

---

## SimbaEngine X 10.1.5

Released 2018-1-26

These release notes provide details of features and known issues in SimbaEngine SDK X version 10.1.5.

### Enhancements & New Features

The following enhancements and new features are made to SimbaEngine SDK X 10.1.5.

#### New methods added

- A new `simba_strtok` method has been added. This method is a wrapper for `strtok_r` (POSIX) and `strtok_s` (Windows) functions.
- A new `SQLDataEngine.getSharedSqlConverterGenerator()` method has been added that allows access to the shared `SqlConverterGenerator`.

#### New logging property

[ODBC] The new driver property `DSI_DRIVER_LOG_QUERY_AT_PREPARE` has been added. This allows you to set the log level for logging the query when it is prepared.

#### Error messages

JDBC error messages have been updated with new translations.

#### SSL error messages

Additional error messages regarding SSL have been added to help diagnose connection issues.

#### Customized return values

You can now customize the return value of `DataBaseMetadata.supportsBatchUpdates()` in our JDBC client.

#### Improved column and table logic

The column and table resolution logic has been updated to correctly take `SQL_IDENTIFIER_CASE` and `SQL_QUOTED_IDENTIFIER_CASE` into account.

## Improved support for Cast function in JSQL

In the JSQLEngine parse tree, `CAST` can now accept Boolean expressions as a Cast operand as defined by SQL 2003 grammar. Also, the cast `TYPE` can now be followed by a parameter list.

## New functions in JSQLEngine

`TIMESTAMPADD` and `TIMESTAMPDIFF` functions are also now supported by the JSQL engine up to the parse tree.

## Log file prefix

Users now have the option to add a prefix to driver log files.

## Support for calculations

SQLEngine now supports `DATE/TIME/TIMESTAMP`, plus/minus `INTERVAL` operations.

## Support for `atod()` method

A new driver property, `DSI_DRIVER_USE_STDLIB_STRTOD`, allows DSII choose whether to use the standard library `strtod()` or the faster but less accurate `atod()` method.

## Setting a default locale

You can now use the `SimbaSettingReader::SetDriverDefaultLocale()` function to set a custom default locale, rather than the default fallback of `en-US`.

## User ID in log files

Log files can now be set to recognize `uid` as sensitive information, and show only as asterisks.

## Support for input passdown on `INSERT` statements

SQLEngine now supports passdown on the input in `INSERT` statements.

## Support for resource optimization

A `Reset()` method has been added to the `Simba::DSI::IQueryExecutor` interface to notify DSII that the cursor is closed. This function will be invoked when `SQLCloseCursor` or `SQLFreeStmt` with the `SQL_CLOSE` option are called. [ClientServer]

## Error messaging

Users can now throw a checked exception from the `IReplacer.replace()` method in `JDBCEscaper` using the superinterface for `IReplacer`.

## Changes and Resolved Issues

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

### SimbaEngine 10.1.5

[Support] `WideStreamConverter` incorrectly converts partial codepoints.

[Support] In some cases drivers would produce incorrect intervals. This has been resolved. Checks have been added for invalid interval literals during conversion to interval data types for two missing cases.

- Interval literals that do not contain a value for every field that is implied in the interval qualifier
- Interval literals that do not conform to Gregorian calendar constraints in trailing fields

[Support] Conversions from `C_TIMESTAMP` to `SQL_WCHAR` do not recognize the value set in the `SQL_DESC_PRECISION` field when setting the precision for seconds. This has been resolved. If no value is set in the field, the driver records up to 6 digits.

[Support] Errors can sometimes occur when converting `SQL_C_FLOAT` and `SQL_C_DOUBLE` to `SQL_BIGINT`. This issue is resolved.

[ODBC] The `TableName` argument for catalog functions does not recognize escape patterns when checking against the maximum table name length. This issue is resolved.

[ODBC] Error or invalid argument values for `SQLGetStmtAttr`, `SQLSetConnectAttr`, and `SQLBindParamter` are not handled by the program. This issue is resolved.

[ClientServer] Key names are not populated in error messages where parsing of the connection string fails. This issue is resolved.

[Support] During installation, drivers may show a system message regarding creating a temporary swap file name. This issue is resolved.

[ODBC] The `SQL_DESC_DATA_PTR` function is reset after `SQLSetDescField` is called on data type related attributes. This issue is resolved.

[ODBC] When using iODBC as the driver manager, error or invalid values for `SQLSetStmtAttr`, `SQLSetDescField`, `SQLGetFunctions` and `SQLSetConnectAttr` may not be handled correctly. This issue is resolved.

[Support] Conversion from `C_TYPE_TIMESTAMP` to `SQL_CHAR` where precision is not at 9 digits or better produce incorrect results. This issue is resolved.

[Support] Conversion from `SQL_INTERVAL_SECOND` to `C_NUMERIC` that involved fractional seconds produced incorrect results. This issue is resolved.

[Support] The `TypeConverter::toBigInteger` method does not handle `SQL_WCHAR`, `SQL_WVARCHAR`, `SQL_WLONGVARCHAR` conversions. This issue is resolved.

[Support] The application name output in the logfile is incorrect on OSX. This issue is resolved.

[SQLEngine] The `SQLEngine` does not alert users for some invalid SQL queries that select non-aggregated columns which are not referred to in `GROUP BY`. This issue is resolved.

[JSQLEngine] The `Identifier` class can produce unexpected comparison results due to its reliance on Java's default implementation of `hashCode()` and `equals()`. This has been resolved. These two functions are now overridden using strings constructed from the Catalog, Schema and Table names for hashing and comparison.

[ODBC] iODBC does not prevent the application from calling `SQLSetConnectAttr` while there were asynchronously-executing ODBC API functions on child statements of that connection. This has been resolved. Logic to check for calls to `SQLSetConnectAttr` while the connection has child statements in the `async` or `NEED DATA` states has been added.

[Support] An undesired null terminator is added to the result of SQL to SQL conversions when converting from float/real/double to char/wchar data types. This issue is resolved.

[Support] An overflow issue sometimes occurs during a C to SQL conversion between interval types, which can cause error checking to be skipped. Errors may also occur when converting fractional seconds in various interval to interval conversions. This issue is resolved.

[JSQLEngine] Issues with `DSIExtJResultSets` returned from `SqlDataEngine.openTable()` are not being communicated to users. This issue is resolved.

[ODBCClient] System alerts during conversion of retrieved results are ignored; this can cause the application to inadvertently read invalid values without alerting the user. This issue is resolved.

[ODBC] iODBC allows users to set `SQL_ATTR_ENABLE_AUTO_IPD` as a connection attribute in ODBC 3.X mode. This issue is resolved.

[JDBC Client] Service entries of `java.sql.Driver` type are missing from client jars. This prevents the Auto-Loading feature for JDBC client from functioning. This issue is resolved.

[Support] The function `EncodingInfo::GetMaxCodeUnitsInCodePoint()` may not return the correct value for encoding type `ENC_EUC_JP` and `ENC_KANJIEUC_OU_TD`. This can lead to miscalculated column length and unexpected truncation errors. This issue is resolved.

[SQL Engine] Values lists that include NaNs may not sort correctly. This has been resolved. NaN is now always treated as the largest value in the list.

[Simba.Net] Users may not be able to access the driver properties `DSI_DRIVER_LOG_QUERY_AT_PREPARE` and `DSI_DRIVER_USE_STDLIB_STRTOD` from `CLIDSI` properly. This issue is resolved.

[Support] Under some circumstances target buffers cannot hold the source data of `CInterval` data type, which results in a memory corruption issue. This issue is resolved.

[ODBC] Calling `SQLDescField SQL_DESC_BIND_TYPE` and a `NULL ValuePtr` can cause the application to quit unexpectedly. This issue is resolved.

[SEN] Under some circumstances the `CToSql` converter may be created before metadata is properly set. This issue is resolved.

[ODBC] When `SQLGetData()` is called with an insufficient buffer it returns truncated data as expected, but also consume the rest data so that following calls to `SQLGetData()` will return no data. This issue is resolved.

[API] The description for `GenerateCatScalarFn` incorrectly stated the parameter list requires exactly 2 children. This has been resolved. The description now reads that it requires at least 2 children.

[SQL Engine] Issues could result when clients did not implement `Execute` and `RetrieveData` of `DSIExtScalarFunction.h`. This has been resolved. Default implementations of `Execute` and `RetrieveData` of `DSIExtScalarFunction.h` now alert the user if Client doesn't implement them.

[ODBC] Users are not alerted when the result for `SQLRowCount` is truncated. This issue is resolved.

[JavaSupport] Commas in the argument list of the function `JDBCESaper` may not be correctly parsed. This issue is resolved.

[ODBC] On Solaris Sparc, `SQLGetRowCount ( )` may return incorrect results when `ROW_COUNT_UNKNOWN` is expected. This issue is resolved.

[ODBC] The error message `StrRightTruncErr` cannot be retrieved under certain conditions. This issue is resolved.

### SimbaEngine 10.1.4

Encoding errors in the error message XML files have been corrected. Japanese translation of error messages has been improved.

[JDBC] Dates that require a four-digit year can now be converted to strings correctly.

[ODBC] Drivers that use different application encoding per connection would, on some occasions, experience encoding problems during conversion. This issue is resolved.

[Support] In some cases a date to string conversion could return some randomly incorrect characters. This issue is resolved.

[Support] When a column is rebound between two `SQLFetch` calls to the same type but with different precision, scale, or length attributes, the attributes cached at the time of first column binding would not be updated. This issue is resolved.

[Support] An error message is generated if `UseSSL` or `IntegratedSecurity` are set to invalid values, rather than failing silently.

[Support] When performing a conversion from UTF-8 to LATIN 1 on AIX, HP\_UX, and Solaris SPARC, invalid UTF-8 encoding in the source could cause the program to stop responding. This issue is resolved.

[Support] A `RegexMatch ( )` function has been added to `simba_wstring`.

[SimbaClient] In debug mode, an `Assert` can no longer be triggered when a second `Execute` is used after `SQLMoreResults` is called.

[SimbaClient] An issue linked to the `UseIntegratedSecurity` attribute could make it impossible to enable Kerberos and SSL. This issue is resolved.

[SQLEngine] The `ETConvert` node would not always correctly handle wide-character to binary conversion. This issue is resolved.

[SQLEngine] The order in which tables are involved in a JOIN operation will now be deterministic, helping optimize performance.

### 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.



[SQLEngine] 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.