The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
PeopleTools Developer Series: Building and Consuming Web Services

Dave Bain
Product Management, PeopleTools
Program Agenda

- Why/When Web Service?
- Web Service Enabling PeopleTools Objects
- Provide Web Service Using Component Interfaces (CI)
- Provide Web Service Using Application Packages
- Provide Restful Web Service
- Consuming a 3rd Party Web Service
- Testing a Web Service
- Other Resources
Web Services with PeopleTools
When are they the right option

• Web Services are only one of many integration options

• When are Web Services good options?
  – Multiple consumers of a single interface
  – If consumer is external and needs to be authenticated
  – Consuming partner integrations
  – Integrating with Standards Based Tools
    • SOA technology
    • Development Tools
    • Business Intelligence
Web Service Enabling PeopleTools
Web Service Enabling PeopleTools

- PeopleTools Objects deployed as Web Services
Web Service Enabling PeopleTools

- PeopleTools Objects deployed as Web Services
Web Service Enabling PeopleTools

• Web Service Routing is always Any-To-Local
Pre-requisite Configuration

- Integration Gateway must be Configured
  - Gateways Component
  - Gateway Introspection for Target Connectors
  - Modify IntegrationGateway.properties file
Pre-requisite Configuration

• Integration Gateway must be Configured
  – Gateways Node Configuration

PeopleSoft Node Configuration

URL: http://rtdc79575vmc/PSIGW/PeopleSoftListeningConnector

Gateway Default App. Server

<table>
<thead>
<tr>
<th>App Server URL</th>
<th>User ID</th>
<th>Password</th>
<th>Tools Release</th>
<th>Domain Password</th>
<th>Virtual Server Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>/rtdc79575vmc:9012</td>
<td>QEDMO</td>
<td>*********</td>
<td>8.51</td>
<td>●</td>
<td>PSFT_HR</td>
</tr>
</tbody>
</table>

PeopleSoft Nodes

<table>
<thead>
<tr>
<th>Node Name</th>
<th>App Server URL</th>
<th>User ID</th>
<th>Password</th>
<th>Tools Release</th>
<th>Domain Password</th>
<th>Ping Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSFT_HR</td>
<td>/rtdc79575vmc:9012</td>
<td>QEDMO</td>
<td>*********</td>
<td>8.51</td>
<td>●</td>
<td>+</td>
</tr>
<tr>
<td>PSFT_CRM</td>
<td>/rtdc79575vmc_cr:9012</td>
<td>VP1</td>
<td>***</td>
<td>8.51</td>
<td>●</td>
<td>Ping Node +</td>
</tr>
</tbody>
</table>

Advanced Properties Page

[OK] [Cancel] [Save]
Pre-requisite Configuration

• Configuration
  – Service Configuration Component
    • Include Namespaces, Target Location, System Status
Pre-requisite Configuration

• Configuration
  – PT 8.52 – Integration Network
    • Streamlines Integration Broker Setup and Administration
    • Shows all configured IB Components in one page
Integration Broker
Web Services

• SOAP or Rest?
• Component Interface or AppClass?
• Rowset or non-rowset message?
• Parts and containers or Document?
Provide Web Service

Using a Component Interface
Web Service Enabling PeopleTools

- PeopleTools Objects deployed as Web Services
Provide Web Service Using Component Interfaces

- Synchronous Service Operation for each Component Interface methods
- Automatically creates Nonrowset-based Messages with Schemas based on the CI definition
- Builds handler definition that points the CI
- Any to Local routing is defined
- Naming convention for Web Service is CI_<name>
- Supports user defined methods
Provide Web Service Using Component Interfaces

• 2 Step Process
  – Step 1: CI-Based Services Wizard
    • Generates Messages and Schemas, Services, Service Operations, SO Handlers, and SO Routings via CI-Based Services Page
  – Step 2: Provide Web Service Wizard
    • Publish as Web Service by selecting the above generated Service definition
Web Service From a Component Interface
Web Service From a Component Interface

CI-Based Services
Select Component Interfaces

Component Interface Name: MOBILE_PREFS_CI
Component Name:
Owner ID:

Search

Select CIs

<table>
<thead>
<tr>
<th>Select</th>
<th>CI Name</th>
<th>Description</th>
<th>Component Name</th>
<th>Object Owner ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐</td>
<td>MOBILE_PREFS_CI</td>
<td></td>
<td>MOBILE_PREFS_COMP</td>
<td>PeopleTools</td>
</tr>
</tbody>
</table>

Review CI Status
Notify
Web Service From a Component Interface
Web Service From a Component Interface
Web Service From a Component Interface

Provide Web Service Wizard
Step 2 of 4

Select Service Operations
Select one or more operations for each service.

Service: CI_MOBILE_PREFS_CI
Description: CI_MOBILE_PREFS_CI

<table>
<thead>
<tr>
<th>Operations</th>
<th>Description</th>
<th>Operation Type</th>
<th>Request Message</th>
<th>Response Message</th>
<th>Fault Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI_MOBILE_PREFS_CI_F.V1</td>
<td>CI_MOBILE_PREFS_CI_F</td>
<td>Synchronous</td>
<td>M774977.V1</td>
<td>M947040.V1</td>
<td>M546912.V1</td>
</tr>
<tr>
<td>CI_MOBILE_PREFS_CI_G.V1</td>
<td>CI_MOBILE_PREFS_CI_G</td>
<td>Synchronous</td>
<td>M314636.V1</td>
<td>M106692.V1</td>
<td>M317505.V1</td>
</tr>
</tbody>
</table>

Select All  Clear All
Web Service From a Component Interface
Web Service From a Component Interface
Web Service From a Component Interface

Specify Publishing Options

The WSDL for the selected services will be published to the PeopleSoft WSDL Repository. You can also publish the WSDLs to one or more UDDI Servers.

- Publish to UDDI
- WSDL Repository
Web Service From a Component Interface

Provide Web Service Wizard

Confirm Results

View the WSDL Generation Log to confirm the results of the wizard.

WSDL Generation Log:

Service: CI_MOBILE PREFS_CI has been exported.
Inserted WSDL: CI_MOBILE_PREFS_CI.1 in the repository

Generated WSDL URL:

http://mo214135/PSIGW/PeopleSoftServiceListeningConnector/E910B21/CI_MOBILE_PREFS_CI.1.wsdl

Provide Another Service  Generate SOAP Template
Web Service From a Component Interface

```xml

<wsdl:wsdl>
  <wsdl:types>
    <xsd:schema elementFormDefault="qualified" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    </xsd:schema>
  </wsdl:types>

  <wsdl:message name="M774977.V1">
    <wsdl:documentation>http://www.oracle.com/Enterprise/FSCM/service/CI_MOBILE_PREFS_CI</wsdl:documentation>
    <wsdl:part element="M774977.V1:Find__CompIntf__MOBILE_PREFS_CI" name="parameter"/>
  </wsdl:message>

  <wsdl:message name="M947040.V1">
    <wsdl:documentation>http://www.oracle.com/Enterprise/FSCM/service/CI_MOBILE_PREFS_CI</wsdl:documentation>
    <wsdl:part element="M947040.V1:Find__CompIntf__MOBILE_PREFS_CIResponse" name="parameter"/>
  </wsdl:message>

  <wsdl:message name="M546912.V1">
    <wsdl:documentation>http://www.oracle.com/Enterprise/FSCM/service/CI_MOBILE_PREFS_CI</wsdl:documentation>
    <wsdl:part element="M546912.V1:Get__CompIntf__MOBILE_PREFS_CI" name="parameter"/>
  </wsdl:message>

  <wsdl:message name="M314636.V1">
    <wsdl:documentation>http://www.oracle.com/Enterprise/FSCM/service/CI_MOBILE_PREFS_CI</wsdl:documentation>
    <wsdl:part element="M314636.V1:Query__CompIntf__MOBILE_PREFS_CI" name="parameter"/>
  </wsdl:message>

  <wsdl:message name="M314636.V2">
    <wsdl:documentation>http://www.oracle.com/Enterprise/FSCM/service/CI_MOBILE_PREFS_CI</wsdl:documentation>
    <wsdl:part element="M314636.V2:Query__CompIntf__MOBILE_PREFS_CIResponse" name="parameter"/>
  </wsdl:message>
</wsdl:definitions>
```

Provide Web Service Using Component Interfaces

Limitations

• This is the easiest way to provide a Web Service but it comes at a cost

• The interaction between the Web Service and Component Interface is Generated and Can Not be Modified
  – State Management can be a problem
  – There is no support for level deletes

• Consider using an AppClass that manipulates the Component Interface as an alternative
Provide Web Service

Using Application Packages
Web Service Enabling PeopleTools

- PeopleTools Objects deployed as Web Services
Provide Web Service Using App Packages

• Web Services can be provided using any of the PeopleTools message types
  – Rowset-based
    • PeopleSoft to PeopleSoft with proprietary XML
    • Legacy - PSCAMA tags
  – Nonrowset-based (schema based)
    • 3rd Party Integration
  – Message Parts and Containers
    • Web Services
    • Standard XML
  – PT 8.51 - Document-based Messages
Provide Web Service Using App Packages

• PeopleTools Document Object Goals
  – Better XML support
  – Increase Developer Productivity
  – Light-weight structure definition
  – Geared for reuse
  – Insulated to change
Documents - definition

- Meta data that defines structured data
- Binding to XML schema or PeopleSoft record
- Structure composed of elements: primitives, compounds (document), and collections (of primitive or compounds).
  - Primitives represent scalar types (int, bool, etc).
  - Compounds represent complex types
  - Collections represent multiple occurrences of an element
- Rich PeopleCode API that does not require parsing
Document is

• Provide a light-weight structural specification that describes any XML Document
• Great reuse with compound documents
• Single programming api for the document definition
• Support mapping to different physical types
• Insulation from change to physical layer
• Support for XML, namespaces, attributes, non-rowset structures.
• Easy construction or generation of Documents (create from xsd, wsdl, or record)
• Managed object in PeopleTools

*Increase Application Developer Productivity*
Provide Web Service Using App Packages

- PeopleTools Document Object
  - Strengthens PeopleTools Consume Web Service model
    - Documents are built from XML schemas (XSD)
    - Creates a first-class Tools Object like a rowset
  - Replaces the need to use non-rowset based messages
    - Provides flexibility with the ease of use of rowsets and rowset parts
  - Can be extended to support other physical types in the future
  - Framework that will be used for future IB enhancements
Provide Web Service Using App Packages

• Sample PeopleCode for Documents

```xml
/** Sample code to create non-rowset based request message */
&request = CreateMessage(Operation.QE_FLIGHTPLAN_XMLDOC, %IntBroker_Request);
&xmlobj = &request.GetXmlDoc();
&ac_number = &xmlobj.CreateDocumentElement("flightplan");
&msi_sensor = &ac_number.AddElement("msi_sensor");
&msi_sensor.NodeValue = "flir";
&ofp = &ac_number.AddElement("ofp");
&ofp.NodeValue = "8.44";
&actype = &ac_number.AddElement("actype");
&actype.NodeValue = "F-18";
&callsign = &ac_number.AddElement("callsign");
&callsign.NodeValue = "DogDish";
```

```xml
/** Sample code to create non-rowset based request message */
&request = CreateMessage(Operation.QE_FLIGHTPLAN_DOCUMENT, %IntBroker_Request);
&docobj = &request.GetDocument();
&elem = &docobj.DocumentElement;
&elem.GetPropertyByTagName("msi_sensor") . Value = "flir";
&elem.GetPropertyByTagName("ofp") . Value = "8.44";
&elem.GetPropertyByTagName("actype") . Value = "F-18";
&elem.GetPropertyByTagName("callsign") . Value = "DogDish";
```
Provide Web Service Using App Packages

- PeopleTools Document Object
  - New message of document type can be defined
Provide Web Service Using App Packages

• Define Service
  – Logically group Service Operations
Provide Web Service Using App Packages

• Configure Service Operations
  – Ties Messages, Queues and Service Operation Versions
  – Defines Routings and Handlers for Operations
  – Service Operation Versions
    • Create new version when message structure is updated
    • Previous version is inactive unless transformation is implemented for the message structure
Provide Web Service Using App Packages

- Define Service Operation Handlers
  - Contains the main operation logic for inbound messages
  - AppClass must exist before the Service Operation
  - Implemented via Application Class, Component Interface or Bulk Load (DMS)
Provide Web Service Using App Packages

- Define Service Operation Routings
  - Routings can be generated automatically except for Point-To-Point
Provide Web Service Using App Packages

• Provide Web Service Wizard
Provide Web Service

Representative State Transfer
- REST
Provide Web Service - REST

• REST is
  – Architecture that consists of clients and servers (or requestors and providers)
  – There exist a number of resources that can be any arbitrary object
  – Resource has a representation that captures the current state of the resource
  – A resource may contain links to other resources
Provide Web Service - REST

• Support for Representative State Transfer (RESTful) Services
  – Defined as a Service type in Integration Broker
  – Alternative to WSDL/SOAP based web services
  – Support for
    • HTTP Based operations (post, get, put, delete, head)
    • Supported type – JSON, XML, HTTP, Text
  – Gives developers more options – one that is being heavily adopted
  – Easier to build/use expect gains in developer productivity
Provide Web Service - REST

- A new Service Type is added
  - RESTful or Non-RESTful
  - Defined at the Service, not Service Operation
Provide Web Service - REST

- Distinguish between provider and consumer
- Use Web Access Descriptive Language (WADL) not Web Service Descriptive Language (WSDL)
- Rest Method is an HTTP method, not Service Operation
Provide Web Service - REST
Provide Web Service - REST

- In this example, the URL to call the service is:
  
  http://rtdc79492vmc:813/PSIGW/PSIGW/RESTListeningConnector/Employees.v1/employees
Provide Web Service - REST

- In this example, the URL to call the service is one of these:
  - http://10.242.11.95/:813/PSIGW/RESTListeningConnector/AKTT/WeatherStation.v1/weather/CA
  - http://10.242.11.95/:813/PSIGW/RESTListeningConnector/AKTT/WeatherStation.v1/weather/CA/Oakland
Provide Web Service - REST

- REST Service Content Type
  - XML for standard messages
  - JSON (Java Script Object Notation) for UI based messages
Provide Web Service - REST

From this:

```xml
<?xml version="1.0"?>
<soapenv:Envelope xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
    <soapenv:Header xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
        <wsse:Security soap:mustUnderstand="1" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
            <wsse:UsernameToken wsu:Id="UsernameToken-1" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
                <wsse:Username>USER</wsse:Username>
                <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
            </wsse:UsernameToken>
            <wsse:Security>
                <wsse:Security>
                    <soapenv:Body xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"/>
                </soapenv:Header>
            </soapenv:Body>
        </soapenv:Header>
    </soapenv:Body>
</soapenv:Envelope>
```

To this: http://PSPC:813/PSIGW/RESTListeningConnector/Employee_Details.v1/details/KF0016
Consume Web Service

From an External Source
Consuming a 3rd Party Web Service

• Wizard to create the PeopleTools Components
• WSDL can be read from a UDDI Server, WSDL URL, WSIL URL, Local File or Legacy WSDL
• Generates Messages and Schemas, Services, Service Operations, and SO Routings based on the WSDL Definition
• PT 8.51 – Messages generated are Document-based
Consuming a Web Service

• Consume Web Service Wizard
Web Service

Testing
Testing a Web Service

- Testing Components
  - Integration Broker
    - Generate SOAP Template
      - Creates sample SOAP Messages
      - Can also invoke the Service Operation directly
    - Handler and Service Operation Tester
      - Create input data or use data from Service Operation Monitor
      - Tree interface to populate the message data
      - Handles Documents
  - Transformation Tester
    - Validate Logical and Physical Transforms
  - Schema Tester
    - Validate XML Data with Message Schema
Testing a Service

- Generate Simple Object Access Protocol (SOAP) Template
Testing a Service

- Transformation Tester

**Transformation Test**

- **Project Name**: PT_IBTRANSFORM_TEST
- **Program Name**: TRANSFORMTST
- **Source Node Name**: GE_LOCAL
- **Destination Node Name**: GE_LOCAL
- **File Name**: D:\TRANSFORMTST.xml

![Transformation Test Form](image)
Testing a Service

• Schema Tester
Testing a Web Service

• Testing Components
  – Documents
    • Document Tester
      – Creates a sample XML based on the Document Shape
    • Document Schema Tester
      – Validate XML Data with Document Shape
Web Service

Security
Web Service
WS-Security

- WS-Security is used to secure web services
  - Designed to secure open exposed endpoints
    - Authentication happens at endpoint
    - Something is passed acknowledging the authentication – Certificate or Username/PW
  - Many different policies for different levels and types of security
  - Not all policies will work with PeopleSoft
Web Service
WS-Security

- Rules of Integration Broker determine the User that is used to process the request
  - If no ws-security in SOAP header then the user is taken from the Anonymous Node
  - If Username/token or clear text password ws-security then user is taken from the ws-security name tag
  - If Security Assertion Markup Language (SAML) based ws-security, then user is derived from the saml user table based on the value in the ws-security name tag
Other Resources
Other Resources

• Instructor Lead Training
  – PeopleTools Integration Tools

• Self Service CDs

• Instructional Viewlets
  – http://download.oracle.com/peopletools/viewlets.html
Hosted & Mobile PeopleBooks - PeopleTools PeopleBooks are available in three formats: Hosted PeopleBooks, PDF’s, and Amazon’s Kindle format. All can be accessed here:

http://www.oracle.com/technetwork/documentation/psftent-090284.html

Doc Home Pages – constantly updated direct links to PeopleBooks, PeopleBook Updates, Release Notes, Installation and Upgrade Guides, and other useful product documentation, all accessible from one My Oracle Support location.

PeopleTools 8.51 Documentation Home Page [ID 1127534.1]
https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1127534.1

Information Portal - locate the documentation, training, and other info needed to help with your implementation process. Customers searching for this information should make this their first online destination.

Cumulative Feature Overview (CFO)- Providing concise descriptions of new and enhanced solutions and functionality that have become available starting with the 8.4 release through our latest 8.51 release.

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=SYSTEMDOC&id=793143.1

Upgrade Resource Report Tools - helps you find all the documentation, scripts, and files you need for your upgrade project.

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=SYSTEMDOC&id=1117047.1

Follow us on twitter PeopleSoft_Info
For More Information

search.oracle.com

or

oracle.com
SOFTWARE. HARDWARE. COMPLETE.
The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.