Oracle Service Registry - Oracle Enterprise Gateway Integration Guide
Disclaimer

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.
1 Introduction .............................................................................................................. 4
2 Prerequisites for Connecting to OSR................................................................. 4
3 References.............................................................................................................. 4
4 How-to guide.......................................................................................................... 5
5 Adding the Policy Attachment Category Systems to OSR......................... 9
1 Introduction

This guide describes how to configure the Policy Studio to connect to the Oracle Service Registry (OSR) in order to browse service data held in the registry, and also to publish service data to the registry.

The OEG Gateway does not connect to the OSR. The Policy Studio connects to the OSR. However, the Policy Studio must be connected to a Gateway in order to access the registry. This guide applies to OEG Gateway software products, from version 6.1 upwards.

1.1. Oracle Service Registry

Oracle Service Registry is an implementation of v3 of the UDDI (Universal Description, Discovery and Integration) specification. A UDDI registry stores service data and provides a standards-based mechanism to classify and manage services so that they can be located and invoked by applications.

1.2. Setup Used for this Guide:

This guide uses the following products:

- OEG Policy Studio 11.1.1.x
- OEG Gateway 11.1.1.x
- Oracle Service Registry 11.1.1.x

2 Prerequisites for Connecting to OSR

Before connecting to OSR, you must know the following:

1. The hostname and port on which the OSR is running. This document refers to the OSR host as oegeval and the port as 8082.
2. You may need to have access to the machine on which the OSR is running in order to update the WSDL files if the address locations are invalid.
3. If you wish to connect to OSR over SSL you will need the OSR SSL certificate.
4. If you are publishing WSDL to the OSR and including WS-Policy as remote policy expressions, or, reusable policy expressions, you will need access to the web-based OSR Registry Control (e.g. http://oegeval:8082/uddi/web), and an administrator’s login credentials. This is needed in order to add the WS-Policy Attachment category systems, see References.

3 References

<table>
<thead>
<tr>
<th>Name</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Services Policy 1.5 – Attachment</td>
<td><a href="http://www.w3.org/TR/ws-policy-attach/">http://www.w3.org/TR/ws-policy-attach/</a></td>
</tr>
</tbody>
</table>
4 How-to guide

4.1 Check the Registry Service WSDL URLs

Enter the following URL into a browser http://oegeval:8082/uddi/doc/wsdl/uddi_api_v3.wsdl. Change the hostname and port appropriately. You should see the following WSDL.

```xml
<definitions
  name="UDDI_API_V3"
  xmlns="http://schemas.xmlsoap.org/wsdl/
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/
  xmlns:api_v3_binding="urn:uddi-org:api_v3_binding"
  xmlns:tns="urn:uddi-org:api_v3" targetNamespace="urn:uddi-org:api_v3">
  <import
    namespace="urn:uddi-org:api_v3_binding"
    location="uddi_api_v3_binding.wsdl" />
  <service
    name="UDDI_Security_SoapService">
    <port
      name="UDDI_Security_PortType"
      binding="api_v3_binding:UDDI_Security_SoapBinding">
      <soap:address location="http://oegeval:8082/uddi/security" />
    </port>
  </service>
  <service
    name="UDDI_Publication_SoapService">
    <port
      name="UDDI_Publication_PortType"
      binding="api_v3_binding:UDDI_Publication_SoapBinding">
      <soap:address location="http://oegeval:8082/uddi/publishing" />
    </port>
  </service>
  <service
    name="UDDI_Inquiry_SoapService">
    <port
      name="UDDI_Inquiry_PortType"
      binding="api_v3_binding:UDDI_Inquiry_SoapBinding">
      <soap:address location="http://oegeval:8082/uddi/inquiry" />
    </port>
  </service>
</definitions>
```

If http://oegeval:8082/uddi/doc/wsdl/uddi_api_v3.wsdl is not a valid URL, then refer to Appendix A in order to use the OSR Registry Control to find the WSDL URLs. If the WSDL does not contain an entry for each of the following services:-
- UDDI_Inquiry_SoapService
- UDDI_Security_SoapService
- UDDI_Publication_SoapService, then refer to Appendix A in order to use the OSR Registry Control to find the other WSDL URLs.

The `<soap:address>` location attribute may contain urn:unknown-location-uri. If this is the case, the WSDL file must be updated on the OSR machine. If this step is required you will need to login to the machine running OSR and edit the WSDL file so that it contains the correct URLs for accessing the inquiry, security and publishing services. The WSDL documents may be found in /opt/oracle/osr/doc/wsdl, or similar on the OSR machine.
4.2 Connecting to OSR as a UDDI v3 Registry

You connect to a registry in Policy Studio before browsing or publishing. There are a few points at which you may do this. For example, click the **Web Services Repository** node, right-click, and select **Browse UDDI Registry**. You will see the screen below.

![Browse UDDI Registry dialog box](image)

Click the **Add** button to add a new registry connection.
Select **UDDI v3**, and enter the WSDL URLs for the inquiry, publishing and security services that were found in the last step into the **Inquiry URL**, **Publish URL** and **Security URL** fields. (Note if you wish to browse only, you may leave the publishing and security URLs blank). In order to perform any registry interactions that require authentication, (e.g. publishing), you must enter a **Username** and **Password** into the screen above. You can register and create a new account via the OSR Registry Control, refer to the **Oracle Service Registry User's Guide** for more information.

When the connection is configured, click **OK** and click **Next** and proceed to browse or publish. Refer to the OEG User documentation for more information.

You may also connect to the OSR using **UDDI v2** interfaces. Refer to Appendix B for more information.
4.3 SSL Connections to OSR

You may wish to communicate with the registry over SSL. Not all communication needs to be over SSL, for example, you may wish to publish over SSL, while all inquiry calls are done without SSL.

For a UDDI v3 connection, there may be a mix of http and https URLs for the WSDL URLs and also for the service address locations. If any URLs, i.e. WSDL or service address location are https, then we must configure the Policy Studio so that it trusts the OSR SSL certificate.

Assuming mutual authentication is not required, simplest way to get an SSL connection configured between the Policy Studio and OSR is to add the OSR certificate to the Policy Studio’s default trust store, i.e. the cacerts file. Perform the following steps with the Policy Studio:-

- Browse to Certificates
- Click on Create and click on Import Certificate
- Browse to the OSR SSL Certificate file and click on Open
- Tick the Use Subject box next to the Alias field and click on Ok
- The OSR SSL Certificate is now imported into the Certificate store
- It now needs to be added to the JAVA keystore
- Click on Keystore in the Certificate window
- Next to the Keystore field click on the browse button
- Browse to the following file:
  `install_dir/policystudio/jre/security/cacerts`
- Click on Open and enter the Keystore password. Default password is: changeit
- Click Add to Keystore
- Browse to the OSR SSL Certificate imported before and select it and click on OK
- Restart the Policy Studio. You should now be able to connect to OSR over SSL.

If mutual SSL authentication is required, i.e. if the Policy Studio must authenticate to the OSR, then the Policy Studio will need to have an SSL private key and certificate. In this case a keystore containing the Policy Studio’s key and certificate should be created. The Policy Studio must be configured to load this file. Edit the `install_dir/policystudio/policystudio.ini` file and add the following arguments:-

```
-Djavax.net.ssl.keyStore=/home/vordel/osr-client.jks
-Djavax.net.ssl.keyStorePassword=changeit
```

Note that the Policy Studio may also be used to create a new keystore (jks) file. Click New keystore instead of browsing to the cacerts file as described above.

4.4 Using a Proxy

In some cases you may need to communicate with the OSR via a proxy. The proxy settings are configured via the Registry Connection Details screen. The settings are the same regardless of whether the connection is **UDDI v2** or **UDDI v3**. Simply enter the **Proxy Host** and **Proxy Port** in the fields provided. For example:-
If the proxy requires authentication, enter the **Username** and **Password** in the **HTTP Proxy** section.

If any of the communication is over SSL (WSDL URLs or service address location URLs), and you wish it to pass through a proxy, you must configure an **SSL Proxy Host** and **SSL Proxy Port**.

### 5 Adding the Policy Attachment Category Systems to OSR

If you wish to publish WSDL to OSR using the Policy Studio and include WS-Policy as described in the WS-Policy Attachment specification (see References), then you may need to add the WS-Policy Attachment category systems to OSR using the OSR Registry Control. For more
information regarding publishing WSDL to a registry using Policy Studio, please refer to the OEG user documentation.

Open the Registry control in a browser (http://oegeval:8082/uddi/web). You should first of all check if the category system exists already in the OSR. Do this as follows:-

Click the **Browse** tab. You should see the following:-

Click **Filter** and enter http://www.w3.org/ns/ws-policy/remotepolicyreference into the **name** field.
Click **Apply Filter**.

If the **Taxonomy list** tree displays any entries then the category system already exists in the OSR.
Repeat the above steps for the other two category systems used to attach WS-Policy, namely:

- http://www.w3.org/ns/ws-policy/localpolicyreference

If the category systems do not exist in the OSR they must be added (if you wish to publish WS-Policy to the registry). Do this as follows:

- Login to the OSR Registry Control as an administrator user, e.g. with username/password oracle/changeit.
- Click the Manage tab.
- Click the Taxonomy management button. You should see the following screen:

  ![OSR Screen Screenshot](image)

- Click the Add Taxonomy button.
- For the http://www.w3.org/ns/ws-policy/remotepolicyreference category system enter fields as follows:
  1. Name: http://www.w3.org/ns/ws-policy/remotepolicyreference
  2. tModel Key: uddi:w3.org:ws-policy:v1.5:attachment:remotepolicyreference
  3. Description: Category system used for UDDI entities to point to an external Web services policy attachment policy that describes their characteristics. See Web Services Policy 1.5 - Attachment specification for further details.
  4. Categorization: Check the categorization box.
  5. Compatibility: Check all boxes.
- Click Save taxonomy button.
Repeat to add 2 more category systems as follows:

For the [http://www.w3.org/ns/ws-policy/localpolicyreference](http://www.w3.org/ns/ws-policy/localpolicyreference) category system enter fields as follows:

1. **Name**: `http://www.w3.org/ns/ws-policy/localpolicyreference`
2. **tModel Key**: `uddi:w3.org:ws-policy:v1.5:attachment:localpolicyreference`
3. **Description**: Category system used for UDDI entities to point to a Web services policy policy expression tModel that describes their characteristics. See Web Services Policy 1.5 - Attachment specification for further details.
4. **Categorization**: Check the **categorization** box.
5. **Compatibility**: Check all boxes.

For the [http://www.w3.org/ns/ws-policy/policytypes](http://www.w3.org/ns/ws-policy/policytypes) category system enter fields as follows:

1. **Name**: `http://www.w3.org/ns/ws-policy/policytypes`
2. **tModel Key**: `uddi:w3.org:ws-policy:v1.5:attachment:policytypes`
3. **Description**: Web services policy types category system used for UDDI tModels to characterize them as Web services policy-based policy expressions.
4. **Categorization:** Check the **categorization** box.

5. **Compatibility:** Check all boxes.

### 6. Conclusion

This document demonstrates how to configure the OEG Policy Studio to connect to OSR. The Policy Studio can be used to find services definitions in the registry, and also to publish services to the registry. These features are out of the scope of this document, but are covered in other documents that are available on ORACLE at [http://www.oracle.com](http://www.oracle.com).
Appendix A - Using OSR Registry Control to Find WSDL URLs

In order to configure a connection to the registry using UDDI v3 interfaces, the Policy Studio requires the WSDL URLs of the registry services it uses, i.e. inquiry, publishing and security. The Policy Studio will use the address location URLs listed in the WSDL in order to invoke these services.

The easiest way to get the WSDL service URLs is to look them up the registry. Using your browser open the OSR Registry Control using the following URL http://oegeval:8082/uddi/web.

Click the Search tab.

Enter UDDI_Inquiry_SoapService in the Service name field, and click Find Service.

You will be brought to a Search results page as shown below.

Click on any of UDDI_Inquiry_SoapService listed in the name column. You will then see the service details which include the WSDL URL in the WSDL location field.
Load the WSDL into your browser. You should see the following WSDL.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<definitions name="UDDI_API_V3" xmlns="http://schemas.xmlsoap.org/wsdl/

xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/

xmlns:api_v3_binding="urn:uddi-org:api_v3_binding"

xmlns:tns="urn:uddi-org:api_v3" targetNamespace="urn:uddi-org:api_v3">
<import namespace="urn:uddi-org:api_v3_binding" location="api_v3_binding.wsdl" />

<service name="UDDI_Inquiry_SoapService">
  <port name="UDDI_Inquiry_PortType" binding="api_v3_binding:UDDI_Inquiry_SoapBinding">
    <soap:address location="http://oegeval:8082/uddi/inquiry" />
  </port>
</service>

<service name="UDDI_Publication_SoapService">
  <port name="UDDI_Publication_PortType" binding="api_v3_binding:UDDI_Publication_SoapBinding">
    <soap:address location="http://oegeval:8082/uddi/publishing" />
  </port>
</service>

<service name="UDDI_Security_SoapService">
  <port name="UDDI_Security_PortType" binding="api_v3_binding:UDDI_Security_SoapBinding">
    <soap:address location="http://oegeval:8082/uddi/security" />
  </port>
</service>
</service>
</definitions>
```
This WSDL has the 3 service definitions, one for each of UDDI_Inquiry_SoapService, UDDI_Security_SoapService, UDDI_Publication_SoapService. If this is the case, then you can use the same WSDL URL for each of the services when you configure your connection to the registry in the Policy Studio. If a service is missing, then you will need to repeat the procedure above in order to find the URL of the WSDL that defines that service, (unless you are not using the service, i.e. you are browsing and not publishing).

**Appendix B - Connecting to OSR as a UDDI v2 Registry**

OSR is a fully compliant UDDI v3 registry, therefore it is recommended that you use the UDDI v3 services to talk to it. However, it is also possible to configure the Policy Studio to use the UDDI v2 interfaces to talk to OSR.

For a UDDI v2 connection, select the **UDDI v2** option. The URLs entered for UDDI v2 connections are not WSDL URLs, instead they are the actual service URLs. Note that the **Security URL** is not used for UDDI v2.

In order to perform any registry interactions that require authentication, (e.g. publishing), the **Type** should be set to **UDDI_GET_AUTHTOKEN**.