



IBM Software Group

# Building JavaServer Faces Applications

***St. Louis Java User Group***

**Rational** software



**ON DEMAND BUSINESS™**

Tim Saunders  
ITS  
Rational Software

[tim.saunders@us.ibm.com](mailto:tim.saunders@us.ibm.com)

© 2005 IBM Corporation

# Agenda

- JSF Vision
- JSF Overview
- IBM Rational Application Developer v6.0
- Build a JSF application

## JSF Vision – Bring RAD to J2EE

- ... ease the burden of writing (UI) applications
- ... accessible to a wide variety of developer types

### **Mission:**

Deliver an easier, visual way to build, test and deploy rich Web applications that leverage the J2EE runtime environment with minimal coding

- ... JSF is designed to be tooled ...

### **Eclipse Plugins for JSF**

**IBM Rational Web Developer v6.0**

**IBM Rational Application Developer v6.0**

**IBM WebSphere Studio Application Developer v5.1.2**

## J2EE Technology before JSF

- Servlet + JSP using MVC pattern
  - ▶ JSP 0.92 spec patterns: “Model 2”
- Good:
  - ▶ Separation of concerns
    - Servlet is Controller
    - JSP is View
  - ▶ Potential for reuse
  - ▶ Comparative ease of maintenance
- Bad:
  - ▶ Repetitive coding – especially in controller

## J2EE Technology before JSF

- Struts
  - ▶ Framework for controller and view
- Good:
  - ▶ Repetitive work in config and framework
  - ▶ Widely used (e.g. WAS 5.x console)
  - ▶ Studio Tooling
- Bad:
  - ▶ Still complex
  - ▶ Low productivity compared to RAD tools
  - ▶ Lack of flexibility in View

# JavaServer Faces (JSF)

- Standards-based Web Application framework (JSR 127)
  - ▶ Plug-and-play other JSF components easily
  - ▶ Library of User Interface components
  - ▶ Extensible
  - ▶ IDEs which support JSF will greatly simplify Web Application development
- Targeted for Web Application developers with little Java background
  - ▶ Reduces the skill level required to develop rich Web applications
- Event driven architecture
  - ▶ Server-Side Rich UI components respond to client events
- UI components are decoupled from its rendering
  - ▶ Allows for other technology (e.g. WML, etc) to be used
- Designed for use with tools to increase productivity
  - ▶ Reduces the skill level required to develop complex Web Application
  - ▶ Eliminates much of the hand-coding involved with integrating web applications into back-end systems

## JSF – A Web Framework

JavaServer Faces technology **simplifies** building user interfaces for JavaServer applications ...

developers of **varying skill levels** can quickly and easily build web applications ...

assembling reusable UI **components** in a page

- JSR 127 – JSF specification v1.1
  - ▶ JSF 1.2 in progress (JSR 252)
  - ▶ JSP 2.1 (JSR 245) will align with JSF
- JSF spec lead was the Struts architect



# Framework for Developing Apps

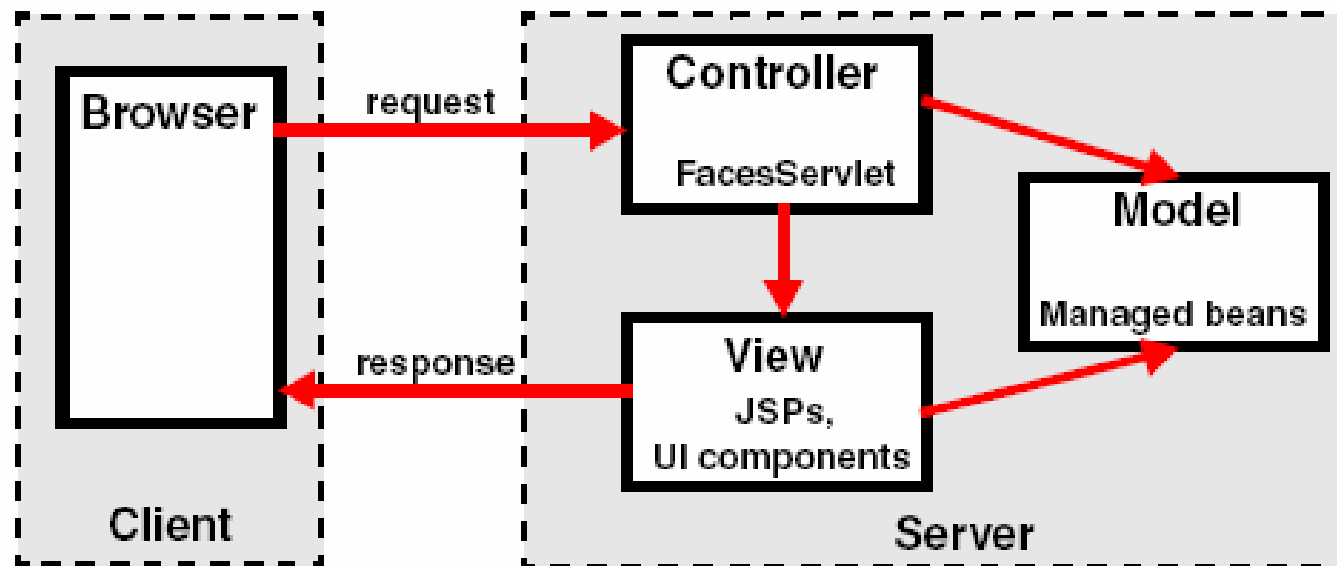
- User Interface Development
  - ▶ Direct binding of UI components to model data
  - ▶ Extensive libraries of prebuilt UI components
- Navigation
  - ▶ Layer of separation between business logic and the UI
  - ▶ Flexible rules control page flow
- Session and object management
  - ▶ Manages model data throughout lifecycle
- Validation and error feedback
  - ▶ Direct binding of reusable validators to UI components
  - ▶ Simplified error messaging and feedback to the user
- Internationalization
  - ▶ Number, currency, time and data formatting



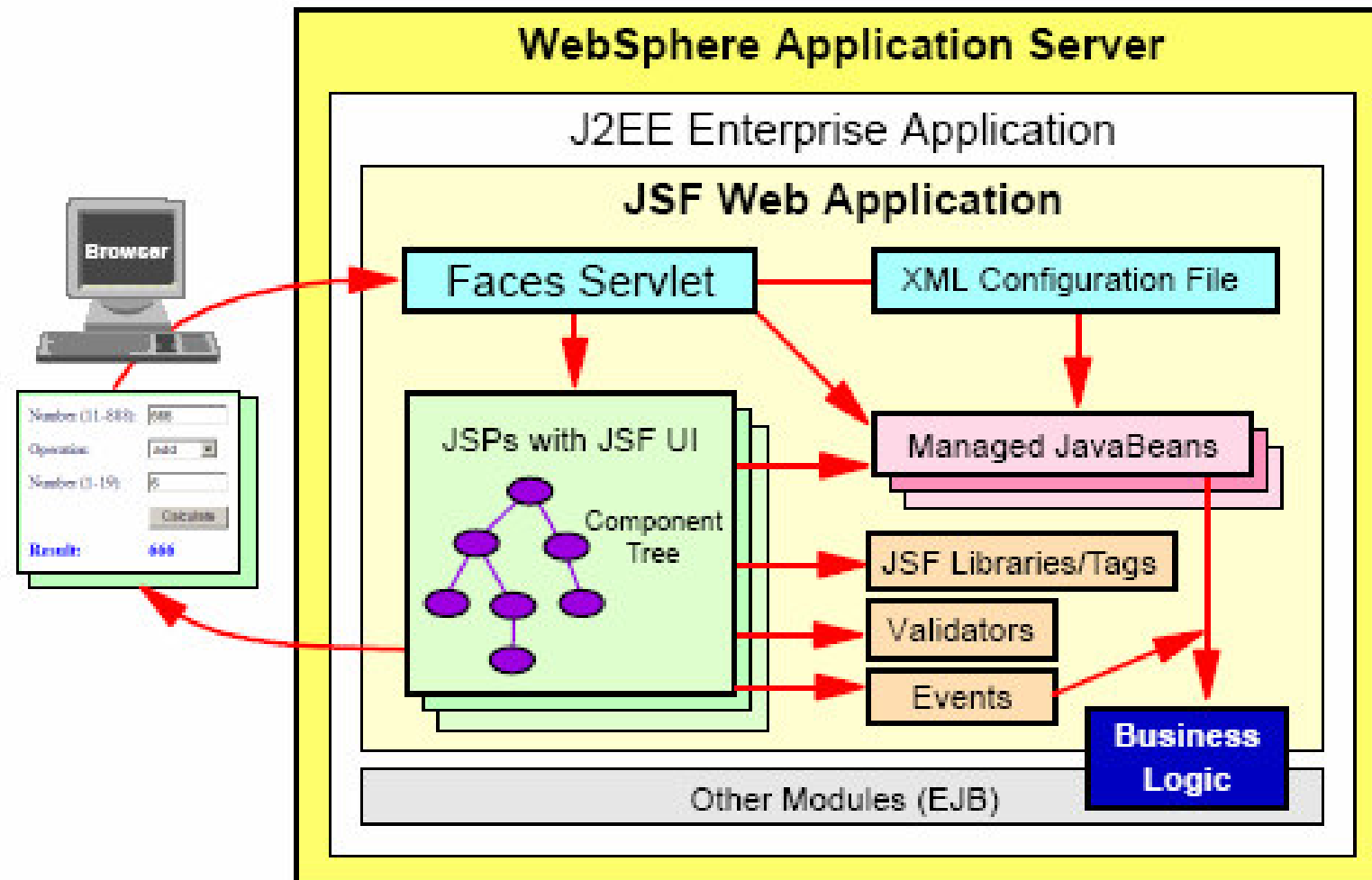


## Uses MVC architecture

- Model – Managed beans interface with reusable business logic components or external systems
- View – JSPs created by combining model data with predefined and custom-made UI components
- Controller – FacesServlet drives navigation and object management



# Application Structure

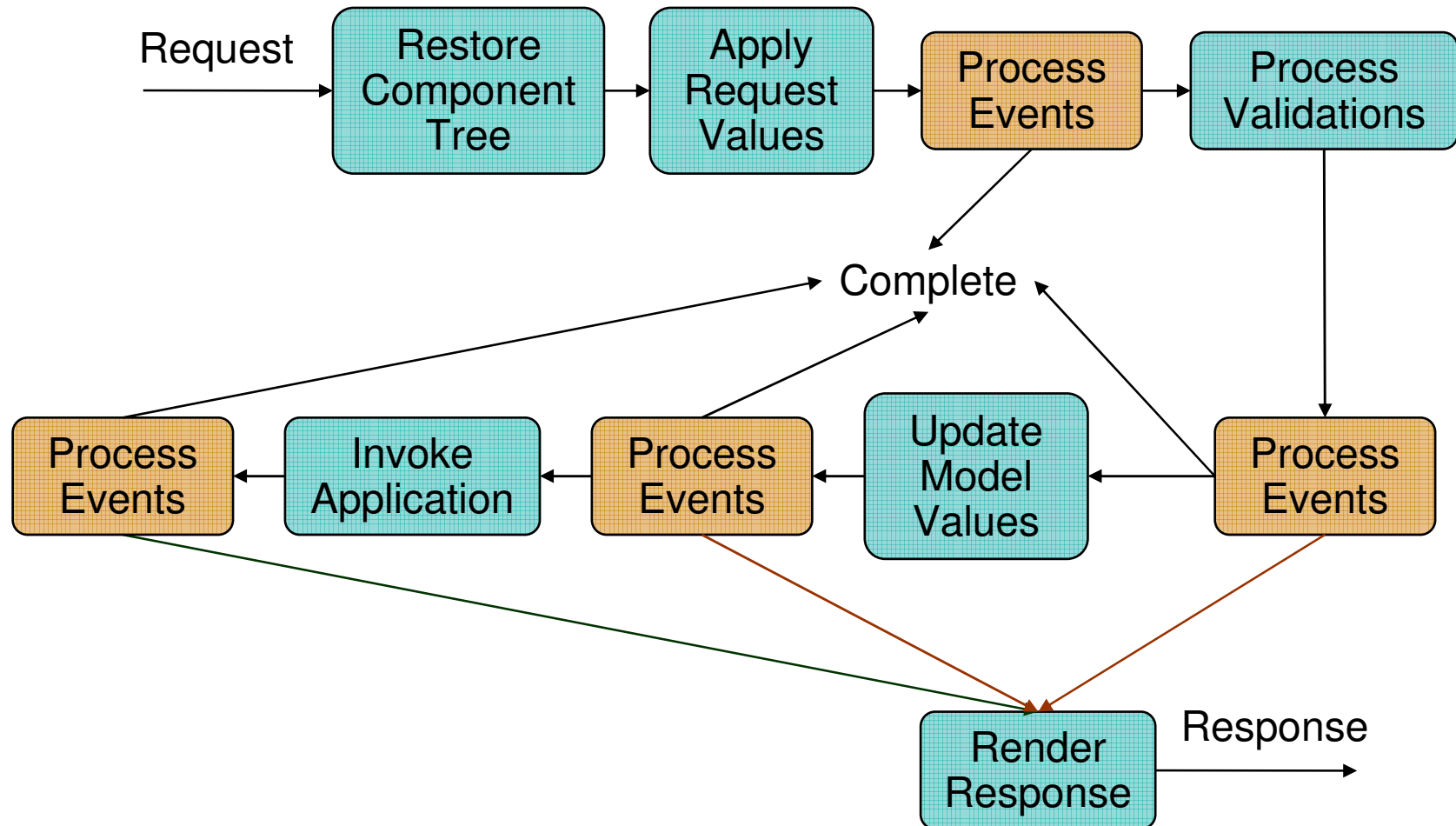


## Application Structure

- JSF Pages – JSPs built from JSF components; each component has a server-side class
- Faces Servlet – One servlet controls execution flow
- Configuration file – XML file (faces-config.xml) that contains navigation rules, validators and managed beans
- Tag libraries
- Validators – Java classes to validate content of JSF components
- Managed beans – Hold data from JSF components; passed between business logic and UI; JSF moves data between managed beans and UI components
- Events – Java code executed in the server for events; event handling passes managed beans to business logic



## FacesServlet Request Life Cycle



## JSF and Struts

- Some similarities
  - ▶ Built on JavaServer Page and Servlet technology
  - ▶ Add custom tag libraries
  - ▶ Support page navigation
  - ▶ Use Model View Controller architecture
  - ▶ Support internationalization
- Some differences
  - ▶ Struts focuses on the Controller dispatching business logic
  - ▶ JSF focuses on the View (User Interface)
  - ▶ JSF offers more user interface features since UI components are separate from their rendering (Struts is HTML centric)
- JSF will not replace Struts immediately
- Lead author of Struts is one of the specification leaders for JSF
  - ▶ Lead author has written a library to use Struts and JSF together



## Struts vs Faces Comparison

	Struts	Faces
Page Navigation	Struts-config.xml	Faces-config.xml
Form & GUI Processing	ActionForms, Actions	Session & object lifecycle, listener scripts
Visual Controls	Extended JSP Tag Libraries	Standard or custom UI components with events
Validators	Framework and ActionForms	Framework and predefined validators
Data Model	Actions	Service Data Objects, Data Access Beans



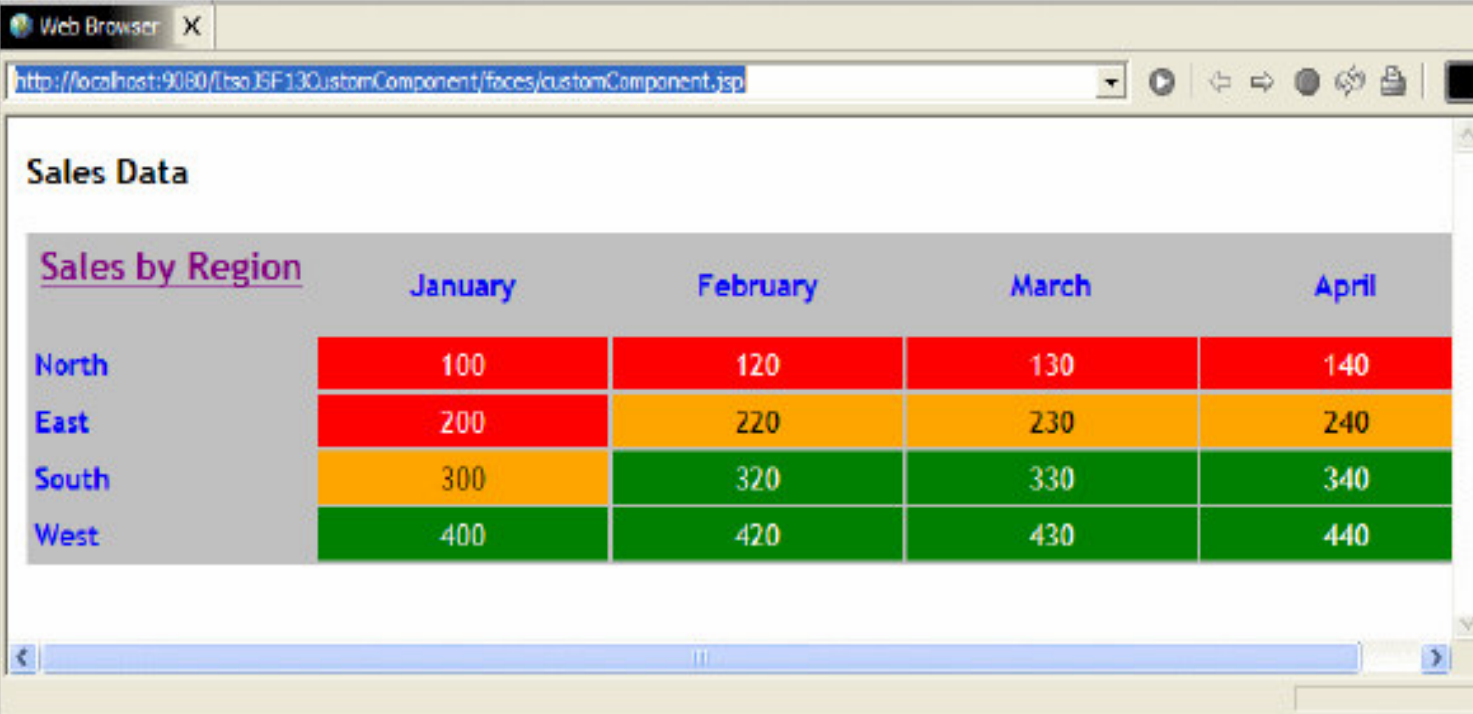
## Faces HTML UI Component Tags

- Check Box
- Check Box Group
- Combo Box
- Command Button
- Command Hyperlink
- Component Label
- Data Table
- Display Errors
- File Upload
- Form
- Grid Panel
- Group Box Panel
- Hidden Field
- Image
- Inline Message
- Input
- Input Text Area
- Input Password
- Label
- Link Action
- Media Player
- Message List
- List Box
- Output
- Output Formatted Text
- Panel Menu Bar
- Page Fragment Box
- Radio Button Group
- Rich Text Area
- Tabbed Panel



## JSF Extensibility - component

- Create new UI components
  - ▶ See JSF Redbook for Traffic Light example



Web Browser X

http://localhost:9080/TrsaJSF13CustomComponent/faces/customComponent.jsp

**Sales Data**

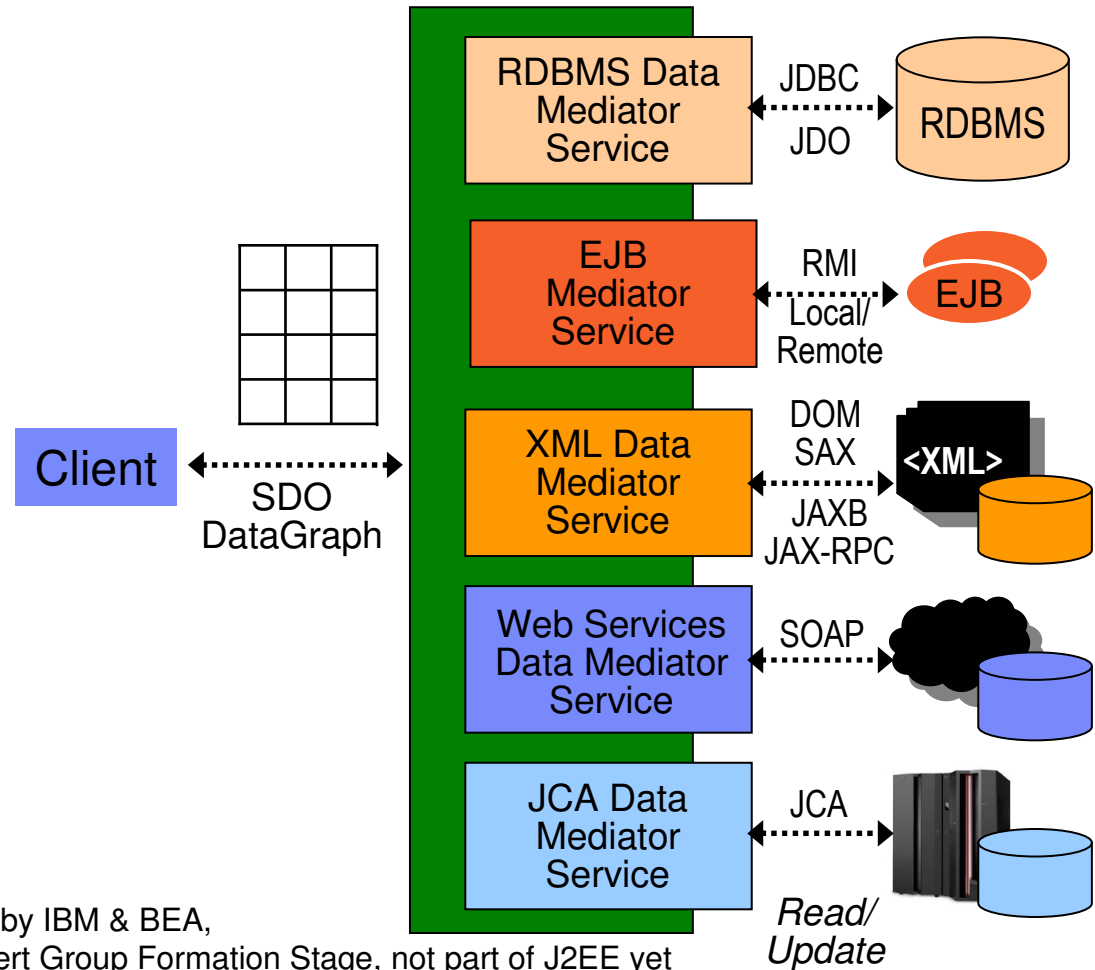
<u>Sales by Region</u>	January	February	March	April
North	100	120	130	140
East	200	220	230	240
South	300	320	330	340
West	400	420	430	440



# 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

# Faces Client Components

- Library of rich user interface controls for the Web Developer
- Easily develop rich interactive User Interfaces in Web Applications (without using plug-ins)
- Extends JavaServer Faces, allowing a developer to quickly create highly interactive Web pages that run client-side
  - ▶ Thin client: low maintenance (no installed client code to upgrade)
  - ▶ More responsive web applications
  - ▶ Reduced bandwidth (fewer round trips to the server)
- Components:
  - ▶ Data grid, tabbed panel, rich text editor, input formatters, tree control, chart (bar/line/pie), client-side model object



# Faces Client Components

treeest.html - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Copy Paste

Links Customize Links Free Hotmail IBM Business Transformation Homepage IBM Business Transformation IBM Home Page IBM Internal Help Homepage

Address D:\bobz\Odyssey\JSControls\Tree\treetest.html

Current Account Account # - Chen

user

- portfolio
  - name
  - position
    - symbol
      - MACR
        - price 49.47
        - quantity 108
        - total 5342.76
      - position
      - position

symbol	price	quantity
MACR	49.47	108
JP	43.94	213
JSDO	48.1	281

**Tree Control**

**Tabbed Panel**

D:\bobz\Odyssey\JSControls\Tab\tab\_sample.htm - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Copy Paste

Links Customize Links Free Hotmail IBM Business Transformation Homepage IBM Business Transformation IBM Home Page IBM Internal Help Homepage

Address D:\bobz\Odyssey\JSControls\Tab\tab\_sample.htm

Employee Office Address

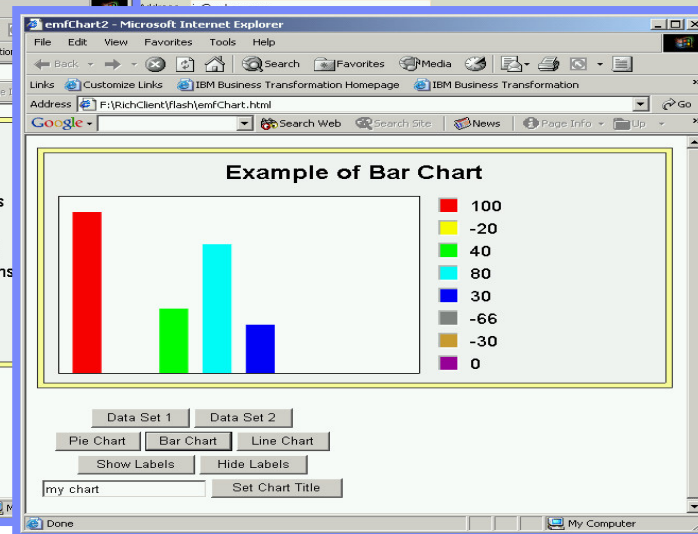
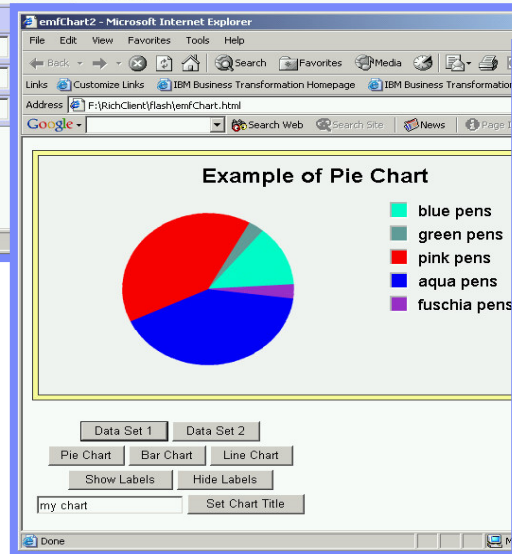
First Name Gegi LastName Thomas

Display Name Mr. Gegi Thomas

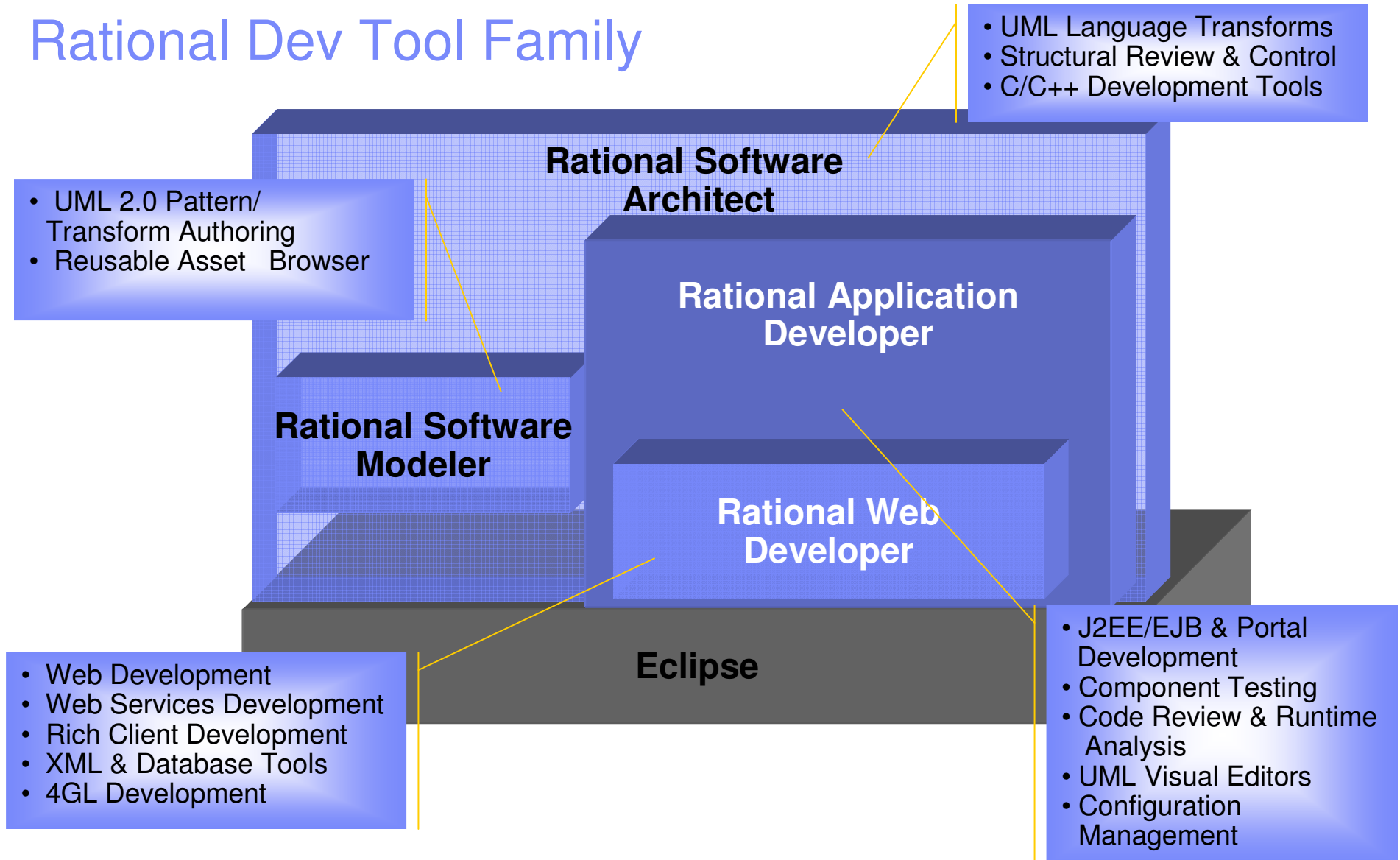
Organization Services

Consultant

**Chart**

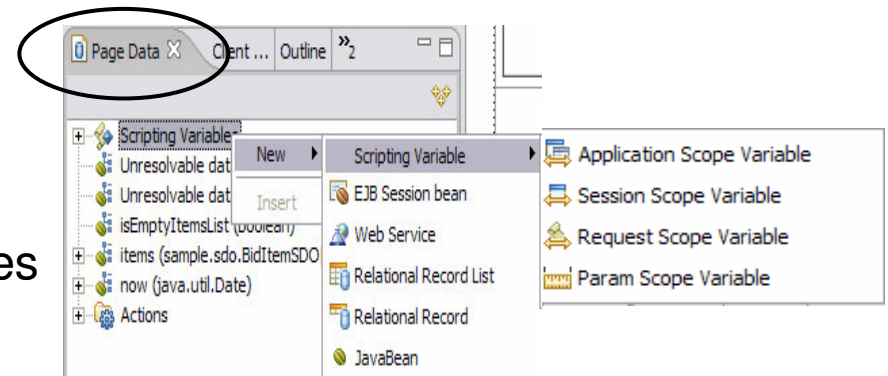
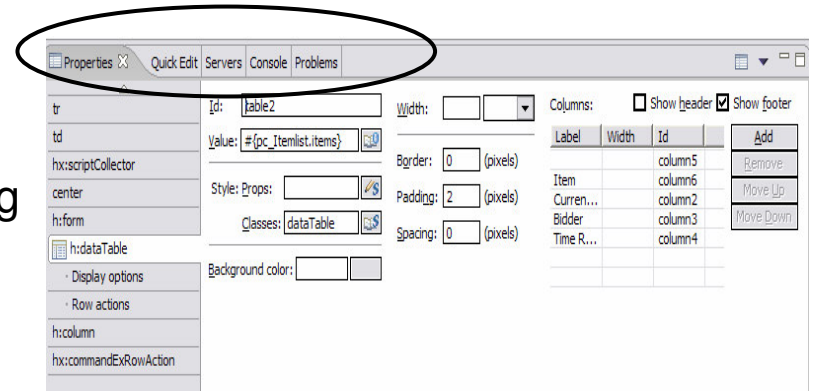


# Rational Dev Tool Family



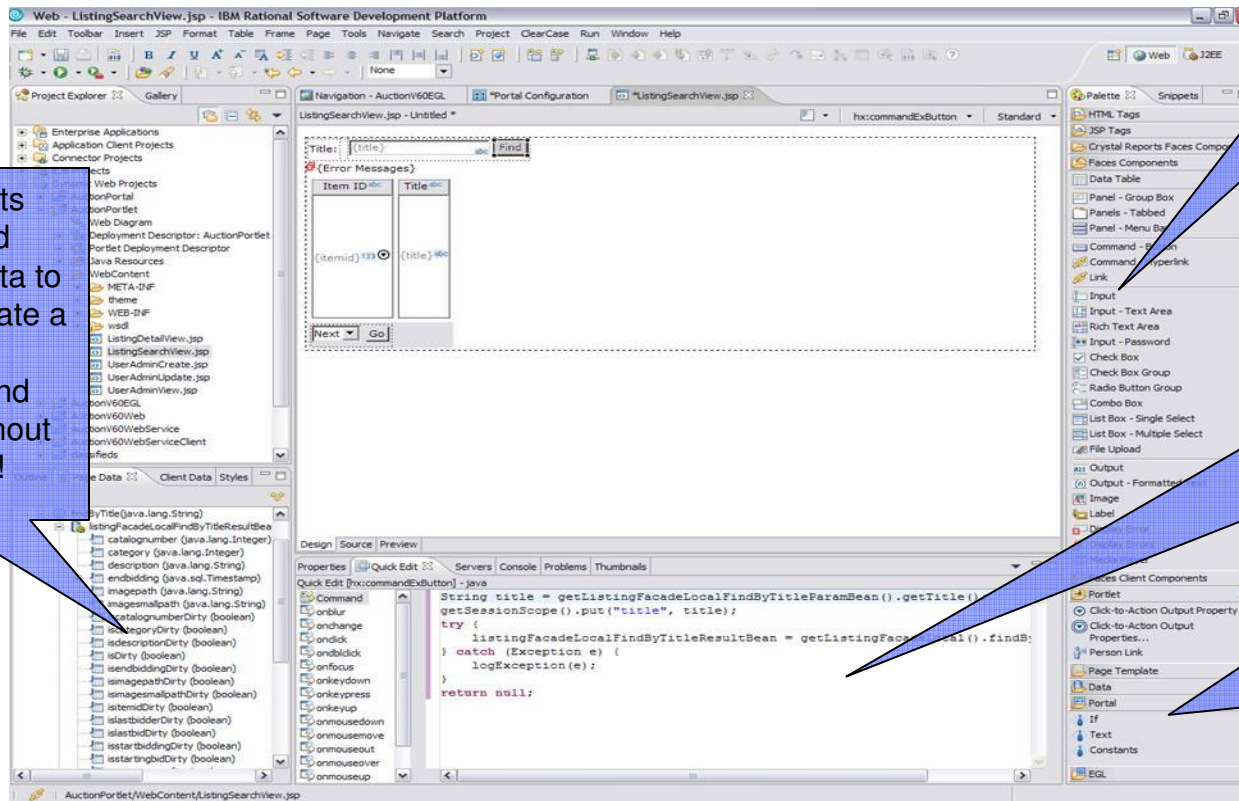
# JSF support in Rational Developer Products

- Visual development of JSF-based pages using Page Designer
  - ▶ Built-in Component Property Editor
  - ▶ Built-in tools simplify/automate event handling
  - ▶ Built-in tools simplify page navigation
- Web Diagram Tool to visually layout the flow of a JSF-based Web Application
- Automatic code generation for data validation, formatting and CRUD functions for data access
- Data abstraction objects for easy data connectivity (Service Data Objects)
  - ▶ Relational databases, EJBs, Web Services
  - ▶ Data objects can be bound easily to user interface components



# Visual Portlet Development

The Data View lets you configure and drag and drop data to automatically create a data bound UI. SDO accesses and updates data without writing any code!



Construct pages by dragging and dropping rich web components from a customizable Palette

The Quick Edit View allows scripting of client or server side events in Java or Java Script. Code assist writes the code for you!

Add Click-2-Action and People Awareness capability by dragging and dropping these components

## ▪ Design, Deploy, Debug, Test and Preview Portlets

- Portlet project wizard to create basic portlets, JSF portlets, and Struts portlets
- Web services client Portlet
- Portlet project support for the IBM portlet API and the JSR 168 portlet API
- Business Portlet Development with WPAI (5.1)—SAP, Seibel portlets
- Testing and debugging of portlets within the workbench using the WebSphere Portal Test Environment or a remote server
- Click-to-Action





## Additional Resources

- **JavaServer Faces Sites**

- ▶ <http://www.jsfcentral.com>
- ▶ <http://java.sun.com/j2ee/javaserverfaces/index.jsp>

- **Service Data Objects**

- ▶ <http://dev2dev.bea.com/technologies/commonj/sdo/index.jsp>
- ▶ <http://www-106.ibm.com/developerworks/library/j-sdo/>

- **Articles on IBM Rational Software developerWorks**

- ▶ <http://www-128.ibm.com/developerworks/library/j-jsf1/>
- ▶ <http://www-128.ibm.com/developerworks/java/library/j-jsf2/>
- ▶ [http://www-128.ibm.com/developerworks/websphere/techjournal/0401\\_barcia/barcia.html](http://www-128.ibm.com/developerworks/websphere/techjournal/0401_barcia/barcia.html)

- **IBM Redbooks - WebSphere Studio 5.1.2 JavaServer Faces and Service Data Objects**

- ▶ <http://www.redbooks.ibm.com/redpieces/abstracts/sg246449.html>

- **Craig McLanahan's web log and interview**

- ▶ <http://blogs.sun.com/roller/page/craigmcc/20040927>
- ▶ <http://www.theserverside.com/talks/videos/CraigMcClanahan2/interview.tss?bandwidth=dsl>

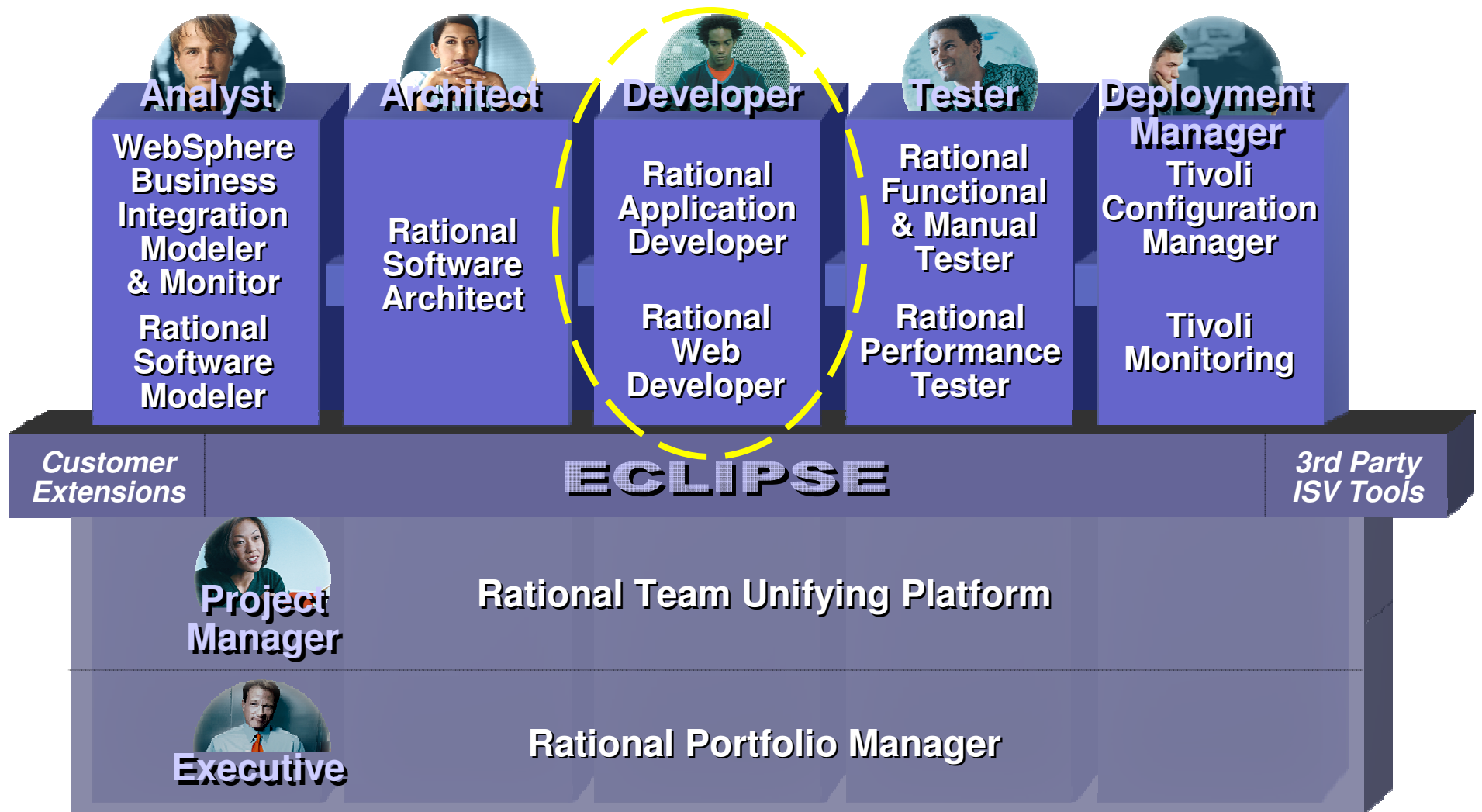


# Demo



Thank  
you

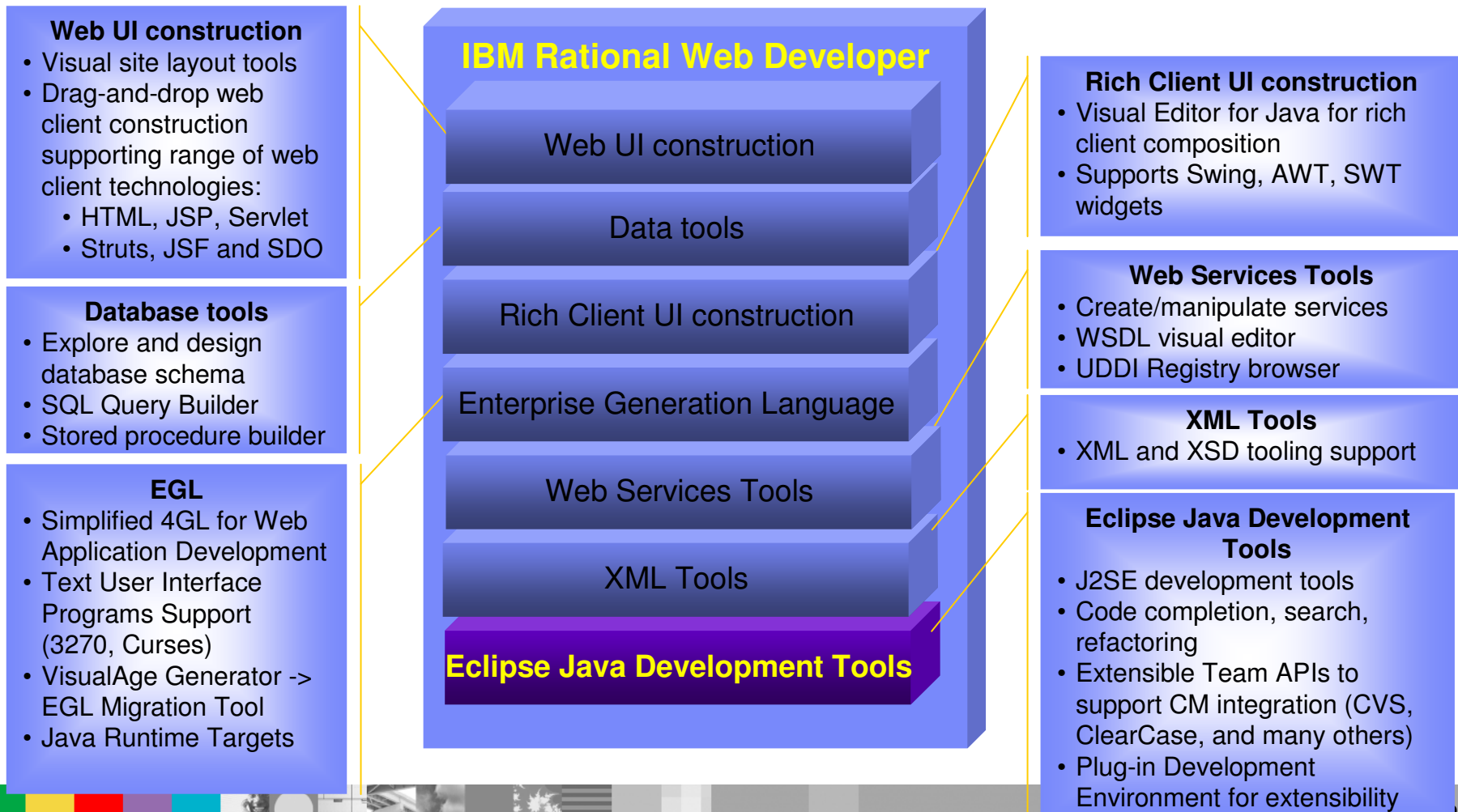
# The IBM Rational Software Development Platform



# IBM Rational Web Developer for WebSphere Software

*Follow on to WebSphere Studio Site Developer*

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



# IBM Rational Application Developer for WebSphere Software

*Follow on to WebSphere Studio Application Developer*

## J2EE/EJB Tooling

- J2EE construction tools
- Supports WebSphere and WebLogic
- Supports J2EE 1.2, 1.3, 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

**IBM Rational Web Developer for WebSphere Software**

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

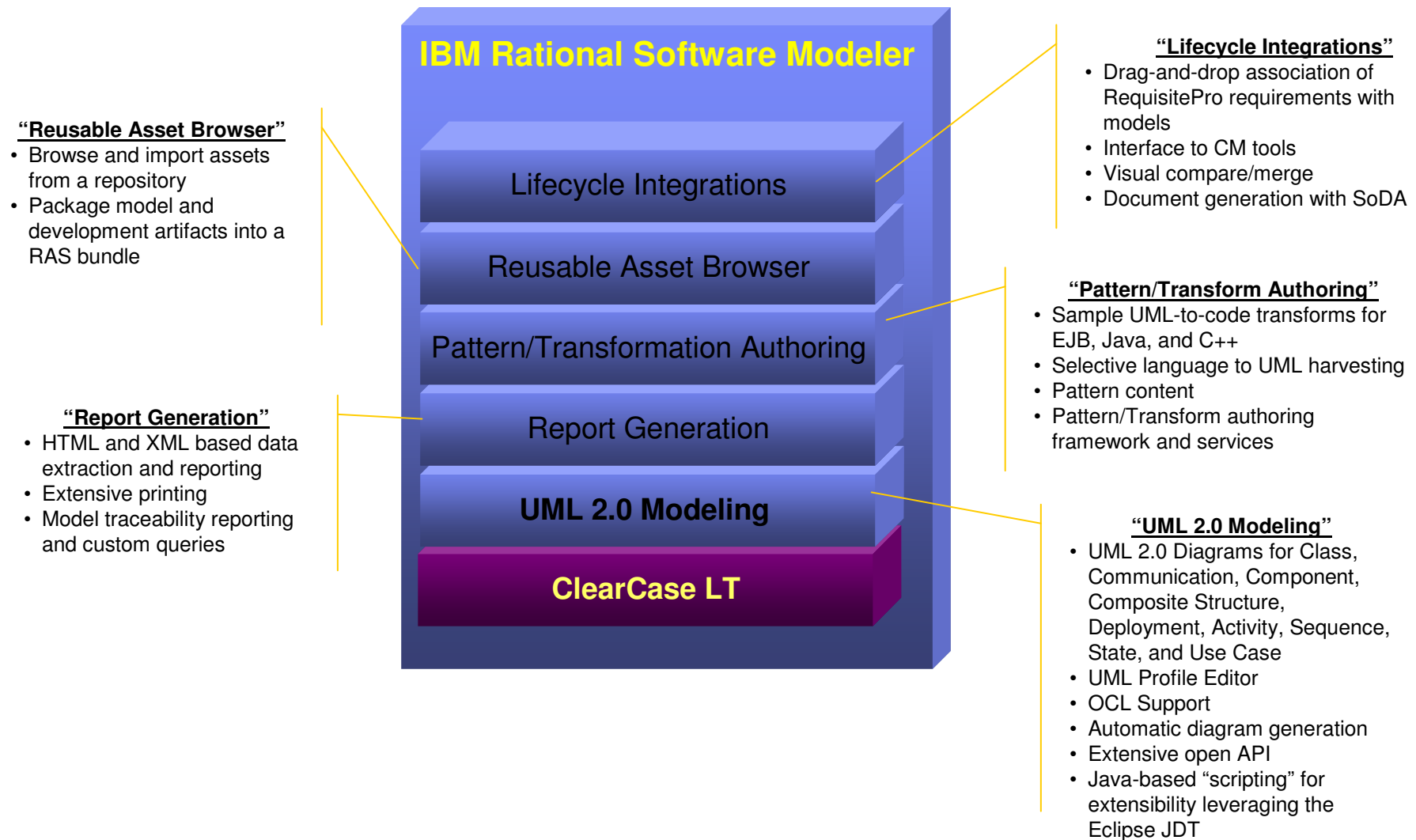
- Visual Portal site and page layout tools
- Visual Portlet layout and preview
- Integrated Portal test environment

## 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 Product Overview



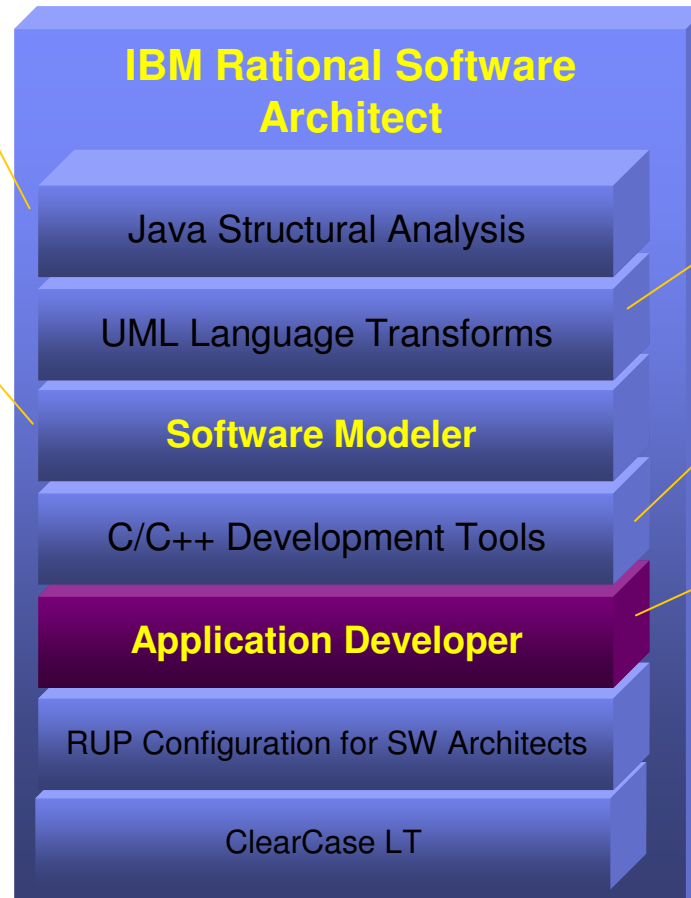
# IBM Rational Software Architect Product Overview

## “Application Analyzer”

- Automatic anti-pattern and pattern detection
- Architectural discovery, analysis, metrics, and stability reporting
- Implementation level architectural rules

## “Modeler”

- UML 2.0 Diagrams for Class, Communication, Component, Composite Structure, Deployment, Activity, Sequence, State, and Use Case
- OCL Support
- Automatic diagram generation
- Pattern content
- Pattern/Transform authoring framework and services
- Extensive open API
- Java-based “scripting” for extensibility
- HTML and XML based data extraction and reporting
- Extensive printing
- RAS tools



- Sample UML-to-code transforms for EJB, Java, and C++
- Selective language to UML harvesting

- C/C++ editors and build management
- Compiler and debugger integration
- UML code editors

## “WSAD v6”

- JSF, SDO, Struts
- Java GUI editor
- Web diagram editor
- Site designer
- Web Services development tools
- Database editing tools
- EGL
- EJB development tools
- UML code editors for EJB, Java, and Data
- Static Analysis
- Runtime Analysis
- Component test automation
- Portal/Portlet development tools