E34

SOA Tools for z/VSE: Generating Web Services in CICS using CICS2WS

Ingo Franzki, IBM

IBM
SYSTEM z9 AND zSERIES EXPO
October 9 - 13, 2006

Orlando, FL

© IBM Corporation 2006

IBM System z9 and zSeries EXPO

E34 - SOA Tools for z/VSE:
Generating Web Services in CICS using CICS2WS

Ingo Franzki – ifranzki@de.ibm.com

August 24, 2006

© 2006 IBM Corporation

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and / or other counties.

- CICS*
- DB2*
- DB2 Connect
- DB2 Universal Database
- e-business logo*
- Enterprise Storage Server
- HiperSockets
- IBM*
- IBM logo*
- IMS
- Intelligent Miner
- Multiprise*
- MQSeries*
- OS/390*
- S/390*
- SNAP/SHOT*
- Virtual Image Facility
- VM/ESA*
- VSE/ESA
- VisualAge*
- VTAM*
- WebSphere*
- xSeries
- z/Architecture
- z/VM
- z/VSE
- zSeries

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

LINUX is a registered trademark of Linus Torvalds

Tivoli is a trademark of Tivoli Systems Inc.

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.

Agenda

What is SOA?

Terms and theory

- SOA, SOAP, WSDL, XML

§ **Using Web Services with Java/.Net**

- Proxy code

§ **Using Web Services with VSE**

- VSE SOAP Engine, programming Interfaces

§ **CICS2WS Tool**

- Live demo

What is Service-Oriented Architecture (SOA)?

- **SOA is an IT architectural style**
  - supports integrating your business as linked services that can be accessed when needed over a network, enabling your business to adapt to changing conditions and requirements
  - These services are self-contained and have well-defined interfaces to let the users of those services -- called clients or consumers -- know how to interact with them
- **SOA results in "loosely coupled" application components**
  - The code is not necessarily tied to a particular database, or even a particular infrastructure.
- **It is this loose coupling that enables the combination of services into diverse applications.**
  - It also enables much greater code reuse, cutting your workload at the same time that it increases your capabilities.
- **Because a service and the client accessing that service are not tied to each other**
  - a service used to process an order could be completely replaced, and the client-services placing orders would never know.

What is Service-Oriented Architecture (SOA)?

- **From a business standpoint, a Service-Oriented Architecture is focused on**
  - developing technology that helps you accomplish your business tasks
  - rather than allowing technological constraints to dictate your activities.
- **For example, the process of selling, manufacturing, shipping, and getting paid for an item may involve dozens of steps and several different databases and computer systems.**
- **But at the heart of things, the process encompasses a handful of human activities, for example:**
  - Salesmen finds a likely customer
  - Customer orders product
  - Production department produces product
  - Production department ships product
  - Billing department bills for product
  - Customer pays for product

What is Service-Oriented Architecture (SOA)?

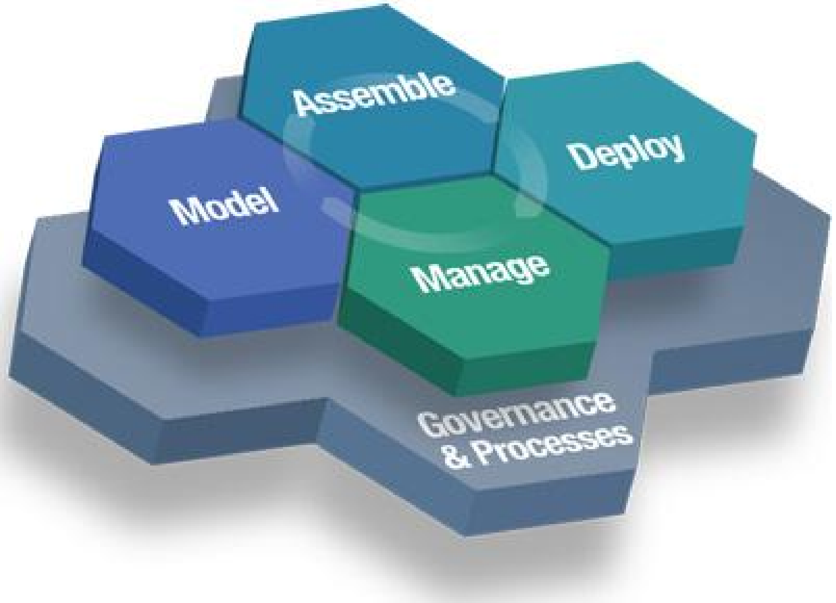
- **Implementing SOA can bring you a great number of benefits, including the following:**
  - Greater alignment of business and IT
  - Component-based systems
  - Loosely coupled components and systems
  - A network-based infrastructure, enabling geographically and technologically diverse resources to work together
  - On-demand, built-on-the-fly-applications
  - Greater code reuse
  - Better process standardization throughout the enterprise
  - Easier centralization of corporate control

What is Service-Oriented Architecture (SOA)?

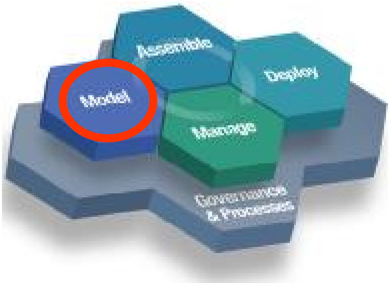
- **Web services are the most common technology standards used to implement SOA**
  - However, they are not the only technology one can use to develop the parts of an SOA
- **Many SOAs -- most, in fact -- involve the integration of legacy data**
  - contained in systems that use technology such as MQSeries and Common Object Request Broker Architecture (CORBA) or even CICS.
- **Many of these technologies have been adapted for the SOA world, and they can be used with or without a Web services wrapper.**
- **But, Web services is rapidly becoming the de facto standard used to support SOA.**

SOA lifecycle

- Because SOA touches so many parts of your business, undertaking SOA projects involves careful planning and design from the outset.
- You need to think of the entire lifecycle of the project, from its inception to its first realization and on to possible revisions and reuses.



SOA lifecycle - Model



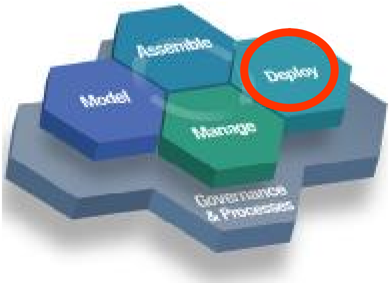
- **The first step in a SOA project has little to do with technology and everything to do with your business.**
- **Service orientation, remember, treats the activities performed by your business as services**
  - so the first step is to establish what these business activities or processes actually are.
  - Documenting your business architecture can be used not only to plan your SOA, but also to optimize your actual business processes.
- **The level to which you model your business process is going to depend on the depth of your anticipated implementation.**

SOA lifecycle - Assemble

Assemble
Deploy
Model
Manage
Governance & Processes

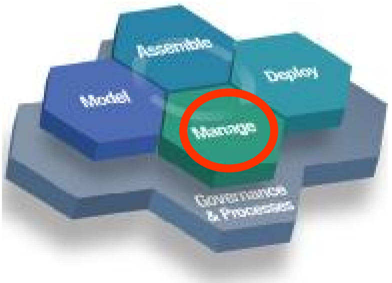
- **Once business processes have been modeled and optimized, developers can implement them by building new services and/or reusing existing services, then assembling them to form composite applications.**
- **In the Model step, you determined what kinds of services you needed and what kinds of data they would access.**
  - Some of the software you needed to implement those services, or access that data, already exists in some form.
- **The Assemble step is about finding functionality that already exists and service-enabling it.**
- **It is also about creating new services that provide functionality and access to data sources that you need to meet the range of business processes to be addressed by your SOA**

SOA lifecycle - Deploy



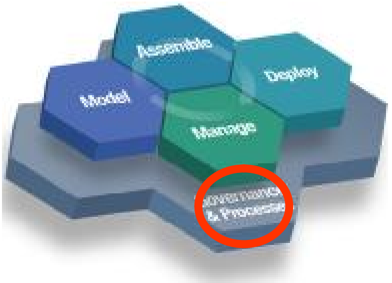
- **Once modeled and assembled, the assets that make up your SOA are deployed into a secure and integrated environment.**
- **This environment avails itself of specialized services that integrate people, processes, and information within your business.**
  - This level of integration helps ensure that all the key elements of your company are connected and working together.
- **Additionally, the deployment needs to meet the performance and availability needs of your business and provide the flexibility to incorporate new services -- and decommission old ones -- without major impact to the system as a whole.**

SOA lifecycle - Manage



- **The system is in place and everything is running smoothly.**
  - You can just let everything run by itself now, right? Wrong.
- **Once deployed, your system needs to be managed and monitored, both from an IT and a business perspective.**
- **In this step, you monitor and optimize the system, finding and correcting inefficiencies and problems.**
  - You will need to deal with issues such as quality of service, security, and general system administration.
- **The completion of this step is the start of a new Model step.**
  - The data gathered during the Manage step is used to revisit the entire SOA lifecycle and start again.

SOA lifecycle - Governance



- **An SOA is a decentralized system**
  - it can include services from different parts of the organization
  - even services that come from outside the organization.
- **Without proper governance, the system can easily spin out of control.**
- **Governance underpins all the lifecycle stages, providing guidance and oversight for the entire SOA system.**
- **It provides both direction and control, shielding both service providers and consumers from encountering the unexpected.**

SOA adoption phases:

- **It's not likely that you will implement a complete business transformation around SOA all at once.**
- **In fact, an all-or-nothing approach increases risk of failure.**
- **Instead, you'll work your way iteratively through the stages of adoption, starting by developing a few services for a pilot project**
  - and then increasingly revamping your IT systems as services that work within an SOA.
- **We'll look at the following SOA adoption phases:**
  - Build services
  - Integrate
  - Transform IT
  - Transform the business

Build services: As needed services with ad hoc linkage

- **In the first phase of SOA adoption, companies typically build SOA services opportunistically.**
- **In other words, as they need to solve specific problems, they choose a service-oriented approach rather than a traditional one.**
- **In this phase, service building will be more focused on solving a specific problem than on transforming the enterprise.**
- **The IT department will build some new services, perhaps transform some existing applications into a set of Web-based services.**
- **Linkage between them will be provided on an as-needed basis, not according to a thorough architecture.**

Integrate: Systematic standardized service interfaces

- **Once you've discovered the advantages, convenience, and maintainability of a loosely coupled architecture, the next step is to start putting that flexibility to work creating new, composite applications by combining services.**
- **For example, an employee status service can be combined with a manager approval service to create a vacation request service.**
- **This process can happen from the top down, that is focusing on the end result and looking for component pieces.**
- **Or it can take place from the bottom up, focusing on the pieces and seeing what can be built from them.**
- **Linkage between them is preplanned and well-defined.**

Transform IT: Composite assemblies of reusable services drawing on functionality from multiple sources

- **This next phase involves transforming your information technology infrastructure to fully take advantage of the SOA.**
- **In this phase, all systems are converted to services based applications, in which loosely coupled becomes the norm and not the exception.**
- **All components of the system are integrated and connected according to the SOA, and no part of the IT system does not work within the SOA.**

Transform the business: Dynamic, event-driven reconfiguration of services

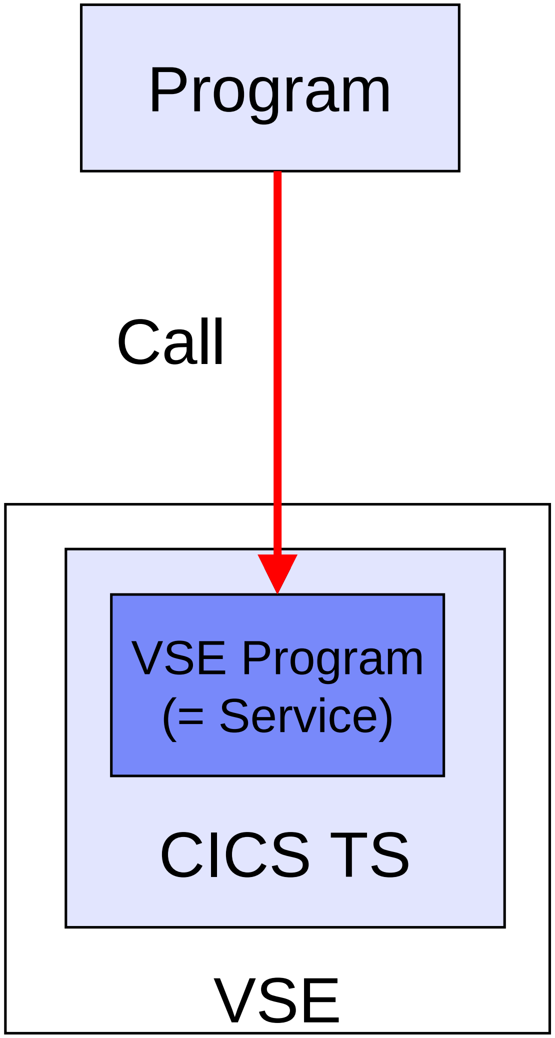
- **In the ultimate phase of SOA maturity**
  - the business integrates completely with the SOA
  - moving to the point at which all appropriate business activities are seen as services that can ultimately be modeled, analyzed, and instantiated in the technical architecture.
- **Reaching this phase takes maximal effort and commitment by the business**
  - however, by reaching this phase the business sees the maximum return from a Service-Oriented Architecture

Why would a VSE customer do SOA ?

- **SOA is modern (hype) and strategic**
  - The management says: We also have to do SOA
- **Easy integration of existing VSE programs into the modern world**
  - Reducing the interface complexity
  - Reuse of existing applications as services
  - Use of standard protocols (XML, SOAP, HTTP)
- **Encapsulation of VSE programs**
  - Disconnecting business and display logic
- **Integration of VSE into a Microsoft .Net environment**
  - You do not wan to use Java
  - You already have a Microsoft environment

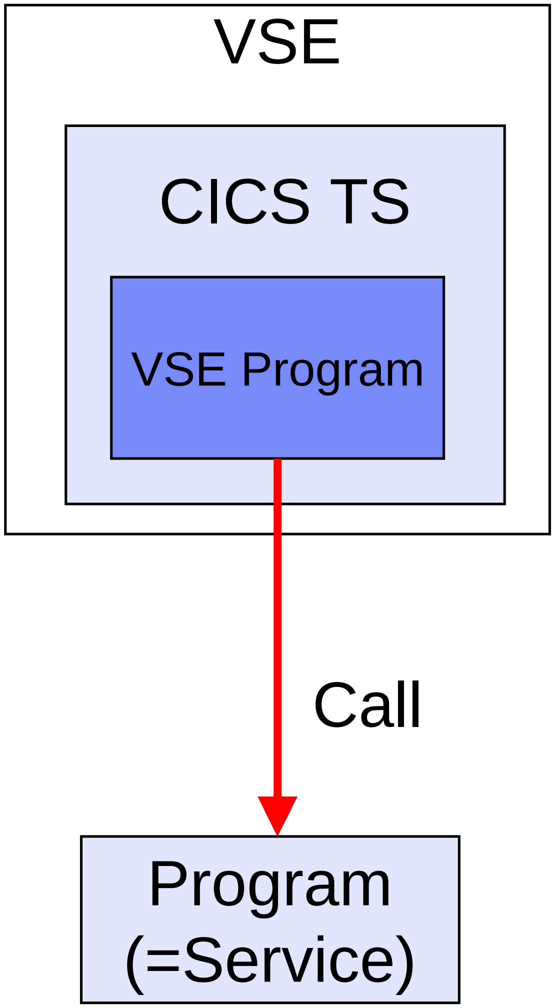
What is a Web Service?

- **Assume you have a VSE program that implements some kind of important business logic**
- **Someone else (outside VSE) wants to use this program**
  - 1. Possibility: Rewrite the same logic
    - May need access to VSE data
    - Changes/Fixes in VSE code needs to be re-done in new code also
  - 2. Possibility: Call the VSE program from remote
    - VSE program can be treated as a Web Service
    - VSE is the Web Service provider



What is a Web Service?

- **Assume someone has a program that implements some kind of important business logic**
- **You want to use this program inside a VSE application**
  - 1. Possibility: Rewrite the same logic
    - May need access to the remote data
    - Changes/Fixes in code needs to be re-done in VSE code also
  - 2. Possibility: Call the external program from VSE
    - External program can be treated as a Web Service
    - VSE is the Web Service Requestor



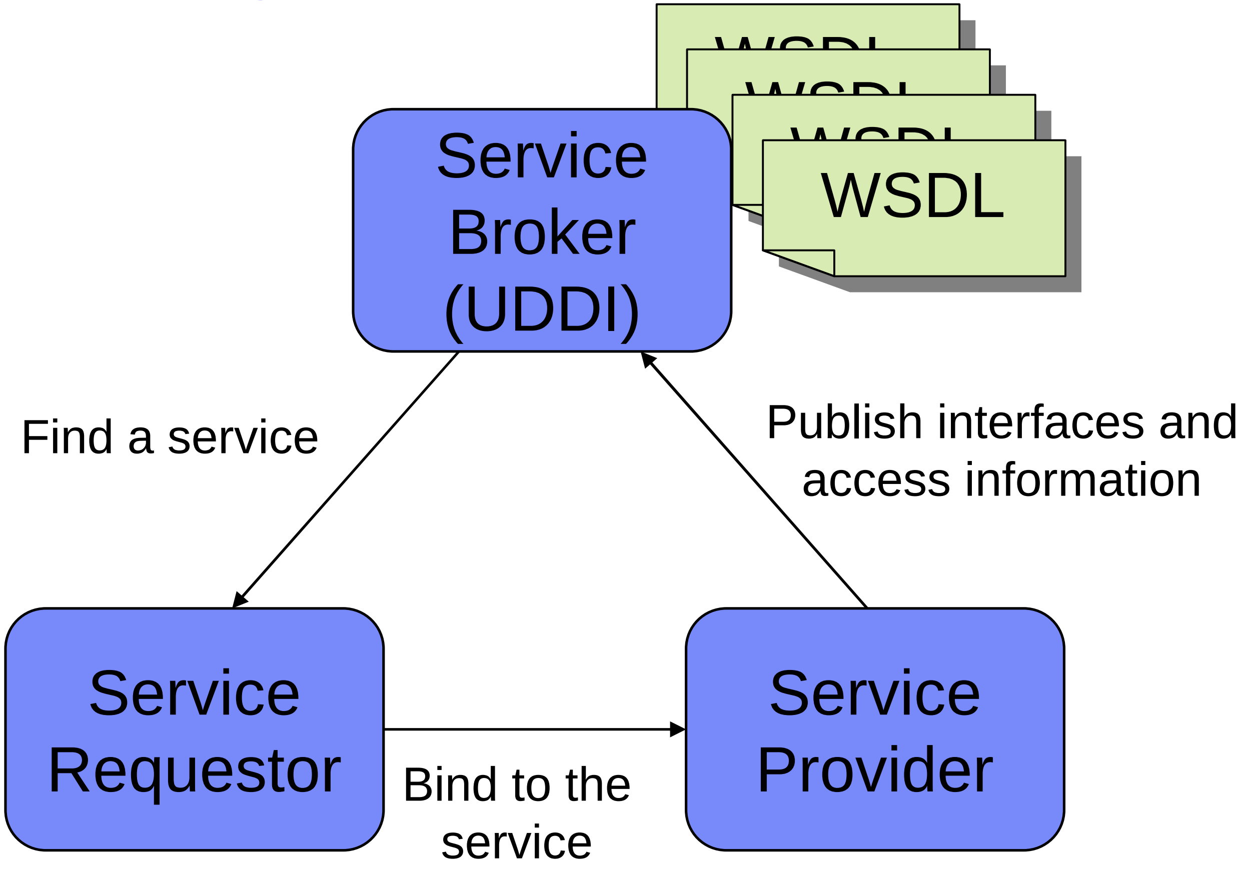
Web Services - Summary

XML <xsl:stylesheet version="1.0" xmlns:xsl="http://

SOAP -

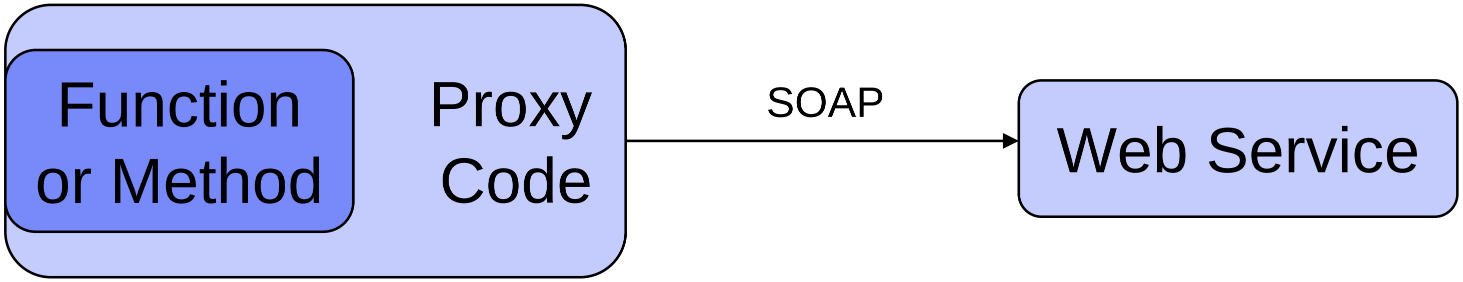
HTTP - Transport Service

TCP/IP - Street

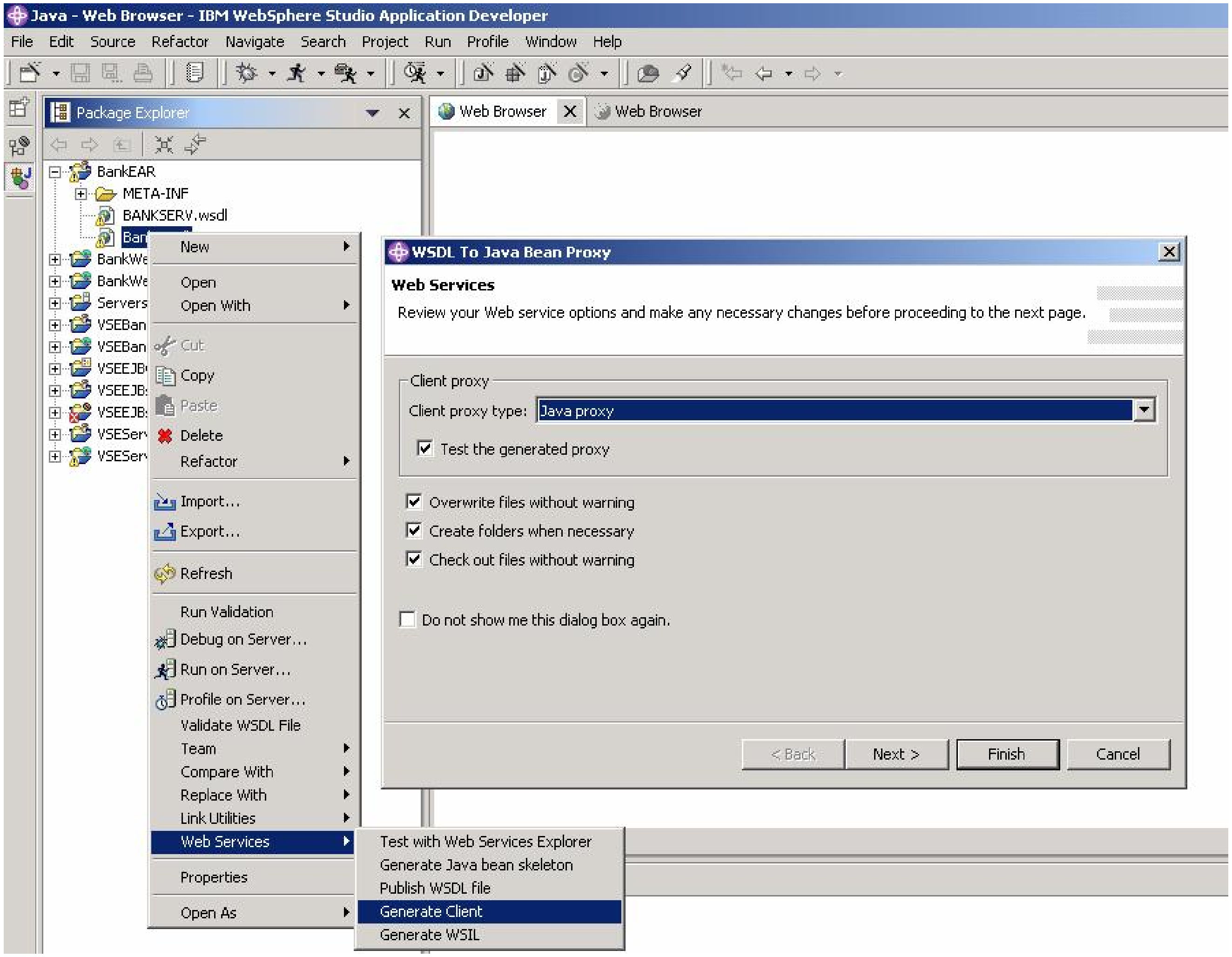


Using Web Services with Java or MS .Net

- **Use/Call an existing Web Service**
  - You know that a specific Web Service exists
  - Locate the Web Service Description (WSDL)
  - Use a tool like Rational Application Developer (RAD/WSAD) or Microsoft Visual Studio and import the WSDL
    - Generate “proxy code” that implements all things needed to invoke the Web Service
    - Applications will call a function or method of the proxy code as it would implement the service locally



Using Web Services with Java or MS .Net



WSDL To Java Bean Proxy

Web Services

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

Client proxy

Client proxy type: Java proxy

☑ Test the generated proxy

☑ Overwrite files without warning

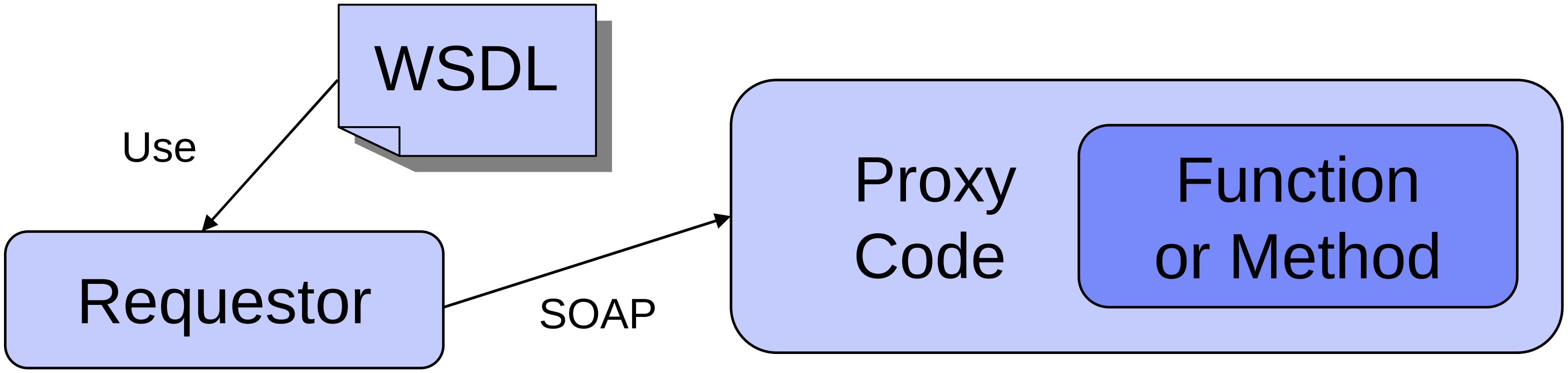
☑ Create folders when necessary

☑ Check out files without warning

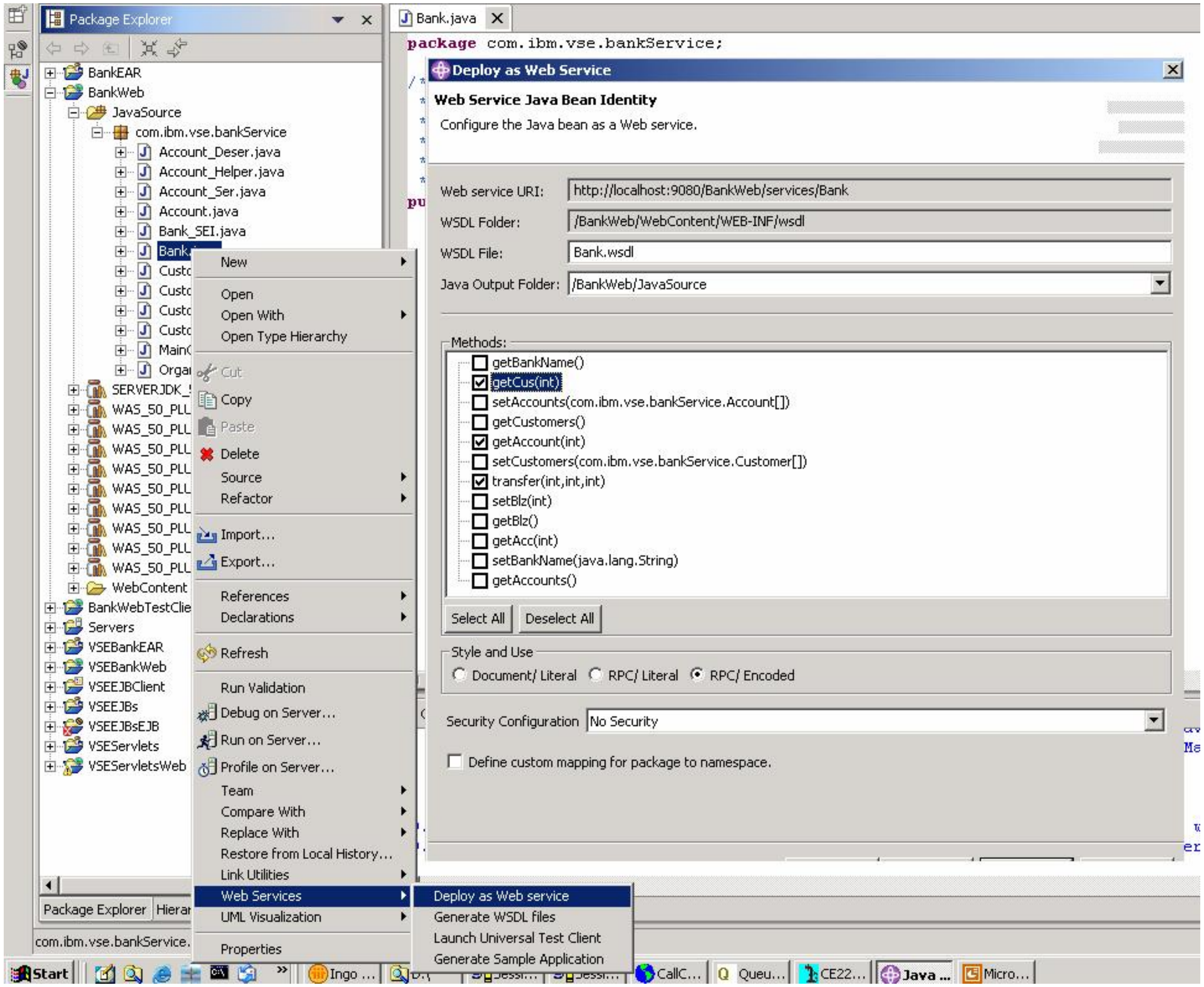
□ Do not show me this dialog box again.

Using Web Services with Java or MS .Net

- **Create/provide a new Web Service**
  - You have a function or method that implements some kind of service that you want to provide
  - Use a tool like Rational Application Developer (RAD/WSAD) or Microsoft Visual Studio to model a Web Service
    - Generate a Web Service Description (WSDL) and publish it
    - Generate “proxy code” that makes the function or method callable from outside as a Web Service via SOAP
    - Deploy it in an application server



Using Web Services with Java or MS .Net



```
package com.ibm.vse.bankService;
```

Deploy as Web Service

Web Service Java Bean Identity

Configure the Java bean as a Web service.

Web service URI: http://localhost:9080/BankWeb/services/Bank

WSDL Folder: /BankWeb/WebContent/WEB-INF/wsdl

WSDL File: Bank.wsdl

Java Output Folder: /BankWeb/JavaSource

Methods:

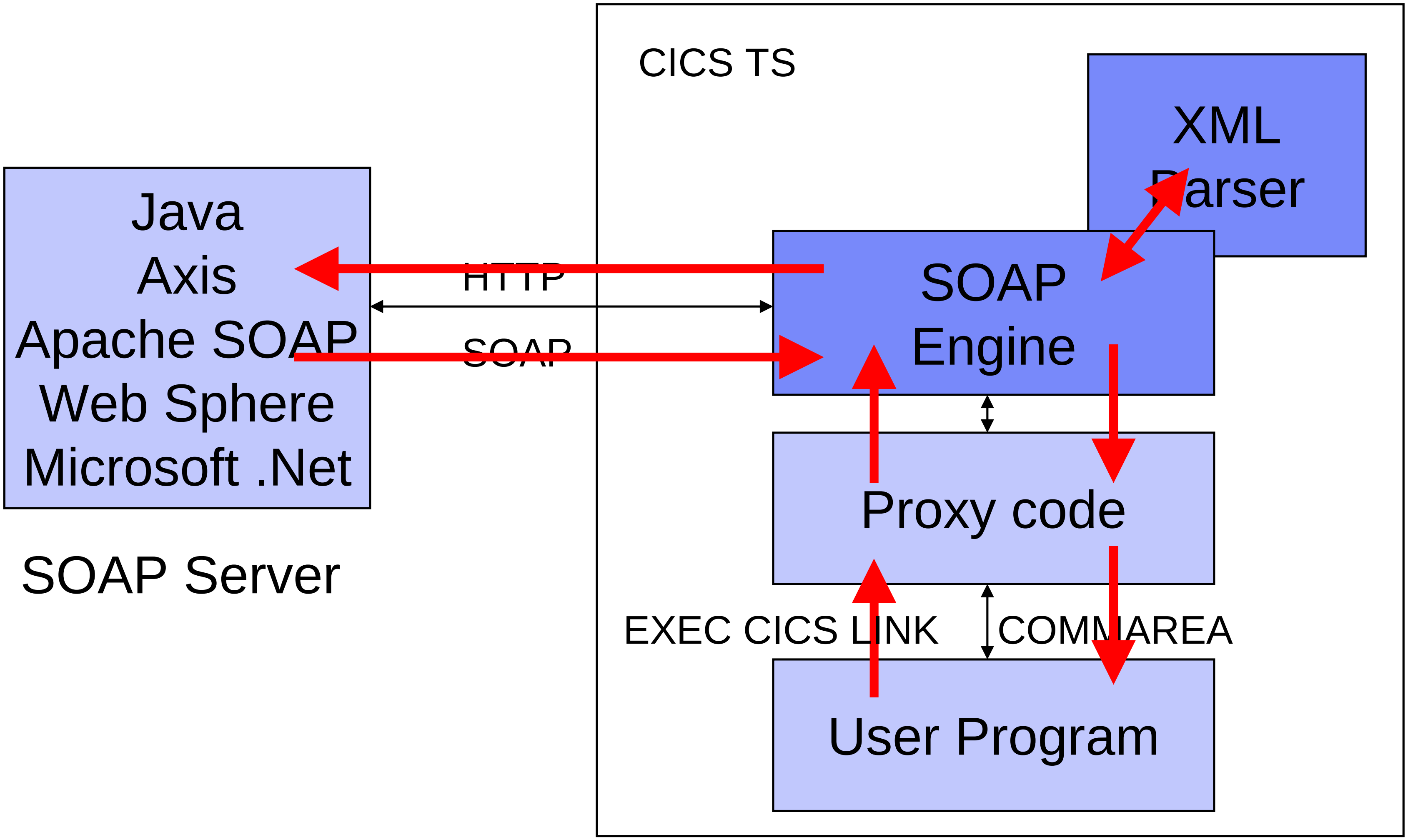
- □ getBankName()
- ☑ getCus(int)
- □ setAccounts(com.ibm.vse.bankService.Account[])
- □ getCustomers()
- ☑ getAccount(int)
- □ setCustomers(com.ibm.vse.bankService.Customer[])
- ☑ transfer(int,int,int)
- □ setBlz(int)
- □ getBlz()
- □ getAcc(int)
- □ setBankName(java.lang.String)
- □ getAccounts()

Style and Use: ○ Document/ Literal ○ RPC/ Literal ● RPC/ Encoded

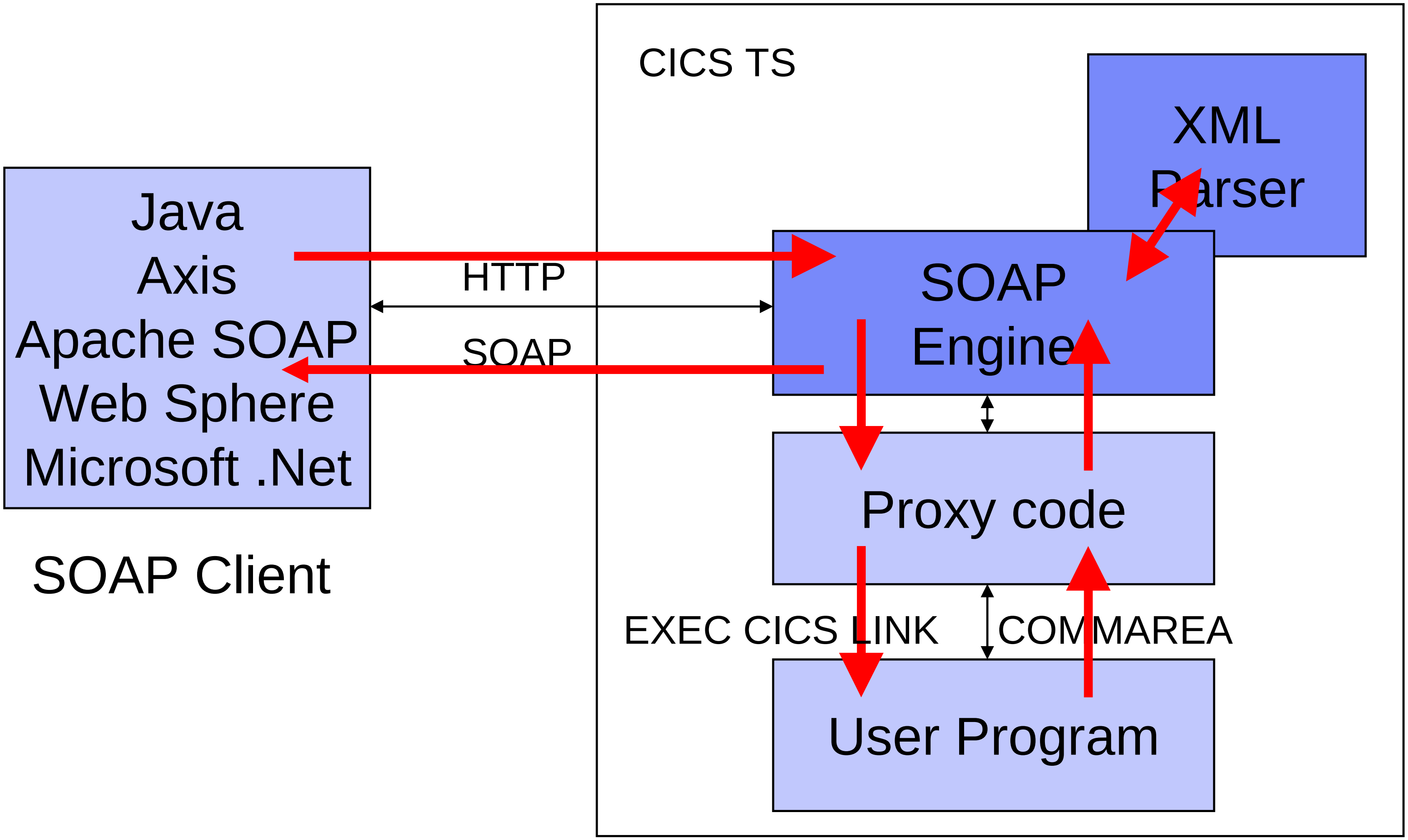
Security Configuration: No Security

□ Define custom mapping for package to namespace.

Using Web Services with VSE – SOAP client



Providing Web Services with VSE – SOAP server



VSE SOAP Engine

- **Part of VSE Connectors**
- **Translates SOAP messages into/from a native format**
  - Receives/sends the SOAP message via HTTP
  - Parses/Generates XML
  - Interprets/creates the SOAP message
  - Serializes/de-serializes the input and output values
    - Translates it from textual format into a native format and vice versa
  - Calls service provider program
  - Gets called when invoking external Web Services
- **System requirements**
  - VSE/ESA 2.6, 2.7 or z/VSE 3.1
  - CICS Transaction Server
    - CICS/VSE 2.3 can not be used
  - TCP/IP

VSE SOAP Engine

- **Input/Output parameters**
  - Each parameter is represented by a TS-Queue entry
    - Parameter name (e.g. “StockPrice”)
    - Parameter type (e.g. “String”)
    - Parameter value (e.g. “34.5”)
    - Length of the parameter data
  - Input parameters are put onto the Input-Queue
  - Output parameters are read from the Output Queue

```
01 SOAP-PARAM-HDR.
   05  NAME                       PIC X(16).
   05  TYPENAME                   PIC X(16).
   05  LENGTH                     PIC 9(8)  COMP.
   05  TYPECODE                   PIC 9(8)  COMP.
   05  VALUE                      PIC X(20).

EXEC CICS WRITEQ TS QUEUE(OUTQUEUE)
               FROM(SOAP-PARAM-HDR)
               LENGTH(TS-QUEUE-LENGTH-OUT)
               RESP(COMMAND-RESPONSE)
               END-EXEC.
```

Why use a proxy program?

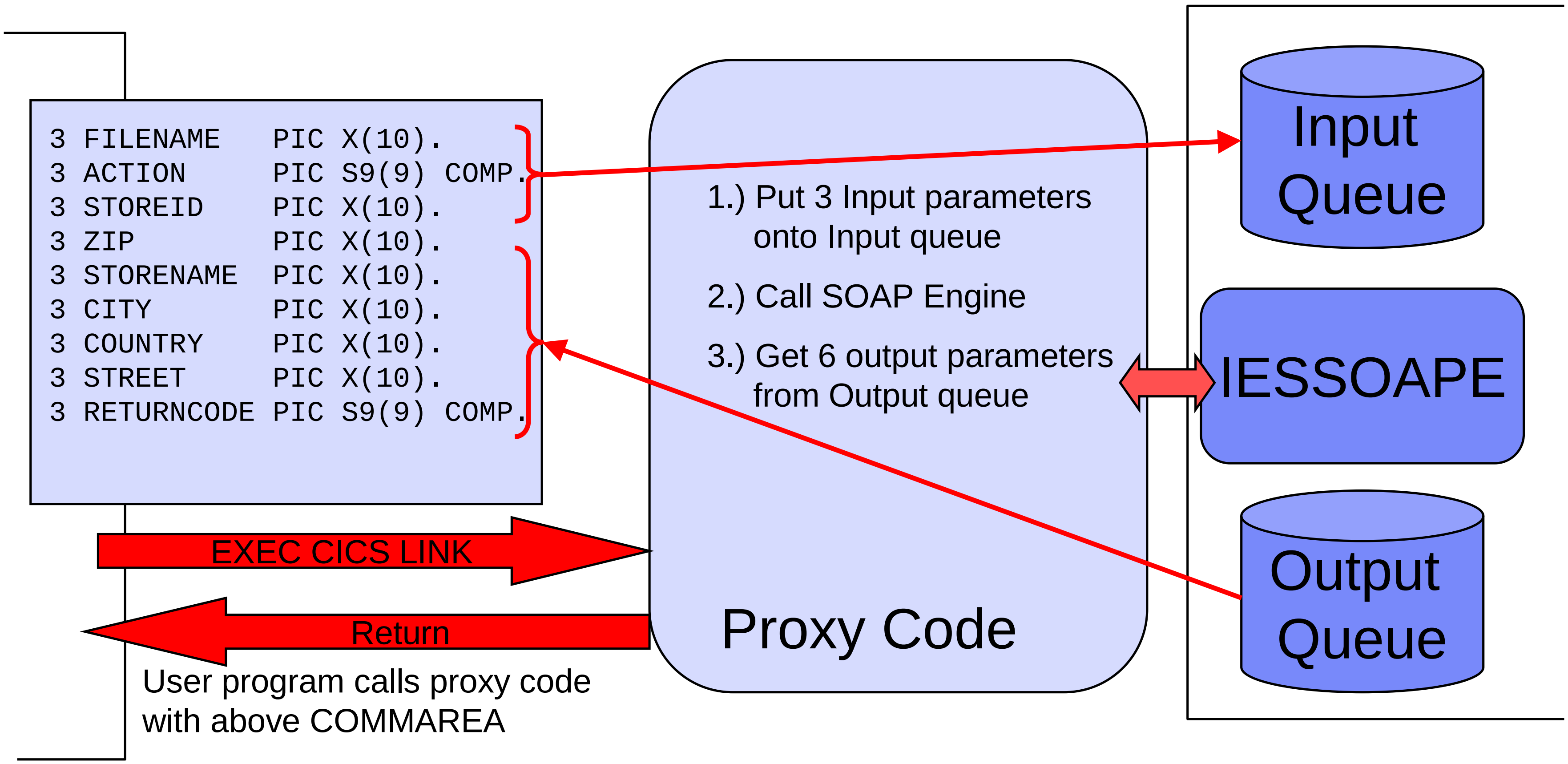
- **Although the SOAP Engine interface uses standard CICS methods, special coding is needed to interface with the VSE SOPA Engine.**
- **Proxy code ‘maps’ between standard COMMAREA interface and SOAP Engine**
  - All SOAP specific handling is done in proxy code
  - User applications calls the proxy code or gets called by the proxy code like a local program call (EXEC CICS LINK) using a COMMAREA
  - User COMMAREA format can be freely defined by user
  - Proxy code copies fields from COMMAREA into TS queue entries and vice versa

Why use a proxy program?

- **All SOAP implementations use some kind of “proxy code”**
  - Java (RAD/WSAD)
  - Microsoft .Net
  - …
- **The proxy code maps the implementation specifics of the SOAP engine to a common interface**
- **The proxy code is generated using the information from the WSDL**
- **The proxy code is usually not modified directly by user**
- **VSE uses the same technique as other SOAP implementations**

What does the proxy code do?

§To call an external Web Service



```
3 FILENAME   PIC X(10).
3 ACTION     PIC S9(9) COMP.
3 STOREID    PIC X(10).
3 ZIP        PIC X(10).
3 STORENAME  PIC X(10).
3 CITY       PIC X(10).
3 COUNTRY    PIC X(10).
3 STREET     PIC X(10).
3 RETURNCODE PIC S9(9) COMP.
```

User program calls proxy code with above COMMAREA

1.) Put 3 Input parameters onto Input queue

2.) Call SOAP Engine

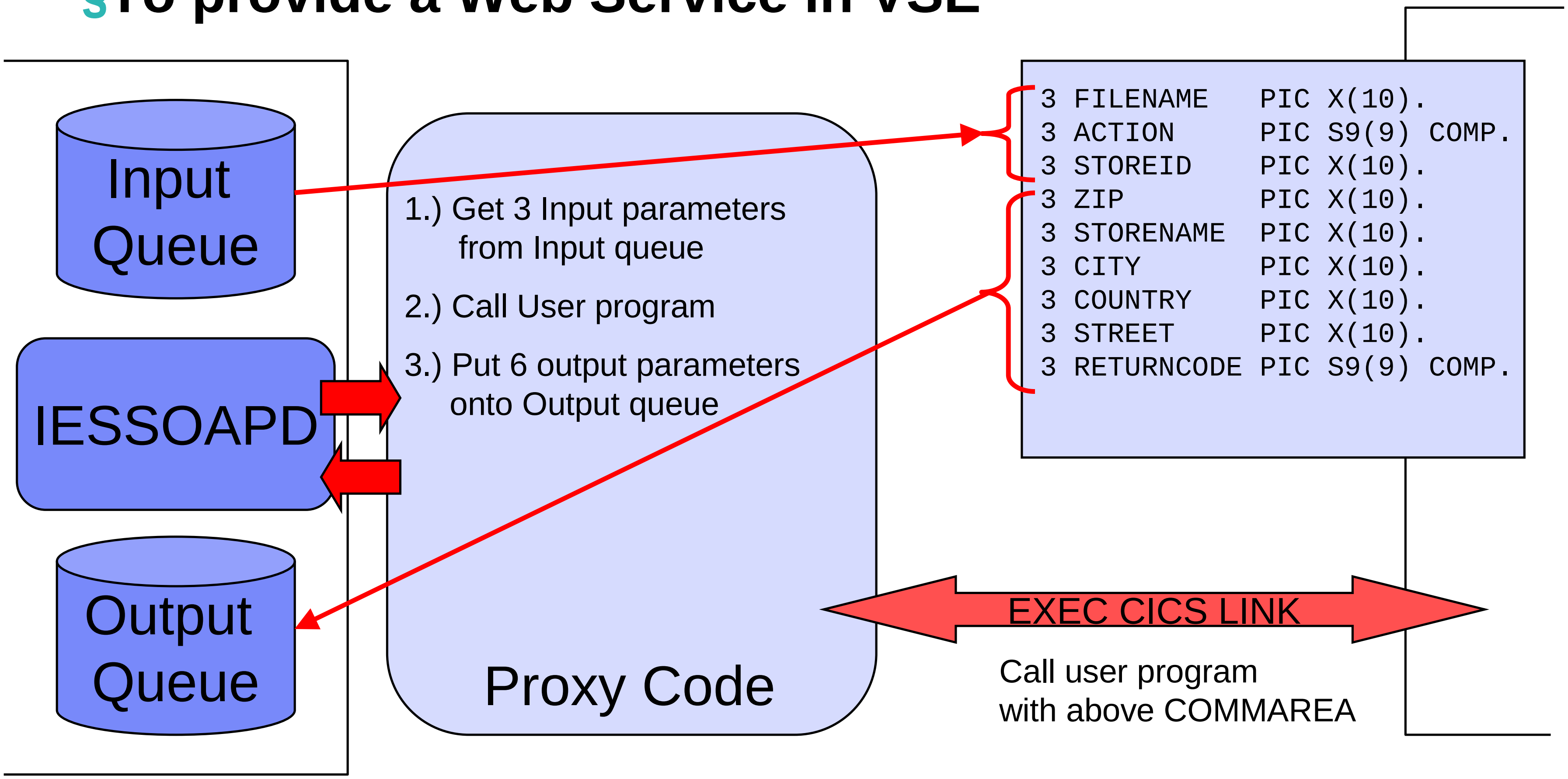
3.) Get 6 output parameters from Output queue

What does the proxy code do?

- **To call an external Web Service**
  - Proxy code gets invoked via EXEC CICS LINK by user program
  - Put input parameters onto input queue
  - Setup parameter area for Web Service call
    - Endpoint URL
    - Name of method to call
    - Names of Input and Output queues
  - Call SOAP Engine
    - EXEC CICS LINK to IESSOAPE
  - On return
    - Check for errors
    - Get output parameters from output queue
    - Return to user program

What does the proxy code do?

- To provide a Web Service in VSE



1.) Get 3 Input parameters from Input queue

2.) Call User program

3.) Put 6 output parameters onto Output queue

```
3 FILENAME   PIC X(10).
3 ACTION     PIC S9(9) COMP.
3 STOREID    PIC X(10).
3 ZIP        PIC X(10).
3 STORENAME  PIC X(10).
3 CITY       PIC X(10).
3 COUNTRY    PIC X(10).
3 STREET     PIC X(10).
3 RETURNCODE PIC S9(9) COMP.
```

Call user program
with above COMMAREA

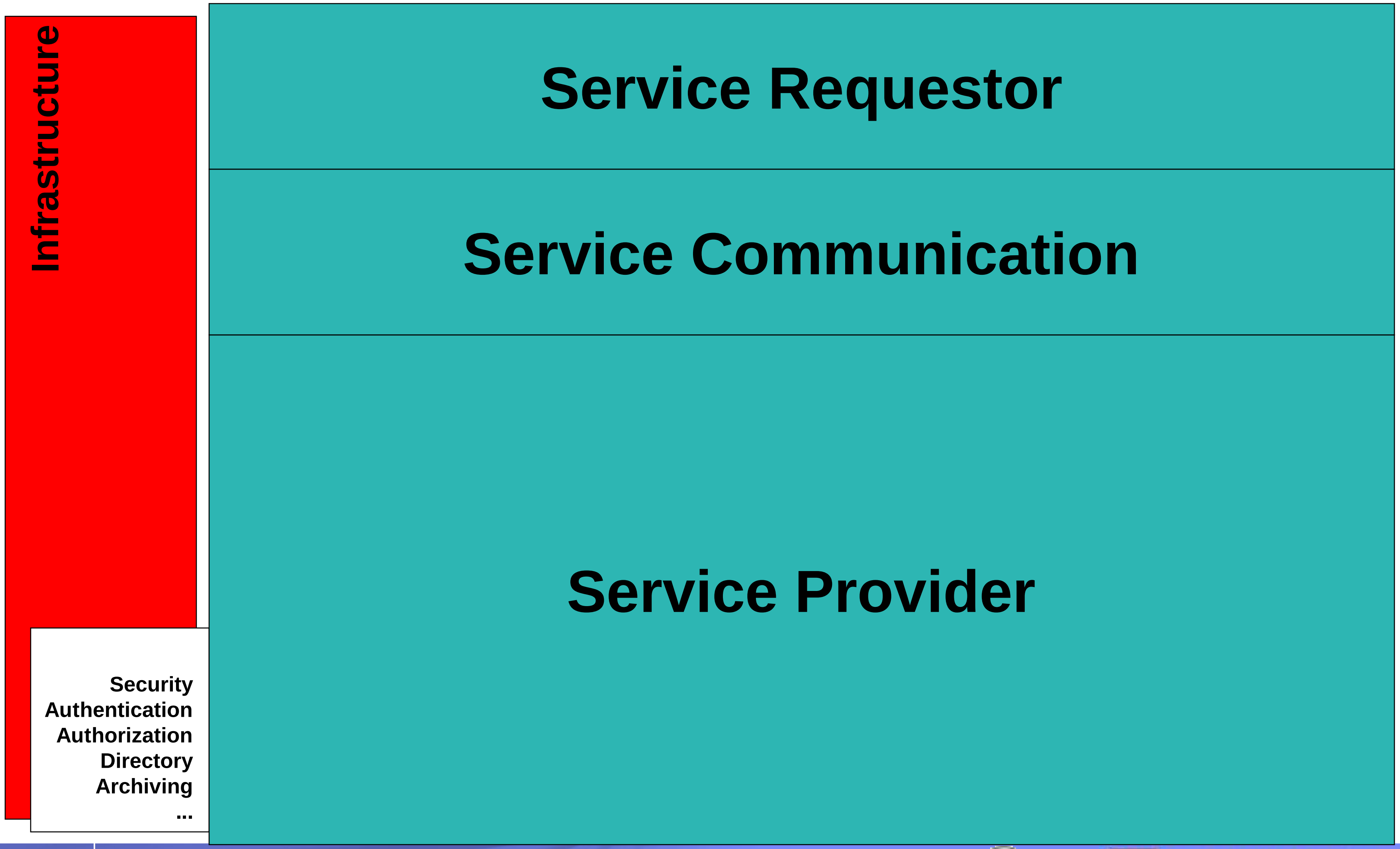
What does the proxy code do?

- **To provide a Web Service in VSE**
  - Proxy code gets called by SOAP Engine
  - Get input parameters from input queue
  - Prepare user COMMAREA
  - Call user program
    - EXEC CICS LINK service provider program (user program)
  - On return
    - Check for errors
    - Put output parameters onto output queue
    - Return to SOAP engine

Which programs can be used with Web Services?

- **Which VSE programs can be used as an Web Service?**
  - All CICS TS programs that implement the “service” you want to provide
    - In any programming language (COBOL, PL/1, C, Assembler)
  - You should separate business logic from user interface
    - 3270 screens or BMS maps can not be used
  - The proxy code calls your program with EXEC CICS LINK and an user defined COMMAREA
- **Which VSE programs can call an external Web Service?**
  - All CICS TS programs that can call another program with a COMMAREA
    - In any programming language (COBOL, PL/1, C, Assembler)
  - Your program calls the proxy code with EXEC CICS LINK and an user defined COMMAREA

Layered Software Architecture



Infrastructure

Service Requestor

Service Communication

Service Provider

Security
Authentication
Authorization
Directory
Archiving
...

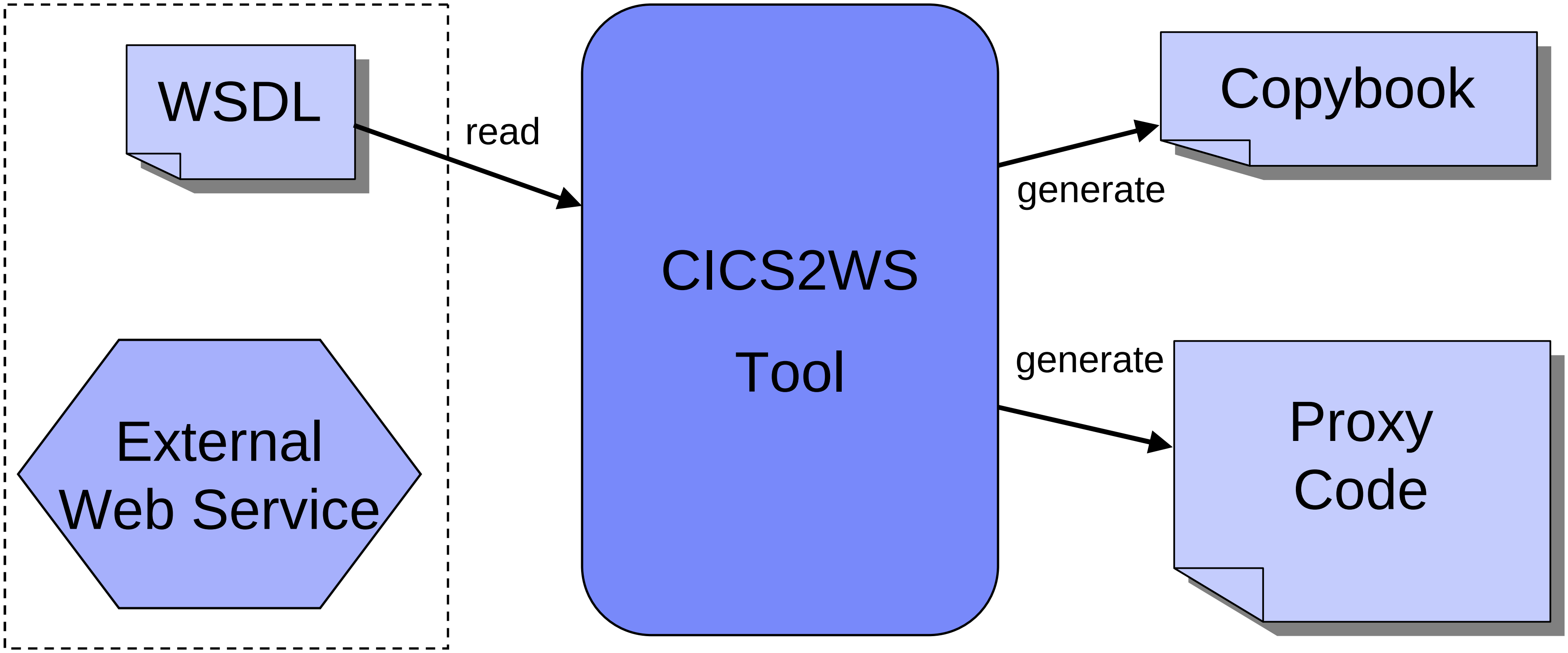
How to write the proxy code

- **You can write the proxy code “by hand”**
  - Not very difficult, use samples as skeleton
  - COBOL Example (from Rich Smrcina):
    - ftp://ftp.software.ibm.com/eserver/zseries/zos/vse/download/xmps/soap_cobol_rsmrcina.zip
- **Use the new CICS2WS tool**
  - Generates proxy code and WSDL files
  - Proxy code is in assembler language
    - No extra charged compiler needed (e.g. COBOL or PL/I)
    - Code is very simple, straight forward
    - Usually no manual changes needed in proxy code

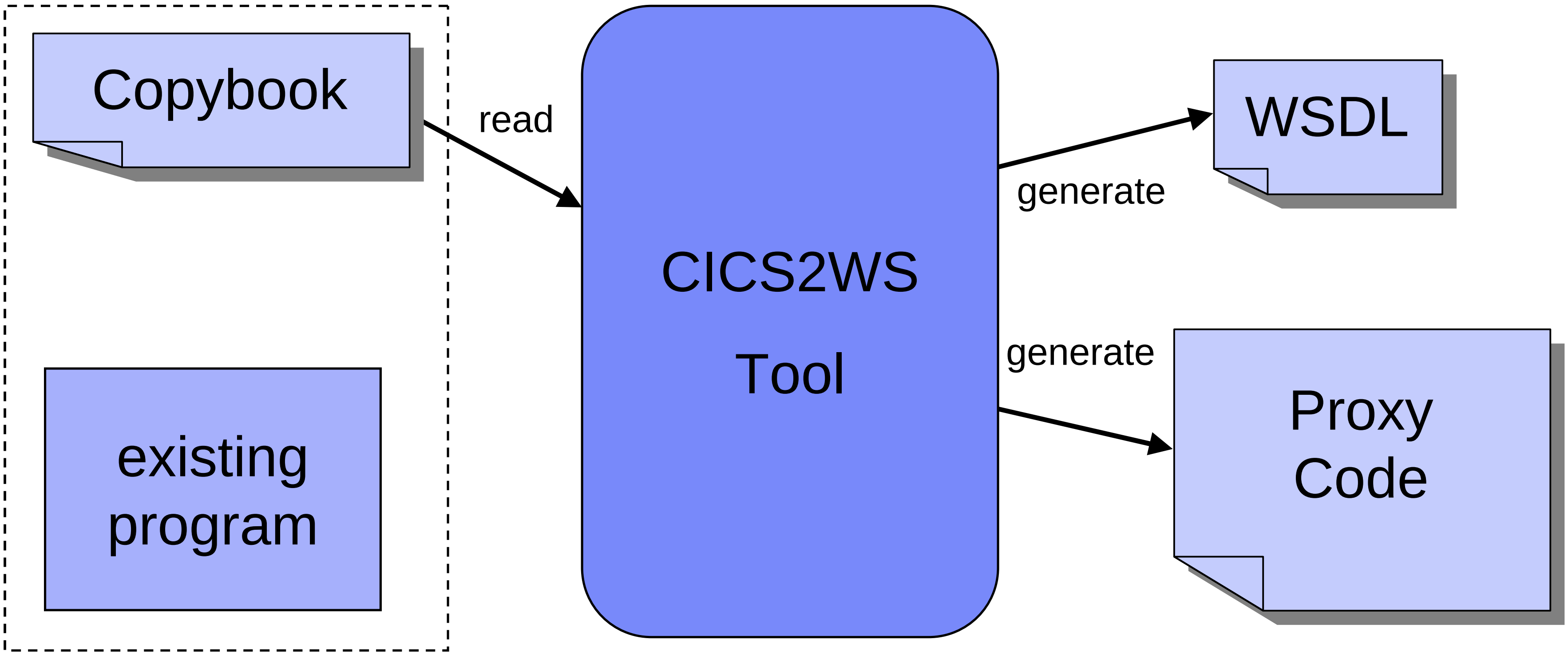
New CICS2WS Tool

- **The tool runs on your PC or workstation**
- **Implemented in Java**
- **VSE as a SOAP client (service requestor)**
  - Reads the WSDL file
  - Generates the proxy code (Assembler)
  - Generates a COMMAREA mapping (copybook)
    - in COBOL, PL/I or Assembler
- **VSE as a SOAP server (service provider)**
  - Reads a given COMMAREA mapping (copybook)
    - in COBOL, PL/I or Assembler
  - Generates the proxy code (Assembler)
  - Generates the WSDL file

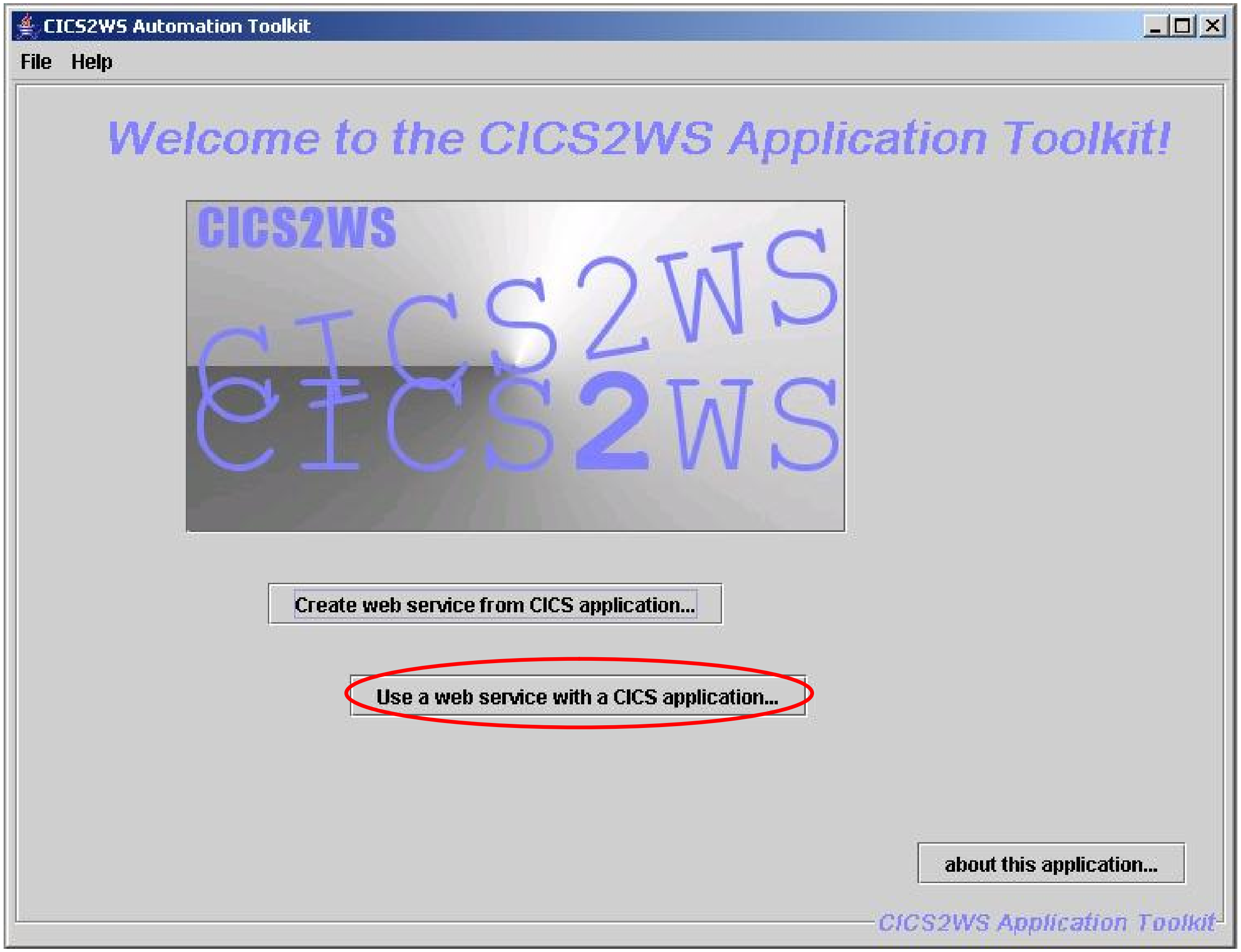
VSE as a SOAP client (service requestor)



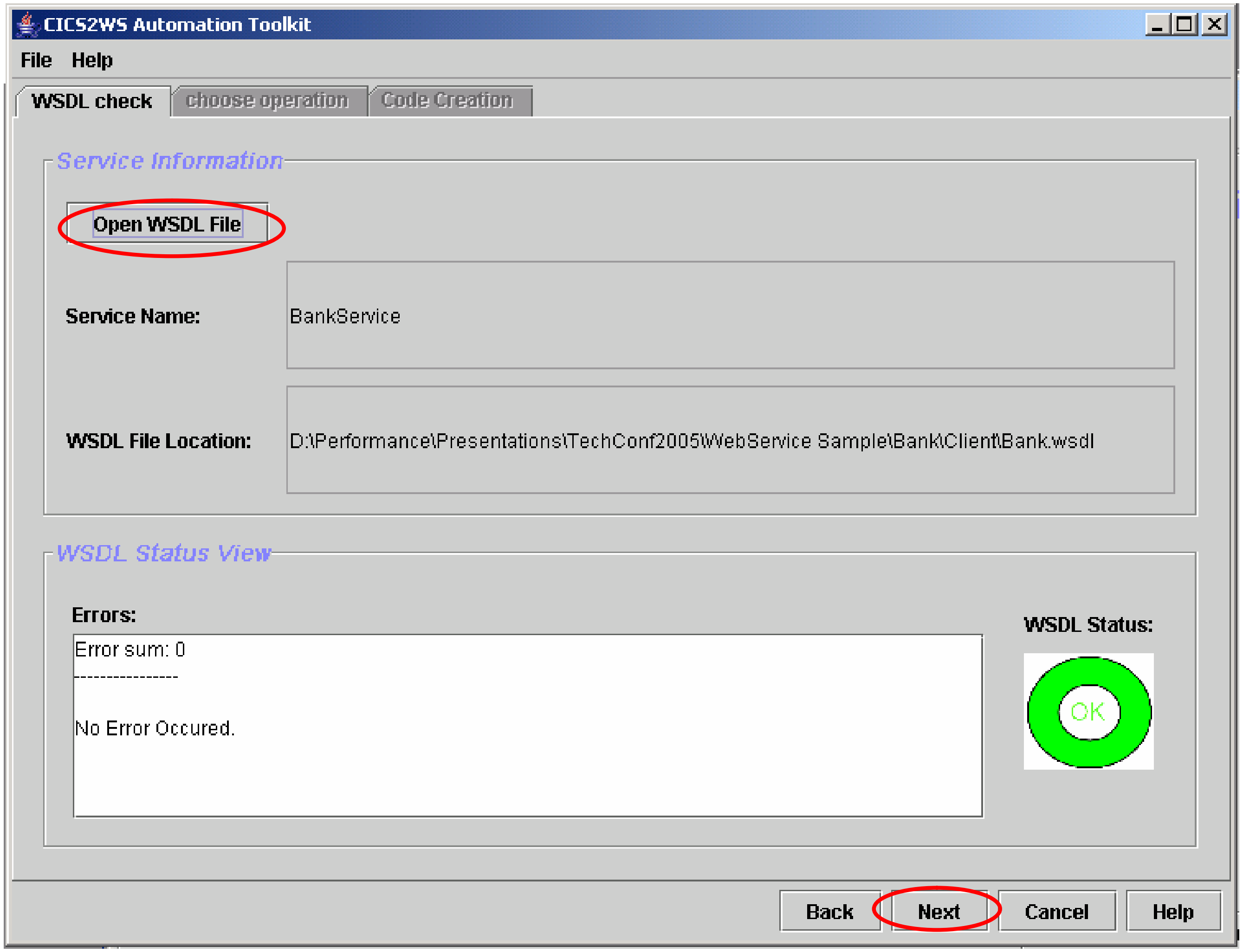
VSE as a SOAP server (service provider)



Live Demo – Use a Web Service


CICS2WS Automation Toolkit
File
Help
Welcome to the CICS2WS Application Toolkit!
CICS2WS
CICS2WS
CICS2WS
Create web service from CICS application...
Use a web service with a CICS application...
about this application...
CICS2WS Application Toolkit

Live Demo – Use a Web Service



Service Name: BankService

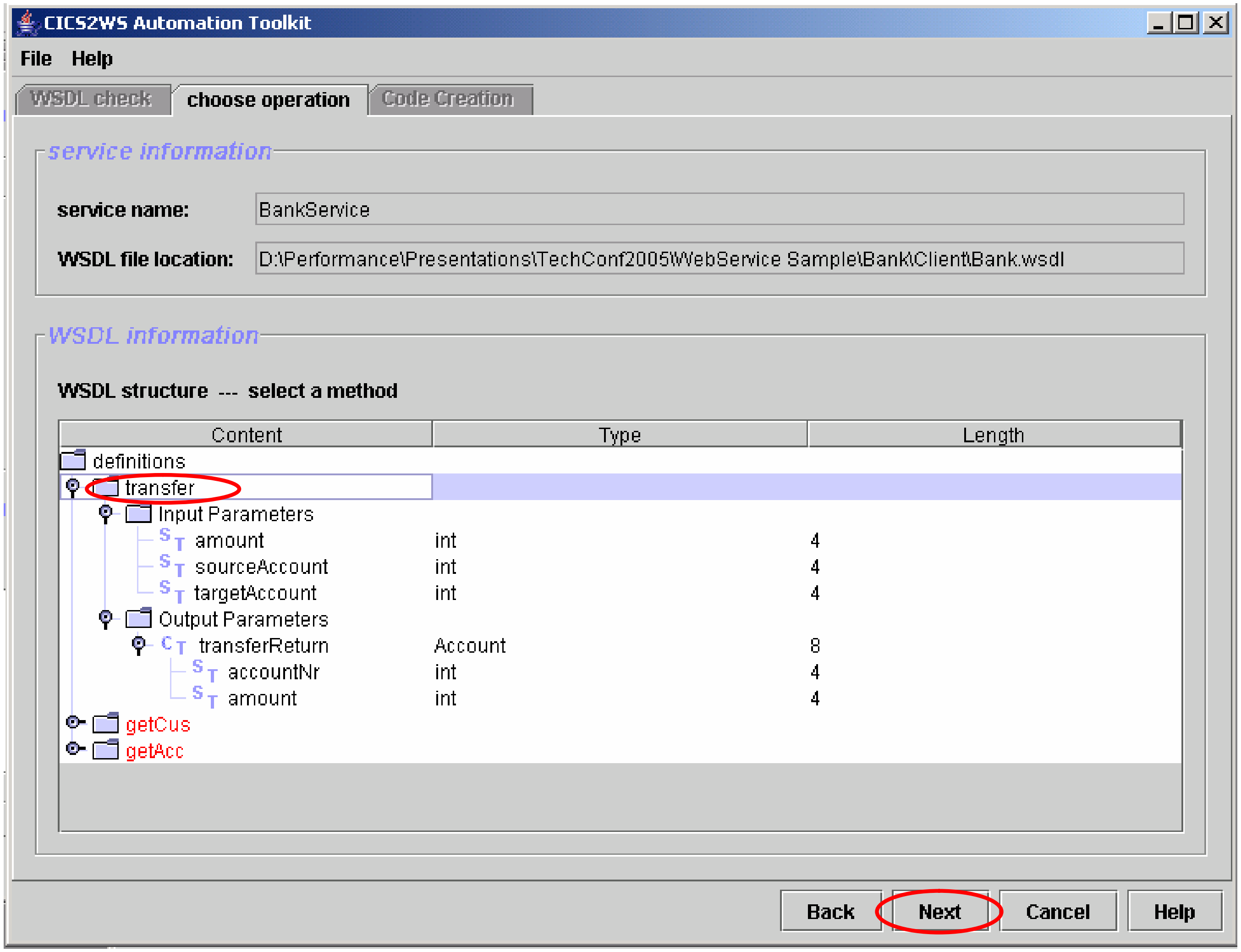
WSDL File Location: D:\Performance\Presentations\TechConf2005\WebService Sample\Bank\Client\Bank.wsdl

Errors:

```
Error sum: 0
----------------

No Error Occured.
```

Live Demo – Use a Web Service



service information

service name: BankService

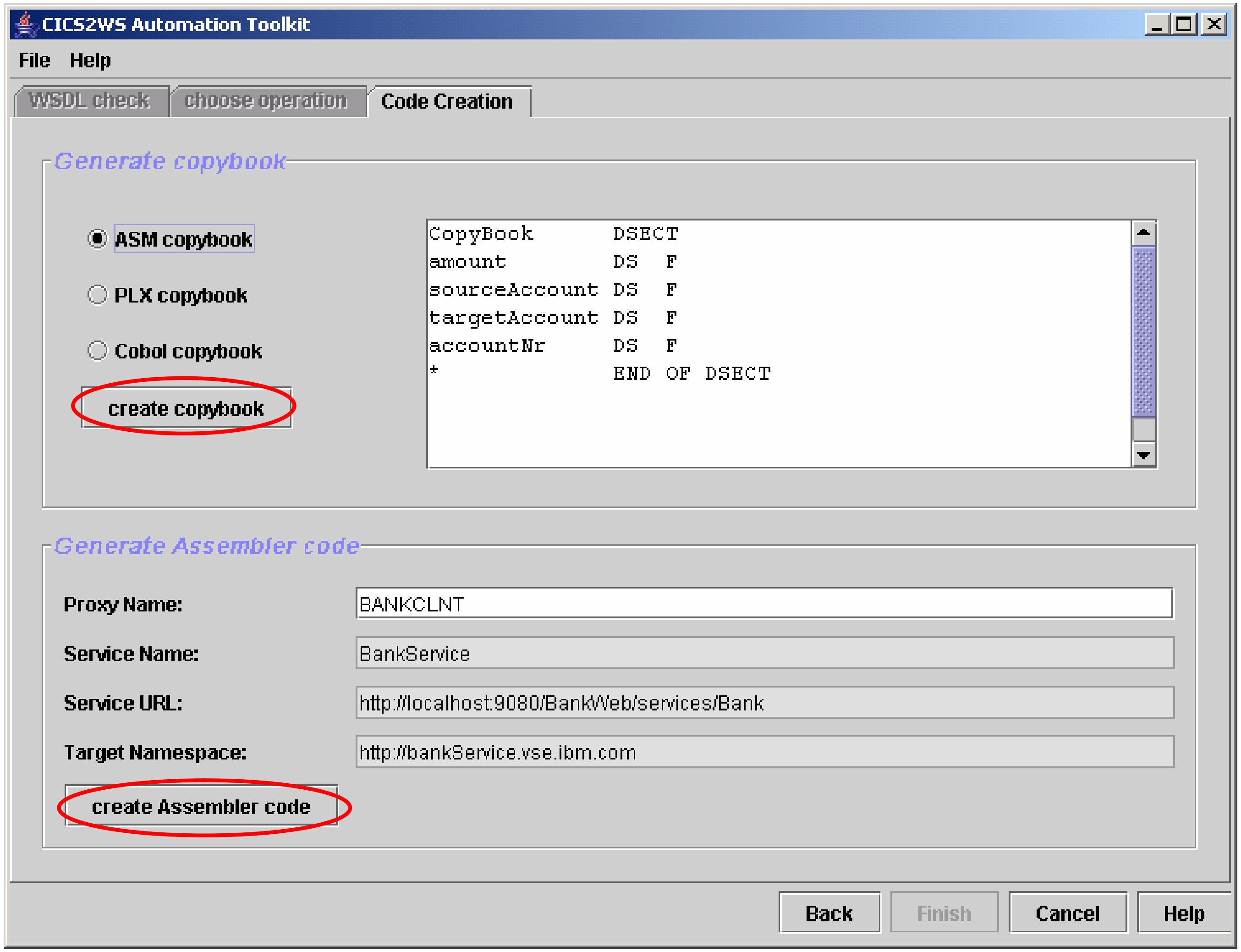
WSDL file location: D:\Performance\Presentations\TechConf2005\WebService Sample\Bank\Client\Bank.wsdl

WSDL information

WSDL structure --- select a method

Content	Type	Length
definitions		
transfer		
Input Parameters		
amount	int	4
sourceAccount	int	4
targetAccount	int	4
Output Parameters		
transferReturn	Account	8
accountNr	int	4
amount	int	4
getCus		
getAcc		

Live Demo – Use a Web Service



Generate copybook

● ASM copybook

○ PLX copybook

○ Cobol copybook

create copybook

```
CopyBook      DSECT
amount        DS   F
sourceAccount DS   F
targetAccount DS   F
accountNr     DS   F
*             END  OF  DSECT
```

Generate Assembler code

Proxy Name: BANKCLNT

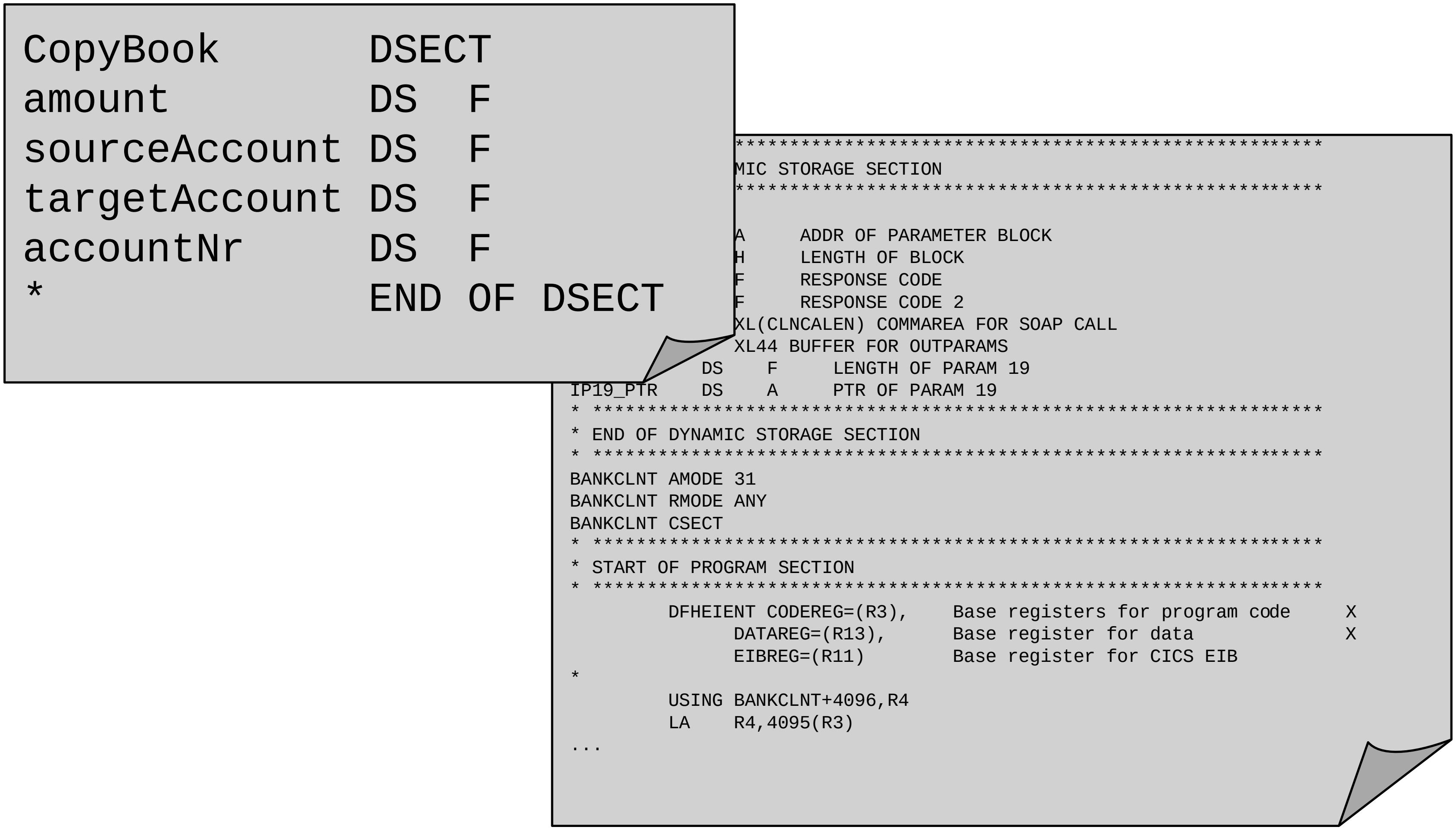
Service Name: BankService

Service URL: http://localhost:9080/BankWeb/services/Bank

Target Namespace: http://bankService.vse.ibm.com

create Assembler code

Live Demo – Use a Web Service

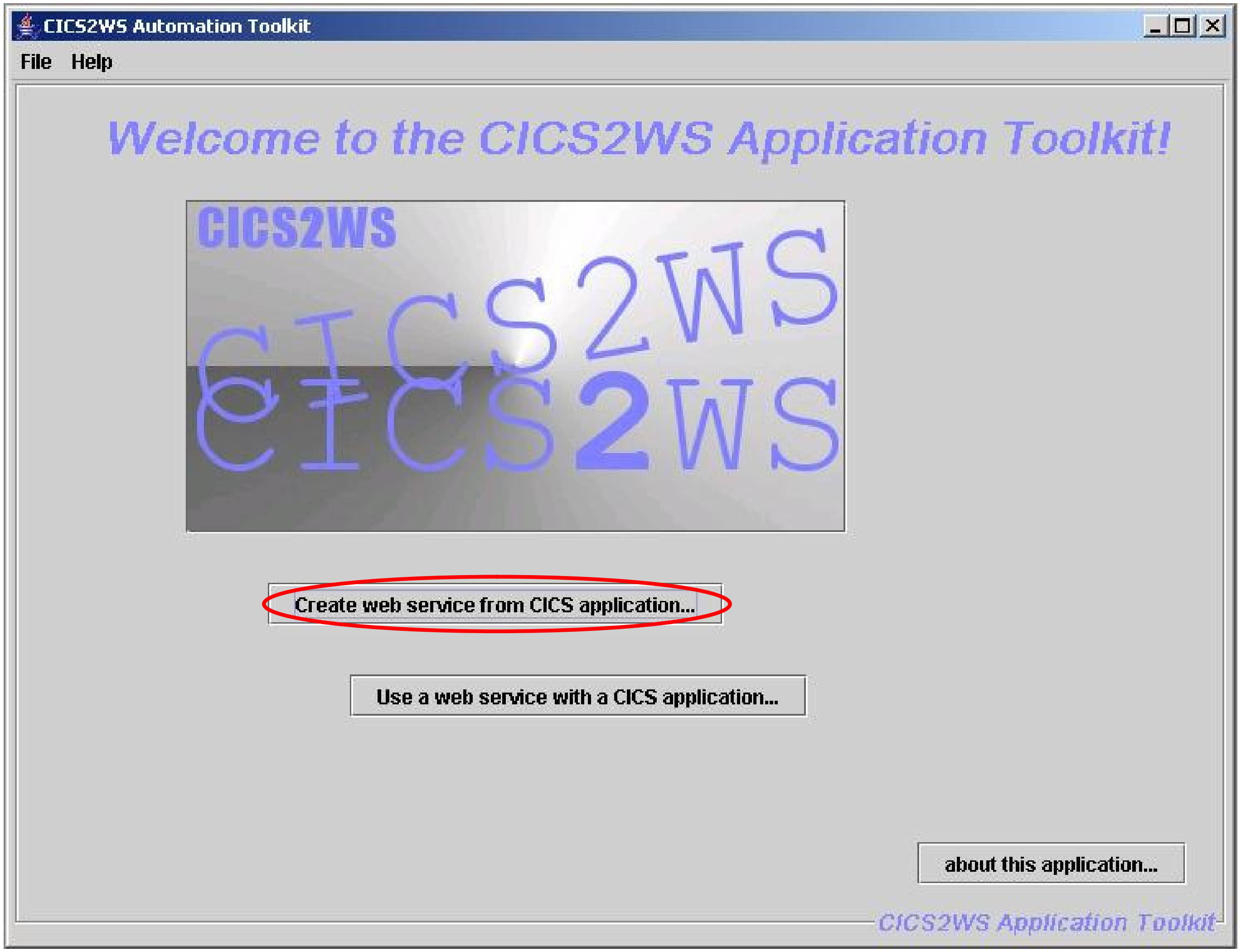


```
CopyBook      DSECT
amount        DS   F
sourceAccount DS   F
targetAccount DS   F
accountNr     DS   F
*             END OF DSECT
```

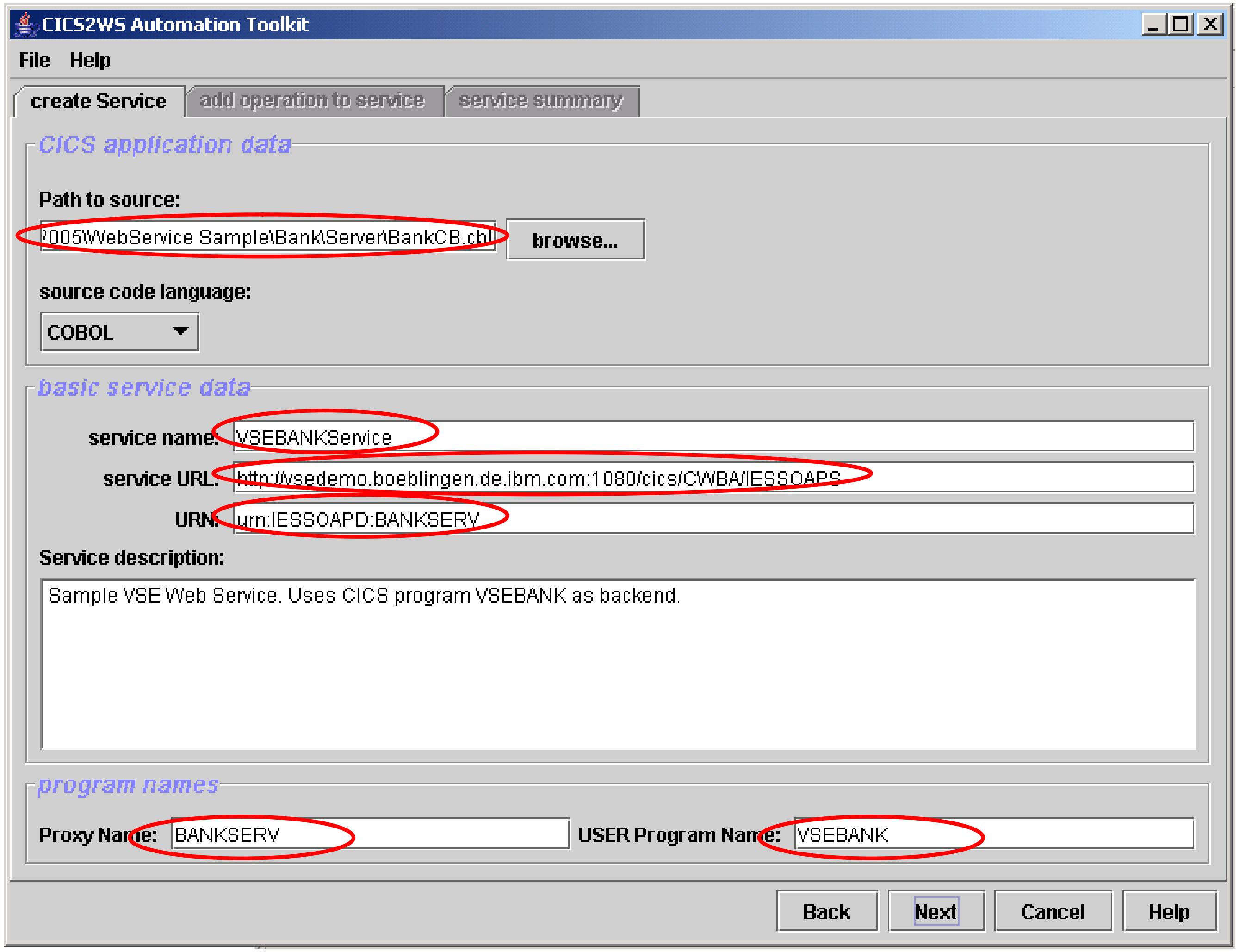
```
               ***********************************************************
               MIC STORAGE SECTION
               ***********************************************************

               A     ADDR OF PARAMETER BLOCK
               H     LENGTH OF BLOCK
               F     RESPONSE CODE
               F     RESPONSE CODE 2
               XL(CLNCALEN) COMMAREA FOR SOAP CALL
               XL44 BUFFER FOR OUTPARAMS
            DS    F     LENGTH OF PARAM 19
IP19_PTR    DS    A     PTR OF PARAM 19
* *******************************************************************
* END OF DYNAMIC STORAGE SECTION
* *******************************************************************
BANKCLNT AMODE 31
BANKCLNT RMODE ANY
BANKCLNT CSECT
* *******************************************************************
* START OF PROGRAM SECTION
* *******************************************************************
         DFHEIENT CODEREG=(R3),    Base registers for program code     X
               DATAREG=(R13),      Base register for data              X
               EIBREG=(R11)        Base register for CICS EIB
*
         USING BANKCLNT+4096,R4
         LA    R4,4095(R3)
...
```

Live Demo – Create a Web Service


CICS2WS Automation Toolkit
File
Help
Welcome to the CICS2WS Application Toolkit!
CICS2WS
CICS2WS
CICS2WS
Create web service from CICS application...
Use a web service with a CICS application...
about this application...
CICS2WS Application Toolkit

Live Demo – Create a Web Service



CICS application data

Path to source: 2005\WebService Sample\Bank\Server\BankCB.cbl

source code language: COBOL

basic service data

service name: VSEBANKService

service URL: http://vsedemo.boeblingen.de.ibm.com:1080/cics/CWBA/IESSOAPS

URN: urn:IESSOAPD:BANKSERV

Service description:

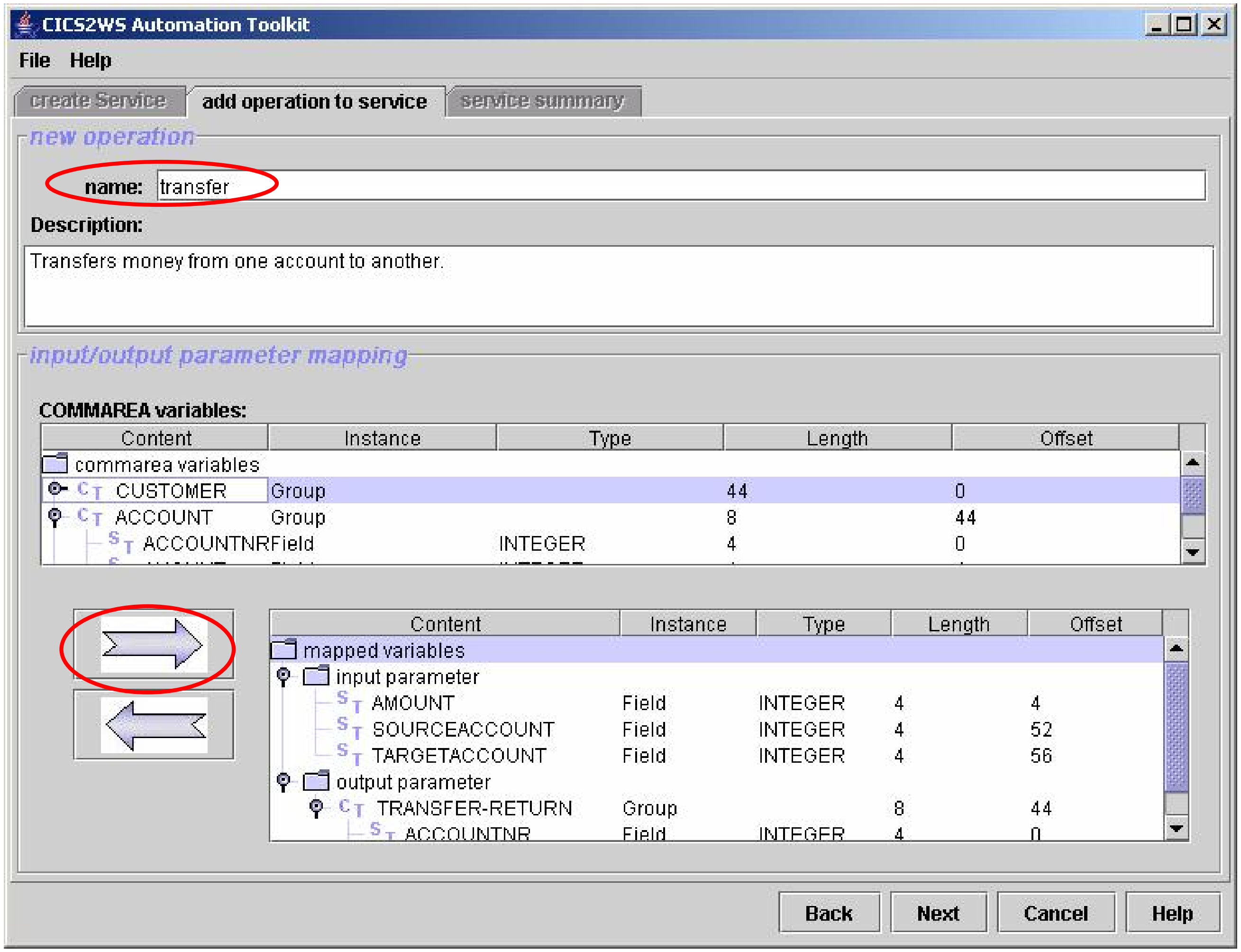
Sample VSE Web Service. Uses CICS program VSEBANK as backend.

program names

Proxy Name: BANKSERV

USER Program Name: VSEBANK

Live Demo – Create a Web Service



new operation

name: transfer

Description:

Transfers money from one account to another.

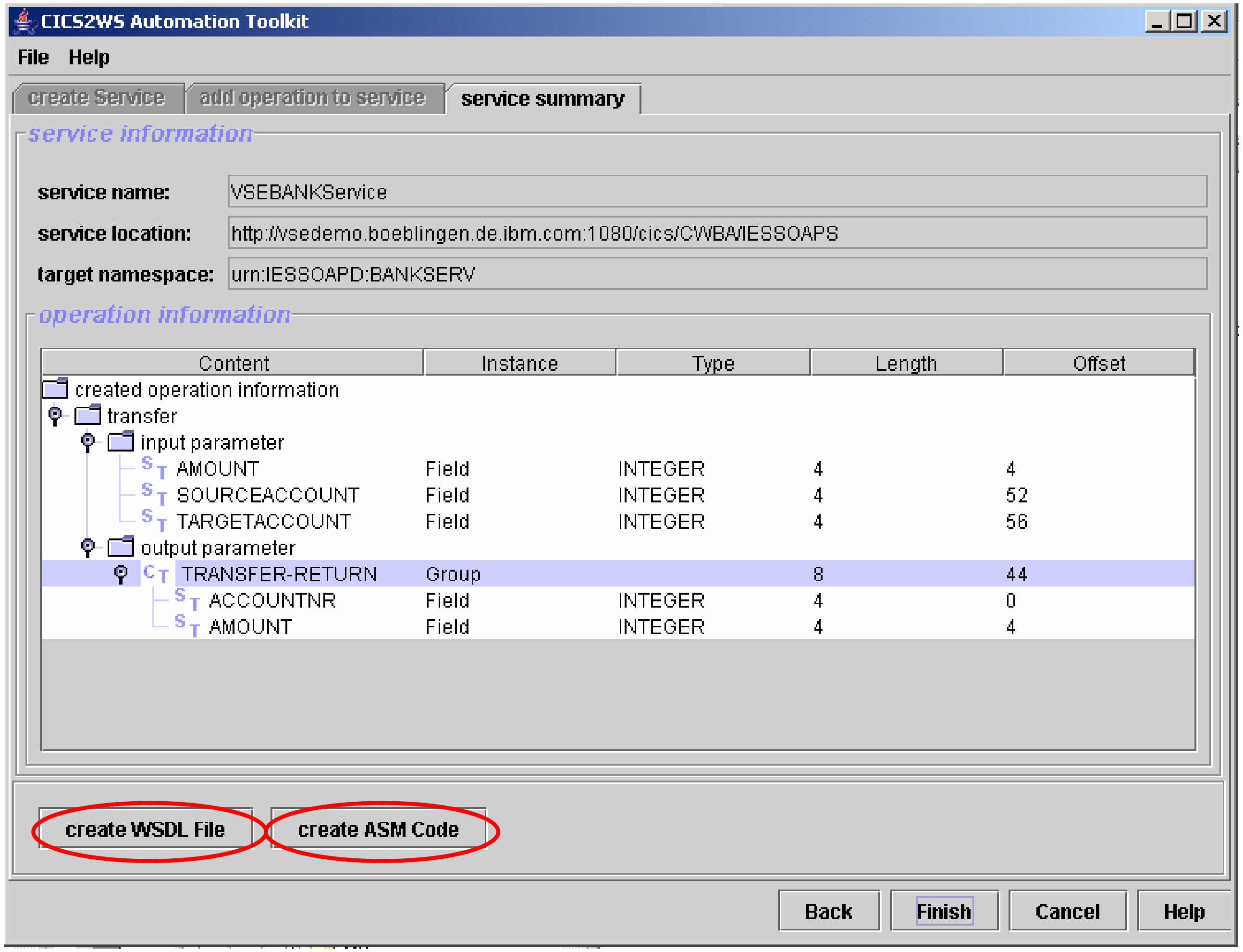
input/output parameter mapping

COMMAREA variables:

Content	Instance	Type	Length	Offset
commarea variables				
CUSTOMER	Group		44	0
ACCOUNT	Group		8	44
ACCOUNTNR	Field	INTEGER	4	0

Content	Instance	Type	Length	Offset
mapped variables				
input parameter				
AMOUNT	Field	INTEGER	4	4
SOURCEACCOUNT	Field	INTEGER	4	52
TARGETACCOUNT	Field	INTEGER	4	56
output parameter				
TRANSFER-RETURN	Group		8	44
ACCOUNTNR	Field	INTEGER	4	0

Live Demo – Create a Web Service



service information

service name: VSEBANKService

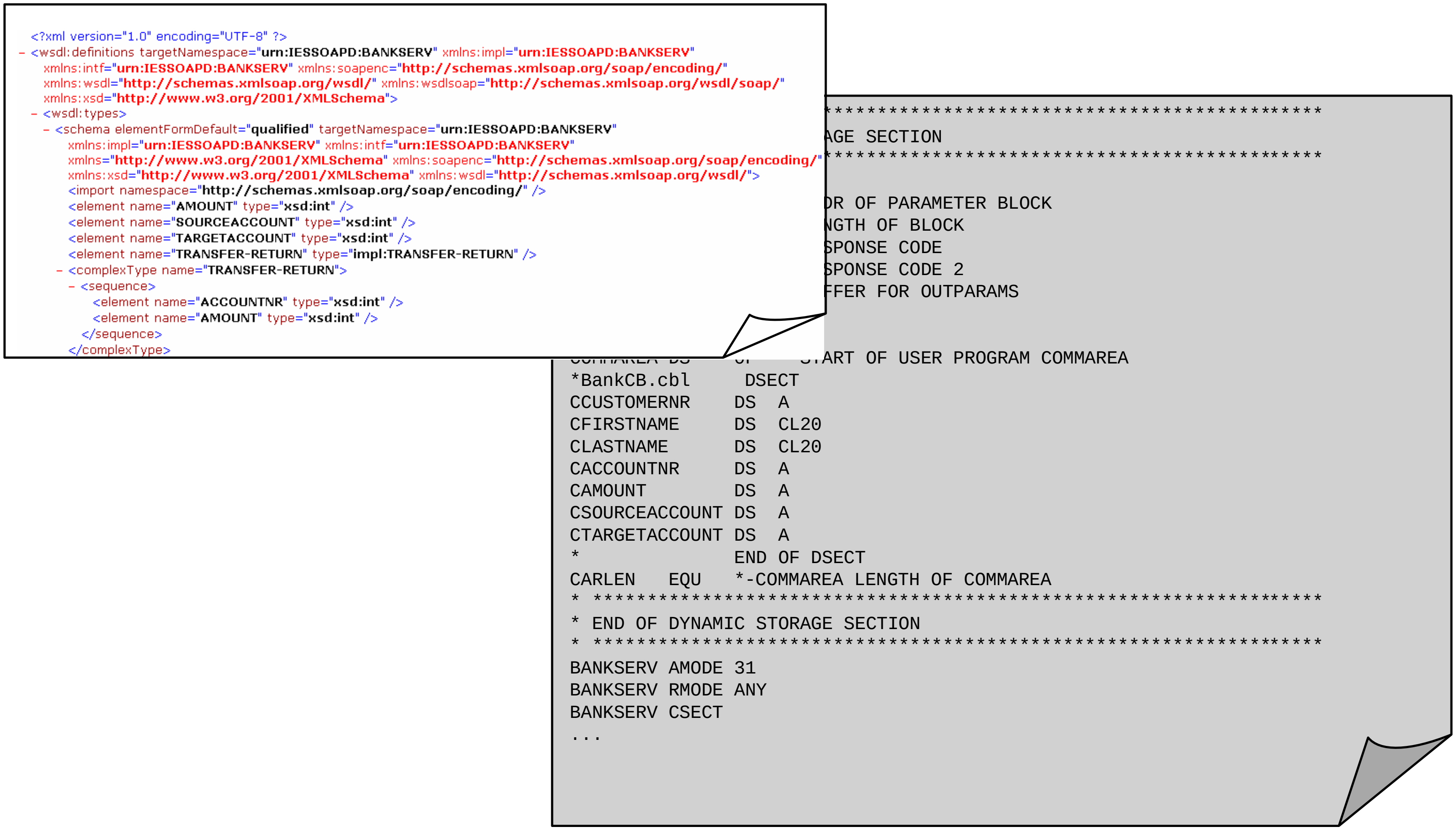
service location: http://vsedemo.boeblingen.de.ibm.com:1080/cics/CWBA/IESSOAPS

target namespace: urn:IESSOAPD:BANKSERV

operation information

Content	Instance	Type	Length	Offset
created operation information				
transfer				
input parameter				
AMOUNT	Field	INTEGER	4	4
SOURCEACCOUNT	Field	INTEGER	4	52
TARGETACCOUNT	Field	INTEGER	4	56
output parameter				
TRANSFER-RETURN	Group		8	44
ACCOUNTNR	Field	INTEGER	4	0
AMOUNT	Field	INTEGER	4	4

Live Demo – Create a Web Service



```
<?xml version="1.0" encoding="UTF-8" ?>
- <wsdl:definitions targetNamespace="urn:IESSOAPD:BANKSERV" xmlns:impl="urn:IESSOAPD:BANKSERV"
    xmlns:intf="urn:IESSOAPD:BANKSERV" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  - <wsdl:types>
    - <schema elementFormDefault="qualified" targetNamespace="urn:IESSOAPD:BANKSERV"
        xmlns:impl="urn:IESSOAPD:BANKSERV" xmlns:intf="urn:IESSOAPD:BANKSERV"
        xmlns="http://www.w3.org/2001/XMLSchema" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
        <import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
        <element name="AMOUNT" type="xsd:int" />
        <element name="SOURCEACCOUNT" type="xsd:int" />
        <element name="TARGETACCOUNT" type="xsd:int" />
        <element name="TRANSFER-RETURN" type="impl:TRANSFER-RETURN" />
      - <complexType name="TRANSFER-RETURN">
        - <sequence>
            <element name="ACCOUNTNR" type="xsd:int" />
            <element name="AMOUNT" type="xsd:int" />
          </sequence>
        </complexType>
```

```
                       *********************************************
                       AGE SECTION
                       *********************************************

                       DR OF PARAMETER BLOCK
                       NGTH OF BLOCK
                       SPONSE CODE
                       SPONSE CODE 2
                       FFER FOR OUTPARAMS

                     START OF USER PROGRAM COMMAREA
*BankCB.cbl     DSECT
CCUSTOMERNR    DS   A
CFIRSTNAME     DS   CL20
CLASTNAME      DS   CL20
CACCOUNTNR     DS   A
CAMOUNT        DS   A
CSOURCEACCOUNT DS   A
CTARGETACCOUNT DS   A
*              END OF DSECT
CARLEN   EQU   *-COMMAREA LENGTH OF COMMAREA
* ******************************************************************
* END OF DYNAMIC STORAGE SECTION
* ******************************************************************
BANKSERV AMODE 31
BANKSERV RMODE ANY
BANKSERV CSECT
...
```

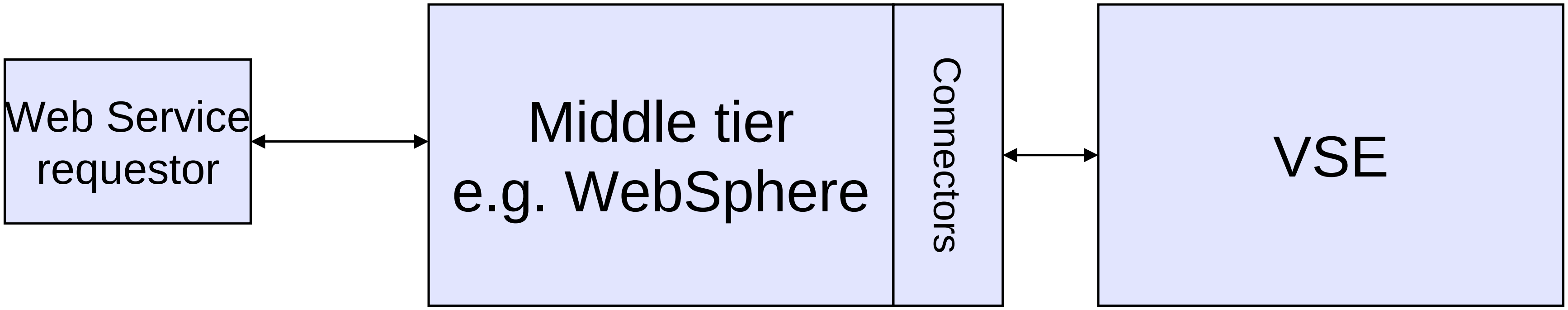
Disadvantages of Web services

- **When should you not use Web Services?**
  - When you have very high performance requirements
    - Communication using SOAP/XML is very time consuming
  - When you transport large amounts of data
    - XML data can get really huge
  - If you require transaction security
    - No 2 phase commit
  - When you want to access the data directly
    - SOAP is program to program communication

- **Similar functions provide**
  - CICS Transaction Gateway
  - MQ Series

Other possibilities to participate into SOA solutions

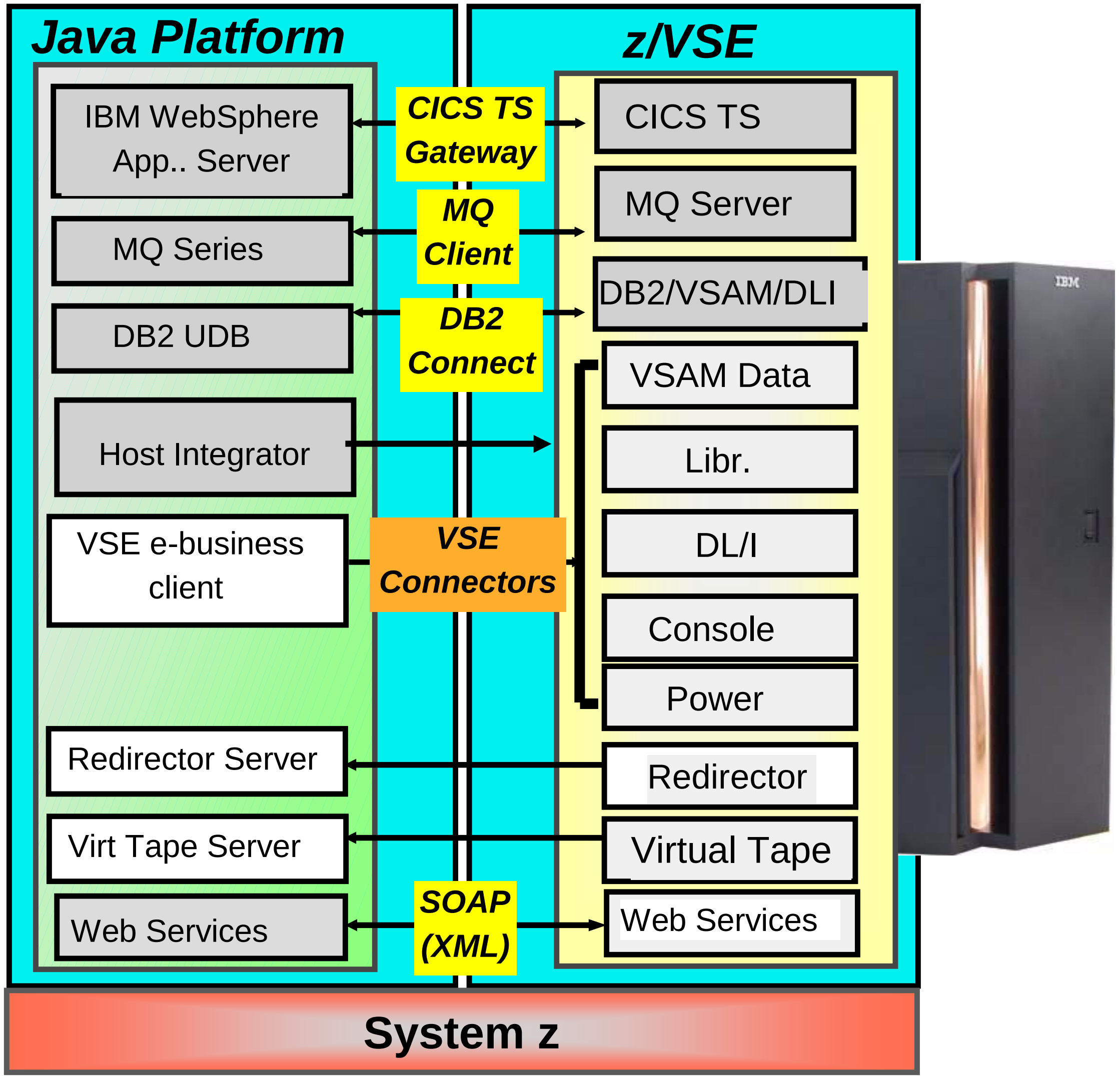
- **2 Tier Solutions**
  - The Web Service requestor or provider runs on VSE itself
- **3 Tier Solutions**
  - The Web Service is implemented on a middle tier system, but accesses VSE data or programs



3 tier SOA solutions

- **Access to VSE using connectors**
  - CICS Transaction Gateway (CICS programs)
  - DB2 Connect (DB2 data)
  - VSE Java-based Connectors (VSAM, DL/I, Jobs, …)
  - MQ Series
- **Middle tier**
  - Using modern technology and products
  - E.g.. WebSphere SOA Products (Enterprise Service Bus, WebSphere Process Server)
  - Can also run on Linux on System z

Middleware relations with VSE



Documentation

Web Services in VSE (from Rich Smrcina)

- http://www.zjournal.com/index.cfm?section=article&aid=281
- http://www.zjournal.com/index.cfm?section=article&aid=320
- Includes COBOL sample code

Web Services

- http://www.ibm.com/servers/eserver/zseries/zvse/documentation/ebusiness.html#soap

What is SOA?

- http://www.ibm.com/developerworks/webservices/newto/
- http://webservices.xml.com/pub/a/ws/2003/09/30/soa.html

z/VSE e-business Connectors, User's Guide (SC33-8231)

Questions ?

?