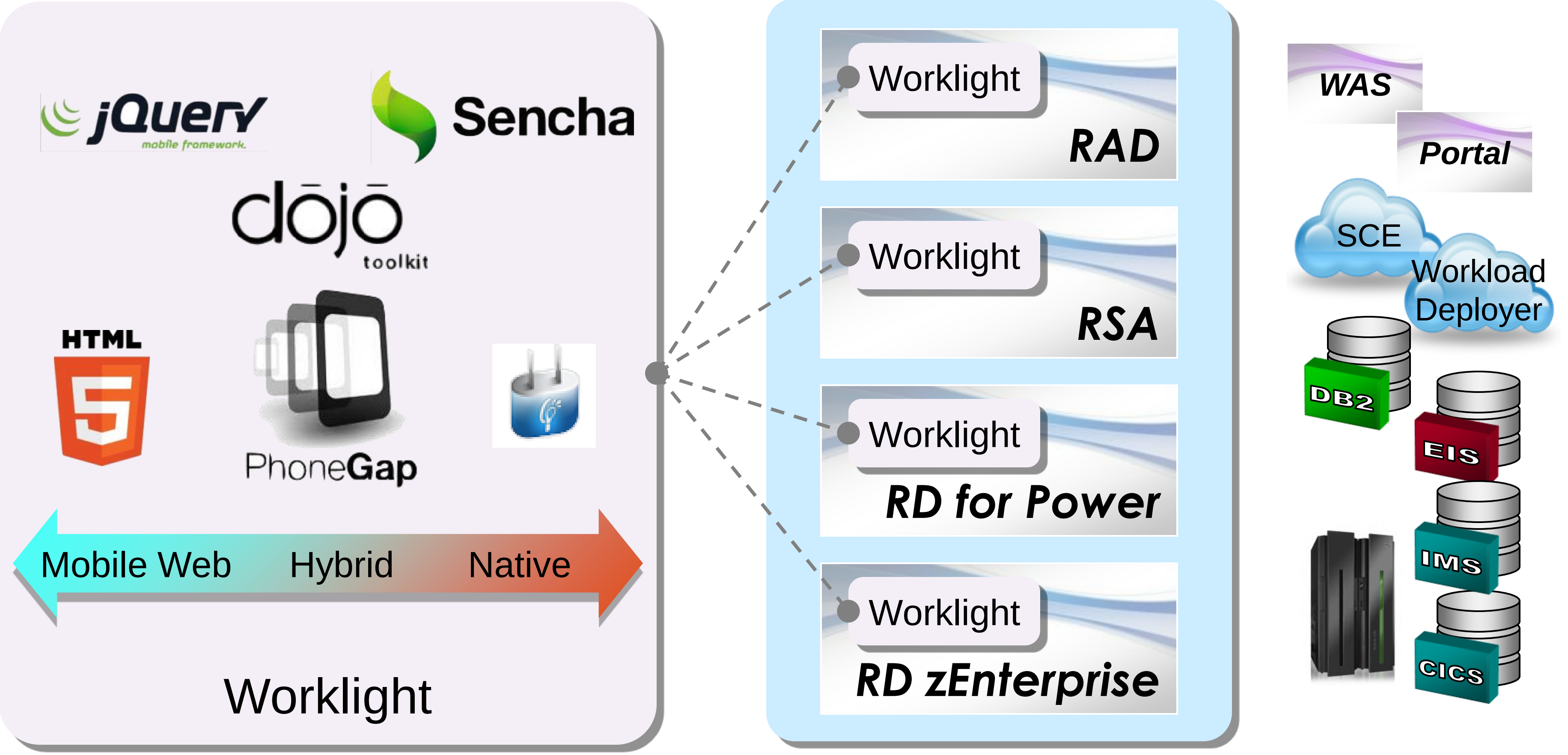
- Supports JSF 1.2 and Facelets with JSF 2.0
- Easily build custom components
- Any component can be configured to initiate Ajax requests
- JSF Trace tools aids in understanding and debugging
- Rich Page Editor support for JSF 2.0



Insert an input component for passwords

```
:panelGroup styleClass="panelGroup" id="group2">
i:repeat size="5" var="product" value="#{testBean.catal
  <h:panelGroup styleClass="panelGroup" id="group1">
      <h:selectBooleanCheckbox styleClass="selectBoolea
          id="checkbox1" value="#{p.active}"></h:select
      <h:inputText styleClass="inputText" id="text2" va
      <h:selectOneMenu styleClass="selectOneMenu" id="m
          <f:selectItems value="#{testBean.comboOptions
      </h:selectOneMenu>
      <h:message styleClass="message" id="message1"></h
      <br />
  </h:panelGroup>
ui:repeat>
h:panelGroup>
i:remove>this comment won't show up</ui:remove>
:panelGrid styleClass="panelGrid" id="grid1" columns="2
  <h:selectOneRadio disabledClass="selectOneRadio_Disab
      enabledClass="selectOneRadio Enabled" styleClass=
```

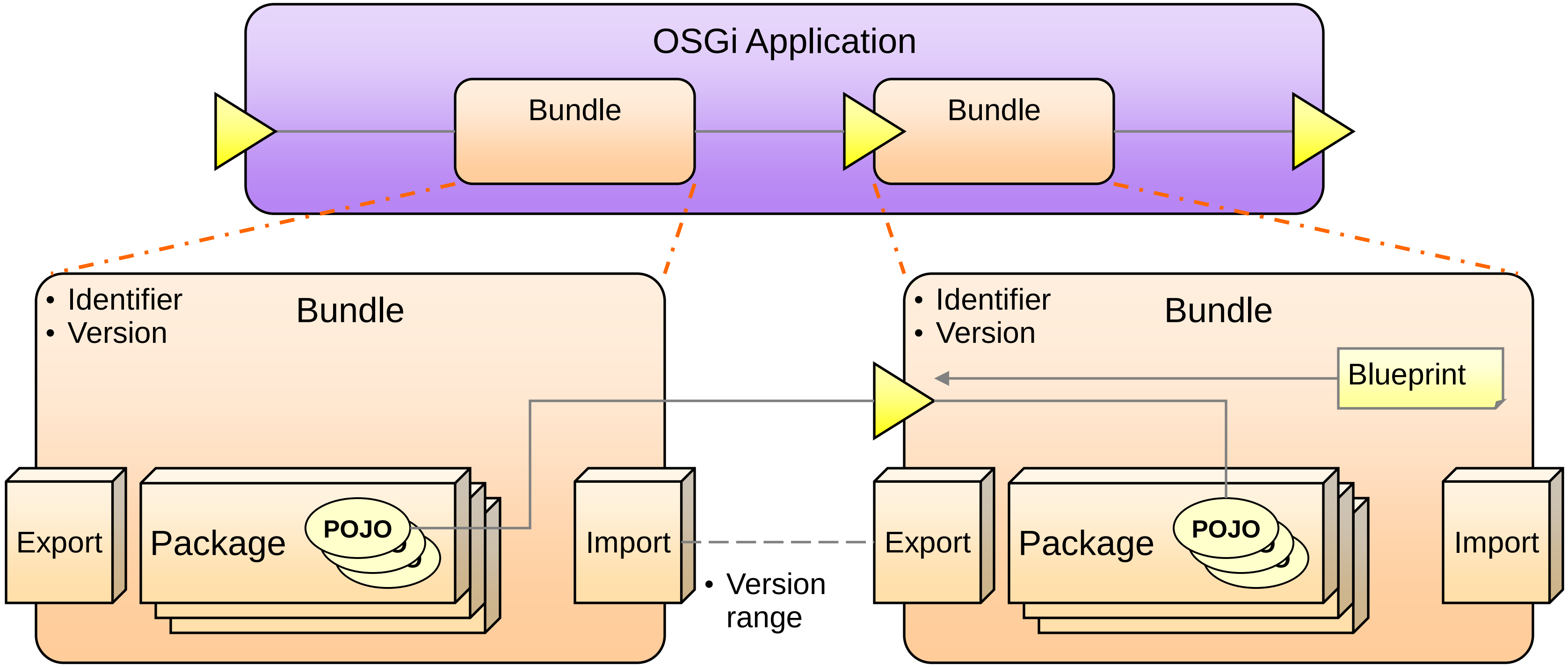
Extend the Enterprise with Mobile Application Development Rational IDEs including IBM Worklight (for development use)

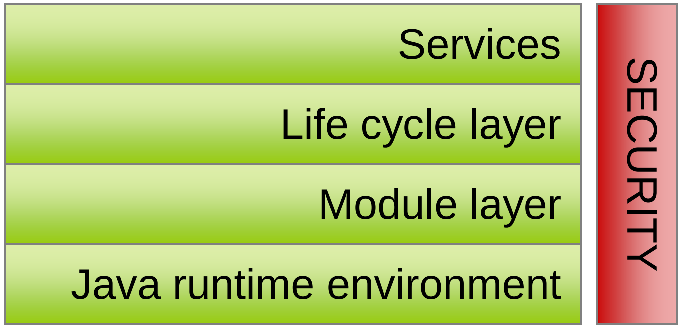


Create and test a mobile UI … to extend … enterprise application development

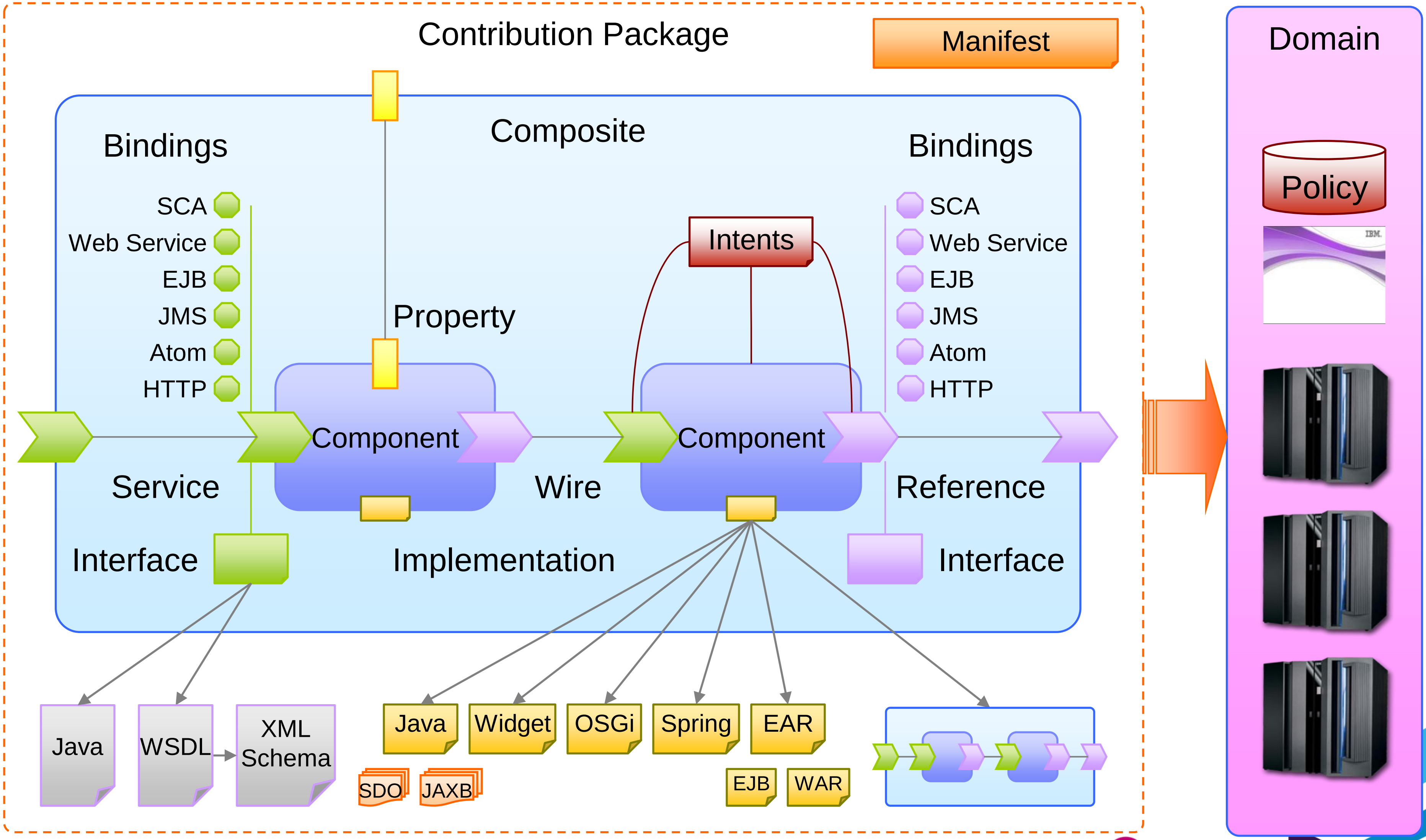
Next NOW!

The general architecture of OSGi



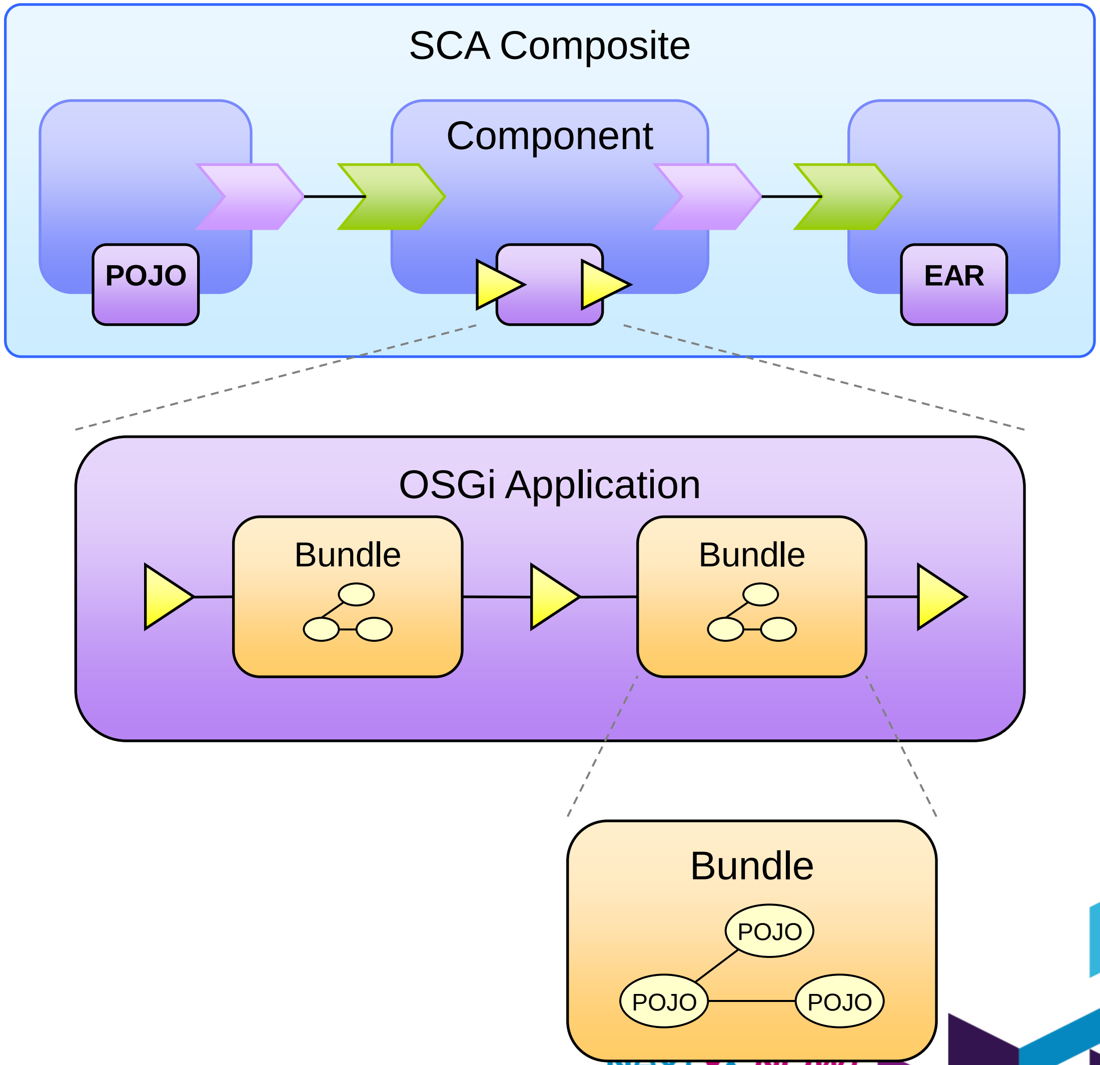


The general architecture of SCA



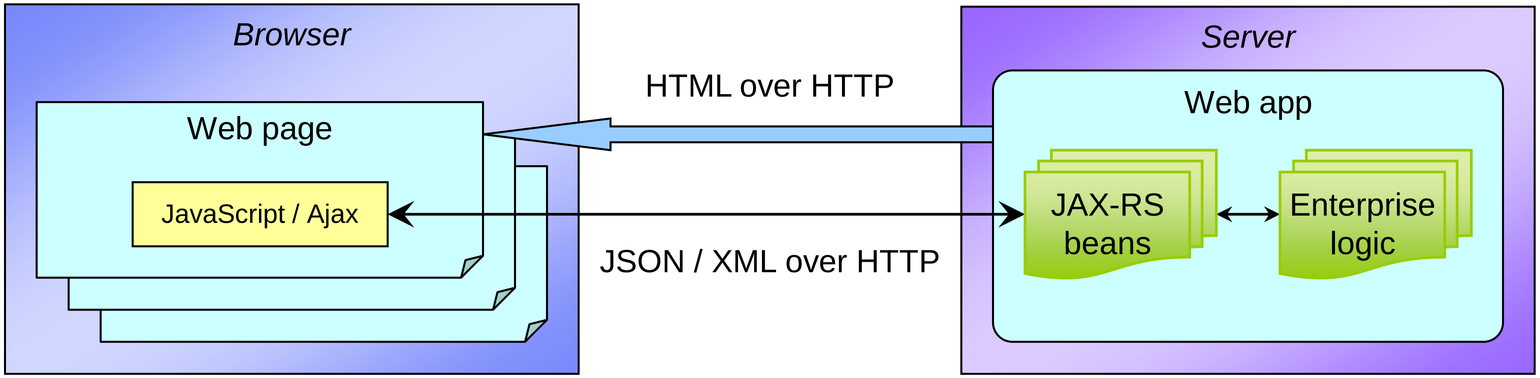
SCA & OSGi for coarse & fine grained SOA
Practically made for each other

- Use SCA to assemble reusable composites out of services realized by diverse technologies, including…
- OSGi applications, which consist of bundles that offer and consume Blueprint services thru a service registry, where the services are…
- Plain Java objects described in a Blueprint.



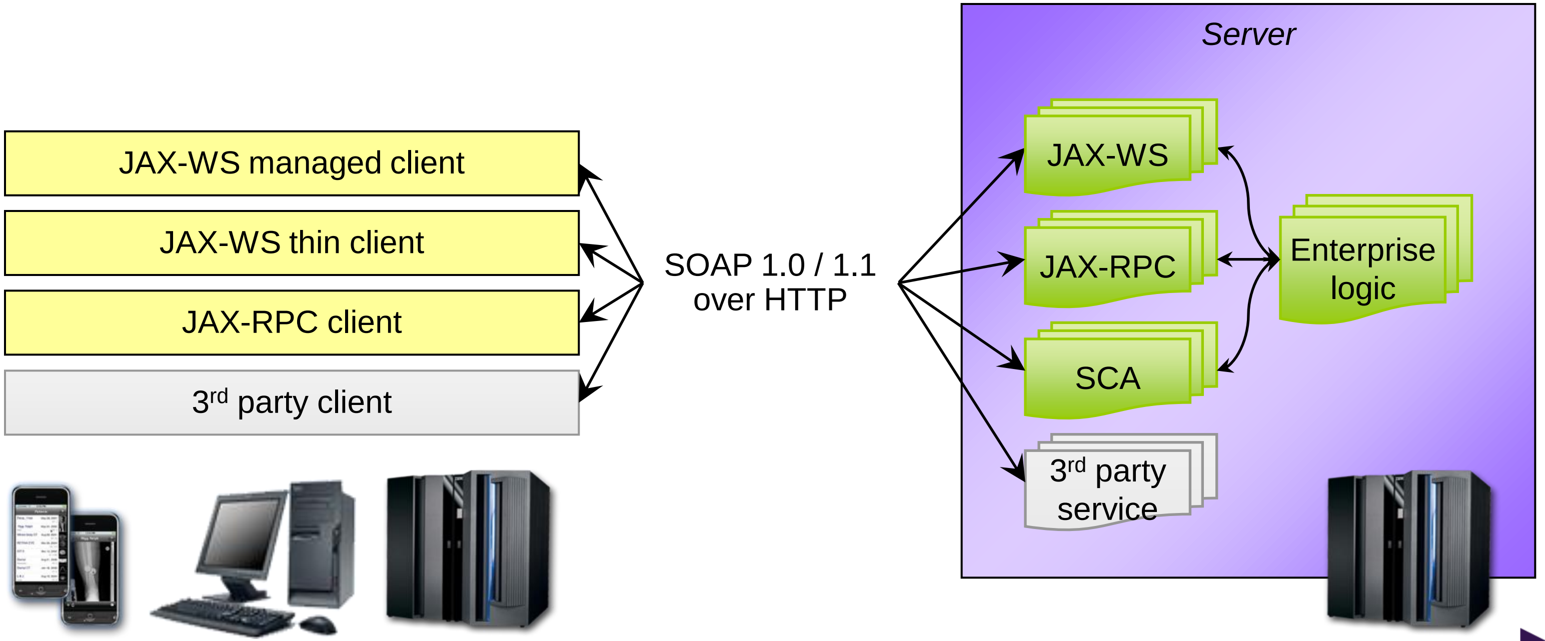
Java EE REST services

- JAX-RS
  - Supported by WAS V6.1, V7, V8 and V8.5 (Full and Liberty Profiles)
  - Includes JAX-RS 1.1 library providers for the target run-times
  - Easily enabled with the REST template in the Web Project Wizard
  - Easily integrated with Dojo (or other JavaScript) Ajax applications



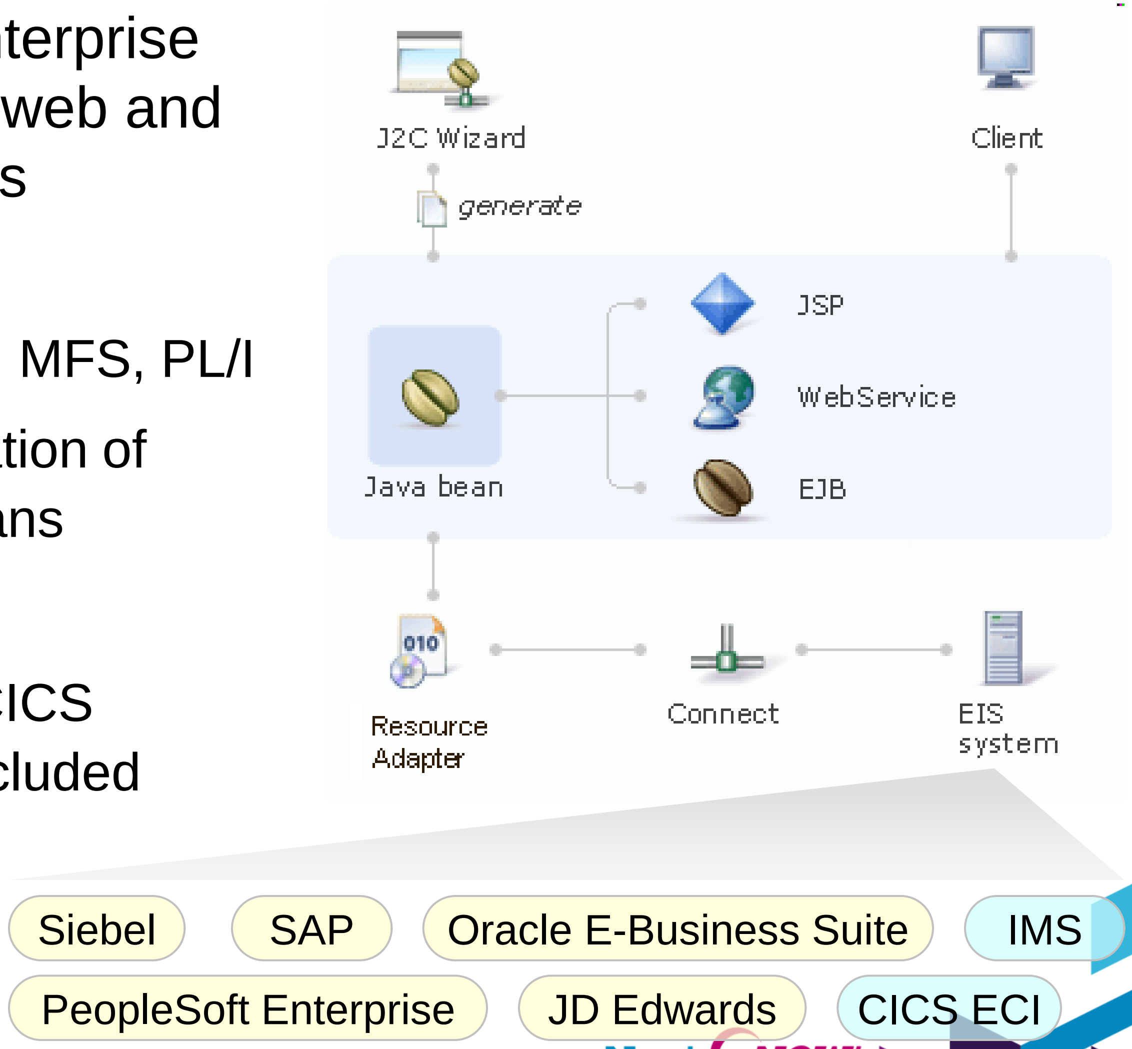
Java EE web services

- JAX-WS
  - Supported on WAS V6.1, V7, V8 and V8.5 (Full Profile, JDK 6 or 7)
  - Support for service, managed client and thin client development
  - Support for transport and message level security



Java EE – JCA
aka. J2C

- Rapidly integrate your enterprise information systems into web and other modern applications
- CICS/IMS
  - – Languages - COBOL, C, MFS, PL/I
  - – Record/playback generation of connection and data beans
- CICS
  - – Development use only CICS Transaction Gateway included
  - – JCA 1.6 for CICS 8.1



Siebel

SAP

Oracle E-Business Suite

IMS

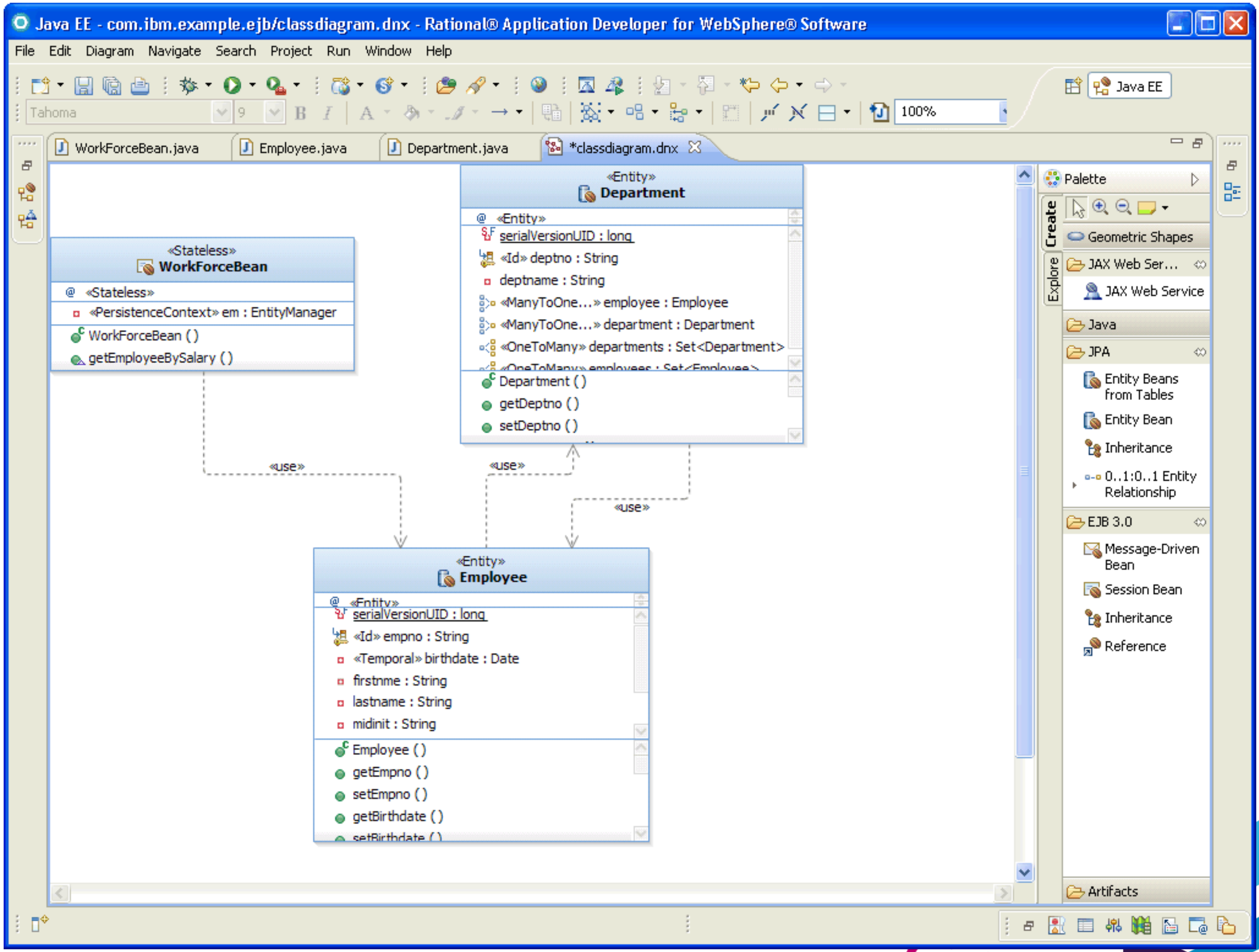
PeopleSoft Enterprise

JD Edwards

CICS ECI

Visualizing code with UML

- How can I possibly grok this pile of code I just inherited?
- Visualize classes, beans and web services with class, sequence and topic diagrams



Getting the bugs out

- Intuitive, integrated debugging aids
  - – Step-by-step debugging automatically stops on entry to objects
  - – Logical display of simple or complex variables, incl. EMF and DOM
  - – Hide “internal” (application server) segments of stack traces
  - – Conditional breakpoint expressions, including stack trace patterns
- Multiple languages and programming models
  - – Java, EJB, SCA, JSP/JSF, JavaScript, XSLT, Jython…
  - – WebSphere Application Server and Portal Server
- Share debug sessions collaboratively with Rational Team Concert

Going headless
Ant tasks and the RAD Build Utility

- Build process automation, efficiency and consistency can be improved by offloading manual developer tasks in the RAD UI to headless tasks integrated into the build
- RAD includes Apache Ant tasks that perform several operations available from the RAD workbench, but without the “head”, like:
  - Export and import tasks for common types of projects
  - Application installation tasks
  - EJB access bean and deployment code generation tasks
  - Workspace server management tasks
- RAD Build Utility
  - A separately installable, stand-alone feature limited to a headless RAD and its Apache Ant tasks

Software development on planet Earth is a 24/7 global exercise

How Can I Manage the Volume and Variety of My Teams?

Can I Handle Single Vendor, Multi-Vendor Trade-offs and Cultural Change?

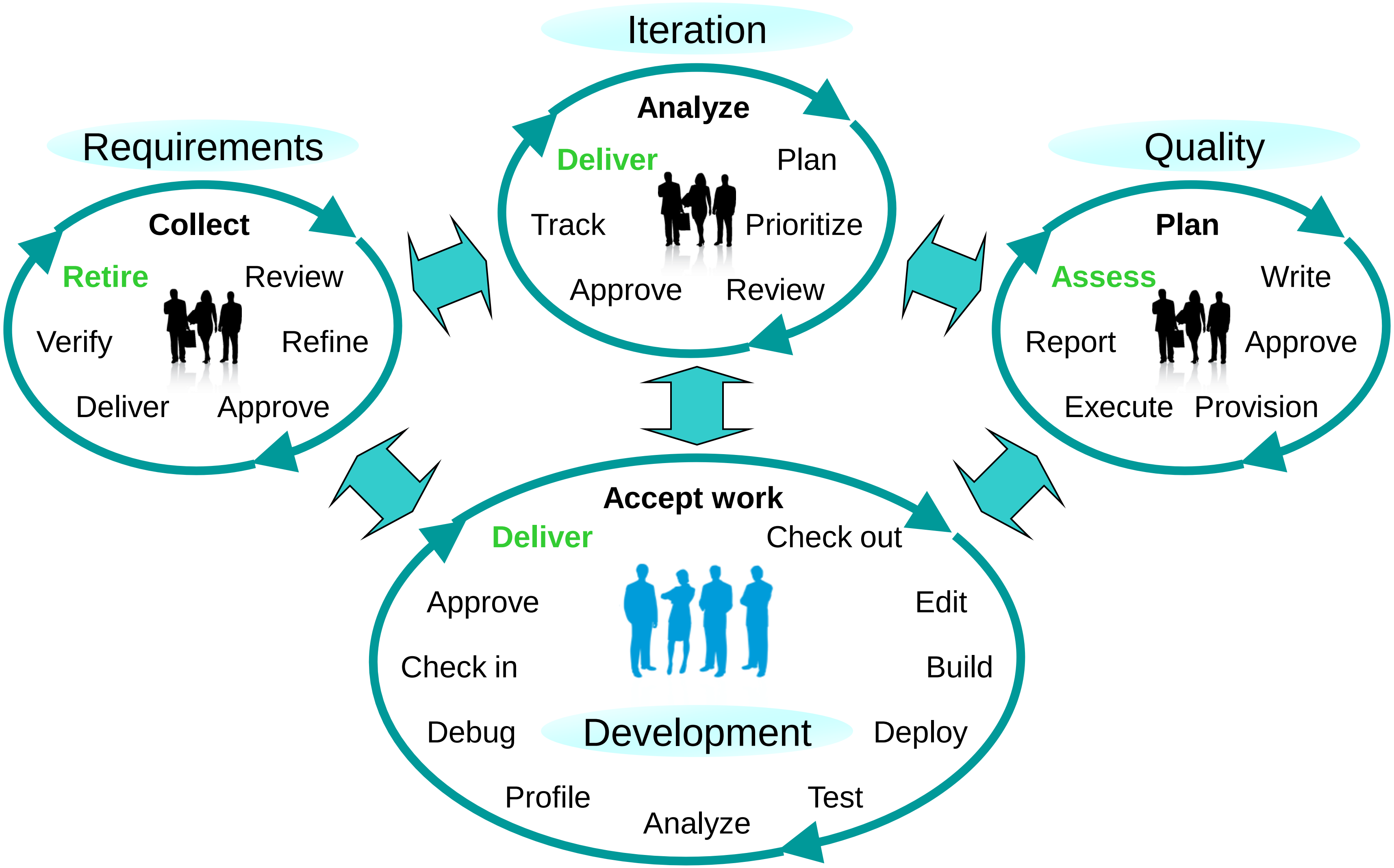
How Do I Align Day-to-Day Efforts with Executive Expectations?

Can I Integrate Disparate Information Silos Without Disrupting Processes?

Silos of people, process, and projects

Geo-social barriers	Organizational barriers	Infrastructure barriers
• Poor communication • Language, culture, time • Process gaps • Duplication and rework • High degree of friction	• Lack of collaboration • Weak project governance • Lack of domain expertise • Poor LOB oversight • Security of IP outsourcing	• Incompatible tools • Incompatible repositories • Unreliable access artifacts • Lengthy on-boarding • Inflexible tool integration

Understanding the lifecycles of development



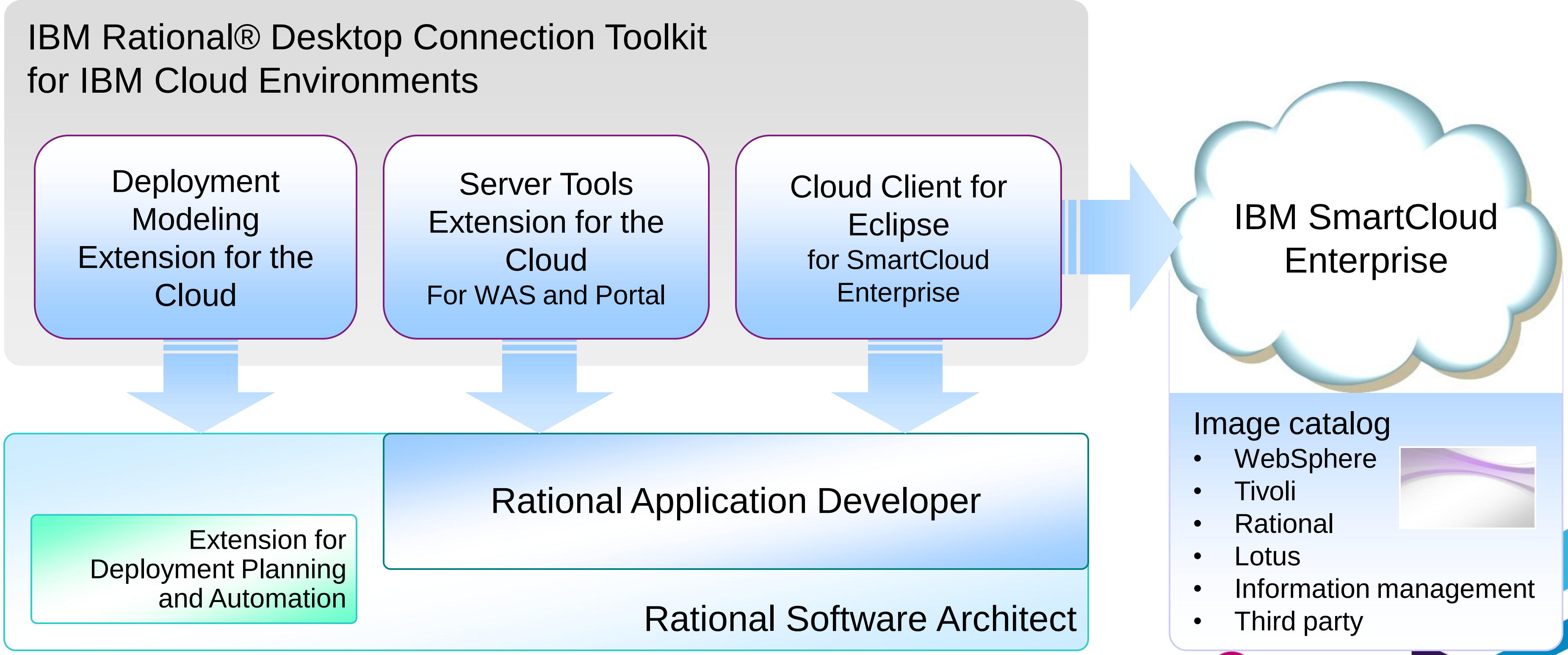
What we’re hearing

- Improve the development experience
  - Inadequate desktop power - poor performance
  - Installing and updating software is expensive and error-prone
  - Working from anywhere, especially mid stream, is hard to do
- Improve the administration / maintenance experience
  - Network updates (required for consistency and security) take too long
  - Update distribution tools can’t handle large update images
  - On-boarding individuals, teams and projects takes too long (procurement)
  - Versions and configurations of desktop software get out of sync easily
  - Governing which updates reach who, and when, is costly or inflexible.

Only in RAD

Integrating with IBM SmartCloud Enterprise

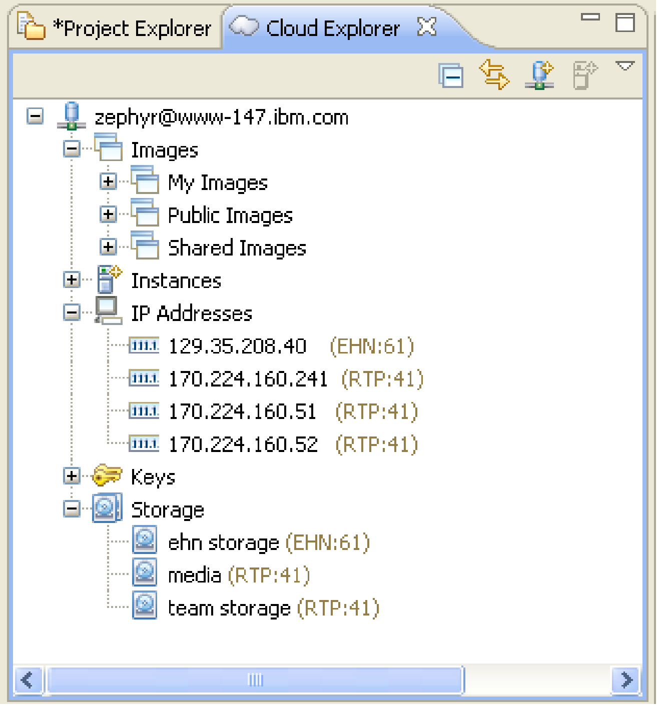
- Manage IBM SmartCloud Enterprise resources from RAD or RSA
- Offload WebSphere Application Server and Portal Server test environments to the public cloud


IBM Rational® Desktop Connection Toolkit for IBM Cloud Environments
Deployment Modeling Extension for the Cloud
Server Tools Extension for the Cloud For WAS and Portal
Cloud Client for Eclipse for SmartCloud Enterprise
IBM SmartCloud Enterprise
Extension for Deployment Planning and Automation
Rational Application Developer
Rational Software Architect
Image catalog
WebSphere
Tivoli
Rational
Lotus
Information management
Third party

Only in RAD

IBM Rational Desktop Connection Toolkit for IBM Cloud Environments – For RAD and RSA

- Cloud Client
  - Provision, explore and manage images, instances, storage, keys and addresses on SmartCloud Enterprise from the comfort of the IDE
- Cloud server tools
  - Provision instances of WebSphere Application Server or WebSphere Portal Server on SmartCloud Enterprise in just minutes using the “New Server” wizard
  - Develop, deploy, test and debug your code on the resulting server as normal



- zephyr@www-147.ibm.com
  - Images
    - My Images
    - Public Images
    - Shared Images
  - Instances
  - IP Addresses
    - 129.35.208.40 (EHN:61)
    - 170.224.160.241 (RTP:41)
    - 170.224.160.51 (RTP:41)
    - 170.224.160.52 (RTP:41)
  - Keys
  - Storage
    - ehn storage (EHN:61)
    - media (RTP:41)
    - team storage (RTP:41)

Next NOW!

The journey to Platform as a Service
IBM Workload Deployer

1 Hardware Virtualization

- Virtualization of hardware resources in the data center
- Management of virtualized infrastructure

Infrastructure Management

2 Image Virtualization

- Virtualized infrastructure leads to creation of “virtual” software images
- Proliferation of virtual software images leads to management challenges

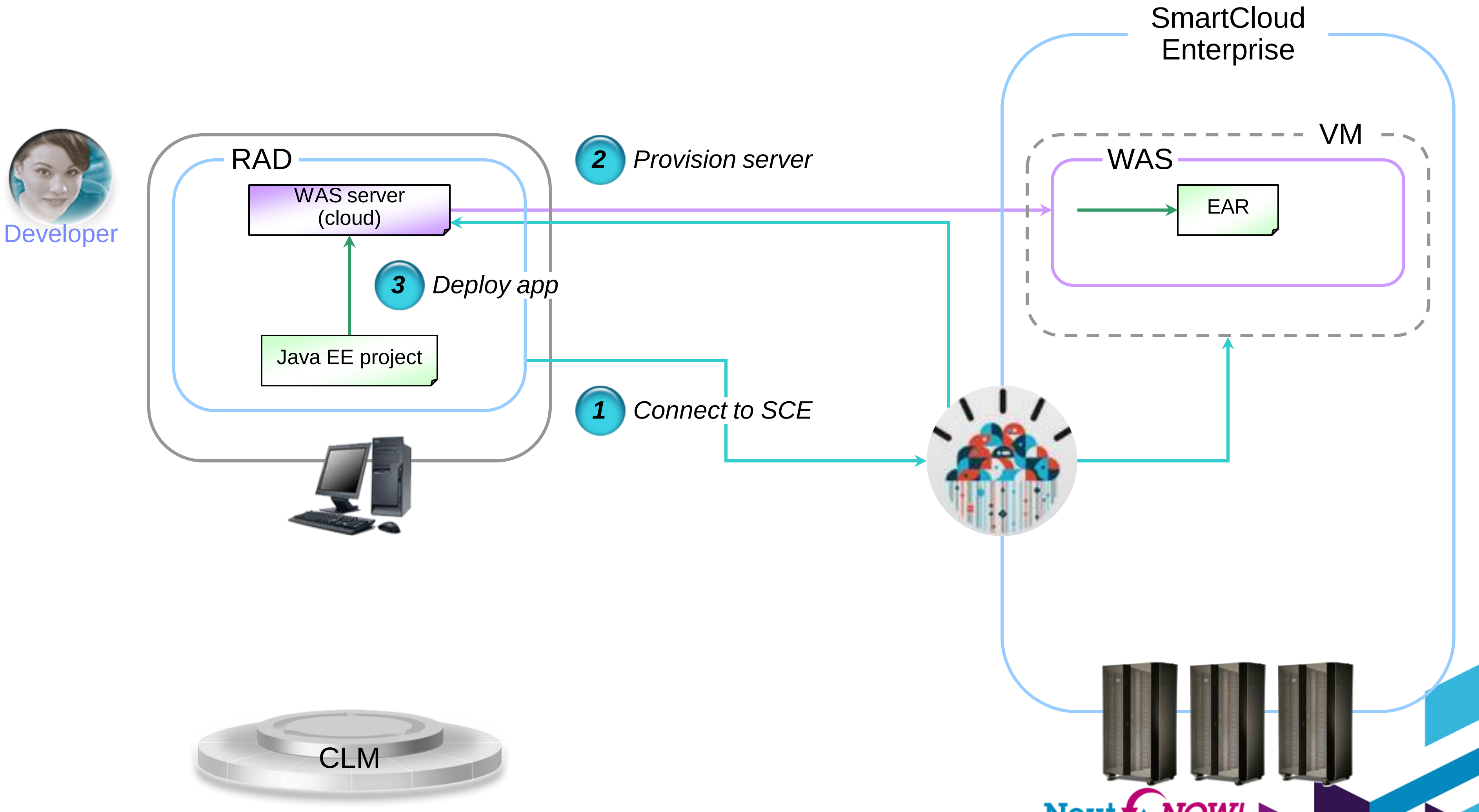
Image Management

3 Workload Virtualization

- Images are combined into patterns representing middleware workloads
- Workloads encapsulate well defined combinations of integrated middleware

Integrated Middleware

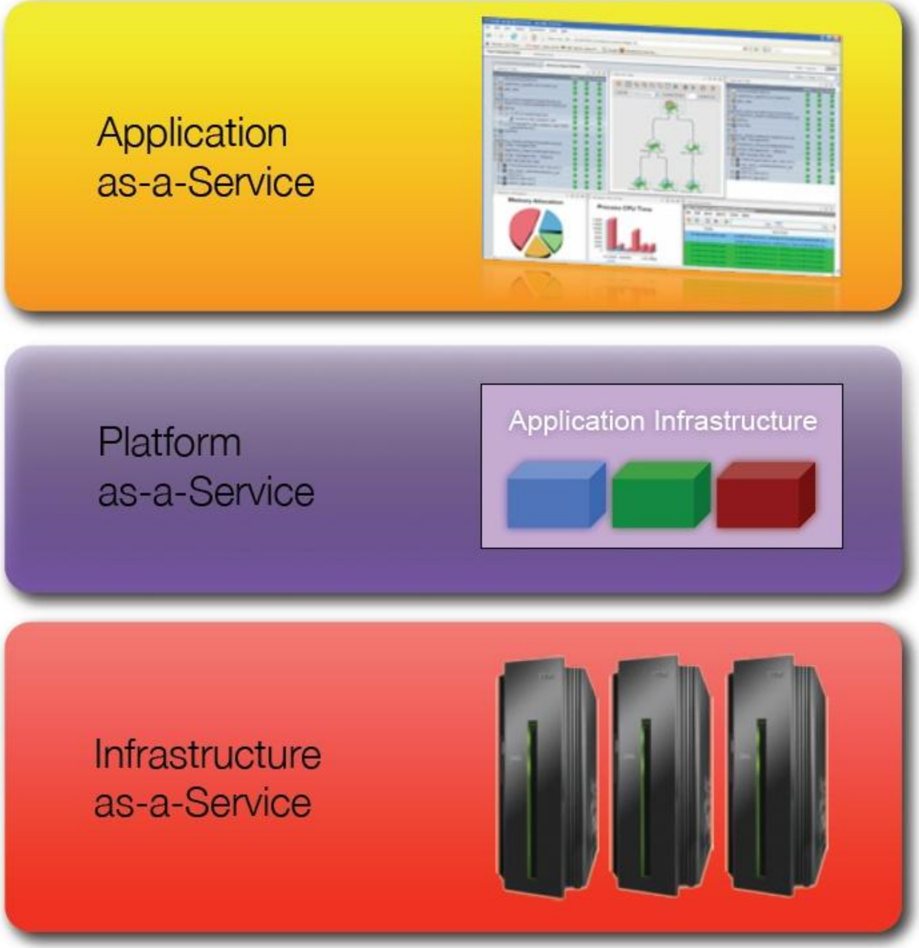
Developing applications for the WebSphere platform on IBM SmartCloud Enterprise



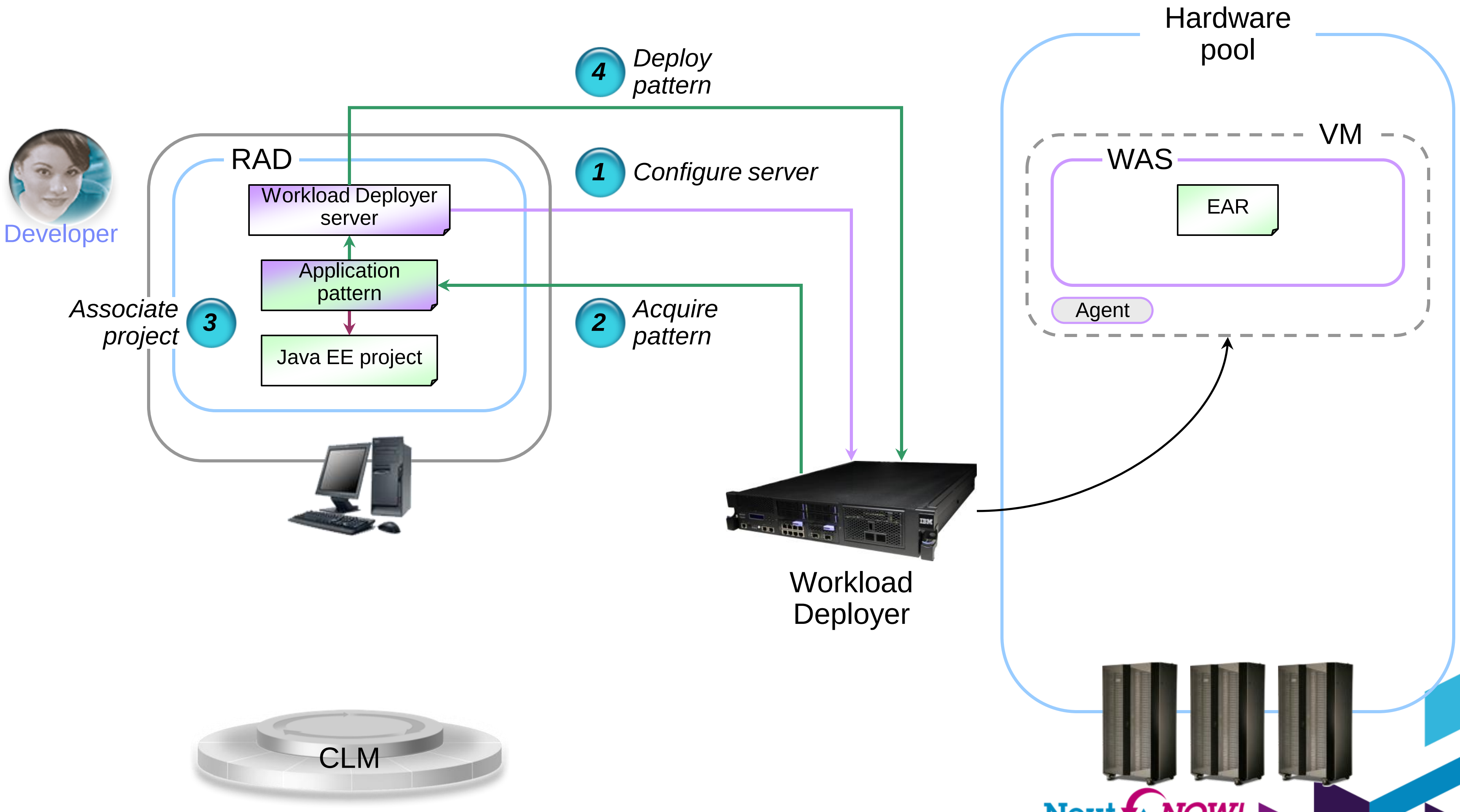
Infrastructure vs. Platform as a Service

- PaaS
  - Central abstraction: Application
  - Cloud provides middlewares, DBs, security, reliability, connectivity, queues, caching, routing, storage.
  - Cloud perceives an application as supporting a set of programming models and understands the needs of the app.

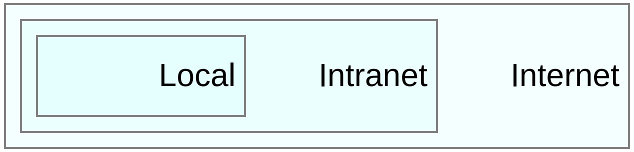
- IaaS
  - Central abstraction: Virtual Machine
  - Cloud provides CPU, network, storage.
  - IaaS neither understands nor cares about the software running on the VM.



Developing virtual application patterns



Help me!



- Information center
  - Hosted on the Internet (default)
  - Can be hosted on your intranet
  - Can be stored on your desktop
- Tutorials and samples
  - Watch and Learn tutorials performed for you.
  - Do and Learn tutorials yourself.
  - Ready to run examples of applications created with the tools
- RAD wiki, Redbooks, developerWorks articles and more

Migration

Workspace migration

- From RAD 8.0, 7.5 or 7.0
- Migration techniques
  - Check projects out of a repository
  - Import projects
  - Open a workspace
- Workspace migration wizard
  - Select projects to migrate and review files that will be touched
  - Tag obsolete files for removal
  - Update target run-times
  - Review detailed results

Programming model migration

- Specification migration wizard
  - Migrate earlier version Java EE projects to Java EE 6
  - Compatible target run-times identified and updated
  - Deployment descriptors migrated
    - web.xml, webservices.xml, ejb-jar.xml, application.xml, application-client.xml, …
  - Java source is not affected

RAD performance tuning

Many techniques exist for improving the performance of RAD

- Improve publication performance
  - Multiple tactics available for this
  - Try the WAS V8.5 Liberty Profile!
- Use source when you need it. Use binaries the rest of the time.
- Control WAS annotation scanning
- Tune validation preferences
- Upgrade hardware or go virtual
- Only install what you need
- Tune your anti-virus software
- Offload your unit test server to another system or to the cloud
- Avoid circular dependencies on build paths
- The basics: Defragment!

Learn more on the RAD information center
Troubleshooting and support → Improving performance → Performance tips