Practical Web Services for RPG
IBM Integrated Web services for i

Dan Hiebert  IBM dhiebert@us.ibm.com

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
Web Services Made Easy
An IBM i Project

Power = Integration + Solutions

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/
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)

Introduction to Web services and SOA

Power = Efficiency - Cost
**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."

**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
SOA Is Like Musical Notes…

Each musical notes represents a business service

SOA allows for flexible composition of music

Web services are a good start…

Turn this …

...into this (web services).

- 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

POWER = INTEGRATION + SOLUTIONS

The New Power Equation

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
What is a RPG Service?

• Function
  – Get information
  – Perform action

• Properties
  – Encapsulated
  – Reusable
  – Stateless
  – Event driven
  – Loosely coupled

• Modular

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
Create, Deploy and Testing an RPG Web service

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
NEW Improvements for YOU! – October/08

Embedding Web services information with RPG on IBM i
Example: RPG Find Customer

V5R4 - Enablement

```
0001.00 h noman bndir('FLGHT400M')  PGINFO(*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      101 0 const
0003.62 d CountRet      101 0
0003.63 d CustList      likeds(CustInfo) dim(100)
0003.64 d options(*vsize)
```
**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’
Web Admin: Install Web Service


Click on the Create New Web Services Server link

Web Admin: Install Web Service

• Step 1: Create the server to run Web services.
Web Admin: Install Web Service

• Step 2: Select to Install a new Web service

Create Web Services Server
Externalize an i5/OS Program as a Web Service - Step 2 of 8

You may externalize an i5/OS program object as a Web service. The program object must be an existing Integrated Language Environment (ILE) program (IP04) or service program (ISP04) object. Currently, only program objects written using the COBOL or RPG programming languages are supported.

Externalize an i5/OS program as a Web service:

☐ Deploy new service to externalize an i5/OS program.

Back  Next  Cancel

Web Admin: Install Web Service

• Step 3: What program contains the services?

Create Web Services Server
Deploy New Service: Specify Location of i5/OS Program Object - Step 3 of 8

This i5/OS object to be externalized as a Web service must be an existing ILE program (IP04) or service program (ISP04) object located on the system. Currently, only program objects written using the COBOL or RPG programming languages are supported.

Specify the library and program object for the Web service:

☐ Specify i5/OS library and ILE program object name (Recommended)

You can specify the program object location by entering the name of the library that contains the program object, as well as the name of the program object. There is an optional and recommended way to locate the program object.

Library name: **WSSRI**
ILE Object name: **NFS409_TH2**
ILE Object type: **SRVPGM**

☐ Browse the integrated file system for the i5/OS program object

Back  Next  Cancel
Web Admin: Install Web Service

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

• Step 5: What in the ILE program should be externalized as a Web service?
Web Admin: Install Web Service

• Step 6: Specify User for the Web service

- Create Web Services Server
  - Deploy New Service. Specify User ID for this Service - Step 6 of 6

The service requires an i5/OS user ID to run the program object that contains the Web service business logic.

Specify User ID for this Service:

- Use server's user ID
- Specify an existing user ID

The user ID must have the necessary authority to this program object and any other additional program objects.

User ID: MyUser

Update the server's user ID to have IUSE authority to this user ID.

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

- Create Web Services Server
  - Deploy New Service: Specify Library List - Step 7 of 8

The functionality of the i5/OS program you want to extend as a Web service may depend upon other i5/OS programs in the system. Specify all libraries in which programs exist that the Web service programs depend on. If no library is specified, default libraries will be used.

Library entries:

- NYSRI
- QGPL
- DTEMP
- FLIGHT43

Add Remove Remove All Move Up Move Down Continue

Back Next Cancel
Web Admin: Install Web Service

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

Create Web Services Server

Summary - Step 8 of 8

When you click Finish, everything necessary to run the Web services server is created:

Server

Sample services name | Sample service definition URL
---------------------|-----------------------------------

User deployed service:

Name: FindCustomers

Description: NF3400_THR

Service install path: http://ip:/ ricland.ibm.com:10071/web/services.FindCustomers

User ID for service: *WEBSERV (GOSERVICE)

Library:

Program: NF3400_THR (SRV)00

Library list for service: *WEB (DEL, 2.2 TEMP 1400)

Web service definition URL: http://ip:8080/ricland.ibm.com:10071/web/services/FindCustomers?wsdl

HTTP Server Information

Server name: WSERVICE

HTTP server description: HTTP server created by the Create Web Services Server wizard.

Port: 8074


Server root: www/WSERVICE

Server association: WSERVICE
Web Admin: Install Web Service

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

When you click Finish everything necessary to run the Web services server is created.

User deployed service:
Name: NFS400_THR
Description: NFS400_THR
Service install path: /software/services/NFS400_THR
User ID for service: SERVER (WSERVICE)
Library: /WS/$LIB
Program: NFS400_THR.DRV
Library list for service: /software/services/NFS400_THR.DRV,NFS400_THR.DRV
Web service definition URL: http://10.11.11.11/Web/services/NFS400_THR

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

The Web services server provides a convenient way to externalize existing programs running on iSeries, such as RPG and COBOL programs, as Web services. Web service clients can then interact with these iSeries program-based services from the Internet or intranet using Web service based industry standard communication protocols such as SOAP. The clients can be implemented using a variety of platforms and programming languages such as C, C++, Java and .NET. An easy to use wizard is provided to configure the Web services server and the services for iSeries program objects. Other management functions such as starting, stopping and deleting services are also provided.

For more information, please visit: http://www-03.ibm.com/systems/i/software/ws/
Web Admin: Install Web Service

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

Web Admin: Install Web Service

• Manage installed Services – view and install new services to this server
Web Admin: Install Web Service

- View the WSDL file

```xml
<definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mime="http://www.w3.org/2006/02/addressing/mime"
targetNamespace="http://findcustomers.webbeans.iseseries/">
  <types>
    <element name="findcustomers">
      <complexType>
        <sequence>
          <element minOccurs="0" name="param0" nillable="true" type="xsd:string"/>
        </sequence>
      </complexType>
    </element>
    <element name="findcustomersResponse">
      <complexType>
        <sequence>
          <element minOccurs="0" name="return" nillable="true" type="xsd:string"/>
        </sequence>
      </complexType>
    </element>
  </types>
  <message name="FindCustomers">
    <part name="param0" type="xsd:string"/>
  </message>
  <message name="FindCustomersResponse">
    <part name="return" type="xsd:string"/>
  </message>
  <portType name="FindCustomersPortType">
    <operation name="FindCustomers">
      <input message="FindCustomers"/>
      <output message="FindCustomersResponse"/>
    </operation>
  </portType>
  <binding name="FindCustomersSoap11Binding" type="soap:Bindng">
    <operation name="FindCustomers">
      <soap:operation soapAction="http://findcustomers.webbeans.iseseries/">
        <input>
          <soap:body use="encoded" encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/>
        </input>
      </soap:operation>
    </operation>
  </binding>
  <service name="FindCustomersService" xmlns="http://findcustomers.webbeans.iseseries/">
    <port name="FindCustomersPort" type="FindCustomersPortType">
      <soap:address location="http://localhost:8080/WSERVICE/V1.3/FindCustomersPort"/>
    </port>
  </service>
</definitions>
```

The New Power Equation
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)
Consuming Web services with RPG on IBM i

The Unlimited Potential

Power = Integration + Solutions

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
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/

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)
Web Service Deployment Review

Where to find my WSDL
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`
FindCustomersPortType.h

```c
/* Functions relating to web service client proxy */
#define ------- */
#define ------- */

extern AXISHANDLE get FindCustomersPortType_stub(const char* pchEndPointUri);
extern void destroy FindCustomersPortType_stub(AXISHANDLE pStub);
extern int get FindCustomersPortType_status(AXISHANDLE pStub);
extern void set FindCustomersPortType_ExceptionHandler(AXISHANDLE pStub, AXIS_EXCEPTION_HANDLER_FUNC fp);

/* Functions relating to web service methods */

extern AXISCHAR* findCustomers_XML(AXISHANDLE pStub, FINDCUSTOMERSInput* pValue);
extern FINDCUSTOMERSResult* findCustomers(AXISHANDLE pStub, FINDCUSTOMERSInput* pValue);
```

FINDCUSTOMERSInput.h

```c
/* Custom type */
#define ------- */
#define ------- */

typedef struct FINDCUSTOMERSInputTag {
  xdoc_int  COUNTAQP;
  xdoc_string LISTTYPE;
  xdoc_string POSITION;
} FINDCUSTOMERSInput;

/* Functions to create/delete, serialize/deserialize custom type */
#define ------- */
#define ------- */

extern int Axis_Deserialize_FINDCUSTOMERSInput(FINDCUSTOMERSInput* param, AXISHANDLE pDZ);
extern void Axis_Create_FINDCUSTOMERSInput(FINDCUSTOMERSInput* initSize);
extern void Axis_Delete_FINDCUSTOMERSInput(FINDCUSTOMERSInput* param, int nSize);
extern int Axis_Serialize_FINDCUSTOMERSInput(FINDCUSTOMERSInput* param, AXISHANDLE pS2, AxisBool bArray);
```
RPG Call Find Customers Program

RPG Find Customer Web Service

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

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

The New Power Equation
Web Services Client for ILE

Step 2: Compile/Bind and Invocation

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

IWS - Web site Information:

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

• Server/Client Tutorials:

• Limitations and Restrictions:

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

- Integrated Web Services for IBM i - June 2008

- A Walk in the Park - RPG developer Perspective: Dec 2008

- "Cover Article" - Diving Into Web Services and SOA on IBM i: Dec 2008

- Getting Started With the Integrated Web-Services Server - Dec 2008

- Power Under the Hood - Dec 2008

- Securing Web-Service Messages: - January 2009

- Integrated Web Services Server for IBM i - Feb 2009
  http://www.mcpressonline.com/application-software/ibm/integrated-web-services-server-for-ibm-i.html

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
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

Questions?

Thank You
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.

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.
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.

The New Power Equation

© 2008 IBM Corporation