



# Modern Application Development Featuring Web 2.0 for System z

Introduction and Agenda

**Rational.** software

# Speakers



## **Will Smythe**

Product Line Manager

Rational Business Developer, EGL, HATS



## **Dave Myers**

Product Line Manager

Rational Developer for System z,

Rational Team Concert for System z

## Goals

- This briefing will explore the innovations provided by IBM Rational Enterprise Modernization software development tools, their broad range of functionality, and their use throughout the entire software development process.
- Attendee will get an understanding of the breadth and depth of these Rational tools, how well they're integrated, and how easy they are to use.
- This session will also help attendees understand the value of Web 2.0 to their businesses.

# Solutions from IBM

- **This briefing will provide information on the following products:**
  - Rational Asset Manager (RAM)
  - Rational Asset Analyzer (RAA)
  - Rational Transformation Workbench (RTW)
  - Rational ClearCase (CC) / ClearQuest (CQ)
  - Rational Build Forge
  - Rational Developer for System z with EGL (RDz EGL)
  - Rational Business Developer (RBD)
  - Rational EGL Rich UI
  - Rational Host Access Transformation Services (HATS)
  - IBM PD Tools
  - IBM CICS
  - IBM WebSphere Application Server

# Agenda

9:00 am	Introduction to Enterprise Modernization
	What is Web 2.0? Why Web 2.0 for System z?
10:30 am	Break
	Discover and Transform Application Assets (Hands on)
	Modern Application Architectures
12:00 pm	Lunch
	Modern Application Architectures (Hands on)
2:30 pm	Break
	Create and Integrate Web 2.0 Applications (Hands on)
	Application Lifecycle: Manage, Build, Deploy
	Summary, Resources and Next Steps
4:45 pm	End of seminar

The background of the slide is a vibrant blue with a series of thin, white, wavy lines that create a sense of motion and depth, resembling a stylized wave or a modern architectural design. The lines are most prominent on the left side and curve towards the right.

# Modern Application Development Featuring Web 2.0 for System z

IBM Enterprise Modernization

**Rational.** software

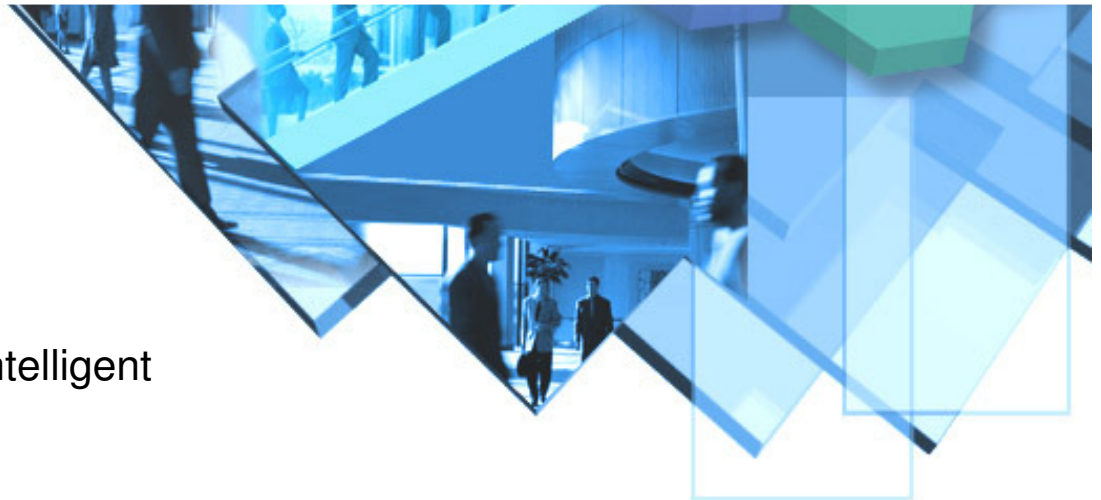
# Agenda



- **Accelerating Software Innovation**
- **The Rational Software Delivery Platform:  
An Enterprise Modernization View**
- **Q&A**

## Innovation will require a significant investment in software

- For businesses and institutions everywhere, there have never been so many ***new possibilities*** at our disposal
- Imagine the amount of software necessary to:
  - Deliver a green world
  - Take advantage of information base explosion
  - Communicate with a trillion devices
- We are moving to a new generation of intelligent software and systems
- To accelerate innovation and take advantage of these new possibilities, clients must get better at ***managing effective software delivery***

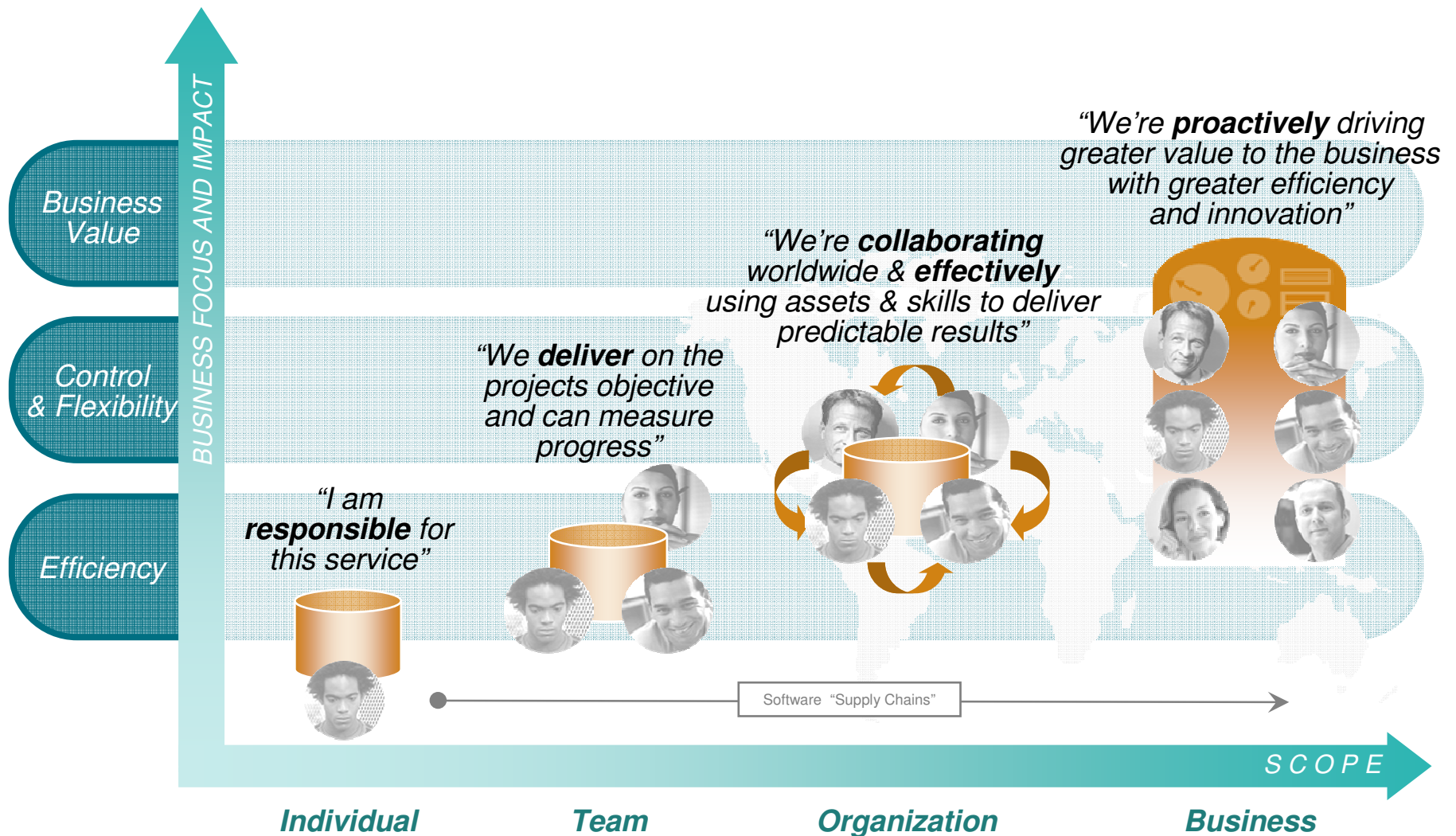




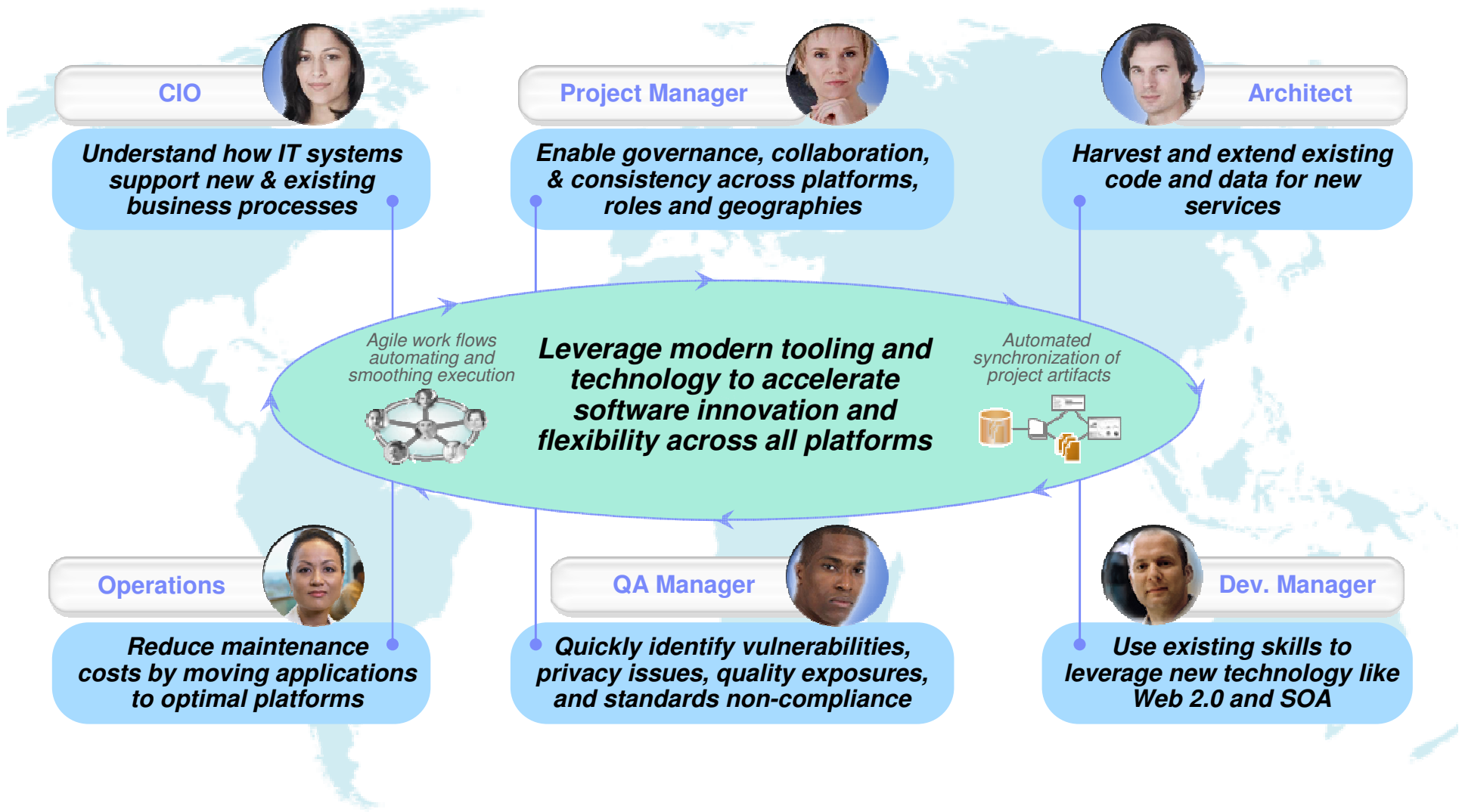
# Customers are asking...How can we ?...

- Speed software delivery?
  - Meet organizational commitments?
  - Make time-critical market windows?
  - Deliver software projects to operations on time and on budget, with greater reliability?
- Improve software quality?
  - Ensure that business-critical applications meet end-user requirements and performance criteria?
  - Comply with regulatory standards?
  - Pinpoint critical application security vulnerabilities?
- Integrate solutions across platforms and reduce complexity?
  - Drive organizational alignment?
  - Improve efficiency, reduce errors and reduce friction?
  - Harness globally distributed resources for development, delivery and operations?

# Needed: A mature approach to software delivery



# Needed: Solutions that eliminate barriers to innovation

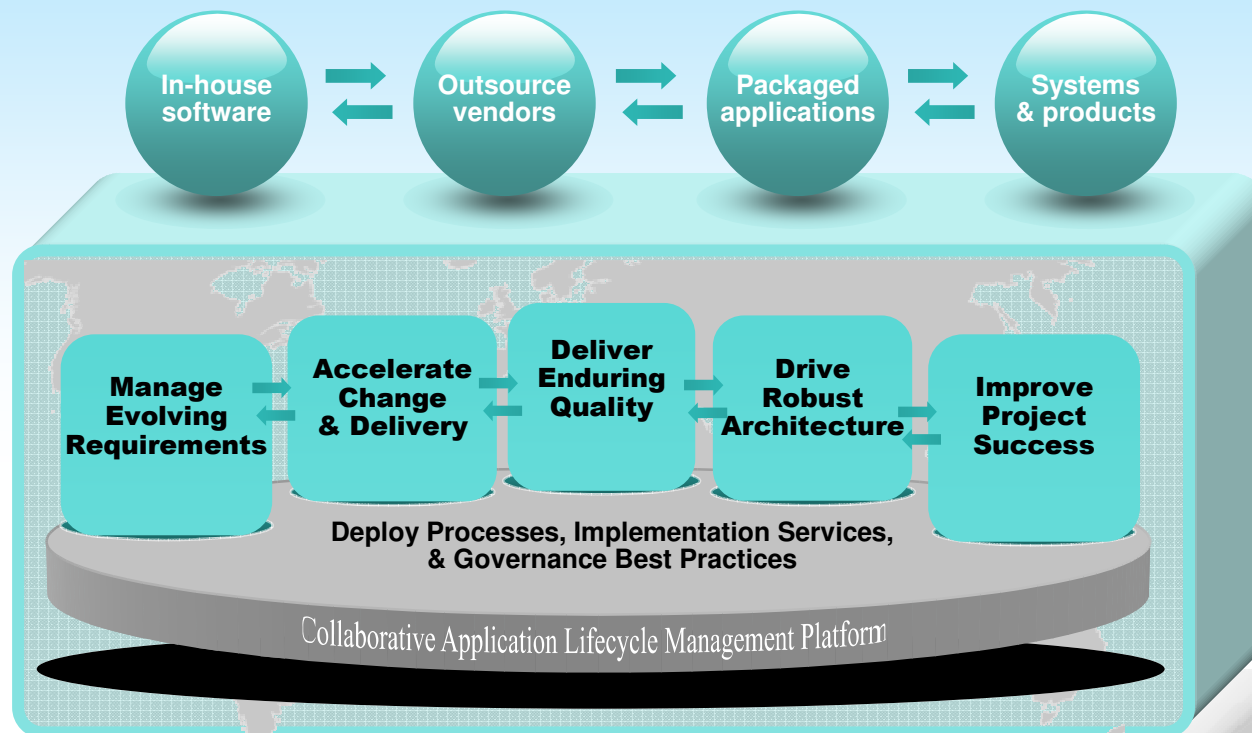


# IBM Rational Software Delivery Platform

*Accelerating software innovation for the entire enterprise*

**Rational.** software

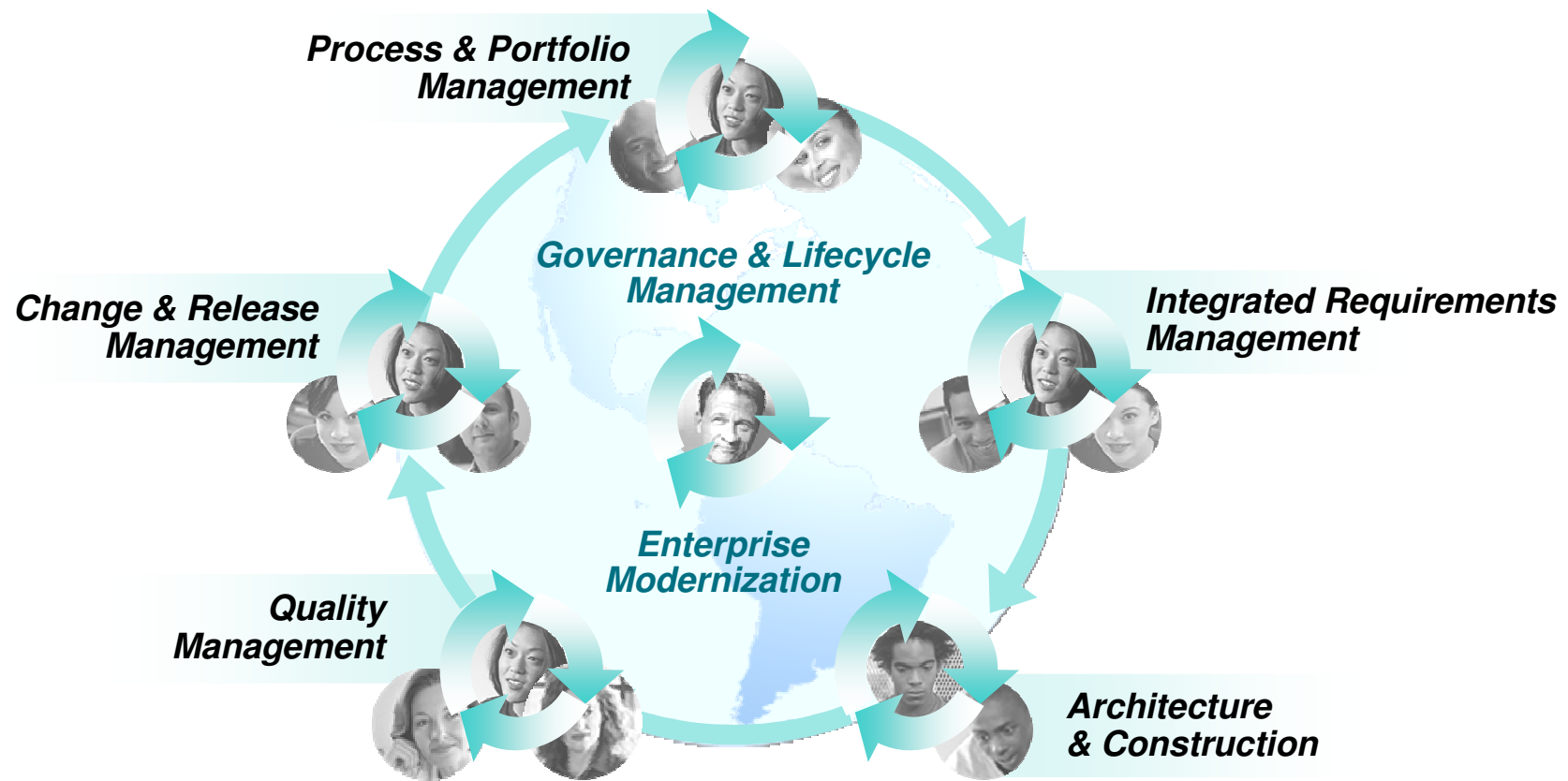
*Solutions to help customers achieve greater value and performance from their investments in delivering software*



- *Enterprise Modernization and Transformation*
- *Organizational Governance*
- *Skill Development and Community*
- *Implementation Services*

# Rational's business

*Governing the business process of software and systems delivery*

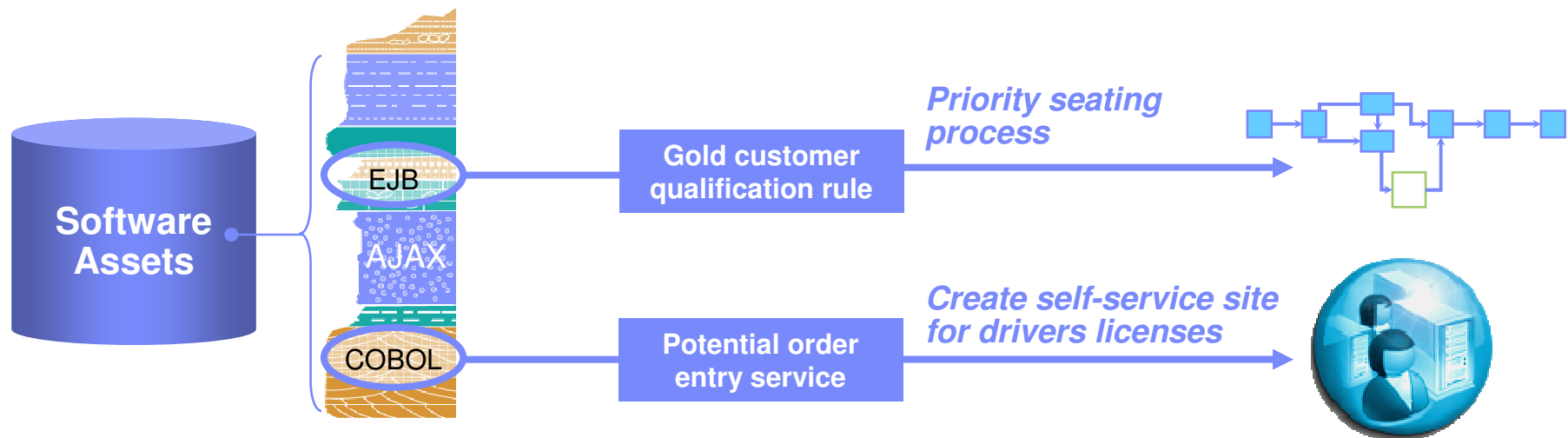


# Managing Software Delivery - for System z ... and distributed

- ▶ Multi-platform development is the norm; System z is key in most enterprises
- ▶ However, many think of z development as “different and unique”, constraining efficiency & flexibility
- ▶ Rational’s Enterprise Modernization solutions provide:
  - a consolidated platform for multi-platform application development
  - a collaborative, integrated environment that supports innovation
  - the same power and flexibility for all types of applications, including z
- ▶ With totally modern tools for System z, customers can choose their deployment platform for new applications based upon runtime requirements (e.g. QoS, proximity to data)  
... and NOT based on out-dated perceptions of z development
- ▶ It is time to re-think how z development is done!

# Companies want to

Understand how their IT systems support their business processes, and then enhance them in a predictable manner

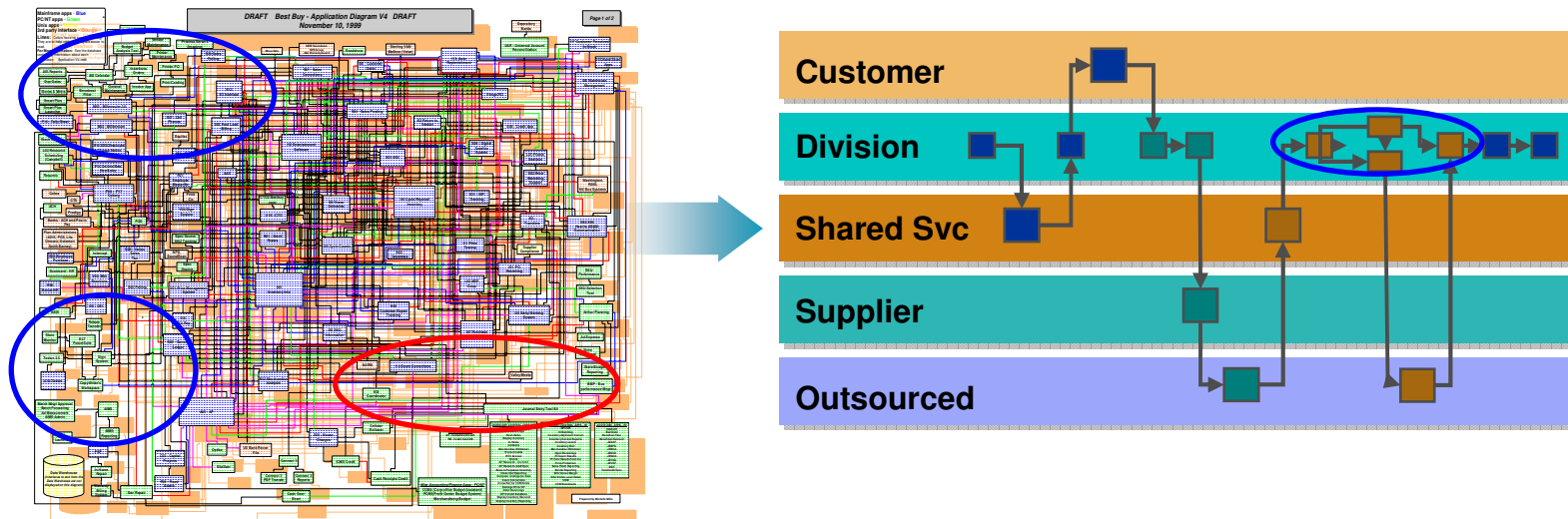


*But...*

*They have no business intelligence inventory of their application assets, relying instead on manual processes & incomplete knowledge*

# Companies want to

Have an IT architecture that quickly adapts to support current business needs



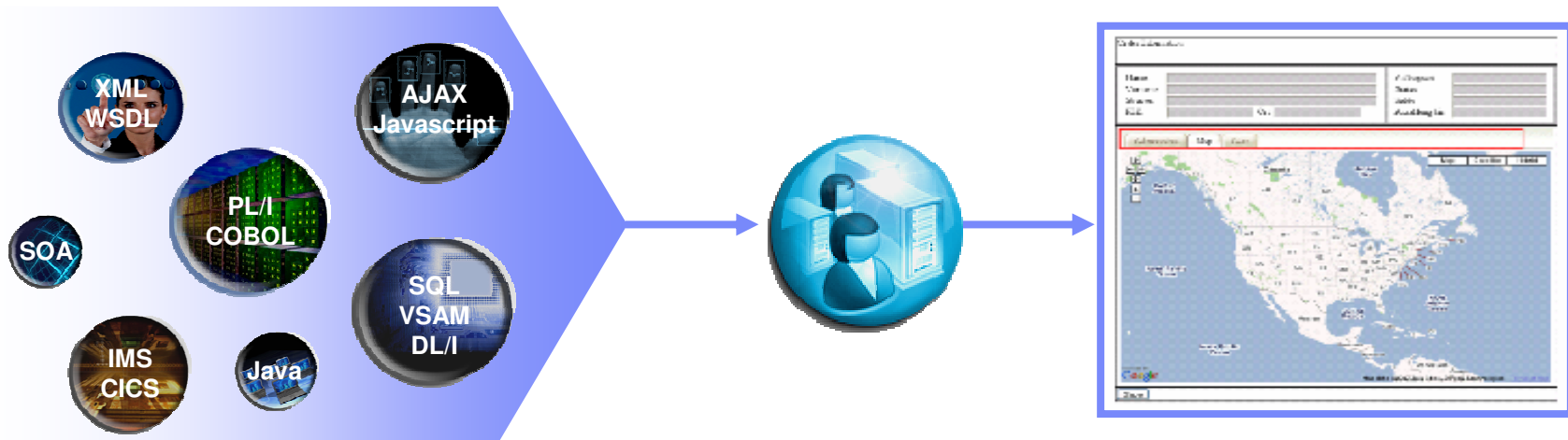
*But...*

*They have application architectures that were incrementally built over the last 20-30 years that are no longer flexible or agile*



# Companies want to

Use existing developers to build modern cross-platform apps that leverage new technologies like Web 2.0

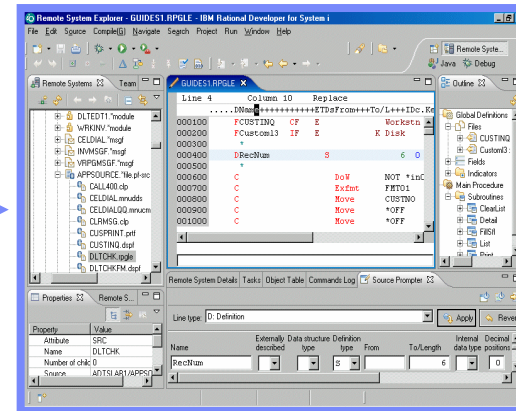
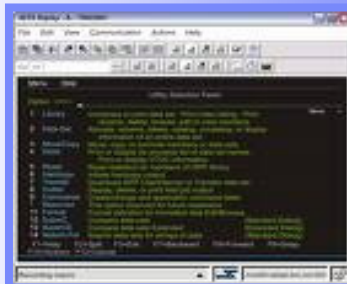


*But...*

They have developers that do not know all of these new technologies, and hiring all new people or doing massive retraining is unrealistic

# Companies want to

Increase productivity of their “backend” development staff, as well as attract new employees to this work

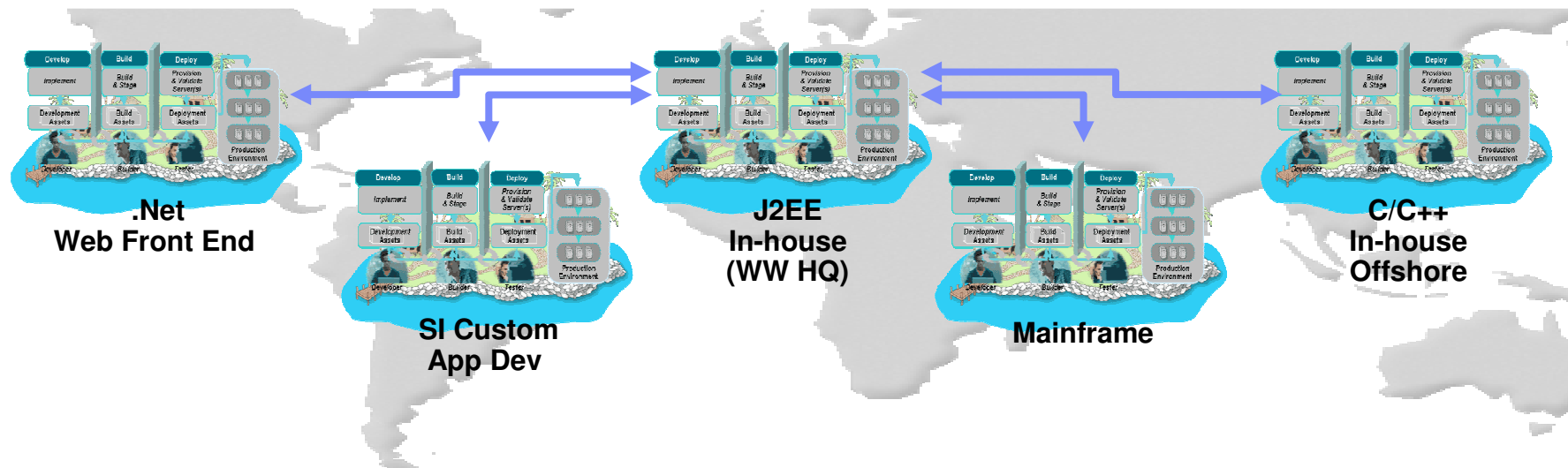


*But...*

They have antiquated development tools which deliver low productivity and are repulsive to younger staff

# Companies want to

Enable multi-platform and cross-geo collaboration and governance

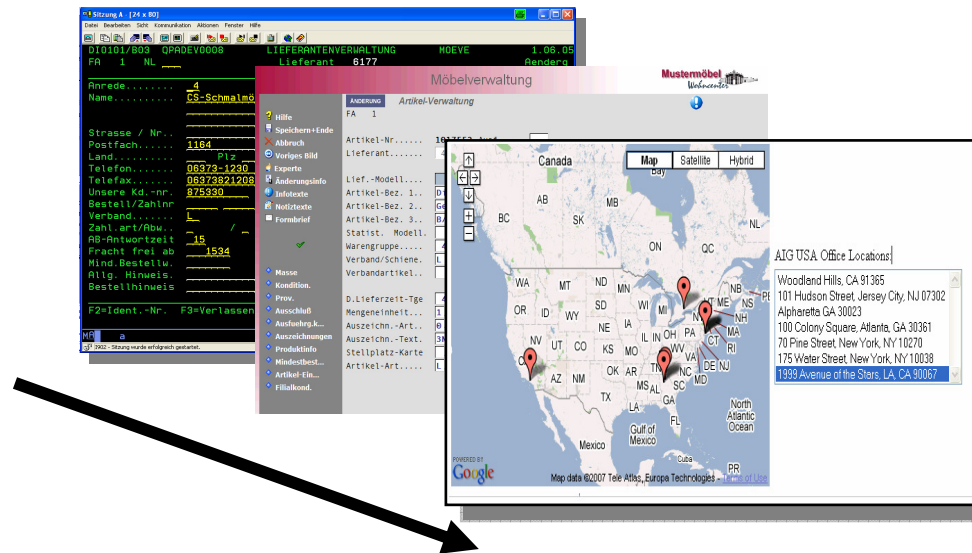


*But...*

They have different tools, processes and team infrastructure for each deployment platform, greatly complicating development of composite apps

# Companies want to

Present a sophisticated, modern user interface to their customers and employees

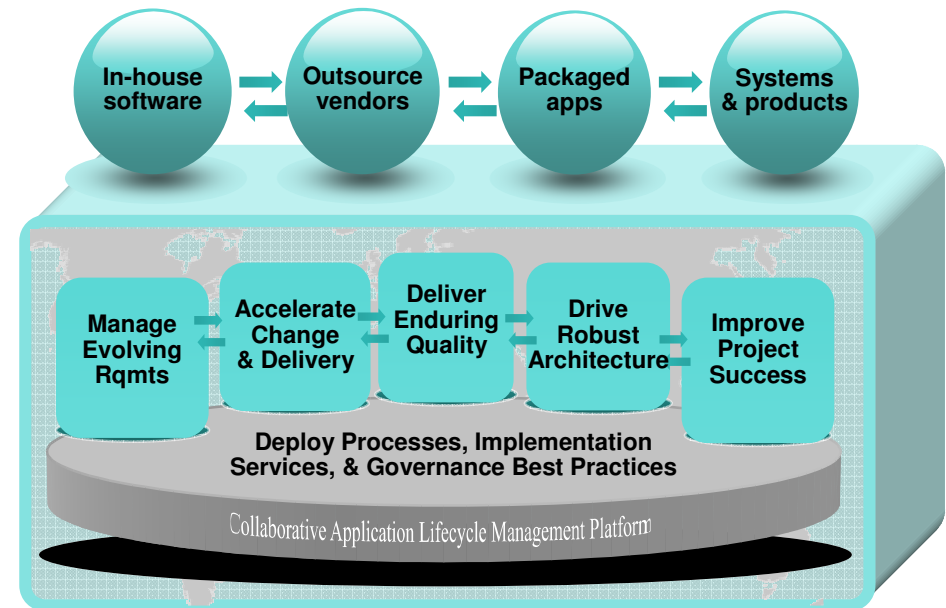
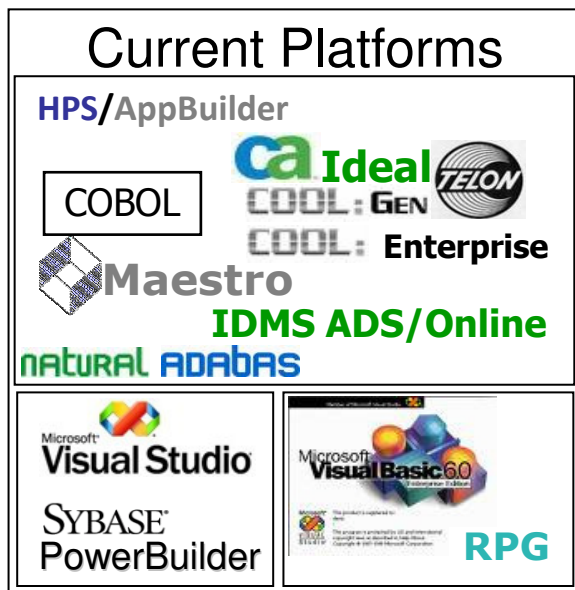


*But...*

They don't know how to upgrade their archaic green screen applications without rewriting their applications

# Companies want to

Reduce costs and move to modern platforms, languages, and tools for all development



*But...*

They are stuck on unsupported, expensive, legacy development platforms

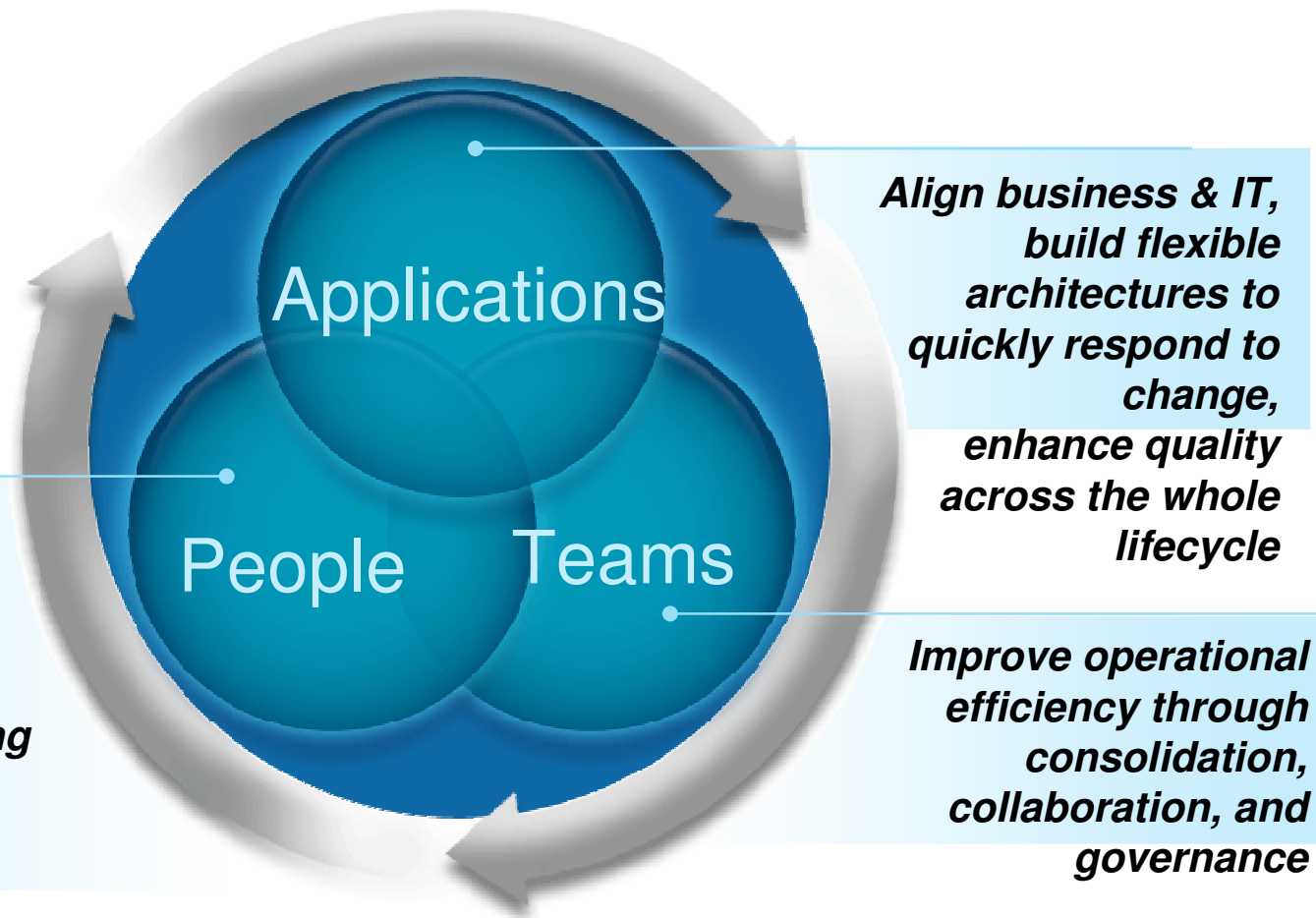
# The Road to Enterprise Modernization



- **Companies need a unique modernization strategy based on:**
  - business requirements
  - budget
  - staffing constraints
- **With our Enterprise Modernization expertise, we can help formulate a customized roadmap for each of our customers**
- **Here are some of our solutions...**

## Enterprise Modernization – Solution Overview

**Connecting applications and people across teams to improve business and IT flexibility**



For more information, visit:  
[www.ibm.com/rational/modernization](http://www.ibm.com/rational/modernization)

# Modernize your Asset Management

I want to improve my business processes, but where do I start?



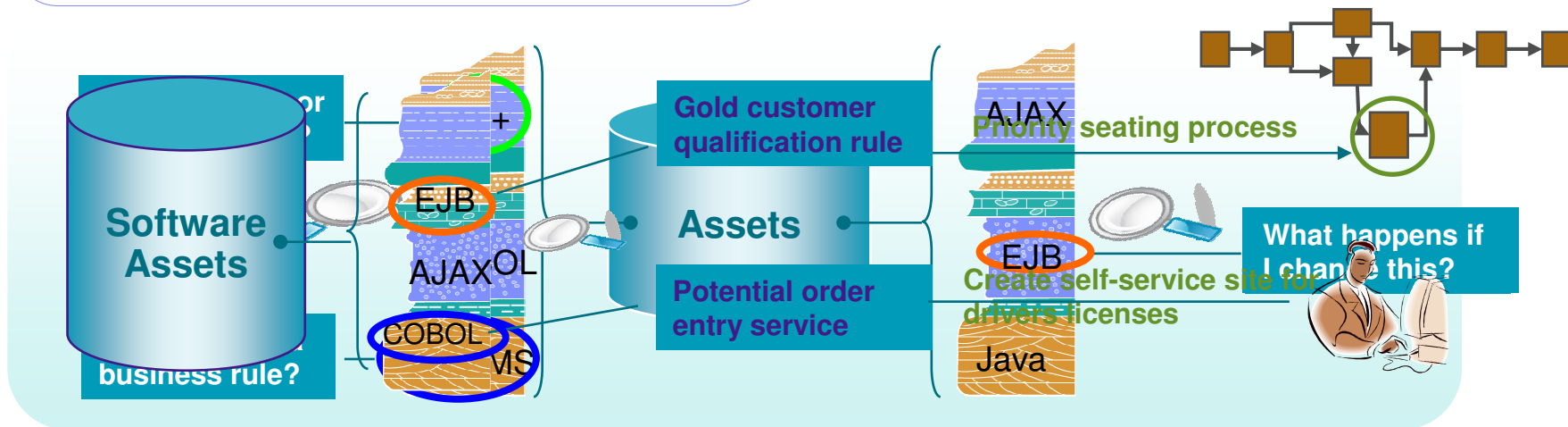
**Business Challenge**  
Difficulty understanding and leveraging existing IT systems to support business processes

**How can we...**

- ▶ Understand how our IT systems support our business processes so we can predictably enhance them
- ▶ Avoid application downtime due to unforeseen code dependencies?
- ▶ Maintain control over code provided by outsourcers, business partners, or acquired through a merger?
- ▶ Accelerate SOA efforts?

**Solutions**  
Comprehensive inventory of assets for impact analysis & reuse

- ✓ Map business processes to IT systems
  - *WebSphere Studio Asset Analyzer* New!
  - *Rational Asset Analyzer*
  - *Rational Transformation Workbench*
- ✓ Manage assets during development
  - *Rational Asset Manager*
- ✓ Manage deployed services
  - *WebSphere Service Registry & Repository*





# Modernize your Architectures

That would make a great web service!



## Business Challenge

Inflexibility in adapting applications in support of business needs

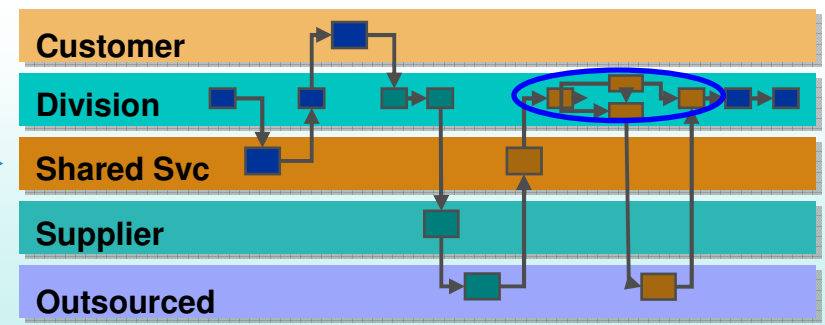
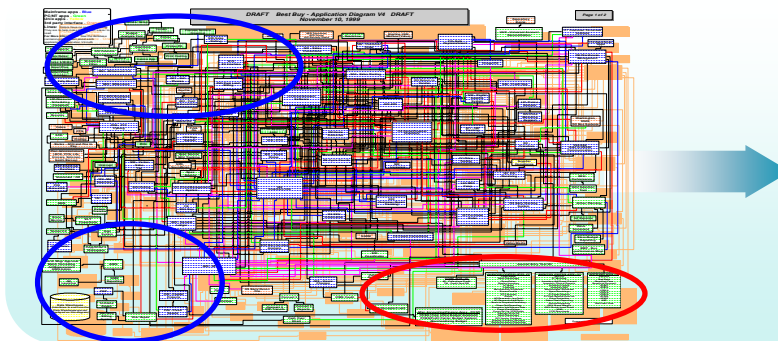
### How can we...

- ▶ Progress our SOA maturity given all our existing code assets, including CICS, IMS, IBM i and green screen applications?
- ▶ Build services once and deploy them across my heterogeneous production environment?
- ▶ Ensure our new SOA system will be more easy to extend and maintain than my current ones?

## Solutions

Flexible architectures to enable business agility

- ✓ Expose 3270 and 5250 applications as web services
  - *Host Access Transformation Services*
- ✓ Quickly create web services from existing COBOL, RPG, PL/I, Java, or EGL applications
  - *Rational Developer for System z (SFM)*
  - *Rational Business Developer*
- ✓ Create new platform-independent services
  - *Rational Business Developer (EGL Service keyword)*
- ✓ Leverage MDD - xform UML to EGL, COBOL, WSDL, etc
  - *Rational Software Architect*



# Modernize your Skills - Languages

I built that cool web 2.0 app with EGL



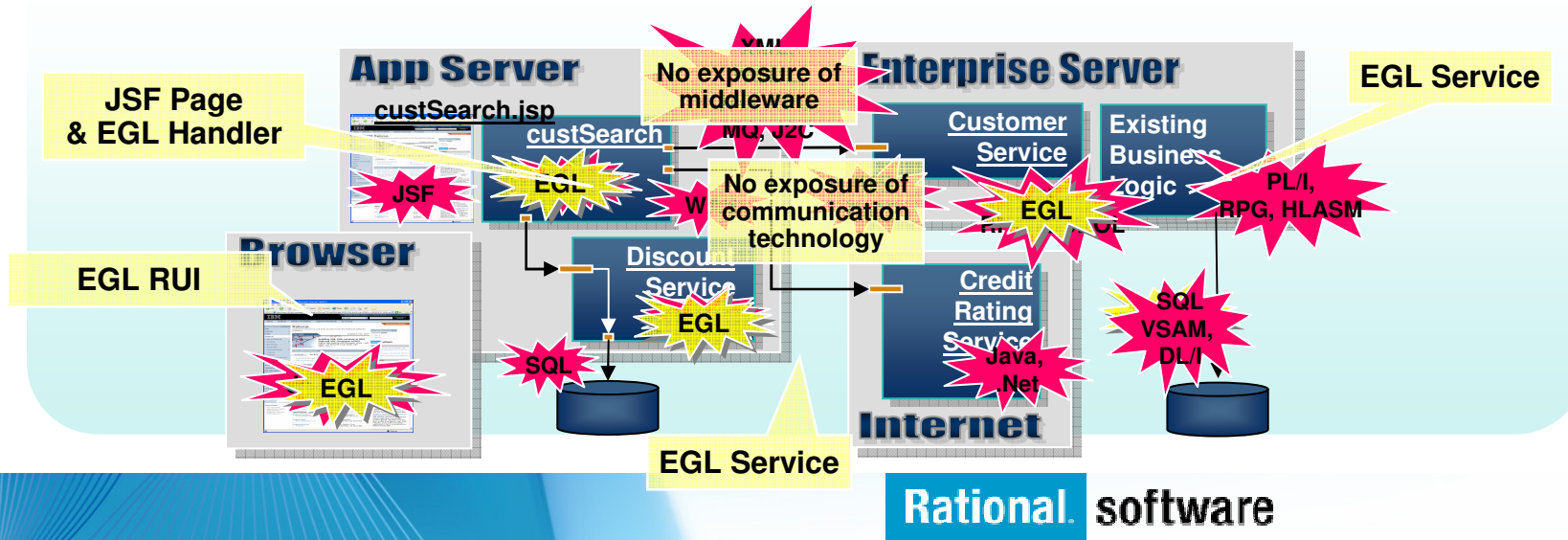
**Business Challenge**  
Staff inflexibility due to “skills silos”

How can we...

- ▶ Exploit new technologies and innovation without retraining existing staff that knows the business?
- ▶ Use new employees on any project, independent of the target platform?
- ▶ Provide modern web UIs leveraging existing skills and systems?

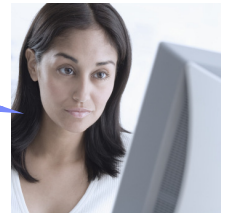
**Solutions**  
Powerful tools & abstractions for cross-platform development using today’s skills & tomorrow’s new employees

- ✓ Apply existing business-knowledgeable staff to build all elements of multi-platform applications ... end-to-end from Web 2.0 to services to business transactions
- ✓ Attract new staff with “IBM’s newest business language”
  - Rational Business Developer (EGL)



# Modernize your Skills - IDEs

I am NOT going to miss my green screen IDE



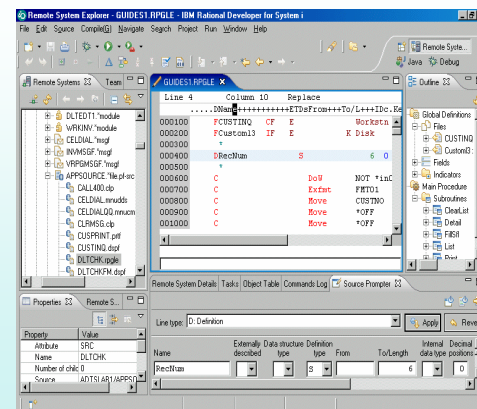
**Business Challenge**  
Low productivity, non-competitive tools for new developers

**How can we...**

- ▶ Achieve higher productivity for our enterprise developers?
- ▶ Attract new and younger developers to backfill retiring System z and IBM i developers?
- ▶ Simplify license management by standardizing on a single, multi-platform development IDE?
- ▶ Free up System z MIPS for production use?

**Solutions**  
Modern IDEs to achieve high productivity and attract new talent

- ✓ Use modern IDEs to develop and maintain code
- ✓ Debug and test from workstation
  - Rational Developer for System z
  - Rational Developer for System i
  - Rational Business Developer
  - Rational Developer for System i for SOA Construction
  - Rational Application Developer



# Modernize Your Team Infrastructure

The only island I want to see has palm trees



## Business Challenge

Inefficiencies and quality problems in multi-platform development

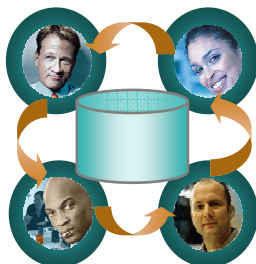
### How can we...

- ▶ Increase quality and improve responsiveness for multi-platform development teams?
- ▶ Simplify management of team infrastructure for multiple development teams?
- ▶ Reduce team infrastructure costs across the enterprise?

## Solutions

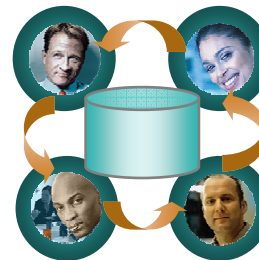
Coordination, traceability, consistency across platforms

- ✓ Consolidate enterprise and distributed source code management
  - *Rational ClearCase*
- ✓ Automate and orchestrate defect tracking and configuration management across the enterprise
  - *Rational ClearQuest*
- ✓ Produce coordinated, traceable, automated, and cross-platform builds
  - *Rational BuildForge*



Distributed Development Environment

RDz RDi RPP



Enterprise Development Environment

Enterprise Portal

Enterprise Asset Repository

# Delivering the first wave of New Rational Jazz Offerings

## Collaborative Application Lifecycle Management

New!

### Requirements Composer

*Business Expert Collaboration*

Elicit, capture, elaborate, discuss and review requirements using a variety of techniques & notations

New!

### Quality Manager

*Business Driven Quality*

Coordinate and manage quality assurance plans, processes and resources across the lifecycle

*Coming soon...*

- Real-time enterprise reporting
- Collaborative project management
- Process design & enactment
- **Application Development**



## Jazz TEAM SERVER

*In-context collaboration, real-time project health, event notification, process enactment & enforcement, global search & query, security, role-based access, automated traceability*

New!

### Team Concert

*Software Innovation Through Real-time Project Team Collaboration*

#### Work Item Management

Automated and flexible software change management improving project control and visibility

*ClearQuest Connector*

#### Configuration Management

Source code control, traceability and versioning leveraging dynamic workspaces & parallel development

*Clearcase Connector*

#### Build Management

Automates build planning, project construction, build of materials and release processes

*Build Forge Integration*



Web



Eclipse

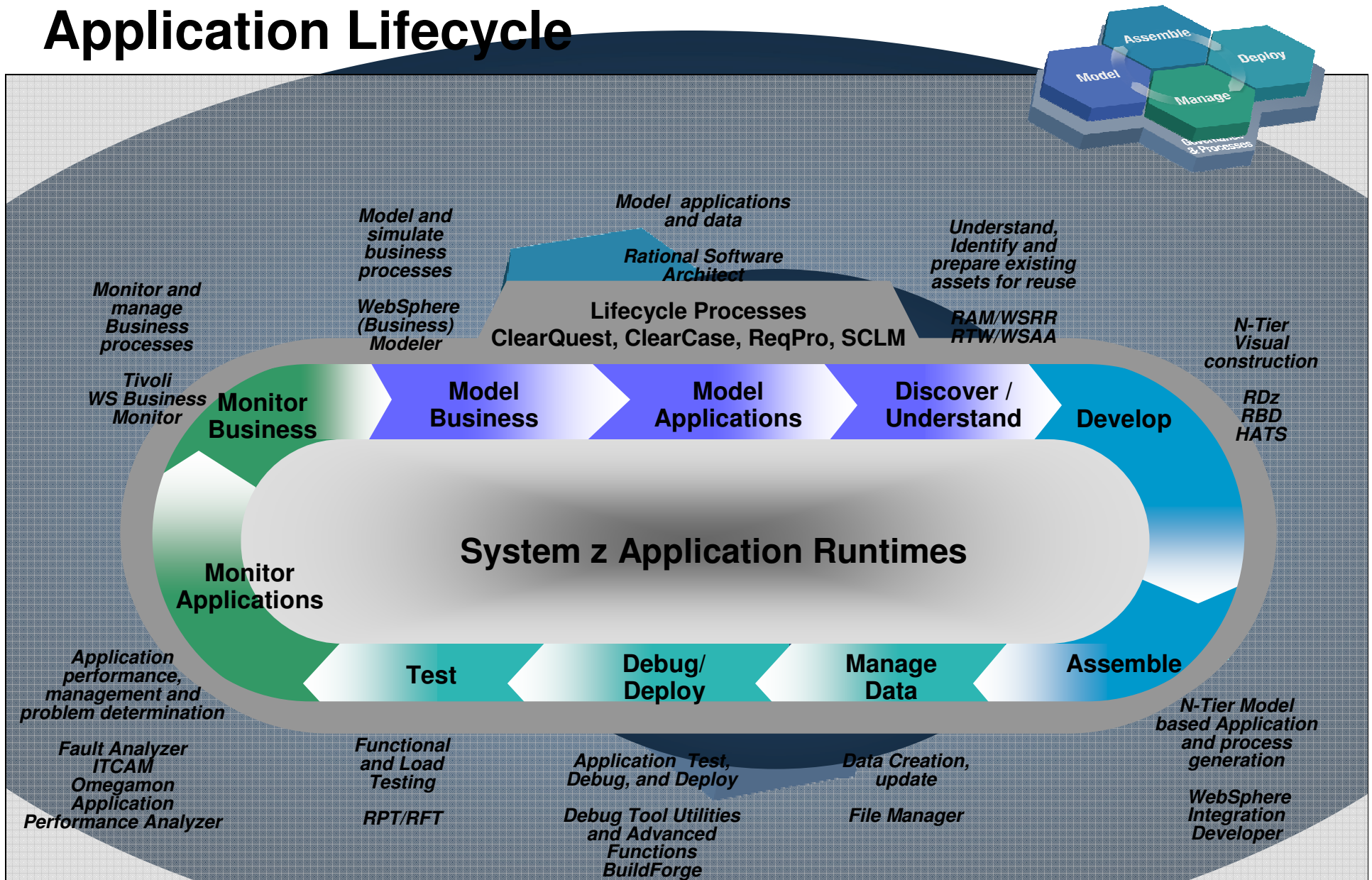


Visual Studio  
(future)

New and Enhanced!

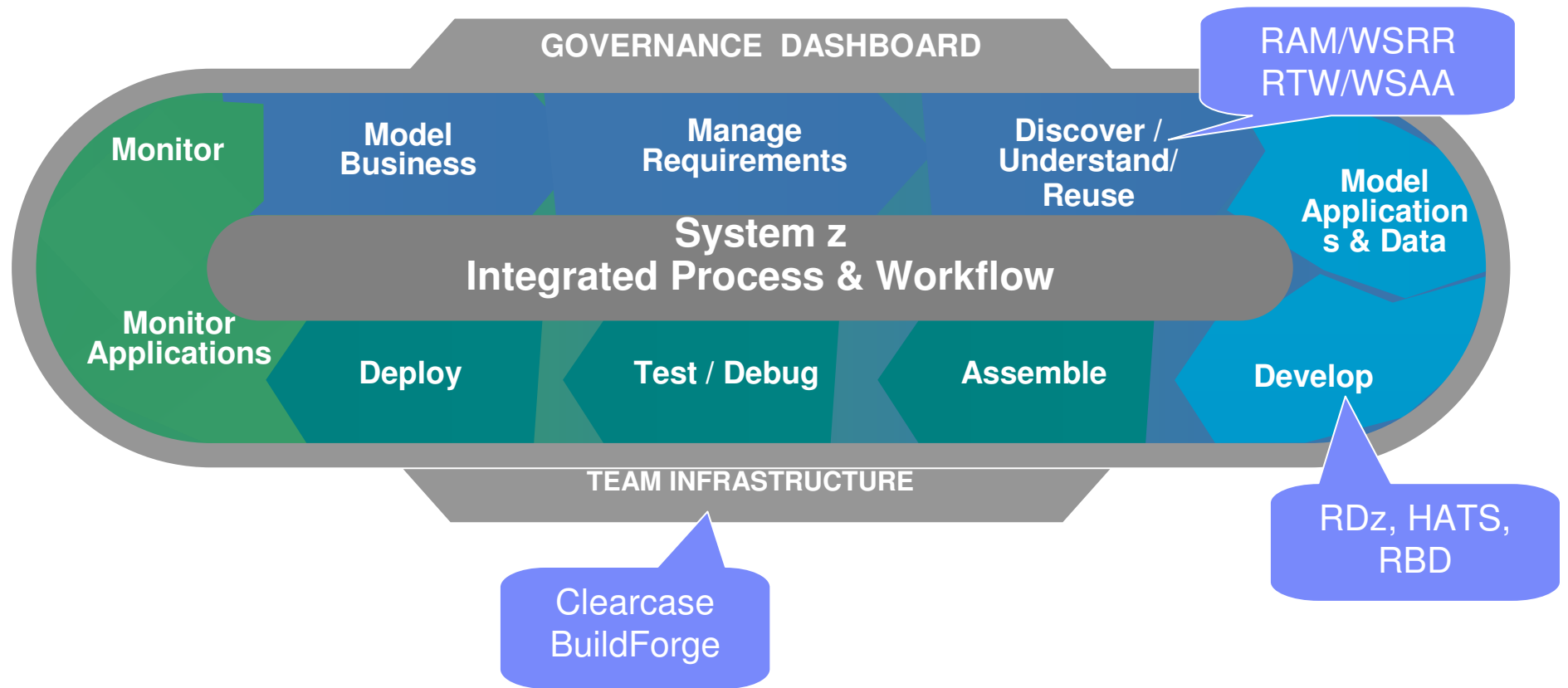
Jazz Enhancements to ClearQuest, Build Forge and ClearCase

# Application Lifecycle



# The IBM Software Delivery Platform

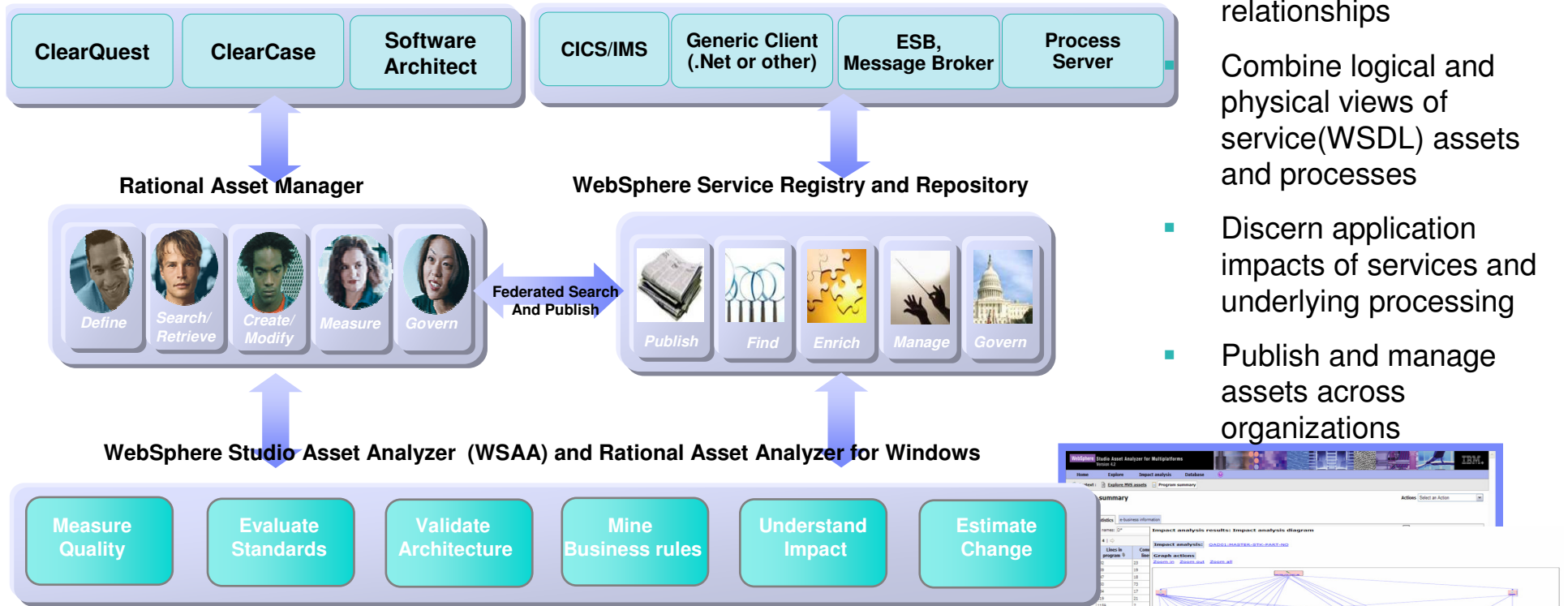
*A complete, coherent and integrated set of AD solutions*



# Bridge the gap between services and application processing

Manages information that is useful for developing, re-using and managing all types of reusable assets

Manages information that is useful for the runtime operation, management and development use of services



- Gain asset clarity with respect to assets, services and relationships
- Combine logical and physical views of service(WSDL) assets and processes
- Discern application impacts of services and underlying processing
- Publish and manage assets across organizations

**IBM WebSphere Studio Asset Analyzer v5.1 (WSAA)** assists in enterprise wide application understanding helping organizations to understand application architecture, perform impact analysis, evaluate application quality an speed a developers overall throughput and lower maintenance costs. It also is part of a broad solution addressing SOA governance providing insight into the processing contained in services. It is a complement to products including RSA, WSRR, and RTW.



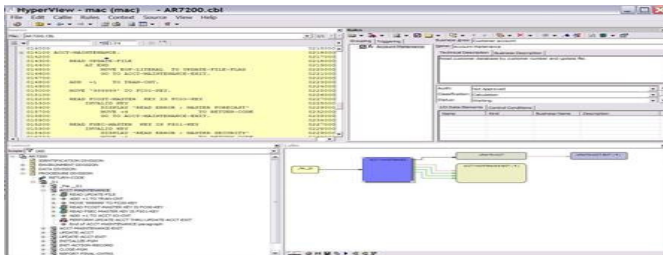
# Extend value of existing enterprise assets

*Accelerate your path to reuse and SOA with Rational Transformation Workbench*

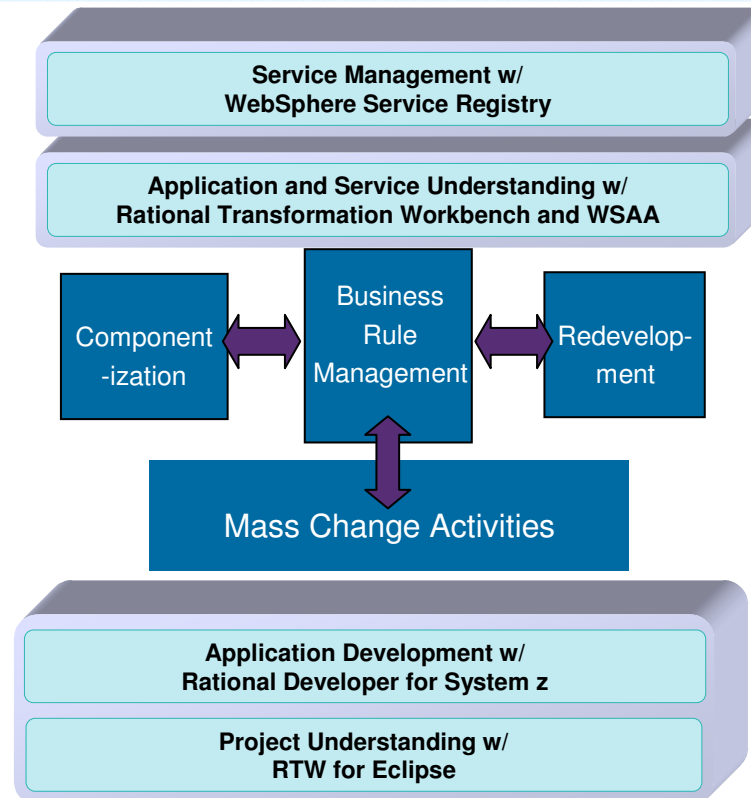
**IBM Rational Transformation Workbench v3.1 (RTW)** assists in restructuring monolithic applications to help reduce complexity and maintenance costs and speed development of SOA services. It provides deep project-level application analysis to complement the enterprise-wide application insight of IBM WebSphere Studio Asset Analyzer software.

## New in RTW 3.1

- ▶ Demystify enterprise applications and their inter-and intra-relationships
- ▶ Gain intellectual control; make fact-based decisions; develop transformation roadmap
- ▶ Identify key assets, mine business rules and restructure for reuse in SOA; Reduce cost and time of modernization projects



IBM Rational Transformation Workbench



# Drive innovation with technology advancements

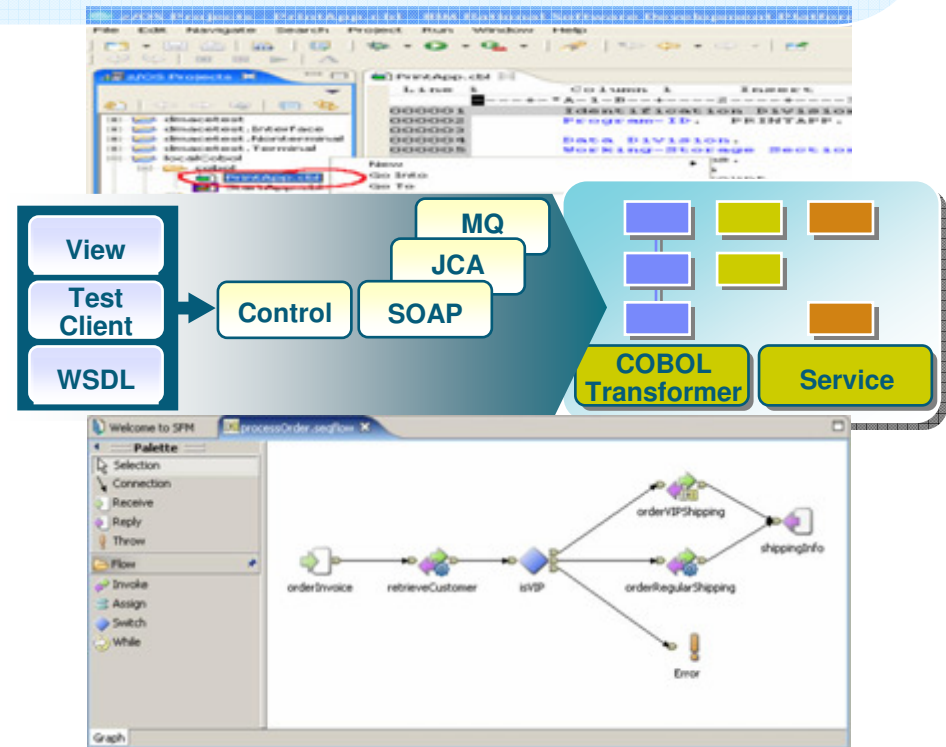
## *Build traditional and composite applications with Rational Developer for System z*

**IBM Rational Developer for System (RDz)** provides powerful eclipse-based tools for building COBOL, PL/I, C/C++, Java, HLASM CICS, IMS and batch applications. RDz is designed to optimize and simplify application development for traditional processes, and for the SOA you're building to address your needs today and in the future—through best practices, visual tools, templates, code generation and a comprehensive development environment.

### RDz 7.5

- ▶ Leverages service flow modeler to automate development of CICS services
- ▶ Develop z/OS applications and CICS and IMS Web services more productively than ever
- ▶ Manage core and composite artifacts from z/OS
- ▶ Speed software delivery with integrated, automated end-to-end processes

- Enables traditional; and modern UI's (via RBD)
- Supports local and remote development
- Creates connectivity via web services, CICS SFM, CTG, IMS Connect
- UML to COBOL speeds Model driven development
- Delivers distributed debugging support for SOA applications via Debug Tool
- Integrates to File Manager, Fault Analyzer

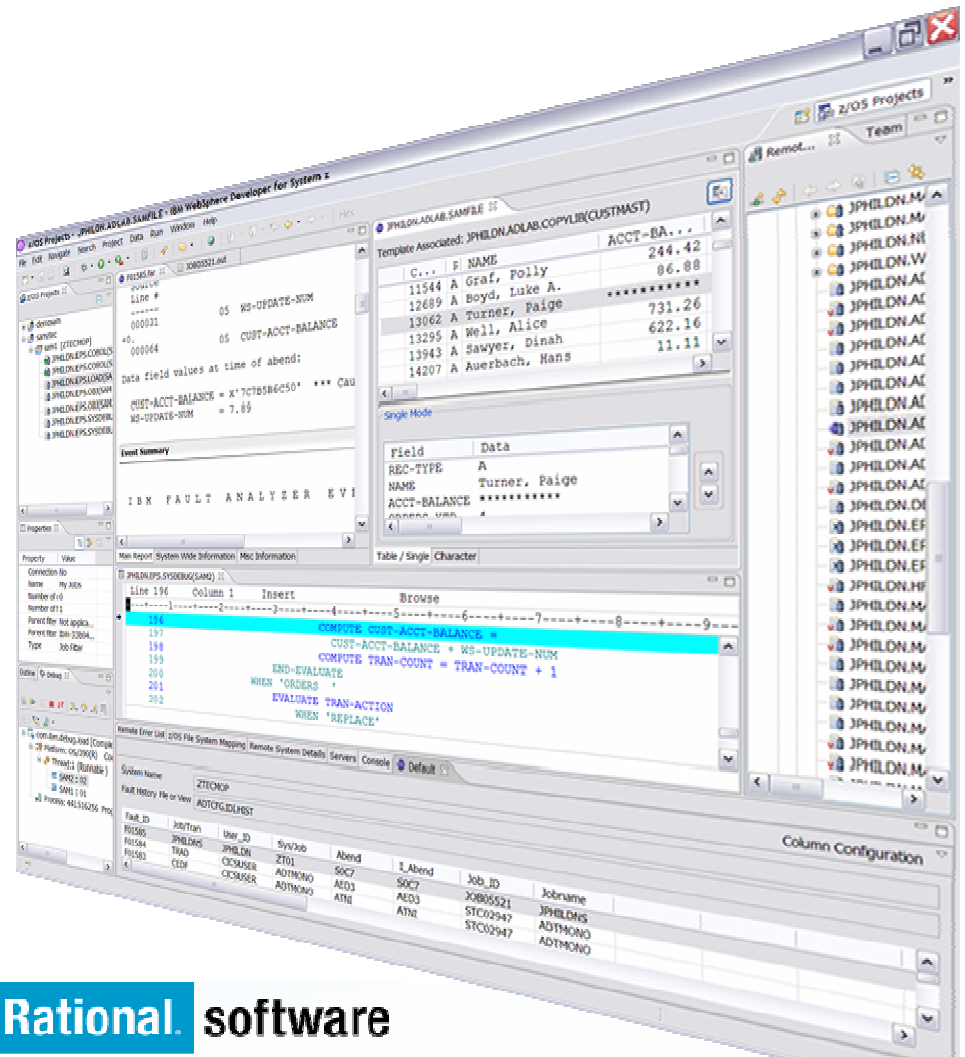


IBM Rational Developer for System z

# Problem Determination Tools Suite Features & Capabilities

Supporting development and testing of SOA and composite applications

- Optimize and manage the performance of application resource
- Compile, debug and test applications, and convert code quickly and easily
- Analyze and correct application failures with minimal down-time
- Manage and generate data files, including XML data files
- Extract/manipulate production data for testing applications (DB2, IMS, VSAM) and provide data privacy functionality
- Conduct stress, performance, regression, function and capacity planning tests



# Problem Determination Tools Suite for z/OS

IBM Problem Determination Tools

## IBM 2009 Offerings

Debug Tool  
for z/OS V9

File Manager  
for z/OS V9

Fault Analyzer  
for z/OS V9

Application  
Performance  
Analyzer for  
z/OS

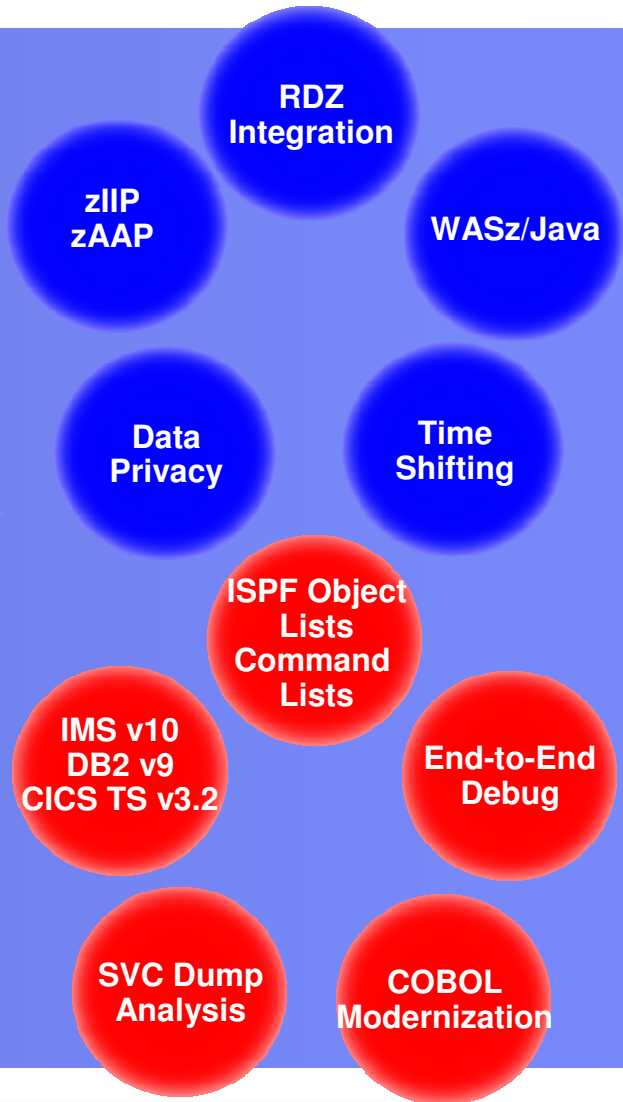
Workload Simulator  
for z/OS & OS/390  
Rational Functional Tester Ext  
Rational Performance Tester z/OS

Optim Move  
for DB2

Application Time  
Facility

ISPF  
Productivity Tool

[www.ibm.com/software/awdtools/deployment](http://www.ibm.com/software/awdtools/deployment)



# Extend value of existing enterprise assets

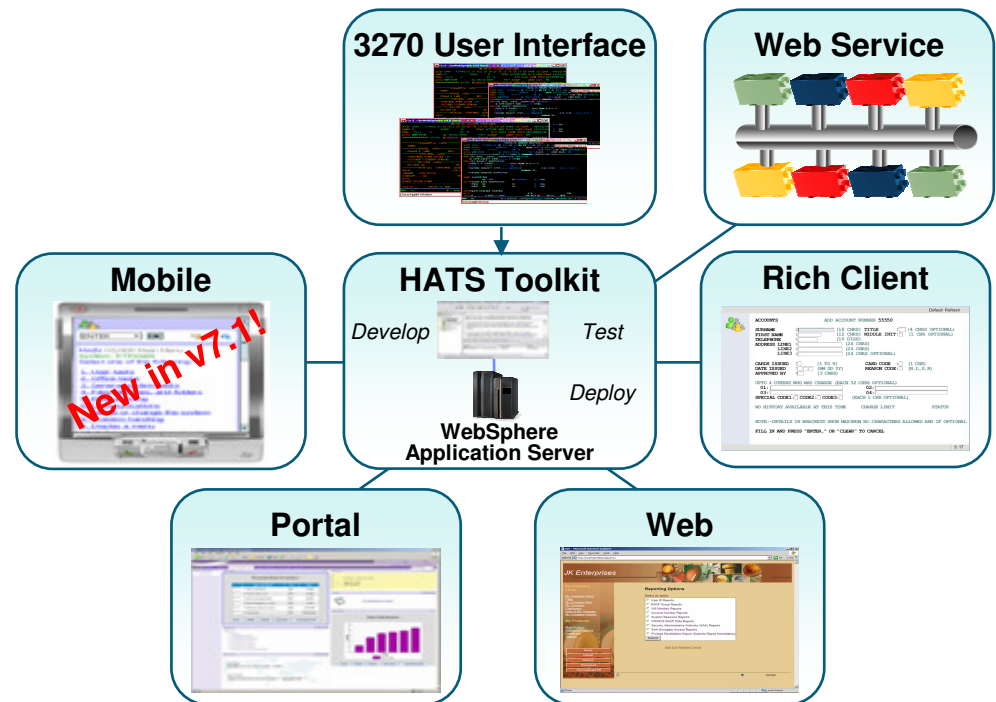
## Transform user experience with Host Access Transformation Services for Multiplatforms\*

With **IBM Rational Host Access Transformation Services for Multiplatforms (HATS)** customers can quickly improve the user interface of their existing System z applications and achieve integration with other business information with minimal development cost. The result is an easy-to-use Web interface to the application, which reduces training cost, simplifies the application, and makes it easy to understand for new users.

### HATS 7.5

- ▶ Rename/Rebrand/Reprice/Repackage
- ▶ WAS 7.0, Portal 6.1, Mobile App Support
- ▶ Visual Macro Editor (Full support)
- ▶ JSR 168 support
- ▶ Web services usability improvements

- Enhance 3270 and 5250 green screen applications with a GUI interface
  - Combine data from multiple screens, applications and databases and deploy to Web, Portal, and rich client
- Use HATS as on ramp to your SOA environment
  - Extends terminal app. tasks as Web Services and insert them in business processes



IBM Rational Host Access Transformation Services for Multiplatforms

# Drive innovation with technology advancements

## Accelerate adoption of Web 2.0 and SOA with Rational Business Developer

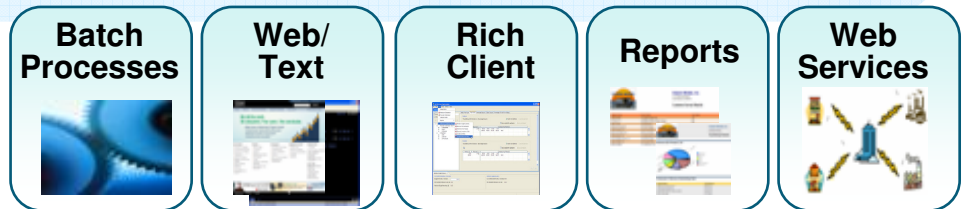
New!

IBM Rational Business Developer (RBD) helps business developers build cross-platform Web and service-oriented applications through “EGL”. COBOL and PL/I developers can leverage their domain using “EGL, IBM’s newest and most powerful business language”, to design, develop, test, and deploy new solutions in record time.

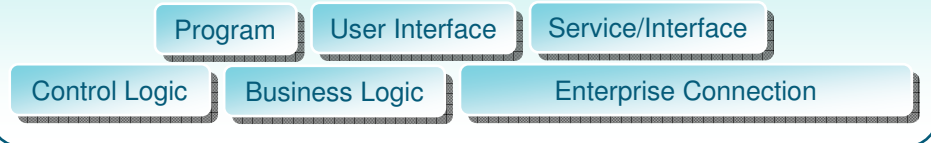
### New in RBD 7.5

- ▶ Create Web 2.0 and AJAX applications using EGL’s support for Javascript (RBD V7.5.1)
- ▶ Support for MQ, CSV, DL/1, TUI (3270, 5250), BIRT reporting
- ▶ Deploy generated program to IMS
- ▶ Improved support for System i, Portal development
- ▶ Target for migration from VAGen

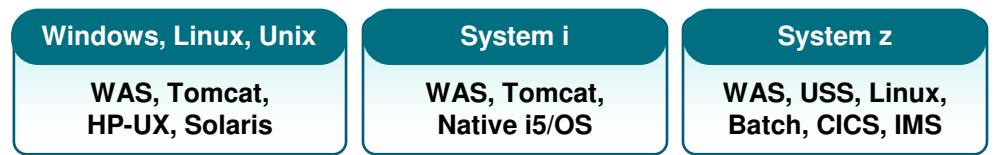
- Simplify delivery of modern User Interfaces
- Modern syntax appeals to today’s developers
- UML to EGL speeds Model Driven Development
- Leverage business developer skills
- Enhance integration to existing processing and services and improve application QOS
  - Deploying to WAS, CICS, IMS



### Write application and control logic with EGL



### Encapsulate existing or create new resources



IBM Rational Business Developer

# Consolidate your Team Infrastructure

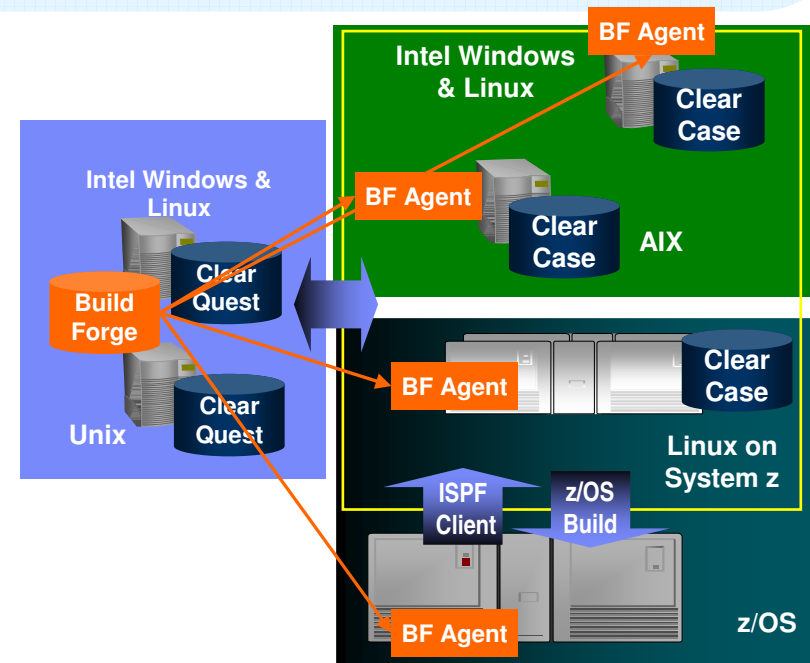
*Provide Coordination, traceability, consistency across platforms*

With **IBM Rational ClearCase v7.0.1, ClearQuest 7.0.1, and Build Forge v7.0.2** customers can reduce development costs with a single infrastructure for software development environments and processes across platforms and technologies; improve quality through improved traceability, automation of processes, and improved reporting; and benefits from lower costs due to elimination of duplicate processes and processes.

## New CRM Solutions

- ▶ **ClearCase 7.0.1**
  - Improved Build Processing
- ▶ **ClearQuest 7.0.1**
  - Enhanced lifecycle integration
- ▶ **BuildForge 7.0.2**
  - Driving distributed builds across platforms

- Provides:
  - single point of control, single artifact repository and set of user interfaces
  - Provides consistent process paradigm across platforms
- Extends power of SDP, Unified change management, and cross platform deployment to z/OS



IBM Rational Change and Release Management Solutions

# Questions





# Modern Application Development Featuring Web 2.0 for System z

Overview and Business Value of Web 2.0

**Rational.** software

# Agenda

- **Basics of Web 2.0**
  - Overview
  - Technology
- **Why Web 2.0?**
  - Business Value
  - Enterprise 2.0 Solution Patterns
  - Sample Business Scenarios
- **Enterprise Modernization and Web 2.0**
  - Identifying opportunities for your existing assets

Web 2.0!

## History of Web 2.0

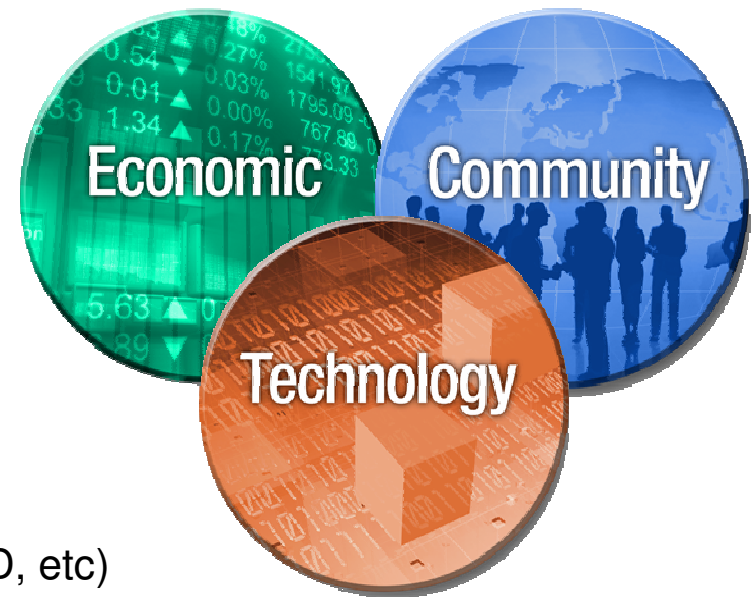
- Web 2.0 was first coined by O'Reilly Media in 2003.
- It was then popularized by the first Web 2.0 conference in 2004.
- According to Tim O'Reilly, "Web 2.0 is the **business revolution** in the computer industry caused by the move to the **internet as platform**, and an attempt to understand the rules for success on that new platform."
- Ultimately Web 2.0 services are expected to replace desktop computing applications for many purposes.



Tim O'Reilly

# What is Web 2.0?

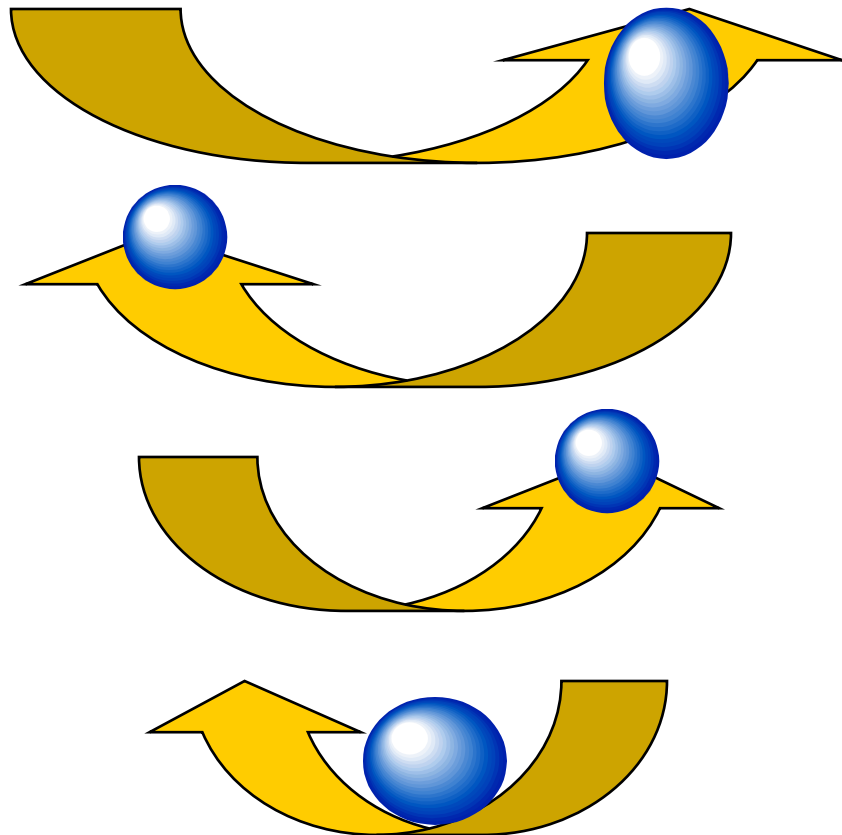
- **An important trend in delivering software applications**
- **An enabler for richer web applications**
  - New business models
  - Peer-to-peer user participation
  - New technologies
  - Interactive filtering, presentation, data entry
- **A combination of core technology components**
  - Rich user experience (maps, grids, animation, D&D, etc)
  - Loose-coupling, composite applications via reuse and “mash-ups”
  - Standards (SOAP, REST, JSON, Atom, etc)



## Web 2.0 technologies highlight the next pendulum swing between client and server function.

Client

Server



### Mainframe computing

“Dumb” green screen clients  
Omnipotent big mainframe servers

### Client-server computing

“Smart” Personal Computer clients  
Simple file and database servers

### Web (1.0) computing

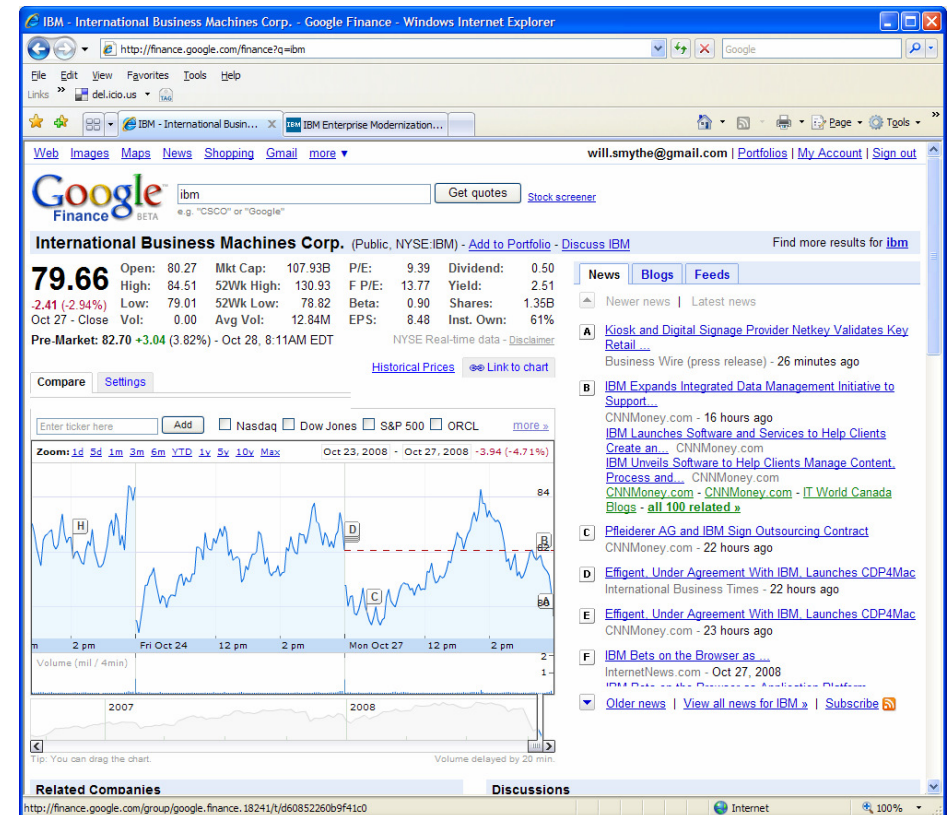
Light Web Browser clients  
Rich application and database servers

### Web 2.0 computing

Rich Internet Application clients  
Lighter application and database servers

# Web 2.0 / Rich Internet Application Characteristics

- Rich user experience
  - Minimal page transitions
  - Dynamic content
  - Data asynchronously retrieved via REST or SOAP service calls
  - Client-side validation
  - Simplified user interface
- User encouraged to add value
- Integration of relevant data from multiple sources



# Mashups

Combine content from more than one source into an integrated experience

**Google Map**

**Hotel information — separate database**

**Directions — come from somewhere else**

**Send to a phone — Additional functionality**

**Disney's Yacht & Beach Club Resorts Yacht Club**  
★★★★☆ 430 reviews - [more info](#)  
1800 Esplanade Blvd.  
Lake Buena Vista, FL 32830  
(407) 934-8000  
[go.com](#)

[Get directions](#) - [Search nearby](#)  
[Save to My Maps](#) - [Send to phone](#)

## Why?

- Rapid application development
- Reuse existing services
- Avoid reinventing the wheel
- Empowers users

# Technology Attributes of Web 2.0

Rich User Experience

**AJAX** incorporating: XHTML and CSS, DOM, XML and XSLT, XMLHttpRequest and JavaScript allowing information to be mashed up into new interactive portals."

Lightweight Programming Model

**XML or JSON data over HTTP**, in a lightweight approach sometimes referred to as **REST** (Representational State Transfer) as an alternative to SOAP.

Info-ware

**"DATA is the new HTML."** Database management is a core competency of Web 2.0 companies.

Feeds

**RSS/ATOM** allows someone to link not just to a page, but to **subscribe** to it, with notification every time that page changes.

Perpetual Beta

Users must be treated as co-developers, in a reflection of open source development practices. The open source dictum, **"release early and release often"**





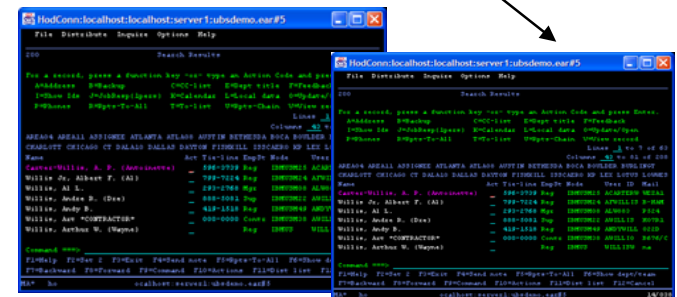
Google Finance

# Value of Existing Applications

**30 million lines of code could be worth \$3 billion!**



- Existing applications are among the most valuable assets a company owns
  - Fully functional applications that run the business today
  - Significant investments over the years
  - Fine-tuned for dependable optimal performance
  - Run within fine-tuned reliable, robust, scalable IT infrastructure and platforms
- Accessed via character / text based terminals
  - Non-intuitive, difficult to navigate user interfaces
  - Difficult to integrate in modern business processes
- 5X less expensive to reuse existing applications than to write new applications from scratch
- Reusing proven, time-tested applications can result in significantly lower risk and faster time to market



# Why Web 2.0 for the Enterprise?

- **Web 2.0 is**

- Creating new markets
- Lowering competitive barriers
- Encouraging creativity to come from anywhere
- Harnessing community created media
- Tapping into the wisdom of the crowds
- Enhancing communications and making information more impactful

- **What does it means for the enterprise?**

- Greater collaboration and innovation across the value chain
- Increased levels of customer intimacy
- Simplification of complex IT and business infrastructure
- Business model flexibility to capitalize on new market opportunities

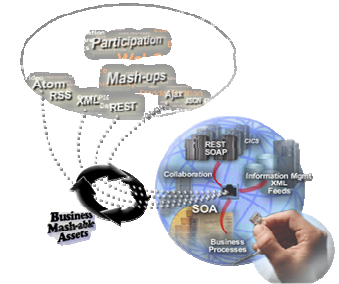
*It's fast becoming a Web 2.0 business world, but innovation never goes out of style.*

**According to Gartner:**

By 2008, the majority of Global 1000 companies will have adopted several technology-related aspects of Web 2.0

Missing out on the non-technology aspects of Web 2.0 means that many organizations will also miss out on some of the positive business benefits

# New Business Opportunities



- **For example...**

- Leverage Web 2.0 in your business critical applications
- Investigate customization, integration and services opportunities in rolling out web 2.0-based community collaboration environments.


- **Benefits to your customers:**

- Instant business value through rapid new product deployment, with reduced costs
- Reduction in time to market opens new business opportunities
- Reducing costs through customer self-service
- Unlocking valuable business data opens new business opportunities
- Extracting value by collecting and sharing the wisdom in the organization

## Modernization with Web 2.0

*Reuse what you already have!*

- Build a new Web front-end for your existing order entry application, but continue to use your proven business logic.
- Create a self-service portal that provides your customers with details about past and business logic to perform a calculation.
- Expose a portion of current orders.
- Mashup information from multiple CRM systems.
- Reuse existing COBOL customer data locked away in a green screen application to another team in your organization.
- **Web 2.0 is not just for Internet-accessible applications – can be used to improve productivity of internal users!**



# Modern Application Development Featuring Web 2.0 for System z

Real Estate Demo Application Overview

**Rational.** software

# Overview

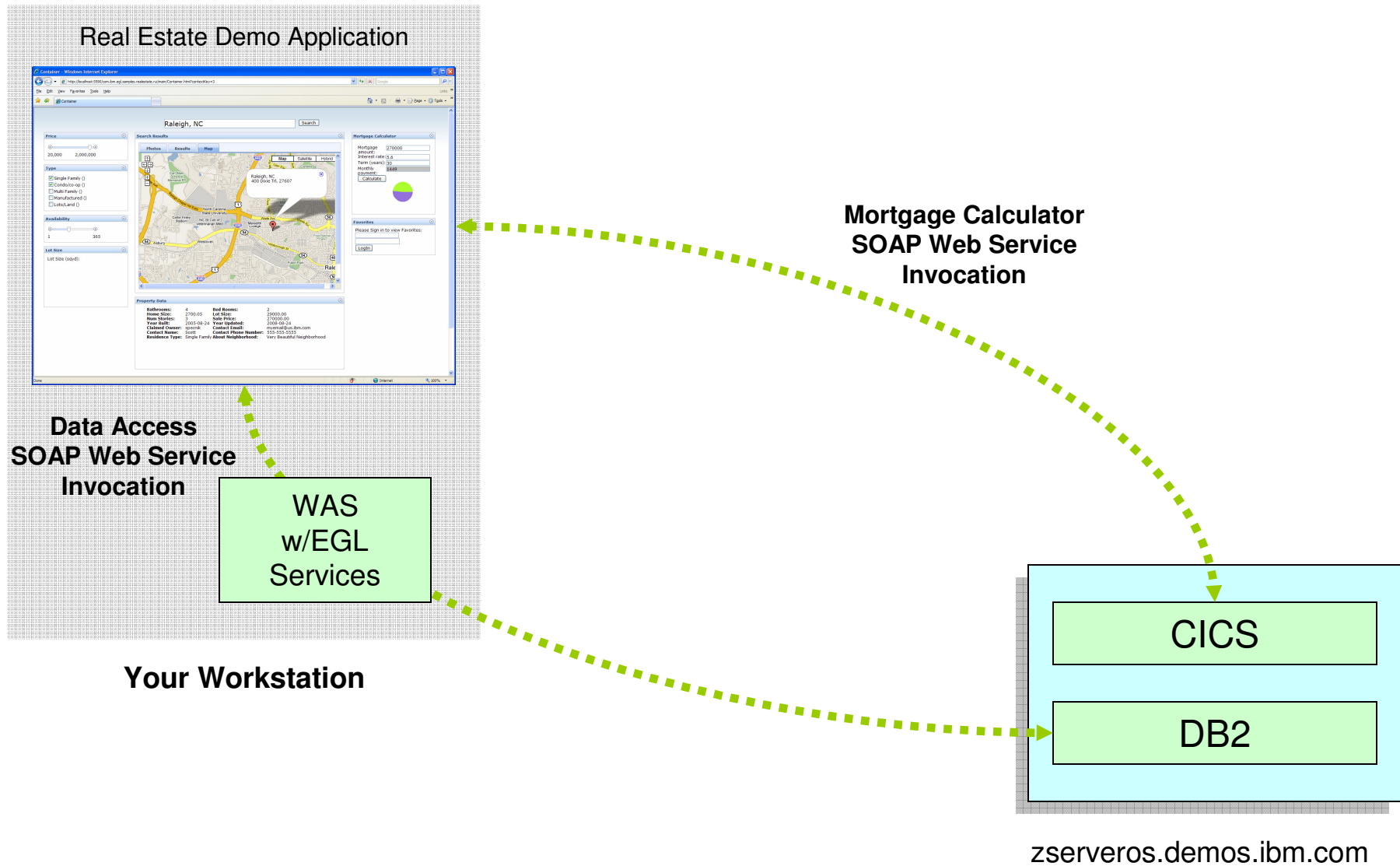
- Throughout the day, we will be working with a sample real estate application that has been created for this seminar. This application shows how existing data and applications can be easily exposed in a new Web 2.0-style application.

The screenshot shows a web browser window titled "Container - Windows Internet Explorer" displaying a real estate application. The address bar shows the URL: `http://localhost:5590/com.ibm.epl.samples.realestate.ru/main/Container.html?contextKey=3`. The application interface includes a search bar with "Raleigh, NC" entered, a "Search" button, and a "Search Results" section with tabs for "Photos", "Results", and "Map". The "Map" tab is active, showing a map of Raleigh, NC, with a red pin marking a property at "Raleigh, NC 400 Dixie Trl, 27607". To the left of the map are filter panels for "Price" (range 20,000 to 2,000,000), "Type" (checkboxes for Single Family, Condo/co-op, Multi Family, Manufactured, Lots/Land), "Availability" (range 1 to 365), and "Lot Size" (input field). To the right of the map is a "Mortgage Calculator" with fields for Mortgage amount (270000), Interest rate (5.6), Term (years) (30), and Monthly payment (1449), along with a "Calculate" button and a pie chart. Below the map is a "Property Data" section with the following information:

<b>Bathrooms:</b>	4	<b>Bed Rooms:</b>	2
<b>Home Size:</b>	2700.05	<b>Lot Size:</b>	29000.00
<b>Num Stories:</b>	3	<b>Sale Price:</b>	270000.00
<b>Year Built:</b>	2005-08-24	<b>Year Updated:</b>	2008-08-24
<b>Claimed Owner:</b>	specnik	<b>Contact Email:</b>	myemail@us.ibm.com
<b>Contact Name:</b>	Scott	<b>Contact Phone Number:</b>	555-555-5555
<b>Residence Type:</b>	Single Family	<b>About Neighborhood:</b>	Very Beautiful Neighborhood

At the bottom right of the application is a "Favorites" section with a "Please Sign in to view Favorites:" message and a "Login" button. The browser's status bar at the bottom shows "Done" and "Internet" with a 100% zoom level.

# Architecture Overview







## Real Estate Demo Application

The background features a series of thin, light blue lines that flow and curve across the frame, creating a sense of motion and depth. The overall color palette is various shades of blue, from deep navy to light sky blue.

## **Modern Application Development Featuring Web 2.0 for System z**

Discover and Transform Application Assets

**Rational.** software

# Discover and Transform Application Assets

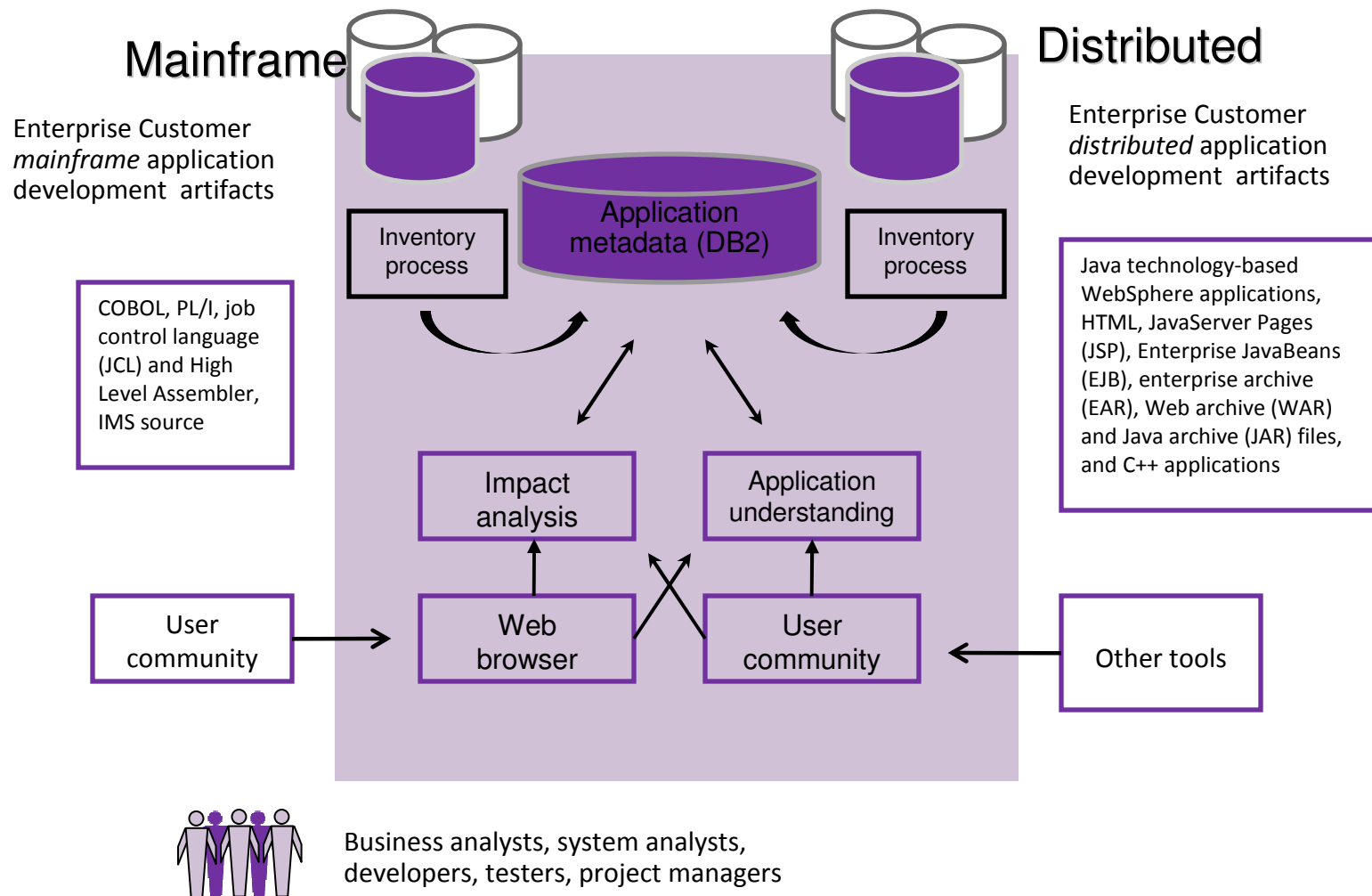
- The purpose of this section is to show how solutions from IBM Rational can help you discover and reuse existing assets in modern applications.
- **Agenda**
  - WebSphere Studio Asset Analyzer Overview
  - Rational Asset Analyzer Overview
  - Rational HATS Overview and Demo

## What is WebSphere Studio Asset Analyzer (WSAA)?

- Provides a browser-based interface to search, explore and report on the gathered information
- Support for System z, enterprise wide software applications
- Scans source “where it lives”
- Assists in the development and test phases to identify impacts across the enterprise and identifying areas to test, thereby reducing risk and improving test coverage
- Identifies relationships among the artifacts
  - Gathers metrics, counts related to the IT artifacts in the enterprise

# WebSphere Studio Asset Analyzer

## Architectural View



# What is Rational Asset Analyzer (RAA)?

## A sister product of WSAA

- Based on proven WSAA technology
- Support for the same languages/subsystems as WSAA

### WSAA

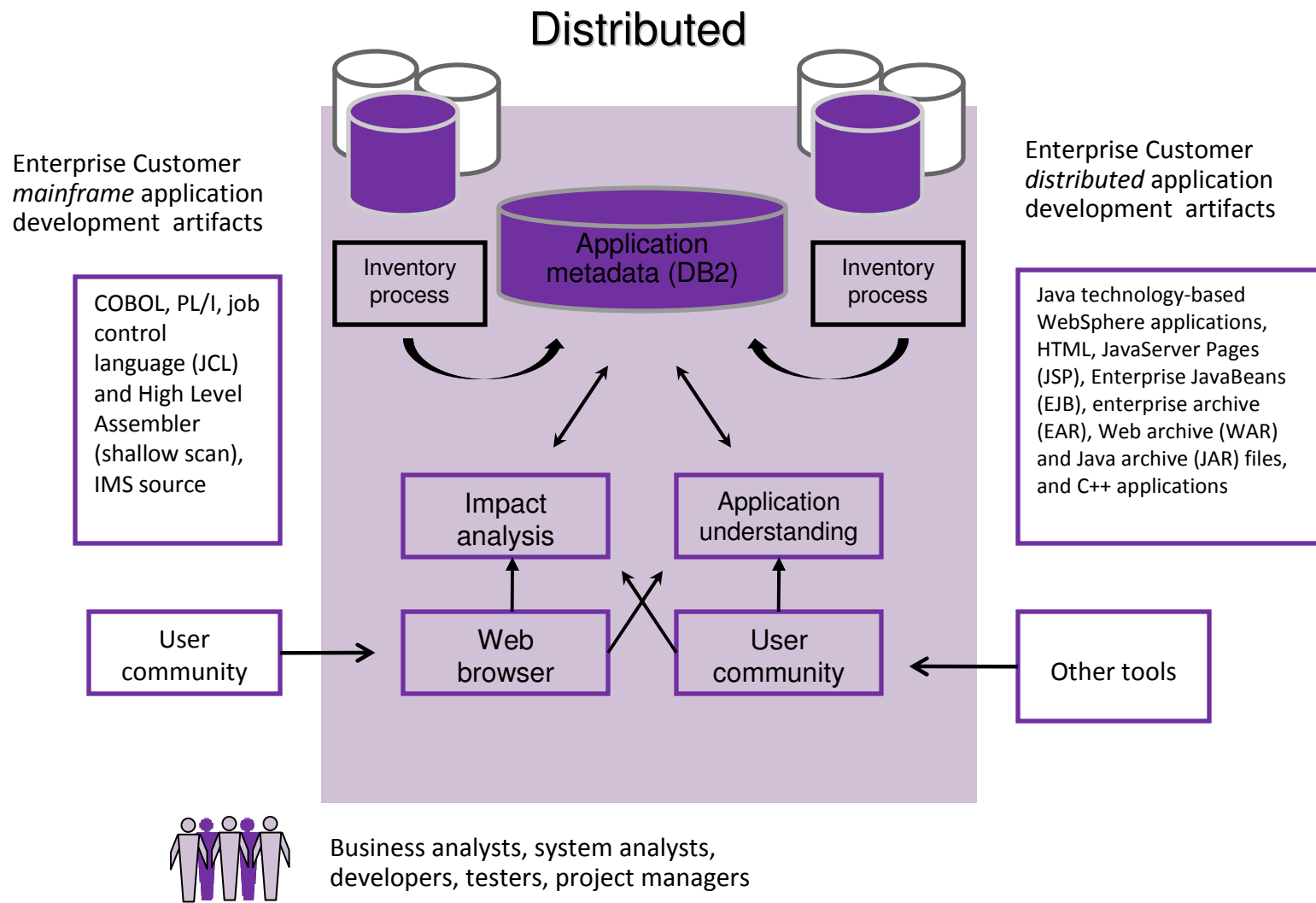
- Installs on z/OS
- Scans source “where it lives” - on z/OS, Windows and AIX
- Typically used for large teams
- Browser-based user interface
- Identifies relationships among the artifacts
- Gathers metrics, counts related to the IT artifacts

### RAA

- **Installs on Windows**
  - **Scans source on Windows**
  - Can be used for large **or small teams**
  - Browser-based user interface
  - Identifies relationships among the artifacts
  - Gathers metrics, counts related to the IT artifacts
- 
- **Dashboards for counts & metrics**
  - **Ability to control depth of inventory scanning**
  - **Very fast directory scans to gather counts & metrics quickly**
  - **Eliminates need for DB2 on System z**
  - **RESTful API (documented in External Interface Guide)**

# Rational Asset Analyzer

## Architectural View



# RAA – Home Page

**Home - Rational Asset Analyzer - Microsoft Internet Explorer**

**Rational. Asset Analyzer**

Home Explore Impact analysis Database ?

Search enterprise assets: [input] Go  Type mixed case

**Welcome to Rational Asset Analyzer!**

To begin, consider looking over the items in the **Getting started** section of the Task Help located at the right.

Common assets Metrics - overview Metrics - detailed Metrics - dashboard

**Common assets Total**

Common assets	Total
Application	5
Container	676
File	100760
Impact analysis	0
Site	2

**Lines in file**

Language	Percentage
Text /js	11.4%
C/C++ /c	17.8%
Java /java	21.6%
XML /xml	27.1%

**File size (bytes)**

Category	Percentage
Unknown /svn-	34.6%

**Task Help**

- Getting started
  - Learning terms used in Rational Asset Analyzer
  - Learning the types of users
  - Navigating Rational Asset Analyzer
- Taking inventory [All](#)
- Customizing Rational Asset Analyzer
- Exploring assets - common [All](#)
- Exploring assets - distributed [All](#)
- Exploring assets - MVS [All](#)
- Using applications
- Determining the impact of changes [All](#)
- Annotating your assets
- Creating reports
- Using Composite Applications
- Getting additional information

Done **RAA status area** Jul 9, 2008 6:31:23 PM Internet



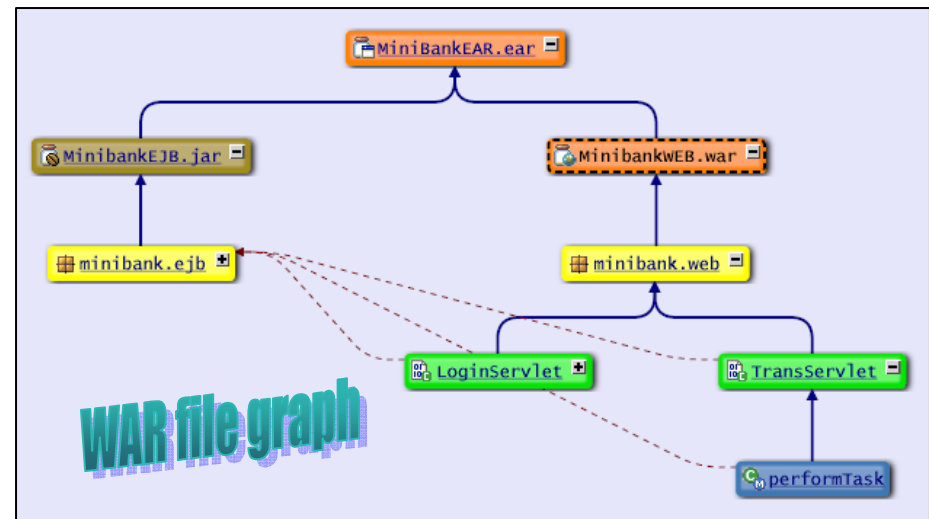
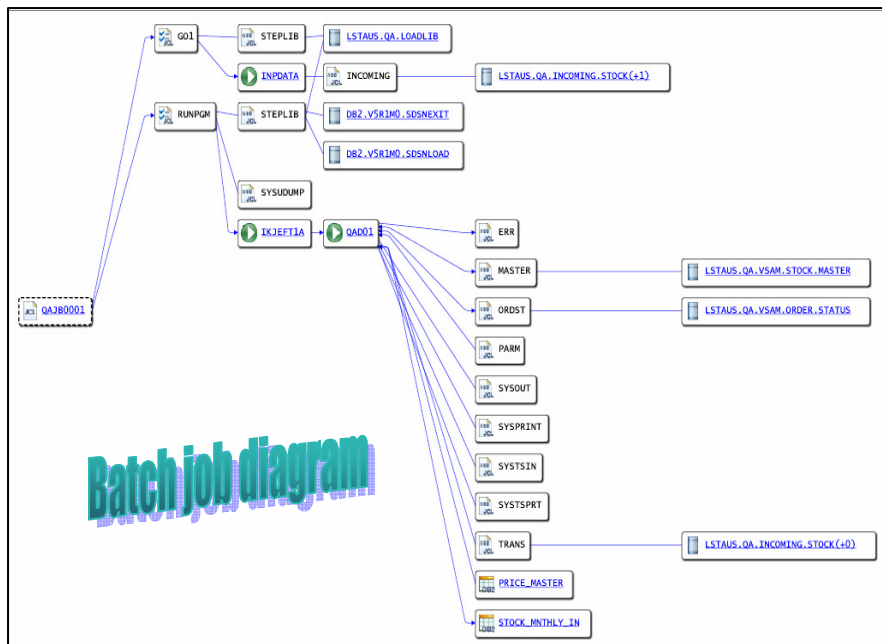
# The Value of Asset Analyzer

- **Accelerate project delivery in response to business drivers**
  - Reduce risk
  - Increase productivity
  - Improve quality of application changes
- Gain **intellectual control** of applications
- Gain **transparency** into outsourced development
- **Customize** RAA to organizational processes and IT environments

# Rational Asset Analyzer – Application Understanding

*Quickly understand code with little or no documentation, and relationships across the enterprise*

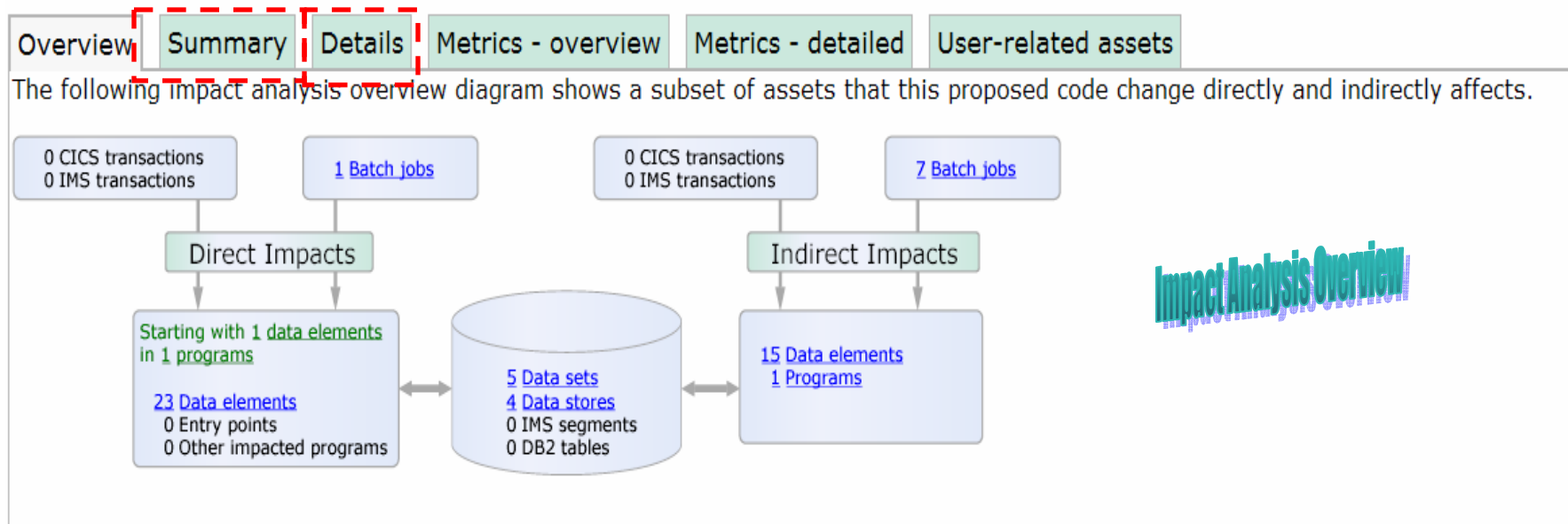
- Group artifacts into **user-defined groups** called Applications\_QA to limit scope to area of interest
- Use **various types of diagrams** for understanding how the application “hangs together”
- Use **annotations** to capture knowledge from SMEs e.g. Business function, description, etc.
- Create **user-defined relationships** for situations where relationships cannot be determined through static analysis
- Perform **enterprise-level keyword searches**



# Rational Asset Analyzer – End to End Impact Analysis

*Reduce time to market & risk of downtime by understanding change impact upfront*

- Reduce time to determine **scope of change** whether for new enhancements, or even maintenance efforts
- View the metrics for impacted artifacts to determine the **risk of change** i.e. cyclomatic complexity, lines in file, etc.
- Traverse user-defined relationships to determine **impacts across platforms** i.e. follow dependencies from mainframe to J2EE and back.
- Create a “**bill of materials**” of impacted artifacts by evaluating the details page



# Rational Asset Analyzer – Synergies

## *RAA enhances value of other tools*

- **Rational Developer for System z (RDz)**

- *Combine productivity gains of RDz with the enterprise-level insight in RAA to minimize time-to-market and minimize risk due to lack of visibility into impacted artifacts*

- **Rational Asset Manager (RAM)**

- *Use RAA to identify artifacts/assets of interest, then use RAM to publish those assets for reuse, and manage/govern the development process around changes to those assets. That is, jumpstart RAM deployments for a quicker ROI.*

- *Use RAA to perform deeper level analysis for change requests coming in to RAM.*

- **Rational Transformation Workbench (RTW)**

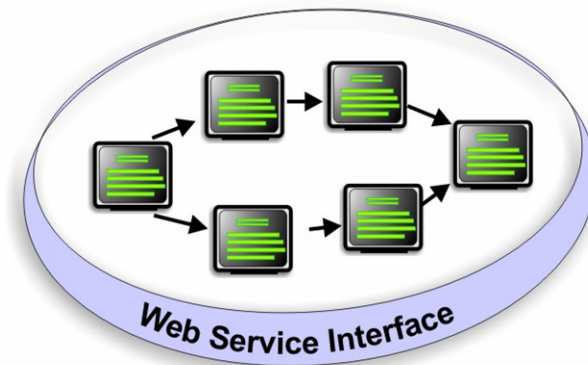
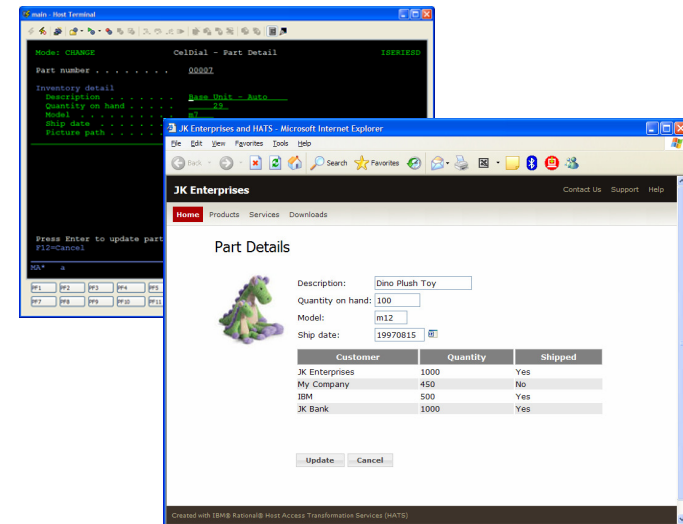
- *Use RAA to gain visibility and control at enterprise level, then*

- ▶ *Use RTW Business Rules Extension to extract, document and manage business rules.*

- ▶ *Use RTW Architect Extension to identify and remove dead code, and accelerate componentization efforts.*

# IBM Rational HATS

- **What can you do with HATS?**
  - Quickly and easily transform green screen applications to the Web, Portal, mobile device, or rich client without rewriting
    - Easy to use
    - Highly customizable
    - Iterative development process
    - Transformation “on the fly”
  - Extend terminal application tasks as Web services
- **Benefits**
  - Extend host applications to new users
  - Improve the navigation of your host applications
  - Reuse your existing assets in a Service Oriented Architecture
  - Low risk

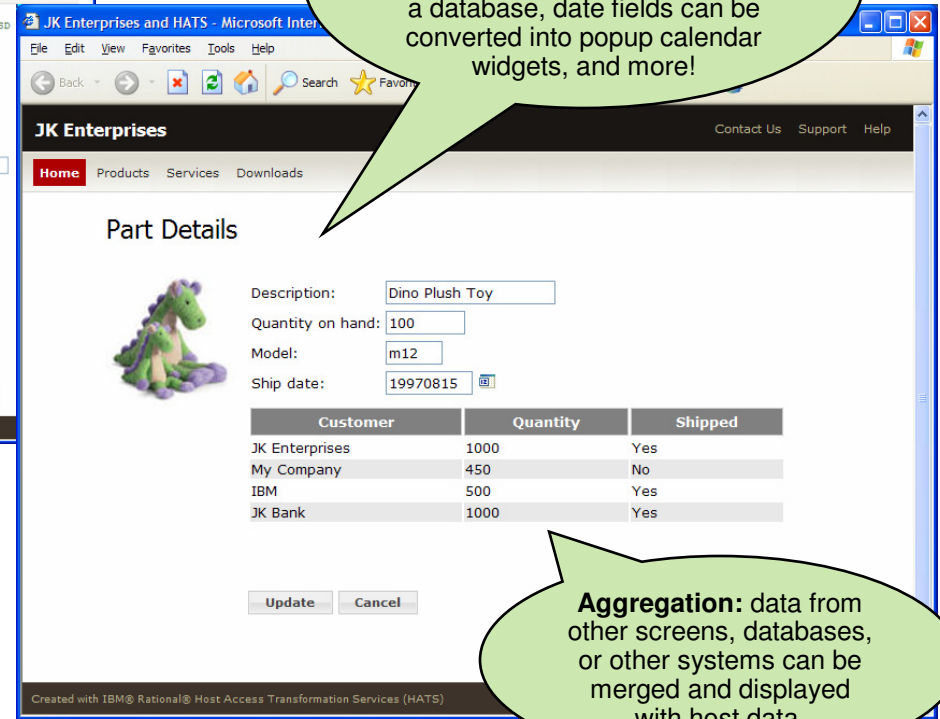
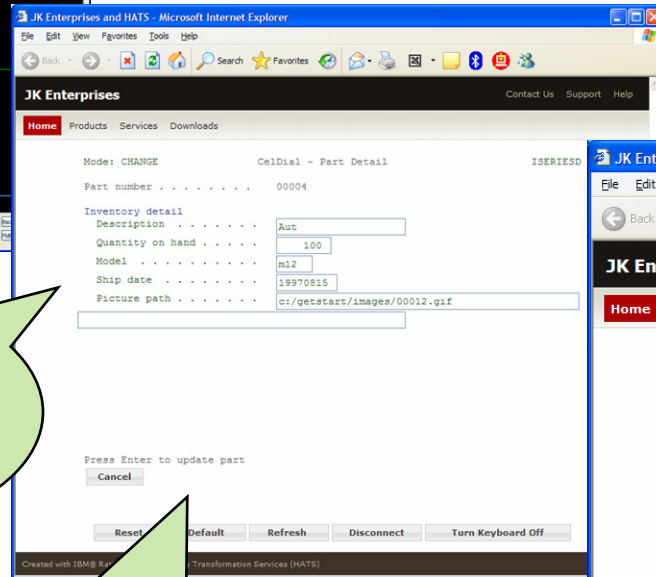
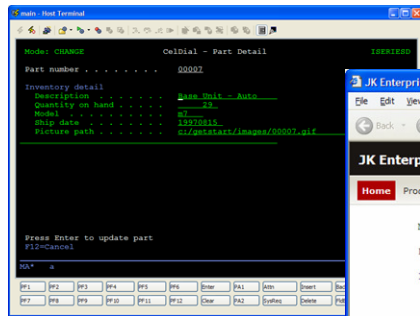


# Modernization At Your Own Pace

Today

Day 1

Tomorrow



**Modern UI:** All pages share the same theme and style, which can be based on an existing Web site.

**Instant Value:** HATS default rendering automatically transforms actions into clickable links and buttons. No changes required to the host application.

**Integration:** Text can be replaced with images, input fields can be populated from data on other screens or from a database, date fields can be converted into popup calendar widgets, and more!

**Aggregation:** data from other screens, databases, or other systems can be merged and displayed with host data.

# Business Value of HATS

- **Improved appearance, usability and navigation of existing, proven applications**
  - Reduce transaction times
  - Reduce training time
  - Reduce employee turnover
- **Speed to web, with less cost and less risk**
  - Reusing existing assets rather than re-writing
  - No changes to existing applications
  - Build on existing knowledge and skills
- **Externalize legacy transactions and data**
  - Reach new end users, markets, sales channels through web self-service
  - 24 X 7 access
  - Viewable, printable
- **Build self-service transactions**
  - Improve the availability of information and transactions
  - Off-load call centers, voice response units and clerical employees
- **Integrate information from multiple applications and databases “at the glass”, into Portals**
- **Make existing applications “integration ready”, using Web Services to link existing applications to new web applications**

# Environments

*Modernizing green screen applications without rewriting, refactoring, or reengineering*

## Browser



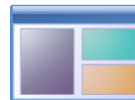
- Zero footprint
- HTML and JavaScript only (no applet or browser plug-in required)
- Access using all popular browsers, including Internet Explorer and Firefox.

## Rich Client



- Integration at the desktop with other Eclipse-based applications
- Client side processing
- Rich set of user interface widgets
- Support for Lotus Expeditor

## Portal



- Integration at the glass
- Cooperative portlet support
- JSR 168 compliant

## Mobile



- Access host applications from mobile devices
- Extend applications to mobile users

## Web Services



- Build self-service transactions
- Expose host business processes as Web Services
- Provide controlled access to vital host applications and host data.

- *Quick ROI : Host applications can be quickly deployed with a new user interface*
- *Low cost : No need to rewrite application*
- *Low risk : Leverage open, proven platforms and technologies*
- *Increase productivity and reduce training costs*
- *Improving work flow from multiple applications*
- *Provides integration of host business processes and data with other Web, portal, and rich client applications*

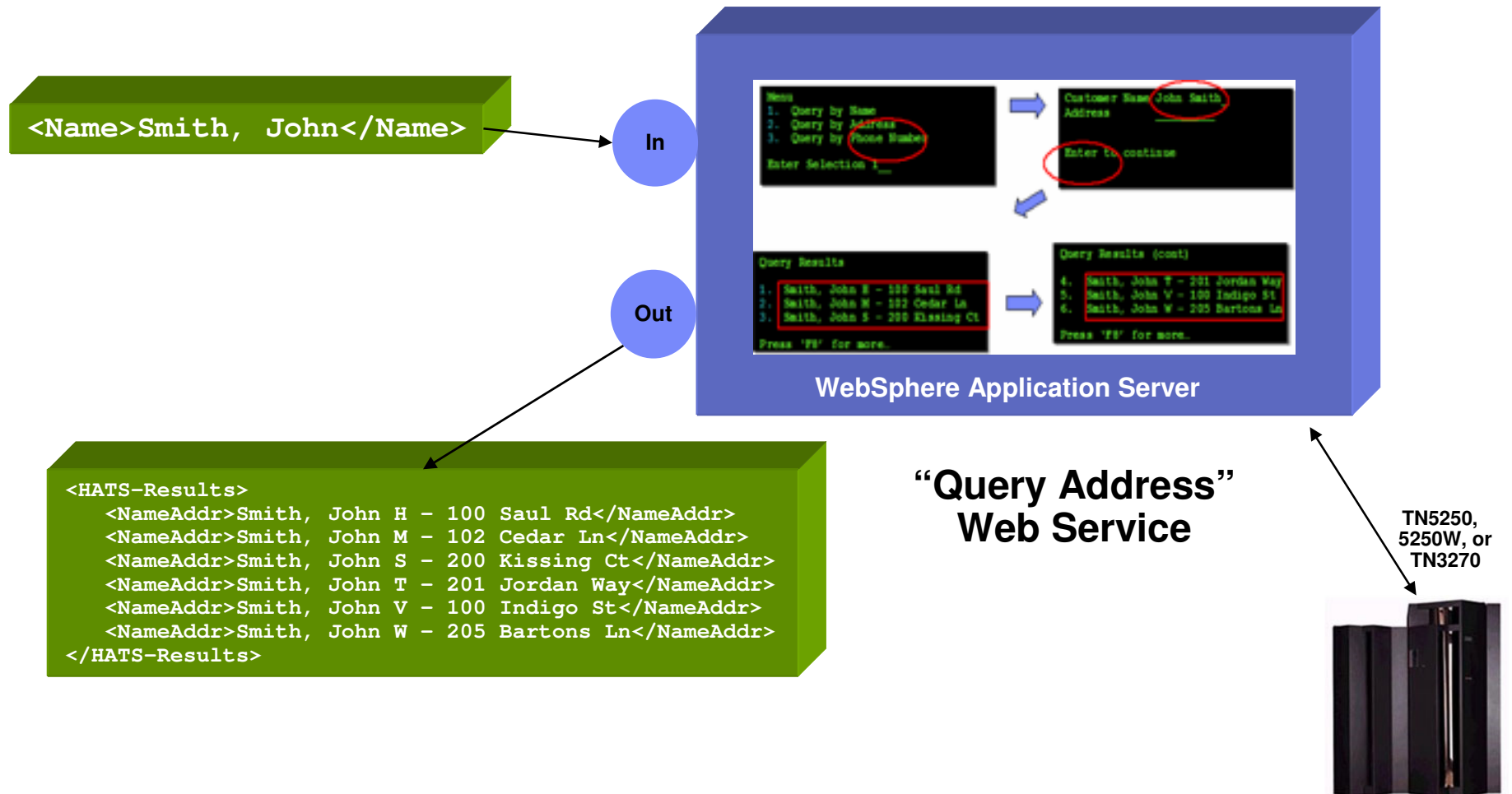


# The Value of HATS SOA

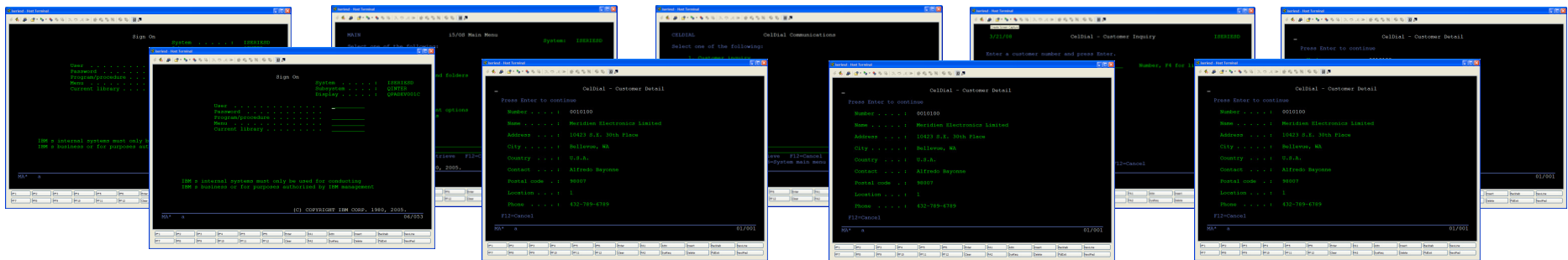


- HATS does not require application reengineering or rewriting
  - Rewriting can consume significant amounts of time and resources.
  - Rewriting requires access to and understanding of application source code.
  - Rewriting inherently introduces new bugs that must be found and fixed.
  - Simply put - rewriting is not an option for some customers!
- HATS is unique
  - HATS Web Service-enables 3270, including CICS and IMS, 5250, and VT terminal applications – great for customers with a mix of applications (one tool can be used to enable all of them).
- HATS generates standard Web Service interfaces
  - HATS services can be integrated into an existing SOA.

# How It Works - A simple example



# Hiding Screen Navigation Complexity

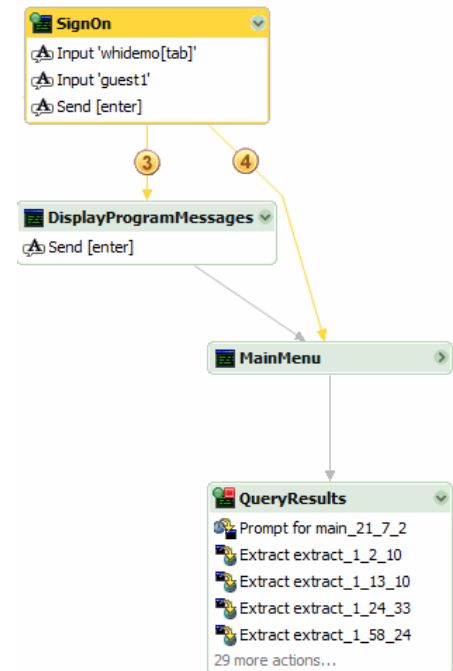


**Input:** Customer number

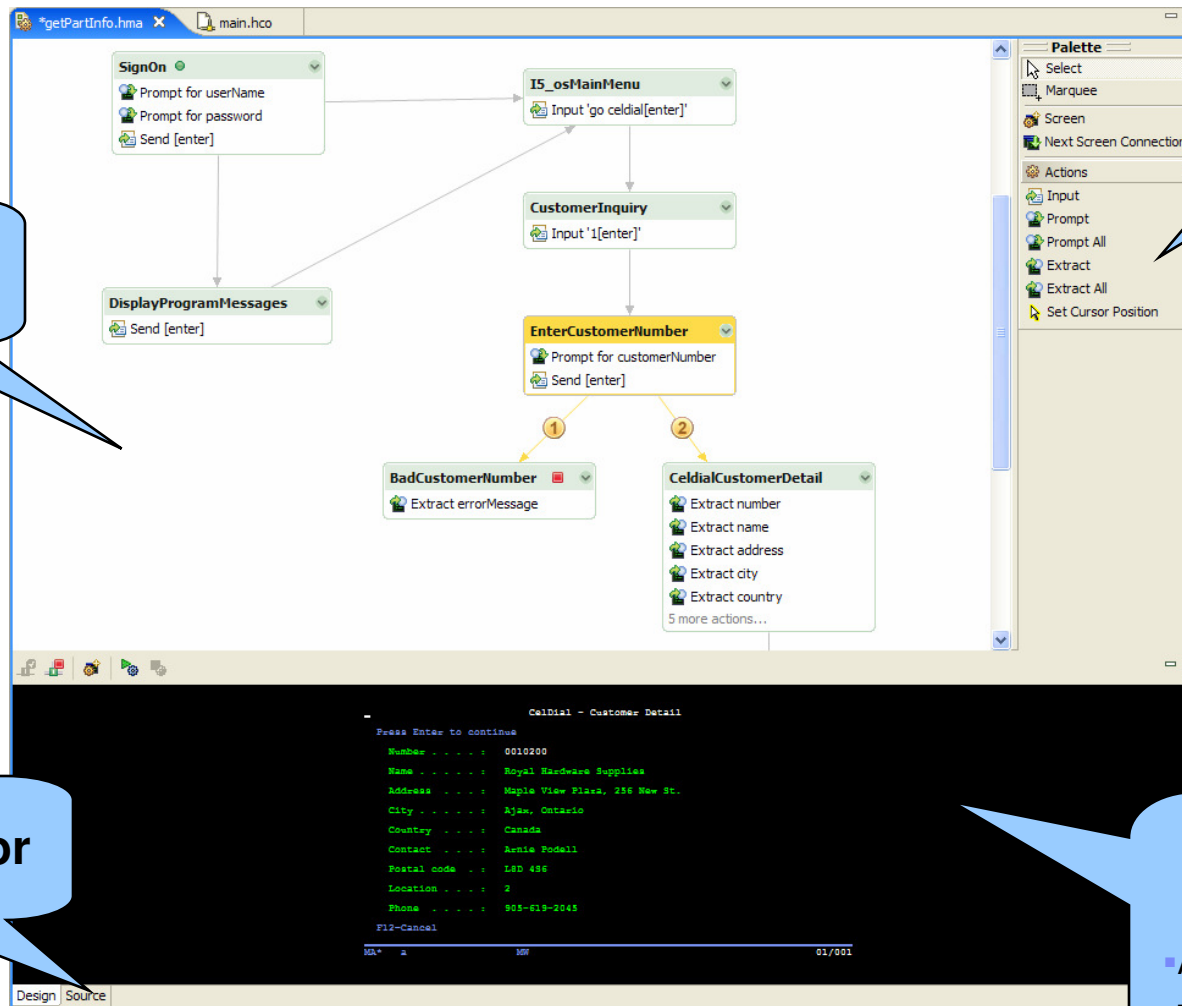
**Output:** Customer name, address, city, state, zip, phone, contact name, past order history, etc.

# Visual Macro Editor (VME)

- **The VME is tool for visually developing HATS macros**
  - Macros are the scripts that navigate and interact with 3270, 5250, VT host applications.
  - Macros are the foundation for the Web Service support provided by HATS.
  - Inputs and outputs in a HATS macro are eventually manifested as inputs and outputs in a Web Service (screen interactions are hidden behind the Web Service interface).
- **The VME provides value by:**
  - Significantly increasing the productivity of HATS macro developers.
  - Allowing for easier handling of alternate / error flows.
  - Helping decrease the number of logic / flow problems (since the flow can be clearly analyzed by both developers, expert users, and business analysts).
- **Fully supported in HATS 7.5 (November 2008)**



# HATS Visual Macro Editor



**Design Canvas**

**Palette**

- Add new screens and actions.
- Connect screens together.

**Source Editor**

**Integrated Terminal**

- Add new screens
- Test macro



HATS

The background of the slide is a vibrant blue with a series of thin, white, wavy lines that create a sense of motion and depth. The lines are most prominent on the left side and curve towards the right, creating a dynamic, organic feel. The overall color palette is various shades of blue, from deep navy to light sky blue.

## **Modern Application Development Featuring Web 2.0 for System z**

Rational Developer for System z

**Rational.** software

# Agenda

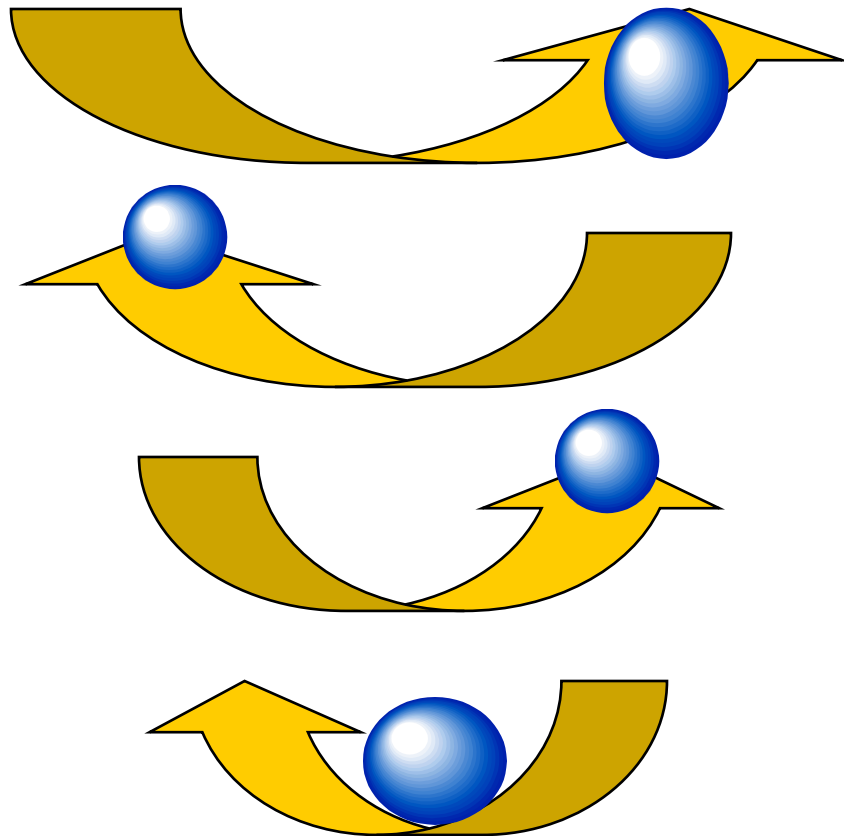
- **Introduction**
- **Architectures**
  - Multi-tier Web Architecture
  - Web 2.0 – the web as a platform
- **Client technologies**
  - HTML, CSS, Javascript, AJAX, JSON
- **Connectivity**
  - Web Services, XML, SOAP, REST WSDL, Mashups
- **Business Tier**
  - CICS
  - COBOL



## Web 2.0 technologies highlight the next pendulum swing between client and server function.

Client

Server



### Mainframe computing

“Dumb” little green screen clients  
Omnipotent big mainframe servers

### Client-server computing

“Smart” Personal Computer clients  
Simple file and database servers

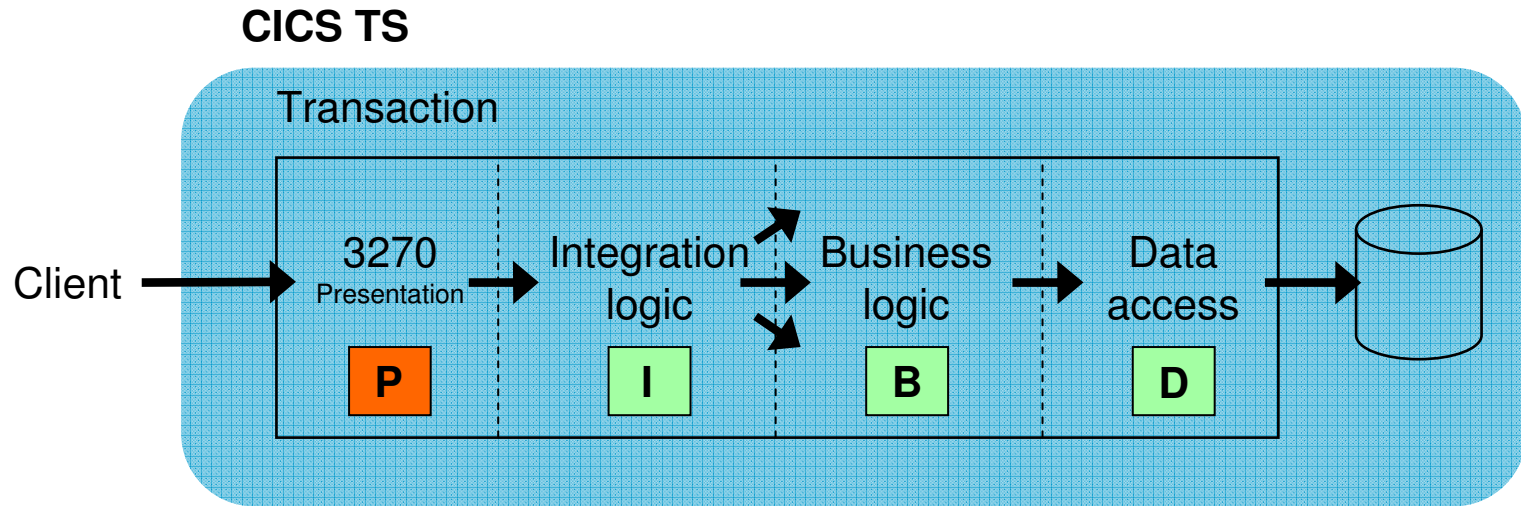
### Web (1.0) computing

Light Web Browser clients  
Rich application and database servers

### Web 2.0 computing

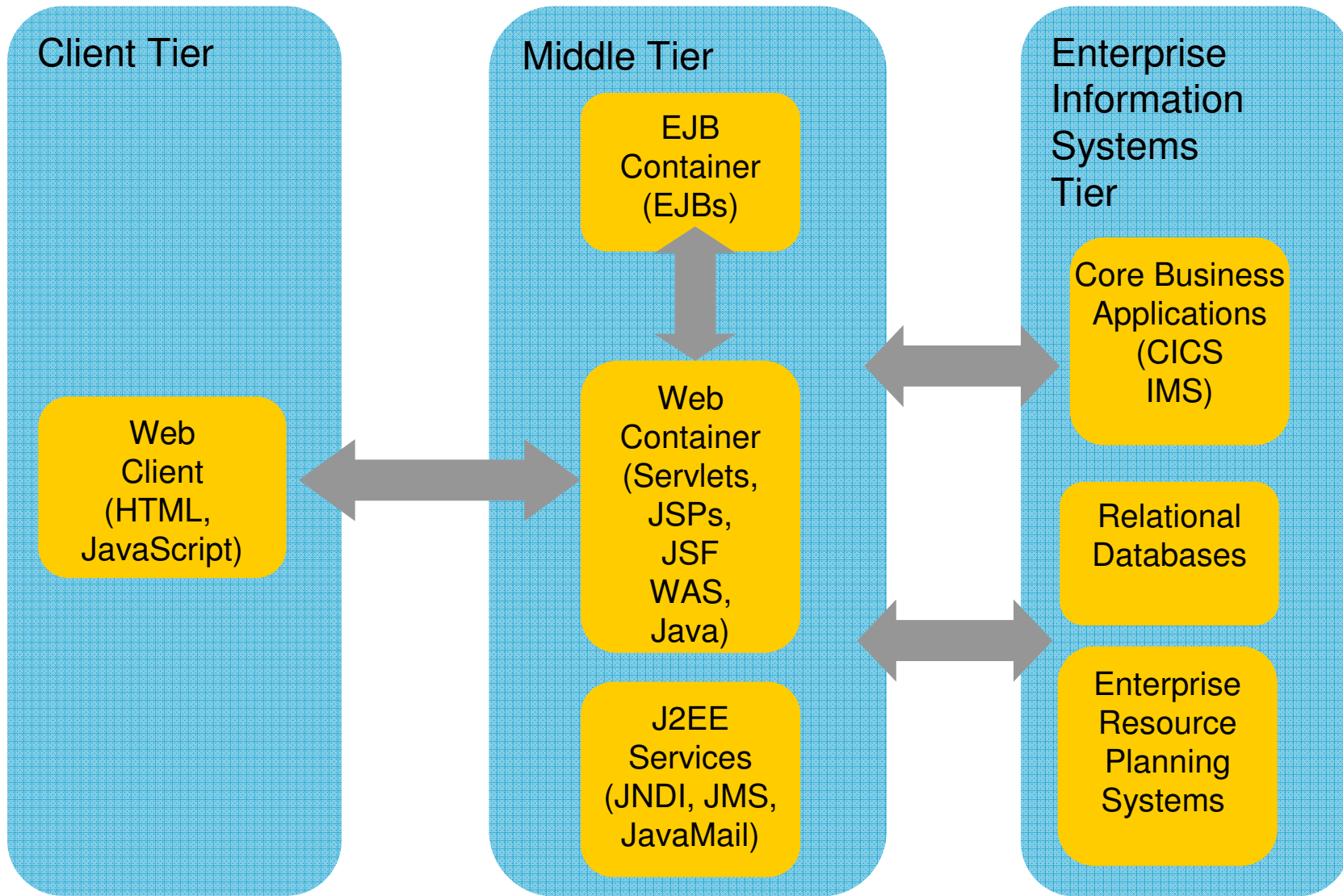
Rich Internet Application clients  
Lighter application and database servers

# Modern “CICS” architecture



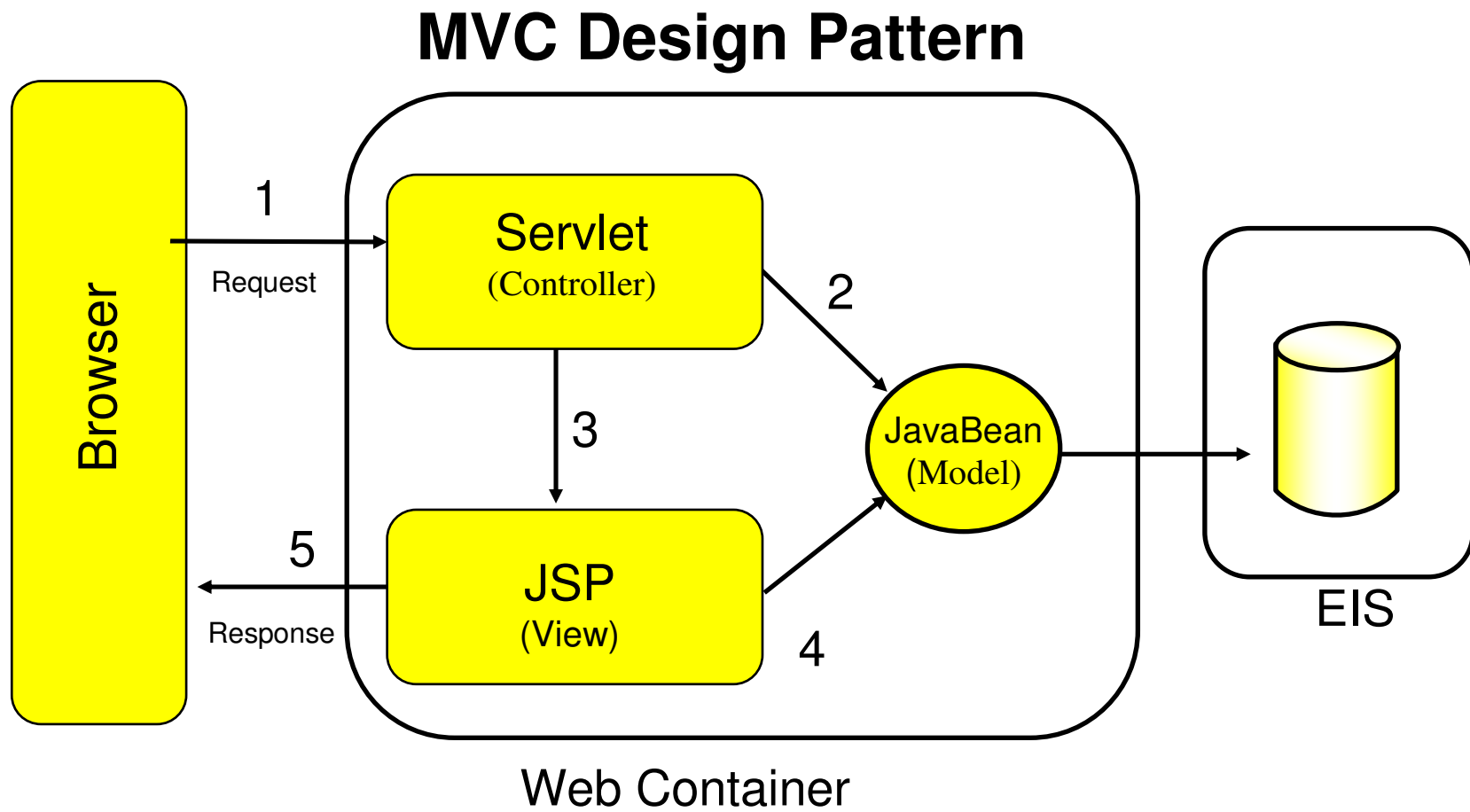
- **Best practice in CICS application design is to separate key elements of the application, in particular:**
  - Presentation logic                      3270, HTML, XML
  - Integration or aggregation logic      Menu, router, tooling
  - Business logic                              COBOL, PL/I, Reusable component
  - Data access logic                          VSAM, DB2, IMS, ...
- **Provides a framework for reuse and facilitates separation of concerns, clear interfaces, ownership, and optimisation**

# Web 1.0 Architecture

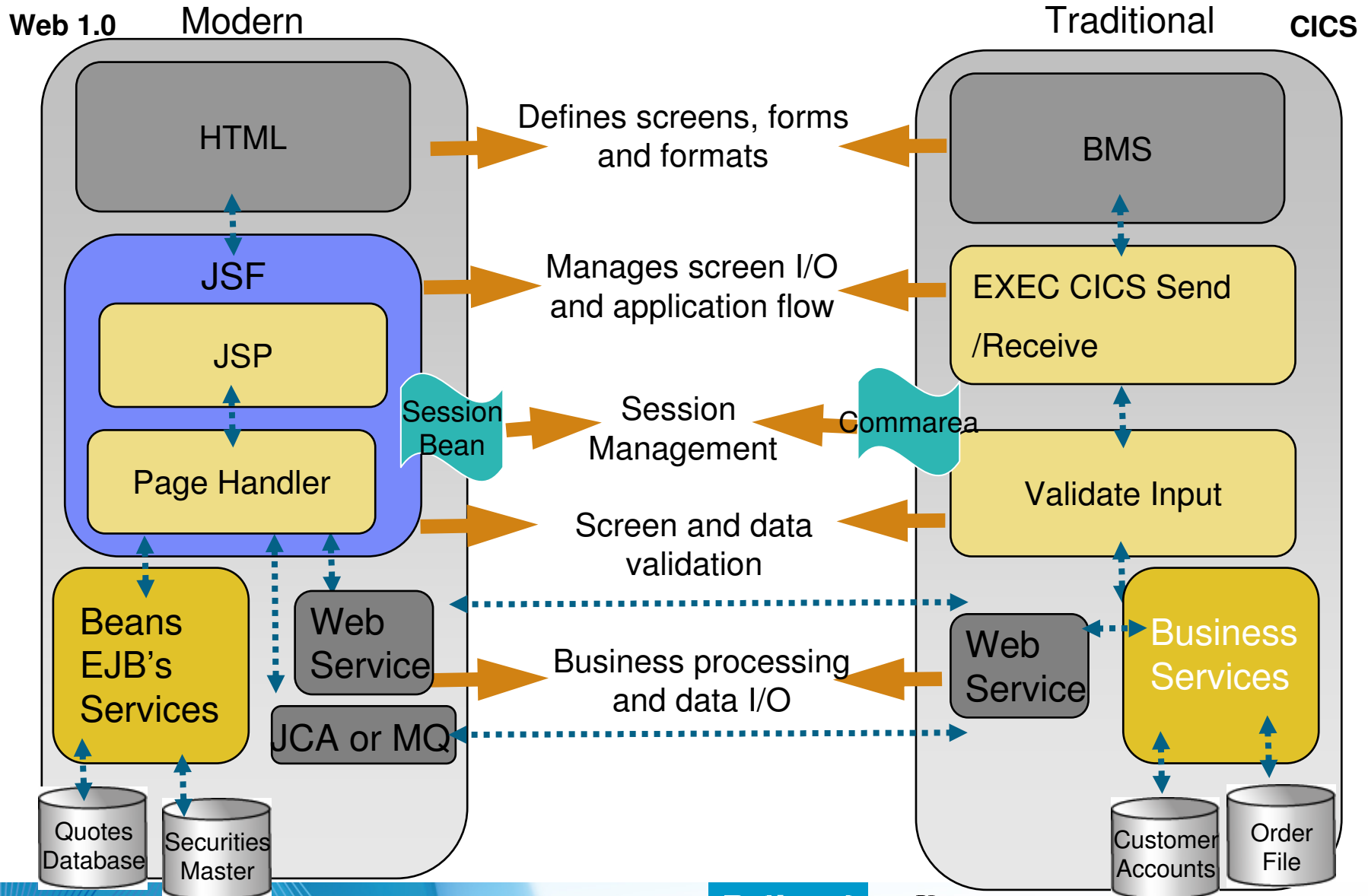


## Web 1.0 Application Model

- A request is sent to a servlet that generates dynamic content and calls a JSP page to send the content to the browser, as shown:



# It's not that different



# Web 1.0 Example – TODO list

## Page 1



### Example List

[Add another item](#)

1. User clicks “Add item” link
2. Controller locates “Add item” business logic in model
3. Server retrieves data to display
4. Server builds page 2 and formats with view with input box
5. User gets new page and types in “New TODO item”. Clicks “Add item” button sending add-data to the server
6. Controller receives add-item data from client and locates *ADD* business logic in model
7. Server updates model data and retrieves other data for display
8. Server builds page 3 and formats with view to display new TODO list

## Page 2



### Example List

New TODO item

[+ Add this item](#) [Close](#)

## Page 3

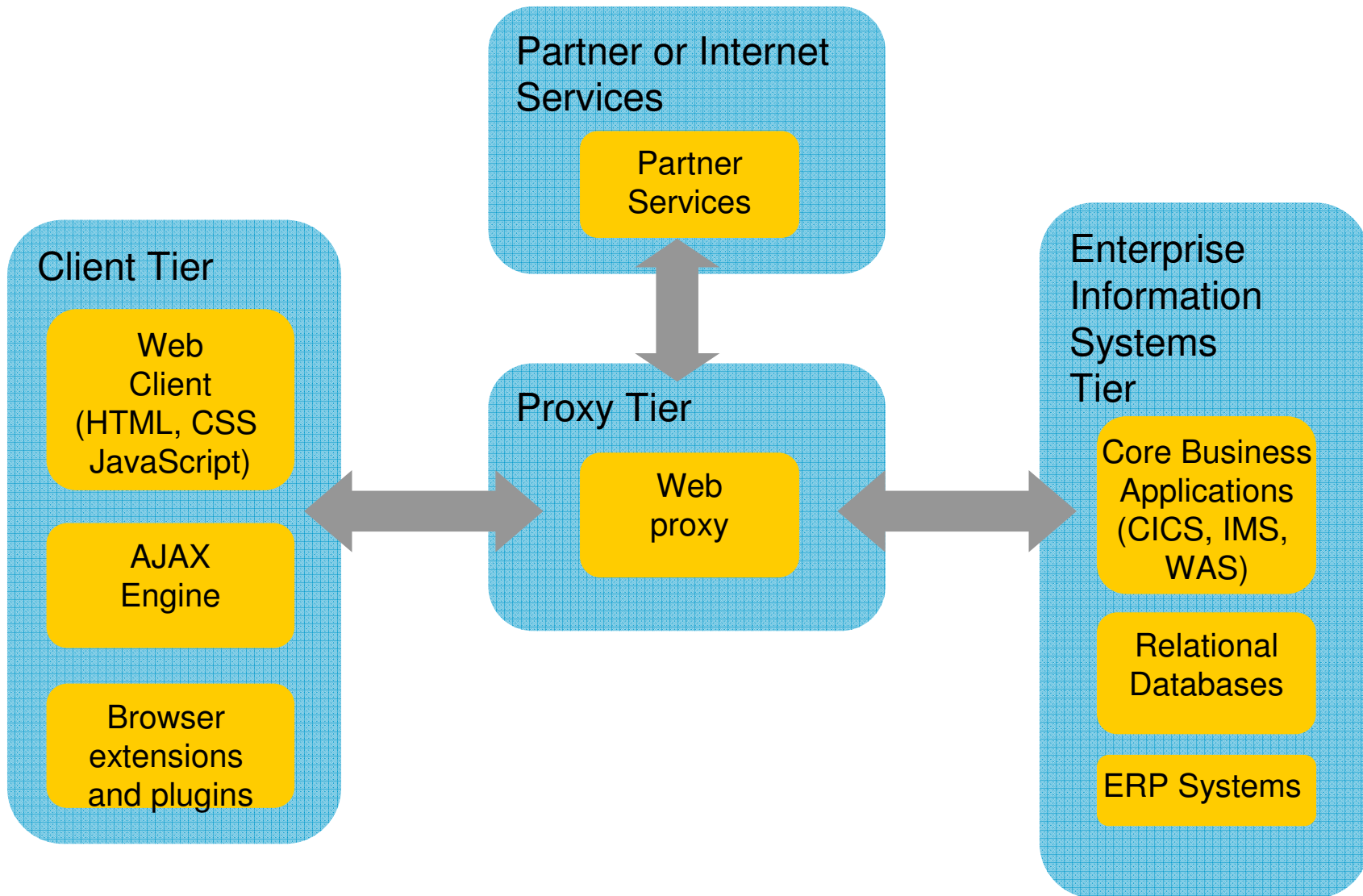


### Example List

New TODO item

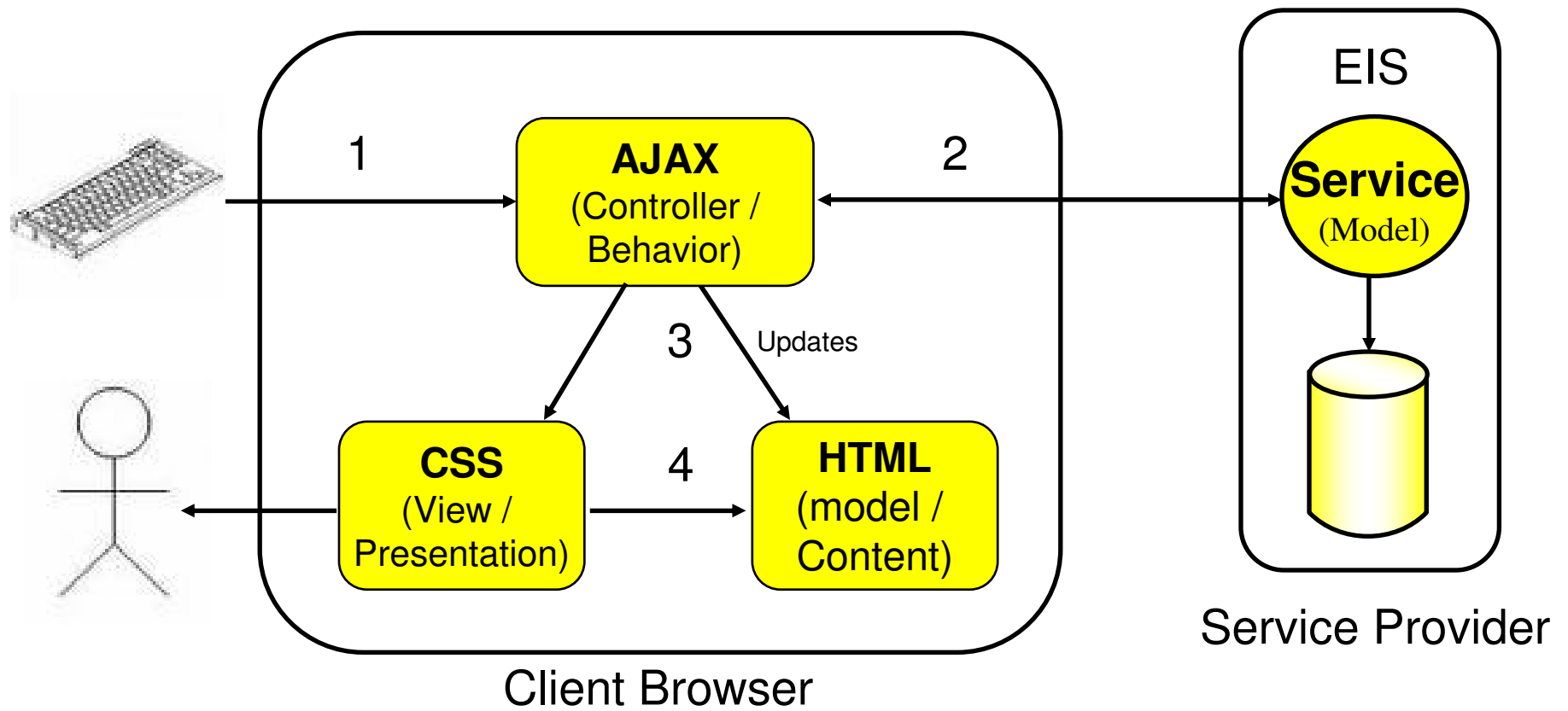
[Add another item](#)

# Web 2.0 Architecture



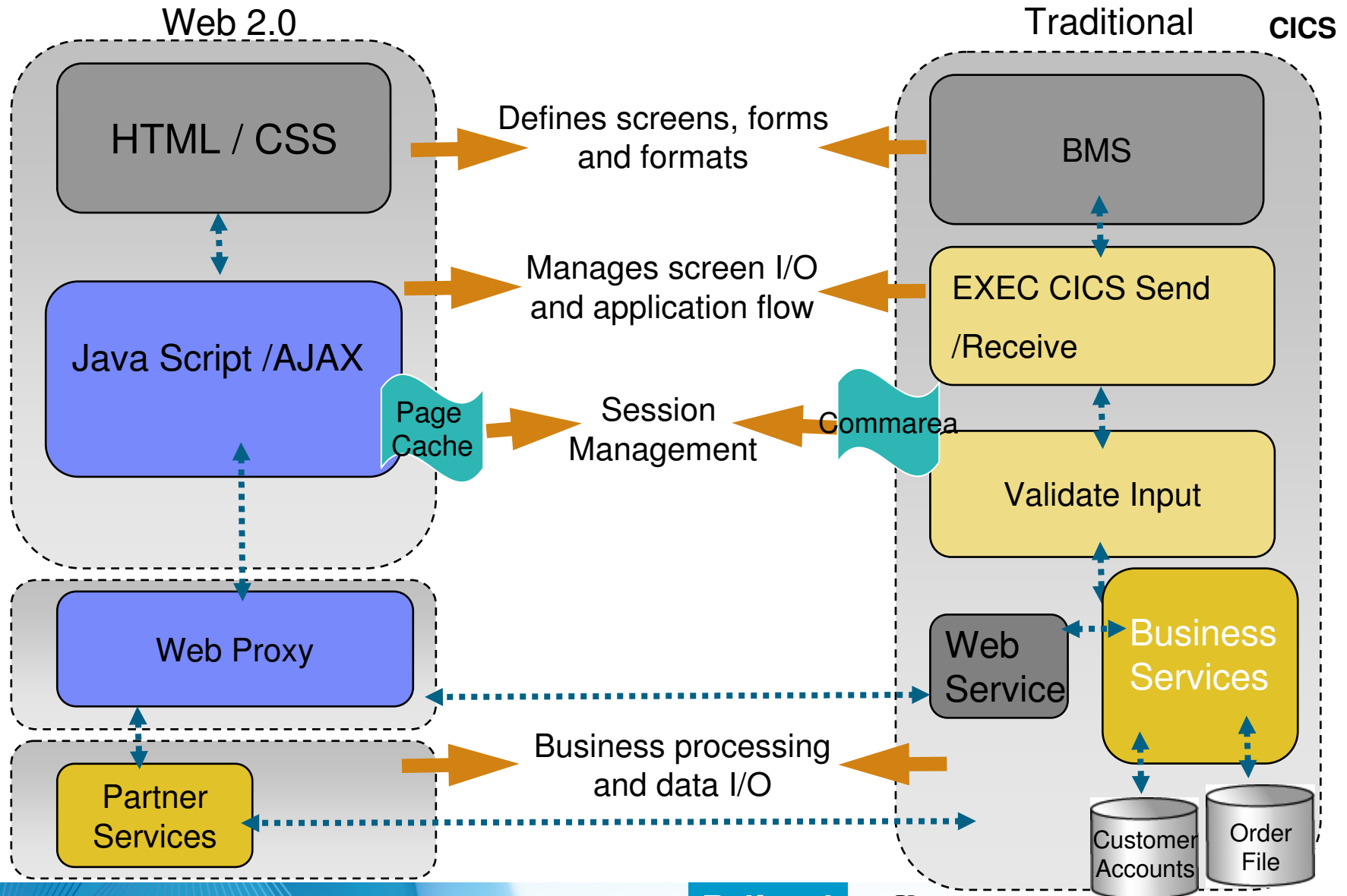
# Web 2.0 Application Model

- User enters data on a page. The new data is formatted into a service call, and sent to the service provider. The response is returned to the client and the page is updated.



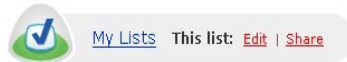


# It's not that different



# Web 2.0 Example – TODO list

## Page 1



### Example List

[Add another item](#)

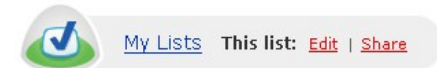
## Page 2



### Example List

[+ Add this item](#) [Close](#)

## Page 3





### Example List

New TODO item

[Add another item](#)

1. User clicks “Add item” link
2. Script is executed
3. Client creates a new input box dynamically and adds it to the page
4. User types in “New TODO item” and clicks “Add this item” button.
5. The new-item name is added into a *ADD* service call and sent to the service provider
6. Service provider acknowledges that the *ADD* was completed
7. Client dynamically removes the input box and creates a new item to display on the TODO list

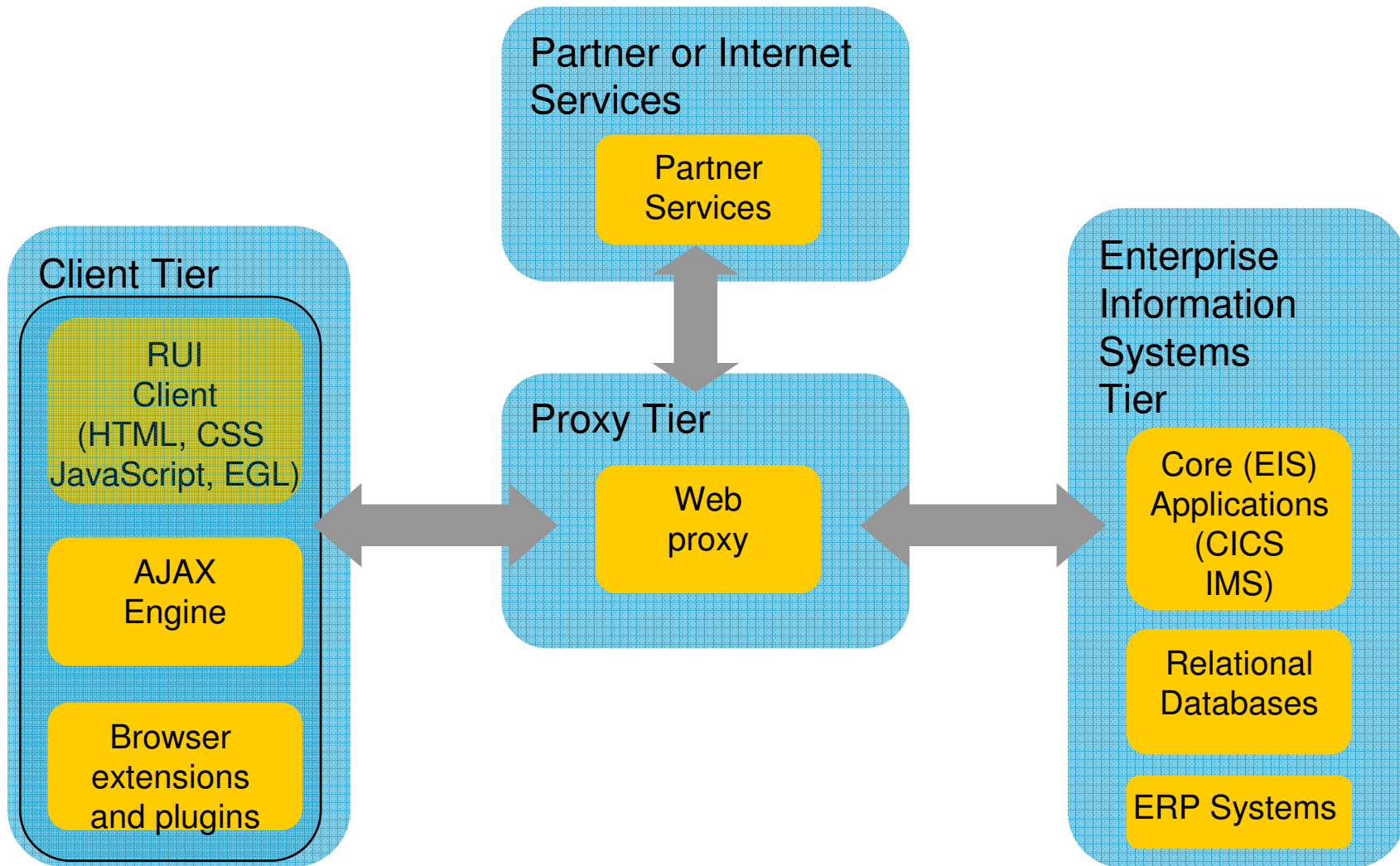
# What is RichUI?

- **RichUI stands for Rich User Interface.**
  - This is a phrase commonly used when talking about an interface that provides dynamic rendering of its individual parts – notably, on the client-browser, as opposed to server-side processing
- **It is a technology that will allow developers of any background to create rich web pages like one would see on a leading-edge, interactive site, such as:**
  - www.digg.com 
  - www.hulu.com 
- **You might also have heard the term: “Rich Internet Application” (RIA)– which is often used synonymously with RichUI.**
- **The benefits of RichUI or RIA include:**
  - Improved user-responsiveness
    - The most successful RichUI implementations can achieve almost a “Windows-desktop” look and feel to users
  - “Rich-er” functionality – beyond the simple rendering of HTML, to include dynamic widgets and components
  - Improved browser/server load-balancing – as more of the business functionality can be distributed to the desktop (browsers)

## RichUI and Client Processing

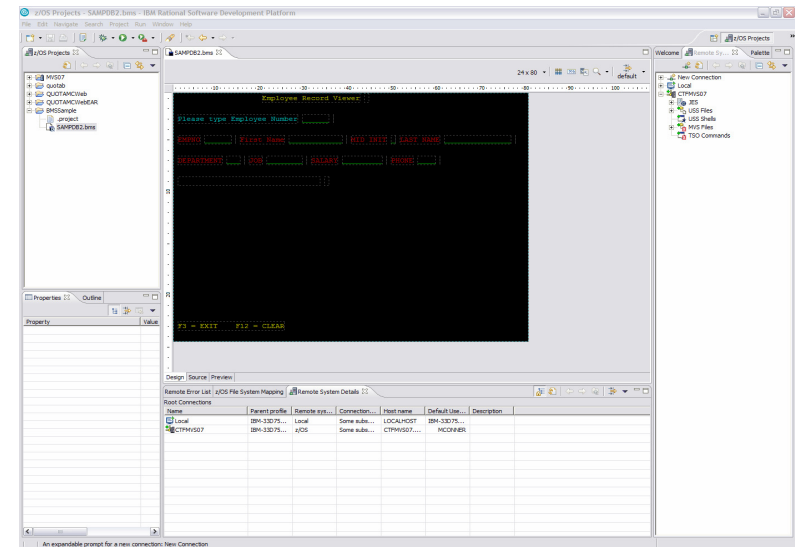
- **RichUI leverages the generation capabilities of EGL; JavaScript is the runtime**
  - JavaScript is a language that can only be run in a browser, (FireFox, Internet Explorer, Opera, etc.) – not on the server like EGL-generated COBOL or Java
- **It is JavaScript that renders your page in the browser and manipulates labels, data, graphics and controls the page's behavior**
  - No static HTML is created
  - JavaScript does all the work
- **Much of the U.I. is also implemented using leading-edge internet technologies such as:**
  - AJAX
  - DOJO
  - JSON
  - FLEX
  - Web Services

# Web 2.0 Architecture



# HTML

- **HTML performs similar processing as BMS or MFS maps. It defines the screens and fields, colors, and interactions, although the technologies and implementations of course are different.**
- **Hypertext Markup Language consists of:**
  - **Hypertext.** The way of creating web documents – and of linking multiple documents together. HTML offers support for both document as well as multimedia links.
  - **Tags or controls.** Pieces of code that are used to create links. All browsers let you know when you've selected an active area of the screen.
    - For example <head> marks where a heading starts and </head> marks where it ends.
    - Popular tags include:
      - Text Tags – Logical structure for content
      - Link Tags – to links such as hyperlinks, image links
      - Style sheet tags – how content is rendered
      - and many more....
- See the green screenshot – displayed inside of the RDz BMS Map Editor, together with with the BMS Macros that are input to generate the code – that upon execution causes the “green screen” to be displayed.
  - RDz provides similar support for HTML screens



# HTML vs BMS

## BMS

Name and overall format of map - Includes items such as input/output, whether keyboard should be enabled, types of terminal, colors, size etc. are defined.

### SAMPDB2 DFHMSD

```
TYPE=&SYSPARM,MODE=INOUT,LANG=COBOL,STORAGE=AUTO, *
CTRL=FREEKB,EXTATT=YES,TERM=3270-2,TIOAPFX=YES, *
MAPATTS=(COLOR,HILIGHT,OUTLINE,PS,SOSI), *
DSATTS=(COLOR,HILIGHT,OUTLINE,PS,SOSI)
```

```
MAP1 DFHMDI SIZE=(24,80), *
COLUMN=1, *
LINE=1
```

Headings and text fields. Defined with DFHMDF macro. You see position, length, initial value, and field attribute below.

```
DFHMDF POS=(3,1),LENGTH=27, *
INITIAL='Please type Employee Number', *
ATTRB=(PROT,NORM)
```

Input Fields. Defined with DFHMDF macro. You see a name (which ultimately defines storage size (and Cobol copybook field definition), and a difference with the field defined as unprotected – information can be entered.

```
EMPONUMINPUT DFHMDF POS=(3,29),LENGTH=6, *
ATTRB=(UNPROT,NORM),HILIGHT=UNDERLINE
```

## HTML

Headings – Overall definition, including whether Java Server faces tags will be used, a heading, and stylesheet definition.

### <HEAD>

```
<%@ taglib uri="http://java.sun.com/jsp/core" prefix="f"%>
<%@ page language="java" contentType="text/html; charset=CP1252"
pageEncoding="CP1252"%>
<META http-equiv="Content-Type" content="text/html; charset=CP1252">
<META name="GENERATOR" content="IBM Software Development Platform">
<META http-equiv="Content-Style-Type" content="text/css">
<LINK href="theme/Master.css" rel="stylesheet" type="text/css">
<TITLE>MAP1</TITLE>
```

Text headings including location definition, colors, attributes, etc.

### f:view> <BODY>

```
<hx:scriptCollector id="scriptCollector1"><h:form styleClass="form" dir="ltr"
id="form1"><table><tr><td colspan="20">&nbsp;</td>
<td colspan="22" nowrap><font color="#ffff00">Employee Record Viewer</font></td>
<td>&nbsp;</td>
<td nowrap><font color="#0000ff"></font></td>
<td colspan="36">&nbsp;</td>
<tr><td colspan="80">&nbsp;</td>
<tr><td>&nbsp;</td>
<td colspan="27" nowrap><font color="#00ffff">Please type Employee
Number</font></td>
```

Input fields

```
<td>&nbsp;</td>
<td colspan="6" nowrap><h:inputText
value="#{pc_MAP1Page.map1Bean.emponumininput}" required="false"
style="color: #00ff00" size="6" id="emponumininput"></h:inputText></td>
<td>&nbsp;</td>
<td colspan="44">&nbsp;</td>
<tr><td colspan="80">&nbsp;</td>
<tr><td>&nbsp;</td>
```

# Cascading Style Sheets (CSS)

- Define colors, fonts, and formatting in a modular manner
- Able to include on tags, individual (named) elements, or classes of element
  - Flow hierarchically
- Dynamically change style via JavaScript

```
h1 {  
  border: 1px dashed red;  
}  
.clazz {  
  font-family: arial, helvetica;  
  font-size: 12pt;  
  margin-left: 10px;  
  color: green  
}  
#named {  
  font-style: italic;  
  font-family: Courier;  
  background: lightgrey;  
  border: none;  
}
```

Default

using a class

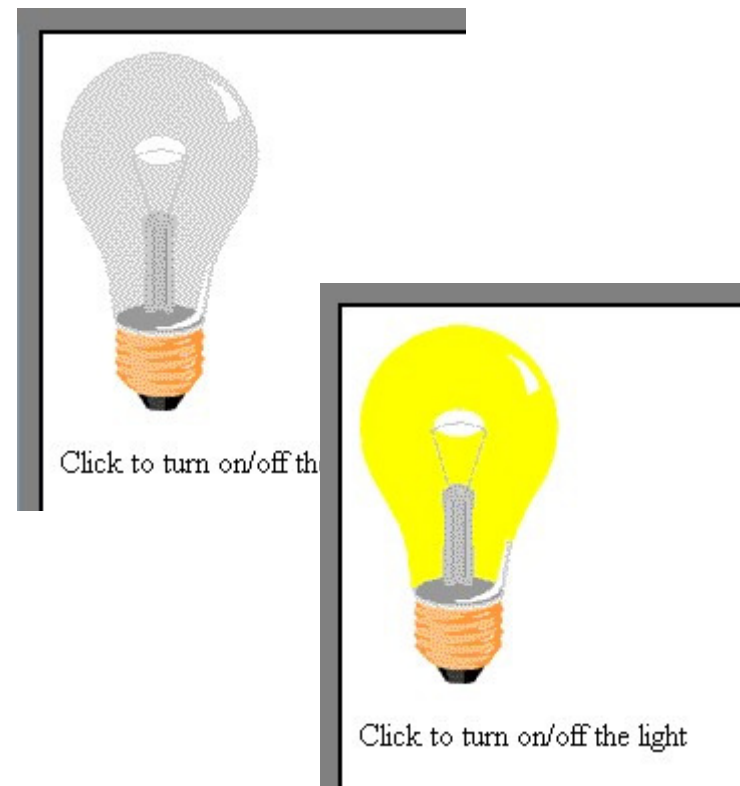
*Named Element*



# JavaScript

- **JavaScript is a scripting language for the web**
- **JavaScript is a script, not Java**
- **Small bits of JavaScript interact with HTML and CSS**
  - provides interactivity
  - allows dynamic changes
  - completely within the browser

```
function changeimage() {  
  if (cc==0){  
    cc=1;  
    document.getElementById('myimage').src="bulbon.gif";  
  } else {  
    cc=0;  
    document.getElementById('myimage').src="bulboff.gif";  
  }  
}
```



# AJAX

- **AJAX – Stands for *Asynchronous JavaScript and XML*.** RichUI makes extensive (almost wholesale) use of AJAX, utilizing it whenever it makes a service call. RichUI never executes a traditional HTML Form Submit.
- **JSON – *JavaScript Object Notation* is lightweight format used by JavaScript to exchange data.** JSON is able to serialize structured data, such as arrays, and exchange it among host and client machines.
- **DOJO – An open source JavaScript toolkit.** The DOJO project sets out to create widgets using only JavaScript. RichUI is able to interface with DOJO code in order to pull in some of their widgets.

– <http://dojotoolkit.org/>

Column 0	Column 1	Column 2	Column 3	Column 4
Apples	Oranges	Mangos	Eat lots of fruit	\$2.50
One	Two	Three	Four	Five
Apples	Oranges	Mangos	Eat lots of fruit	\$2.50
One	Two	Three	Four	Five
Apples	Oranges	Mangos	Eat lots of fruit	\$2.50

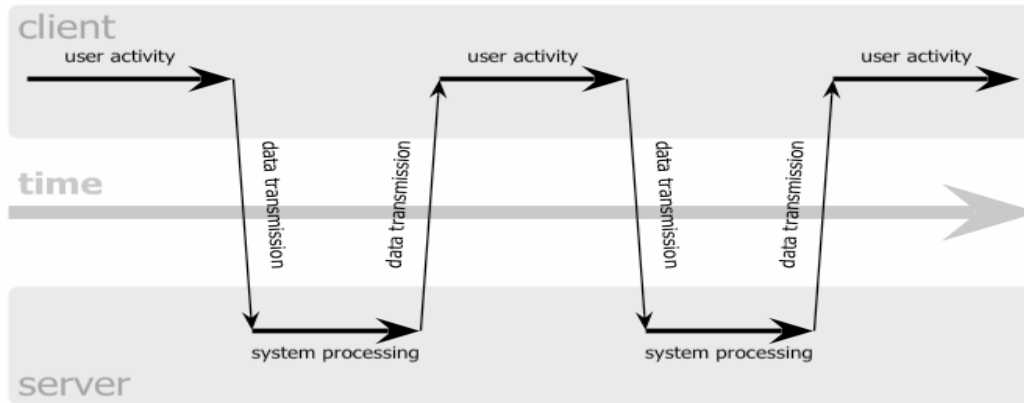
```

{
  "firstName": "John",
  "lastName": "Smith",
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": 10021
  },
  "phoneNumbers": [
    "212 555-1234",
    "646 555-4567"
  ]
}

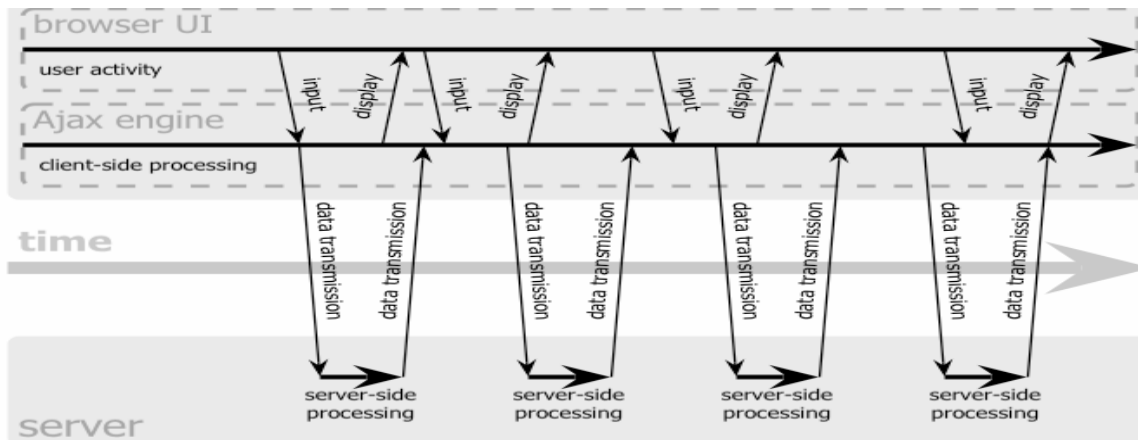
```

# Request Response with Ajax

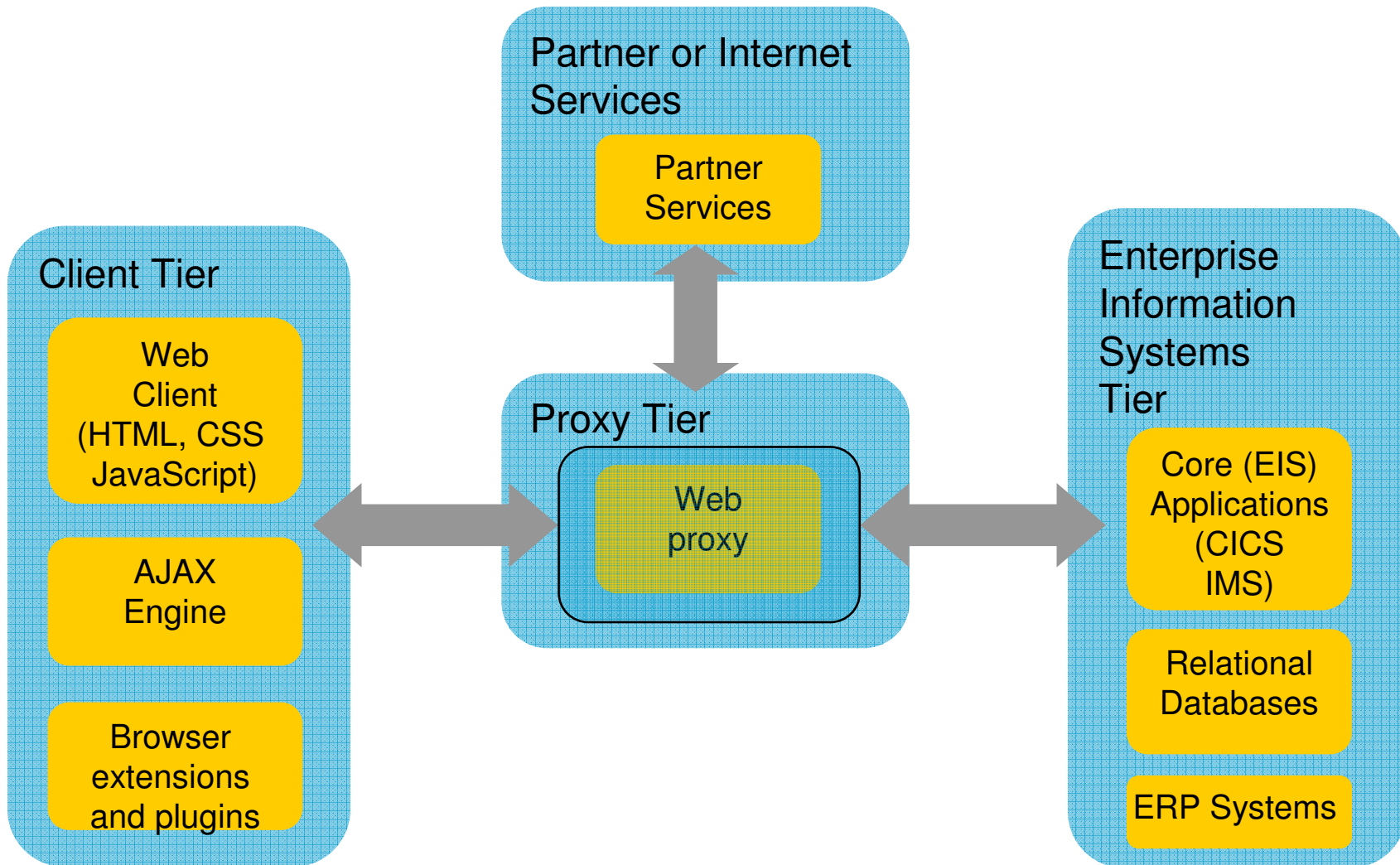
Classic web application model (synchronous)



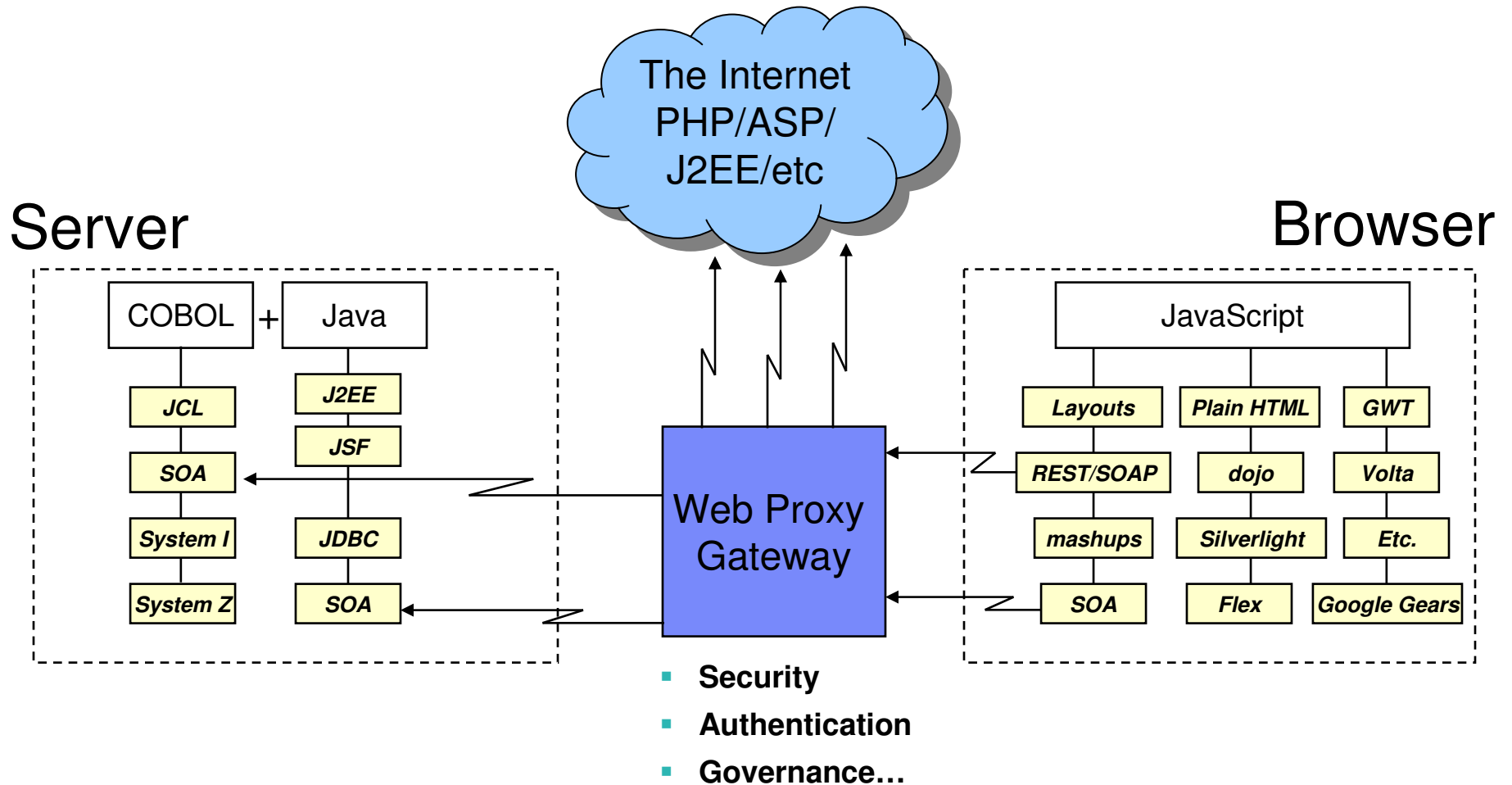
Ajax web application model (asynchronous)



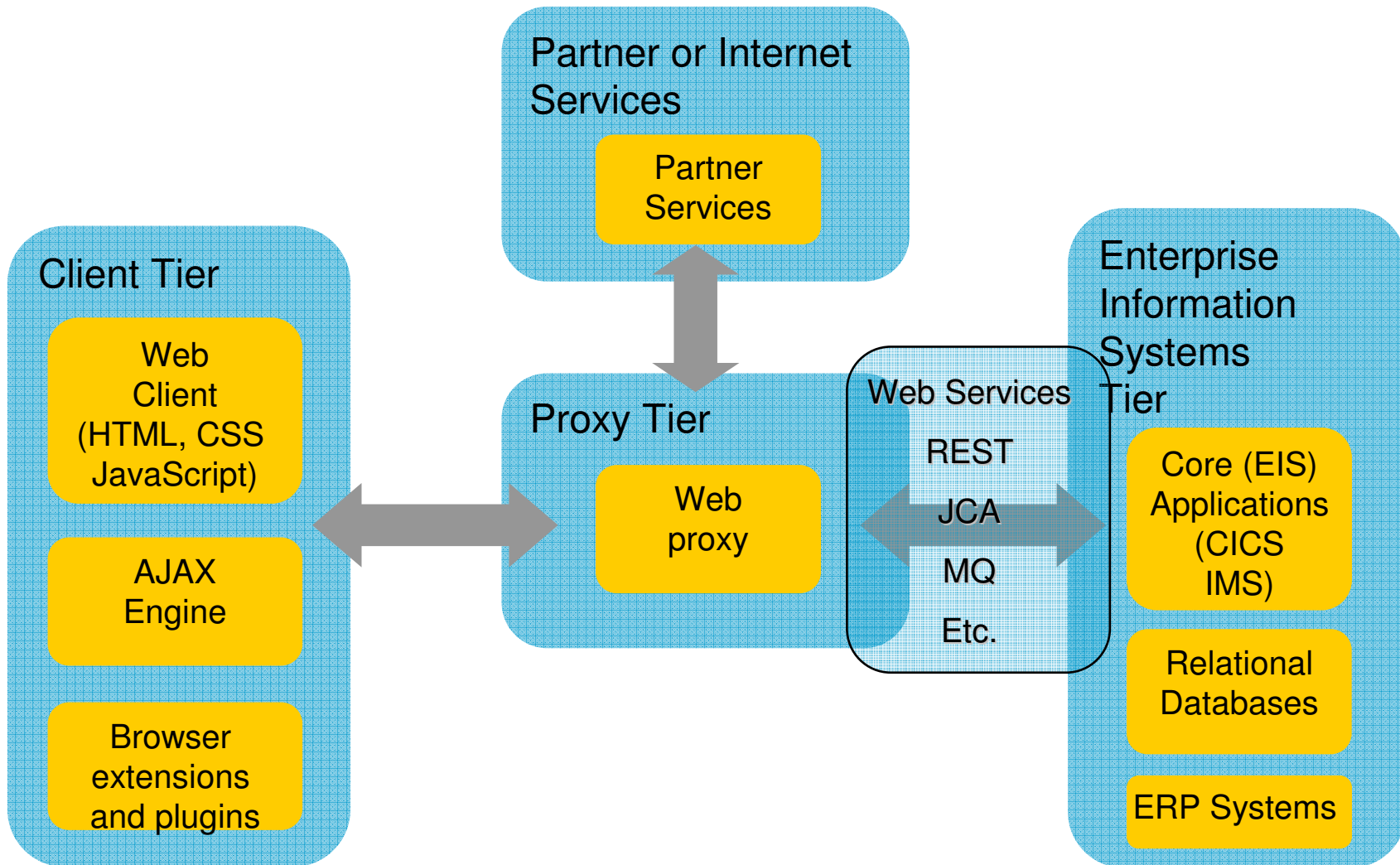
# Web 2.0 Architecture



# Web Proxy



# Web 2.0 Architecture



## SOA and Web Services

- **Rich U.I. makes extensive use of services, and SOA – Service Oriented Architecture, which is a way to modularize and deploy code so that it can be consumed anywhere in the world using any language.**
  
- **There are two types of Web Service calls used by Rich U.I.**
  - 1. SOAP service calls** - A type of service call that is more popular in enterprise. It requires the exchange of XML messages between the client and host system.
  - 2. RESTful service calls** – A call made through the HTTP service-interface. Once the call is made, a result is passed back to the requestor in XML or JSON format.
  
- **By utilizing web services, one can create a truly scalable system.**

# Web Services

## Architecture for

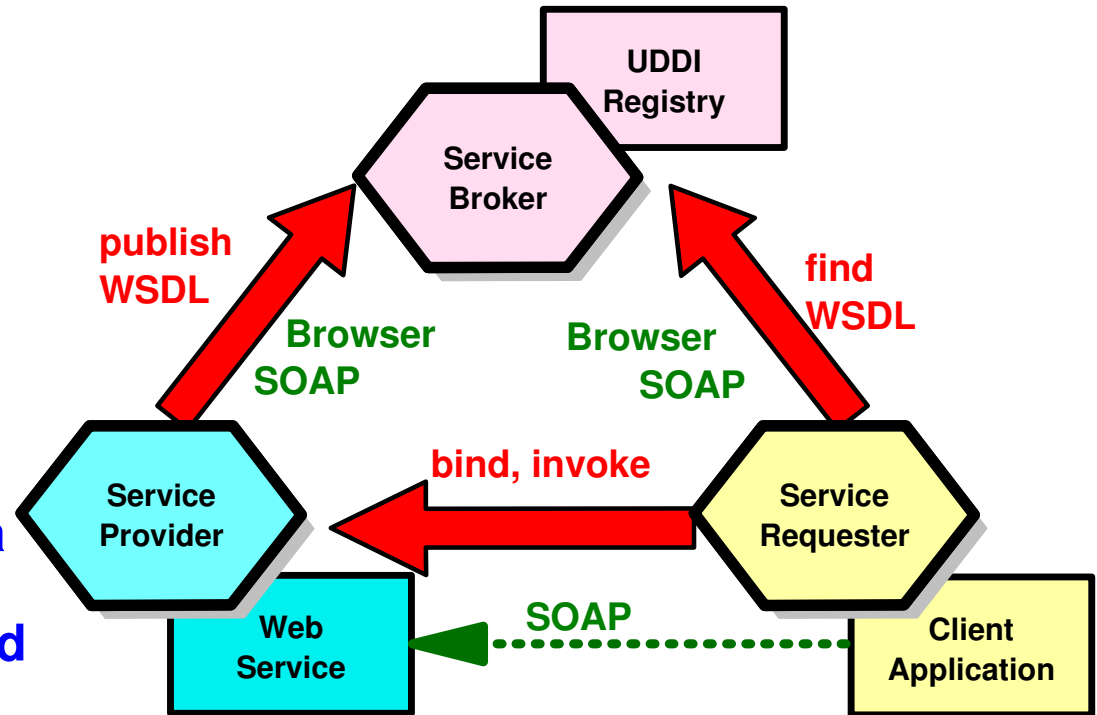
- Application to application
  - Communication
  - Interoperation

## Definition:

- Web Services are **software components described via WSDL** that are capable of being accessed via **standard** network protocols such as SOAP over HTTP

## WS-I.org (Web Services Interoperability Organization)

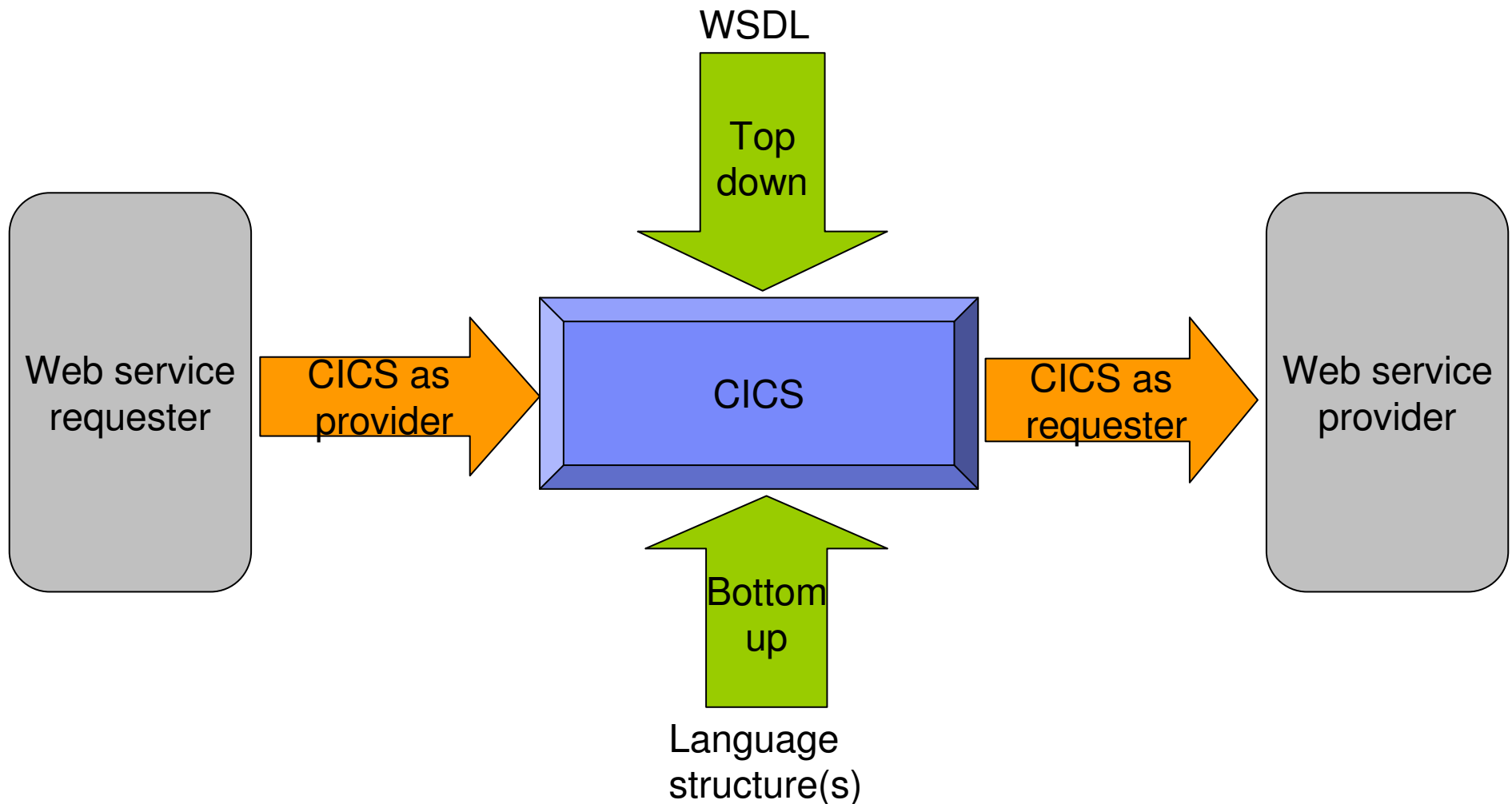
- Ensure interoperability



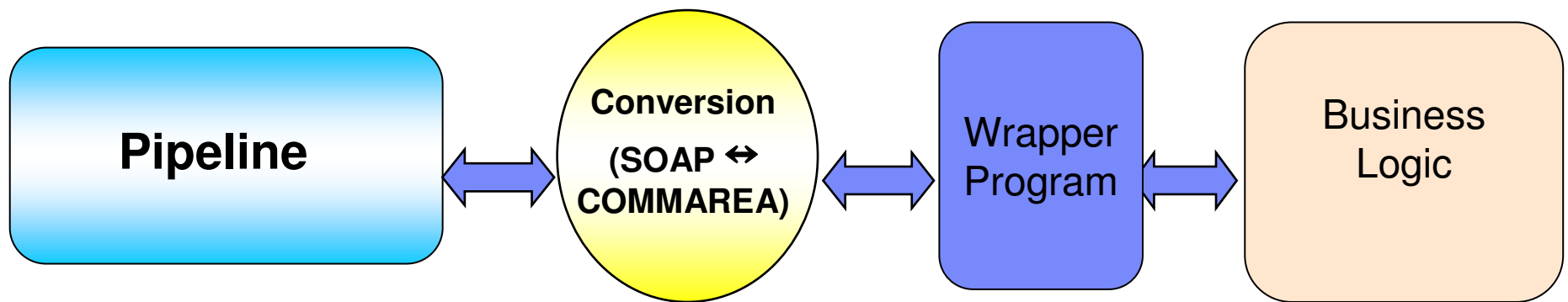
The entire industry is agreeing on one set of standards !!



# Web Services Enablement Styles



## Where a wrapper program fits



## XML Terminology

- **SOAP** and **WSDL** are based on XML
- **A tag / attribute based syntax**
- **Format of XML file described in**
  - **DTD** – Document Type Definition
  - **XSD** – XML Schema Definition
- **XML files are**
  - Well-formed (syntax is ok – matching tabs, etc.)
  - Valid (obeys rules in DTD or XSD) (CICS can validate)
- **Namespaces**
  - Avoids name collisions
  - A set of names (XML tags) that apply to a certain space in a document

# XML Basic Parts

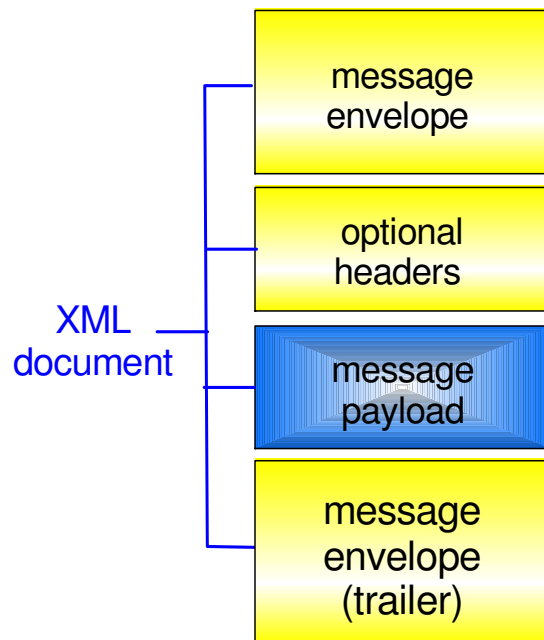
```

<?xml version="1.0" standalone="no" encoding="UTF-8" ?> ← XML Declaration
<!DOCTYPE shirt SYSTEM "http://shirts.com/xml/dtds/shirt.dtd"> ← Document
<shirt>                                                    ← root element
  <model>CICS Tee</model>                                  ← child of root
  <brand>Tommy Hilltop</brand>                             ← end tag
  ▲
  ─────────────────────────────────────────────────── start tag
  <price currency="USD">10.95</price>                     ← attribute
  <fabric content="70%">cotton</fabric>                  ← attribute
  <fabric content="30%">polyester</fabric>
  <on_sale/>                                              ← empty element
  <options>
    <colorOptions>
      <color>red</color>
      <color>white</color>
    </colorOptions>
    <sizeOptions>
      <!-- Medium and large are out of stock -->        ← comment
      <size>small</size>
      <size>x-large</size>
    </sizeOptions>
  </options>
  <order_info>Call &phone;</order_info>                  ← entity reference
</shirt>

```

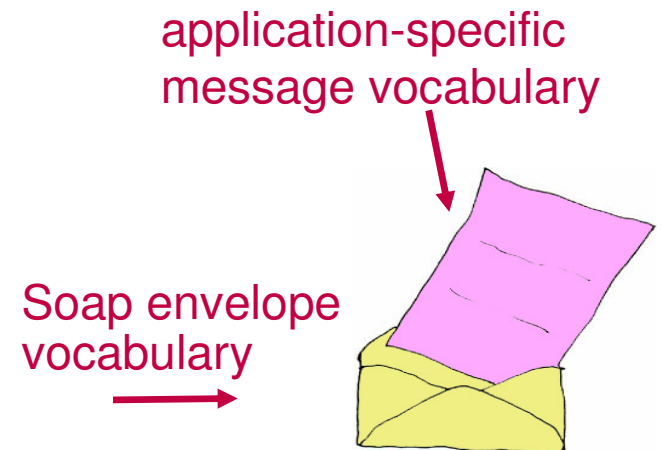
# Simple Object Access Protocol (SOAP)

- An XML-based protocol for exchanging of information in a decentralized, distributed environment
- An open standard whose main goal is to facilitate interoperability
- A protocol which is not tied to any operating system, transport protocol, programming language, or component technology



- **XML Message Envelope**
  - service requested
  - routing information
  - message type
  - date/time stamp
- **XML Message Headers**
  - authentication
  - transaction context
- **XML Message Payload**
  - data understood by target application
- **XML Message Trailer**
  - closing tags
  - optional message digest

*SOAP spec defines how to do this!*



## SOAP: Request Message

```
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV=
    "http://www.w3.org/2001/06/soap-envelope"
  SOAP-ENV:encodingStyle=
    "http://www.w3.org/2001/06/soap-encoding">

  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI">
      <symbol>IBM</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

app-specific  
message

SOAP envelope

# SOAP: Response Message

```
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV=
    "http://www.w3.org/2001/06/soap-envelope"
  SOAP-ENV:encodingStyle=
    "http://www.w3.org/2001/06/soap-encoding">

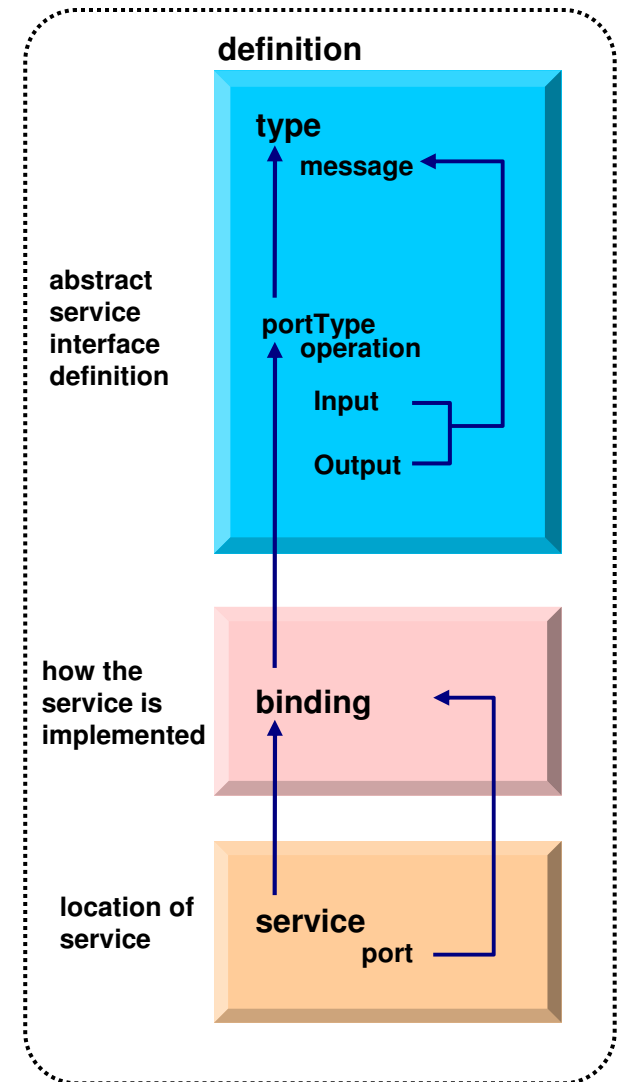
  <SOAP-ENV:Body>
    <m:GetLastTradePriceResponse
      xmlns:m="Some-URI">
      <Price>134</Price>
    </m:GetLastTradePriceResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Result  
returned  
in Body

app-specific  
message

# WSDL - Web Service Description Language

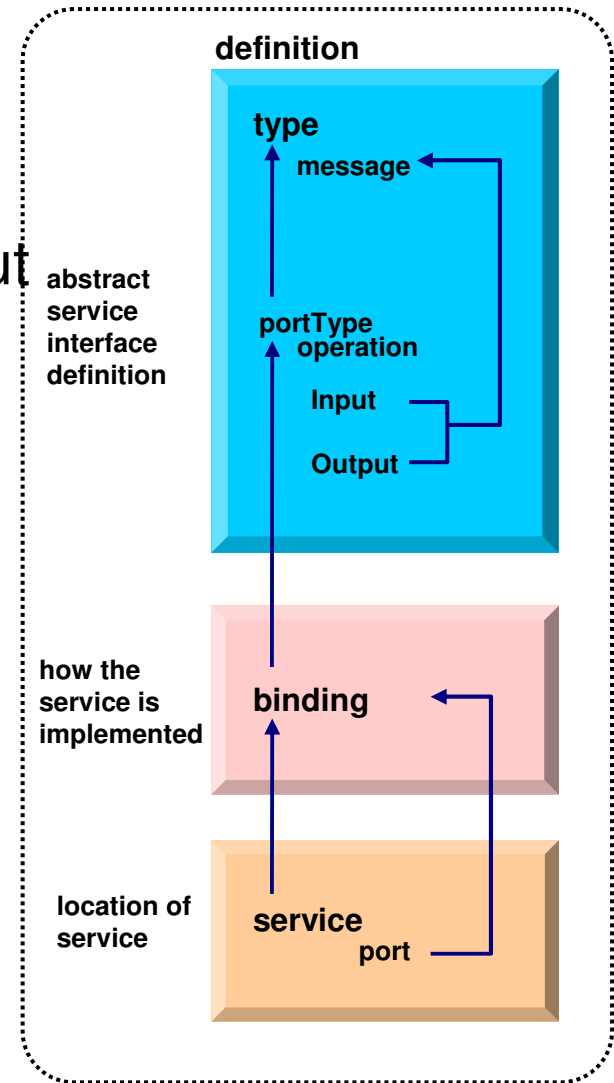
- Open Standard
- XML resume describing what a Web Service can do, where it resides, and how to invoke it
- Machine readable, generated, used by IDEs
- Similar in purpose to IDL, but in XML form
- Can be One or multiple documents
- Major sections are:
  - Service Interface (operations, input, output)
  - Service binding (protocol binding)
  - Service implementation (location of service)





# WSDL: Logical Contents

- **Service Interface**
  - Operation (business functions)
    - Input Message ( 0 or 1 ) and Output Message ( 0 or 1 )
      - 1 or more parts
      - Parts may be simple or complex
      - Complex parts may have multiple elements
- **Service binding**
  - Definition of the physical service interface implementation
- **Service Implementation**
  - Location of the service



## RESTful services

- **Representational State Transfer**
  - Nouns (URLs) indicate what is being worked on
  - Verbs (GET, PUT, POST, DELETE) indicate the action to perform
- **A lighter-weight, simpler mechanism than Web Services**
  - SSL vs WS-Security
  - HTTP vs. WS-Authentication
  - Unformatted data vs. XML Schema
- **Standards are well defined by IETF**
- **\*\*\* Technology stack is in place in the browser**

## REST Example

### ■ Request...

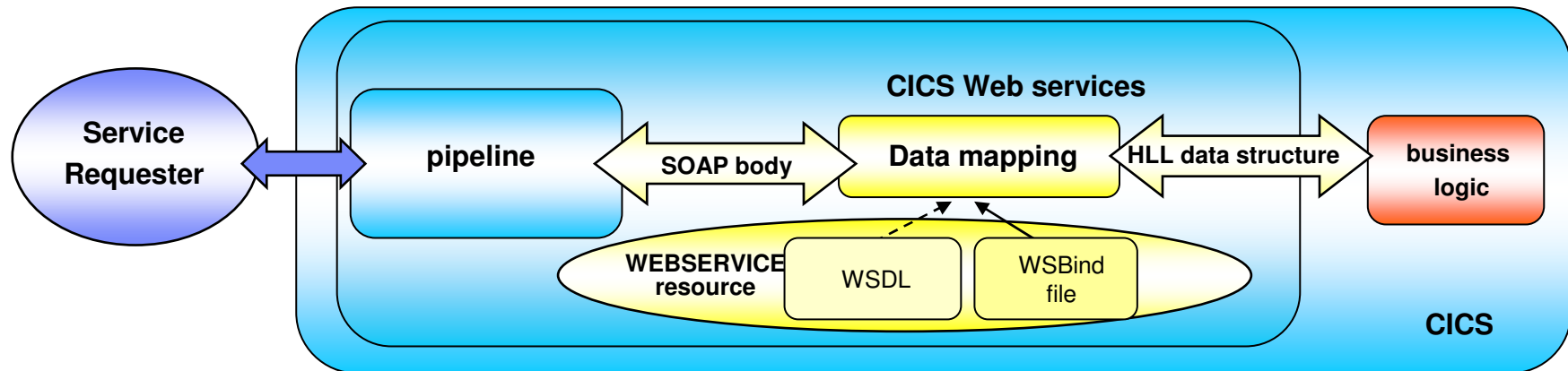
```
GET /mortgage/231677 HTTP/1.1  
Host: www.example.com  
Accept-Language: en  
Charset: UTF-8
```

### ■ Response...

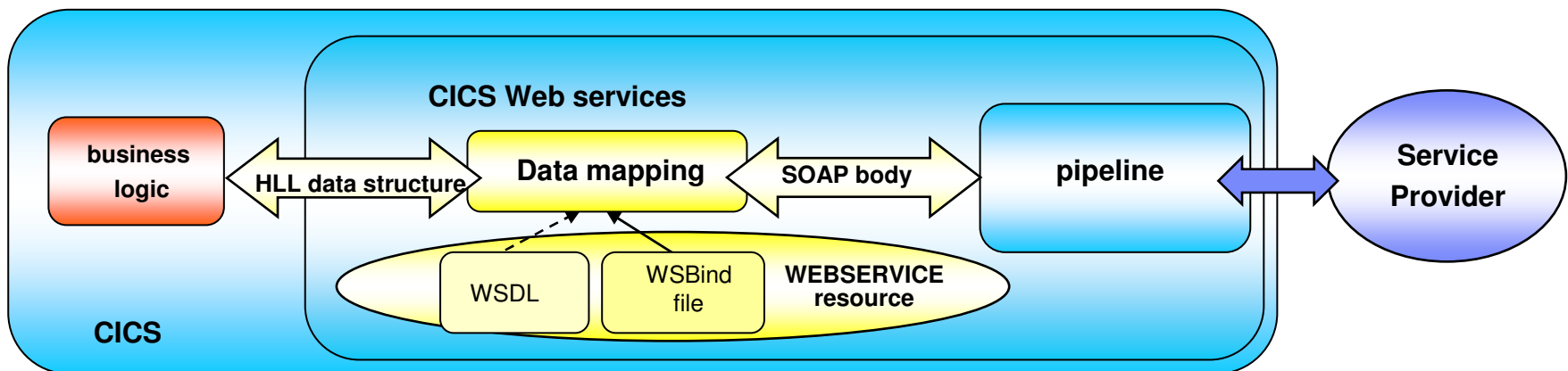
```
HTTP/1.1 200 OK  
Language: en_us  
Charset: UTF-8  
<mortgage>  
  <principle>238000</principle>  
  <rate>3.5</rate>  
  <type>5/1 ARM</type>  
</mortgage>
```

# CICS Web Services Implementation

- CICS as a service provider



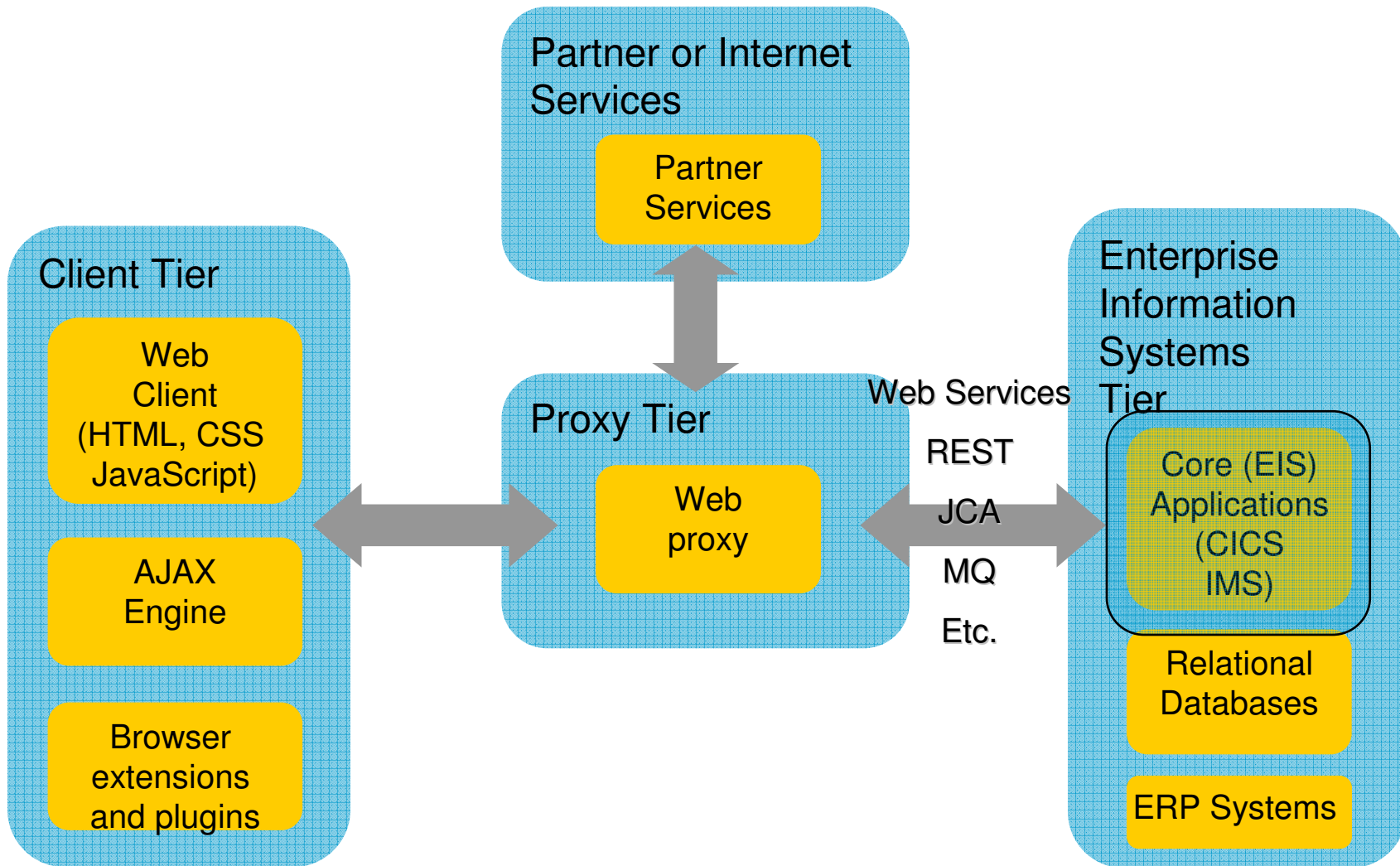
- CICS as a service requester



## CICS Web Services Resources

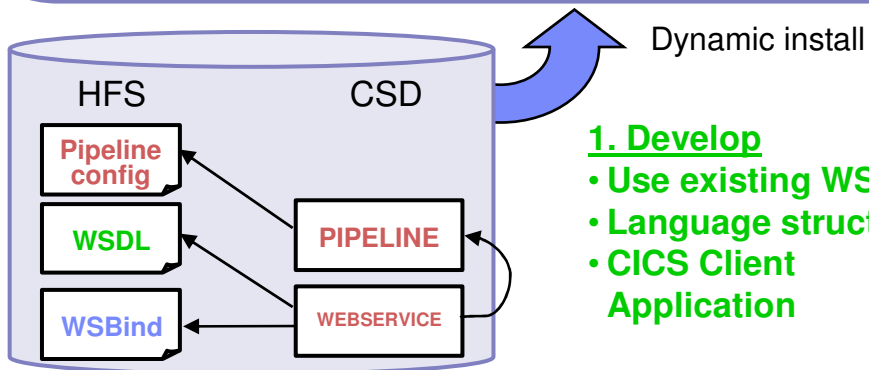
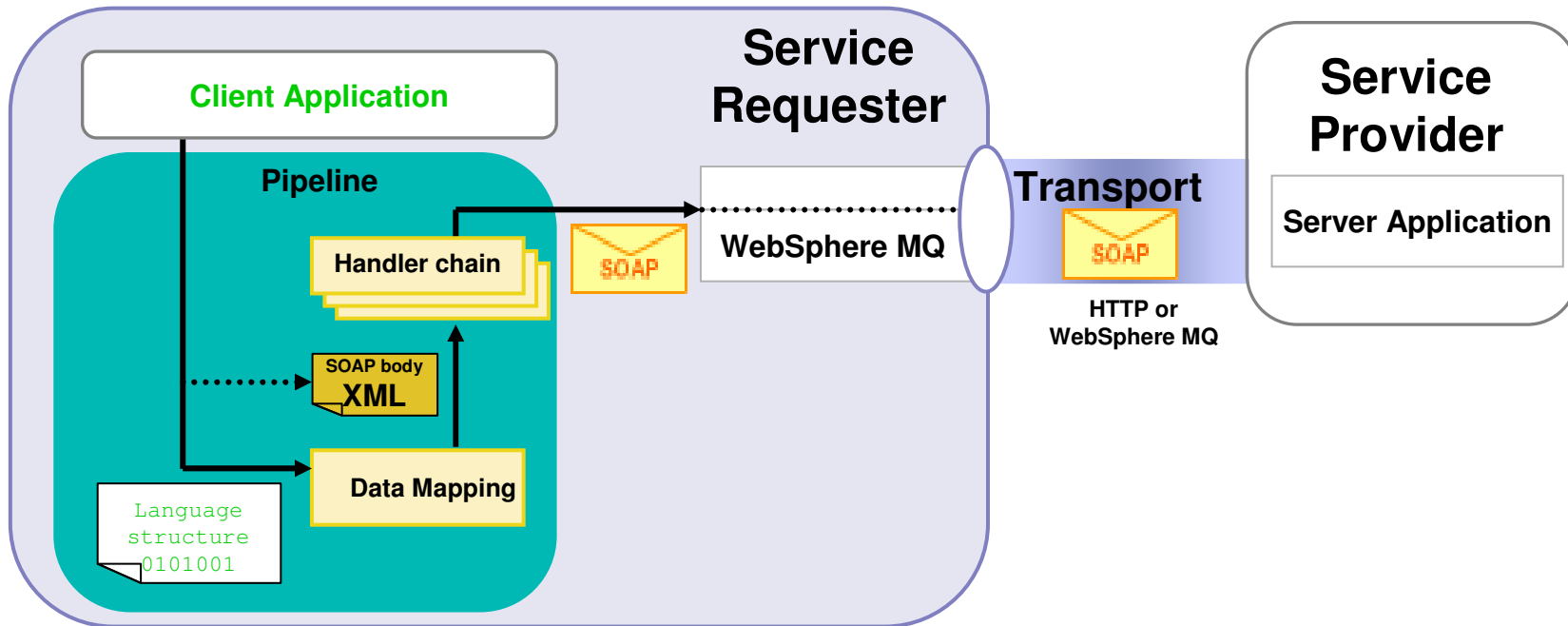
- **Define a TCPIP SERVICE (or WMQ) and a PIPELINE**
- **Then install the PIPELINE definition and issue CEMT PERFORM PIPELINE SCAN**
- **CICS uses the PIPELINE definition to**
  - Locate the WSBIND file
  - From the WSBIND file, CICS will dynamically create a WEBSERVICE resource
  - CICS will also dynamically create a URIMAP definition
- **Can define everything individually if preferred**

# Web 2.0 Architecture



# CICS as a Web Service Requester

CICS TS V3.1



- 1. Develop**
- Use existing WSDL
  - Language structure
  - CICS Client Application

- 2. Generate**
- Language structure
  - WSBIND

- 3. Configure**
- Pipeline
  - ✓ Pipeline configuration
  - WEBSERVICE

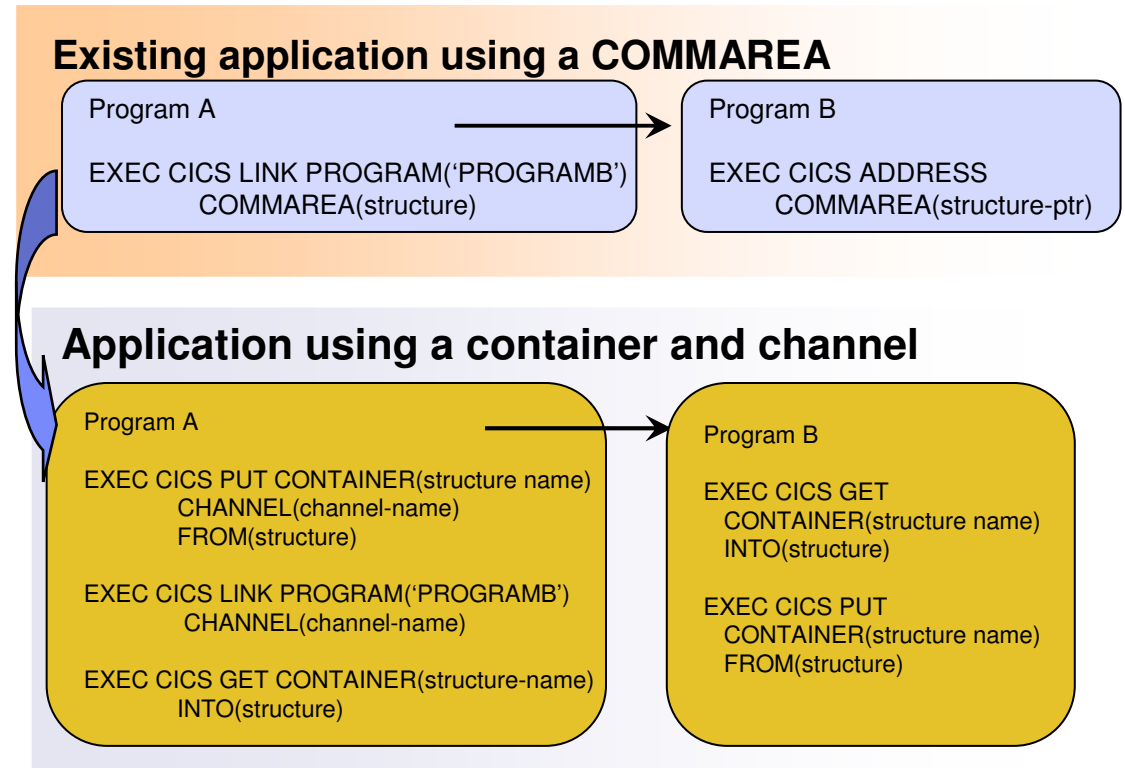
## CICS Programs invoking Web Services

- **Invoking a Web Service from a CICS application program**
  - CICS as a service requester
    - EXEC CICS INVOKE WEBSERVICE ( )  
CHANNEL ( ) URI ( ) OPERATION ( )
      - ▶ WEBSERVICE: name of the Web Service to be invoked
      - ▶ CHANNEL: name of the channel containing data to be passed to the Web Service (DFHWS-DATA container)
      - ▶ URI: Universal Resource Identifier of the Web Service (optional)
      - ▶ OPERATION: name of the operation to be invoked

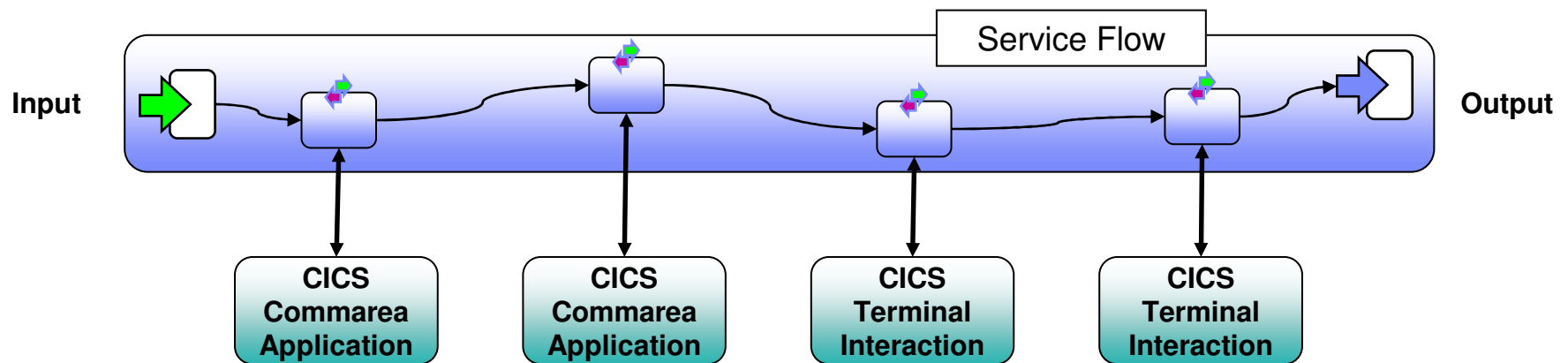


# Data Exchange between CICS programs with Containers and Channels

- Offers a more flexible and intuitive alternative to the COMMAREA
- Enables large amounts of data to be passed between CICS applications
  - Not subject to 32KB restriction
- Optimized and managed by CICS
- Requires minimal application changes required to use



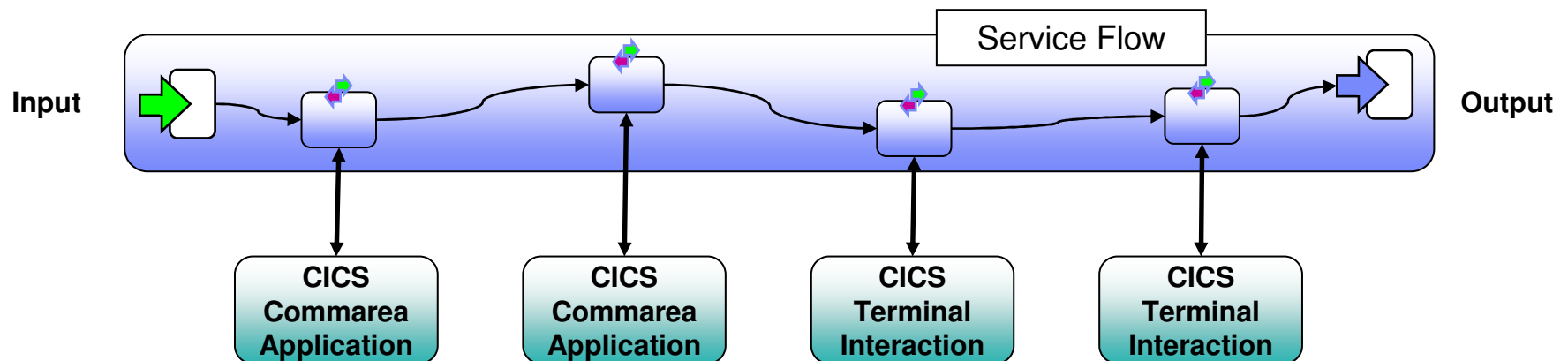
# What is a Service Flow?



- A service flow is a non-interruptible micro-flow that is constructed from a collection of nodes that represent the invocation of CICS resources
- The flow describes the navigation of the nodes and allows data mapping between the nodes
- A single request may cause the execution of many CICS resources
- Allows for the development of coarse grained services from fine grained resources

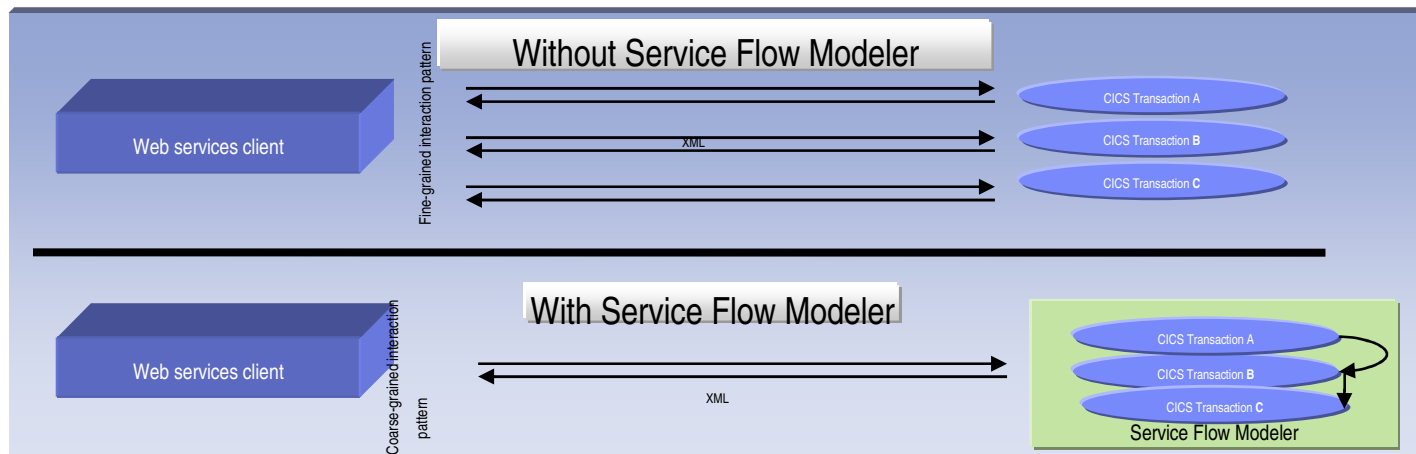
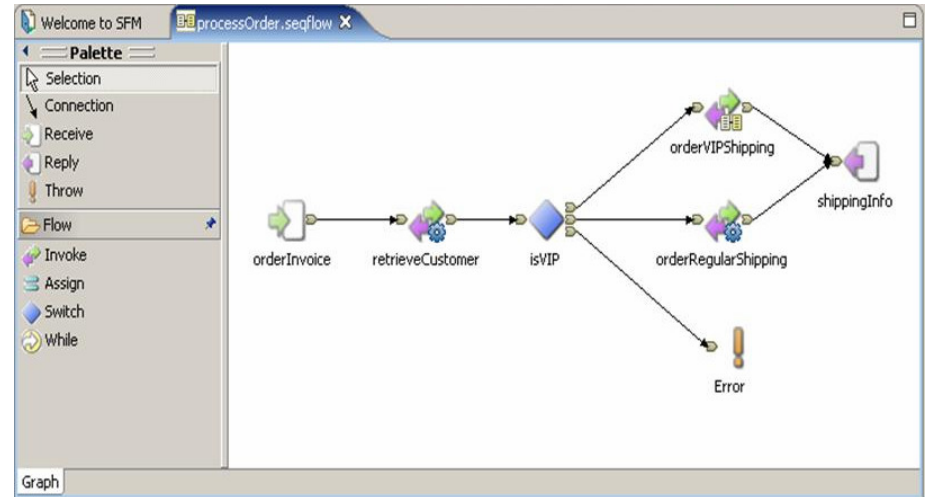
# What is the Service Flow Runtime?

- The CICS Service Flow Runtime is a no-charge, fully supported, orderable feature for CICS TS v3.X
- A business service integration adapter for all CICS applications
- Create optimized CICS services orchestrating CICS terminal, non-terminal, and web services into a composite service
  - In a fast moving market, companies need to embrace new technologies whilst leveraging their existing assets.
  - Compose new services by aggregating these existing proven assets into new services.



# What is Service Flow Modeler?

- **Service Flow Modeler is the tooling required to develop service flows**
- **SFM only ships with Rational Developer for System z**
- **Model, Deploy, and Test Service Flows**
  - Aggregates multiple CICS transactions into high-level business processes through visual modeling
  - Supports CICS BMS (terminal-based) applications & CICS commarea/container/channel applications
  - Deploys to the optimized CICS Service Flow Runtime



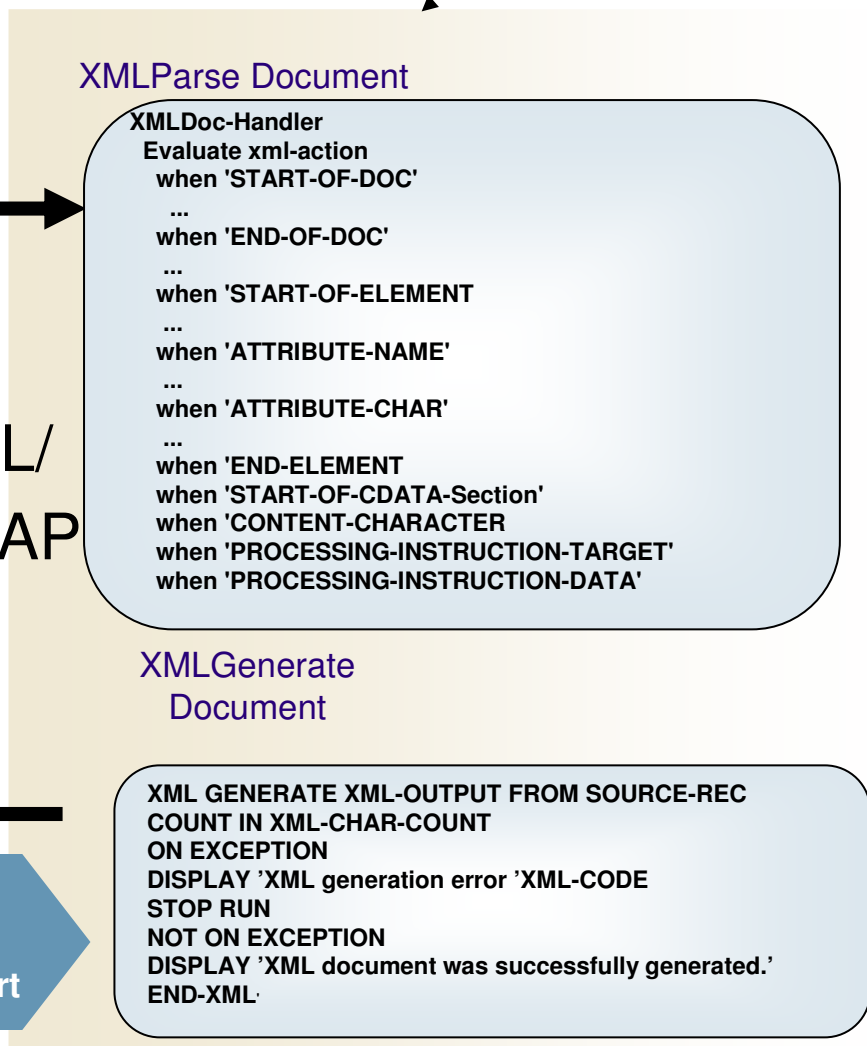
# Enterprise COBOL

- XML Language based generation from COBOL data structure
  - XMLGenerate Verb
  - WebSphere EJB support
  - DB2 V8
- High speed XML Sax based parsing
- Object Oriented Support for Java COBOL Interoperability
- Unicode support
- CICS and DB2 integrated preprocessor
- Raise 16Mb COBOL data size limit
  - Picture clause replication:  
01 A PIC X(134217727).
  - OCCURS:::  
05 V PIC X OCCURS 134217727 TIMES.

XML/  
SOAP



## CICS/IMS/Batch/DB2 COBOL




*COBOL is an excellent business language*

## Why COBOL and EGL deployed to COBOL?

- **Large portfolios**
- **Many developers**
- **High performance**
- **Self documenting**
- **Proven Maintainability**
- **Business oriented, eases technology burden**

## Summary

- **Web 2.0 applications provide highest levels of visual service**
- **MVC application model provides high levels of flexibility**
- **EGL provides support for all tiers of the Web 2.0 and MVC models**
- **CICS provides leading edge support of Web Services**
  - Allows for re-use of existing business assets and new development of high QOS assets
- **Developers need “complete” application skills**

The background of the slide is a vibrant blue gradient with intricate, flowing white and light blue lines that create a sense of movement and depth. The lines are most prominent on the left side, curving and overlapping to form a complex, organic pattern that fades into a smoother gradient towards the right.

# Modern Application Development Featuring Web 2.0 for System z

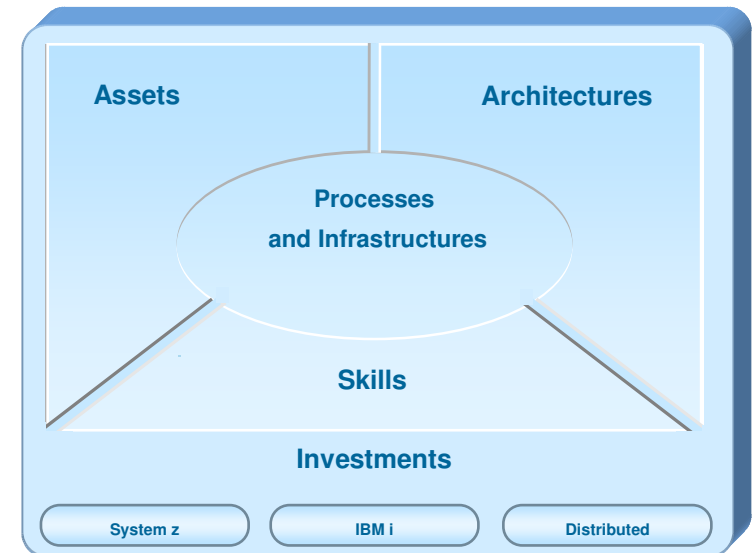
Rational Business Developer and EGL

**Rational.** software

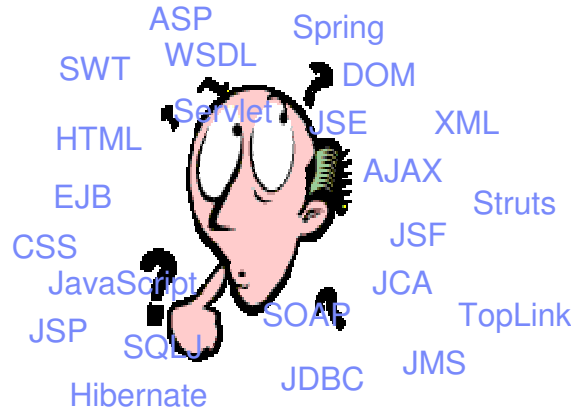
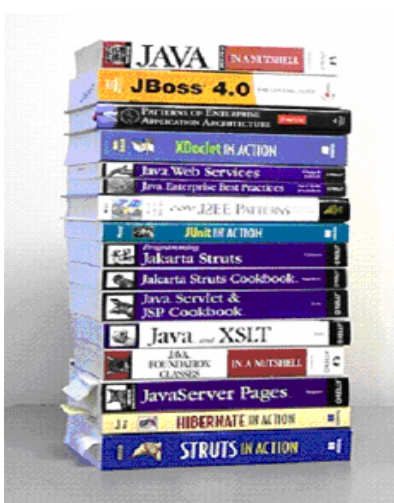


# EGL for Enterprise Modernization

- Enables COBOL, RPG, 4GL developers to create Web applications and SOA solutions with minimum learning curve
- Provides a modern programming paradigm for “legacy” platforms, attract new staff to your trusted box
- Enables to easily connect to, wrapper and extend trusted valuable assets
- Affords maximum flexibility of deployment options and architectures
- Delivers a modern language that adapts more easily to changing technologies
- Can be used as a target for legacy programs conversion
- Enables migration from Natural or RPG



# Why EGL? Because building applications today is not easy



- Learn and master myriads of
  - programming languages and semantics
  - middleware interfaces
  - programming paradigms and styles
  - frameworks and libraries
- Constantly emerging new technologies

- Code at a more abstract and simpler level
- Easy to learn, modern and comprehensive language
- Keeps up with emerging technologies
- Inter-operates seamlessly with legacy

# EGL - The power of the Language

*Simple and familiar*



## Hello World

Basic EGL Program

```
1 // Hello World basic program
2
3 program hello type BasicProgram
4
5 // Data Declarations
6 name string = "World";
7
8 function main()
9     writeStdOut("Hello " + name);
10 end
11
12 end
13
```

Declare program type and name

Declare a variable and assign a value

EGL Function

EGL Built-in Function

Comments

End of Program

Literal

Variable

# EGL the Language

*Powerful and complete*

- **Rich data types**
  - Simple (int, string, boolean, etc.) or Complex (any, static arrays, dynamic arrays, dictionaries, array dictionaries, etc.)
- **Keywords**
  - Case, if-then-else, while, for loop, for loop cycling through a database result set, etc.
- **High power language capabilities**
  - Automated Casting (e.g. using AS operator)
  - Mixing data types in assignments and expressions
  - Exception handling
- **Rich libraries of built-in functions**
  - Math, string, date/time, system, etc.
- **Robust integrate with existing functions or low level APIs**
  - Call RPG, COBOL, C, etc.
  - Full Java interoperability
    - Invoke Java from EGL (map Java classes with EGL External Types)
    - Invoke EGL from Java



# EGL - The Power of Declarative Programming

*Annotations apply semantics in multiple contexts*

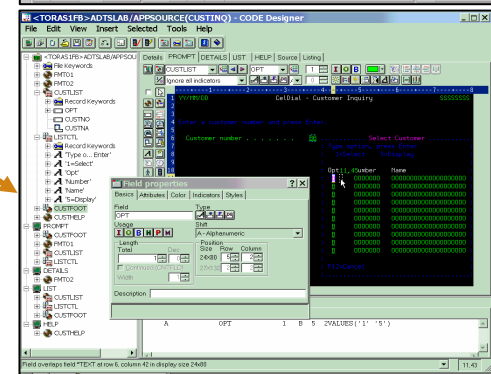
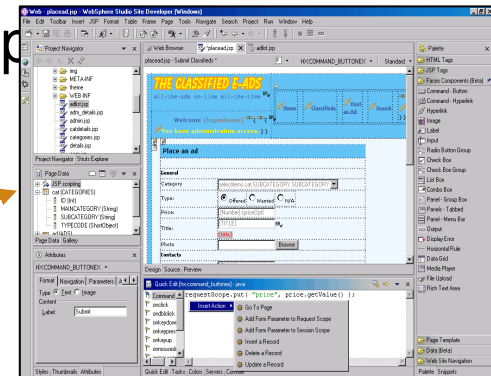
## Validation and Editing Rules

- Set properties in “Data Items”
- Define formatting & validation rules in a common place
- Reuse data items for Records, screens, web pages

```

DataItem SSN Password char(9) {
    validatorFunction = "ValidateSSN()",
    displayUse = secret,
    pattern = "XXX-XX-XXXX",
    displayName = "Social Security No",
    inputRequired = yes
}
end
  
```

Validation is consistently applied whether the data is bound to a field in a JSF-based web page, field on a 3270 screen, or Rich UI page.





# EGL - The Power of Abstractions

## ▪ Data access

- Access SQL, Indexed, Relative, Serial, DL/I, and Service data through “Records”
- Use common Verbs for data access (**Get, Add, Replace, Delete**)
- Use common Error Handling

```
*sampleProgram.egl x
function allLoans()
    loans LoanRec[];
    get loans;
end
```

## ▪ Remote Invocation

- Call COBOL, RPG, C, Java
- Keep linkage info separated from code
- Resolve data mapping and protocol invocation at runtime... NO code necessary!

```
*sampleProgram.egl x
function callHelloWorldOniSeries()
    salutation char(30);
    call helloworld salutation;
end
```

# The power of Services - Built into the language



## ■ Service part:

- a generatable part containing code that will be accessed:
  - from EGL code by way of a local or TCP/IP connection (*EGL Service*).
  - from any code by way of an HTTP connection (*EGL Web service*).

```
customerService.egl x
// service
Service CustomerService
  Function getCustomer(custid String) returns (string)
//
  ...
end
//
...
end
```

## ■ Interface part:

- Used to access external services provide separation of concern.

```
customerService.egl creditCheck.egl x
// interface
Interface creditCheck
  function checkCredit(SSAN string in) returns (string);
// ...
end
```

# EGL - The Power of Services

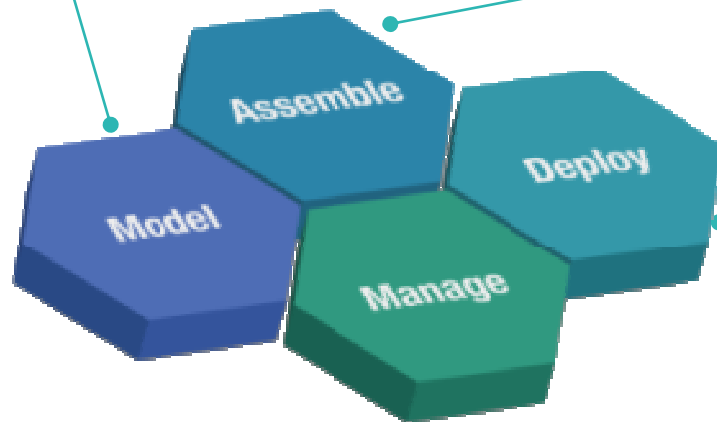
*Cross platform language for business oriented services development.*

## At development time...

- Focus on the business logic
- Implement SOA design elements: services and interfaces
- Leverage existing business developers for new SOA development
- Ignore deployment targets/technology while coding/testing

## Leverage external web services...

- EGL Interfaces
  - represent external web services
  - Are created via import from WSDL
  - Allow the EGL developer to stay within the context of the EGL programming model



## Deploy EGL services...

*To any platform*

- Java to WAS/Tomcat/etc.
- COBOL to CICS, iSeries (1Q 2007)
- COBOL to IMS (2H 2007)

*As...*

- A Web service (uses SOAP)
- A private service (uses CICS ECI or TCP)
- Other SOA runtimes when they reach critical mass

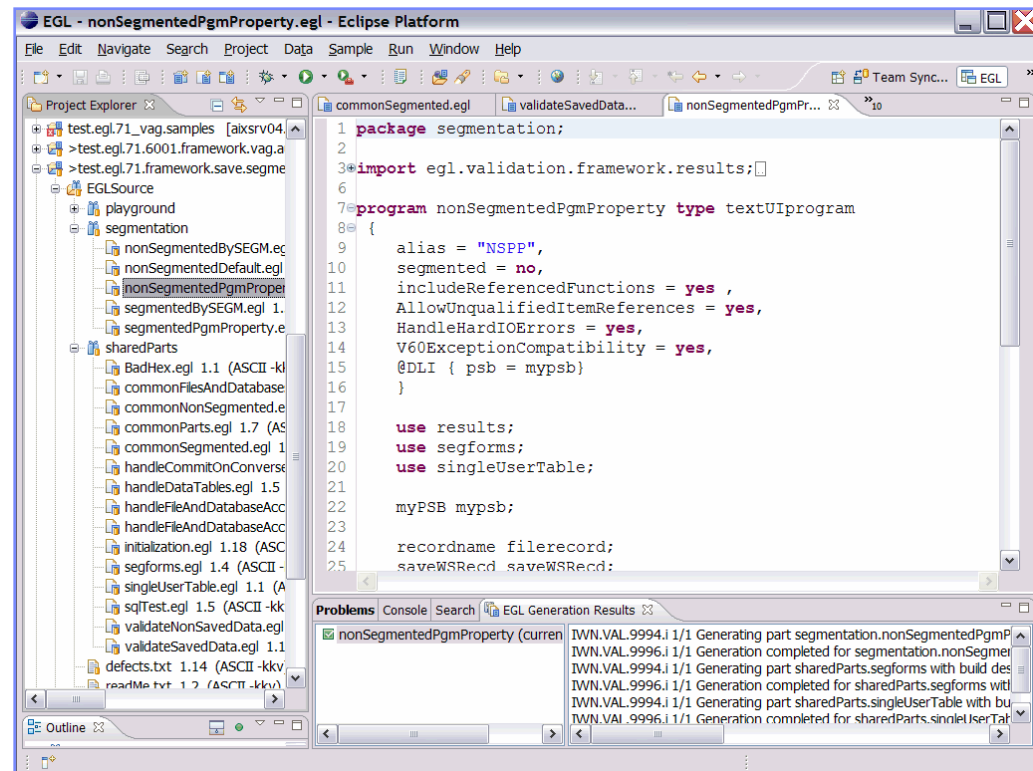




# EGL - The Power of Tools

*First class Eclipse workbench*

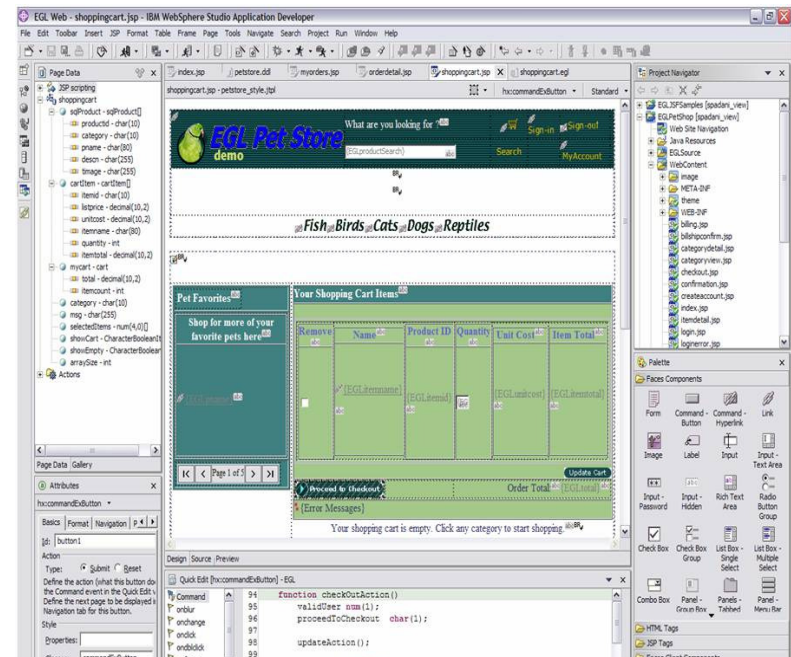
- Folders and views
- Smart EGL editor
- Code templates and snippets
- Code completion
- Import data items from tables
- SQL visualization and editing
- SQL validation
- References and declarations
- Open on selection
- Refactoring
- Cheat sheets and Dynamic help



# The power of tools: Robust Page Design



- **First Class integration with Page Designer and JSF tools**
  - Drop EGL data structures on JSP:
    - Validation, editing, formatting rules from EGL Data Items applied.
    - Appropriate UI controls rendered pre-bound to data declared in EGL Page.
  - Server-side event handlers in EGL within context of page designer.
- **Integration is totally seamless.**
- **No Java coding required to wire EGL data to JSF.**
- **EGL logic can be used to handle user interaction with the JSP.**
- **AJAX capability built in...partial refresh, etc...**
- **Portlet support**

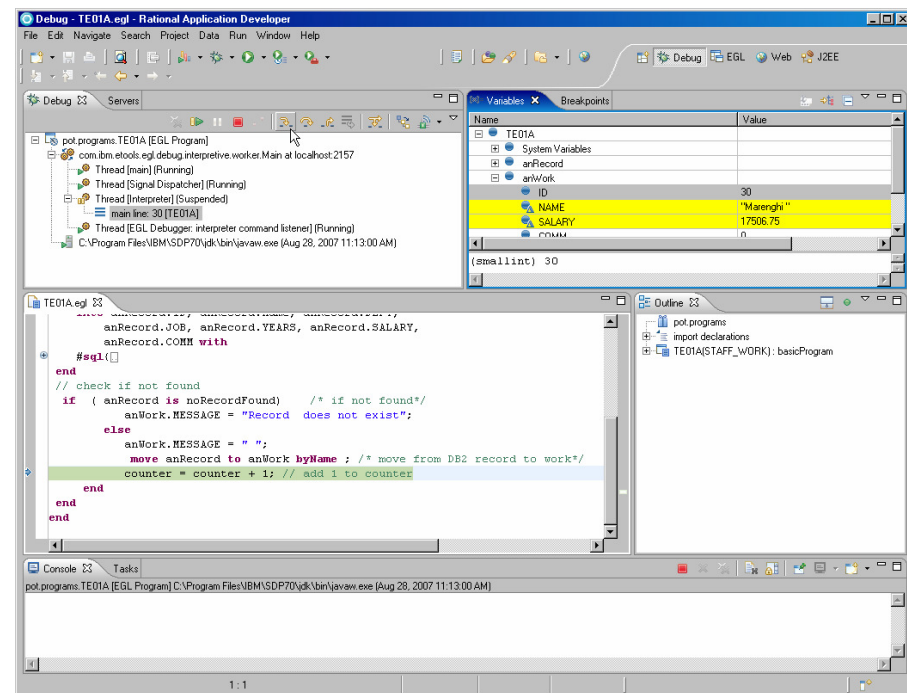




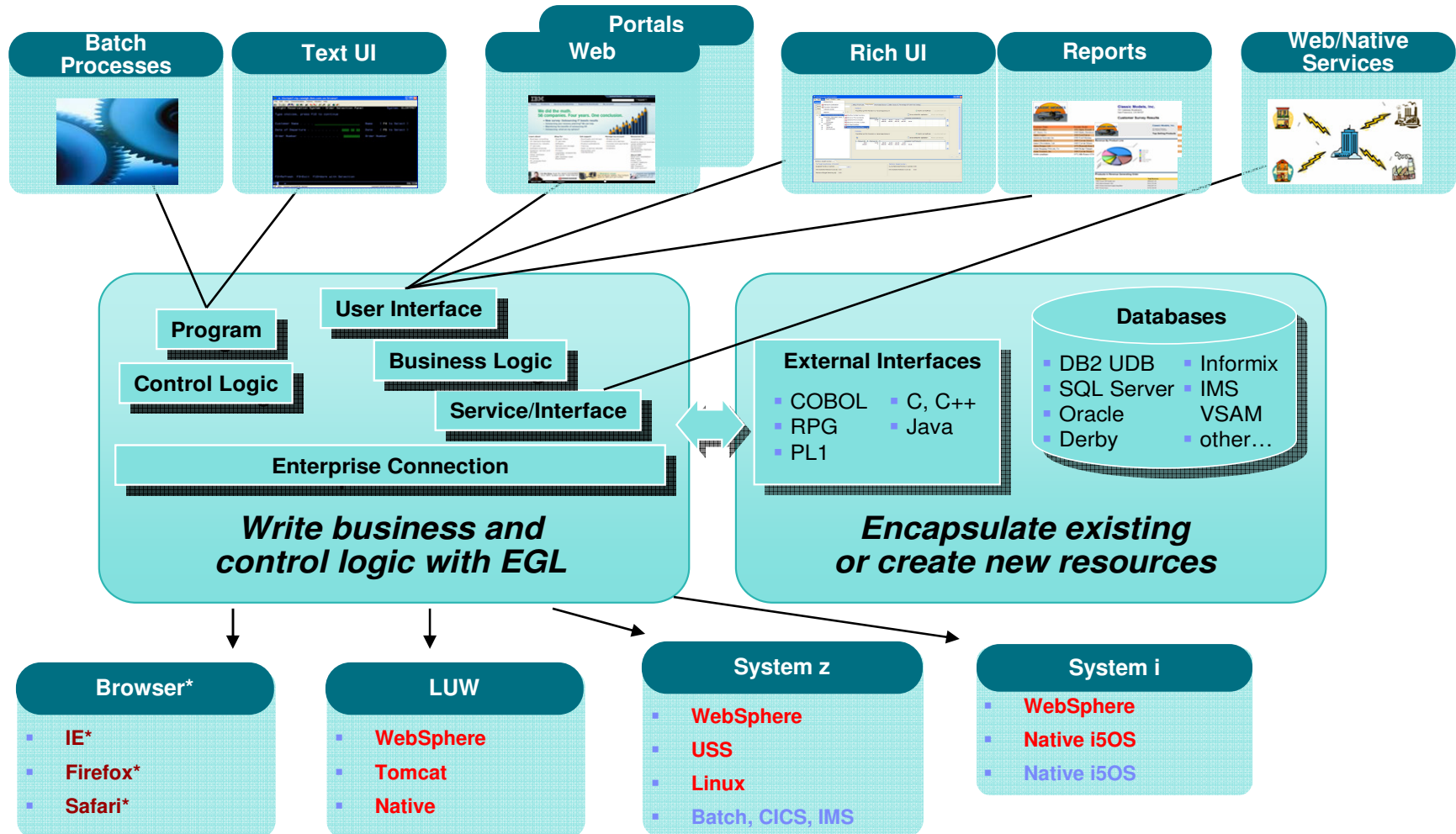
# EGL - The Power of Tools

*Integrated debug environment*

- **Debug entire application regardless of ultimate deployment targets**
  - Debug EGL, JSP, Java, etc.
- **Use features of the EGL source debugger**
  - Set breakpoints
  - Watch variables
  - Change variable values
  - Dynamic re-positioning
- **Additional EGL features**
  - Remote VSAM access

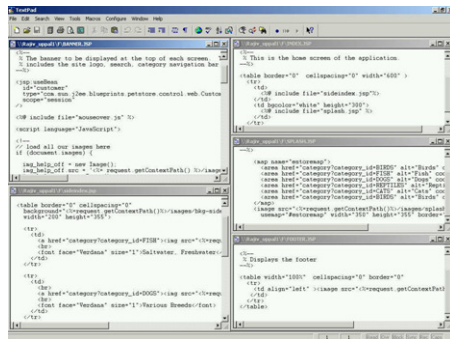


# Respond to Broadest Application Needs



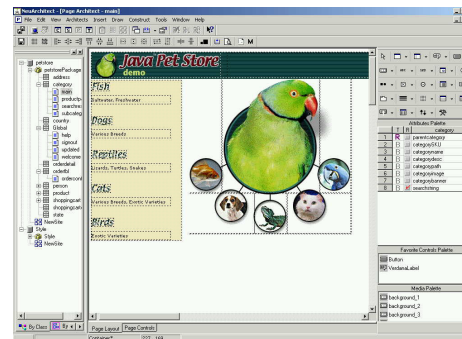
# Accelerating Application Delivery

## Hand-Coded



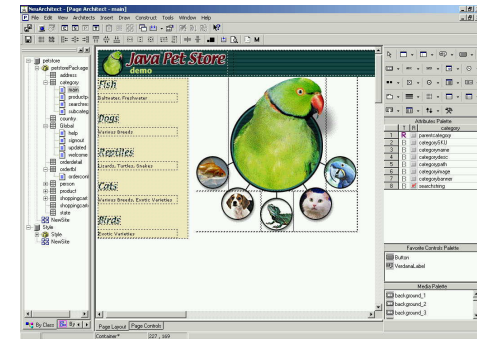
507 Hours

## Java/J2EE IDE



330 Hours

## RBD



60 Hours

- RBD is **dramatically faster** than traditional development\*
- RBD is more productive than MS VS 2005\*\*

\* Internal benchmark using Sun PetStore application

\*\* Branham Study April 2008

# EGL Rich UI

*Web 2.0 made easy*

## Why RIA?

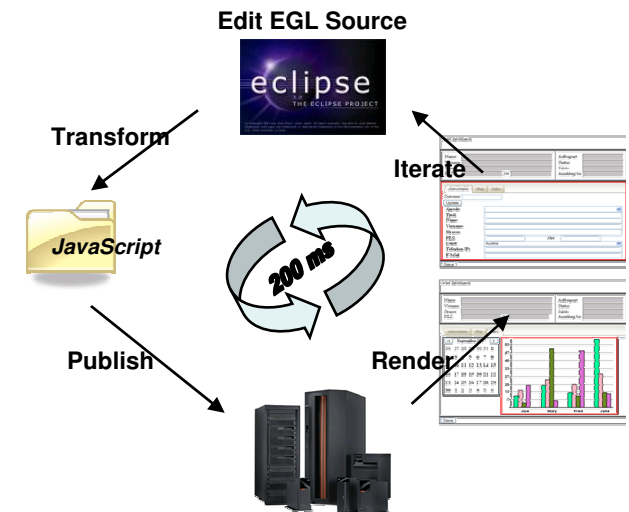
- Applications are responsive and user friendly (user productivity)
- Unleash innovation leveraging services “mush-up

## Why is it hard to create RIA?

- Currently domain of “tech heads”
- Need to know many low level intricate technologies
- Compound the skill/tool silos and fragmentation

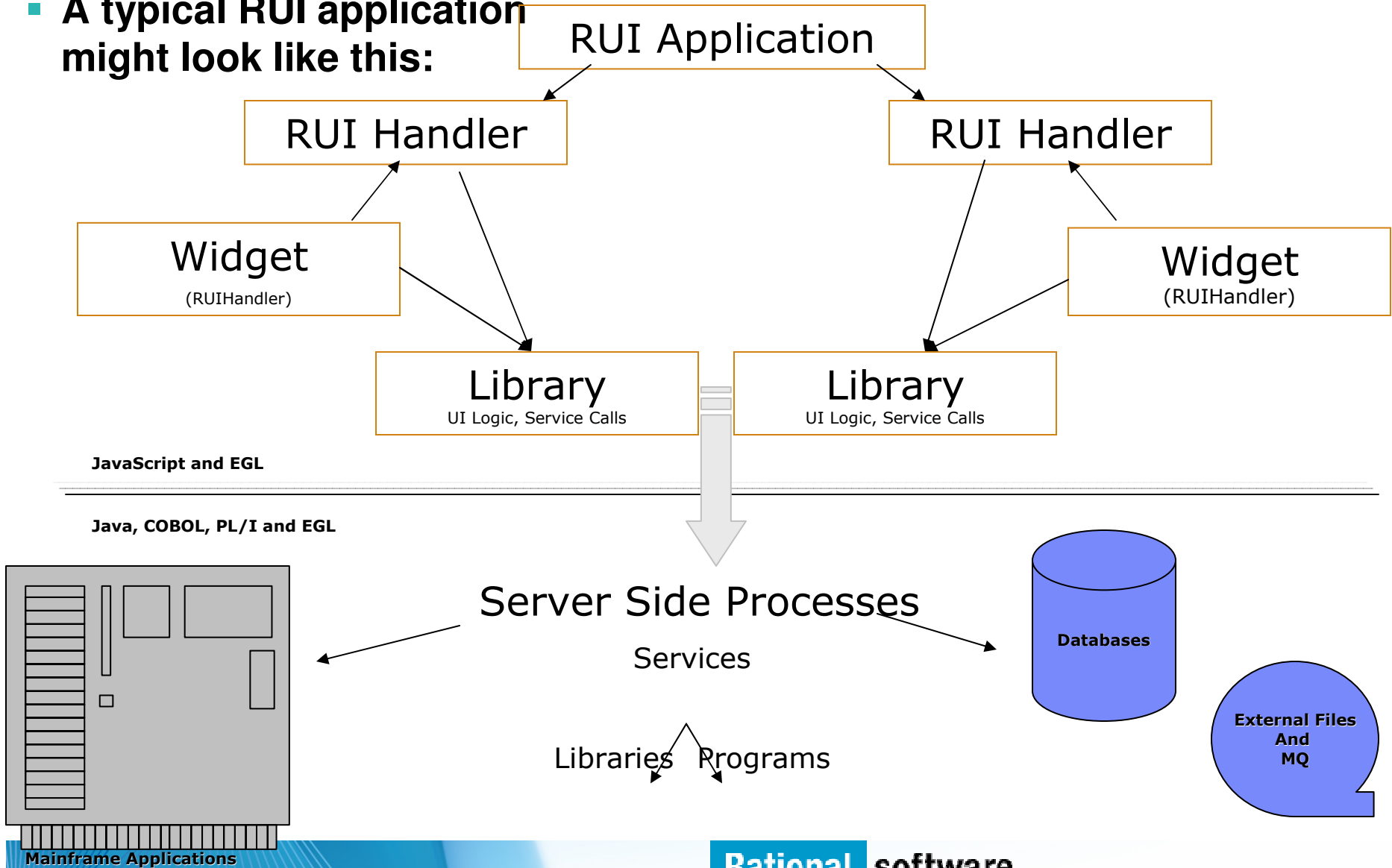
## EGL Rich UI

- Includes visual composition, libraries of RUI widgets
- Instant deploy/visualize while you code
- Fully open and extensible
- Easily integrate/consume any service (REST or SOAP)
- Single language end-to-end
- Brings legacy data and assets to Web 2.0



# RUI Programming – Overview

- A typical RUI application might look like this:



## RUI Programming - EGL

1.

2.

3.

### Single Language

- EGL in 3 tiers:
  1. Data and Logic
  2. Soap/Rest Services
  3. Declarative UI
- Just 1 language
- Skill transfer

### Rich User Interfaces

- Declarative UI
- Ajax support
- Rich set of widgets
- Easily extensible
- Mashup using SOA

### Mobility

- Disconnect/Sync
- Easy installation
- No version pain
- Flexibility



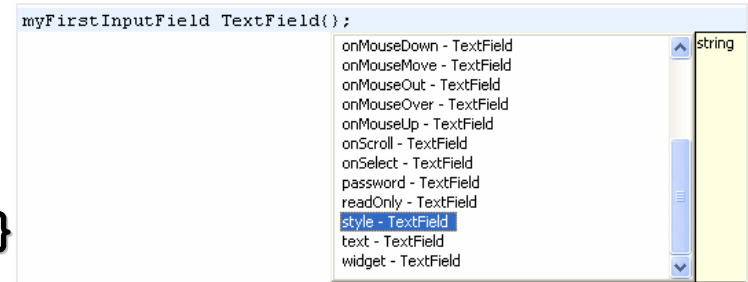
# Basics of RUI Programming

- **Designing web pages using RichUI is unlike using EGL with JSF and the traditional RBD Page Designer – where every .JSP page has EGL “backing” server-side code.**
- **The language is still EGL, but the structure of pages and U.I. logic parts will change with Rich UI.**
- **The RUIHandler part takes the place of both the EGL JSFHandler, and the JSP page (and its JSF Components):**
  - RUIHandlers are similar to JSFHandlers – and they’re written entirely in EGL
  - RUIHandlers control the way your page is rendered through the code that you write.
    - With JSF, one would create a variable in a JSFHandler, then bind it to controls on a jsp.
  - With RichUI, if a variable of a widget is created in a RUIHandler, it is automatically put onto the page
  - Wizards are available to assist you with creating these elements in your applications

## Basics of RUI Programming – TextFields

- To put a basic input field onto a page, you will first create a variable of type TextField.

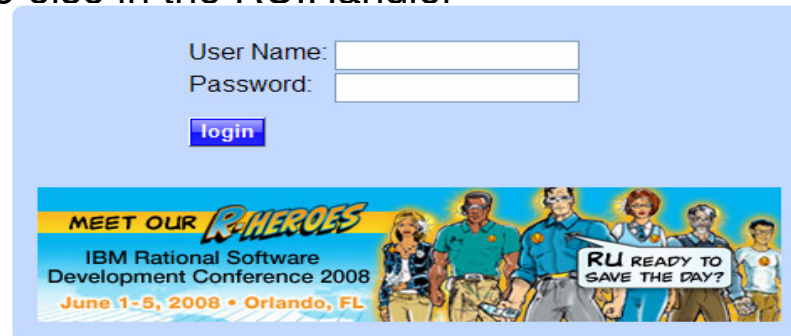
```
myFirstInputField TextField{}
```



- In EGL RichUI, variable properties (specified within the {} following the variable declaration) play a huge role.
  - Just declaring a variable of type TextField will give you the ability to put an input field onto a web page, but giving it initial values etc. need all be done within variable properties.

## Basic of RUI Programming - Children

- **Web pages created with RichUI are extremely flexible.**
  - For example, by simply changing the order in which children are assigned, a web page can be completely turned upside down ☺
  - On the previous slide, we showed that our mainContent VBox had children
    - [image, loginBox, login]
  - Changing the order of our children can render the following
    - [loginBox, login, image]
    - Note that each child is an individual widget declared just like EGL variables somewhere else in the RUIHandler

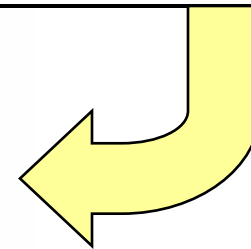
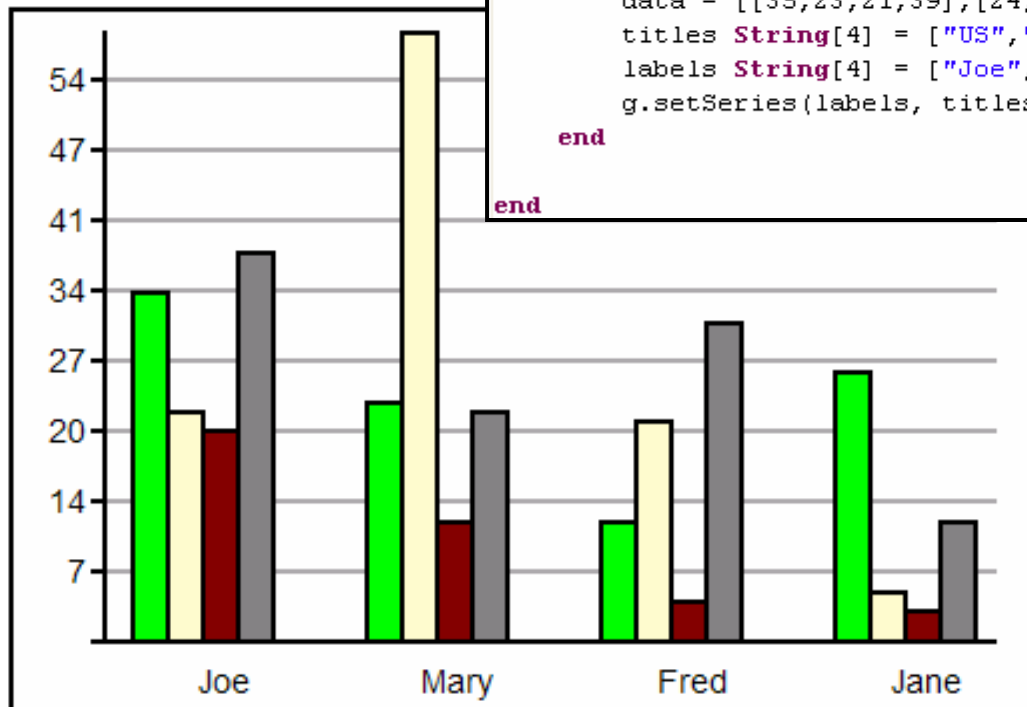


# Basics of RUI Programming - Widgets

- So what types of widgets can I use when programming in RichUI?
- The answer to this question is infinite!
- RichUI comes ships with a wide range of widgets, but also has the ability to pull in widgets from other technologies like DOJO and Silverlight. And last but not least, developers can create custom widgets using EGL RichUI!
- The list of widgets shipped with EGL is best found through Content Assist but here are a few (remember all are created by simply creating an egl variable of “widget type”):
  - Combo Boxes
  - Radio Buttons
  - Check Boxes
  - Image Fields
  - Charts
  - Single and Multi Selection List Boxes
  - Calendars
  - This list goes on!!!

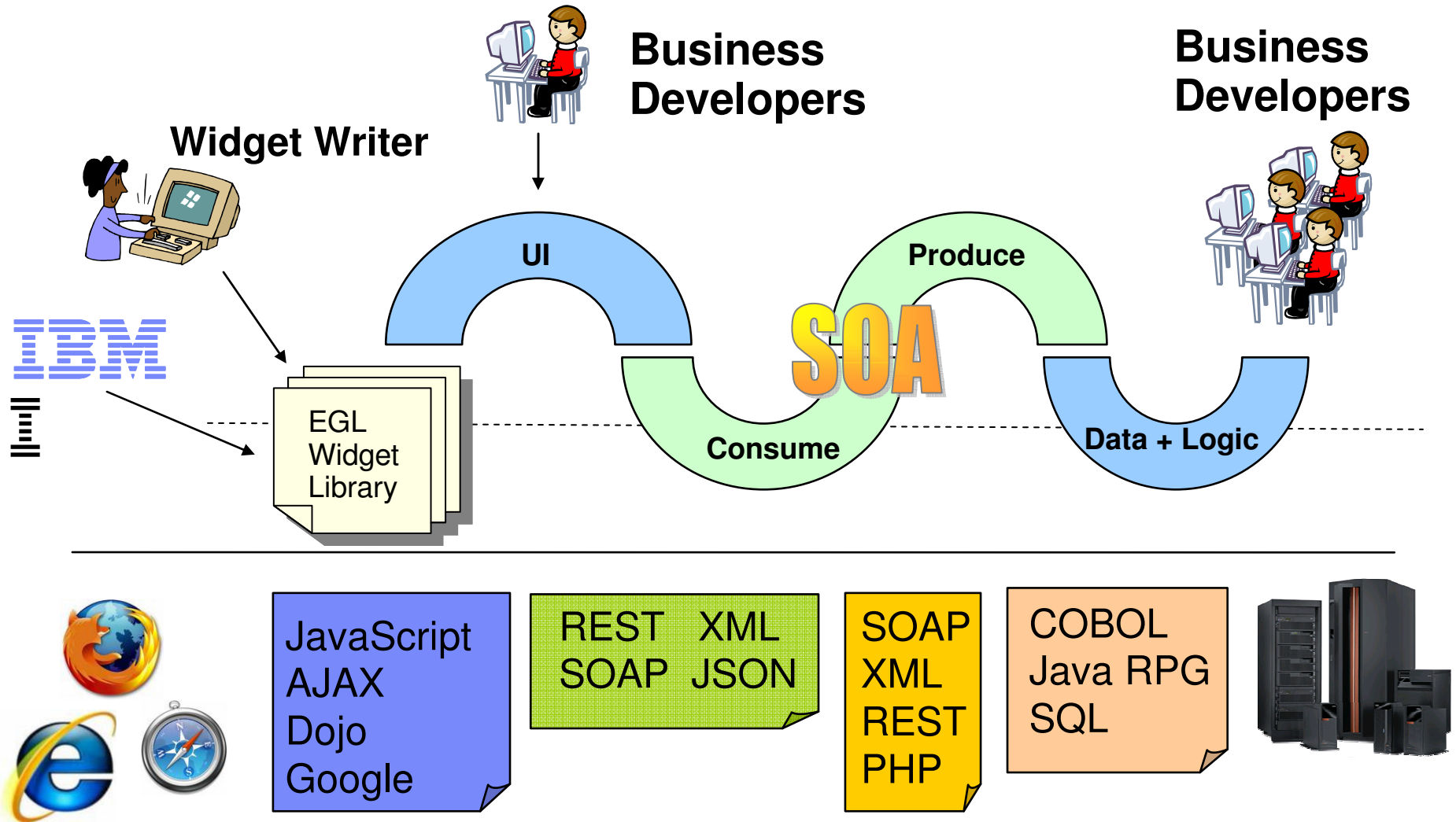
## EGL Simple Example (Drawing Graphics)

```
Handler GraphDemo Type RUIHandler { onConstructionFunction = myStartupFunction }  
  g Graph ( );  
  
  Function myStartupFunction(parent DOMElement)  
    g.init(parent);  
    data int[4][4];  
    data = [[35,23,21,39],[24,61,12,23],[12,22,4,32],[27,5,3,12]];  
    titles String[4] = ["US", "Asia", "Europe", "Africa"];  
    labels String[4] = ["Joe", "Mary", "Fred", "Jane"];  
    g.setSeries(labels, titles, data);  
  
  end  
end
```



Graph widget was user-written  
in 120 lines of Javascript

# EGL: Shielding Complexity



# Demo Tasks

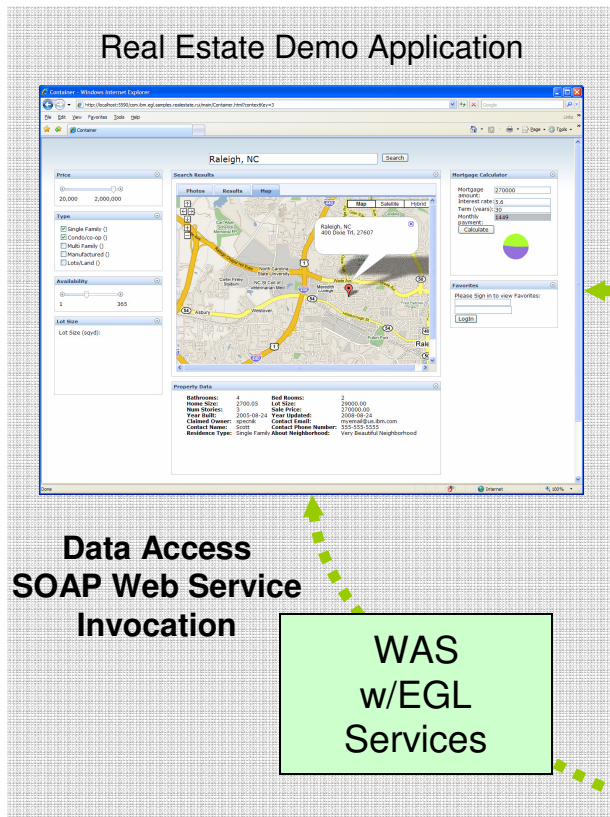
## Task 1 - Adding Mortgage Calculator

1. Consume CICS Web service in EGL Rich UI project
2. Create EGL interface code so service can be invoked
3. Create mortgage calculator UI
4. Add event listener to call service on

## Task 2 - Populating Data Table

1. Explorer database using tooling
2. Create EGL code representing data we want to use
3. Create EGL services to return data to RUI application
4. Examine data table code

lays  
life of

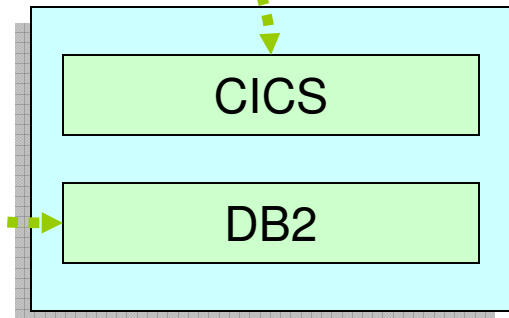


Data Access  
SOAP Web Service  
Invocation

WAS  
w/EGL  
Services

Your Workstation

Mortgage  
SOAP Web Service  
Invocation




zserveros.demos.ibm.com



## EGL Rich UI

Mortgage Calculator, Data Table, and Google Map





# Modern Application Development Featuring Web 2.0 for System z

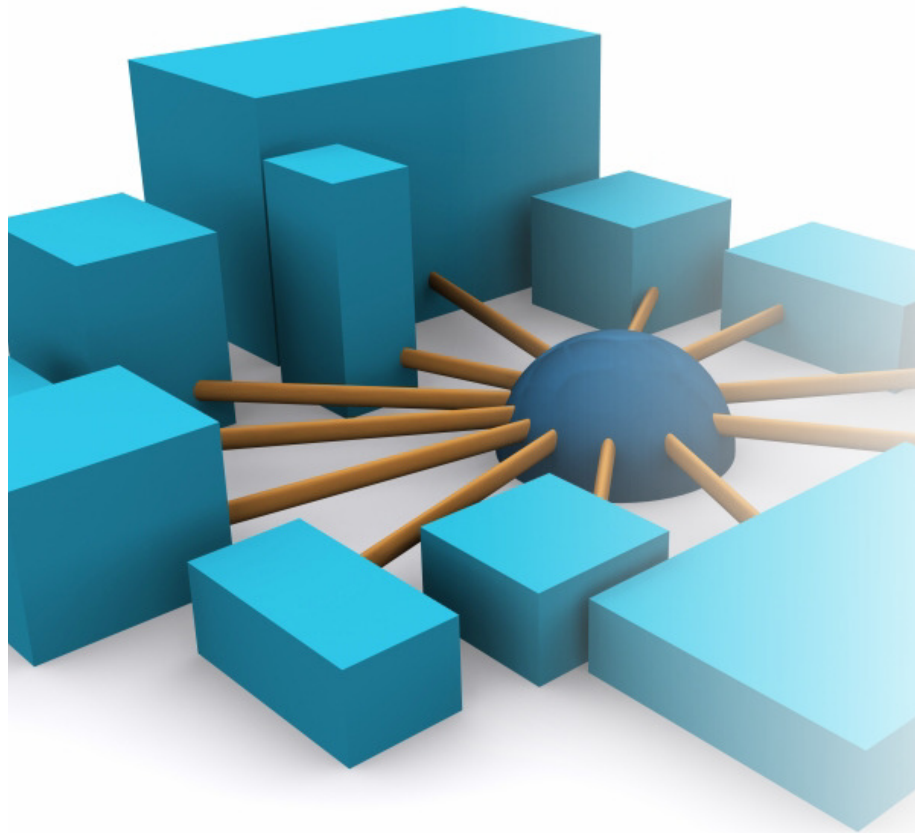
Application Lifecycle Management with  
ClearCase, ClearQuest, and Build Forge

**Rational.** software

# Management Challenges for n-tier Applications

- **Heterogenous platforms/consistent functions**
  - z/OS, i5os
  - Linux/UNIX variants
  - Windows variants
- **Parallel activities (new development, maintenance)**
- **Consistent user interface & process paradigms**
- **Automated, heterogeneous builds**
- **Coordinated promotions & backout**
- **Impact analysis**
- **Deployment support**

## ***Solution: Simplify your Environment Standardize your cross-platform application development***



### **Rational Developer for System z**

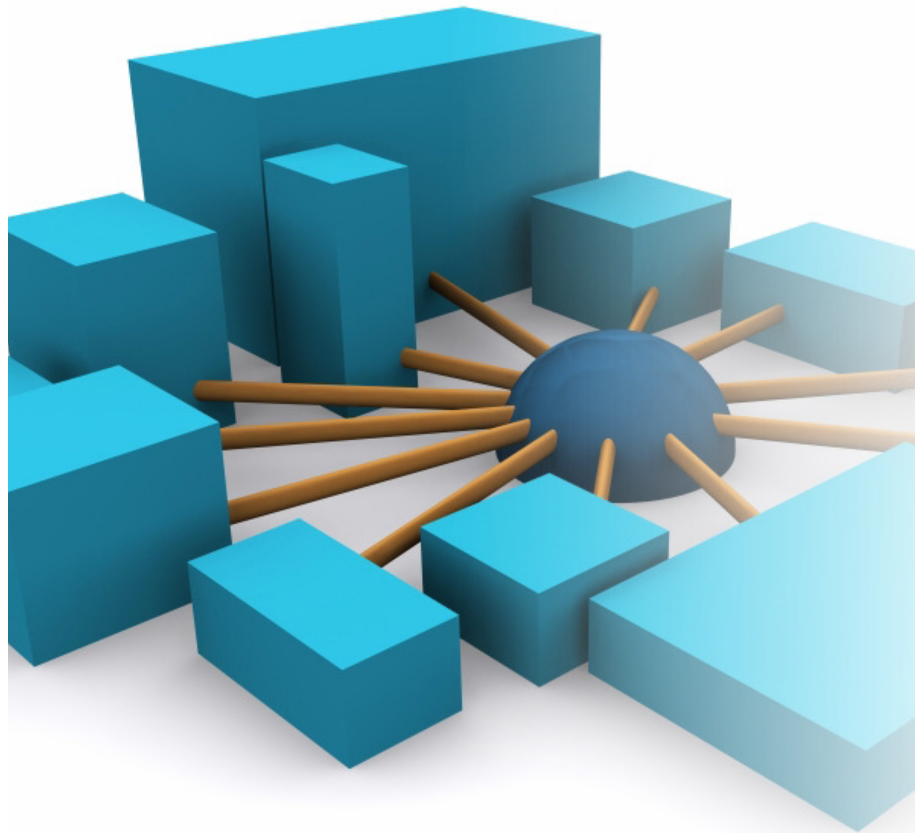
#### **Integrated Host/desktop IDE**

- Local COBOL/PL1 edit, syntax check
- Integrated java, C/C++ development
- Automated host/local filesystem synchronization
- Remote test, debug, job monitoring

*Rational Developer for System z*

**Rational.** software

***Solution: Simplify your Application Lifecycle  
Standardize your development process across your enterprise***



**IBM Rational ClearQuest**

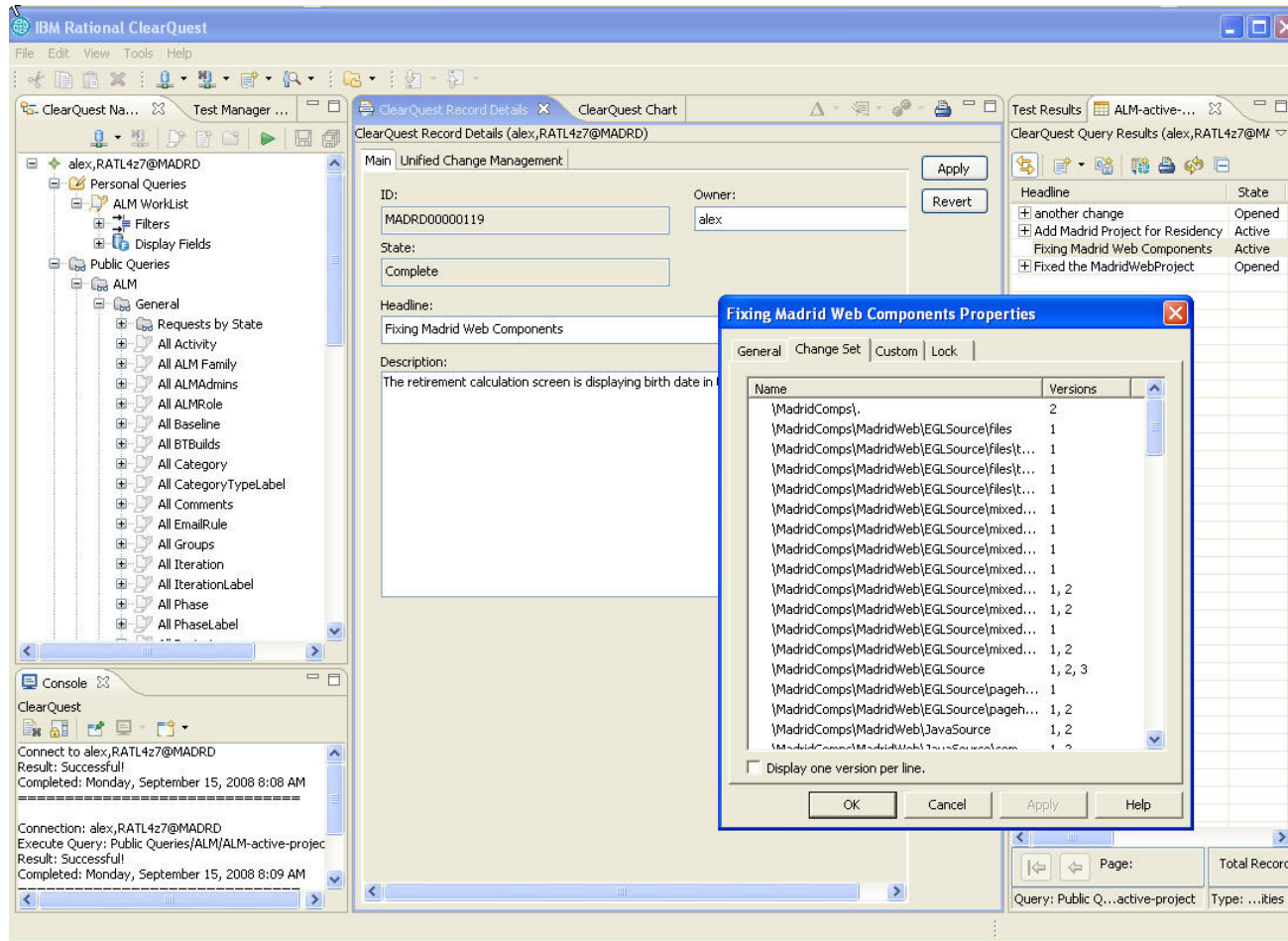
**Activity and Deployment  
Management**

- Configurable, enforceable workflows
- Central repository, real-time status
- Graphical and text-based reporting
- RDz (Eclipse), Web, Windows clients

*IBM Rational ClearQuest*

**Rational.** software

# Rational ClearQuest: Automating Activity Management



Windows



UNIX/Linux



Eclipse



e-mail

# Automating Workflow Management

The screenshot displays the IBM Rational Developer for System z environment. The main window shows the ClearQuest Navigator on the left, listing various queries and reports. The central pane displays the 'ClearQuest Chart: Active Defects by Owner' window, which contains a 3D bar chart. The chart shows the count of active defects for each owner. The x-axis lists the owners: admin, alex, Brian, chris, dale, dana, devon, jan, morgan, sandy, and tracy. The y-axis represents the count, ranging from 0 to 5. The bar for 'dale' is the tallest, reaching a count of 5. A 'Capture Image' button is visible at the bottom right of the chart area, with a tooltip indicating the keyboard shortcut <CTRL><F11>.

Owner	Count
admin	2
alex	1
Brian	1
chris	3
dale	5
dana	1
devon	1
jan	2
morgan	3
sandy	2
tracy	1

Console Log:

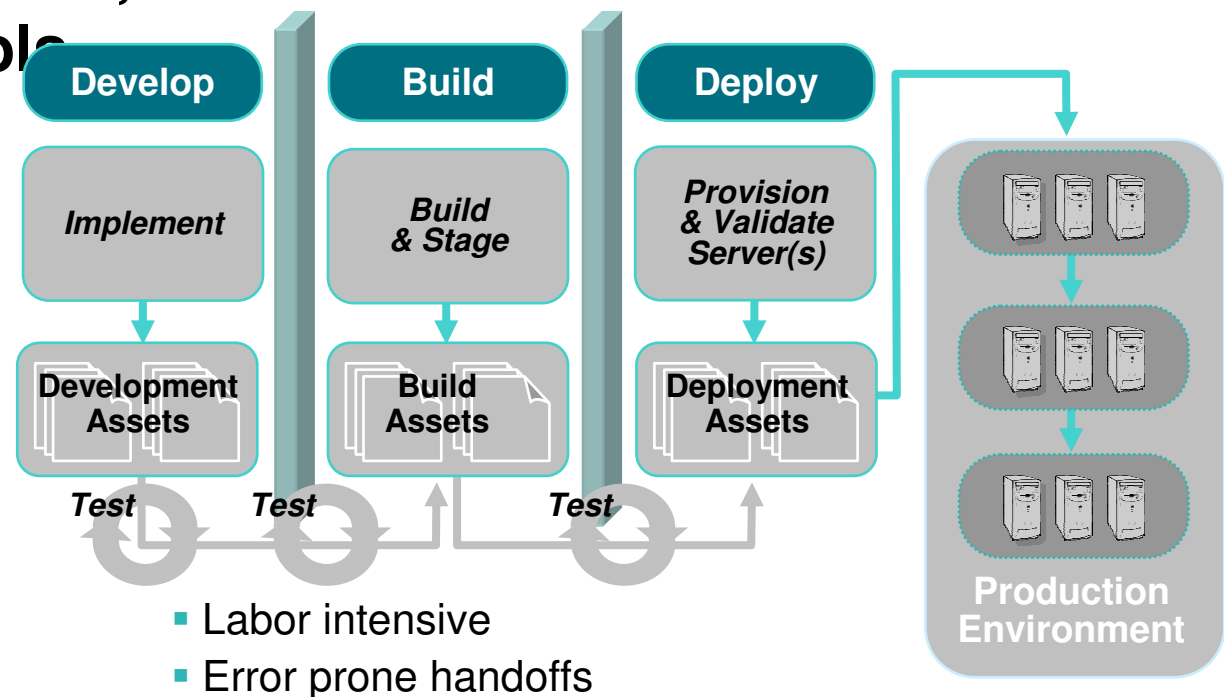
```
ClearQuest
Connect to admin,RATL427@AUCT
Result: Successful!
Completed: Thursday, January 24, 2008 12:36 PM

-----

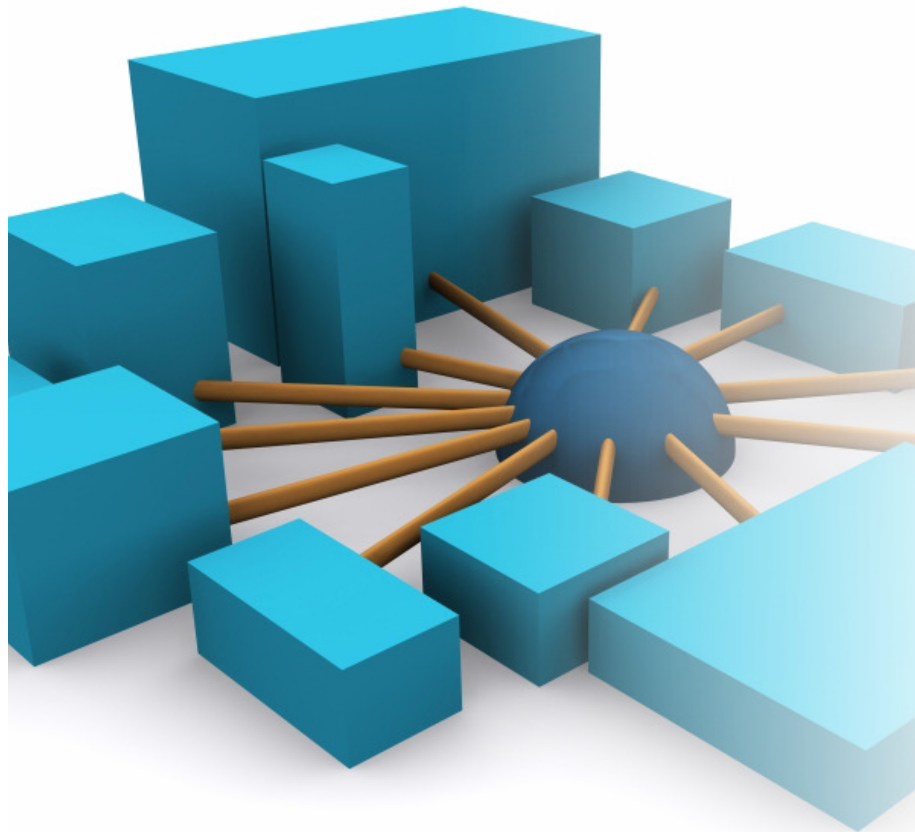
Connection: admin,RATL427@AUCT
Execute Chart: Public Queries/Distribution Charts-All Projects/Active Defects by O
Result: Successful!
Completed: Thursday, January 24, 2008 12:41 PM
```

## Challenge: Silos Between Development, Build and Deployment

- Difficulty tracking testing status of multiple builds
- No shared processes, artifacts or controls
- “Over the wall” communication
- Minimal reuse



## ***Solution: Simplify Builds and Deployments*** ***cross-platform application builds and deployments***



### **Rational Build Forge**

#### **Synchronize enterprise builds**

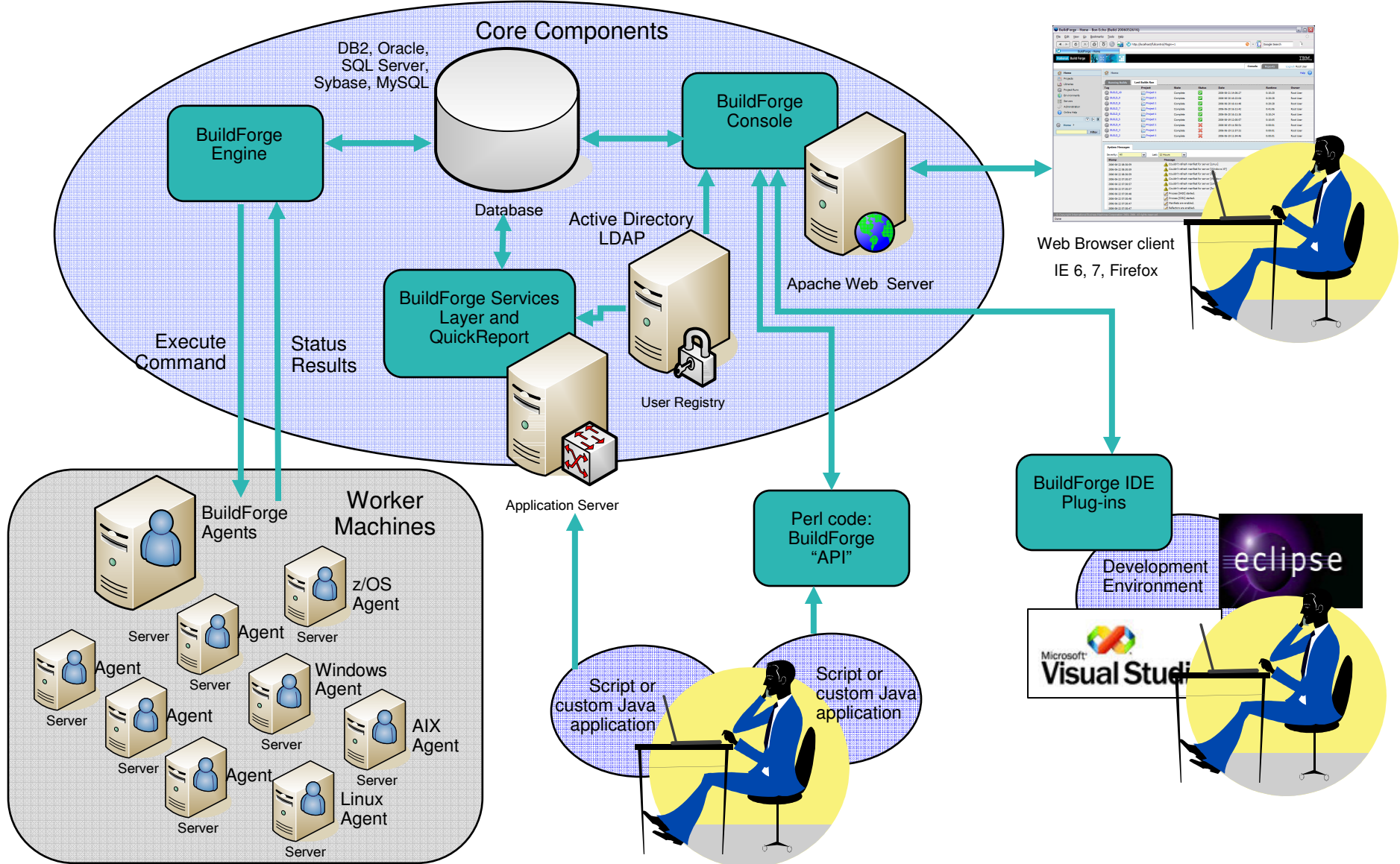
- Centralized build management
- Local (host and distributed) agents
- RDz (Eclipse), Web clients
- Comprehensive audit trails

*Rational Developer for System z*

**Rational.** software



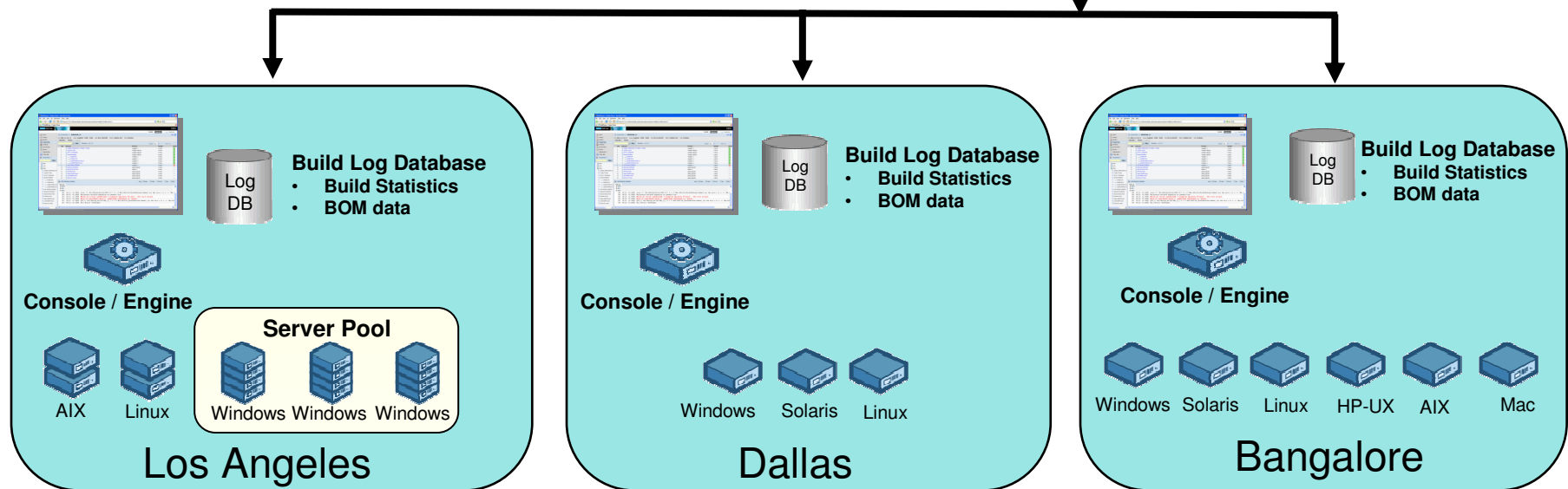
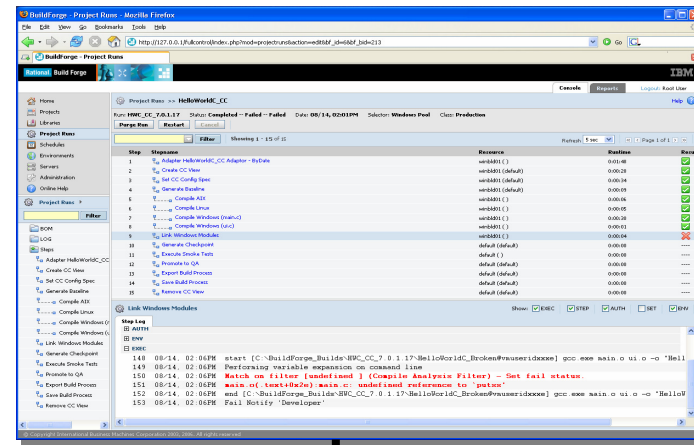
# Build Forge System Architecture



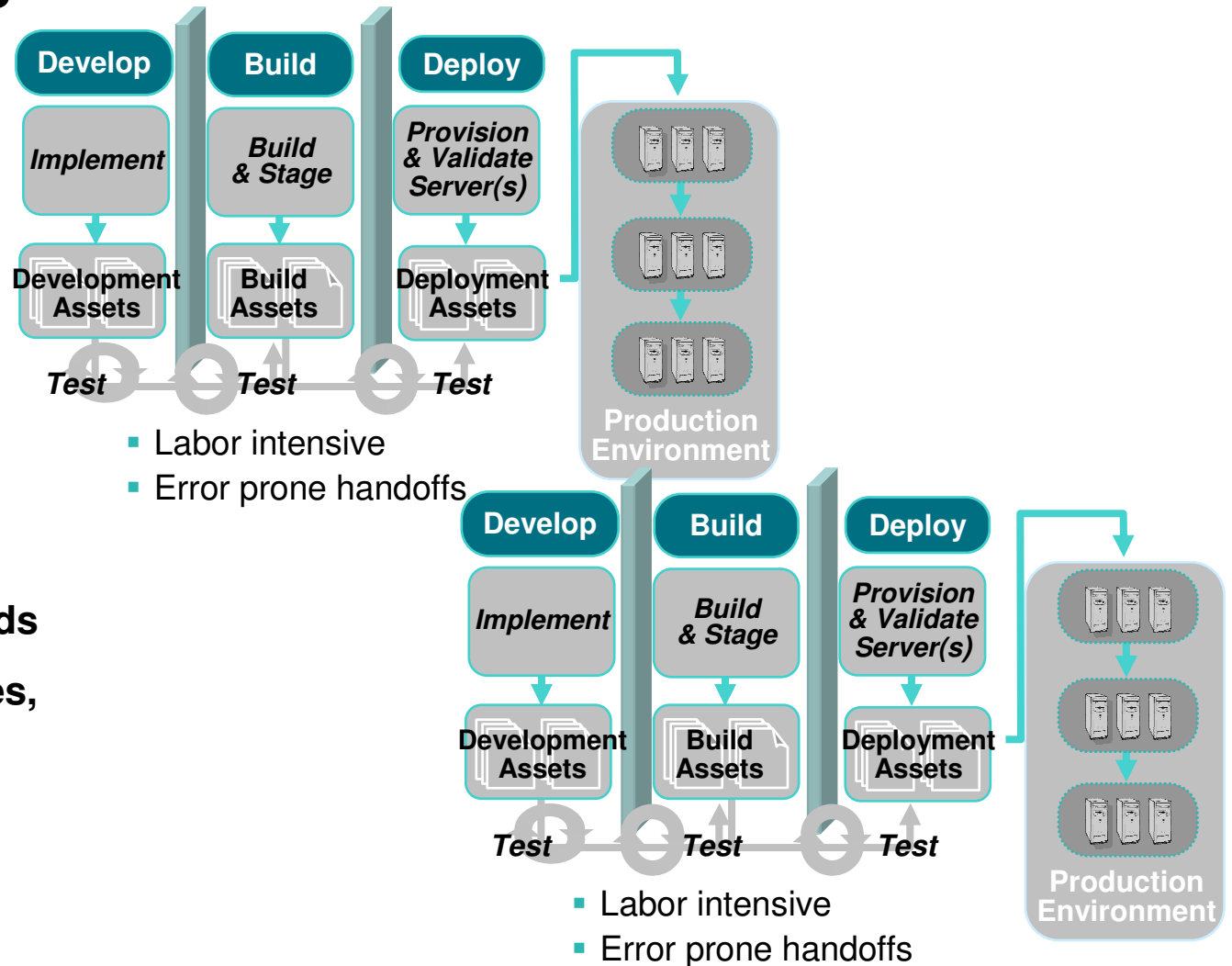
# IBM Rational Build Forge

## Centralized management across geographically distributed teams

- Web-based management console
- Consolidated, real-time project view
- Safe, secure access via user authentication
- Unicode support for international projects

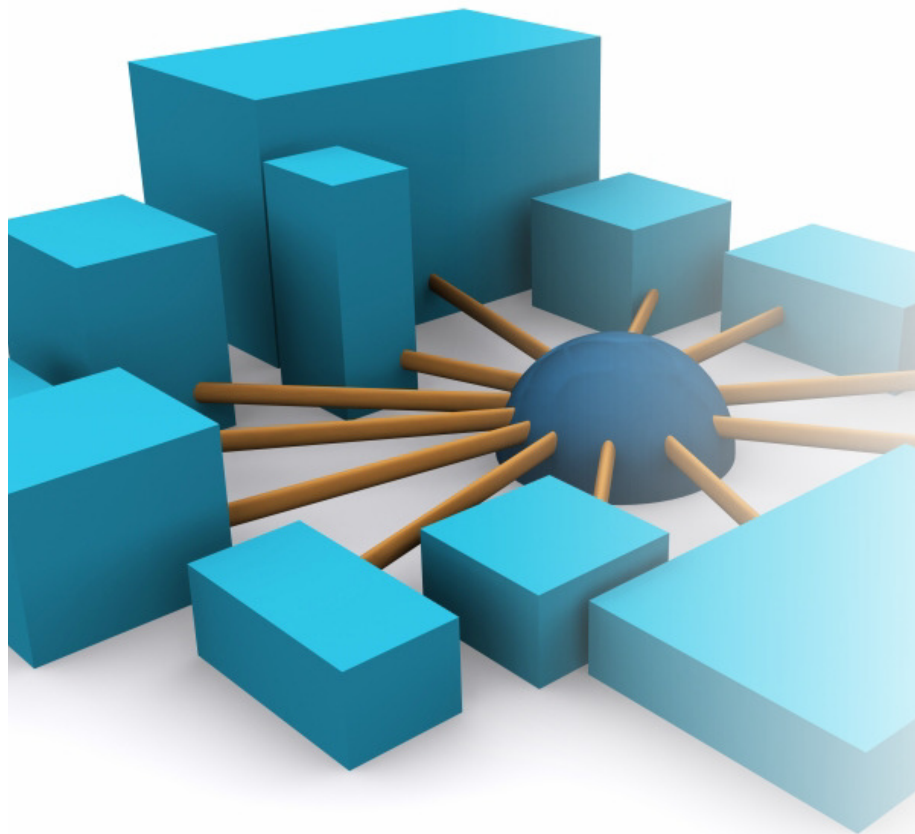


# Challenge: Silos Between z/OS & Distributed Environments



- Difficulty tracking migrations and builds
- No shared processes, artifacts or controls
- “Over the wall” communication

## ***Solution: Simplify Software Asset Management*** ***Manage enterprise assets with a single repository solution***



### **Rational ClearCase**

#### **Enterprise-level SCM**

- Secure hosting of software assets
- Desktop, RDz, z/OS, web clients
- Distributed and z/OS build capability
- Built-in extensible process model

*Rational ClearCase*

# Rational ClearCase: Automating SCM

## Workspace management

- Work in isolated workspaces
- Work in parallel with other developers
- Manage multiple releases

**Stability**

## Version control

- Version all types of files and directories in secure repositories
- Merge versions of elements

**Safety**

## Build management

- Ensure the integrity of builds
- Accurately reproduce every release
- Trace and reproduce builds

**Reproducibility  
Traceability**

## Process control

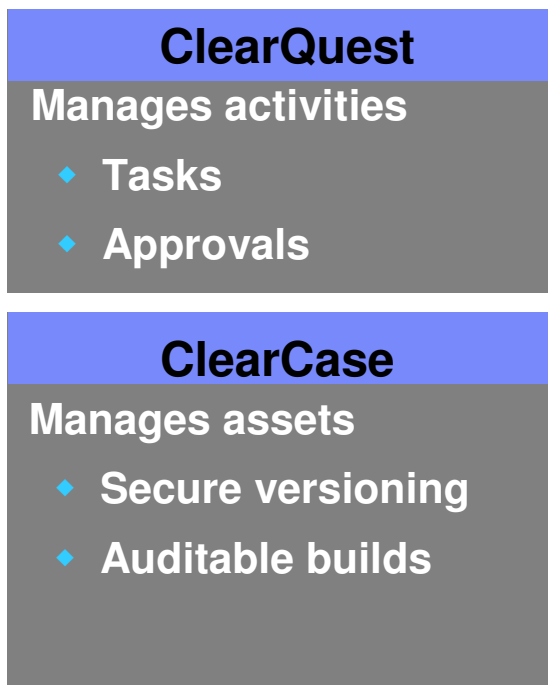
- Record and report actions, history, and milestones
- Customize process
- Automate defect and change tracking

**Auditability  
Control**

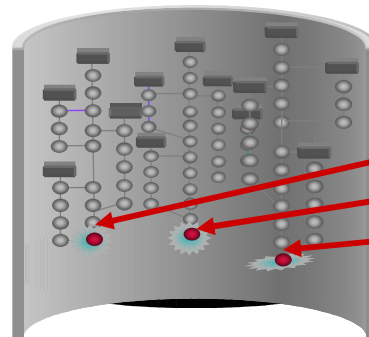


# The Process Model: Unified Change Management

- Automatically maps ClearCase change sets to ClearQuest activities
- Built-in, fully customizable promotion model
- Implements a component-based application lifecycle approach

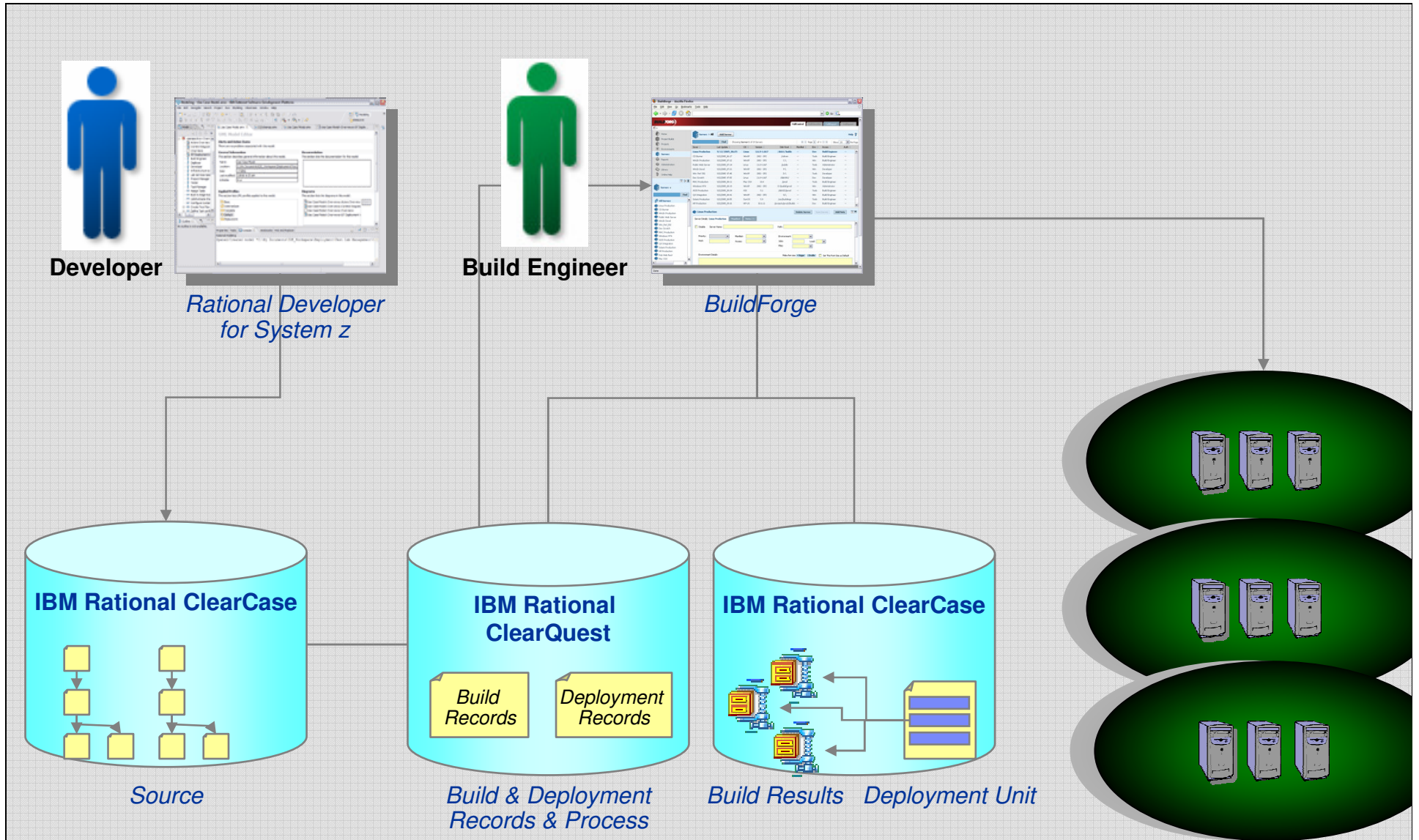


ClearQuest: Organized Activities			
Request	Priority	Owner	
Special Promo	1	Terry	
Bug 527	2	Sandy	
Add GUI button	2	Kim	



Change Set	
Special Promo	
a. html	V5
b. xml	V3
c. jpg	V8

# Bringing it all together . . .



# Building with RDz and ClearCase Remote Build

The screenshot illustrates the workflow for building a project in IBM Rational Developer for System z using ClearCase Remote Build. The main editor shows the JCL file for the program 'MADLO2D', which is identified as 'Web Services for CICS TS 3.x Converter Driver'. The 'Properties for CC-Remote-Build' dialog box is configured with the following details:

- Name:** CC-Remote-Build
- Connection:** CC-view
- Build Invocation:** "C:\Program Files\Rational\ClearCase\bin\rccbuild.exe"
- Working Directory:** Y:\MadridComps\MadridDemo
- Arguments:** -h demomvs.demopkg.ibm.com@3606 -b COMP -ft COMP.BCL -v USR=DNET471 -v MBR=MADLO2
- Project:** Madrid Demo

The Command Shell - Running window shows the execution of the build command:

```
Y:\MadridComps\MadridDemo>
cd Y:\MadridComps\MadridDemo

08/09/15 09:38:53 (c) Copyright, IBM Corp. 2001
Copyright (c) 2002 Rational Software Corporation
08/09/15 09:39:00 *** Success ***
08/09/15 09:39:01
RCCI-003
Program Name : 'IGYCRCTL'.
PARM : 'EXIT(ADEXIT(ELAXNGUX)),&DATA,LIB,TEST(NONE,SYM,SEP),LIST,FLAG(I,
RCCI-004
The MVS step 'COMP' return code is '000000'.
08/09/15 09:39:01 Message files from build:
08/09/15 09:39:01 1:COMP.SYSPRINT
08/09/15 09:39:02 *
```

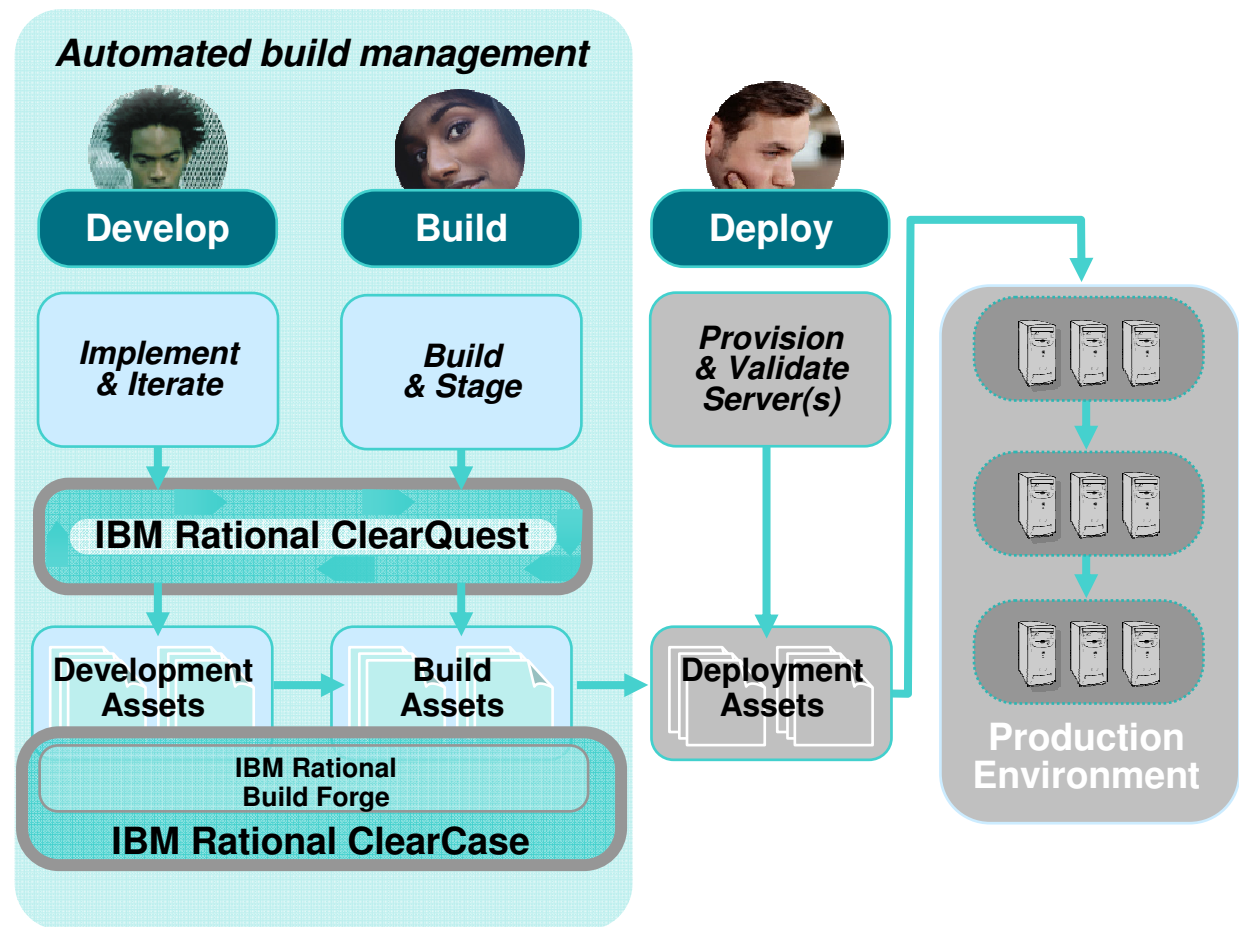
The Command Shell - Command window shows the full command used for the remote build:

```
Command "C:\Program Files\Rational\ClearCase\bin\rccbuild.exe" -h demomvs.demopkg.ibm.com@3606 -b
```

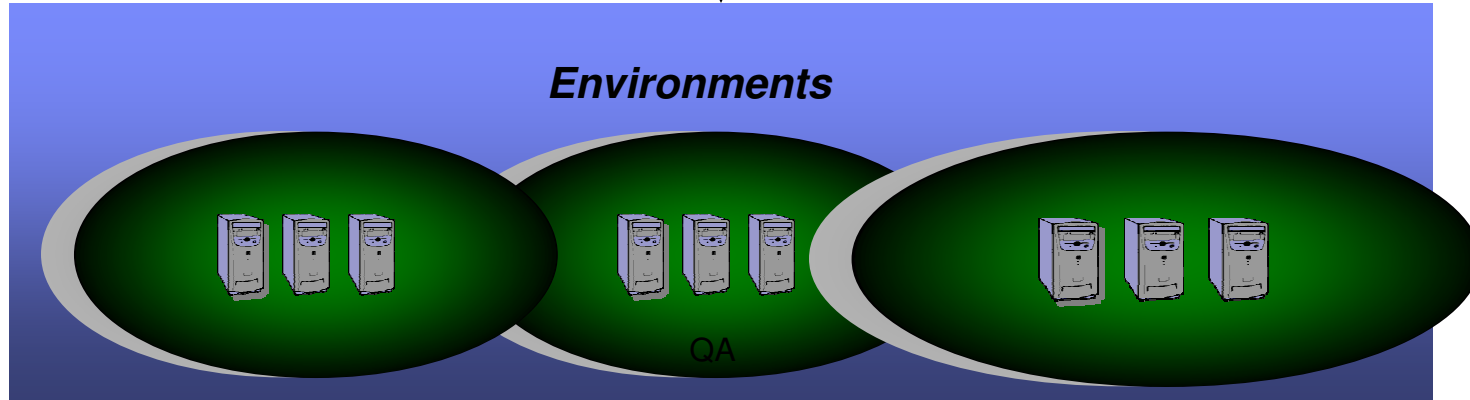
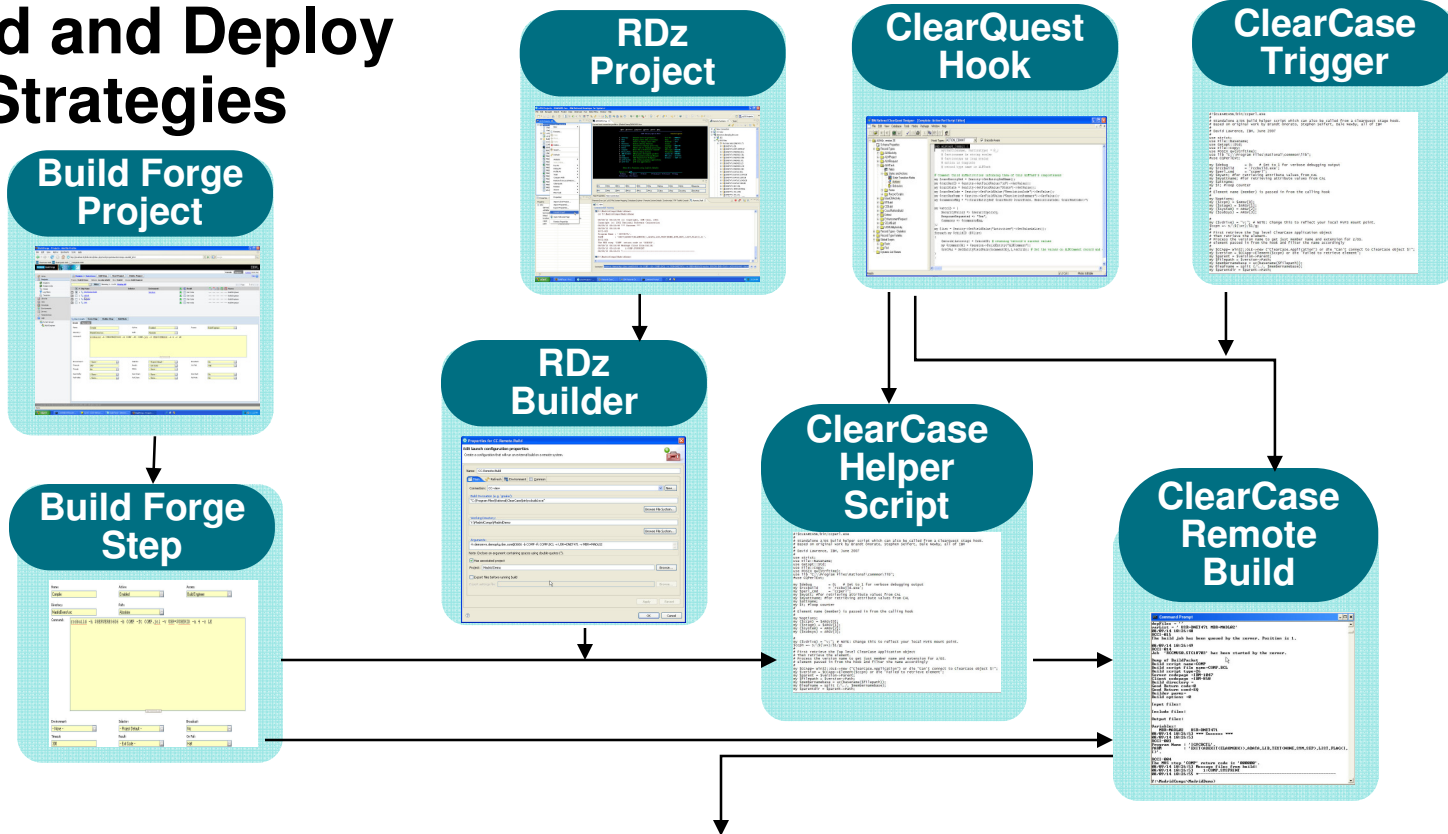



## IBM Rational ClearQuest and IBM Rational ClearCase with IBM Rational Build Forge integration

- Automated build management
- Detailed record of build content
- Support for continuous builds
- New IBM Rational ClearQuest build record for tracking and reporting



# Build and Deploy Strategies



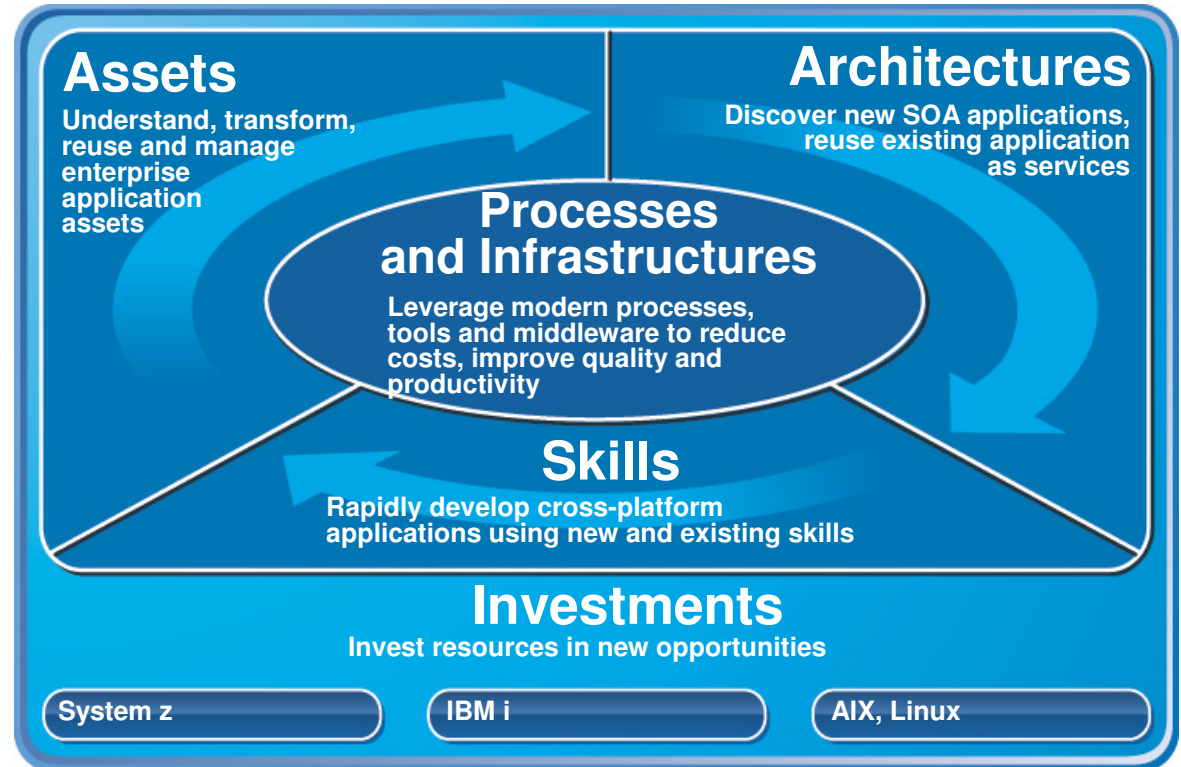


# Modern Application Development Featuring Web 2.0 for System z

Summary and Next Steps

**Rational.** software

# Summary



## Rational and WebSphere ...

- are working closely with the System z hardware team
- have a strong portfolio of tools for System z application development
- are investing in new tools for application development for System z

# Resources to help you kick-start modernization

## New Educational Resources

### System z Sandbox

*Example assets and best practices providing low-risk, practical, hands-on path to leveraging IBM solutions*

### EGL Distance Learning

*10 days of FREE instructor-led virtual class*

### Seminar Series: Web 2 Architectures for IBM Mainframe Developers

*For managers and technologists focused on extending value of mainframe applications in modern Web 2 Architectures*

## Get quick ROI today

**Rational Asset Analyzer** – find out what assets can be reused today

**Host Access Transformation Services** – quickly turn your green screens into web UIs

**Rational AppScan** – make sure your web sites are hacker-proof

## System z Solution Videos and Demos

*Translated solution videos and demo to help you raise awareness of Enterprise Modernization offerings*



- English
- German
- Korean
- Spanish
- Mandarin Chinese
- Japanese
- French
- Italian

<http://rational.dfw.ibm.com/index.html>

**Jump-start  
your  
modernization  
projects!**

<http://www.ibm.com/rational/modernization>



## z/OS Enterprise Modernization POTs

- A Proof of Technology, is designed to give attendees a "deep dive" of the technology through demonstration, presentation, and hands on labs in a controlled environment.
- PoTs can run from one to three days in length, and are usually held in a Technical Exploration Center.
- The objective is to expose the usability, documentation, and manageability of an IBM product.

Free of charge

<u>Discovering the value of WebSphere Studio Asset Analyzer</u>	<b>1 Day</b>
<u>Discovering the value of IBM Host Access Transformation Services (HATS)</u>	<b>1 Day</b>
<u>Discovering the value of IBM Rational Developer for System z</u>	<b>2 Days</b>
<u>Discovering the value of Rational Business Developer and EGL to develop Java or COBOL Applications</u>	<b>2 Days</b>

Contact your IBM representative for dates

## Rational EM tools on the Web

- **RDz with Java or RDz with EGL**  
<http://www.ibm.com/software/awdtools/rdz/>
- **Rational Business Developer**  
<http://www.ibm.com/software/awdtools/developer/business/index.html/>
- **Rational Host Access Transformation Services**  
<http://www.ibm.com/software/awdtools/hats/>
- **Rational Transformation Workbench**  
<http://www.ibm.com/software/awdtools/rtw/>
- **WebSphere Studio Asset Analyzer**  
<http://www.ibm.com/software/awdtools/wsa/>
- **Rational Asset Analyzer**  
<http://www.ibm.com/software/awdtools/raa/>
- **Rational Team Concert**  
<http://www.ibm.com/software/awdtools/rtc/index.html>



# Want to learn more?

- **Get this complimentary kit for mainframe developers for a practical, how-to guide for making the most of an existing development environment, including the skills and infrastructure already in place at an established enterprise.**
  - Did you say mainframe? e-kit  
<http://www.ibm.com/developerworks/integrate/kits/mainframe/>
    - This kit contains the following:
      - **Podcast series:** Did you say mainframe?
      - **Webcasts**
      - **Tutorials**
      - **White and red papers**
      - **Demos**
      - **developerWorks articles and other resources**
- **Find more Enterprise Modernization System z resources**
  - <http://www.ibm.com/software/info/developer/solutions/em/systems/z/>
- **You can also find out more about other IBM Enterprise Modernization offerings at this site**
  - <http://www.ibm.com/software/info/developer/solutions/em/>



# Enterprise Modernization Sandbox for System z

***Quickly try practical scenarios guided by self-paced exercises 24/7***

## ***System z Sandbox***

*Examples and best practices provide a low-risk, practical, hands-on path to understanding*

- Rational Developer for System z
- Rational Business Developer
- Rational Transformation Workbench
- Rational Host Access Transformation Services



- ✓ ***Launched on 2/26/08***
- ✓ ***Over 250 customer registrations to-date***
- ✓ ***13 hands-on tutorials are live***
- ✓ ***Additional scenarios coming soon***

### ***New scenarios planned for 2H08***

- Rational Asset Analyzer
- IMS, DataStudio, Info 2.0, Rational Data Architect
- WAS, CICS, PD Tools
- Rational ClearCase, ClearQuest, and BuildForge

*Full version software trials*

*'Try online' hosted environments*

*Tutorials*

*Online resources*

<http://www.ibm.com/developerworks/downloads/emsandbox/>

## Getting Started With EGL

- **No Charge 2 week Distance Learning Course offered by IBM Rational**
- **Next Class**
  - Check Web Site
  - Register on-line
- **Register at:**

<http://www.ibm.com/developerworks/offers/techbriefings/details/rbdtws.html>

### Distance Learning

- Week 1 – 4 days
  - ✓ 2 hours lecture
  - ✓ 4 hours lab
- 1-2 week break
- Week 2 – 4 days
  - ✓ 2 hours lecture
  - ✓ 4 hours lab
- Extensive instructor interaction

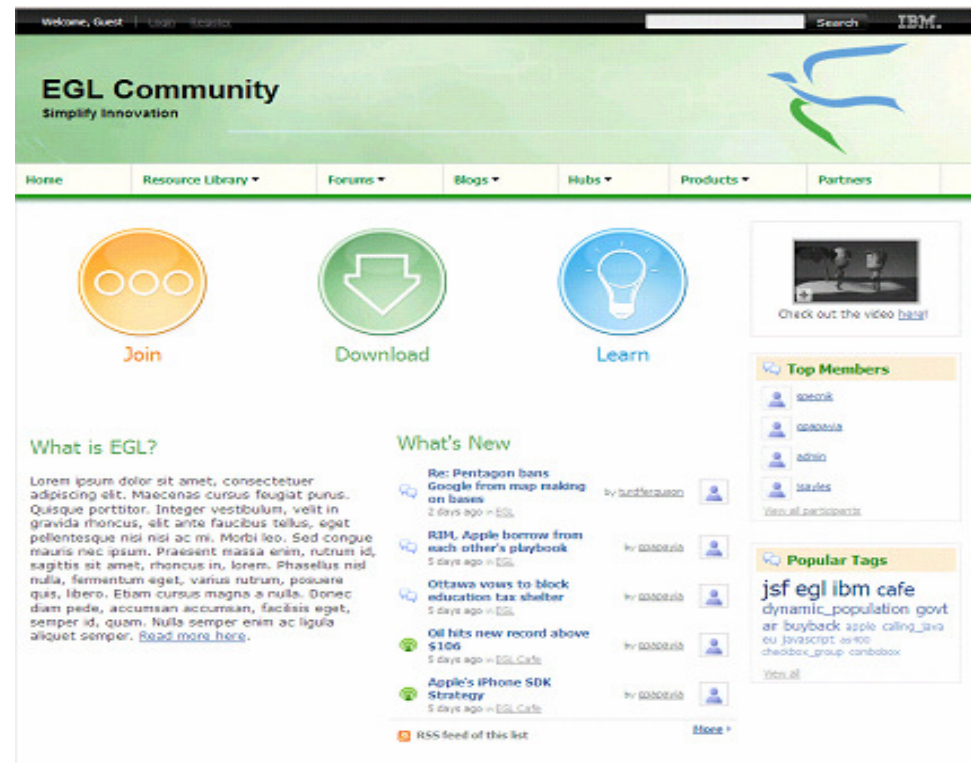
# Rational Business Developer – EGL Café

## *Building a Community for Business Developers*



- Attract participation with compelling experience – Partners, Users, Prospects
- Capture “Wisdom of the Crowd” – Experience knowledge base
- Sustainable Participation – Team, Users
- Leverage re-use to reduce effort
- Extending community to CICS, IMS, COBOL, and PL/I developers

<http://ibm.com/rational/eglcafe>



**Resources:** Download, Learn, Presentations, Video/viewlet, Sample Code

**Community:** Clients, Partners, Influencers, Press, News and Events

**Collaboration:** Blogs, Forums, Tips and Techniques Comments, Ratings

**Testimonials:** Case Studies, Celebrations!

# Questions



Learn more at: [IBM Enterprise Modernization Solution](http://www.ibm.com/software/info/developer/solutions/em/)  
(<http://www.ibm.com/software/info/developer/solutions/em/>)

- [IBM Rational software](http://www.ibm.com/software/rational/)  
(<http://www.ibm.com/software/rational/>)
- [IBM Rational Software Delivery Platform](http://www.ibm.com/software/info/developer/index.html)  
(<http://www.ibm.com/software/info/developer/index.html>)
- [Process and portfolio management](http://www.ibm.com/software/rational/offerings/ppm/)  
(<http://www.ibm.com/software/rational/offerings/ppm/>)
- [Change and release management](http://www.ibm.com/software/rational/offerings/crm/)  
(<http://www.ibm.com/software/rational/offerings/crm/>)

© Copyright IBM Corporation 2008. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.



## Other links for information and learning

### Quality management

(<http://www.ibm.com/software/rational/offerings/quality/>)

### Architecture management

(<http://www.ibm.com/software/rational/offerings/architecture/>)

### Rational trial downloads

([http://www.ibm.com/developerworks/rational/downloads/?S\\_TACT=105AGX23&S\\_CMP=RCD](http://www.ibm.com/developerworks/rational/downloads/?S_TACT=105AGX23&S_CMP=RCD))

### Leading Innovation Web site

(<http://www.ibm.com/software/rational/leadership/leaders/>)

developerWorks Rational (<http://www.ibm.com/developerworks/rational>)

### IBM Rational Business Partners

(<http://www.ibm.com/software/rational/partners/>)

© Copyright IBM Corporation 2008. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.