Agenda Key: 32CG
Session Number: 540191

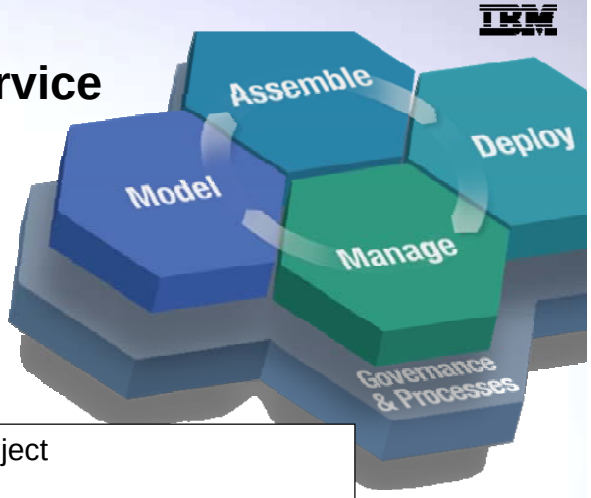
Practical Web Services for RPG

IBM Integrated Web services for i

Dan Hiebert IBM dhiebert@us.ibm.com

© Copyright IBM Corporation, 2009. All Rights Reserved.
This publication may refer to products that are not currently available in your country. IBM makes no commitment to make available any products referred to herein.

Presentation RPG Web Service Business Logic



- Integrated Web services for IBM i – Project
- Intro to SOA/WS on IBM i
- Web Services coding for RPG considerations
- Create, Deploy and Testing an RPG Web service
- Consuming Web services with RPG on IBM i

The New Power Equation

© 2009 IBM Corporation

Web Services Made Easy
An IBM i Project

The New Power Equation

© 2009 IBM Corporation

Web Services Made Easy – An IBM i Project

- IBM Integrated Web Services for i
  - "An Easy Step to starting with SOA on System i"
- Simplify the process of externalizing RPG/COBOL business logic as a service.
- Externalize various RPG/COBOL business tasks as services.
- Abstracts the hidden complexities of Web services for IBM i.
- Provide RPG/COBOL Developer easy to use Web interface, not requiring additional tools or skills

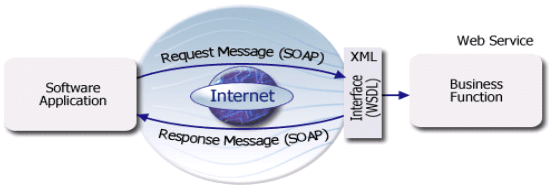
http://www.ibm.com/systems/i/software/iws/

The New Power Equation

© 2009 IBM Corporation

IBM Integrated Web services server for

- Merged the Development Process and Deployment Server
- 2 Steps to Create a Web services server on IBM i
- 7 Steps to Deploy an RPG/COBOL Service
- Built on IBM Integrated Web Application Server for I
  - **Note:** 2 Methods returned for every procedure or program
    - XML - Automated Data (Including Data Structures)
    - Standard Serialized objects.
- Embedded Axis 2 Engine into IBM i (5722SS1)



© 2009 IBM Corporation

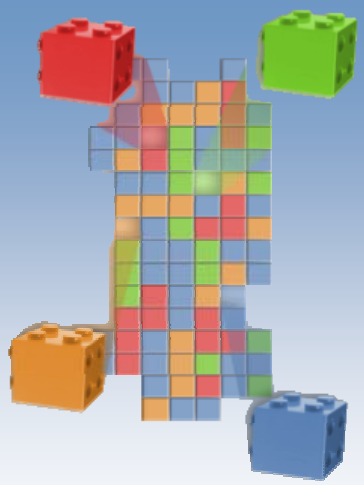
Introduction to Web services and SOA

© 2009 IBM Corporation

What is …..?

… a service?

A **repeatable business task** – e.g., check customer credit; open new account



… service oriented architecture (SOA)?

An IT **architectural style** that supports integrating your business as linked services

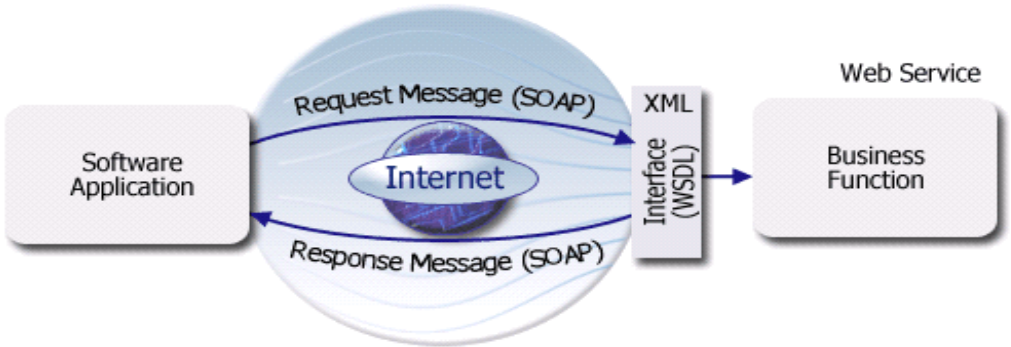
"SOA impacts every aspect of IT and business."

Gartner

© 2009 IBM Corporation

Web Services Overview

Definition: Self-Contained with well-defined interfaces that provide functionality that is accessible over the Internet/Intranet

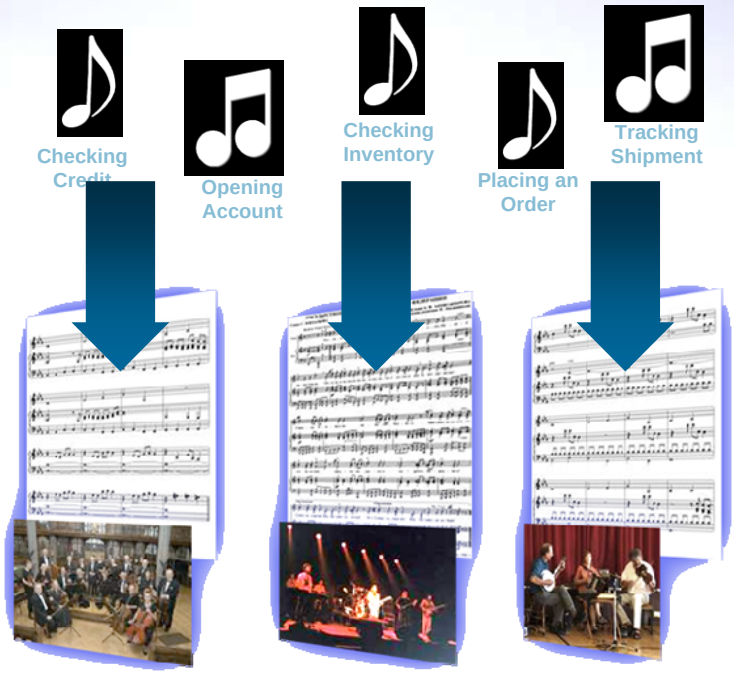


Key Technologies: XML, WSDL, SOAP, UDDI

© 2009 IBM Corporation

SOA Is Like Musical Notes…

Each musical notes represents a business service

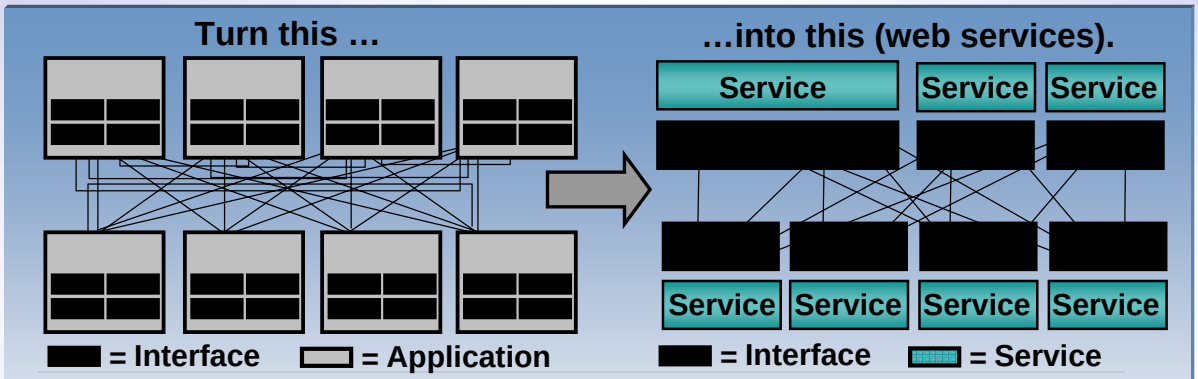


SOA allows for flexible composition of music

The New Power Equation

© 2009 IBM Corporation

Web services are a good start…



- ✓ Rich business abstractions describe the application interface
- ✓ Decouples the interfaces from the business applications
- ✓ The number and complexity of the interfaces is reduced
- ✓ Business applications and their interfaces become reusable

The New Power Equation

© 2009 IBM Corporation

RPG Coding Considerations

The New Power Equation

© 2009 IBM Corporation

RPG Best Practices – Quick Reference

- Use Free Form
- Utilize ILE Techniques
  - Procedures
  - Binding Directories
  - Service Programs
  - Exports – Hints & Tips
- Centralize Declarations
- Expand Naming Conventions
- Write Indicatorless Code
- Use Structured Programming Techniques
- Use Comments
- Avoid Obsolescence

The New Power Equation

© 2009 IBM Corporation

What is a RPG Service?

- Function
  - Get information
  - Perform action
- Properties
  - Encapsulated
  - Reusable
  - Stateless
  - Event driven
  - Loosely coupled
- Modular

© 2009 IBM Corporation

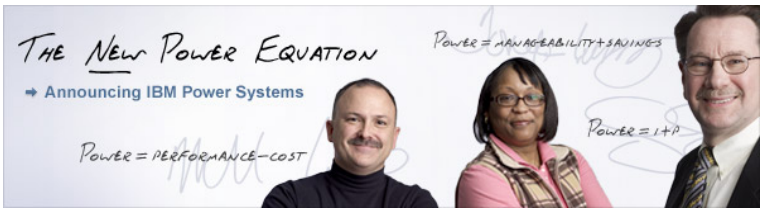
Properties Comparison Review – Service

Service
- Encapsulated
  - Access through interface
- Reusable
  - Write once – use everywhere
- Stateless
  - Information not retained
- Event driven
  - No required order
- Loosely coupled
  - Callable from anywhere

Traditional – Sub Routine
- Global data
  - Access directly
- Reuse by copy
  - Maintain everywhere
- Stateful
  - Information retained in job
- Application driven
  - Fixed order
- Tightly coupled
  - Tied to application

© 2009 IBM Corporation

Create, Deploy and Testing an RPG Web service



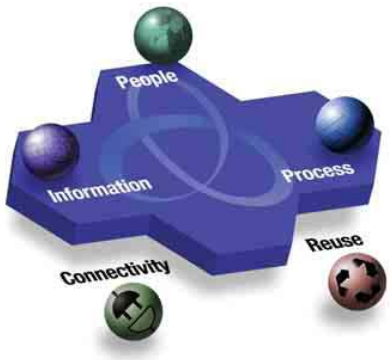
The New Power Equation

© 2009 IBM Corporation

Introducing - IBM Integrated Web Services Server on i

Key Features of IBM I Web Services server

- Integrated into IBM i
- WS-Basic Profile – Compliant
- Developed for RPG/COBOL
- Open Source Technology
- Removes Complexities of Web Services for IBM i Developer
- Easy to Use - Web Admin Interface
- IBM i Web Services Test Client
- Externalizes IBM i Program Objects
- Tracing – WS-Message & Program Objects
- Scripting Support



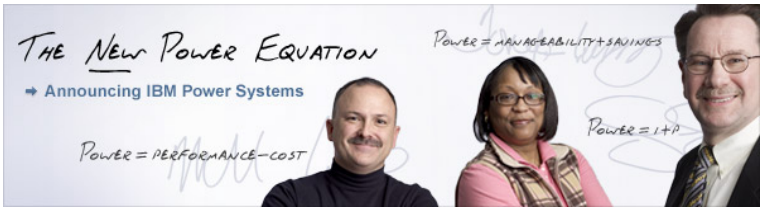
The New Power Equation

© 2009 IBM Corporation

NEW Improvements for YOU! – October/08

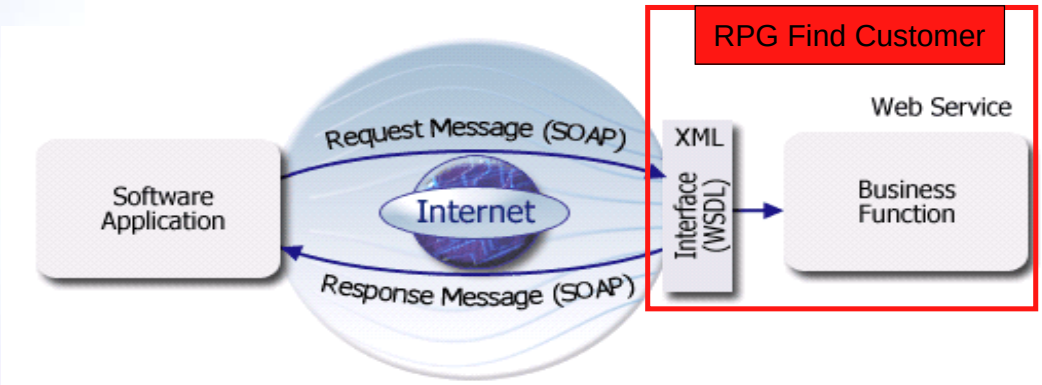
© 2009 IBM Corporation

Embedding Web services information with RPG on IBM i



© 2009 IBM Corporation

Example: RPG Find Customer



© 2009 IBM Corporation

V5R4 - Enablement

```
0001.00 h nomain bnddir('FLGHT400M') PGMINFO(*PCML:*MODULE)
0001.01
0001.06 fCUSTNAME  IF   E           K DISK
0001.07 FCUSTOMER  IF A E           K DISK    RENAME(CUSTR:CUSTOMERR)
0001.08 FCUSTOMRZ  IF   E           K DISK
0001.12
0001.13  /copy nfs405pr
0001.61
0003.55  ******************************************************
0003.56 p FindCustomers   b                   export
0003.57  ******************************************************
0003.58 d FindCustomers   pi
0003.59 d  Position                     64    const
0003.60 d  ListType                      1    const
0003.61 d  CountReq                     10i 0 const
0003.62 d  CountRet                     10i 0
0003.63 d  CustList                           likeds(CustInfo) dim(100)
0003.64 d                                     options(*varsize)
```


© 2009 IBM Corporation

Binding RPG Business Logic to Program/Service Program

- Service Info embedded with RPG or COBOL program objects (PCML)
  - For V6R1 need to recompile specifying:
    CRTRPGMOD PGMINFO(*PCML *MODULE)
    CRTCBLMOD PGMINFO(*PCML *MODULE)
  - For V5R4 – recompile specifying following option in the source:
    - For RPG
      H PGMINFO(*PCML:*MODULE)
    - For COBOL
      PROCESS OPTIONS PGMINFO(PCML MODULE)
- Service Information can alternatively be generated in IFS
- Restrictions:
  - Program objects must be 'Stateless'
  - Further Information - http://www.ibm.com/systems/i/software/iws/

© 2009 IBM Corporation

Externalizing Web services with RPG on IBM i

(Demo) - Simplification and Ease of Use

© 2009 IBM Corporation

Web Admin: Install Web Service

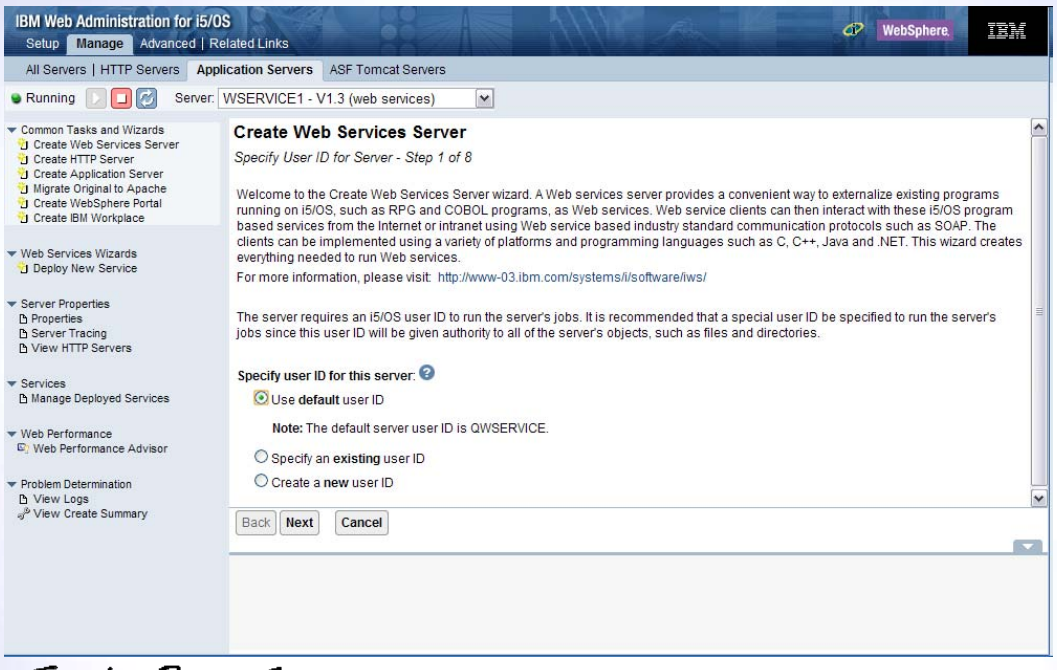
Access Web Admin http://*hostname*:2001/HTTPAdmin

Click on the Create New Web Services Server link

© 2009 IBM Corporation

Web Admin: Install Web Service

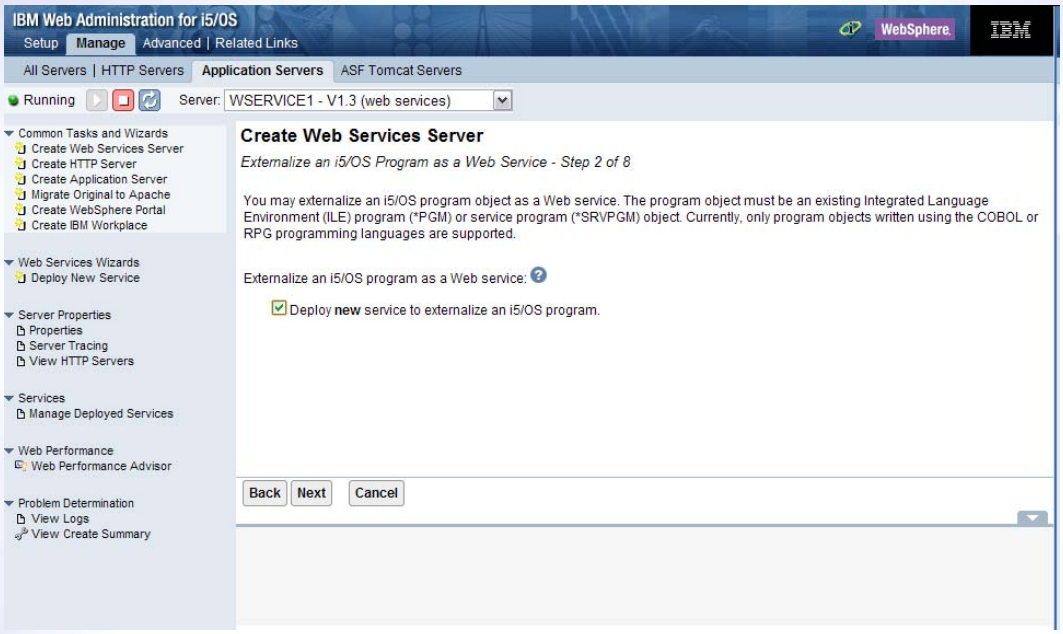
- Step 1: Create the server to run Web services.



© 2009 IBM Corporation

Web Admin: Install Web Service

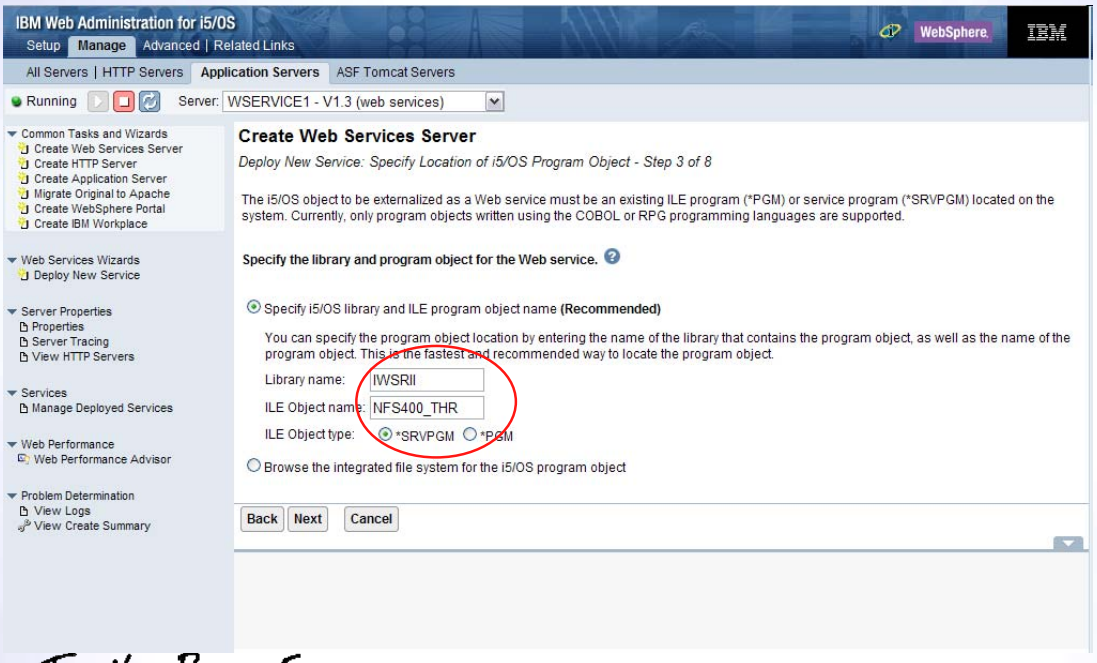
- Step 2: Select to Install a new Web service



© 2009 IBM Corporation

Web Admin: Install Web Service

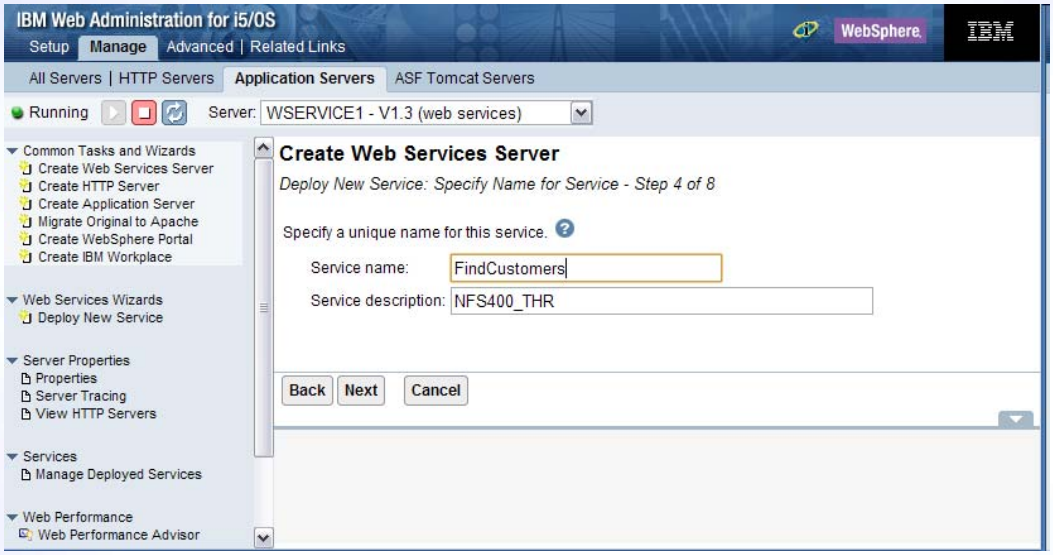
- Step 3: What program contains the services ?



© 2009 IBM Corporation

Web Admin: Install Web Service

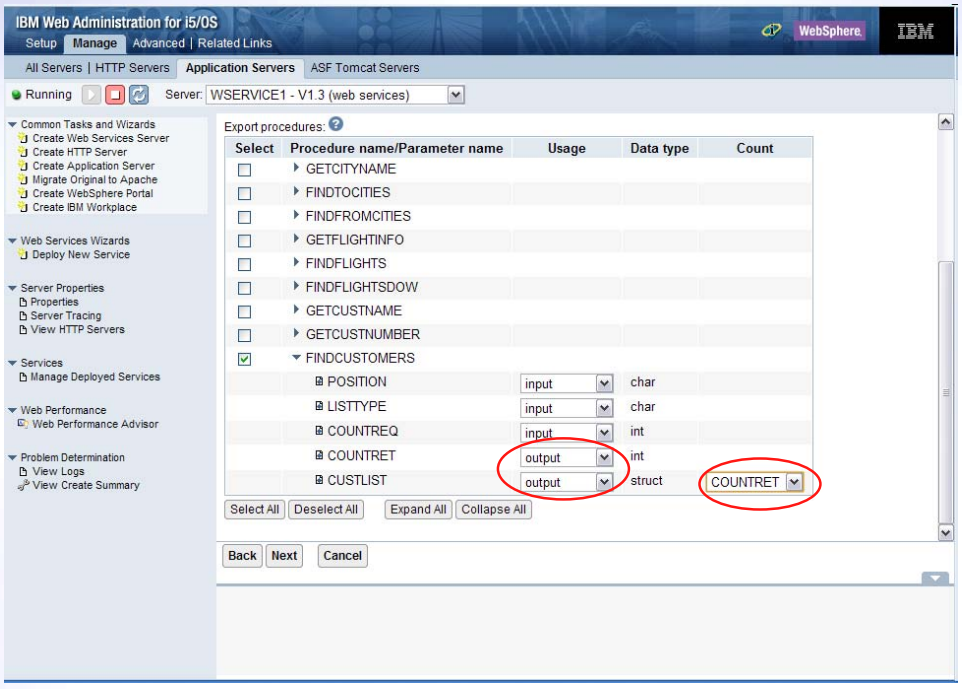
- Step 4: What should we call this new Web service?



© 2009 IBM Corporation

Web Admin: Install Web Service

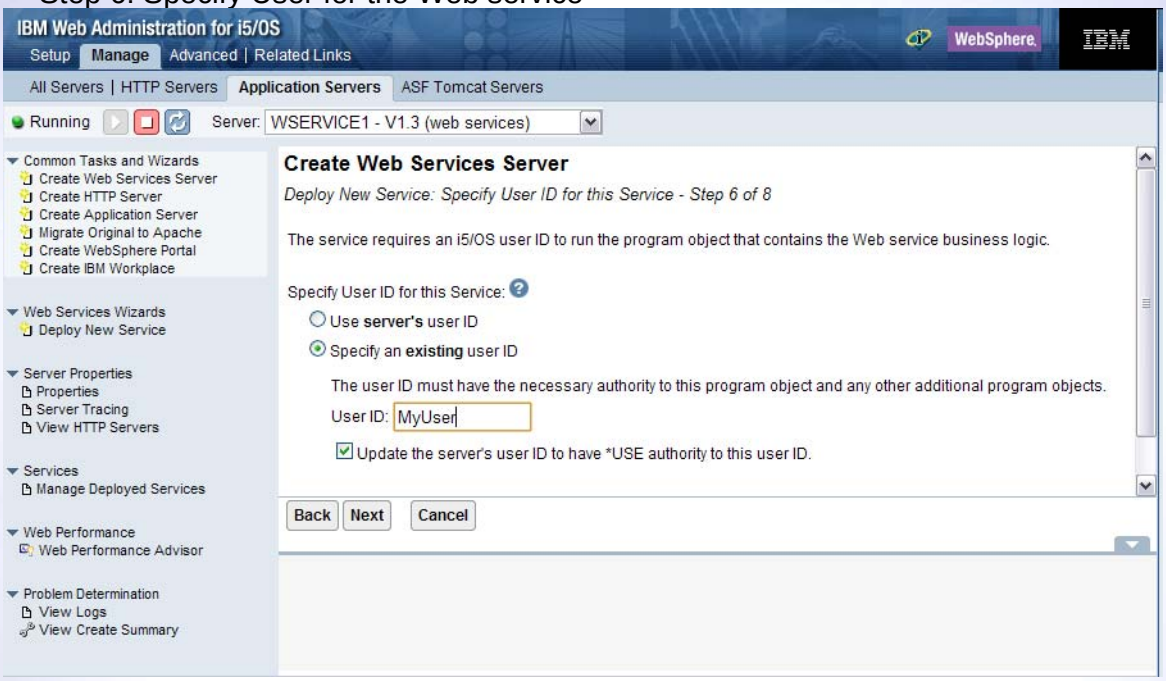
- Step 5: What in the ILE program should be externalized as a Web service?



© 2009 IBM Corporation

Web Admin: Install Web Service

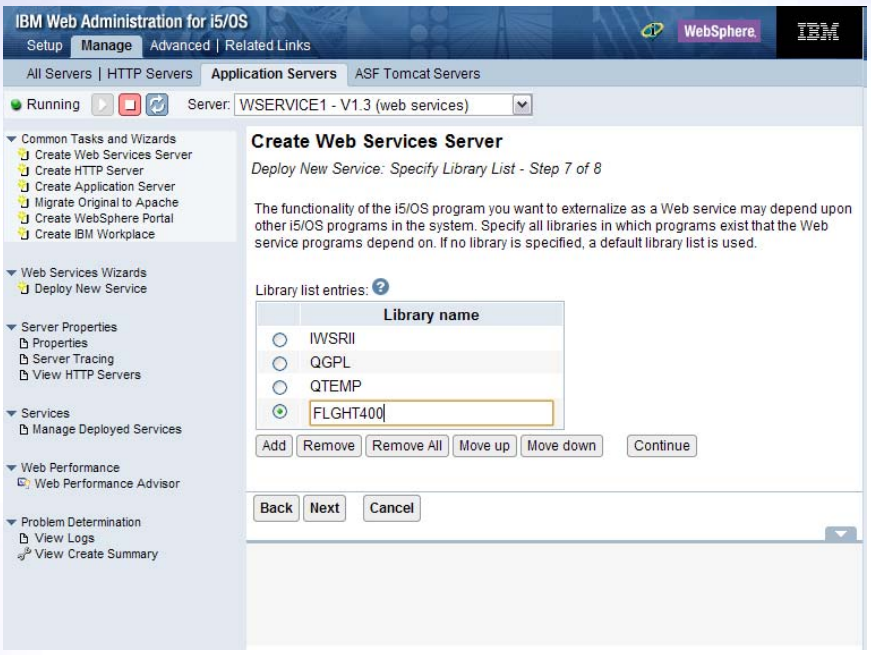
- Step 6: Specify User for the Web service



© 2009 IBM Corporation

Web Admin: Install Web Service

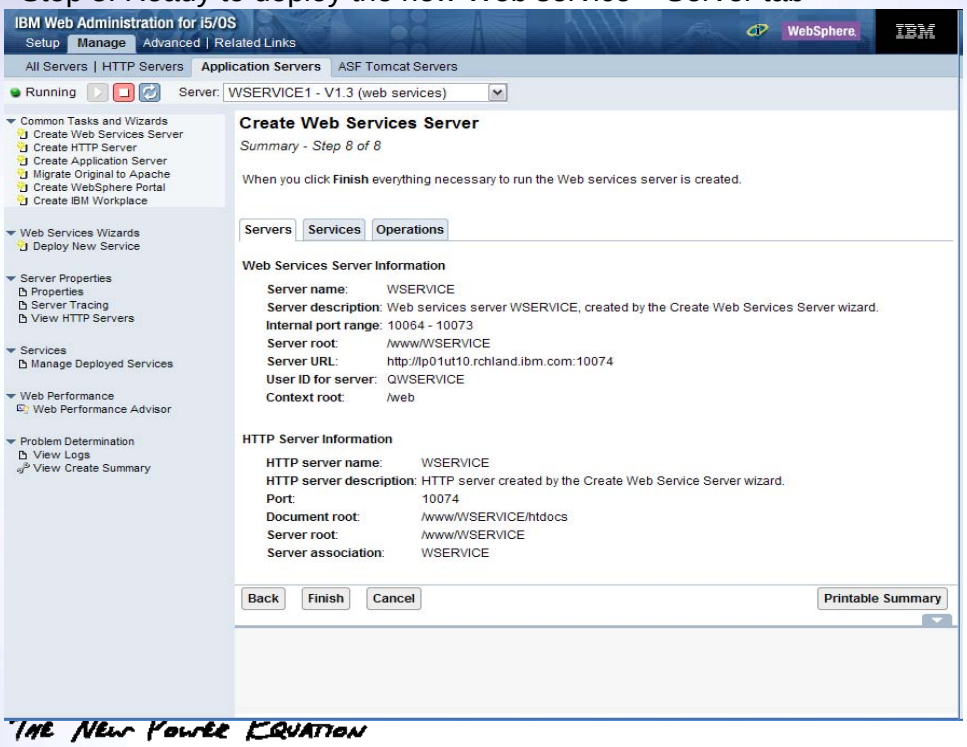
- Step 7: Configure IBM i resources for the Web service



© 2009 IBM Corporation

Web Admin: Install Web Service

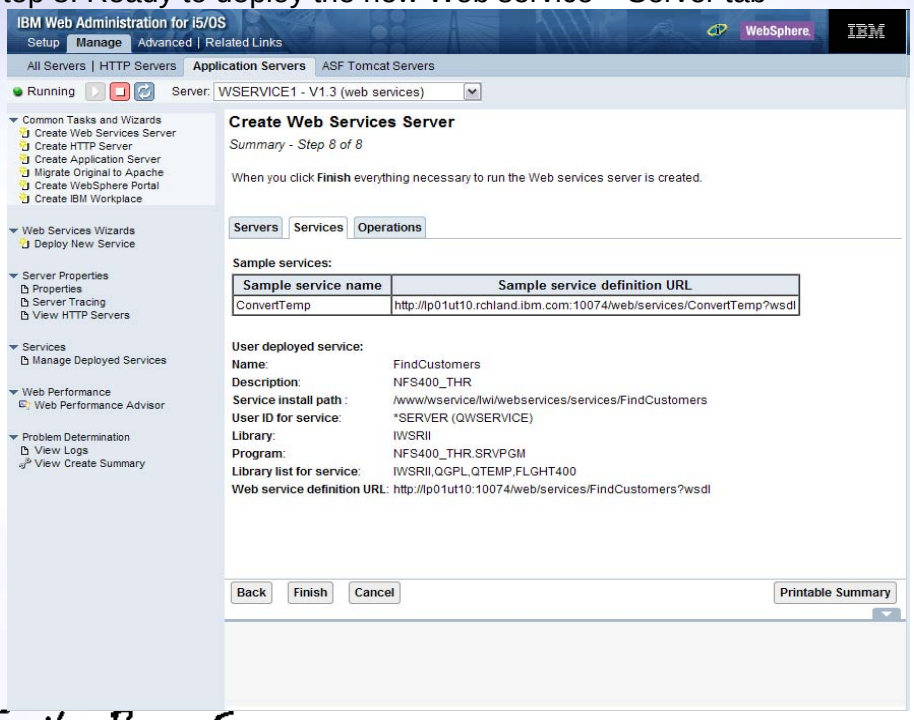
- Step 8: Ready to deploy the new Web service – Server tab



© 2009 IBM Corporation

Web Admin: Install Web Service

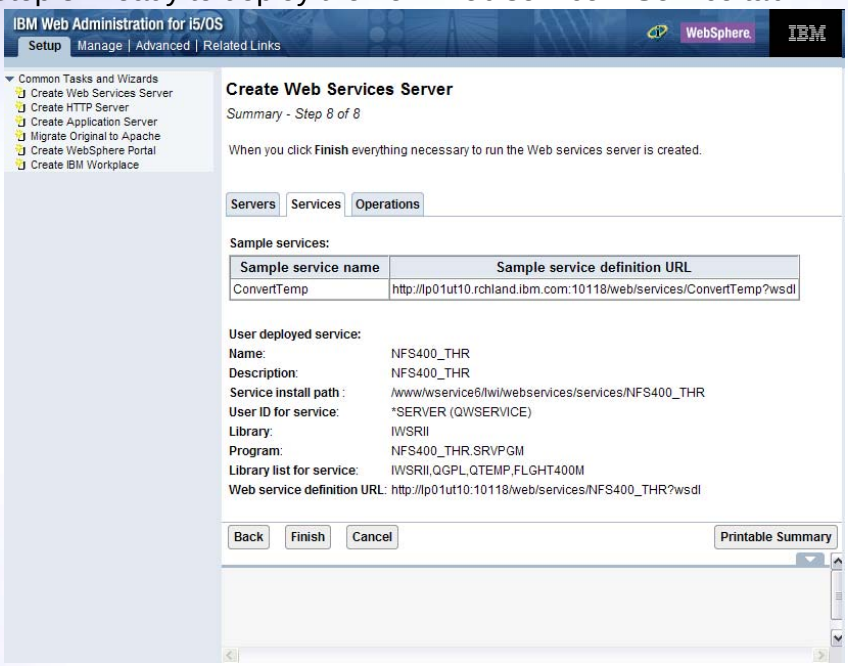
- Step 8: Ready to deploy the new Web service – Server tab



© 2009 IBM Corporation

Web Admin: Install Web Service

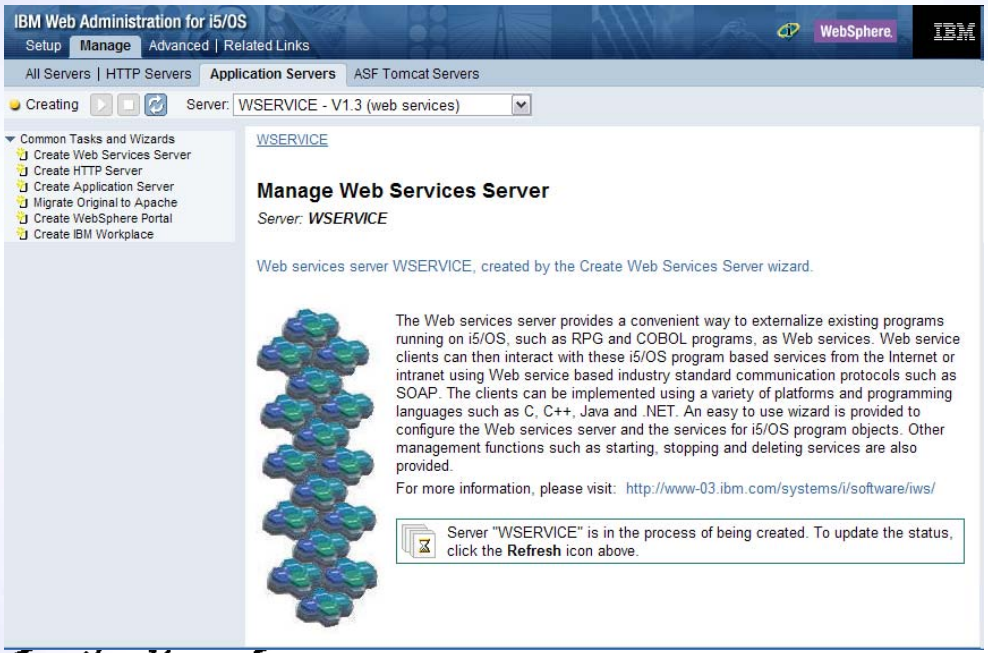
- Step 8: Ready to deploy the new Web service – Service tab



© 2009 IBM Corporation

Web Admin: Install Web Service

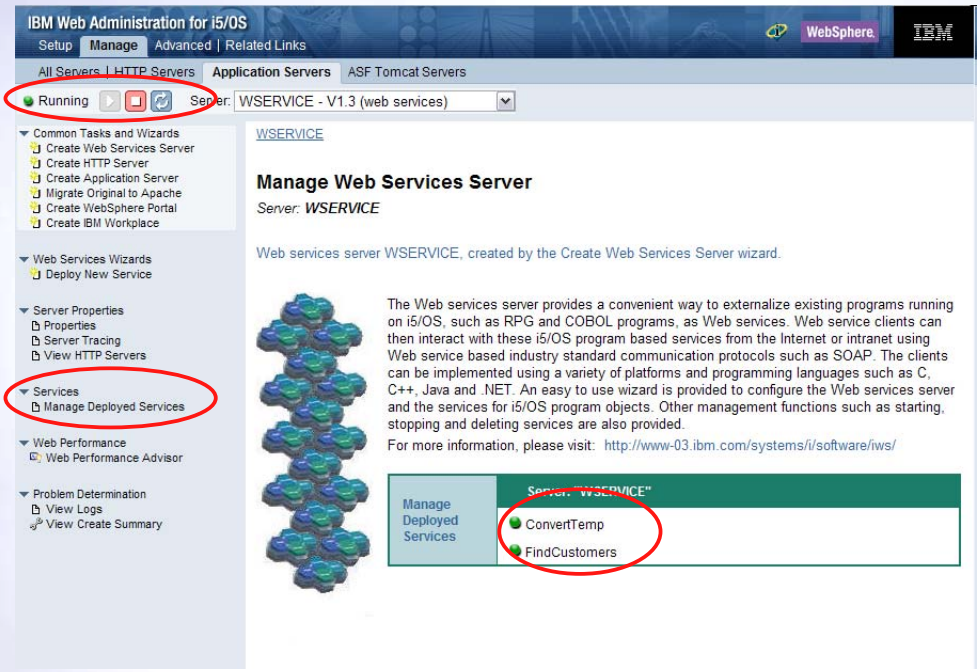
- Create server - after short period of time (seconds) server created and service deployed



© 2009 IBM Corporation

Web Admin: Install Web Service

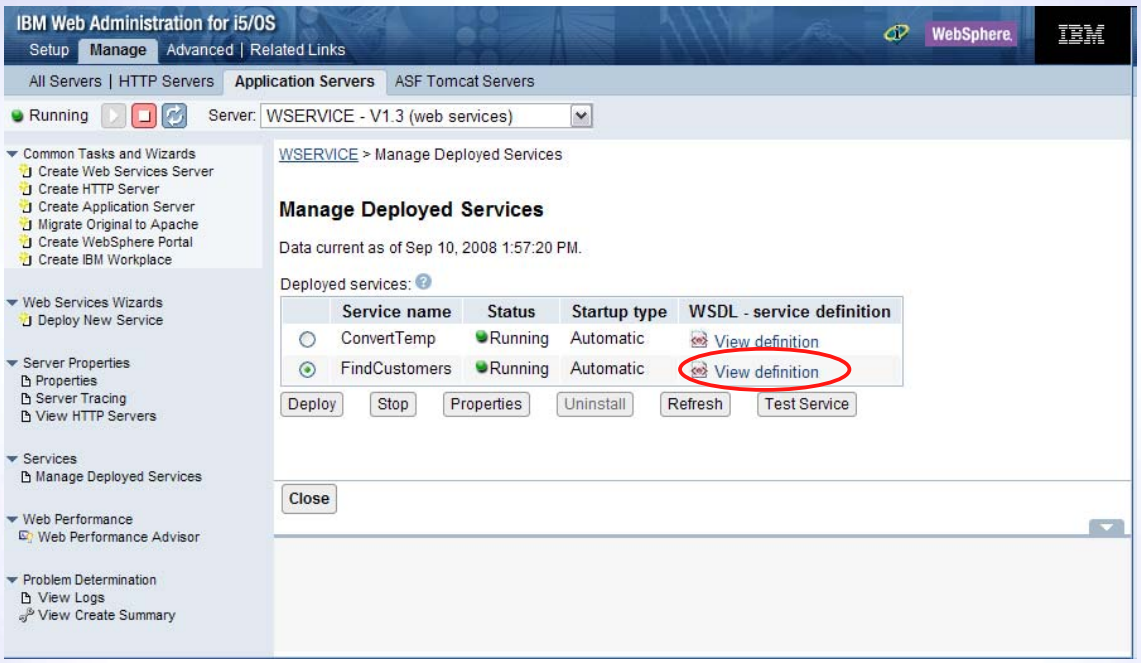
- Once created, the server is started and deployed service started



© 2009 IBM Corporation

Web Admin: Install Web Service

- Manage installed Services – view and install new services to this server



© 2009 IBM Corporation

Web Admin: Install Web Service

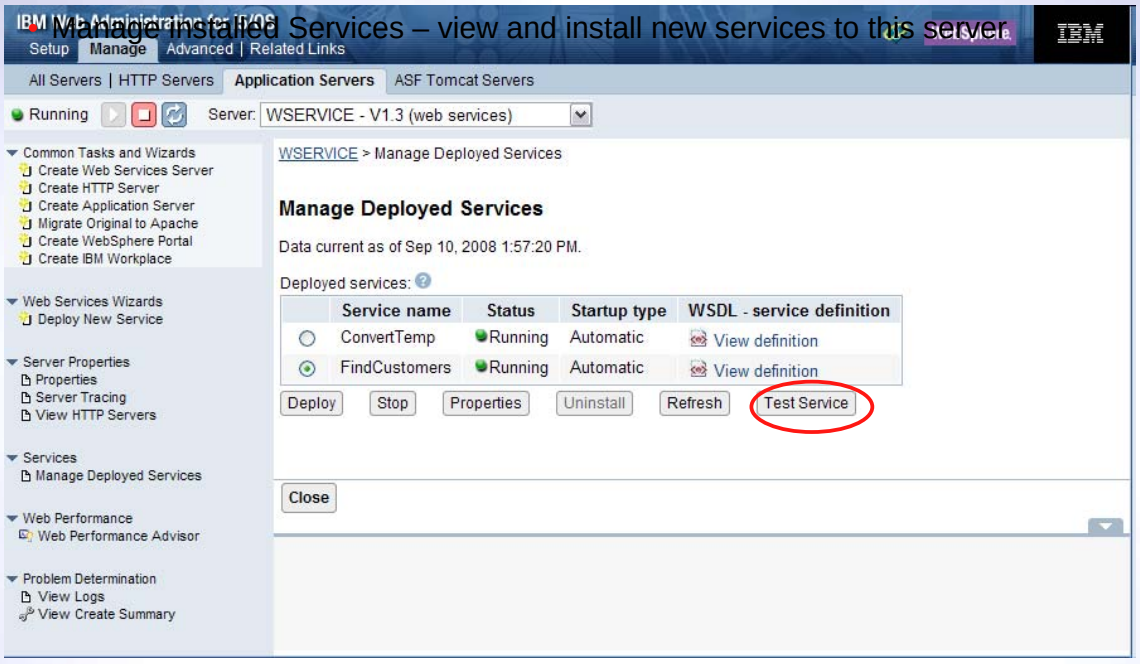
- View the WSDL file

```
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:axis2="http://findcustomers.wsbeans.iseries" xmlns:ns1="http://org.apache.axis2/xsd"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl" xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:ns0="http://findcustomers.wsbeans.iseries/xsd" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
targetNamespace="http://findcustomers.wsbeans.iseries">
    <wsdl:documentation>FindCustomers</wsdl:documentation>
    <wsdl:types>
        <xs:schema xmlns:ns="http://findcustomers.wsbeans.iseries/xsd" attributeFormDefault="qualified"
elementFormDefault="qualified" targetNamespace="http://findcustomers.wsbeans.iseries/xsd">
            <xs:element name="findcustomers">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element minOccurs="0" name="param0" nillable="true"
type="ns:FINDCUSTOMERSInput"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:complexType name="FINDCUSTOMERSInput">
                <xs:sequence>
                    <xs:element minOccurs="0" name="_COUNTREQ" nillable="true" type="xs:int"/>
                    <xs:element minOccurs="0" name="_LISTTYPE" nillable="true" type="xs:string"/>
                    <xs:element minOccurs="0" name="_POSITION" nillable="true" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:element name="findcustomersResponse">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element minOccurs="0" name="return" nillable="true"
type="ns:FINDCUSTOMERSResult"/>
```


© 2009 IBM Corporation

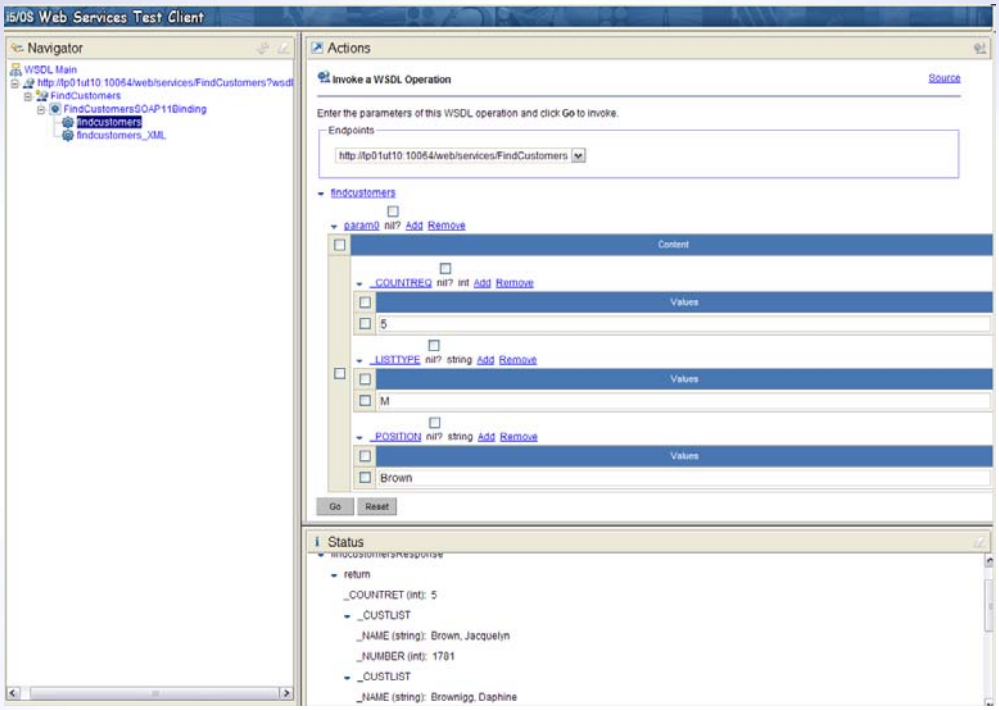
Web Admin: Install Web Service

- Manage Installed Services – view and install new services to this server



© 2009 IBM Corporation

IBM i Web Services Test Client



© 2009 IBM Corporation

Security Discussion

Security Concerns

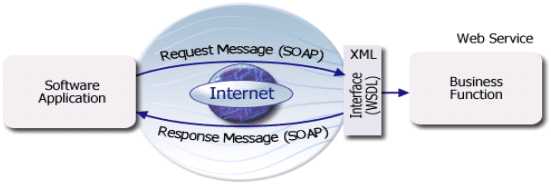
Messages are being sent over the intranet/internet – Are they Secure?

Web Services – Security

- HTTP
- WS-Security

Three Available Security Mechanism's

- HTTP - SSL encryption and authorization (Certificates) (Basic authorization)
- WS-Security (No support on IBM Integrated Web services server for i)
- IBM Datapower Appliances – (Pretty cool utility)



© 2009 IBM Corporation

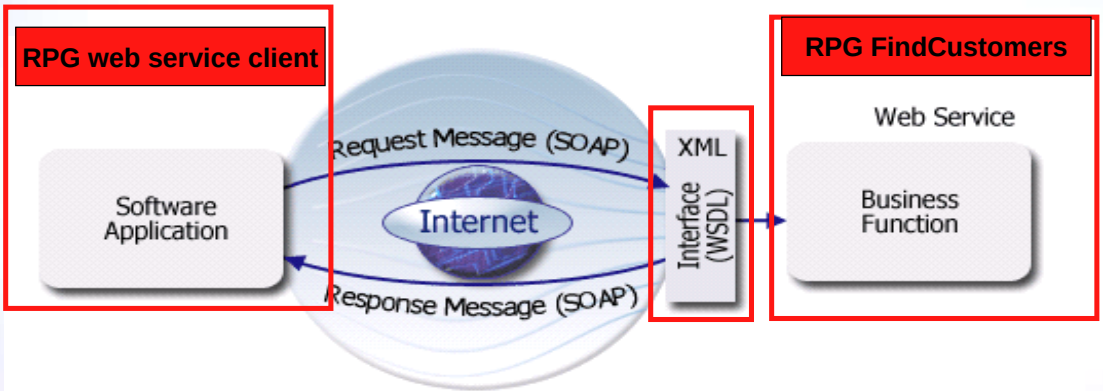
Consuming Web services with RPG on IBM i
The Unlimited Potential

The New Power Equation

© 2009 IBM Corporation

Web Services – a client view

Definition: Self-Contained with well-defined interfaces that provide functionality that is accessible over the Internet/Intranet



Key Technologies: XML, WSDL, SOAP, UDDI

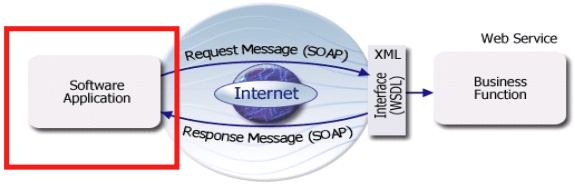
The New Power Equation

© 2009 IBM Corporation

Integrated Web services client for IBM i

- **Natural for ILE RPG/COBOL Developer**
- **4 Step Static Development Process**
  1. **Use WSDL to generate Web service proxy code in C**
  2. **Build RPG stub code from C proxy code**
  3. **Compile/Bind RPG & Web service stub code**
  4. **Invoke RPG/COBOL Web service client**

http://www.ibm.com/systems/i/software/iws/



© 2009 IBM Corporation

Web Services Client for ILE

- Overview
  - Based on Apache AXIS C++ Version 1.5+
  - Consists
    - Tools
      - Convert service's WSDL to C/C++ APIs
    - SOAP client
      - ILE Service Program
  - Supported today
    - Packaged with 5733-XT1
    - V5R1, V5R2, V5R3, V5R4
  - Availability
    - 4Q/07 – IBM i SS1 Option 3
    - V5R4 and later
- Supports
  - C, C++, RPG, COBOL
  - Web Services Description Language (WSDL) - document literal only
  - Web Services Invocation (WSI) 1.1 basic profile compliance
  - Secure Sockets Layer (SSL)

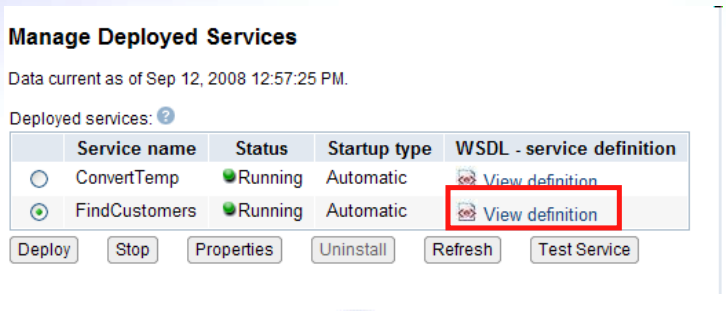
© 2009 IBM Corporation

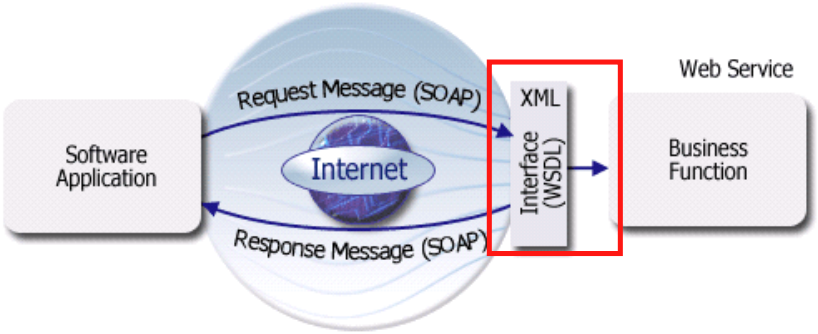
Web Service Deployment Review

```
0001.00 h nomain bnddir('FLGHT400M') PGMINFO(*PCML:*MODULE)
0001.01
0001.06 fCUSTNAME  IF   E           K DISK
0001.07 FCUSTOMER  IF A E           K DISK    RENAME(CUSTR:CUSTOMERR)
0001.08 FCUSTOMRZ  IF   E           K DISK
0001.12
0001.13  /copy nfs405pr
0001.61
0003.55 ********************************************
0003.56 p FindCustomers   b                   export
0003.57 ********************************************
0003.58 d FindCustomers   pi
0003.59 d  Position                     64    const
0003.60 d  ListType                      1    const
0003.61 d  CountReq                     10i 0 const
0003.62 d  CountRet                     10i 0
0003.63 d  CustList                           likeds(CustInfo) dim(100)
0003.64 d                                     options(*varsize)
```


© 2009 IBM Corporation

Where to find my WSDL





© 2009 IBM Corporation

FindCustomers.WSDL

```
<wsdl:definitions xmlns:ws...eries" xmlns:ns1="http://org.apache.axis2/xsd"
xmlns:wsaw="http://www.w3...lns:ns0="http://findcustomers.wsbeans.iseries/xsd"
xmlns:xs="http://www.w3.o...http://schemas.xmlsoap.org/wsdl/mime/"
xmlns:soap12="http://schen...s">
    <wsdl:documentation>Fi...
    <wsdl:types>
        <xs:schema xmlns:ns="http://findcustomers.wsbeans.iseries/xsd" attributeFormDefault="qualified" elementFormDefault="qualified"
targetNamespace="http://findcustomers.wsbeans.iseries/xsd">
            <xs:element name="findcustomers">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element minOccurs="0" name="param0" nillable="true" type="ns:FINDCUSTOMERSInput"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:complexType name="FINDCUSTOMERSInput">
                <xs:sequence>
                    <xs:element minOccurs="0" name="_COUNTREQ" nillable="true" type="xs:int"/>
                    <xs:element minOccurs="0" name="_LISTTYPE" nillable="true" type="xs:string"/>
                    <xs:element minOccurs="0" name="_POSITION" nillable="true" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:element name="findcustomersResponse">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element minOccurs="0" name="return" nillable="true" type="ns:FINDCUSTOMERSResult"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:complexType name="FINDCUSTOMERSResult">
                <xs:sequence>
                    <xs:element minOccurs="0" name="_COUNTRET" nillable="true" type="xs:int"/>
                    <xs:element maxOccurs="unbounded" minOccurs="0" name="_CUSTLIST" nillable="true" type="ns:CUSTINFO"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="CUSTINFO">
                <xs:sequence>
                    <xs:element minOccurs="0" name="_NAME" nillable="true" type="xs:string"/>
                    <xs:element minOccurs="0" name="_NUMBER" nillable="true" type="xs:int"/>
                </xs:sequence>
            </xs:complexType>
            <xs:element name="findcustomers_XML">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element minOccurs="0" name="param0" nillable="true" type="ns:FINDCUSTOMERSInput"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element name="findcustomers_XMLResponse">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element minOccurs="0" name="return" nillable="true" type="xs:string"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:schema>
    </wsdl:types>
```

© 2009 IBM Corporation

```
    <wsdl:message name="findcustomers_XMLRequest">
        <wsdl:part name="parameters" element="ns0:findcustomers_XML"/>
    </wsdl:message>
    <wsdl:message name="findcustomers_XMLResponse">
        <wsdl:part name="parameters" element="ns0:findcustomers_XMLResponse"/>
    </wsdl:message>
    <wsdl:message name="findcustomersRequest">
        <wsdl:part name="parameters" element="ns0:findcustomers"/>
    </wsdl:message>
    <wsdl:message name="findcustomersResponse">
        <wsdl:part name="parameters" element="ns0:findcustomersResponse"/>
    </wsdl:message>
    <wsdl:portType name="FindCustomersPortType">
        <wsdl:operation name="findcustomers_XML">
            <wsdl:input message="axis2:findcustomers_XMLRequest" wsaw:Action="urn:findcustomers_XML"/>
            <wsdl:output message="axis2:findcustomers_XMLResponse" wsaw:Action="urn:findcustomers_XMLResponse"/>
        </wsdl:operation>
        <wsdl:operation name="findcustomers">
            <wsdl:input message="axis2:findcustomersRequest" wsaw:Action="urn:findcustomers"/>
            <wsdl:output message="axis2:findcustomersResponse" wsaw:Action="urn:findcustomersResponse"/>
        </wsdl:operation>
    </wsdl:portType>
    <wsdl:binding name="FindCustomersSOAP11Binding" type="axis2:FindCustomersPortType">
        <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
        <wsdl:operation name="findcustomers_XML">
            <soap:operation soapAction="urn:findcustomers_XML" style="document"/>
            <wsdl:input>
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output>
                <soap:body use="literal"/>
            </wsdl:output>
        </wsdl:operation>
        <wsdl:operation name="findcustomers">
            <soap:operation soapAction="urn:findcustomers" style="document"/>
            <wsdl:input>
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output>
                <soap:body use="literal"/>
            </wsdl:output>
        </wsdl:operation>
    </wsdl:binding>
    <wsdl:service name="FindCustomers">
        <wsdl:port name="FindCustomersSOAP11port_http" binding="axis2:FindCustomersSOAP11Binding">
            <soap:address location="http://lp01ut10:10064/web/services/FindCustomers"/>
        </wsdl:port>
    </wsdl:service>
</wsdl:definitions>
```


© 2009 IBM Corporation

FindCustomers.WSDL

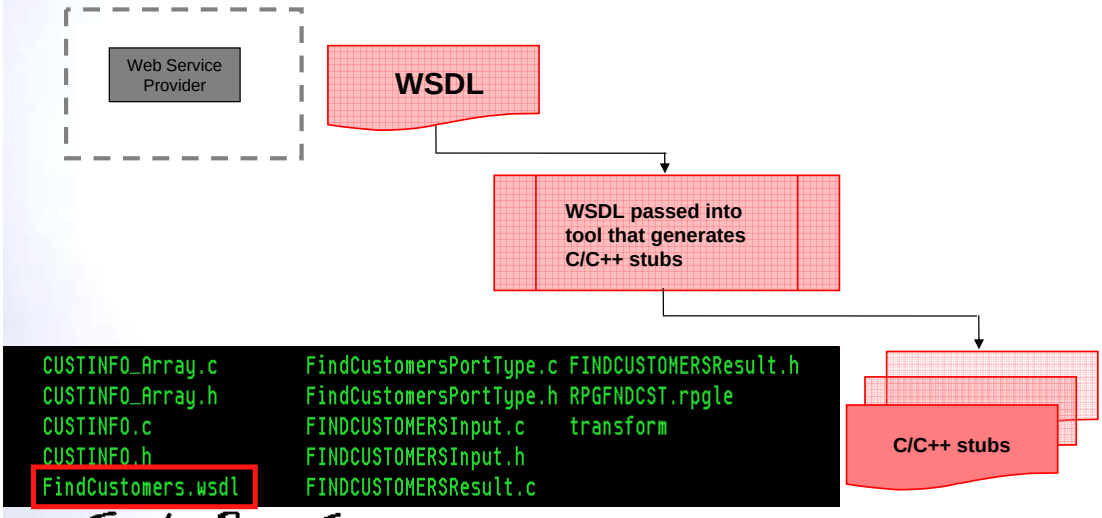
```
<wsdl:definitions xmlns:ws...eries" xmlns:ns1="http://org.apache.axis2/xsd"
xmlns:wsaw="http://www.w3...lns:ns0="http://findcustomers.wsbeans.iseries/xsd"
xmlns:xs="http://www.w3.o...ttp://schemas.xmlsoap.org/wsdl/mime/"
xmlns:soap12="http://schen...s">
    <wsdl:documentation>Fi...
    <wsdl:types>
      <xs:schema xmlns:ns="http://findcustomers.wsbeans.iseries/xsd" attributeFormDefault="qualified" elementFormDefault="qualified"
targetNamespace="http://findcustomers.wsbeans.iseries/xsd">
        <xs:element name="findcustomers">
            <xs:complexType>
                <xs:sequence>
                    <xs:element minOccurs="0" name="param0" nillable="true" type="ns:FINDCUSTOMERSInput"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:complexType name="FINDCUSTOMERSInput">
            <xs:sequence>
                <xs:element minOccurs="0" name="_COUNTREQ" nillable="true" type="xs:int"/>
                <xs:element minOccurs="0" name="_LISTTYPE" nillable="true" type="xs:string"/>
                <xs:element minOccurs="0" name="_POSITION" nillable="true" type="xs:string"/>
            </xs:sequence>
        </xs:complexType>
        <xs:element name="findcustomersResponse">
            <xs:complexType>
                <xs:sequence>
                    <xs:element minOccurs="0" name="return" nillable="true" type="ns:FINDCUSTOMERSResult"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:complexType name="FINDCUSTOMERSResult">
            <xs:sequence>
                <xs:element minOccurs="0" name="_COUNTRET" nillable="true" type="xs:int"/>
                <xs:element maxOccurs="unbounded" minOccurs="0" name="_CUSTLIST" nillable="true" type="ns:CUSTINFO"/>
            </xs:sequence>
        </xs:complexType>
        <xs:complexType name="CUSTINFO">
            <xs:sequence>
                <xs:element minOccurs="0" name="_NAME" nillable="true" type="xs:string"/>
                <xs:element minOccurs="0" name="_NUMBER" nillable="true" type="xs:int"/>
            </xs:sequence>
        </xs:complexType>
        <xs:element name="findcustomers_XML">
            <xs:complexType>
                <xs:sequence>
                    <xs:element minOccurs="0" name="param0" nillable="true" type="ns:FINDCUSTOMERSInput"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="findcustomers_XMLResponse">
            <xs:complexType>
                <xs:sequence>
                    <xs:element minOccurs="0" name="return" nillable="true" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
      </xs:schema>
    </wsdl:types>
```

© 2009 IBM Corporation

Web Services Client for ILE

Step 1: Stub Generation – Creating a Web Services Proxy

- Developer generates stubs using:
  - Java tools (wsdl2ws.jar)
  - Qshell script - ***wsdl2ws.sh -lc STOCKQ.wsdl***



The New Power Equation

© 2009 IBM Corporation

FindCustomersPortType.h

```
29
30 /* ********************************************************************* */
31 /* --- Functions relating to web service client proxy                --- */
32 /* ********************************************************************* */
33
34 extern AXISCHANDLE get_FindCustomersPortType_stub(const char* pchEndPointUri);
35 extern void destroy_FindCustomersPortType_stub(AXISCHANDLE pStub);
36 extern int get_FindCustomersPortType_Status(AXISCHANDLE pStub);
37 extern void set_FindCustomersPortType_ExceptionHandler(AXISCHANDLE pStub, AXIS_EXCEPTION_HANDLER_FUNCT fp);
38
39 /* ********************************************************************* */
40 /* --- Functions relating to web service methods                     --- */
41 /* ********************************************************************* */
42
43 extern xsdc__string findcustomers_XML(AXISCHANDLE pStub, FINDCUSTOMERSInput* Value0);
44 extern FINDCUSTOMERSResult* findcustomers(AXISCHANDLE pStub, FINDCUSTOMERSInput* Value0);
45
```


© 2009 IBM Corporation

FINDCUSTOMERSInput.h

```
40
41 /* ********************************************************************* */
42 /* --- Custom type                                                   --- */
43 /* ********************************************************************* */
44
45 typedef struct FINDCUSTOMERSInputTag {
46     xsdc__int* _COUNTREQ;
47     xsdc__string _LISTTYPE;
48     xsdc__string _POSITION;
49 } FINDCUSTOMERSInput;
50
51
52 /* ********************************************************************* */
53 /* --- Functions to create/delete, serialize/deserialize custom type --- */
54 /* ********************************************************************* */
55
56 extern int Axis_DeSerialize_FINDCUSTOMERSInput(FINDCUSTOMERSInput* param, AXISCHANDLE pDZ);
57 extern void* Axis_Create_FINDCUSTOMERSInput(int nSize);
58 extern void Axis_Delete_FINDCUSTOMERSInput(FINDCUSTOMERSInput* param, int nSize);
59 extern int Axis_Serialize_FINDCUSTOMERSInput(FINDCUSTOMERSInput* param, AXISCHANDLE pSZ, AxiscBool bArray);
60
```


© 2009 IBM Corporation

```
000002     H DFTNAME(GETCINFORP) bnddir('QC2LE')
000004      *
000005      * Quick code declaration to make use of C printf()
000006      *--------------------------------------------------------------
000007     D printf          pr            10i 0 extproc('printf')
000008     D fmt                            *   value options(*string)
000009     D extra                          *   value options(*string:*nopass)
000011      *
000012      * Prototypes for the service interface and operations
000013      * obtained from FindCustomersPortType.h  file.
000014      *--------------------------------------------------------------------------
000015     DgetStub          PR             *   ExtProc('get_FindCustomers+
000016     D                                    PortType_stub')
000017     D pEndpoint                      *   Value
000018      *
000019     DFindCustomers    PR             *   ExtProc('findcustomers_XML')
000020     D pWsStub                        *   Value
000021     D pTempInF                       *   Value
000022      *
000023     DdestroyStub      PR                 ExtProc('destroy_FindCustomers+
000024     D                                    PortType_stub')
000025     D pWsStub                        *   Value
000027      *--------------------------------------------------------------------------
000028      * Prototype for function to free dynamic storage obtained from
000029      * FINDCUSTOMERSResult.h file.
000030      *--------------------------------------------------------------------------
000031     DdestroyResult    PR                 ExtProc('Axis_Delete+
000032     D                                    _FINDCUSTOMERSResult')
000033     D pResult                        *   Value
000034     D size                          9B 0 Value
000035     D
```

1

2

3

© 2009 IBM Corporation

```
000041      *--------------------------------------------------------------------------
000042      * Input data structure used by the orderinventory operation.  The
000043      * structure layout is obtained from FINDCUSTOMERSInput.h file.
000044      *--------------------------------------------------------------------------
000045     D Input           DS
000046     D   CountReq                     *
000047     D   ListType                     *
000048     D   Position                     *
000057      *--------------------------------------------------------------------------
000058      * Miscellaneous declarations.
000059      *--------------------------------------------------------------------------
000060     D WsStubP         S              *
000062     D CustomersP      S              *
000063     D Endpoint        S            100A
000065     D tempListType    S             25A
000066     D tempPosition    S             25A
000067     D tempCountReq    S             10I 0
000068     D customersXML    S           9999A   based(CustomersP)
000072
000073     D newl            S              2A
000074     D null            S              2A
```

4

5

© 2009 IBM Corporation

```
000088      *---------------------------------------------------------------
000089      * Program entry point.
000090      *---------------------------------------------------------------
000091
000092      *---------------------------------------------------------------
000093      * Web service logic. The code will attempt to invoke a Web
000094      * service that returns a customer list.
000095      *---------------------------------------------------------------
000096      /FREE
000097
000098          newl = x'15';
000099          null = x'00';
000105
000106          // Get a Web service stub. The host and port for the endpoint may need
000107          // to be changed to match host and port of Web service. You can pass
000108          // *NULL to getStub() if the endpoint in the WSDL file is correct.
000109          Endpoint = 'http://lp01ut10.rchland.ibm.com:10064/web/+
000110                     services/FindCustomers'+null;
000111          WsStubP  = getStub(%Addr(Endpoint));
000112
000113          // Initialize input parameters
000114          tempListType = 'M'     + null;
000115          tempPosition = 'Brown'+ null;
000118          tempCountReq = 5;
000119          ListType     = %addr(tempListType);
000120          Position     = %addr(tempPosition);
000121          CountReq     = %addr(tempCountReq);
000122
000123          // Invoke the FindCustomers_XML Web service operation.
000124          CustomersP   = FindCustomers(WsStubP:%Addr(Input));
000125
000126          // Print Message
000127          printf(newl + 'RPG Find Customer Web Service' + newl +newl);
000128
000129          // Check for NULL
000130          if (CustomersP <> *NULL);
000131            //Print XML Customer Return Content
000132            printf (CustomersP);
000133          ELSE;
000134            //Print Error Message
000135            printf(newl+'*** RPG Find Customer Contains no Data ***' + newl);
000137          ENDIF;
000138
000139          // Clean up storage for the result and Web service stub.
000181          destroyResult(CustomersP:0);
000182          destroyStub(WsStubP);
000185          *inlr = *on;
000187      /END-FREE
```

6

7

8

9

10

© 2009 IBM Corporation

http://www.ibm.com/systems/i/software/iws

Step 2. Build the client application

At this point it is assumed that the generated stubs are in a directory called /myconverttemp. The RPG code that uses the generated stubs is located in **<inst_dir>**/samples/ConvertTemp /CNVRTTMPC.RPGLE.

To build the CNVRTTEMP client application, complete the following steps from the CL command line.

1. Create a library that will contain the module and program objects. In this example, we create a library called CVTTEMP by issuing the following CL command:
   `CRTLIB CVTTEMP`
2. Change the current working directory to '/myconverttemp' by issuing the following CL command:
   `CD '/myconverttemp'`
3. Copy the RPG code that uses the generated stubs to ?/myconverttemp' by issuing the following CL command:
   `COPY OBJ('<inst_dir>/samples/converttemp/converttempclient.rpgle') TODIR('/myconverttemp')`
4. Update the following line in converttempclient.rpgle so that the host name and port is set to the host name and port of the Web services server. The code currently has host name set to "localhost" and the port set to 10000.
   Endpoint = 'http://localhost:10000/web/services/ConvertTemp' + X'00';
5. Build the client application using the following commands.

```
CRTRPGMOD MODULE(CVTTEMP/CVTTEMPCL)
SRCSTMF(ConvertTempClient.RPGLE)

CRTCMOD MODULE(CVTTEMP/CVTTEMPIN)
SRCSTMF('CONVERTTEMPInput.c')
INCDIR('/qibm/proddata/os/webservices/v1/client/include') ENUM(*INT)

CRTCMOD MODULE(CVTTEMP/CVTTEMPRES)
SRCSTMF('CONVERTTEMPResult.c')
INCDIR('/qibm/proddata/os/webservices/v1/client/include') ENUM(*INT)

CRTCMOD MODULE(CVTTEMP/CVTTEMPPT)
SRCSTMF('ConvertTempPortType.c')
INCDIR('/qibm/proddata/os/webservices/v1/client/include') ENUM(*INT)

CRTPGM PGM(CVTTEMP/CNVRTTEMP)
MODULE(CVTTEMP/*ALL) BNDSRVPGM(QSYSDIR/QAXIS10CC)
```


© 2009 IBM Corporation

RPG Call Find Customers Program

```
  RPG Find Customer Web Service

  <?xml version="1.0" encoding="UTF-8"?>
 <FINDCUSTOMERS>
   <COUNTRET>5</COUNTRET>
   <CUSTLIST>
     <NAME>Brown, Jacquelyn</NAME>
     <NUMBER>1781</NUMBER>
   </CUSTLIST>
   <CUSTLIST>
     <NAME>Brownigg, Daphine</NAME>
     <NUMBER>2350</NUMBER>
   </CUSTLIST>
   <CUSTLIST>
     <NAME>Brownigg, Ellen</NAME>
     <NUMBER>5215</NUMBER>
   </CUSTLIST>

===> _
F3=Exit F4=End of File F6=Print F9=Retrieve F17=Top
F18=Bottom  F19=Left   F20=Right F21=User Window
```


© 2009 IBM Corporation

Quick Review of Steps

Process to Run RPG Web Service:

1. Stub Generation – Creating a Web Services Proxy (“intermediary” for ILE RPG, COBOL)
2. Use Stubs to build RPG prototypes
3. Compile/Bind and Invocation
4. Run Program

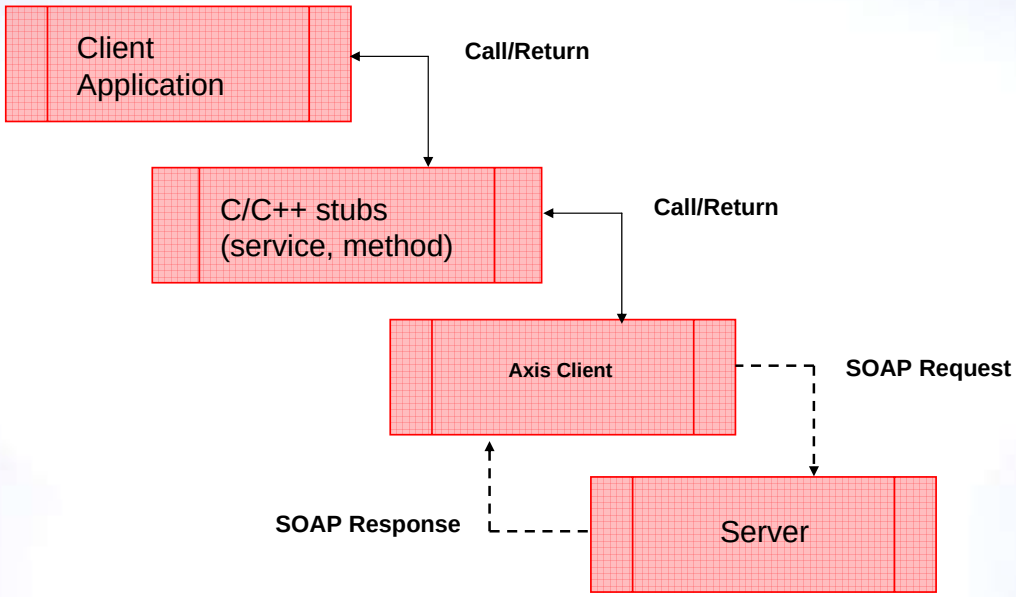
- Prerequisites
  - C++ Compiler (Compiler - ILE C++, licensed program product ID 5722WDS, option 52)
  - Java (IBM Developer Kit for Java, JDK 1.4, licensed program product ID 5722JV1, option 6)=
  - C Compiler (Compiler - ILE C, licensed program product ID 5722WDS, option 51)
    - Only needed if generating C stubs

© 2009 IBM Corporation

Web Services Client for ILE

Step 2: Compile/Bind and Invocation

- Create the application that uses the stubs to invoke the Web service



© 2009 IBM Corporation

IWS - Web site Information:

- **Integrated Web Services for IBM i - Website:**
  http://www.ibm.com/systems/i/software/iws/

- **Server/Client Tutorials:**
  http://www.ibm.com/systems/i/software/iws/quickstart.html
  http://www.ibm.com/systems/i/software/iws/quickstart_server.html
  http://www.ibm.com/systems/i/software/iws/quickstart_client.html

- **Limitations and Restrictions:**
  http://www.ibm.com/systems/i/software/iws/limitations.html

- **How to Order: - Group PTF - for V5R4 and V6R1**
  http://www.ibm.com/systems/i/software/iws/support.html

© 2009 IBM Corporation

Published Articles: - Chronologically

- **Integrated Web Services for IBM i - June 2008**
  http://www.ibmsystemsmag.com/ibmi/june08/trends/20795p1.aspx
- **A Walk in the Park - RPG developer Perspective: Dec 2008**
  http://www.ibmsystemsmag.com/ibmi/december08/features/23166p1.aspx
- **"Cover Article" - Diving Into Web Services and SOA on IBM i: Dec 2008**
  http://www.ibmsystemsmag.com/ibmi/december08/coverstory/23164p1.aspx
- **Getting Started With the Integrated Web-Services Server - Dec 2008**
  http://www.ibmsystemsmag.com/ibmi/december08/tipstechniques/23351p1.aspx
- **Power Under the Hood - Dec 2008**
  http://www.ibmsystemsmag.com/ibmi/december08/lessonsfromthelab/23339p1.aspx
- **Securing Web-Service Messages: - January 2009**
  http://www.ibmsystemsmag.com/ibmi/january09/administrator/23600p1.aspx?ht=watkins%20watkins%20watkins
- **Integrated Web Services Server for IBM i - Feb 2009**
  http://www.mcpressonline.com/application-software/ibm/integrated-web-services-server-for-ibm-i.html

© 2009 IBM Corporation

Other - Resources

- Integrated Web Services for IBM I
  http://www.ibm.com/systems/i/software/iws/
- IBM Technical Information and Example
  http://www.ibm.com/developerworks/
- WebSphere Enterprise Service Bus
  http://www.ibm.com/software/integration/wsesb/
- WebSphere Process Server
  http://www-306.ibm.com/software/integration/wps/

© 2009 IBM Corporation

Summary

- Integrated Web services for IBM i – Project
- Intro to SOA/WS on IBM i
- Web Services coding for RPG considerations
- Create, Deploy and Testing an RPG Web service
- Consuming Web services with RPG on IBM i

© 2009 IBM Corporation

Questions?

© 2009 IBM Corporation

Trademarks and Disclaimers

© IBM Corporation 1994-2009. All rights reserved.
References in this document to IBM products or services do not imply that IBM intends to make them available in every country.
Trademarks of International Business Machines Corporation in the United States, other countries, or both can be found on the World Wide Web at http://www.ibm.com/legal/copytrade.shtml.

Adobe, Acrobat, PostScript and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Cell Broadband Engine and Cell/B.E. are trademarks of Sony Computer Entertainment, Inc., in the United States, other countries, or both and are used under license therefrom.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

The customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Some information addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Prices are suggested U.S. list prices and are subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

© 2009 IBM Corporation