

# SimbaEngine ODBC SDK

## SimbaODBC – An ODBC Interface SDK Component



### ODBC interface component for building ODBC drivers for SQL and non-SQL databases

SimbaODBC is a component of SimbaEngine ODBC SDK. It provides the processing for ODBC drivers built with SimbaEngine ODBC SDK and can be used anywhere that an ODBC 3.52 compliant interface implementation is required.

### Features

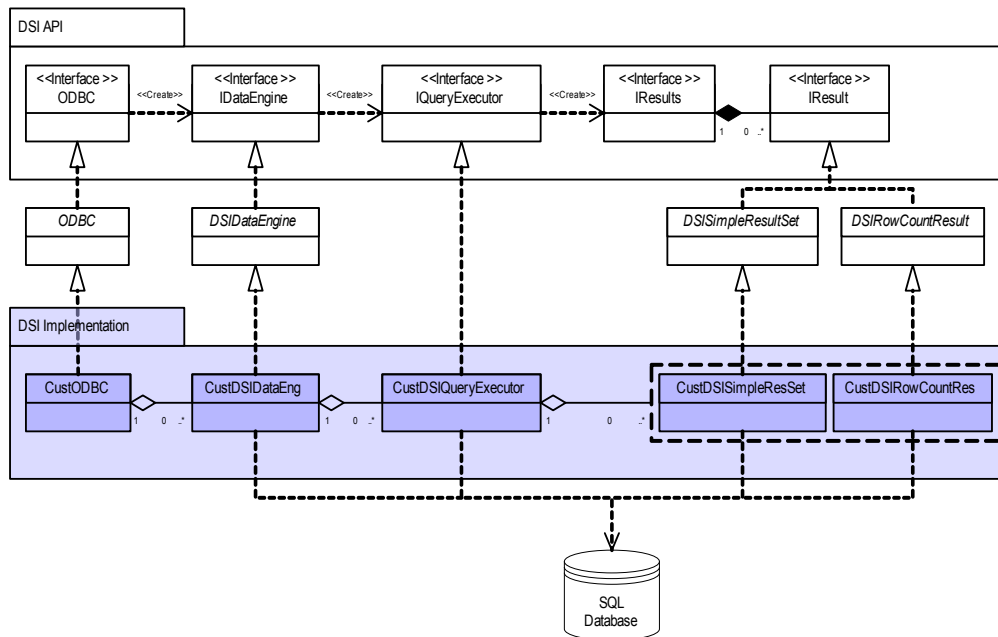
- Complete ODBC 3.52 processing component.
- Designed for access by common reporting tools like Microsoft Excel, SAP BusinessObjects Crystal Reports and MicroStrategy.
- Can be used in two different ways:
  1. Directly as the ODBC interface for any SQL-capable relational database, or
  2. With Simba SQL Engine to provide complete ODBC/SQL access to any non-SQL database.
- Rebrandable and localizable – make it your driver.
- Adds no SQL processing overhead.
- Standard Simba Data Store Interface (DSI) API – works seamlessly with Simba SQL Engine and custom DSI implementations.
- Flexible – start with minimal ODBC functionality and add to it to meet demand.
- Create your own ODBC data driver, or have Simba do the work for you.

### Description

SimbaODBC is a complete ODBC 3.52 processing component that can be used to build ODBC drivers for any SQL database. Used with Simba SQL Engine, you can also build ODBC drivers for non-SQL data sources such as web services, object-oriented, network, real time (e.g. SCADA) and ISAM databases.

SimbaODBC enables all popular reporting and Business Intelligence (BI) applications, such as Microsoft Excel and Access, SAP BusinessObjects Crystal Reports and MicroStrategy to connect to your data. It also contains the ODBC 3.52 processing and housekeeping functionality expected by these applications and by ODBC Driver Managers. SimbaODBC ensures that the behavior of your data driver is robust, predictable and graceful.

It provides custom ODBC driver developers with a simple and logical interface – the Simba Data Store Interface or DSI – that is easy to understand and use. It follows the hierarchical structure of ODBC itself without restricting the kind of database that can be accessed. SimbaODBC adapts to the capabilities of your database and reflects those capabilities to the application. For instance, if your database supports transactions, SimbaODBC will report that to the application when queried. Data retrieval through SimbaODBC is optimized for speed and data copying is minimized. If data conversion is not required, data buffers are passed straight through. If data conversion is needed, SimbaODBC has robust and efficient data conversion functions you can use.



**Block Diagram:** The internal organization of a SimbaODBC driver

Referring to the diagram on the previous page, a custom ODBC driver developer only needs to derive the highlighted classes from Simba's ODBC interface framework and connect them to the database. The functionality can be developed incrementally. You can use SimbaODBC to build a custom ODBC Driver for your SQL data source in a few short days.

SimbaODBC is designed to work seamlessly with the other SimbaEngine ODBC SDK components. Simba SQL Engine links directly to SimbaODBC to create a framework for ODBC drivers for non-SQL data sources. When working together, SimbaODBC and Simba SQL Engine require only that your data store can be presented as if it contained tables and columns and rows of data. Everything else is handled by the SimbaEngine ODBC SDK components. For remote access, your DSI implementation can be linked directly to SimbaServer (with no changes to your ODBC driver code). Applications can then use SimbaClient for ODBC, JDBC or ADO.NET to connect over the high-performance, secure Simba Client/Server Protocol to your centrally-located data source. OLE DB client access is also available via a bridge.

## Specifications

- Supports the ODBC 3.52 database access interface standard.
- Comprehensive support for ODBC 3.52 data types and conversions between data types.
- Support for ANSI and Unicode UTF-8, -16 and -32 strings for both data and metadata.
- Available in 32-bit and 64-bit versions for Windows, Linux, Mac, Solaris, HP-UX and AIX.
- Maximum result set size limited only by memory and the implementation of the IResult sub-classes.
- Up to 65,356 columns allowed per result-set.
- Up to 2 GB of data per column-value on 32-bit systems, 4 GB on 64-bit systems.