Revitalize your CICS

Session 2: Revitalize your Applications

SOA on your terms and our expertise – www.ibm.com/cics

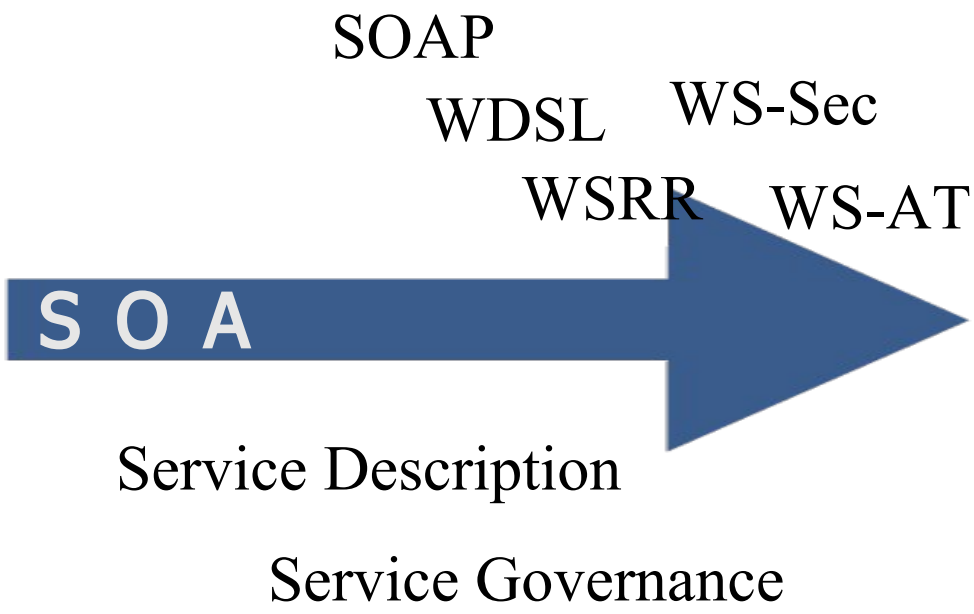
Agenda

- Spectrum of Service enablement formality
  - Multiple Accounts/Portfolio scenario
- Web Services
  - Service Component Architecture
  - WS-Addressing
- Bundles
- WSRR
- XML Transforms and offload
- Java and JavaServers

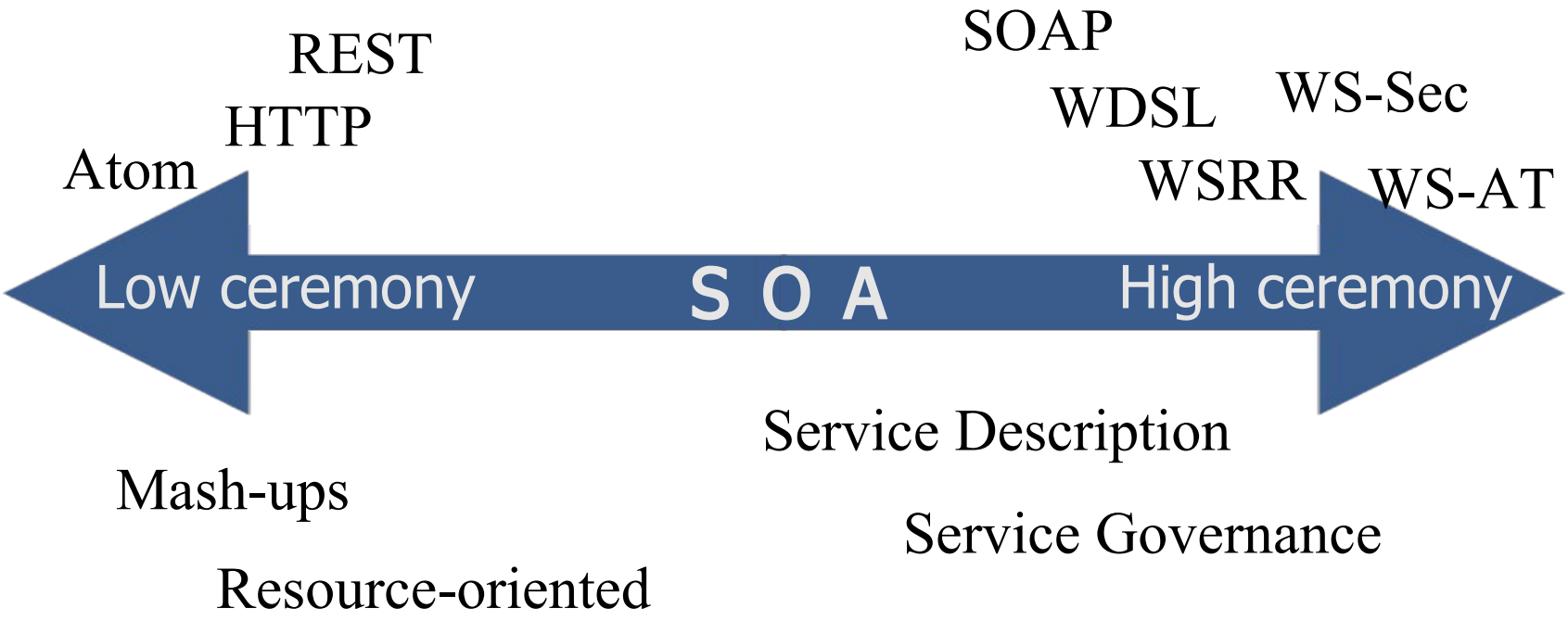
Accounts in multiple banks

- Trend towards customers having multiple accounts in different banks
  - How can customers see all their accounts in one consolidated view?
  - How can our Revitalized Bank be seen as the customer's primary bank?
- Multi-account mashup provided by the Revitalized Bank
  - Both the provider of account information and on-the-glass integrator
  - Uses account information feeds, which can then also be used in other ways
- Use feeds of transaction data from an account, and of location data of payment terminals

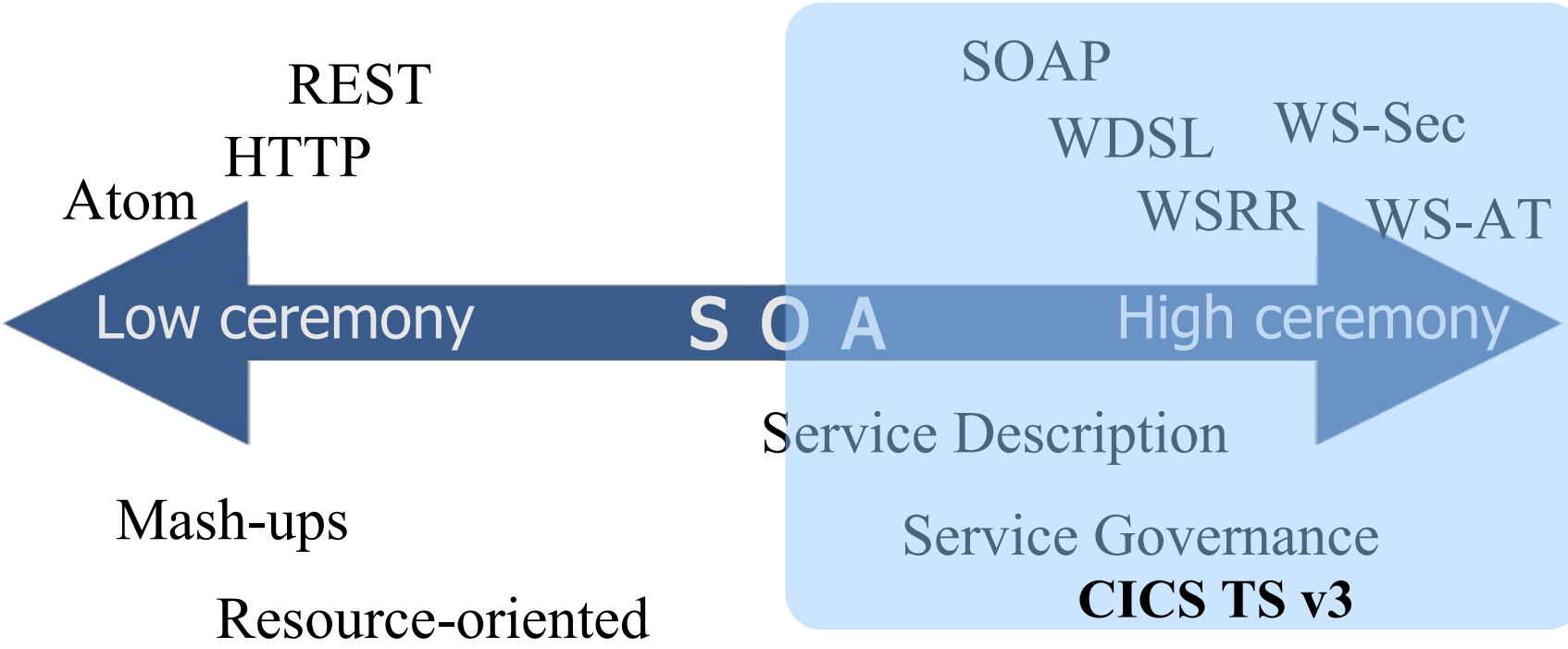
A Spectrum of Service Enablement



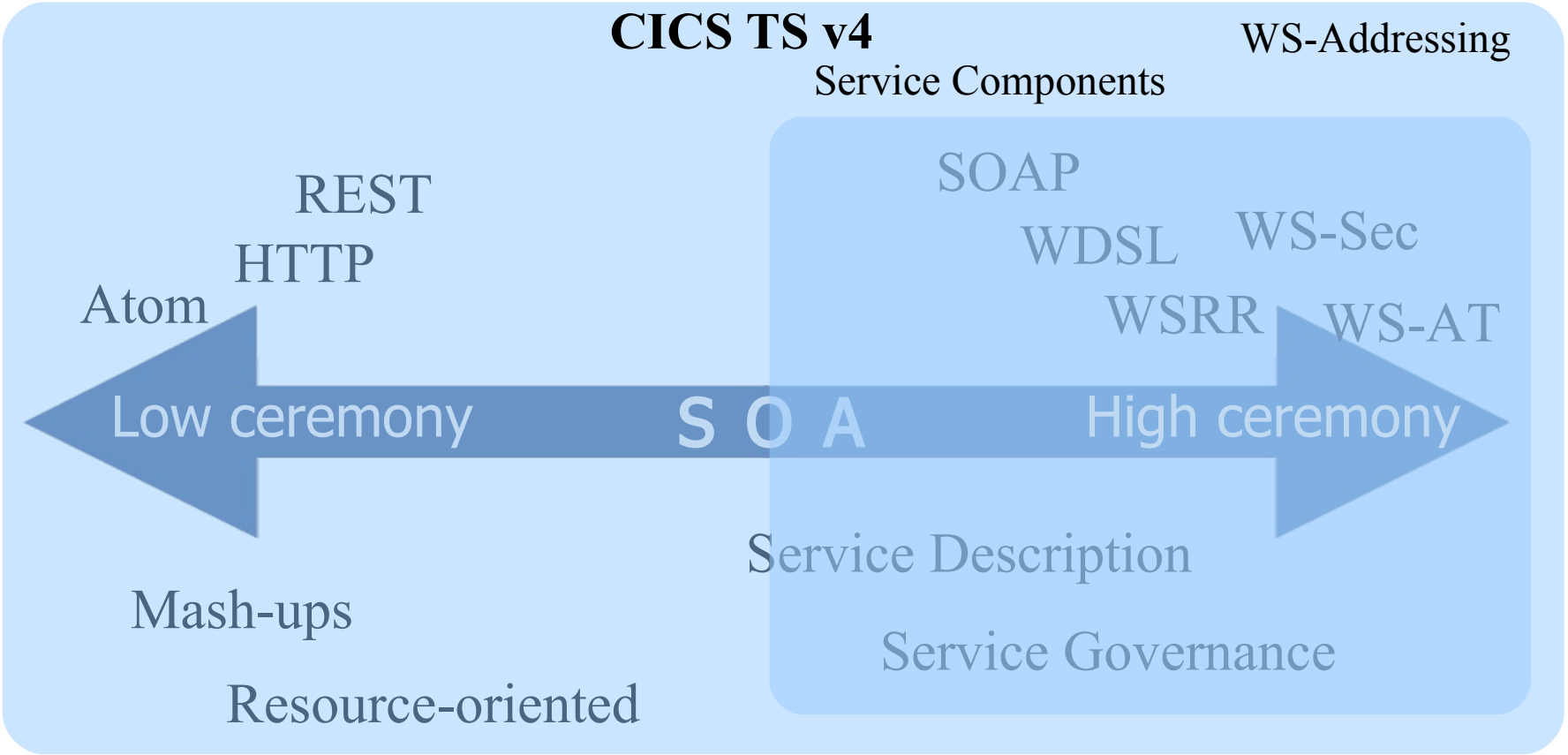
A Spectrum of Service Enablement



A Spectrum of Service Enablement



A Spectrum of Service Enablement



Agenda

- Spectrum of Service enablement formality
  - Multiple Accounts/Portfolio scenario
- **Web Services**
  - **Service Component Architecture**
  - WS-Addressing
- Bundles
- WSRR
- XML Transforms and offload
- Java and JavaServers

Sub-agenda – Service Component Architecture

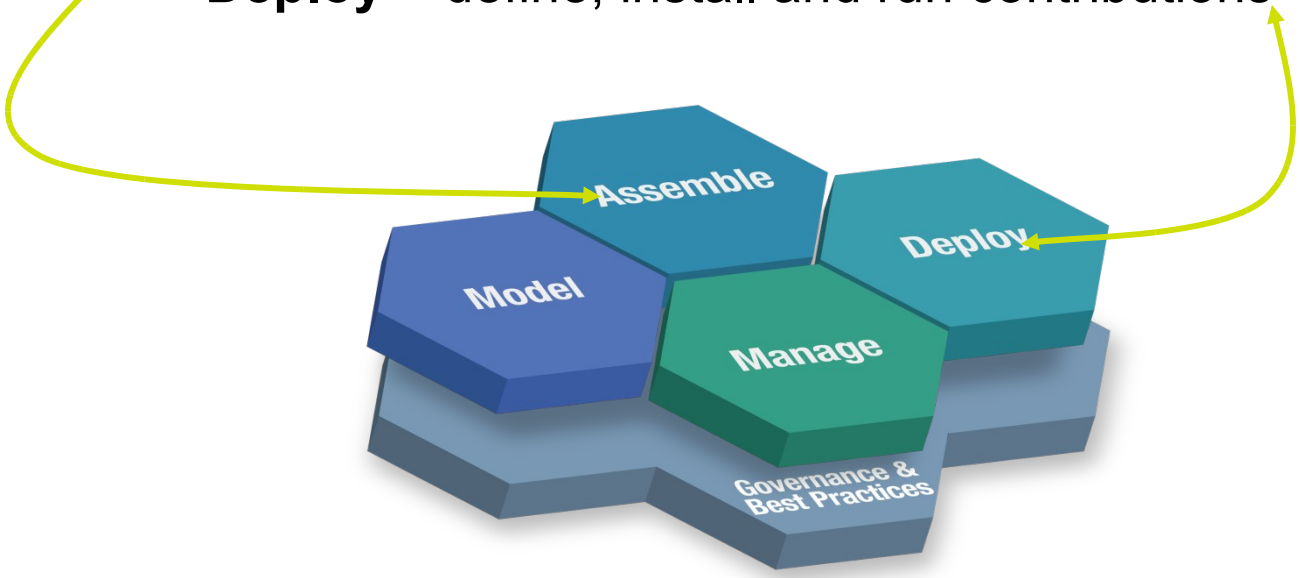
- **What is SCA**
- Introduction to SCA support in CICS TS v4.1
- The Portfolio Scenario
- Defining and implementing SCA components using Rational Developer for System z
- Deploying and running SCA components in CICS TS

SCA Terms and Relations to SOA Foundation

SCA is the development, deployment model of the **SOA** Foundation.

SCA is the open standard model for service assembly.

Assemble = develop interfaces, implementations, composites.
Deploy = define, install and run contributions



Service Component Architecture

- A concrete manifestation of an ***SOA way of thinking***.
  - Designed for building agile service oriented applications.
- ***A framework*** for implementing, assembling, composing and deploying services.
  - Supports loose or tight coupling of coarse or fine grained services.
- Extends, exploits and ***complements existing technologies*** and standards.
  - Language, Application Environment, Framework and Vendor neutral.
- Supports Java and Web Services, ***and more***
- An extensible set of:
  - Protocol bindings (eg. SCA, WS, RMI, …)
  - Implementation languages (eg. Composite, Java, …)
  - Interface definitions (eg. WSDL, Java, …)
  - Pluggable Data bindings (eg. PoJo, JAXB, …)
  - Policies and Intents (eg. Integrity, Confidentiality).
- "Open SCA" refers to Service Component Architecture as defined by the industry at both the OSOA collaboration
  - "Classic SCA" refers to Service Component Architecture as it is defined and built by IBM supported in a variety of WebSphere Family products starting with V6.

SCA: What it is NOT

- Does not model individual ***workflows***
  - use BPEL or other workflow languages
- Is not solely ***Web services***
  - SCA can use / may use Web services, but can also build solutions with no Web services content
- Is not tied to a specific runtime environment
  - distributed, heterogeneous, large, small
- Does not force use of specific programming languages and technologies
  - aims to encompass many languages, frameworks, technologies
- embrace not replace
  - adaptable to new technology.

Key benefits of SCA

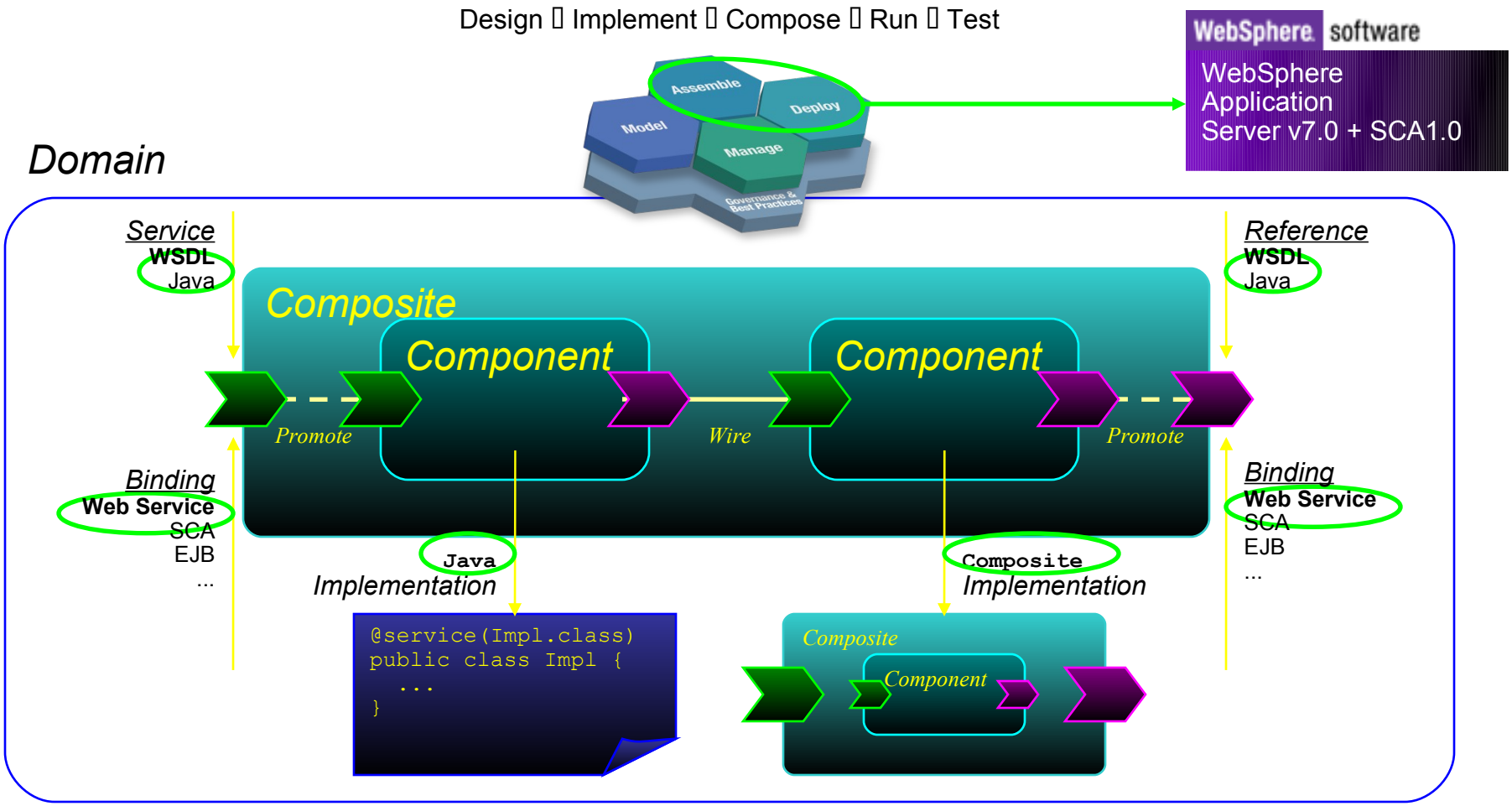
- ***Separation of Concerns*** Developers in an SOA need only be concerned with what they need to be.
- ***Loose Coupling*** components integrate without need to know how others are implemented
- ***Flexibility*** components can easily be replaced by other components
- ***Services*** can be easily invoked either synchronously or asynchronously
- ***Composition*** of solutions: clearly described
- ***Productivity*** easier to integrate components to form composite application
- ***Heterogeneity*** multiple implementation languages, communication mechanisms
- ***Declarative*** application of infrastructure services
- ***Simplification*** for all developers, integrators and application deployers

SCA v1.0

- OSOA – Consortium of industry vendors
  - http://www.osoa.org
- The OASIS Open Composite Services Architecture (CSA) Member Section advances open standards that simplify SOA application development.
  - http://www.oasis-opencsa.org/
  - Open CSA brings together vendors and users from around the world to collaborate on the further development and adoption of the Service Component Architecture (SCA) and Service Data Objects (SDO) families of specifications
- Apache Tuscany simplifies the task of developing SOA solutions by providing a comprehensive infrastructure for SOA development and management that is based on Service Component Architecture (SCA) standard.
  - http://tuscany.apache.org/

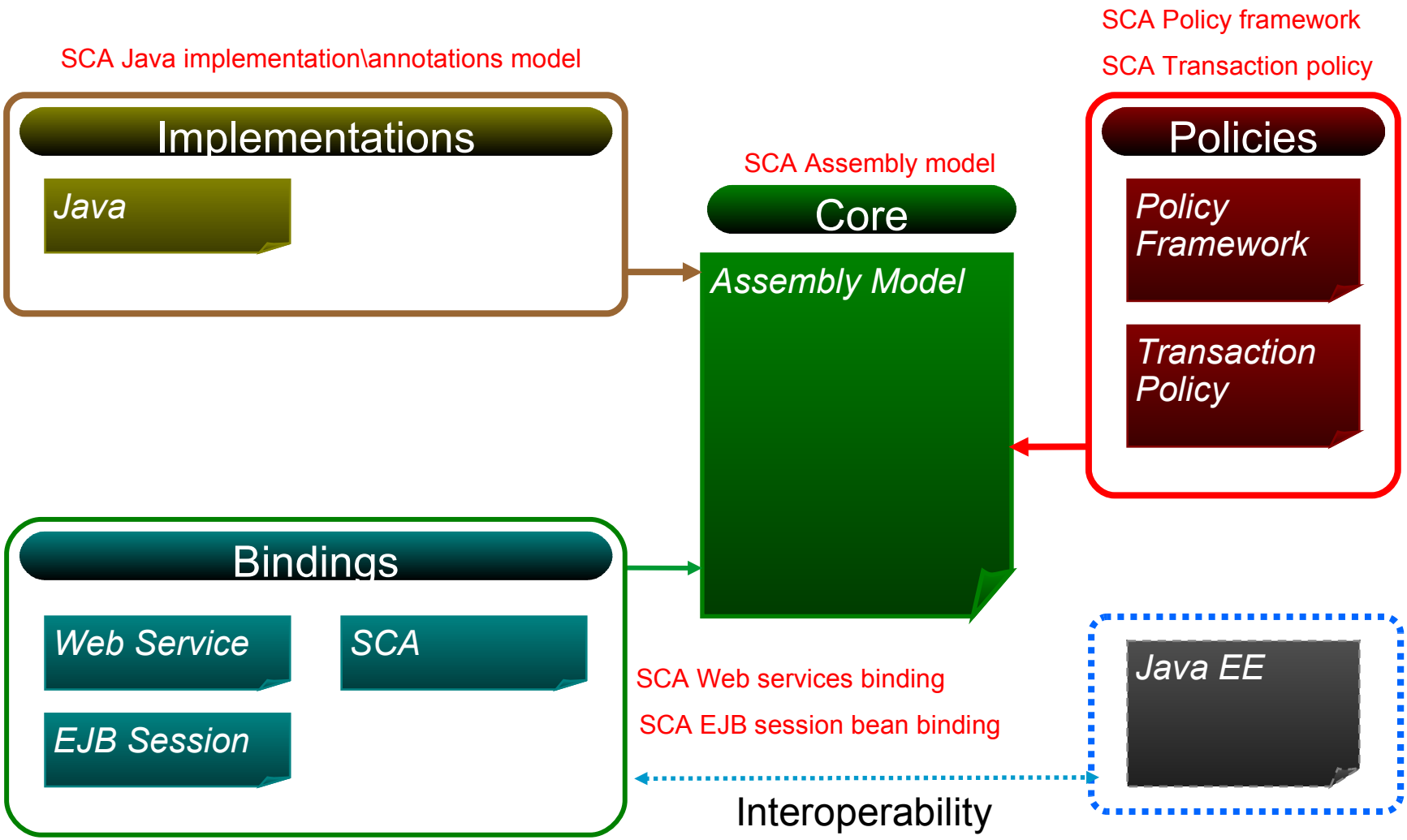
OASIS Open CSA

SCA Key Concepts



Reusability, Connectivity, Flexibility, Extensibility

SCA v1.0 Specifications – Flexible & Extensible



Sub-agenda – Service Component Architecture

- Quick SCA in WAS recap
- **Introduction to SCA support in CICS TS v4.1**
- The Portfolio Scenario
- Defining and implementing SCA components using Rational Developer for System z
- Deploying and running SCA components in CICS TS

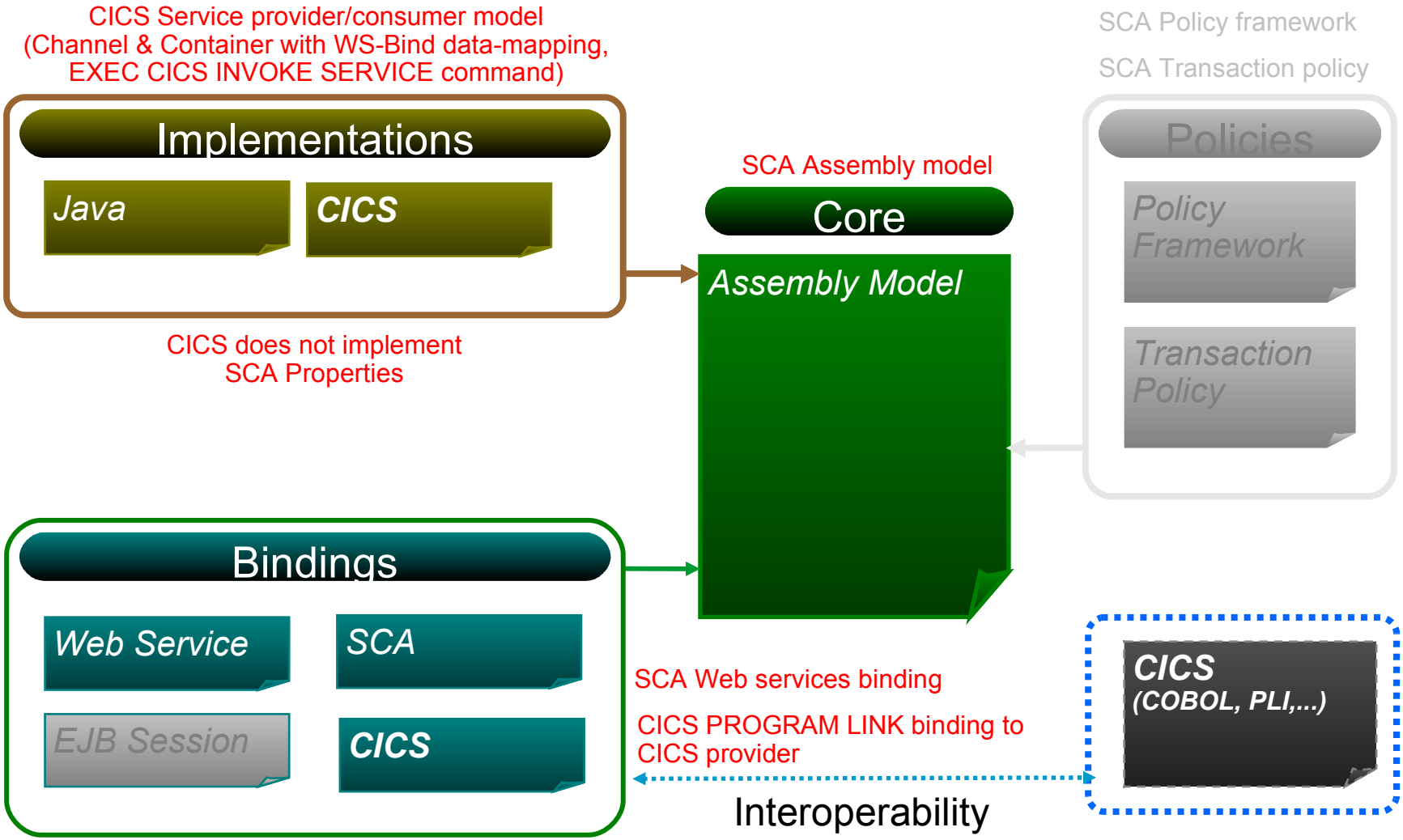
CICS TS v4.1 Component Architecture

- Provide capability to easily develop flexible and reusable CICS application components
  - Rapid assembly and deployment of new Services
  - Express existing applications as re-usable components
- Separation of bindings from application code allows flexible infrastructure changes
- Reduce skills and effort required to view and manage business applications

Component Architecture in CICS TS v4.1

- Ability to install and manage business applications as single CICS components
  - Abstract away from programs, transactions, resources
- Ability to describe CICS application as SCA components (using SCDL)
- Application bindings provided by CICS and configured using SCDL
  - Services and References
  - Invocation locally and via web services
  - EXEC CICS INVOKE SERVICE
- RDz providing CICS component tooling to enable component definition, assembly and deployment

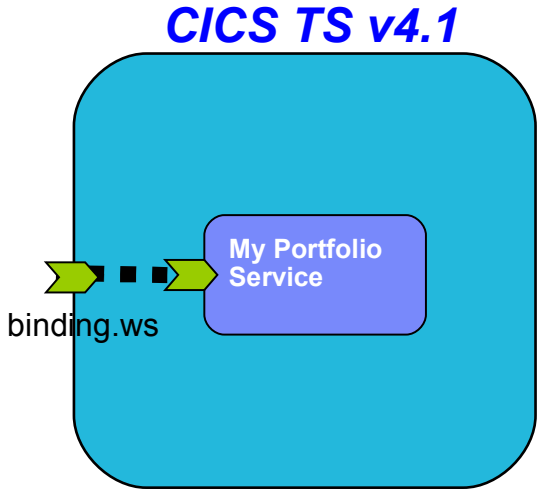
SCA v1.0 Specifications – CICS TS v4.1 capabilities



Sub-agenda – Service Component Architecture

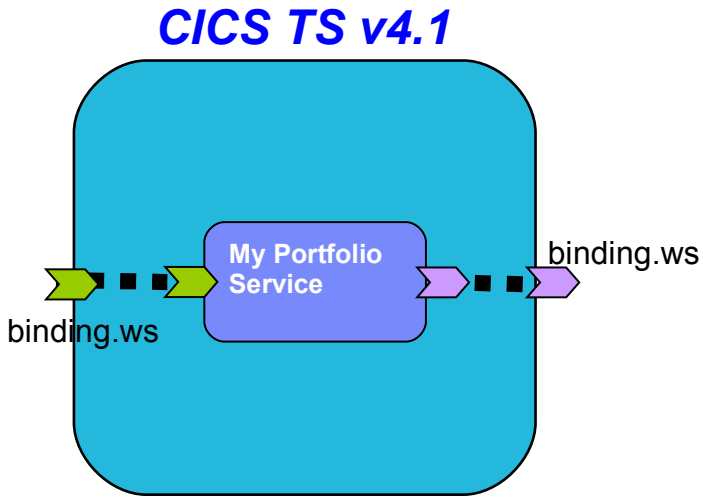
- Quick SCA in WAS recap
- Introduction to SCA support in CICS TS v4.1
- **The Portfolio Scenario**
- Defining and implementing SCA components using Rational Developer for System z
- Deploying and running SCA components in CICS TS

Service Component Architecture - Scenario



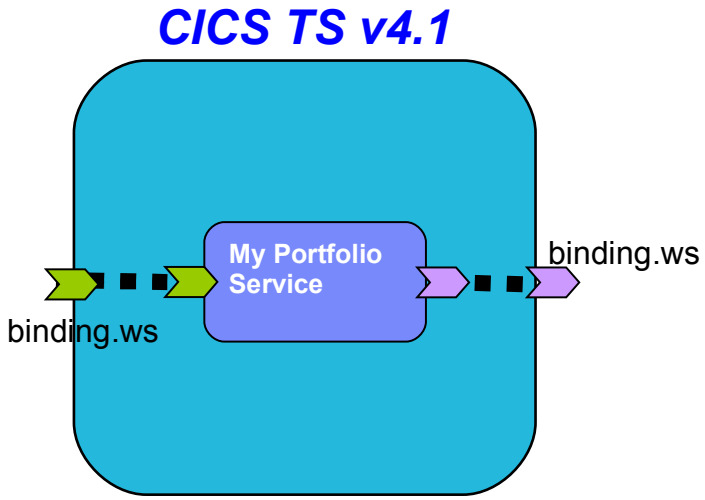

```
<component name = "MyPortfolioComponent">
      <implementation.CICS program="PORTFOLI"/>
      <service name="MyPortfolioService"
            <binding.ws ... >
      </service>


</component>
```

Service Component Architecture – Scenario



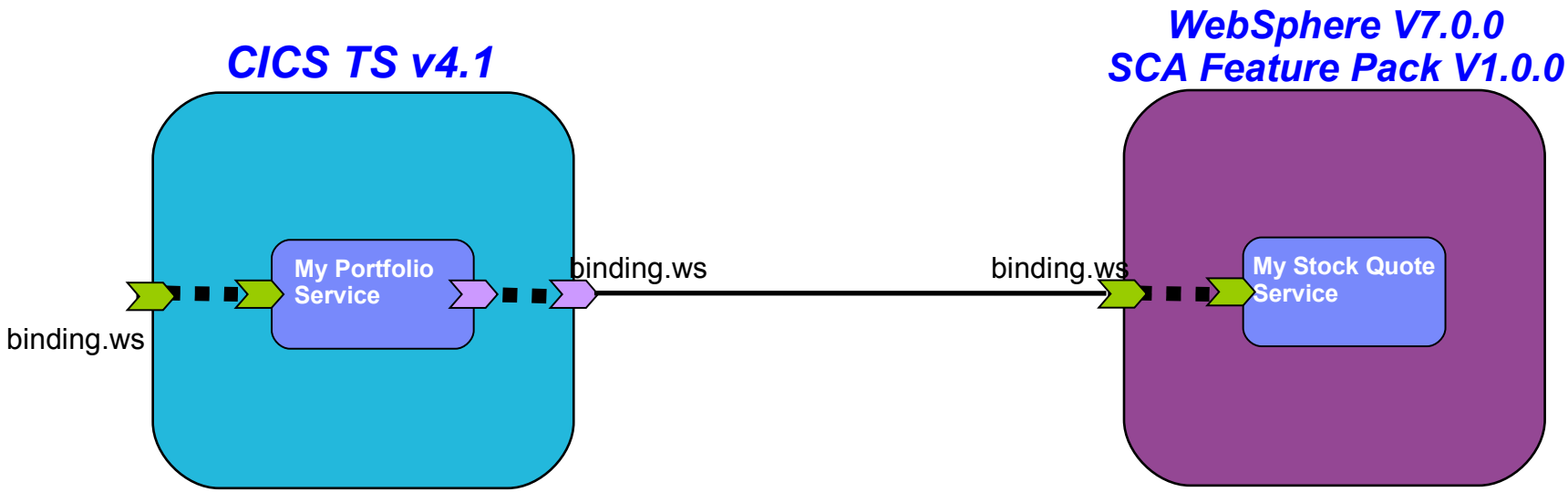

```
<component name = "MyPortfolioComponent">
      <implementation.CICS program="PORTFOLI"/>
      <service name="MyPortfolioService"
            <binding.ws ... >
      </service>
      <reference name="MyStockQuoteService">
            <binding.ws ... >
      </reference>
</component>
```

Service Component Architecture – Programming Model




```
CICS COBOL Program

EXEC CICS PUT CONTAINER("SYMBOL")
              CHANNEL("QUOTESERVICE")
              FROM("IBM ")
.

EXEC CICS INVOKE SERVICE("MyStockQuoteService")
                 CHANNEL("QUOTESERVICE")
```

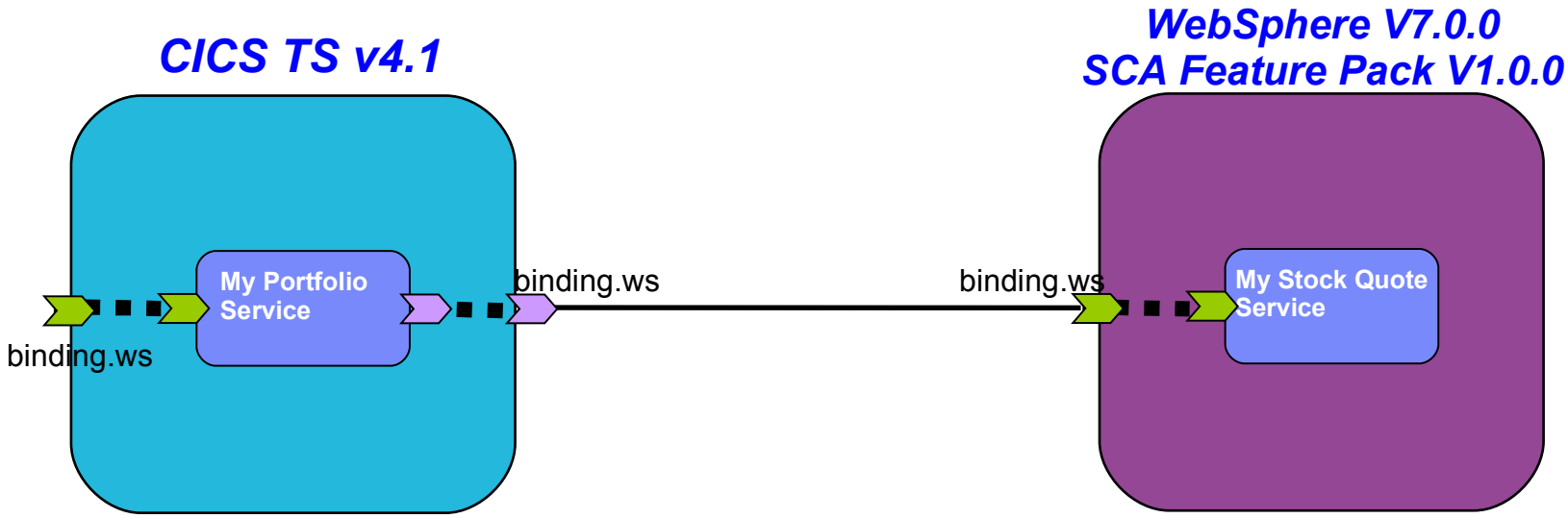
Service Component Architecture – Scenario




```
<component name = "MyPortfolioComponent">
        <implementation.CICS program
        <service name="MyPortfolioSe
                <binding.ws ... >
        </service>
        <reference name="MyStockQuot
                <binding.ws ... >
        </reference>
</component>
```

```
<component name = "MyStockQuoteComponent">
        <implementation.java
class="mystockquoteImpl.class"/>
        <service name="MyStockQuoteService">
                <binding.ws ... >
        </service>
</component>
```

Service Component Architecture – Programming Model



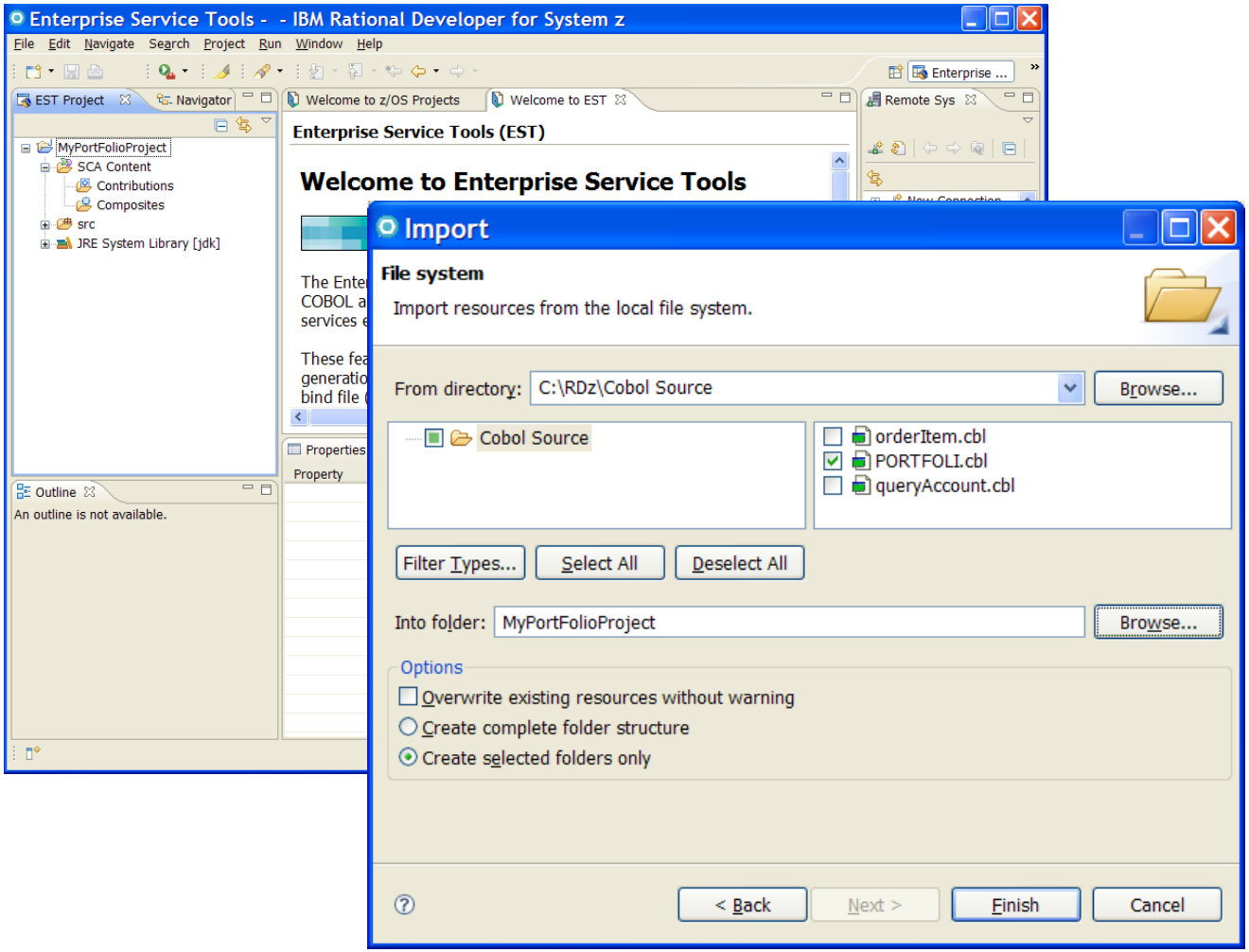

```
CICS COBOL Program

EXEC CICS PUT CONTAINER("SYMBO
              CHANNEL("QUOTESE
              FROM("IBM  ")
.

EXEC CICS INVOKE SERVICE("MySt
                CHANNEL("QUOT
```

```
WebSphere POJO

@Remotable
public interface stockQuote{
   public String myStockQuoteService();}

@Service(stockQuoteImpl.class)
 public class stockQuoteImpl implements
stockQuote
```

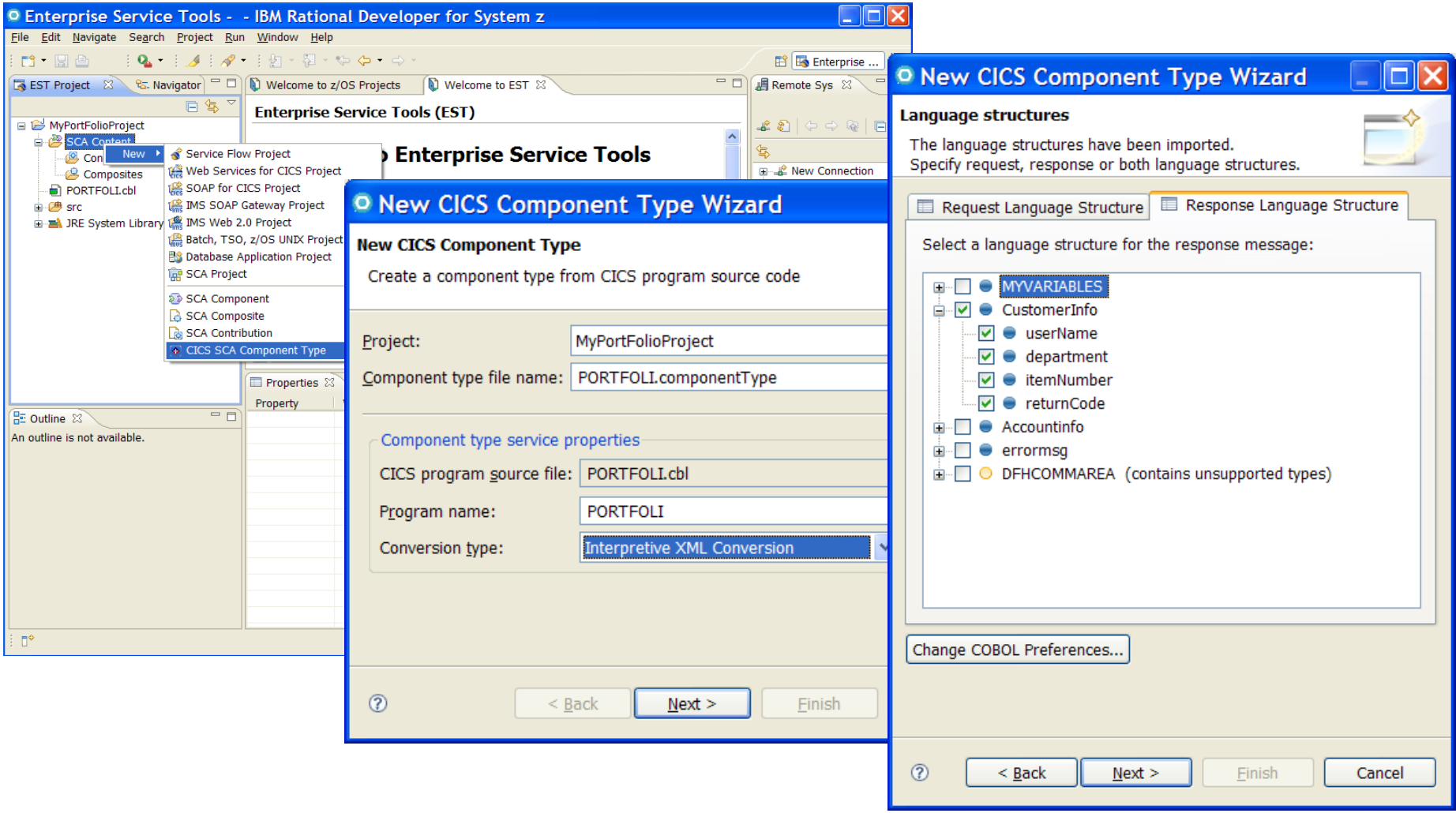
Sub-agenda – Service Component Architecture

- Quick SCA in WAS recap
- Introduction to SCA support in CICS TS v4.1
- The Portfolio Scenario
- **Defining and implementing SCA components using Rational Developer for System z**
- Deploying and running SCA components in CICS TS

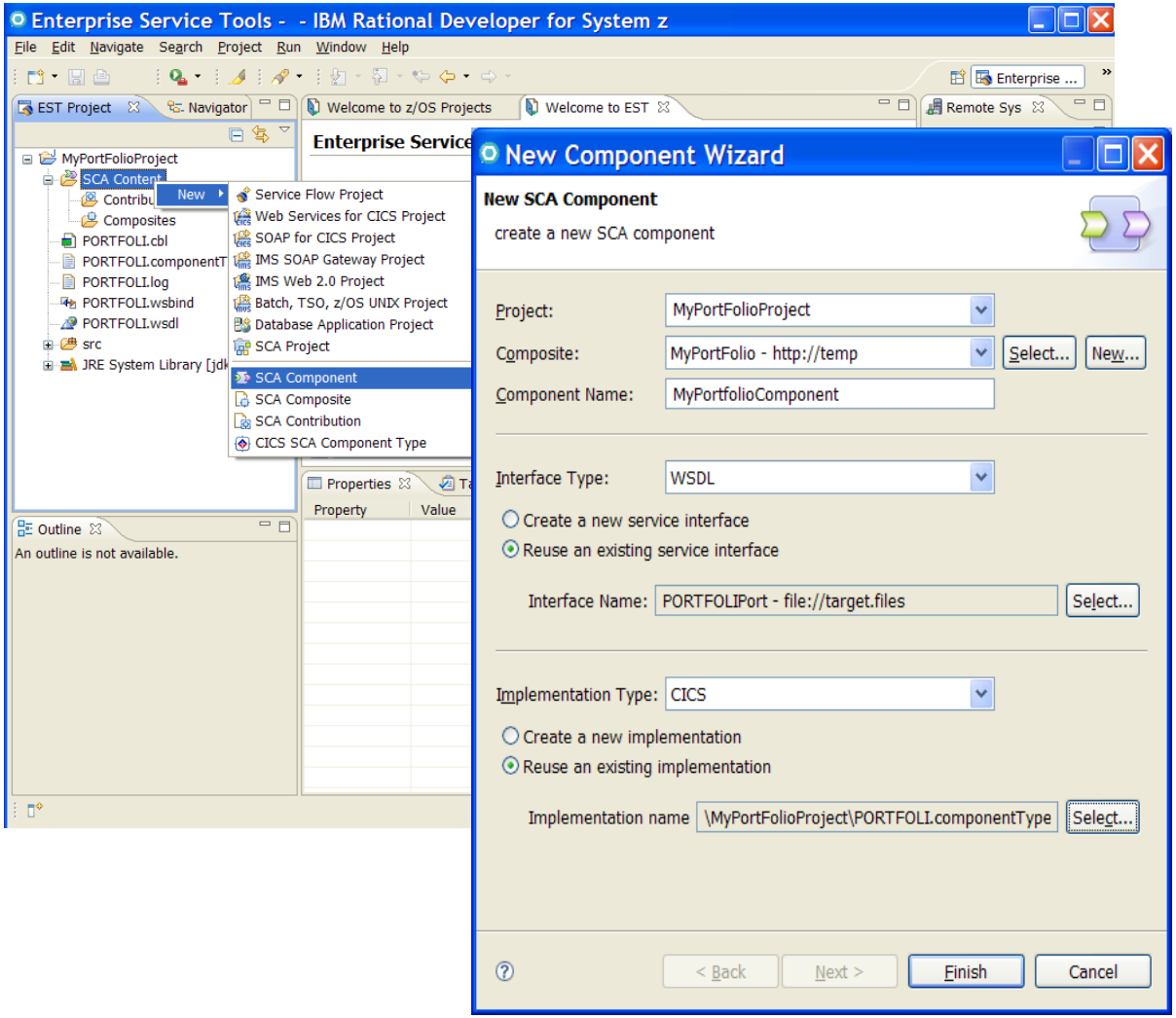
RDz SCA Tooling Support



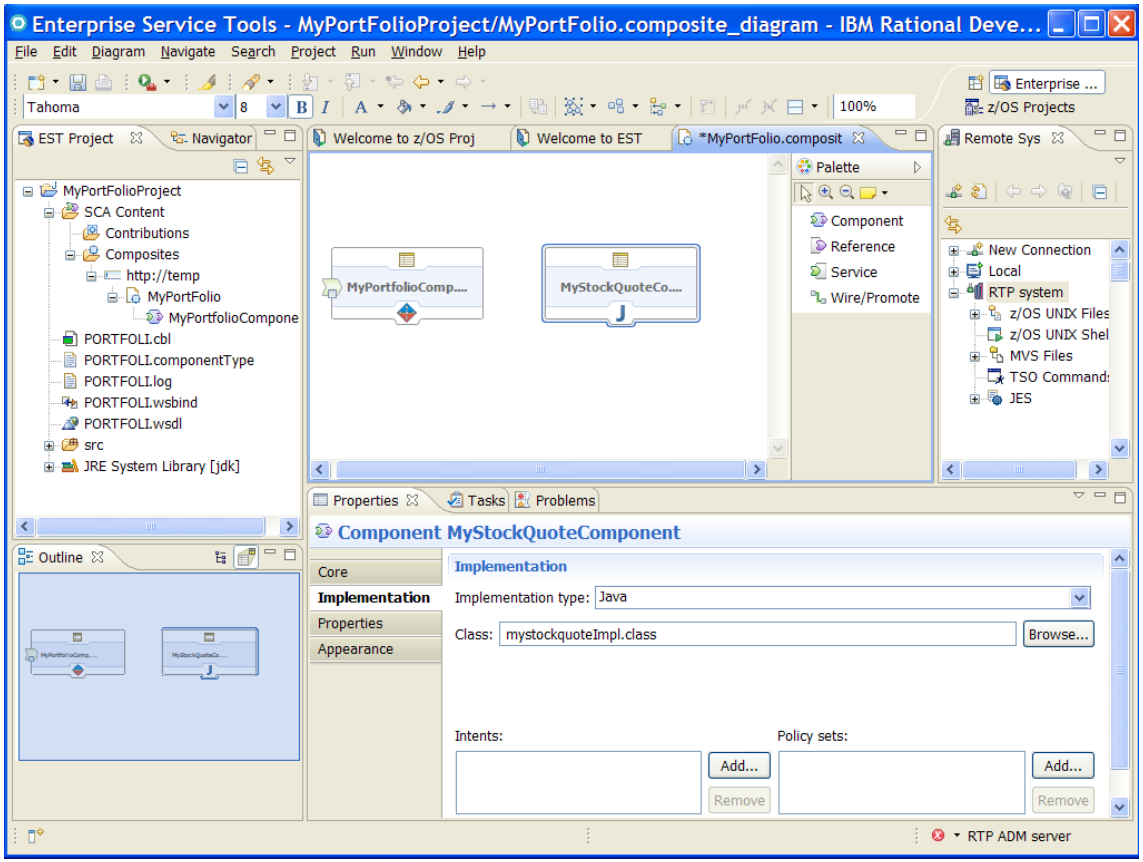
RDz SCA Tooling Support



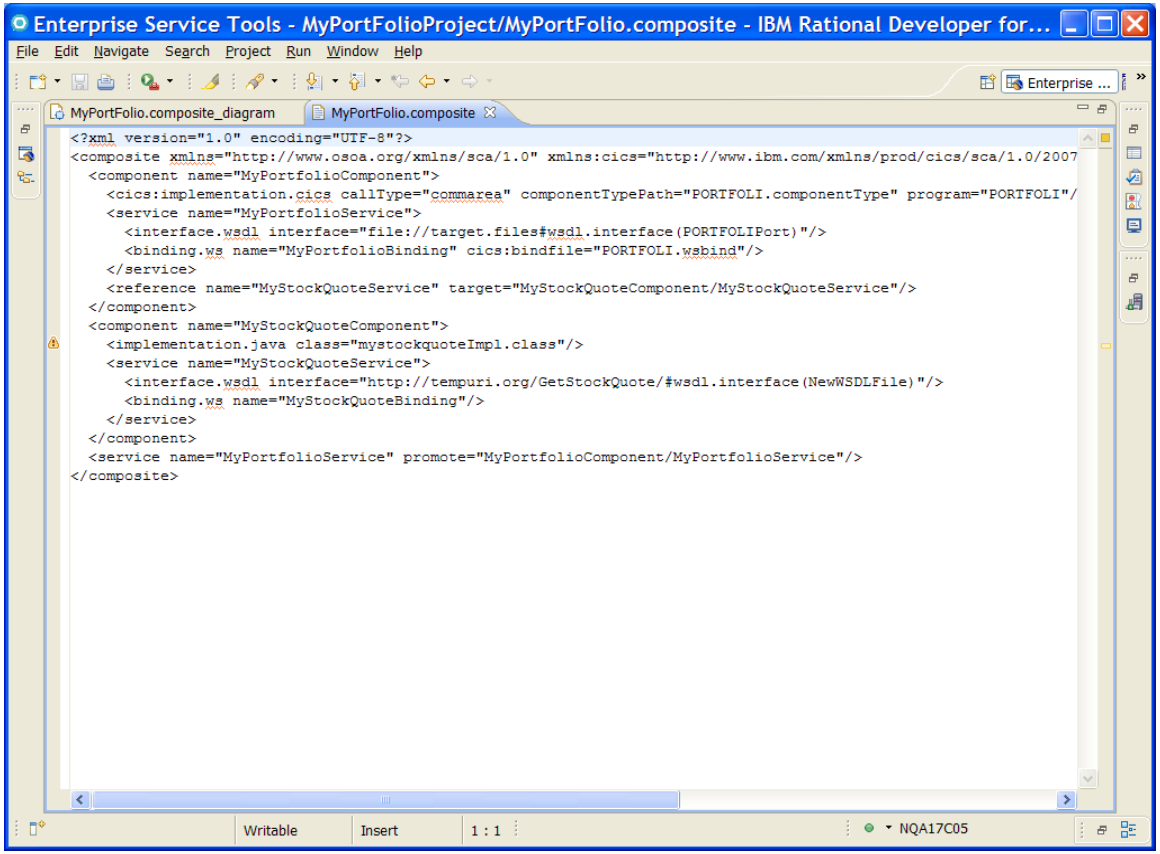
RDz SCA Tooling Support



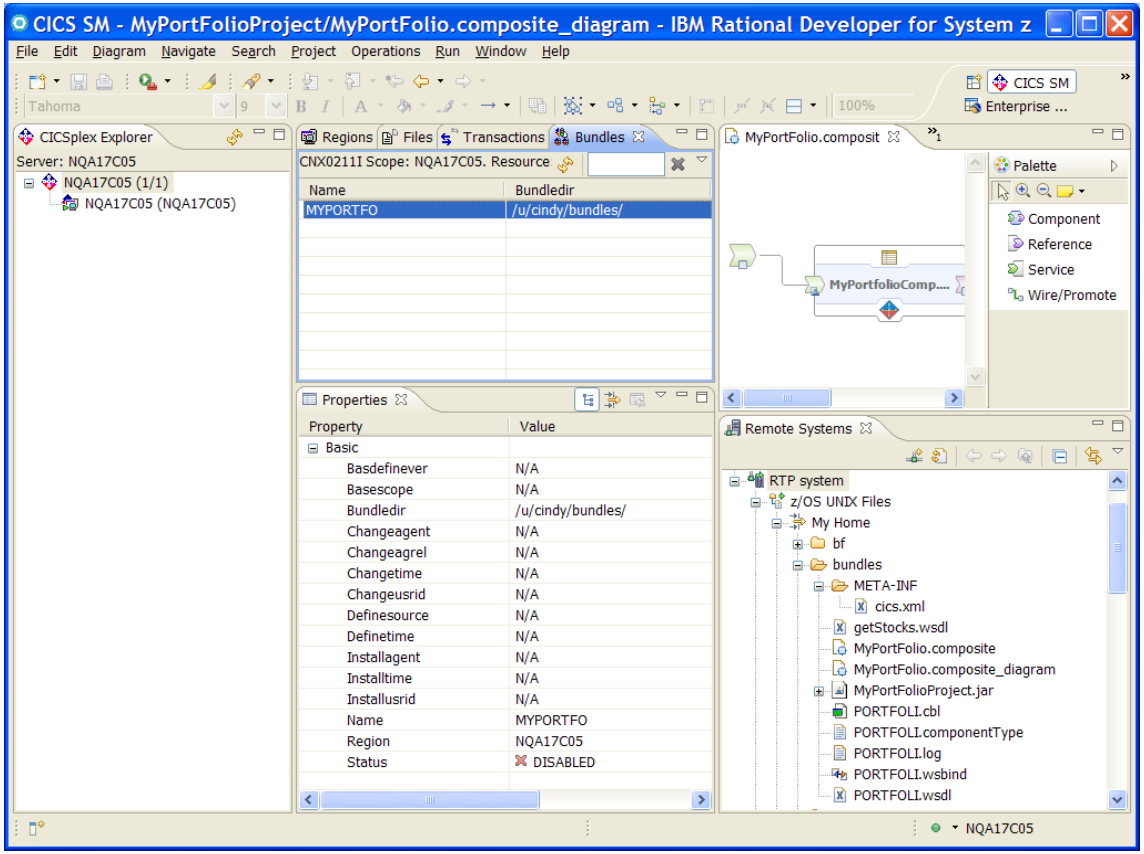
RDz SCA Tooling Support



RDz SCA Tooling Support



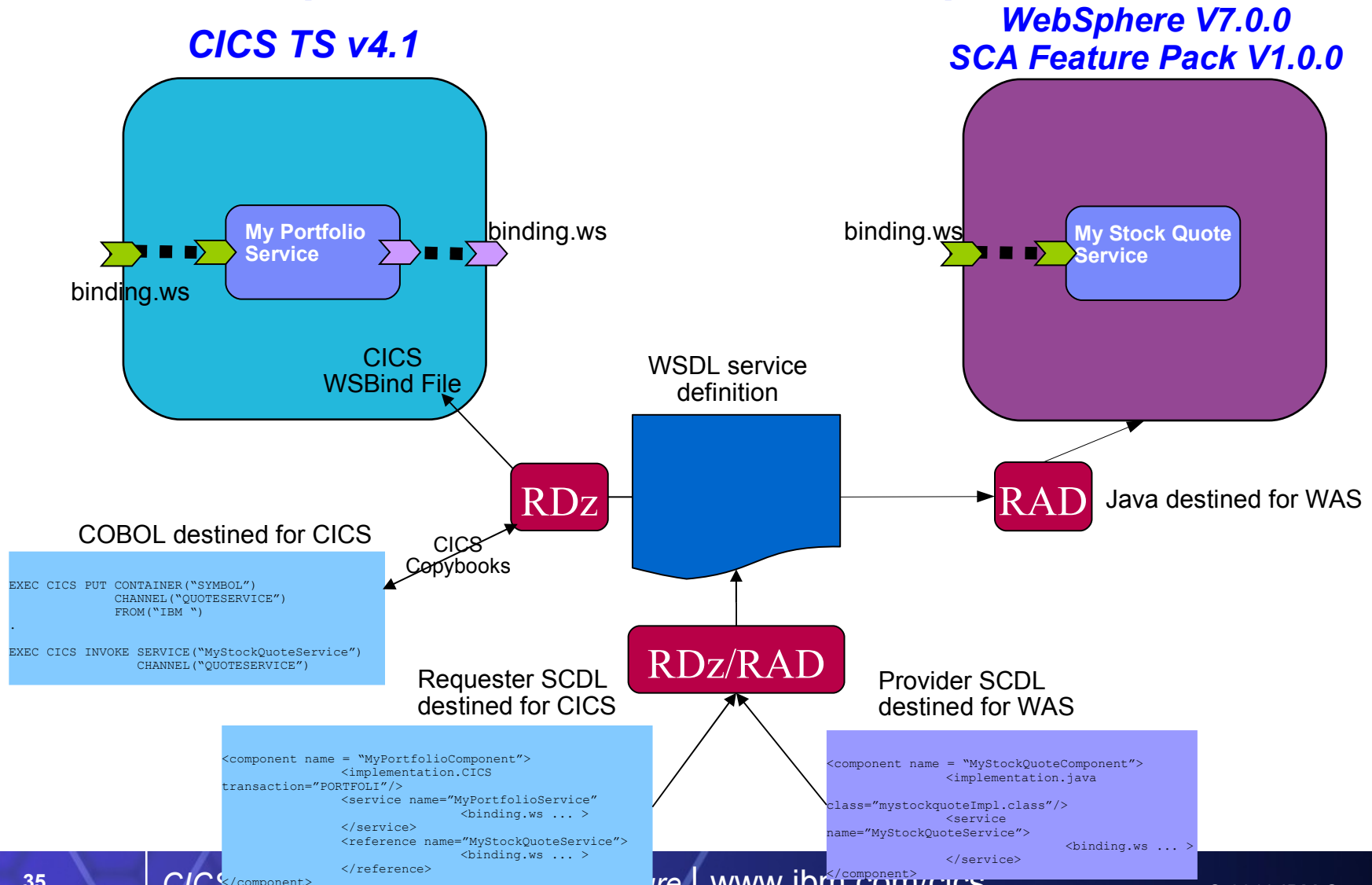
RDz SCA Tooling Support



Sub-agenda – Service Component Architecture

- Quick SCA in WAS recap
- Introduction to SCA support in CICS TS v4.1
- The Portfolio Scenario
- Defining and implementing SCA components using Rational Developer for System z
- **Deploying and running SCA components in CICS TS**

Service Component Architecture – Development Artifacts



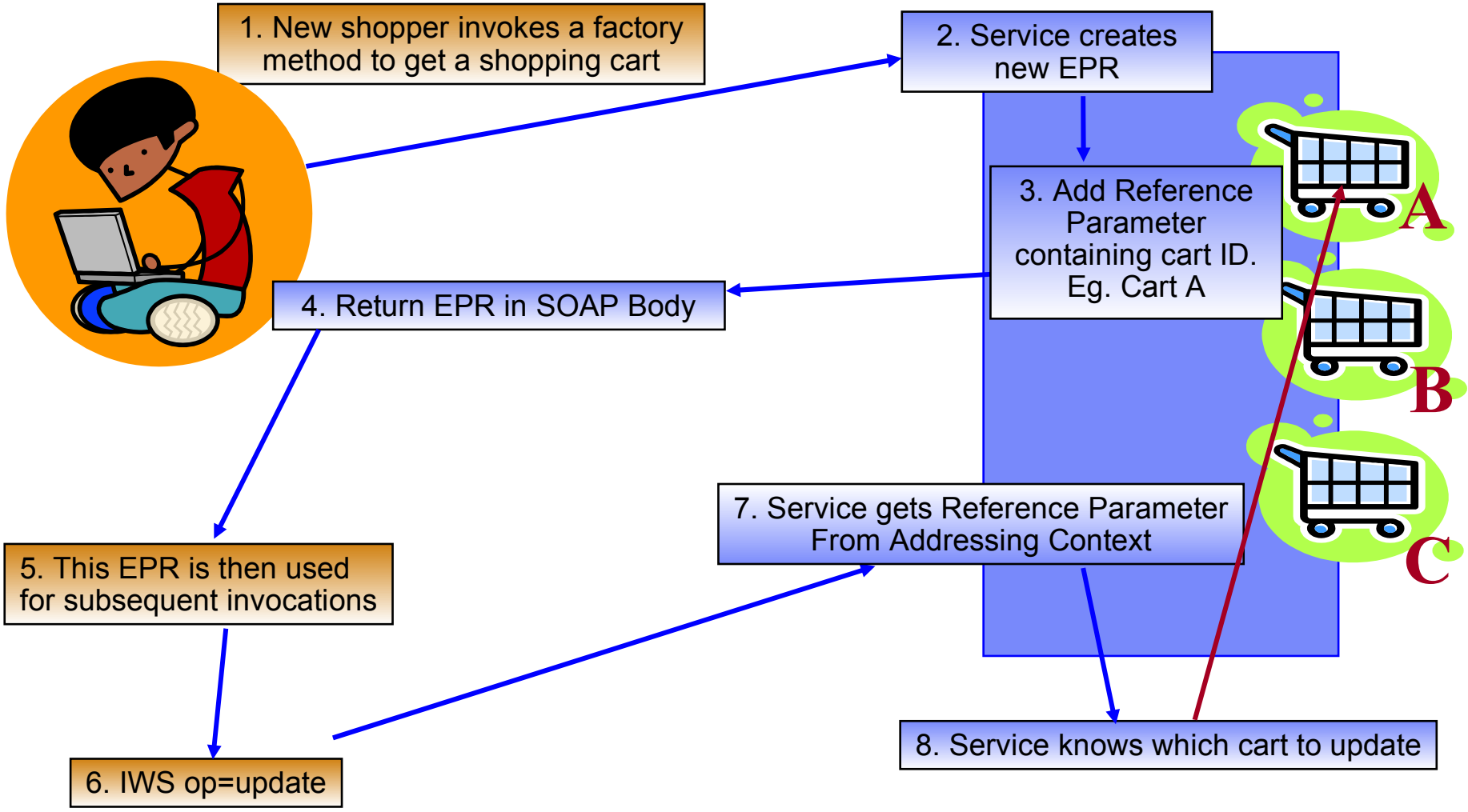
Agenda

- Spectrum of Service enablement formality
- Web Services
  - Service Component Architecture
  - **WS-Addressing**
- Bundles
- WSRR
- XML Transforms and offload
- Java and JavaServers

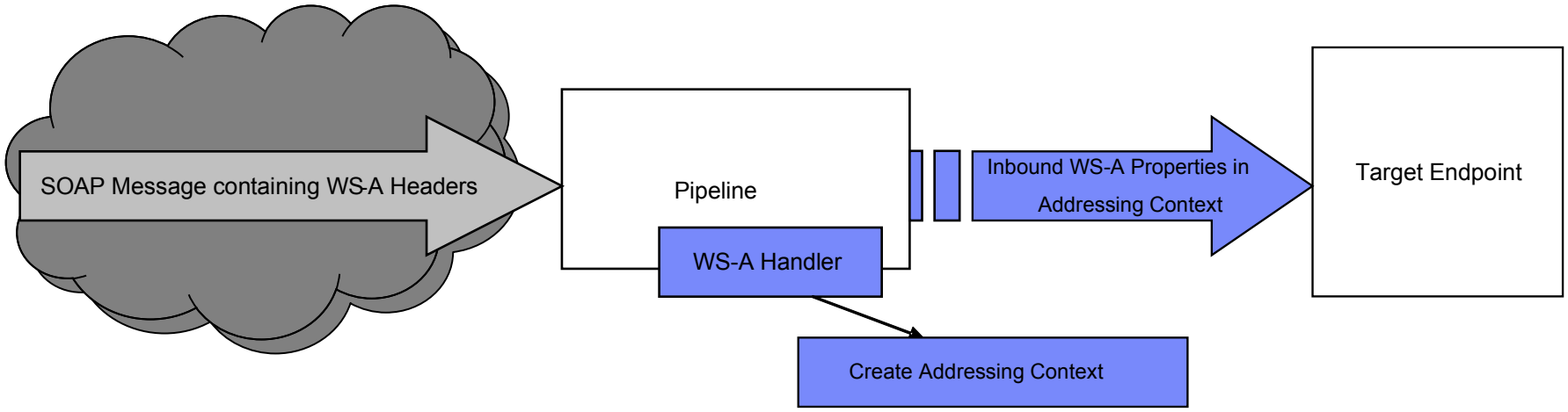
WS-Addressing goals

- Defines transport-neutral mechanisms to address Web services and support message transmission through networks
- Improves interop with other Web Services implementations such as .NET
- XML elements to identify Endpoints: EndpointReferences (EPRs)
  - More than just a URI
  - Can have Reference Parameters and metadata
  - Allows for Psuedo-Conversational style web service requesters in CICS
- WS-Addressing Message Addressing Properties (MAPs)
  - Standard placeholders in the SOAP header for WS-Addressing information
  - Plus reference parameters in target EPR

Example of a WS-Addressing Resource Access Pattern



WS-Addressing in CICS



WS-Addressing in CICS

- **Pipeline configuration**
  - Configure Requester pipeline to use WS-Addressing handler giving specification version
  - Configure Provider pipeline to use WS-Addressing handler
- **Requester**
  - Requester application not aware requests are WS-Addressed
    - CICS handles the required addressing responses
  - Requester WS-A aware
    - Uses EXEC CICS API to set Message Addressing Properties (MAPs)
- **Provider**
  - Provider application not aware request/response is WS-Addressed
    - CICS handles the required addressing responses and routing
  - Provider WS-A aware
    - uses EXEC CICS API to interrogate Addressing Context (e.g. Fetch To EndPointReference (EPR) to extract Reference parameters)

WS-Addressing in CICS

- Requester
  - Requester optionally uses API to create an Addressing Context and modify/add MAPs
  - On INVOKE WEBSERVICE context MAPs converted to WS-A SOAP Headers
  - On return WS-A SOAP Headers converted to MAPs in Addressing Context
- Provider
  - Provider optionally uses API to interrogate Addressing Context and modify/add MAPs
  - On service return MAPs converted to WS-A SOAP Headers
  - ReplyTo or FaultTo EPR used for reply endpoint. An Anonymous address is the default. I.E. Reply back to requester.

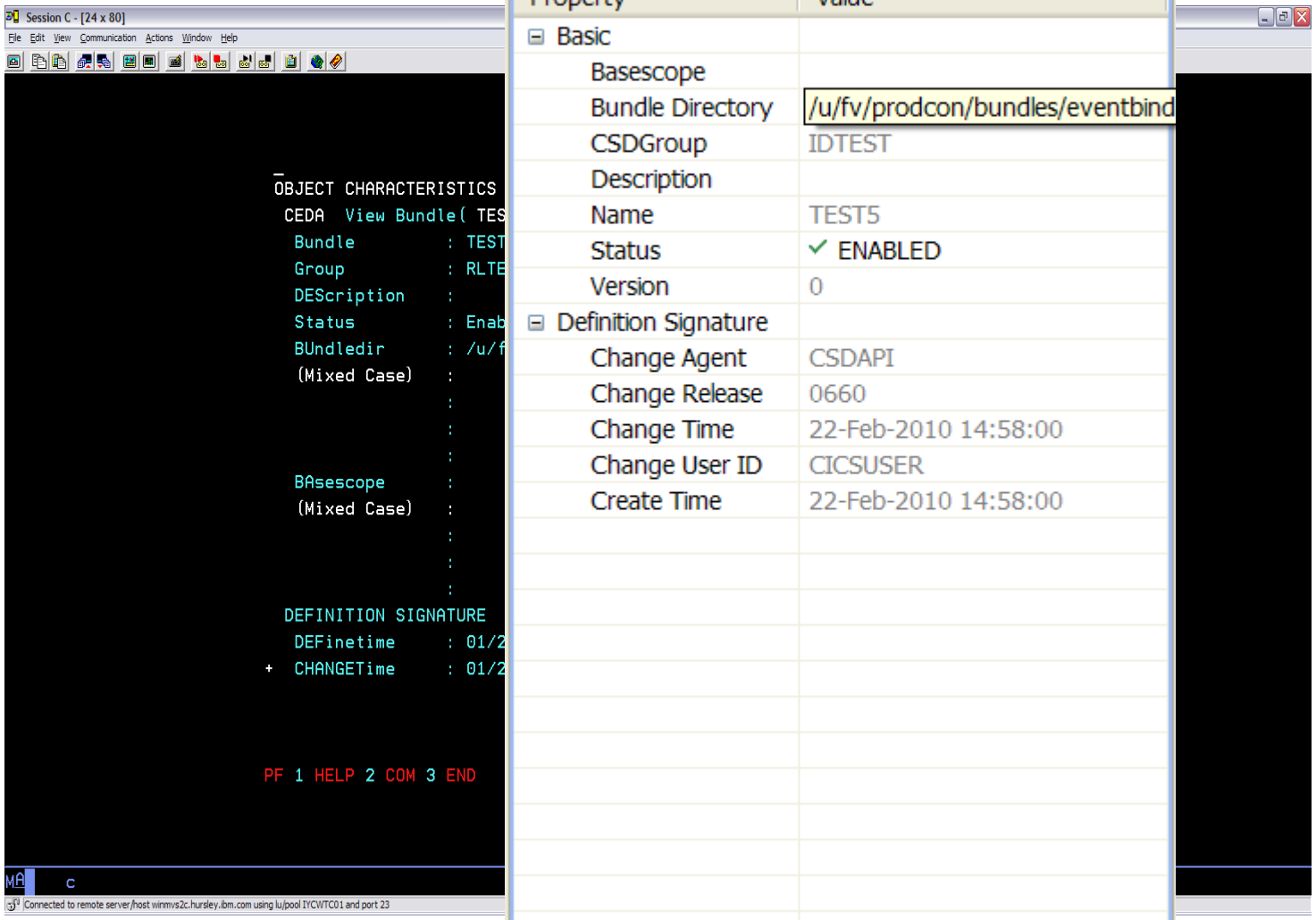
Agenda

- Spectrum of Service enablement formality
- Web Services
  - Service Component Architecture
  - WS-Addressing
- **Bundles**
- WSRR
- XML Transforms and offload
- Java and JavaServers

Why Bundles?

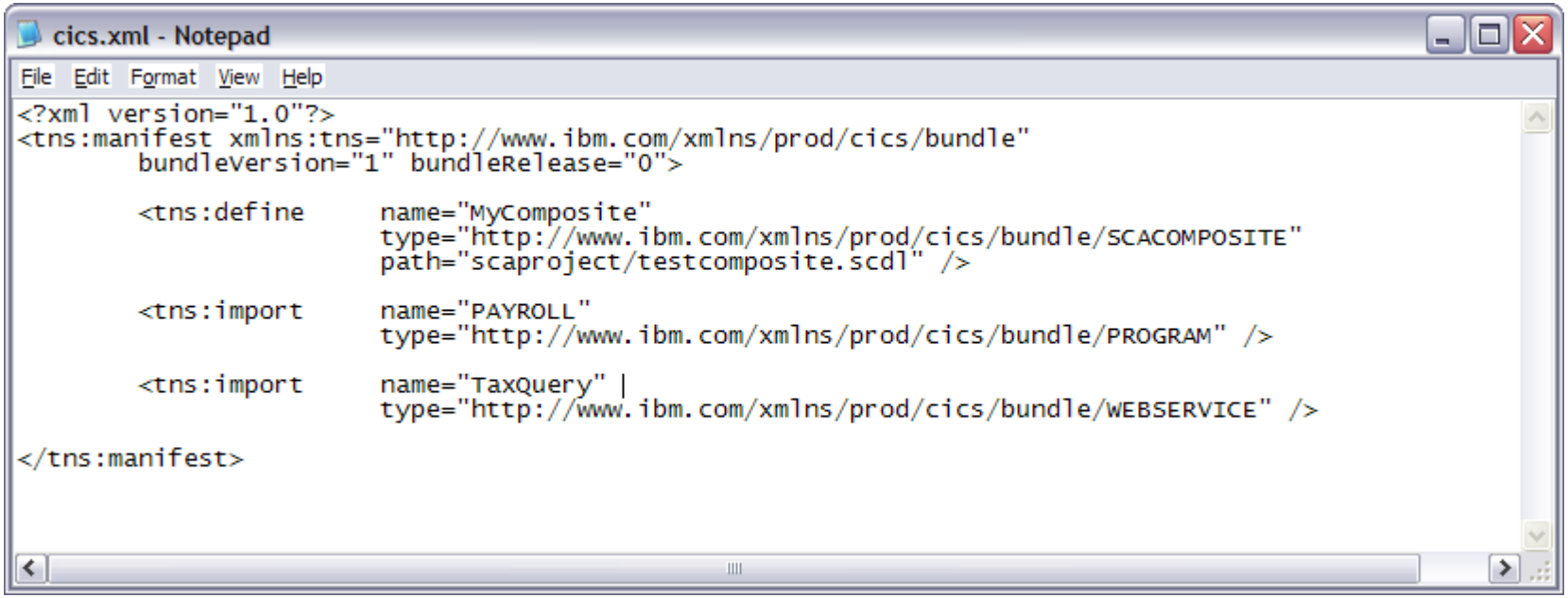
- Similar in concept to OSGI bundles for Java / Eclipse / WAS
- Provide a deployment and life cycle grouping for related application artefacts
  - Provides a single point of management and control
  - The artefacts can be from a number of resource spaces
- Allow such a grouping to express and police its dependencies on other
  - Can express functional or resource related dependencies
- Extensible
  - Provide an extension point for Vendor or User artefacts to be deployed and managed alongside CICS Resources
  - Manifest File describes contents - “Imports”, “exports” , “defines”
  - User extensible via Callback program
- CICS Resources which are “bundle-enabled”:
  - Event Binding, XSD Bindfile, SCA Composite

BUNDLE Resource

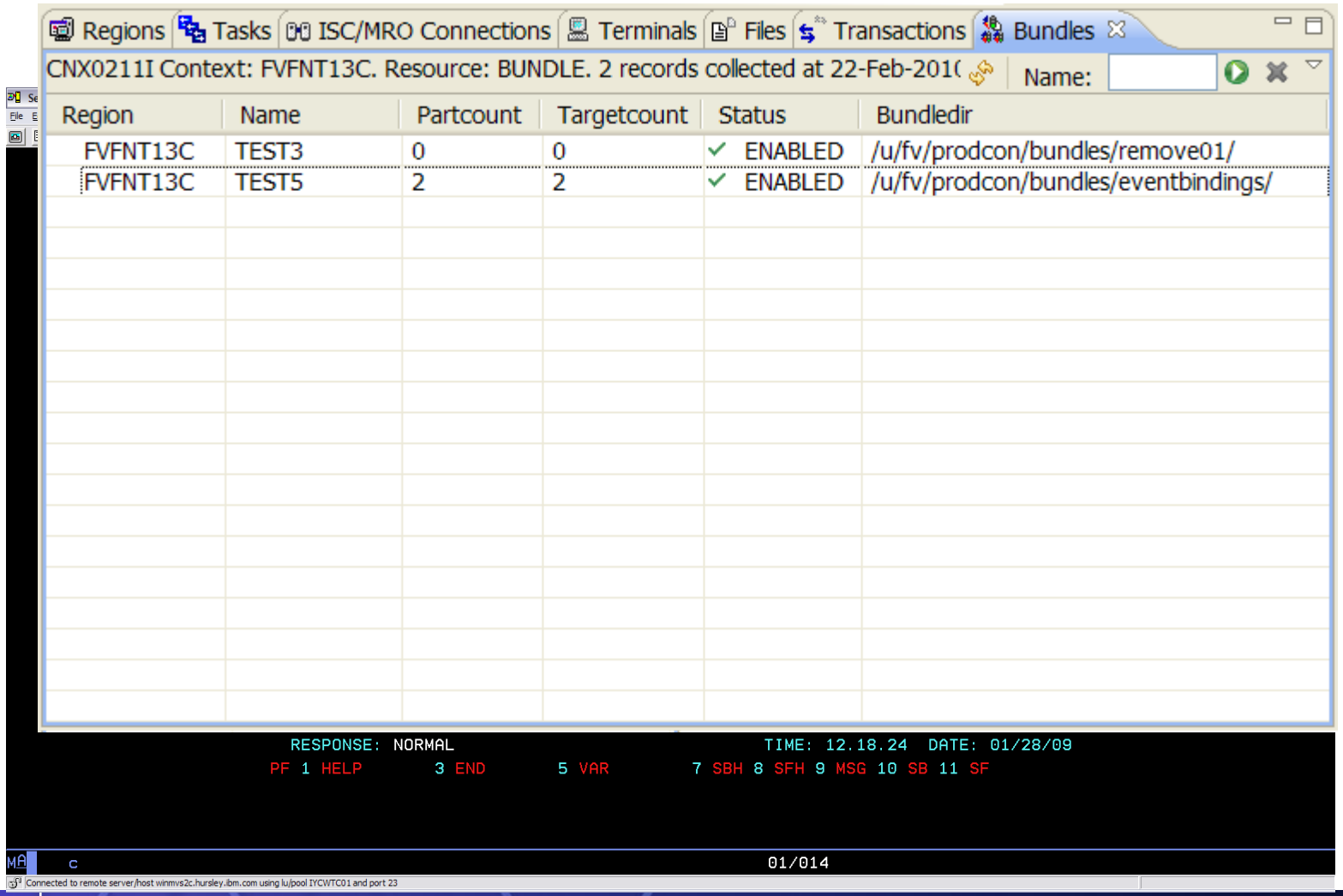


Bundle Contents

```
.../bundle5
      /META-INF
            cics.xml
      /scaproject
            testcomposite.scdl
```



Bundle Operations



Agenda

- Spectrum of Service enablement formality
- Web Services
  - Service Component Architecture
  - WS-Addressing
- Bundles
- **WSRR**
- XML Transforms and offload
- Java and JavaServers

WebSphere Service Registry and Repository

- ***Enables governance*** – Configurable service life-cycle, classifications and access controls
- ***Manages service meta-data*** – Providing better search granularity than most UDDI-based products
- ***User-friendly UI*** – Facilitates design time discovery
- Provides ***location transparency*** through runtime access
- Stores all service artifacts – ***Not just WSDL***
- Provides ***fully configurable*** functionality to classify services
- ***Supports state model functionality*** – Manages service life-cycles in a shared environment
- ***Service notification*** – facilitates communication between service consumers and providers
- ***Enforces*** consumer access to services
- ***Simple*** version management functionality

Notes

- WebSphere Service Registry and Repository provides a central repository for storing service interfaces and associated meta-data.
- It enables the management of services throughout their life-cycle, from concept through to retirement. The life-cycle is fully configurable using WebSphere Integration Developer. Thus, the SOA governance model for the service life-cycle can be represented by defining the life-cycle of the services in your organization and defining the authorization process for an application to move from one phase of the life-cycle to another. Once the life-cycle is defined, it can be installed into WSRR and used as the basis for managing the life-cycle of your services.
- Another area in which WSRR enables governance is through the use of classifications. The classification systems defined as part of your organizations governance model can be imported into WSRR and used to classify the services described in it.
- WSRR also enables documents to be tagged with properties and relationships to be defined between documents. The WSRR user interface is web based and provides search capabilities to enable services to be easily located.

CICS Support for WSRR

- DFHLS2WS
  - Can now publish the generated WSDL to WSRR
  - Allows specification of the WSDL meta-data
  - SSL support
- DFHWS2LS
  - Can now retrieve WSDL from WSRR

Notes

- The Web services assistant batch jobs DFHWS2LS and DFHLS2WS have new parameters to support interoperability with the IBM WebSphere Service Registry and Repository (WSRR) server. You can optionally secure the network connection to WSRR using secure socket layer (SSL) encryption.
- This support requires mapping level 1.2 or higher.

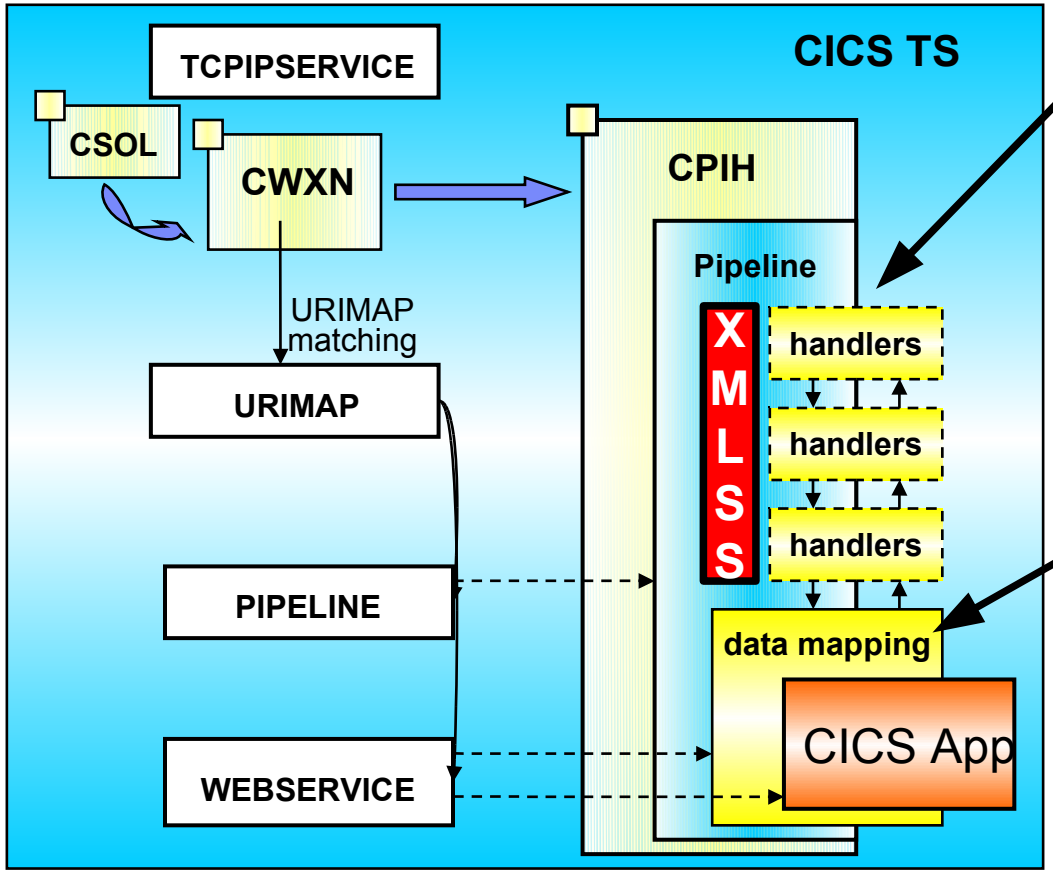
Agenda

- Spectrum of Service enablement formality
- Web Services
  - Service Component Architecture
  - WS-Addressing
- Bundles
- WSRR
- **XML transforms and XMLSS offload**
- Java and JavaServers

XML to language structure mapping services

- New API to convert between XML and application data
  - Map between XML and language structure
- `EXEC CICS TRANSFORM TRANSFORMTYPE(XMLTODATA ¦ DATATOXML)`
- Command options depend on the direction of the transformation
  - XMLTODATA requires an XMLTRANSFORM resource, providing metadata used for the transformation
  - XMLTRANSFORM resource installed via Bundle support

CICS and z/OS XML System Services Parser (XMLSS)



- First parse of message now uses XMLSS
  - Locates the SOAP headers
  - Handler execution is outside of XMLSS
- XMLSS is zAAP eligible
  - Offloads MIPS for this element of the processing
- "Shredding" of body into Containers or Commareas is ***NOT*** XMLSS
  - and so ***NOT*** zAAP eligible.

Agenda

- Spectrum of Service enablement formality
- Web Services
  - Service Component Architecture
  - WS-Addressing
- Bundles
- WSRR
- XML transforms and XMLSS offload
- **Java and JavaServers**

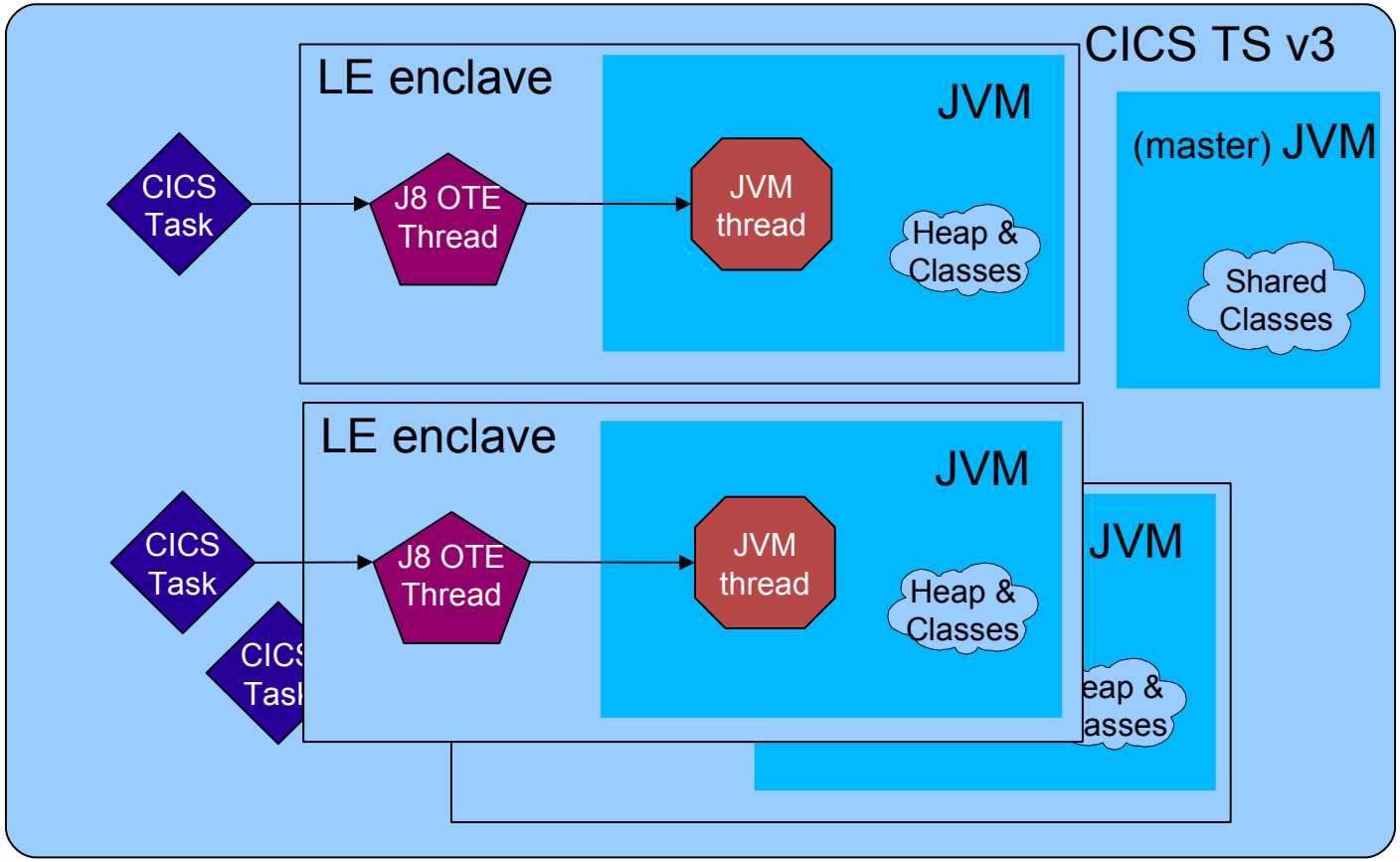
JVMServers

- JVM implementation in CICS continues to evolve
  - JDK 1.1.8, HPJ and Hotpooling, IBM Persistent Reusable JVM (Shiraz), Continuous mode, Java 5, Java 6, ...
- JVMs in CICS TS V2 and V3
  - Single task, serial reuse
  - Large memory footprint
  - Excellent isolation characteristics
- JVMServers in CICS TS V4.1
  - Multiple tasks (threads) in a JVM concurrently
  - Larger capacity
  - Risk of collateral damage
  - Not for customer application use in V4.1

JVMPool Architecture - CICS TS v3 (and v2)

Single CICS task dispatched into a JVM in the pool at a time. So concurrent task count limited to the number of JVMs that can fit in the region.

Result is about 20 tasks/JVMs concurrently in each region.

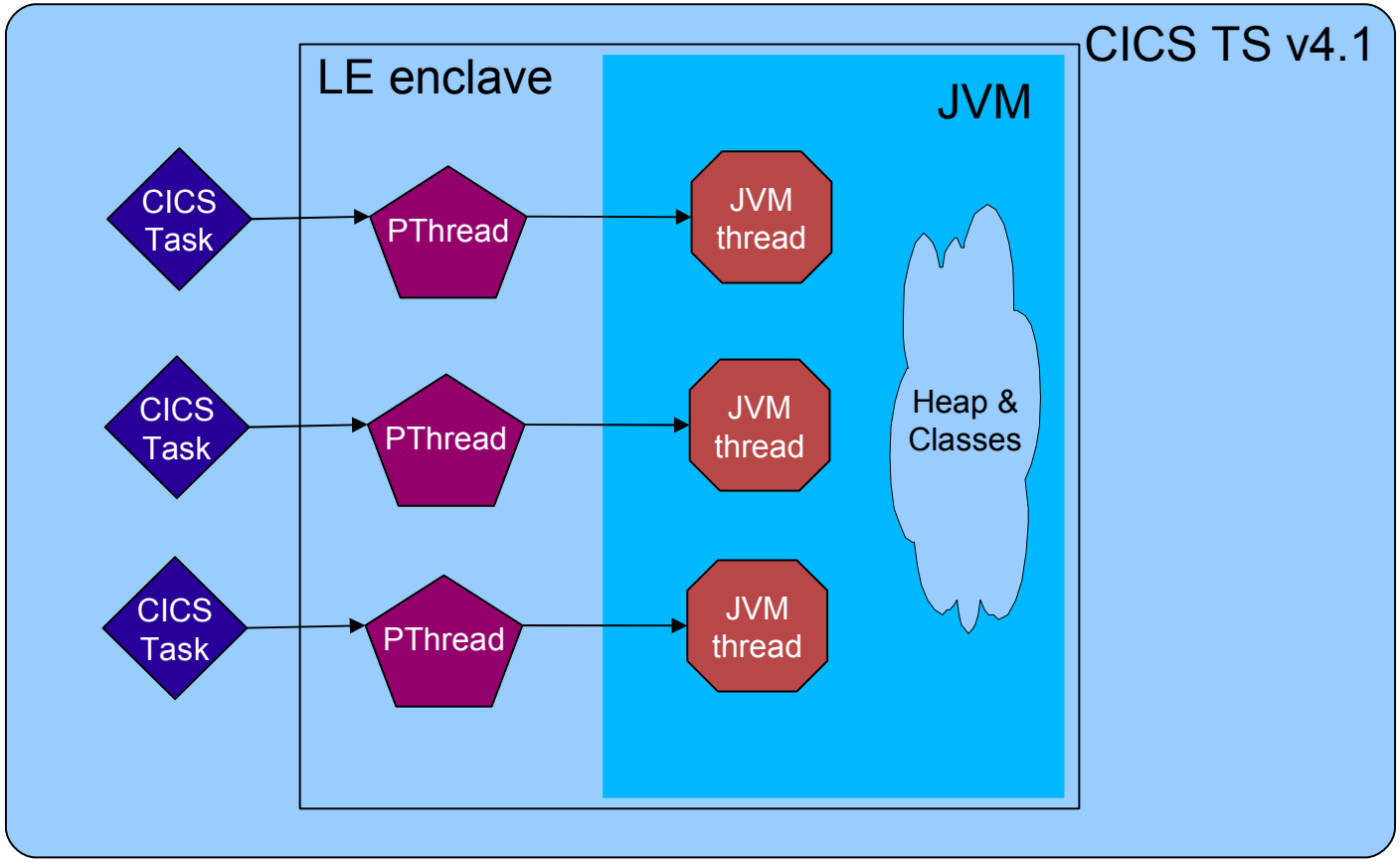


JVMServer Architecture

Can attach multiple pthread/T8/CICS tasks to the JVM at the same time.

Therefore serve more requests using a single JVM.

Result is 100s tasks per region.



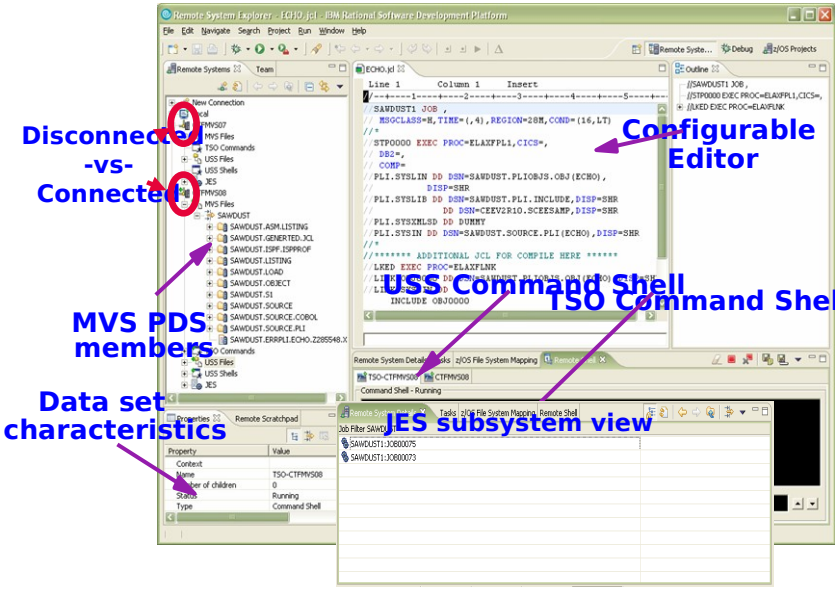
Revitalizing CICS Applications

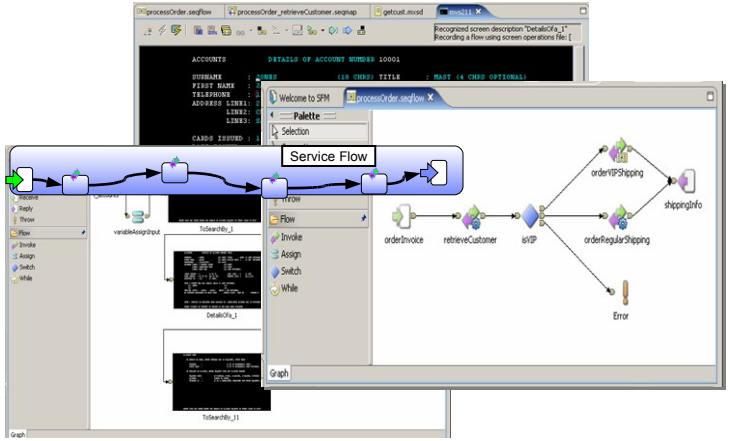
Developing Applications with Rational Developer for System z

SOA on your terms and our expertise – www.ibm.com/cics

Rational Developer for System z (RDz)

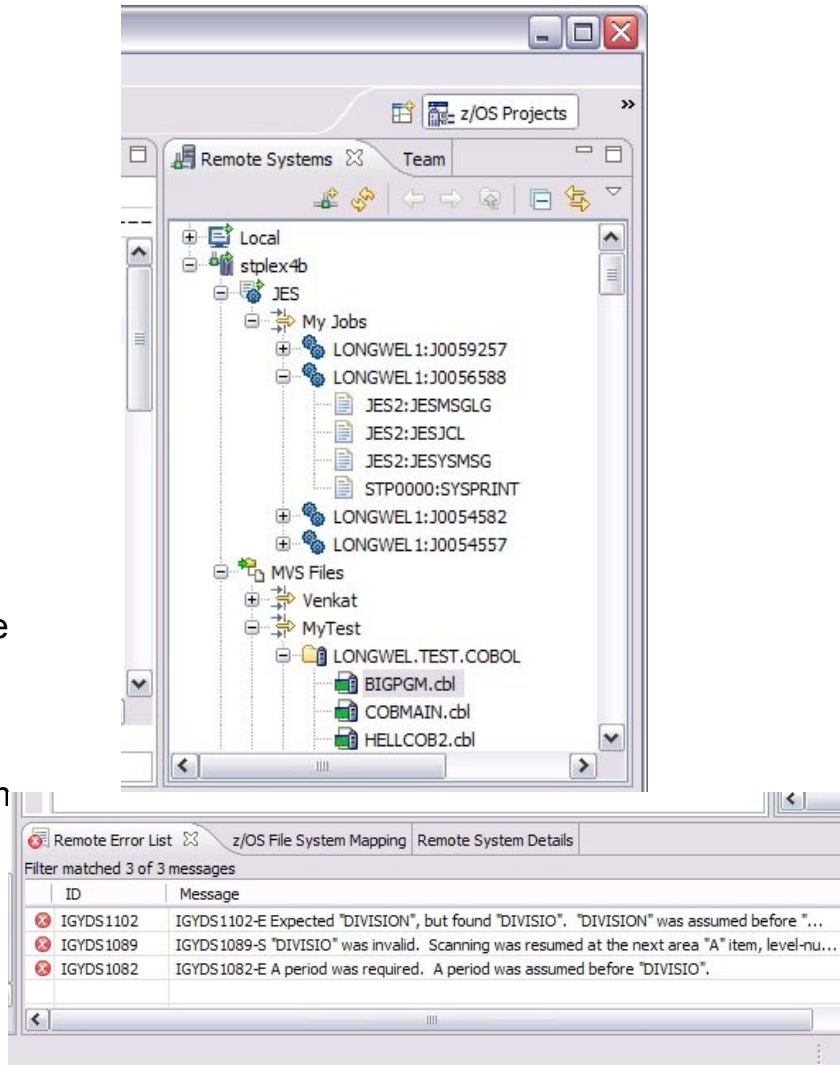
- **What is RDz**
  - Eclipse-based IDE speeding modern composite (SOA) application development
- **RDz supports Enterprise Modernization**
  - Links WAS and core system z processing
  - Supports common IDE for COBOL, PL/I, C, C++, HLASM, Java, and web services
  - Transforms UML to COBOL source code
  - Provides interactive access to z/OS for development, debug, job generation, submission, monitoring, command execution
  - Supports new and existing runtimes (CICS, IMS, Batch, USS, DB2 SP, WAS)
- **RDz supports SOA**
  - Enables CICS and IMS applications for web services and SOA including System z runtime micro-flows
  - Supports for J2EE, JCA, XML, web services





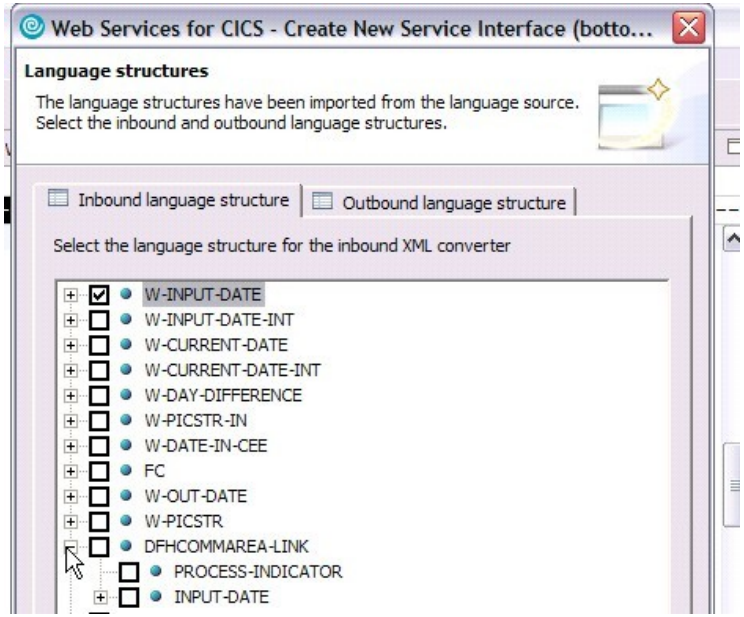
Gain productivity and ROI

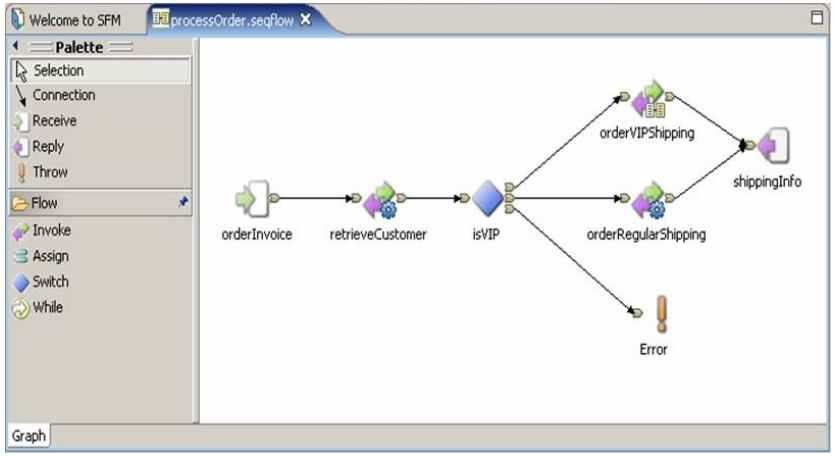
- **Real-time mainframe connection with Remote System Explorer**
  - Direct access to host subsystems and file system
  - Project assets connected to host resources
- **Eclipse-based editor (LPEX)**
  - Green-screen compatible environment but with eclipse flavor
    - Content assist; Discover/Link to include files; Error markers, etc
- **Local and Remote syntax check**
  - Mainframe programs can be compiled and errors listed in the problems view
  - Saves time and cost (MIPS reduction) over traditional compile process on host
- **Remote Debugging with Debug Tool integration**
  - Debug programs as they execute on the host in real-time from eclipse unified debugger
  - Debug composite applications linking seamlessly from WAS to host runtimes



Move to SOA easily

- Customers today tend to focus on applications which already exist on the mainframe (limited new workload creation)
- RDz can help maintain and componentise existing mainframe applications to prepare host systems for a SOA architecture
- RDz offers XML services for the Enterprise
  - Wizard-driven web-service creation
  - Works with existing host programs without changing program code (lost cost, low risk)
  - Creates program skeleton for new program creation
  - Works for major mainframe runtimes (Batch, CICS, IMS)
- CICS Service flow
  - Create orchestration logic for combining multiple green screen transactions or service calls into a single web service





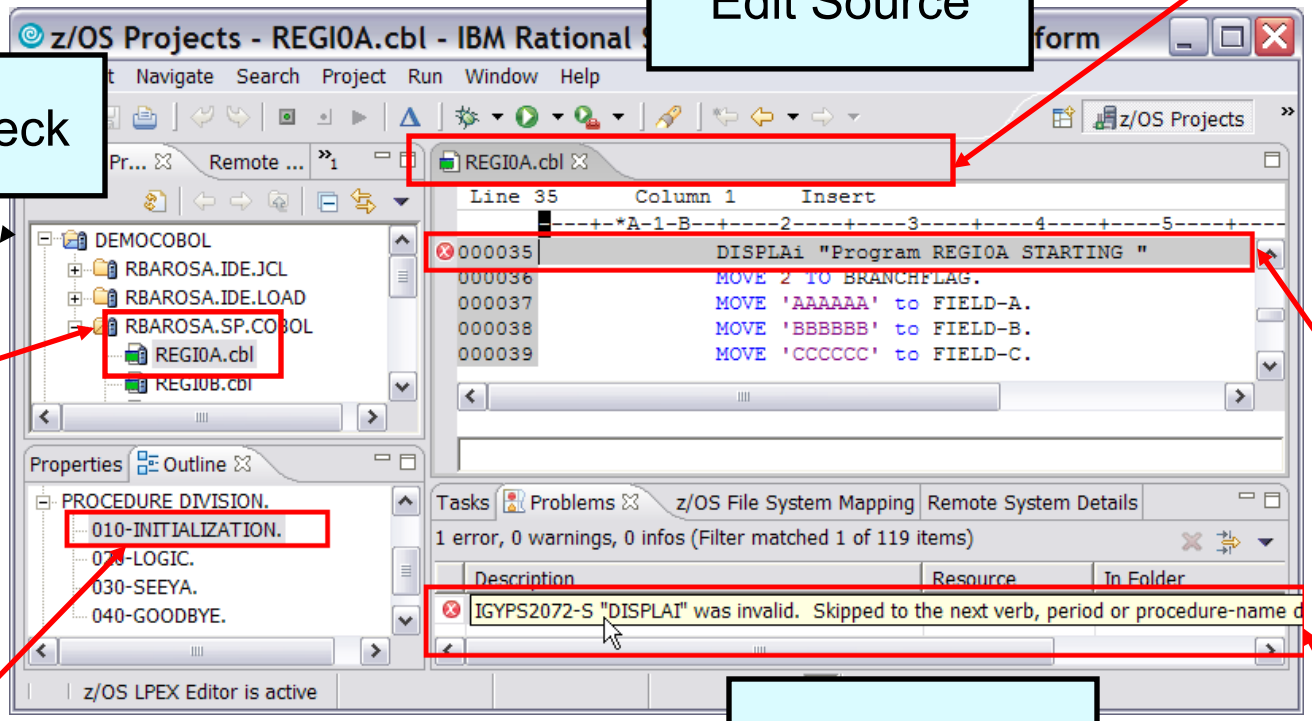
RDz-based development

- Common development environment for COBOL, PL/I, C/C++, and Java
- Simplified development with more information at your fingertips

Edit Source

Open and edit multiple source and JCL members simultaneously

Syntax Check



Submit jobs, access job output, or open source members with a single click

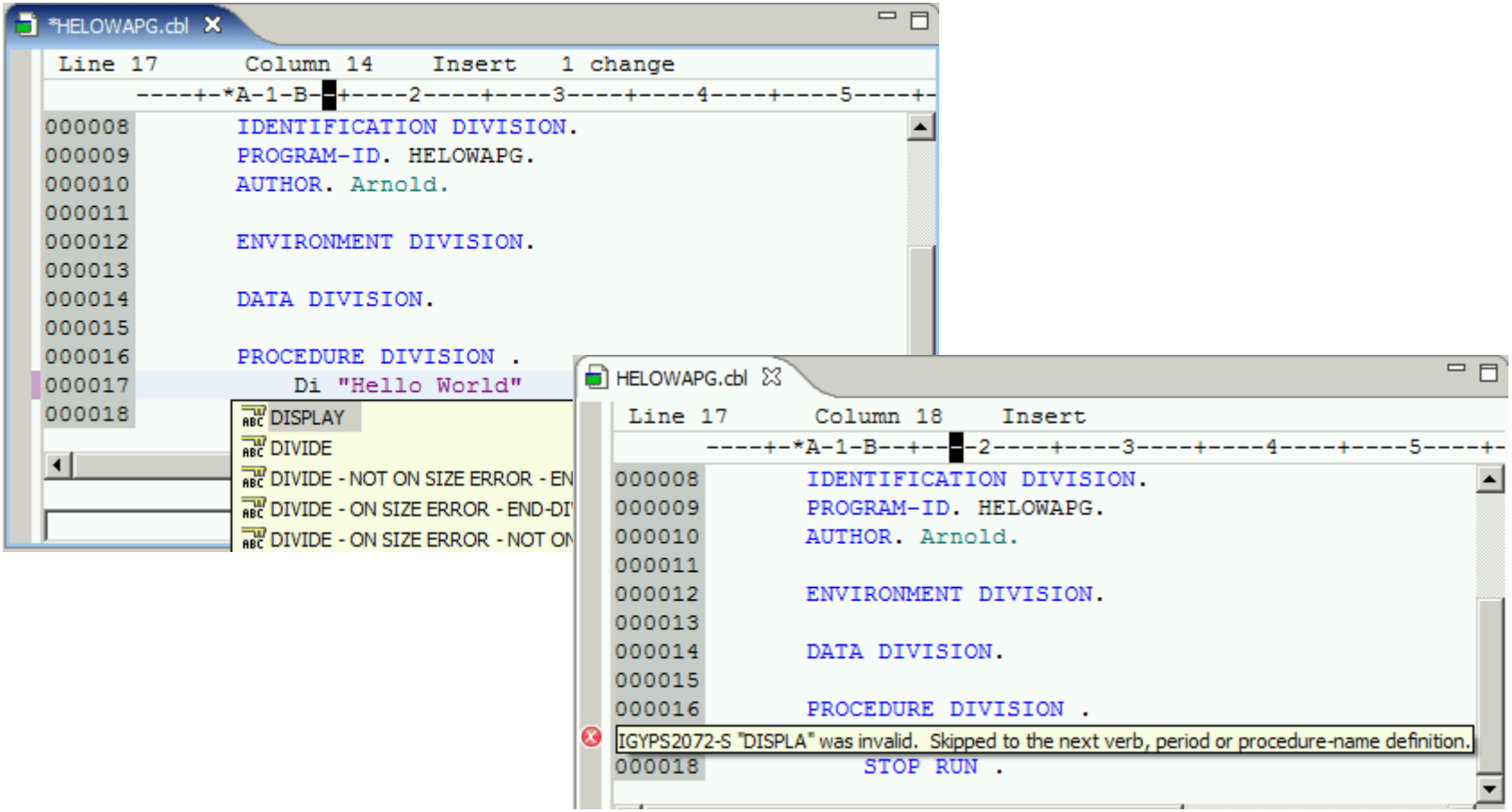
Statement in error indicated in source

Outline view presents COBOL structure

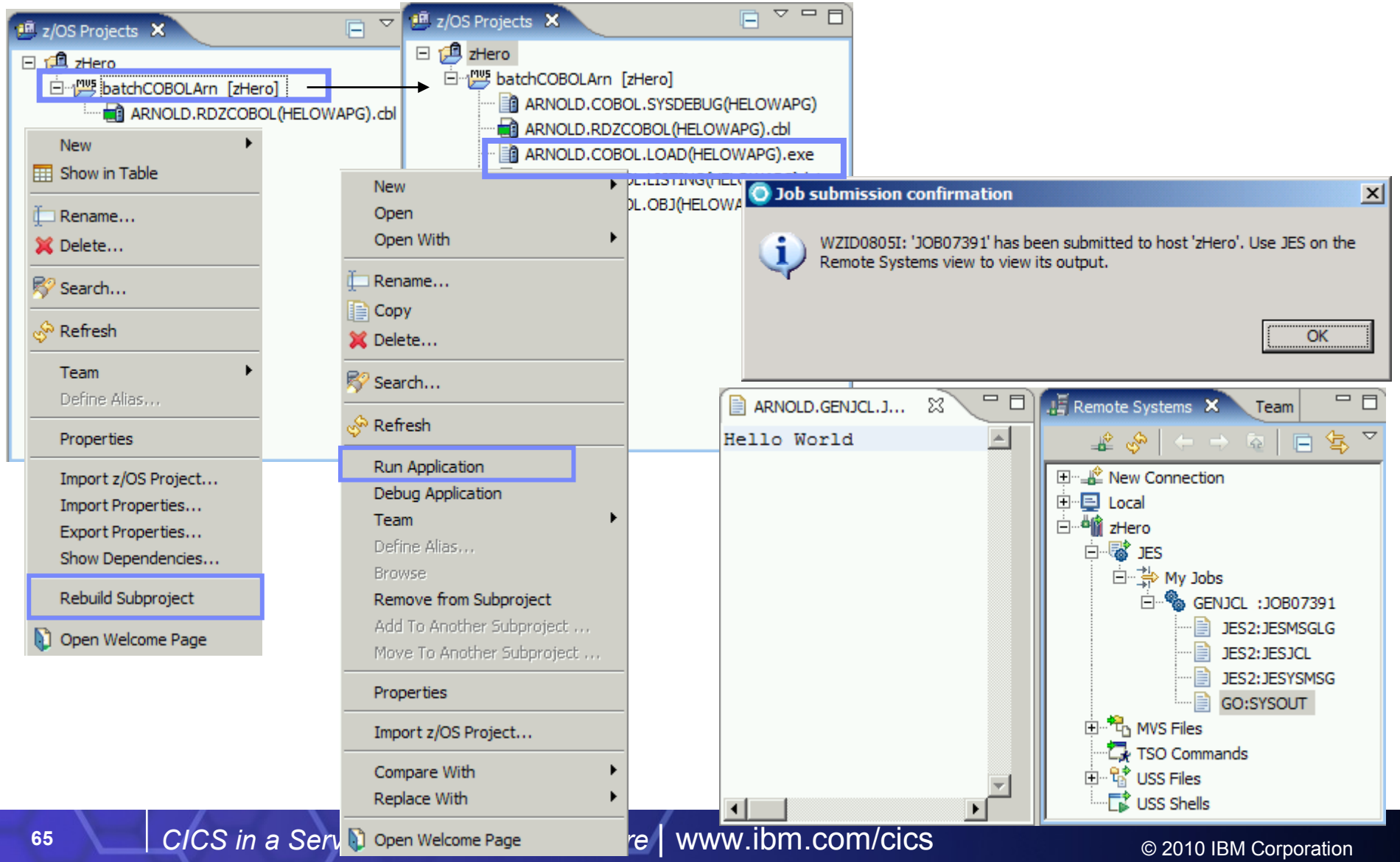
Double-Click on the Error

Error list in Problems view

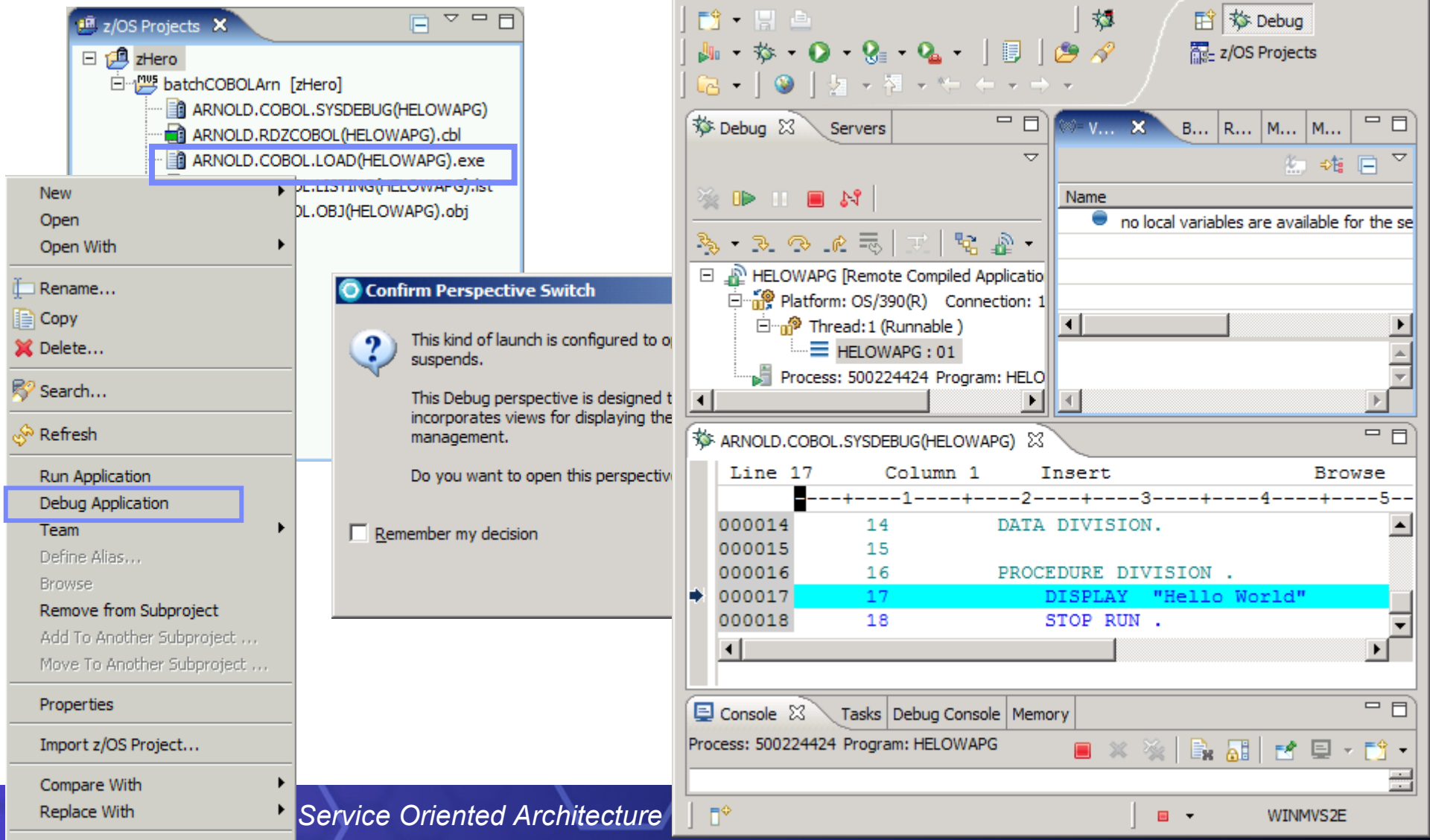
Member – Edit & Syntax Check RDz



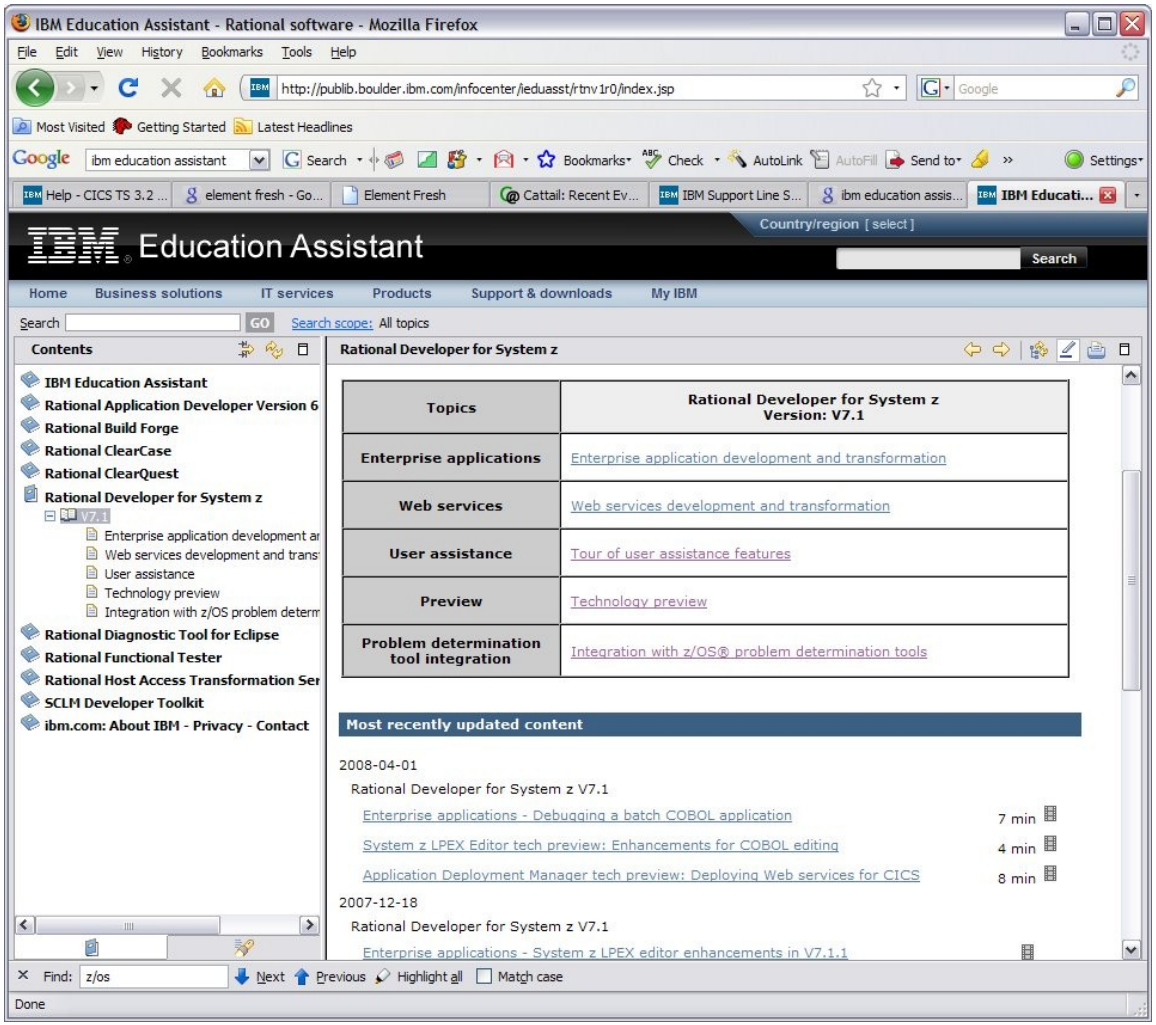
Member – Compile, Link, Go



Member - Debug



And here are some more RDz flash demos to show you how it all works



http://publib.boulder.ibm.com/i

Note : it is possible to hack around with the URL and download the swf files. There is over 30 RDz demos (>100MB total) of really good stuff to grab

Revitalizing CICS Applications

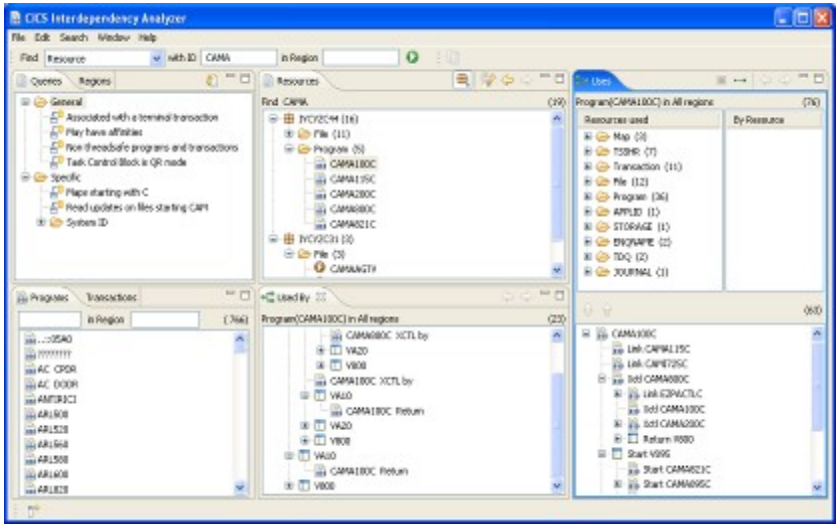
How CICS IA can help

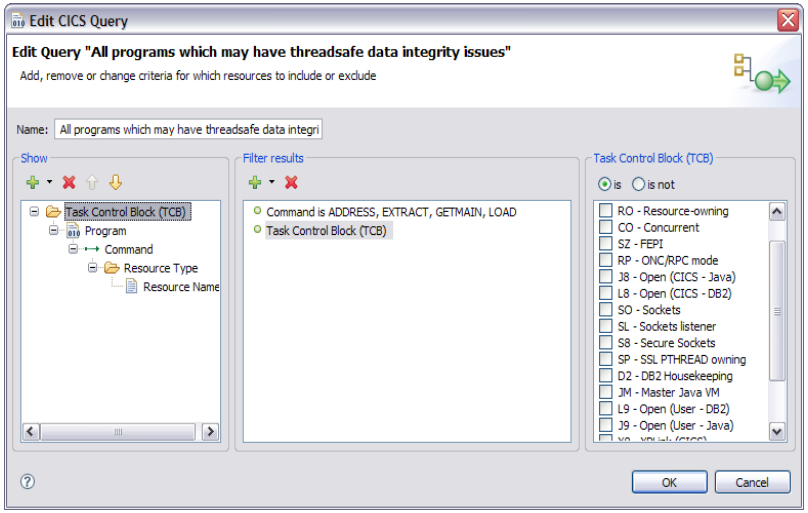
SOA on your terms and our expertise – www.ibm.com/cics

Application Insight

CICS Interdependency Analyzer

Need to extend existing CICS applications





Solution:

- CICS IA collects key relationship data
- CICS Explorer plug-in provides insight:
  - Using Programs as Web services
  - Locating Business Event sources
  - Identifying Atom feeds, thread-safe programs
  - Programs using files, message queues, and database tables
  - Command Flows, Natural and Adabas, etc etc

Value:

- Reduced risk of problems changing complex applications when under time pressure or without deep CICS skills
- Bring new staff up to speed more quickly
- Deliver situational applications to broader user community

What is CICS Interdependency Analyzer ?

- CICS Interdependency Analyzer for z/OS
  - Run-time tool for use with CICS TS for z/OS
  - Identifies the sets of resources used by CICS transactions, and their relationships to other resources
  - Consists of,
    - run-time collector
    - query interface
    - batch reporter
    - load module scanner
    - CSECT scanner
    - CICS IA plug-in for CICS Explorer
- Program Product - 5655-U86
  - Not part of CICS Transaction Server for z/OS

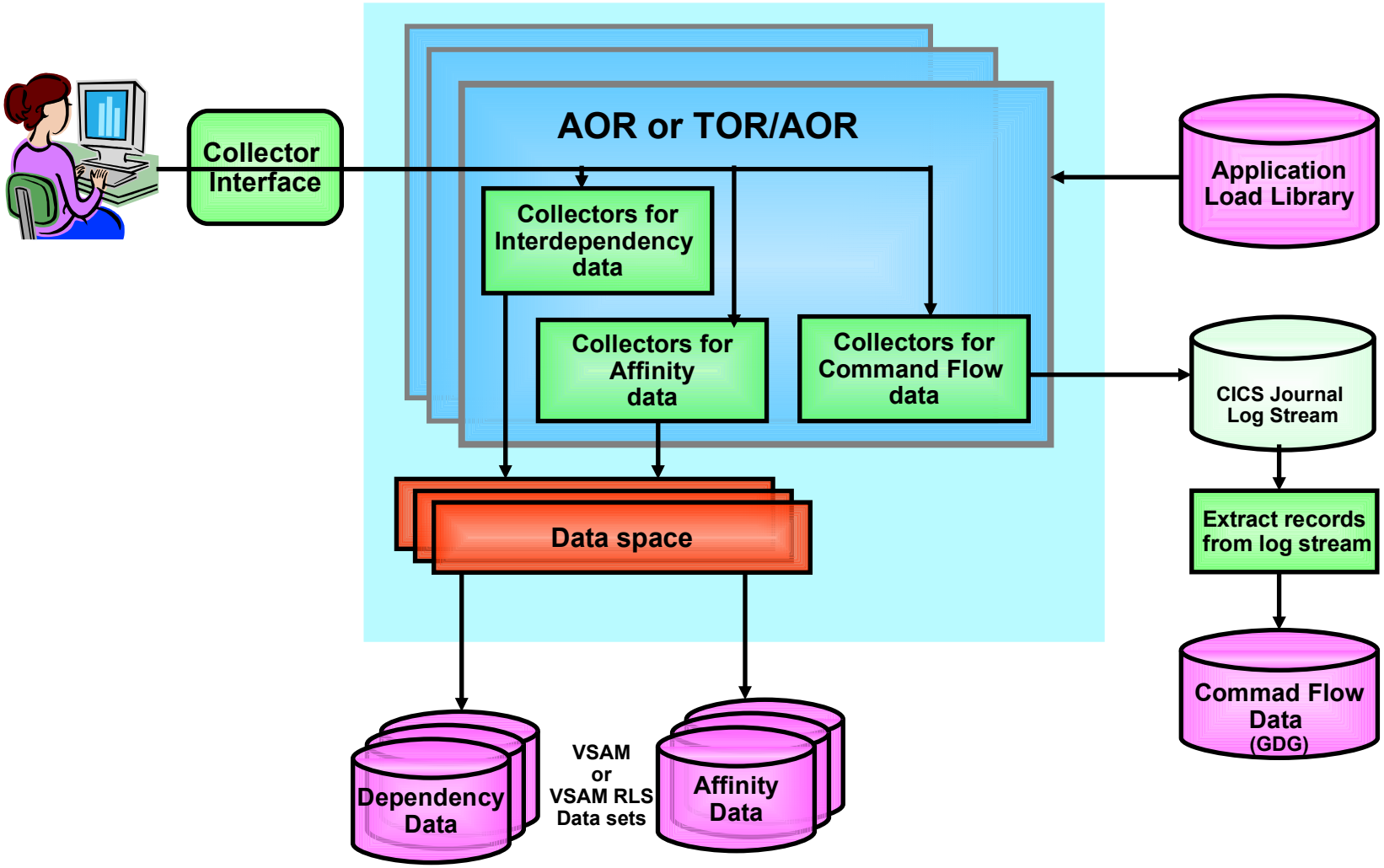
CICS IA Benefits

- Enables you to understand the relationships between resources used by CICS and its applications.

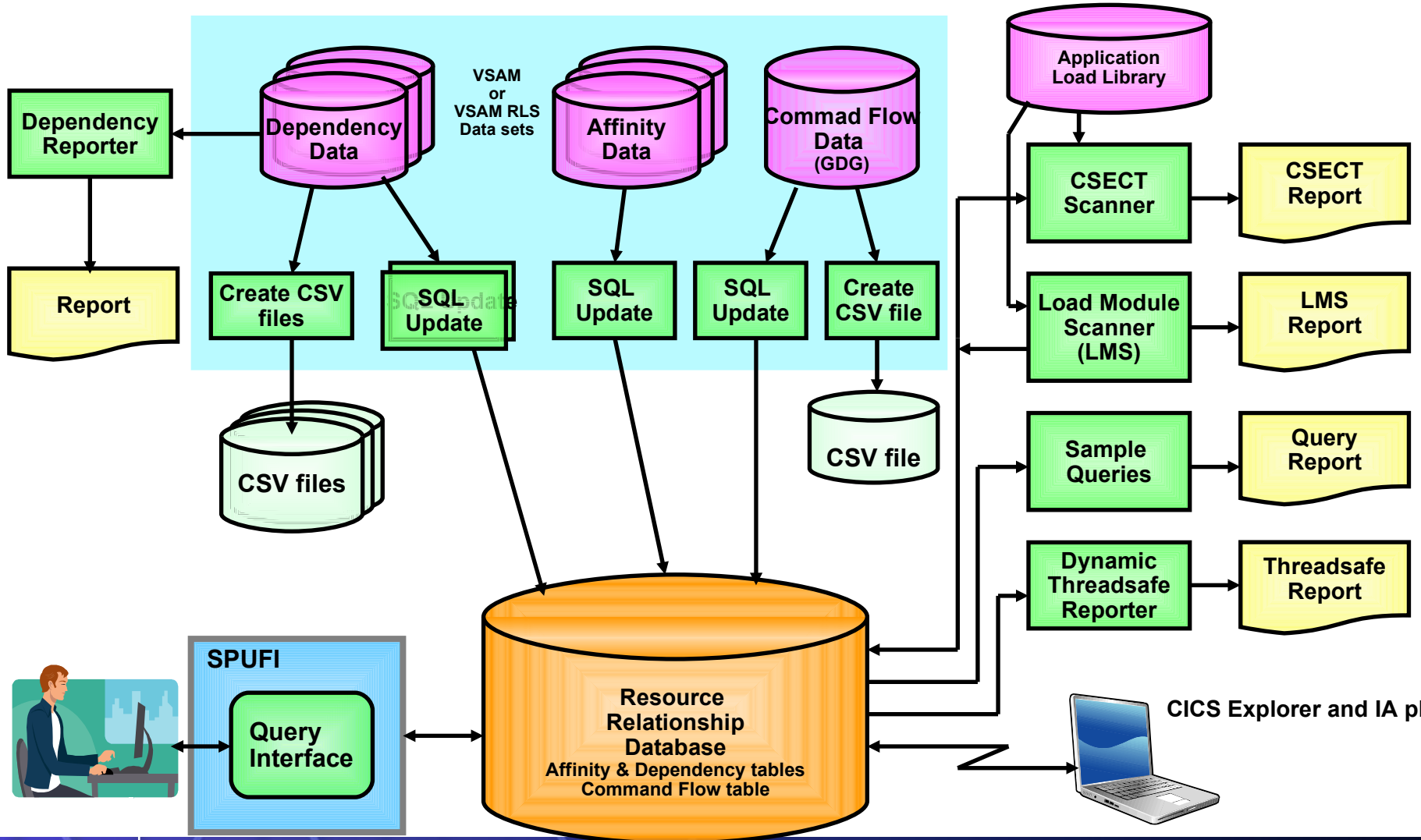
- You can see
  - what resources a CICS region uses
  - what resources a transaction needs in order to run
  - which programs use which resources
  - which resources are no longer used

- Ability to maintain, enhance, modify or redistribute your applications is much improved

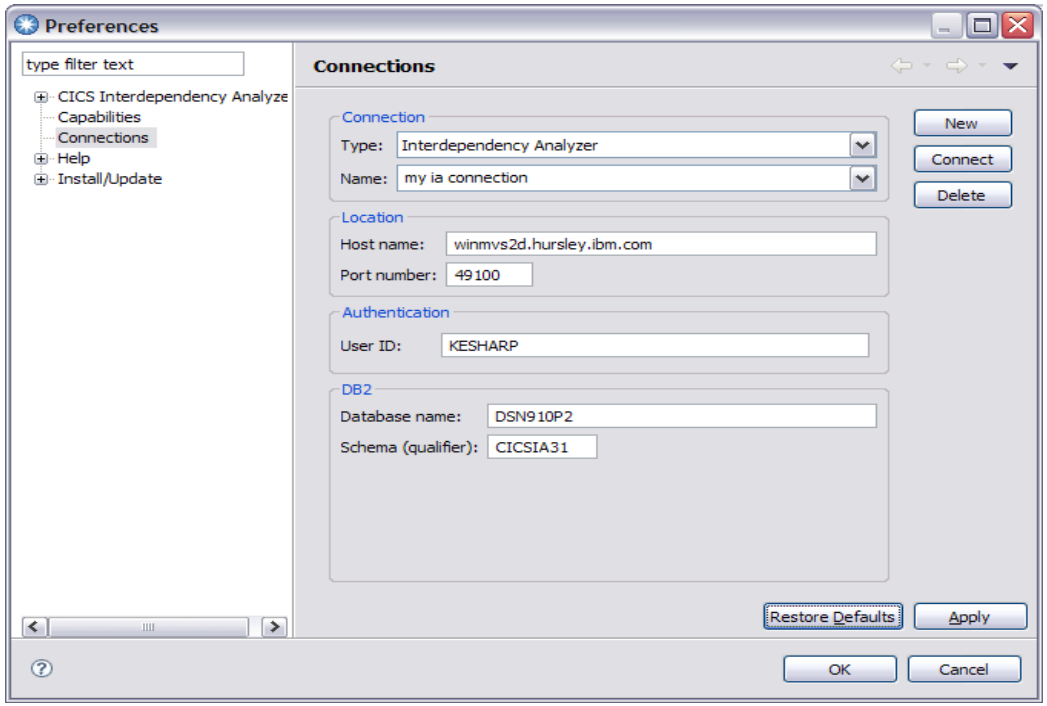
CICS IA Architecture (Collector structure)



CICS IA Architecture (Reporting Structure)



Defining a Connection

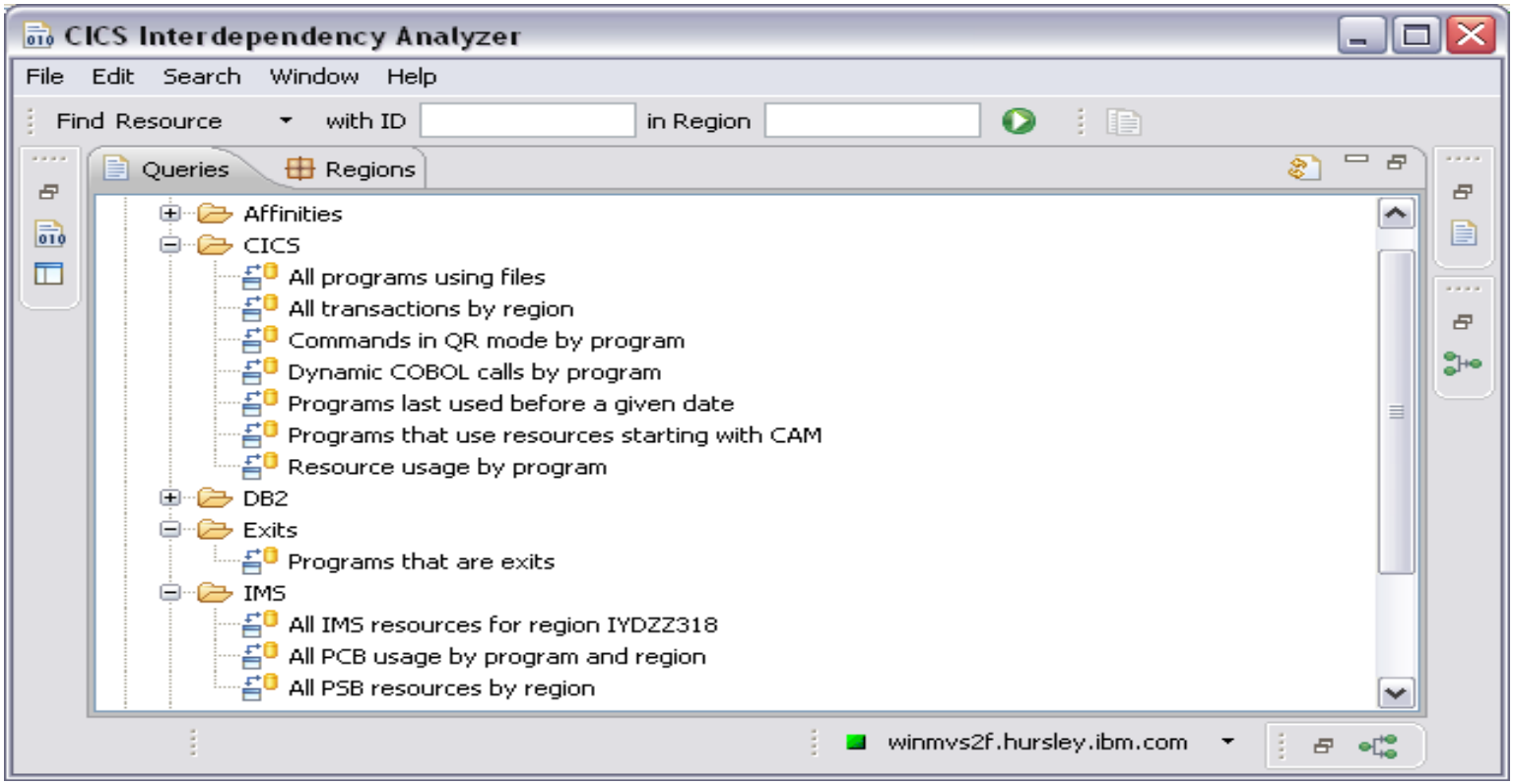


Use the Connections Preference page to define a connection of type “Interdependency Analyzer”

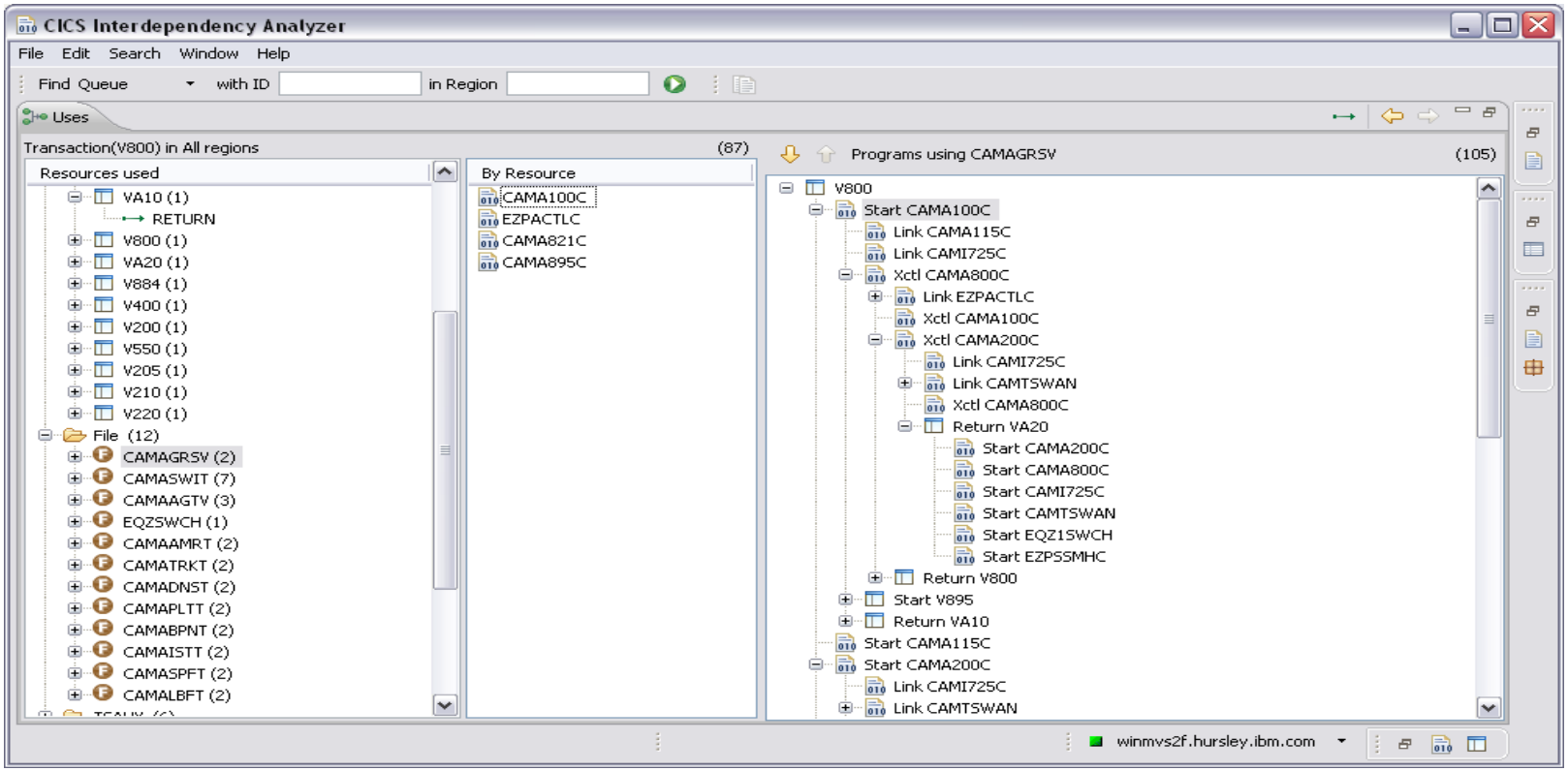
Click New to define a new connection, Apply to save.

Password prompted for on connect

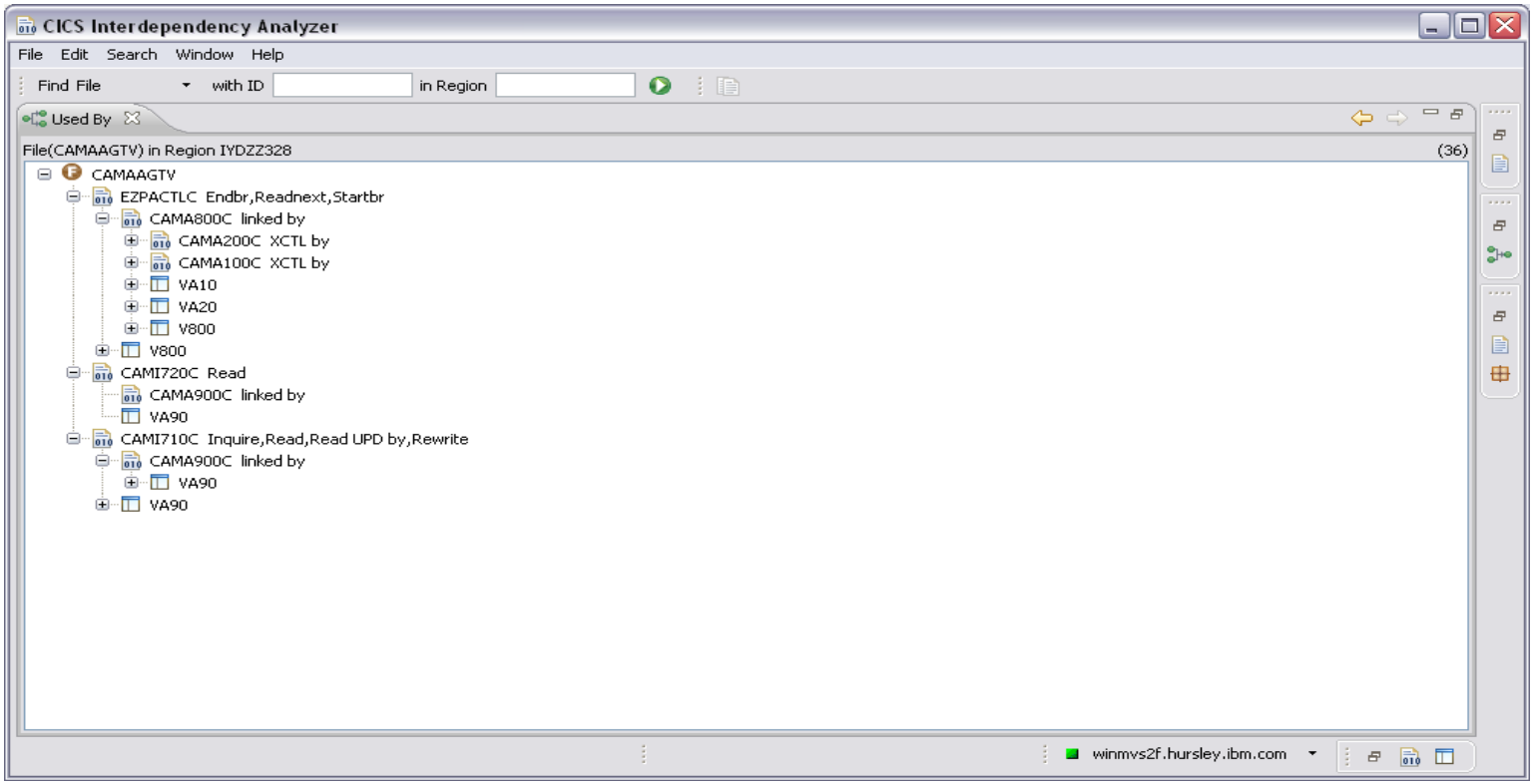
IA Explorer – Queries folder



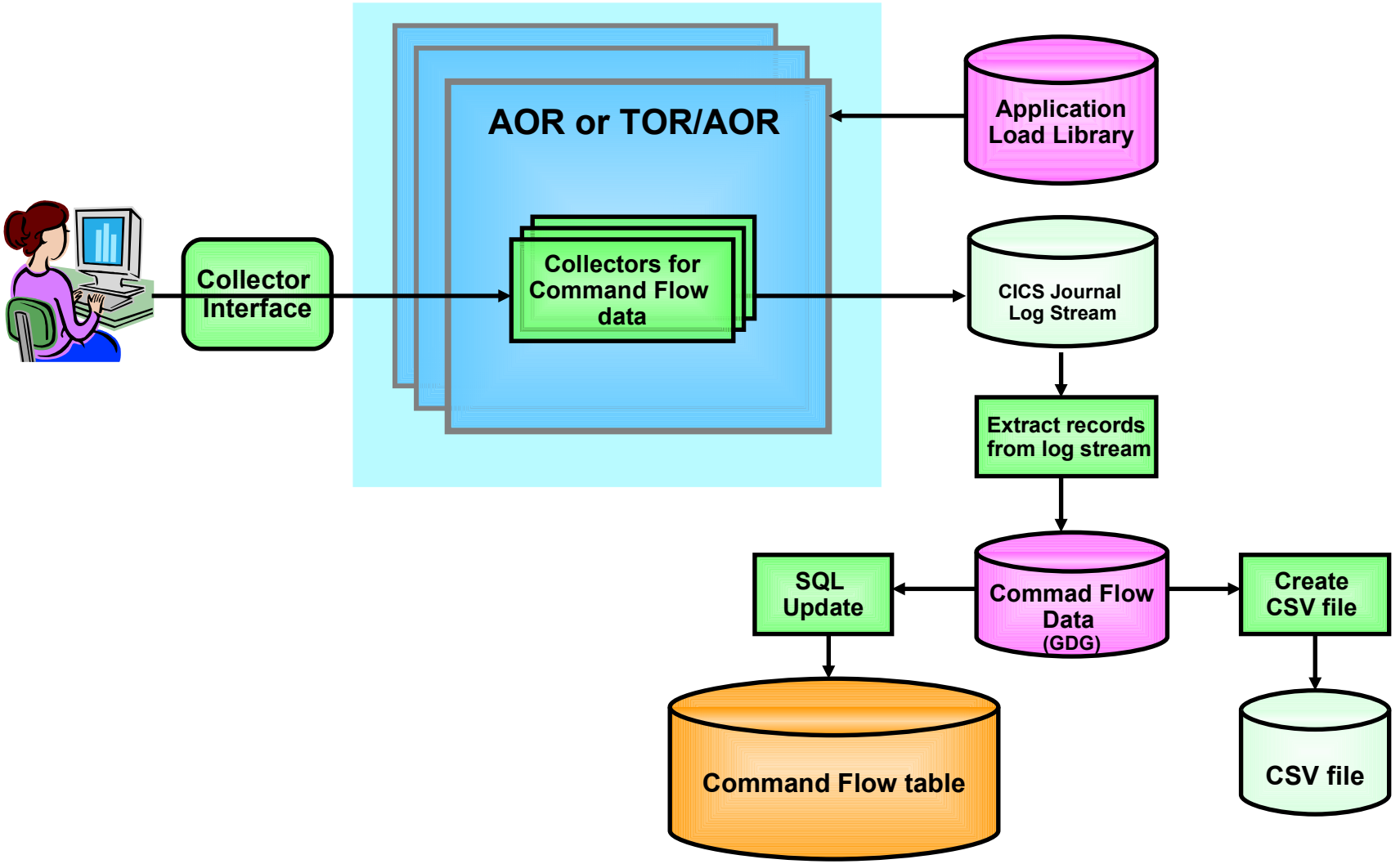
IA Explorer – Uses folder



IA Explorer – Used by folder

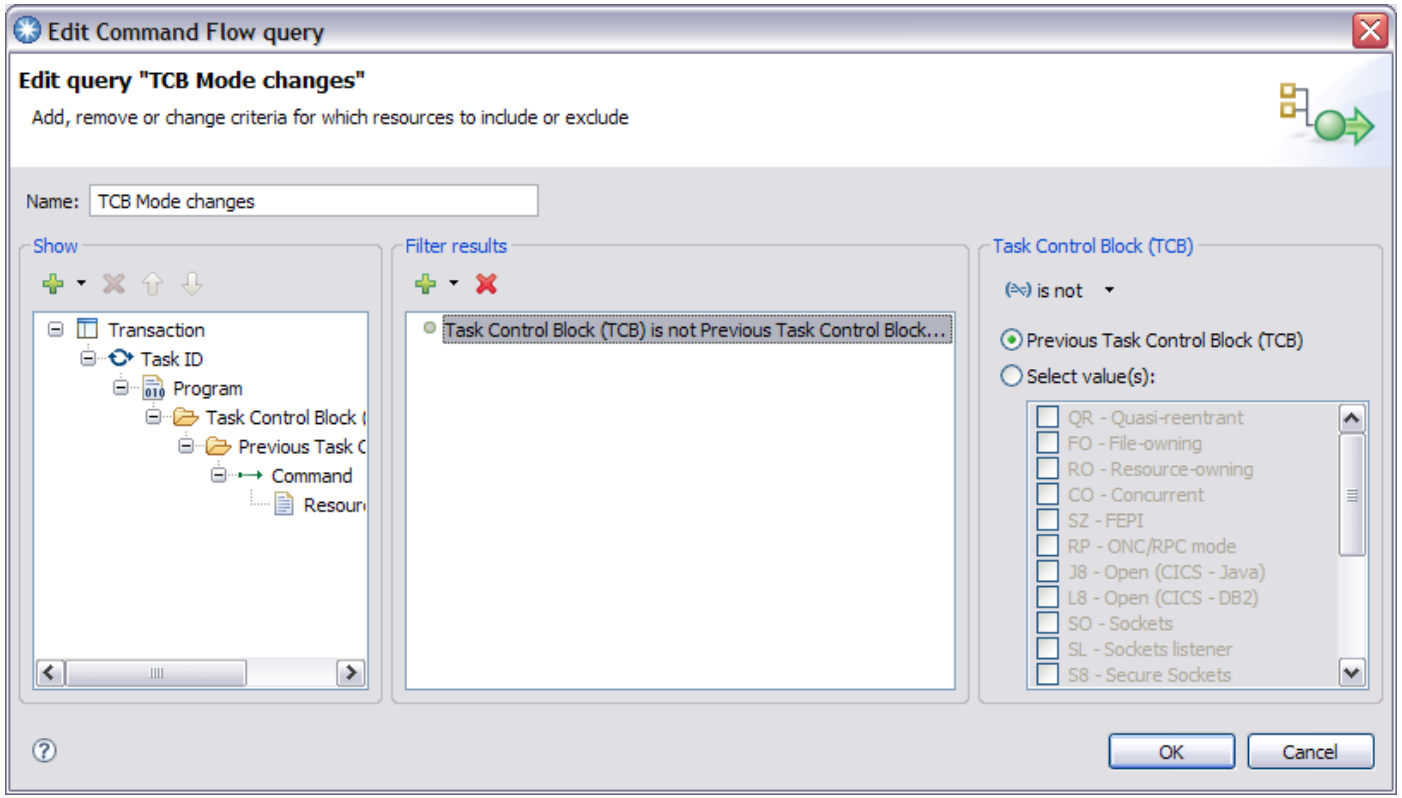


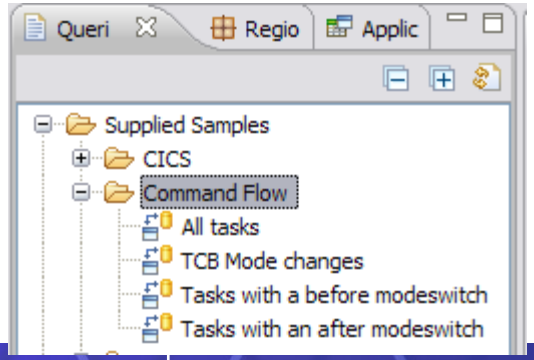
Command Flow option structure



Command Flow queries

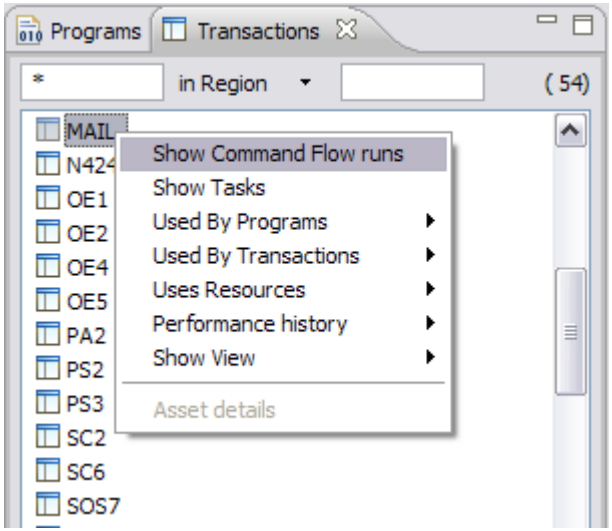
Use command flow query editor to query command flow table directly



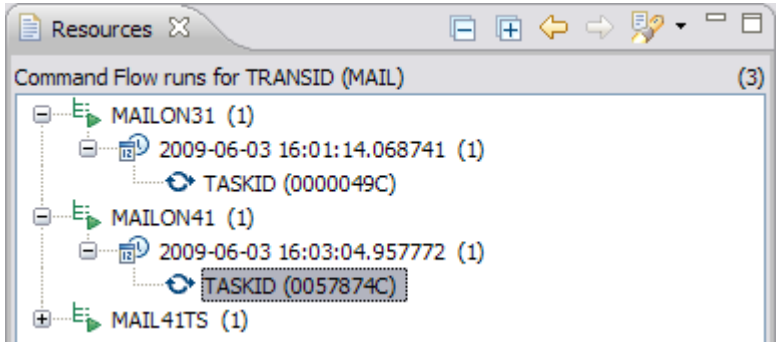


Sample queries provided

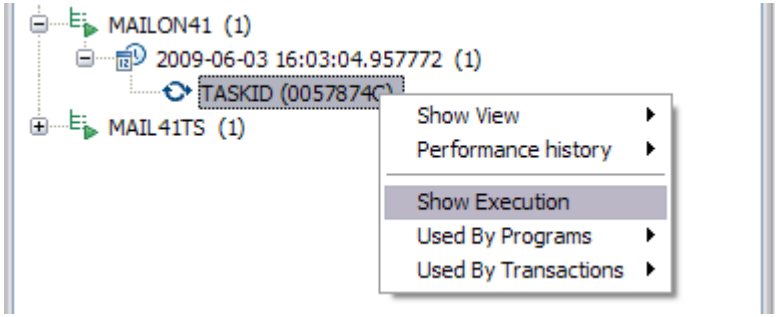
Command Flow



Show collected Command Flow runs for a Transaction

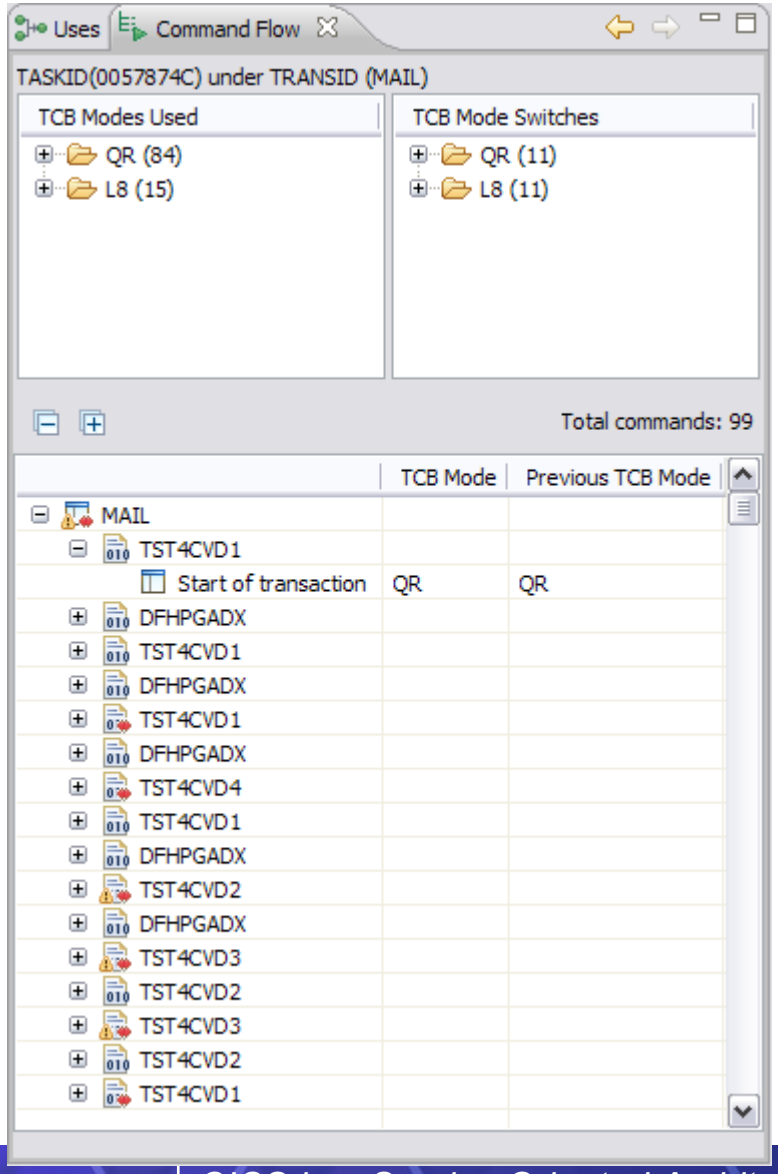


Select which task you are interested in

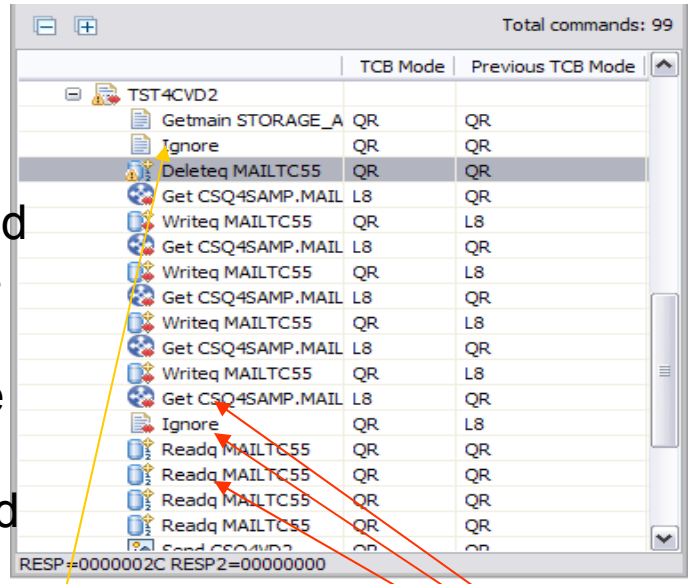


Show the execution of that task

Command Flow Continued…



Displays commands in time order alongside summary of TCB Modes used and any TCB mode switches



Command with non-zero response code decorated with warning. Codes shown on selection

TCB Mode switches decorated with red switch arrow

Revitalizing your CICS Application data using CICS VT

SOA on your terms and our expertise – www.ibm.com/cics

DB2 Benefits to Business and Application

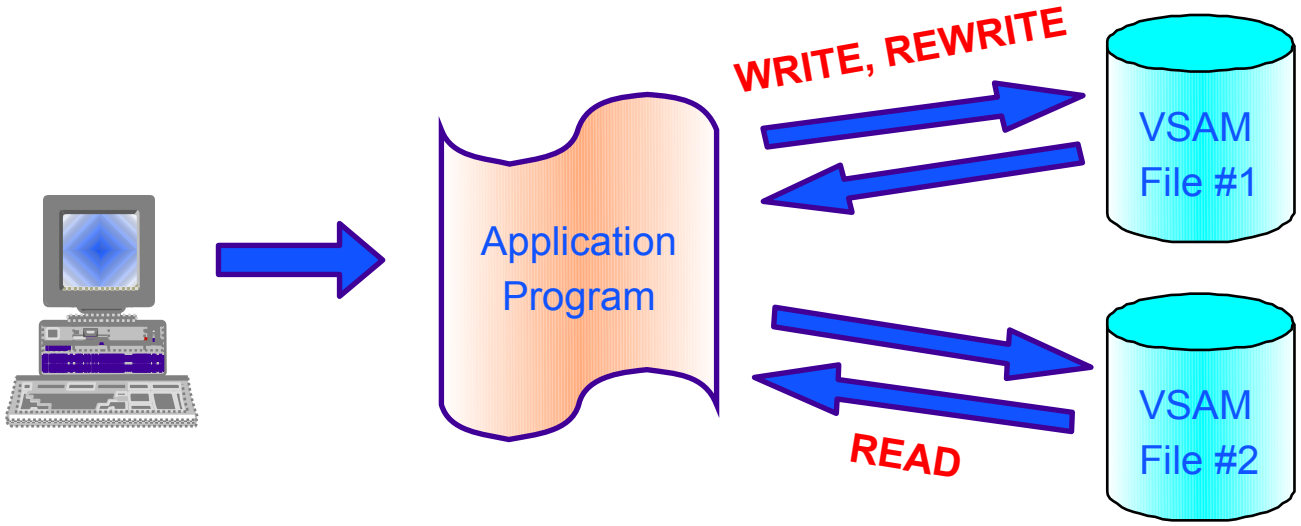
- Faster business need fulfillment
  - Increase application productivity and functionality by leveraging a database platform that supports multiple 3rd party and rapid application development tools
- Mitigate risk of losing market share
  - Legacy database limits application development choices, programmer productivity and ability to meet the changing market demands
- Reduce risk of application failures
  - Avoid maintaining complex legacy applications and file structures that cannot be easily modified
- Data Integrity
  - The existing VSAM files do very little to ensure the data is accurate or consistent.
- Separation of Data from Business Rules
  - Today, master file data cannot be interpreted without applying business logic that is embedded in the programs
  - Data stored in a database needs to be viewed as having its own existence outside of the application processes that maintain it – ADHOC reporting

Re-vitalizing application data using CICS VT

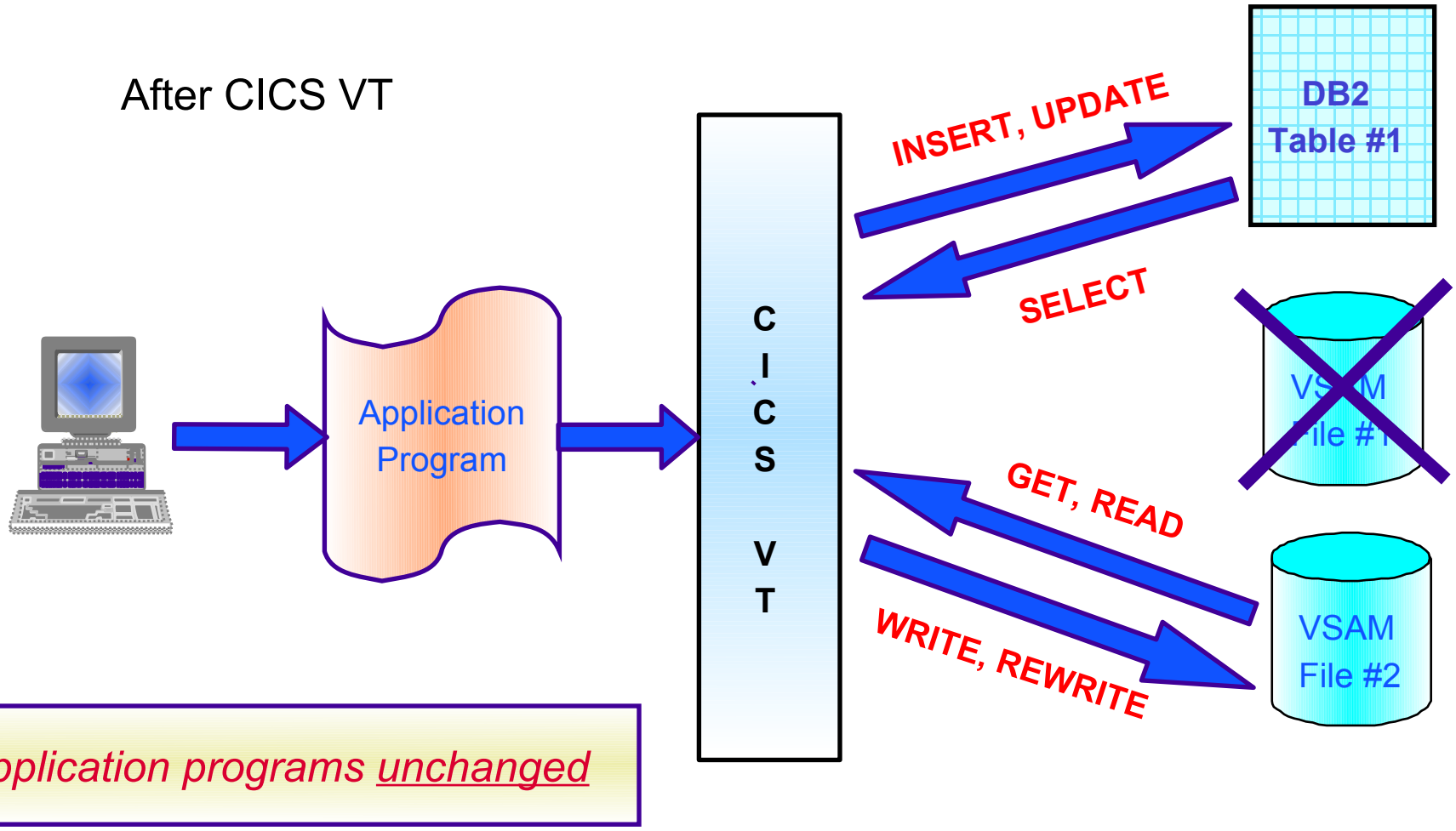
- Migrate VSAM files to DB2 without changing application programs
- Legacy programs access DB2 data using driver modules CICS VT generates for each migrated data set
- Migrated data can be accessed by SQL in new programs
- Existing programs can be enhanced using SQL
- Lowest risk migration strategy

Re-vitalizing application data using CICS VT

- Before CICS VT



Re-vitalizing application data using CICS VT



CICS VT HIGHLIGHTS

- Completely separate from application program
- Transparent access to data in DB2
- 100% static SQL
- Migrate on a file by file basis
- Single live copy of data
  - Dual mode available for test purposes
- Data can be re-engineered
  - Same data returned to VSAM programs
  - Enhanced data available using SQL

CICS Development Technical Services

Helping to solve Business Problems and exploiting new Business Opportunities

Engage the Hursley development team to ensure that you get the maximum value from your CICS investments.

Consultants are now available via a funded services engagement directly from CICS development providing a complete range of CICS services - no one has more experience!

- **System Health Check**
  - How can you be sure that your CICS environment in running optimally? Be sure that your CICS environment is in the best of health and delivers maximum value to you
- **Upgrade Services**
  - Do you want to improve your competitive edge by offering new CICS services? Exploit new CICS features such as: Web services, IP connectivity, CICS Explorer, Event Processing, Web 2.0, Atom Feeds and PHP scripting in CICS

CICS Development Technical Services

Helping to solve Business Problems and exploiting new Business Opportunities

- **Security and Governance**
  - Worried about your CICS security? Standards, regulatory compliance and policies getting you down? This services helps you assess and ensure the security and compliance of your CICS assets.
- **Performance Optimization**
  - Is your goal to maximize your performance and minimize your costs? We review CICS performance data from SMF records and/or CICS tooling to provide tuning recommendations and Capacity Planning advice
- **Web Services Enablement**
  - Need to unlock and modernize your CICS assets for SOA exploitation? We provide education and implementation for CICS Web services support including Rational Developer for z tooling. We can help with identifying assets for re-use in an application and legacy modernization project
- **Integration and Connectivity**
  - Interested in lowering complexity? Need to connect CICS with WebSphere using the latest technologies? We provide consultancy for migration from SNA to IP connectivity as well as using connection technologies such as the CICS Transaction Gateway and Web services to provide the availability of your CICS business assets to a wider audience in an SOA

CICS Development Technical Services

Helping to solve Business Problems and exploiting new Business Opportunities

- **Systems Management**
  - Concerned about the availability of your CICS environment? Our Systems Management offering is aimed at ensuring that your CICS environment is being managed to Best Practice standards. Improve your service levels and lower your costs

- **Event Processing**
  - Respond more quickly to business opportunities without changing your existing CICS application code. CICS Event Processing also helps you with regulatory compliance and Service Level Agreement monitoring through Business Dashboards and Key Performance Indicators. This offering helps you integrate your CICS environment into an Event Processing framework.

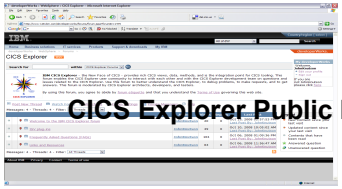
- **Customized Workshops**
  - Skills issue? Need the best CICS product and solution training? Ensure the effective transfer of CICS best practices to the next generation of IT Staff. We offer Customized Workshops to meet your education needs for the CICS Family of products. These workshops are staffed by senior CICS developers and can be run either at your location, or at an IBM location that meets with your preference.

- For further information please contact: CICSDTS@uk.ibm.com

CICS Communities and Information

zSeries PD/CICS/Icing Sales - CICS Commu

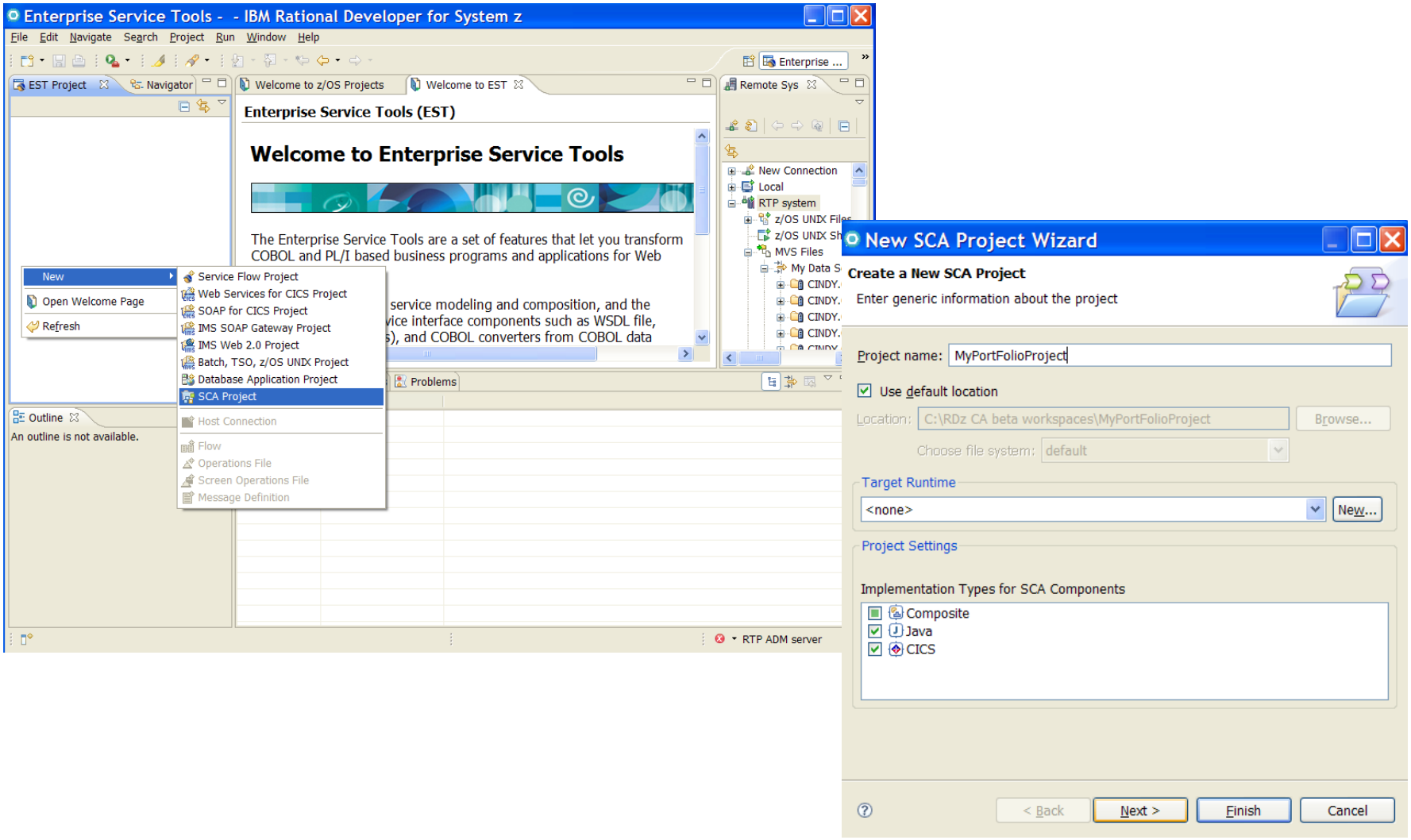
- CICS Transaction Server V4.1
  - http://ibm.com/cics/tserver/v41/
- CICS Explorer home page
  - Remember this link ibm.com/cics/explorer
- CICS Explorer Forum
  - http://tinyurl.com/68bndw
  - IBM developerWorks forum with FAQs, Links and resources, ISV Contributions, etc. Ask questions, suggest improvements, report problems, chat
- New! CICS Hub on the Rational COBOL Café
  - http://ibm.com/software/rational/cafe/community/cobol/cics
- Twitter - Subscribe to the IBM_System_z channel & CICSfluff channel to get CICS news flashes
- CICS Blog - Comment and opinion at TheMasterTerminal.com
- CICS eNews - Subscribe for news about CICS and related products
- CICS Links regular updates all in a single presentation deck
- YouTube channels
  - CICS Explorer - Videos, demos and other cool stuff
  - CICSFluff - Other CICS videos



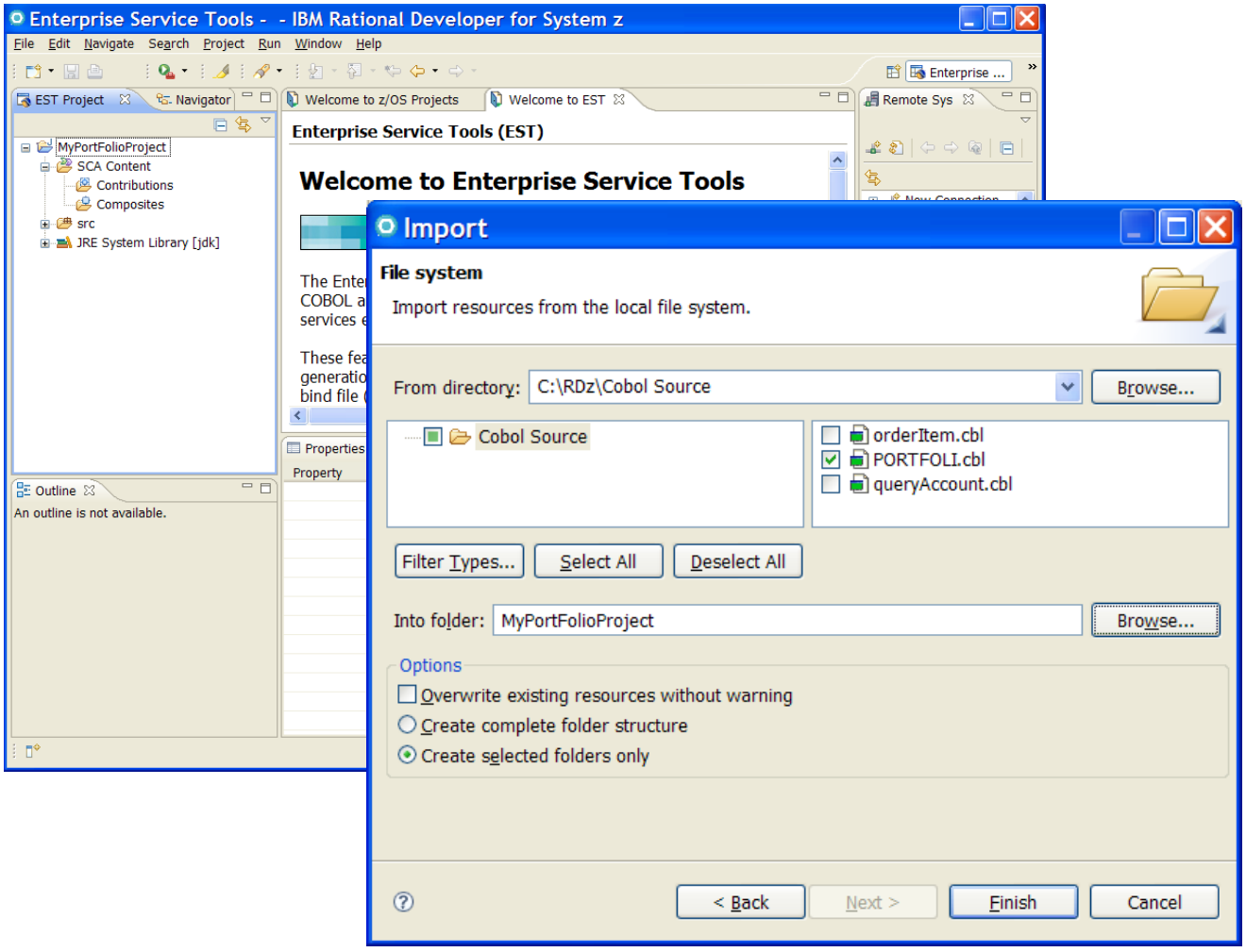
Supplementary Material

SOA on your terms and our expertise – www.ibm.com/cics

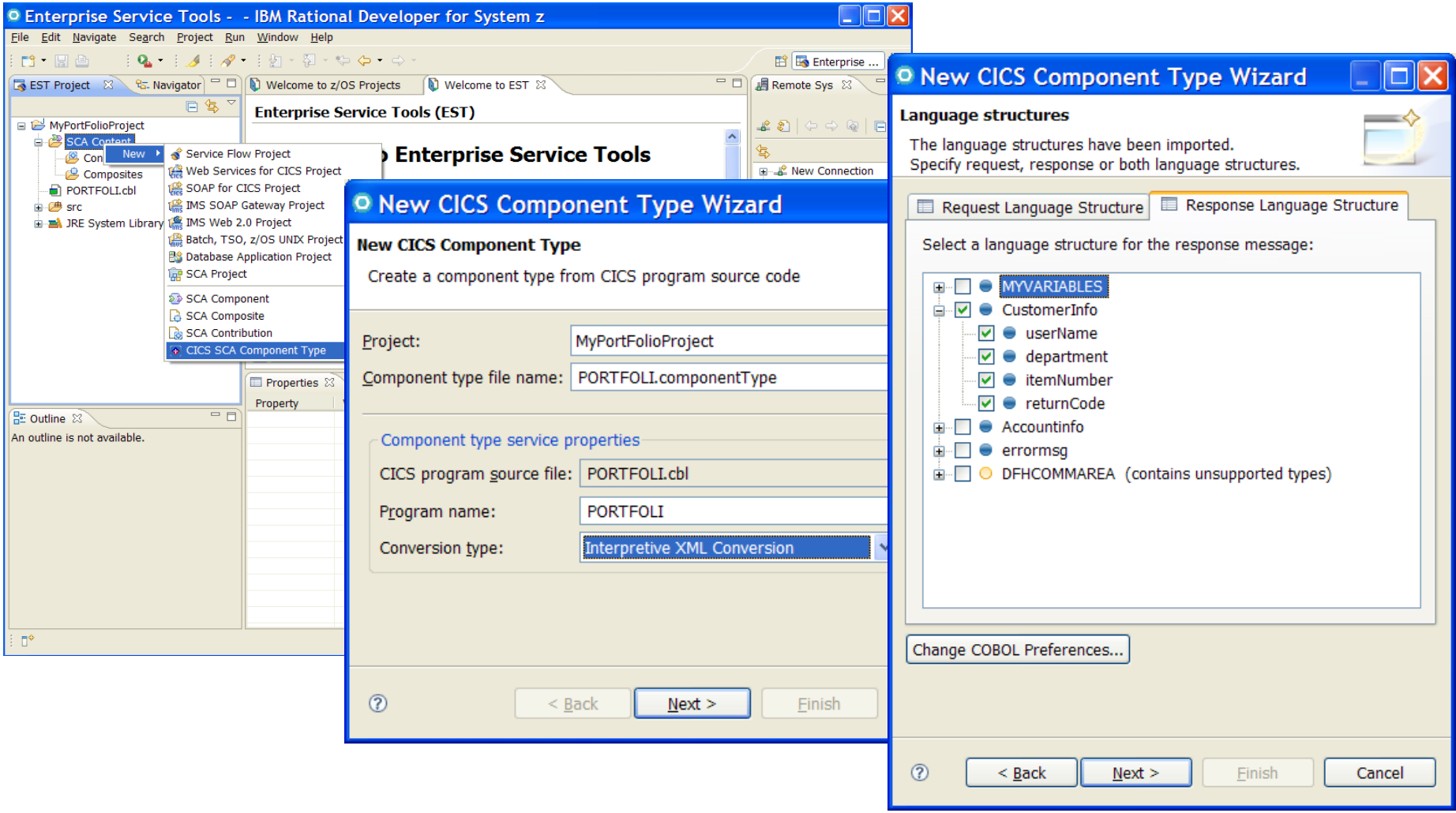
RDz SCA Tooling Support



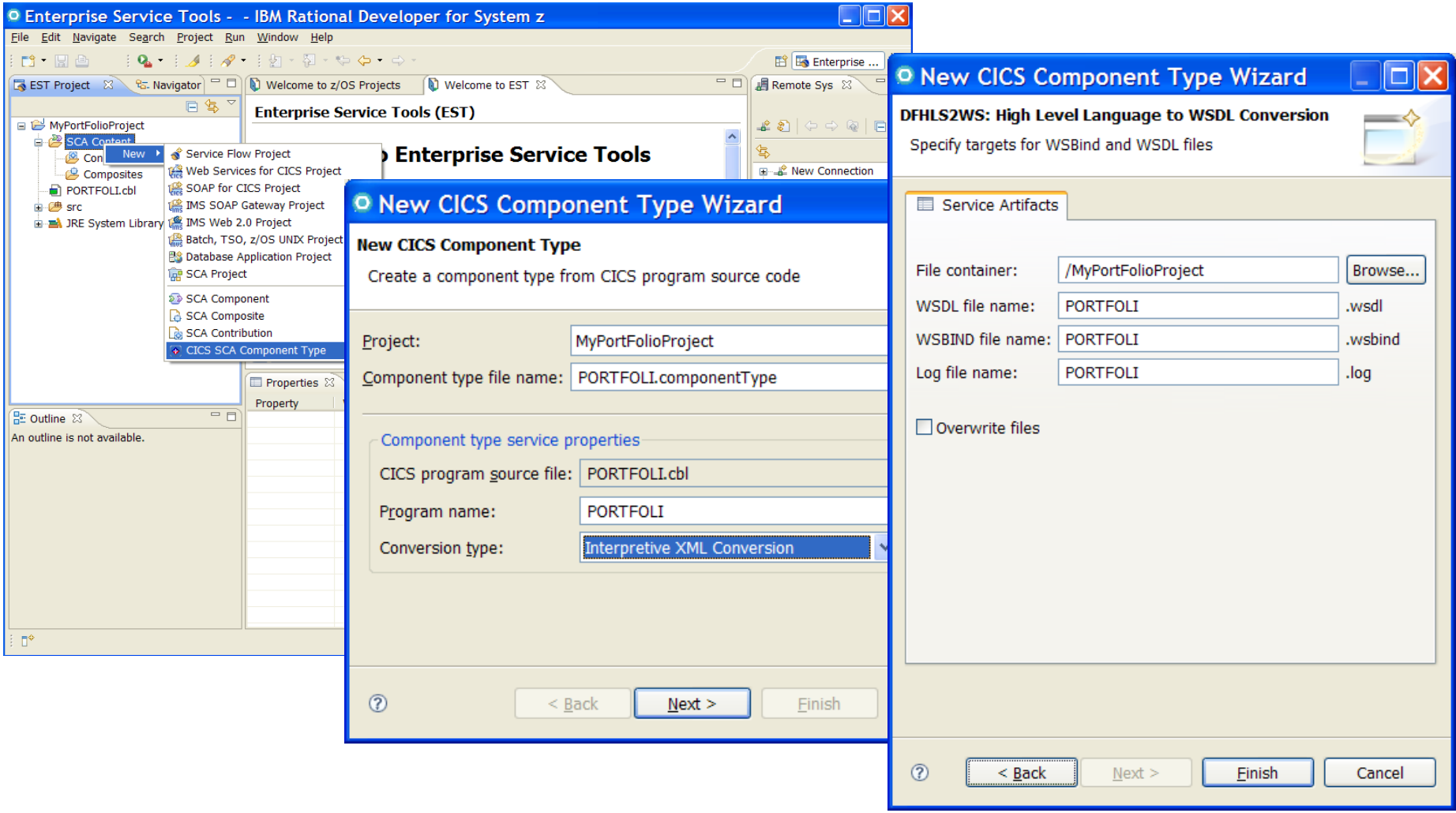
RDz SCA Tooling Support



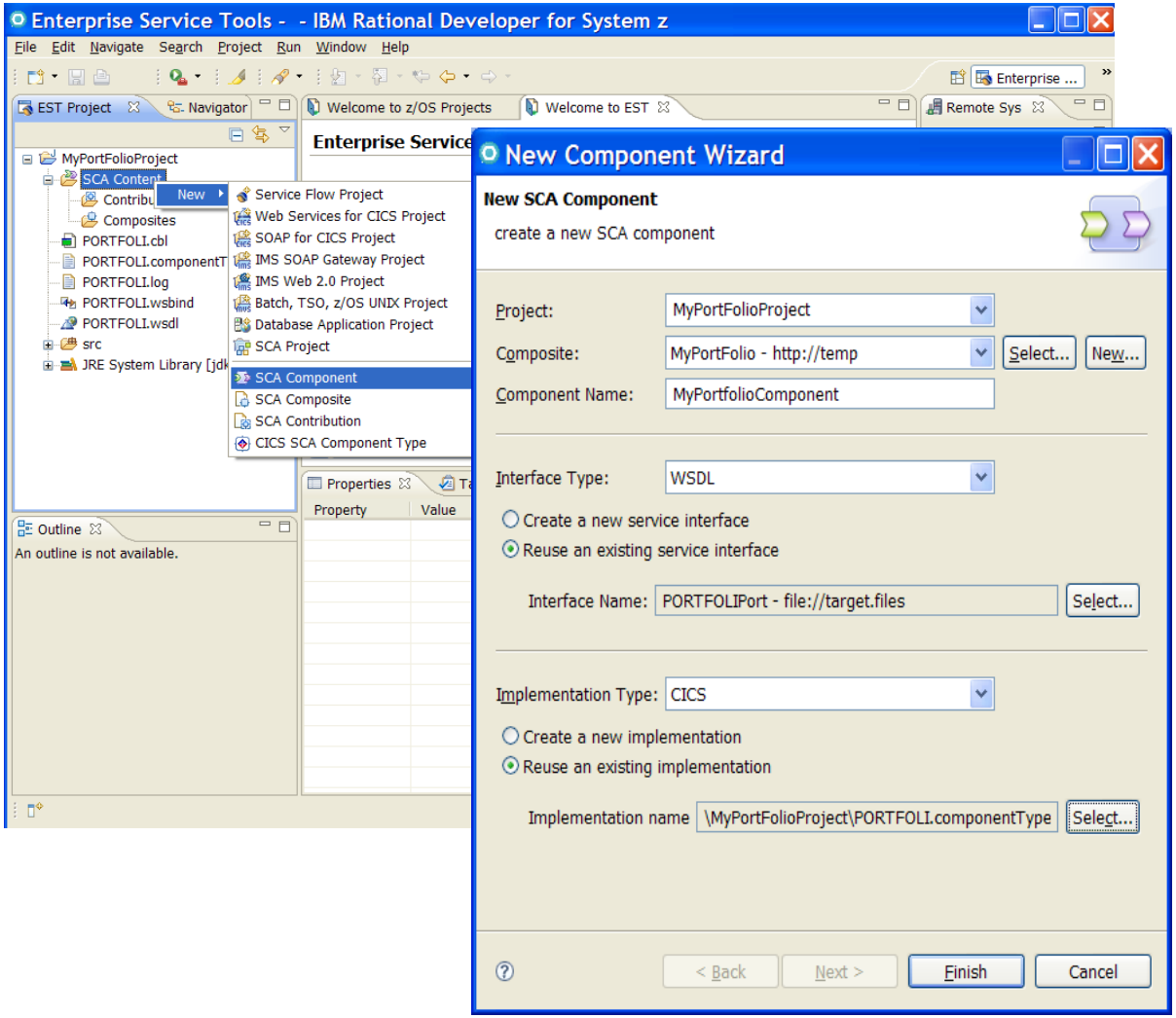
RDz SCA Tooling Support



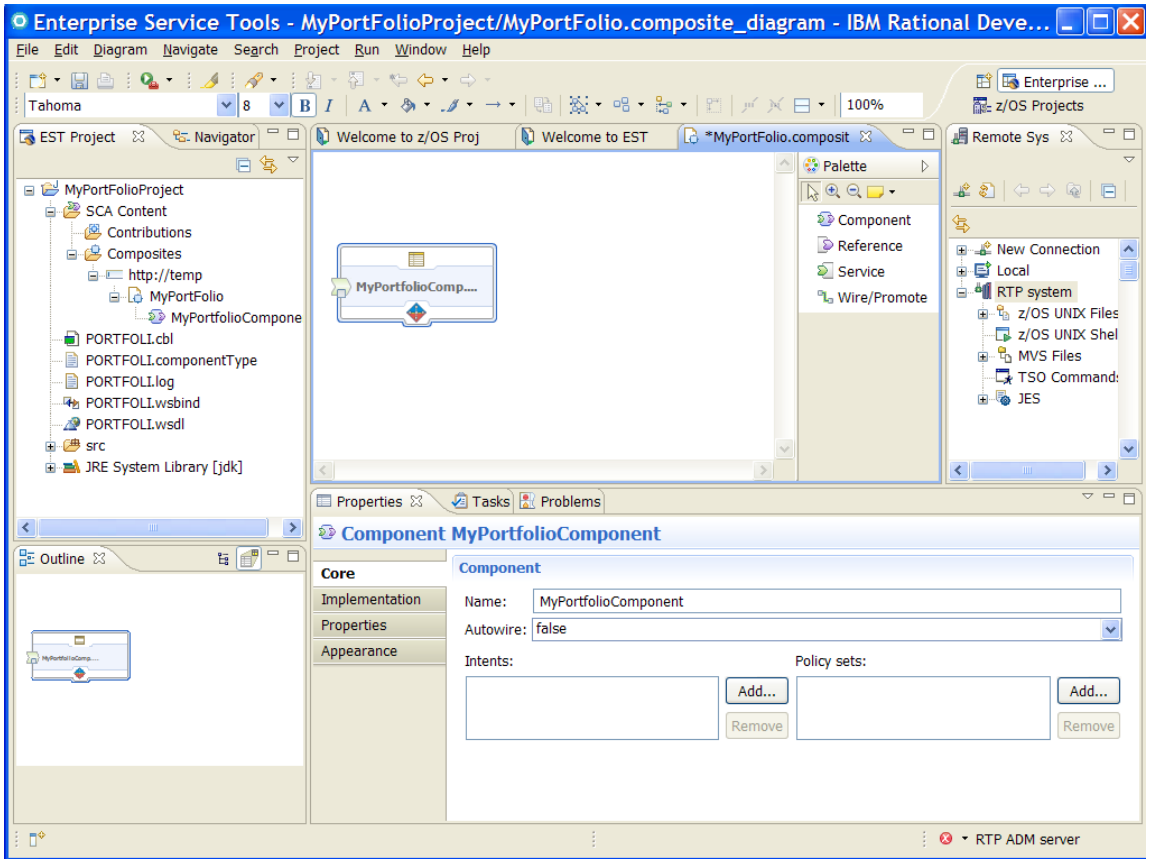
RDz SCA Tooling Support



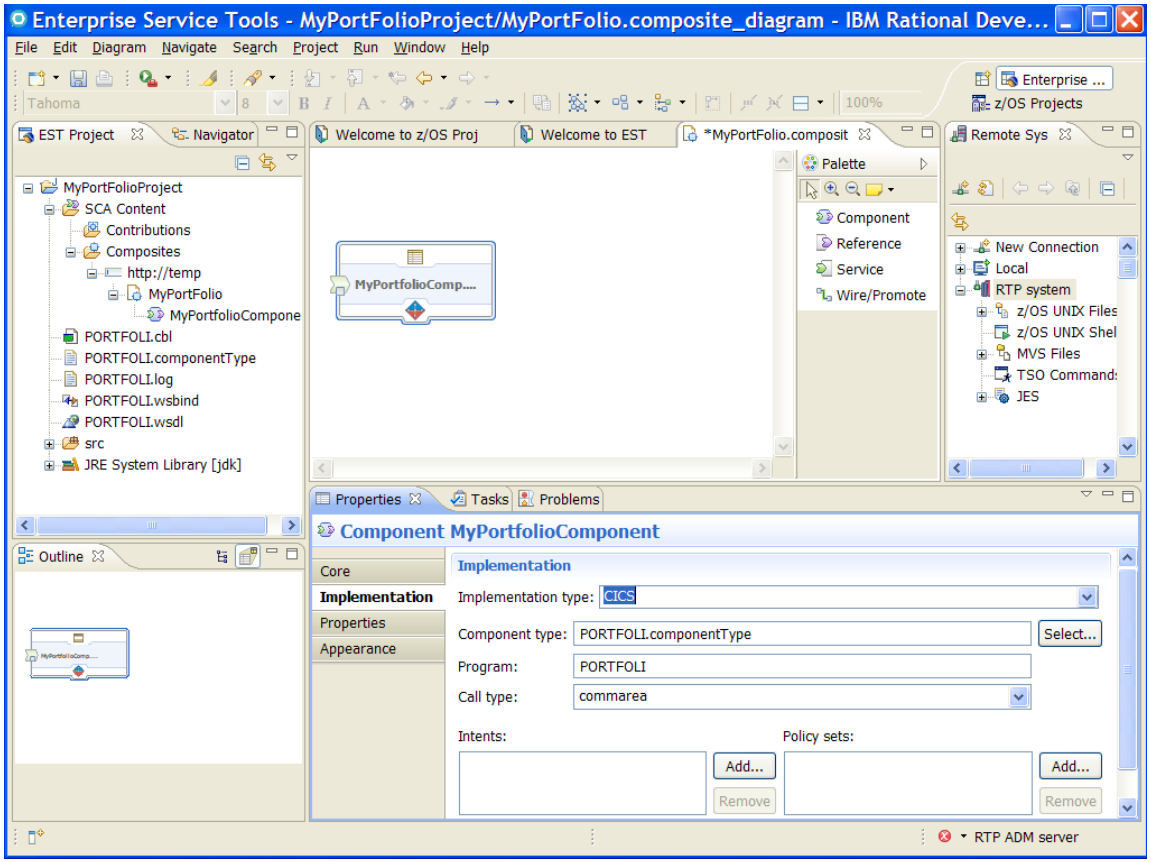
RDz SCA Tooling Support



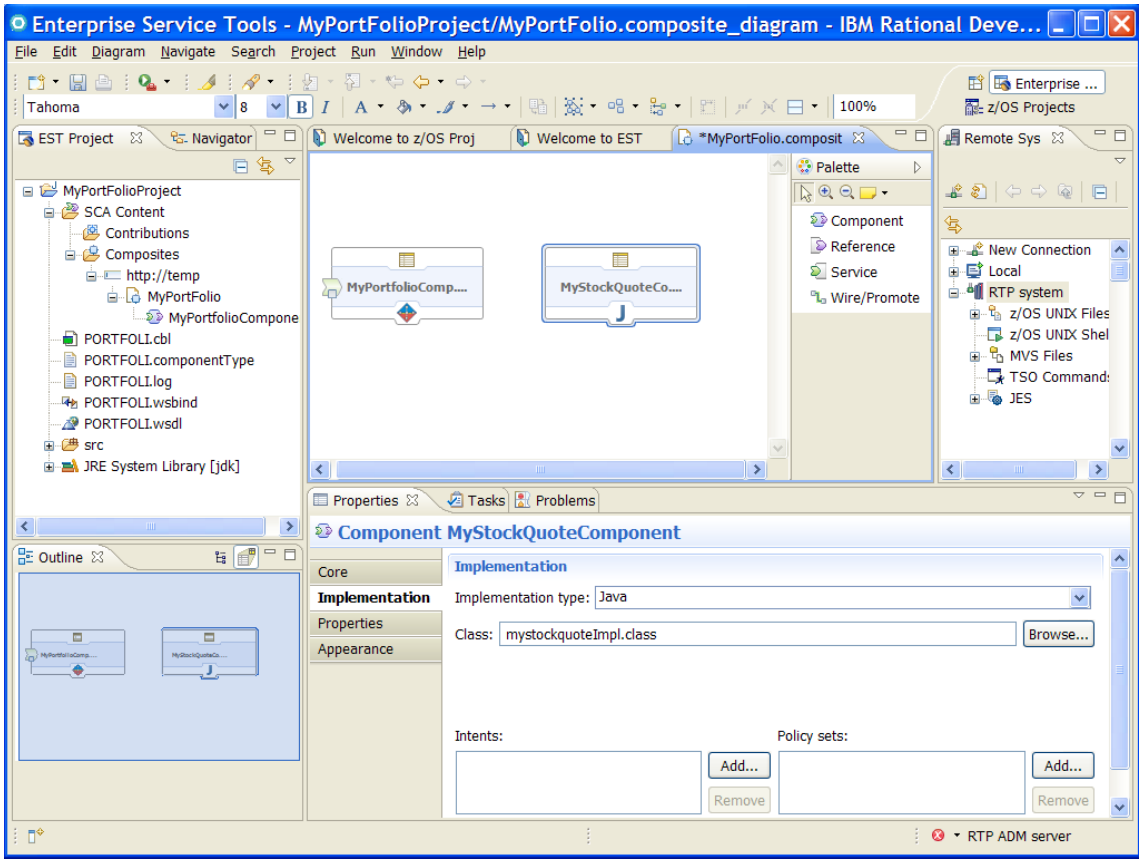
RDz SCA Tooling Support



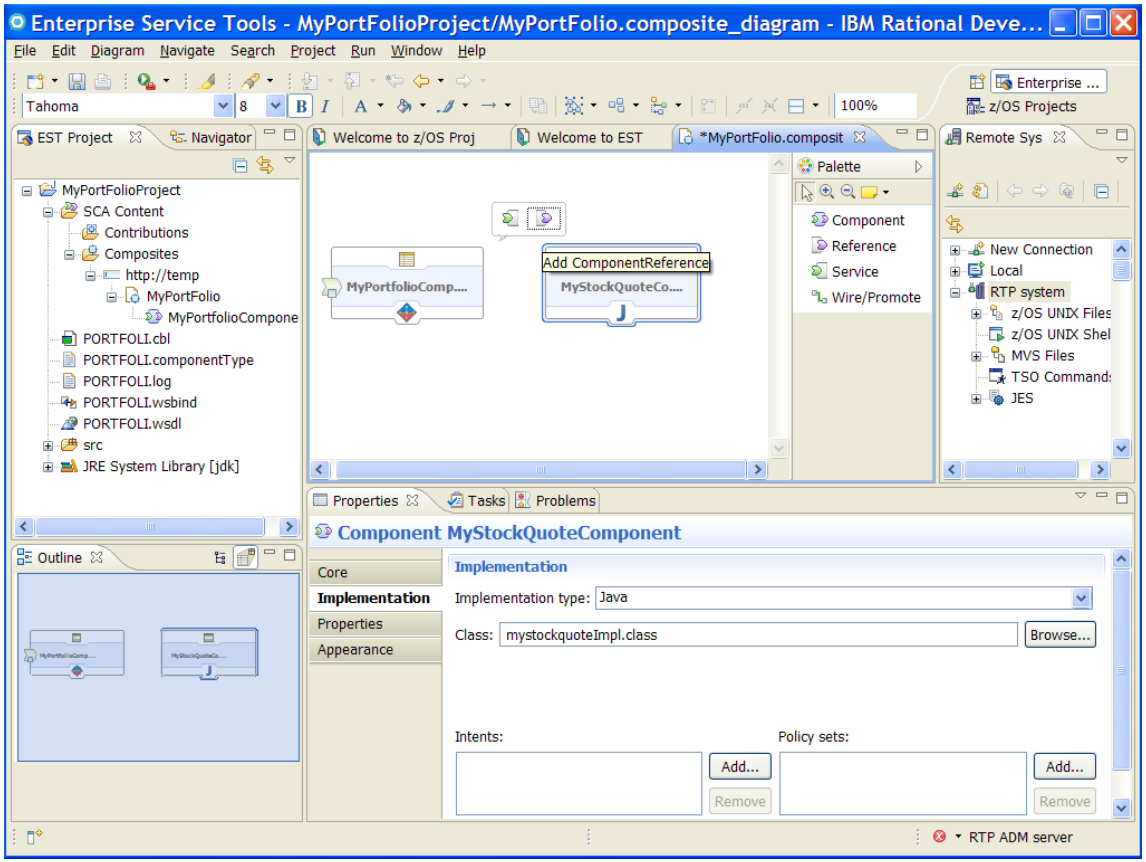
RDz SCA Tooling Support



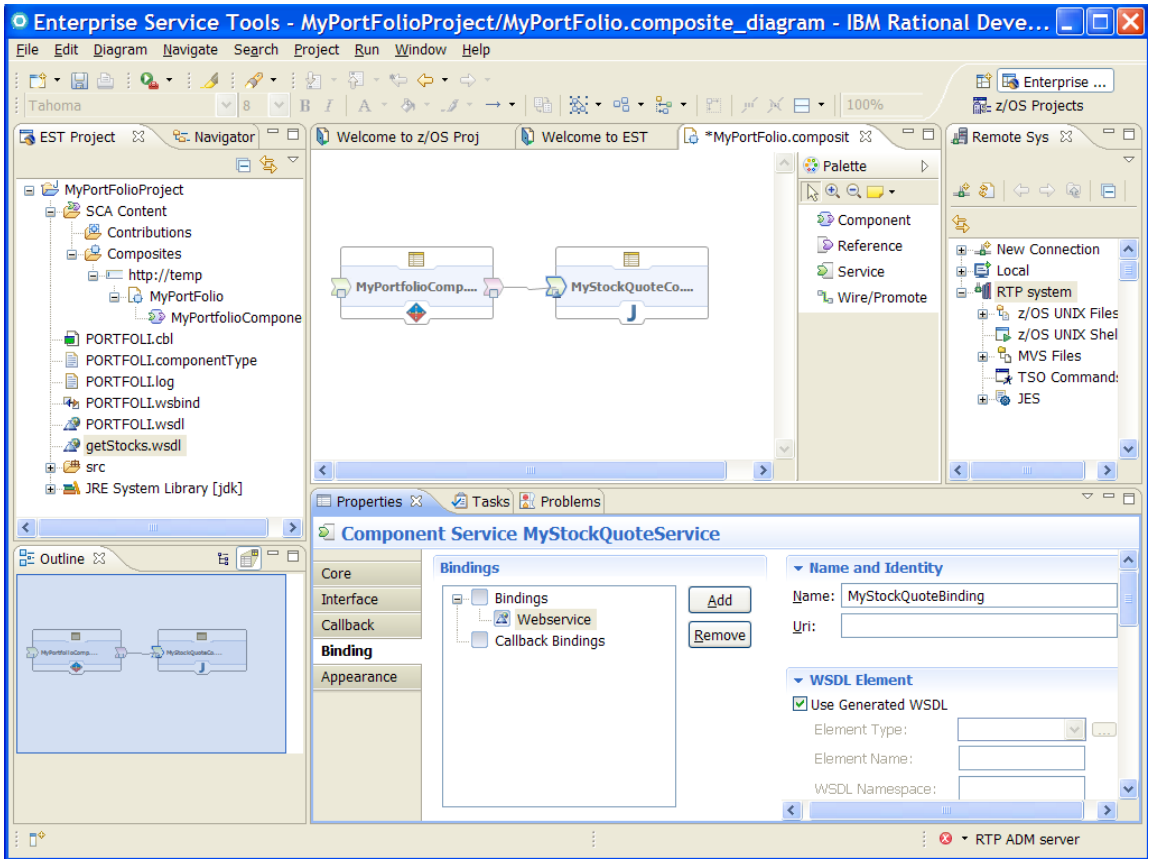
RDz SCA Tooling Support



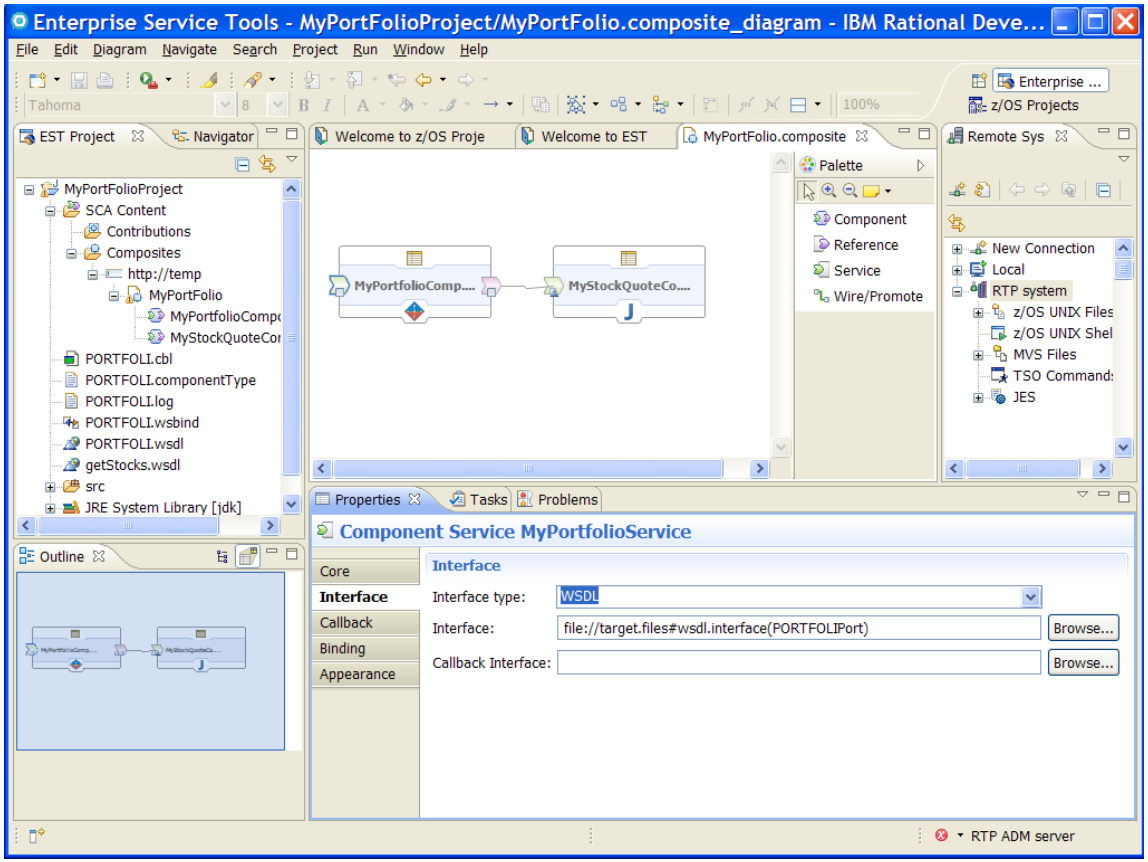
RDz SCA Tooling Support



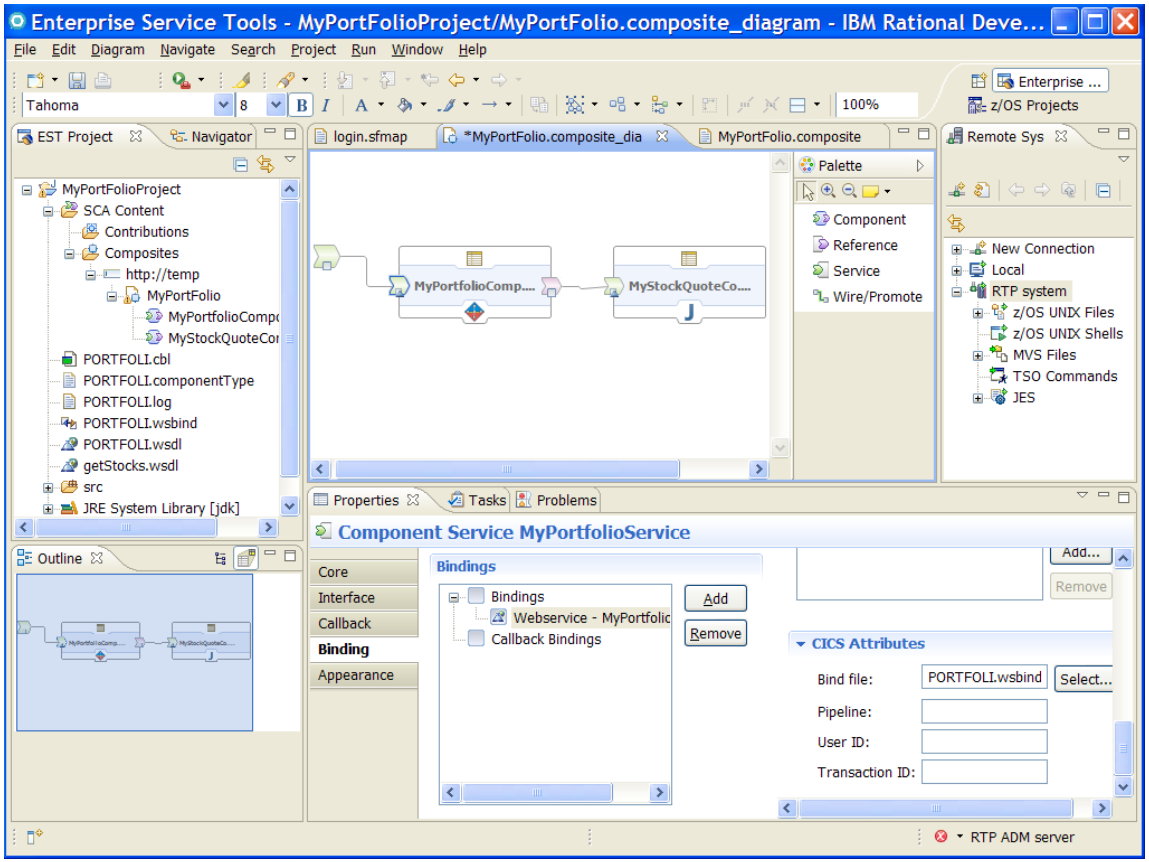
RDz SCA Tooling Support



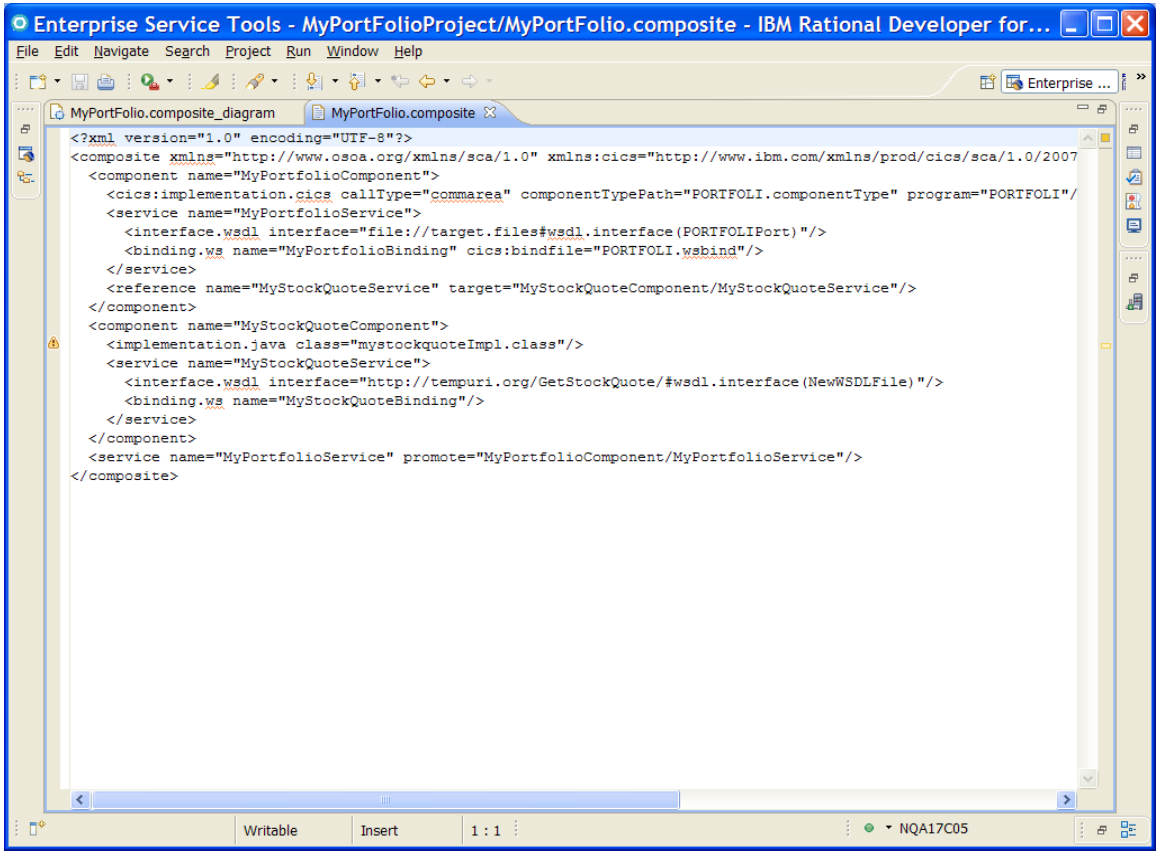
RDz SCA Tooling Support



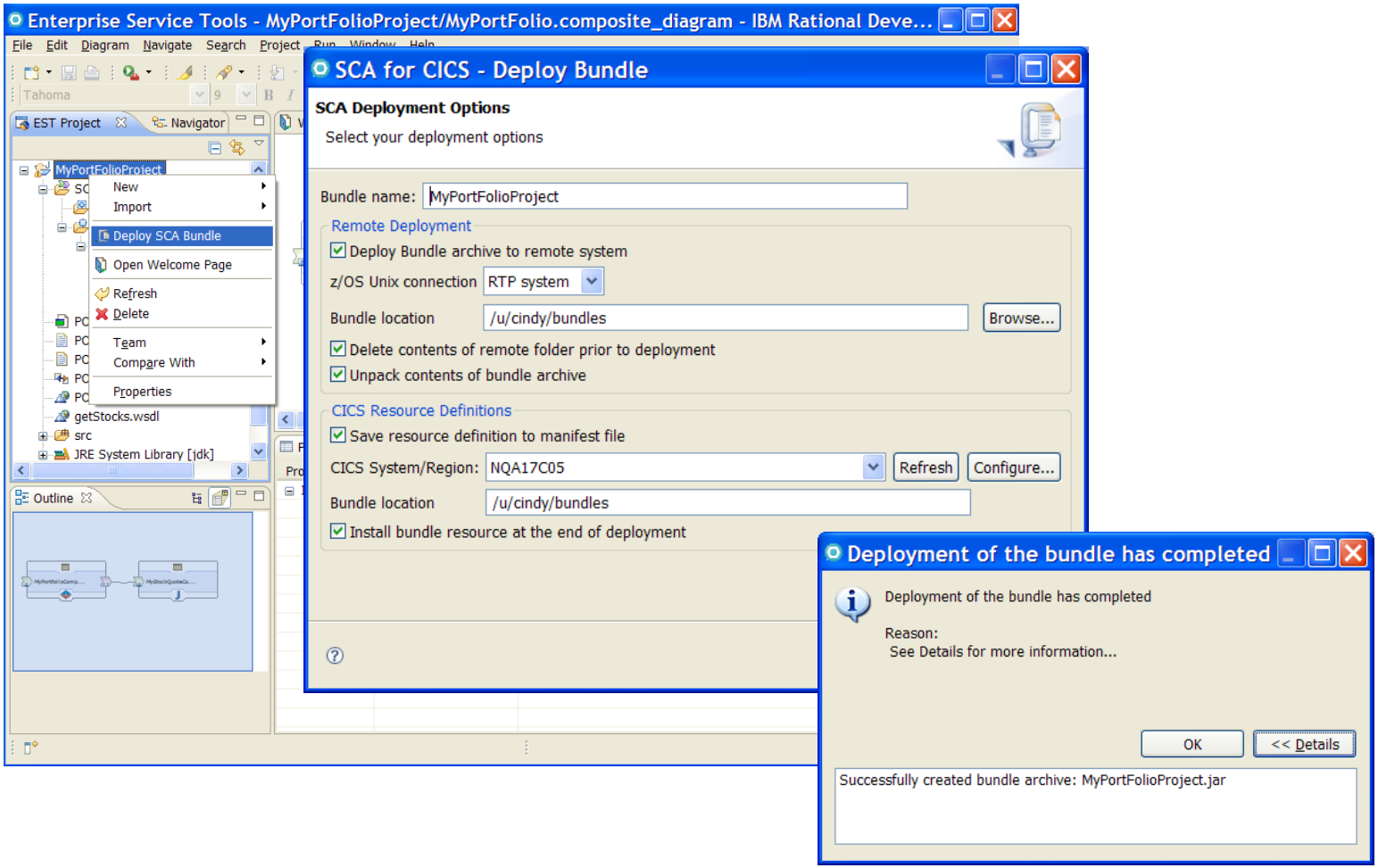
RDz SCA Tooling Support



RDz SCA Tooling Support



RDz SCA Tooling Support



RDz SCA Tooling Support