Your options for optimizing your web service creation with IBM Rational offerings

Michelle A. Cordes
Rational Ecosystem team
mcordes@us.ibm.com

Rational software

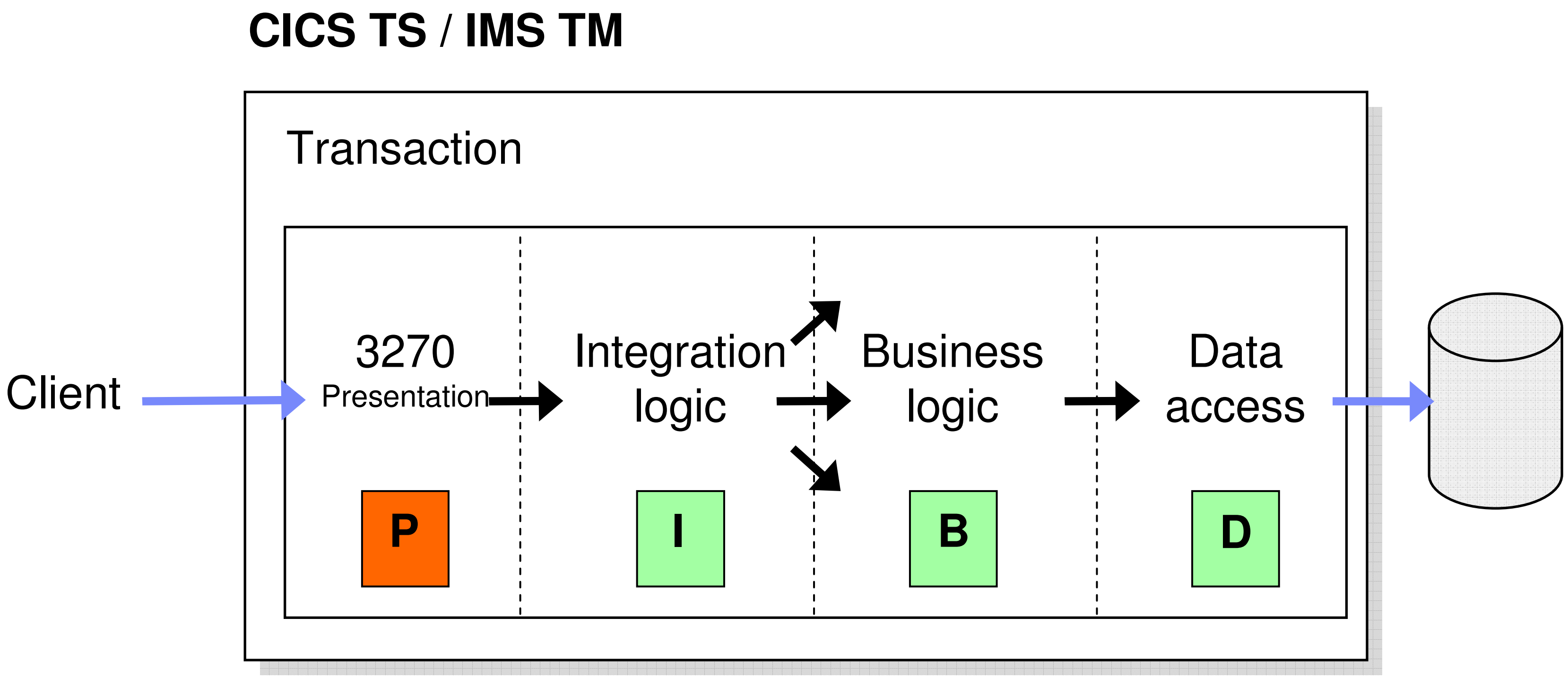
Agenda

- **What are web services?**
  - **How do they play in a mainframe environment?**
- Approaches to creating web services in the mainframe environment
  - Wrapping
  - Re-engineering
  - Redeveloping
- Summary
- Q/A

Transforming the Enterprise

- Many leading enterprises have chosen IBM Mainframes running CICS or IMS software to provide service functions
- The proven strengths of System z has, over time, led to a huge investment in application code and skills
- To be agile in today's fast-moving marketplace, companies must embrace new technologies while leveraging these assets
- System z maintains and adds value by enabling traditional, proven assets to be consumed as Web Services

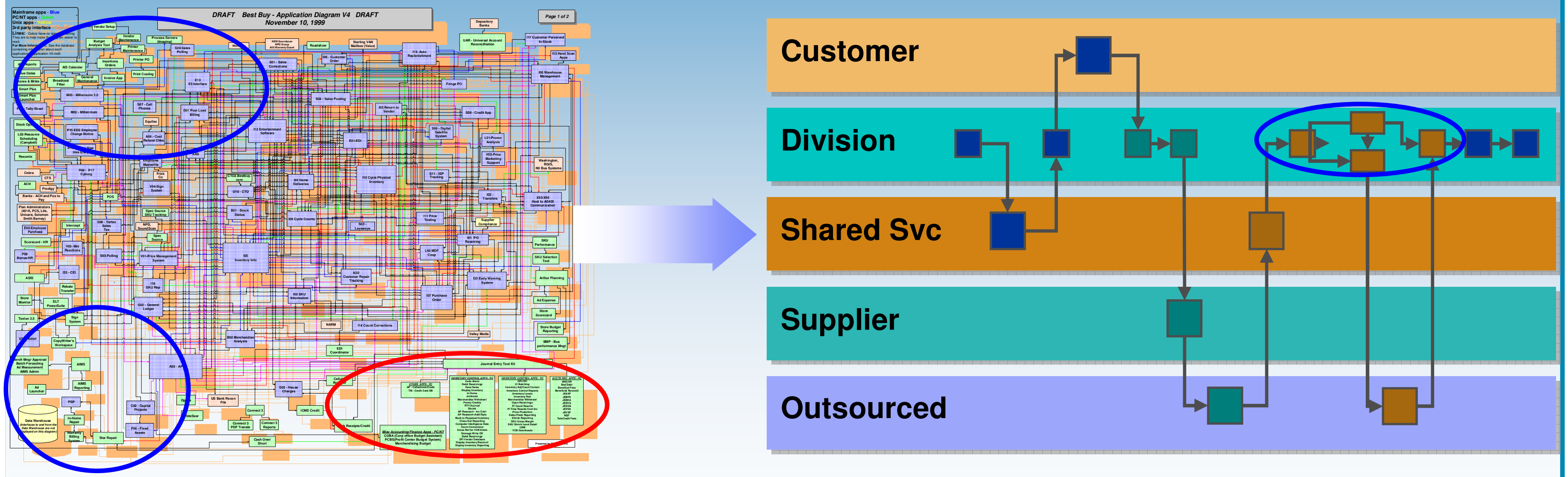
Modern “System z” architecture



Modernize Architectures

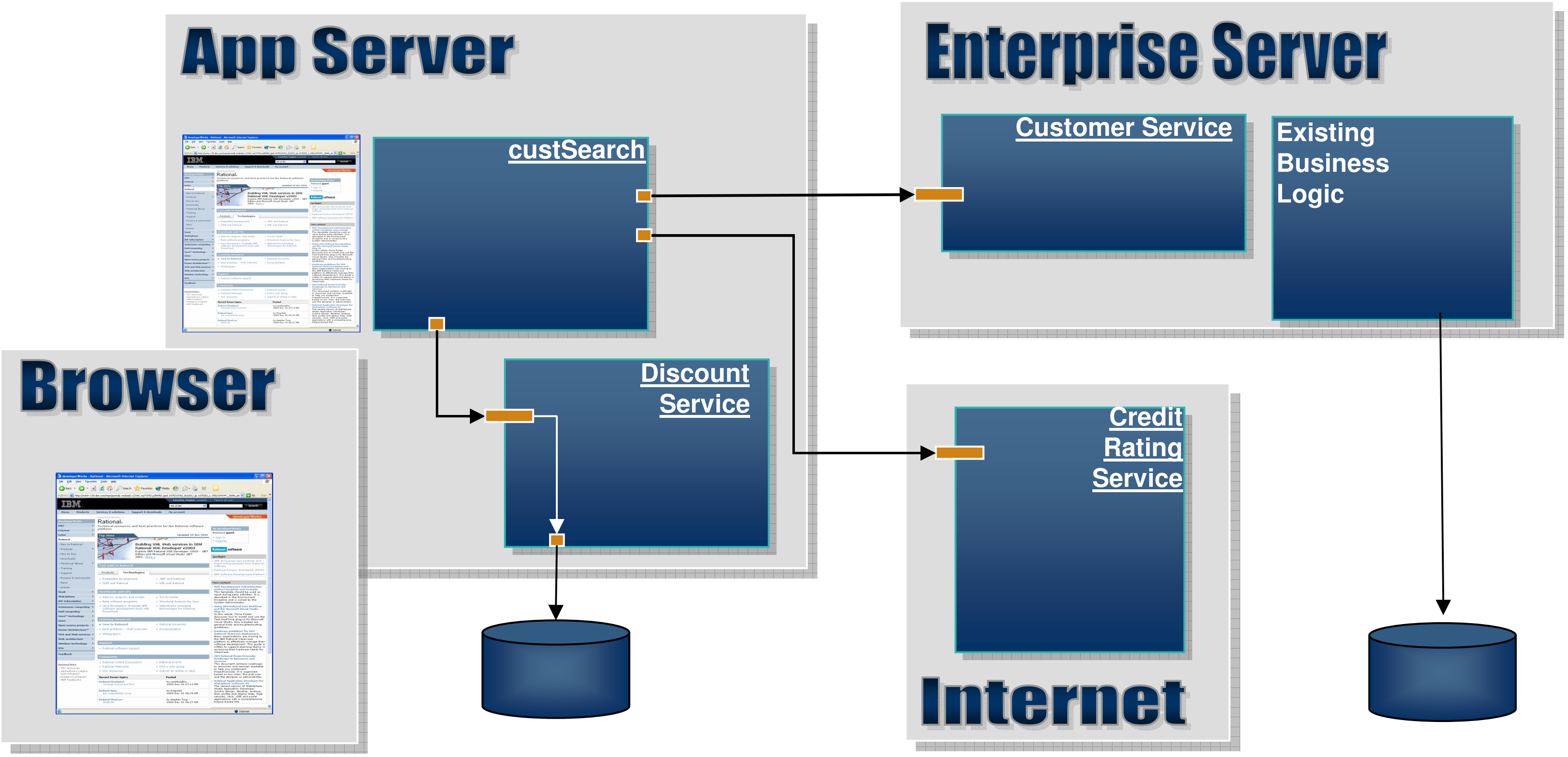
Flexible architectures to enable business agility

- Easily create services from existing code, including CICS, IMS, and terminal applications
- Define new services for all deployment platforms from initial design to implementation
- Separate service flow from service implementation to attain optimal flexibility
- Expose existing business logic out to new channels and consumers



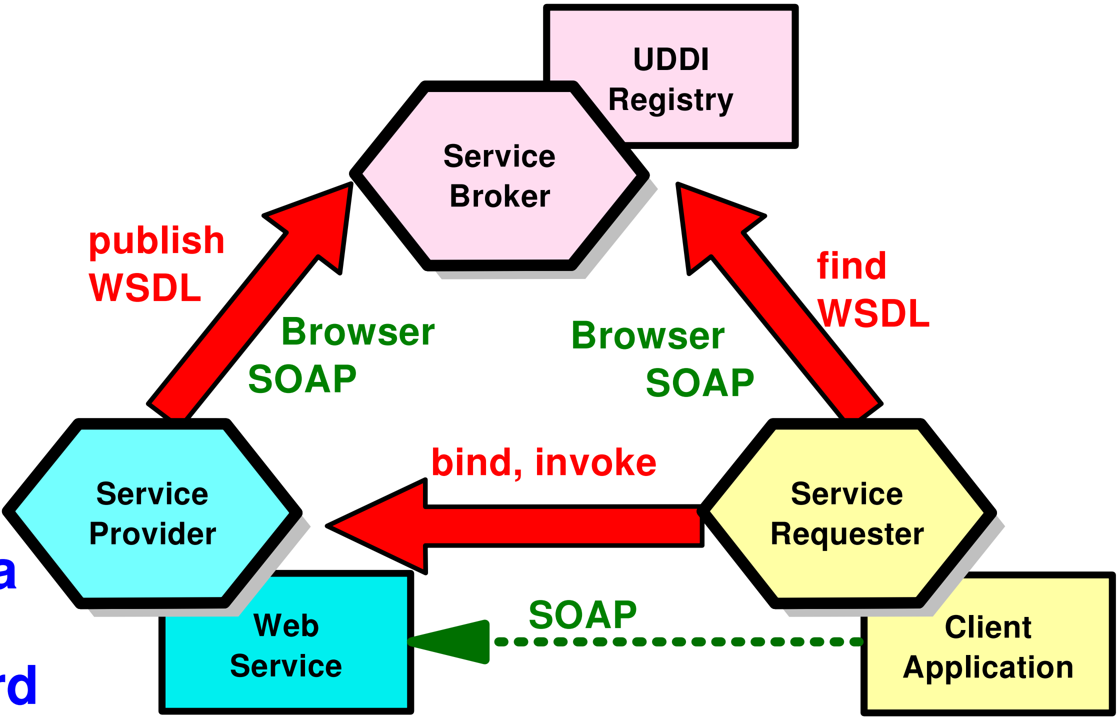
"SOAs cost 20% less to implement and saves 50% more with each reuse than traditional component-based development… the level of reuse in SOA development averages 2.5 times more than non-SOA development"
Jeffrey Poulin, PH.D. and Alan Himler, MBA, 2006, "The ROI of SOA – Based on Traditional Component Reuse"

Today’s applications



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 Interoperablity Organization)**
  - Ensure interoperability

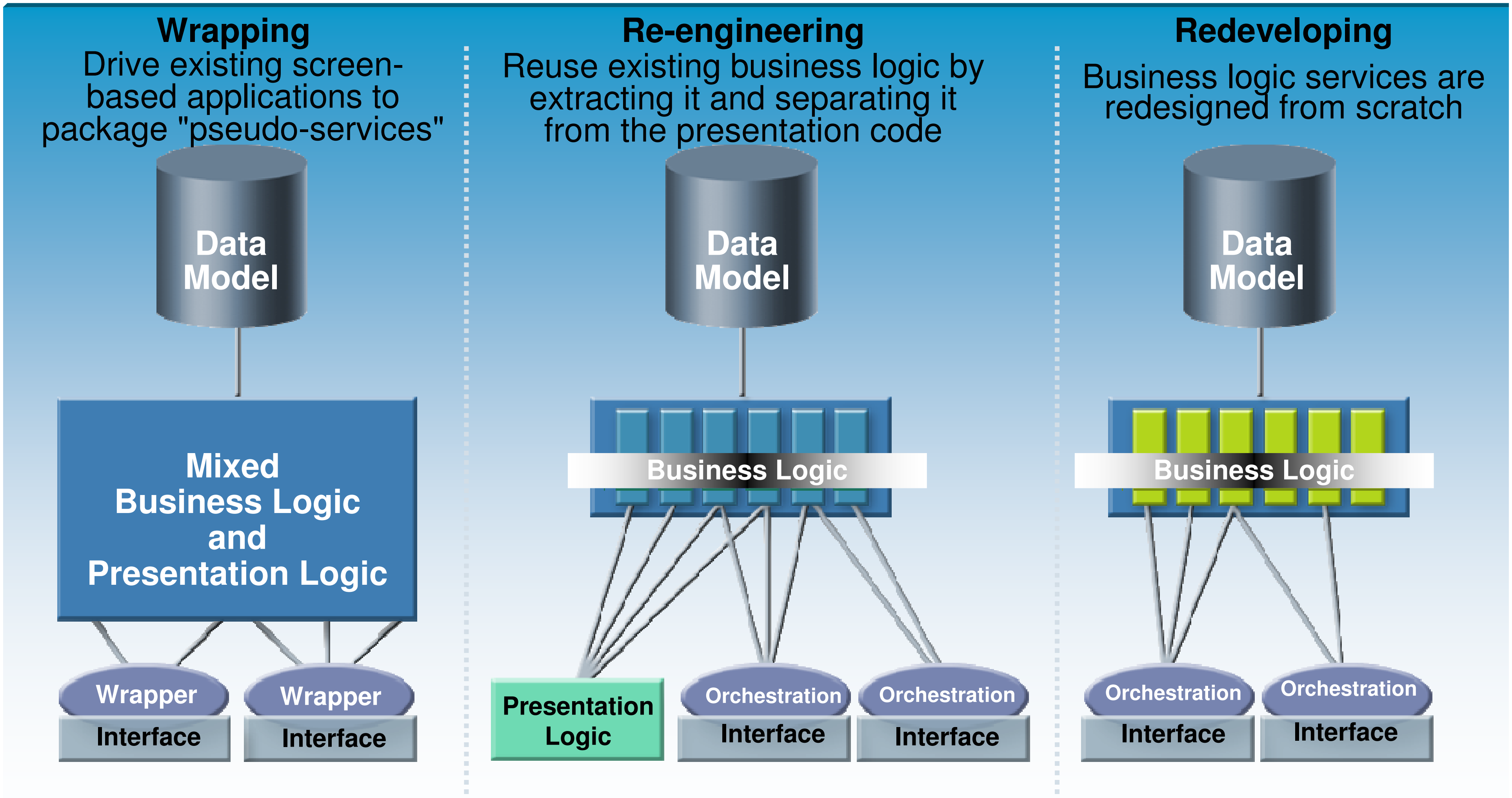


WS-I
WEB SERVICES INTEROPERABILITY ORGANIZATION

uddi.org

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

Three ways of re-using services from of existing applications


Wrapping
Drive existing screen-based applications to package "pseudo-services"
Data Model
Mixed Business Logic and Presentation Logic
Wrapper
Interface
Wrapper
Interface
Re-engineering
Reuse existing business logic by extracting it and separating it from the presentation code
Data Model
Business Logic
Presentation Logic
Orchestration
Interface
Orchestration
Interface
Redeveloping
Business logic services are redesigned from scratch
Data Model
Business Logic
Orchestration
Interface
Orchestration
Interface

Source: Gartner

Agenda

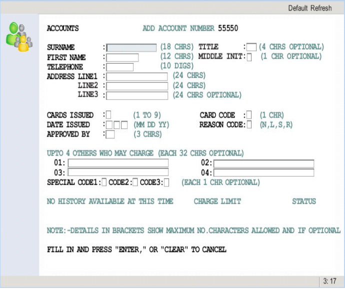
- What are web services?
  - How do they play in a mainframe environment?
- **Approaches to creating web services in the mainframe environment**
  - **Wrapping**
  - Re-engineering
  - Redeveloping
- Summary
- Q/A

Rational Host Access Transformation Services

Modernize user interfaces and create Web Services

- **Modernize and streamline “green screen” applications**
- **Combine data from multiple screens, applications and databases**
- **Non-invasive**

Rich Client



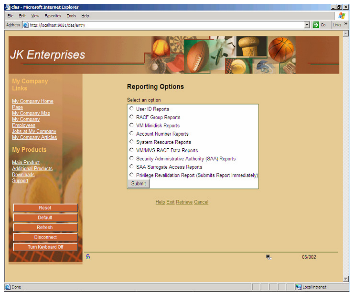
```
ACCOUNTS                ADD ACCOUNT NUMBER 55550

SURNAME       :[      ] (18 CHRS) TITLE       :[ ] (4 CHRS OPTIONAL)
FIRST NAME    :[    ]   (12 CHRS) MIDDLE INIT:[ ]  (1 CHR OPTIONAL)
TELEPHONE     :[    ]   (10 DIGS)
ADDRESS LINE1 :[        ] (24 CHRS)
        LINE2 :[        ] (24 CHRS)
        LINE3 :[        ] (24 CHRS OPTIONAL)

CARDS ISSUED  :[ ]     (1 TO 9)          CARD CODE   :[ ]  (1 CHR)
DATE ISSUED   :[ ][ ][ ] (MM DD YY)      REASON CODE:[ ]   (N,L,S,R)
APPROVED BY   :[ ]     (3 CHRS)

UPTO 4 OTHERS WHO MAY CHARGE (EACH 32 CHRS OPTIONAL)
  01:[                ]            02:[                ]
  03:[                ]            04:[                ]
SPECIAL CODE1:[ ] CODE2:[ ] CODE3:[ ]   (EACH 1 CHR OPTIONAL)

NO HISTORY AVAILABLE AT THIS TIME     CHARGE LIMIT          STATUS

NOTE:-DETAILS IN BRACKETS SHOW MAXIMUM NO.CHARACTERS ALLOWED AND IF OPTIONAL

FILL IN AND PRESS "ENTER," OR "CLEAR" TO CANCEL
```

- **Integration at the desktop with other Eclipse applications**
- **Client side processing**
- **Rich set of user interface widgets**
- **Built on the standard, open Eclipse foundation**
- **3270e print directly to end user’s printer**

Example: CICS App

3270 or 5250 Data stream

Web



Reporting Options

Select an option

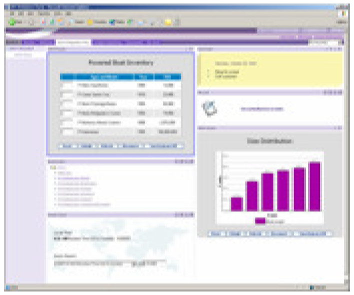
○ User ID Reports
○ RACF Group Reports
○ VM Minidisk Reports
○ Account Number Reports
○ System Resource Reports
○ VM/MVS RACF Data Reports
○ Security Administrative Authority (SAA) Reports
○ SAA Surrogate Access Reports
○ Privilege Revalidation Report (Submits Report Immediately)

Submit

Help Exit Retrieve Cancel

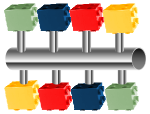
- **Zero footprint**
- **View through your favorite browser**

Portal



- **Integration at the glass**
- **Click-to-Action support**

Web Service



- **Build self-service transactions**

Host Access Transformation Services Values

Rich Client

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

Browser

- Zero footprint
- Pure HTML
- Access through your favorite browser, including Internet Explorer and Firefox.

Portal

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

Mobile

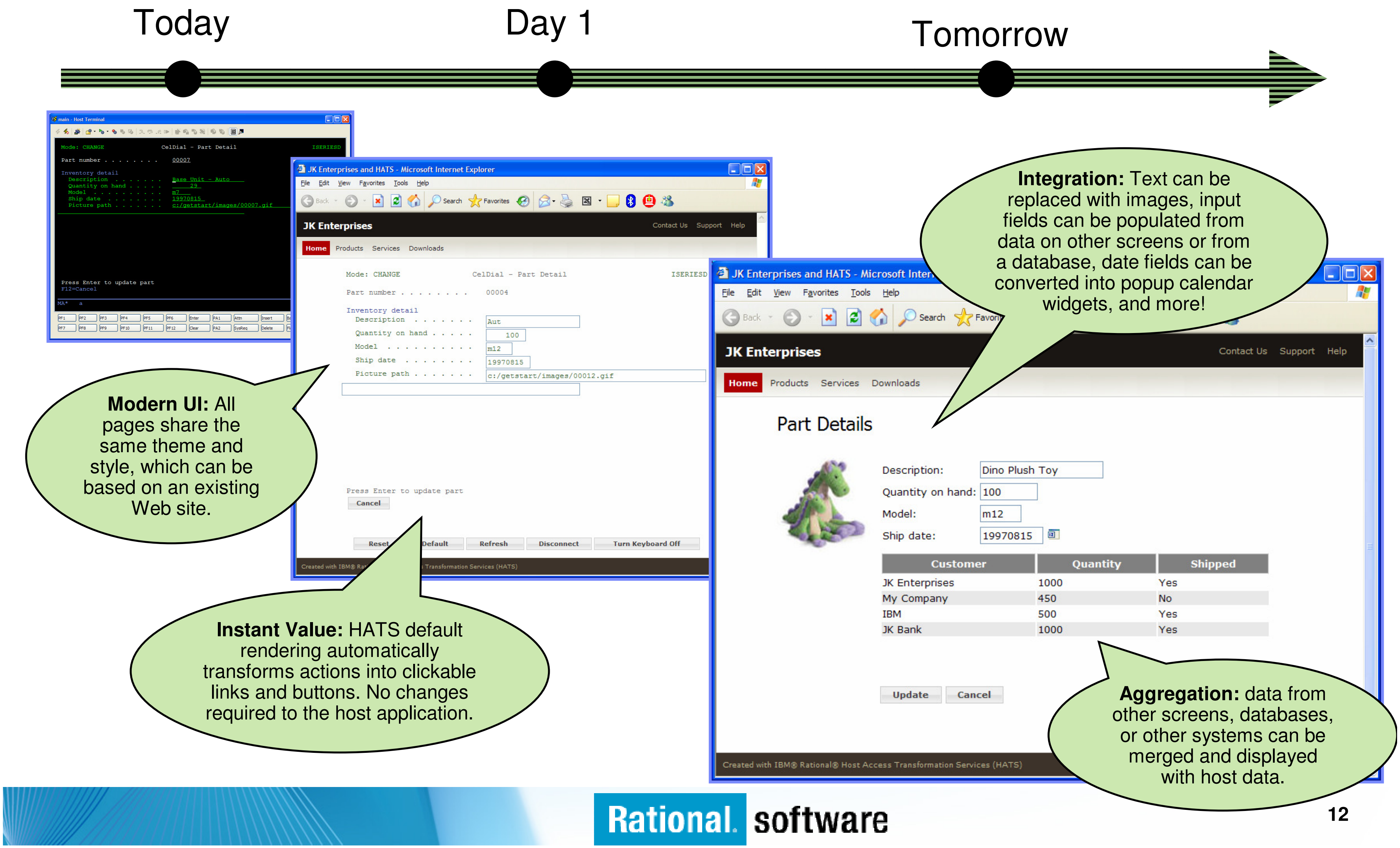
- Access host applications from mobile devices

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*

HATS - Modernization At Your Own Pace



main - Host Terminal

```
Mode: CHANGE                CelDial - Part Detail                ISERIESD

Part number . . . . . . . .    00007

Inventory detail
  Description  . . . . . . .   Base Unit - Auto
  Quantity on hand . . . . .         29
  Model  . . . . . . . . . .   m7
  Ship date  . . . . . . . .   19970815
  Picture path . . . . . . .   c:/getstart/images/00007.gif


Press Enter to update part
F12=Cancel

MA*   a
```

PF1 PF2 PF3 PF4 PF5 PF6 Enter PA1 Attn Insert
PF7 PF8 PF9 PF10 PF11 PF12 Clear PA2 SysReq Delete

JK Enterprises and HATS - Microsoft Internet Explorer

File Edit View Favorites Tools Help
Back Search Favorites
JK Enterprises
Contact Us Support Help
Home Products Services Downloads

```
Mode: CHANGE                CelDial - Part Detail                ISERIESD

Part number . . . . . . . .    00004

Inventory detail
  Description  . . . . . . .   Aut
  Quantity on hand . . . . .   100
  Model  . . . . . . . . . .   m12
  Ship date  . . . . . . . .   19970815
  Picture path . . . . . . .   c:/getstart/images/00012.gif

Press Enter to update part
```

Cancel
Reset Default Refresh Disconnect Turn Keyboard Off
Created with IBM® Ra… Transformation Services (HATS)

JK Enterprises and HATS - Microsoft Inter

File Edit View Favorites Tools Help
Back Search Favori
JK Enterprises
Contact Us Support Help
Home Products Services Downloads

Part Details

Description: Dino Plush Toy

Quantity on hand: 100

Model: m12

Ship date: 19970815

Customer	Quantity	Shipped
JK Enterprises	1000	Yes
My Company	450	No
IBM	500	Yes
JK Bank	1000	Yes

Update
Cancel
Created with IBM® Rational® Host Access Transformation Services (HATS)

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!

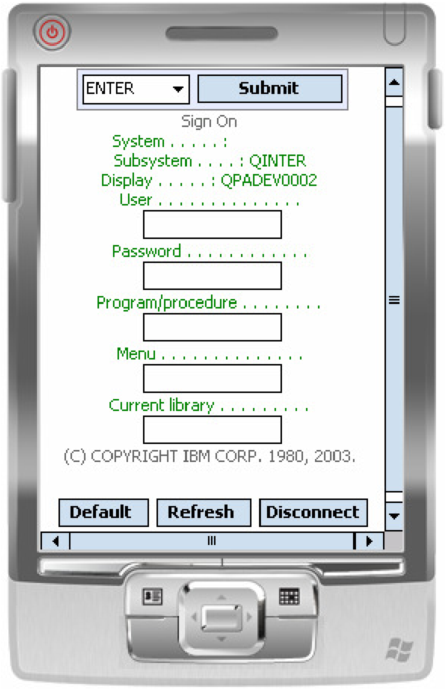
Modern UI: All pages share the same theme and style, which can be based on an existing Web site.

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

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

What is HATS mobile device support for v7.1?

- **HATS transformation applications can now be accessed by pervasive devices such as cellular phones, data collection terminals, and Personal Digital Assistants**
- **This allows for the HATS developer to:**
  - Customize and transform individual screens
  - Add business logic to screen interactions
  - Enhance the presentation, layout, and flow of the original host systems
- **Supported browser: Internet Explorer Mobile 5.0 or later**
- **Reminder: HATS Integration Object or Web service data can also be used in conjunction with mobile devices**



Sign On

System . . . . . :

Subsystem . . . . : QINTER

Display . . . . . : QPADEV0002

User . . . .

Password . . . .

Program/procedure . . . .

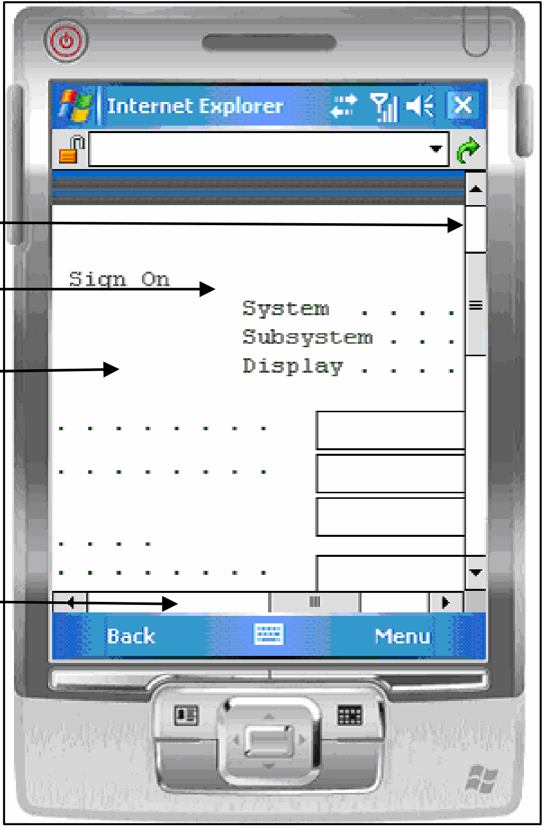
Menu . . . .

Current library . . . .

(C) COPYRIGHT IBM CORP. 1980, 2003.

HATS Compact Default Rendering Example

Normal default rendering



Sign On

System . . . . .

Subsystem . . . .

Display . . . . .

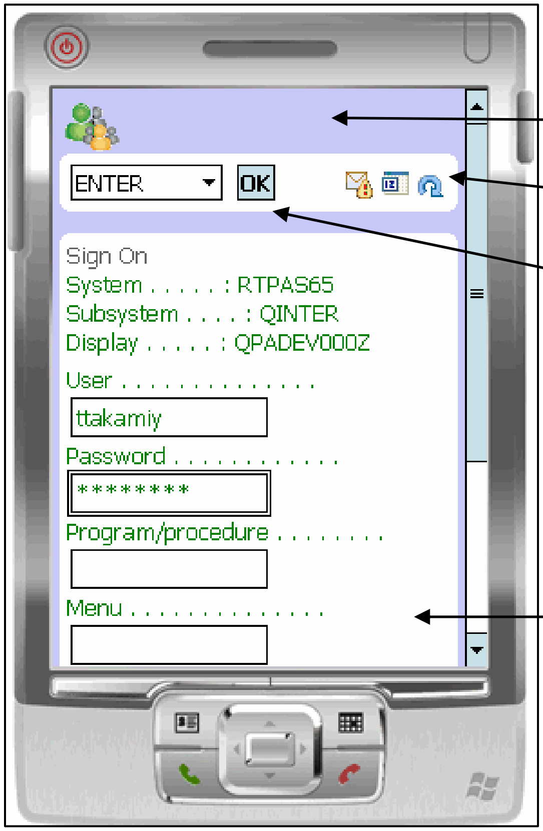
Larger scroll area

Oversized
Default rendering

Blank template

Vertical scrolling

Compact Default Rendering



Sign On

System . . . . . : RTPAS65

Subsystem . . . . : QINTER

Display . . . . . : QPADEV000Z

User . . . .

ttakamiy

Password . . . .

****

Program/procedure . . . .

Menu . . . .

New template

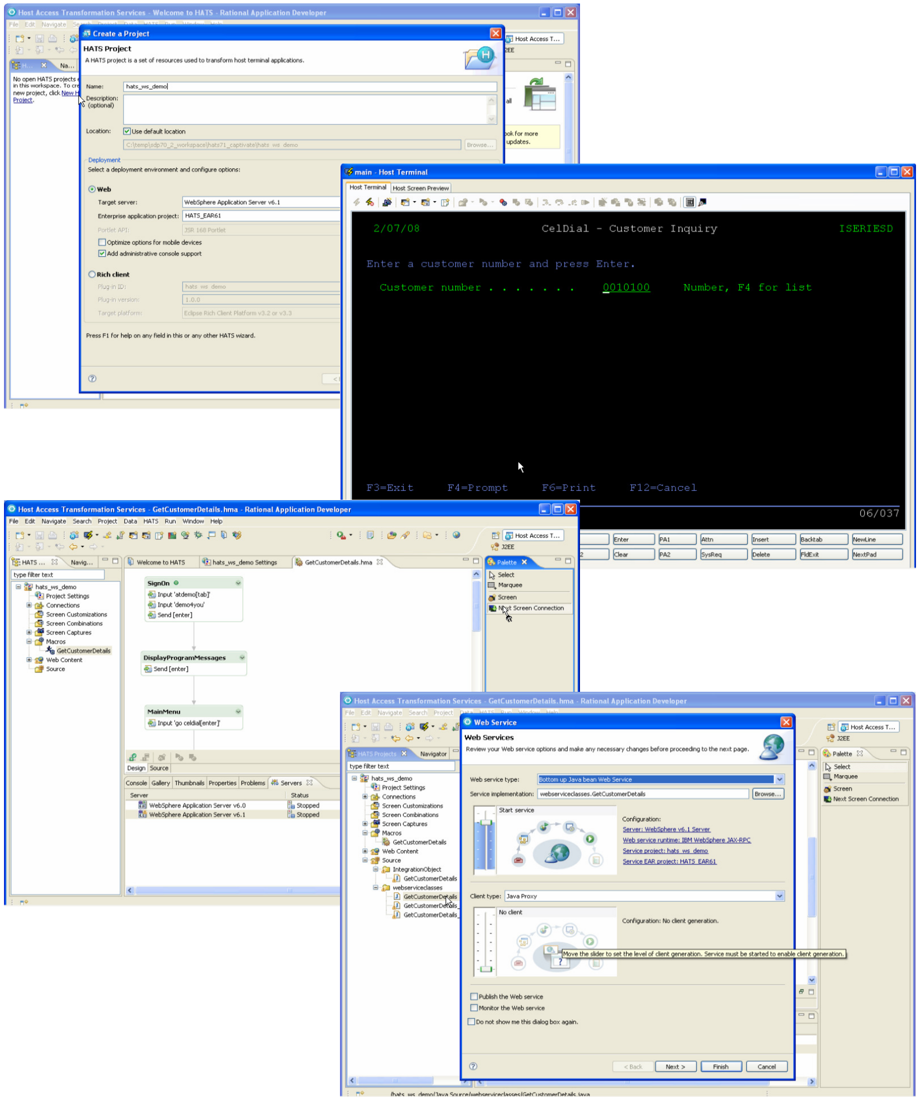
Icon application keypad

optimized default settings

Compact
default rendering

HATS Developer Tasks

1. Create a HATS Web project using the HATS toolkit.
2. Create a macro to navigate the screens, supply input, and extract output.
3. Use the HATS toolkit to generate a Java bean wrapper for the macro.
4. Use the Rational tooling to generate a Web service (and WSDL file) to drive the Java bean.
5. Deploy to WebSphere Application Server.



No open HATS projects in this workspace. To create a new project, click New HATS Project.

HATS Project

A HATS project is a set of resources used to transform host terminal applications.

Name: hats_ws_demo

Description: (optional)

Location: ☑ Use default location

C:\temp\sdp70_2_workspace\hats71_captivate\hats_ws_demo

Deployment

Select a deployment environment and configure options:

● Web

Target server: WebSphere Application Server v6.1

Enterprise application project: HATS_EAR61

Portlet API: JSR 168 Portlet

□ Optimize options for mobile devices

☑ Add administrative console support

○ Rich client

Plug-in ID: hats_ws_demo

Plug-in version: 1.0.0

Target platform: Eclipse Rich Client Platform v3.2 or v3.3

Press F1 for help on any field in this or any other HATS wizard.

```
 2/07/08              CelDial - Customer Inquiry                 ISERIESD

 Enter a customer number and press Enter.

  Customer number . . . . . . .    0010100     Number, F4 for list




 F3=Exit      F4=Prompt      F6=Print      F12=Cancel
                                                                  06/037
```

SignOn
- Input 'atdemo[tab]'
- Input 'demo4you'
- Send [enter]

DisplayProgramMessages
- Send [enter]

MainMenu
- Input 'go celdial[enter]'

Server	Status
WebSphere Application Server v6.0	Stopped
WebSphere Application Server v6.1	Stopped

Web Services

Review your Web service options and make any necessary changes before proceeding to the next page.

Web service type: Bottom up Java bean Web Service

Service implementation: webserviceclasses.GetCustomerDetails

Start service

Configuration:
Server: WebSphere v6.1 Server
Web service runtime: IBM WebSphere JAX-RPC
Service project: hats_ws_demo
Service EAR project: HATS_EAR61

Client type: Java Proxy

No client

Configuration: No client generation.

Move the slider to set the level of client generation. Service must be started to enable client generation.

□ Publish the Web service

□ Monitor the Web service

□ Do not show me this dialog box again.

/hats_ws_demo/Java Source/webserviceclasses/GetCustomerDetails.java

Agenda

- What are web services?
  - How do they play in a mainframe environment?
- **Approaches to creating web services in the mainframe environment**
  - Wrapping
  - **Re-engineering**
  - Redeveloping
- Summary
- Q/A

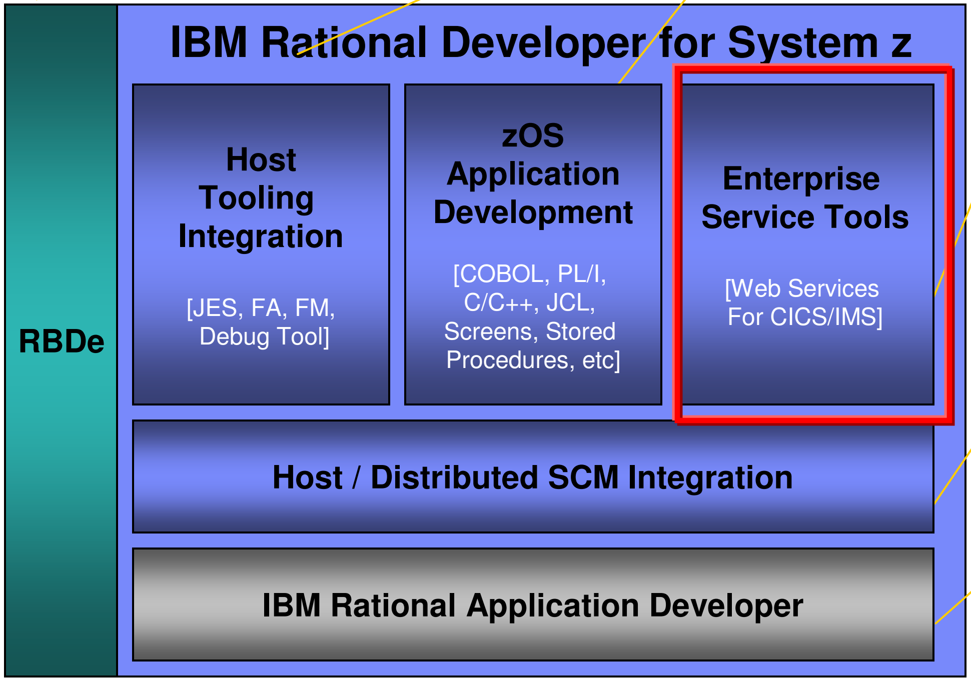
IBM Rational Developer for System z

JES and PD Tools

•Read/Write/Update VSAM datasets via integration with IBM File Manager
•Access IBM Fault analyzer reports for analyzing ABENDS and associating back to source code
•Interact with the Job Entry Subsystem (JES) to submit jobs, monitor jobs, and review job output
•Debug zOS applications from with workstation as they execute live in the remote runtime

Integration with EGL using RBDe

- Quick and easy development of modern enterprise applications for procedural programmers
- Simplify and speed up creation of Web applications and services without having to learn Java or J2EE



RBDe

IBM Rational Developer for System z

Host Tooling Integration

[JES, FA, FM, Debug Tool]

zOS Application Development

[COBOL, PL/I, C/C++, JCL, Screens, Stored Procedures, etc]

Enterprise Service Tools

[Web Services For CICS/IMS]

Host / Distributed SCM Integration

IBM Rational Application Developer

Traditional Development

Development Environment

- Connect to z/OS systems
- Work with z/OS resources like COBOL, PL/1 , C, C++, JCL, etc.
- Perform dataset management actions like allocating datasets and migrating datasets
- Perform typical edit, compile, and debug tasks on remote z/OS resources from the workstation
- Create, build, and catalog DB2 stored procedures on zOS
- Compile and test programs locally to ensure correctness

Screen design

- Visually create, modify, build, and deploy BMS maps sets or MFS/IMS maps remotely or on the local workstation

Code Generation

- Generate CRUD DB2 program code from UML, which can also be easily integrated into web service applications

zOS Web Service and Flow Creation

- Implements SOA and Web Services
- SOA access to CICS V3.2 and IMS V9 COBOL applications
- Bottom-up/Top-down or meet-in-the-middle COBOL to XML mapping support
- Integrated COBOL XML converters, XML schemas, and WSDL generation
- Service Flow Modeler to build/deploy service flows out of your existing Commarea, Channel, MQ, and Terminal CICS applications.

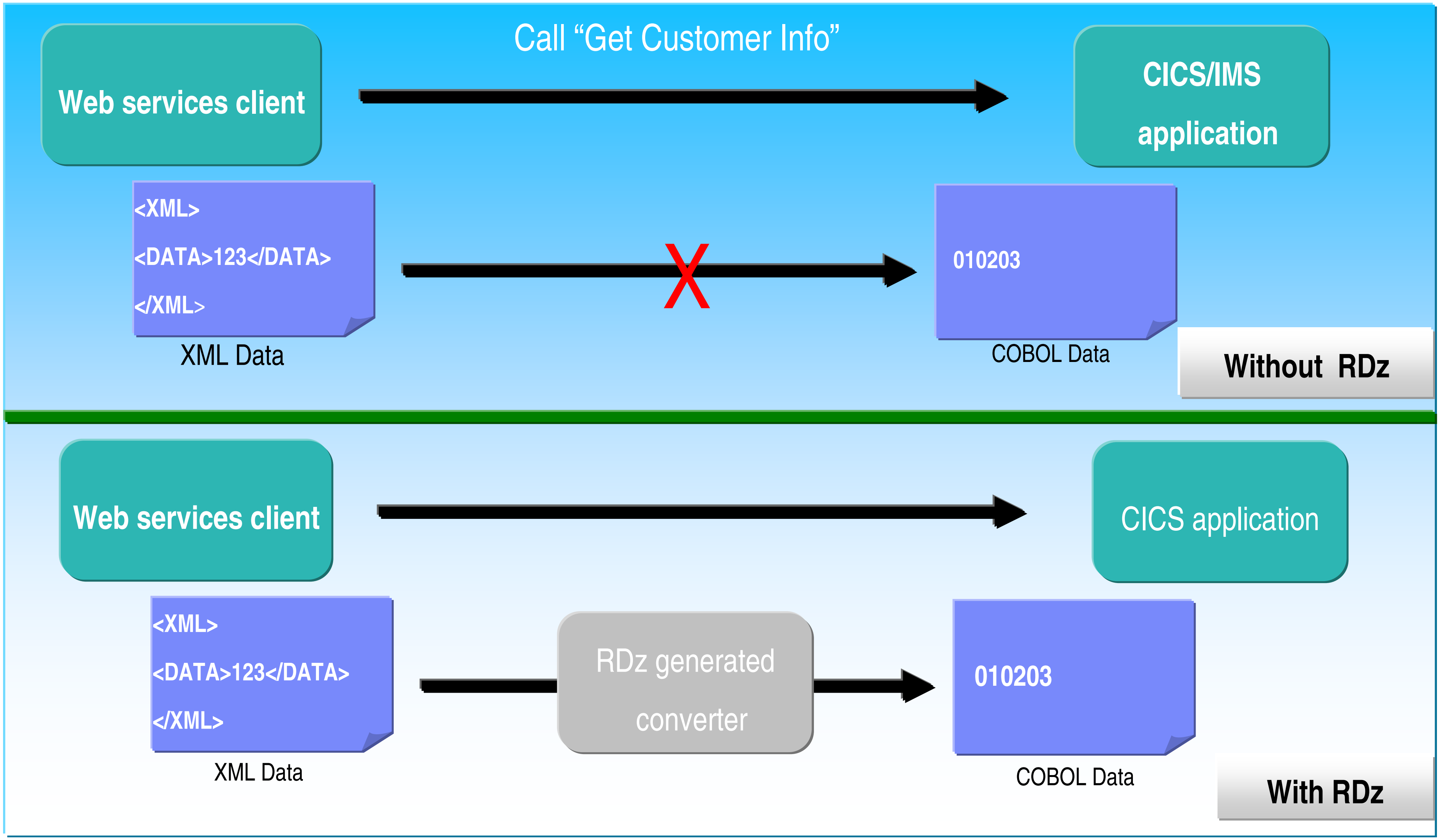
SCM Support

- Access to host SCMs such as SCLM
- Framework for writing/deploying custom SCM integration code
- Support for storing zOS resources in distributed SCMs such as ClearCase

Web and JEE Development

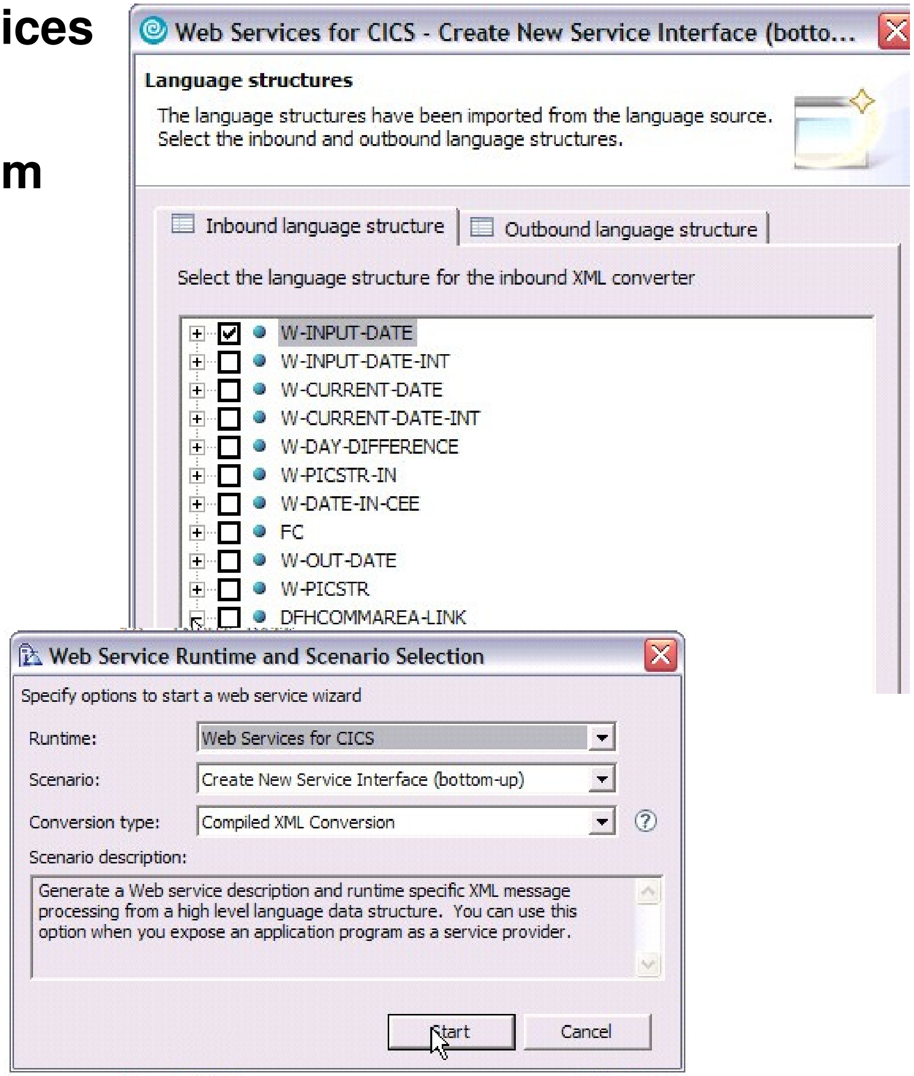
- Create Web Pages / JSF / Struts
- JEE/Java Development
- JCA Connectors
- Distributed debugger
- Web Services and Test environment

We need interfaces to talk “XML” ....



Create web services for zOS runtimes

- **Build, Deploy, and Test Web services from existing applications**
- **Create source code skeletons from web service definitions**
- **Map web service definitions to existing application modules**
- **Supports traditional languages**
  - COBOL
  - PL/I
- **Supports zOS specific runtimes**
  - CICS
  - IMS
  - Batch



Web Services for CICS - Create New Service Interface (botto...

Language structures

The language structures have been imported from the language source.
Select the inbound and outbound language structures.

Select the language structure for the inbound XML converter

- ☑ W-INPUT-DATE
- □ W-INPUT-DATE-INT
- □ W-CURRENT-DATE
- □ W-CURRENT-DATE-INT
- □ W-DAY-DIFFERENCE
- □ W-PICSTR-IN
- □ W-DATE-IN-CEE
- □ FC
- □ W-OUT-DATE
- □ W-PICSTR
- □ DFHCOMMAREA-LINK

Web Service Runtime and Scenario Selection

Specify options to start a web service wizard

Runtime: Web Services for CICS

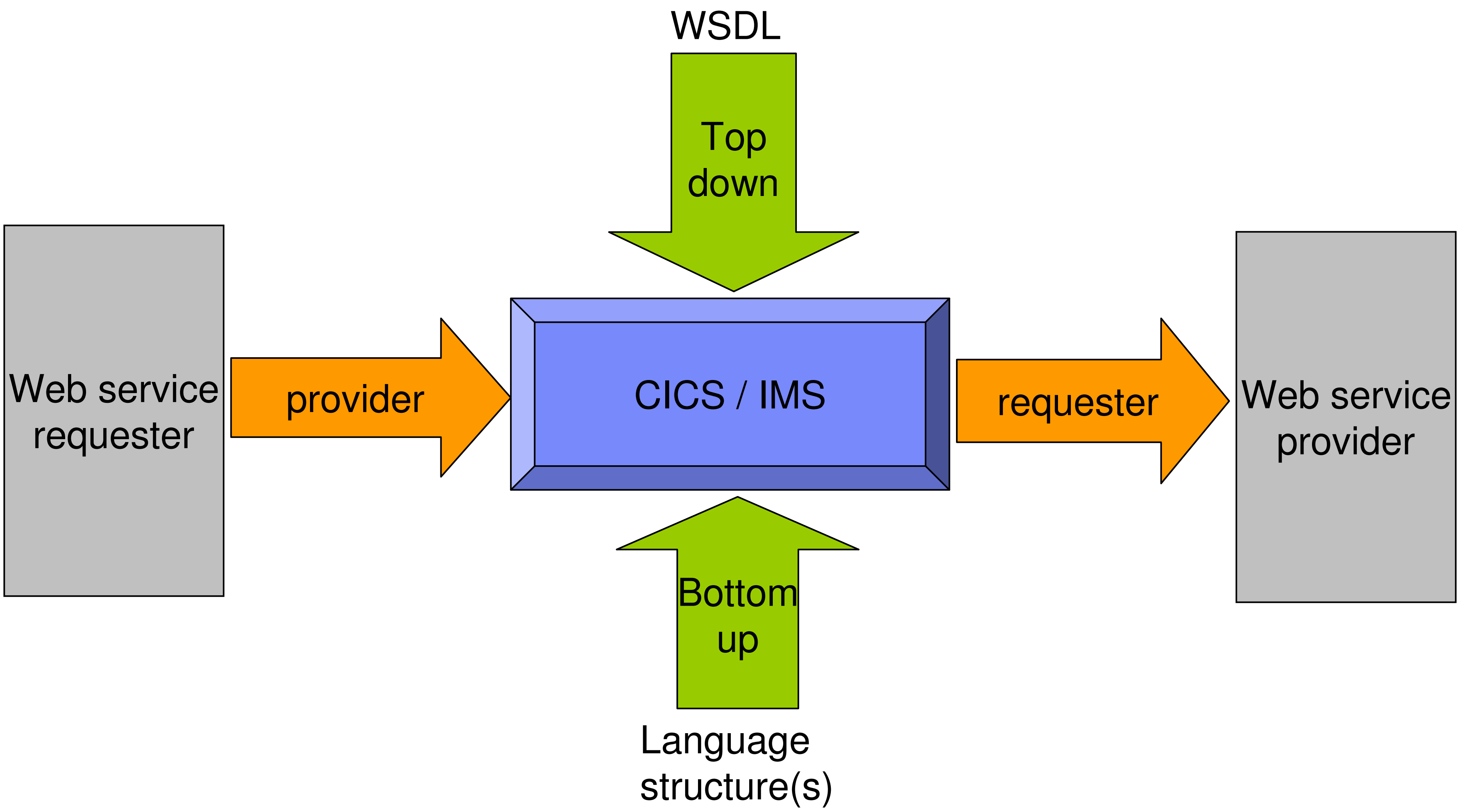
Scenario: Create New Service Interface (bottom-up)

Conversion type: Compiled XML Conversion

Scenario description:

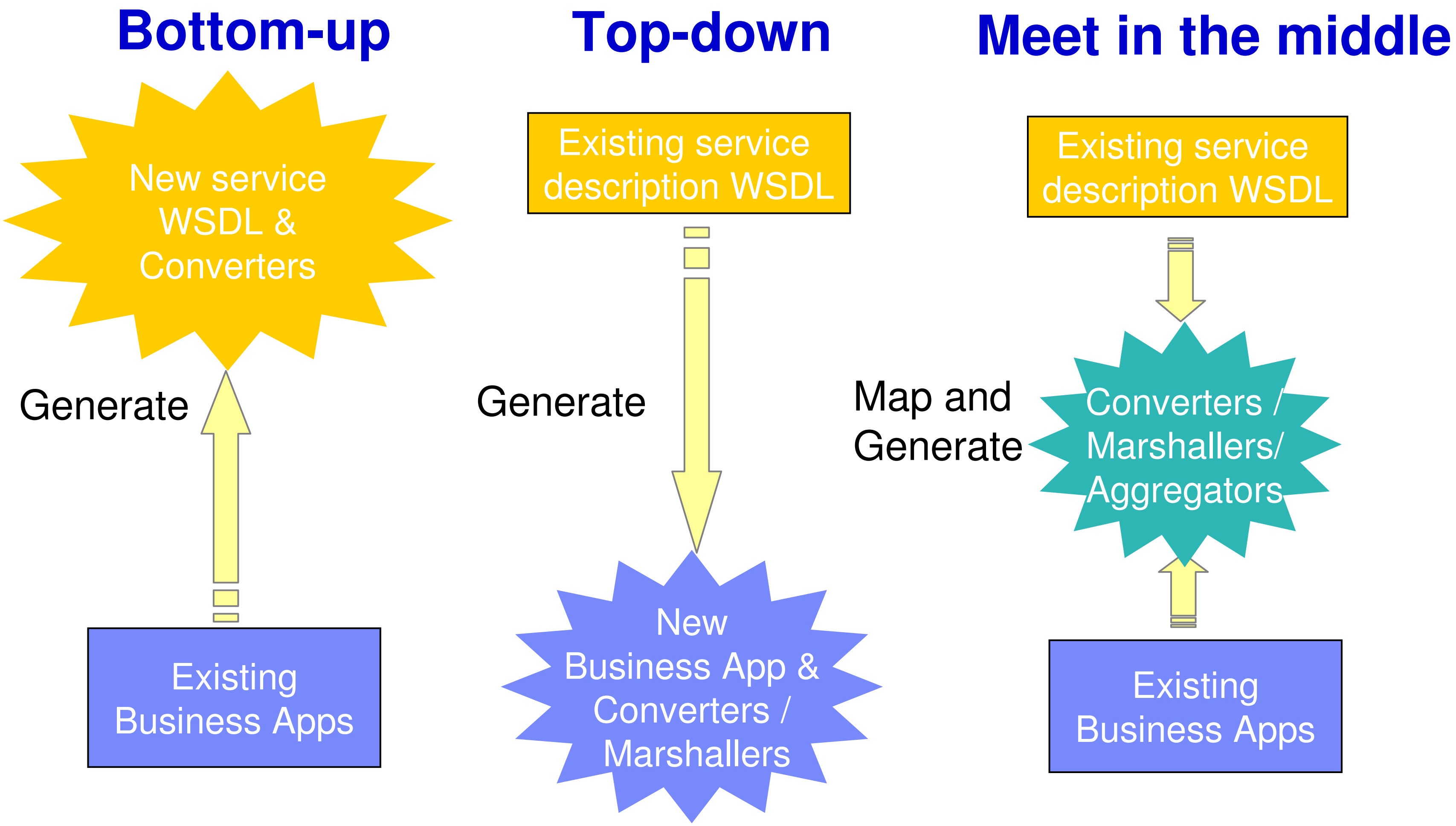
Generate a Web service description and runtime specific XML message processing from a high level language data structure. You can use this option when you expose an application program as a service provider.

Web Services Enablement Styles



Enterprise Service Tools

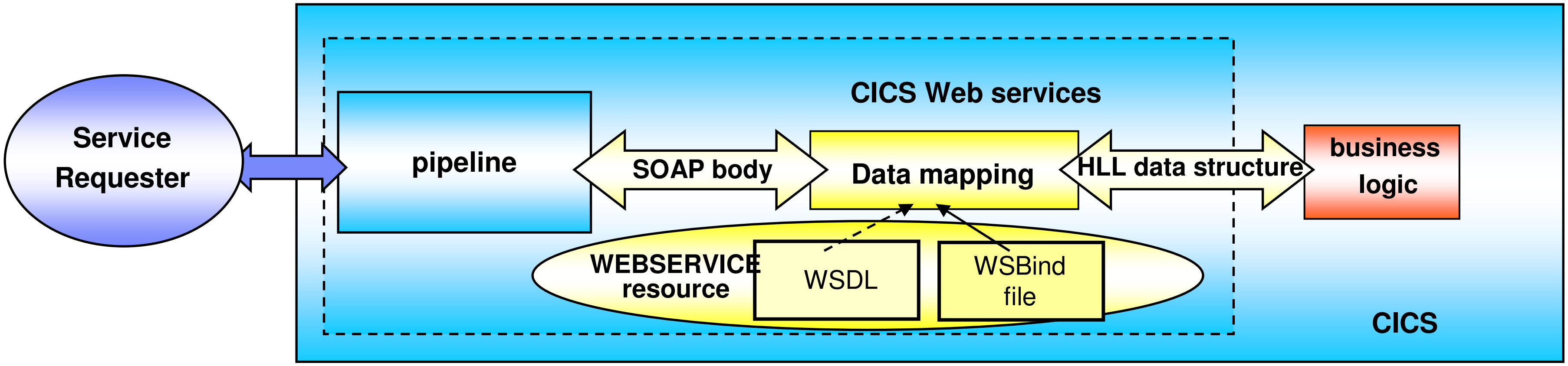
Web Service Enablement Styles



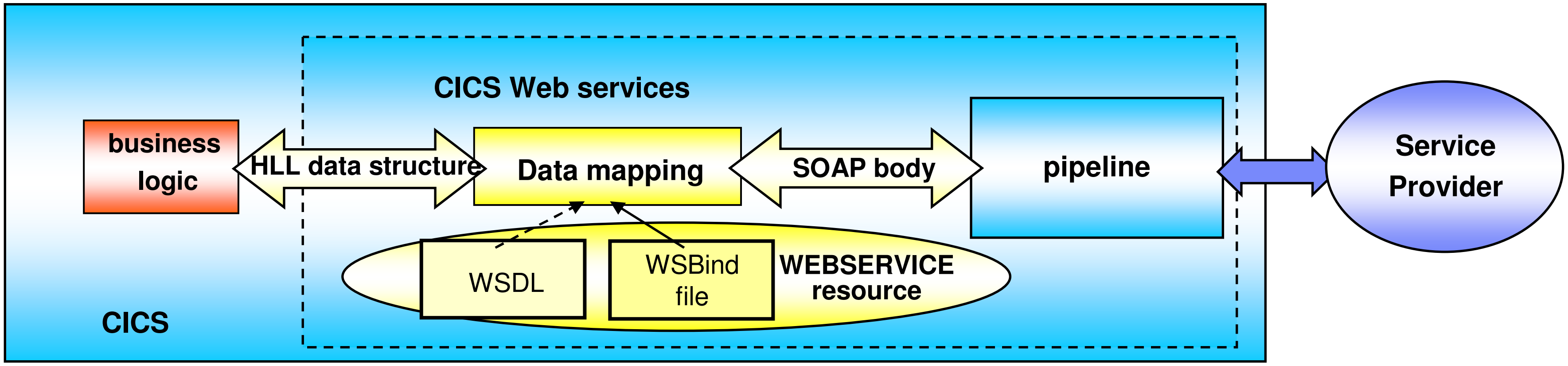
Bottom-up
New service WSDL & Converters
Generate
Existing Business Apps
Top-down
Existing service description WSDL
Generate
New Business App & Converters / Marshallers
Meet in the middle
Existing service description WSDL
Map and Generate
Converters / Marshallers/ Aggregators
Existing Business Apps

CICS Web Services

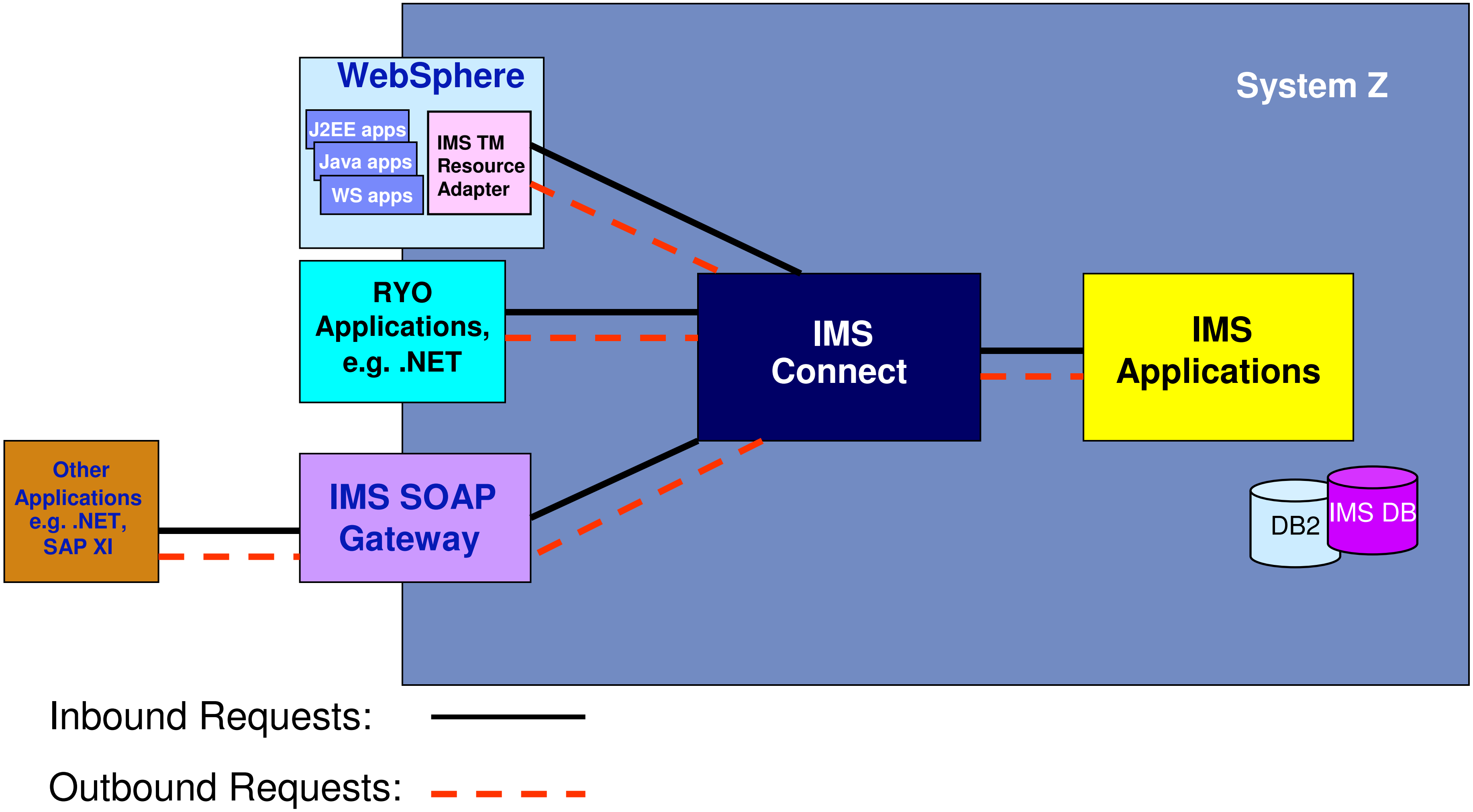
- CICS as a service provider



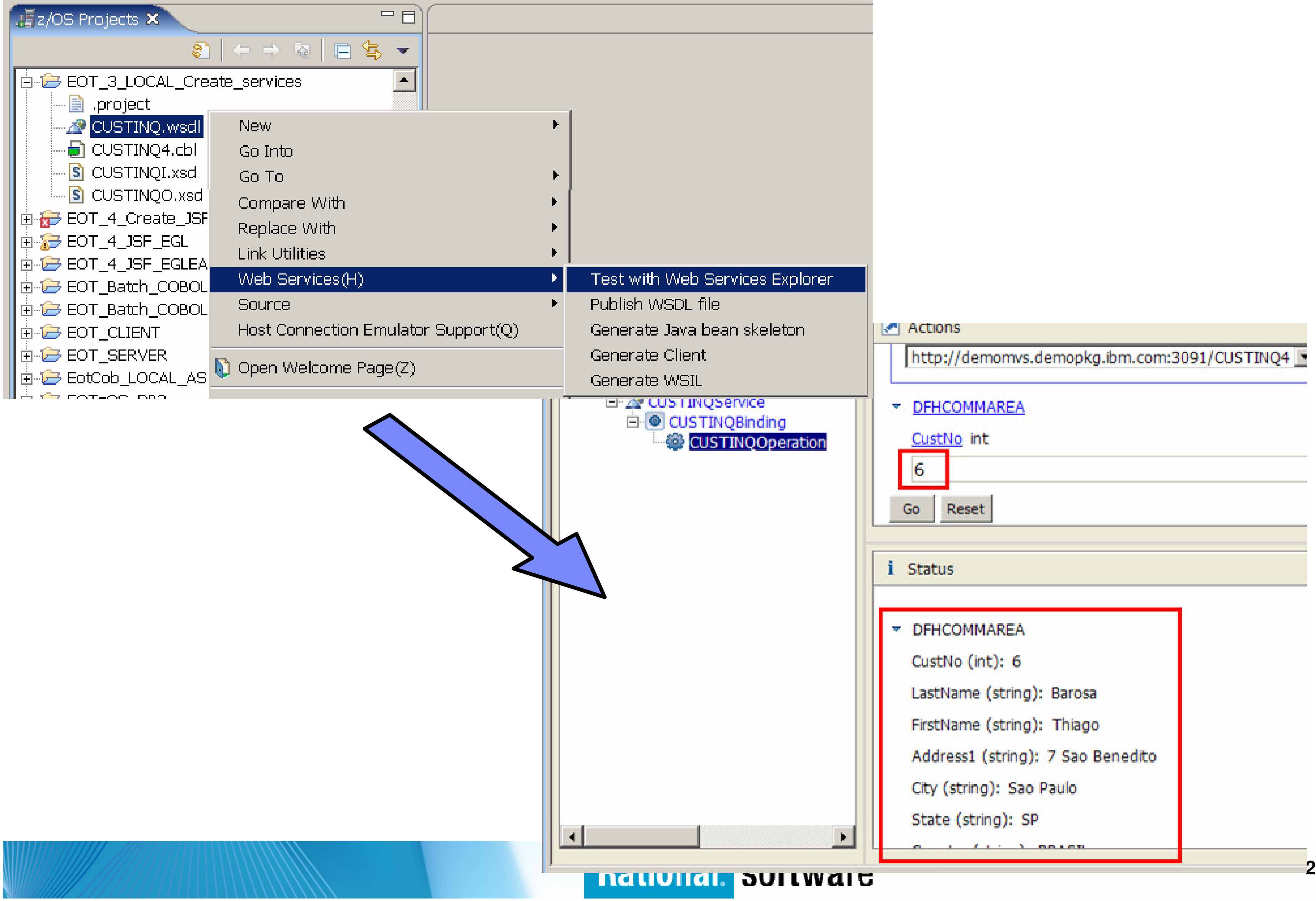
- CICS as a service requester



IMS SOA Solutions – Service Provider and Service Consumer



Example: Testing using RDz



http://demomvs.demopkg.ibm.com:3091/CUSTINQ4

CustNo int

6

DFHCOMMAREA

CustNo (int): 6

LastName (string): Barosa

FirstName (string): Thiago

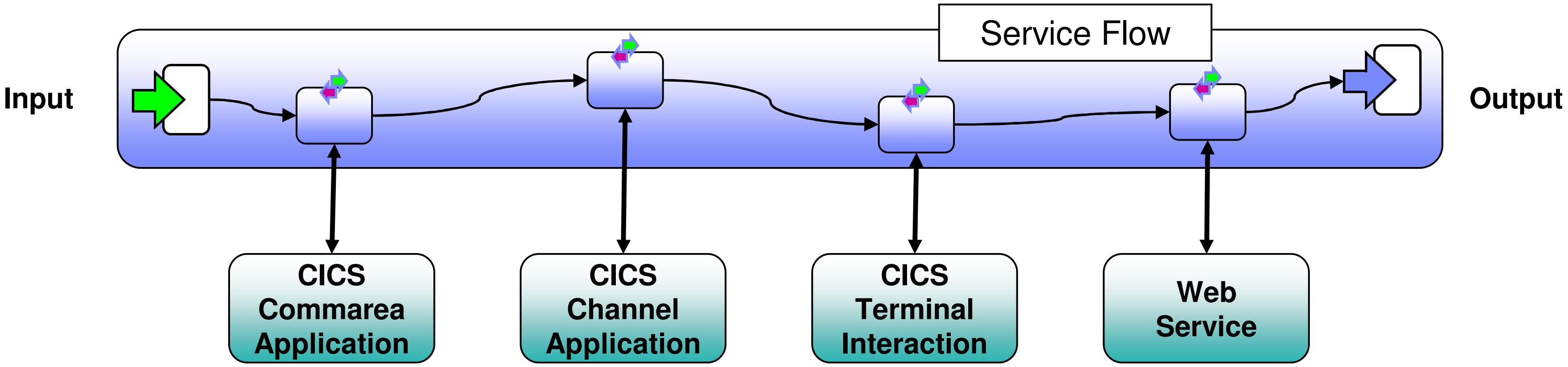
Address1 (string): 7 Sao Benedito

City (string): Sao Paulo

State (string): SP

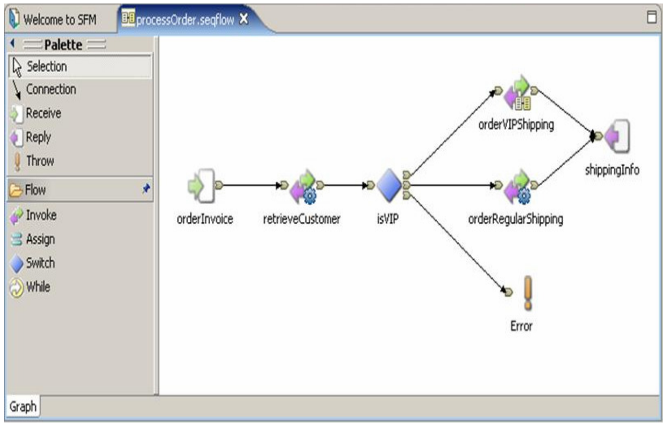
What is the Service Flow Feature?

- **CICS Service Flow Feature provides the capability to aggregate existing CICS applications into composed business services which may be integrated into an SOA environment**
  - Aggregate multiple calls to CICS applications into one business level service call
  - Automate the interaction with 3270 terminal based applications and expose as a business level service

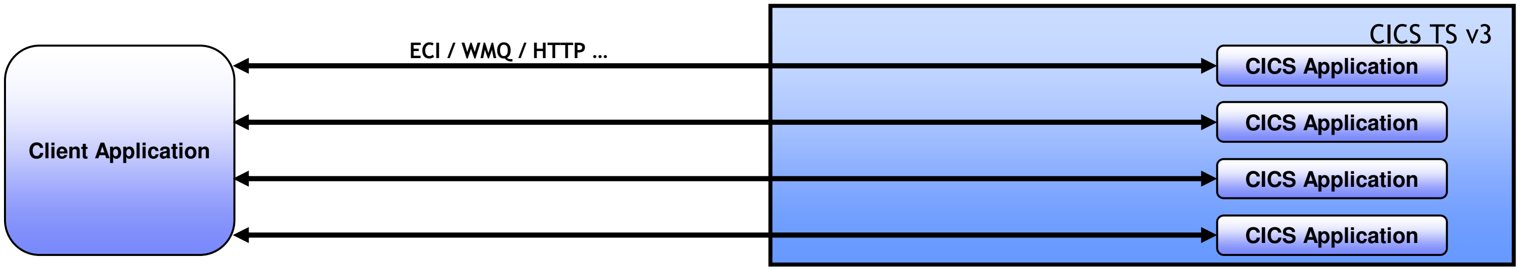


Orchestrate CICS services and screens

- **Model, Deploy, and Test Service Flows using Service Flow Modeler**
  - Aggregates multiple CICS transactions into high-level business processes through visual modeling
  - Supports CICS BMS (terminal-based) applications & CICS commarea/container/channel applications
  - Highly optimized CICS runtime supporting Web services and XML interfaces



Without Service Flow Modeler



Client Application

ECI / WMQ / HTTP ...

CICS TS v3

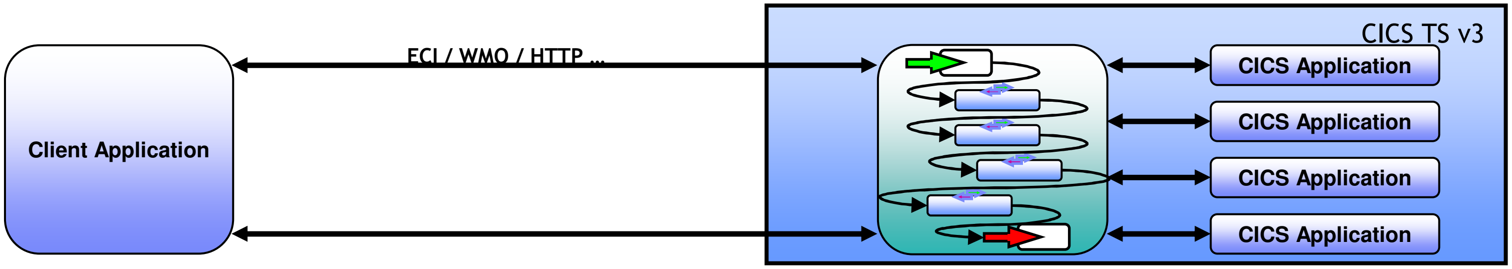
CICS Application

CICS Application

CICS Application

CICS Application

With Service Flow Modeler



Client Application

ECI / WMQ / HTTP ...

CICS TS v3

CICS Application

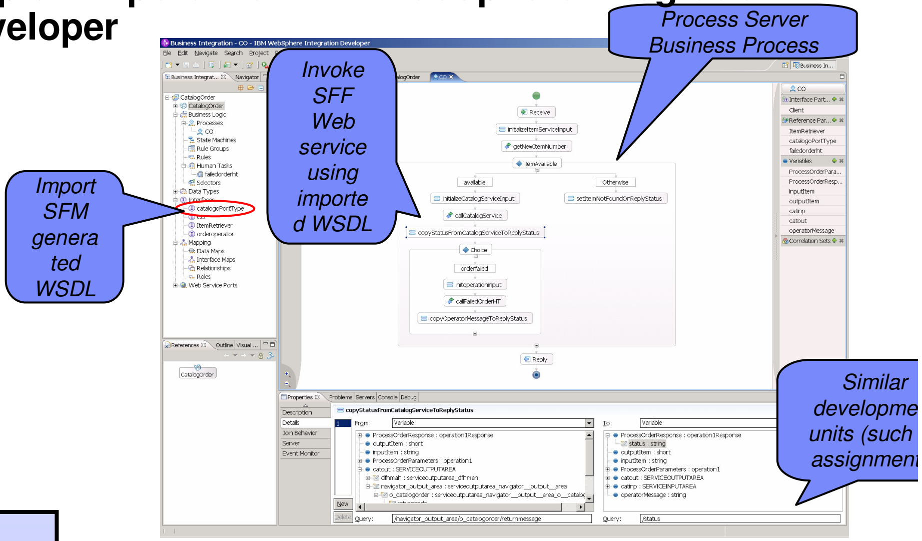
CICS Application

CICS Application

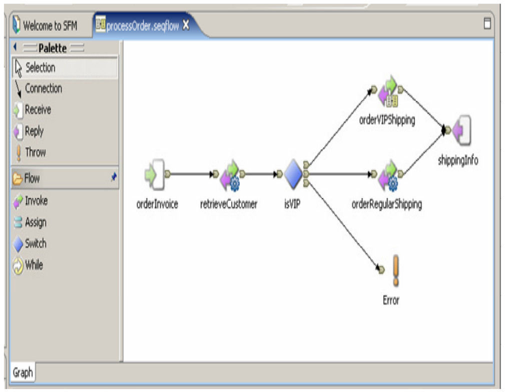
CICS Application

CICS Service Flow Feature and IBM SOA

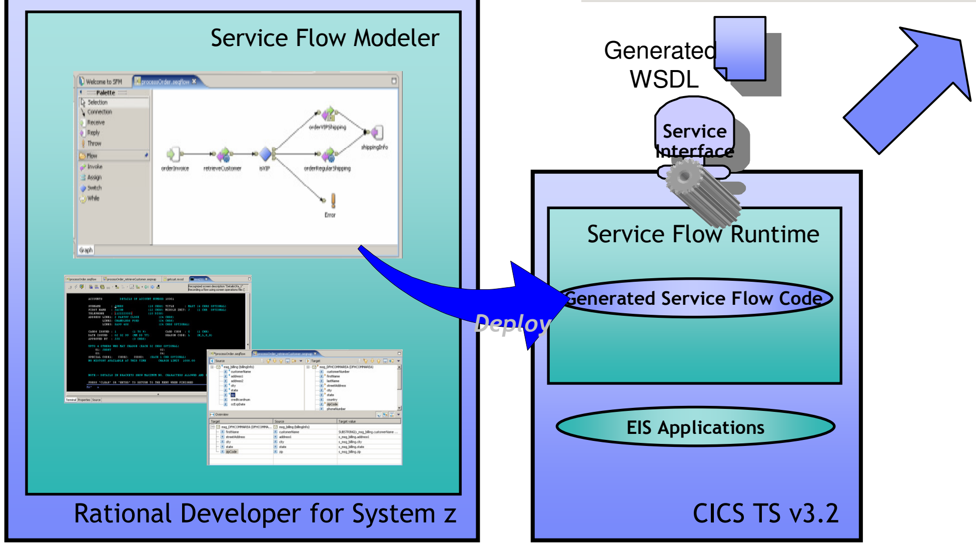
Step 3 - Import WSDL in WebSphere Integration Developer


Process Server Business Process
Invoke SFF Web service using importe d WSDL
Import SFM genera ted WSDL
Similar developme units (such assignmen
Business Integration - CO - IBM WebSphere Integration Developer
CatalogOrder
catalogPortType
Receive
initializeItemServiceInput
getNewItemNumber
itemAvailable
available
Otherwise
initializeCatalogServiceInput
callCatalogService
setItemNotFoundOnReplyStatus
copyStatusFromCatalogServiceToReplyStatus
Choice
orderFailed
initoperationInput
callFailedOrderHT
copyOperatorMessageToReplyStatus
Reply
Properties
copyStatusFromCatalogServiceToReplyStatus

Step 1 - Model Service Flow with SFM



Step 2 - Deploy as Web Service to CICS SFR

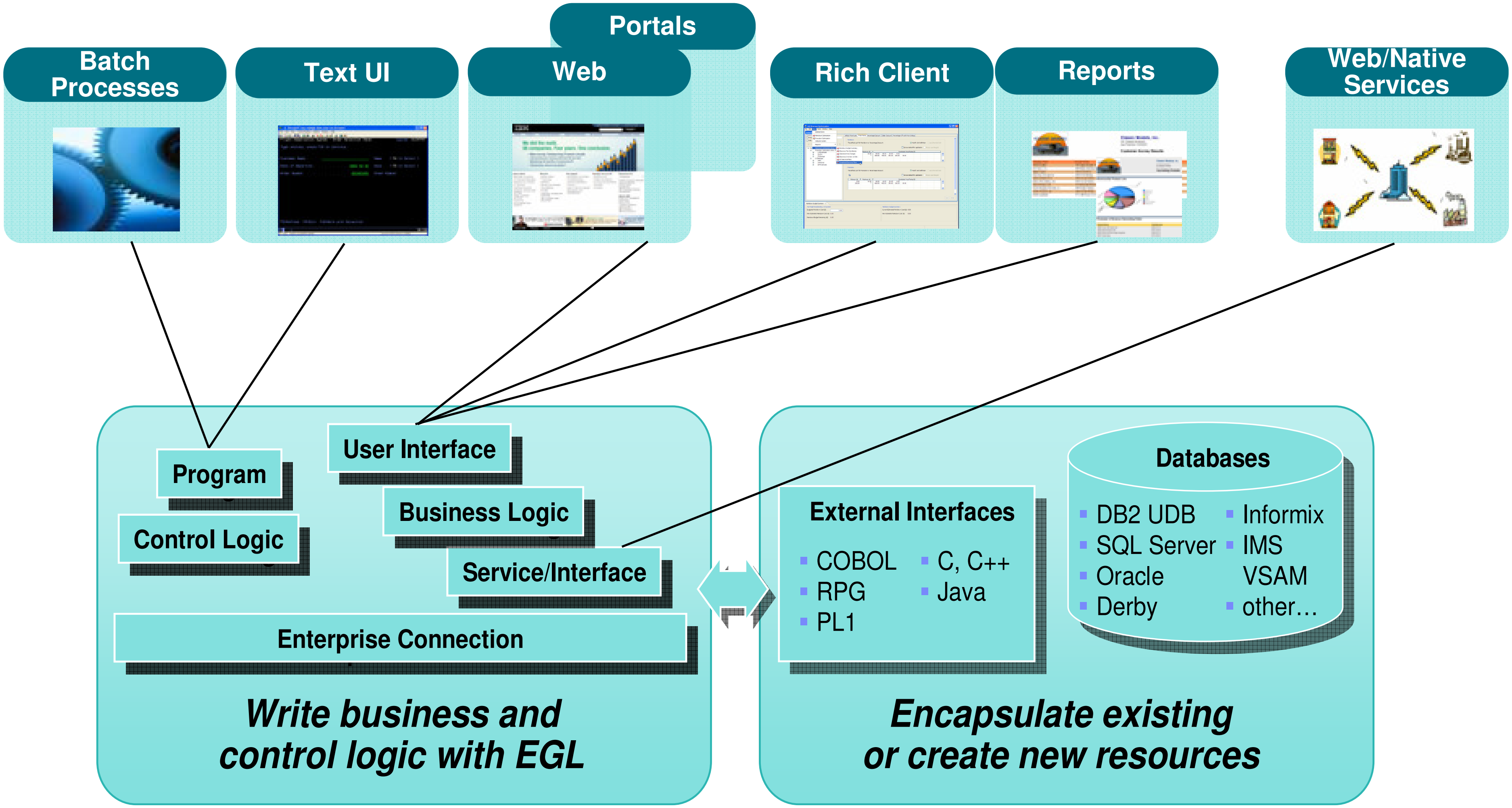


Agenda

- What are web services?
  - How do they play in a mainframe environment?
- **Approaches to creating web services in the mainframe environment**
  - Wrapping
  - Re-engineering
  - **Redeveloping**
- Summary
- Q/A

IBM Rational Business Developer

Unified approach to end-to-end construction that shields developers from intricacies of runtimes and middleware



Batch Processes

Text UI

Portals

Web

Rich Client

Reports

Web/Native Services

Program

Control Logic

User Interface

Business Logic

Service/Interface

Enterprise Connection

Write business and control logic with EGL

External Interfaces

- COBOL
- RPG
- PL1
- C, C++
- Java

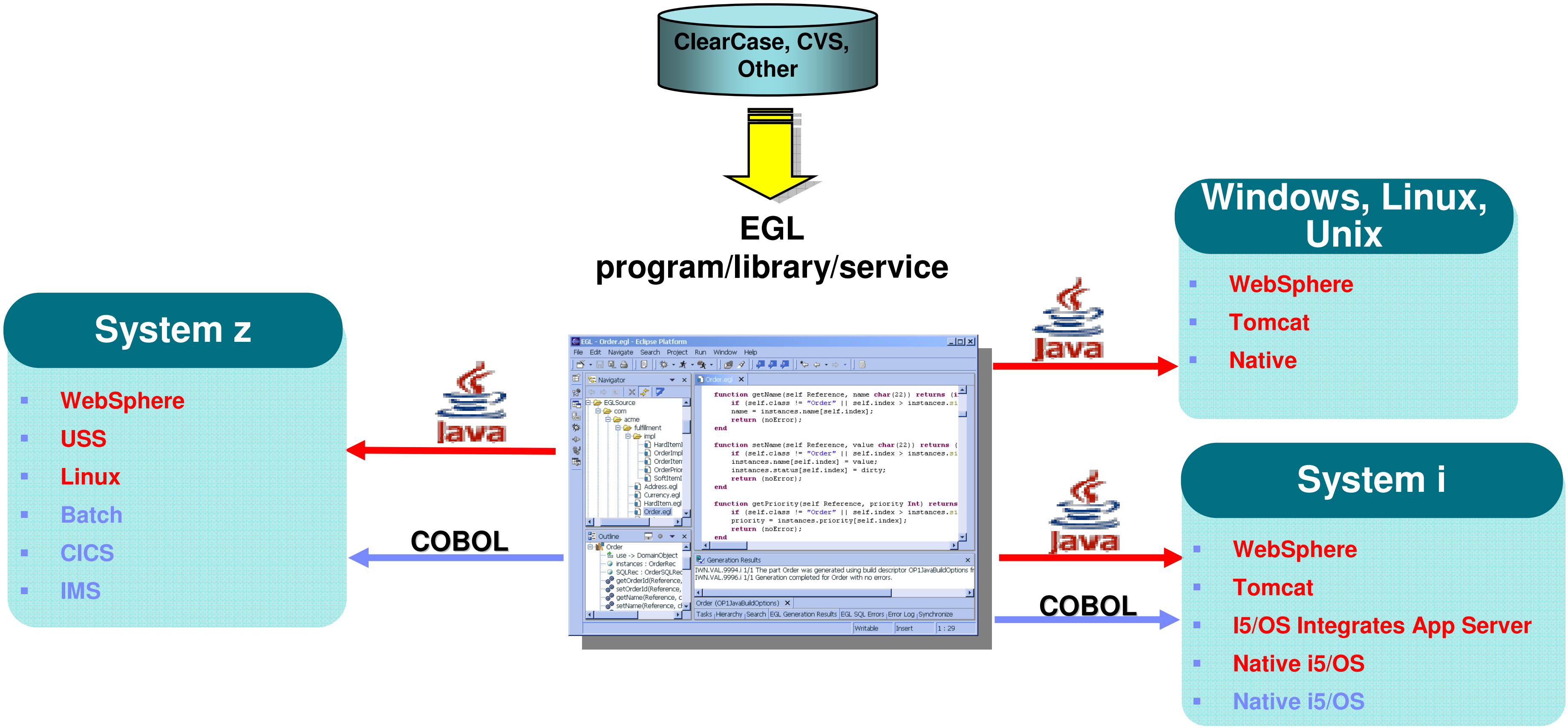
Databases

- DB2 UDB
- SQL Server
- Oracle
- Derby
- Informix
- IMS
- VSAM
- other…

Encapsulate existing or create new resources

The power of Generation

Platform Flexibility



ClearCase, CVS, Other

EGL program/library/service

System z

- WebSphere
- USS
- Linux
- Batch
- CICS
- IMS

COBOL

Windows, Linux, Unix

- WebSphere
- Tomcat
- Native

System i

- WebSphere
- Tomcat
- I5/OS Integrates App Server
- Native i5/OS
- Native i5/OS

COBOL

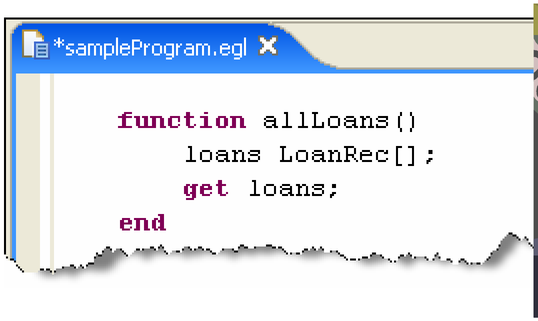
```
function getName(self Reference, name char(22)) returns (i
    if (self.class != "Order" || self.index > instances.si
    name = instances.name[self.index];
    return (noError);
end

function setName(self Reference, value char(22)) returns (
    if (self.class != "Order" || self.index > instances.si
    instances.name[self.index] = value;
    instances.status[self.index] = dirty;
    return (noError);
end

function getPriority(self Reference, priority Int) returns
    if (self.class != "Order" || self.index > instances.si
    priority = instances.priority[self.index];
    return (noError);
end
```

Generation Results

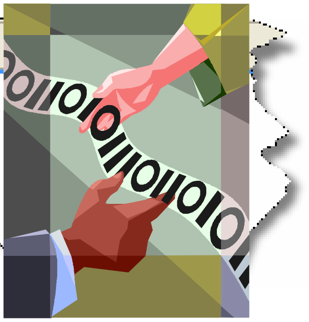
```
IWN.VAL.9994.i 1/1 The part Order was generated using build descriptor OP1JavaBuildOptions fr
IWN.VAL.9996.i 1/1 Generation completed for Order with no errors.
```

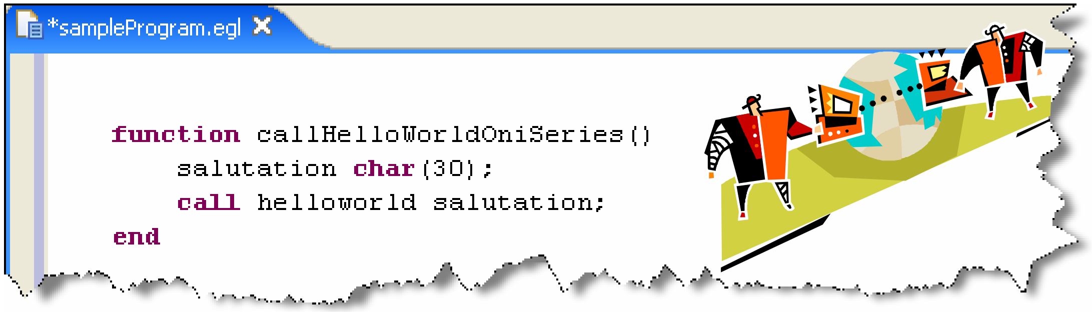
The power of abstractions

- **Data access:**
  - "Records" provide access to:
    - SQL, Indexed, Relative, Serial, DL/I, MQ, Service data
  - Common Verbs for data access (**Get, Add, Replace, Delete**)
  - Allows complete access to SQL statement if needed
  - Common Error Handling



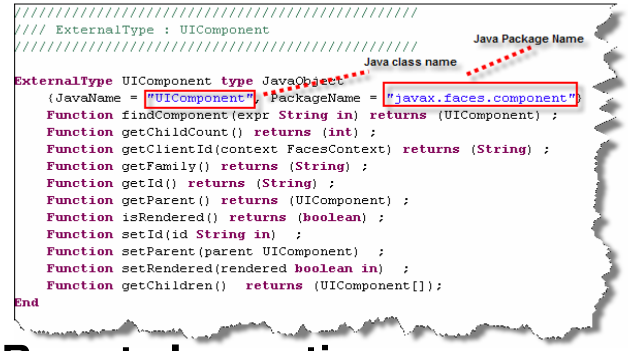
```
function allLoans()
    loans LoanRec[];
    get loans;
end
```





```
function callHelloWorldOniSeries()
    salutation char(30);
    call helloworld salutation;
end
```

- **External application access**
  - External Type to access Java classes
    - Can define constructors, fields, methods (functions), and throw exceptions
    - NEW operator to create instances
  - Intuitive interface
  - Simple function call with the ability to pass and receive parameters



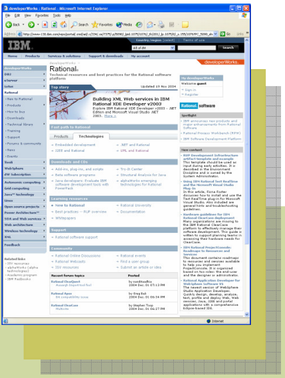
```
////////////////////////////////////////////////
//// ExternalType : UIComponent
////////////////////////////////////////////////

ExternalType UIComponent type JavaObject
    {JavaName = "UIComponent", PackageName = "javax.faces.component"}
    Function findComponent(expr String in) returns (UIComponent) ;
    Function getChildCount() returns (int) ;
    Function getClientId(context FacesContext) returns (String) ;
    Function getFamily() returns (String) ;
    Function getId() returns (String) ;
    Function getParent() returns (UIComponent) ;
    Function isRendered() returns (boolean) ;
    Function setId(id String in)  ;
    Function setParent(parent UIComponent)  ;
    Function setRendered(rendered boolean in)  ;
    Function getChildren()  returns (UIComponent[]);
End
```

- **Remote Invocation**
  - Call COBOL, RPG, C, Java
  - Linkage information separated from code
  - Data mapping, protocol invocation all resolved at runtime, NO code necessary!

EGL – simple programming model

Page Handlers

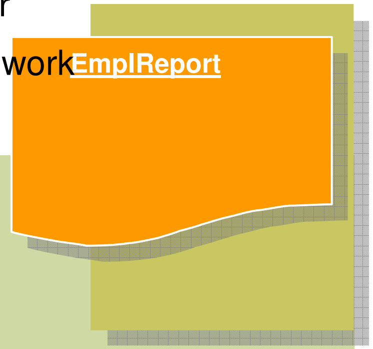
- Contain functions and data related to a .jsp
- Should be mostly “Controller Logic”



```
PageHandler customerInfoPage {
      view = "customerInfoPage.jsp",
      title = "Customer Information",
      onPageLoadFunction = "onPageLoad"}

function onPageLoad()
      loans LoanRec[];
      get loans;
end
…
end
```

Report Handler

- Call-out to EGL “ReportHandler”
- Open Source Reporting Framework



```
ReportHandler customerList

function afterPageInit()
...
end

end
```

Controller Logic/User Interaction

Business Logic

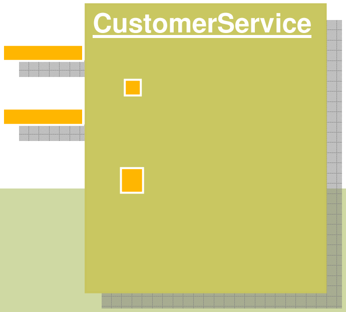
Programs

- Used for single point of entry situations
- TUI program, Batch program, GUI program

```
Program MyProgram

function myFunction()
      salutation char(30);
      call helloworld salutation;
end
```

Services

- “Business Logic” for web apps
- Multiple entry points
- Invoke by function



```
Service Customer

Function getAllCustomers()
end
...

end
```

Java Server Faces Integration

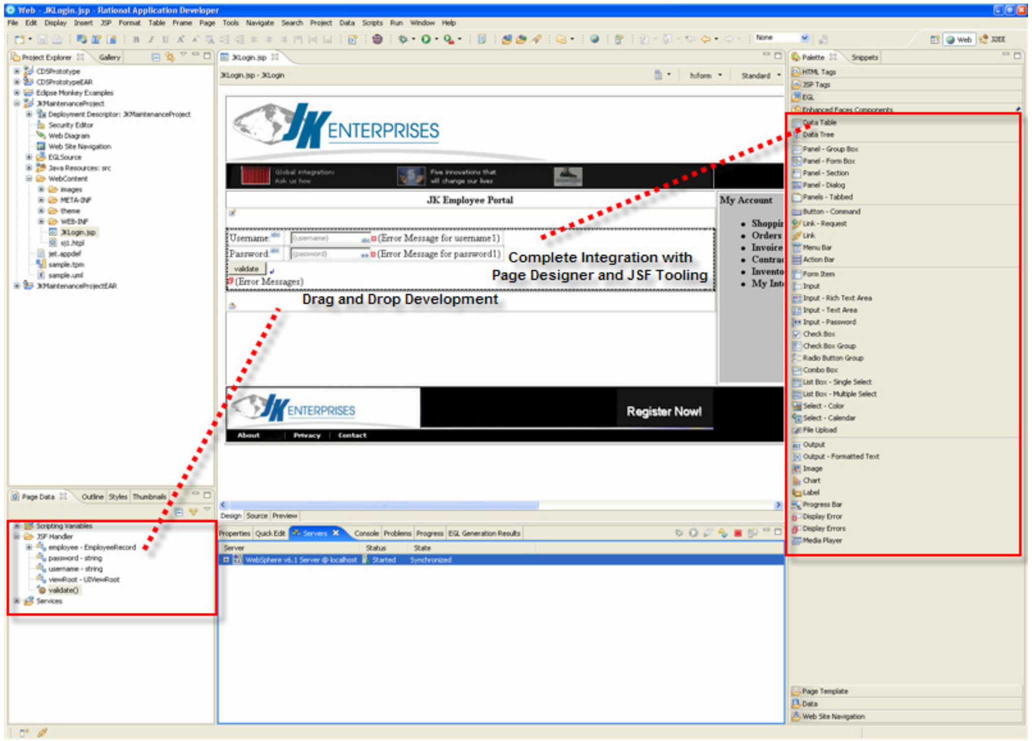
- **First Class integration with Page Designer and JSF tools**
  - Drop EGL data structures on JSP
    - Validation, editing, formatting rules from EGL Data Items applied
    - Appropriate UI controls rendered pre-bound to data declared in EGL Page
  - Server-side event handlers in EGL within context 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**

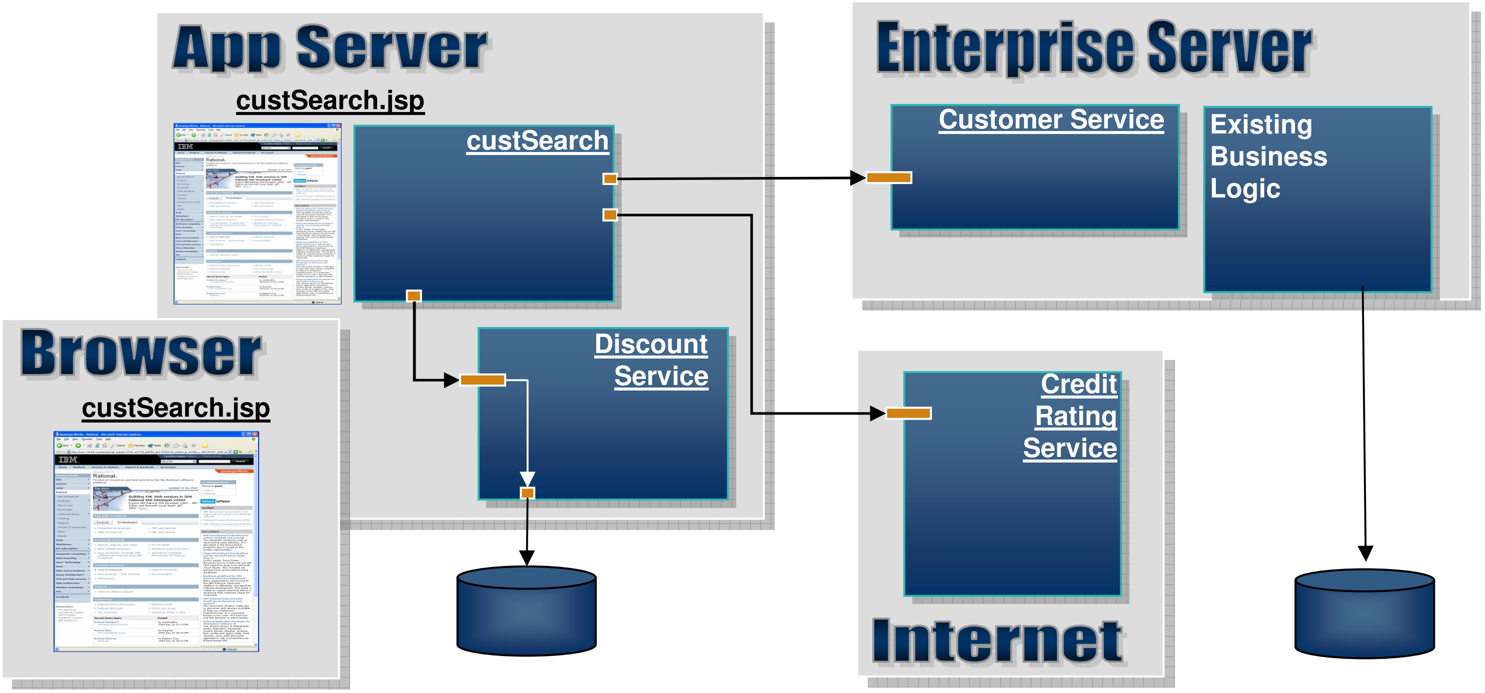
- **AJAX capability built in…partial refresh, etc…**



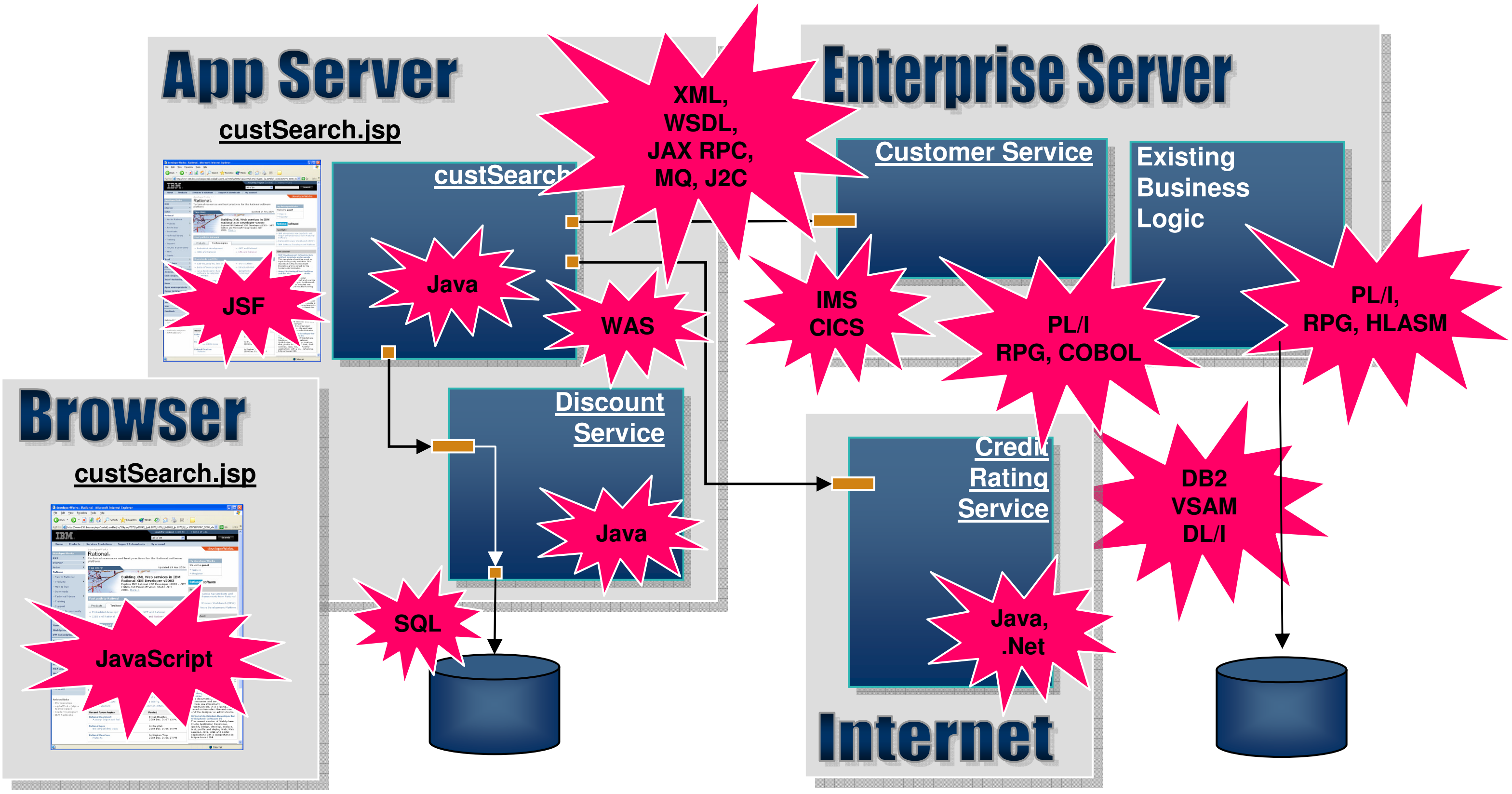
Complete Integration with Page Designer and JSF Tooling

Drag and Drop Development

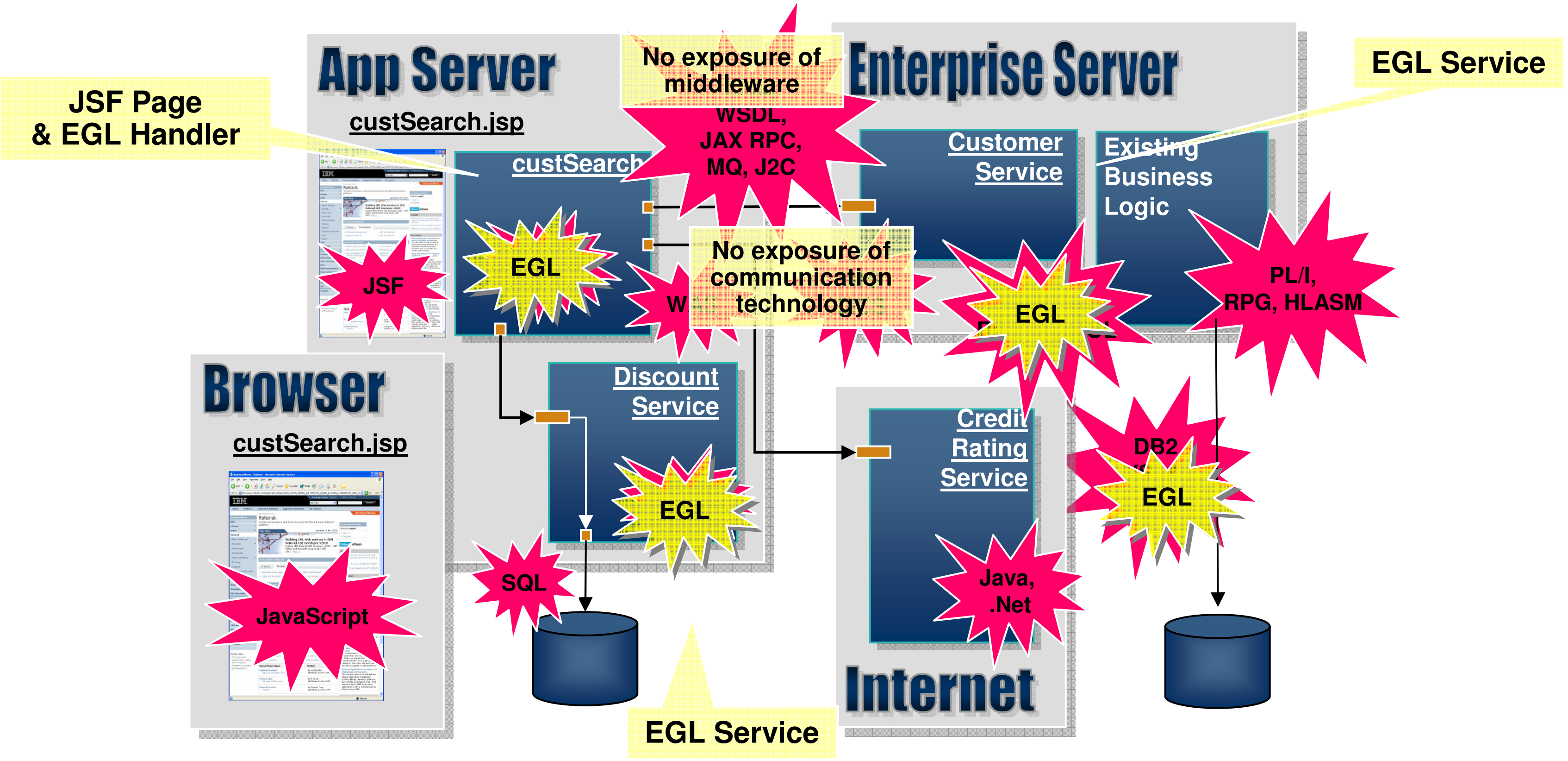
Current J2EE architecture and technology – part 1



Current J2EE architecture and technology – part 2



Current J2EE architecture and technology – part 3



EGL Web – System z Programming Similarities

Page Data
~
Define Map Fields

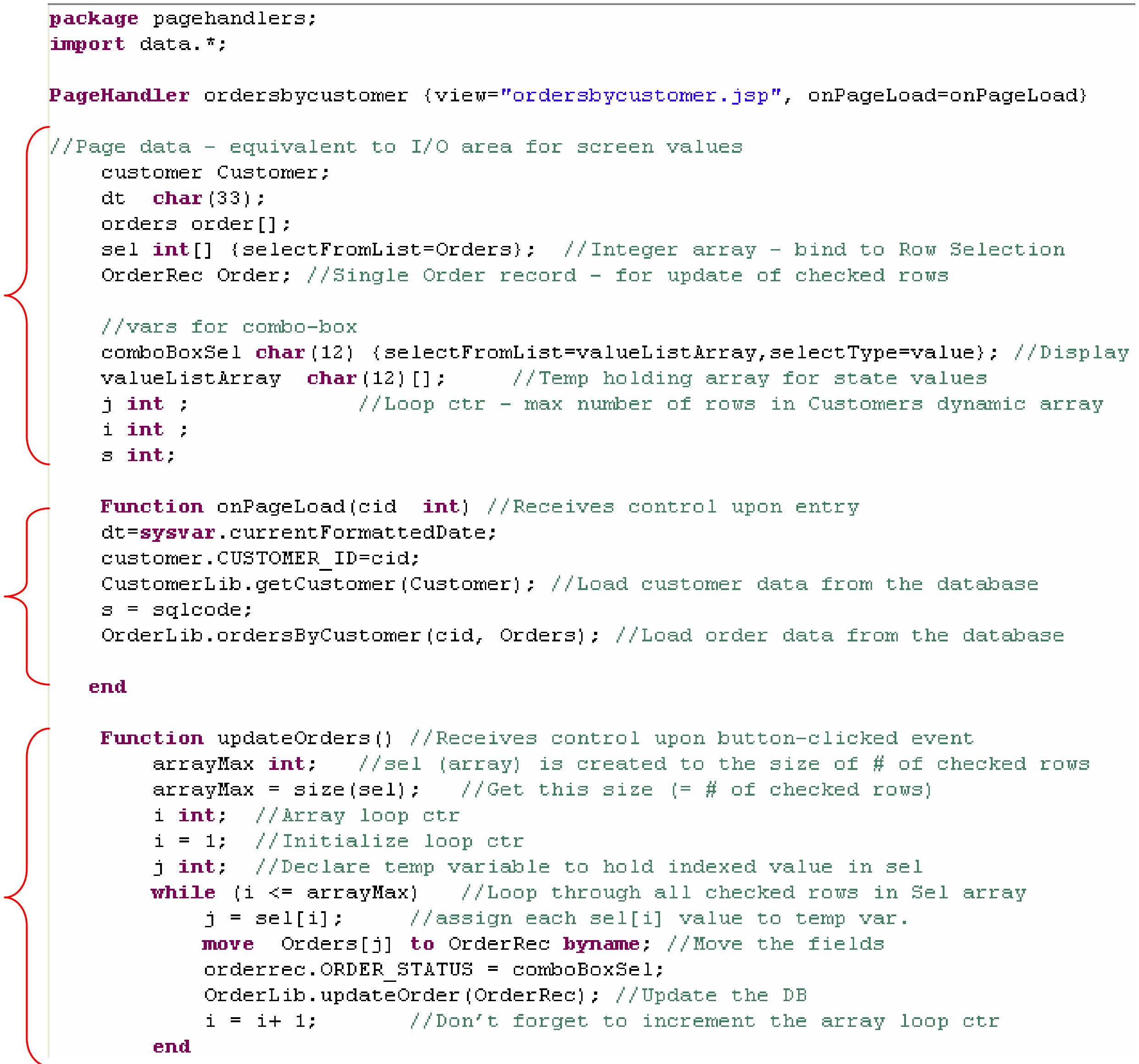
Load values from the database

“Send map”

“Receive map”

Process user-input values

Update Database



```
package pagehandlers;
import data.*;

PageHandler ordersbycustomer {view="ordersbycustomer.jsp", onPageLoad=onPageLoad}

//Page data - equivalent to I/O area for screen values
    customer Customer;
    dt  char(33);
    orders order[];
    sel int[] {selectFromList=Orders};  //Integer array - bind to Row Selection
    OrderRec Order; //Single Order record - for update of checked rows

    //vars for combo-box
    comboBoxSel char(12) {selectFromList=valueListArray,selectType=value}; //Display
    valueListArray  char(12)[];     //Temp holding array for state values
    j int ;             //Loop ctr - max number of rows in Customers dynamic array
    i int ;
    s int;

    Function onPageLoad(cid  int) //Receives control upon entry
    dt=sysvar.currentFormattedDate;
    customer.CUSTOMER_ID=cid;
    CustomerLib.getCustomer(Customer); //Load customer data from the database
    s = sqlcode;
    OrderLib.ordersByCustomer(cid, Orders); //Load order data from the database

   end

    Function updateOrders() //Receives control upon button-clicked event
        arrayMax int;   //sel (array) is created to the size of # of checked rows
        arrayMax = size(sel);   //Get this size (= # of checked rows)
        i int;  //Array loop ctr
        i = 1;  //Initialize loop ctr
        j int;  //Declare temp variable to hold indexed value in sel
        while (i <= arrayMax)   //Loop through all checked rows in Sel array
            j = sel[i];     //assign each sel[i] value to temp var.
            move  Orders[j] to OrderRec byname; //Move the fields
            orderrec.ORDER_STATUS = comboBoxSel;
            OrderLib.updateOrder(OrderRec); //Update the DB
            i = i+ 1;       //Don't forget to increment the array loop ctr
        end
```

Rational. software

Rich Internet Applications

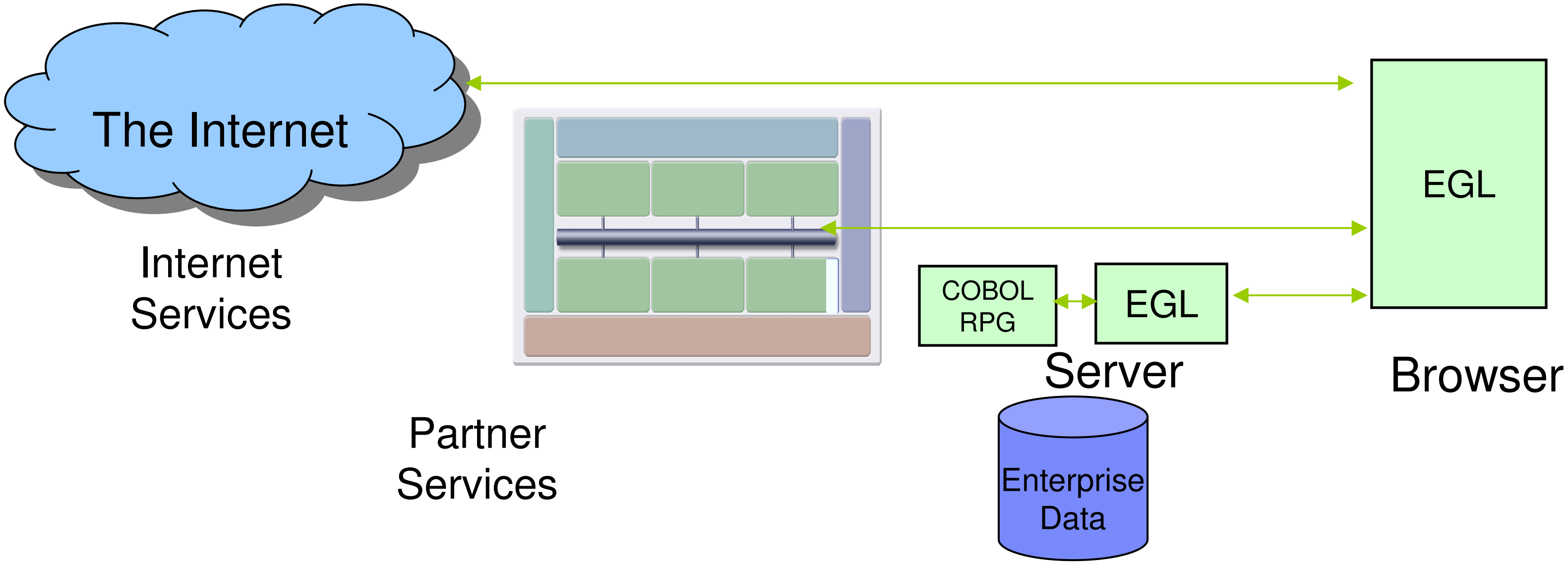
Why RIA?

- **Applications are responsive and user friendly (user productivity)**
- **Lower provisioning costs**
- **Get back to Web architecture basics (REST)**
  - Browser keeps state, Server is stateless thus….
  - Better performance/response (caching)
  - Frees up server CPU and Memory, increase scalability
  - Less complex server-side stack, lower software and administrative costs

Why is it hard to create RIA?

- **Need to know many technologies**
  - JavaScript, HTML, CSS, Ajax, Dojo, SilverLight, SOAP, Flex, XML, JSON, PHP/C#/Java
  - Required technologies operate on a very detailed, low level
  - These technologies use different metaphors and abstractions, hard to mix and match
  - Require highly skilled adaptable software hackers (not typical of traditional business developers)

Simplifying Innovation: EGL Web2.0 Support*



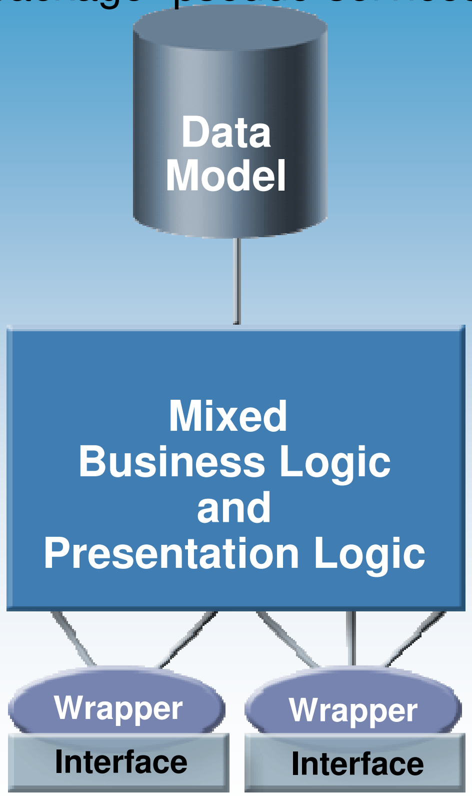
- One language end-to-end: EGL
- One unifying IDE: RBD
- Mobility of Skills
- Innovative Editing Approach

*** Tech Preview :** http://www.alphaworks.ibm.com/tech/reglrws

Summary - Three ways of re-using services from of existing applications

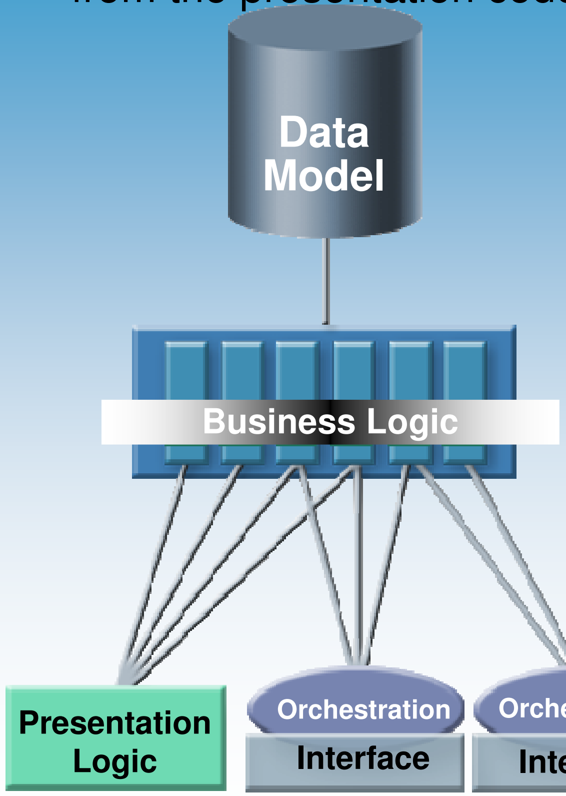
Wrapping

Drive exisitng screen-based applicatons to package "pseudo-services"



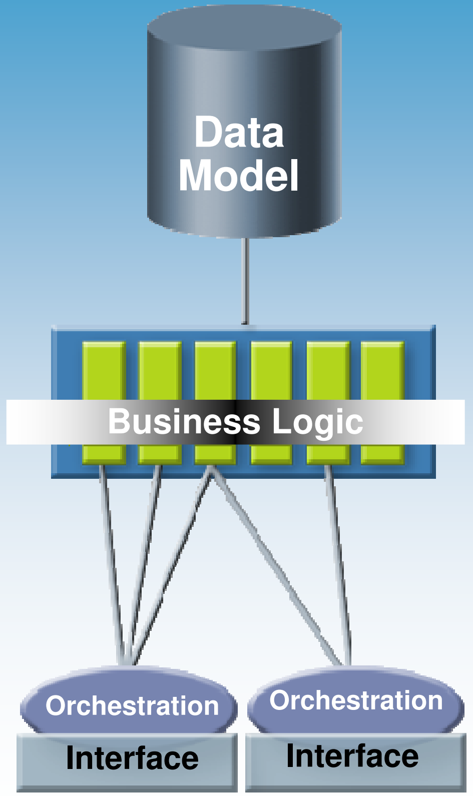
Re-engineering

Reuse existing business logic by extracting it and separating it from the presentation code



Redeveloping

Business logic services are redesigned from scratch



Source: Gartner

Where can you get more information?

System z Sandboxes

Fully hosted online System z environments to experiment, try and buy

www.ibm.com/developerworks/downloads/emsandbox/

More details

Videos, product factsheets, whitepapers, demonstrations

www.ibm.com/software/info/developer/solutions/em/

Education Series

Modern Application Architecture for COBOL Developers

Learn how to design and integrate composite applications across CICS and WebSphere

EGL Distance Learning

10 days of FREE instructor-led virtual class!

EGL Cafe

Resources:	Download, Learn, Presentations, Video/viewlet, Sample Code
Community:	Clients, Partners, Influencers, Press, News & Events
Collaboration:	Blogs, Forums, Tips & Techniques Comments, Ratings
Testimonials:	Case Studies

www.ibm.com/rational/eglcafe

Learn more at:

- IBM Enterprise Modernization Solutions
  - http://www.ibm.com/rational/modernization
- IBM Rational Software Delivery Platform
  - http://www-306.ibm.com/software/info/developer/index.jsp
- Process and portfolio management
  - http://www-306.ibm.com/software/rational/offerings/lifecycle.html
- Change and release management
  - http://www-306.ibm.com/software/rational/offerings/lifecycle.html
- Quality management
  - http://www-306.ibm.com/software/rational/offerings/testing.html
- Architecture management
  - http://www-306.ibm.com/software/rational/offerings/design.html
- Rational trial downloads
  - http://www.ibm.com/developerworks/rational/downloads/?S_TACT=105AGX23&S_CMP=RCD
- developerWorks Rational
  - http://www.ibm.com/developerworks/rational
- IBM Rational TV
  - http://www-306.ibm.com/software/info/television/index.jsp?cat=rational&media=video&item=en_us/rational/xml/M259765N40519Z80.xml
- IBM Rational Business Partners
  - http://www-306.ibm.com/software/rational/partners/

Questions