

Oracle® Enterprise Data Quality for Product Data

Web Service Access to Oracle DataLens Servers Interface Guide

Release 5.6.2

E25661-01

November 2011

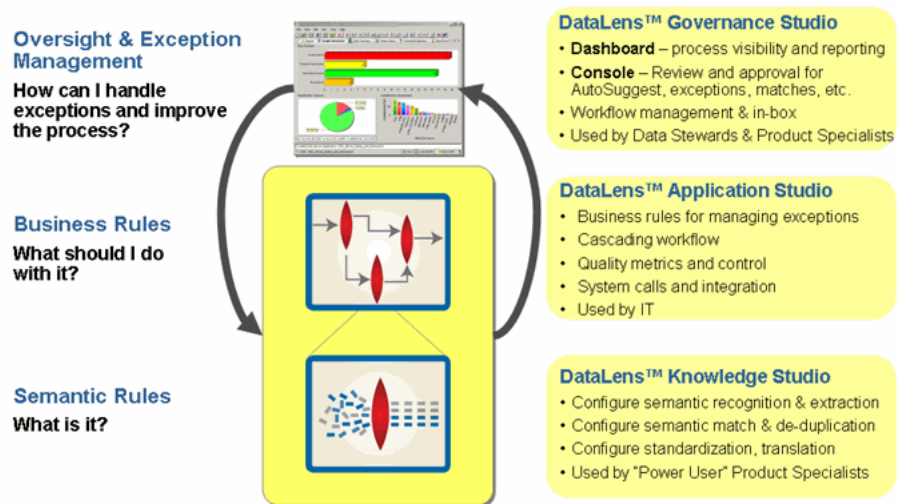
This document describes the Oracle Enterprise Data Quality for Product Data Web Access to Oracle DataLens Servers and contains the following:

- ["Overview"](#) on page 1
- ["Obtaining the EDQP WSDL Document"](#) on page 2
- ["Communicating with a DSA"](#) on page 2

Overview

Oracle DataLens Server is built on industry-leading DataLens™ Technology to standardize, match, enrich, and correct product data from different sources and systems. The core DataLens Technology uses patented semantic technology designed from the ground up to tackle the extreme variability typical of product data.

Oracle Enterprise Data Quality for Product Data, formerly Oracle Product Data Quality, uses three core DataLens Technology modules: Governance Studio, Knowledge Studio, and Application Studio. The following figure illustrates the process flow of these modules.



Web service access is provided to the Oracle DataLens Server as a Document-Literal Web Service. The following sections describe how to set up this access and the Web Service Operations that you can use to interface with Enterprise DQ for Product (EDQP) Data Service Applications (DSAs) including examples.

Obtaining the EDQP WSDL Document

To integrate with an Enterprise DQ for Product DSA as a Web Service, you need software that will interface with the Oracle DataLens Web Services. There are a number of tools available to generate this software from a Web Services Description Language (WSDL) document, which is an XML format for describing network services. You can view the WSDL for the Oracle DataLens Web Services by using an Internet browser.

Browse to the following Web site:

<http://localhost:2229/datalens/ws/Processor?wsdl>

Note: The host name and port number may differ.

This displays the WSDL document, which can be saved by right-clicking in the document in the browser, selecting **View Source**, and then saving the file from within your browser. For instance, the file can be saved as `Processor.wsdl`.

Client Web Service Software

For your Web Service clients, client-side software can be generated from this WSDL document to access the Oracle DataLens Server.

Communicating with a DSA

There is a single Service called "ProcessorService", which uses a PortType (Interface) called "Processor".

The following three Oracle DataLens Web Services Operations can be used to process data and communicate with a DSA by creating a job:

`ProcessorList`

This requires an input array of strings, or multiple lines of data, and returns an output array of strings. One DSA job is created.

`ProcessorOneLine`

This requires a single string of input, or one line of data, and returns a single string of output. If you have several lines of data, use `ProcessorList` to process them because a single DSA job is created rather than the multiple DSA jobs that would be created using the `ProcessorOneLine` operation for each line of data. For example, if you have four lines of data to process, the `ProcessorList` creates one DSA job while the `ProcessorOneLine` creates four separate DSA jobs.

`ProcessorDB`

This requires a database query (defined in the Transform Map) and returns a job Id of the DSA Job that handled the request. The output is assumed to be a database update, e-mail, or FTP. One DSA job is created.

Points to Consider

- Reviewing the EDQP WSDL will help you to understand what operations and parameters can be used to communicate with DSAs.
- Simple Object Access Protocol (SOAP) is used to communicate with DSAs. For more information about SOAP, see the W3C SOAP 1.1 Web site:

<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

- You must provide the exact input fields that are defined in the DSA to ensure proper operation and avoid errors. For example, if the input nodes in the DSA are `Id`, `Category`, and `Description`, you must provide all three fields in your SOAP processing operation.
- You must separate input fields using a defined separator; the tab character "`^`" is recommended. For example, using the tab character to separate the `Id`, `Category`, and `Description` input fields:

```
<lineOfData>3^2334^res, 37ohm, 39watt, 30%</lineOfData>
```

- The best practice is to define all of the operation parameters (even those that may be labeled as optional) to avoid errors. The EDQP WSDL and the DSA define the necessary parameters not the SOAP UI you are using; some SOAP UIs erroneously mark parameters as optional as in the examples in the following sections.

The SOAP request structure is the same for each type of request though some of the parameters may not apply. For example, `processListRequest` and `processOneLineRequest` do not use the `dbparameters` parameter; `processDBRequest` does not use `line` or `linesofdata` parameters. The parameters used are dependent on the operation request.

Additionally, defining all parameters allows the most information possible to be related to the DSA job, and then reviewed and verified in the Oracle DataLens Server Administration Web Page.

processListRequest and processOneLineRequest Operations

The difference between these two operations is that `processListRequest` takes an array of lines and `processLineRequest` takes a single line of data as a string. The transformed data is returned. This call is synchronous. The parameters are as follows:

dsaName

The name of the DSA to run on the Oracle DataLens Server.

line Or linesOfData

One line or multiple lines of input or output data respectively.

dbParameters

Not used.

priority

The priority the DSA job will be given on the Oracle DataLens Server for processing; the default is Medium. The priority is set numerically as follows:

3 for High
2 for Medium (the default)
1 for Low

runtimeLocale

A run time locale must be entered; the default locale of `en_US` should be entered if not known.

fieldSeparatorChar

A character that will be used to separate fields; the tab character "`^`" is recommended.

application

This application name initiated the client request to the server. This name is used to accumulate server statistics on the Oracle DataLens Server Administration Web Pages.

description

A description of a particular job and relates to a DSA Job ID.

The Oracle DataLens Server Administration DSA Job Status Web page illustrates the DSA Job ID for both the processListRequest and processLineRequest operations:

ORACLE Enterprise Data Quality for Product Data

Jobs
 Job Status
 Run a Job
 Schedule a Job
 Edit Scheduled Jobs

Server
 Data Lenses
 Data Lens Groups
 DSAs
 Reports

Configuration
 Platform Topology
 Configuration
 Server Groups
 Role Administration
 User Administration

Database Connections
 Web Services
 FTP Connections

User Access
 Data Lenses
 DSAs

DSA Job Status since Fri Sep 09 12:01:25 MDT 2011

Active DSA Jobs (Running)

Job ID	Owner	Status	Start	Input Line Count	Description	Server	Priority	Action
There are currently 0 active job(s) running								

Pending DSA Jobs (in Queue)

Job ID	Owner	Status	Start	Input Line Count	Description	Server	Priority	Action
There are currently 0 pending job(s) waiting								

DSA Job History (Completed, Canceled, Failed)

Job ID	Owner	Status	Start	Duration	Input Line Count	Description	Server	Priority
81	ClientCall	Completed	2011-09-12 12:01:17	0:0:0	2	Test List Job	LVALLAD-LAP	High
80	ClientCall	Completed	2011-09-12 11:56:16	0:0:0	1	Test Job	LVALLAD-LAP	High

There are currently 2 finished job(s) archived

Multiple Line Request
 Single Line Request

Additionally, the priority parameter that was set is displayed in the Priority column.

The following DSA Job Details Web page illustrates the operation parameters supplied in the processListRequest operation:

ORACLE Enterprise Data Quality for Product Data

Jobs
 Job Status
 Run a Job
 Schedule a Job
 Edit Scheduled Jobs

Server
 Data Lenses
 Data Lens Groups
 DSAs
 Reports

Configuration
 Platform Topology
 Configuration
 Server Groups
 Role Administration
 User Administration

Database Connections
 Web Services
 FTP Connections

User Access
 Data Lenses
 DSAs

Dashboard

DSA Job Details for Job 81

Property	Value
Job ID	81
Status	Completed
Definition	My_DSA
Description	Test List Job
Start Time	September 12, 2011 12:01:17 PM MDT
Finish Time	September 12, 2011 12:01:18 PM MDT
Duration	0:0:0
Created by	ClientCall
Input Line Count	2
Output Line Count (Good)	2
Output Line Count (Not Processed)	0
Output Path/File	Not Used
Run-time Locale	en_US

DSA Step Details

Step Name	Type	Status	Description	Start Time	End Time	Duration	Input Line Count	Output Line Count	Comment
Text	Input	Completed	Text Data Input	2011-09-12 12:01:17.909	2011-09-12 12:01:17.933	0:0:0	0	2	
pass_data	Processing	Completed	null	2011-09-12 12:01:17.946	2011-09-12 12:01:17.966	0:0:0	2	2	
results	Output	Results Retrieved	null	2011-09-12 12:01:17.983	2011-09-12 12:01:17.999	0:0:0	2	2	

The following sections provide examples of request and response operations for both the processListRequest and processLineRequest operations:

SOAP Doc-Lit Multi-Line ProcessList Request Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ws="http://www.oracle.com/ws">
<soapenv:Header/>
<soapenv:Body>
```

```

<ws:processList>
  <dsaName>sampleDSA</dsaName>
  <!--Zero or more repetitions:-->
  <linesOfData>1^res, 17ohm, 19watt, 10%</linesOfData>
  <linesOfData>2^res, 27ohm, 29watt, 20%</linesOfData>
  <linesOfData>3^res, 37ohm, 39watt, 30%</linesOfData>
  <!--Zero or more repetitions:-->
  <dbParameters>?</dbParameters>
  <!--Optional:-->
  <priority>1</priority>
  <!--Optional:-->
  <runtimeLocale>en_US</runtimeLocale>
  <!--Optional:-->
  <fieldSeparator>^</fieldSeparator>
  <!--Optional:-->
  <application>ClientCall</application>
  <!--Optional:-->
  <description>Example list Doc-Lit client call</description>
</ws:processList>
</soapenv:Body>
</soapenv:Envelope>

```

SOAP Doc-Lit Multi-Line ProcessList Response Example

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
<S:Body>
  <ns2:processListResponse xmlns:ns2="http://www.oracle.com/ws">
    <return>1^Resistor, 17 Ohm, 10%, 19 Watt^32121609^Fixed
resistors^Resistor^Item_Name^RESISTOR^Item_Type^^Resistance^17
OHM^Power^19^Tolerance^10%^Package_Size^^Construction^^Mounting^^Pin_Count
^^For_sale_packaging^</return>
    <return>2^Resistor, 27 Ohm, 20%, 29 Watt^32121609^Fixed
resistors^Resistor^Item_Name^RESISTOR^Item_Type^^Resistance^27
OHM^Power^29^Tolerance^20%^Package_Size^^Construction^^Mounting^^Pin_Count
^^For_sale_packaging^</return>
    <return>3^Resistor, 37 Ohm, 30%, 39 Watt^32121609^Fixed
resistors^Resistor^Item_Name^RESISTOR^Item_Type^^Resistance^37
OHM^Power^39^Tolerance^30%^Package_Size^^Construction^^Mounting^^Pin_Count
^^For_sale_packaging^</return>
  </ns2:processListResponse>
</S:Body>
</S:Envelope>

```

SOAP Document-Literal One Line Request Example

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ws="http://www.oracle.com/ws">
<soapenv:Header/>
  <soapenv:Body>
    <ws:processOneLine>
      <dsaName>sampleDSA</dsaName>
      <!--Optional:-->
      <line>1^res, 17ohm, 19watt, 20%</line>
      <!--Zero or more repetitions:-->
      <dbParameters>?</dbParameters>
      <!--Optional:-->
      <priority>1</priority>
      <!--Optional:-->

```

```

        <runtimeLocale>en_US</runtimeLocale>
        <!--Optional:-->
        <fieldSeparator>^</fieldSeparator>
        <!--Optional:-->
        <application>ClientCall</application>
        <!--Optional:-->
        <description>Example Doc-Lit client call</description>
    </ws:processOneLine>
</soapenv:Body>
</soapenv:Envelope>

```

SOAP Document-Literal One Line Response Example

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
<S:Body>
    <ns2:processOneLineResponse xmlns:ns2="http://www.oracle.com/ws">
        <return>1^Resistor, 17 Ohm, 20%, 19 Watt^32121609^Fixed
resistors^Resistor^Item_Name^RESISTOR^Item_Type^^Resistance^17
OHM^Power^19^Tolerance^20%^Package_Size^^Construction^^Mounting^^Pin_Count
^^For_sale_packaging^</return>
    </ns2:processOneLineResponse>
</S:Body>
</S:Envelope>

```

processDBRequest

This call takes the database parameters as input and returns the DSA Job ID. This call is asynchronous. The parameters are as follows and are the same as described in ["processListRequest and processOneLineRequest Operations"](#) on page 1-3:

```

dsaName
dbParameters
priority
runtimeLocale
fieldSeparatorChar
application
description

```

However, the dbParameters parameter is the value that is input to the DSA database query and is related to the DSA job run.

The following sections provide example request and response operations for the processDb operation:

SOAP Document-Literal processDb Request Example

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ws="http://www.oracle.com/ws">
<soapenv:Header/>
    <soapenv:Body>
        <ws:processDB>
            <dsaName>SampleDSADbInput</dsaName>
            <!--Zero or more repetitions:-->
            <dbParameters>1</dbParameters>
            <dbParameters>2</dbParameters>
            <!--Optional:-->
            <priority>2</priority>
            <!--Optional:-->

```

```

        <runtimeLocale>en_US</runtimeLocale>
        <!--Optional:-->
        <fieldSeparator>|</fieldSeparator>
        <!--Optional:-->
        <application>ClientCall</application>
        <!--Optional:-->
        <description>Example Db Input Doc-Lit client call</description>
    </ws:processDB>
</soapenv:Body>
</soapenv:Envelope>

```

The field separator will be used when the output from the database job is a text file.

SOAP Document-Literal processDb Response Example

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
<S:Body>
    <ns2:processDBResponse xmlns:ns2="http://www.oracle.com/ws">
        <return>784</return>
    </ns2:processDBResponse>
</S:Body>
</S:Envelope>

```

The return value of the preceding processDb call is the DSA Job ID, in this case 784. This job reads data from the database and updates other fields in the database as in the following example:

DSA Job Details for Job 784									
Property	Value								
Job ID	784								
Status	Completed								
Definition	sampleMapIdDefDBInput								
Description	Example Db Input Doc-Lit client call								
Start Time	January 27, 2010 1:14:56 PM MST								
Finish Time	January 27, 2010 1:14:56 PM MST								
Duration	H:0 M:0 S:0								
Created by	ClientCall								
Input Line Count	0								
Output Line Count (Good)	0								
Output Line Count (Not Processed)	0								
Output Path/File	Not Used								
Run-time Locale	en_US								
DSA Step Details									
Step Name	Type	Status	Description	Start Time	End Time	Duration	Input Line Count	Output Line Count	Comment
DB Input	DB Input	Completed	Input from Database	2010-01-27 13:14:56.177	2010-01-27 13:14:56.183	H:0 M:0 S:0	0	0	
ProcessResistors	Processing	Completed	null	2010-01-27 13:14:56.203	2010-01-27 13:14:56.413	H:0 M:0 S:0	2	2	
updateValue	Processing	Completed	null	2010-01-27 13:14:56.425	2010-01-27 13:14:56.478	H:0 M:0 S:0	2	2	
add_resistors	DB Output	Completed	update className in Db	2010-01-27 13:14:56.483	2010-01-27 13:14:56.491	H:0 M:0 S:0	2	0	

Related Documents

For more information, see the following documents in the documentation set:

- The *Oracle Enterprise Data Quality for Product Data R12 PIM Connector Installation Guide* provides installation and configuration of Enterprise DQ for Product R12 PIM Connector.
- The *Oracle Enterprise Data Quality for Product Data R12 PIM Connector User's Guide* provides highlights of the core process steps and features of Enterprise DQ for Product R12 PIM Connector.
- The *Oracle Enterprise Data Quality for Product Data Oracle DataLens Server Installation Guide* provides detailed Oracle DataLens Server installation instructions.

- The *Oracle Enterprise Data Quality for Product Data Oracle DataLens Server Administration Guide* provides information about installing and managing an Oracle DataLens Server.
- The *Oracle Enterprise Data Quality for Product Data COM Interface Guide* provides information about installing and using the Oracle DataLens Server COM APIs.
- The *Oracle Enterprise Data Quality for Product Data Java Interface Guide* provides information about installing and using the Oracle DataLens Server Java APIs.
- The *Oracle Enterprise Data Quality for Product Data Glossary* provides definitions to commonly used Enterprise DQ for Product technology terms.

See the latest version of this and all documents listed at the Oracle Enterprise Data Quality for Product Data Documentation Web site at:

http://download.oracle.com/docs/cd/E20593_01/index.htm

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Oracle Enterprise Data Quality for Product Data Web Service Access to Oracle DataLens Servers Interface Guide, Release 5.6.2
E25661-01

Copyright © 2011 Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.