Web Services in Rhapsody

Presented by: Jamie Titak

Outline

- What are Web Services?
- IHE Profiles
- Web Services in Rhapsody
  - Host-based Web Services in Rhapsody
  - Utilizing the Rhapsody Web Service Client to connect to a Web service
- Communication Points, Operations and Parameters in Rhapsody
- IHE Toolkit
- Monitoring Web Services
- Web Services Demo

What are web services ?

- A method of communication that allows disparate systems to communicate
  - For example Windows and Unix
- Web services can be created regardless of the programming language
- Provides query/response or update services
- Rules for communication are defined in a WSDL
  - Encoded in XML
- Web Services have two uses
  - Reusable application-components
  - Connect existing software

Web Services Description Language (WSDL)

- Written in XML
- Used to describe and locate web services
- http://localhost:8880/services/GetPatientID?wsdl
- 5 major elements (can contain other elements)
  - <types>
  - <message>
  - **<portType>**
  - <binding>
  - <Service>

Operations

- Defined in the <port type> element
- 4 Types of Operations
  - One-way
  - Request-response
  - Solicit-response
  - Notification

WSDL Example Definition


```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="http://localhost:8880/services/GetPatientID"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:ns0="http://localhost:8880/services/GetPatientID" 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/">
<wsdl:documentation>Get Patient ID</wsdl:documentation>
        <types>
        <message>
        <portType>
        <binding>
        <Service>
<wsp:Policy wsu:Id="servicePolicy" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"><wsp:ExactlyOne><wsp:All><wsam:Addressing wsp:Optional="true"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"><wsp:Policy><wsam:AnonymousResponses/></wsp:Policy></wsam:Add
ressing></wsp:All></wsp:ExactlyOne></wsp:Policy>
</wsdl:definitions>
```

WSDL Example <Type> Portion

```
 <wsdl:types>
  <xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified“ targetNamespace="http://localhost:8880/services/GetPatientID">
<xs:element name="GetpatientID">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="FirstName" type="xs:string"/>
        <xs:element minOccurs="0" name="LastName" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetpatientIDResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="PatientID" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetpatientIDFault">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="payload" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
 </wsdl:types>
```

WSDL Example <Message> Portion

```
<wsdl:message name="GetpatientIDRequest">
  <wsdl:part name="parameters" element="ns0:GetpatientID">
  </wsdl:part>
 </wsdl:message>
 <wsdl:message name="GetpatientIDFault">
  <wsdl:part name="parameters" element="ns0:GetpatientIDFault">
  </wsdl:part>
 </wsdl:message>
 <wsdl:message name="GetpatientIDResponse">
  <wsdl:part name="parameters" element="ns0:GetpatientIDResponse">
  </wsdl:part>
 </wsdl:message>
```

WSDL Example <PortType> Portion

```
<wsdl:portType name="GetPatientIDPortType" wsp:PolicyURIs="#servicePolicy">
  <wsdl:operation name="GetpatientID">
<wsdl:documentation>GetpatientID</wsdl:documentation>
    <wsdl:input message="ns0:GetpatientIDRequest" wsaw:Action="urn:GetPatientID">
  </wsdl:input>
    <wsdl:output message="ns0:GetpatientIDResponse"
wsaw:Action="http://localhost:8880/services/GetPatientID/GetPatientIDPortType/GetpatientIDR
esponse">
  </wsdl:output>
    <wsdl:fault name="GetpatientIDFault" message="ns0:GetpatientIDFault"
wsaw:Action="http://localhost:8880/services/GetPatientID/GetPatientIDPortType/GetpatientIDF
ault">
  </wsdl:fault>
  </wsdl:operation>
</wsdl:portType>
```

WSDL Example <Binding> Portion

```
<wsdl:binding name="GetPatientIDSoap12Binding" type="ns0:GetPatientIDPortType">
  <soap12:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsaw:UsingAddressing wsdl:required="false"/>
  <wsdl:operation name="GetpatientID">
    <soap12:operation soapAction="urn:GetPatientID" style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
    <wsdl:fault name="GetpatientIDFault">
      <soap12:fault name="GetpatientIDFault" use="literal"/>
    </wsdl:fault>
  </wsdl:operation>
</wsdl:binding>
```

WSDL Example <Service> Portion

```
<wsdl:service name="GetPatientID">
  <wsdl:port name="GetPatientIDHttpSoap12Endpoint"
binding="ns0:GetPatientIDSoap12Binding">
    <soap12:address
location="http://titakj2.galenhealthcare.com:8880/services/GetPatientID.GetPatientIDHttpSoap12Endpoint"/>
  </wsdl:port>
</wsdl:service>
```

Simple Object Access Protocol (SOAP)

- Provides xml based structure for managing communications
- Contains the following elements which are declared in the namespace
  - Envelope
  - Header
  - Body
  - Fault

Anatomy of a SOAP Request

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:pat="http://www.galenhealthcare.com/LookupPatientID">
   <soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
      <wsa:Action>urn:LookupPatientID</wsa:Action>
      <wsa:MessageID>messageid</wsa:MessageID>
   </soap:Header>
   <soap:Body>
      <pat:lookupPatientID>
         <pat:FirstName>Jamie</pat:FirstName>
         <pat:LastName>Titak</pat:LastName>
      </pat:LookupPatientID>
   </soap:Body>
</soap:Envelope>
```

Anatomy of a SOAP Response

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:RelatesTo>messageID<wsa:RelatesTo>
  </soapenv:Header>
  <soapenv:Body>
    <pat:LookupPatientID xmlns:pat="http://www.galenhealthcare.com/LookupPatientID">
      <pat:PatientId>1243534</pat:PatientId>
    </pat:LookupPatientID>
  </soapenv:Body>
</soapenv:Envelope>
```

Integrating the Healthcare Enterprise (IHE)

- Non-profit solving the problem on how to cost effectively implement regional integration between healthcare systems
- Promotes the coordinated use of established standards such as DICOM and HL7
- Domains
  - Anatomic Pathology, Cardiology, Dental, Eye Care, IT Infrastructure, Laboratory, Patient Care Coordination, Patient Care Devices, Pharmacy, Quality, Research and Public Health, Radiation Oncology, Radiology
- Integration Profiles
  - Describe specific solutions to integration problems and can be used as an implementation guide
- Example Profile: Patient Demographic Query (PDQ)
  - Allows applications to query a central patient information server and retrieve a patient's demographic and visit information
- Resources
  - http://www.ihe.net
  - http://wiki.ihe.net

Web Services in Rhapsody

- Rhapsody provides an easy-to-use interface for harnessing Web services, Independent of
  - Whether the Web service is hosted by Rhapsody
  - Whether Rhapsody uses an external Web service.
- Vendors can exploit Rhapsody's secure Web services for application integration
- Supports SOAP/WSDL as a host and client
- Provides a limited set of REST-based operations

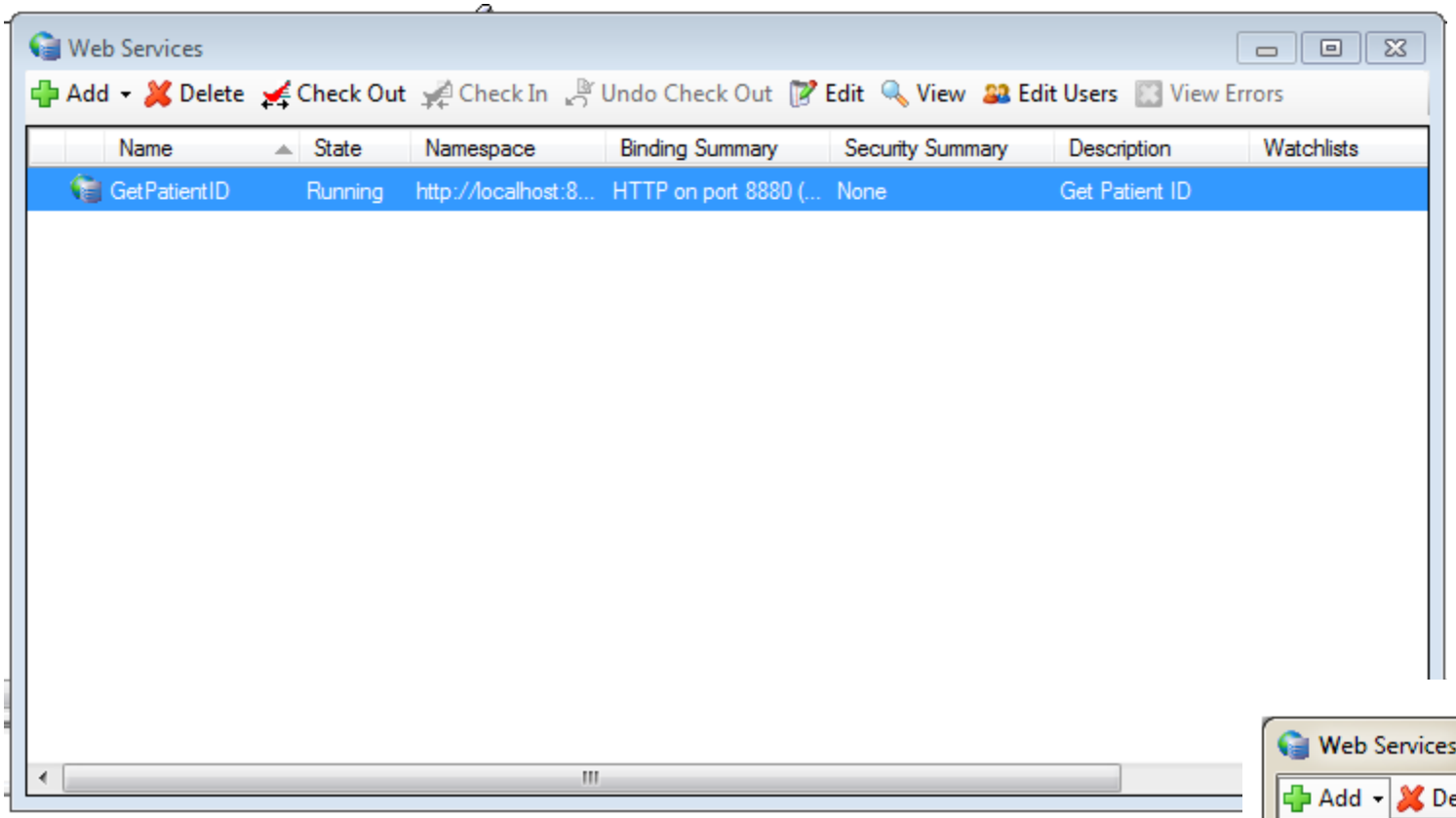
Web services in Rhapsody Continued

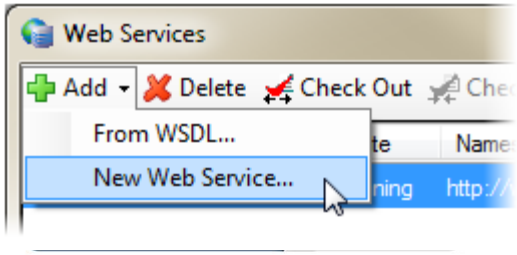
- Web Services are created and managed utilizing the Web Services Management dialog in the IDE environment
- Activity and state of web services may be monitored utilizing the Monitoring / Web Services item in the Web Management Console.
- IHE toolkit to implement IHE profiles
- Web Service Host hosts web services whereas the Web Service Client connects to web services

Web Service Client

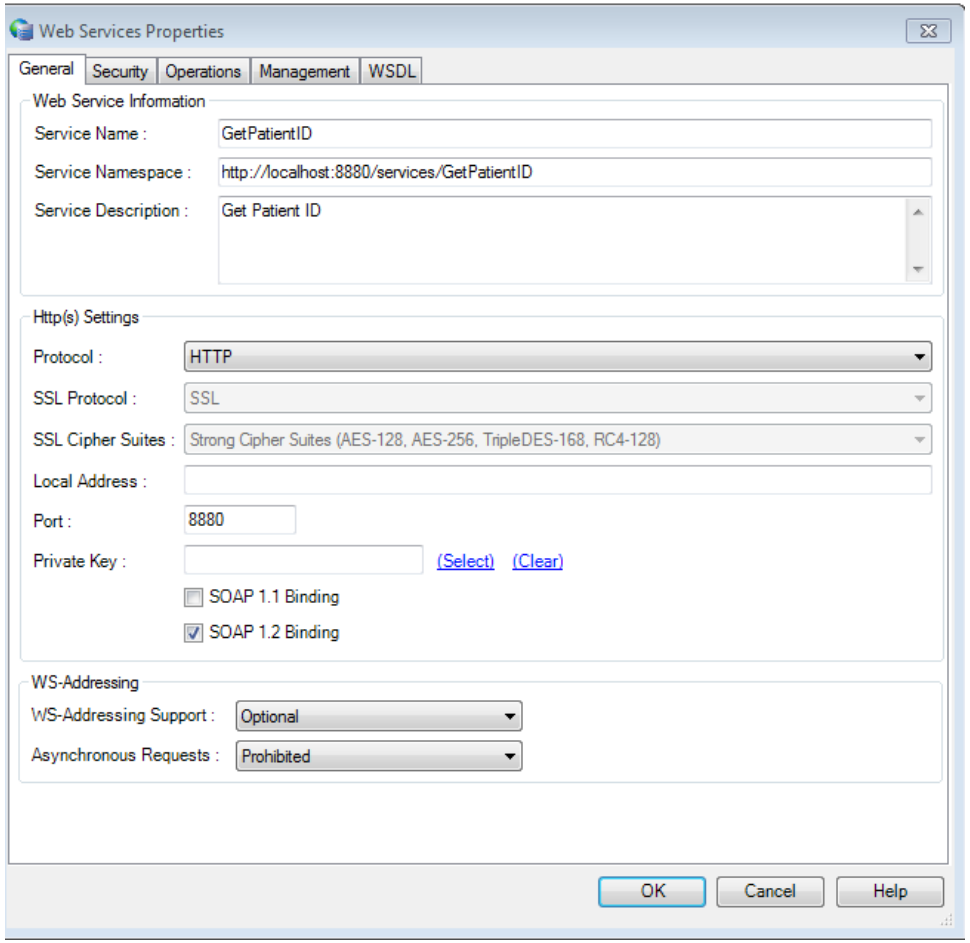
Web Service Hosting

Hosted Web Services

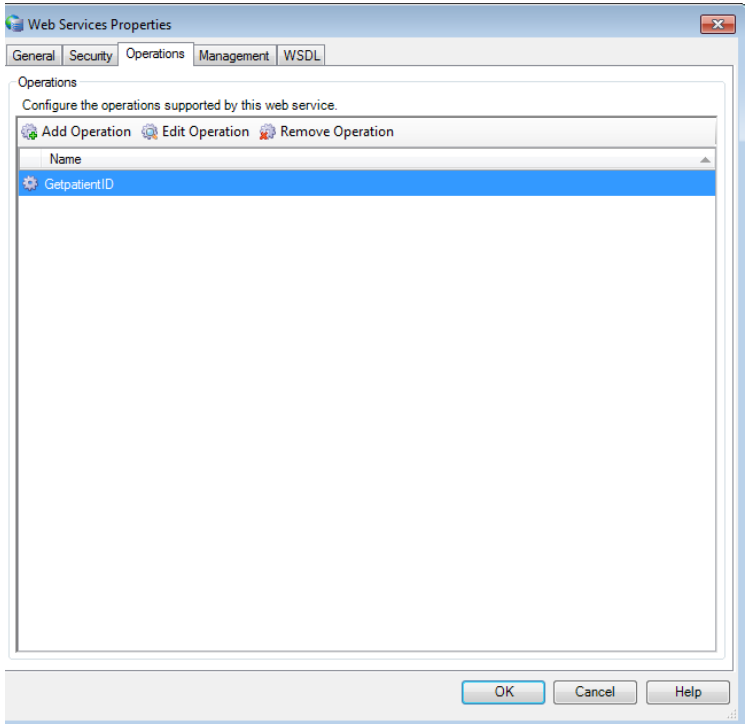




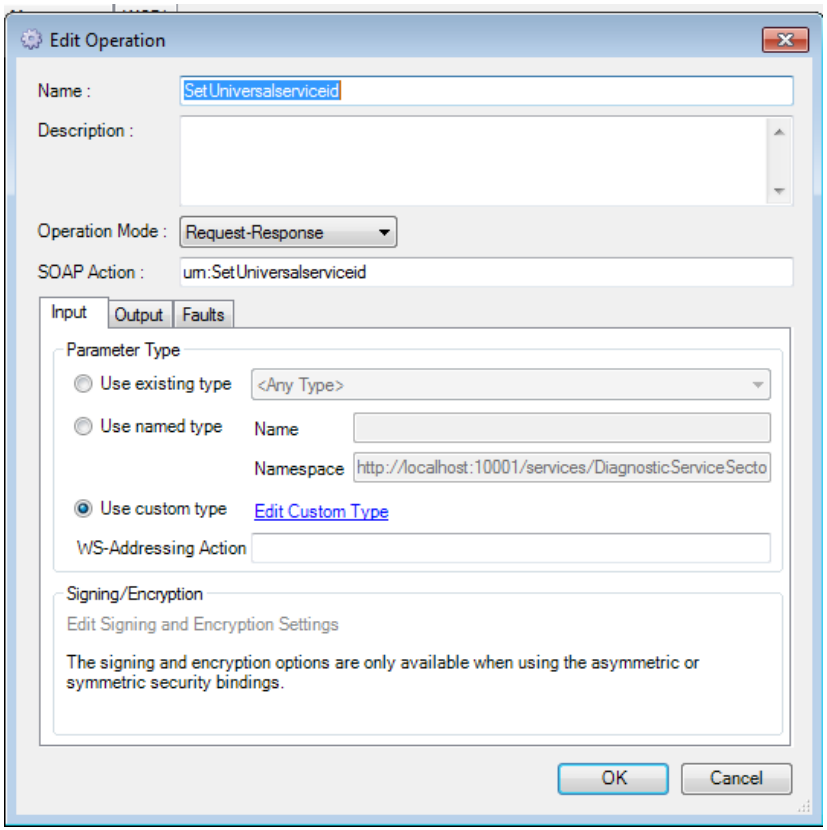
Rhapsody Web Services Properties for Web Service Hosting

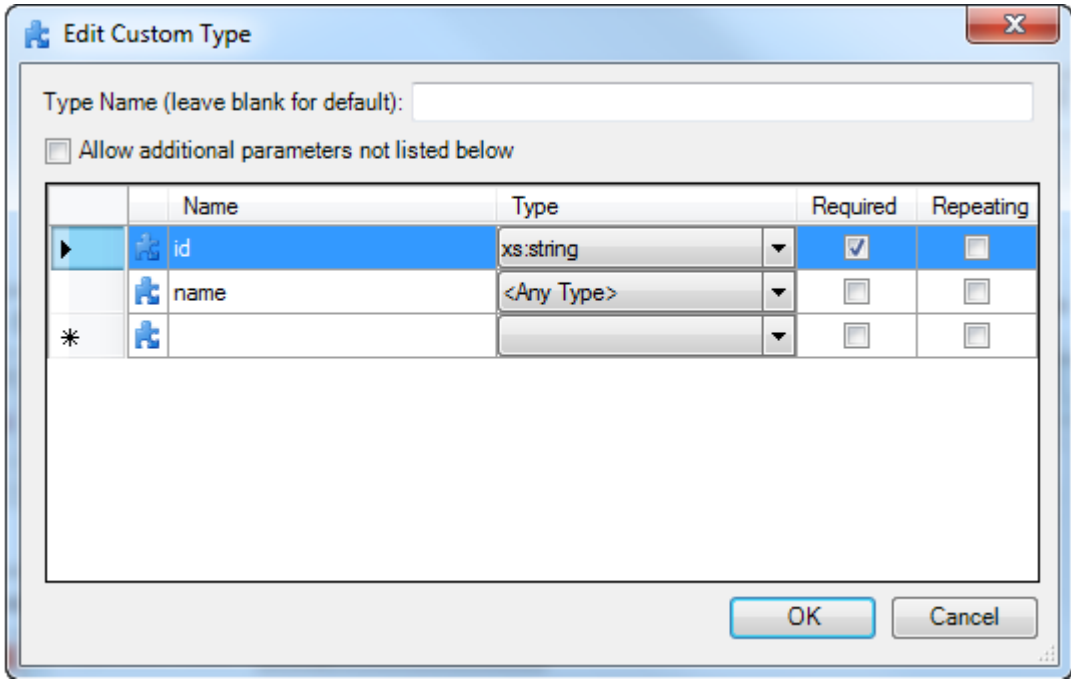


Edit Operations for Web Service Hosting

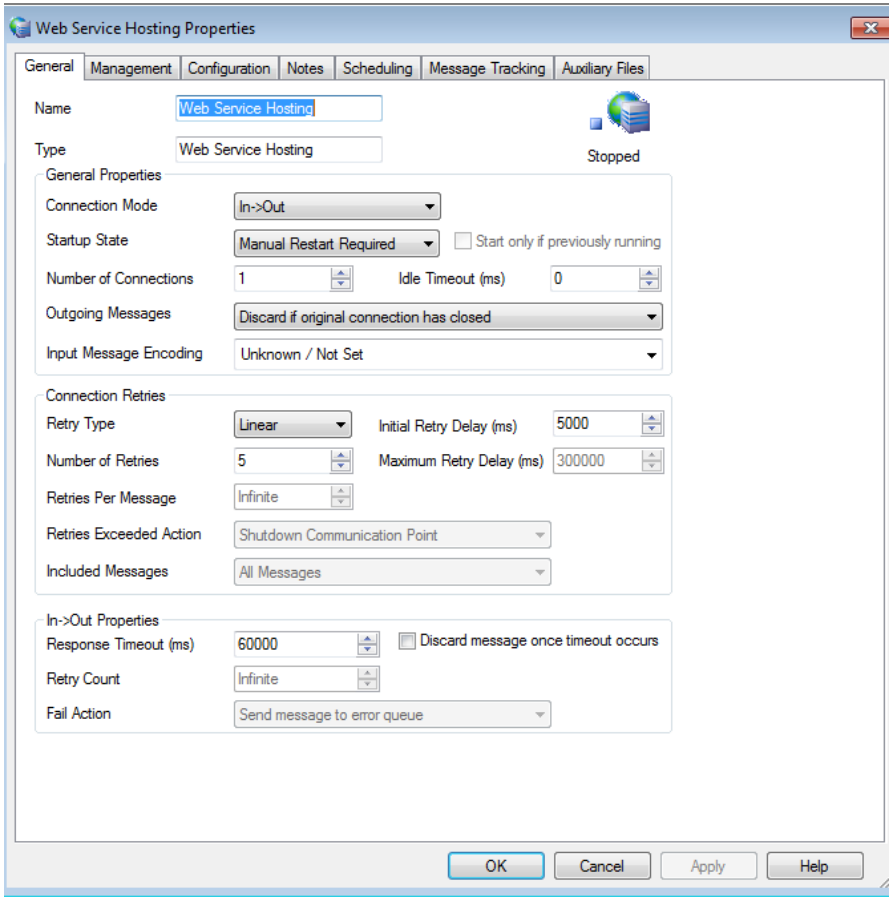


Edit Parameters for Web Service Hosting

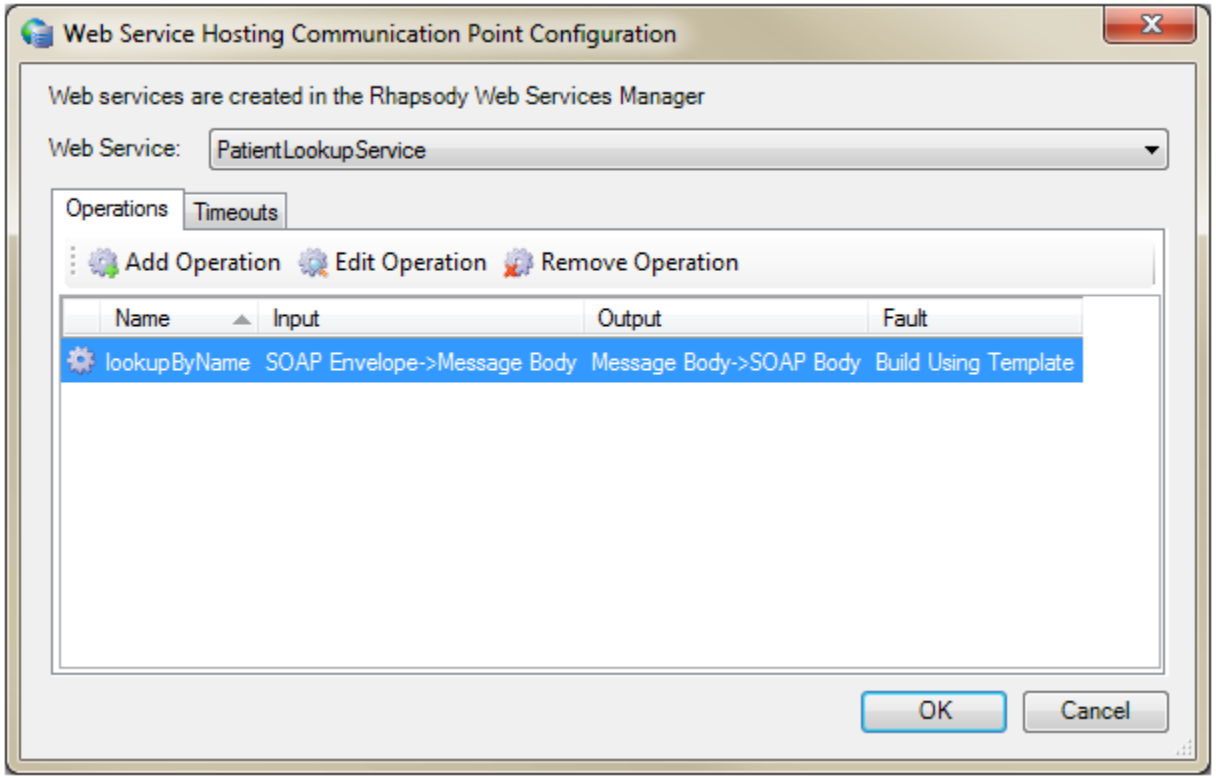




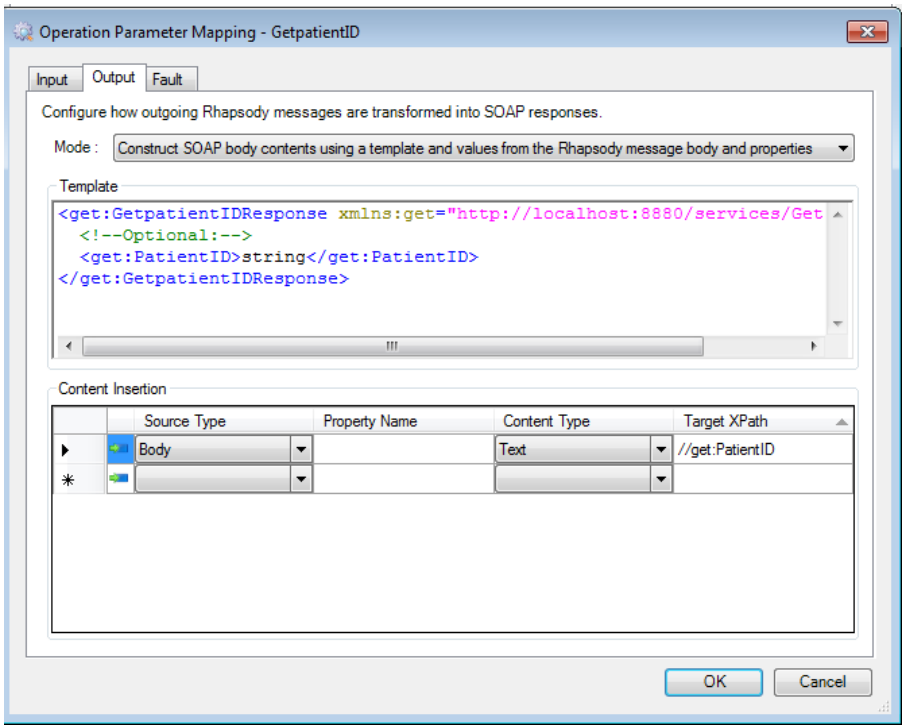
Communicating with Web Services (Host Web Service communication point)



Edit Operation for Web Service Hosting Communication Point



Edit Parameters for Web Service Hosting Communication Point



Communicating with Web Services (Web Service Client)

Top-down approach

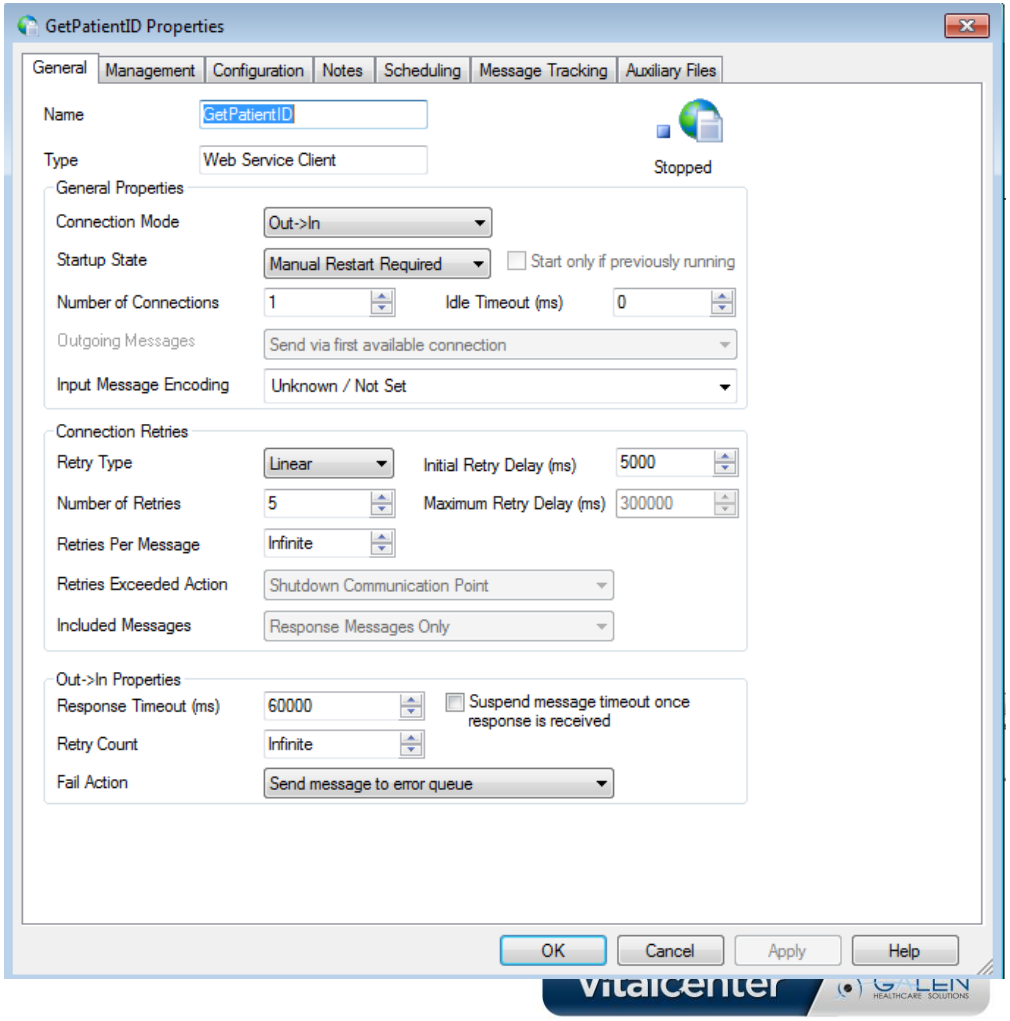
- Uses an existing Web Services Description Language (WSDL).
- The WSDL is imported and is used to generate the Web service configuration. You can then modify the configuration, and your changes will be reflected in the WSDL that is hosted by Rhapsody.

Bottom-Up approach

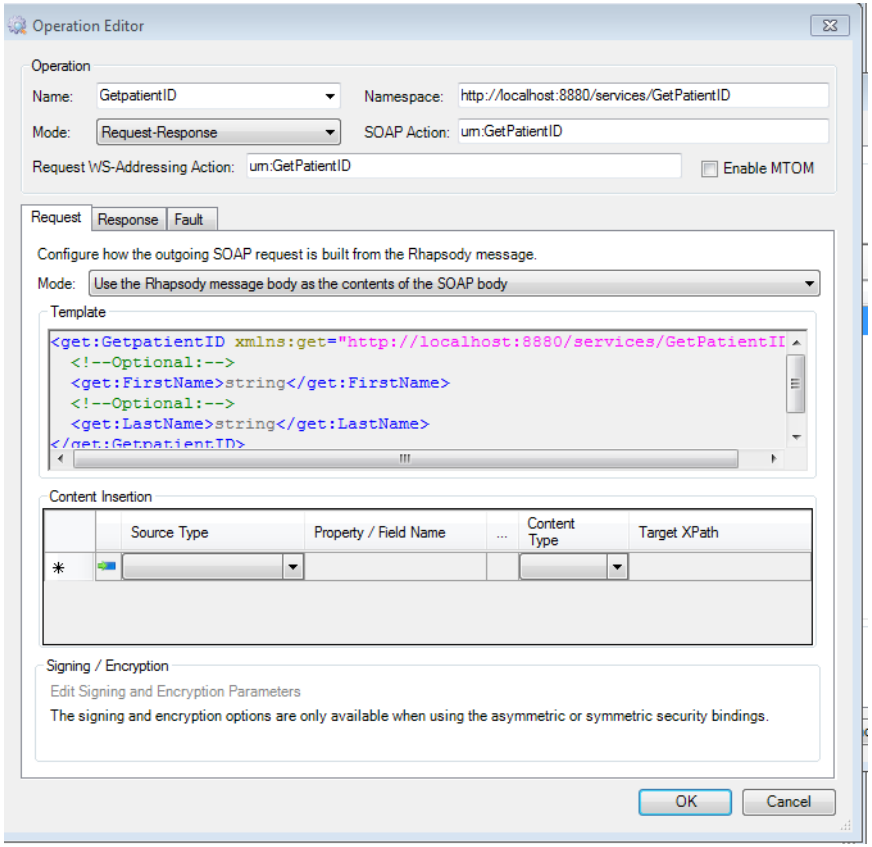
- The bottom-up approach does not require a WSDL.
- Instead you create a new Web service, and configure all the settings manually.
- We recommend this approach when exposing an existing Rhapsody route as a Web service.

Web Services Client Configuration

- Connects to Web Services hosting by WSDL or by manually configuring the settings
  - General Settings
  - Operations
    - In-Only
    - Request-Response
  - Endpoint

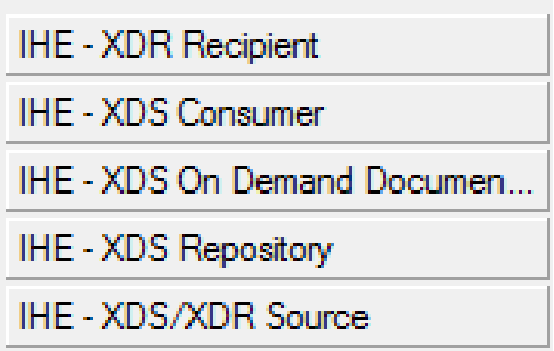


Edit Operations for Web Services Client



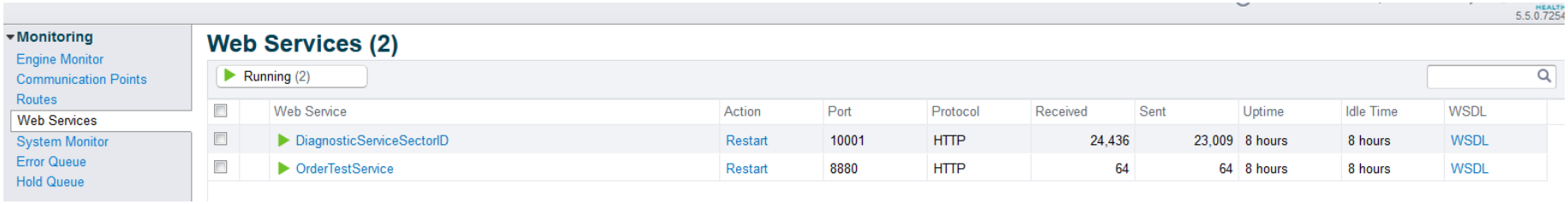
IHE Toolkit

- Optional add-on to Rhapsody 5, which simplifies the understanding and implementation of IHE-based regional integration efforts
- The IHE Toolkit includes:
  - Five new filters and one new communication point
  - Sample configurations showing the usage of typical IHE scenarios
  - Documentation explaining IHE
- Allows users to rapidly IHE enable your products or network without costly development or needing to understand the complexity of IHE specifications
- Keep up to date with the latest healthcare standards with new versions of the IHE Toolkit



Monitoring Web Services in Rhapsody

- Monitor web services in web management console
- Access to WSDL
- Start/Stop Webserver
- Messages Sent and Received
- How long the Web service has been idle



DEMO

GALEN HEALTHCARE SOLUTIONS

For the Galen take on industry news, EHR developments, interesting solutions we have developed, and new Galen products check out our blog…

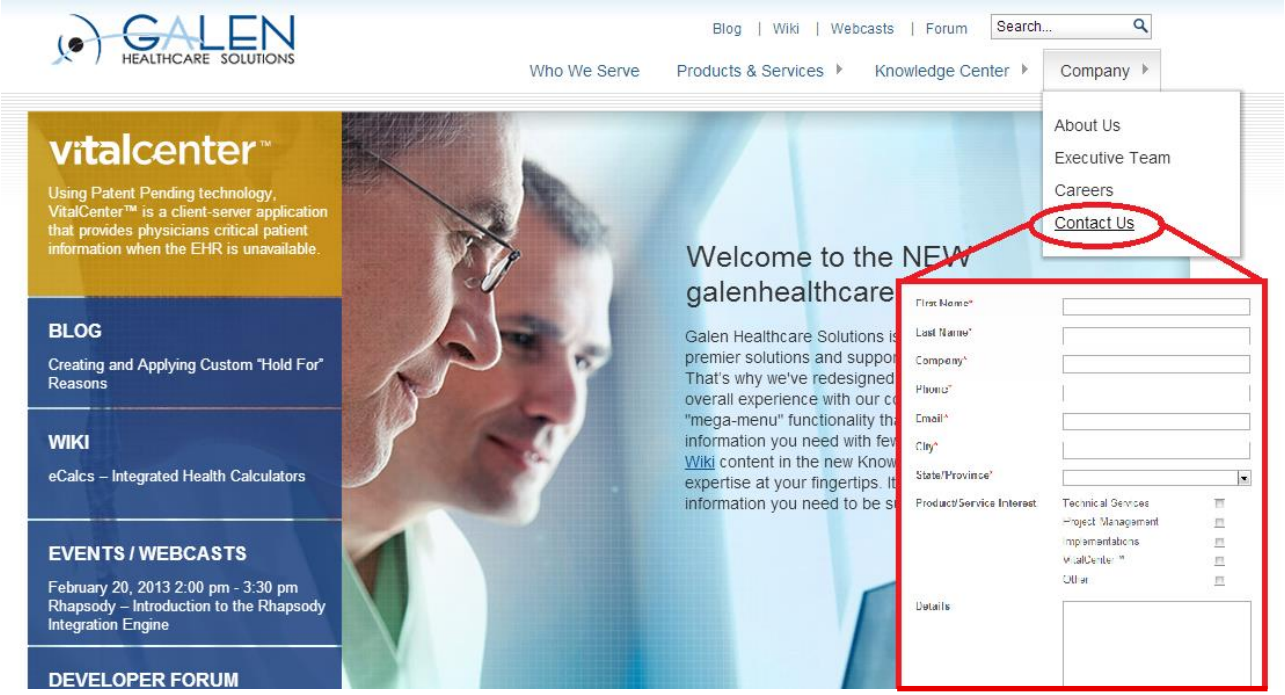
blog.galenhealthcare.com

vitalcenter GALEN HEALTHCARE SOLUTIONS

Thank you for joining us today, for additional assistance….

You can contact us through our ***NEW*** website at

www.galenhealthcare.com



vitalcenter GALEN HEALTHCARE SOLUTIONS