
SimbaEngine X 10.1.3

Released 2017-5-29

These release notes provide details of features and known issues in SimbaEngine SDK X version 10.1.3.

Enhancements & New Features

The following enhancements and new features are made to SimbaEngine X 10.1.0.

Improved Makefiles

Significant improvements are made to the makefiles and Visual Studio projects to streamline the build process on all supported platforms.

INI configuration file on Windows

Drivers can retrieve their driver-specific configuration from an `.ini` file instead of the Windows registry.

Mac OS X Installer

Installation of the SimbaEngine X SDK on Mac OS X platforms is simplified with the creation of an installer.

Event Trace for Windows (ETW) capability added

Instead of logging errors and messages to a text file, SimbaEngine X SDK now supports logging to ETW.

HTML Documentation

To improve usability and facilitate search, the *SimbaEngine SDK Developer Guide* is divided into two guides: *Developing Drivers for SQL-Aware Data Stores* and *Developing Drivers for Data Stores Without SQL*. These guides are available online in HTML format at <http://www.simba.com/resources/sdk/documentation/>.

Breaking Changes

The following breaking changes to SimbaEngine X SDK version 10.1 may require you to modify and recompile your SimbaEngine X SDK version 10.0. code.

SimbaEngine 10.1.3

Visual Studio 2015 Update 3 is required

In this release, the Visual Studio 2015 builds are generated with VS2015 Update 3. If you encounter the linker error "Mismatch between 'p1' version <number> and 'p2' version <number>", you must upgrade your version of Visual Studio 2015 to Update 3.

SimbaEngine 10.1.2

MemoryManager::ReleaseMemoryResources() is updated

The method `Simba::DSI::MemoryManager::ReleaseMemoryResources()` is updated to address support issue 00092758. This method now releases the memory but no longer releases the token. To release the token, call `MemoryManager::ReleaseMemoryToken()`.

The method `MemoryManager::GetUniqueMemoryToken()` is renamed to `MemoryManager::GetUniqueMemoryToken()`.

SimbaEngine 10.1.0

[DSI] CreateHasher() method added to the ICollation interface

`ICollation` has a new interface method, `CreateHasher()`. If your custom driver code extends `ICollation`, for example to create new collation rules for a custom data type, you need to implement this method. See *Developing Drivers for Data Stores Without SQL* at <http://www.simba.com/resources/sdk/documentation/> for more information.

[DSI] GetHasher() method added to the IColumn interface

`IColumn` has a new interface method `GetHasher()`. If your custom driver code extends `IColumn` you need to implement this method. See *Developing Drivers for Data Stores Without SQL* for more information.

[DSI] Additional parameter added to MetadataFilter methods

In `DSIMetadataFilterFactory.h`, the parameter `in_escapeChar` is added to the following methods:

- `MakeFiltersForPrimaryKeysMetadata()`
- `MakeFiltersForForeignKeysMetadata()`
- `MakeFiltersForStatisticsMetadata()`

- `MakeFiltersForSpecialColumnsMetadata()`

Use `in_escapeChar` to specify the escape character to use for filtering.

[DSI] A reference to `IConnection` is used rather than a pointer

The constructors in the following filters use a reference to `IConnection` instead of a pointer:

- `IDMetadataFilter.h`
- `OAMetadataFilter.h`
- `PVMetadataFilter.h`
- `SmallIntMetadataFilter.h`
- `StringMetadataFilter.h`
- `VLMetadataFilter.h`

[Support] `IHasher` interface moved to a different component

`IHasher` interface is moved from DSI to Support.

[SQLEngine] `DSIEXT_DATAENGINE_TEMPORARY_INDEXES`

The data engine property `DSIEXT_DATAENGINE_TEMPORARY_INDEXES` is removed.

[DSI] A new interface, `IMemoryContext`, is introduced. Any method that requires memory from the memory manager must pass a pointer to this interface along with the request.

[DSI] Significant changes are made to the `MemoryManager` interface. The following methods are removed or have a new method signature:

- `AllocateBlocks()`
- `ReleaseBlock()`
- `ReleaseAllAssignedBlocks()`
- `ReserveBlocks()`

The following methods are added:

- `NotifyCancellation()`
- `CleanUpMemoryRecords()`
- `GetUniqueMemoryToken()`

For more information, see `MemoryManager.h` in the `DataAccessComponents\Include\DSI` folder of the SimbaEngine X SDK installation directory.

[Makefile] Simplified makefiles

Significant improvements are made to the makefiles and Visual Studio projects to simplify the build process on all supported platforms. Changes include:

- Commands to run the makefile, including default settings and available options.
- The names of the libraries and executables include "32", "64", or "3264" to indicate their bitness.
- The libraries and executable names no longer include the suffix 'debug' or '_MTDLL'. Instead, the folder structure indicates whether a library is release or debug.
- The folder structure of the libraries and executables is changed to include information about bitness, platform, and compiler, and build configuration.

Changes and Resolved Issues

The following is the list of changes and resolved issues in the SimbaEngine X SDK release.

SimbaEngine 10.1.3

[SQLEngine] In previous releases, the SQL Engine would pass inner joins down to the DSII, but not cross joins. In this release, the SQL Engine is updated to pass down cross joins as well. Passing cross joins down to the DSII allows the driver to handle the SQL statement more efficiently than the default SQL Engine implementation.

The MiniParser can now handle the CONVERT scalar function in non-escaped form. For more information, see the MiniParser documentation in the SDK Development guide.

[00092741] The method `IResult::GetRowCount()` is overloaded with a new method, which can return 64-bit rowcounts. This allows `SQLRowCount()` to return additional warnings and errors. The previous version of `GetRowCount()` is deprecated.

[00093301] When `SQLDescribeParam` is provided with an invalid number, an incorrect `SQLState` is returned. This issue is resolved by changing the `SQLState` returned from `HY000` to `07009`.

[SQLEngine] Errors that may have occurred in some cases, when passing down `TopNSort`, are resolved.

[00093295][Support] The conversion between C interval and SQL Numeric may not have returned a correct value in all cases. This issue is resolved.

[00092734][Codebase][JavaQuickstart][JavaQuickJson][Quickstart] The sample drivers Codebase, JavaQuickstart, JavaQuickJson, and Quickstart incorrectly reported that they supported some string functions which were not actually supported. This caused errors when the unsupported string functions were used. This issue is resolved.

[00092800] [ODBC] When executing statements with multiple queries, the row count may not be updated correctly in all cases, resulting in an incorrect row count being returned from a call to `SQLGetDiagField()`. This issue is resolved.

[00092741] [ODBC] `SQLRowCount()` now checks for truncation, then posts a warning if truncation occurs.

[00092740][DSI] It is now possible to override the column metadata for catalog functions when using `DSIMetadataSource`.

[ODBC] In some cases, `SQL_C_TINYINT` may not have been converted from `unsigned` to `signed`. This issue is resolved.

[Server] The server may have closed the `IResult` cursor multiple times when the client closed the cursor. This issue is resolved.

SimbaEngine 10.1.2

[00092758] Issues in the `Simba::DSI::MemoryManager` class may have caused the SimbaEngine X SDK to terminate unexpectedly. This issue is resolved by updating the `ReleaseMemoryResources()` method to release the memory without releasing the token. To release the token, call `ReleaseMemoryToken()`.

Support for the Create Table as Select (CTAS) function is added. For more information, see the SimbaEngine X SDK Developer's guide, "*Developing Drivers for Data Stores Without SQL*".

The SimbaEngine X SDK includes the MiniParser feature, which allows drivers to replace ODBC escape sequences with commands that are specific to the data store. In previous releases, the MiniParser was supported for ODBC escape sequences only. In this release, the MiniParser is supported for both ODBC and JDBC escape sequences.

[00092331] In some cases, error messages created by the driver may not have been retrieved by the SimbaEngine X SDK, resulting in an “error message not found” error. This issue is resolved.

[JNI] JNIDSJ drivers use `DSIMessageSource` to load error messages. In this release, the following methods are added to `DSIMessageSource` in order to enhance configuration: `setMessageSourceVendorName()`, `setMessageSourcePrefixesVendorName()`, and `setMessageSourceComponentName()`. For more information, see the guide “*Developing Drivers for Data Stores Without SQL*”.

[SQL Engine] The hybrid hash algorithm used in the SQL Engine to perform Equi Join operations is optimized, resulting in a 10% increase in performance for a nested join query executed in an environment with a low memory limit.

[SQL Engine] `DataNeeded` flag may not have been properly set for pass-down results. This issue is resolved.

[Support] Initialization of `m_overflow` is added to the `TDWExactNumeric` constructor, preventing possibly errors due to incorrect initialization.

[JDBC] Calling `setAutoCommit(false)` may have incorrectly ended a transaction. This issue is resolved.

[JDBCClient] During repeated executions of a prepared statement in the JDBC client, disconnection may have occurred. This issue is resolved.

[JDBCClient][ODBCClient][Server] Timeouts during login may have occurred incorrectly. This issue is resolved.

[JDBC] Support for `DriverManager.setLoginTimeout()` is added for JDBC drivers.

[SQL Engine] When an exception is thrown during the execution of an incorrect SQL query, the state of some `Executor` objects may not have been reset correctly, resulting in errors. This issue is resolved.

[ODBC] In some cases, the encoding strategy specified in connection properties was not used. Instead, the native character encoding was used. This issue is resolved so that the specified encoding strategies are respected.

SimbaEngine 10.1.1

[00090513][DSI] A `clone()` method is added to `DSIMetadataFilter`.

[00090617][DSI] The connection string constants are updated to conform to the ODBC 4.0 specification updates.

[00091070][SQLEngine] The length of Unicode string literals may not have been handled correctly when writing to temporary tables. This issue is resolved.

Performance improvements are made to the miniParser feature.

[DSI] The `SQLTable()` catalog function should filter out all the tables when an empty string is passed in. In previous releases, this functionality was included in release mode only. In this release, SimbaEngine X SDK is updated to include this functionality when running in debug mode as well.

[SQL Engine] The C++ SQL engine includes support for stored procedures. In this release, support for stored procedures is added to the Java SQL engine.

Contact Us

For more information or help using this product, please contact our Technical Support staff. We welcome your questions, comments, and feature requests.

Note: To help us assist you, prior to contacting Technical Support please prepare a detailed summary of the version and development platform that you are using.

You can contact Technical Support via the Magnitude Support Community at <http://magnitudesoftware.com/online-support/>.

You can also follow us on Twitter @SimbaTech and @Mag_SW.