



SimbaEngine X version 10.1

OLE DB Implementation Guide

Simba Technologies Inc.

November 2016



Copyright ©2015 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.

Trademarks

Simba, the Simba logo, SimbaEngine, SimbaEngine C/S, SimbaExpress and SimbaLib 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

Third Party Trademarks

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2014 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

OpenSSL License

Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young(ey@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com). Copyright (c) 1998-2008 The OpenSSL Project. All rights reserved.

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (ey@cryptsoft.com)

All rights reserved.

This package is an SSL implementation written by Eric Young (ey@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

"This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)"

The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).

4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:

"This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

Expat License

"Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the ""Software""), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ""AS IS"", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NOINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE."

Stringencoders License

Copyright 2005, 2006, 2007

Nick Galbreath -- nickg [at] modp [dot] com

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the modp.com nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This is the standard "new" BSD license:

<http://www.opensource.org/licenses/bsd-license.php>

dtoa License

The author of this software is David M. Gay.

Copyright (c) 1991, 2000, 2001 by Lucent Technologies.

Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software.

THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.

Table of Contents

Introduction	9
Implemented OLE DB CoTypes.....	9
Adding OLE DB Support to an Existing Custom ODBC Driver	13
Implementing the GetOLEDBBranding Function.....	14
Populating the m_settingInfo Map.....	14
Updating Custom SQL Data Type Converters	14
Implementing Support for OLE DB Metadata.....	15
Handling No Filters Passed to the DSII.....	15
Limitations.....	15
Testing Your OLE DB Provider	16
Installing Your OLE DB Provider.....	16
Creating Registry Entries.....	16
Installing a Cartridge File	19
Contact Us	20

Introduction

SimbaEngine 9.4 and later includes support for OLE DB. Microsoft® created OLE DB to provide the ability to implement uniform access to data stored outside of traditional production databases, including but not limited to data in:

- File systems
- Indexed-sequential files
- Personal databases
- Spreadsheets
- Project management planners
- Electronic mail

For example, you can use OLE DB to access data in SQL Server® Analysis Services (SSAS).

OLE DB is a set of COM-based interfaces that support the amount of database management system functionality appropriate to the data store, enabling the data store to share its data. For more details on OLE DB, refer to the *OLE DB Programmer's Guide* at [http://msdn.microsoft.com/en-us/library/windows/desktop/ms713643\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/ms713643(v=vs.85).aspx).

The *SimbaEngine OLE DB Implementation Guide* explains how to add OLE DB support to an existing custom ODBC driver created using SimbaEngine . Adding OLE DB support to your custom driver involves linking your working Data Store Interface Implementation (DSII) to the SimbaEngine OLE DB library.

Note: For details on creating an ODBC driver using SimbaEngine, see the *Build a C++ ODBC Driver in 5 Days* guide, available for download from <http://www.simba.com/wp-content/uploads/2014/06/Build-a-C++-ODBC-Driver-in-5-Days1.pdf>. Also, refer to the *SimbaEngine Developer Guide*.

Implemented OLE DB CoTypes

The following tables list OLE DB interfaces and methods implemented in SimbaEngine 9.4 by object, including details of the implementation as needed:

- *Table 1 Data Source Object Implementation* on page 10
- *Table 2 Session Object Implementation* on page 11
- *Table 3 Command Object Implementation* on page 12
- *Table 4 Rowset Object Implementation* on page 13

Note: Empty fields in the Notes column in *Table 1* to *Table 4* are intentionally left blank.

Interface and Method	Notes
IDBCreateSession::CreateSession	
IDBInfo::GetKeywords	
IDBInfo::GetLiteralInfo	
IDBInitialize::Initialize	
IDBInitialize::Uninitialize	
IDBProperties::GetProperties	
IDBProperties::GetPropertyInfo	
IDBProperties::SetProperties	
IPersist::GetClassID	
ISupportErrorInfo::InterfaceSupportsErrorInfo	

Table 1 Data Source Object Implementation

Interface and Method	Notes
IDBCreateCommand::CreateCommand	
IDBSchemaRowset::GetRowset	
IDBSchemaRowset::GetSchemas	<p>Supported schemas are:</p> <ul style="list-style-type: none"> • DBSCHEMA_CATALOGS (if the DSII supports catalogs) • DBSCHEMA_SCHEMATA (if the DSII supports schemas) • DBSCHEMA_TABLES • DBSCHEMA_COLUMNS • DBSCHEMA_FOREIGN_KEYS • DBSCHEMA_PRIMARY_KEYS • DBSCHEMA_TABLE_PRIVILEGES • DBSCHEMA_COLUMN_PRIVILEGES • DBSCHEMA_PROVIDER_TYPES • DBSCHEMA_PROCEDURES (if the DSII supports stored procedures) • DBSCHEMA_PROCEDURE_COLUMNS (if the DSII supports stored procedures)
IGetDataSource::GetDataSource	

Interface and Method	Notes
IOpenRowset::OpenRowset	Implemented as 'Select * from <i>TableName</i> '. IConnection::ToNativeSql is not called. There is currently no way to configure what query is generated.
ISessionProperties::GetProperties	
ISessionProperties::SetProperties	
ISupportErrorInfo::InterfaceSupportsErrorInfo	

Table 2 Session Object Implementation

Interface and Method	Notes
IAccessor::AddRefAccessor	
IAccessor::CreateAccessor	<ul style="list-style-type: none"> • DBACCESSOR_PASSBYREF is not supported. • Null accessors are not supported. • Currently, optimized accessors are not handled specially.
IAccessor::GetBindings	
IAccessor::ReleaseAccessor	
IColumnsInfo::GetColumnInfo	<p>Type mappings (also applicable to ICommandWithParameters::GetParameterInfo):</p> <ul style="list-style-type: none"> • TDW_SQL_BIT -> DBTYPE_BOOL • TDW_SQL_STINYINT -> DBTYPE_I1 • TDW_SQL_UTINYINT -> DBTYPE_UI1 • TDW_SQL_SSMALLINT -> DBTYPE_I2 • TDW_SQL_USMALLINT -> DBTYPE_UI2 • TDW_SQL_SINTEGER -> DBTYPE_I4 • TDW_SQL_UINTEGER -> DBTYPE_UI4 • TDW_SQL_SBIGINT -> DBTYPE_I8 • TDW_SQL_UBIGINT -> DBTYPE_UI8 • TDW_SQL_REAL -> DBTYPE_R4 • TDW_SQL_DOUBLE/TDW_SQL_FLOAT -> DBTYPE_R8 • TDW_SQL_CHAR/ TDW_SQL_VARCHAR/ TDW_SQL_LONGVARCHAR -> DBTYPE_STR • TDW_SQL_WCHAR/ TDW_SQL_WVARCHAR/

Interface and Method	Notes
	TDW_SQL_WLONGVARCHAR -> DBTYPE_STR <ul style="list-style-type: none"> • TDW_SQL_BINARY/ TDW_SQL_VARBINARY/ TDW_SQL_LONGVARBINARY -> DBTYPE_BYTES • TDW_SQL_NUMERIC/TDW_SQL_DECIMAL -> DBTYPE_NUMERIC • TDW_SQL_TYPE_DATE -> DBTYPE_DBDATE • TDW_SQL_TYPE_TIME -> DBTYPE_DBTIME • TDW_SQL_TYPE_TIMESTAMP -> DBTYPE_DBTIMESTAMP • TDW_SQL_GUID -> DBTYPE_GUID • Single-field intervals (except TDW_SQL_INTERVAL_SECOND) -> DBTYPE_I4. • All other interval types -> DBTYPE_STR
IColumnsInfo::MapColumnIDs	
ICommand::Cancel	
ICommand::Execute	
ICommand::GetDBSession	
ICommandPrepare::Prepare	
ICommandPrepare::Unprepare	
ICommandProperties::GetProperties	
ICommandProperties::SetProperties	
ICommandText::GetCommandText	
ICommandText::SetCommandText	
ICommandWithParameters::GetParameterInfo	See IColumnsInfo::GetColumnInfo
ICommandWithParameters::SetParameterInfo	
ICommandWithParameters::MapParameterNames	
ISupportErrorInfo::InterfaceSupportsErrorInfo	

Table 3 Command Object Implementation

Interface and Method	Notes
IAccessor::AddRefAccessor	

Interface and Method	Notes
IAccessor::CreateAccessor	
IAccessor::GetBindings	
IAccessor::ReleaseAccessor	
IColumnsInfo::GetColumnInfo	
IColumnsInfo::MapColumnIDs	
IParentRowset::GetChildRowset	
IRowset::AddRefRows	
IRowset::GetData	
IRowset::GetNextRows	Only one active row handle at a time, per rowset, is supported.
IRowset::ReleaseRows	
IRowset::RestartPosition	Currently, calling this method always re-executes the query (or schema result), as scrollable cursors are not supported.
IRowsetInfo::GetProperties	
IRowsetInfo::GetReferencedRowset	
IRowsetInfo::GetSpecification	
ISupportErrorInfo::InterfaceSupportsErrorInfo	

Table 4 Rowset Object Implementation

Adding OLE DB Support to an Existing Custom ODBC Driver

Note: For an example OLE DB implementation, see the Quickstart driver found in the Examples subfolder in your SimbaEngine installation folder. For more details on the Quickstart example driver, see the *Build a C++ ODBC Driver in 5 Days* guide.

To add OLE DB support to an existing ODBC driver created using SimbaEngine:

- Implement the function `Simba::OLEDB::GetOLEDBBranding`
- Populate the `m_settingInfo` map
- Update custom SQL data type converters
- Implement support for OLE DB metadata
- Handle no filters passed to the DSII

Implementing the GetOLEDBBranding Function

Your DSII must implement the `Simba::OLEDB::GetOLEDBBranding` function in `Main_Windows.cpp`. The function is an entry point that returns branding information, including a unique GUID, for registering your OLE DB provider in the Windows Registry. When your driver starts, `GetOLEDBBranding` is called to retrieve the branding information.

Populating the m_settingInfo Map

The `IConnection::GetConnectionSettingInfo` method has a default implementation in the `DSIConnection` class. If you inherit the method from the `DSIConnection` class, then you need to populate the `m_settingInfo` map. For example, populate the `m_settingInfo` map in the constructor of your `DSIConnection` subclass. The `m_settingInfo` map is used to set initialization properties for the OLE DB provider.

Implement the `IConnection::GetConnectionSettingInfo` method if necessary.

Note: For more information on OLE DB initialization properties implemented in SimbaEngine, see documentation related to the `IConnection::GetConnectionSettingInfo` method in the *SimbaEngine Developer Guide*. The topic *OLE DB Initialization Properties: Quick Reference* on MSDN at <http://msdn.microsoft.com/en-us/library/windows/desktop/ms723996%28v=vs.85%29.aspx> provides general information.

Updating Custom SQL Data Type Converters

SimbaEngine allows DSII implementers to add custom SQL data types based on existing types. Conversion is required between SQL data types used in the data source and C data types used in the application. If you define custom SQL data types, then you need to update your converters to support the OLE DB requirement to retrieve the length of a result without retrieving the result. The length of a result is required to declare a buffer to contain the result.

In a SimbaEngine converter, the `SQLCData` object encapsulates application-side data. To support OLE DB, a `SQLCData` flag indicates whether a valid buffer has been bound to the object. Converters must test the flag using the `SQLCData::IsBufferValid` function prior to storing a result in the `SQLCData` object. If `IsBufferValid` returns the value `False`, then the converter needs to call `SQLCData::SetConvertedLength`, passing in the total, untruncated length of the data to store after conversion. When `IsBufferValid` returns the value `True`, then the converter can store data in the `SQLCData` object.

Converters should post any conversion warnings to `IWarningListener`.

Note: For more details on using custom SQL data types in SimbaEngine, see the *SimbaEngine Developer Guide*. The MSDN topic *Data Types in ODBC* at

<http://msdn.microsoft.com/en-us/library/ms712514%28v=vs.85%29.aspx> discusses SQL data types, C data types and data type conversions.

Implementing Support for OLE DB Metadata

In SimbaEngine, when fetching metadata you return either an IResult object or a DSIMetadataSource object.

Note: For details on fetching metadata, refer to the *SimbaEngine Developer Guide*.

Differences exist between metadata defined in OLE DB, compared with ODBC. To support OLE DB, SimbaEngine defines additional metadata table identifiers. For details on table identifiers created in SimbaEngine to support OLE DB that can be returned, refer to comments in the file DSIMetadataColumnIdentifierDefns.h

If you return a DSIMetadataSource object, then you need to define the metadata table identifiers related to OLE DB. For optional or unused metadata, returning NULL is sufficient.

Handling No Filters Passed to the DSII

In ODBC, JDBC or ADO.NET specifications, a filter is always passed to the DSII. To support OLE DB, your DSII must handle the possibility of no filters being passed to the DSII.

Limitations

OLE DB support in SimbaEngine is subject to the following limitations:

- Transactions are not supported.
- Scrollable cursors are **not** supported. Only non-scrollable—that is, forward-only—cursors are supported. Moving backwards through a rowset is not supported.
- Only one row handle at a time can be active per rowset.

Note: For details on row handles in OLE DB, see <http://msdn.microsoft.com/en-us/library/windows/desktop/ms719753%28v=vs.85%29.aspx>.

- SimbaEngine supports creating rowset objects:
 - When executing a command
 - Using IDBSchemaRowset::GetRowset
 - Using IOpenRowset::OpenRowset
- Streaming data is not supported.

- The following OLE DB data types are **not** supported:
 - DBTYPE_CY
 - DBTYPE_BSTR (for output)
 - DBTYPE_IDISPATCH
 - DBTYPE_ERROR
 - DBTYPE_VARIANT
 - DBTYPE_DECIMAL
 - DBTYPE_ARRAY
 - DBTYPE_VECTOR
 - DBTYPE_HCHAPTER
 - DBTYPE_VARNUMERIC
 - DBTYPE_FILETIME
 - DBTYPE_PROPVARIANT
 - DBTYPE_XML

Testing Your OLE DB Provider

You can test an OLE DB provider you create using SimbaEngine using the Rowset Viewer application included with Microsoft Data Access SDK 2.8, available for download from <http://www.microsoft.com/en-ca/download/details.aspx?id=5793>. For instructions on using Rowset Viewer, see the MSDN topic *Rowset Viewer Sample* at <http://msdn.microsoft.com/en-us/library/windows/desktop/ms714336%28v=vs.85%29.aspx>.

Installing Your OLE DB Provider

Installing your OLE DB provider—for example, to use the provider with SQL Server Data Tools (SSDT), Business Intelligence Development Studio (BIDS) or SQL Server Analysis Services (SSAS)—involves the following tasks:

- You need to create entries in the Windows Registry.
- If your data source uses custom SQL and you use your provider with SQL Server Analysis Services or SQL Server Data Tools, then then you must install a cartridge file.

Creating Registry Entries

You can install your OLE DB provider in any location. Windows Registry entries record the installation location.

After you implement the `Simba::OLEDB::GetOLEDBBranding` method, you can use the `Regsvr32` tool to register your provider.

Note: For details on implementing `GetOLEDBBranding`, see *Implementing the GetOLEDBBranding Function* on page 14.

If you prefer, you can use an installer-specific tool. For example, for a WiX installer you can use the Heat tool to generate a .WXS file containing WiX format Registry entries, which require only minor modification. For an MSI installer, you can set the Register flag to register the provider.

You can also record the required Registry entries in the installer manually.

Minimum Required Registry Entries

The following sample .REG file lists the minimum Windows Registry entries required for an OLE DB provider created using SimbaEngine. Most of the required information for the Registry entries is defined in the Simba::OLEDB::GetOLEDBBranding method that you implemented in the section *Implementing the GetOLEDBBranding Function* on page 14.

Depending on the features of your provider, more Registry entries may be required.

Note: Placeholders definitions appear following the sample .REG file.

```
[HKEY_CLASSES_ROOT\ProgID]
  @="ProviderDescription"

[HKEY_CLASSES_ROOT\ProgID\CLSID]
  @="ProviderGUID"

# For 32-bit Providers on 64-bit Windows (Note: for 64-bit
# Providers on 64-bit Windows or 32-bit Providers on 32-bit
# Windows, simply remove "Wow6432Node" from the paths):

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ProviderGUID]
  @="ProgID"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ProviderGUID\Extended
Errors]
  @="ExtendedErrorServiceDescription"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ProviderGUID\Extended
Errors\ErrorLookupGUID]
  @="ErrorLookupServiceDescription"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ProviderGUID\InprocServer32
]
  @="ProviderInstallLocation"
  "ThreadingModel"="Both"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ProviderGUID\OLE DB
Provider]
  @="ProviderDescription"
```

```
[ HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ProviderGUID\OLEDB_SERVICES
]
@=dword:ffffffff

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ProviderGUID\ProgID]
@="ProgID"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ErrorLookupGUID]
@="ErrorLookupServiceDescription"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ErrorLookupGUID\InprocServer3
2]
@="ProviderInstallLocation"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ErrorLookupGUID\InprocServer3
2\ThreadingModel]
@="Both"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ErrorLookupGUID\ProgID]
@="ProgID"

[HKEY_CLASSES_ROOT\Wow6432Node\CLSID\ErrorLookupGUID\VersionInde
pendentProgID]
@="VersionIndependentProgID"
```

Where:

- *ProgID* is a provider-defined ID for the provider, for example SimbaSampleProvider
- *ProviderDescription* is a readable description for the provider, for example Simba Sample OLE DB Provider
- *ProviderGUID* is a static random GUID in the form {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX} to uniquely identify the provider.
- *ExtendedErrorServiceDescription* is a readable description of the extended error service, for example Extended Error Service
- *ErrorLookupGUID* is a static random GUID in the form {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX} to uniquely identify the error lookup service.
- *ErrorLookupServiceDescription* is a readable description of the error lookup service, for example Simba Sample Error Lookup Service
- *ProviderInstallLocation* is the installation location of the provider specified by the installer.
- *VersionIndependentProgID* is a version-independent instance of the *ProgID* for the provider.

Installing a Cartridge File

You need to install a cartridge file **only** if your data source uses custom SQL and you use your provider with SQL Server Analysis Services or SQL Server Data Tools. A cartridge file is an XSL file used to transform queries from abstract query language to the SQL defined by your data source.

If you need a cartridge file, contact Technical Support. For details, see *Contact Us* on page 20.

Important: Due to path dependencies, prior to installing a cartridge file, install all necessary versions of SQL Server, Visual Studio, Business Intelligence Development Studio and SQL Server Data Tools.

SQL Server Analysis Services

Cartridge file installation locations are tested using:

- SQL Server 2008 R2 Service Pack 3 (32- and 64-bit versions) running on Windows Server 2008 R2 (64-bit version)
- SQL Server 2012 Service Pack 2 running on Windows Server 2012 (64-bit version)

Note: Placeholder definitions appear following the procedure.

To install the cartridge file:

1. Copy the cartridge file to the folder *SSASInstallFolder\AS OLEDB\AnalysisServicesVersion\Cartridges*
2. If you are using SQL Server 2008 R2, then copy the cartridge file to the folder *MSSQLServerInstallFolder\MSSQLServerVersion\Tools\Binn\VSShell\Common7\IDE\DataWarehouseDesigner\UIRdmsCartridge*

OR

- If you are using SQL Server 2012, then copy the cartridge file to the folder *MSSQLServerInstallFolder\MSSQLServerVersion\Tools\Binn\ManagementStudio\DataWarehouseDesigner\UIRdmsCartridge*
3. Copy the cartridge file to the folder *MSSQLServerInstallFolder\SSASServerInstanceName\OLAP\bin\Cartridges*

Where:

- *SSASInstallFolder* is the installation location of SQL Server Analysis Services, for example *C:\Program Files\Microsoft Analysis Services*
- *AnalysisServicesVersion* is the version number of the target Analysis Services instance, for example 10, 110 or 120

- *MSSQLServerInstallFolder* is the installation location of MS SQL Server, for example C:\Program Files\Microsoft SQL Server
- *MSSQLServerVersion* is the version number of the target SQL Server instance, for example 90, 110 or 120
- *SSASServerInstanceName* is a composition of the version of SSAS and the name of the instance, for example MSAS11.MSSQLSERVER or MSAS10_50.MSSQLSERVER

SQL Server Data Tools

Cartridge file installation locations are tested using:

- Business Intelligence Development Studio running on Windows Server 2008 R2 (64-bit version)
- SQL Server Data Tools for Visual Studio 2012 (32-bit version) running on Windows 7 Professional (64-bit version)
- SQL Server Data Tools for Visual Studio 2012 (32-bit version) running on Windows Server 2012 (64-bit version)

To install the cartridge file:

1. Copy the cartridge file to the folder
VSInstallFolder\Common7\IDE\PrivateAssemblies\
DataWarehouseDesigner\UIRdmsCartridge where *VSInstallFolder* is the installation location of the target version of Visual Studio, for example C:\Program Files (x86)\Microsoft Visual Studio 10.0
2. If you are using Visual Studio 2010 or higher with SQL Server Data Tools, then copying the cartridge file to the folder
VSInstallFolder\Common7\IDE\PrivateAssemblies\Business Intelligence Semantic Model\Cartridges is recommended, but not required.

Contact Us

If you have difficulty using the SimbaEngine, please contact our Technical Support staff. We welcome your questions, comments and feature requests.

Technical Support is available Monday to Friday from 8 a.m. to 5 p.m. Pacific Time.

You can contact Technical Support via:

- **E-mail:** support@simba.com
- **Web site:** www.simba.com
- **Telephone:** (604) 633-0008 Extension 3

- **Fax:** (604) 633-0004

You can also follow us on Twitter @SimbaTech