



LAE 5.0.1

Release Notes

Version 1.0

LAVASTORM
analytics



Copyright

© THE CONTENTS OF THIS DOCUMENT ARE THE COPYRIGHT OF LAVASTORM ANALYTICS LIMITED. ALL RIGHTS RESERVED. THIS DOCUMENT OR PARTS THEREOF MAY NOT BE REPRODUCED IN ANY FORM WITHOUT THE WRITTEN PERMISSION OF LAVASTORM ANALYTICS.

Disclaimer

No representation, warranty or understanding is made or given by this document or the information contained within it and no representation is made that the information contained in this document is complete, up to date or accurate. In no event shall Lavastorm Analytics be liable for incidental or consequential damages in connection with, or arising from its use, whether Lavastorm Analytics was made aware of the probability of such loss arising or not.

Contents

- 1.0 Lavastorm Analytics Engine 5.0.1 3**
 - 1.1 Overview 3
 - 1.2 Supported Platforms 3
- 2.0 New Features and Enhancements..... 3**
 - 2.1 Password Policy 3
- 3.0 Lavastorm Analytics Library..... 4**
- 4.0 Bug Fixes and Corrections..... 4**
- 5.0 Backwards Compatibility 7**
- 6.0 Java Node Development..... 7**

1.0 Lavastorm Analytics Engine 5.0.1

1.1 Overview

Release 5.0.1 of the Lavastorm Analytics Engine is a maintenance fix release, containing several bug fixes as well as the Password Policy feature included in other recent releases.

1.2 Supported Platforms

For the BRE client and Lavastorm Desktop:

- Windows 8 (32-bit and 64-bit)
- Windows 7 (32-bit and 64-bit)—note that BRE and Desktop are 32-bit applications that run on either 32-bit or 64-bit versions of Windows7

For the LAE Server:

- Linux (x86; 32-bit and 64-bit)
 - Red Hat Enterprise Linux 5
 - Red Hat Enterprise Linux 6
 - Oracle Enterprise Linux 5
 - Oracle Enterprise Linux 6
- Oracle Solaris
 - Solaris 11 (SPARC; 64-bit)
 - Solaris 10 (SPARC; 64-bit)
- Windows Server 2008

Please note that we have dropped support for Windows XP. Microsoft dropped support for Windows XP in April 2014, and we are following suit

2.0 New Features and Enhancements

2.1 Password Policy

LAE now allows the changing of passwords directly from BRE in addition to the existing command line utilities, and it provides support for password policy enforcement including:

- Password expiration
- Password expiration warning periods
- Password complexity rules (e.g. password must contain an uppercase character, number, non-alphanumeric, etc.)

The password policy can be configured via a new LAE property, *ls.brain.passwdPolicy*, which points to a password policy configuration file. An example of this file has been provided in the *conf/brain/server* directory of your installation folder named *passwdPolicy*. The *ls.brain.passwdPolicy* property is undefined by default in the *ls_brain.prop* file found in the *conf/brain* directory of your installation folder.

Additional notes about this feature:

- Upon enabling a password policy that includes password expiration, all existing users' passwords will immediately expire.
- The *changeLaePassword* utility has been modified so that providing the existing password for a user is no longer necessary. This is to facilitate administration of passwords that have

been lost or forgotten under the assumption that access to this utility is restricted to administrator-type users.

- An expired password cannot be changed through BRE. An administrator must change the password from the server side.

3.0 Lavastorm Analytics Library

Installed along with LAE 5.0.1 is the Lavastorm Analytics Library (LAL), including Simba Technologies Hive drivers. In order for the Hive Metadata Query, Hive Sample, and Hive Join nodes to function, a reboot of the system might be required following installation of these drivers. An option to reboot the system is presented at the end of the installation.

This option is not presented during silent installation, or if Visual Studio 2010 Redistributables is already present on the target machine. Please reboot the system manually following silent installation to enable the Hive nodes.

4.0 Bug Fixes and Corrections

1. Clarified the help content for the `bool()` BRAINscript function to accurately reflect its behavior when converting numeric values to booleans. Issue 1495
2. In the documentation for 'filter' the example given is:
`filter(¬-null, list(null, 1, null, 2, 3, null, 4)) # value:`
`{ 1 2 3 4 }`
`not-null.filter(list(null, 1, null, 2, 3, null, 4)) # value: { 1 2 3 4 }`
 This has been replaced with '&isNotNull'. Issue 1395
3. When running BRXs with `laeControl` or LXAs with the logistics manager, if the BRXs or LXAs contained nodes which produced output with column names outside of the 7-bit US-ASCII, the run would fail with an error of "Error while lexing spec format: Input length = 1". This has been fixed. Issue 1392
4. A bug with the "Copy as Excel" and "Save as Excel" functionality in the BRD Viewer has been fixed.
 Previously, when exporting BRD data to excel via the BRD Viewer, "\" characters were interpreted as unicode escape sequences that needed to be converted.
 This would lead to errors when trying to export strings like "c:\tmp\myFile.txt" to Excel using the BRD Viewer. Issue 1364
5. Logistics Manager now automatically filters out certain characters in all name fields. The disallowed characters are filtered out as the user types.
 The disallowed characters are the following:
`/, ", ', @, !, #, $, %, \, ^, &, *, ~, ` , ; , : , | , < , > , (,) , { , } , [,] , ,, . , ?`
 Issue 1341
6. Fixed an issue where, in rare circumstances, data written to an output pin of a Java-based node containing an escape sequence could become corrupted due to the escape character being stripped out of the data. Issue 1317
7. When saving a BRX or LXA within BRE, newlines were encoded using the windows '\r\n' newline characters.
 However, when BRE attempts to execute a node and sends the node information to the server, multi-line parameters were sent with unix '\n' newline separators.

In general this would not cause problems except for some multi-line parameters. In these cases, the server is expecting to receive the parameters with '\n' newline separators, which could result in some issues with the execution of some BRX and LXA files.

This has now been fixed to ensure BRE doesn't force the use of windows newline characters.

For BRX's and LXA's which have previously been saved and exhibit this problem, opening the corresponding BRG within BRE and re-saving as a BRX/LXA will resolve the problem.

Issue 1149

8. A regression was introduced in the JDBC Store node in LAE 5.0.0 (and LAE 4.6.2) as a result of fixing issue 4294, which allowed for users to enter table and schema names with spaces in them.

The node was modified to automatically add double-quote characters for all table and schema names in the generated SQL INSERT statement.

The issue further stated that the DbTable, SchemaName, and CatalogName parameters should not contain quoting characters (e.g. "" or []) - as the node would insert the required quote characters automatically.

While using the double-quote character is correct ANSI SQL syntax, not all databases accept this as a quoting character.

Further, the use of quoting within table and schema names can be used to indicate case-sensitivity rather than just ensuring that the table/schema name will work when it contains things such as space characters.

In order to resolve this, the JDBC driver is consulted to determine which characters are allowed in identifier names.

If there are any characters within the schema or table name which are not allowed in the identifier names according to the driver, then the node will automatically insert quote characters around the identifier (schema or table) to ensure that this works correctly.

Rather than simply using double-quotes, the quote character to use is obtained from the JDBC driver.

There is an existing parameter on the JDBC Store node: TableAndSchemaCase.

This parameter is used to determine whether all table and schema names should preserve the case, convert to upper case, or convert to lower case.

Previously, this had a default of "To Upper", which would convert the case of all table and schema names provided to upper case - regardless of whether they are quoted or not.

A new option has been added to the parameter: "Use Driver Settings".

This option ensures that the driver is queried to determine whether such identifiers should be converted to upper case, to lower case, or have their case preserved.

In general, this option should always be used as it will provide the correct interpretation of the case preservation mechanism according to the database.

The use of the other options will therefore be deprecated over time.

If the "Use Driver Settings" option is selected and the user enters a table or schema which is surrounded by the quote characters recognized by the JDBC driver, or enters a table or schema which needs to be quoted, then the database settings reported by the driver are used to determine what to do with the case of such quoted identifiers.

Using this setting, the node will handle these differently than unquoted identifiers (which do not need to be automatically quoted).

This allows the user more control of the case settings of the table and schema.

Using an Oracle database, all identifiers are automatically converted to upper case unless they are quoted, where the case is preserved.

As an example, if a user is created within an Oracle database, using:

```
create user "myUser" identified by ....
```

And subsequently a table is created, using:

```
create table "myUser".myTable
```

Then the database will have a user "myUser" with a table "MYTABLE" - as the username was quoted, while the table was not.

When entering this into the JDBC Store node, with the "Use Driver Settings" (default) option selected, if the SchemaName parameter is set to: "myUser", while the DbTable parameter is set to: myTable, the table will correctly be converted to uppercase in the insert statement, and the case of the schema will correctly be preserved.

While the "Use Driver Settings" option should always provide the correct behavior, since the default of the TableAndSchemaCase parameter has been modified, this default can be reset on a server wide basis by modifying the compatibility-5.0.1.0.prop file found in the LAE server installation directory, under the conf/brain/ subdirectory.

In order to revert to the previous "To Upper" default for the TableAndSchemaCase, simply open the compatibility-5.0.1.prop file and uncomment (remove the '#' at the start of the line) the following line:

```
#ls.brain.node.java.DBLoader.tableAndSchemaCase="To Upper"
```

In addition to this a new parameter has been added "EmptySchemaAsNull" which specifies what to do when the SchemaName parameter is not populated. When set to true, this parameter will set the schema to null when retrieving the metadata from the database. When set to false (default), .

When the default of false is used, this will set the schema to the empty string (""). This retrieves the metadata for tables without a schema. When set to null this means that the schema name should not be used to narrow the search. Generally this parameter should be left as the default - which would be the same as the existing behavior. However, you can explicitly choose for this to be null - in particular this seems to work around an issue connecting using the MS SQL Server 4.0 driver when attempting to publish to a table which was created without a schema name.

Issue 962

9. The BRAINScript expression double now converts the strings NaN, Inf, and Infinity into the IEEE floating point values for Not-a-Number and Infinity. Issue 1451
10. The section of the BRE Help regarding Advanced Preferences has been updated to correctly match the changes to the dialog layout made in LAE 5.0.
In addition, the Help button on each of the tabs within the Advanced Preferences dialog is now correctly linked to the corresponding sub topic for that tab in the help. Issue 969
11. When running the Salesforce nodes provided in LAL from BRE, temporary BRD files would remain on disk after node execution causing the LAE temporary directory to eventually grow in size permanently. This issue has now been fixed. Issue 1797

5.0 Backwards Compatibility

1. As noted in Bug Fix #8, the JDBC Store Node has been reworked. Please refer to Bug Fix #8 for details about possible compatibility issues with older LAE versions.

6.0 Java Node Development

The LAE Java library versioning scheme has been updated to facilitate easier development of Java nodes across releases.