



IBM Software Group

# IBM Application Development **Ease-of-Use** with JavaServer Faces (JSF)

Go to <http://JavaDevTools.com> for presentations, demos, tutorials and other resources

(a.k.a. This is the SAME website as <http://RationalCentral.com> & <http://WebSphereCentral.com>)



Name  
Title  
Company

© 2003 IBM Corporation



# Application Development Challenges

- It takes too long to deliver applications  
“We need to *Design, Develop and Deploy applications more quickly*”
- J2EE is perceived to be too complex  
“We need to leverage open standards, keep it simple and make it easy for Microsoft heritage developers to get on board”
- Integration is cost prohibitive  
“We need to integrate our people, processes and information using web services, databases and existing systems”



# Developing Applications More Quickly

- Drag-n-drop development of J2EE applications (*No Coding Required*)

The screenshot displays the IBM WebSphere Portal interface in Microsoft Internet Explorer. The browser title is "IBM WebSphere Portal - Microsoft Internet Explorer". The address bar shows "http://yclidesktop.raleigh.ibm.com/wps/myportal/.scr/LoggedInIn". The page features a navigation menu with tabs: Home, Banking Account, Portfolio, Mortgage/Loans, Bill Pay, and ExtremePlace. The main content area is divided into several panels:

- Accounts:** A sidebar menu with options like Order Checks, Payment, and Help.
- Banking Accounts:** A table listing accounts with columns for Account #, Account Type, and Balance.
 

Account #	Account Type	Balance
96,666	Credit	2,000.00
74,444	Checking	1,432.44
85,555	Saving	2,567.56
- Account Details:** A table showing transaction history for Account Number: 96666.
 

Trans #	Amount	Date	Type	Expense Purpose
355	72.62	2000-11-06	Visa	Taxes
356	31.78	2001-06-16	Visa	Entertainment/Travel
357	61.02	2000-08-25	Visa	Clothing
358	21.75	2000-01-25	Visa	Transportation
359	13.01	2001-03-24	Visa	Telephone
360	24.51	2002-01-20	Visa	Transportation
361	15.87	2001-02-11	Visa	Food
362	13.81	2001-05-23	Visa	Telephone
- Transfer Funds:** A form titled "Make Transfer" with fields for "From:" (Select Account), "To:" (Select Account), and "Amount: \$".
- Expense Allocation:** A pie chart showing the distribution of expenses across various categories.
 

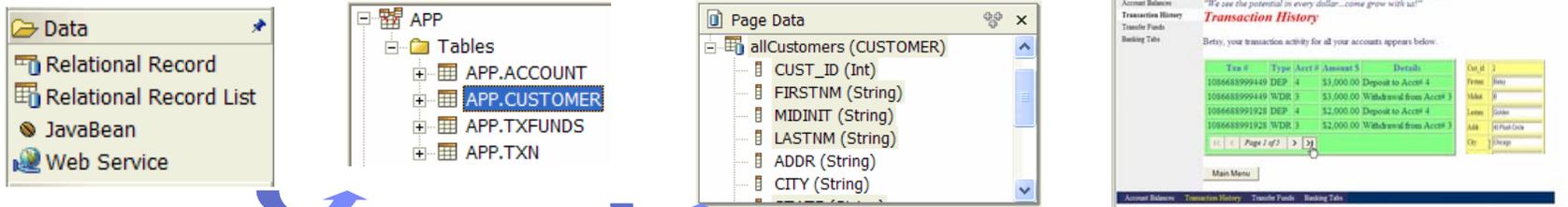
Category	Color
Mendical/Dental	Light Pink
Mortgage/Rent	Red
Food	Yellow
Gas/Electric	Light Green
Cloth	Cyan
Insurance	Blue
Dependent Care	Grey
Taxes	Brown
Entertainment/Travel	Purple
Transportation	Light Grey
Telephone	Black

Callout boxes highlight the following features:

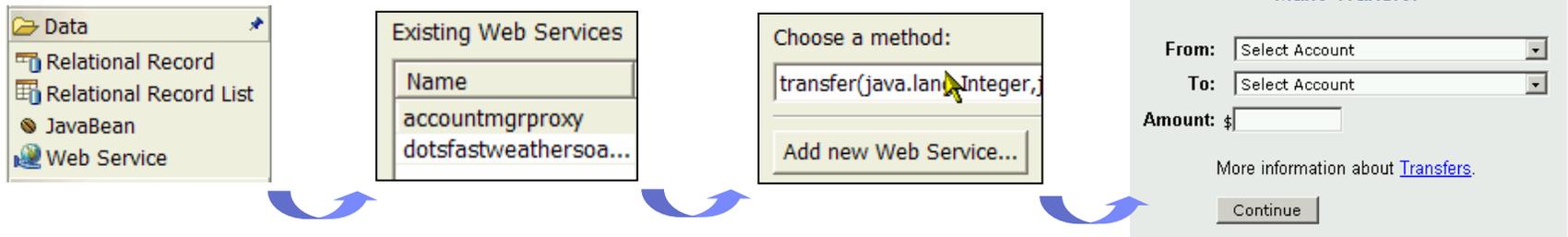
- Database Query:** Points to the Banking Accounts table.
- Web Service:** Points to the Transfer Funds form.
- Graphing Controls:** Points to the Expense Allocation pie chart.
- Tabbed Panels:** Points to the navigation tabs at the top.
- Spreadsheet Control:** Points to the Account Details table.

# Simplifying J2EE Development

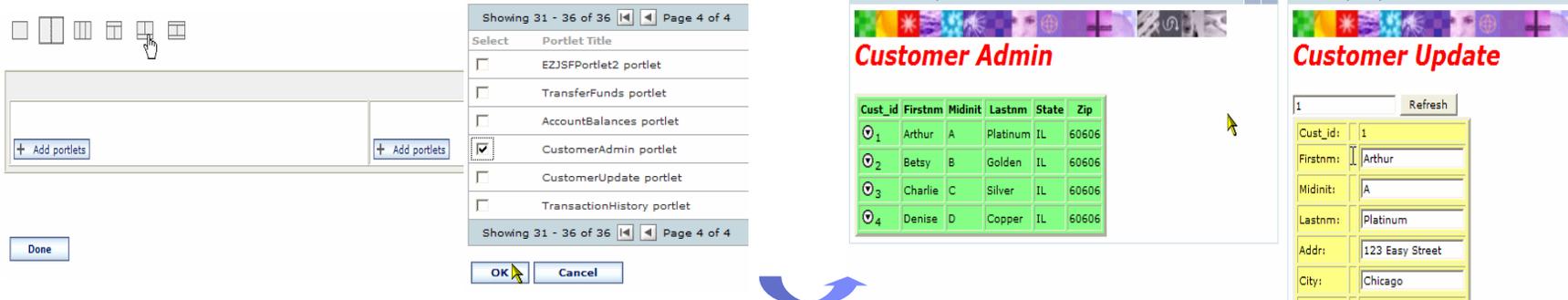
- Database development drag-n-drop ease-of-use <sup>1 2</sup>



- Web Services development drag-n-drop ease-of-use <sup>1 2</sup>



- Portals drag-n-drop ease-of-use <sup>1 2</sup>



# Integrating People, Processes and Information

- Model your business, simulate a process and assess cost savings

The screenshot shows a workflow simulation interface. On the left, a 'Risk Assess Business...' task leads to a decision diamond 'Achieve Final Approval?'. The 'Yes' path leads to 'Send Confirmat...' and the 'No' path leads to 'Send Risk Rejection...'. On the right, a 'Resource Cost Analysis' window displays the following data:

Resource Cost Analysis	
	Start Time
	Thursday, January 1, 2004 12:00:00 AM CST
Total	
Average Cost Per Time Unit	\$68.31 / 1 Hours
Average One Time Cost	\$0.00
Average Cost Per Quantity	
Average Cost Per Quantity & Time Unit	
Annual Per TimeUnit Cost	\$590,163.93

- Describe everything as a web service and integrate using workflow tools

The screenshot displays two views of a loan application process. On the left is a flowchart showing the process from 'Customers' (Call Center and Branch Offices) through 'Receive Application', 'Create Application' (using CICS), 'Check Credit' (using Microsoft .NET), an 'Auto Approve?' decision, a 'Loan Officer' role, 'Risk Assessment', and finally 'Approved' or 'Denied' outcomes. On the right is a BPMN diagram for 'QuickLoanProcess.bpel'. It features a 'Receive' start event, followed by a split into 'Sequence1' (containing 'CreateApplication') and 'Sequence2' (containing 'PrepCreditCheck' and 'CreditCheck'). A 'Partner Links' panel on the right lists various services like 'EISApp', 'CheckCredit', 'AutoApprove', etc.



# Application Development Challenges Addressed

- It takes too long to deliver applications  
“We need to *Design, Develop and Deploy applications more quickly*”
  - ✓ IBM’s new JavaServer Faces (JSF) tools accelerate development using a drag-n-drop approach (no coding required)
- J2EE is perceived to be too complex  
“We need to leverage open standards, keep it simple and make it easy for Microsoft heritage developers to get on board”
  - ✓ IBM’s new JSF tools make J2EE easier and very attractive to Microsoft heritage developers
- Integration is cost prohibitive  
“We need to integrate our people, processes and information using web services, databases and existing systems”
  - ✓ IBM’s new JSF tools automate web services and database integration
  - ✓ IBM’s business process integration and workflow tools save time





IBM Software Group

# IBM Application Development **End-to-End Development**

➔ V6 Update and RAD Top 10 Highlights ←

(For Complete “RAD Top 10” Presentation visit <http://JavaDevTools.com>)

*“Delivering applications faster, easier and with higher ROI”*



Name  
Title  
Company

© 2003 IBM Corporation

# V6 Tools Highlights - Areas of Innovation and Enhancements

## • Lifecycle Integrations (Enhanced Rational/WebSphere Eclipse based Integrations)

- Requirements Explorer View
- Model Explorer View
- ClearCase Views for managing changing artifacts
- ClearQuest Views for managing collaborative activities
- Testing Tools Views

## • RAD Construction Tooling Enhancements

- JSF, SDO, Web Services, etc.
- J2EE 1.4/1.3/1.2 and WAS V6.x/V5.x
- Portal/Portlet Integration
- Enhanced Unit/Component Testing
- Code Review and Runtime Analysis for Profiling
- Built-in Coder-friendly UML Visualizations for Class/Sequence diagrams
- Crystal Reports
- And much more...

## • New UML2 Engine

- New code base
- End-2-End UML2 modeling
- Easier GUI development
- Transformations between models and generating Java/C/C++ code

## • Testing Tools Enhancements (Manual, Functional and Performance Testing)



## V6 App Server Highlights

- **New Infocenter and Education Assistant**
- **Concepts and Terminology**
- **Programming Model**
- **System Management**
- **Rapid Deployment**
- **Enterprise Service Bus**
- **Clustering and High Availability**
- **Security**



# New Infocenter, including “Education Assistant”

- New Infocenter  
<http://ibm.com/software/webservers/appserv/infocenter.html>
- Education Assistant
  - Presentations narrated with sound
  - Tutorials
  - Reference information
- Installed with WAS and on [ibm.com](http://ibm.com)

WebSphere Application Server Library - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address <http://ibm.com/software/webservers/appserv/infocenter.html>

United States [change] Terms of use

IBM

Home Products Services & solutions Support & downloads My account

All software products Software > Application Servers > WebSphere Application Server >

WebSphere Application Server

Features and benefits

System requirements

Library

Success stories

News

Trials and betas

How to buy

Events

Training and certification

Services

Support

Support for z/OS

Related software

- WebSphere Application Server - Express
- WebSphere Application Server
- WebSphere Application Server Network Deployment
- WebSphere Application Server for iSeries
- WebSphere Application Server for z/OS
- WebSphere Extended Deployment
- Rational Application Developer for WebSphere Software
- Rational Web Developer for WebSphere Software
- Macromedia ColdFusion MX for WebSphere

Related solutions

- WebSphere Business Integration Server Foundation
- WebSphere Industry Solutions

**WebSphere Application Server**

WebSphere software

V 6.0 V 5.1.x V 5.0.x V 4.0.x

WebSphere Application Server Version 6.0.x Information Center

Express - distributed platforms  
[Show | Hide](#)

WebSphere Application Server - distributed platforms  
[Show | Hide](#)

Network Deployment - distributed platforms  
[Show | Hide](#)

Documentation - Information center format

- [WebSphere Application Server Network Deployment, Version 6.0.x](#)
- [WebSphere Help System plug-in: Network Deployment](#)
- [WebSphere Help System plug-in: Javadoc](#)

Documentation - Book format

- [Migrating, coexisting, and interoperating](#) (PDF)
- [Installing your application serving environment](#) (PDF)
- [Administering applications and their environment](#) (PDF)
- [Setting up the application serving environment](#) (PDF)
- [Using the administrative clients](#) (PDF)
- [Securing applications and their environment](#) (PDF)
- [Developing and deploying applications](#) (PDF)
- [Troubleshooting and support](#) (PDF)

Evaluation and overview

[Literature: Case studies, data sheets, and more](#)

**Tutorials, samples and demos**

- [IBM Education Assistant](#)
- [Accessing the Samples Gallery \(requires product installation\)](#)

Release notes

[Release notes](#)

**Supported hardware and software**

- [Supported hardware and software](#)
- [Supported hardware](#)
- [Supported software](#)

**Papers and articles**

- [WebSphere Application Server zone on developerWorks](#)
- [WebSphere Developer Technical Journal](#)
- [White papers](#)
- [Links recommended by product support](#)

**Supplemental books**

- [WebSphere Redbooks Domain](#)

Resources

- [WebSphere Help System \[?\]](#)

Related libraries

- [WebSphere Application Server for iSeries library](#)
- [WebSphere Application Server Express library](#)
- [IBM HTTP Server library](#)
- [WebSphere Application Server - Edge Components library](#)
- [WebSphere Extended Deployment library](#)
- [WebSphere Business Integration Server Foundation library](#)
- [WebSphere Business Integration Server Foundation for z/OS library](#)

# “Education Assistant” – RAD V6 and WAS V6

IBM Education Assistant - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites

Address <http://www-306.ibm.com/software/info/education/assistant/noflash.shtml>

Country/region [select]

Home | Products | Services & solutions | Support & downloads | My account

## IBM Education Assistant

IBM Education Assistant

**Related links:**

- WebSphere training and technical enablement

### Software for the e-business on demand era

IBM Education Assistant provides an organized collection of self-study educational materials that are simple, short, and focused. You can access content in several formats:

- Narrated Presentations - include audio explanation of the task or concept
- Show Me Demonstrations - illustrate how to complete a specific task
- Tutorials - step-by-step instructions to complete bigger or more complex tasks
- Additional resources - links to product documentation, Redbooks, and whitepapers

### How it works

The IBM Education Assistant user interface allows you to select your configuration and specify how you want to display links to the educational material. Clicking on an education module link presents a navigational hierarchy of available educational content.

→ [Click here to go to the education modules](#)

# “Education Assistant” – RAD V6 and WAS V6

## IBM Education Assistant

### IBM® WebSphere® Application Server V6.0

#### Overview and new features

Return to the IBM Education Assistant selection page

Icon key

New to WebSphere Application Server V6

[Architecture overview \(7 min\)](#)



869K

[Topology and terminology \(11 min\)](#)



907K

[Topology and terminology - Express \(5 min\)](#)



346K

Overview and new features

[Main themes, product packaging and development tools \(11 min\)](#)



378K

[Overview of new features \(18 min\)](#)



508K

Download the full content of this flow in zip format

See all of the above topics in a list and provide feedback on this material

## IBM Education Assistant

### IBM® Rational® Application Developer V6.0

#### Overview and new features

Return to the IBM Education Assistant selection page

Icon key

Overview and new features

[Rational Application V6 Overview and new features \(37 min\)](#)



789K

Download the full content of this flow in zip format

See all of the above topics in a list and provide feedback on this material

Click here to expand or collapse a category and see education flowcharts

A technical overview that provides background on a topic

A guided demonstration that shows how to complete a specific task

Step by step instructions to follow for completing a task

Additional reading material

Includes audio

To view PDF format, get [Adobe Acrobat Reader](#)



# “Education Assistant” – WAS V6 – Modules

## IBM Education Assistant



IBM Education Assistant

**Related links:**

- WebSphere training and technical enablement

Select your configuration:

IBM® WebSphere® Application Server V6.0

Select how to display results:

Task based view

Click on links below to view educational material.

Tasks	IBM WebSphere Application Server V6 and V6.0.1 for Distributed
Big picture and overview	<a href="#">Big picture, overview, product packaging, new V6 features, topology and terminology</a>
Installation and migration	<a href="#">Installation (WebSphere Application Server, IBM HTTP Server and Web server plug-in) and migration</a> <a href="#">WebSphere Application Server profiles</a>
Programming model	<a href="#">J2EE™ 1.4 programming model</a> <a href="#">JavaServer™ Faces (JSF) and Service Data Objects (SDO)</a> <a href="#">Web services</a> <a href="#">Programming model extensions (PME)</a>
System and application management	<a href="#">System, application and Web Server plug-in management</a>
Security	<a href="#">Security architecture and details</a>

Service integration technologies	<a href="#">Service integration technology, including messaging</a>
Scalability and high availability	<a href="#">Work load management and high availability details</a>
Other runtime functions	<a href="#">Class loader and JNDI naming</a> <a href="#">Transport Channel Service</a>
Performance	<a href="#">Performance, caches and tools</a>
Problem determination and serviceability	<a href="#">Problem determination - overview and details</a>



# “Education Assistant” – RAD V6 – Modules

## IBM Education Assistant



Select your configuration:

IBM® Rational® Application Developer V6.0 ▾

Select how to display results:

Task based view ▾

Go

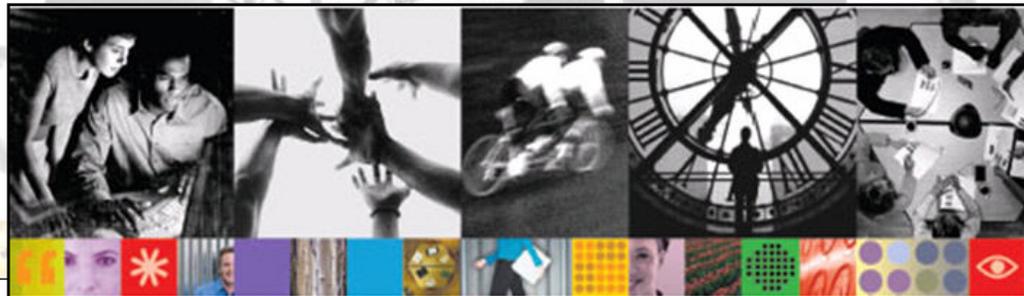
Click on links below to view educational material.

Application development	IBM Rational Application Developer V6
Getting started	<a href="#">Tool overview</a> <a href="#">Eclipse overview</a>
Installation and migration	<a href="#">Installation and migration</a>
J2EE™ application development	<a href="#">Developing Java™ and J2EE applications with Unified Markup Language (UML)</a> <a href="#">Analyzing coding practices in Java and J2EE applications</a> <a href="#">J2EE Web and EJB™ development</a>
Web application development	<a href="#">Developing Web applications (including J2EE Web applications)</a> <a href="#">Developing Web applications with JavaServer™ Faces (JSF)</a>
Portal application development	<a href="#">Developing portal applications</a>

Data application development	<a href="#">Developing applications with Service Data Objects (SDO)</a> <a href="#">Developing applications for databases</a>
XML application development	<a href="#">XML application development</a>
Web services applications	<a href="#">Developing Web services applications</a> <a href="#">Securing Web services applications</a> <a href="#">Testing Web services applications</a>
Test, debug and profile applications	<a href="#">Testing applications with WebSphere Application Server V6</a> <a href="#">Component testing applications</a> <a href="#">Debugging Java and J2EE applications</a> <a href="#">Profiling Java and J2EE applications</a>
Applying security	<a href="#">Applying J2EE security to EJBs and Web applications</a>



# New WebSphereCentral.com Tutorial Demos and more...


[JavaDevTools.com](#)
[IBM Links](#)
[About Bill Hahn](#)

## JavaDevTools.com

*"Navigating WebSphere, Rational, Workplace and other IBM Eclipse Based Tools..."*

Homepage for: [RationalCentral.com](#), [WebSphereCentral.com](#), [WorkplaceDev.com](#), [JavaSOA.com](#), [Bill-Hahn.com](#)

Disclaimer: This is a private web site intended for IBM Eclipse Tools enthusiasts. The materials on this site are offered as-is, no support, no warranties. This is not an official IBM website. To achieve the official IBM experience, please visit <http://ibm.com>.

**Welcome!** Rational Application Developer (RAD) is the foundation of the IBM Software Development Platform (SDP). The **links**,  **demos**  and  **presentations**  below can speed your navigation of IBM development tools. **Executives, managers, architects** and  **developers**  can use this site to learn more about Rational, WebSphere, Lotus and other IBM Eclipse-based development tools.

### Featured Demo

- o [RAD V6 Introduction and Tour Demo](#)

*This demonstration showcases RAD in action and concisely explains the breadth and depth of RAD V6. Executives, managers, architects and developers can use this overview and the referenced resources to quickly learn about the power and value of RAD V6.*

### Most Popular Links

- o [Presentations and Seminar Demos \(Downloads\)](#)
- o [Eclipse - Platform, Evolution, Ecosystem, and beyond \(Article\)](#)
- o [Rational - IBM Software Development Platform \(SDP\) Overview \(Whitepaper\)](#)
- o [WebSphere - IBM Enterprise Service Bus + WBI \(Whitepaper\)](#)
- o [Workplace - IBM Workplace Overview and Resources \(website\)](#)
- o [Portal DevZone - V5.0 Whitepaper](#) and [Portal V5.1 Enhancements](#)

### 2005/03/24 Updates

- o [RAD V6 Tutorial Demo Series has arrived!](#)

- o [RAD V6 Tutorial Demo Series has arrived!](#)

*(Getting Started Guide, Introduction and Tour)*

**Four scenarios with > 75 demos**

- 1) End-2-End (Soup-to-nuts J2EE Build Scenarios)
- 2) Quick Success (Short Ad-hoc Build Scenarios, < 10 min each)
- 3) Banking and Finance (Web App/Site Scenario)
- 4) Financial Advisor Dashboard (Portal/Portlet Scenario)

WebSphereCentral.com  
RationalCentral.com  
WorkplaceDev.com  
JavaDevTools.com  
JavaSOA.com  
Bill-Hahn.com

# RAD V6 Tutorial Demo Series

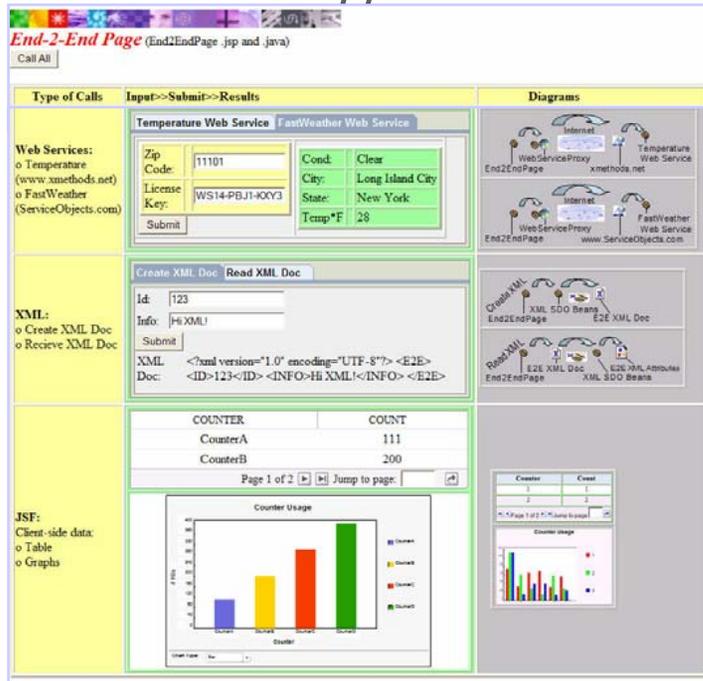
## Executives and Managers

- *Watch and Learn:* Simply view the "Tutorial Demo" movies to see ROI in action...in just minutes!

## Architects and Developers

- *Watch and Do:* Use the "Tutorial Demo" movies to build the reference applications from scratch!

### #1 End-2-End Application



**End-2-End Page** (End2EndPage.jsp and .java)

Call All

**Type of Calls** Input->Submit->Results

**Diagrams**

**Web Services:**

- o Temperature (www.smthods.net)
- o FastWeather (ServiceObjects.com)

**XML:**

- o Create XML Doc
- o Receive XML Doc

**JSF:**

- o Client-side data:
- o Table
- o Graphs

**Temperature Web Service** FastWeather Web Service

Zip Code: 11101 Cond: Clear  
 City: Long Island City  
 License Key: WS14-PBJ1-00Y3 State: New York  
 Temp°F: 28

**Create XML Doc** Read XML Doc

Id: 123  
 Info: Hi XML  
 XML: <?xml version="1.0" encoding="UTF-8"?> <E2E>  
 Doc: <ID>123</ID> <INFO>Hi XML!</INFO> <E2E>

**COUNTER** COUNT

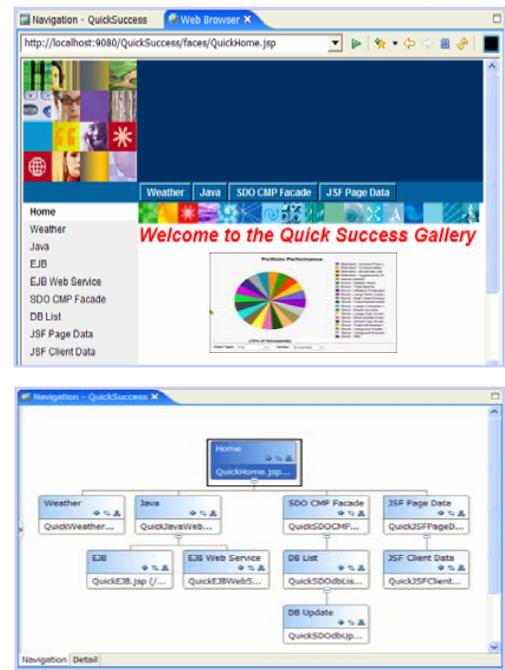
CounterA	111
CounterB	200

Page 1 of 2 | Jump to page: [ ]

**Counter Usage**



### #2 Quick Success Apps



Navigation - QuickSuccess

http://localhost:9060/QuickSuccess/faces/QuickHome.jsp

Home  
 Weather  
 Java  
 EJB  
 EJB Web Service  
 SDO CMP Facade  
 DB List  
 JSF Page Data  
 JSF Client Data

**Welcome to the Quick Success Gallery**

Navigation - QuickSuccess - X

Home  
 QuickHome.jsp...

Weather  
 QuickWeather...

Java  
 QuickJavaWeb...

SDO CMP Facade  
 QuickSDOcmp...

JSF Page Data  
 QuickJSFPageD...

EJB  
 QuickEJB.jsp...

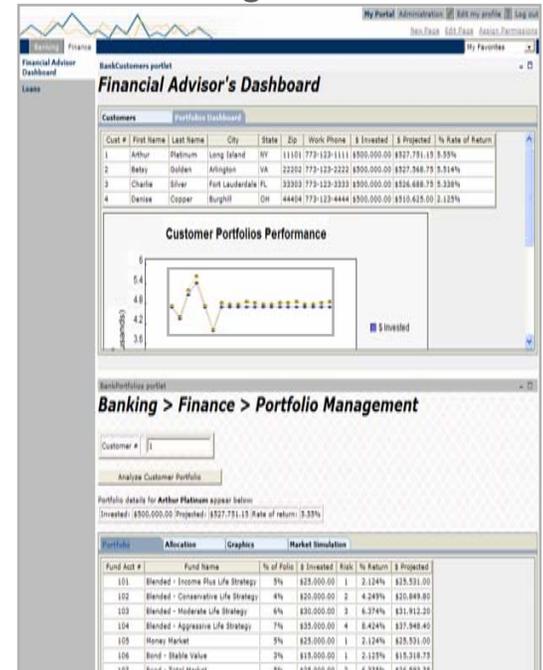
EJB Web Service  
 QuickEJBWebS...

DB List  
 QuickDOList...

JSF Client Data  
 QuickJSFClient...

DB Update  
 QuickSDOUpd...

### #3 Banking & Finance



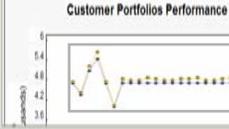
Financial Advisor Dashboard

Financial Advisor's Dashboard

Customers

Customer #	First Name	Last Name	City	State	Zip	Work Phone	Invested	Projected	% Rate of Return
1	Arthur	Platinum	Long Island	NY	11101	773-123-1111	\$500,000.00	\$527,751.13	5.55%
2	Betty	Golden	Arlington	VA	22202	773-123-2222	\$500,000.00	\$527,568.73	5.51%
3	Charlie	Silver	Fort Lauderdale	FL	33303	773-123-3333	\$500,000.00	\$526,688.73	5.33%
4	Dennis	Copper	Burghil	OH	44404	773-123-4444	\$500,000.00	\$510,625.00	2.12%

**Customer Portfolios Performance**



**Banking > Finance > Portfolio Management**

Customer # 1

Analyze Customer Portfolio

Portfolio details for Arthur Platinum appear below:  
 Invested: \$500,000.00 Projected: \$527,751.13 Rate of return: 5.55%

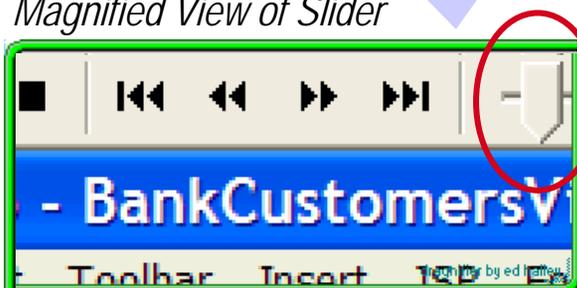
Fund Act #	Fund Name	% of Folio	Invested	Risk	% Return	Projected
101	Blended - Income Plus Life Strategy	5%	\$25,000.00	1	2.124%	\$25,531.00
102	Blended - Conservative Life Strategy	4%	\$20,000.00	2	2.249%	\$20,849.00
103	Blended - Moderate Life Strategy	6%	\$30,000.00	3	6.374%	\$31,912.00
104	Blended - Aggressive Life Strategy	7%	\$35,000.00	4	8.424%	\$37,948.40
105	Honey Market	3%	\$15,000.00	1	2.124%	\$15,318.00
106	Bond - Stable Value	3%	\$15,000.00	1	2.123%	\$15,318.75
107	Bond - Total Market	5%	\$25,000.00	2	6.375%	\$26,592.75

# RAD V6 Tutorial Demo Series – How They Work...

- The embedded movie player has playback controls like a DVD player.
- The SLIDER allows accelerated forward/reverse playback control.



Magnified View of Slider



BankCustomers portlet

### Bank Customers

Customers      Portfolios Dashboard

Cust #	First Name	Last Name	Address	City	State	Zip	e-mail	Work Phone
1	Arthur	Platinum	123 Easy Street	Long Island	NY	11101	Cash.A.Plenty@mail.com	773-123-1111
2	Betsy	Golden	45 Plush Circle	Arlington	VA	22202	Gold.P.Lated@mail.com	773-123-2222
3	Charlie	Silver	67 Humble Avenue	Fort Lauderdale	FL	33303	Simply.E.Nuff@mail.com	773-123-3333
4	Denise	Copper	89 Trailer Park Drive	Burghill	OH	44404	Always.N.Debt@mail.com	773-123-4444

BankPortfolios portlet

### Bank Portfolio Management

Customer #

Analyze Customer Portfolio

Portfolio      Allocation      Graphics      Market Simulation

Fund Acct #	Fund Name	% of Folio	\$ Invested	Risk	% Return	\$ Projected
301	Blended - Income Plus Life Strategy	5%	\$25,000.00	1	2.125%	\$25,531.25
302	Blended - Conservative Life Strategy	5%	\$25,000.00	2	4.25%	\$26,062.50
303	Blended - Moderate Life Strategy	5%	\$25,000.00	3	6.375%	\$26,593.75
304	Blended - Aggressive Life Strategy	5%	\$25,000.00	4	8.425%	\$27,106.25
305	Money Market	5%	\$25,000.00	1	2.125%	\$25,531.25
306	Bond - Stable Value	10%	\$50,000.00	1	2.125%	\$51,062.50

Paused 00:59/01:06

## End-2-End Application

Demo

"Accelerate development...  
increase productivity...  
maximize ROI! \$"

## Iterative Development Demos

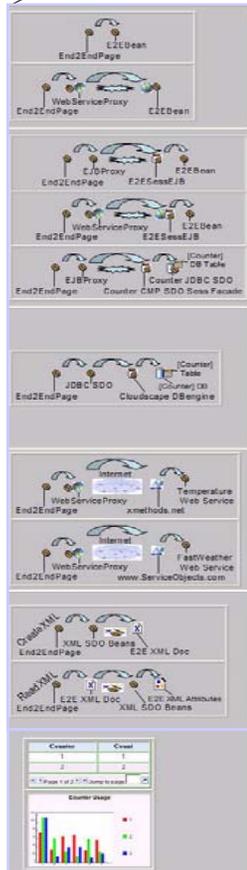
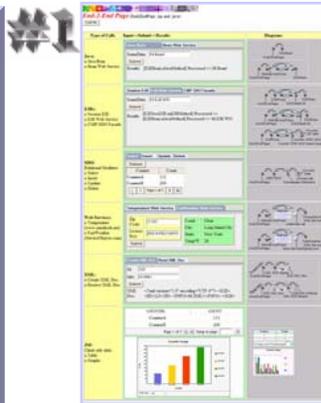
➔ Model, Build, Test, Debug, Deploy

- UML Model Visualizations
- Dynamic Development
- Automated Deployment

## J2EE+ Builder Demos

➔ Includes JSF, SDO, Web Services

- Java
- Databases
- Web Services
- Drag-n-Drop GUI



## Quick Success Applications

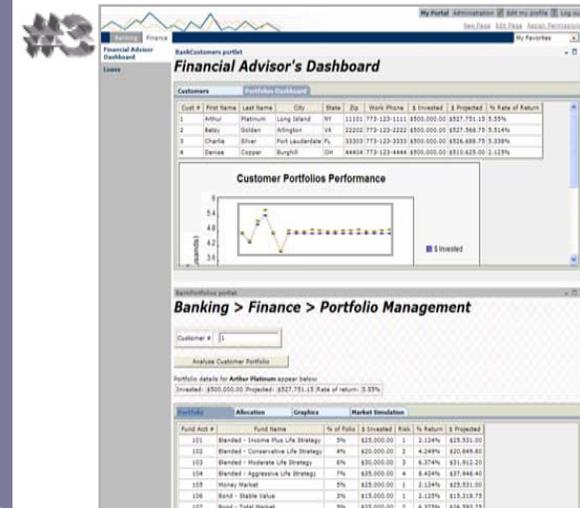
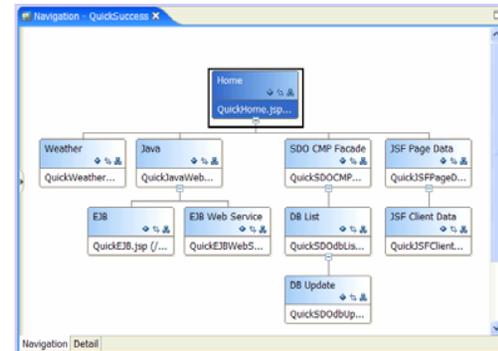
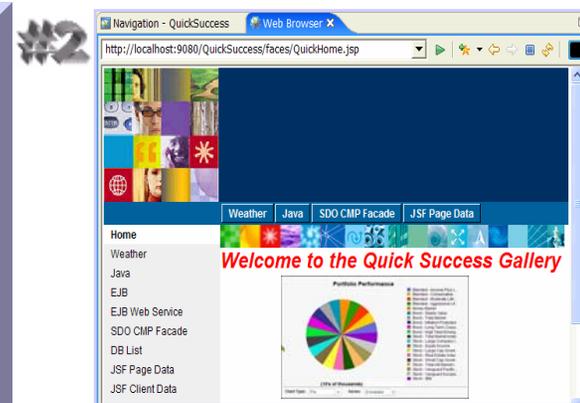
Demo

- Easy build scenarios
- You watch or build
- < 10 minutes each

## Banking & Finance Web/Portal Applications

Demo

- Banking Accounts
- Portfolio Management
- Financial Advisor Dashboard



# End-2-End Application

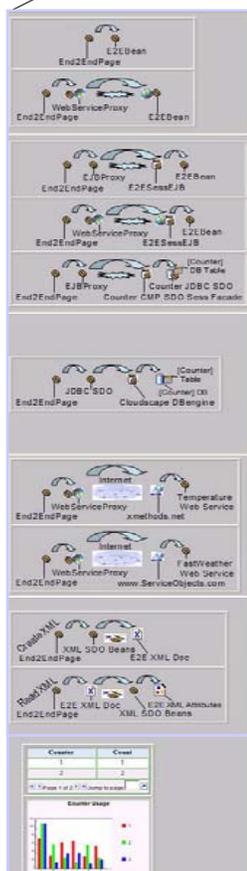
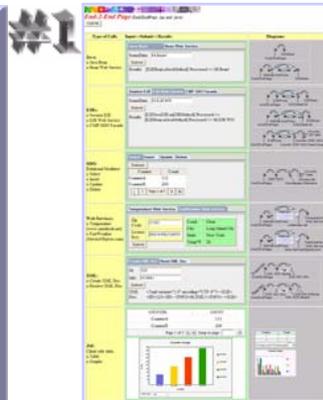
"Accelerate development...  
increase productivity...  
maximize ROI !\$! "

## Iterative Development Demos

- ➔ Model, Build, Test, Debug, Deploy
- UML Visualizations
- Dynamic Development
- On-the-fly code changes
- Incremental build engine
- Testing for Java, EJBs, Web Services, etc...
- Dynamic Interactive Debugger
- Automated Application Packaging and Deployment

## J2EE+ Builder Demos

- ➔ Includes JSF, SDO, Web Services
- Java and EJBs (Session, CMP)
- Service Data Objects (SDO)
- SDO CMP Session Facades
- XML and XSD Schemas
- XML SDO JavaBean Bindings
- JavaServer Faces (JSF)
- Database and SQL
- Web Services Creation and Consumption



## End-2-End Page (End2EndPage.jsp and java)

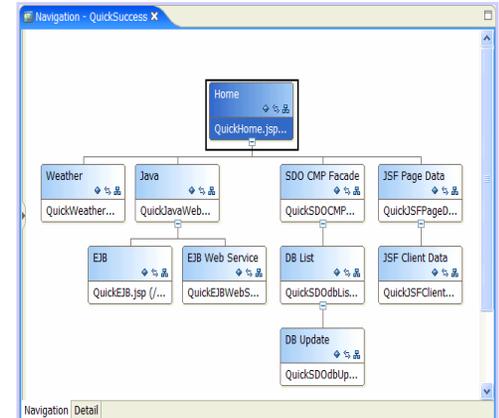
Call All

Type of Calls	Input>>Submit>>Results	Diagrams						
<b>Java:</b> o Java Bean o Bean Web Service	Java Bean    Bean Web Service SomeData:  Hi Bean! Submit Results: [E2EBean.aJavaMethod] Processed >> Hi Bean!							
<b>EJBs:</b> o Session EJB o EJB Web Service o CMP SDO Facade	Session EJB    EJB Web Service    CMP SDO Facade SomeData:  Hi EJB WS! Submit Results [E2ESessEJB.aEJBMethod] Processed >> [E2EBean.aJavaMethod] Processed >> Hi EJB WS!							
<b>SDO:</b> Relational Mediator: o Select o Insert o Update o Delete	Select    Insert    Update    Delete Refresh <table border="1"> <thead> <tr> <th>Counter</th> <th>Count</th> </tr> </thead> <tbody> <tr> <td>CounterA</td> <td>111</td> </tr> <tr> <td>CounterB</td> <td>200</td> </tr> </tbody> </table> Page 1 of 2	Counter	Count	CounterA	111	CounterB	200	
Counter	Count							
CounterA	111							
CounterB	200							
<b>Web Services:</b> o Temperature (www.xmethods.net) o FastWeather (ServiceObjects.com)	Temperature Web Service    FastWeather Web Service Zip Code:  11101    Cond:  Clear City:  Long Island City License Key:  WS14-PBJ1-KXY3    State:  New York Temp*F  28 Submit							
<b>XML:</b> o Create XML Doc o Recieve XML Doc	Create XML Doc    Read XML Doc Id:  123 Info:  Hi XML! Submit XML <?xml version="1.0" encoding="UTF-8"?><E2E> Doc: <ID>123</ID> <INFO>Hi XML!</INFO> </E2E>							
<b>JSF:</b> Client-side data: o Table o Graphs	<table border="1"> <thead> <tr> <th>COUNTER</th> <th>COUNT</th> </tr> </thead> <tbody> <tr> <td>CounterA</td> <td>111</td> </tr> <tr> <td>CounterB</td> <td>200</td> </tr> </tbody> </table> Page 1 of 2    Jump to page:	COUNTER	COUNT	CounterA	111	CounterB	200	
COUNTER	COUNT							
CounterA	111							
CounterB	200							

## Quick Success Gallery Applications

- Easy J2EE build scenarios
- Internet Weather Web Service
- JavaBean Web Service
- EJB Web Service
- Service Data Objects (SDO)
  - Database Mediator: Select, Insert, Update, Delete
- Java Server Faces (JSF)
  - Rich UI Controls, Charts/Graphs, drag-n-drop
  - Custom coding, Event Driven, MVC Framework

#2



## Banking & Finance Web/Portal Applications

- Transfer Funds
- Transaction History
- Account Management
- Portfolio Management
- Asset Allocation
- Graphical portfolio analysis
- Stock Market Simulation
- Mutual Funds, Stocks, Bonds, Money Mkt

#3

**Financial Advisor's Dashboard**

Customers	Portfolios Dashboard																																																		
<table border="1"> <thead> <tr> <th>Cust #</th> <th>First Name</th> <th>Last Name</th> <th>City</th> <th>State</th> <th>Zip</th> <th>Work Phone</th> <th>\$ Invested</th> <th>\$ Projected</th> <th>% Rate of Return</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Arthur</td> <td>Platinum</td> <td>Long Island</td> <td>NY</td> <td>11101</td> <td>773-123-1111</td> <td>\$500,000.00</td> <td>\$527,751.15</td> <td>5.55%</td> </tr> <tr> <td>2</td> <td>Betsy</td> <td>Golden</td> <td>Arlington</td> <td>VA</td> <td>22202</td> <td>773-123-2222</td> <td>\$500,000.00</td> <td>\$527,568.75</td> <td>5.514%</td> </tr> <tr> <td>3</td> <td>Charlie</td> <td>Silver</td> <td>Fort Lauderdale</td> <td>FL</td> <td>33303</td> <td>773-123-3333</td> <td>\$500,000.00</td> <td>\$526,688.75</td> <td>5.338%</td> </tr> <tr> <td>4</td> <td>Denise</td> <td>Copper</td> <td>Burghill</td> <td>OH</td> <td>44404</td> <td>773-123-4444</td> <td>\$500,000.00</td> <td>\$510,625.00</td> <td>2.125%</td> </tr> </tbody> </table>	Cust #	First Name	Last Name	City	State	Zip	Work Phone	\$ Invested	\$ Projected	% Rate of Return	1	Arthur	Platinum	Long Island	NY	11101	773-123-1111	\$500,000.00	\$527,751.15	5.55%	2	Betsy	Golden	Arlington	VA	22202	773-123-2222	\$500,000.00	\$527,568.75	5.514%	3	Charlie	Silver	Fort Lauderdale	FL	33303	773-123-3333	\$500,000.00	\$526,688.75	5.338%	4	Denise	Copper	Burghill	OH	44404	773-123-4444	\$500,000.00	\$510,625.00	2.125%	<p><b>Customer Portfolios Performance</b></p>
Cust #	First Name	Last Name	City	State	Zip	Work Phone	\$ Invested	\$ Projected	% Rate of Return																																										
1	Arthur	Platinum	Long Island	NY	11101	773-123-1111	\$500,000.00	\$527,751.15	5.55%																																										
2	Betsy	Golden	Arlington	VA	22202	773-123-2222	\$500,000.00	\$527,568.75	5.514%																																										
3	Charlie	Silver	Fort Lauderdale	FL	33303	773-123-3333	\$500,000.00	\$526,688.75	5.338%																																										
4	Denise	Copper	Burghill	OH	44404	773-123-4444	\$500,000.00	\$510,625.00	2.125%																																										

**Banking > Finance > Portfolio Management**

Customer #

Analyze Customer Portfolio

Portfolio details for **Arthur Platinum** appear below:  
 Invested: \$500,000.00 | Projected: \$527,751.15 | Rate of return: 5.55%

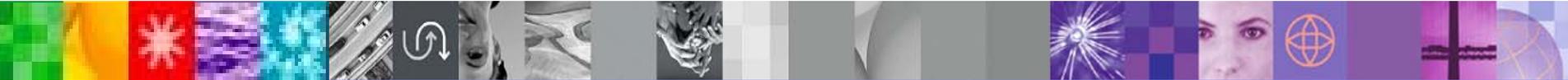
Portfolio	Allocation	Graphics	Market Simulation			
Fund Acct #	Fund Name	% of Folio	\$ Invested	Risk	% Return	\$ Projected
101	Blended - Income Plus Life Strategy	5%	\$25,000.00	1	2.124%	\$25,531.00
102	Blended - Conservative Life Strategy	4%	\$20,000.00	2	4.249%	\$20,849.80
103	Blended - Moderate Life Strategy	6%	\$30,000.00	3	6.374%	\$31,912.20
104	Blended - Aggressive Life Strategy	7%	\$35,000.00	4	8.424%	\$37,948.40
105	Money Market	5%	\$25,000.00	1	2.124%	\$25,531.00
106	Bond - Stable Value	3%	\$15,000.00	1	2.125%	\$15,318.75
107	Bond - Total Market	5%	\$25,000.00	2	6.375%	\$26,593.75



IBM Software Group

## ***Rational/WebSphere V6 Family and RAD/RSA Highlights***

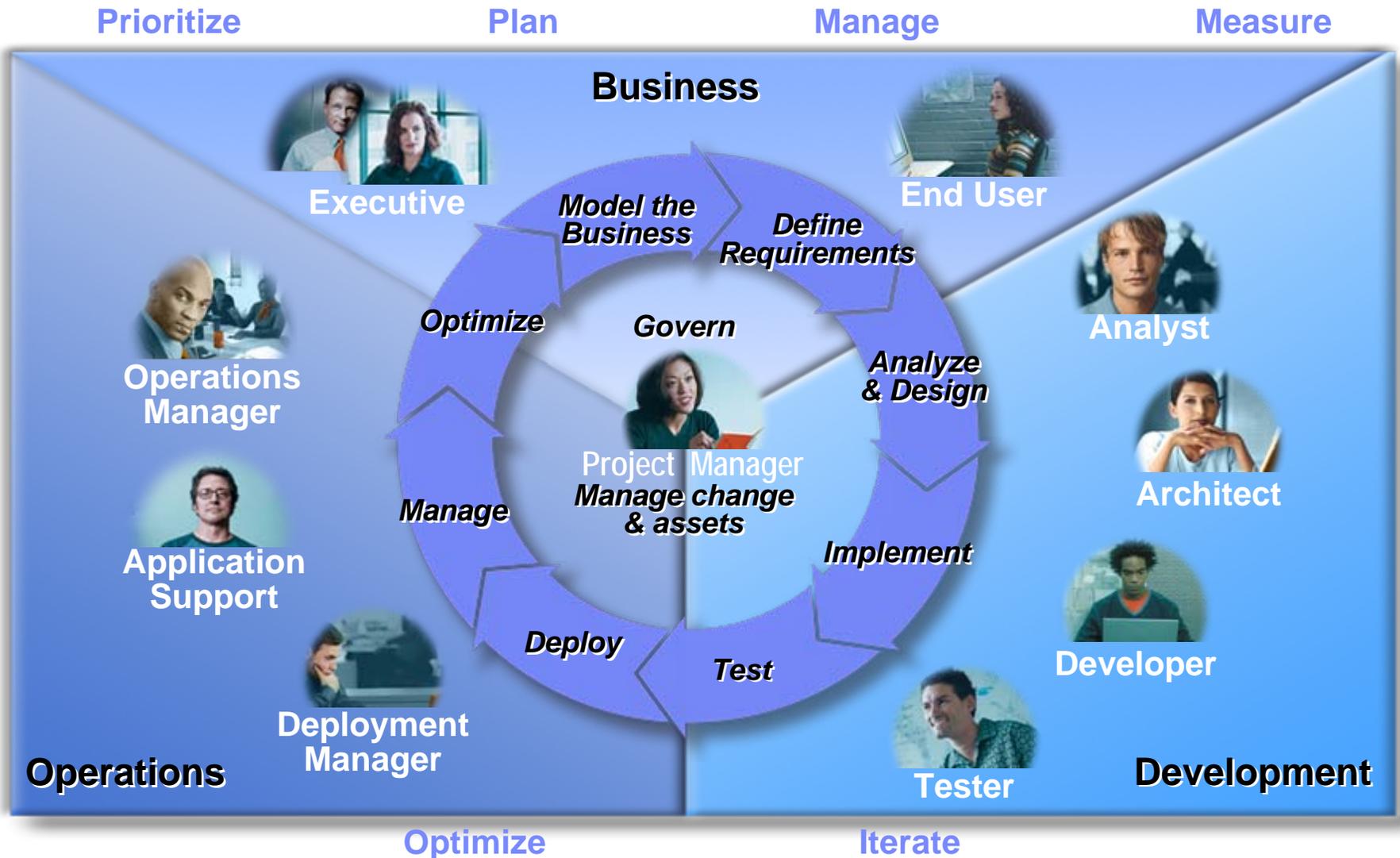
**➔ “The next generation of WSAD...IBM’s Foundational Eclipse based Tools” ←**



Name  
Title  
Company

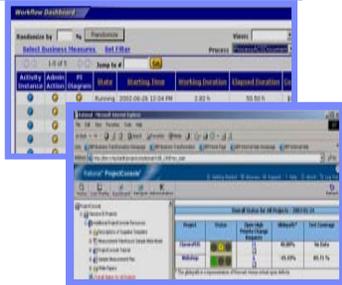
© 2003 IBM Corporation

# The business-driven development lifecycle

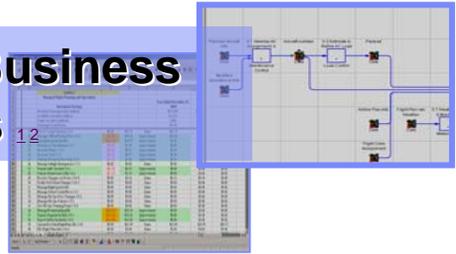


# Full Lifecycle Development *IBM Software Development Platform*

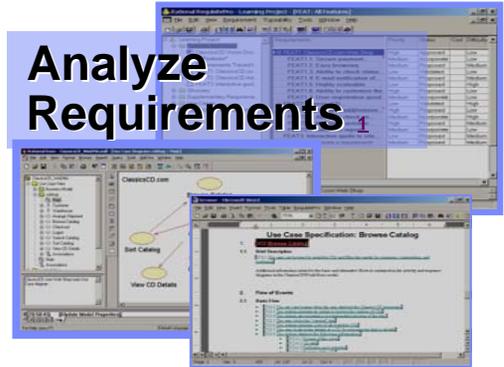
## Monitor



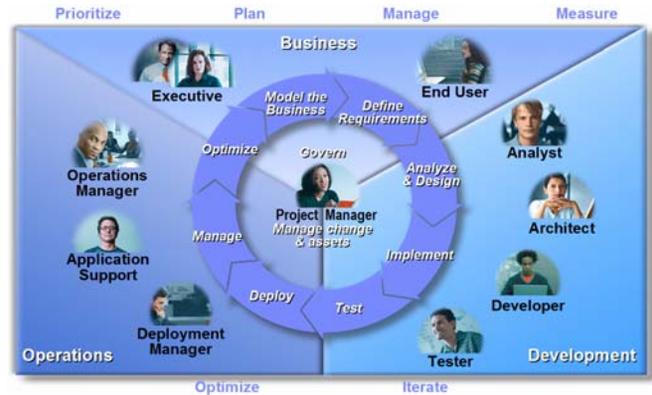
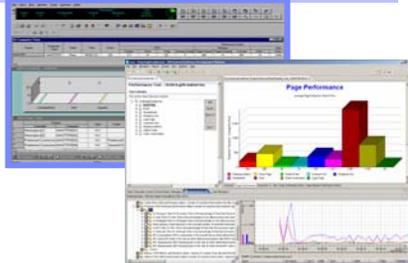
## Model Business Process



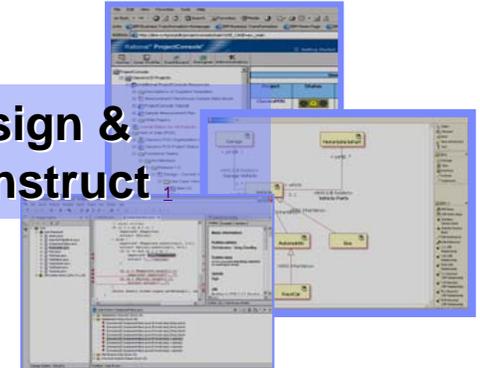
## Analyze Requirements



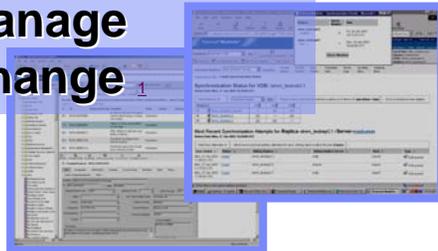
## Test & Deploy



## Design & Construct



## Manage Change



# Lifecycle Tools – RUP Development Disciplines

## Disciplines

Process Modeler Business Modeling

RSA (RequisitePro Views) Requirements

RSA/RAD Analysis & Design

RAD Implementation

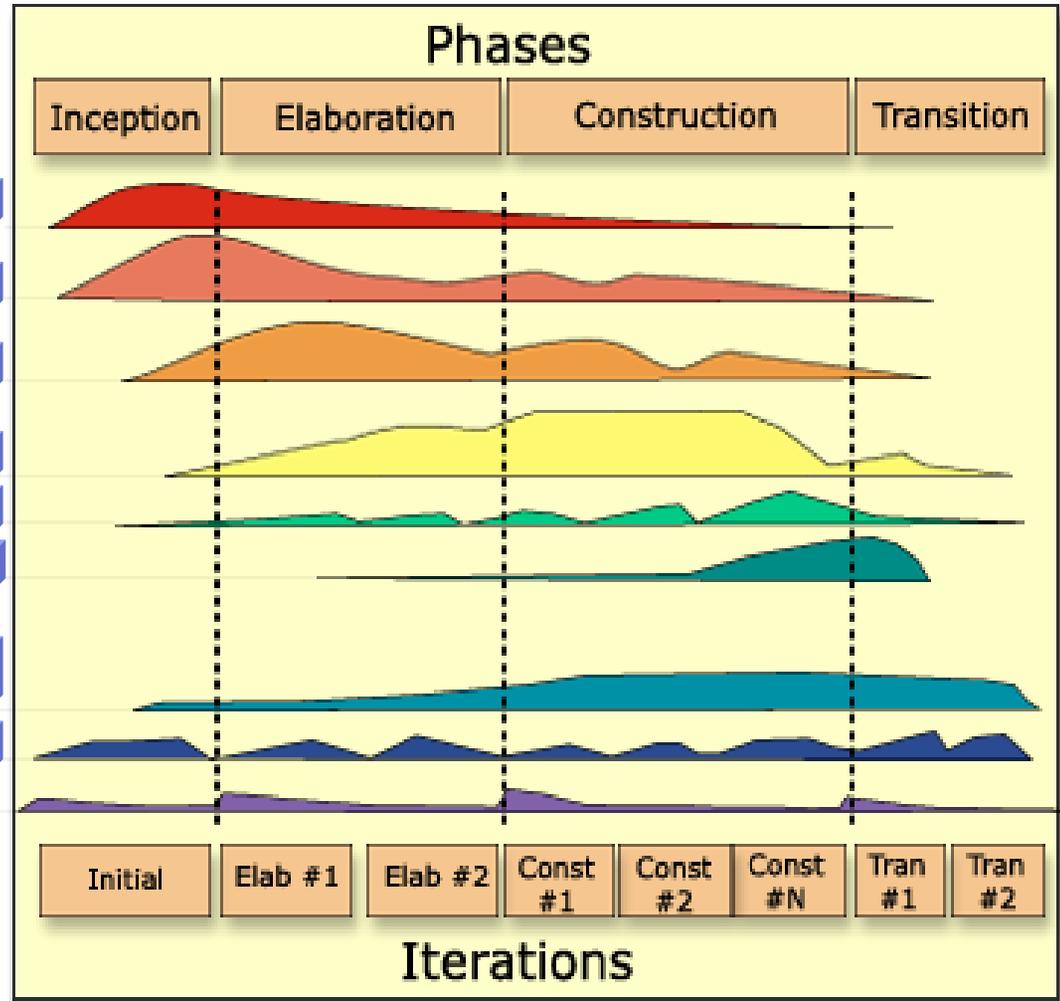
RAD Test

RAD Deployment

RAD (ClearCase Views + ClearQuest Views) Configuration & Change Mgmt

RAD (ClearQuest + Project Console) Project Management

Environment



## Simplified Packaging and Names

V4 & V5	V6
WSAD	RAD (Rational App Developer)
Rose XDE	RSM (Rational SW Modeler)
WSAD + XDE	RSA (Rational SW Architect)



# Existing Tools In WSAD V5.x and RAD V6

## Disciplines

Business Modeling  
Requirements

*UML 2-way Modeling* Analysis & Design

*J2EE, Web Services, JSF, SDO...* Implementation

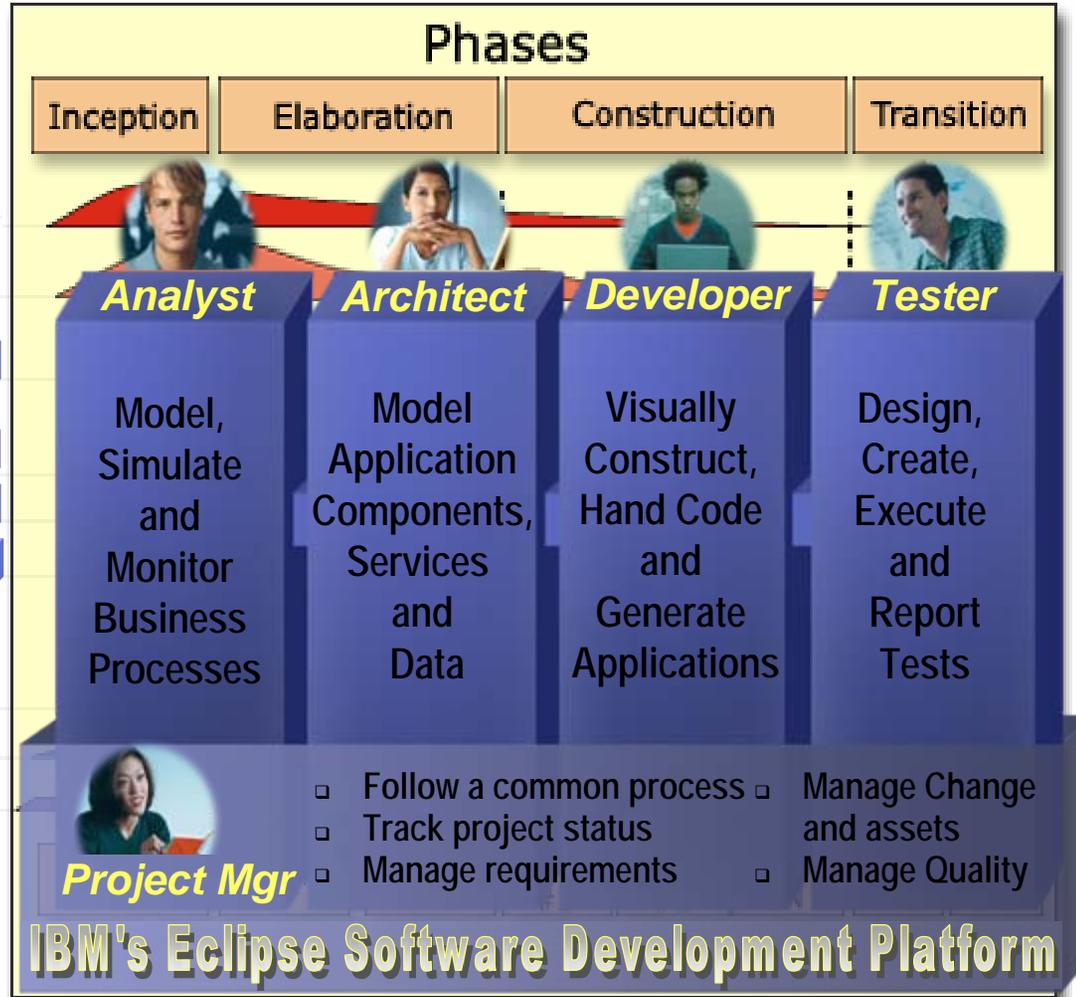
*Unit Testing, etc.* Test

*Automated J2EE Packaging* Deployment

Configuration & Change Mgmt  
Project Management  
Environment

### Simplified Packaging and Names

V4 & V5	V6
WSAD	RAD (Rational App Developer)
Rose XDE	RSM (Rational SW Modeler)
WSAD + XDE	RSA (Rational SW Architect)



# New V6 Integrated Views/Perspectives

## Disciplines

*Process Modeler* Business Modeling

*RequisitePro Eclipse Views* Requirements

Analysis & Design

Implementation

*Functional, Manual, Performance, Analysis* Test

Deployment

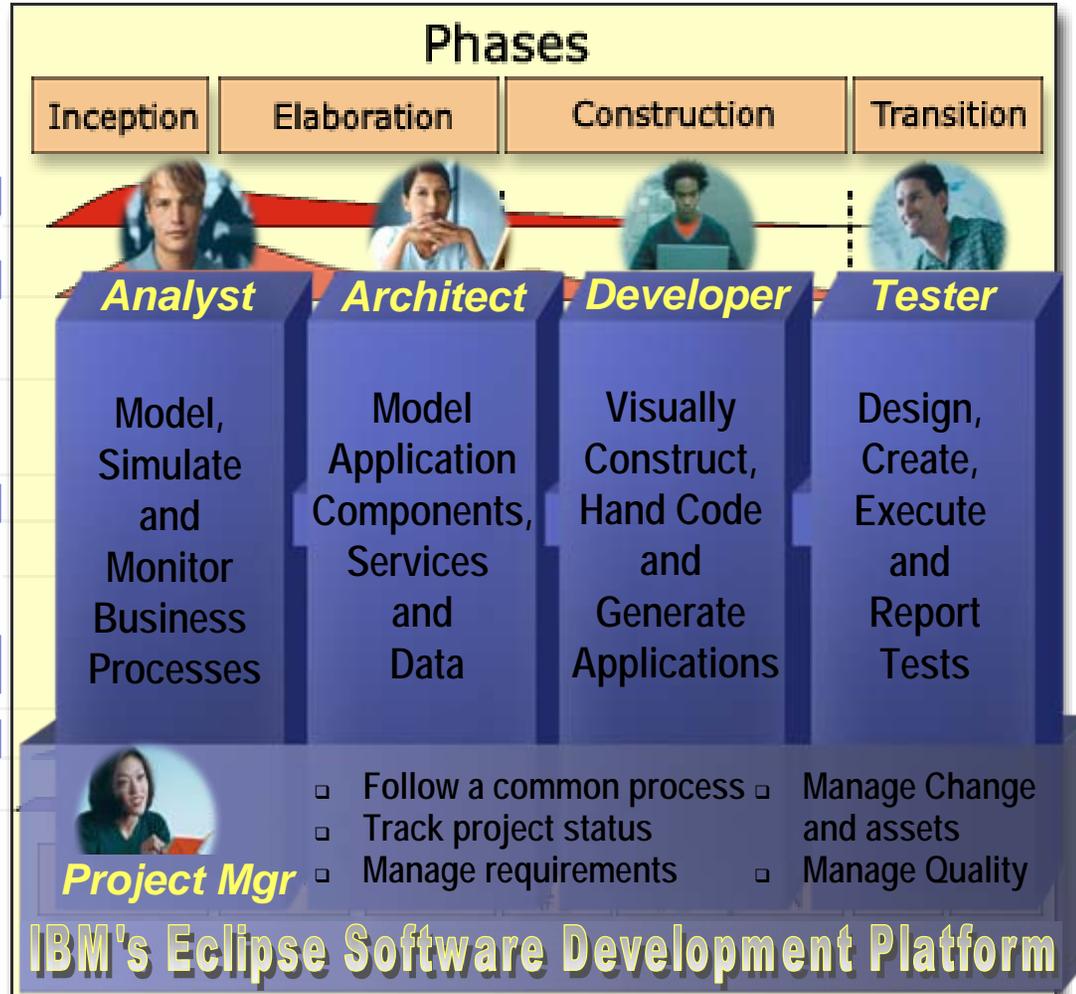
*ClearCase Eclipse Views* Configuration & Change Mgmt

*ClearQuest Eclipse Views* Project Management

Environment

### Simplified Packaging and Names

V4 & V5	V6
WSAD	RAD (Rational App Developer)
Rose XDE	RSM (Rational SW Modeler)
WSAD + XDE	RSA (Rational SW Architect)



# Lifecycle Tools are Integrated as Views in V6 <sup>1</sup>

- Requirement Management perspective
- ClearCase perspective
- ClearQuest perspective
- Functional Test perspective

## Disciplines

- Business Modeling
- Requirements
- Analysis & Design
- Implementation
- Test
- Deployment
- Configuration & Change Mgmt
- Project Management
- Environment

The screenshot displays the IBM Rational Software Development Platform interface. The main window shows a UML Use Case Diagram with an actor named 'Attendee' connected to five use cases: 'Register for Conference', 'Login to RSDUC site', 'Check weather', and 'Get directions to conference location'. The 'Requirement Explorer' on the left shows a tree structure for 'RSDUC UML Model' with 'Use Cases' expanded. The 'Properties' window at the bottom right shows details for a selected use case: Name: 'Use Cases', Type: 'Usecase', and Description: (empty).



# Rational V6 Tools – Product Packaging Overview



**System Analyst**



**Architect**



**Developer**



**Tester**

**Rational Software Architect**

**Rational Performance Tester**

**Rational Software Modeler**

**Rational Application Developer**

**Rational Functional Tester**

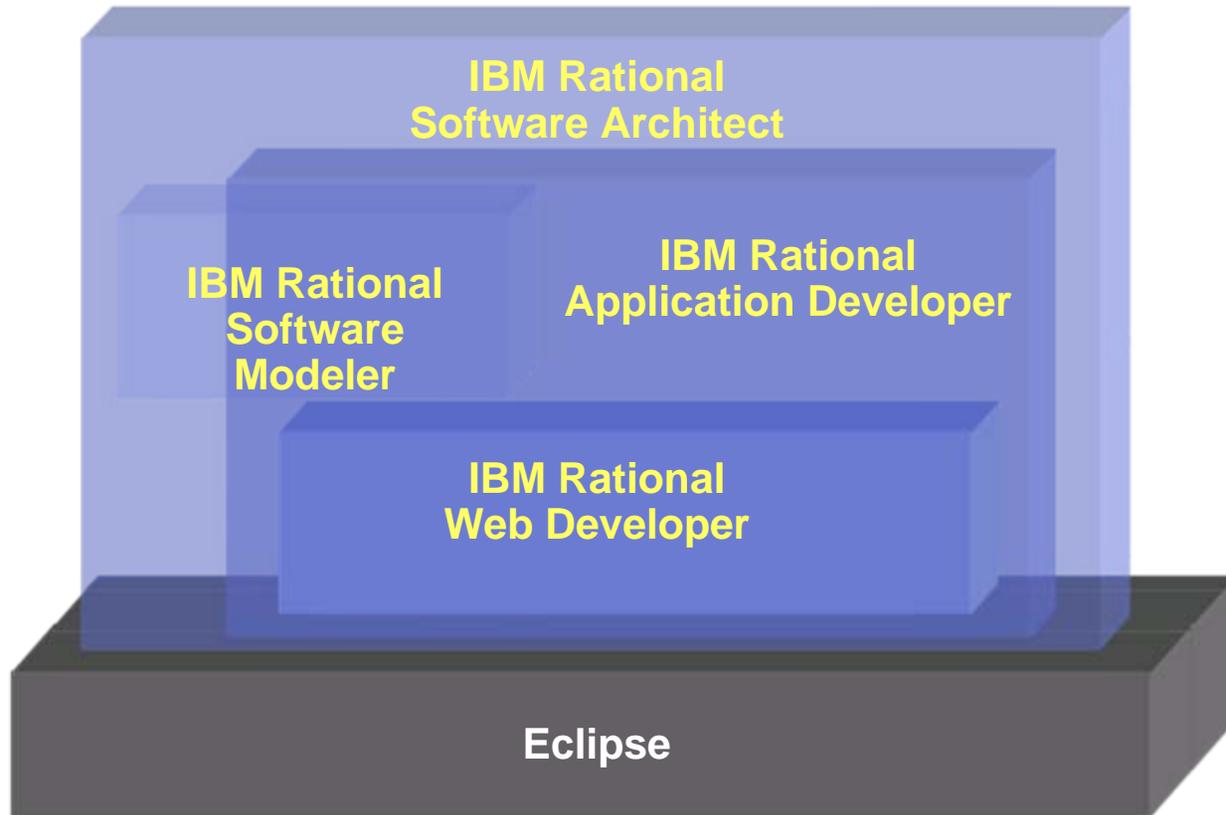
**Rational Web Developer**

**Rational Manual Tester**

**Eclipse 3.0**



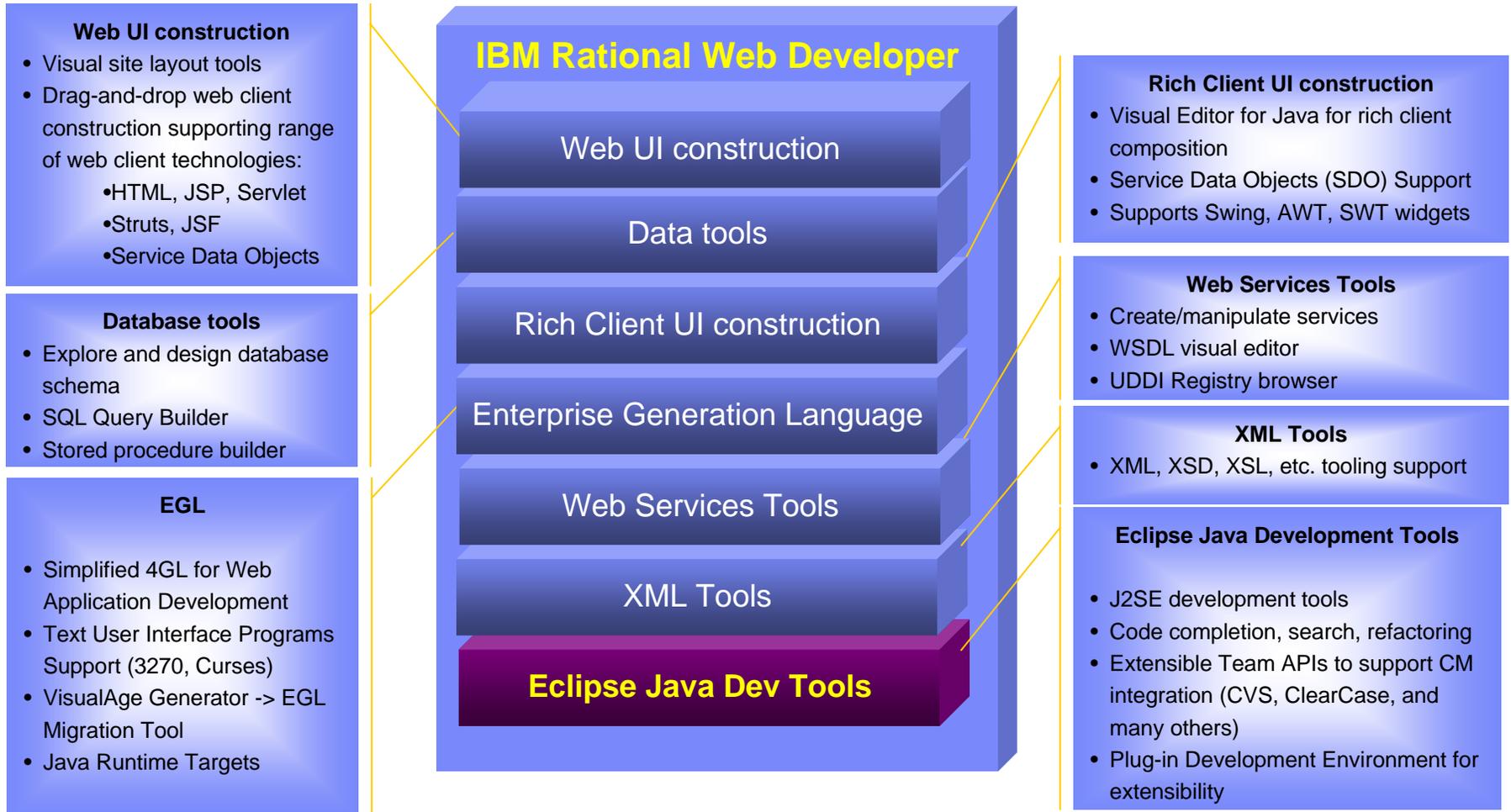
# Design and construction tools



# IBM Rational Web Developer

*Follow on to WebSphere Studio Site Developer*

*New Rapid Application Development capabilities focused on ease of use*



# IBM Rational Application Developer

Follow on to WebSphere Studio Application Developer

## J2EE/EJB Tooling

- J2EE construction tools
- Supports WebSphere and WebLogic
- Supports J2EE 1.3 and 1.4
- Rapid Deployment for WAS v6
- Integrated WAS test environments
- O/R mapping

## Code Analysis Tools

- Both static source code analysis and dynamic runtime analysis
- Source code analysis for rule violation detection: Globalization, J2EE/J2SE Best Practices, Private API Usage, Performance
- Sequence diagram runtime trace with performance, thread and code coverage annotation
- Advanced memory leak detection
- Remote data collection across multiple servers
- User-defined run-time probes
- Interactive reports and metrics

## IBM Rational Application Developer

Component/Unit Test

J2EE/EJB Tooling

Portal/Portlet Design Tools

Code Analysis Tools

UML Visual Editors

Rational Web Developer

RUP Configuration for J2EE

ClearCase LT

## Component/Unit Test

- JUnit-based testing of Java components, EJBs and Web Services
- Test prioritization recommendation based on code analysis
- Automated test case generation through usage of test patterns
- Datapool editor for data-driven testing
- Optimized unit testing through a set of metrics in the test creation wizard
- Test creation wizard offers a set of test patterns to automatically generate test

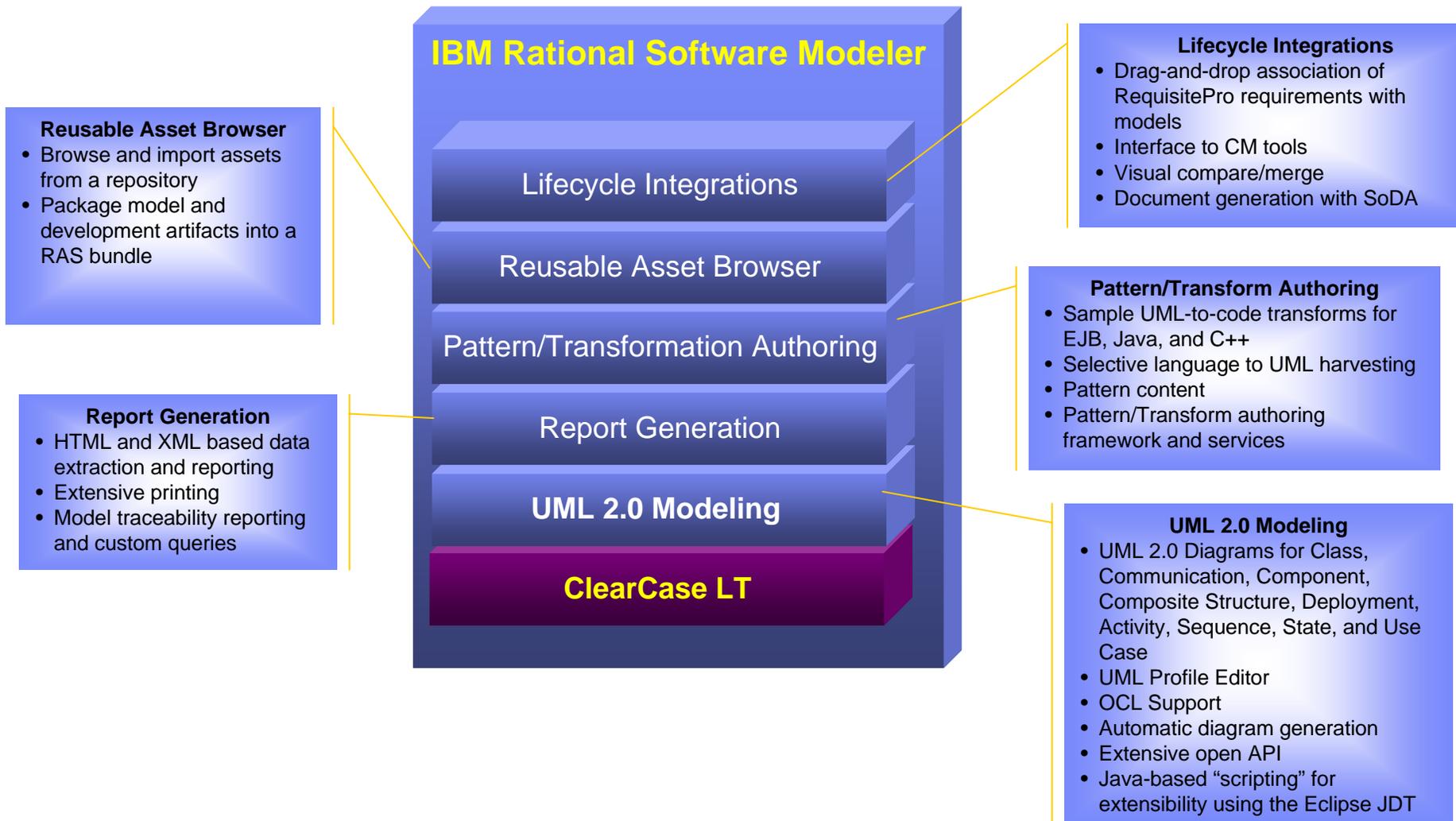
## Portal/Portlet Design Tools

- Visual Portal site and page layout tools
- Visual Portlet layout and preview
- Integrated Portal test environment

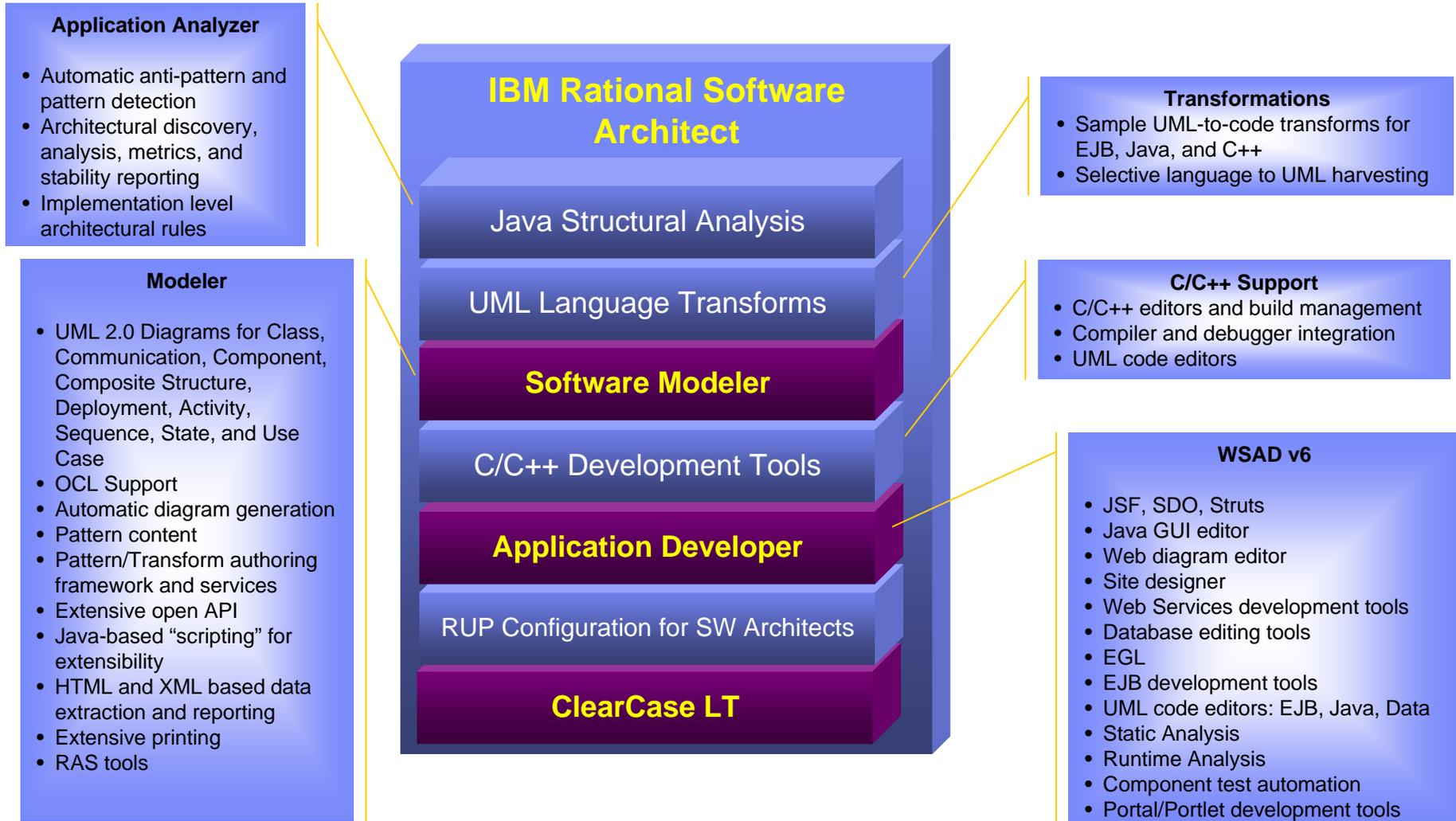
## UML Visual Editors

- Class diagrams for Java/EJB structure
- Sequence diagrams for method body visualization
- IDEF1X/IE diagrams for Database and XML schema
- Dynamic topic diagrams
- Javadoc integration
- Visual refactoring

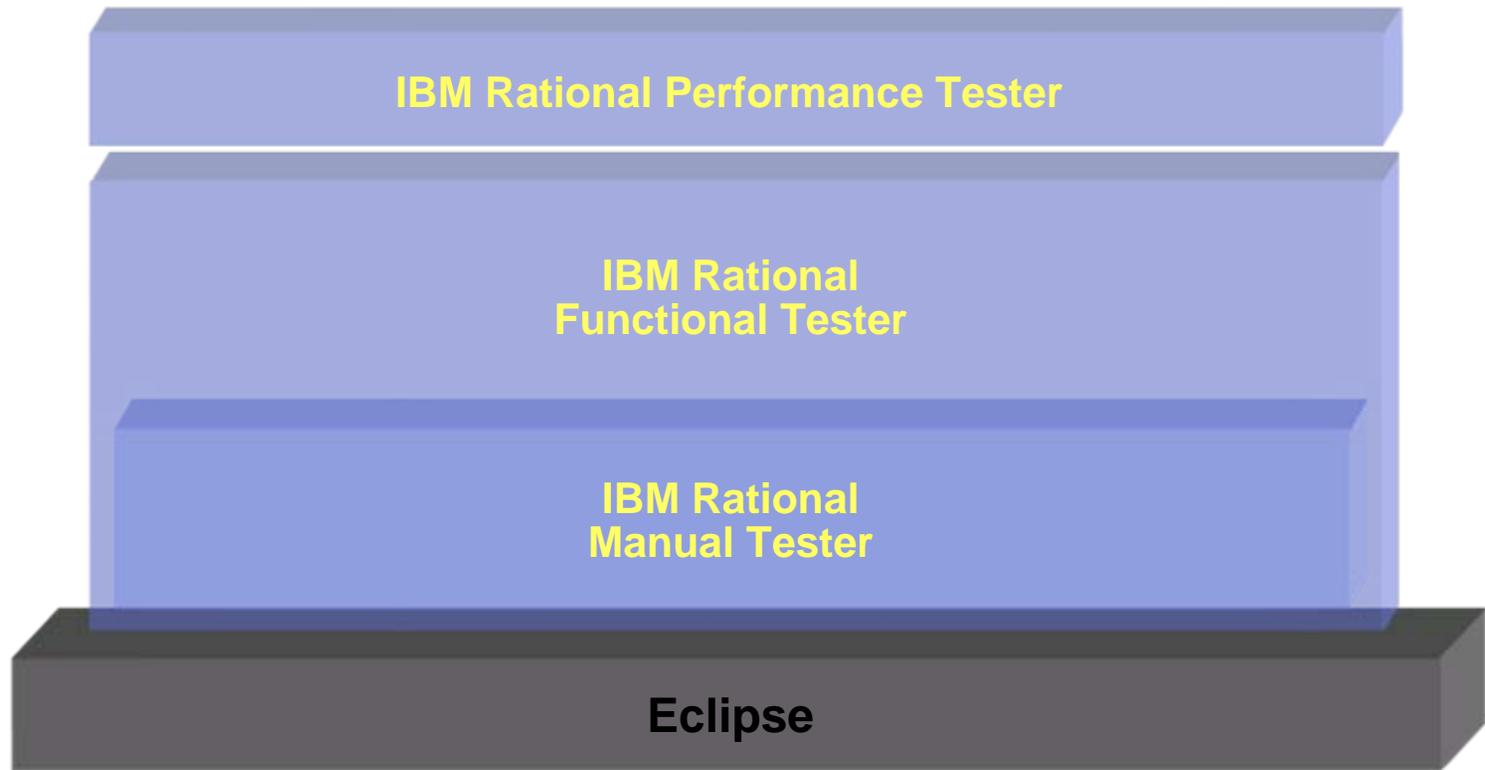
# IBM Rational Software Modeler



# IBM Rational Software Architect

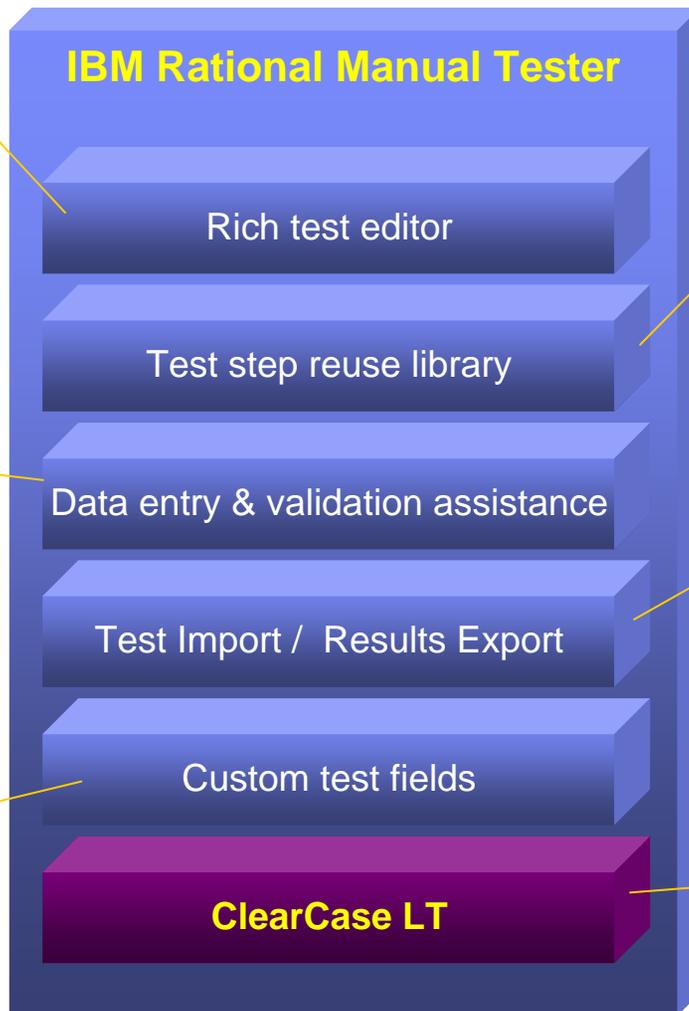


# Software quality tools



# IBM Rational Manual Tester

**Manual test authoring and execution**



- Rich Text Editor**
- Modify test font, size, color; attach images and files
  - Supports creation of both test steps and verification points
  - Ensures clarity of test step direction and consistency of test execution

- Tool Assistance**
- Automated data entry
  - Automated data comparison and results storage
  - Reduces opportunity for manual error during test execution

- Customizable**
- Add new data fields to individual test steps
  - Apply internally-defined naming conventions

- Reuse Library**
- Library for common test procedures
  - Drag-and-drop construction of new tests
  - Automatic global update when modifying any one instance of linked content

- Import/Export**
- Imports manual test source files from Microsoft Word, Microsoft Excel and Rational TestManager
  - Exports test results to CSV format for additional analysis

- ClearCaseLT "in the box"**
- Supports parallel development of manual tests
  - Enables remote test access for distributed teams



# IBM Rational Functional Tester

**Automated functional regression testing**

**GUI Technologies**

- Automates regression testing of Java, .NET, Web and terminal emulator-based applications

**Data-driven Testing**

- Simplified data-driven testing enabled by wizards and Excel-like data tables

**Eclipse availability**

- When using Java scripting, runs inside Eclipse/RSA/RAD shell
- Full Eclipse JDT features are accessible
- Accessible to developers from within their IDE

**ClearCaseLT "in the box"**

- Supports parallel development of manual tests
- Enables remote test access for distributed teams



**Test Script Language**

- User has choice of two test script languages – Java and VB.NET
- Use of Java implies editing and debugging in Eclipse/RAD/RSA-based editor
- Use of VB.NET implies usage of Microsoft VS.NET-based editor/debugger
- Identical capabilities in both editors

**Improved Linux support**

- Users can author and modify test scripts on Linux (no script recorder)
- Scripts can also be executed on Linux

**Manual Tester in the box**

- Customers receive one Manual Tester license per Functional Tester license
- Improve return on manual testing efforts while adopting automated test methods



# IBM Rational Performance Tester

**Load and performance testing**

**Load testing technology**

- New load test recorder and execution engine
- Supports HTTP/S; additional protocol support

**High scalability**

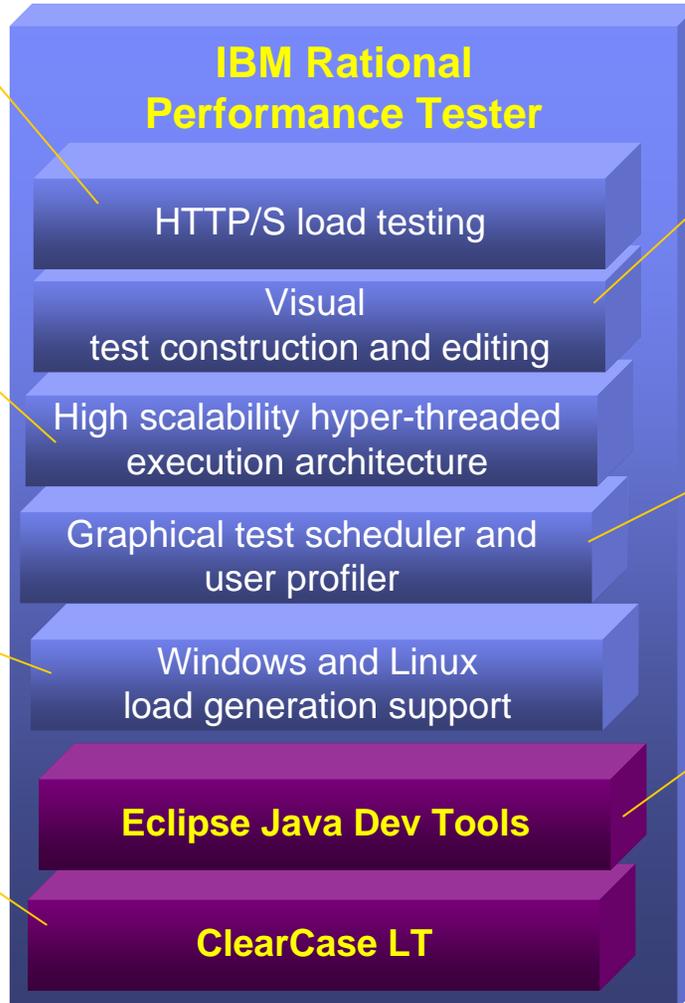
- High scalability, low footprint due to hyper-threaded architecture
- Significant improvement over Robot technology

**Platform Support**

- Multiple platform support for load generation

**ClearCaseLT “in the box”**

- Supports parallel development of manual tests
- Enables remote test access for distributed teams



**Visual Test Editor**

- Test viewer requires no coding for test modification and execution
- Automated identification and variation of dynamic data
- Insertion of Java code is supported for advanced customization needs

**Test scheduler**

- Graphical user profiling and transaction modeling
- Eases modeling of user populations

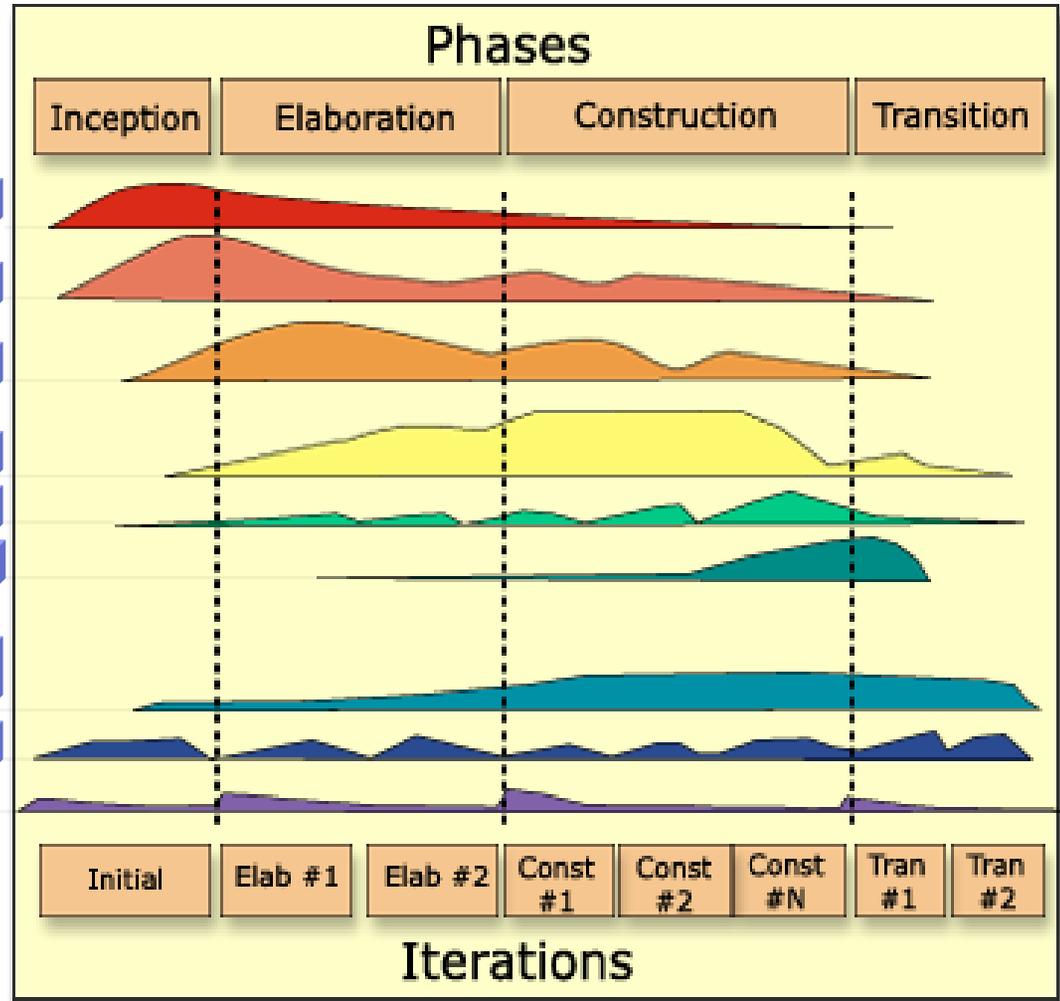
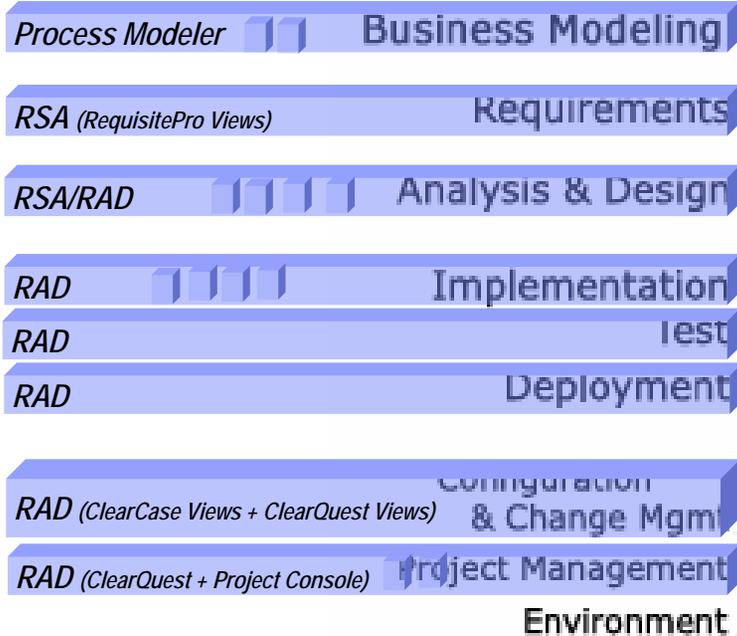
**Eclipse-based**

- Runs inside Eclipse/RSA/RAD shell
- Accessible to developers from within their IDE



# Lifecycle Tools – RUP Development Disciplines

## Disciplines



Simplified Packaging and Names	
V4 & V5	V6
WSAD	RAD (Rational App Developer)
Rose XDE	RSM (Rational SW Modeler)
WSAD + XDE	RSA (Rational SW Architect)



# RAD Top 10 of IBM AD Tools (Adaptable, Automated, Accelerated)

## **Adaptable** - Eclipse-based Workbench

**#1** Customizable **Perspectives** and **Views** support **roles-based** tools

## **Automated** - IDE Infrastructure (Code, Build, Test, Debug, Deploy)

**#2** **Dynamic build engine** incrementally updates and hot-deploys (on-the-fly, while you type)

**#3** **Testing and Debugging** for Java, Web Services, XML, database and EJB

## **Accelerated** - End-2-End Builders

**#4** **Java and UML** core coding tools

**#5** **Web App Builders**: Pages, Sites, Portals, Portlets, Struts, JavaServer Faces

**#6** **Web Services** and **XML** Builders

**#7** **Database and SDO** (Service Data Objects) Builders

**#8** **J2EE** and **EJB** Builders

**#9** **SOA workflow** builders (Service Oriented Architecture)

**#10** **Legacy/Enterprise** development, integration and Web Services tools



# RAD Top 10 of IBM AD Tools (Adaptable, Automated, Accelerated)

## Adaptable - Eclipse-based Workbench

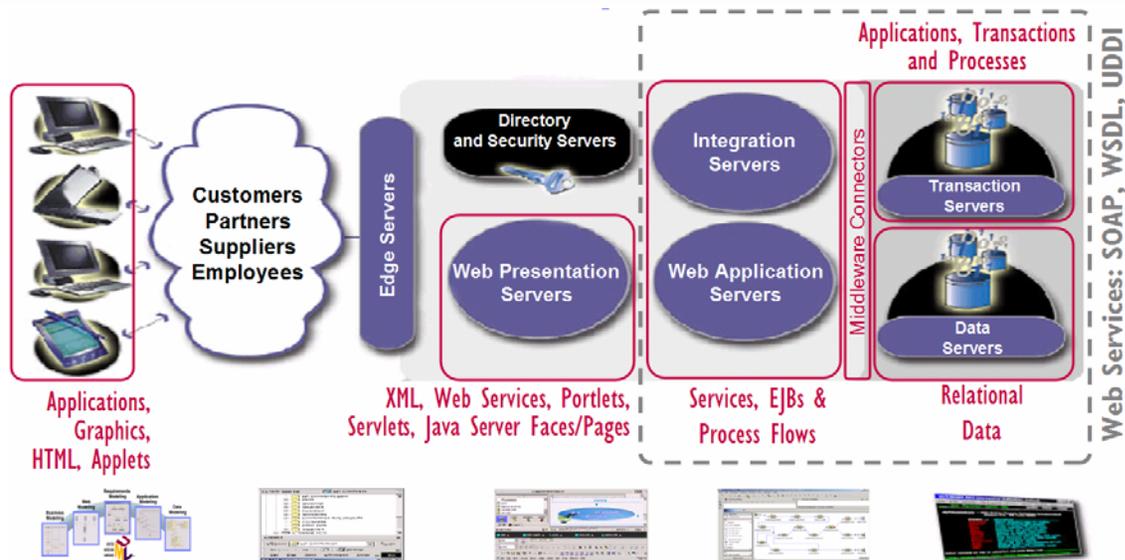
#1 Customizable **Perspectives** and **Views** support **roles-based** tools

## Automated - IDE Infrastructure (Code, Build, Test, Debug, Deploy)

#2 **Dynamic build engine** incrementally updates and hot-deploys (on-the-fly, while you type)

#3 **Testing and Debugging** for Java, Web Services, XML, database and EJB

## Accelerated - End-2-End Builders



# RAD Top 10 of IBM AD Tools (Adaptable, Automated, Accelerated)

## **Adaptable** - Eclipse-based Workbench

**#1** Customizable **Perspectives** and **Views** support **roles-based** tools

## **Automated** - IDE Infrastructure (Code, Build, Test, Debug, Deploy)

**#2** **Dynamic build engine** incrementally updates and hot-deploys (on-the-fly, while you type)

**#3** **Testing and Debugging** for Java, Web Services, XML, database and EJB

## **Accelerated** - End-2-End Builders

**#4** **Java and UML** core coding tools □□

**#5** **Web App Builders**: Pages, Sites, Portals, Portlets, Struts, JavaServer Faces □□□□□

**#6** **Web Services and XML** Builders □□□

**#7** **Database and SDO** (Service Data Objects) Builders □

**#8** **J2EE and EJB** Builders □

**#9** **SOA workflow** builders (Service Oriented Architecture) □□

**#10** **Legacy/Enterprise** development, integration and Web Services tools □□□





# J2EE Development *Automated by JSF*

Input → Process → Output

## • Input

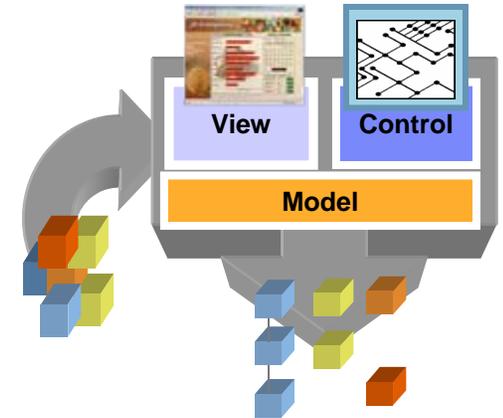
- Constructing web page forms, formatting, validation, etc.
- GUI widget richness
- Calling appropriate "Action" when button pressed, etc.
- Passing data to business logic

## • Processing

- Integrating to web services, databases, backends, biz logic, etc.
- Maintaining user session and application state information
- MVC layering, page flow and business logic modularity

## • Output

- Forwarding control to results screen/page
- Displaying data, GUI controls, formatting, etc.

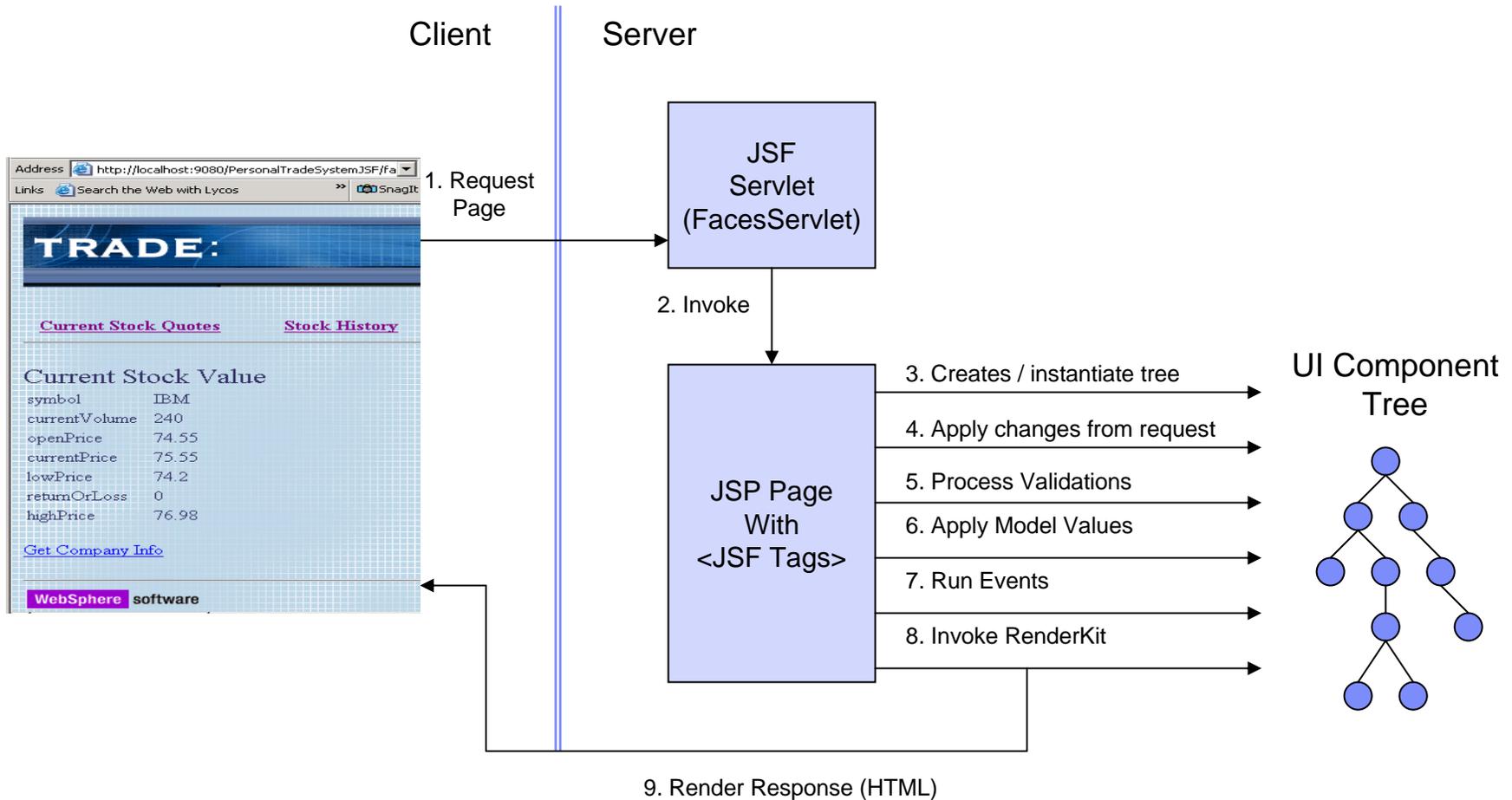


# JavaServer Faces (JSR 127) – What Does it Provide?

- **Based on MVC design pattern**
  - Clean separation of Model, View, and Controller
- **User Interface Component Model – Set of standard widgets**
  - Specification allows extending custom GUI components (i.e. DataTable, Graphing, Tree, Panel)
- **User Interface components are decoupled from its rendering**
  - Allows for other technologies and pervasive devices (e.g. WML, etc) to be used
- **Validation Framework allows server-side validation**
- **Server side management of User Interface, session and state info**
- **Event driven architecture**
  - Server-side rich user interface components respond to client events
- **Actions organize and process business logic**
- **Navigation model manages page flow using JSP path aliases**



# Java Server Faces- How it works



# Application Developer V6 JSF Design Points

- **Accelerate GUI, Web Services, Database, Portlet and MVC development**
  - Visual drag-n-drop development of UI, events, actions, page navigation, etc.
- **Simplify J2EE development for non-Java programmers**
- **Enhance Web Application Development for J2EE experts**
- **Appeal to Microsoft developers (VB/Visual Studio .Net)**
- **Reduce development and deployment time and cost**
- **Standards based technologies**
  - JavaServer Faces - JSR 127
  - Service Data Objects - JSR 235
  - Portlets – JSR 168



# JavaServer Faces Tools in RAD V6

The screenshot displays the IBM Rational Application Developer (RAD) V6 interface in the Web Perspective. The main workspace shows a JSP page titled "home.jsp" with a design view. The page content includes a header "THE CLASSIFIED E-ADS", a navigation bar with "Home", "Classified", and "Post an Ad" links, a "Welcome {logonName}" message, and a "Sign in" form with "Log on" and "Password" fields. A "NEW! Latest Ads" section is also visible.

Several toolbars and panels are highlighted with blue callouts:

- Web Perspective**: Located at the top center of the IDE window.
- Project Navigator**: Located on the left side, showing a tree view of the project files including admin.jsp, catdetails.jsp, categories.jsp, details.jsp, error.jsp, and home.jsp.
- Page Data**: Located below the Project Navigator, showing a tree view of the page's data sources like latestAds (ADS), news (NEWS), login (beans.Login), and logonPassword (java.lang.Str).
- Attributes**: Located below the Page Data, showing the configuration for the selected "HX::COMMAND\_BUTTONEX" component, including its ID (logonButton) and action (Submit/Reset).
- Page Designer**: The central workspace where the JSP page is visually designed.
- Event Panel**: Located at the bottom center, showing the event listener code for the "onclick" event: `gotoPage ("home.jsp");` and `return null;`.
- Palette**: Located on the right side, providing a library of components and tags such as HTML Tags, JSP Tags, Faces Components (Beta), Command - Button, Command - Hyperlink, Hyperlink, Image, Label, Input, Radio Button Group, Check Box, Check Box Group, List Box, Combo Box, Panel - Group Box, Panels - Tabbed, Panel - Menu Bar, Output, Display Error, Horizontal Rule, Data Grid, Media Player, Page Template, Data (Beta), Web Site Navigation, and Snippets.



# Agenda

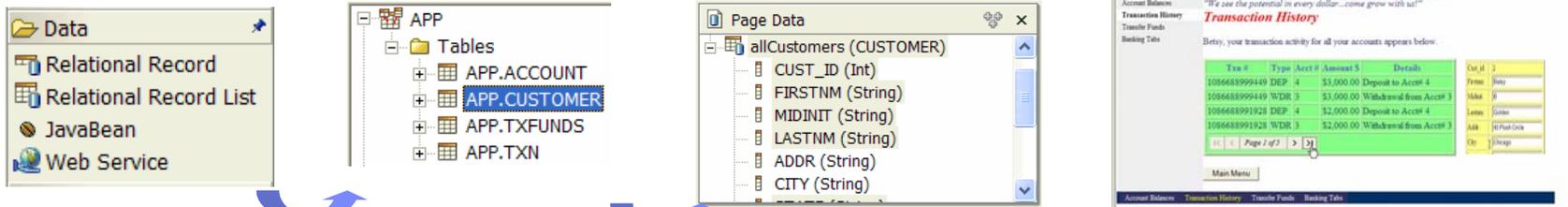
- **JSF in Action**

- Web Services Creation and Consumption
- Service Data Objects (SDO) Heterogeneous data
- MVC Web App Rich Thin Clients
- Portlet Apps and Portal Integration

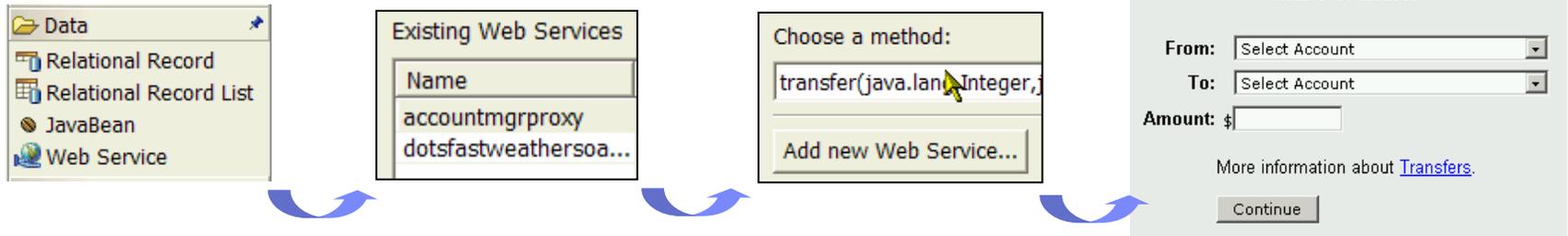


# Simplifying J2EE Development

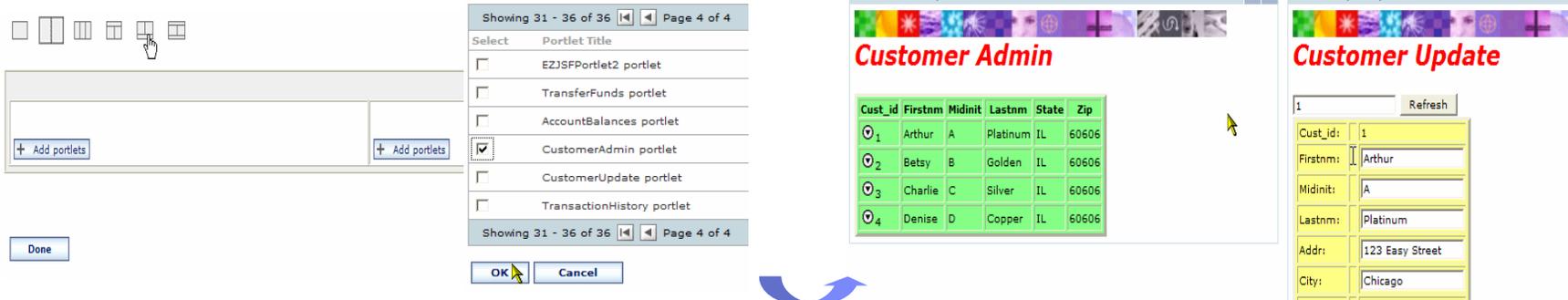
- Database development drag-n-drop ease-of-use <sup>1 2</sup>



- Web Services development drag-n-drop ease-of-use <sup>1 2</sup>

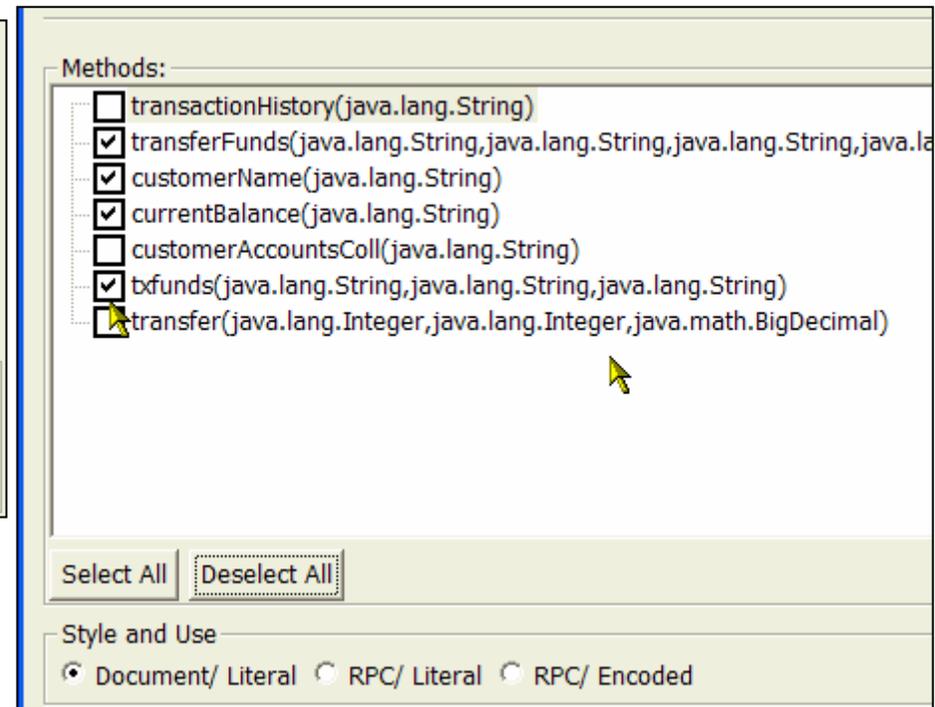
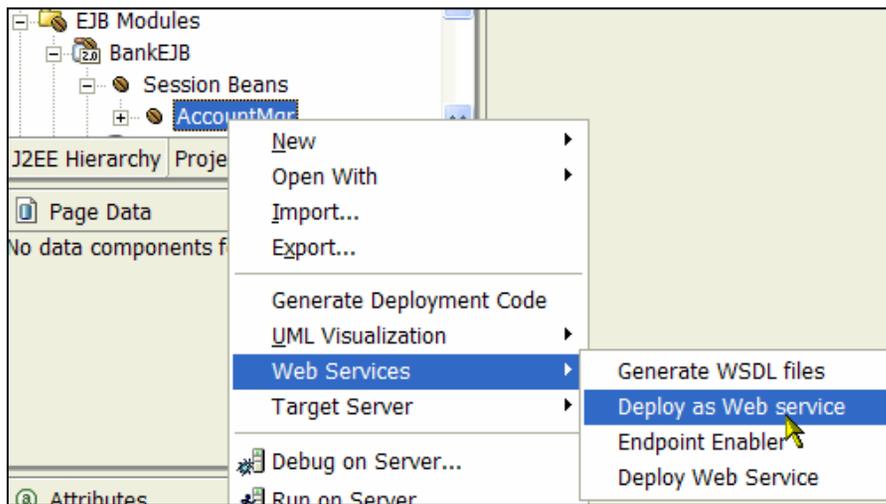


- Portals drag-n-drop ease-of-use <sup>1 2</sup>



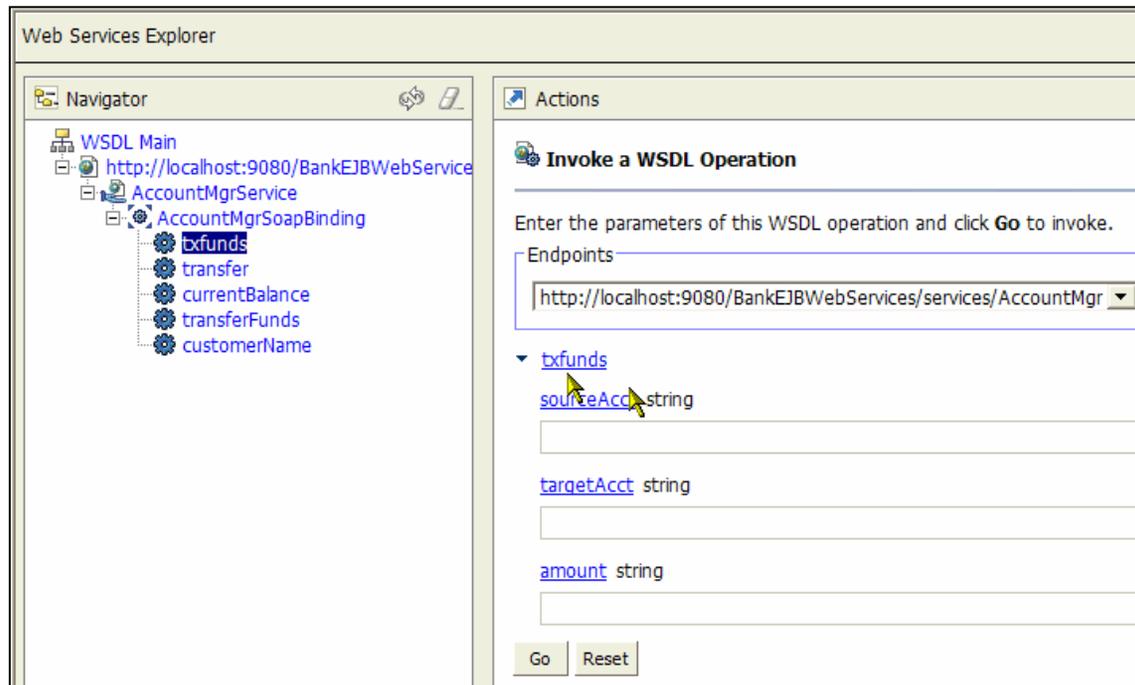
# Creating New Web Services

- **Generates all web services artifacts** <sup>12</sup>
  - Easily choose Java Class/EJB and methods to expose
  - WSDL service definition file for chosen Class/EJB and methods
  - Other infrastructure: Deployment descriptor, JAX/RPC request/response data definitions



# Consuming Existing Web Services – Discover and Test

- Discovery tools use UDDI, a well known URL or a WSDL file <sup>1</sup>
- Web Services Explorer automates discovery and testing
  - Launch separately or while building Faces JSPs
  - Browse Web Service operations, request input parameters, response output
  - Great general purpose unit testing environment



# Consuming Existing Web Services – Calling From Application

## • JSF Web Services Wizards Automate Construction <sup>1</sup>

- Creates web service proxy bean
- Creates Action method that calls proxy
- Creates input parameter bean and output results bean
- Drag-n-drop to create UI controls
- Binds input, output and action to UI controls

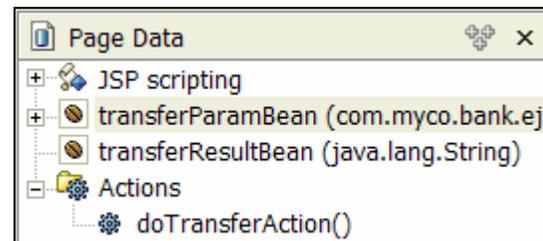
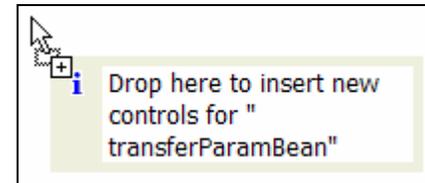
SourceAcct:	{sourceAcct}	123
TargetAcct:	{targetAcct}	123
Amount:	{amount}	123
* {Error Messages} <sup>BR</sup>		
Submit		

Name	Class
accountmgrproxy	com.myco.bank.ejb.AccountMgr...
dotsfastweathersoa...	com.serviceobjects.www.DOTS...

Choose a method:

transfer(java.lang.Integer,java.lang.Integer,java.math.BigDecimal)

Add new Web Service...



# Web Services Ease-of-Use Summary

## • External Web Services <sup>1</sup>

- Web Services Explorer
  - Great for unit testing and experimenting
  - Launched separately or while building a new Faces JSP
- Adding a Web Service to a project
  - Proxy created that knows how to call the Web Service
  - Input Parameters Bean
  - Results Bean
  - Action method created that uses the proxy bean
- Drag-n-drop is used to create the UI
- Once a Web Service is added to a project, any Faces JSP can use it

## • Local Web Services <sup>2</sup>

- Easily explored using Web Services Explorer
- Easily added from local workspace



# Agenda

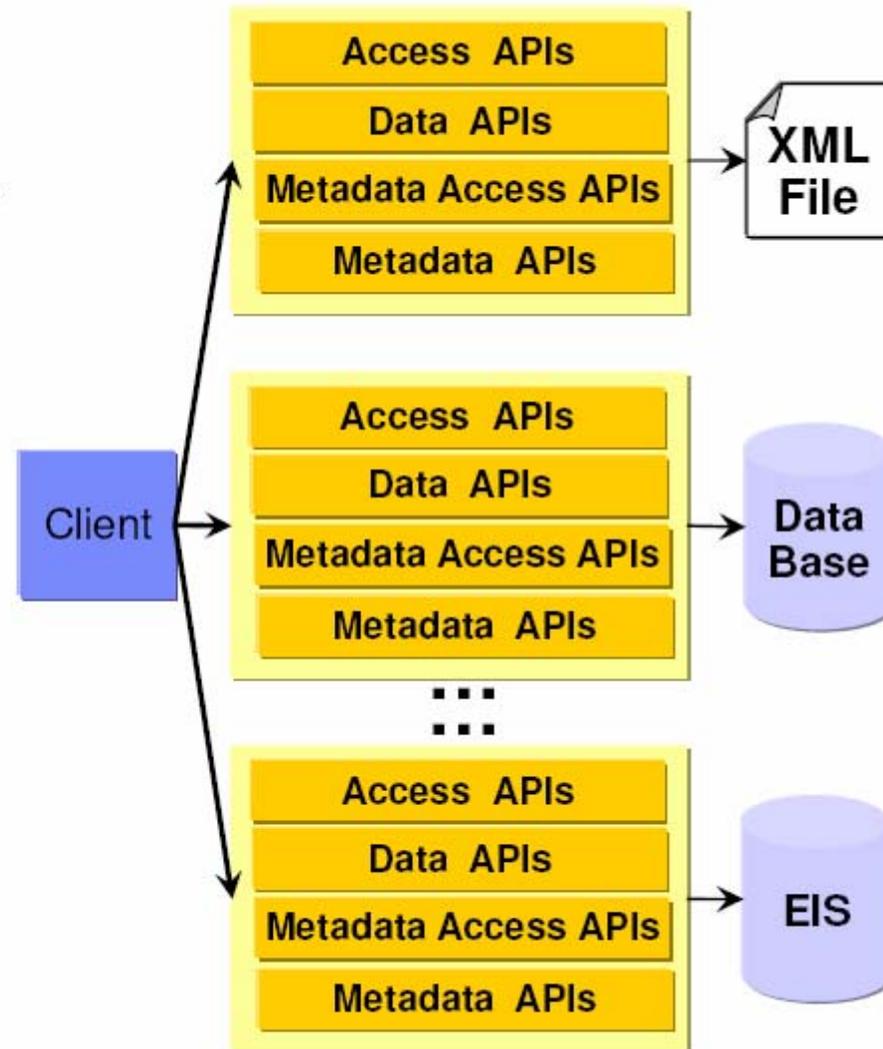
- **JSF in Action**

- Web Services Creation and Consumption
- Service Data Objects (SDO) Heterogeneous data
- MVC Web App Rich Thin Clients
- Portlet Apps and Portal Integration



# Current Data Access Challenges

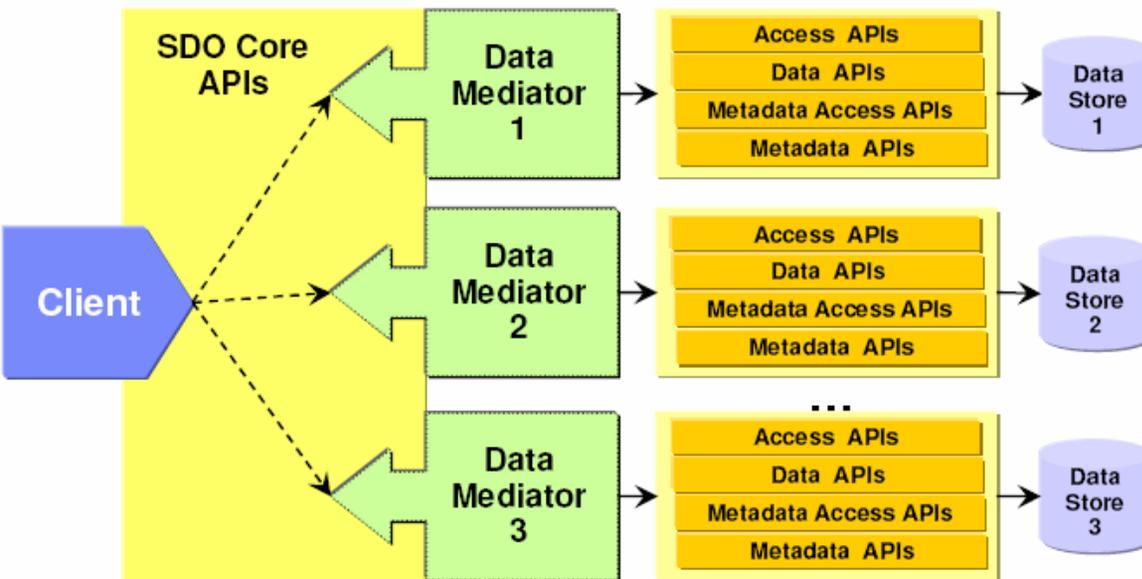
- Many different models/APIs for Data and Metadata retrieval and representation
  - ▶ Relational Databases (JDBC), XML files, JMS, Web Services (JAX-RPC), Enterprise Information Systems (EIS)
- Lack of support for standard application patterns
  - ▶ Optimistic concurrency, pagination of large data-sets
  - ▶ Asynchronous invocation
  - ▶ Stateful services
- Consequences:
  - ▶ Programmers forced to focus on the technology, rather than the business problem
  - ▶ Requires programmers to do a lot of low-level coding
  - ▶ Tools do not provide a higher-level task orientation data abstraction



# Current Data Access Challenges

## Solution → Service Data Objects (SDO)

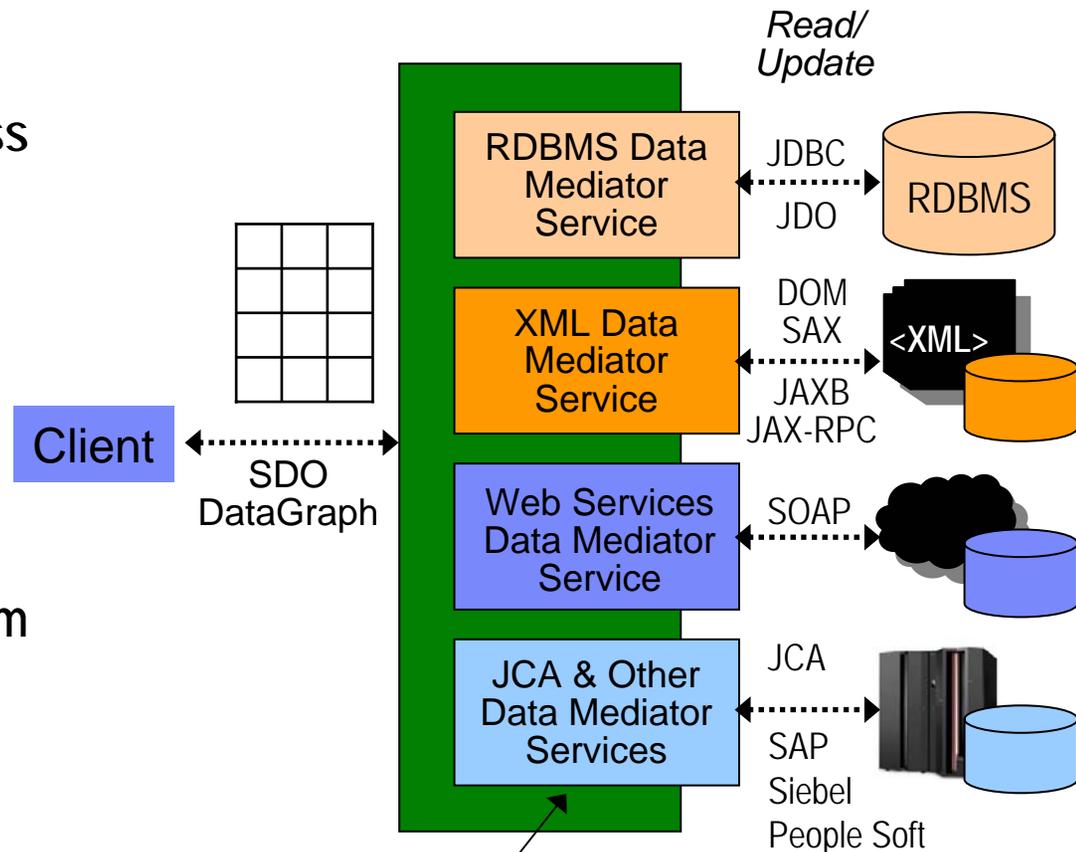
- Unified data representation & retrieval across heterogeneous data sources in a disconnected, source-independent format
- Exploitable by tools to provide simple application development experience
- Support of XML typed data
- Support for dynamic and statically type data



- Supported Mediators:
  - ▶ Relational database
  - ▶ EJB
- Future
  - ▶ Web Services
  - ▶ XML
- RAD V6.x drag-n-drop tooling today
  - Relational
  - EJB
  - Web Services
  - JCA
  - SAP, Siebel, People Soft

# Service Data Objects (SDO): Unified Access to Heterogeneous Data Sources

- Provides a common programming interface to access data from multiple heterogeneous sources
- Supports both static and dynamic data APIs
- Supports disconnected operation
- Decouples application code from data access code
- Designed for use with tools to increase programmer productivity

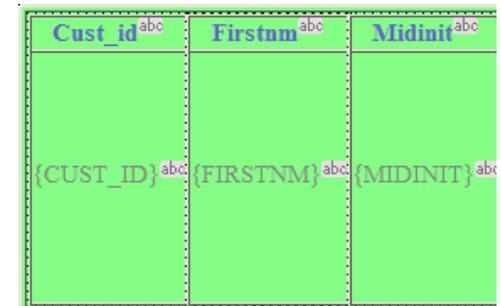
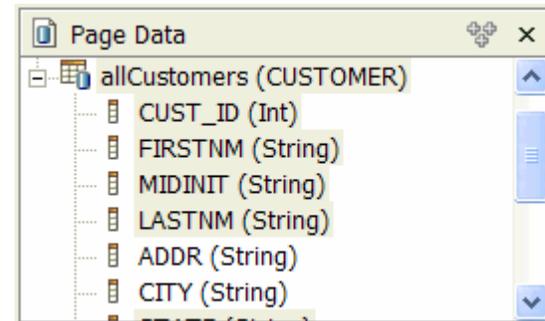
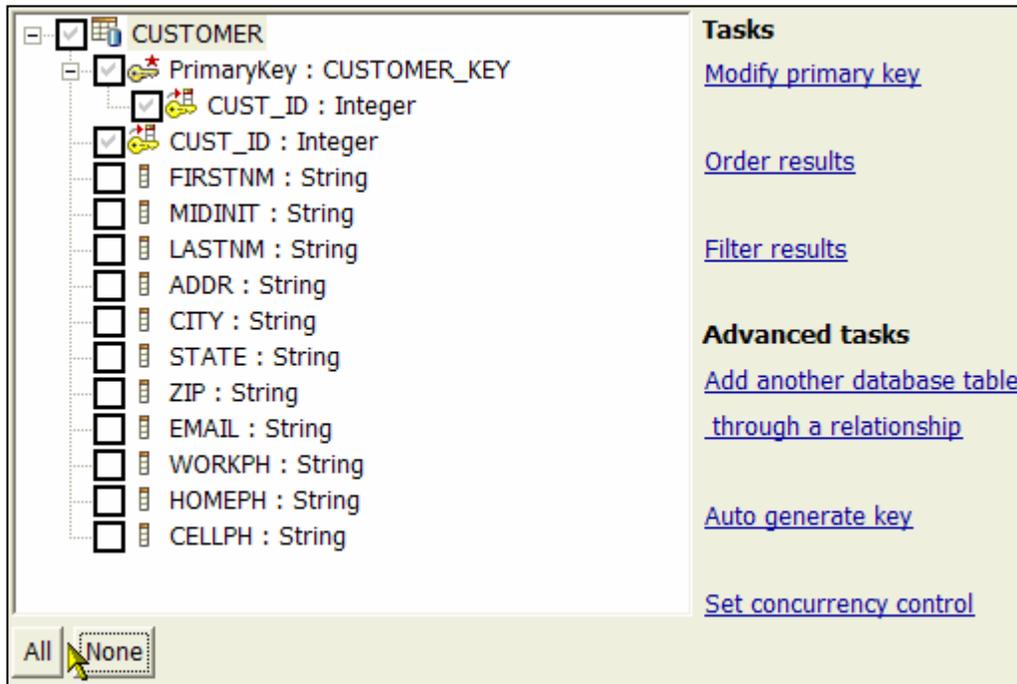


*Query data sources, create data graphs containing data objects, apply changes back to the data source*

Service Data Objects (SDO) is a specification created by IBM & BEA, submitted to JCP as JSR 235 in December 2003. Expert Group Formation Stage, not part of J2EE yet

# Creating Database Apps – Select “Relational Record List”

- **Relational wizards automate Select queries** <sup>12</sup>
  - Connect to any relational database, select table(s) and columns graphically
  - Construct query without knowledge of SQL, including input parameters and conditions
  - Drag-n-drop to create UI controls for data display (DataTable)
  - Binds input, output and action to UI controls
  - Format data display for numbers, currency, dates, text, masks, etc.



# Creating Database Apps – Insert, Update, Delete Rows “Relational Record”

- **Relational wizards automate Update queries** <sup>12</sup>
  - Connect to any relational database, select table(s) and columns graphically
  - Construct query without knowledge of SQL, including input parameters and conditions
  - Drag-n-drop to create UI controls for data display (Form)
  - Binds input, output and action to UI controls
  - Format data input/output display for numbers, currency, dates, text, masks, etc.

Create controls for:

Displaying an existing record (read-only)

Updating an existing record

---

Fields to display:

Field Name	Label	Control Type
<input checked="" type="checkbox"/> CUST_ID (int)	Cust_id:	Output Field
<input checked="" type="checkbox"/> FIRSTNM (java.lang.String)	Firstnm:	Input Field
<input checked="" type="checkbox"/> MIDINIT (java.lang.String)	Midinit:	Input Field
<input checked="" type="checkbox"/> LASTNM (java.lang.String)	Lastnm:	Input Field
<input checked="" type="checkbox"/> ADDR (java.lang.String)	Addr:	Input Field
<input checked="" type="checkbox"/> CITY (java.lang.String)	City:	Input Field
<input checked="" type="checkbox"/> STATE (java.lang.String)	State:	Input Field

All None Options...

Cust_id:	{CUST_ID} abc
Firstnm:	{FIRSTNM} abc
Midinit:	{MIDINIT} abc
Lastnm:	{LASTNM} abc
Addr:	{ADDR} abc
City:	{CITY} abc
State:	{STATE} abc
Zip:	{ZIP} abc

# Relational Data Ease-of-Use Summary

- **Service Data Objects (SDO)** <sup>12</sup>
  - Records and Record Lists abstract data access
  - Build a query without knowledge of SQL, JDBC, etc.
  - Select, Insert, Update, Delete
  - Drag-n-drop tables and forms of data onto UI
  - UI data display is automatically bound to Record/List queries
  - Supports conditions, relationships, optimistic concurrency, etc.
  - Easy to pass input parameters to queries
  - Drill-down using hyperlinks
  - Format display types like Currency, Decimal, String, Custom Masks, etc.
  - Once created, Record and Record List queries can be re-used across the app/pages
  - Drag-n-drop is granular and iterative...you can change your mind and not be limited by the style of a wizard
  - Foreign Key relationships easy to navigate (leading to complex queries through an easy and intuitive process)
  - Paging controls for automatic cursor control
  - Service Data Objects (SDO) are a standard (JSR235)



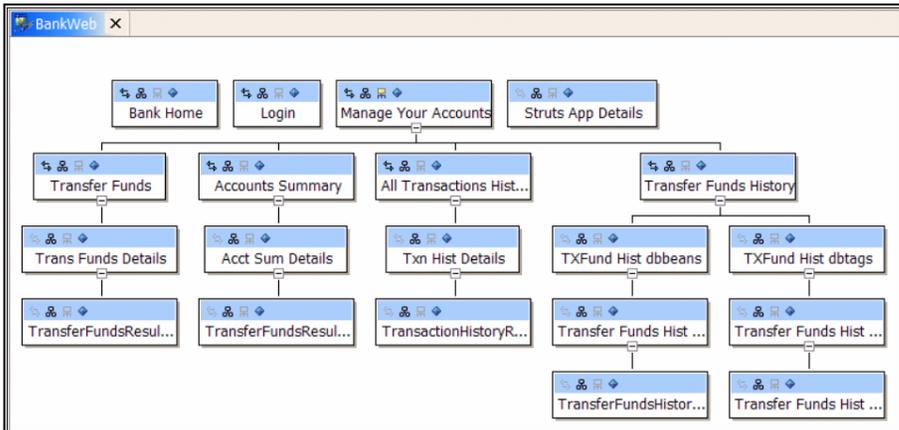
# Agenda

- **JSF in Action**
  - Web Services Creation and Consumption
  - Service Data Objects (SDO) Heterogeneous data
  - MVC Web App Rich Thin Clients
  - Portlet Apps and Portal Integration

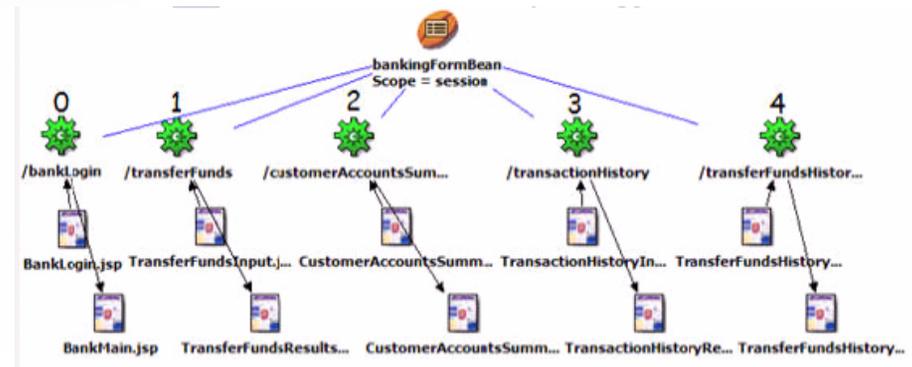


# Banking and Finance Application Scenarios <sup>1</sup>

- Building a J2EE application



A screenshot of a banking application interface. At the top, there are navigation tabs for 'Struts Application Details', 'Transfer Funds', 'Accounts Summary', 'Transaction History', and 'Transfer Funds History'. Below this is a 'Manage Your Accounts' section with a sidebar menu listing various options like 'Struts Application Details', 'Transfer Funds', 'Accounts Summary', etc. The main content area features a banner with the text 'We see the potential in every dollar...come grow with us!' and a 'Manage Your Accounts' heading. Below this, a welcome message reads 'Welcome, Arthur Platinum, to our online bank! You may manage your accounts below or logout.' A 'Main Menu' box contains four numbered items: 1) Transfer funds between your accounts, 2) Summary of your account balances, 3) History of all your transactions, and 4) History of all your fund transfers.



# JSF Ease-of-Use Banking Application Scenario

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- Account Balances page using Relational Record SDO and user session
- Transfer Funds page using Web Service and SDO account numbers combo box
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Transaction history: DB foreign keys , Tab Panels, Odyssey browser client FW, charts/graphs
- Portal Portlet development, testing, customization and Click-to-Action integration
- Portal Site/Page Designer, Page Layout and adding Portlets to Pages (People-Awareness not covered in this scenario)



# JSF Ease-of-Use Banking Application Scenario 12

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- Account Balances page using Relational Record SDO and user session
- Transfer Funds page using Web Service and SDO account numbers combo box
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Transaction history: DB foreign keys , Tab Panels, Odyssey browser client FW, charts/graphs
- Portal Portlet development, testing, customization and Click-to-Action integration

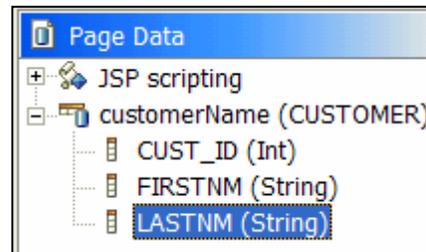
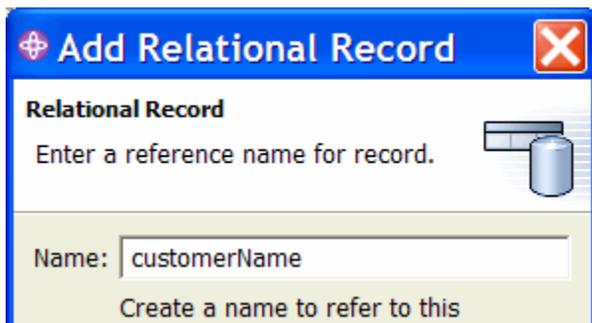
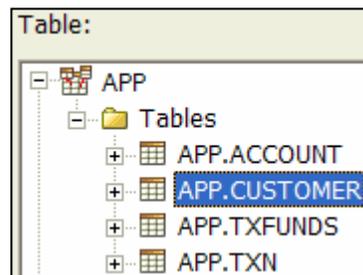
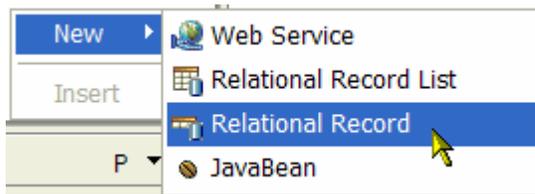
The screenshot shows the IBM WebSphere IDE interface. On the left, the **Project Navigator** displays a project named **JSFWeb** with a tree structure including **Web Site Navigation**, **Web Deployment Descriptor**, **Java Resources**, **WebContent**, and **images**. Below this is the **J2EE Hierarchy** and **Project Navigator** tabs. The **Page Data** pane shows the **sessionScope** with a **customerNumber (java.lang.String)** attribute selected. The main editor window shows a **Web Browser** view of **Logon.jsp** with a form containing **Customer ID#** and **Password** fields. A tooltip indicates: "Drop here to bind 'customerNumber' to the control 'text1'".

The screenshot shows the rendered **Logon** page. It features a decorative header with a colorful pattern. Below the header, the word **Logon** is displayed in a red serif font. A form is present with two input fields: **Customer ID#** and **Password**. The **Customer ID#** field contains the text `{customerNumber}` and `abc`. A yellow dashed box highlights the **Customer ID#** field, and a yellow arrow points to it from the right.

The screenshot shows the **Attributes** panel for the **h:inputText** component. The panel has tabs for **Basics**, **Format**, **Access**, and **Validation**. The **Basics** tab is active, showing the **Id** property set to `text1` and the **Value** property set to `#{sessionScope.customerNu}`. A small icon is visible next to the value property.

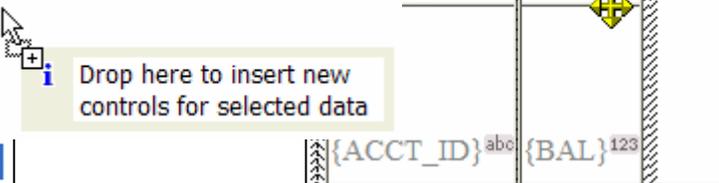
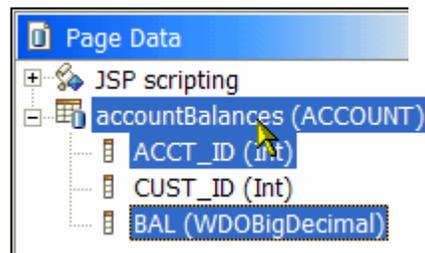
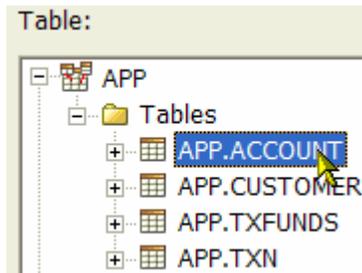
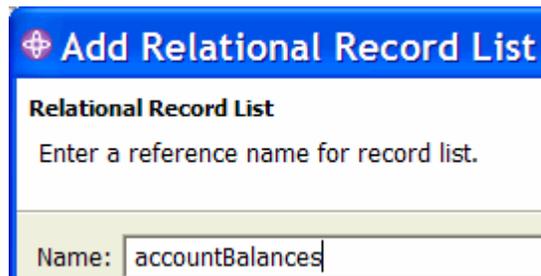
# JSF Ease-of-Use Banking Application Scenario 12

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- Account Balances page using Relational Record SDO and user session
- Transfer Funds page using Web Service and SDO account numbers combo box
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Transaction history: DB foreign keys , Tab Panels, Odyssey browser client FW, charts/graphs
- Portal Portlet development, testing, customization and Click-to-Action integration



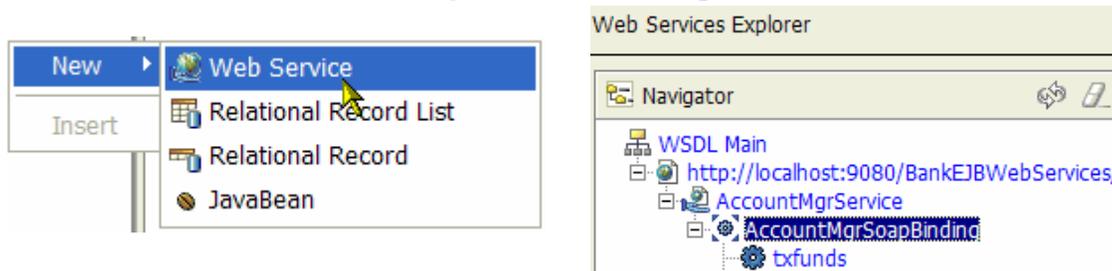
# JSF Ease-of-Use Banking Application Scenario 3a 3b

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- **Account Balances page using Relational Record SDO and user session**
- Transfer Funds page using Web Service and SDO account numbers combo box
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Transaction history: DB foreign keys , Tab Panels, Odyssey browser client FW, charts/graphs
- Portal Portlet development, testing, customization and Click-to-Action integration



# JSF Ease-of-Use Banking Application Scenario 4a 4b 4c

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- Account Balances page using Relational Record SDO and user session
- **Transfer Funds page using Web Service and SDO account numbers combo box**
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Transaction history: DB foreign keys , Tab Panels, Odyssey browser client FW, charts/graphs
- Portal Portlet development, testing, customization and Click-to-Action integration



### Web Service Discovery Home

[Web Services from a UDDI Registry](#)  
Search for Web services by name or key in a public or private UDDI registry.

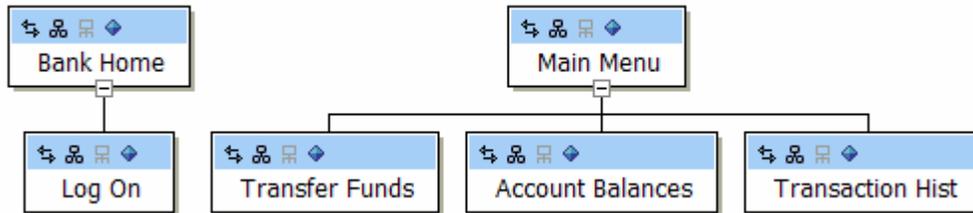
[Web Services from a known URL](#)  
Search for Web services using a URL to a document.

[Web Services from your workspace](#)  
Search for Web services deployed and running in my current workspace.

The image shows a web form titled 'Transfer Funds'. At the top, there is a decorative banner with various icons. Below the banner, the text reads: 'Transfer Funds<sup>BR,↓</sup>  
BR,↓  
{LASTNM}<sup>abc</sup>, you may transfer funds below. BR,↓'. The form contains three input fields: 'SourceAcct' with a dropdown menu showing 'item 1', 'TargetAcct' with a dropdown menu showing 'item 1', and 'Amount' with a text input field containing '{amount}' and a small '123' to its right. Below the input fields, there is a red asterisk icon followed by '{Error Messages}'<sup>BR,↓</sup>. At the bottom of the form is a 'Transfer Funds' button.

# JSF Ease-of-Use Banking Application Scenario w

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- Account Balances page using Relational Record SDO and user session
- Transfer Funds page using Web Service and SDO account numbers combo box
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Transaction history: DB foreign keys , Tab Panels, Odyssey browser client FW, charts/graphs
- Portal Portlet development, testing, customization and Click-to-Action integration



# JSF Ease-of-Use Banking Application Scenario <sup>5</sup>

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- Account Balances page using Relational Record SDO and user session
- Transfer Funds page using Web Service and SDO account numbers combo box
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Transaction history: DB foreign keys , Tab Panels, Odyssey browser client FW, charts/graphs
- Portal Portlet development, testing, customization and Click-to-Action integration



**Keeping Tabs On Your Accounts**

Charlie, your may manage your accounts below.

Acct_id	Bal
5	93,000
6	107,000

Refresh



**Keeping Tabs On Your Accounts**

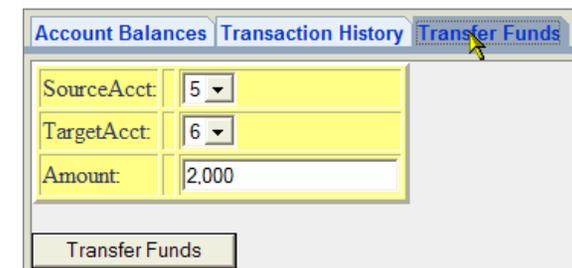
Charlie, your may manage your accounts below.

Txn_id	Txn_type	Acct_id	Amount	C
1086689270349	DEP	6	2,000	Deposit t
1086689270349	WDR	5	2,000	Withdraw
1086688659851	DEP	6	1,000	Deposit t



## Keeping Tabs On Your Accounts

Charlie, your may manage your accounts below.



Account Balances Transaction History **Transfer Funds**

SourceAcct: 5

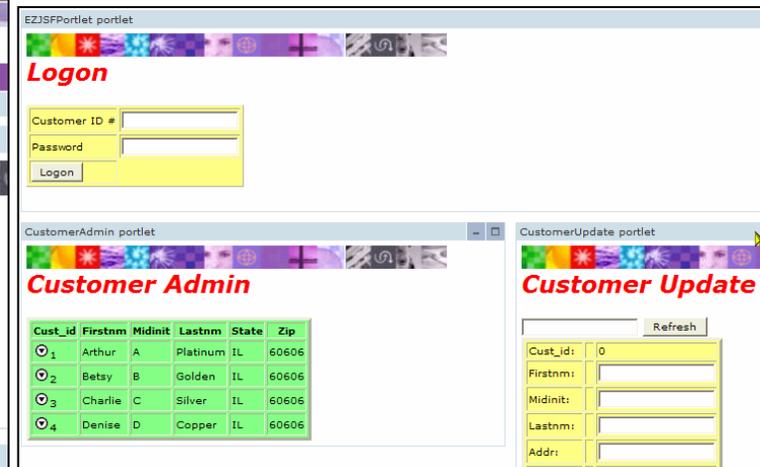
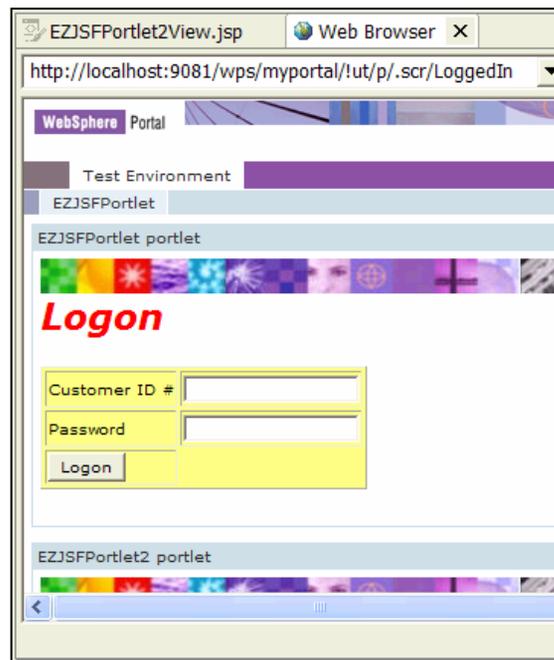
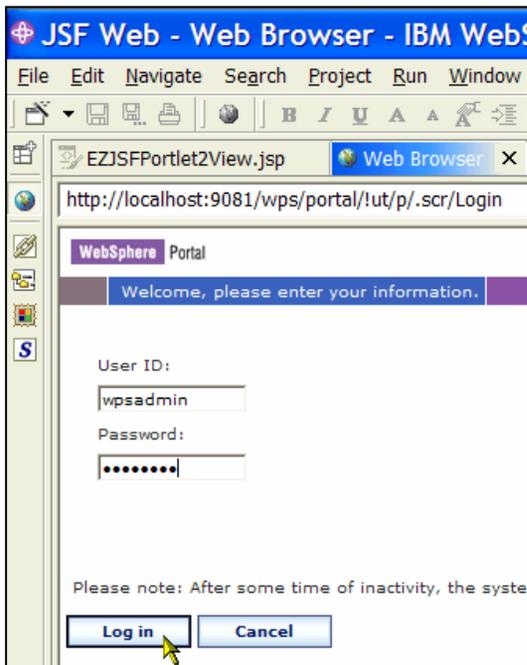
TargetAcct: 6

Amount: 2,000

Transfer Funds

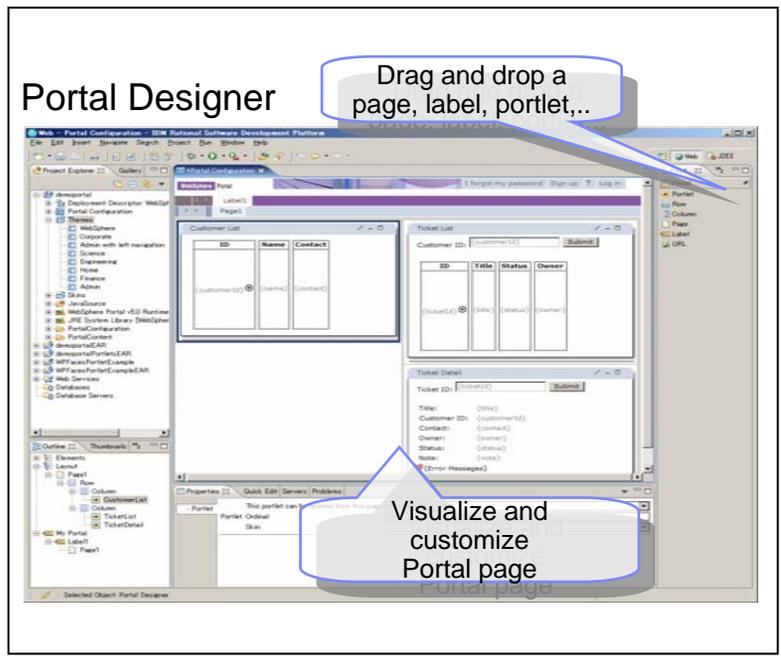
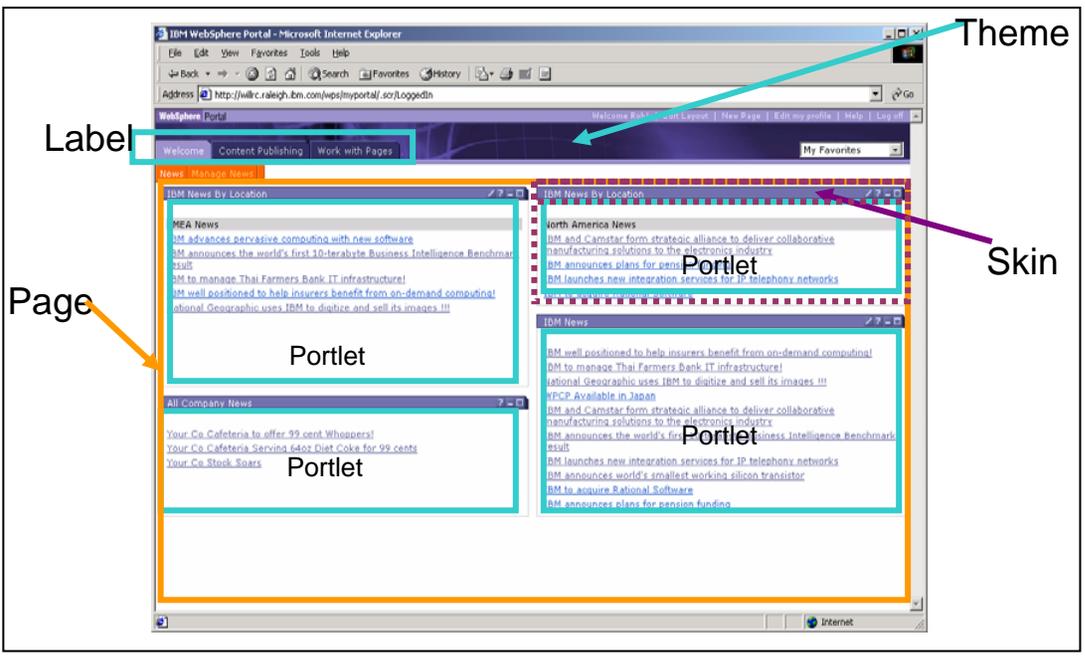
# JSF Ease-of-Use Banking Application Scenario 1 2 3 4

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- Account Balances page using Relational Record SDO and user session
- Transfer Funds page using Web Service and SDO account numbers combo box
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Rich thin client tab panel view for account summary consolidation
- Click-2-Action integrated, drag-n-drop automation

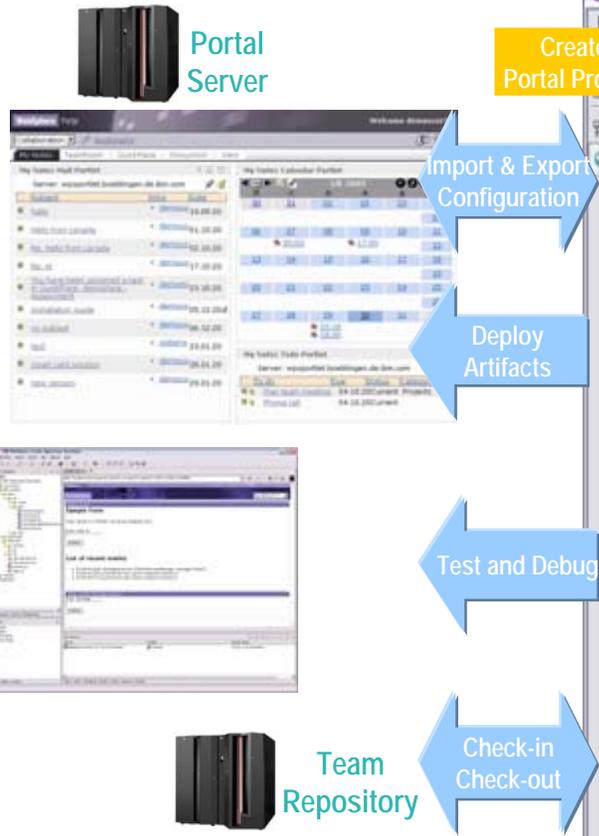


# JavaServer Faces Ease-of-Use <sup>4</sup>

- Logon page and binding customer number into user session/state
- Main Menu page with welcome greeting using user session & Relational Record SDO
- Account Balances page using Relational Record SDO and user session
- Transfer Funds page using Web Service and SDO account numbers combo box
- Web site navigation and common look-and-feel using Web Site Designer & Templates
- Rich thin client tab panel view for account summary consolidation
- Portal Designer, Page Layout and adding Portlets to Pages (note: People Awareness Not Covered )



# Portal Designer: Site, Page, Themes, Skins 1234



The screenshot shows the **IBM WebSphere Studio Application Developer** interface with the following elements:

- Project Explorer (Left):** Shows the project structure for **MyPortal**, including **Web Deployment Descriptor**, **Java Source**, **Web Content**, **META-INF**, **skins**, **themes**, **WEB-INF**, and **Libraries**.
- Server Configuration (Middle-Left):** Shows **Portal Configuration** for **MyPortal**, listing **Pages** (Welcome, Home\_ScreamingMedia, My\_Portlets), **Portal Administration**, and **Themes** (Admin, AdminLeftNavigation, Corporate, Engineering, Finance, Science, WebSphere).
- Main Editor (Right):** Displays a hierarchical tree of the portal structure:
  - MyPortal** (Root)
  - Welcome** (Place)
  - Portal Admi...** (Place / Page Group)
  - Home\_Scre...** (Page)
  - My\_Portlets** (Page Group) containing **FT Portl...** and **World CL...** (Portlets)
  - Portal Setti...** (Page)
- Thumbnail (Bottom-Right):** A **Preview File (preview.gif)** window showing a grid of eight different theme preview images. A yellow arrow labeled **Apply themes** points to this window.

**Portal Site Designer**



# Demonstration Agenda

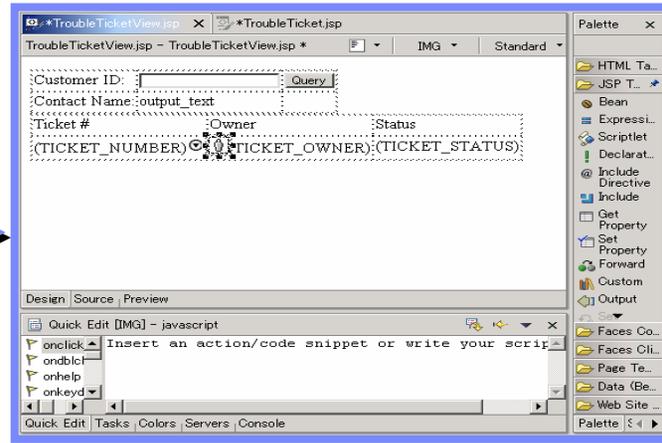
- **JSF in Action**

- Web Services Creation and Consumption
- Database App CRUD and UI
- MVC Web App Rich Thin Clients
- Portlet Apps and Portal Integration



# Portlet/Portal Development Ease-of-Use 1 2 3 4

- Visual portlet development uses the JSF styles of construction
- Visual Portal Designer makes site-wide themes/skins look-n-feel easy
- Portlet specific syntax and deployment descriptors are generated
- Click-2-Action and People Awareness drag-n-drop development
- Portal Server Test Environment is turnkey (Improved Portal Toolkit installation)
- Bottom line:
  - Web Services, UI and MVC enhancements in JSF apply to portlets
  - Portlet development is significantly enhanced and easier than ever before



# Portlet Development

**Create a Portlet Project**

**Define the Portlet Project**  
Create a portlet project.

**JSF and Struts portlet creation by a wizard**

Project name:

Project location:

Select a portlet type:

- Create empty portlet
- Create basic portlet
- Create Faces Portlet
- Create Struts Portlet**

Description:

Create a Struts portlet. You will be asked to select several options for generating sample code.  
You may customize the code to integrate the portlet into your portal.

Configure advanced options

**Visual Portlet JSF editing in Page Designer**

Visual Portlet JSF editing in Page Designer

Click-to-Action

People Awareness

Drag & Drop



# Demonstration Agenda

- **JSF in Action**

- ✓ Web Services Creation and Consumption
- ✓ Service Data Object (SDO) Heterogeneous Data
- ✓ MVC Web App Rich Thin Clients
- ✓ Portlet Apps and Portal Integration



# J2EE Development *Automated by JSF*

Input → Process → Output

## • Input

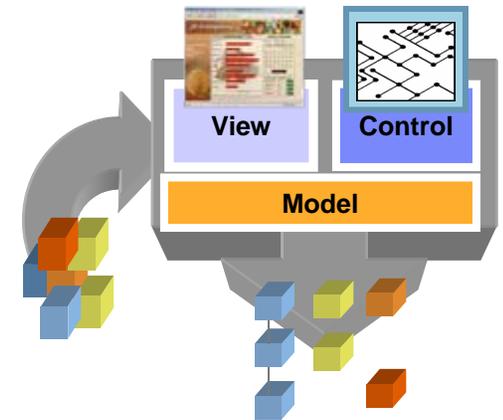
- Constructing web page forms, formatting, validation, etc.
- GUI widget richness
- Calling appropriate "Action" when button pressed, etc.
- Passing data to business logic

## • Processing

- Integrating to web services, databases, backends, biz logic, etc.
- Maintaining user session and application state information
- MVC layering, page flow and business logic modularity

## • Output

- Forwarding control to results screen/page
- Displaying data, GUI controls, formatting, etc.



# IBM RAD JSF Tools Summary

- ✓ **Takes Web, J2EE and Portal development to a new level of productivity**
  
- ✓ **Appeals to Web Developers**
  - ✓ J2EE is more powerful than .NET and JSF now makes it both easy and attractive to Microsoft heritage developers
  - ✓ JSF Simplifies and enhances RAD development for both J2EE newbies and experts
  
- ✓ **Appeals to Portal Developers**
  - ✓ Improved integration, visual portlet construction and JSF tools provide a comprehensive portlet development environment
  
- ✓ **Bottom Line**
  - ➔ Accelerates GUI, Web Services, Database, Portlet and MVC Web App development



# Application Development Challenges Addressed

- It takes too long to deliver applications
  - “We need to *Design, Develop and Deploy applications more quickly*”
  - ✓ IBM’s new JavaServer Faces (JSF) tools accelerate development using a drag-n-drop approach (no coding required)
- J2EE is perceived to be too complex
  - “We need to leverage open standards, keep it simple and make it easy for Microsoft heritage developers to get on board”
  - ✓ IBM’s new JSF tools make J2EE easy and very attractive to Microsoft heritage developers
- Integration is cost prohibitive
  - “We need to integrate our people, processes and information using web services, databases and existing systems”
  - ✓ IBM’s new JSF tools automate web services and database integration
  - ✓ IBM’s business process integration and workflow tools save time





# Lifecycle Tools – Integrated into a single IDE Platform

## Disciplines

Process Modeler Business Modeling

RSA (RequisitePro Views) Requirements

RSA/RAD Analysis & Design

RAD Implementation

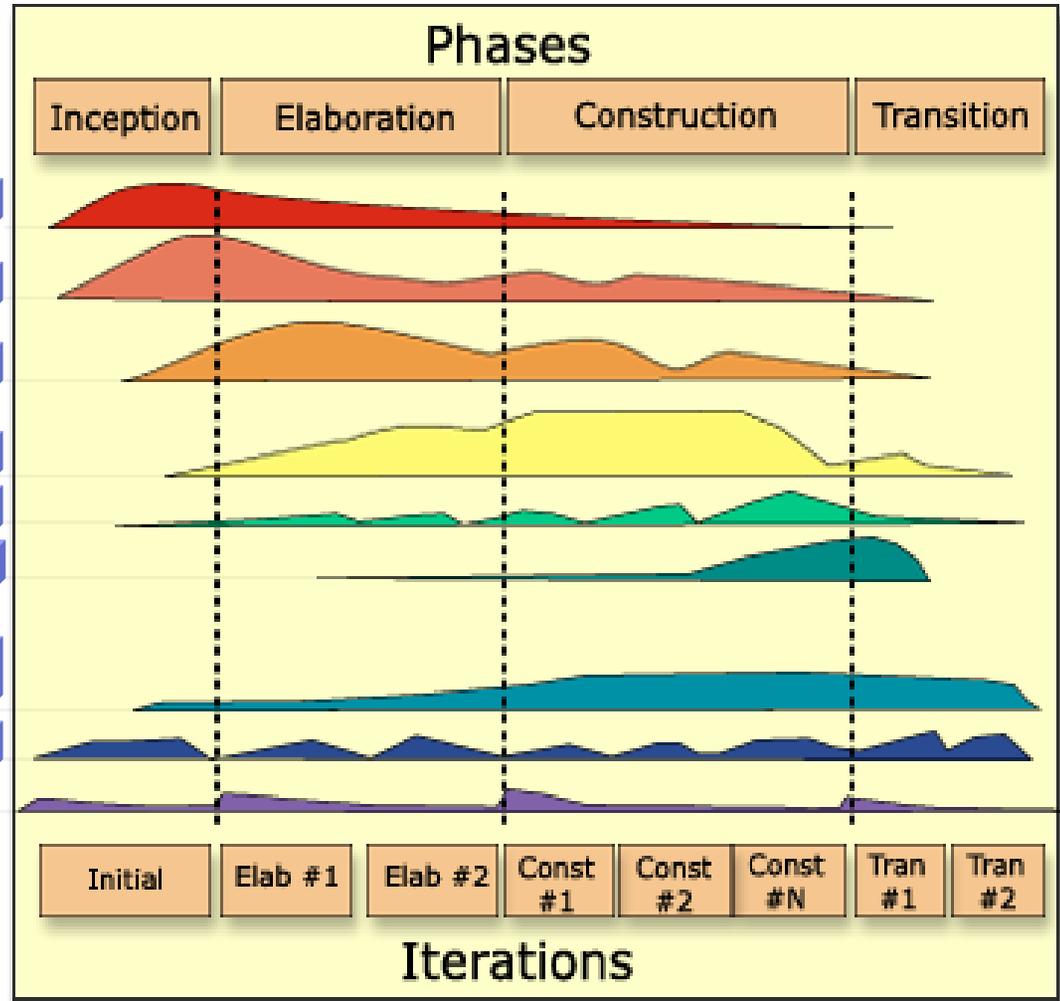
RAD Test

RAD Deployment

RAD (ClearCase Views + ClearQuest Views) Configuration & Change Mgmt

RAD (ClearQuest + Project Console) Project Management

Environment

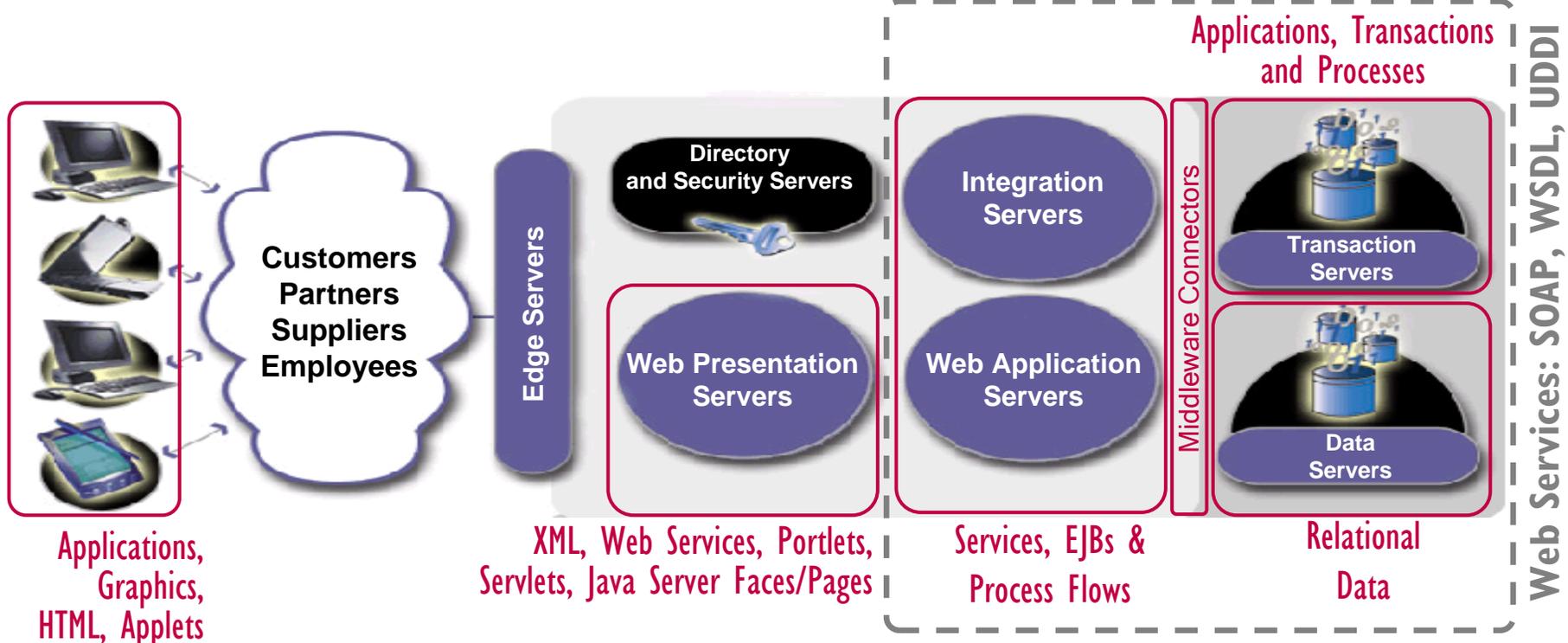


### Simplified Packaging and Names

V4 & V5	V6
WSAD	RAD (Rational App Developer)
Rose XDE	RSM (Rational SW Modeler)
WSAD + XDE	RSA (Rational SW Architect)



# Build Tools - End-2-End Design & Construction



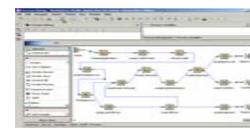
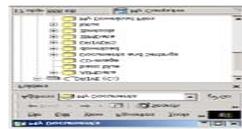
Applications, Graphics, HTML, Applets

XML, Web Services, Portlets, Servlets, Java Server Faces/Pages

Services, EJBs & Process Flows

Relational Data

Web Services: SOAP, WSDL, UDDI



## Development Roles

Application Modeling

Java/J2EE Development

Web/Portal Development

XML Web Services Development

Database App Development

Business Process SOA Integration

Cobol, CICS/IMS, 4GL Development



# RAD Top 10 of IBM AD Tools (Adaptable, Automated, Accelerated)

## **Adaptable** - Eclipse-based Workbench

**#1** Customizable **Perspectives** and **Views** support **roles-based** tools

## **Automated** - IDE Infrastructure (Code, Build, Test, Debug, Deploy)

**#2** **Dynamic build engine** incrementally updates and hot-deploys (on-the-fly, while you type)

**#3** **Testing and Debugging** for Java, Web Services, XML, database and EJB

## **Accelerated** - End-2-End Builders

**#4** **Java and UML** core coding tools

**#5** **Web App Builders**: Pages, Sites, Portals, Portlets, Struts, JavaServer Faces

**#6** **Web Services** and **XML** Builders

**#7** **Database and SDO** (Service Data Objects) Builders

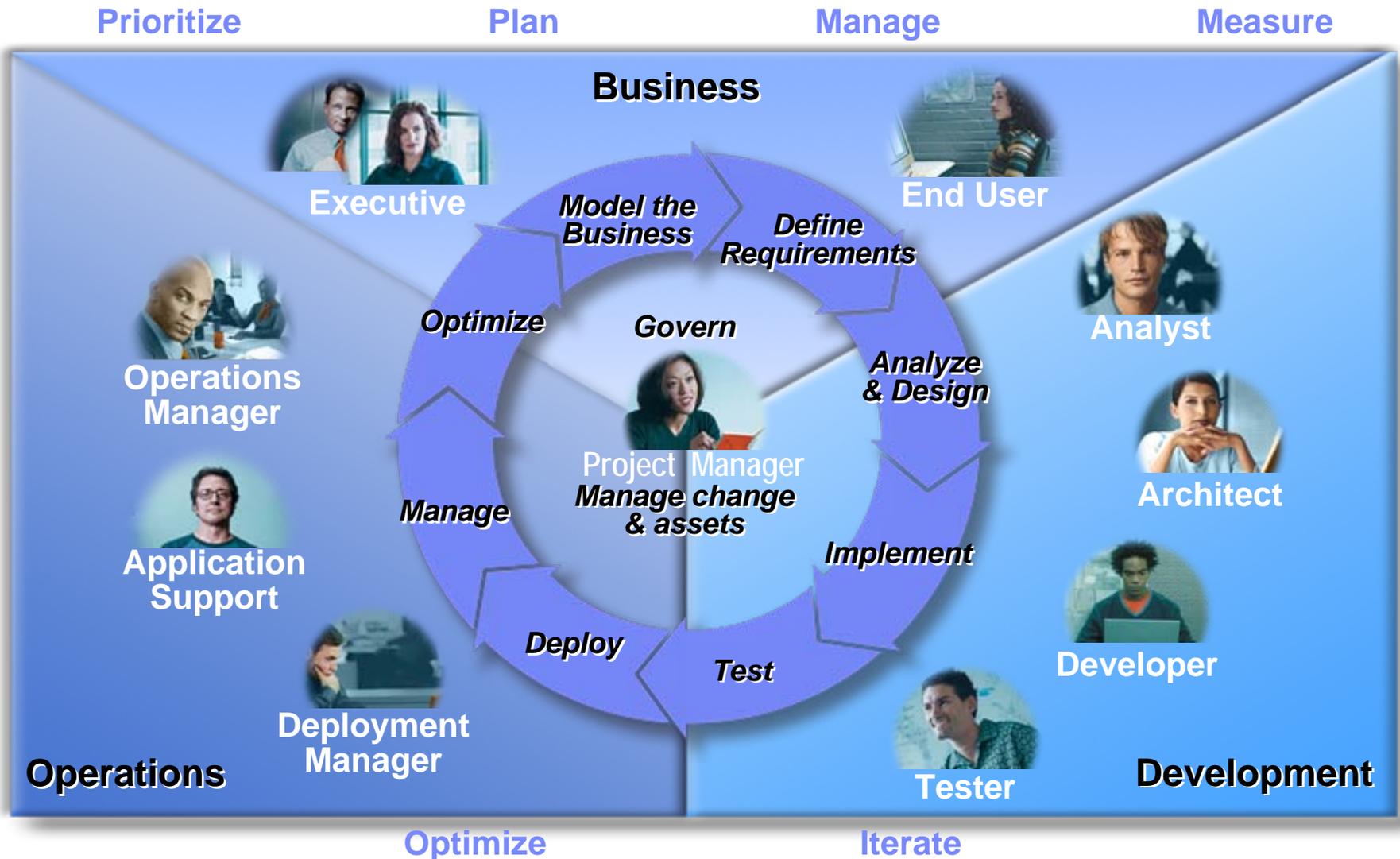
**#8** **J2EE** and **EJB** Builders

**#9** **SOA workflow** builders (Service Oriented Architecture)

**#10** **Legacy/Enterprise** development, integration and Web Services tools



# The Business-Driven Development Lifecycle



# Resources

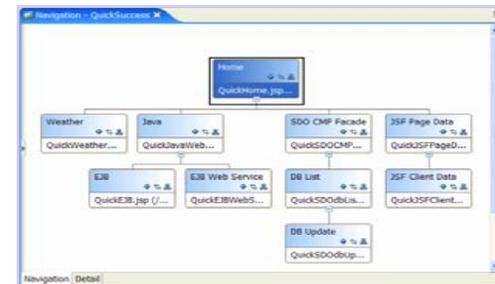
- <http://ibm.com/developer>  
(IBM developerWorks for technologies and products)
  - [ibm.com/developer/websphere](http://ibm.com/developer/websphere)  
(WebSphere Developers Domain)
  - [ibm.com/developer/rational](http://ibm.com/developer/rational)  
(WebSphere Developers Domain)
  - [ibm.com/developer/ibm](http://ibm.com/developer/ibm)  
(All other product developer domains)
- <http://www.redbooks.ibm.com>  
(How To Books)
- <http://ibm.com/software/ad>  
(IBM App Dev Tools Homepage)
- <http://ibm.com/websphere>  
(WebSphere Family Homepage)
- <http://JavaDevTools.com> (a.k.a. RationalCentral.com and WebSphereCentral.com)  
(Presentations, references, tutorials, white papers, articles, links, etc.)





# Quick Success Scenarios – Ad Hoc Build Tasks

- Weather Web Service – Consuming an internet based Web Service
- Java Web Service – Creating and consuming a JavaBean Web Service
- Session EJB - Creating, coding, testing, and web service enablement
- CMP EJBs – Mapping to a relational database
- CMP SDO Facade – Creating an SDO façade for a CMP EJB
- SDO CRUD – Create, retrieve, update and delete relational data
- JSF Client Data – Enhanced Browser cached JSF UI Components
- Web Site Designer – Constructing and configuring web site navigation



# Quick Success Scenarios – Ad Hoc Build Tasks

- Weather Web Service – Consuming an internet based Web Service
- Java Web Service – Creating and consuming a JavaBean Web Service
- Session EJB - Creating, coding, testing, and web service enablement
- CMP EJBs – Mapping to a relational database
- CMP SDO Facade – Creating an SDO façade for a CMP EJB
- SDO CRUD – Create, retrieve, update and delete relational data
- JSF Client Data – Enhanced Browser cached JSF UI Components

**Web Service Discovery Dialog**

Web Services from a UDDI Registry

Select or enter a registry's inquiry URL followed by a partial or complete service name or key and click Go. You may use '%' wildcards when searching by the service name.

UDDI Registry:

Search text:

Search by:  Service name  Service key

UDDI Services:

Name	Description	WSDL URL
Weather - Temperature	Endpoint for service	<a href="http://www.xmethods.net/sd/2001/Ten">http://www.xmethods.net/sd/2001/Ten</a>
Weather Fetcher	Endpoint for service	<a href="http://qlkev.webs.innerhost.com/qlkev">http://qlkev.webs.innerhost.com/qlkev</a>
Weather Services from EJSE, Inc.	Endpoint for service	<a href="http://www.ejse.com/WeatherService/">http://www.ejse.com/WeatherService/</a>

New: Scripting Variable, Web Service, EJB Session bean, Relational Record List, Relational Record, JavaBean

QuickWeatherWebService.jsp

Web Browser

http://localhost:9080/QuickSuccess/faces/QuickWeatherWebService.

**Quick Weather Web Service**

ZipCode:

Submit

Location: Fort Lauderdale, FL

Forecast: Mostly Cloudy

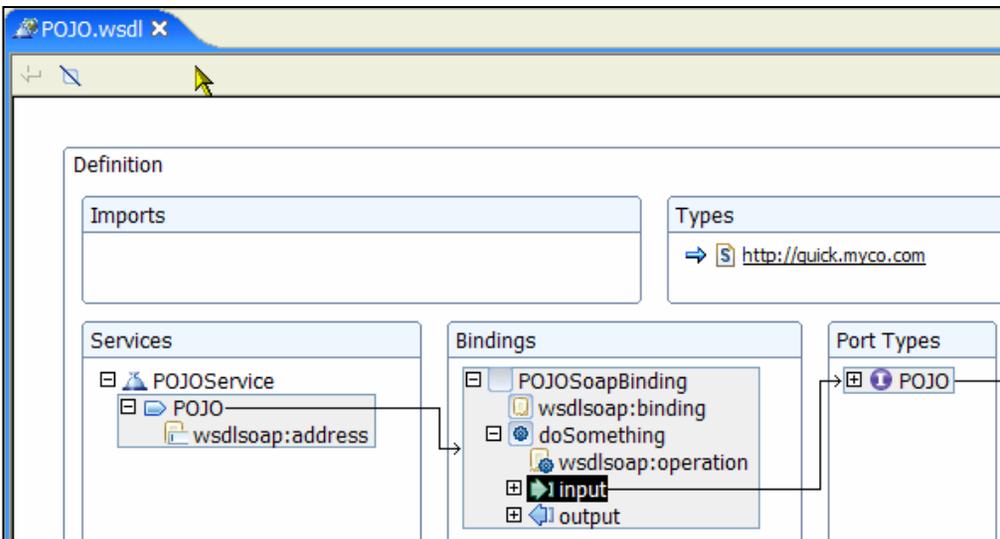
Temperature: 75°F

FeelsLike: 75°F

Humidity: 90%

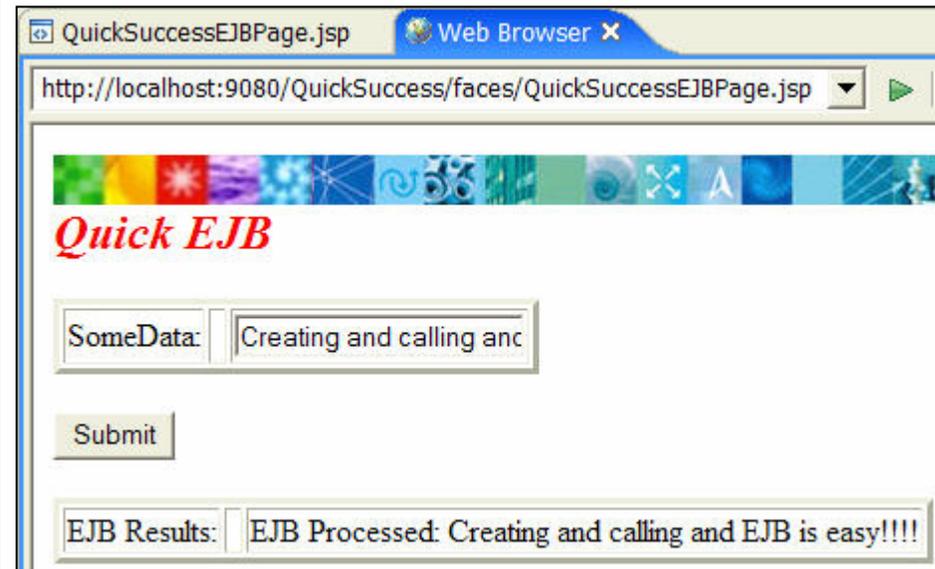
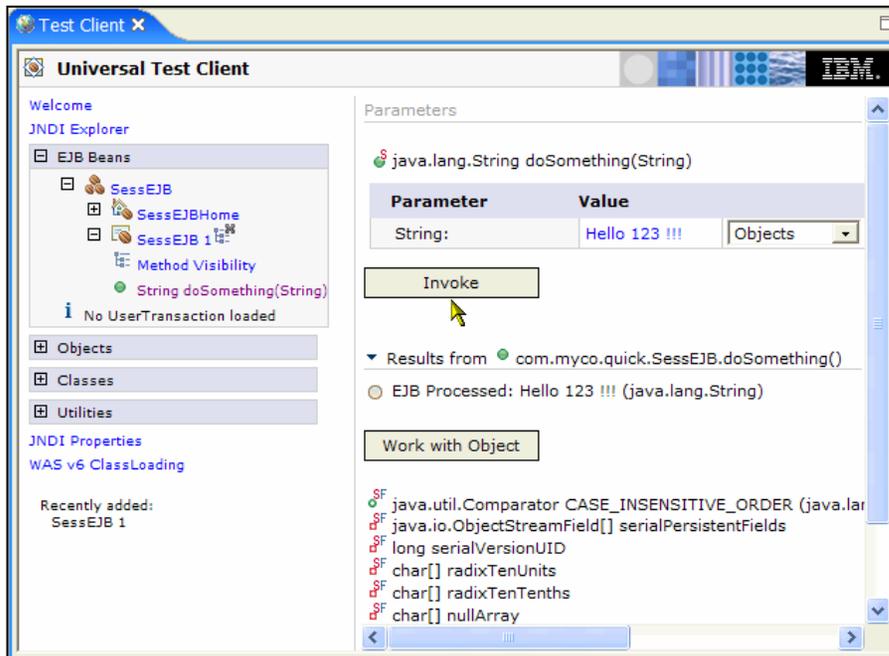
# Quick Success Scenarios – Ad Hoc Build Tasks

- Weather Web Service – Consuming an internet based Web Service
- Java Web Service – Creating and consuming a JavaBean Web Service
- Session EJB - Creating, coding, testing, and web service enablement
- CMP EJBs – Mapping to a relational database
- CMP SDO Facade – Creating an SDO façade for a CMP EJB
- SDO CRUD – Create, retrieve, update and delete relational data
- JSF Client Data – Enhanced Browser cached JSF UI Components



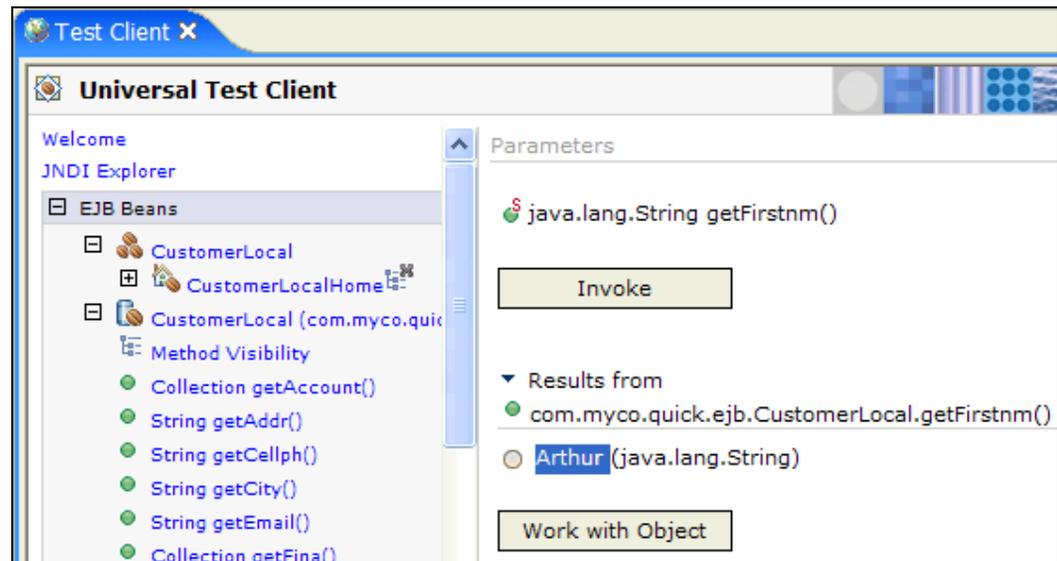
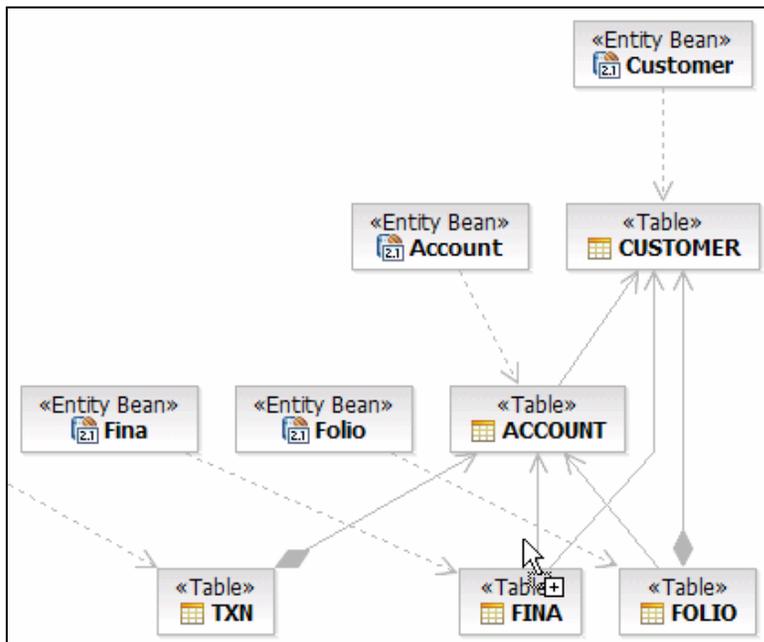
# Quick Success Scenarios – Ad Hoc Build Tasks

- Weather Web Service – Consuming an internet based Web Service
- Java Web Service – Creating and consuming a JavaBean Web Service
- **Session EJB** - Creating, coding, testing, and web service enablement
- CMP EJBs – Mapping to a relational database
- CMP SDO Facade – Creating an SDO façade for a CMP EJB
- SDO CRUD – Create, retrieve, update and delete relational data
- JSF Client Data – Enhanced Browser cached JSF UI Components



# Quick Success Scenarios – Ad Hoc Build Tasks

- Weather Web Service – Consuming an internet based Web Service
- Java Web Service – Creating and consuming a JavaBean Web Service
- Session EJB - Creating, coding, testing, and web service enablement
- **CMP EJBs** – Mapping to a relational database
- CMP SDO Facade – Creating an SDO façade for a CMP EJB
- SDO CRUD – Create, retrieve, update and delete relational data
- JSF Client Data – Enhanced Browser cached JSF UI Components



# Quick Success Scenarios – Ad Hoc Build Tasks

- Weather Web Service – Consuming an internet based Web Service
- Java Web Service – Creating and consuming a JavaBean Web Service
- Session EJB - Creating, coding, testing, and web service enablement
- CMP EJBs – Mapping to a relational database
- **CMP SDO Facade** – Creating an SDO façade for a CMP EJB
- SDO CRUD – Create, retrieve, update and delete relational data
- JSF Client Data – Enhanced Browser cached JSF UI Components

**Create Session Bean Facade**

**Session Bean Facade**  
Create a session bean facade for CMP entity beans.

Name:

Package Name:

Source Location:

Available CMP beans:

- Customer
- Account
- Fina
- Folio
- Txn

QuickSDOCMPFacadePage.jsp Web Browser x

http://localhost:9080/QuickSuccess/faces/QuickSDOCMPFacadePage.

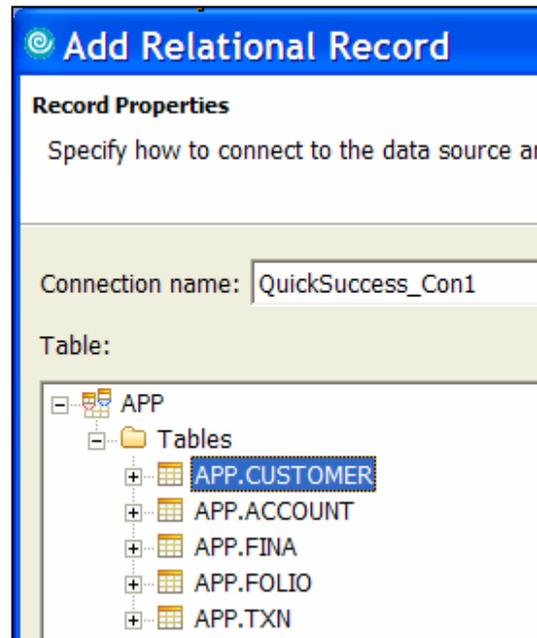
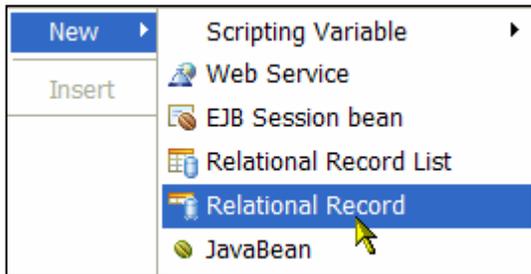


## Quick SDO CMP EJB Session Bean Facade

Cust_id	Firstnm	Lastnm	Addr	City	State	Zip
1	Arthur	Platinum	123 Easy Street	Long Island	NY	11101
2	Betsy	Golden	45 Plush Circle	Arlington	VA	22202
3	Charlie	Silver	67 Humble Avenue	Fort Lauderdale	FL	33303
4	Denise	Copper	89 Trailer Park Drive	Burghill	OH	44404

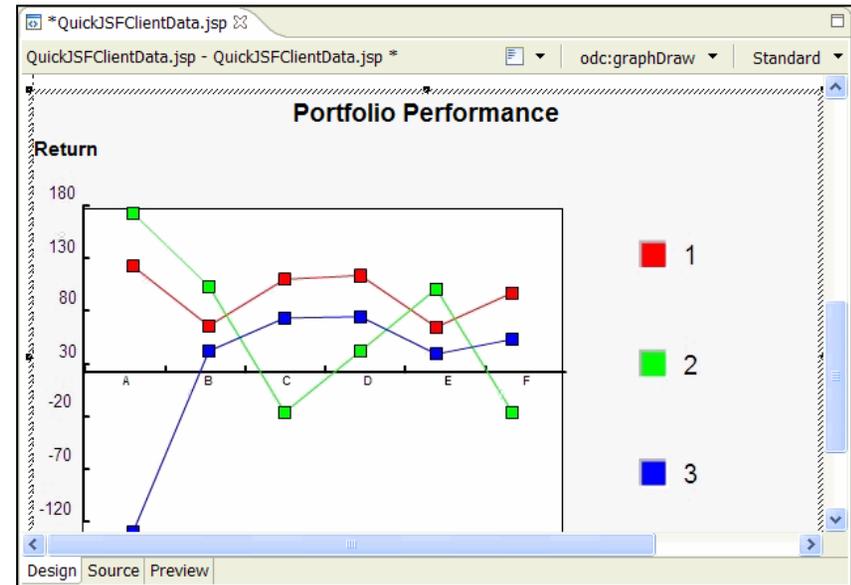
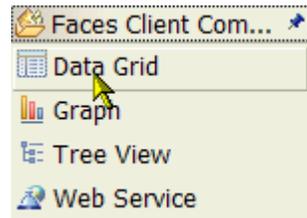
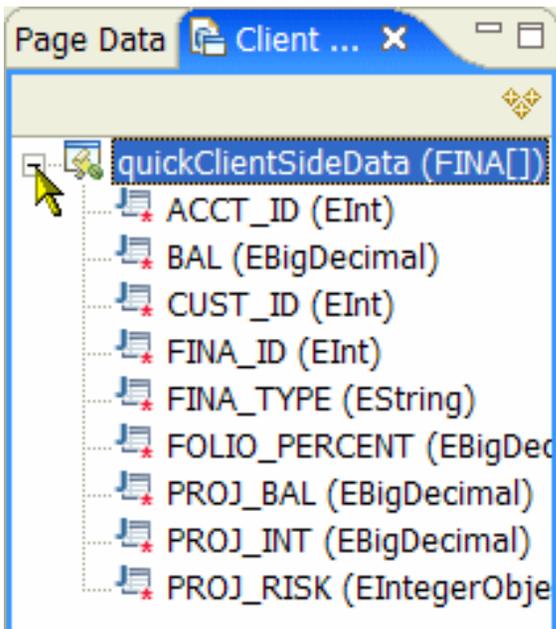
# Quick Success Scenarios – Ad Hoc Build Tasks

- Weather Web Service – Consuming an internet based Web Service
- Java Web Service – Creating and consuming a JavaBean Web Service
- Session EJB - Creating, coding, testing, and web service enablement
- CMP EJBs – Mapping to a relational database
- CMP SDO Facade – Creating an SDO façade for a CMP EJB
- **SDO CRUD** – Create, retrieve, update and delete relational data
- JSF Client Data – Enhanced Browser cached JSF UI Components



# Quick Success Scenarios – Ad Hoc Build Tasks

- Weather Web Service – Consuming an internet based Web Service
- Java Web Service – Creating and consuming a JavaBean Web Service
- Session EJB - Creating, coding, testing, and web service enablement
- CMP EJBs – Mapping to a relational database
- CMP SDO Facade – Creating an SDO façade for a CMP EJB
- SDO CRUD – Create, retrieve, update and delete relational data
- JSF Client Data – Enhanced Browser cached JSF UI Components



# End-2-End Scenario – Soup-2-Nuts J2EE Build Tasks

- Solution Overview – Single JSP captures all build and run scenarios
- Java Beans – Create, code, test, debug and JSF integration
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application artifacts
- Performance Profiling – Analyzing performance with graphics and reports

The screenshot displays the 'End-2-End Page' interface for 'End2EndPage.jsp and .java'. It is divided into several sections:

- Web Services:** Shows a 'Temperature Web Service' with input fields for Zip Code (11101), License Key (WS14-PBJ1-KXY3), and a 'Submit' button. Results show Cond: Clear, City: Long Island City, State: New York, and Temp°F: 28.
- XML:** Includes 'Create XML Doc' and 'Read XML Doc' sections. The XML content is:
 

```
<?xml version="1.0" encoding="UTF-8"?> <E2E>
      <ID>123</ID> <INFO>H XML</INFO> </E2E>
```
- JSF:** Shows 'Client-side data' with a table:
 

COUNTER	COUNT
CounterA	111
CounterB	200
- Diagrams:** Displays UML diagrams for 'End2EndPage', 'WebServiceProxy', 'FastWeather Web Service', 'Internet', 'XML SDO Beans', and 'E2E XML Doc'.
- Performance:** Includes a 'Counter Usage' bar chart and a 'Counter Usage' table:
 

Counter	Count
1	1
2	2

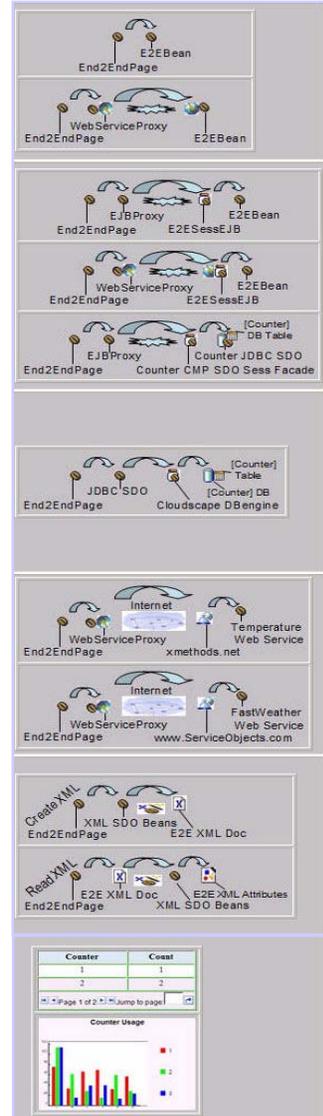
# End-2-End Scenario – Soup-2-Nuts J2EE Build Tasks

- **Solution Overview** – Single JSP captures all build and run scenarios
- Java Beans – Create, code, test, debug and JSF integration
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application artifacts
- Performance Profiling – Analyzing performance with graphics and reports

**End-2-End Page** (End2EndPage.jsp and .java)

Call All

Type of Calls	Input->Submit->Results	Diagrams										
<b>Web Services:</b> o Temperature (www.xmethods.net) o FastWeather (ServiceObjects.com)	<table border="1"> <tr> <td>Temperature Web Service</td> <td>FastWeather Web Service</td> </tr> <tr> <td>Zip Code: 11101</td> <td>Cond: Clear</td> </tr> <tr> <td>License Key: WS14-PBJ1-KCY3</td> <td>City: Long Island City</td> </tr> <tr> <td>Submit</td> <td>State: New York</td> </tr> <tr> <td></td> <td>Temp*F: 28</td> </tr> </table>	Temperature Web Service	FastWeather Web Service	Zip Code: 11101	Cond: Clear	License Key: WS14-PBJ1-KCY3	City: Long Island City	Submit	State: New York		Temp*F: 28	
Temperature Web Service	FastWeather Web Service											
Zip Code: 11101	Cond: Clear											
License Key: WS14-PBJ1-KCY3	City: Long Island City											
Submit	State: New York											
	Temp*F: 28											
<b>XML:</b> o Create XML Doc o Recieve XML Doc	<table border="1"> <tr> <td>Create XML Doc</td> <td>Read XML Doc</td> </tr> <tr> <td>Id: 123</td> <td>Info: Hi XML!</td> </tr> <tr> <td>Submit</td> <td></td> </tr> <tr> <td>XML: &lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;E2E&gt;</td> <td>Doc: &lt;ID&gt;123&lt;/ID&gt;&lt;INFO&gt;Hi XML!&lt;/INFO&gt;&lt;E2E&gt;</td> </tr> </table>	Create XML Doc	Read XML Doc	Id: 123	Info: Hi XML!	Submit		XML: <?xml version="1.0" encoding="UTF-8"?><E2E>	Doc: <ID>123</ID><INFO>Hi XML!</INFO><E2E>			
Create XML Doc	Read XML Doc											
Id: 123	Info: Hi XML!											
Submit												
XML: <?xml version="1.0" encoding="UTF-8"?><E2E>	Doc: <ID>123</ID><INFO>Hi XML!</INFO><E2E>											
<b>JSF:</b> Client-side data: o Table o Graphs	<table border="1"> <thead> <tr> <th>COUNTER</th> <th>COUNT</th> </tr> </thead> <tbody> <tr> <td>CounterA</td> <td>111</td> </tr> <tr> <td>CounterB</td> <td>200</td> </tr> </tbody> </table> <p>Page 1 of 2   Jump to page: [ ]</p>	COUNTER	COUNT	CounterA	111	CounterB	200					
COUNTER	COUNT											
CounterA	111											
CounterB	200											



# End-2-End Scenario – Soup-2-Nuts J2EE Build Tasks

- Solution Overview – Single JSP captures all build and run scenarios
- Java Beans – Create, code, test, debug and JSF integration
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application artifacts
- Performance Profiling – Analyzing performance with graphics and reports

The screenshot displays the IBM Rational Software Development Platform interface during a debug session. The main window is titled "Debug - E2EBean.java - IBM Rational Software Development Platform". The interface includes a menu bar (File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help), a toolbar, and several panels:

- Debug Console:** Shows the execution stack with the following entries:
  - E2EBean.aJavaMethod(String) line: 11
  - CodeSnippet\_3.run() line: 5
  - ScrapbookMain1.eval(Class) line: 20
  - NativeMethodAccessorImpl.invoke0(Method, Object, Object[]) line: 85
  - NativeMethodAccessorImpl.invoke(Method, Object, Object[]) line: 85
  - DelegatingMethodAccessorImpl.invoke(Method, Object, Object[]) line: 85
  - Method.invoke(Object, Object[]) line: 391
- Variables Panel:** Shows the current object: `this= E2EBean (id=48)` with a variable `someData= "Hi, my name is ..."`. A context menu is open over this variable, with options including "Change Value...", "Copy Variables", "Select All", "New Detail Formatter...", "Open Declared Type", "Open Declared Type Hierarchy", "Instance Breakpoints...", "Watch", and "Inspect".
- Source Editor:** Shows the source code of `E2EBean.java`. The package is `com.myco.e2e.java`. The class is `E2EBean` with an author of `Bill Hahn`. The `aJavaMethod` method is highlighted, showing the return statement: `return "[E2EBean.aJavaMethod] Processed >>" + someData;`
- UML Diagram:** Located in the bottom right, it shows a class hierarchy with `com.myco.e2e.java` as the package, `E2EBean` as the class, and `aJavaMethod(String)` as the method.

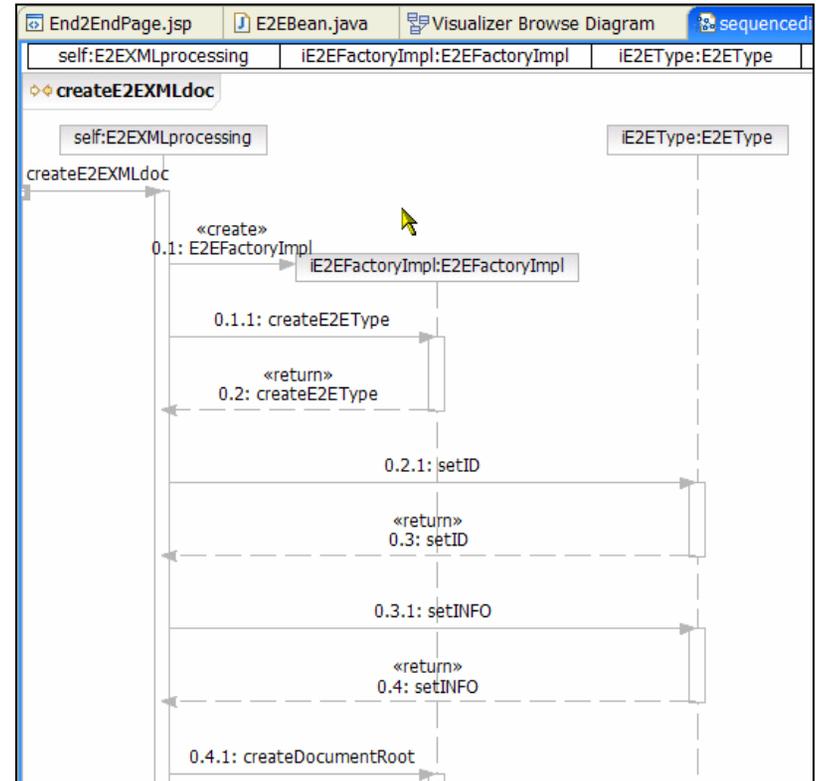
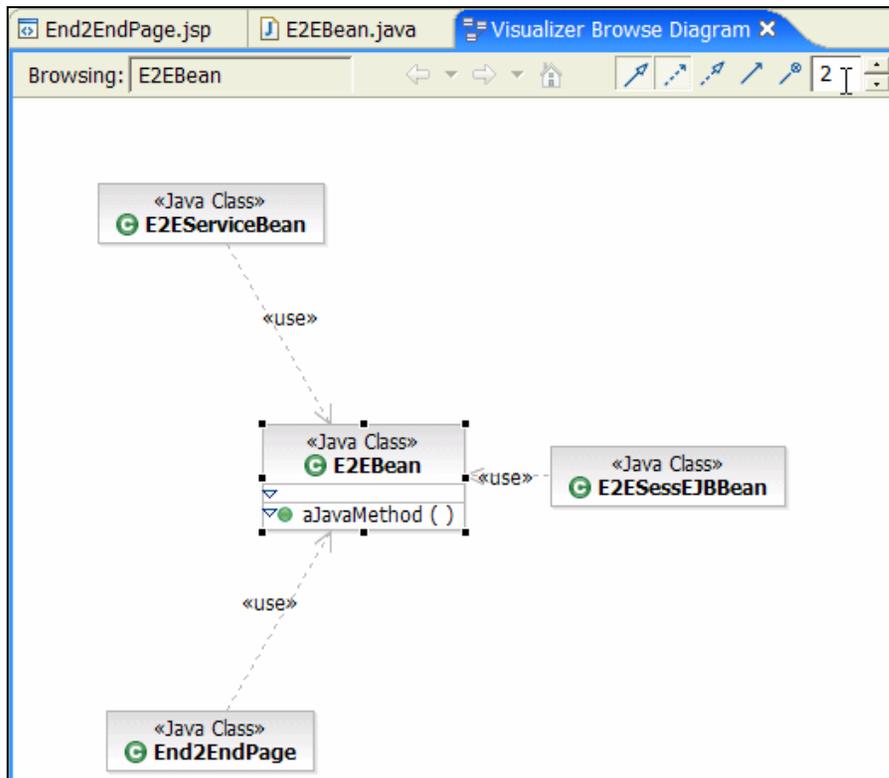
# End-2-End Scenario – Soup-2-Nuts J2EE Build Tasks

- Solution Overview – Single JSP captures all build and run scenarios
- Java Beans – Create, code, test, debug and JSF integration
- **Dynamic Development** – Changing code on-the-fly and debugging
- UML Visualization – Exploring application artifacts
- Performance Profiling – Analyzing performance with graphics and reports

Type of Calls	Input>>Submit>>Results	Diagrams
<b>Java:</b> o Java Bean o Bean Web Service	<div style="border: 1px solid gray; padding: 5px;"> <div style="border-bottom: 1px solid gray; display: flex; justify-content: space-between;"> <span>Java Bean</span> <span>Bean Web Service</span> </div> <div style="padding: 5px;">           SomeData: <input type="text"/>  <input type="button" value="Submit"/> </div> <div style="padding: 5px;">           Results: [E2EServiceBean.aWSJavaMethod] Processed &gt;&gt;                      [E2EBean.aJavaMethod] Processed &gt;&gt; [BP #2]         </div> </div>	<div style="border: 1px solid gray; padding: 5px;"> <pre>           graph TD             End2EndPage --&gt; E2EBean           </pre> </div> <div style="border: 1px solid gray; padding: 5px; margin-top: 5px;"> <pre>           graph TD             End2EndPage --&gt; WebServiceProxy             WebServiceProxy --&gt; E2EBean           </pre> </div>
<b>EJBs:</b> o Session EJB o EJB Web Service o CMP SDO Facade	<div style="border: 1px solid gray; padding: 5px;"> <div style="border-bottom: 1px solid gray; display: flex; justify-content: space-between;"> <span>Session EJB</span> <span>EJB Web Service</span> <span>CMP SDO Facade</span> </div> <div style="padding: 5px;">           SomeData: <input type="text"/>  <input type="button" value="Submit"/> </div> <div style="padding: 5px;">           Results: [E2ESessEJB.anEJBMethod] Processed &gt;&gt;                      [E2EBean.aJavaMethod] Processed &gt;&gt; [BP #3]         </div> </div>	<div style="border: 1px solid gray; padding: 5px;"> <pre>           graph TD             End2EndPage --&gt; EJBProxy             EJBProxy --&gt; E2ESessEJB             E2ESessEJB --&gt; E2EBean           </pre> </div> <div style="border: 1px solid gray; padding: 5px; margin-top: 5px;"> <pre>           graph TD             End2EndPage --&gt; WebServiceProxy             WebServiceProxy --&gt; E2ESessEJB             E2ESessEJB --&gt; E2EBean           </pre> </div>

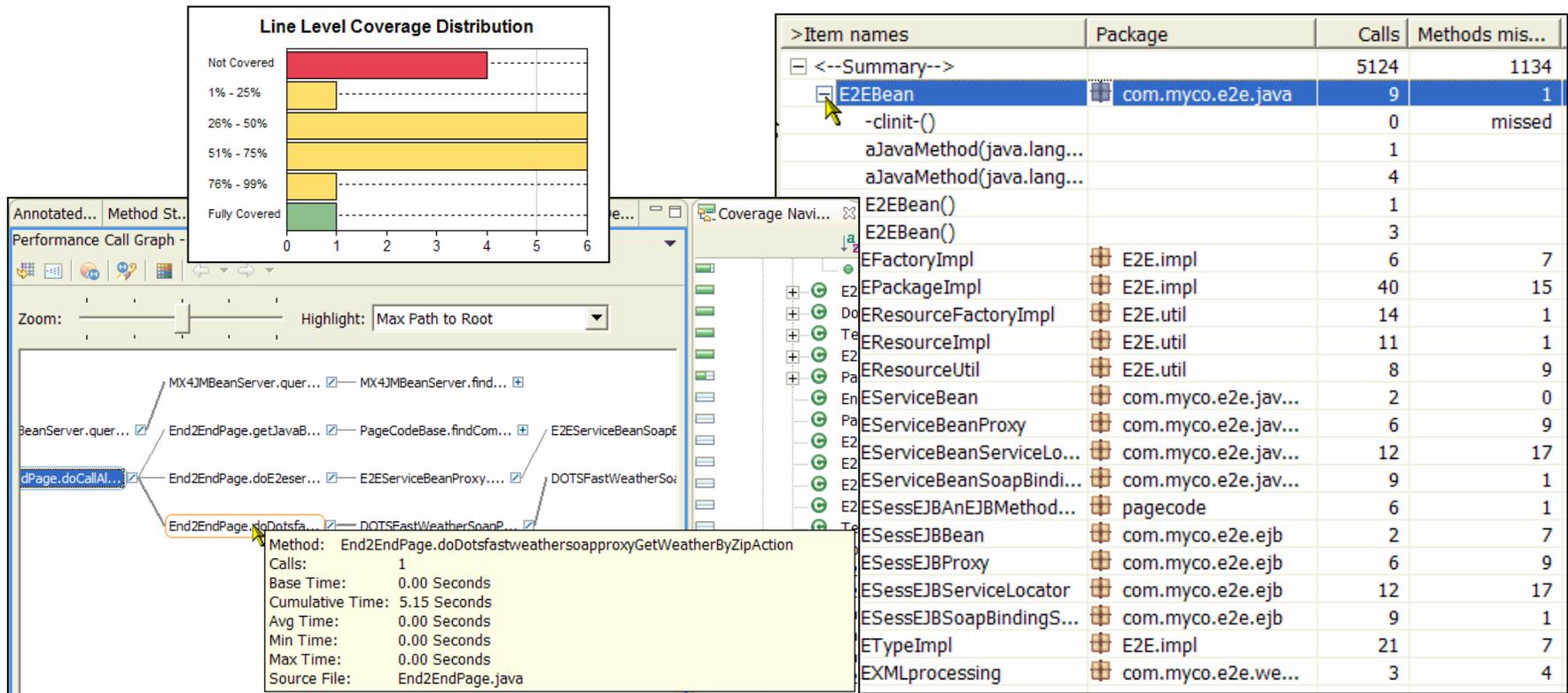
# End-2-End Scenario – Soup-2-Nuts J2EE Build Tasks

- Solution Overview – Single JSP captures all build and run scenarios
- Java Beans – Create, code, test, debug and JSF integration
- Dynamic Development – Changing code on-the-fly and debugging
- **UML Visualization** – Exploring application artifacts
- Performance Profiling – Analyzing performance with graphics and reports



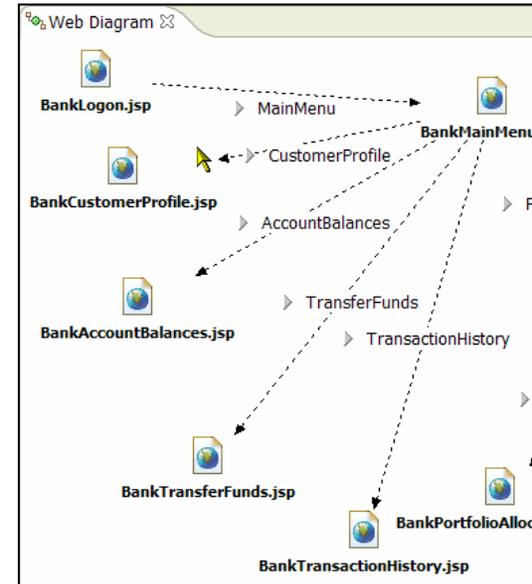
# End-2-End Scenario – Soup-2-Nuts J2EE Build Tasks

- Solution Overview – Single JSP captures all build and run scenarios
- Java Beans – Create, code, test, debug and JSF integration
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application artifacts
- **Performance Profiling** – Analyzing performance with graphics and reports



# Banking & Finance Scenario – Applying End-2-End tasks

- Logon and Main Menu – Logon, user profile managed bean and SDOs
- Transfer Funds – Web Services, SDOs and dynamically populated combo boxes
- Portfolio Management – Asset allocation, performance graphs and market simulation
- Web Site Design – Web site look-n-feel, hierarchy and navigation
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application code with UML Diagrams
- Code Review – Best practices, code proposals and extensible rules
- Profiling Performance – Analyzing performance with graphics and reports



# Banking & Finance Scenario – Applying End-2-End tasks

- Web Diagram, Logon and Main Menu – Logon, user profile managed bean and SDOs
- Transfer Funds – Web Services, SDOs and dynamically populated combo boxes
- Portfolio Management – Asset allocation, performance graphs and market simulation
- Web Site Design – Web site look-n-feel, hierarchy and navigation
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application code with UML Diagrams
- Code Review – Best practices, code proposals and extensible rules
- Profiling Performance – Analyzing performance with graphics and reports

The screenshot shows an IDE interface with three main panels:

- Project Navigator:** Displays a project structure for 'JSFWeb' including 'Web Site Navigation', 'Web Deployment Descriptor', 'Java Resources', 'WebContent', 'images', and 'CustomerAdmin.jsp'. The 'J2EE Hierarchy' tab is active.
- Page Data:** Shows the 'sessionScope' attribute with a 'customerNumber' property of type 'java.lang.String' selected.
- Web Browser:** Displays the rendered 'Logon.jsp' page. It features a header with a colorful banner and the word 'Logon' in red. Below is a form with a 'Customer ID#' field (containing 'abc') and a 'Password' field. A tooltip points to the 'Customer ID#' field with the text: 'Drop here to bind "customerNumber" to the control "text1"'. The 'Customer ID#' field is highlighted with a dashed border and a yellow background.

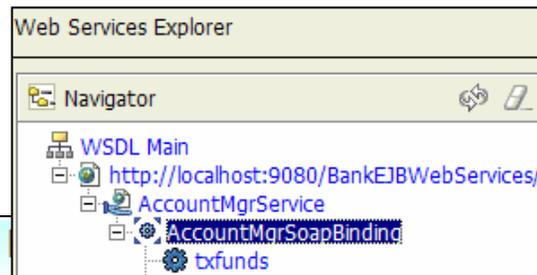
The screenshot shows a web browser displaying the 'Logon' page. The page has a header with a colorful banner and the word 'Logon' in red. Below the header is a form with two input fields: 'Customer ID#' and 'Password'. The 'Customer ID#' field contains the text 'abc' and is highlighted with a dashed border and a yellow background. A tooltip points to the 'Customer ID#' field with the text: 'Drop here to bind "customerNumber" to the control "text1"'. The 'Customer ID#' field is highlighted with a dashed border and a yellow background.

Below the browser window is an 'Attributes' panel for the selected 'h:inputText' component. The panel shows the following attributes:

- Basics:** Format, Access, Validation
- Id:** text1
- Value:** #{sessionScope.customerNu

# Banking & Finance Scenario – Applying End-2-End tasks

- Logon and Main Menu – Logon, user profile managed bean and SDOs
- **Transfer Funds** – Web Services, SDOs and dynamically populated combo boxes
- Portfolio Management – Asset allocation, performance graphs and market simulation
- Web Site Design – Web site look-n-feel, hierarchy and navigation
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application code with UML Diagrams
- Code Review – Best practices, code proposals and extensible rules
- Profiling Performance – Analyzing performance with graphics and reports



### Web Service Discovery

[Web Services from a UDDI Registry](#)  
Search for Web services by name or key in a public or private UDDI registry.

[Web Services from a known URL](#)  
Search for Web services using a URL to a document.

[Web Services from your workspace](#)  
Search for Web services deployed and running in my current workspace.

A screenshot of a web browser window showing the 'Bank Transfer Funds' page. The page title is 'Bank Transfer Funds'. Below the title, it says 'Charlie, you may transfer funds between your accounts below:'. There are two buttons: 'Account Balances' and 'Return to Main Menu'. Below these are three input fields: 'Source Account #' with a dropdown menu showing '8', 'Target Account #' with a dropdown menu showing '7', and 'Amount \$' with a text input containing '1,000'. At the bottom, there is a 'Transfer Funds' button and a green box containing 'Transfer Details: [From/To Acct Bal: \$100000.00 / \$10000.00]'.

# Banking & Finance Scenario – Applying End-2-End tasks

- Logon and Main Menu – Logon, user profile managed bean and SDOs
- Transfer Funds – Web Services, SDOs and dynamically populated combo boxes
- **Portfolio Management** – Asset allocation, performance graphs and market simulation
- Web Site Design – Web site look-n-feel, hierarchy and navigation
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application code with UML Diagrams
- Code Review – Best practices, code proposals and extensible rules
- Profiling Performance – Analyzing performance with graphics and reports

**Bank Portfolio Management**

Arthur, you may analyze your portfolio below:

[Return to Main Menu](#)

*Portfolio Analysis*

- Re-Allocate Funds Across Portfolio
- Re-Calculate Portfolio Balances
- Graph Portfolio Performance

**Bank Portfolio Allocation**

Arthur, you may re-allocate your funds below:

[Update](#)

F01\_bledned\_income\_plus\_life: 6

F02\_bledned\_conservative\_life: 4

F03\_bledned\_moderate\_life: 5

F04\_bledned\_aggressive\_life: 5

**Bank Portfolio Performance**

Arthur, the graphical performance of your funds appears below:

[Return to Manage Your Portfolio](#)

**Portfolio Performance**

Chart Type: Bar Series: Show All Seri...

Fund #	Fund Type	Folio %	Invested Balance	Risk	Return	Projected Balance
101	Blended - Income Plus Life Strategy	5	\$25,000.00	1	2.125%	\$25,500.00
102	Blended - Conservative Life Strategy	5	\$25,000.00	2	4.25%	\$26,000.00
103	Blended - Moderate Life Strategy	5	\$25,000.00	3	6.375%	\$26,500.00
104	Blended - Aggressive Life Strategy	5	\$25,000.00	4	8.425%	\$27,000.00
105	Money Market	5	\$25,000.00	1	2.125%	\$25,500.00
106	Bond - Stable Value	5	\$25,000.00	1	2.125%	\$25,500.00

**Portfolio Performance**

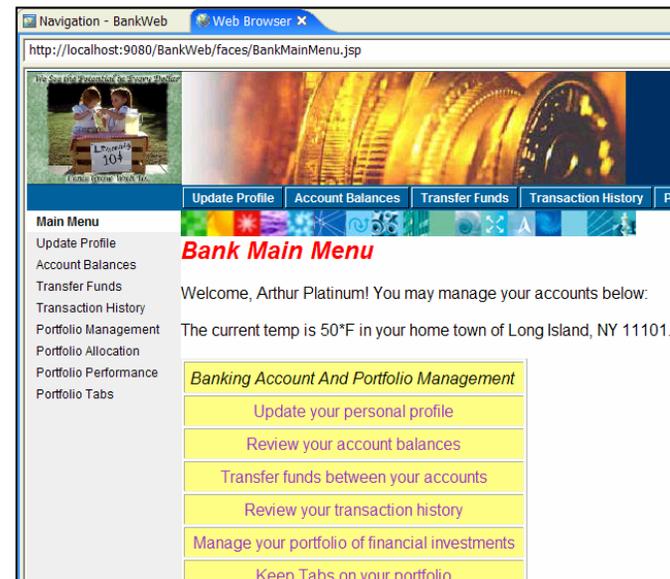
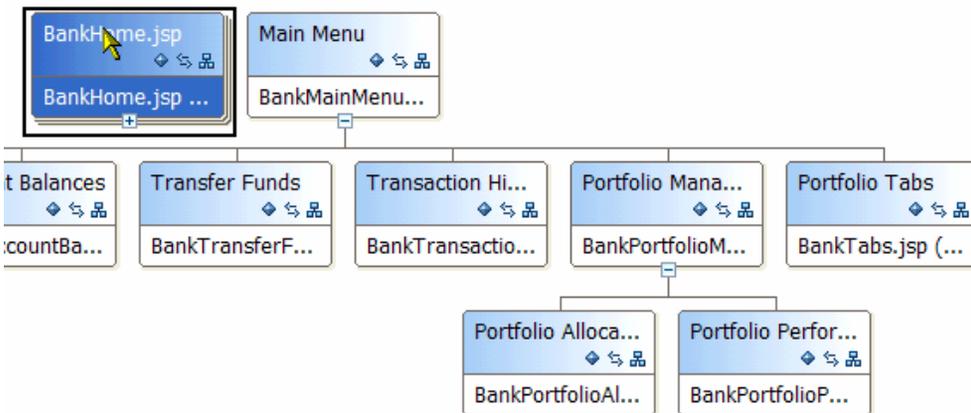
Chart Type: Line Series: Show All Seri...

**Portfolio Performance**

Chart Type: Pie Series: Invested

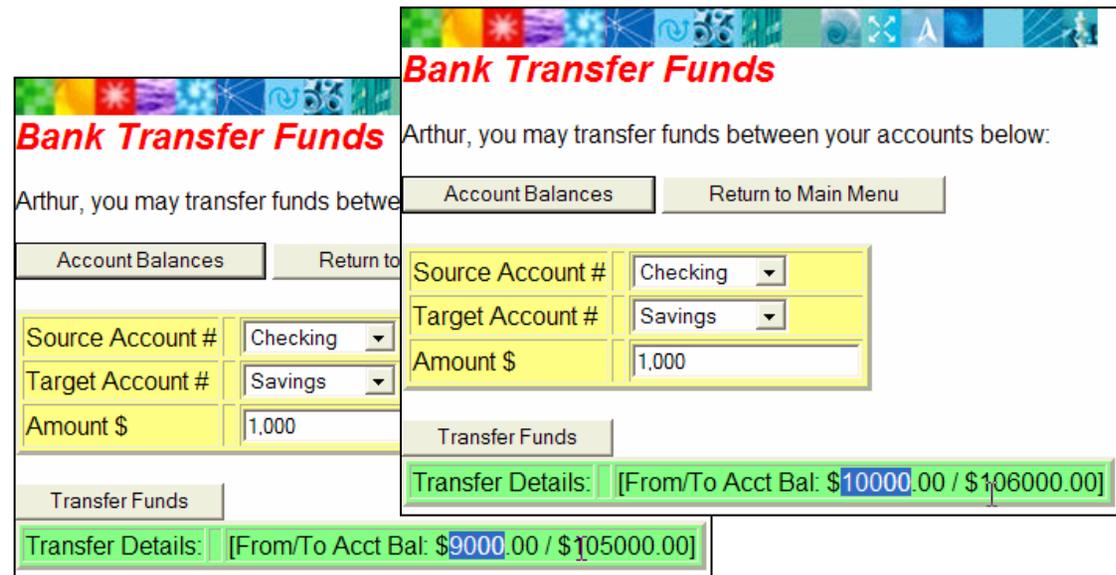
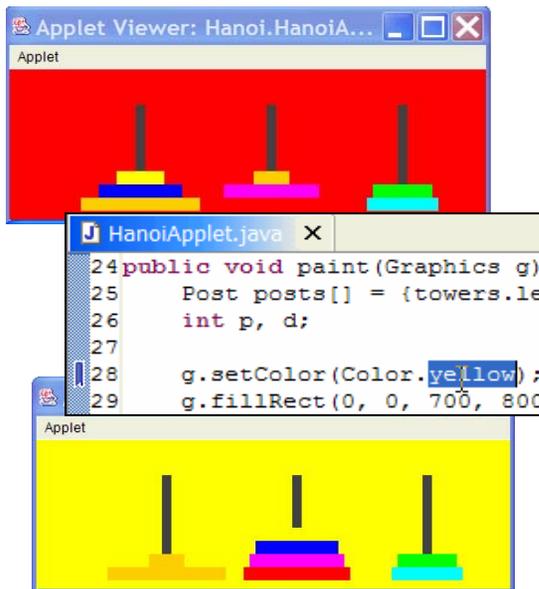
# Banking & Finance Scenario – Applying End-2-End tasks

- Logon and Main Menu – Logon, user profile managed bean and SDOs
- Transfer Funds – Web Services, SDOs and dynamically populated combo boxes
- Portfolio Management – Asset allocation, performance graphs and market simulation
- **Web Site Design** – Web site look-n-feel, hierarchy and navigation
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application code with UML Diagrams
- Code Review – Best practices, code proposals and extensible rules
- Profiling Performance – Analyzing performance with graphics and reports



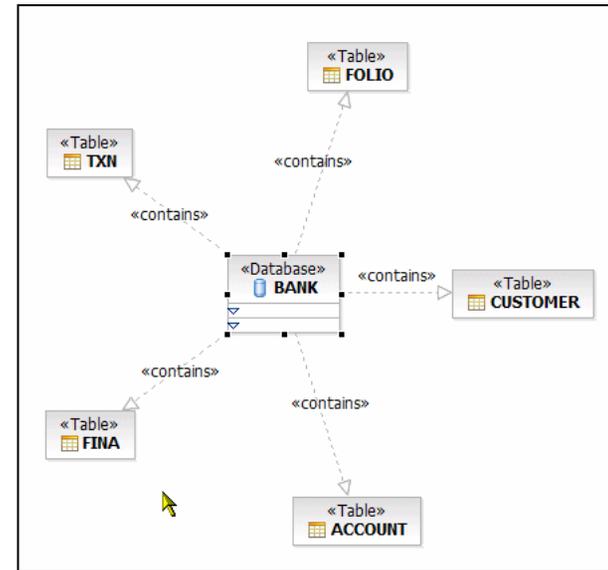
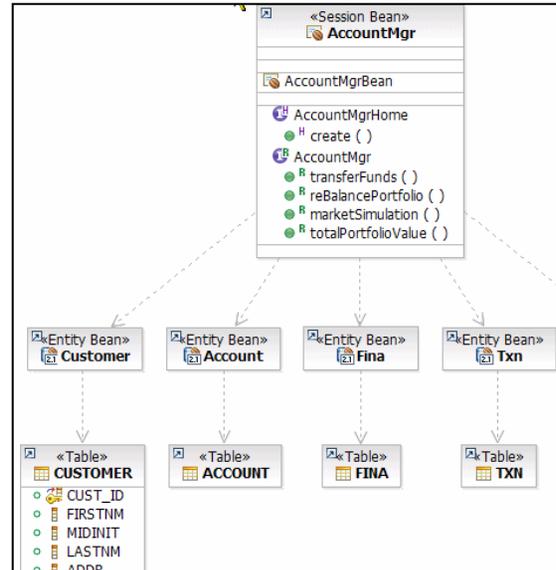
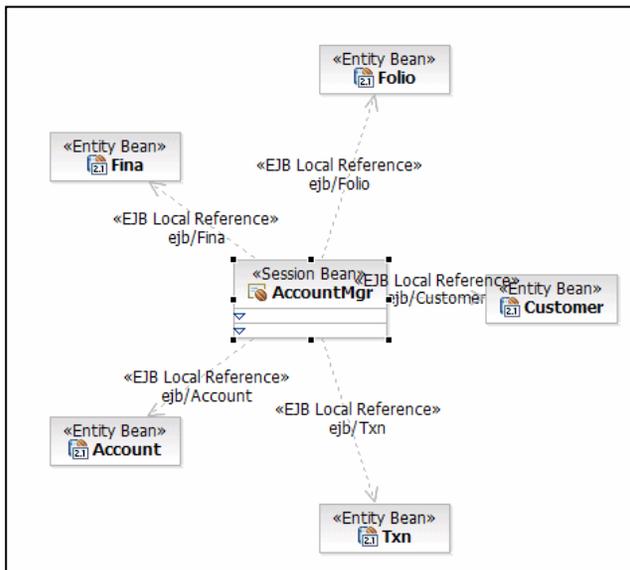
# Banking & Finance Scenario – Applying End-2-End tasks

- Logon and Main Menu – Logon, user profile managed bean and SDOs
- Transfer Funds – Web Services, SDOs and dynamically populated combo boxes
- Portfolio Management – Asset allocation, performance graphs and market simulation
- Web Site Design – Web site look-n-feel, hierarchy and navigation
- **Dynamic Development** – Changing code on-the-fly and debugging
- UML Visualization – Exploring application code with UML Diagrams
- Code Review – Best practices, code proposals and extensible rules
- Profiling Performance – Analyzing performance with graphics and reports



# Banking & Finance Scenario – Applying End-2-End tasks

- Logon and Main Menu – Logon, user profile managed bean and SDOs
- Transfer Funds – Web Services, SDOs and dynamically populated combo boxes
- Portfolio Management – Asset allocation, performance graphs and market simulation
- Web Site Design – Web site look-n-feel, hierarchy and navigation
- Dynamic Development – Changing code on-the-fly and debugging
- **UML Visualization** – Exploring application code with UML Diagrams
- Code Review – Best practices, code proposals and extensible rules
- Profiling Performance – Analyzing performance with graphics and reports



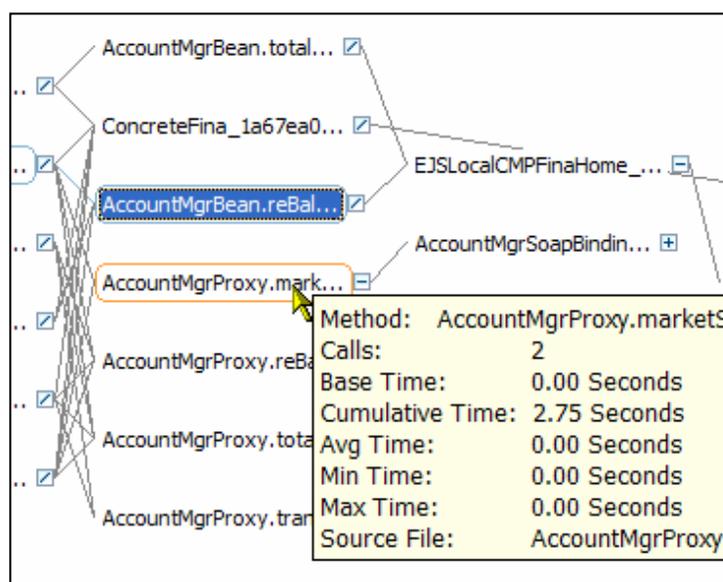
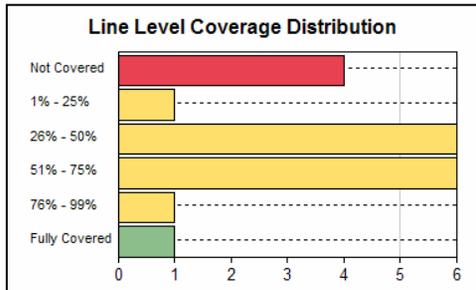
# Banking & Finance Scenario – Applying End-2-End tasks

- Logon and Main Menu – Logon, user profile managed bean and SDOs
- Transfer Funds – Web Services, SDOs and dynamically populated combo boxes
- Portfolio Management – Asset allocation, performance graphs and market simulation
- Web Site Design – Web site look-n-feel, hierarchy and navigation
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application code with UML Diagrams
- **Code Review** – Best practices, code proposals and extensible rules
- Profiling Performance – Analyzing performance with graphics and reports

The Performance rule category contains rules that enforce improving performance and reducing memory footprint in

# Banking & Finance Scenario – Applying End-2-End tasks

- Logon and Main Menu – Logon, user profile managed bean and SDOs
- Transfer Funds – Web Services, SDOs and dynamically populated combo boxes
- Portfolio Management – Asset allocation, performance graphs and market simulation
- Web Site Design – Web site look-n-feel, hierarchy and navigation
- Dynamic Development – Changing code on-the-fly and debugging
- UML Visualization – Exploring application code with UML Diagrams
- Code Review – Best practices, code proposals and extensible rules
- **Profiling Performance** – Analyzing performance with graphics and reports



**Callers**

Caller	Perc...	Calls
Thread: WebContainer : 5	21.53	1
Thread: WebContainer : 7	20.03	1
Thread: WebContainer : 2	19.11	1
Thread: WebContainer : 9	39.33	1

**Descendants**

Descendant
CustomerKey.CustomerKey
FolioKey.FolioKey
EJSLocalCMPFolioHome_6b6f4b12.findByPrimaryKe
EJSLocalCMPFolio_6b6f4b12.getFk_folio_acct_id
AccountKey.AccountKey
EJSLocalCMPAccountHome_955e9689.findByPrima
EJSLocalCMPAccount_955e9689.getBal
FinaKey.FinaKey
EJSLocalCMPFinaHome_1a67ea0b.findByPrimaryKe
EJSLocalCMPFolio_6b6f4b12.getF01_blended_inco
EJSLocalCMPFina_1a67ea0b.setFolio_percent



# Financial Advisor Dashboard – Portlets and Portal Design

- Tabbed Panel Portlets – Bank Customers and Portfolio Management Portlets
- Click-2-Action – Bank Customer and Portfolio SDOs and C2A inter-portlet communication
- Portfolio Summary – Display portfolio summary information using SDOs
- Bank DB CMP EJBs Mapping – Generate CMP EJBs from BANK database tables
- Account Manager Session EJB – Portfolio Management and Dashboard logic
- Portfolio Management – Asset allocation, performance graphs and market simulation
- Dashboard – SDO display of tabular and graphical dashboard
- Portal Site Design – Themes, skins, pages and portal site design



# Financial Advisor Dashboard – Portlets and Portal Design

My Portal
Administration
Edit my profile
Log out

New Page
Edit Page
Assign Permissions

Banking
Finance
My Favorites

**Financial Advisor Dashboard**

Loans

BankCustomers portlet

## Financial Advisor's Dashboard

Customers
Portfolios Dashboard

Cust #	First Name	Last Name	City	State	Zip	Work Phone	\$ Invested	\$ Projected	% Rate of Return
1	Arthur	Platinum	Long Island	NY	11101	773-123-1111	\$500,000.00	\$527,751.15	5.55%
2	Betsy	Golden	Arlington	VA	22202	773-123-2222	\$500,000.00	\$527,568.75	5.514%
3	Charlie	Silver	Fort Lauderdale	FL	33303	773-123-3333	\$500,000.00	\$526,688.75	5.338%
4	Denise	Copper	Burghill	OH	44404	773-123-4444	\$500,000.00	\$510,625.00	2.125%

### Customer Portfolios Performance

BankPortfolios portlet

## Banking > Finance > Portfolio Management

Customer #

Analyze Customer Portfolio

Portfolio details for **Arthur Platinum** appear below:

Invested: \$500,000.00 | Projected: \$527,751.15 | Rate of return: 5.55%

Portfolio
Allocation
Graphics
Market Simulation

Fund Acct #	Fund Name	% of Folio	\$ Invested	Risk	% Return	\$ Projected
101	Blended - Income Plus Life Strategy	5%	\$25,000.00	1	2.124%	\$25,531.00
102	Blended - Conservative Life Strategy	4%	\$20,000.00	2	4.249%	\$20,849.80
103	Blended - Moderate Life Strategy	6%	\$30,000.00	3	6.374%	\$31,912.20
104	Blended - Aggressive Life Strategy	7%	\$35,000.00	4	8.424%	\$37,948.40
105	Money Market	5%	\$25,000.00	1	2.124%	\$25,531.00
106	Bond - Stable Value	3%	\$15,000.00	1	2.125%	\$15,318.75
107	Bond - Total Market	5%	\$25,000.00	2	6.375%	\$26,593.75

