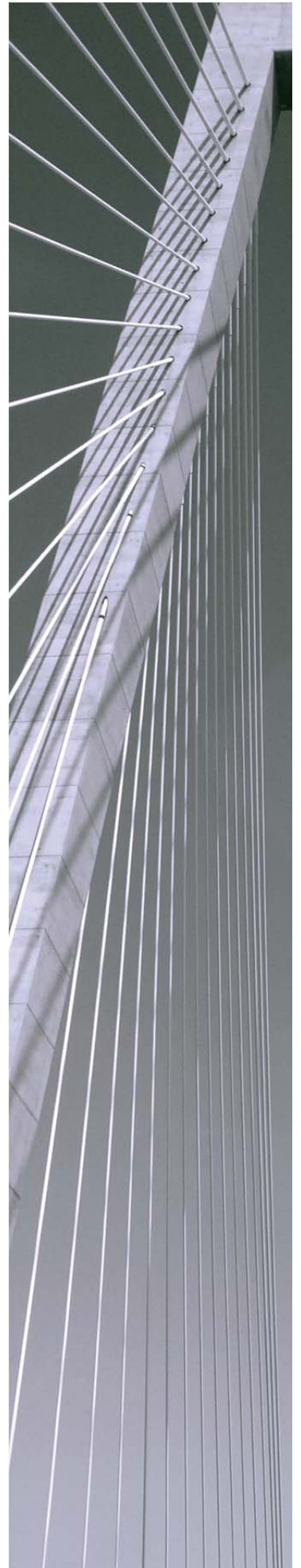




**Enabling MDX Provider for Oracle
OLAP in SAP BusinessObjects
Voyager XI 3.0 and XI 3.1**

Simba Technologies Inc.



Copyright ©2010 Simba Technologies Inc. All Rights Reserved.

Information in this document is subject to change without notice. Companies, names and data used in examples herein are fictitious unless otherwise noted. No part of this publication, or the software it describes, may be reproduced, transmitted, transcribed, stored in a retrieval system, decompiled, disassembled, reverse-engineered, or translated into any language in any form by any means for any purpose without the express written permission of Simba Technologies Inc.

Simba Trademarks

Simba, the Simba logo, SimbaEngine, SimbaEngine C/S, SimbaODBC, SimbaClient, SimbaEngine SDK and SimbaODBC are registered trademarks of Simba Technologies Inc. All other trademarks and/or servicemarks are the property of their respective owners.

Simba Technologies Inc.

938 West 8th Avenue
Vancouver, BC Canada
V5Z 1E5
Tel. +1.604.633.0008
Fax. +1.604.633.0004
www.simba.com

Printed in Canada

V0.5: 2011-06-07

Table of Contents

1	Reference Documents.....	1
2	Introduction	2
3	Adding the MDX Provider to Voyager data driver (Ref[1])	3
4	Configuring the Voyager connection creation User Interface	3
5	Authentication Issues	5
5.1	BusinessObjects XI 3.0.....	5
5.2	BusinessObjects XI 3.1	5
6	Trouble shooting (Ref[1]).....	6
6.1	Verifying the data driver acknowledges the new provider.....	6
6.2	The data driver acknowledges the new provider but it still doesn't appear in the UI ..	6

1 Reference Documents

SAP SDN documentation:

<http://wiki.sdn.sap.com/wiki/display/BOBJ/Enabling+Unsupported+ODBO+providers+-+Voyager+XI+3.0+and+XI+3.1>

2 Introduction

The paper is targeted at SAP BusinessObjects XI users who wish to use SAP BusinessObjects Voyager with Simba's MDX Provider for Oracle OLAP.

As SAP BusinessObjects XI does not officially support the MDX Provider for Oracle OLAP ODBO (OLE DB for OLAP) component, the techniques here are for exploratory purposes only.

In order to have another provider show up in the list of the providers, changes must be made to the Business Objects Enterprise Suite server. The following section describes these changes.

3 Adding the MDX Provider to Voyager data driver (Ref[1])

The data driver can be informed of the MDX for Oracle OLAP ODBO provider by entering the following registry setting.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Business Objects\Suite 12.0\MDA\ODBO]
"SupportedProviders"="OracleMdxProvider.2"
```

The contents of the *SupportedProviders* registry entry is a semi-colon separated list of the versioned provider name for the ODBO providers that you want the data driver to acknowledge. For example, if Voyager is also required to access Microsoft Analysis Services 2000 (in addition to MDX provider for Oracle OLAP), then the entry would look like this:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Business Objects\Suite 12.0\MDA\ODBO]
"SupportedProviders"="OracleMdxProvider.2;MSOLAP.8"
```

Note that the above registry settings also govern the providers available to Web Intelligence.

4 Configuring the Voyager connection creation User Interface

Voyager end-users can only interact with the ODBO providers that are listed in the Voyager user interface. The user interface displays drivers that are obtained by intersecting a list of providers from the data driver (see previous section), and the providers mentioned in a configuration file. Also, the configuration file determines what type of dialog to present for which providers.

The configuration file is an XML file. This XML file is packaged along with other Java classes in a JAR file (Java package) called *VoyagerClient.jar*, and it resides on the BusinessObjects XI server.

This JAR file is located (for a default installation of BusinessObjects Enterprise using Tomcat as the web application server) here:

```
C:\Program Files\Business Objects\Tomcat55\webapps\VoyagerClient\WEB-INF\lib
```

A JAR file has the same format as a ZIP file. Therefore, you can read, update or re-create a JAR file the same way you would with a ZIP file.

The XML file, within this JAR file, is called *providerConfig.xml*, and it is located at the following path:

```
com/businessobjects/multidimensional/cmc/connection/model/provider
```

The contents of the XML file may look similar to the following:

```
<?xml version="1.0"?>
<config>
  <provider name="Hyperion Essbase"
    type="com.businessobjects.multidimensional.cmc.connection.model.p
rovider.essbase.EssbaseProvider"
    builder="com.businessobjects.multidimensional.cmc.connection.mode
l.provider.essbase.EssbasePropertyWidgetBuilder" />

  <provider name="MSOLAP.2,MSOLAP.3,MSOLAP.4"
    type="com.businessobjects.multidimensional.cmc.connection.model.p
rovider.microsoft.MSASProvider"
    builder="com.businessobjects.multidimensional.cmc.connection.mode
l.provider.microsoft.MSASPropertyWidgetBuilder" />

  <provider name="IBMCS"
    type="com.businessobjects.multidimensional.cmc.connection.model.p
rovider.ibm.IBMProvider"
    builder="com.businessobjects.multidimensional.cmc.connection.mode
l.provider.ibm.IBMPropertyWidgetBuilder" />

  <provider name="sapbw_bapi"
    type="com.businessobjects.multidimensional.cmc.connection.model.p
rovider.sap.SAPProvider"
    builder="com.businessobjects.multidimensional.cmc.connection.mode
l.provider.sap.SAPPropertyWidgetBuilder" />

  <provider name="EPMMDX.4"
    type="com.businessobjects.multidimensional.cmc.connection.model.p
rovider.microsoft.EPMMDXProvider"
    builder="com.businessobjects.multidimensional.cmc.connection.mode
l.provider.microsoft.MSASPropertyWidgetBuilder" />

  <provider name="Cartesis"
    type="com.businessobjects.multidimensional.cmc.connection.model.p
rovider.microsoft.CartesisProvider"
```

```
builder="com.businessobjects.multidimensional.cmc.connection.model.provider.microsoft.MSASPropertyWidgetBuilder" />
</config>
```

To add the MDX Provider for Oracle OLAP, append the following lines between `<config>` `</config>` tags as shown below:

```
<config>
...

  <provider name="OracleMdxProvider.2"
    type="com.businessobjects.multidimensional.cmc.connection.model.provider.microsoft.EPMMDXProvider"
    builder="com.businessobjects.multidimensional.cmc.connection.model.provider.microsoft.MSASPropertyWidgetBuilder" />

...
</config>
```

5 Authentication Issues

5.1 BusinessObjects XI 3.0

When creating and using a connection, you are required to provide at least the user name. However, if you use windows authentication, providing a user name and/or password will cause the connection to fail. This is an issue with Voyager Connections that is resolved in XI 3.1 version.

The only authentication method that works properly with XI 3.0 is SSO (or Windows Integrated) authentication. Therefore, the Oracle server with OLAP Option needs to be configured for Windows integrated authentication.

5.2 BusinessObjects XI 3.1

When creating a Voyager connection, always use *Prompt* or *SSO* (single sign-on, or Windows Integrated authentication) as authentication methods. The User Specified method has created problems during our testing with BusinessObjects XI 3.1 Voyager. This issue may be resolved in the future releases of this product.

6 Trouble shooting (Ref[1])

6.1 Verifying the data driver acknowledges the new provider

You can use logging to verify that the data access component has acknowledged the provider as a supported provider. Add the following key to the registry:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Business Objects\Suite 12.0\MDA\Log\
MODULES\ODBOSHAREDUTILITIES]
"Verbosity"=dword:0000000a
```

In the log file, there should now be information about which ODBO providers the data driver supports. To locate the log, look for a registry key similar to the following:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Business Objects\Suite 12.0\MDA\Log]
"LogFile"="C:\\Program Files\\Business Objects\\BusinessObjects
Enterprise 12.0\\Logging\\MDA.log"
```

The location and name of the log file will be indicated by the contents of the *LogFile* entry. For example, the following line from the log indicates a supported provider:

```
Wed Aug 20 10:53:05.968: ODBOSHAREDUTILITIES: ODBOPROVIDERENUMERATOR:
Found Provider: MSOLAP - Microsoft OLE DB Provider for Analysis
Services 9.0
```

However, the following indicates an unsupported provider:

```
Wed Aug 20 10:53:05.978: ODBOSHAREDUTILITIES: ODBOPROVIDERENUMERATOR:
Found unsupported provider: ORACLEMDXPROVIDER - A Brand New ODBO
provider
```

If the Provider appears in the above way, check that you have entered the versioned provider name correctly into the registry as explained above.

6.2 The data driver acknowledges the new provider, but it still doesn't appear in the UI

Even if the XML file has been updated correctly, sometimes the web application server uses a cached version of the jar file and the new provider will not show up in the UI. To try and prevent this from occurring, make sure there are no versions of the *VoyagerClient.jar* file

around, even with different names, or directory structures corresponding to unzipped jar files in the directory where *VoyagerClient.jar* lives.

If the problem persists, remove *VoyagerClient.jar* completely, start the web application server and navigate to the Voyager connections page in the CMC. An error should be displayed as the *VoyagerClient.jar* cannot be found. Now stop the web application server, put *VoyagerClient.jar* with the amended XML back in the correct directory and restart the web application server.