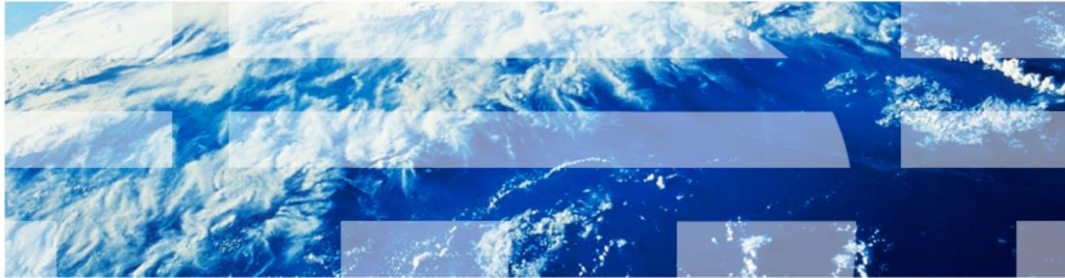


InfoSphere Master Data Management Collaboration Server v10.0

How to create and access the web services using the
user interface



© 2012 IBM Corporation

The presentation explains how to create and access the web services using the user interface.

Terminology

- Aim
 - To create and access a sample web service on InfoSphere® Master Data Management Collaboration Server user interface
- Terminology
 - MDMCS
 - TOP
 - WSDL
 - SOAP
 - Axis
 - UI
 - DOCUMENT_LITERAL
 - XML, XML Schema, XSD, and xmlns
- Assumption
 - You have a working MDMCS instance
- What is not covered
 - Comprehensive details on programming or implementation of web service



The aim of this presentation is to familiarize you with the processes of creating and accessing the web services using the user interface. Through the course of this presentation, certain terminology is used that are unique to the product and web services. Some of the terminology includes MDMCS, which is an acronym for IBM InfoSphere Master Data Management Collaboration Server.

TOP is an environment variable pointing to the installation directory of an MDMCS instance.

Extensible Markup Language, referred to as XML, is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

Web Services Description Language, referred to as WSDL, is written in XML and is used to describe what a web service is designed to do.

Simple Object Access Protocol, referred to as SOAP, is a transport protocol that sends XML messages using HTTP, also referred to as Hyper Text Transfer Protocol, which runs on top of TCP, referred to as Transfer Control Protocol, typically on port 80.

Axis is an XML based, open source framework for developing web services. It is a global standard and currently MDMCS supports Axis 1.4.

UI stands for user interface. In this presentation, the user interface that has been used is a web browser, Internet Explorer. In MDMCS, it is called "Web Service Console".

A WSDL binding describes how the service is bound to a messaging protocol and can be either a Remote Procedure Call, or RPC style binding or a document style binding and one of the models is *document literal style*. The *style* dictates how to translate a WSDL binding to a SOAP message and the *literal* is for the WSDL-to-SOAP mapping.

XML schema is another term used in this presentation. An XML schema is a description of a type of XML document.

XML Schema Definition, referred to XSD, can be used to express a set of rules to which an XML document must conform in order to be considered 'valid' according to that schema.

XML Namespaces, referred to as xmlns, provide a method to avoid element name conflicts. When using prefixes in XML, a so-called namespace for the prefix must be defined. The namespace is defined by the *xmlns* attribute in the start tag of an element.

In this presentation, it is assumed that you already have a working InfoSphere MDMCS instance with the data populated in its database (for example, spec, catalog, items, and so on).

This presentation is not meant to serve as a comprehensive programming or implementation guide to help you implement web services. Rather, it is meant to make you familiar with the whole process of creating and accessing the web services using the user interface.

Introduction

- InfoSphere MDM Collaboration Server user interface, *Web Service Console*, provides ability to create and manage web services
- Sample scenarios of when to use this presentation
 - Used by quality assurance team to test product web service component
 - As educational material
 - To troubleshoot product web service component

A web service that you create in InfoSphere MDMCS is based on WSDL and works with standard SOAP over HTTP protocol.

InfoSphere MDMCS supports SOAP over HTTP web services by using Axis, an open source implementation of SOAP. You can host both RPC or encoded and document or literal style web services within the InfoSphere MDMCS application server, which allows other external clients or applications to interact within InfoSphere MDMCS exactly as required by external clients. You can write the web services either by using scripts or in Java.

You can host simple web services to access MDMCS system entities without the need for extensive familiarity with Axis and other web service tools. In addition, experienced web service developers can host more complex web service scenarios by handling the deployment details themselves.

To deploy the web service in InfoSphere MDMCS you can use a user interface, Java, or script.

The *Web Service Console* is an InfoSphere MDMCS user interface that provides the ability to create and manage web services used by SOAP over HTTP. Your quality assurance team can use this presentation when testing the product web service component after installing the InfoSphere MDMCS product in a new environment. This presentation can be used to educate a user on how to use the MDMCS product, including the web service component. This presentation can also be used to perform a quick test to troubleshoot an issue that may rise while using the product web service component.

Reference

Title	Link
Product information center	http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/index.jsp
Web Service Console	http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.pim.con.doc/panel_help/pim_hlp_chcm_webservicecon.html
Debugging web services	http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.pim.app.doc/webservices/pim_con_webservdebug.html
Known limitations for document literal web services	http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.pim.app.doc/webservices/pim_con_knownlim.html
Understanding WSDL file structures	http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.md mhs.dev.platform.doc/references/r_wsdlfilestructures.html
Web services within InfoSphere MDM Collaboration Server	http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.pim.app.doc/webservices/pim_tsk_dev_webservices.html
Web services outside of InfoSphere MDM Collaboration Server	http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.pim.app.doc/webservices/pim_tsk_implementcustomhostws.html
Document literal style web services	http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.pim.app.doc/webservices/pim_con_docliteralstyle.html

Further details of implementing and deploying web services are beyond the scope of this presentation. Review any of the links displayed on this slide for further details.

Overview

- Enable web service in InfoSphere MDM Collaboration Server instance
- Create a new web service using 'Web Service Console'
- Access web service on a web browser

In this presentation, you are walked through the steps on how to create and access a simple web service using the user interface that is of document/literal style and is written using script. You must have a working InfoSphere MDMCS instance with the data populated (for example: specs, catalog, hierarchy, items, and so on).

If you have a sample WSDL for a document literal type SOAP binding and an InfoSphere MDMCS script implemented for it, see the steps described in this presentation to create and access the web service using the user interface.

The example used in this presentation has a WSDL and InfoSphere MDMCS script implemented for a simple web service. The script has a method, `getListOfMatchingItems`, that returns a list of matching items from an existing catalog for the given attribute path and the attribute value.

Enabling web service

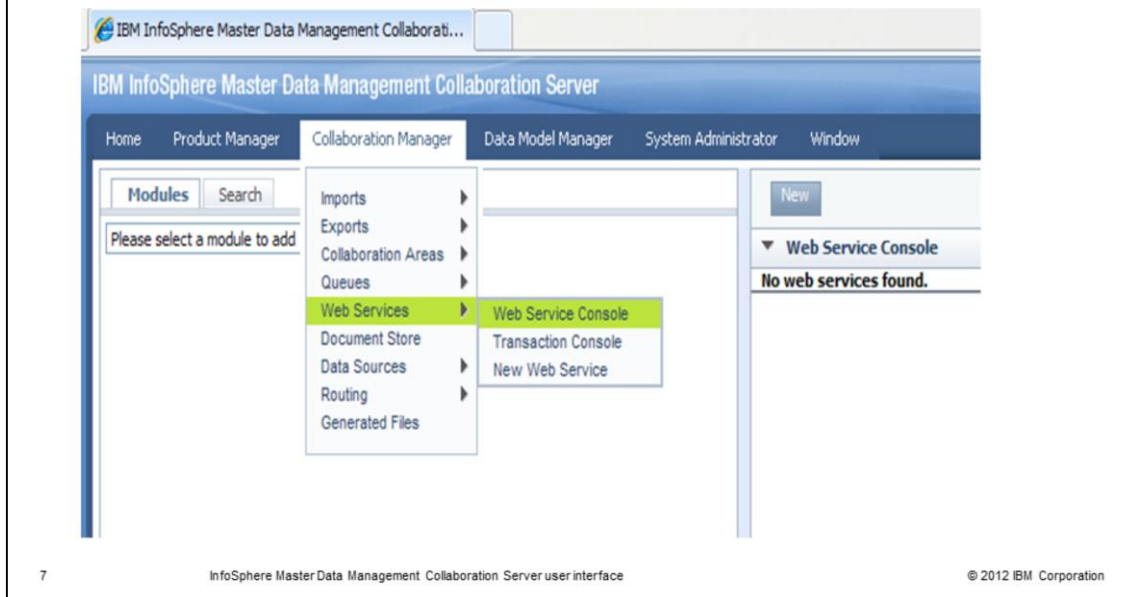
- Define value for properties in `$TOP/etc/default/common.properties`
 - `soap_company`
 - `soap_user`
 - `product_center_url`

For this topic, to enable web service, you have to modify the parameters in the `$TOP/etc/default/common.properties` file. These include the `soap_company` parameter which defines the company credentials. The `soap_user` parameter, which defines the user credentials. Note: When SOAP services are run, they use this company and user to access database and run scripts.

You also need to modify the `product_center_url` parameter, which defines the fully-qualified “web address”, including the port number of the website where you should point your web browser to access your InfoSphere MDMCS instance.

Web Services Console (the MDMCS UI)

- To access Web Service Console
 - Collaboration Manager > Web Services > Web Service Console



InfoSphere MDMCS Java web services support allows hosting of simple web service, without requiring complex setup. In addition, it allows experienced web service developers to host more complex services, by handling the deployment details themselves.

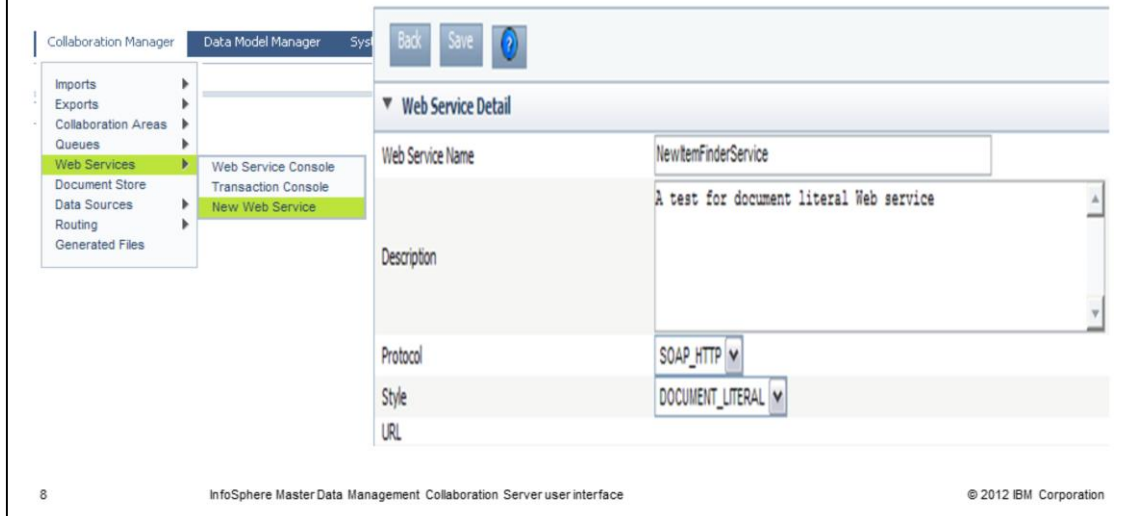
The *Web Service Console* provides the ability to create and manage web services used by SOAP over HTTP. The *Web Service Console* displays the web services as rows in a table.

To access the *Web Services Console*, click *Collaboration Manager*, then *Web Services* then *Web Service Console*. You can create a new web service, view details of a web service, or delete a web service from the *Web Service Console*.

Steps to implement and deploy a web service using UI (1 of 5)

- Implementing a simple web service

http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.pim.app.doc/webservice/pim_tsk_excreatewebserv.html



Simple web services are the ones that use only simple data types such as string and int. You can deploy simple web services on InfoSphere MDMCS without any customization. All that is required is to go to the web service definition page and populate it with the implementation code with either Java or a native application script.

Note: You must define a value for the "soap_company", "soap_user", and "product_center_url" in the \$TOP/etc/default/common.properties file.

First, click *Collaboration Manager*, then *Web Services*, then *New Web Service* as shown in the screen capture displayed on this slide. Next, enter the values for the fields as shown in the screen capture. For example, in the *Web Service Name* field, enter *NewItemFinderService*. Add a description in the *Description* field. For example, *a test for document literal web service*. In the *Protocol* field, enter *SOAP_HTTP*. In the *Style* field, enter *DOCUMENT_LITERAL*. The *URL* field is blank initially, but when the web service is saved, it will show as,

`http://<MDMCS_server_name>:<MDMCS_port_number>/services/NewItemFinderService.`

Steps to implement and deploy a web service using UI (2 of 5)

The screenshot shows a web service configuration interface. On the left, there are two labels: 'WSDL URL' and 'WSDL'. The 'WSDL URL' field is currently blank. The 'WSDL' field contains the following XML content:

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="http://samples.service.pim.ibm.com"
  xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:apachesoap="http://xml.apache.org/xml-soap"
  xmlns:impl="http://samples.service.pim.ibm.com"
  xmlns:intf="http://samples.service.pim.ibm.com"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:tns="http://samples.service.pim.ibm.com"
  xmlns:wsi="http://schemas.xmlsoap.org/wsdl/wsi/"
  xmlns:wsoap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <wsdl:types>
    <schema targetNamespace="http://samples.service.pim.ibm.com"
      xmlns="http://www.w3.org/2001/XMLSchema">
      <element name="getListOfMatchingItems">
        <complexType>
          <sequence>
            <element name="catalogName"
              type="xsd:string" />
            <element name="attributePath" />
          </sequence>
        </complexType>
      </element>
    </schema>
  </wsdl:types>
</wsdl:definitions>
```

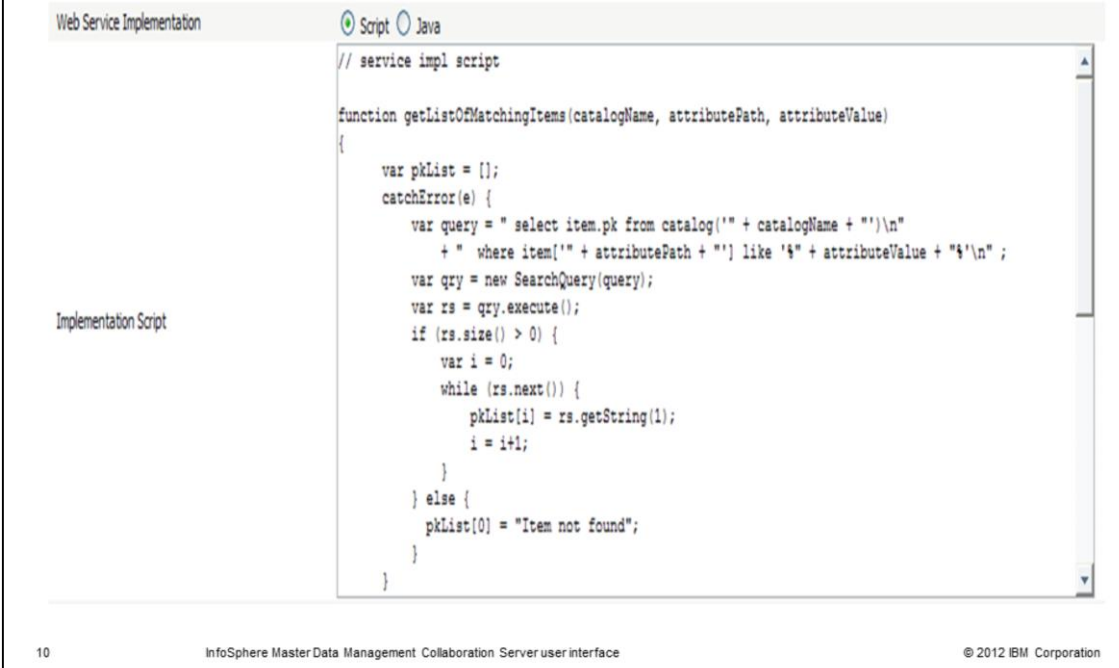
At the bottom of the interface, there is a page number '9', the text 'InfoSphere Master Data Management Collaboration Server user interface', and the copyright notice '© 2012 IBM Corporation'.

The *WSDL URL* field is blank initially, but when the web service is saved, it will show as:

`http://<MDMCS_server_name>:<MDMCS_port_number>/services/NewItemFinderService?wsdl`. In the WSDL field, specify the WSDL XML content as shown in the screen capture displayed on this slide.

Ensure you enter the correct service name, `MDMCS_server_name`, and `MDMCS_port_number` in the `<wsdlsoap:address/>` element.

Steps to implement and deploy a web service using UI (3 of 5)



```
// service impl script

function getListOfMatchingItems(catalogName, attributePath, attributeValue)
{
    var pkList = [];
    catchError(e) {
        var query = " select item.pk from catalog('" + catalogName + "')\n"
            + " where item['" + attributePath + "' ] like '%" + attributeValue + "%'\n";
        var qry = new SearchQuery(query);
        var rs = qry.execute();
        if (rs.size() > 0) {
            var i = 0;
            while (rs.next()) {
                pkList[i] = rs.getString(1);
                i = i+1;
            }
        } else {
            pkList[0] = "Item not found";
        }
    }
}
```

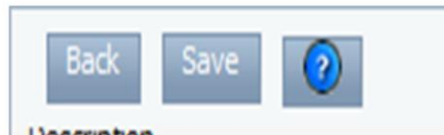
10 InfoSphere Master Data Management Collaboration Server user interface © 2012 IBM Corporation

Next, select a *Web Service Implementation*. In this example, select *Script*. In the *Implementation Script* field, specify the script for the web service as shown in the screen capture.

The script has a function `getListOfMatchingItems` that requires three attribute values; the Catalog Name, the Attribute Path, and the Attribute Value.

Steps to implement and deploy a web service using UI (4 of 5)

Store requests	<input checked="" type="checkbox"/>
Store replies	<input checked="" type="checkbox"/>
Deployed	<input checked="" type="checkbox"/>
Authentication Required	<input type="checkbox"/>
Skip validations of SOAP requests	<input checked="" type="checkbox"/>
Skip validations of SOAP responses	<input checked="" type="checkbox"/>



Specify the remaining values as shown in the screen capture. Select *Store requests* so that you can view the request history from the transaction console. Select *Store replies* so that you can view the response history from the transaction console. Select *Deployed* to deploy the web service. The web service is available to you only if you check this option. Select *Skip validations of SOAP requests* and *Skip validations of SOAP responses*. Finally, click the *Save* button located on the top of the *Web Service Console* screen.

Steps to implement and deploy a web service using UI (5 of 5)

```

<?xml version='1.0' encoding='UTF-8' ?>
- <soapenv:Envelope xmlns:soapenv='http://schemas.xmlsoap.org/soap/envelope/' xmlns:xsd='http://www.w3.org/2001/XMLSchema' xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'>
- <soapenv:Header>
  <ns1:sessionID soapenv:actor='http://schemas.xmlsoap.org/soap/actor/next' soapenv:mustUnderstand='0' href='#id0' xmlns:ns1='http://xml.apache.org/axis/session' />
</soapenv:Header>
- <soapenv:Body>
- <getListOfMatchingItemsResponse xmlns='?'>
  <getListOfMatchingItemsReturn>100</getListOfMatchingItemsReturn>
  <getListOfMatchingItemsReturn>101</getListOfMatchingItemsReturn>
</getListOfMatchingItemsResponse>
  <multiref id='id0' soapenc:root='0' soapenv:encodingStyle='http://schemas.xmlsoap.org/soap/encoding/' xsi:type='soapenc:long' xmlns:soapenc='http://schemas.xmlsoap.org/soap/encoding/'>5021333233115121305</multiref>
</soapenv:Body>
</soapenv:Envelope>

```

12

InfoSphere Master Data Management Collaboration Server user interface

© 2012 IBM Corporation

Invoke the web service in the web browser “Web address” field.

`http://<MDMCS_server_name>:<MDMCS_port_number>/services/NewItemFinderService?method=getListOfMatchingItems&catalogName=NewTestCatalog&attributePath=NewTestPrimarySpec/SKU&attributeValue=1.`

The “Web address” includes the function/method *getListOfMatchingItems* of web service *NewItemFinderService* with attributes of Catalog Name which is *NewTestCatalog*, the Attribute Path of *NewTestPrimarySpec/SKU*, and the Attribute Value of one.

The web service that you invoked in the previous step, if matching items are found that contains the number one, returns the SOAP message on the web browser as shown in the screen capture displayed on this slide.

Note that if you provide the catalog and its items incorrectly or from a different company than the one that is set in the \$TOP/etc/default/common.properties, web service returns a fatal error SOAP message.

Debugging web services

- http://publib.boulder.ibm.com/infocenter/mdm/v10r0m0/topic/com.ibm.pim.app.doc/webservices/pim_con_webservdebug.html
- Open a PMR
<https://www-946.ibm.com/support/servicerequest/>

The Information Center also creates some very useful suggestions on how to debug a web service if you encounter any issues. If you cannot narrow down the root cause using these suggestions, open a PMR and the IBM Support team is glad to assist you.

Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, and InfoSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2012. All rights reserved.