



IBM Software Group

WebSphere Studio Enterprise Developer V5.1.1

WebSphere. software



veroquiblier@fr.ibm.com

1

© 2003, 2004 IBM Corporation

Agenda

- **Positioning**
- WebSphere Studio Enterprise Developer
 - z/OS Application Development Tools
 - Basic J2EE Connector Architecture support
 - XML Enablement for COBOL
- Summary

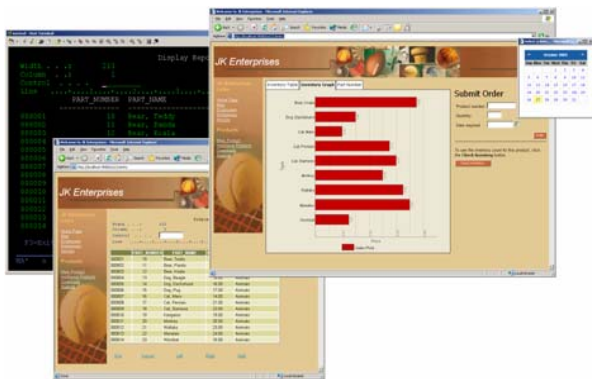


Enterprise Transformation

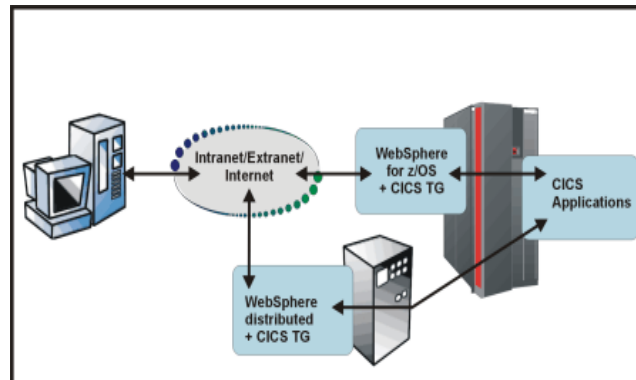


- Create new business value from existing IT systems
- Transform business-critical legacy processes into reusable, shareable business components
- Integrate traditional COBOL and PL/I applications with new Java applications into an efficient mixed-workload environment
- Leverage existing enterprise skills and improve developer productivity

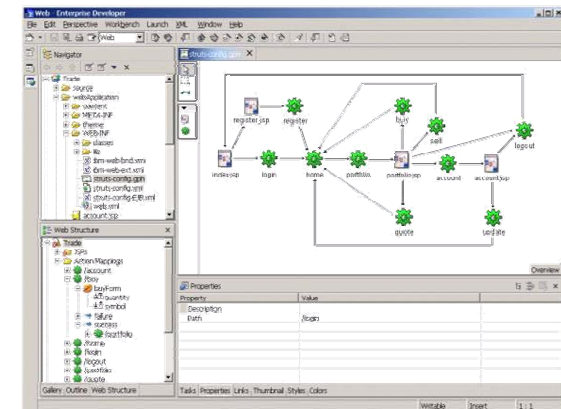
Transformed user interface and workflow for quick return on investment



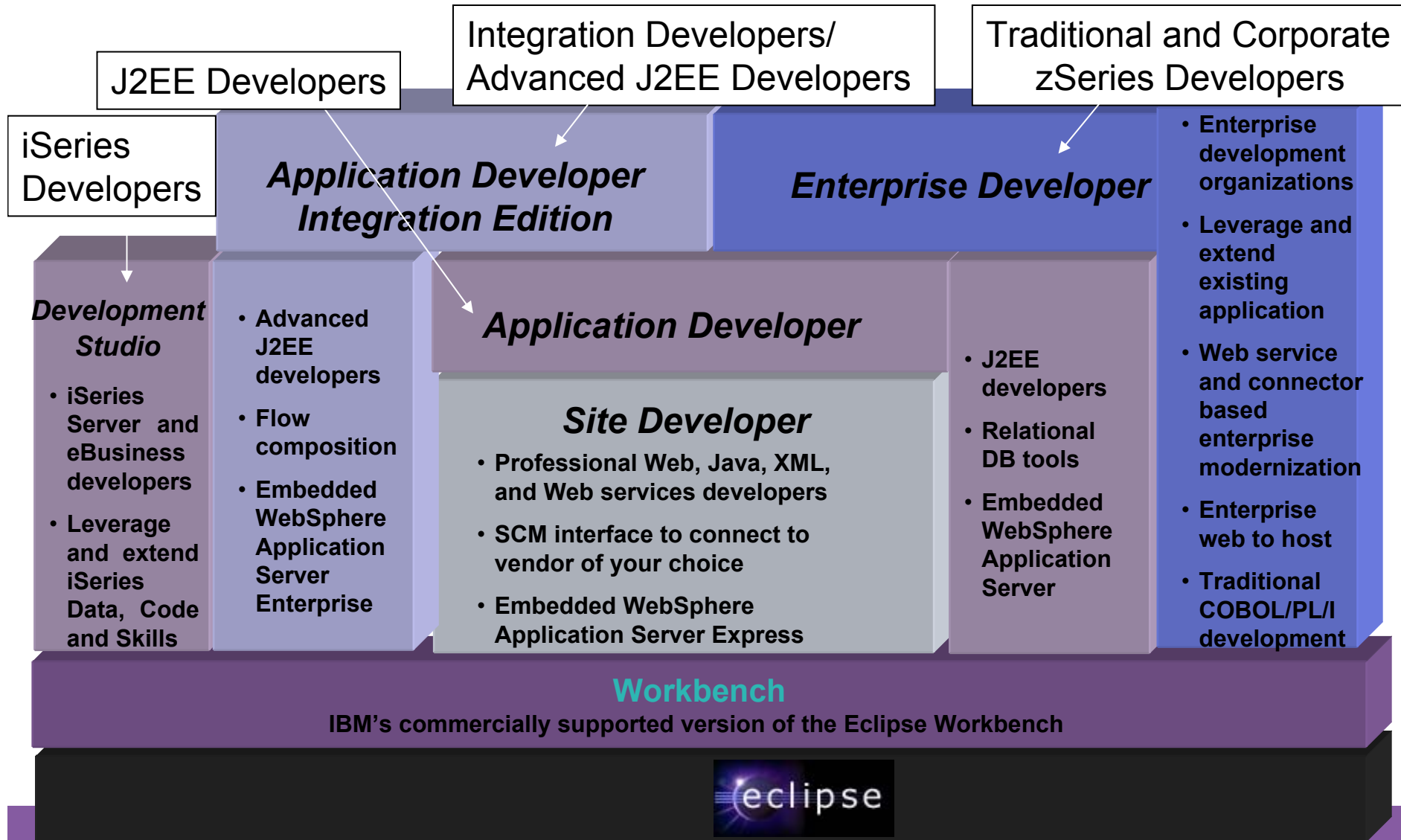
Transformed business processes using Web services and Java connectors



Transformed applications and data with tools for discovery, development and deployment of legacy assets



The WebSphere Studio v5.1 family

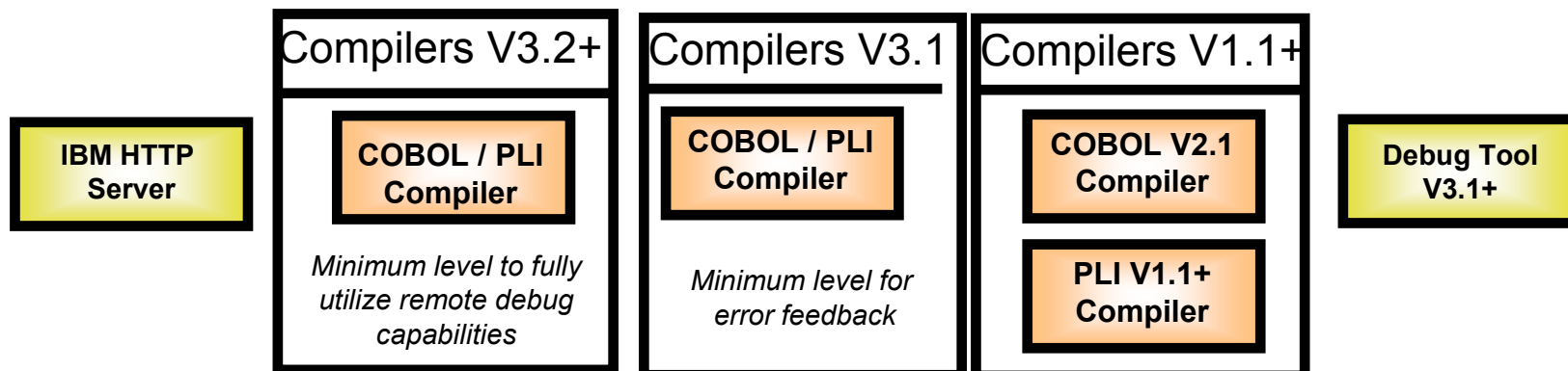
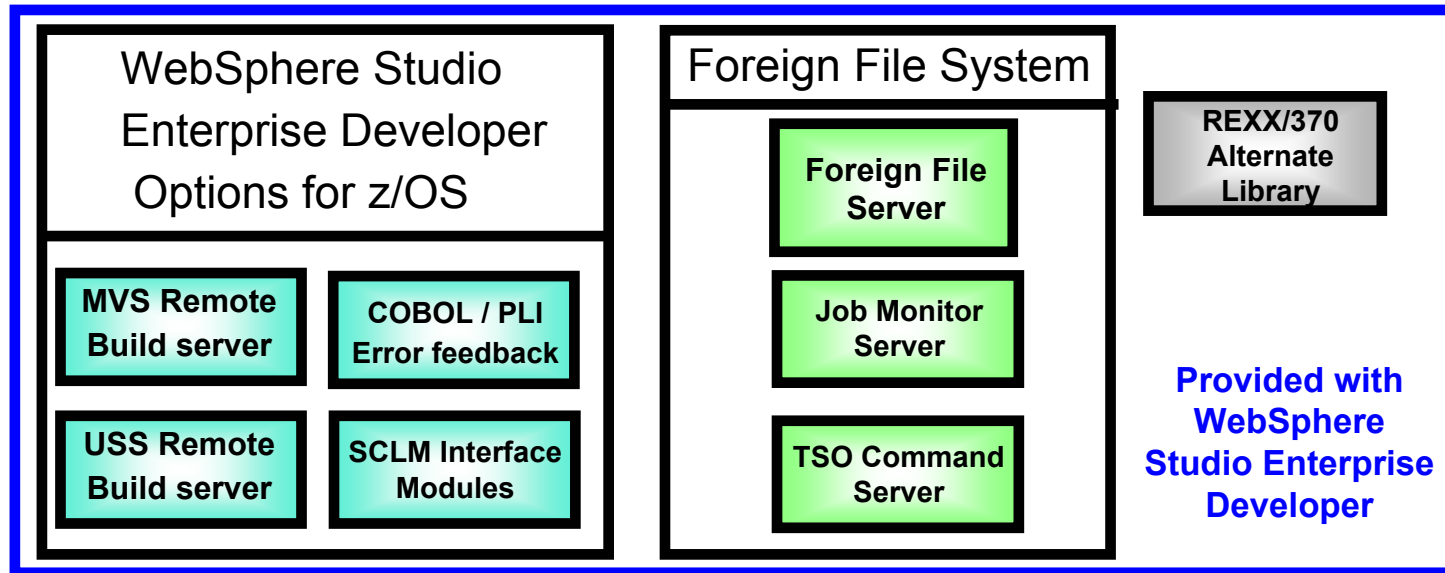


z/OS Application Development

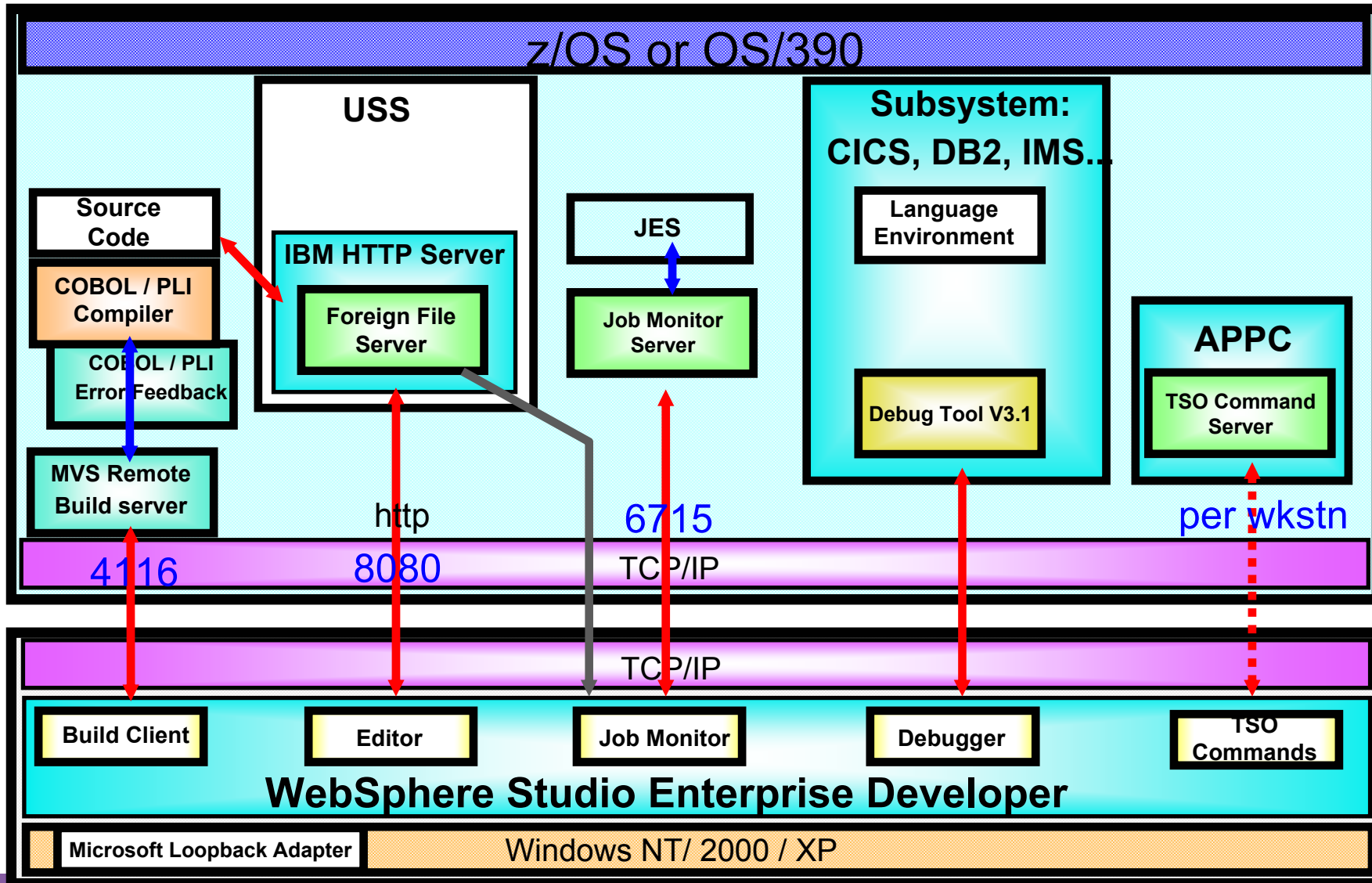
- WebSphere Studio based development environment for z/OS
 - Shared development experience and facilities
 - Perspectives, Views, Workspace
 - Web, Java, J2EE, COBOL, PL/I, HLASM
 - Access to commonly used z/OS facilities
 - Edit, compile, debug
 - **Remote** or local
 - Dataset management
 - Job queue monitoring and management
 - Service Oriented Architecture (SOA) enablement
 - Web and Enterprise Services
 - J2EE Connector Architecture (JCA)
 - XML Enablement for COBOL



Host Components



Host - Workstation Communication

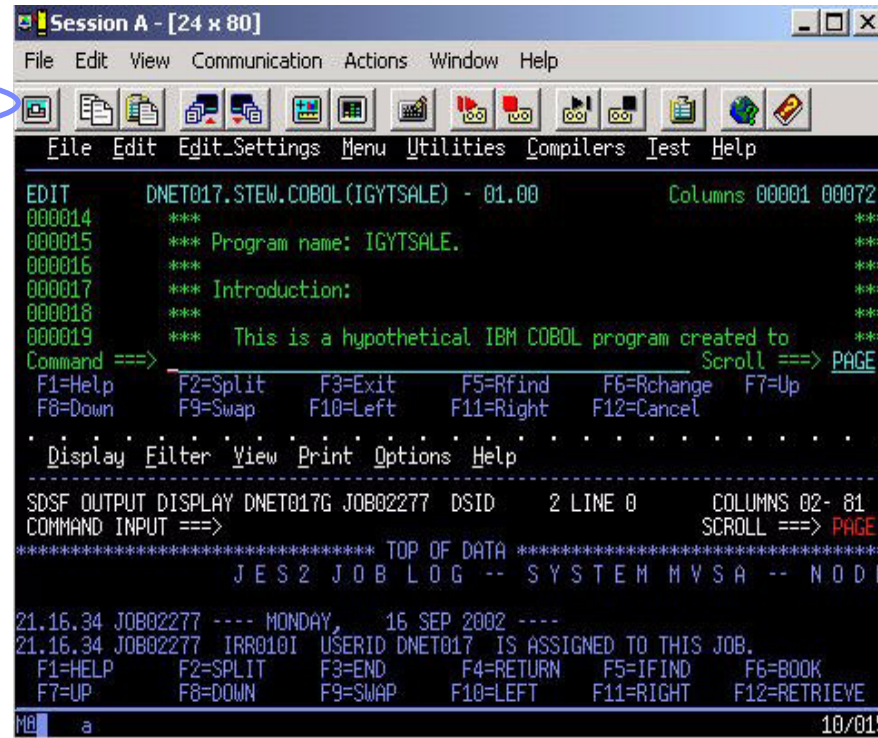
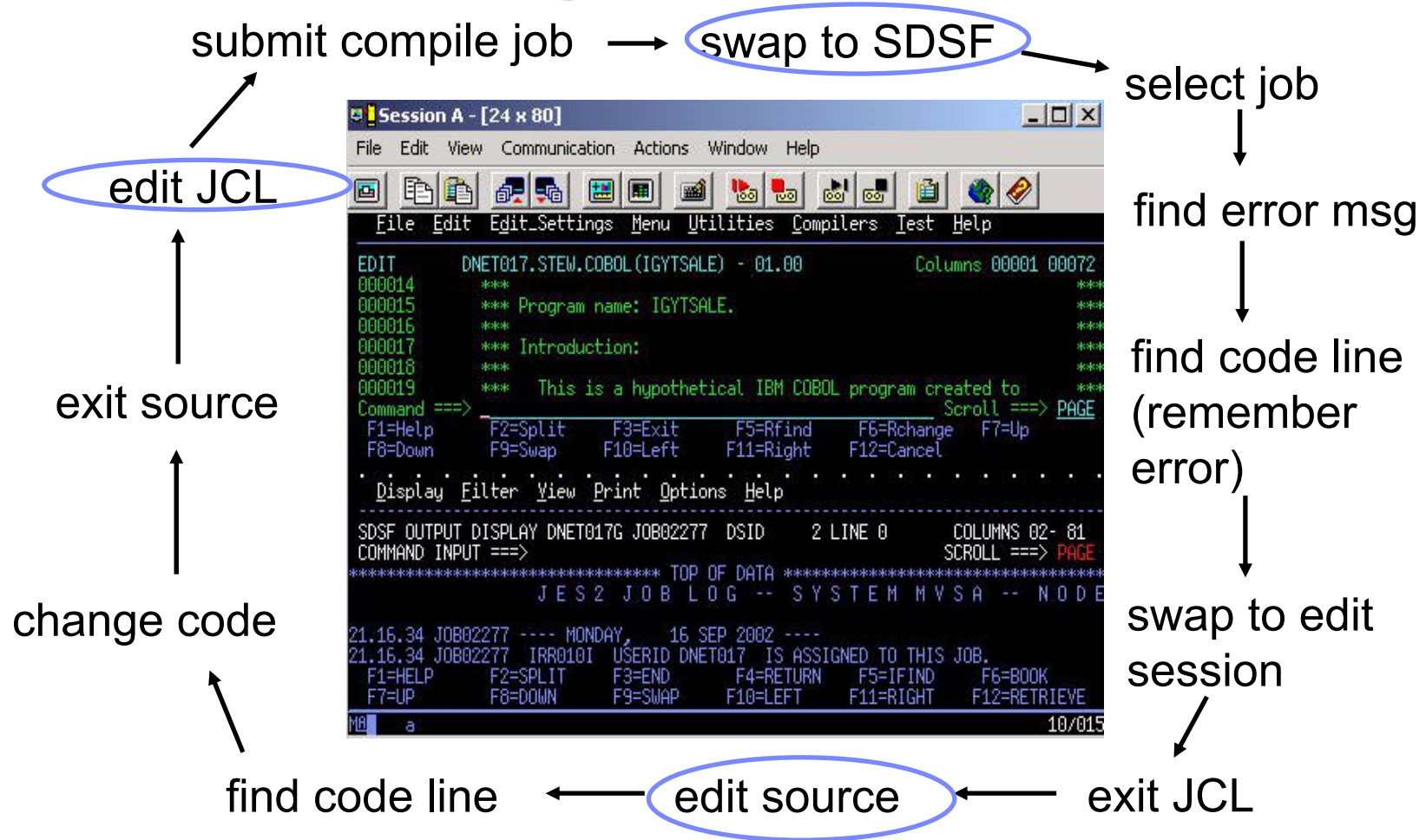


Agenda

- Positioning
- WebSphere Studio Enterprise Developer
 - **z/OS Application Development Tools**
 - Basic J2EE Connector Architecture support
 - XML Enablement for COBOL
- Summary

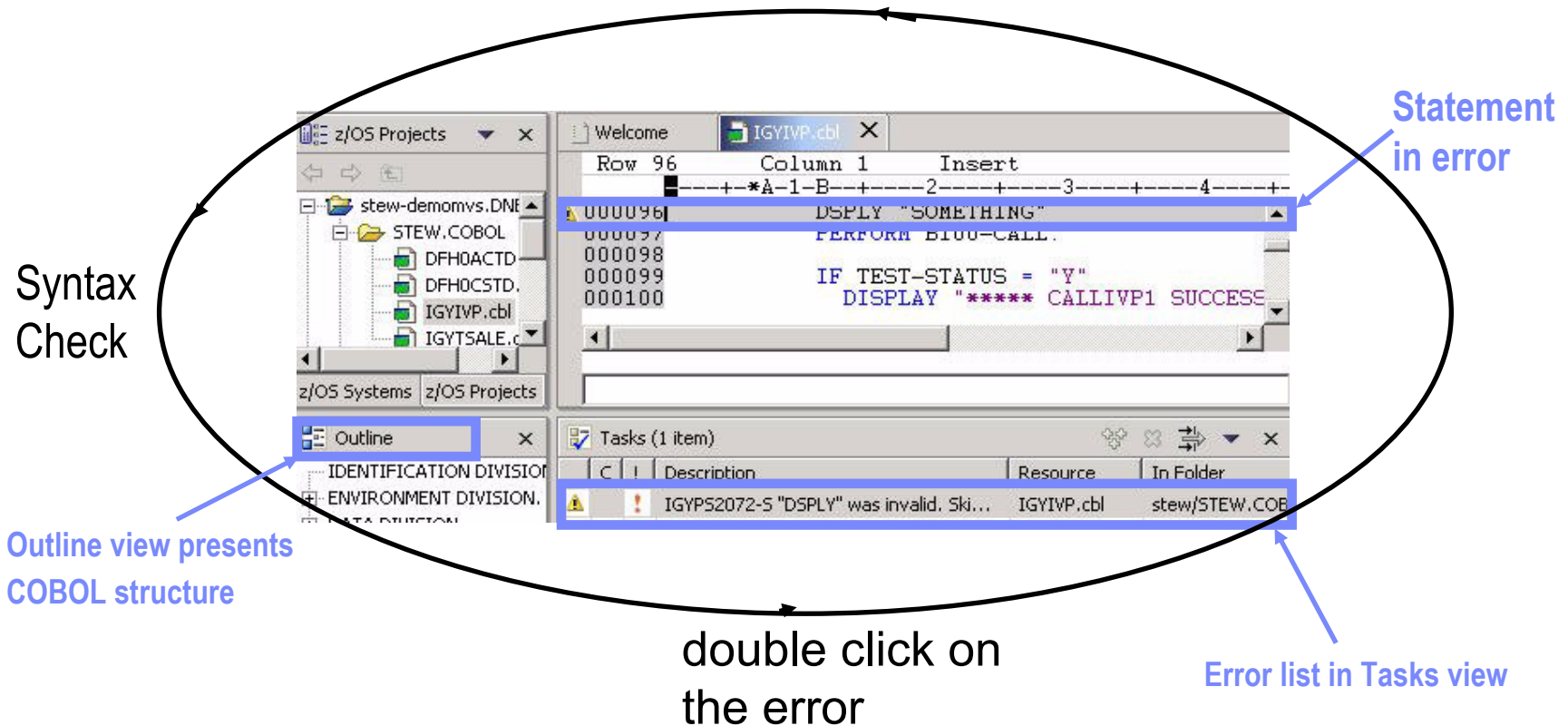


ISPF based Development



WebSphere Studio based Development

edit source

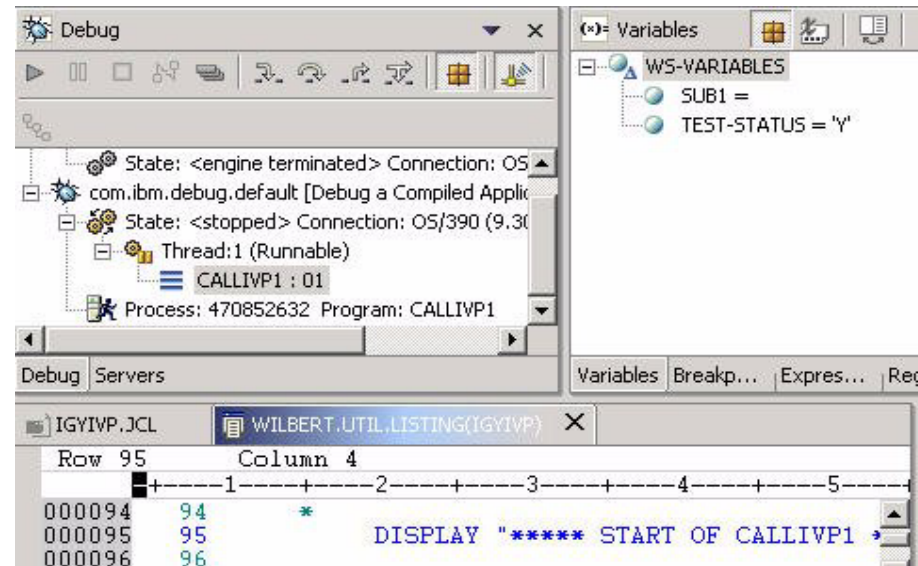
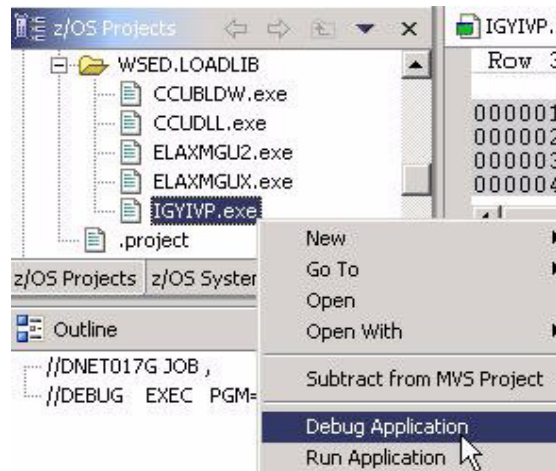


Benefit: Simplified development for COBOL and PL/I on a common development environment



WebSphere Studio based Debugging

Same Debug Perspective



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



z/OS Systems Perspective

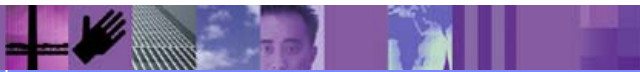
The screenshot displays the 'z/OS Systems' perspective in IBM WebSphere Studio Enterprise Developer. The interface includes a menu bar (File, Edit, Navigate, Search, Project, Run, Window, Help), a toolbar, and several views:

- Defined systems:** A red box highlights the 'z/OS Systems' folder in the left-hand tree view.
- Directories available:** A red box highlights a sub-tree under 'st1' containing 'demomvs' and 'DNET017', with 'DNET017' expanded to show sub-directories like 'DNET017.E'.
- Directory definitions:** A red box highlights the 'Data Set Name Level View' window, which shows the path 'DNET017.*'.
- Dataset mappings:** A red box highlights the 'Host Data Set and Workstation Mapping View' window, which contains a table mapping workstation file extensions to host data sets and transfer modes.

The 'Host Data Set and Workstation Mapping View' table is as follows:

Workstation File Ext...	Host Data Set	Transfer Mode
cbl	**COBOL	text
cpy	**cobcopy	text
pli	**PLI	text
asm	**ASSEMBLE	text
obj	**obj	binary
exe	**load	binary
cmd	**clist	text
jcl	**jcl	text
cmd	**sigyclst	text
jcl	**cntl	text
lst	**listing	text
out	**outlist	text
inc	**include	text
mac	**macro	text
cpy	**copylib	text
obj	**objs	binary
jcl	**jcllib	text

At the bottom left, a 'Properties' window is visible, showing details for the selected system, such as 'Host code ... IBM-037' and 'System name demomvs.de'.



z/OS Projects Perspective

MVS Project

MVS Directory

Outline view

The screenshot displays the IBM WebSphere Studio Enterprise Developer interface in the z/OS Projects perspective. The top menu bar includes File, Edit, Navigate, Search, Project, Launch, Run, Window, and Help. The project browser on the left shows a project named 'stew-demomvs.dnet01' containing a directory 'STEW.COBOL' with various COBOL source files like DFHOACTD.cbl, DFHOCTSD.cbl, IGYIYP.cbl, etc., and a file 'TESTXML.cbl'. The main editor window shows the source code of 'XMLSAMP.cbl', which includes a COBOL program header and an XML document. The z/OS Job Monitor window at the bottom right displays a table of active jobs.

Job ID	Job Name	Owner	Hold Status	Exec Node
demomvs.f1	*	DNET017	*	*
demomvs.f2	XML*	DNET017	*	*
stplex4b.f1	*	WILBERT	*	*

JLPEX editor

z/OS Job Monitor view

z/OS Commands view

COBOL and PL/I Content Assist

```

*ACTDDRV.cbl 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

```

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

JCL generated

z/OS Projects - DNET017.STEW.JCL(DFSIVA34).jcl - IBM WebSphere Studio Enterprise Deve...

File Edit Navigate Search Project Run Troubleshooting Window Help

z/OS Projects

- PhoneBookProxy.ja
- IMSPhoneBookService
- J2C Tool Plugin Import Service
- myIMSPhoneBookService
- PhoneBookUDF
- zIMSCOB
 - DNET017.IMS.COBOUL
 - DFSIVA34.cbl
 - IAP26A.cbl
 - IAPMDI26.cbl
 - DNET017.STEW.JCL(DFSIVA34).jcl**

z/OS Systems

```
Row 1      Column 1      Insert
-----+-----1-----+-----2-----+-----3-----+-----4-----
000001 //GEN001 JOB ,
000002 //  MSGCLASS=H,TIME=(,4),REGION=28M,COND=
000003 //  JCLLIB ORDER=DNET017.STEW.JCL
000004 //*
000005 //DELLIST EXEC PGM=I
000006 //SYSPRINT DD SYSOUT
000007 IF LASTCC = 8 THEN
000008 DELETE DNET017.IDE
000009 IF LASTCC = 8 THEN
000010 /*
000011 //STP0000 EXEC PROC=
000012 // DB2=,
000013 // COMP=
000014 //COBOL.SYSPRINT DD
000015 //                               DISP=SHR
```

z/OS Projects

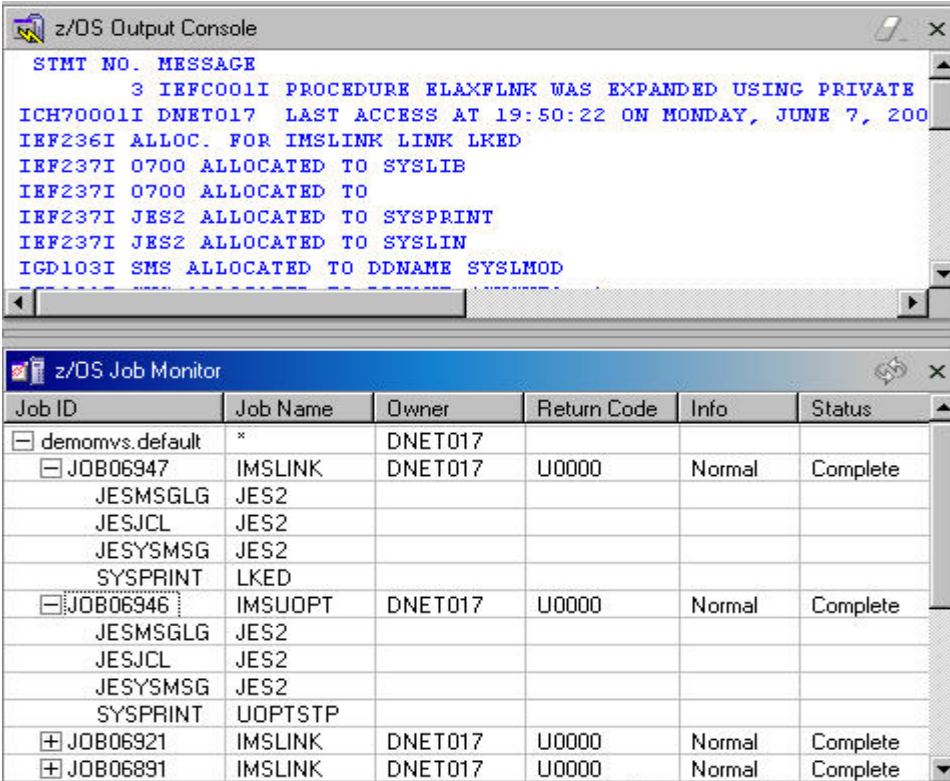
- myIMSPhoneBookService
- PhoneBookUDF
- zIMSCOB
 - DNET017.IMS.COBOUL
 - DFSIVA34.cbl
 - IAP26A.cbl
 - IAPMDI26.cbl
 - DNET017.STEW.JCL(DFSIVA34).jcl**

- Submit
- Remove from MVS Project
- Add To Another MVS Project ...
- Move To Another MVS Project ...
- Browse
- Open
- Open With

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

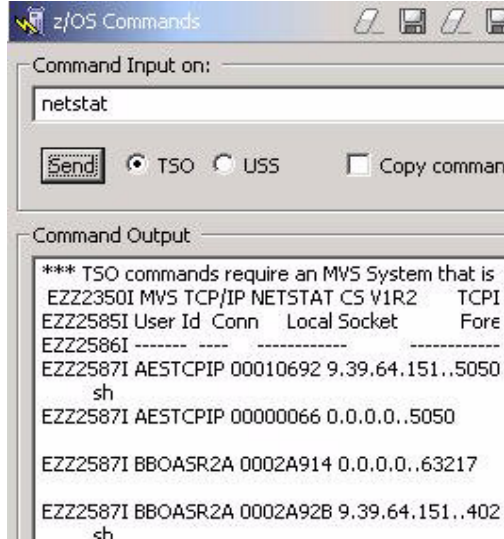


Monitoring Job Output / Issuing Commands



The image shows two windows from the WebSphere Studio Enterprise Developer interface. The top window, titled "z/OS Output Console", displays a list of system messages (STMT NO. MESSAGE) including job allocation and completion details. The bottom window, titled "z/OS Job Monitor", displays a table of job details.

Job ID	Job Name	Owner	Return Code	Info	Status
demomvs.default	*	DNET017			
JOB06947	IMSLINK	DNET017	U0000	Normal	Complete
	JESMSG LG	JES2			
	JESJCL	JES2			
	JESYSMSG	JES2			
	SYSPRINT	LKED			
JOB06946	IMSUOPT	DNET017	U0000	Normal	Complete
	JESMSG LG	JES2			
	JESJCL	JES2			
	JESYSMSG	JES2			
	SYSPRINT	UOPTSTP			
JOB06921	IMSLINK	DNET017	U0000	Normal	Complete
JOB06891	IMSLINK	DNET017	U0000	Normal	Complete



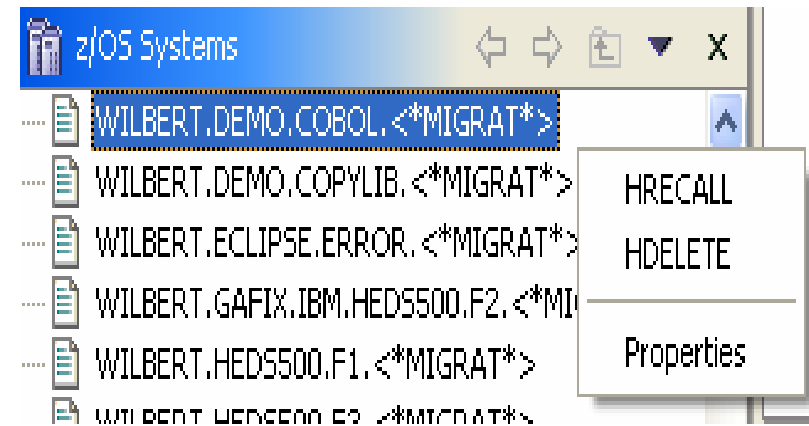
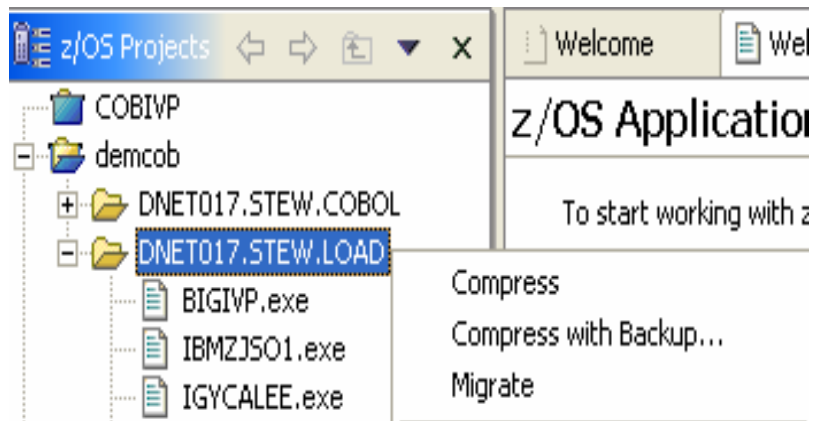
The image shows the "z/OS Commands" window. The "Command Input on:" field contains the text "netstat". The "Send" button is visible, along with radio buttons for "TSO" (selected) and "USS", and a checkbox for "Copy command". The "Command Output:" field displays the following text:

```
*** TSO commands require an MVS System that is
EZZ2350I MVS TCP/IP NETSTAT CS V1R2 TCPI
EZZ2585I User Id Conn Local Socket Fore
EZZ2586I -----
EZZ2587I AESTCPIP 00010692 9.39.64.151..5050
sh
EZZ2587I AESTCPIP 00000066 0.0.0.0..5050
EZZ2587I BBOA5R2A 0002A914 0.0.0.0..63217
EZZ2587I BBOA5R2A 0002A92B 9.39.64.151..402
ch
```

Benefit: Developers do not have to continually switch between systems



z/OS Dataset Management



- Allocate
- Compress
- Compress with Backup
- Migrate
- HRECALL, HDELETE



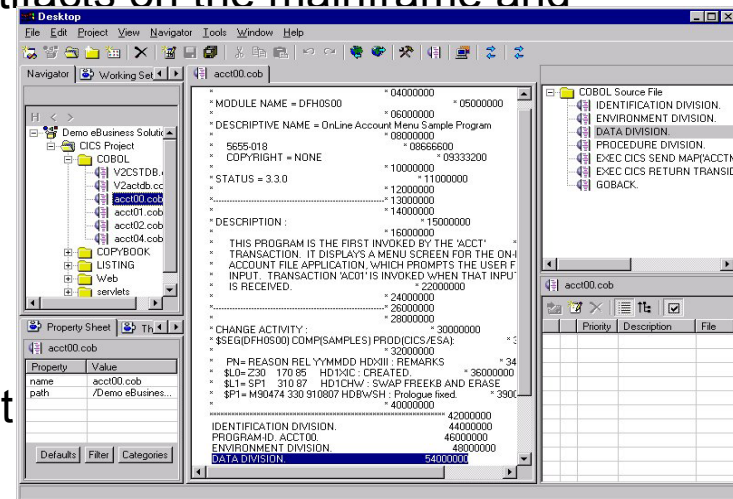
COBOL and PL/I Remote Edit/Compile/Debug

- 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 available on workstat
- ✓ Graphical debugger on workstation

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

- Exploits WebSphere Studio Workbench
 - ✓ Task manager, Projects/Perspectives , etc
- Live host connectivity (TSO Commands, Job queue mgmt, etc)



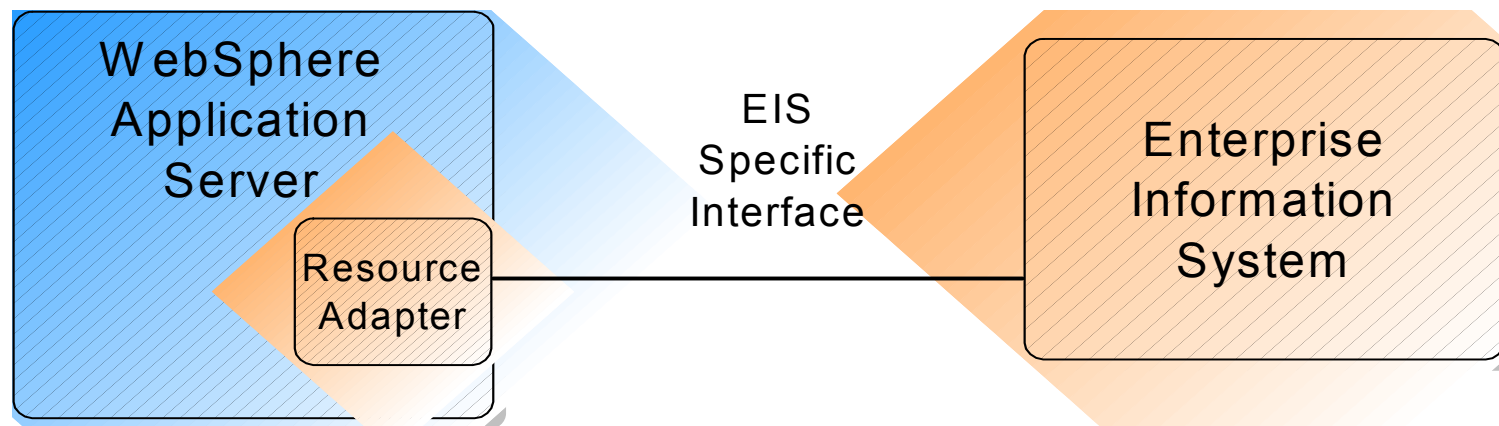
Agenda

- Positioning
- WebSphere Studio Enterprise Developer
 - z/OS Application Development Tools
 - **Basic J2EE Connector Architecture support**
 - XML Enablement for COBOL
- Summary



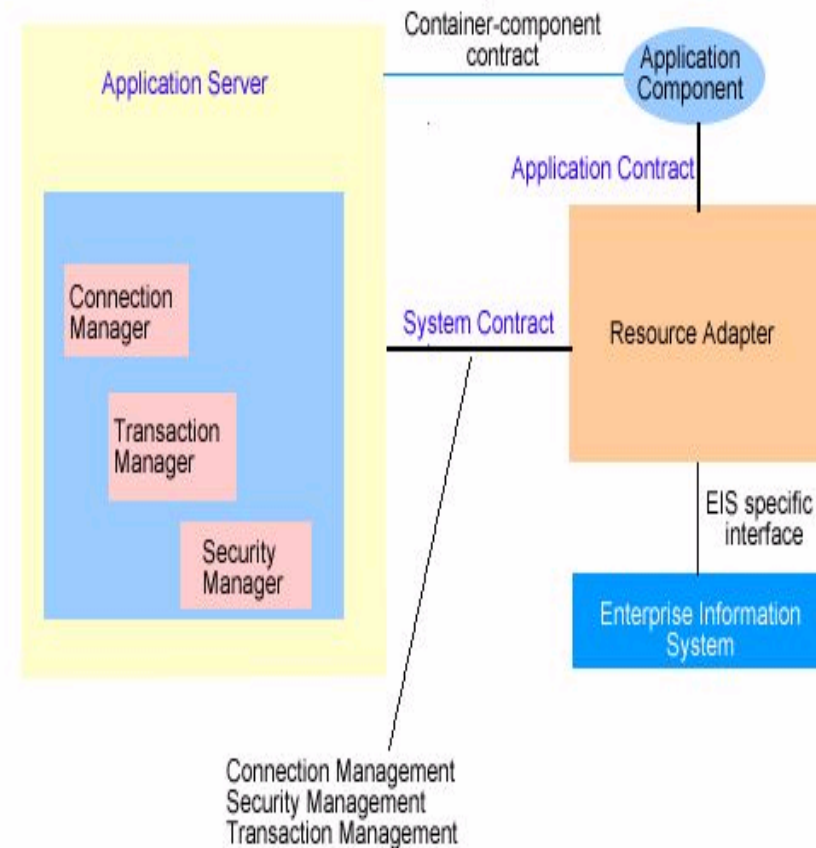
J2EE Connector Architecture (JCA)

- **Standards-based architecture for integrating J2EE systems and heterogeneous back-end (EIS) systems**
 - CICS, IMS, HOD, SAP, Siebel, PeopleSoft, etc.
- **Implemented by resource adapters that can plug into application server**
 - Provides connectivity to EIS system
 - Works with application server to provide pooling, transactions, security



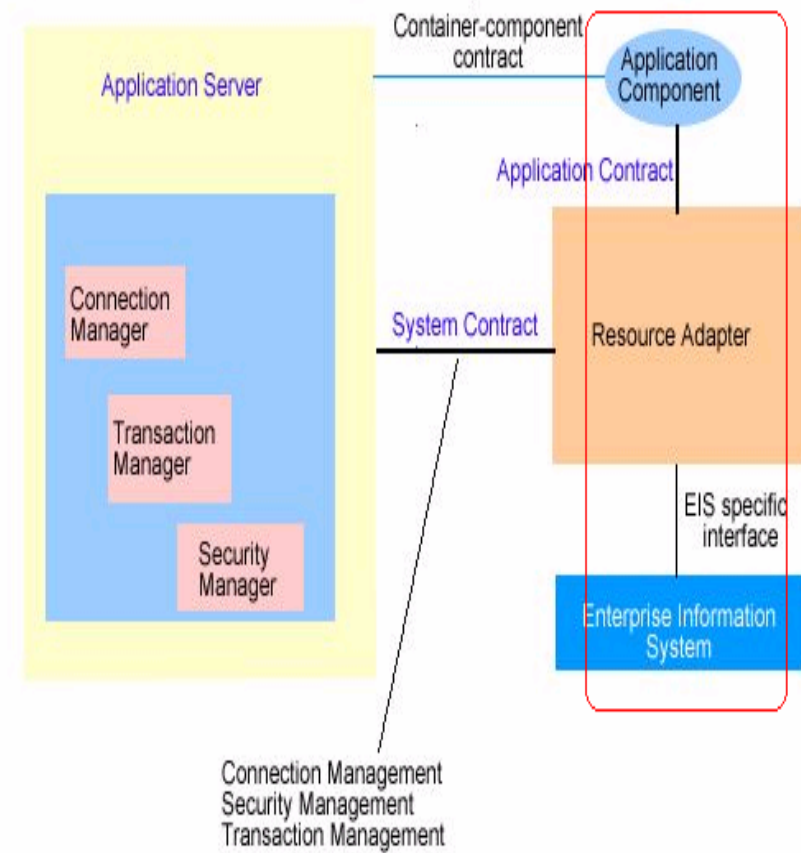
JCA Value

- **Enables EIS vendor to provide a standard resource adapter to talk to its EIS**
 - From any application server
- **Enables application server to have seamless connectivity to multiple EISes**
- **Allows EIS connections to take advantage of application server services**
 - Pooling, transactions, security
- **A Java solution to the problem of connectivity between application servers and EIS systems**



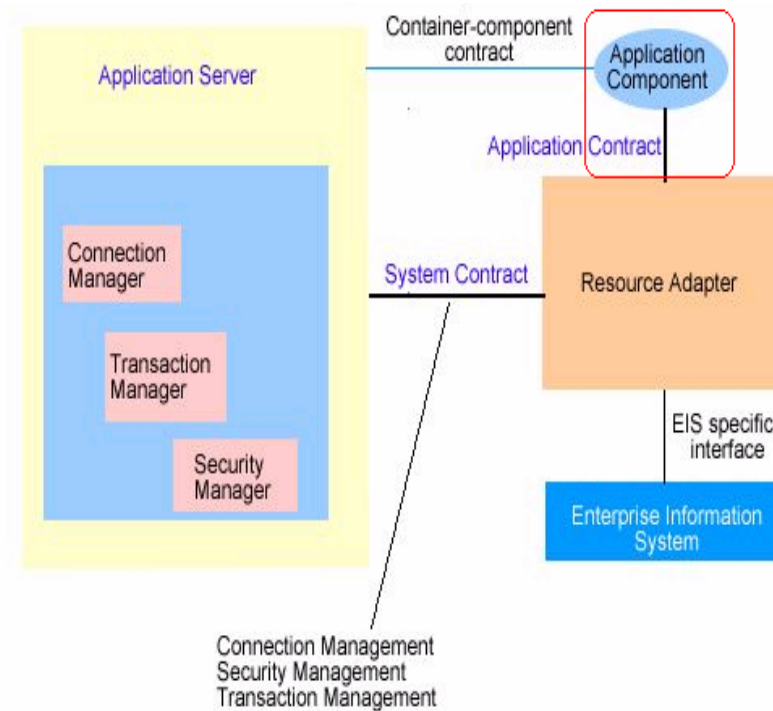
Basic JCA Support in Enterprise Developer

- Facilitates development of JCA services to connect to EIS
 - **Basic JCA** perspective
 - Development tools and runtime
 - Minimal knowledge of details of EIS system needed
- Connectors supported
 - CICS ECI
 - CICS EPI
 - HOD3270
 - IMS



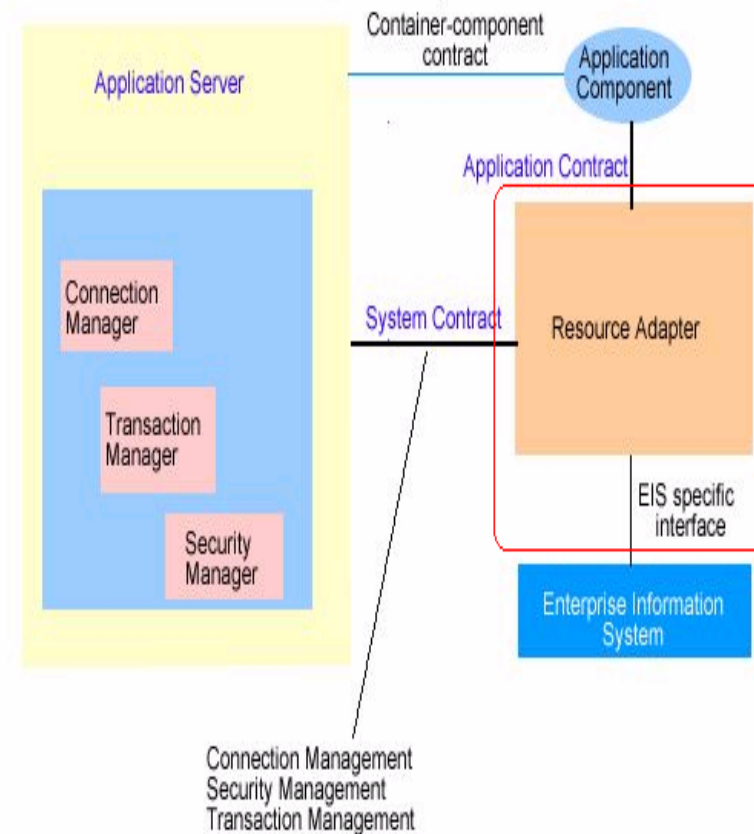
Basic JCA Tools

- Application Component
 - Easily generate service proxy (Java bean) to access the operations in the service
 - Making it easy for Java developers to write applications that access EIS systems



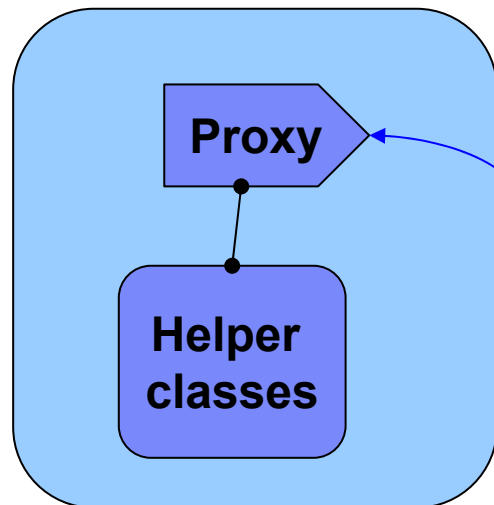
Basic JCA Runtime

- Resource adapter is needed to connect to EIS
 - Resource adapters stored in Resource Adapter Archive (.RAR) files
- Resource adapters
 - CICS ECI
 - CICS EPI
 - Host On Demand (HOD)
 - IMS

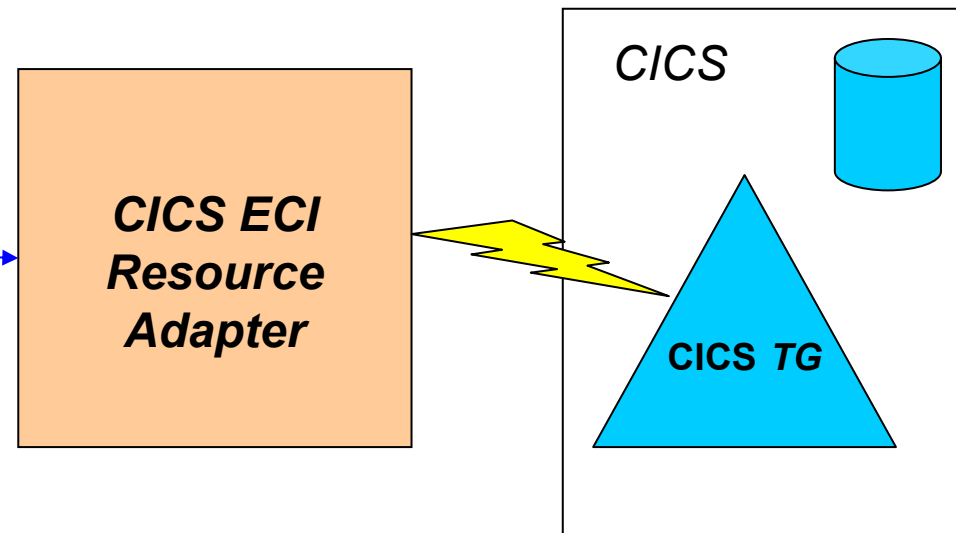


Basic JCA example: Using CICS ECI from a Java client

Java Client



z/OS



Exposing a CICS COBOL Transaction as a Service



Agenda

- Positioning
- WebSphere Studio Enterprise Developer
 - z/OS Application Development Tools
 - Basic J2EE Connector Architecture support
 - XML Enablement for COBOL
- Summary



XML 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 template COBOL driver program
 - ▶ Illustrate the invocation of converters
 - ▶ Illustrate the interaction with existing application
 - ▶ Needs to be updated before run
- Enables communication with XML based systems



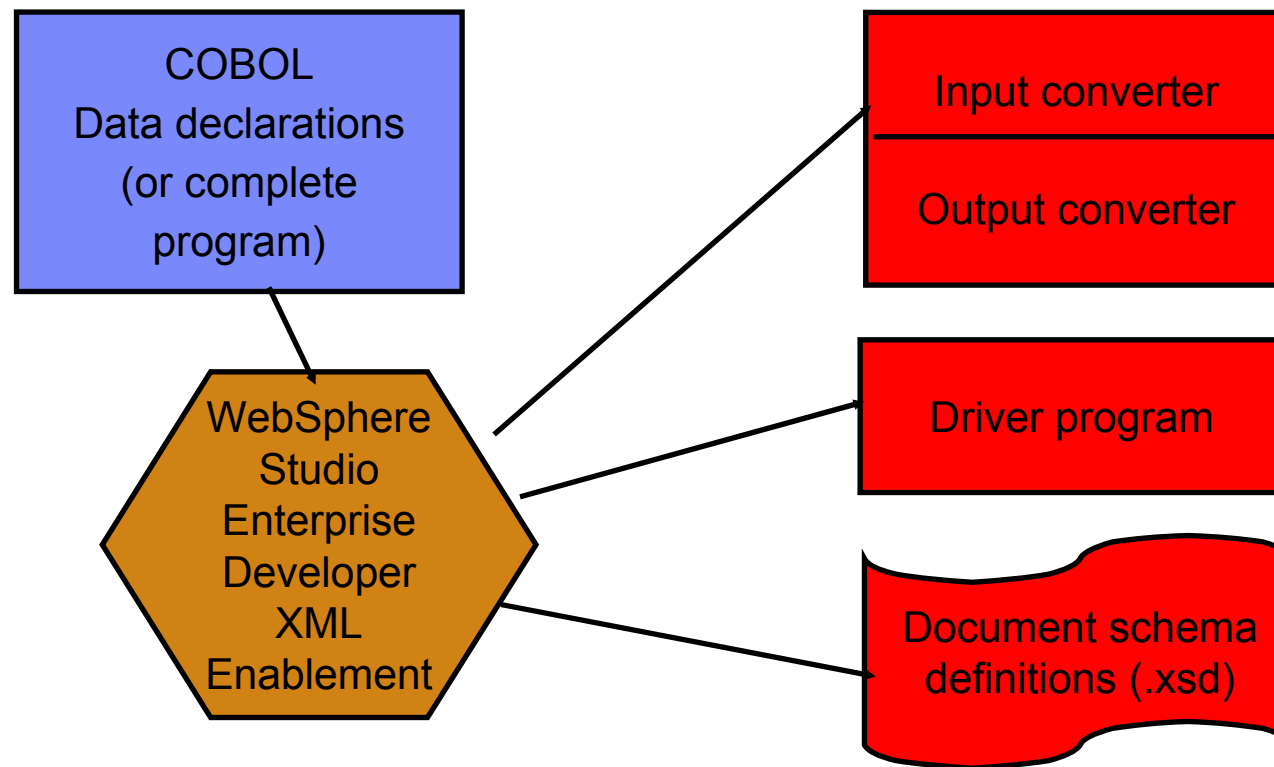
XML Enablement Driver

- SOAP for CICS
- SOAP for IMS
- CICS
- Batch, TSO, USS

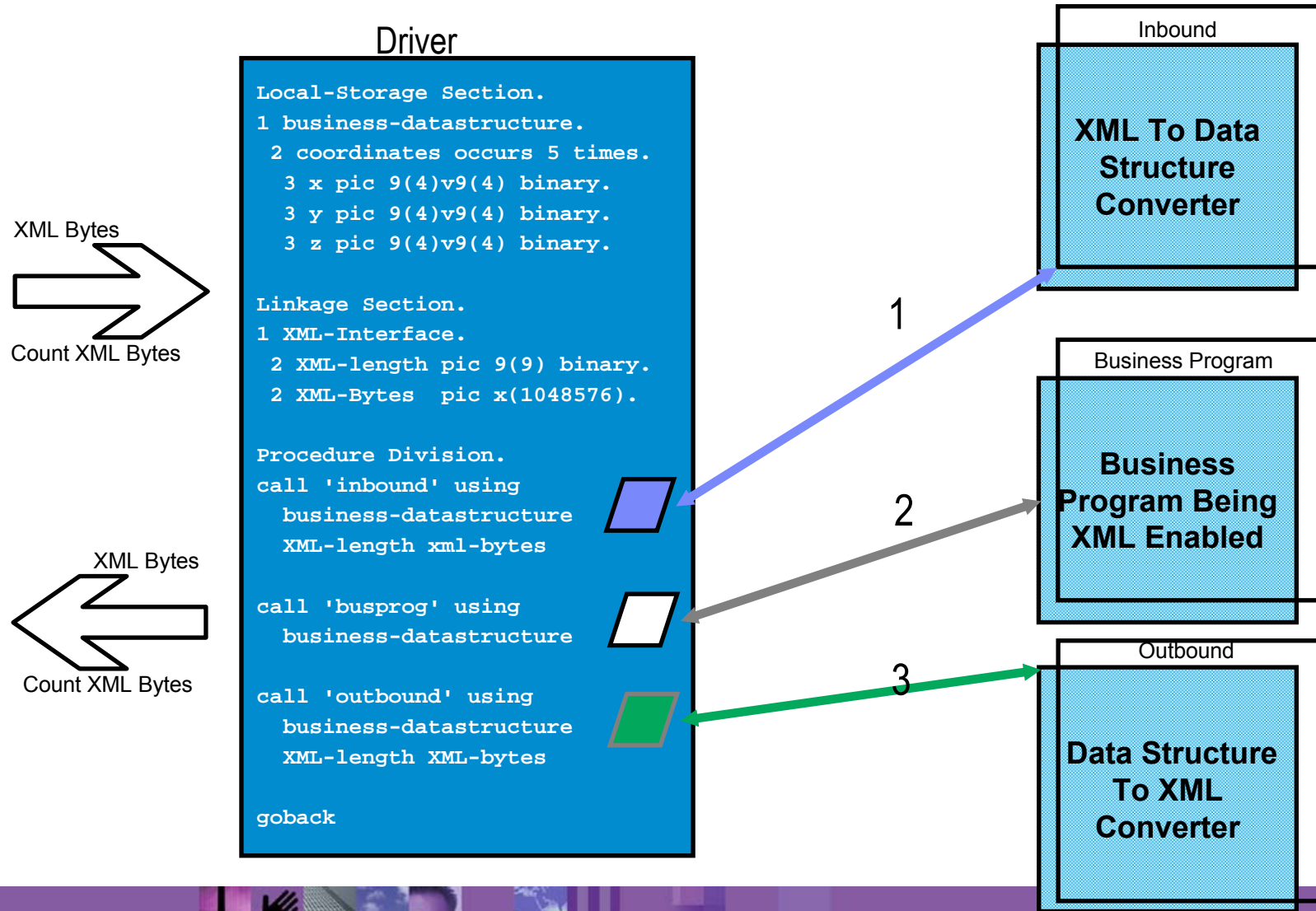


Mapping COBOL Data

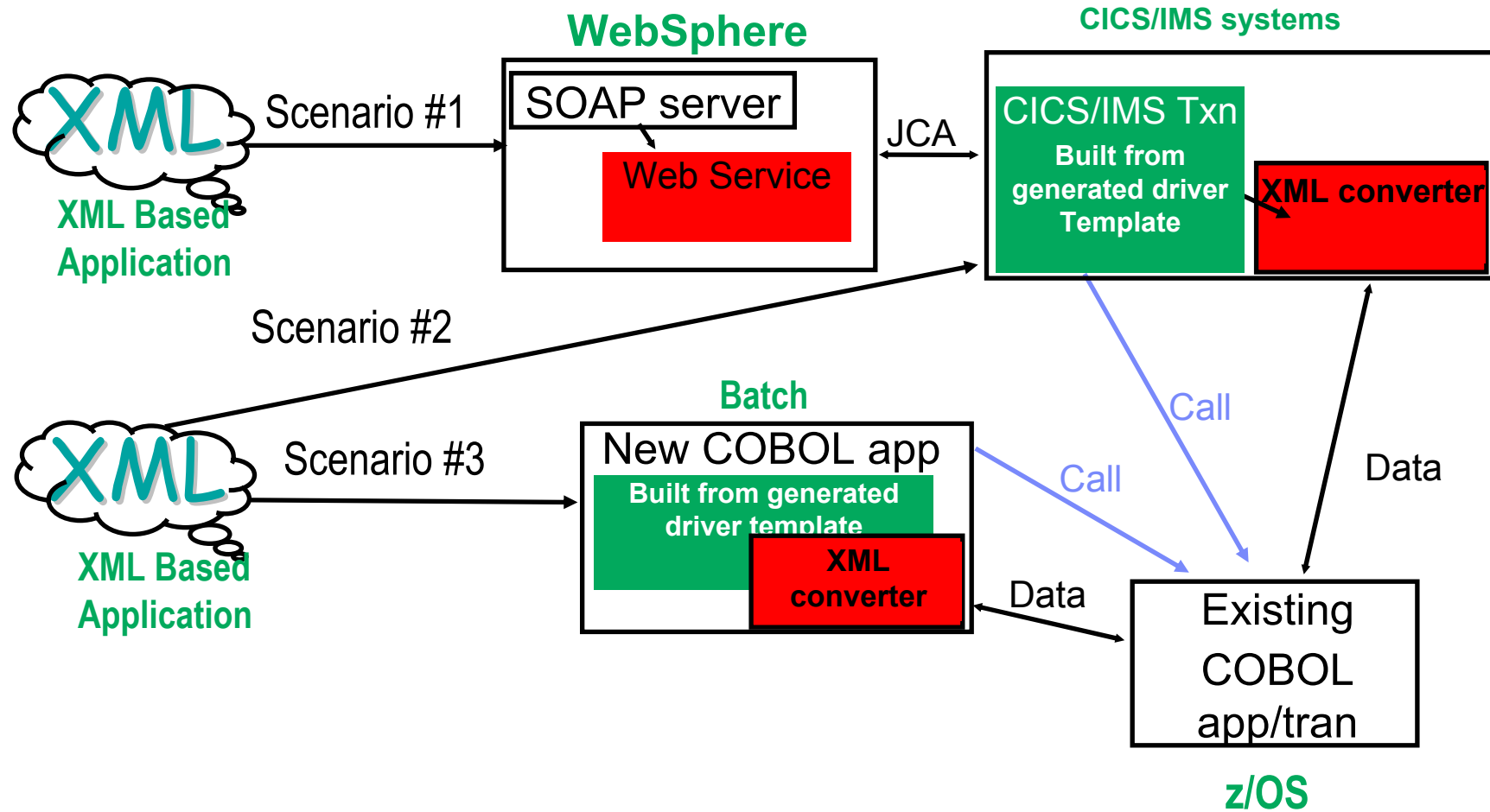
- Enables COBOL-based applications to consume and produce XML messages
 - ▶ Original COBOL program unchanged



Using the XML Converters



Runtime Scenarios



CICS SOAP Support

- XML Enablement plays a key role in this process by taking care of the consuming and producing of the SOAP message body.
- XML produced by the outbound converter is now suitable for direct inclusion in a SOAP message body.
- New option to specify the inbound and outbound namespaces.
- New CICS SOAP / CICS specific driver type is available to help minimize the amount of coding needed in the driver when deploying converters under CICS.
- New option to generate the converters and converter driver to one file or PDS member simplifies building and deployment.

Generate XML converter Wizard

XML converter options
Specify options for the XML converter

Specify identification attributes
Program name: Order
Author name: WSED

Specify XML converter driver type
Driver type: Batch, TSO and USS
Batch, TSO and USS
CICS
CICS SOAP
IMS

Configure XML message processing
Maximum message size (KB): IMS

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

Specify XML namespaces
Inbound namespace: http://www.OrderI.com/schemas/OrderInterface
Outbound namespace: http://www.OrderO.com/schemas/OrderInterface

Generate XML converter Wizard

File, data set, or member selection
Select the source and targets for the XML converter

Select the source for the XML converter
Source file or member: /SOAPCICS/Order.cpy Browse...

Select targets for the XML converter
Converter folder: /SOAPCICS Browse...
Input converter file name: OrderD.cbl
Output converter file name:
Converter driver file name:

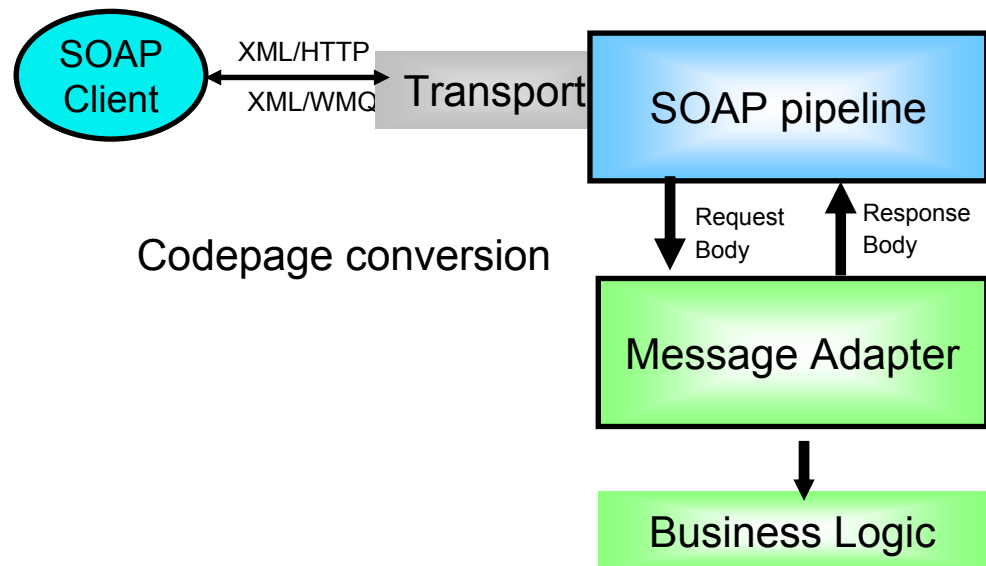
Generate converters and converter driver to the Input converter file



SOAP for CICS feature

Why process SOAP in CICS?

- ▶ Easily and efficiently reuse CICS assets from SOAP clients
 - drive existing CICS programs with user supplied XML<->COMMAREA mapping
 - Supports IBM WebSphere Studio Enterprise Developer generated code for mapping
- ▶ Performance focused
- ▶ Language neutral approach, works for COBOL, C and Java
- ▶ Synergy with WebSphere and industry direction for Web Services



SOAP for CICS

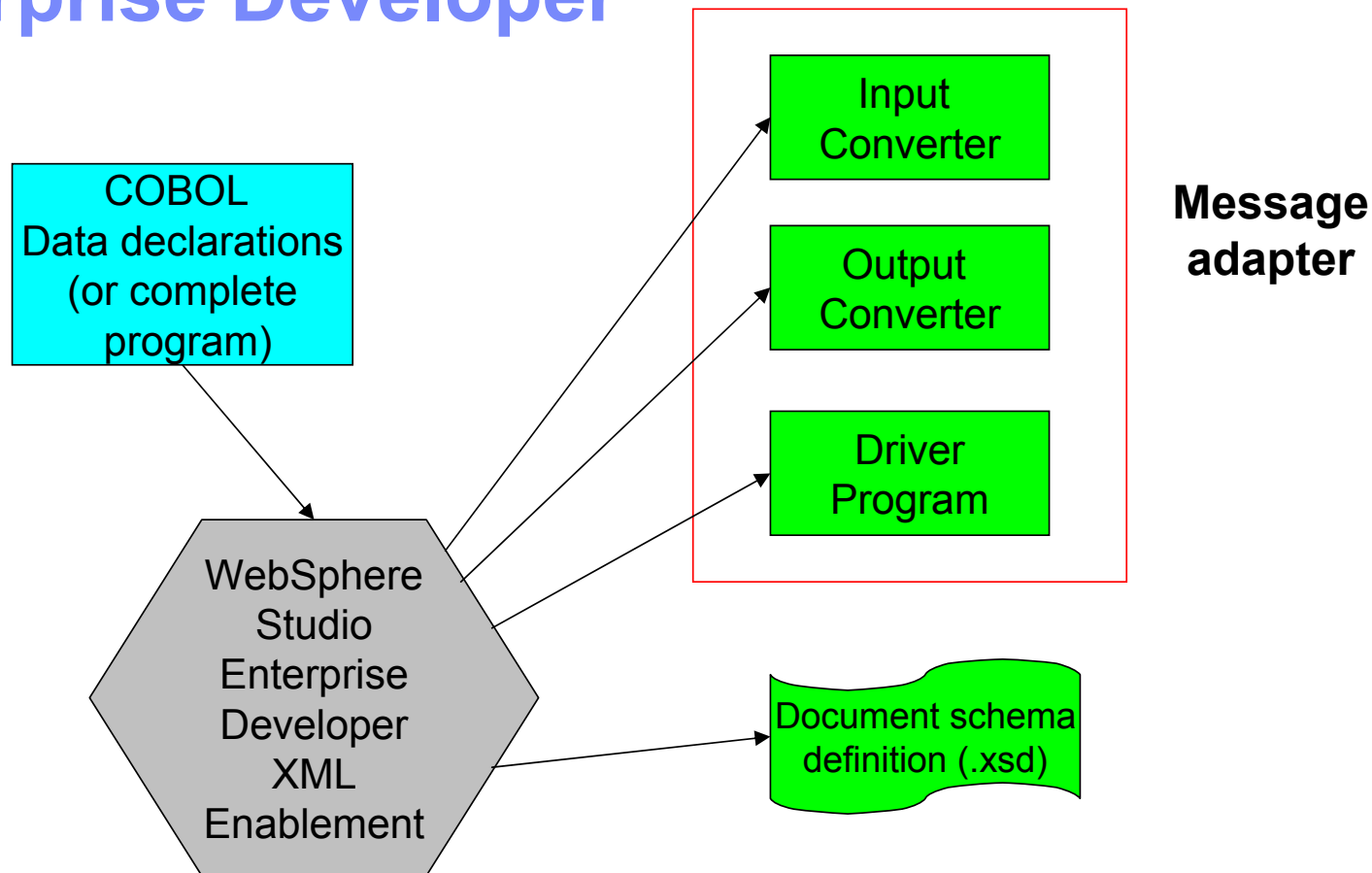
- Transport
- Envelope parsing
- URI mapping
- Envelope building

COBOL code

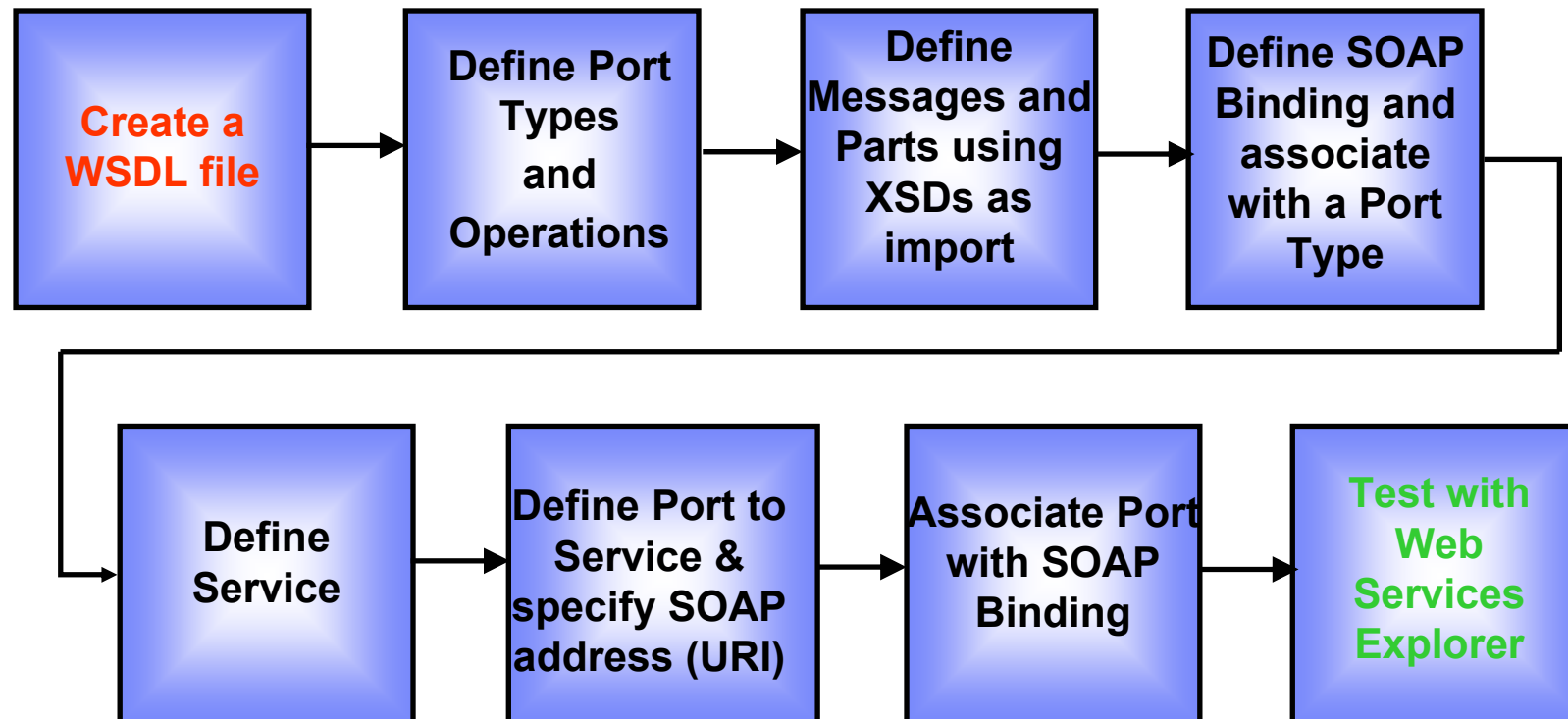
-generated by WebSphere Studio Enterprise Developer



Message Adapter - WebSphere Studio Enterprise Developer



WSDL for the SOAP for CICS Web Service



Web Services tools used: *WSDL creation*, *WSDL Editor*, *Web Services Explorer*



Agenda

- Positioning
- WebSphere Studio Enterprise Developer
 - z/OS Application Development Tools
 - Basic J2EE Connector Architecture support
 - XML Enablement for COBOL
- Summary



dwp Bank Uses IBM Tools to Modernize Legacy Applications and Integrate with J2EE

Challenge

Modernize and extend legacy COBOL and IMS applications and integrate them into an e-business infrastructure

Solution

Enterprise Transformation tools used by dwp Bank include:

WebSphere Studio Enterprise Developer
WebSphere Studio Asset Analyzer
WebSphere Studio Application Monitor
WebSphere Host Integration Solution
WebSphere Application Server for z/OS



Business benefits

- Allow the bank to sustain its transaction processing leadership in Germany and grow market share throughout Europe
- Reduce the cost of each individual transaction through economies of scale and decreased risk

Technology benefits

- Develop an IT architecture that is scalable and flexible, and that will allow the bank to integrate its existing and new systems
- Align the IT environment with business objectives
- Reduce functional and data redundancies



Summary

- **Capability Rich - Comprehensive end-to-end development environment facilitates development of mixed workload applications**
 - ▶ Runtimes: WebSphere Application Server, CICS, IMS, z/OS, USS
 - ▶ Developers: Web, Java, and Enterprise Developers
 - ▶ Tools: Web, Java, J2EE, XML, COBOL, and web services
- **Productivity - Higher-quality applications in a fraction of the time**
 - ▶ Facilitates the development and deployment of a broad range of applications including web, Java, J2EE, XML, and COBOL
 - ▶ Facilitates the building and testing of z/OS applications
 - ▶ Facilitates enterprise transformation of existing COBOL applications to support XML messages
- **Cost efficient - Reduced Total Cost of Ownership (TCO) by adopting a consistent development environment for the enterprise**
 - ▶ Single development environment to manage and deploy vs. multiple
 - ▶ Simplified training requirements

