



- [Site Map](#)
- [Topic Index](#)

[Home](#)[Install](#)[Migrate](#)[Start Authoring](#)[StreamSQL](#)[Test/Debug](#)[Adapters](#)[API](#)[Admin](#)[Samples](#)[Studio](#)[Reference](#)

Current Location: [Home](#) > StreamBase Release Notes

StreamBase Release Notes

StreamBase 6.3.13

Date: 11 Jun 2010

This page contains the release notes for the current version of StreamBase.

Contents

[Before You Install](#)

[Resolved Limitations](#)

[Fixed in Previous Releases](#)

[Known Limitations](#)

Before You Install

Before you install StreamBase:

1. Read this Release Notes document, paying special attention to the [Issues and Cautions](#) section.
2. Confirm that your target system matches the hardware and software requirements discussed on [Supported Configurations](#).
3. Read the [Installation Guide](#).

More Information

- ◆ For a summary of what's new in this release, see [New and Noteworthy](#).
- ◆ If you have used a StreamBase 3.7 release, see the important information in [Migrating to Newer StreamBase Releases](#).
- ◆ If you are new to StreamBase, read [Getting Started](#) after installation.

See the [documentation home page](#) to begin exploring the StreamBase documentation.

If you are reading this document in printed form, note that the online version of the StreamBase documentation is available in several ways:

- On the installation CD, open `doc/index.html` in a browser.

StreamBase Release Notes

- From the installed file system, open the documentation home page in a browser. The default locations are:

UNIX /opt/streambase/doc/index.html

Windows C:\Program Files\StreamBase Systems\StreamBase.n.m\doc\index.html

- In StreamBase Studio, select Help Help Contents from the top menu.
- On Windows, open Start (All) Programs StreamBase n.m StreamBase User Documentation
StreamBase User Guide - Help Viewer
- On UNIX at a terminal window's command prompt, run **sbhelp &**.

Issues and Cautions

Take note of the following issues in the current StreamBase release.

Reset Perspectives

For any 6.x release, if you have installed any preview release of StreamBase Studio on your computer, you must reset your perspectives in both StreamBase Studio and StreamBase Manager to pick up the changes in the final release.

In both StreamBase Studio and StreamBase Manager, invoke Window Reset Perspective. For StreamBase Studio, do this in both SB Authoring and SB Test/Debug perspectives.

Do Not Use Studio 3.x Workspace with Studio 5.x or 6.x

Releases of StreamBase Studio starting with 5.0 use a new Eclipse workspace format that is incompatible with the workspace from previous releases. When Studio 5.0 or later is first started and it prompts for the location of its workspace, use the default location or specify a new, empty directory. Do NOT specify the location of a StreamBase Studio 3.x workspace. Studio 5.0 or later issues a warning dialog if it detects an older format workspace.

Customer Support

[Contact Information](#) contains information about available support options. You can send us e-mail or get information from your StreamBase Systems representative about our support portal.

Resolved Limitations

Limitations Resolved in 6.3.x

Fixed in 6.3.13	
Number	Resolution
SB-17189	A cache used in conjunction with Query Tables was not correctly functioning, which caused excessive object creation during Query Table writes, which in turned caused excessive garbage collection activity. This was fixed.
SB-17185	StreamBase Server collects statistics as it runs and emits those as tuples from the <code>system</code> container's <code>stat</code> stream. The collection of statistics for the <code>stat</code> stream was optimized to improve caching and to prevent allocating excessive objects. The default size of the <code>stat</code> stream's tuple cache was changed to dynamically grow as needed by the application. You can control the <code>stat</code>

StreamBase Release Notes

	stream's cache size by means of the system property <code>streambase.max-stat-tuples-cache</code> . The default value of <code>-1</code> means the stat stream cache can grow as much as memory allows.
Fixed in 6.3.12	
Number	Resolution
SB-17079, SB-16966	Previous releases would store extra information for every row of a Query Table, which doubled the memory requirements for Query Tables. This was corrected.
SB-16800	The StreamBase Alpha Trading Systems EMAPI Order Entry Adapter had some problems with rounding and scaling. These were fixed.
Fixed in 6.3.11	
Number	Resolution
SB-16261	In previous releases, if a CSV File Writer adapter was marked to not start with the application and not open the file during initialization, and the adapter was never started before the application was shut down, a spurious warning would be displayed during shutdown. If an error port was configured for the adapter, an error would be reported when attempting to shut down the application. This was fixed.
SB-16243	A rare condition relating to manipulating arcs in Studio could cause a <code>NullPointerException</code> error. This was corrected.
SB-16375	Using Ctrl+O to open Outline View in Studio could cause a <code>NullPointerException</code> error. This was corrected.
Fixed in 6.3.10	
Number	Resolution
SB-15898	The CSV File Reader adapter now reports a failure to open its specified CSV file on its event port, if enabled.
SB-15797	An error using NaNs in Query Tables was identified and corrected.
Fixed in 6.3.9	
Number	Resolution
SB-15660	A problem was identified when using a Metronome operator with a Query Table downstream that could result in a <code>NullPointerException</code> error. This was corrected.
SB-15230, SB-14140	Query operators were ignoring the Row Limit setting when the operator returned aggregate results, and when group options were set. These problems were corrected.
SB-15211	In previous releases, the JMS adapter could silently drop tuples if a message converter's schema did not exactly match the adapter's schema. A console message and error tuple were added to report this condition.
SB-14772	The FIX adapter was updated to incorporate a change in the way the CameronFIX FIX engine located its data dictionary file.
SB-12071	Two methods in the .NET client API were producing incorrect results for Field objects whose type is <code>DataType.TUPLE: Schema::Field::GetHashCode()</code> and <code>Schema::Field::Equals()</code> . The .NET API was corrected.
Fixed in 6.3.8	
Number	Resolution
SB-14996	Leading and trailing whitespace is now automatically trimmed around network settings in the properties for the TIBCO Rendezvous adapters. Inadvertent spaces were causing "argument conflict" errors.
SB-14922	

StreamBase Release Notes

	In previous releases, the Binary File Reader adapter could show a warning on the console in cases where the Binary File Writer adapter truncated an existing output file (when set to do so), but no records were yet written. The Binary File Writer was updated to correct for these conditions.
SB-14871	A difference in sorting of nulls between memory and disk Query Tables was found and fixed.
SB-14851	The Deutsche Bank AutobahnFX Trading System adapter was updated such that, when building spot, forward, and swap ladders, it filters out quotes that return <code>isValid=false</code> .
SB-14824	StreamBase Studio was updated to flag the current canvas as changed (and thus needing to be saved) for text-only changes in the Description field for components, including Notes. Studio does not run a typechecking pass when it detects changes only in Description fields.
SB-14699	The JMS external adapter could exit with an illegal argument exception when dequeuing with certain schemas. This was corrected.
SB-14583	When the FIX adapter is disconnected from a counterparty, a tuple is generated by the FIX Input adapter on the Admin message output port with the <code>__ExtraInfo</code> field's value set to "Disconnect". As of 6.3.8, this tuple now also has the <code>SenderCompID</code> and <code>TargetCompID</code> fields correctly set to help identify the session that has just disconnected.
Fixed in 6.3.7	
Number	Resolution
SB-14791	The Deutsche Bank Autobahn Trading System adapter would sometimes leave an orphaned process running after disconnection. This was corrected.
SB-14780	When using the Query operator with a Materialized Window data construct, the Query Settings tab could show settings for an ordered index as if it were unordered. This was fixed.
SB-14723	In previous releases, the Deutsche Bank Autobahn Trading System adapter would correctly reconnect to the Autobahn service after disconnection, but would incorrectly ignore forwards after reconnection. The cause was identified and corrected.
SB-14613	The StreamBase Manager utility was updated with the ability to stop and start adapters hosted in submodules.
SB-14401	Studio could appear to hang in cases where you tried to stop a running application with a running feed simulation, but the application was blocked waiting for the feed simulation to disconnect from the server. Prior to this release, you could unblock Studio by force-stopping the sbd and sbd-java processes. As of 6.3.7, changes were implemented to allow Studio to not block if feed simulations are having trouble disconnecting from the server.
Fixed in 6.3.6	
Number	Resolution
SB-14495	In previous releases, a large number of clients rapidly connecting to and disconnecting from the server could cause memory issues. This was corrected.
SB-14437	In previous releases, calling a custom adapter from a child module could result in a null pointer exception. This was fixed.
SB-14430	When using the CameronFIX engine with the FIX adapter, log files now include the full outgoing FIX message.
SB-14421	In previous releases of the client API, using <code>Tuple.setField()</code> on fields of type <code>Tuple</code> with values of type <code>String</code> could cause a null pointer exception. This was fixed.
SB-14401	In the StreamBase client API, the <code>com.streambase.sb.Tuple</code> class now fully implements the standard <code>java.io.Serializable</code> interface.

StreamBase Release Notes

	<p>Note</p> <p>Do not assume that the serialized form of any StreamBase class remains constant between StreamBase releases. An instance of a StreamBase class serialized in one release might not be deserializable in another release.</p>
SB-14092	In the StreamBase Client API, the method <code>Schema.Field.checkType(DataType)</code> was deprecated in favor of <code>Schema.Field.checkType(CompleteDataType)</code> , which correctly handles hierarchical types like list and tuple.
Fixed in 6.3.5	
Number	Resolution
SB-14353	The SQL statement edit control in a Query operator associated with a JDBC Table data construct did not honor cut, copy, paste, and select-all commands and shortcuts. This was fixed.
SB-14306	In releases 6.3.0 through 6.3.4, the expression language's <code>format()</code> function returned an <code>IllegalArgumentException</code> for some test cases that worked as expected under release 6.2. The cause was found and corrected.
SB-14241	If the Limit number of output rows option is not enabled for the Query operator, the hover summary text for that operator no longer includes text describing line limits.
SB-14189	In previous releases, the sbundle command would inadvertently include a JAR file twice in the bundle if it was listed both in the <code>jar</code> element of the server configuration file and was also present in a directory specified in the <code>dir</code> element. This was corrected.
SB-13944	When using <u>separate layout</u> files for your EventFlow modules, and when using a third-party version control plug-in with StreamBase Studio, the layout files by themselves were sometimes ignored when Studio generated lists of changed files ready to be checked in. This was corrected. If you encounter this situation, open the EventFlow <code>.sbapp</code> file associated with the <code>.sblayout</code> file you want to check in. Make a simple change and save the EventFlow file. This clears the incorrect bit on the associated layout file and it is now flagged as ready to check in.
SB-13643	In previous releases, when using the Query operator with a JDBC data source connected to an Oracle database, a SQL statement encapsulated in the PL/SQL <code>begin</code> and <code>end</code> statements would return an error if the statement spanned more than one line. This was corrected, and multiline PL/SQL statements that include <code>begin</code> and <code>end</code> are now processed correctly.
SB-13602	In previous releases, when using the FIX adapter, customer-initiated resend requests resulted in the rejection of the resent messages as out of sequence. This condition was corrected.
SB-13428	In previous releases, disk-based Query Tables did not report old values when reporting the results of an update operation. This was corrected so that disk-based Query Tables now report old values the same way as memory-based Query Tables.
SB-13291	In previous releases, in the Feed Simulation Editor, when using the Data File generation method, selecting the First row as header option resulted in slower CSV file parsing as column names were matched with field names. This was corrected so that, when using a data file with a one-to-one column-to-field mapping, the CSV file is read just as fast with the First row as header option selected as without.
Fixed in 6.3.4	
Number	Resolution
SB-14051	In previous releases, the <code>Schema.Field.checkType()</code> method in the client API always returned an exception indicating that the test failed. This method was fixed to return the exception only when the method's test actually does fail.
SB-13962	

StreamBase Release Notes

	In releases 6.3.0 through 6.3.3, the samples <code>custom-simple-function</code> and <code>custom-aggregate-function</code> ran as expected at the command prompt, but failed to run when loaded in Studio. The cause was identified and corrected.
SB-13760	The sbc command, when dequeuing a tuple that contained a list with zero elements, would show an error <code>Cannot infer the element type of a zero-length list</code> . This was fixed.
SB-13637	Windows only. The presence of a file named <code>Program</code> at the root of <code>C:\</code> prevented StreamBase Server from starting. This was corrected.
SB-13570	In previous 6.3.x releases, using auto-completion in a dynamic variable's updating expression field could result in a null pointer exception. The cause was found and corrected.
SB-13315	In previous releases, the sbargen command did not use the same algorithm to locate the StreamBase installation directory as the sbd command. This was fixed.
SB-12570	In previous releases, StreamBase Server could be seen to leak memory if many client connections were made in a short period of time. This was fixed.
Fixed in 6.3.2	
Number	Resolution
SB-13619	For some supported databases where the JDBC connection was configured to retry when a disconnect occurred, and the SQL statement in the Query operator had field parameter substitutions, the reconnect attempt would fail. This was fixed.
SB-13617	When running StreamBase Server as a service on Windows, the Server could receive a shutdown signal when the service-owning user logged out of Windows. This was corrected.
SB-13614	When using the Reuters RMDS Subscribing adapter and the data from Reuters contains partial updates to string fields (which is a rarely encountered mechanism), update tuples were incorrectly handled and could be lost. There were two cases: when using the adapter's Send Thin Tuples property, partial string updates resulted in lost tuples. When using standard tuples and the Send Unchanged Fields as Null checkbox is checked, string fields in tuples were null. These conditions were corrected.
SB-13567	In the StreamBase C++ API, using <code>TupleList's operator[]</code> with out-of-range data could result in a segfault, instead of the documented behavior, an assert. This was corrected.
SB-13536	Studio release 6.2 added Export view as PNG to the context menu of the EventFlow Editor. However, the image files actually created by this feature were in GIF format, despite their <code>.png</code> extension. This was corrected, and the feature now creates PNG files.
SB-13501	Despite the removal of string lengths in release 6.3.0, typechecking in the Query operator, when connected to a JDBC data source, was still testing for string length maximums in query responses returned from connected databases such as Oracle 10. This was corrected.
SB-13472	The StreamBase to StreamBase Input and Output adapters were not reporting parameter errors until runtime. This was corrected to report such errors at typecheck time.
SB-13448	When using the Query operator to update a row in a Query Table, the presence of the expression <code>if isnull()</code> would inadvertently suppress the output of the <code>old.fieldname</code> fields. This was fixed.
SB-12604	In previous releases, when using groups in the EventFlow Editor, it could be difficult to select exactly one arc inside a group. This was corrected.
SB-11576	In previous Studio releases, if you opened a project, opened an application in the project, deleted the project, and then selected the application in the EventFlow Editor, StreamBase Studio generated duplicate File Deleted dialog boxes. This was corrected.
Fixed in 6.3.1	

StreamBase Release Notes

Number	Resolution
SB-13373	A 64-bit byte-swapping issue was corrected that affected communication between a Windows client and StreamBase Server on Solaris.
SB-13355	In previous Studio releases, if you opened two copies of the same EventFlow application in different projects, Studio could become confused when you closed one of the copies, and thereafter fail to show editable values in the Properties view. This was fixed.
SB-13287	In previous Studio releases, in the Properties view for a Query Table, if you specified a secondary index of type Unordered (hash), that selection was honored and saved correctly. However, the next time you entered the Properties view for the same Query Table, and opened the Edit Secondary Index dialog, the secondary index type reverted to Ordered (btree). This situation was corrected.
SB-13268	In previous Studio releases, when you used Extract as Module to create a new module from components that referenced a named schema, the named schema was not copied to the new module. This was fixed.
SB-13207	In previous Studio releases, when set to use a <u>separate layout file</u> for an EventFlow application, if you moved a component <u>group</u> as a whole, the components inside that group did not preserve their layout information when the group was next opened. This was fixed.
SB-13187	In release 6.3.0, in the C++ Client API, the CompleteDataType's operator== method was performing incomplete comparisons for lists and tuples. As of 6.3.1, CompleteDataType's operator== compares all members of lists and tuples, such that two CompleteDataTypes of type DataType::LIST are only considered equal if both have the same element CompleteDataType, and two CompleteDataTypes of type DataType::TUPLE are only considered equal if both have the same schema.
SB-13101	In previous releases, the SMTP Reader adapter required SMTP commands to be sent with uppercase letters (HELO, RCPT, and so on). This was corrected to accept lowercase and mixed case commands.
SB-12774	In previous releases, a Java bug could trigger a race condition between two instances of StreamBase Server, or between the server and certain external adapters. This release includes a workaround for the Java bug.
SB-12681	In previous releases, when using Query operators with Query Tables, an input string value longer than a string returned from a table would appear as the returned value overwriting the input value. This only occurred when input values other than the primary key were passed into the result tuple, and did not affect the actual value returned from the table. This condition was corrected.
SB-12372	In release 6.2.x, bundled JAR files were not automatically deleted after exiting from running a <u>bundle file</u> on Windows. This was fixed.
SB-11103	In previous releases, an EventFlow application file was marked as needing to be saved as soon as it was opened. This was corrected so that EventFlows are so marked only when the user has made interactive changes.
Fixed in 6.3.0	
Number	Resolution
SB-13139	Previous StreamBase releases failed to call C++ custom aggregate functions when one of the arguments was null. This limitation was resolved.
SB-12954	In previous releases, if Studio was launching an application on a port that was currently in use by a running sbd that was launched outside of Studio, Studio would incorrectly try to use that server, and cause it to shut down when Studio was stopping the launched application. This was corrected, and Studio now avoids using a port already in use by a running StreamBase Server started outside Studio.
SB-12942	

StreamBase Release Notes

	Global editing actions such as copy, cut, paste, select-all, undo, and redo, and the standard keyboard shortcuts for those actions, would affect the Editor subtab of the EventFlow Editor, even if initiated in another subtab. That is, if you copied and pasted an operator in the Editor subtab, then switched to the Parameter subtab and performed editing tasks that included pasting, the Parameter subtab paste operation also inadvertently reapplied the last paste of the Editor subtab. This was corrected.
SB-12779	On Windows, with multiple instances of StreamBase Server running as a service on the same machine, event log messages from each server instance were hard to distinguish. This was fixed by adding your assigned service name to the Source field of each event log entry that comes from the server startup code. Remember that most messages from a running Server now come from the log4j logging system. You can configure those messages with the log4j configuration file, as described in Running StreamBase Server in Background Mode .
SB-12400	In previous releases, when using Studio's Visual Debugger, the Pattern operator was failing to turn blue to show it was reached when stepping through an application. This was corrected.
SB-11341	In previous releases, the method used to start and shut down StreamBase Server prevented use of the Java method <code>Runtime.getRuntime().addShutdownHook()</code> in customer client code. As of release 6.3.0, this limitation is lifted and <code>addShutdownHook()</code> can now be used in client code.
SB-11087	In releases 6.0 through 6.2, when you clicked a Show in Help link in the Properties view's Functions pane, the Studio Help window opened to the top of the Expression Language page. As of 6.3.0, clicking Show in Help now correctly opens directly to the same function in the Help window that is currently selected in the Functions pane. This restores the Show in Help functionality of releases 3.x and 5.x.

Fixed in Previous Releases

For the list of issues resolved in previous releases, see:

- [Release Notes for StreamBase 6.2.x](#)
- [Release Notes for StreamBase 6.1.x](#)
- [Release Notes for StreamBase 6.0.x](#)
- [Release Notes for StreamBase 5.1.x](#)
- [Release Notes for StreamBase 5.0.x](#)
- [Release Notes for StreamBase 3.x](#)

Known Limitations

This section describes known limitations in the current release of StreamBase. Each item includes a tracking number, description, and whenever possible, one or more workarounds.

SB-14862	Custom Java operators and adapters cannot include their own <code>janino.jar</code> .	
	Description	The StreamBase Server executable, <code>sbd[.exe]</code> , incorporates a customized version of the Janino libraries that include bug fixes that have not yet been absorbed by the Janino project. If you include the standard <code>janino.jar</code> implementation of Janino as part of a custom operator or that JAR file conflicts with the version embedded in <code>sbd</code> .
	Workaround	Call Janino as part of StreamBase's <code>server.jar</code> , or see http://code.google.com/p/janino-strict/ for public access to StreamBase's Janino bug fixes.
	sb-config --cxx returns a setting for G++ 4.3 on newer Linux distributions	

StreamBase Release Notes

	Description	On Linux, in StreamBase samples that build C++ code, some of the example Makefiles show sb-config --cxx as the correct usage to determine the G++ compiler to use. On newer distributions, this command returns the distribution's default G++ 4.3 compiler, which is not supported for building StreamBase client code, and sb-config does not honor the CC and CXX environment variables.
	Workaround	On such systems, install GCC and G++ 4.2, and edit the Makefiles to specify CXX=g++-4 .
SB-13398	Filter operators with very large predicate settings may result in a runtime EvalException error.	
	Description	Filter operators with several output ports and complex expressions to evaluate for each port may result in a <code>com.streambase.sb.runtime.exceptions.EvalException</code> error when attempting to run the application that contains such a filter.
	Workaround	Add another Filter operator and split the predicate settings between the two Filter operators.
SB-13144	Variables view does not refresh under some conditions.	
	Description	While debugging an application in Studio, after setting a list element value in the Variables view while the debugger is suspended, the Variables view may intermittently fail to refresh.
	Workaround	Close and reopen the Variables view.
SB-13113	Application output view does not display UTF-8 characters.	
	Description	You can set StreamBase Server to process Unicode characters in streams by setting the Java property <code>streambase.tuple.charset="UTF-8"</code> . However, StreamBase Studio does not default display non-Western characters when showing streams in the Application Output view or other views.
	Workaround	You can set StreamBase Studio to display Unicode characters with two steps: <ul style="list-style-type: none"> • Set <code>streambase.tuple.charset="UTF-8"</code> as part of the <code>STREAMBASE_STUDIO_VMARGS</code> environment variable, or set <code>STREAMBASE_TUPLE_CHARSET=UTF-8</code>. • Set Studio's default font in the preferences for General > Appearance to a Unicode-enabled font that contains the extended character set you want to display. For example, to display Chinese, Arabic, or Hebrew characters, select a font that contains those characters.
SB-13021	JDBC database errors related to string lengths.	
	Description	Due to the removal of string lengths in StreamBase as of release 6.3, some static checking of table sizes is no longer possible. This can lead to database-specific errors, possibly including truncation, if a StreamBase application attempts to insert into a JDBC table string values that are too wide for the JDBC table's schema.
	Workaround	None.
SB-12824	Java and C++ format doubles differently.	
	Description	When displaying integer-like values of type double, Java code outputs the value with a decimal point and zero appended (34.0), whereas C++ code outputs the same value without those features (34). Thus, the same double values output to <code>stdout</code> with sbconsole and jsbcconsole are formatted differently.
	Workaround	None.
SB-12165	Attempting to compare large EventFlow application files can result in an out of memory error.	
	Description	When using the Compare With -> Each Other context menu in Studio's Package Explorer to compare two selected EventFlow application files, you may get an <code>java.lang.OutOfMemoryError</code> for large applications.
	Workaround	Try increasing the minimum heap size of Studio by using the <code>STREAMBASE_STUDIO_VMARGS</code> environment variable, adjusting the <code>-Xms</code> setting to a larger value.

StreamBase Release Notes

SB-12136	On 32-bit Windows, some third-party applications can interfere with StreamBase Server starting.	
	Description	In recent releases, StreamBase Server is started by default with a requested maximum memory of 512 MB. Certain third-party applications, including Apple's Bonjour program, load themselves in the middle of available RAM, which can prevent StreamBase Server from reserving a contiguous block of memory.
	Workaround	You can (1) reduce StreamBase Server's maximum memory size request by adjusting the <code>java_heap_size</code> section of the server configuration file, (2) remove the competing program, such as Bonjour, or (3) upgrade to a supported 64-bit Windows platform.
SB-12016	Studio HTML display issues with Linux and Firefox 3.x.	
	Description	Recent Linux distributions use Firefox 3.x as the default browser. Firefox 3 splits the display into a separate piece called Xulrunner so that it can be called by applications other than Firefox. In a default Linux installation, this can cause some display problems with StreamBase Studio. Some symptoms are: <ul style="list-style-type: none"> • The Welcome page has a white background and no longer shows the bubble graphic. • Studio Help pages and the sbhelp utility open in the browser, not in separate Help windows.
	Workaround	Install the xulrunner compatibility package for your Linux distribution. For example, for Ubuntu use the following command: <code>sudo apt-get install xulrunner-gnome-support</code>
SB-11964	Case-sensitive Unicode regular expressions require Java 1.6.	
	Description	When running StreamBase with Java 1.5, with Unicode support enabled (using <code>-Dstreambase-tuple-charset=UTF8</code>), the <code>regexmatch()</code> expression language function behaves as if an ignore-case option was enabled.
	Workaround	To have both case-sensitive regular expressions and Unicode support, you must run StreamBase with Java 1.6, which is the default JDK shipped with StreamBase.
SB-11850	Properties view Help button shows correct Help page but wrong TOC entry	
	Description	In the Properties view, when you click the circled question mark Help button, Studio Help opens showing the documentation entry for the currently open Properties view tab of the currently selected EventFlow component. However, the TOC entry in the Help window sometimes does not match the open page.
	Workaround	None. The correct Help page is shown.
SB-11387	Visual debugger slows application execution.	
	Description	Running an application while it is being debugged by the StreamBase visual debugger incurs a performance cost in terms of the peak message rate that the application can handle. For this reason, do not enable the debugger for production. Also, be aware of the possible changes in speed when enqueueing large amounts of data to an application being debugged.
	Workaround	None.
SB-11279	JDBC Operator, no blob inserts with DB2 and batching.	
	Description	Due to DB2 driver issues, the JDBC Operator cannot insert blob fields when batching is on. This is specific to DB2 and does not affect other supported databases.
	Workaround	None.
SB-11152	Error message shows on wrong Properties view tab for the Aggregate operator.	
	Description	When setting up an Aggregate operator with a group option, some typechecking error messages appear only in the General tab, not the Group Options tab where the error occurred.
	Workaround	None.

StreamBase Release Notes

SB-11130	Focus in the Debug view is sometimes lost while stepping through debugging.	
	Description	Focus in the Debug view can be intermittently lost while stepping through an application at high speed, such as when frequently pressing the F6 button.
	Workaround	If the focus becomes lost in the Debug view during execution stepping, you can reselect any item in the execution stack of the Debug View and continue stepping.
SB-11117	Test Editor might not update changes in an open Feed Simulation file.	
	Description	If a StreamBase Feed Simulation file is modified while a Test Editor is open that references the simulation, the Test Editor may not update its contents until it is closed and reopened.
	Workaround	Close the Test Editor before editing a Feed Simulation file referenced in the test.
SB-10983	Reuters RFA Publishing Adapter not shipped for Linux	
	Description	Because of a change in C++ libraries, the Reuters RFA Publishing external adapter is not shipped for Linux in StreamBase 6.x.
	Workaround	Use the new Reuters RMDS Publishing embedded adapter.
SB-10792	The Query operator converts a JDBC data type differently when using explicit schemas than when using default data types.	
	Description	When explicit schemas are used, the Query operator does not use database metadata to determine the mapping between JDBC data types and StreamBase data types. The JDBC data type <code>Types.TIMESTAMP</code> is normally converted to an interval timestamp since it has a time, but no date component. But when explicit schemas are used, there is no database type information, so <code>Types.TIMESTAMP</code> is converted to an absolute timestamp.
	Workaround	None.
SB-10971	CSV Reader sample application must be reloaded.	
	Description	The CSV Reader sample application from previous releases no longer passes typechecking in StreamBase 6.x.
	Workaround	Reload the CSV Reader sample shipped with release 6.x.
SB-10486 (1)	On-disk format for query tables changes in 6.x.	
	Description	The on-disk format for disk-based query tables changed in 6.0 as a result of adding support for hierarchical data and upgrading the underlying disk database library.
	Workaround	If you have query tables in existing applications, you must convert them to the new format by dumping to a CSV file in your current StreamBase release and reloading into the new StreamBase release.
SB-10486 (2)	Hierarchical data restrictions.	
	Description	Patterns by value do not support using nested data for their on value portion. Similarly, nested data cannot be used as the index or primary key of a query table.
	Workaround	Design schemas so that the intended sort field or key is a top-level numeric or timestamp field as described in StreamBase Pattern Matching Language .
SB-10363	No StreamBase data type corresponding to Sybase timestamp.	
	Description	When used with Sybase data sources, StreamSQL fails to convert timestamps, resulting in type errors. This is because the Sybase <code>timestamp</code> is based on the Sybase <code>varbinary</code> data type.
	Workaround	In StreamSQL code that will be used for Sybase data sources, use the Sybase <code>datetime</code> data type instead of <code>timestamp</code> .
SB-10298	Hierarchical schemas defined in StreamSQL cannot be re-used in StreamBase Studio.	
	Description	StreamBase does not provide a way to share a hierarchical schema between StreamSQL and EventFlow application types.
	Workaround	Use the Named Schemas and Saved Schemas features of Studio.

StreamBase Release Notes

#73006, SB-7550	Limit on Feed Simulation Editor fields.	
	Description	Very large feed simulations with more than about 160 fields can crash StreamBase Studio when you click the Customize Fields button.
	Workaround	Limit the size of feed simulations or break large simulations into two pieces.
#69299, SB-7446	Starting a StreamBase application fails with a <code>Not enough space</code> error.	
	Description	On small systems, StreamBase may run out of virtual memory during the pre-compilation phase. This is reported as a <code>CompilationException: I/O exception: Not enough space</code> error message from the StreamBase Server.
	Workaround	Increase the size of your system's paging file.
#58027, SB-7170	Module reference sometimes fails to typecheck.	
	Description	StreamBase Studio may report a typecheck error on a module reference, with a message similar to: Duplicate stream <code>stream-name</code> detected When this message appears, StreamBase Studio refuses to typecheck the module reference and prevents you from continuing to develop operators downstream, even if the application has no other errors and is runnable by either StreamBase Server or StreamBase Studio.
	Workaround	StreamBase is investigating this limitation. For now, you might be able to work around the problem by restructuring the flow of your application, moving the location of the module reference in the graph.
#56829, SB-7148	Invalid byte error when opening EventFlow application.	
	Description	Studio sometimes fails to open an EventFlow application with this error: <code>org.xml.sax.SAXParseException: Invalid byte 1 of 1-byte UTF-8 sequence</code>
	Workaround	Perform the following steps to recover: <ol style="list-style-type: none"> 1. Close the EventFlow Editor, if it is open. 2. Open the <code>sbapp</code> file in a text editor. For example, in StreamBase Studio, right-click the file and choose <code>Open With</code> <code>Text Editor</code>. 3. In the source, find any multi-byte characters that might have been pasted from a multi-line source (for example, formatted text that was copied from Microsoft Word into a description field in an EventFlow Properties view). 4. Remove the multibyte characters or any other characters that might be causing trouble. Typically, the characters to remove will appear as a square box. Replace them with characters that you type in manually. 5. Confirm that the application opens in the EventFlow Editor (repeat 2 to reassign that file to the EventFlow editor).
#55871, SB-7125	Group labels can appear truncated.	
	Description	Group labels longer than 12 characters are truncated and elided if the group is open, and truncated to longer than 5 characters if the group is closed.
	Workaround	Pad the end of the group label with space characters to prevent truncation.
#52959, SB-7075	Changing perspectives can rearrange the Studio toolbar.	
	Description	The position of toolbar sections can change when you go from perspective to perspective, especially when going from the Authoring to the Demo perspective and back.
	Workaround	Eclipse toolbars do not have fixed locations, so this is expected behavior.
#49739, SB-6984	Actions unavailable when all folders are selected.	
	Known Limitations	

StreamBase Release Notes

	Description	After you collapse a Group in the EventFlow editor, Select All selects all the items in the group. However, actions performed on the group do not succeed. For example, Delete does not delete items inside a group.
	Workaround	Uncollapse the group before performing the action.
#48285, SB-6862	Not all databases have native blob support.	
	Description	For example, Microsoft SQL Server and Sybase do not have blob support. For these databases you can write blobs into varbinary and possibly other types. But when blobs are read from a select statement, they are returned as strings, not blobs.
	Workaround	Add a Map operator and convert the string to a blob.
#36261, SB-6838	Studio 3.x workspaces not compatible with Studio 5.x.	
	Description	StreamBase Studio 5.0 introduced a new workspace format that is incompatible with the workspace format from previous releases. Studio 5.0 detects an attempt to use a previous version's workspace and prevents you from proceeding.
	Workaround	Start Studio 5.0 with a new, empty workspace. Then use Import StreamBase Projects from StreamBase Studio 3.x. See Migrating Studio 3.x Projects .
#35859, SB-6356	Large applications cannot be debugged in StreamBase Studio.	
	Description	When running an application in debug mode in StreamBase Studio that contains many operators (in the range of hundreds, including all modules), it is possible to cause Studio to display and log an error message about running out of handles. The error message may vary, but the symptom is that Studio is unable to display all the intermediate streams in the Application Output View or the Manual Input View.
	Workaround	Limit the number of output streams the application will expose when running in debug mode by setting the server's JVM property to include the following: <code>streambase.codegen.intermediate-stream-dequeue-regex=(regex_pattern)</code> For example, setting the JVM arguments to include: <code>-Dstreambase.codegen.intermediate-stream-dequeue-regex=Map\d</code> allows any intermediate stream that contains the pattern of a Map followed by a digit to be exposed as a dequeueable stream. All other intermediate streams normally available in debug mode will remain available.
#35743, SB-6346	Custom function C++ code typechecking is delayed until the application is generated.	
	Description	Most portions of a StreamBase application are typechecked in real time as you add features to an EventFlow or StreamSQL application. However, starting with release 5.0, typechecking for a custom function written in C++ is delayed until the application is generated and run.
	Workaround	None. Be aware that typechecking errors may not be reported as usual when using custom C++ functions.
#35429, SB-6327	StreamBase Server cannot run applications that generate class files with too long a name.	
	Description	If you use modules, StreamBase can generate internal class names that are too long for the name system, causing problems running StreamBase Server.
	Workaround	To avoid this problem, use small module reference names and minimize module nesting as much as possible.
#33330, SB-6241	When an application is running in StreamBase Studio on Windows, the Manual Input view and other views are not visible.	
	Description	

StreamBase Release Notes

		This is a known Eclipse bug. The problem occurs when you use the tab key. Borders around other decorations, and some controls can disappear.
	Workaround	Minimize and then maximize the window, or resize the affected view.
#27349, SB-5894	Evaluation exceptions in Data Parallelism key and Subscribe predicate should not be handled by upstream operator.	
	Description	If an evaluation exception occurs for a Data Parallelism key, or in a filtered subscribe predicate, an exception is thrown and caught by the upstream operator, and reported as an error in the upstream operator. One incorrect side effect is that if one predicate in filtered subscribe fails, then predicates that follow it in the list are not executed and tuples are not delivered. Similarly, if other operators are downstream and only one is set for data parallelism, the other operators may not see tuples. Moreover, the order in which operators see tuples is effectively non-deterministic.
	Workaround	Either ensure that you do not have errors in the expressions, or use <code>catchexception()</code> as a wrapper.
#27060, SB-5882	Tuple dimensions do not reset counter when another dimension closes a window early.	
	Description	A tuple-based dimension that closes due to other dimensions (for example, after another dimension times out) does not reset its counter. When this happens, the next window will close based on the number of tuples received, resulting in a partially full dimension.
	Workaround	None.
#26888, SB-5848	Issue with Parameterized Modules and Custom Java Operators or Embedded Adapters.	
	Description	In EventFlow applications, you cannot enter references to parameterized modules in custom operators or embedded adapters. This is due to a GUI limitation, currently under investigation, which prevents you from entering the required values.
	Workaround	Embed your custom Java operator or custom embedded adapter in a StreamSQL-defined module (using the <code>APPLY MODULE</code> statement in the StreamSQL Guide), and use an EventFlow module reference through which you optionally share parameter values.
#26477, SB-5779	SBAPP-to-SSQL Conversion Assistant Resets Module Parameters to Default Values.	
	Description	If you run the SBAPP-to-SSQL Conversion Assistant on an EventFlow application that makes use of module parameter references, the resulting StreamSQL application does not have any module parameter references. They are replaced by their default values.
	Workaround	After the conversion, edit the StreamSQL file to add the non-default module parameter values.
#26255, SB-5749	Feed Simulation Editor's Data Rate Edit and Save Issue on Linux.	
	Description	In StreamBase Studio, the Feed Simulation Editor lets you set the rate (number of tuples per second) at which data will be generated for a stream. On Linux only, if you use the keyboard to enter an integer in the Data Rate text box, the new value might not be saved.
	Workaround	After clicking into the text box and entering the value, press Enter or change the cursor's focus (for example, press Tab). Then press the Save icon or enter Ctrl+S. Note that if you press Ctrl+S while your cursor is within the Data Rate text box, the new value is not saved. You must press Ctrl+S after the cursor has moved away from the text box.
#25228, SB-5628	Include Package with Custom Java Functions.	
	Description	If you use a custom Java function (simple or aggregate) with no package, <code>javac</code> returns a compilation error.
	Workaround	Put the function in a package.
#25009, SB-5602, #25004, SB-5599	Embedded Adapters and StreamSQL.	
	Description	In StreamSQL applications, you cannot specify that an embedded adapter should start in a particular state, and you cannot manage the state of a running adapter in a StreamSQL application (although you can with EventFlow applications).

StreamBase Release Notes

	Workaround	Use an EventFlow application with embedded adapters.
#15898, SB-5078	C++ Custom Aggregate functions cannot return a string.	
	Description	Custom aggregate functions that are implemented in C++ cannot return a <code>string</code> .
	Workaround	Create the custom aggregate in Java.
#14807, SB-4951	Problem running large applications in debug mode.	
	Description	Very large applications that run in standard mode may not start in debug mode, with errors such as <code>code too large</code> .
	Workaround	Organize the application into smaller modules.
#14763, SB-4937	Problem when tuple in top-level module consumes large amounts of CPU time.	
	Description	If the processing of a single tuple in the top-level module consumes a large amount of CPU time, contained parallel modules that are connected directly to output streams will see their output delayed until the tuple/operator releases the CPU and the tuples can be passed to the network.
	Workaround	Put the time-consuming computation into its own parallel module. This workaround may not solve the issue in all cases, because inter-module queues can grow very large, but may help in many cases.
#14745, SB-4931	SBAPP-to-SSQL Conversion Assistant Does Not Support Parallel Operators.	
	Description	If your EventFlow application file has individual operators that are marked for parallel operation, the SBAPP-to-SSQL Conversion Assistant cannot generate equivalent functionality in the generated StreamSQL application because there is no direct analogy for parallel operators in StreamSQL.
	Workaround	After the conversion, if appropriate for your application (no dependencies on data in other operators), you can use an <code>APPLY PARALLEL MODULE</code> statement or <code>APPLY PARALLEL JDBC STATEMENT</code> in the StreamSQL file. For details on the <code>APPLY</code> statement, see the StreamSQL Guide .
#14207, SB-4854	Typecheck problem with DB2 JDBC data sources.	
	Description	SQL <code>INSERT</code> statements submitted to a JDBC data source in a Query operator or in an <code>APPLY JDBC</code> statement may pass typechecking even when they contain errors. This problem was detected when using an IBM DB2 data source. For example, the following statement passes typechecking even if the named table or fields do not exist in the JDBC data source: <pre>APPLY JDBC DB2 "INSERT INTO mytable (id, name) VALUES ({id}, {name})" FROM InStream INTO OutStream;</pre>
	Workaround	Run the StreamBase application. If an error occurs against an <code>INSERT</code> statement, verify that the table name is correct. You can validate the <code>INSERT</code> statement by running it against a test database.
SB-4625	Spurious typecheck error when saving changes to a data construct associated with an operator.	
	Description	In some circumstances StreamBase Studio generates a typecheck error with a message to remove a row that you cannot remove because the row is not displayed in the Properties view. This can occur when you have edited a data construct associated with an operator and then attempt to save the changes.
	Workaround	You can work around the typecheck error by forcing StreamBase Studio to reread your changes. In the Properties view, click the Refresh icon, or restart the application instance, cut and paste an existing row, or change one character and change it back.
#13017, SB-4559	Join timeout is not asynchronous.	
	Description	Join timeout in the Join operator is not asynchronous. That is, the Join operator does not remove a tuple from its buffer until the timeout has passed and another tuple arrives on the same input stream. Tuples received on one input stream do not affect the buffer of the other input stream.
	Workaround	None.
	Some disk drive configurations can corrupt disk-based query tables.	

StreamBase Release Notes

#10674, SB-4225	Description	Some caching disk drives do not guarantee the sequencing of disk writes, nor do they guarantee all the writes to the drive have actually happened. This behavior improves disk performance, but increases the risk of data corruption in the event of a power outage or equipment failure. StreamBase is vulnerable to this kind of data corruption if your StreamBase application uses disk-based query tables.
	Workaround	Disk drives with this behavior usually have a driver-level setting to turn this capability on or off. For example, the driver may present you with a checkbox labeled Enable write caching on the disk properties dialog. Unchecking this box would prevent the problem from occurring.
#10291, SB-4160	JVM crash during StreamBase Application compilation.	
	Description	When StreamBase Server first loads an application, it compiles it to Java code using the Sun Java compiler. Some StreamBase applications can cause the Sun Java compiler to crash during this compilation. If this happens, StreamBase Server displays a stack trace that mentions <code>OopFlow::build_oop</code> . StreamBase is pursuing this issue with Sun, but in the meantime if you encounter this issue, please contact StreamBase Systems for assistance in modifying your application to avoid the error.
	Workaround	Make slight changes to your application until the error no longer occurs. We are unfortunately unable to provide guidance about what specific changes to make.
#9620, SB-3955	Finding sbd process failure details.	
	Description	When StreamBase Studio runs an application, it launches the server process in the background. If the server process fails for any reason, Studio tries to report the reason to you. However, the actual error message that Studio displays may not contain enough information, depending on the failure.
	Workaround	To get more information about the reason for the sbd process failure, open the Eclipse error log. The error is described in Error Log View in the <i>StreamBase Studio Reference</i> .
#8174, SB-3585	Inconsistent support for undo in StreamBase Studio.	
	Description	Many editing operations in Studio cannot be undone. This includes operations such as renaming objects, editing query tables, copying schemas, adding operators from the palette, editing flow diagrams, simulations, and so on.
	Workaround	The Revert command in the File menu can be used to undo all changes since the last time a file was saved.
#7361, SB-3308, #2585, SB-1601	From Windows, the <code>custom-simple-function</code> and <code>custom-aggregate-function</code> samples do not work with Solaris server or 64-bit Linux server.	
	Description	On Windows, the installed <code>custom-simple-function</code> and <code>custom-aggregate-function</code> samples contain library files (<code>log.so</code> and <code>stdev.so</code> , respectively) that are built for 32-bit Linux. These custom functions will work properly only when StreamBase Studio is connected to a 32-bit Linux server, not a Solaris server or a 64-bit Linux server.
	Workaround	To use the <code>custom-simple-function</code> and <code>custom-aggregate-function</code> samples on a Solaris server: <ul style="list-style-type: none"> • On the Windows or Linux machine where you are running StreamBase Studio, use the StreamBase Samples menu item to import the <code>plugin-function</code> or <code>plugin-aggregate</code> sample into your workspace. • Remove <code>log.so</code> or <code>stdev.so</code> files from the sample project. • Copy <code>/opt/streambase/sample/plugin-function/log.so</code> or <code>/opt/streambase/sample/plugin-aggregate/stdev.so</code> from your Solaris server machine to your Windows client machine. • Use the File Import menu item to import the copied <code>log.so</code> or <code>stdev.so</code> into the sample project.
#7191, SB-3260	Description	Using non-member <code>Schema.Field</code> to access a tuple does not throw a useful exception.

StreamBase Release Notes

		<p>The StreamBase client API does not throw a useful exception if you inadvertently reference a schema while using <code>Schema.Field</code> to access a <code>Tuple</code>. Under C++, no exception is thrown and memory corruption may occur. Under Java, an <code>ArrayIndexOutOfBoundsException</code> is thrown.</p> <p>This applies both to using fields from other schemas, or to using the <code>Field</code> objects that were used to create the <code>Schema</code>. Fields that are used to create a schema are not fully initialized, and so you must use the actual <code>Field</code> from the new <code>Schema</code> object in order to access the fields of a <code>Tuple</code>.</p>
	Workaround	None via the API. You must examine your client code and verify that the correct <code>Schema</code> is referenced.
#6860, SB-8698	Aggregate dimensions do not support advance on infinite dimensions.	
	Description	In an Aggregate operator's Properties View, the Dimensions tab includes a setting for Window Policy. One of the size options is <code>Do not close window based on this dimension</code> . If selected, this dimension cannot cause the closing of a new window for the Aggregate. If a new window is never opened, this creates an infinite sized window that never closes (for the life of the StreamBase Server instance). Application designers may set up this feature, for example, to ensure that an Aggregate processes an entire day's volume of streaming data. However, a limitation of Aggregates with infinite dimensions do not support an <code>Advance</code> value that you set in the <code>Open Policy</code> .
	Workaround	None.
#6539, SB-3121	Mozilla issue with StreamBase Studio Welcome page and Help.	
	Description	On Linux, the StreamBase Studio Welcome page and Help require that Mozilla or Firefox is installed and can be located. If you removed Mozilla, or installed it in a non-standard location, or moved the Welcome page or Help may not appear.
	Workaround	Before using Studio, set the environment variable <code>MOZILLA_FIVE_HOME</code> to the path of your Mozilla installation. For example, on Linux, set it to: <pre>MOZILLA_FIVE_HOME=/usr/lib/mozilla-<version> export MOZILLA_FIVE_HOME</pre>
#5400, SB-8640	Java Operator with zero inputs does nothing.	
	Description	While it is not an error to create a Java Operator with zero input ports, the <code>processTuple</code> method will never be called, thus the Java Operator can never emit any tuples.
	Workaround	None.
#5389, SB-2931	Relocatable RPMs install incorrect symbolic links in <code>/usr</code> .	
	Description	The Linux RPM packages are relocatable. This means that the default installation location of <code>/opt/streambase</code> can be changed to a different location. However, if this is done, symbolic links are still installed in <code>/usr/bin</code> and <code>/usr/lib</code> that point to <code>/opt/streambase</code> .
	Workaround	Either edit the symbolic link by hand, or use the tar file instead of the RPMs to install StreamBase on Linux.
#5327, SB-2891	Error messages while creating Java operators lack detail.	
	Description	Any exceptions during the creation of a Java operator from within StreamBase Studio will result in a <code>Can't make new object: (operator name) message</code> , with no additional information.
	Workaround	Run the application in <code>sbd</code> directly for additional diagnostic information. Verify the correct Java classes are specified in your project's Custom Libraries. Verify the classes are in the JAR files. Verify the manifests of the JAR files are correct. For more information, see Creating Java Clients . The error message may have additional diagnostics.
#4824, SB-2617	For Java operators and modules, typechecking may occur regardless of connections.	
	Description	

StreamBase Release Notes

		Java operators and modules are typechecked independently of their connections to other components. For modules, this may result in errors that do not clearly describe the problem. For example, if you have an application containing a module with its inputs connected, then delete an input stream module references, and then return to the referencing application, the arc (connection) will be dropped. However the following typecheck message will appear: "Error in referencing application: <input> lacks required attribute 'stream'".
	Workaround	None.
#4451, SB-2507	Java API IOException when client program is suspended.	
	Description	A program using the StreamBase Java API may receive an unexpected IOException if it is suspended (using either Ctrl-Z , or <code>kill -SIGSTOP</code>). The IOException is thrown when the program is resumed, or placed in the background. The exception may be thrown even if there is no network traffic at the time. This issue does not apply to Windows systems.
	Workaround	Do not suspend StreamBase clients written in Java.
#4289, SB-2482	Data sent to multiple streams may get out of order.	
	Description	StreamBase Server does not preserve the relative ordering of input tuples received on different streams. This affects tools such as sbfeedsim , which replays recorded data. It also affects the Java API, if tuples are enqueued to different streams.
	Workaround	In some cases you may be able to use a Merge operator to reorder your input data. However, the Merge operator can block in some situations and therefore may not be appropriate. In general, it is best to avoid sending data to different input streams in the first place, if the relative order of tuples on those streams must be preserved.
#3236, SB-1969	StreamBase Studio on Windows triggers Windows firewall when running Server.	
	Description	Users of StreamBase Studio on Windows may find that the Windows Firewall, or a third-party firewall product, prompts for permission to continue when first starting StreamBase Server. The permission message may refer to javaw or to StreamBase Server by name. This is expected behavior of most firewalls. You must grant the requested permission in order to continue using StreamBase Studio on this machine.
	Workaround	After granting permission once, you should not be prompted about this again.
#2811, SB-1769	LockDirectoryException error encountered when starting StreamBase Server.	
	Description	There may be a number of cases when you would encounter the LockDirectoryException error. The error message indicates <code>File exists</code> , either another instance of StreamBase Server is currently running, or the prior instance of Server was improperly shut down.
	Workaround	If another StreamBase Server instance is running, change the <code>.sbconf</code> file to point the <code>data</code> element (within the <code><server> ... </server></code> element) to a different directory. If there is another Server instance running, you must manually delete the directory named in the <code>File exists</code> message.
#2414, SB-8297	Files edited outside StreamBase Studio are not refreshed in Studio automatically.	
	Description	For <code>.sbapp</code> and <code>.sbfs</code> files that were created using StreamBase Studio and are still open in Studio, modifications made using an external editor are not refreshed in the current Studio Editor session.
	Workaround	If edits are made outside of Studio and you now want to use the latest file, right-click the project icon that contains the changes you made and select Refresh.
#2197, SB-1289	IPv6 addresses not supported.	
	Description	StreamBase Server can only connect to IPv4 addresses. If you try to connect using IPv6 addresses, the connection is refused.
	Workaround	None.
	Issues with certain older Linux kernel versions.	

StreamBase Release Notes

#2048, SB-1164	Description	Linux kernel versions 2.6.8-10 have a known bug in their thread handling code. This bug can cause sbd to exit immediately after servicing its first client request. For more information, see http://www.ussg.iu.edu/hypermail/linux/kernel/0410.2/1650.html .
	Workaround	This bug was fixed in Linux kernel 2.6.11. This does not affect the Linux distributions currently supported by StreamBase. If you are running an older kernel in the 2.6.8-10 range, then set the environment variable <code>STREAMBASE_IGNORE_SIGHUP=1</code> to work around this problem. Set this variable in the login environment in which StreamBase Server will run.
#1838, SB-1039	C++ code is not binary compatible between releases.	
	Description	If you use either the C++ client API, plug-in functions, or plug-in aggregates, you must recompile them from source in order to use them in each new release. Without recompiling, StreamBase Server will print an error message at runtime: <pre>bin/sbd: error while loading shared libraries: libsbclient.so.4: cannot open shared object file: No such file or directory</pre>
	Workaround	Recompile your C++ code from your original source against the new headers.
#1456, SB-806	Explicitly-provided schema in loops are order sensitive.	
	Description	If you use a loop, the order of the fields in your explicitly-provided schema must match that of the output schema. Otherwise you will get a typecheck error indicating that your explicitly-provided output schema does not match your actual output schema.
	Workaround	Click on the loop arc, then in the Properties view click the option: Update using the schema that StreamBase Studio detected on one of the Union input ports. This action rearranges the order of the explicitly-provided schema to match that of the input schema. You can also manually change the order of the fields using the Properties View for the loop arc.
#1450, SB-804	init.d script uses a hard-coded port number.	
	Description	UNIX only. If you change <code>/opt/streambase/etc/sbd.sbconf</code> to point to a port other than 10000, then the <code>stop</code> , <code>restart</code> , and <code>status</code> commands in the <code>/etc/init.d/streambase</code> script do not work as expected. Those commands assume that sbd is running on port 10000.
	Workaround	Edit the commands in the <code>/etc/init.d/streambase</code> script to use the <code>-p</code> option, to force sbd to run on the same port specified in the modified <code>sbd.sbconf</code> file.
#1395, SB-765	Clients that exceed page limit buffer size disconnect with no explanation.	
	Description	When a StreamBase client exceeds its maximum number of pages (for example, because it's dequeuing fast enough to keep up with the rate at which tuples are being generated), StreamBase Server disconnects the client and prints a message saying that it has done so. However on the client side, no error is reported. Instead, it looks like an unexplained dropped connection.
	Workaround	Edit the server configuration file for the Server that processes your application, and increase the <code>max-client-pages</code> parameter.

- [Copyright](#) © 2004-2010 StreamBase Systems, Inc. All Rights Reserved.
- [Contact Us](#)