



IBM System z Software

## Deploying Application Development for composite COBOL, PL1 and Java Application



**ON DEMAND BUSINESS™**

# Trademarks

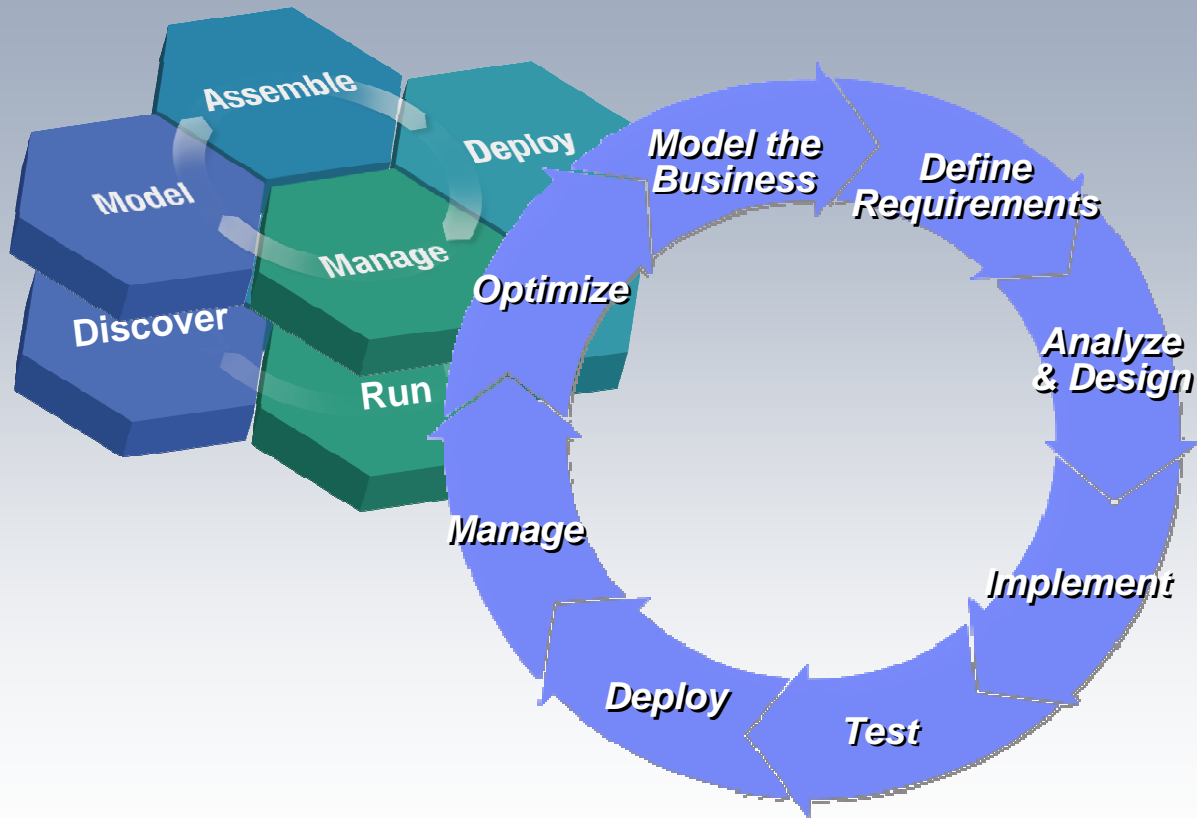
- The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM Trademarks, see [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml)
  - ▶ AS/400, CICS, DB2, Domino, E-business logo, ESCON, eServer, FICON, IBM, IBM Logo, IMS, iSeries, Lotus, MVS, Notes, OS/390, pSeries, Rational, RS/6000, S/390, Tivoli, VM/ESA, VSE/ESA, WebSphere, xSeries, z/OS, zSeries, z/VM
- The following are trademarks or registered trademarks of other companies
  - ▶ Linux is a registered trademark of Linus Torvalds
  - ▶ Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries
  - ▶ UNIX is a registered trademark of The Open Group in the United States and other countries.
  - ▶ Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
  - ▶ SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
  - ▶ Intel is a registered trademark of Intel Corporation
  - ▶ \* All other products may be trademarks or registered trademarks of their respective companies.
- **Notes:**
  - Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
  - IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.
  - All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
  - This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
  - All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.



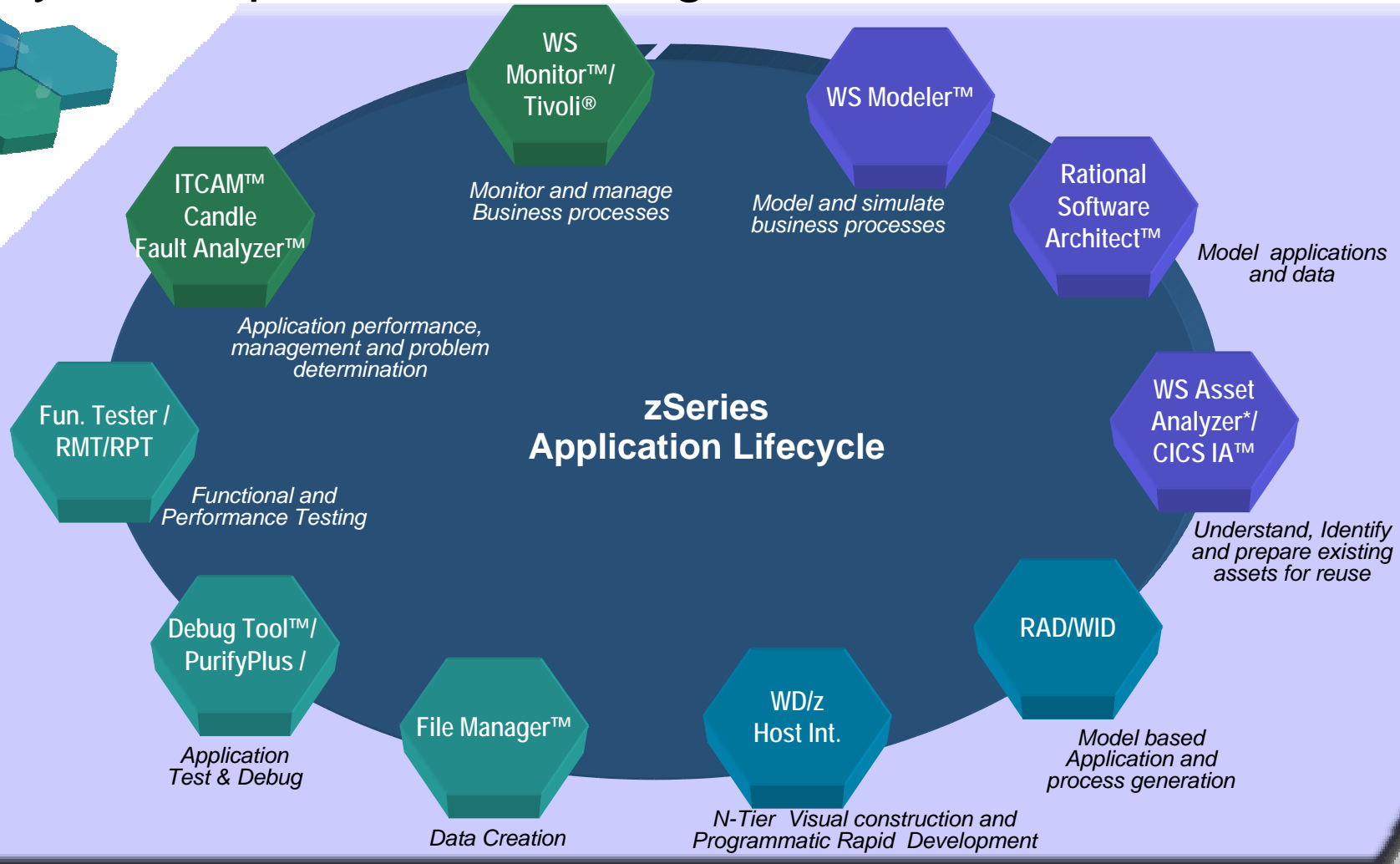
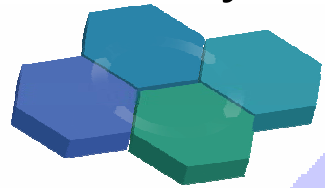
# Agenda

- Where do we fit
- **Introduction SOA and WebSphere Developer for zSeries™**
- **Introduction to tool concepts**
- **Detail information and demonstration of zOS™ development**

# Enterprise Platform – Life Cycle



# IBM System z process and integrated workflow



Common Process and Guidance – Reqpro, ClearQuest, Rational Method Composer

Software Configuration Management – ClearCase, ClearQuest, SCLM

# IBM SDP for System z and Composites

## Model and Discover

- WS Business Integration Modeler™
- Rational Software Architect
- WS Asset Analyzer™/CICS IA
- Facilitate understanding
- Identify reusable components

## Develop and Integrate

- WebSphere Developer for zSeries
  - Speed up and simplify:
    - Development
    - Integration
    - Webification
    - Componentization and assembly

## Test, Deploy, and Manage

- ITCAM/ATA
- RPT/Workload Simulator
- System z PD / CICS™ and DB2™ Tools
- Rational
  - Reduce production downtime
    - Function test
    - Simulate application loads
    - Identify bottlenecks
    - Resolve complex faults

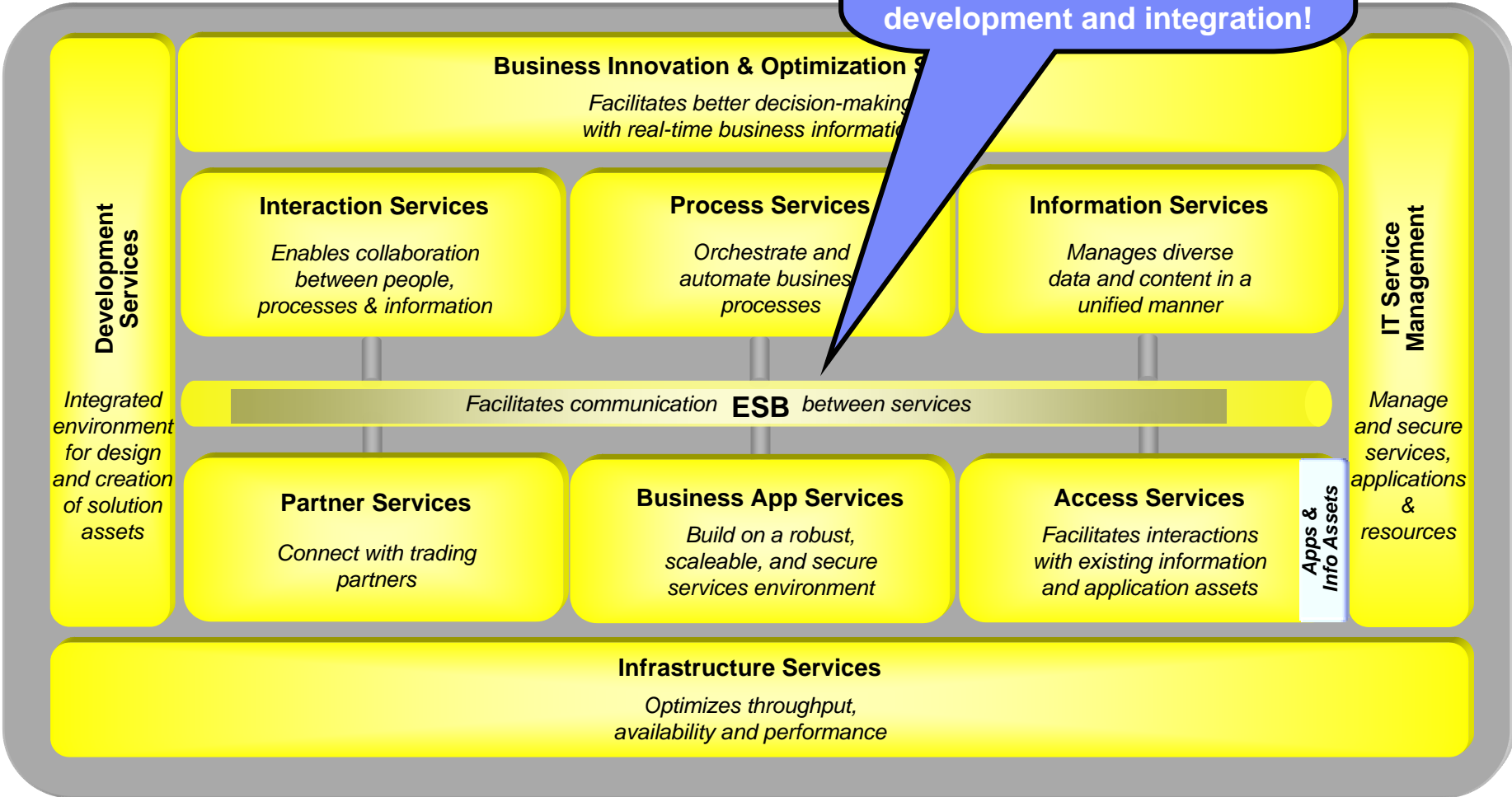
## Run


- WASz, CICS, and IMS™ transactions
- Enterprise COBOL™ and PL/I
- Rational and SCLM™ Tools
  - Highest Qualities of service
  - Broadest ROI's
  - Utilization of standards/process

# SOA Reference Architecture

## Supporting your SOA Lifecycle

Note that all the blocks in this diagram are yellow. The SOA Reference Architecture applies directly to System z for all aspects of service development and integration!



 Leverage zSeries middleware for maximum business flexibility.

# Agenda

- **Where do we fit**
- Introduction Modern SOA, CICS, IMS, WAS and WDz
- **Introduction to tool concepts**
- **Detail information and demonstration of zOS development**



# Enabling a robust, flexible SOA runtime environment

*While maximizing the value of existing assets* **Fully SOA capable!**

## WebSphere Application Server™ V6

March 2005

- Extend existing Java assets with support for Web Services standards and standards-based messaging
- Help ensure 24x7 availability of business-critical applications with clustering and high availability
- Build and deploy Web Services quickly and easily with rapid development and deployment features

## CICS Transaction Server™ V3.1

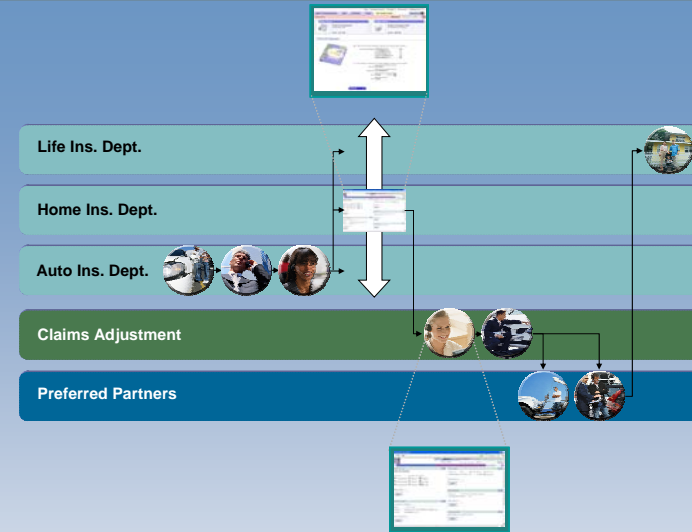
March 2005

- Exploit provider/requestor Web service support for CICS assets, based on full Web service standards
- Extend the value of CICS transactions in a mixed language environment
- Build Web services from CICS transactions with no change to existing applications.

## IMS Transaction and Database™ V9

October 2004

- Exploit Web service support for IMS assets, based on full Web service standards
- Extend the value of IMS transactions in a mixed language environment
- Build Web services from IMS transactions with no change to existing applications



**#1 in market share for  
Application Server software**

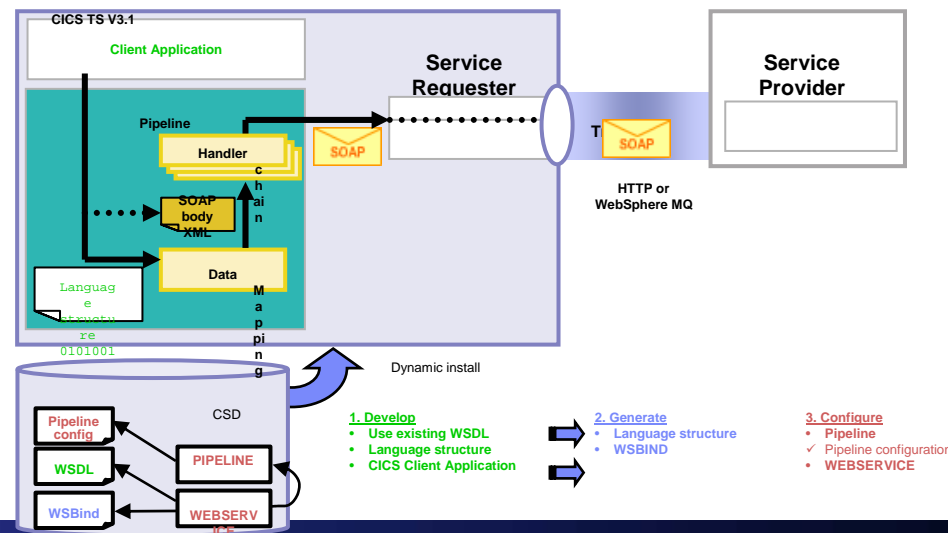
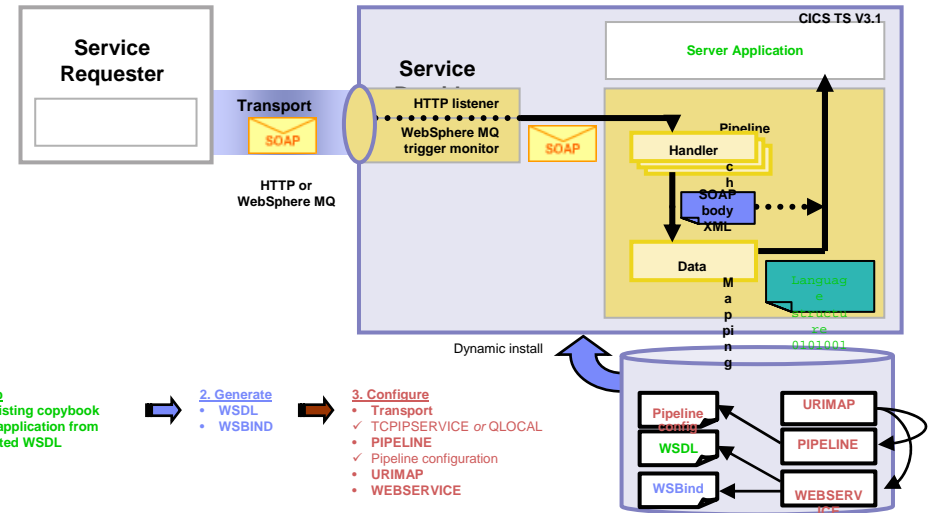


**IBM WebSphere  
Application Server  
comes out on top**

**35+ years of maturity and innovation  
in transaction and data systems**

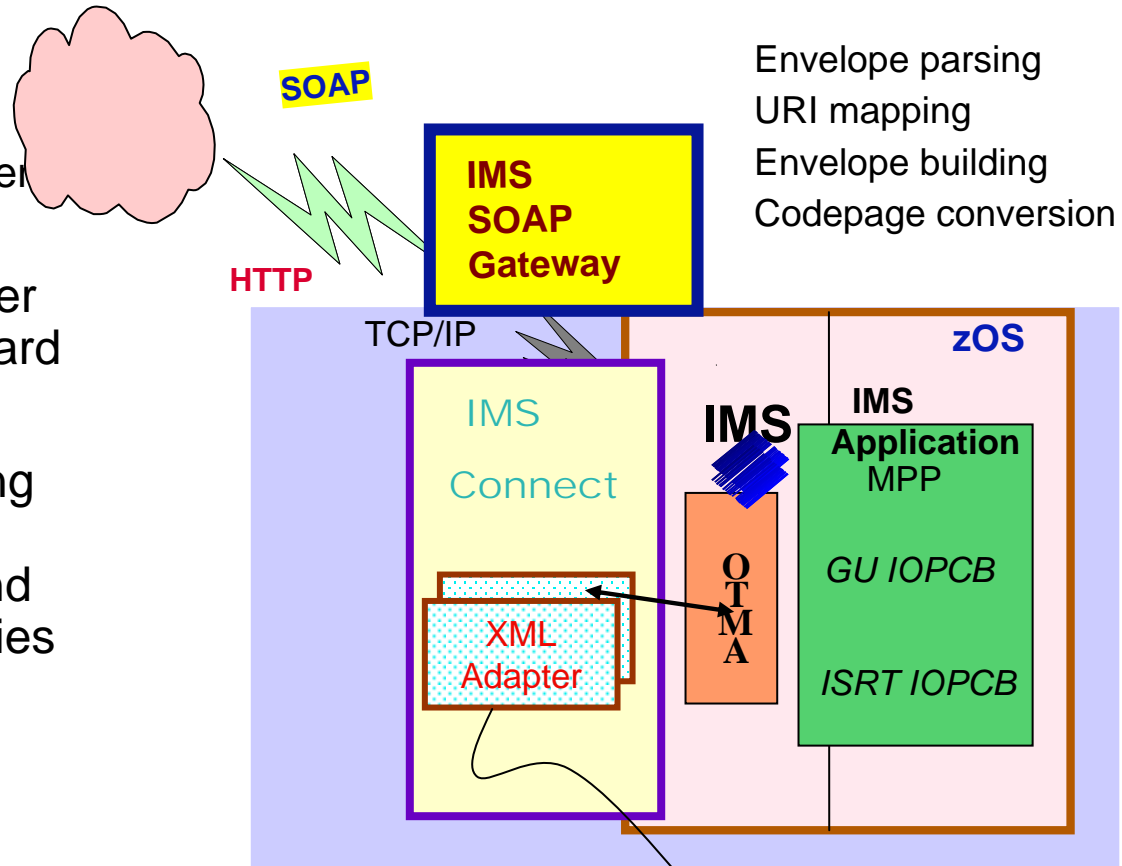
# CICS Web Services

- Web services capabilities extend CICS applications directly to a Service Oriented Architecture
  - A CICS application can now be a Web service provider and requester
- Evolution of SOAP for CICS feature
  - Simplification of pipeline and system management
  - Fully integrated into CICS
    - RDO, problem determination, monitoring & statistics
    - New tooling support for easier application development
  - Guidance provided to assist migration from the SOAP for CICS Feature
- Rich set of Web services standards supported
  1. SOAP 1.1 and 1.2 to send and receive Web services messages
  2. WS-I Basic Profile 1.0a for interoperability with between providers and requesters using SOAP
  3. WS-Coordination extensible coordination framework, and specific coordination of transactions
  4. WS-AtomicTransaction for transaction coordination
  5. WS-Security for authentication and encryption of all or part of a message
    - SOAP Message Security, Username Token Profile 1.0, X.509 Certificate Token
- Both HTTP and WebSphere MQ network layers supported
  - For flexible deployment options dependant on application and IT requirements
  - CICS applications acting as providers or requesters are agnostic to the transport mechanism used



# IMS And Web Services

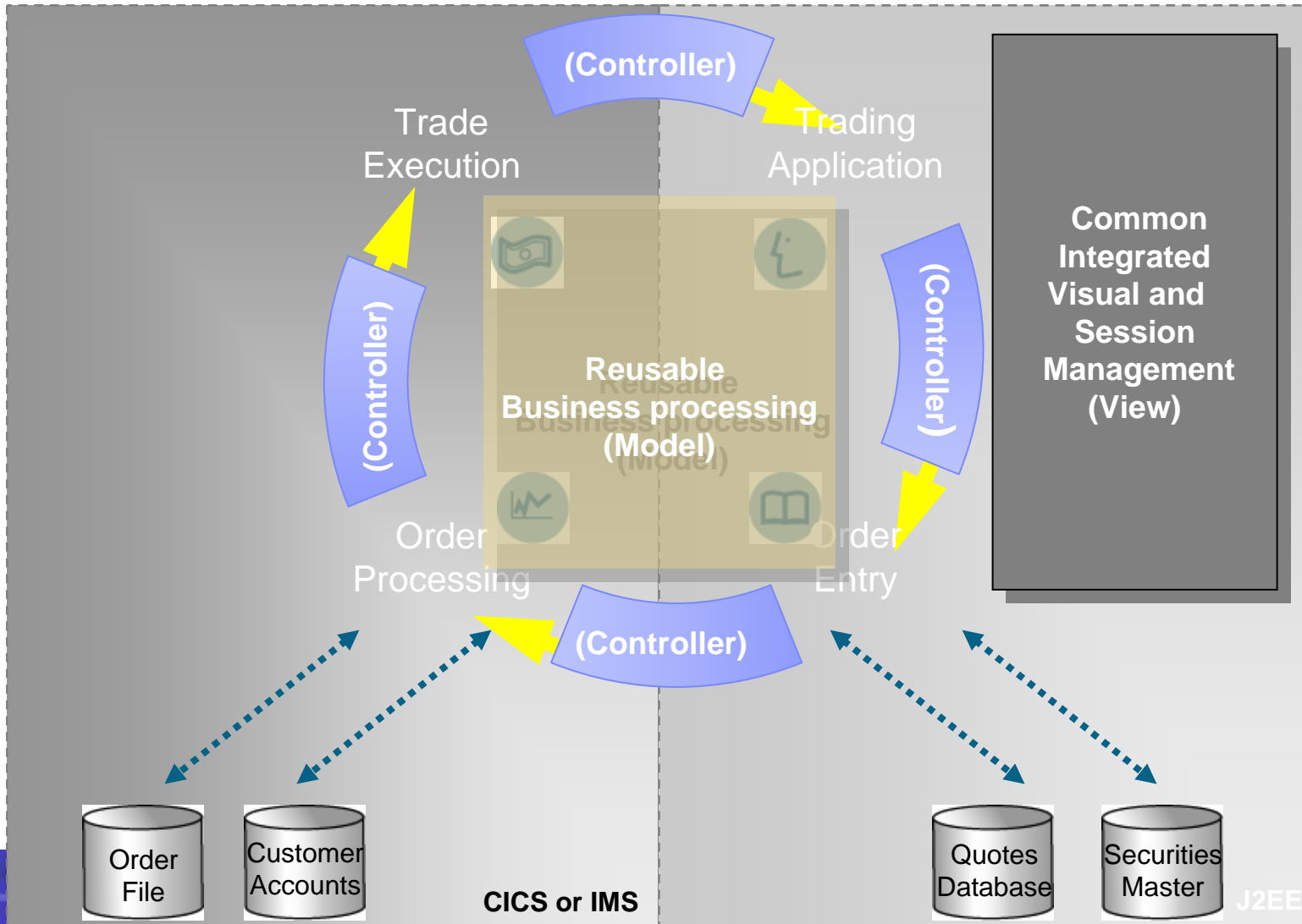
- SOAP for IMS at [www.ibm.com/ims](http://www.ibm.com/ims)
  - Technology preview recently announced in February
- Maximize re-use of customer enterprise assets via standard interfaces
- Support collaboration among IMS and IBM and non-IBM components, both within and beyond enterprise boundaries



WSED-generated XML adapter for COBOL

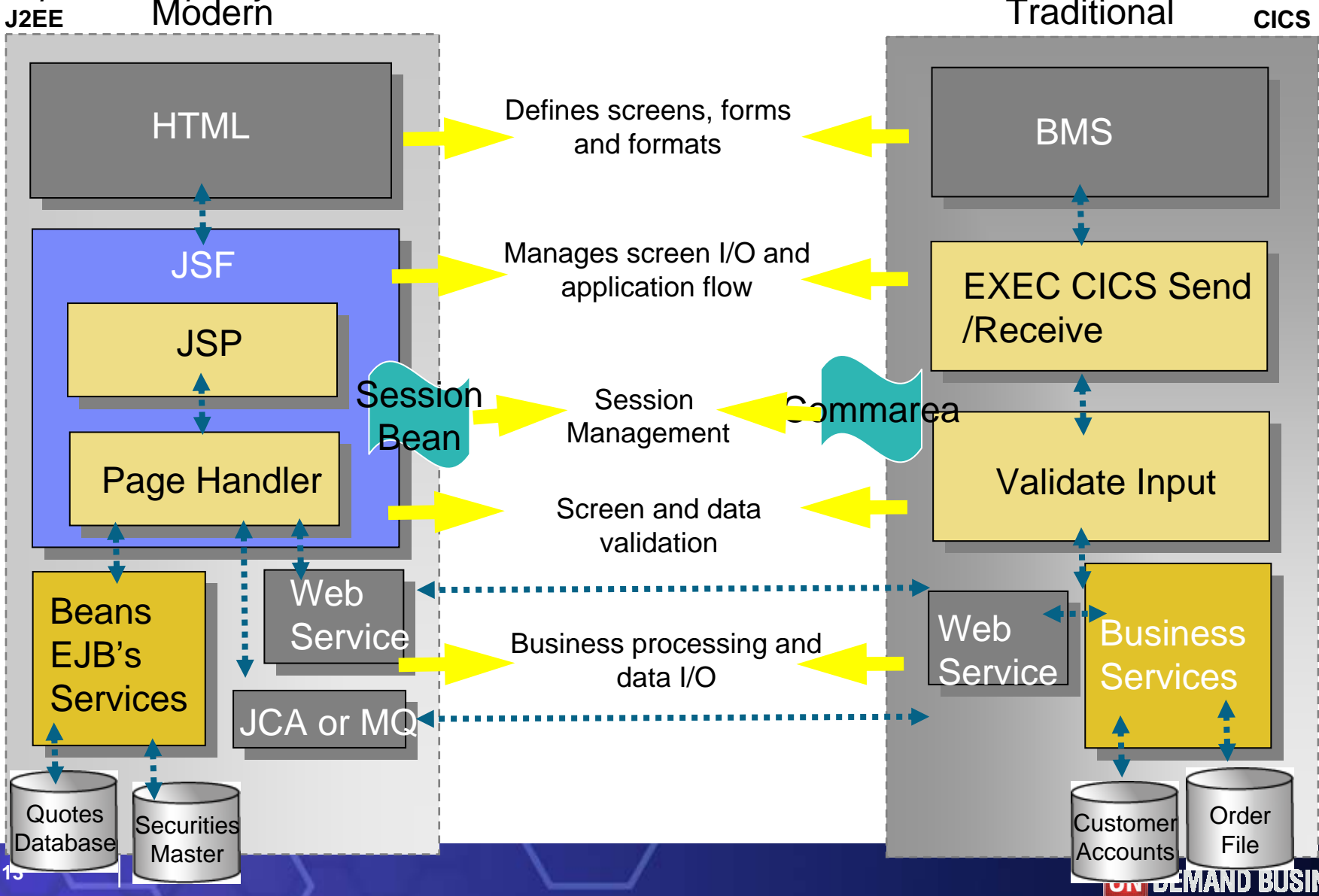
# Mixed Workload Application Components

*Spans multiple system and middleware boundaries*



# It's not that different

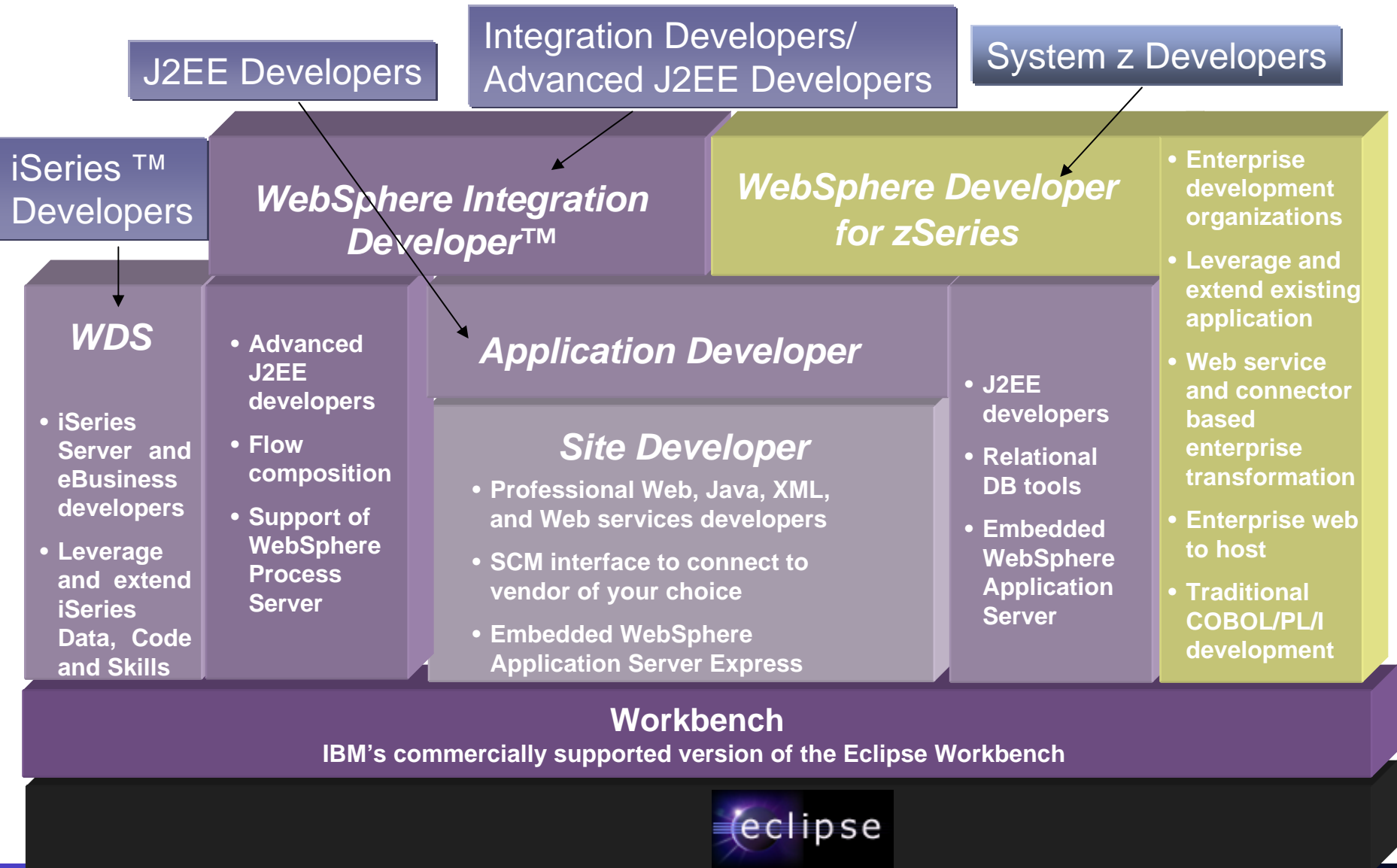
*Spans multiple system and middleware boundaries*



# Agenda

- **Where do we fit**
- **Introduction Modern SOA, CICS, IMS, WAS and WDz**
- Introduction to tool concepts
- **Detail information and demonstration of zOS development**

# WebSphere/Rational Development Family



# WebSphere Developer for z/OS

## What is WebSphere Developer for z/OS?

Brings the power of modern application architectures and rapid application development and robust team support, to diverse enterprise IT organizations

- Intuitive, visual construction based on open standards (JSF and Struts)
- Broad SOA support through Web services and JCA linking visual environments and user sessions to CICS QOS
- Easy to learn, COBOL like language for rapid UI and Business dev.
- Facilities to develop, debug and deploy Java, COBOL, & PL/I applications and services



## V6 New Functionality

CICS V3 exploitation - Subsystem support latest – CICS, WAS, DB2

- Connectivity enhancements
  - WSDL automation from existing processing
  - Support for new CICS WS run timemarshallers
  - XML based COBOL adapter enhancements
  - JCA connectors supporting latest CTG
- Modern Architectural enhancements
  - Service Flow Modeler support (Preview)
  - Leverages support for channels
- Traditional support for:
  - EGL support for VG based Web Transactions
  - BMS Editor
- Integration with other IBM application lifecycle products
- Eclipse V3 exploitation

## V6.0.1

- GA Service Flow Modeler
- 3rd party and open SCM support
- Preview: CICS Patterns “List Detail, CRUD, etc.)
- ...and more

## Benefits

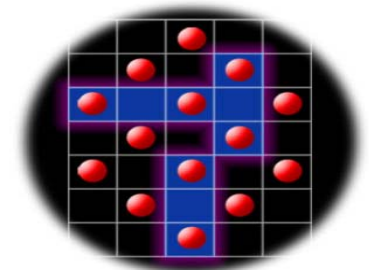
### Single tool for all application transformation

- Increase developer productivity
- Leverage existing processing by enabling legacy assets to be used in SOA’s
- Integrate with lifecycle
- Extend skill sets across the organization
  - Enterprise Generation Language limits need for Java or traditional expertise

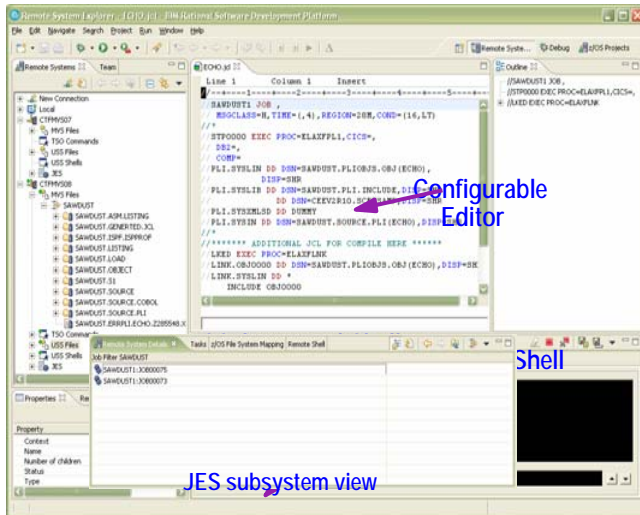


# zOS Application Development tools

- Interactive, workstation-based environment
  - Faster development with less errors
  - Work offline or online
  - Local/workstation projects
- Edit/compile/debug on the workstation
  - Remote or Local
  - Language sensitive editors for COBOL, PL/I, ASM, JCL
  - BMS Map development
- Interactive access to zOS
  - Job generation, submission, and monitoring
  - TSO/USS command execution



*Traditional applications and COBOL/PL/I Services*



Disconnect -vs- Connected

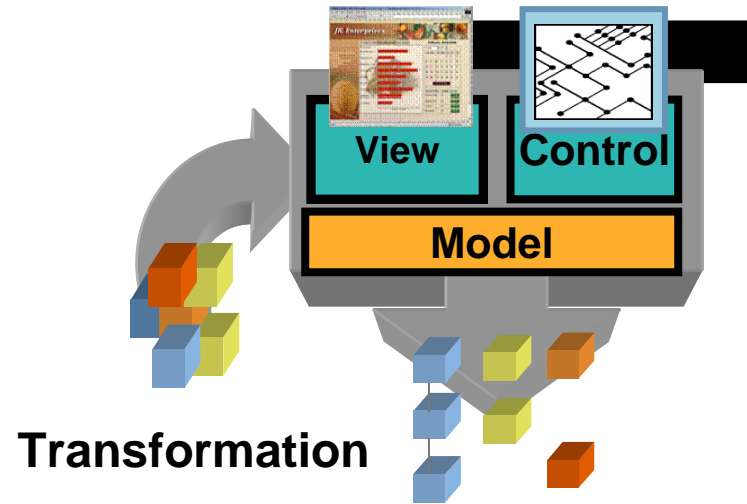
MVS PDS members

Data set characteristics

Configurable Editor

Shell

JES subsystem view

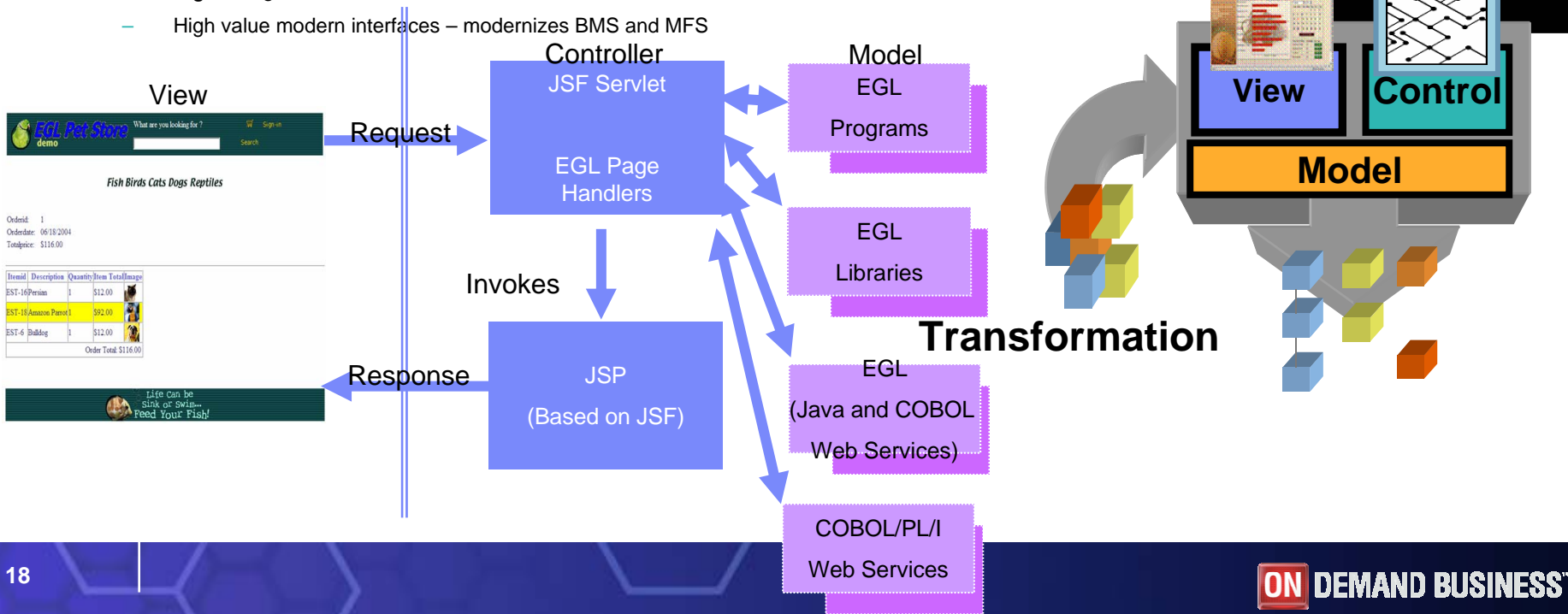


# Web Development tools

- Interactive, Web development
  - Static and Dynamic Web development
  - XML
  
- Java Development
  - Java and J2EE development
  - Java Server Faces
  - Struts
  
- EGL 4GL Java/Web development
  - Generate to language of Choice
  - Tight integration to JSF
  - High value modern interfaces – modernizes BMS and MFS

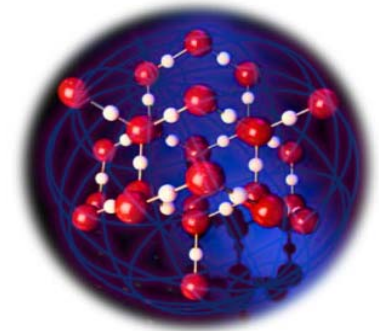


**Web applications and services**

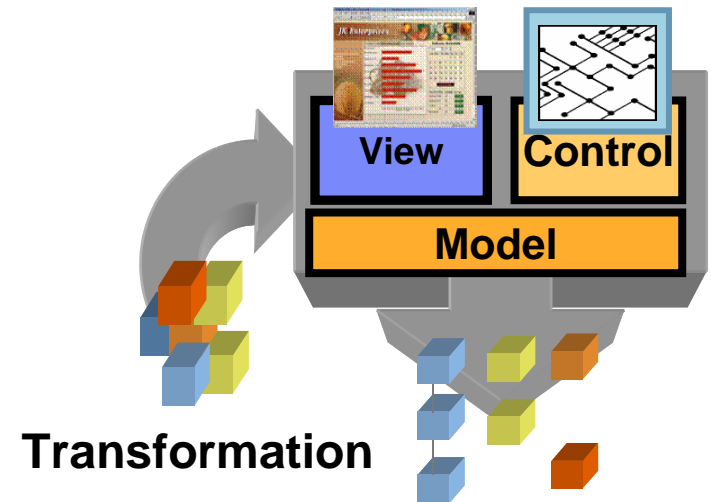
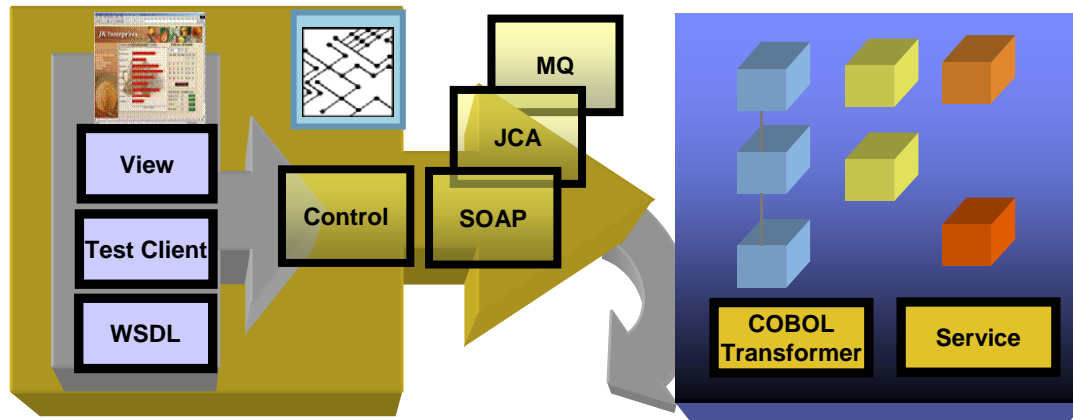


# z/OS Composite Development tools

- Transition of Traditional environments to Web and Mixed Workload or Composite applications
- SOA / SOAP / XML / Enablement
- JCA Support
- Service Flow Modeler
- HATS
- Enterprise Generation Language (EGL) / JSF
  - COBOL/CICS generation
  - Java generation



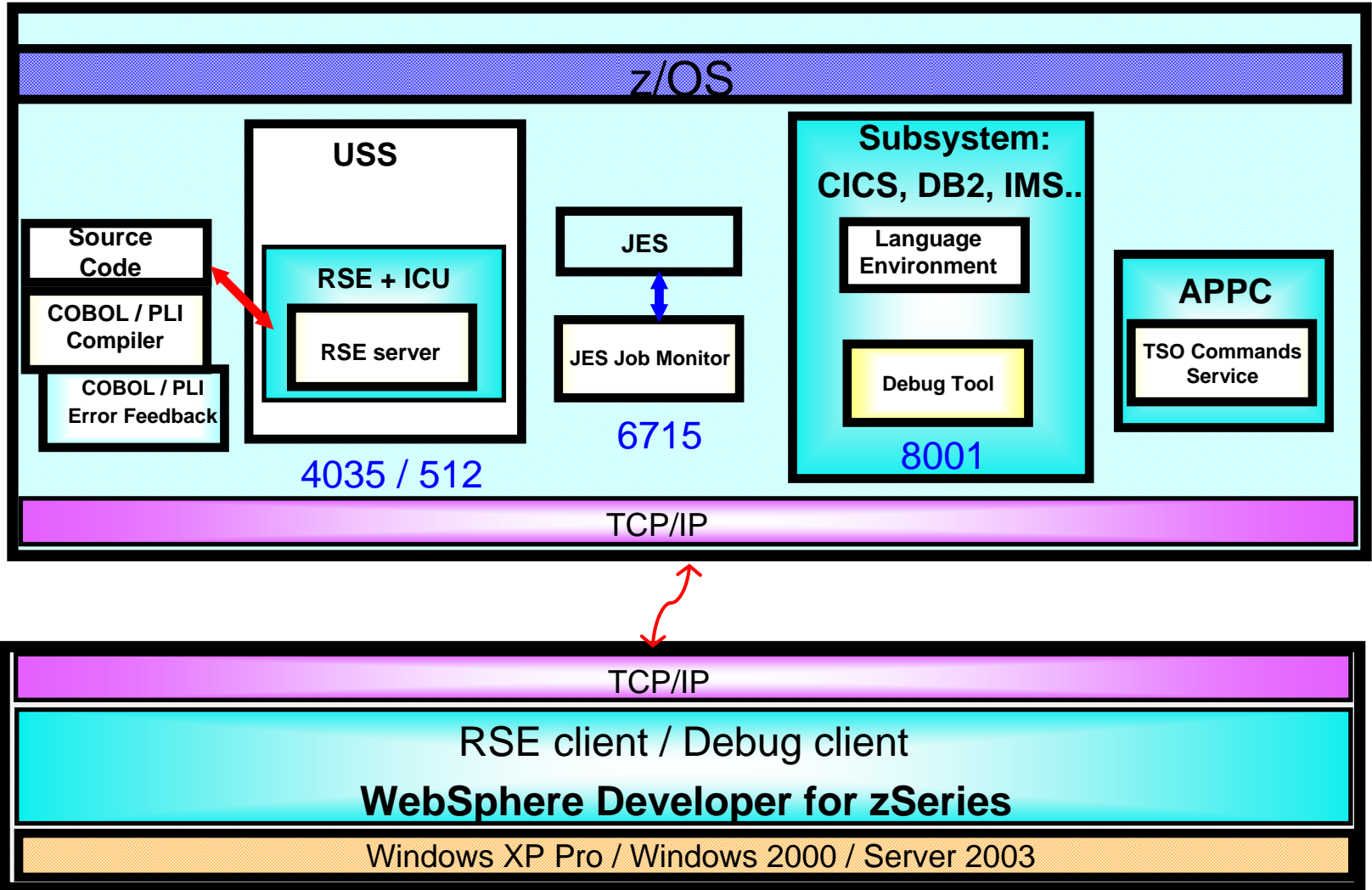
*On Demand*



# Agenda

- **Where do we fit**
- **Introduction Modern SOA, CICS, IMS, WAS and WDz**
- **Introduction to tool concepts**
- **Detail information and demonstration of ZOS development**

# Host / Client Interaction



# Perspectives for zOS Application Development

- What's a Perspective?
  - An arrangement of views (windows) and editors targeted for a particular task
  
- zOS Systems Perspective – Remote Systems Explorer (RSE)
  - View of datasets & members
  - TSO Command Processor, Job Monitor, Edit Window
  
- zOS Projects Perspective
  - IDEs organize work in projects
  - Project properties - compile options, link options, etc apply to artifacts in project
  - Remote projects – application artifacts exist on zOS
  - Local projects – application artifacts existing on the workstation
  
- EGL Perspectives
  - Works in conjunction with JSF
  - Generates COBOL/CICS or Java

# z/OS System Perspective

The screenshot displays the Remote System Explorer interface for a z/OS system. The main window shows a JCL file named ECHO.jcl with the following content:

```
Line 1      Column 1      Insert
//-----1-----2-----3-----4-----5-----+
//SAWDUST1 JOB ,
//  MSGCLASS=H, TIME=(,4), REGION=28M, COND=(16,LT)
// *
//STP0000 EXEC PROC=ELAXFP1,CICS=,
//  DB2=,
//  COMP=
//  PLI.SYSLIN DD DSN=SAWDUST.PLI OBJ (ECHO) ,
//                DISP=SHR
//  PLI.SYSLIB DD DSN=SAWDUST.PLI INCLUDE,DISP=SHR
//                DD DSN=CEEV2R10.SCEESAMP,DISP=SHR
//  PLI.SYSXMSLD DD DUMMY
//  PLI.SYSIN DD DSN=SAWDUST.SOURCE.PLI (ECHO) ,DISP=SHR
// *
//***** ADDITIONAL JCL FOR COMPILE HERE *****
//LKED EXEC PROC=ELAXFLNK
//LINK.OBJ0000 DD DSN=SAWDUST.PLI OBJ (ECHO) ,DISP=SH
//LINK.SYSLIN DD *
//                INCLUDE OBJ0000
```

Annotations on the screenshot include:

- Disconnected -vs- Connected:** Two red circles in the Remote Systems tree highlight the status of connections.
- MVS PDS members:** A purple arrow points to the SAWDUST directory in the tree, which contains various PDS members.
- Data set characteristics:** A purple arrow points to the Properties window at the bottom, which shows details for the selected PDS.
- Configurable Editor:** A purple arrow points to the main JCL editor window.
- USS Command Shell:** A purple arrow points to the Remote System Details window, which shows the Job Filter for the SAWDUST subsystem.
- TSO Command Shell:** A purple arrow points to the Remote System Details window, which shows the Job Filter for the TSO subsystem.
- JES subsystem view:** A purple arrow points to the Job Filter window, which shows the JES subsystem view.

Property	Value
Context	
Name	TSO-CTF...
Number of children	0
Status	Running
Type	Command Shell

# z/OS Project Perspective – remote project

The screenshot displays the IBM Rational Software Development Platform interface for a z/OS project named 'PrintApp.cbl'. The interface is divided into several panes:

- Remote System Explorer (Left):** Shows a tree view of project files. A red circle highlights the 'Remote' project, and another red circle highlights the 'Local' project. A purple arrow points to the 'MVS Projects - able to work offline / online' text.
- Code Editor (Center):** Displays the source code for 'PrintApp.cbl'. A blue arrow points to the 'Language sensitive Editor' text.
- Remote Systems (Right):** Shows a tree view of remote systems. A blue arrow points to the 'System Navigator' text.
- Properties (Bottom Left):** Shows the source outline for the current file. A purple arrow points to the 'Source outline' text.
- z/OS File System Mapping (Bottom Center):** A table showing root connections to local and remote systems.

Name	Parent profile	Remote system type	Connection status	Host name
Local	sawdust	Local	Some subsystems connected	LOCALHOST
CTFMVS07	sawdust	z/OS	Some subsystems connected	CTFMVS07.RTP.RALE
CTFMVS08	sawdust	z/OS	No subsystems connected	CTFMVS08.RTP.RALE



# Integrated Editor

- Language Sensitive editing (COBOL, PLI, JCL, etc)
- Code Assist for COBOL, PL/I, HLASM, JCL source
  - language construct completion
  - variable completion
- Open Copybook/Include/Macro
  - Name is resolved via standard search order.
- Both Local and Remote Syntax Check or Compiles / integration with task list
  - Similar to Java, click on task list entry, opens editor on source file
- User extensible via Java
- Used by the debug Perspective to set breakpoints, etc
- Outline view of source for ease of navigation

# Remote System Explorer perspective

**Select Perspective**

- Data
- Debug
- Host Access Transformation Services
- J2EE (default)
- Java
- Java Browsing
- Java Type Hierarchy
- Remote System Explorer**
- Resource
- Service Flow Modeler
- Web
- z/OS Projects

**Remote System Explorer - IBM Rational Software Development Platform**

File Edit Navigate Search Project Run Window Help

Remote Systems Team z/OS File System Mapping

System: testmvs

Mapping Criterion	Workstation File Extension	Transfer Mode	Host Code Page	Local Code Page
**COBOL	cbl	text	IBM-037 (Inh...	CP1252 (Inhe...
**COBCOPY	cpy	text	IBM-037 (Inh...	CP1252 (Inhe...
**PLI	pli	text	IBM-037 (Inh...	CP1252 (Inhe...
**ASSEMBLE	asm	text	IBM-037 (Inh...	CP1252 (Inhe...
**OBJ	obj	binary	IBM-037 (Inh...	CP1252 (Inhe...
**LOAD	exe	binary	IBM-037 (Inh...	CP1252 (Inhe...
**CLIST	cmd	text	IBM-037 (Inh...	CP1252 (Inhe...
**JCL	jcl	text	IBM-037 (Inh...	CP1252 (Inhe...
**SIGYCLST	cmd	text	IBM-037 (Inh...	CP1252 (Inhe...
**CNTL	jcl	text	IBM-037 (Inh...	CP1252 (Inhe...
**JCL	jcl	text	IBM-037 (Inh...	CP1252 (Inhe...

**Remote System Details** Tasks

Connection testmvs

Name	User ID	Port	Connected
JES	DNET017	0	Yes
USS Files	(Inherited)	0	No
USS Shells	(Inherited)	0	No
MVS Files	(Inherited)	0	No
TSO Commands	(Inherited)	0	No

**Properties**

Property	Value
Connecti...	Some subsystems connected
Default ...	(Inherited)
Description	Dallas TESTMVS system
Host name	TESTMVS.DEMOPKG.IBM.COM

# Flexible Access to Remote Artifacts

The screenshot displays the IBM Rational Software Development Platform interface. The **Remote System Explorer** on the left shows a tree view of local and remote systems. A red box highlights the **Remote Systems** tab, and a yellow callout points to **Files on workstation** under the Local Files section. Another yellow callout points to **Files on remote z/OS** under the Local Shells section. A third yellow callout points to **COBOL and JCL** files (COBSP.cbl, COBTEST.cbl, IGYIVP.jcl) under the **WILBERT.MIXEDBAG.FILES** folder.

The main editor shows the code for **IGYTSALE.cbl** with the following content:

```

Line 1      Column 1      Insert
-----*A-1-B-----2-----3-----4-----
000001      Cbl_noadv,lib,map,nonumber,quote,sequence
000002      ISO100      Title "IGYTSALE * Main Program".
000003      Identification Division.
000004
000005      ISO120      Program-id.      IGYTSALE.
000006
000007      Author.      A. Programmer.
000008      Installation. IBM - Santa Teresa Laborat
000009      Date-written. April 1991.
000010      Date-compiled.
  
```

The **Remote System Details** pane shows the system **ctfmvs07.rtp.raleigh.ibm.com** and a **z/OS File System Mapping** table:

Mapping Criterion	Workstation File Extension	Transfer Mode	Host Code Page
**LOAD	exe	binary	IBM-037 (Inherited)
**CLIST	cmd	text	IBM-037 (Inherited)
**JCL	jcl	text	IBM-037 (Inherited)
**SIGYCLST	cmd	text	IBM-037 (Inherited)
**CNTL	jcl	text	IBM-037 (Inherited)
**LISTING	lst	text	IBM-037 (Inherited)
**OUTLIST	out	text	IBM-037 (Inherited)
**OBJS	obj	binary	IBM-037 (Inherited)
**INCLUDE	inc	text	IBM-037 (Inherited)
**MACRO	mac	text	IBM-037 (Inherited)
**COPYLIB	cpy	text	IBM-037 (Inherited)
**XML	xml	text	IBM-037 (Inherited)
**BMS	bms	text	IBM-037 (Inherited)
**JCLLIB	jcl	text	IBM-037 (Inherited)
**FILES			IBM-037 (Inherited)
COB**	cbl	text	IBM-037 (Inherited)
JCL**	jcl	text	IBM-037 (Inherited)
**JOB	jcl	text	IBM-037 (Inherited)
PLI**	pli	text	IBM-037 (Inherited)

A yellow callout points to the **\*\*FILES** section of the table, labeled **Resource (e.g., member) mapping**. The **Properties** pane at the bottom left shows attributes like **BLKSIZE** (32720) and **DSNTYPE** (DATA\_LIBRARY).

# COBOL and PL/I Content Assist

```

*ACTDDRV.CBI X
Row 100 Column 12 1 change.
-----+*A-1-E-----2-----3-----4-----
000085 * ** New Business Program XML Interfac
000086 * *****
000087 * XML Stream Byte Length
000088 * XML Stream
000089 * 1 DFHCOMMAREA.
000090 1 a-xml-interface.
000091 2 a-xml-int-len pic 9(9) binary.
000092 2 a-xml-int-txt pic x(32768).
000093 * Procedure Division using DFHCOMMAREA
000094 Procedure Division using a-xml-interf.
000095 Mainline Section.
000096 * +-----+
000097 * | Enable Exception Handler |
000098 * +-----+
000099 perform a-register-exception-hand
000100
000101
000102
000103
000104
000105

```

- ABC DIVIDE - NOT ON SIZE ERROR - END-DIVIDE
- ABC DIVIDE - ON SIZE ERROR - END-DIVIDE
- ABC DIVIDE - ON SIZE ERROR - NOT ON SIZE ERROR - END
- ABC EJECT.
- ABC ENTRY
- ABC EVALUATE - WHEN - END-EVALUATE
- ABC EVALUATE - WHEN - WHEN OTHER - END-EVALUATE

```

* procedure DIVISION using DFHCOMMAREA.
Procedure Division using a-xml-interface.
Mainline Section.
* +-----+
* | Enable Exception Handler |
* +-----+
perform a-register-exception-handler
MOVE

```

- 010 a-converter-return-code
- 010 a-error-code
- 010 a-error-description
- 010 a-error-message-number
- 010 a-exception-occurred
- 010 a-failure-data
- 010 a-failure-message-number
- 010 a-failure-response

Benefit: Developers complete code more accurately and efficiently.

# JCL Generation and Submission

The screenshot shows the IBM Rational Software Development Platform interface for a z/OS project named 'IGYIVPCL.jcl'. The project browser on the left shows a tree structure with folders like 'zCOBIVP', 'zCWS', 'zMVS1', and 'COBSAMP1'. A file 'DNET017.STEW.JCL(IGYIVPCL).jcl' is highlighted in red, with a blue arrow pointing to it and the text 'JCL generated'. The main editor window displays the following JCL code:

```

Line 1      Column 1      Insert
//-----1-----2-----3-----4-----5-----6-----7--|-----8
000001 //WKIVP JOB ,
000002 //  MSGCLASS=H,MSGLEVEL=(1,1),TIME=(,4),REGION=28M,COND=(16,LT)
000003 // *
000004 //STP0000 EXEC PROC=ELAXFCOC,CICS=,
000005 //  DB2=,
000006 //  COMP=
000007 //COBOL.SYSPRINT DD DSN=DNET017.STEW.LISTING(IGYIVP),
000008 //          DISP=SHR
000009 //COBOL.SYSLIN DD DSN=DNET017.STEW.OBJS(IGYIVP),
000010 //          DISP=SHR
000011 //COBOL.SYSLIB DD DSN=DNET017.STEW.COPYLIB,DISP=SHR
000012 //COBOL.SYSXMLSD DD DUMMY
  
```

The context menu on the right includes options like 'Cut', 'Copy', 'Paste', 'Select', 'Selected', 'Deselect', 'Filter view', 'Show all', 'Source View', 'Add to Snippets...', 'Save', 'Add Breakpoint', and 'Submit'. The 'Submit' button is highlighted with a mouse cursor.

**Benefit: Developers focused on business logic and not on writing JCL**

# Monitoring Job Output / Issuing Commands

JES2 JOB LOG -- SYSTEM MVSA -- NO

```

14.07.02 JOB07143 ---- THURSDAY, 16 JUN 2005 ----
14.07.02 JOB07143 IRR010I USERID DNET017 IS ASSIGNED TO THIS JOB.
14.07.02 JOB07143 ICH70001I DNET017 LAST ACCESS AT 13:53:30 ON THURSDAY, JU
14.07.02 JOB07143 $HASP373 WKIVP STARTED - INIT 5 - CLASS A - SYS MVSA
14.07.03 JOB07143 IEF403I WKIVP - STARTED - TIME=14.07.03
14.07.05 JOB07143 - --TIMINGS (MINS.
14.07.05 JOB07143 -JOBNAME STEPNAME PROCSTEP RC EXCP CPU SRB CL
14.07.05 JOB07143 -WKIVP STP0000 COBOL 00 792 .00 .00
14.07.05 JOB07143 IEF404I WKIVP - ENDED - TIME=14.07.05
14.07.05 JOB07143 -WKIVP ENDED. NAME- TOTAL CPU TIME
14.07.05 JOB07143 $HASP395 WKIVP ENDED
----- JES2 JOB STATISTICS -----
16 JUN 2005 JOB EXECUTION DATE
15 CARDS READ
211 SYSOUT PRINT RECORDS
0 SYSOUT PUNCH RECORDS
13 SYSOUT SPOOL KBYTES
0.05 MINUTES EXECUTION TIME
1 //WKIVP JOB ,
// MSGCLASS=H,MSGLEVEL=(1,1),TIME=(,4),REGION=28M,COND=(16,LT)

```

Remote Shell Remote System Details Tasks z/OS File System Mapping

TSO-ctfmvs07.rtp.raleigh.ibm.com

Command Shell - Running

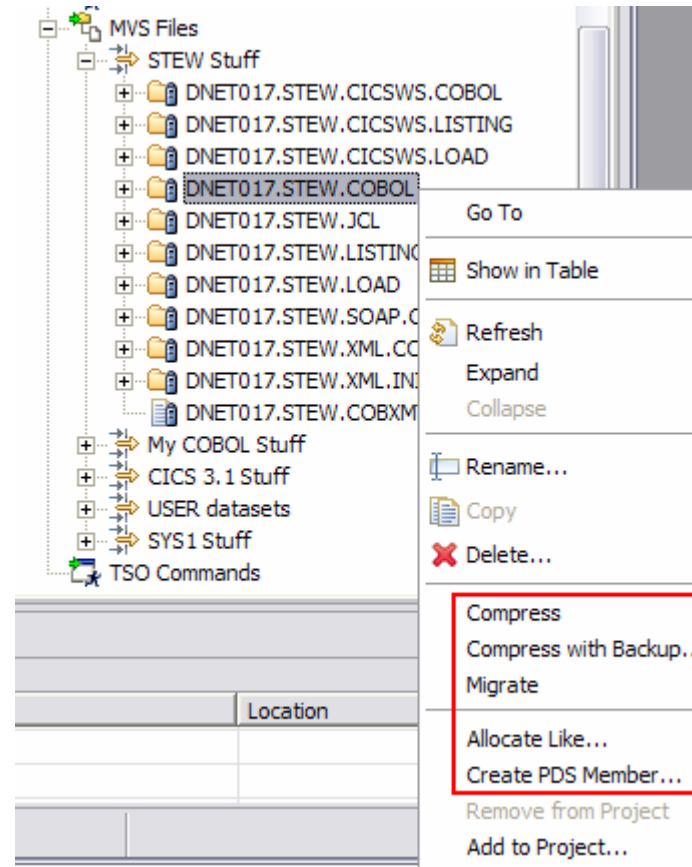
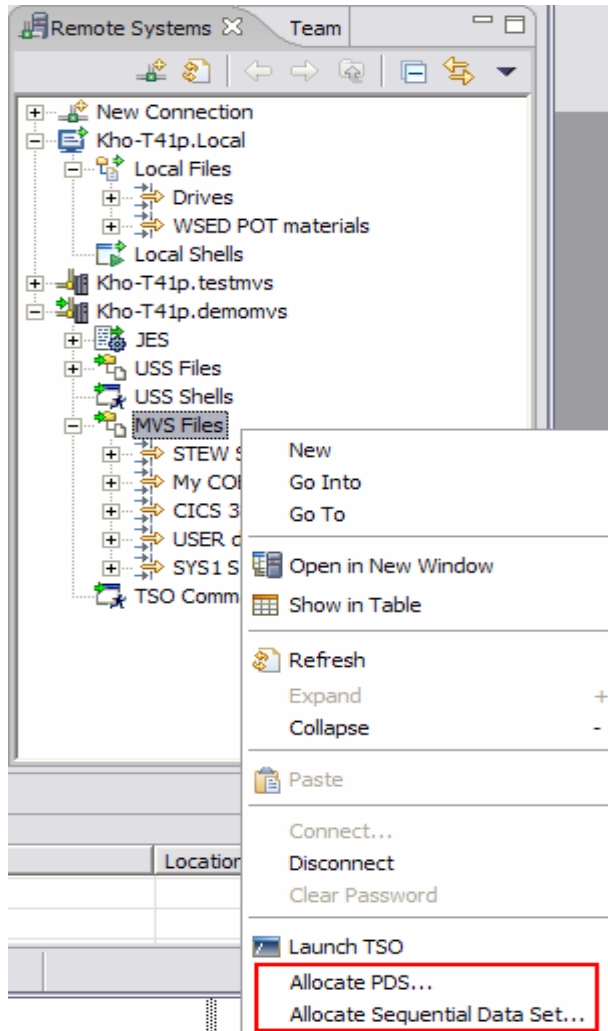
```

Specify a TSO command to run
>LISTALC
CUST.H001600.V6ROM0.SFEKSAMP
CUST.PDS.EXEC
WILBERT.FEKFRSRV.STC00146.D0000109.?
WILBERT.FEKFRSRV.STC00146.D0000110.?
NULLFILE
NULLFILE

```

**Benefit: Developers do not have to continually switch between systems**

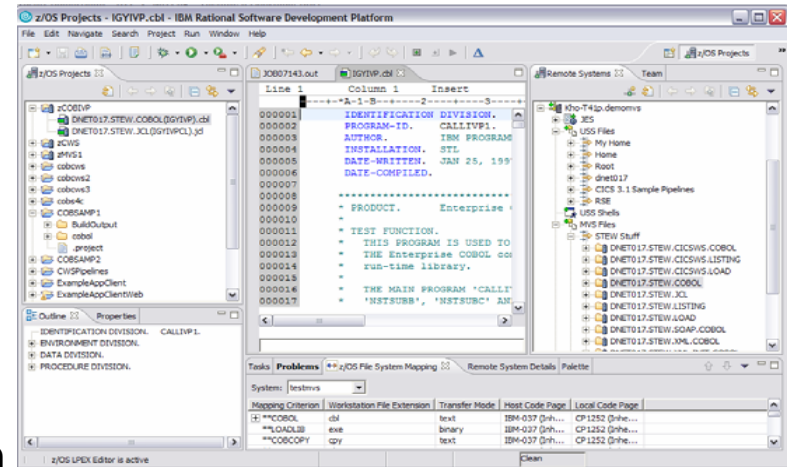
# z/OS Dataset Management



# COBOL and PL/I Remote Edit/Compile/Debug

- Comprehensive state-of-the-art facilities for editing, build and debugging existing or new COBOL or PL/I programs
- Remote edit/compile/debug: keep z/OS artifacts on the mainframe and avoid costly downloads/uploads

- ✓ Powerful smart editing
- ✓ Graphical navigation
- ✓ Syntax check
- ✓ Control of remote compile
- ✓ Compile feedback
- ✓ Graphical debugger on workstation



- Program executes on mainframe (CICS, IMS, Batch, Stored Procedures)

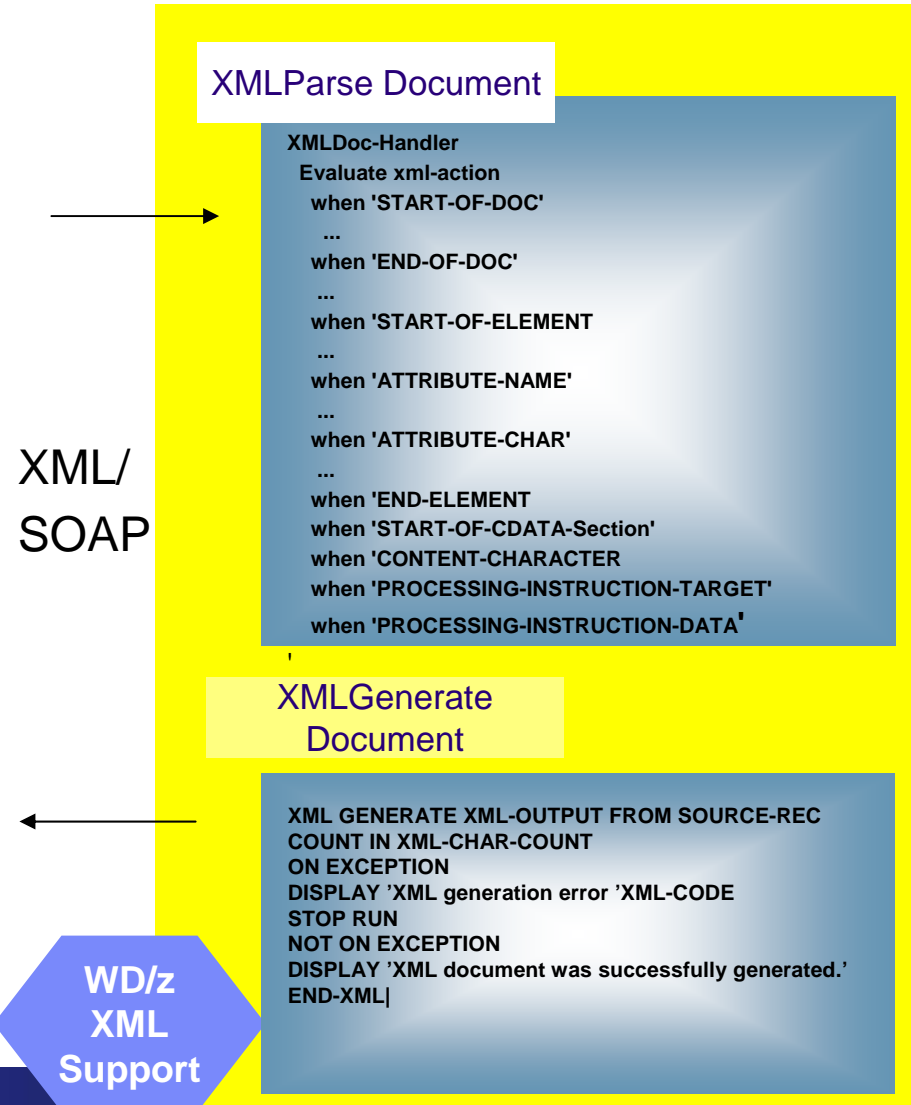
- Exploits IBM Software Development Platform
  - ✓ Task manager, Projects/Perspectives , etc
- Live host connectivity (TSO Commands, Job queue mgmt, etc)



# Using Enterprise COBOL to service-enable z/OS

CICS/IMS/Batch/DB2 COBOL

- **What's new ...**
  - XML Language based generation from COBOL data structure
    - XMLGenerate Verb
  - WebSphere EJB support
  - DB2 V8 preprocessor
  - CICS preprocessor
- High speed XML Sax based parsing
  - XMLParse
  - XMLGenerate
  - Related verbs
- Object Oriented Support for Java COBOL Interoperability
- Unicode support
- Similar XML parsing support available in Enterprise PL/I



# WDz SOA or ESB *Lite* tools

- **Enable Web Services and XML access to existing CICS and IMS transactions**

- **XML and Web Services for the Enterprise (XSE)**
- Quickly maps existing COBOL interfaces to XML and Web Services.
- No code changes for the COBOL application
- Supports IMS, CICS BMS (terminal-based) & CICS commarea applications

- **Model and deploy complex CICS processes to support SOA**

- **Service Flow Modeler (SFM)**
- Aggregate CICS transactions into high-level business processes through visual (drag n drop) modeling
- Highly optimized CICS COBOL runtime to increase overall throughput
- Supports COBOL commarea-based applications and terminal-based applications

- **Recent Announcement of Note: CICS Service Flow Feature**

- *0 Cost feature*
- *Design Web Services / XML flows in WDz*
- *Consists of:*
  - *CICS Service Flow Runtime*
  - *Limited WDz licensed for:*
    - **Service Flow Modeler**
    - **XML and Web ServicesXML**
- *Positioned for all CICS V3 customers needing to simply or programmatically integrate via Web Services*

# XML Services for the Enterprise (XSE)

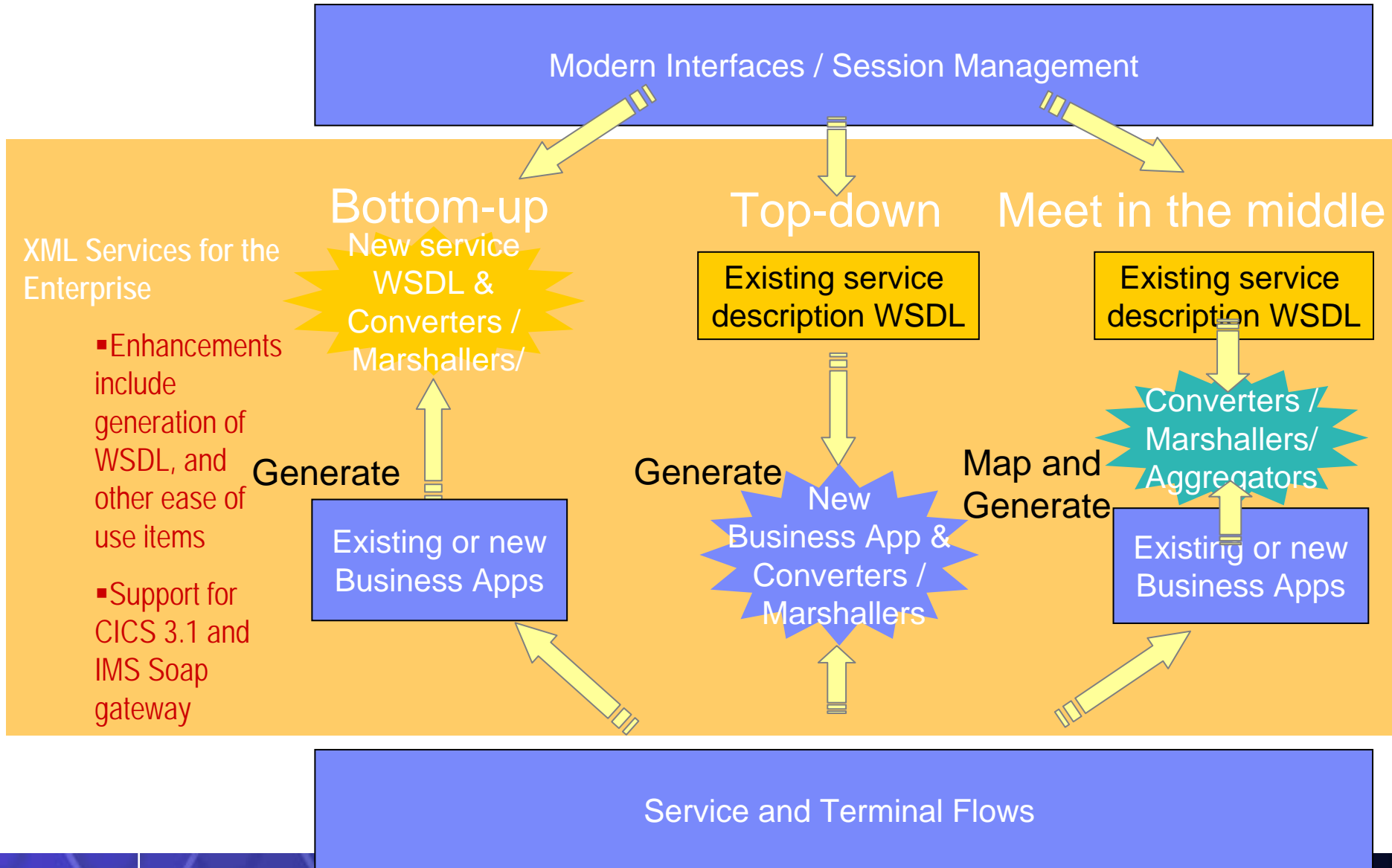
- Provide tools to adapt COBOL based applications
  - Process and produce XML messages
    - Web Services
    - Participate in a larger system that uses XML
- Web Services Enablement wizard
  - Generate Web Service interface from existing COBOL application
  - Bottom-up approach since COBOL at the bottom (base) of the creation process
- XML to COBOL Mapping tool
  - Map existing Web Service interface or XML to existing COBOL app.
  - Meet-in-the-middle since Web Services/XML definition “meets” or maps to the existing COBOL interface
- Batch processor
  - Runs unattended or in batch mode using the bottom-up approach

# XML and Web Services Enablement

## Enables COBOL-based applications to consume and produce XML messages

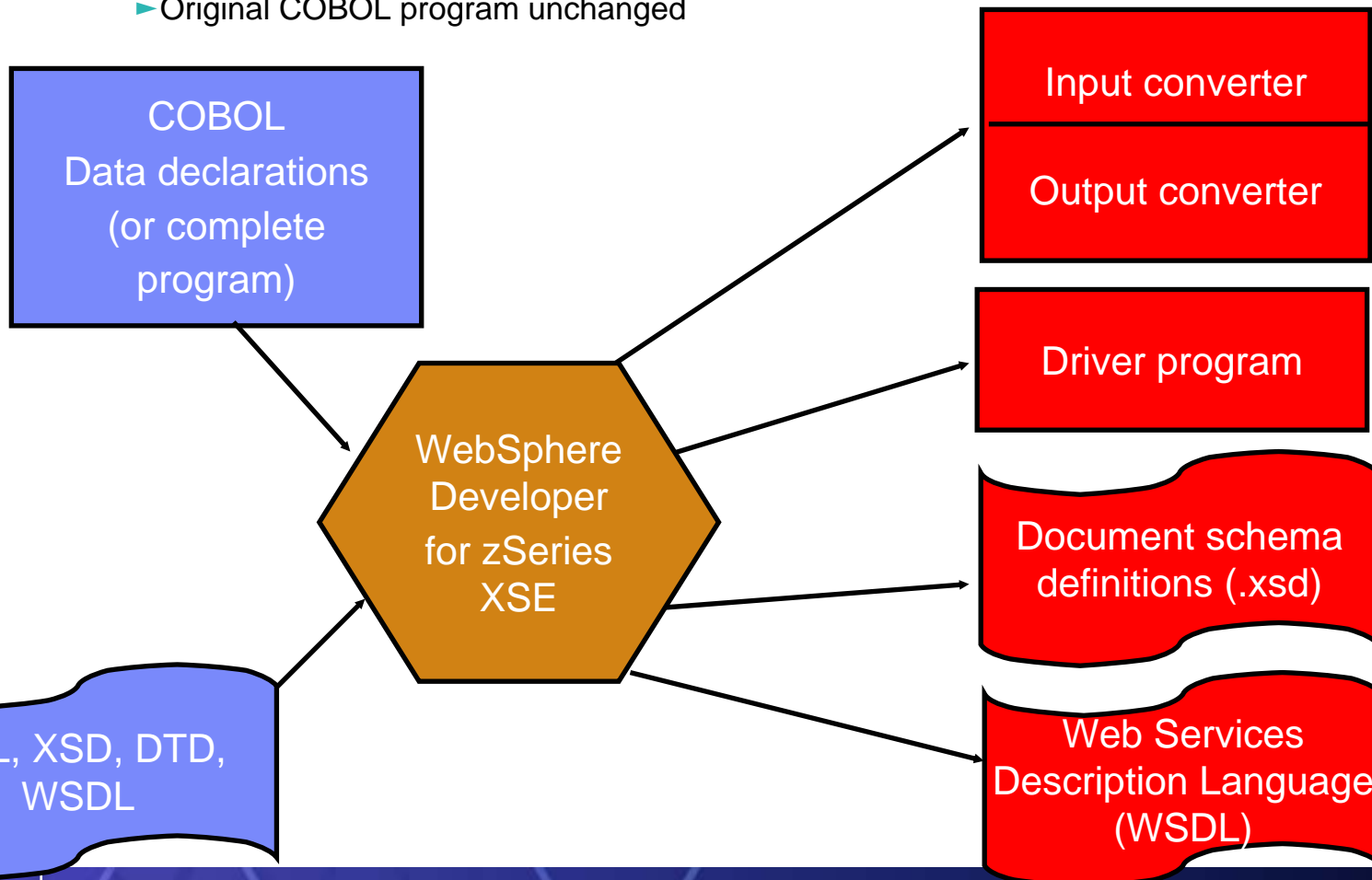
- Leverages XML parsing capabilities of IBM Enterprise COBOL V3.1
- Creates COBOL converter programs
  - ▶ Inbound to convert XML messages into native COBOL data
  - ▶ Outbound to convert native COBOL data into XML messages
- Creates COBOL driver program
  - ▶ Illustrate the invocation of converters
  - ▶ Illustrate the interaction with existing application
- Creates WSDL that describes COBOL based service
- Enables communication with XML based systems
- Batch interface to Web Services Enablement Tool for COBOL
  - Create COBOL adapters and WSDL via command line

# Web Service Enablement Styles (XSE)



# Mapping COBOL Data

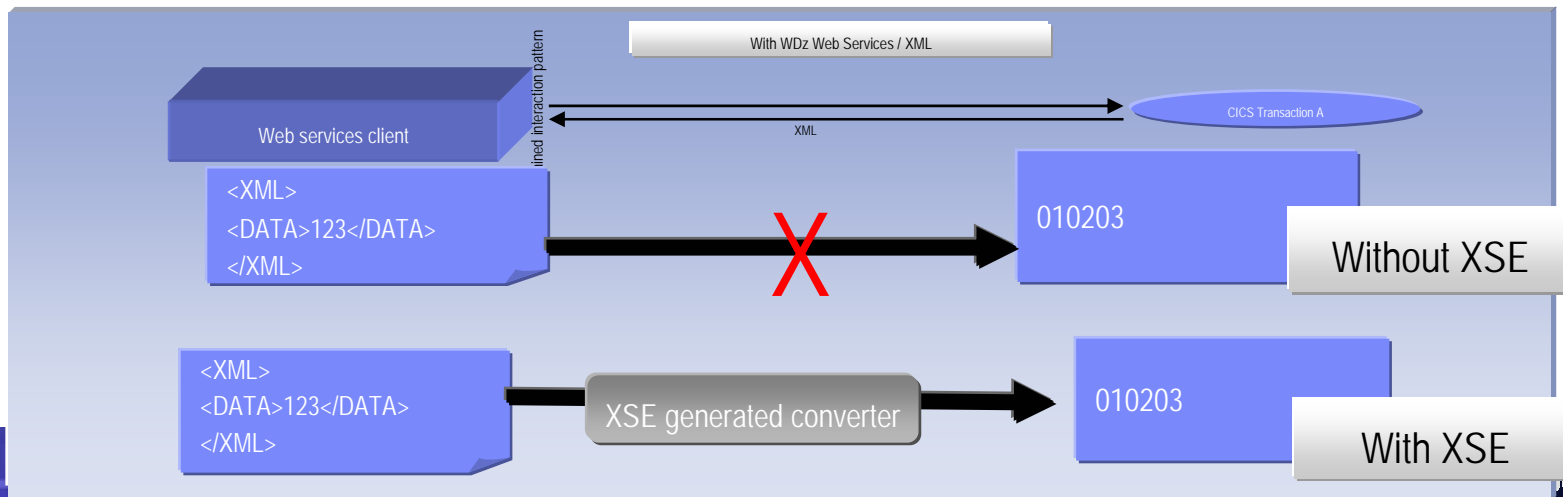
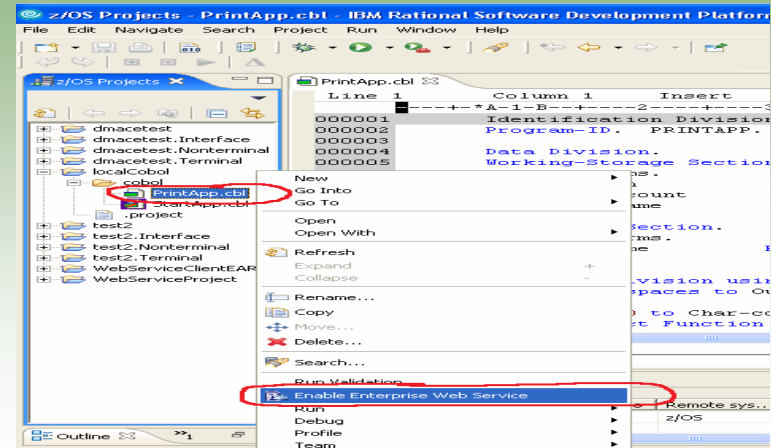
- Enables COBOL-based applications to consume and produce XML messages
  - ▶ Original COBOL program unchanged



# What is XML Services for the Enterprise (XSE)

## *XSE in WebSphere Developer for zSeries*

- Most rapid building of Web services from existing CICS applications
  - Single CICS and IMS transactions enabled for Web Services
  - Supports IMS Message Queue, CICS Commarea and new Channels/Container based applications
  - Rapid generation of WSDL, CICS WSBind, and Adapter generation eliminating complex hand coding of XML to/from language conversions
  - Includes complete Web Services Test and Java generation environment



# Converter Types Supported

- Batch, TSO and USS
- IMS SOAP Gateway
- SOAP for CICS
- Web Services for CICS

← *New!*

Enable Web Service Wizard

**Generation Options**  
Specify generation options for the Web Services enablement artifacts.

XML Converter Options | WSDL and XSD Options

Specify type  
Converter type: Batch, TSO and USS  
Batch, TSO and USS  
IMS SOAP Gateway  
SOAP for CICS  
Web Services for CICS

Specify identification  
Program name prefix: StartApp  
Author name: WD4Z  
Business program name: StartApp

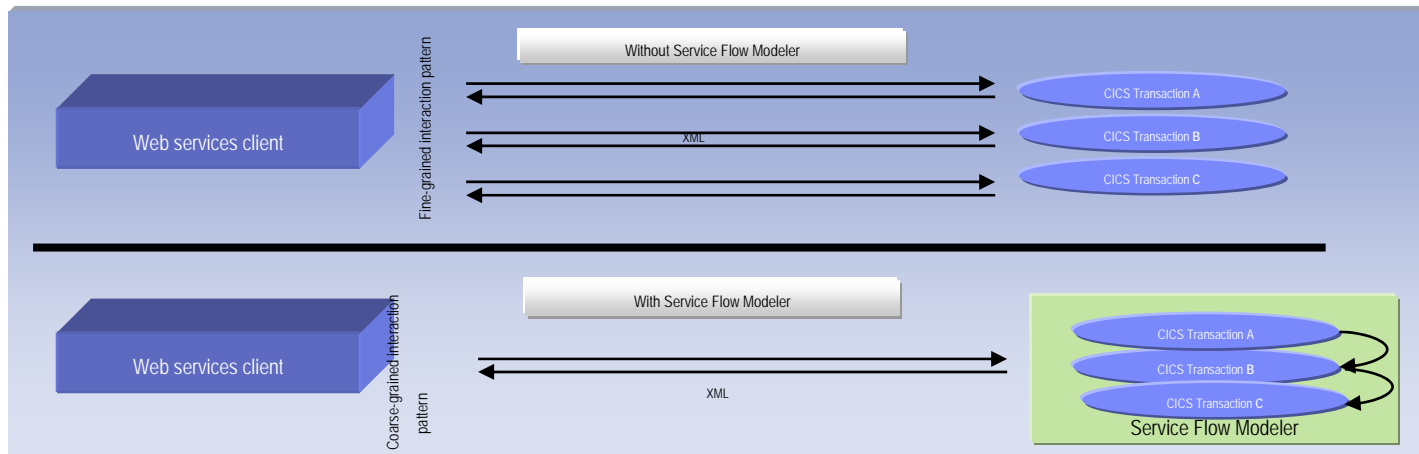
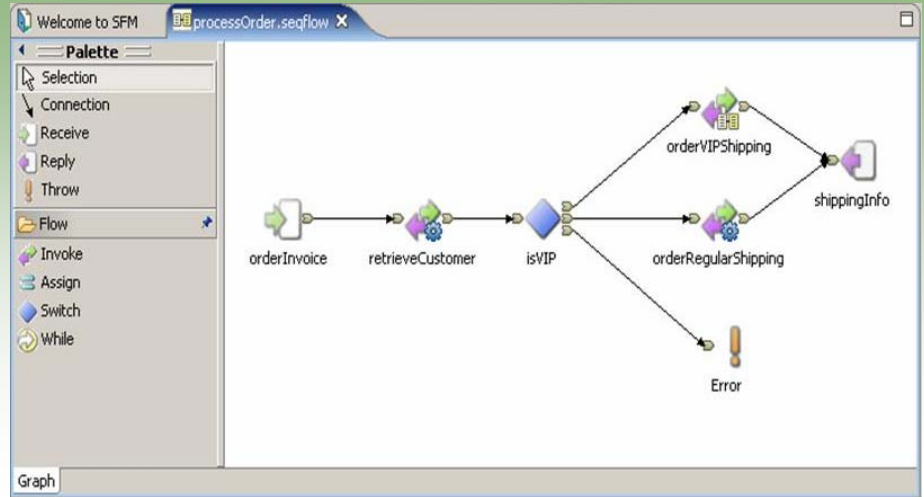
Specify character encodings  
Inbound code page: 1140 USA, Canada, etc. Euro Country Extended  
Host code page: 1140 USA, Canada, etc. Euro Country Extended  
Outbound code page: 1140 USA, Canada, etc. Euro Country Extended



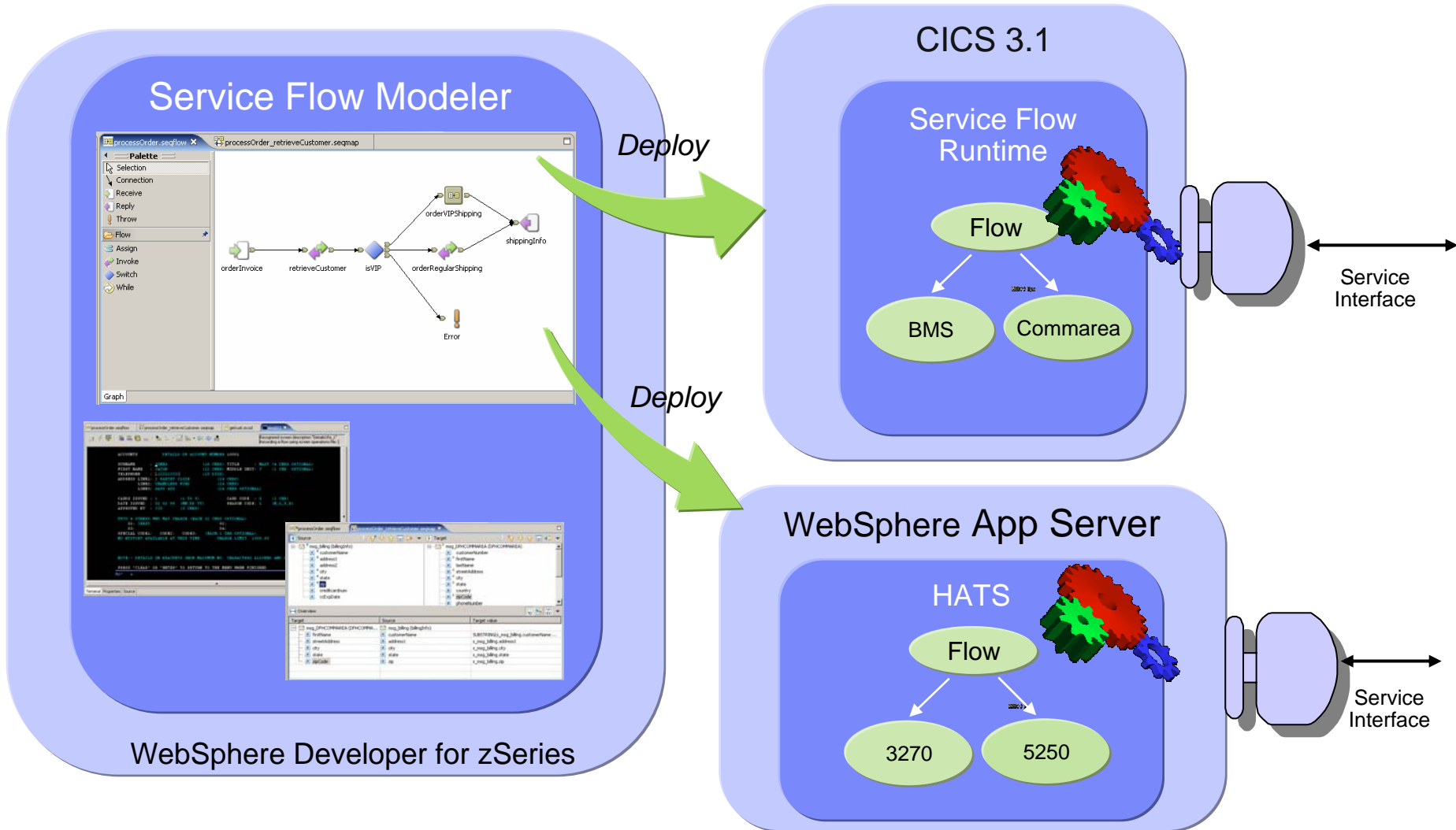
# What is Service Flow Modeler?

## *New Feature!* Service Flow Modeler in WebSphere Developer for zSeries

- **Builds Web services from existing CICS applications**
  - Aggregates multiple CICS transactions into high-level business processes through visual modeling
  - Supports CICS BMS (terminal-based) applications & CICS commarea applications
  - Highly optimized CICS runtime supporting Web services and XML interfaces

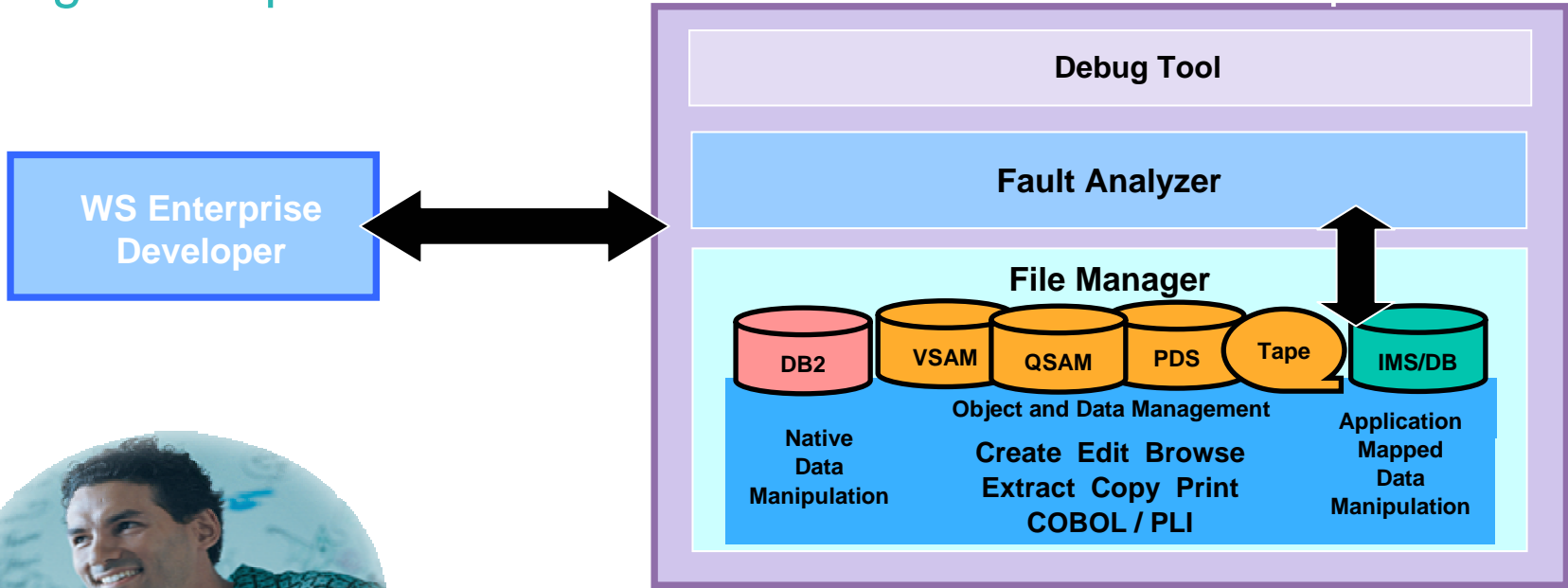


# WDz's Service Flow Modeler Deployment Options



# Test and Problem Determination

## Integration speeds time to market



### Benefits:

- **Simplify development of zSeries test cases**
  - ▶ Data creation for DB2, IMS/DB, VSAM, and QSAM
  - ▶ Extract and load
- **Reduced deployment complexity**
  - ▶ Production data validation and creation
- **Common environment**
  - ▶ Reuse of skills across e-bus and traditional applications

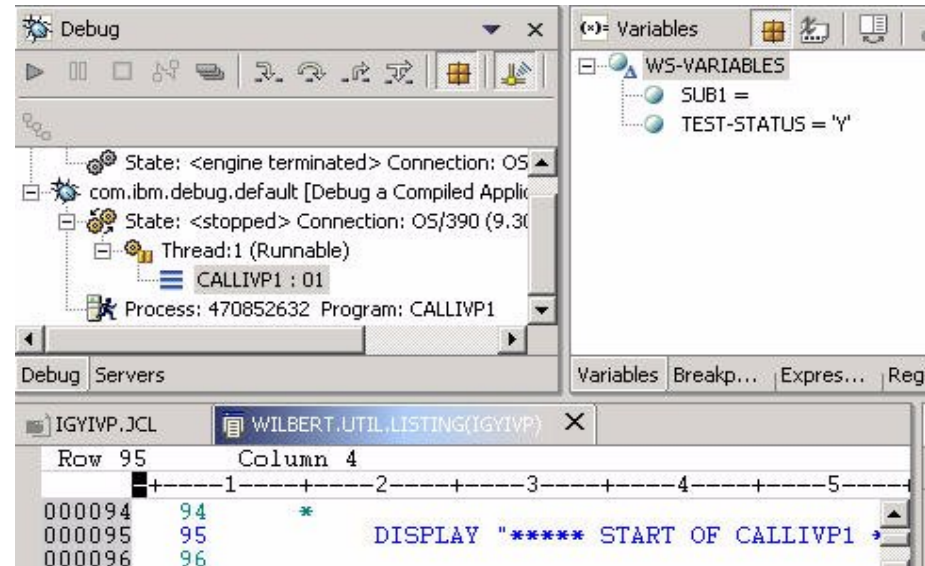
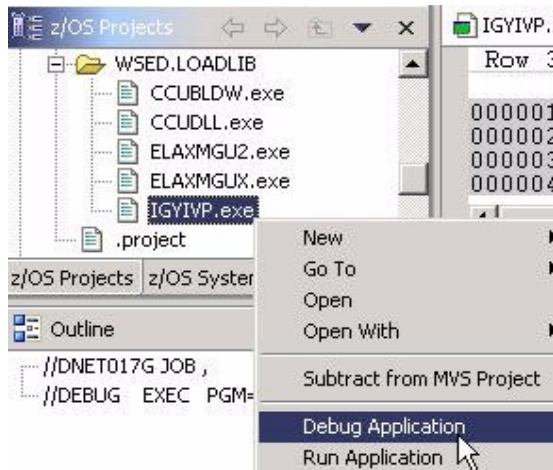
Systems Support Manager

Systems Programmer

Data Center Operator

# WebSphere Developer based Debugging

Same Debug Perspective



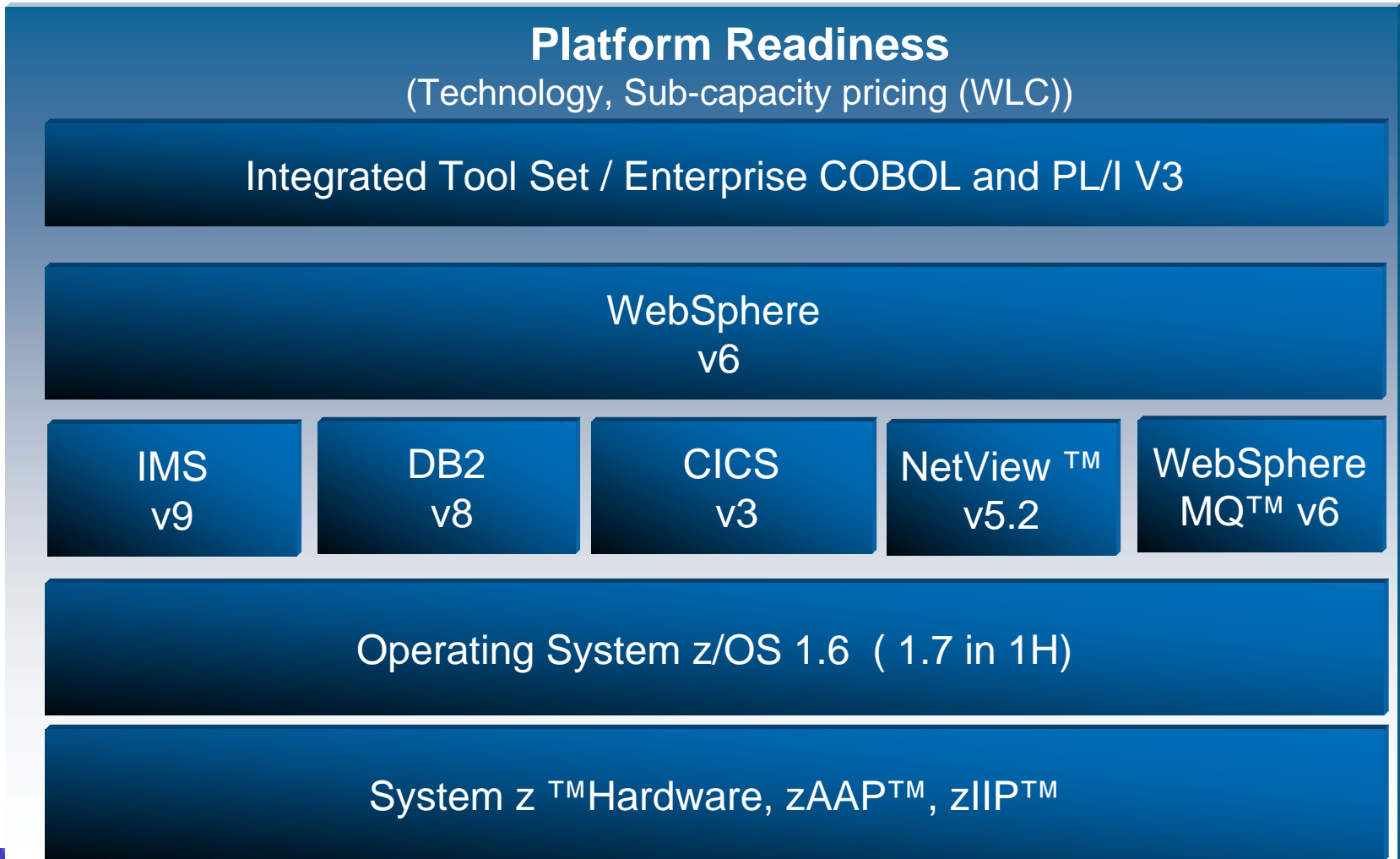
**Benefit: Consistent debugging environment for COBOL, PL/I, Java**

# Gartner: Best Practices for Mainframe SOA

- **Act tactical, think strategic**
- **Evaluate tools that provide good microflow orchestration**
- **Create services that utilize function from across existing application boundaries.**
- **Build a reuse culture and technology infrastructure.**
- **Work with operations to create management/performance-monitoring support.**
- **Use code understanding/inventory/restructuring tools to improve service granularity.**
- **Define the role of the mainframe in future application architecture.**

# IBM zSeries Software Solutions

*Platform Readiness is Key*



# End Game: A Single Point of Access for People and Projects: *Moving to The Developer Dashboard*

Simplify organizational management in mixed workload and distributed environments

