StreamSets Data Collector 2.3.0.0 Release Notes

February 2, 2017

We're happy to announce a new version of StreamSets Data Collector. For important information about upgrading, new features, fixed issues, and known issues, read below.

Important: We highly recommend that you migrate to Java 8 with this version of Data Collector. The Apache Solr and Elasticsearch stage libraries currently require Java 8, and more are sure to follow. We will officially drop support for Java 7 in a future release.

Upgrade

You can upgrade previous versions of Data Collector to version 2.3.0.0. For instructions on upgrading, see the Upgrade Documentation.

Update Elasticsearch Pipelines

As of version 2.3.0.0, Data Collector includes an enhanced Elasticsearch destination that uses the Elasticsearch HTTP API.

The Elasticsearch destination is disabled by default. To upgrade pipelines that use the Elasticsearch destination, you must verify that Java 8 is installed on the Data Collector machine. Elasticsearch is no longer supported on Java 7. You also must remove the Elasticsearch stage library from the blacklist property for stage libraries in the Data Collector configuration file, `$SDC_CONF/sdc.properties`

Due to SDC-5148, upgraded Elasticsearch destinations have the Default Operation property set based on the configuration for the Enable Upsert property:
- With upsert enabled, the default operation is set to INDEX.
- With upsert not enabled, the default operation is set to CREATE which requires a Document ID.

Review all upgraded Elasticsearch pipelines to ensure the Default Operation is set to the correct operation.

Due to SDC-5243, an upgrade of a Cloudera Manager installation might not successfully upgrade Elasticsearch destinations, even after completing the required post upgrade tasks for Elasticsearch destinations. To work around this issue, delete the previous Elasticsearch destination, and then add a new Elasticsearch destination with the same configured properties.

For more information, see Update Elasticsearch Pipelines.

Update Kudu Pipelines

As of version 2.2.0.0, Data Collector supports Apache Kudu version 1.0.x. Earlier Kudu versions are no longer supported. To upgrade pipelines that contain a Kudu destination, upgrade your Kudu cluster to version 1.0.x and then add a stage alias for the earlier Kudu version to the Data Collector configuration file, `$SDC_CONF/sdc.properties` For more information, see Update Kudu Pipelines.

New Features and Enhancements

This version includes new features and enhancements in the following areas.

Multithreaded Pipelines

You can use a multithreaded origin to generate multithreaded pipelines to perform parallel processing.

The new multithreaded framework includes the following changes:

- **HTTP Server origin** - Listens on an HTTP endpoint and processes the contents of all authorized HTTP POST requests. Use the HTTP Server origin to receive high volumes of HTTP POST requests using multiple threads.
- **Enhanced Dev Data Generator origin** - Can create multiple threads for testing multithreaded pipelines.
- **Enhanced runtime statistics** - Monitoring a pipeline displays aggregated runtime statistics for all threads in the pipeline. You can also view the number of runners, i.e. threads and pipeline instances, being used.

CDC/CRUD Enhancements

With this release, certain Data Collector stages enable you to easily process change data capture (CDC) or transactional data in a pipeline. The sdc.operation.type record header attribute is now used by all CDC-enabled origins and CRUD-enabled stages:

CDC-enabled origins:

- The MongoDB Oplog and Salesforce origins are now enabled for processing changed data by including the CRUD operation type in the sdc.operation.type record header attribute.
- Though previously CDC-enabled, the Oracle CDC Client and JDBC Query Consumer for Microsoft SQL Server now include CRUD operation type in the sdc.operation.type record header attribute.
  Previous operation type header attributes are still supported for backward-compatibility.

CRUD-enabled stages:

- The JDBC Tee processor and JDBC Producer can now process changed data based on CRUD operations in record headers. The stages also include a default operation and unsupported operation handling.
- The MongoDB and Elasticsearch destinations now look for the CRUD operation in the sdc.operation.type record header attribute. The Elasticsearch destination includes a default operation and unsupported operation handling.

Multitable Copy

You can use the new JDBC Multitable Consumer origin when you need to copy multiple tables to a destination system or for database replication. The JDBC Multitable Consumer origin reads database data from multiple tables through a JDBC connection. The origin generates SQL queries based on the table configurations that you define.

Configuration

- **Groups for file-based authentication** - If you use file-based authentication, you can now create groups of users when multiple users use Data Collector. You configure groups in the associated `realm.properties`file located in the Data Collector configuration directory, `$SDC_CONF`.

  If you use file-based authentication, you can also now view all user accounts granted access to the Data Collector, including the roles and groups assigned to each user.

- **LDAP authentication enhancements** - You can now configure Data Collector to use StartTLS to make secure connections to an LDAP server. You can also configure the userFilter property to define the LDAP user attribute used to log in to Data Collector. For example, a username, uid, or email address.

- **Proxy configuration for outbound requests** - You can now configure Data Collector to use an authenticated HTTP proxy for outbound requests to Dataflow Performance Manager (DPM).

- **Java garbage collector logging** - Data Collector now enables logging for the Java garbage collector by default. Logs are written to `$SDC_LOG/gc.log`. You can disable the logging if needed.

- **Heap dump for out of memory errors** - Data Collector now produces a heap dump file by default if it encounters an out of memory error. You can configure the location of the heap dump file or you can disable this default behavior.

- **Modifying the log level** - You can now use the Data Collector UI to modify the log level to display messages at another severity level.

Pipelines

- **Pipeline renaming** - You can now rename pipelines.

Origins

- **New HTTP Server origin** - A multithreaded origin that listens on an HTTP endpoint and processes the contents of all authorized HTTP POST requests. Use the HTTP Server origin to read high volumes of HTTP POST requests using multiple threads.

- **New HTTP to Kafka origin -** Listens on a HTTP endpoint and writes the contents of all authorized HTTP POST requests directly to Kafka. Use to read high volumes of HTTP POST requests and write them to Kafka.

- **New MapR DB JSON origin** - Reads JSON documents from MapR DB JSON tables.

- **New MongoDB Oplog origin** - Reads entries from a MongoDB Oplog. Use to process change information for data or database operations.

- **Directory origin enhancement** - You can use regular expressions in addition to glob patterns to define the file name pattern to process files.

- **HTTP Client origin enhancement** - You can now configure the origin to use the OAuth 2 protocol to connect to an HTTP service.

- **JDBC Query Consumer origin enhancements** - The JDBC Consumer origin has been renamed to the JDBC Query Consumer origin. The origin functions the same as in previous releases. It reads database data using a user-defined SQL query through a JDBC connection.

  You can also now configure the origin to enable auto-commit mode for the JDBC connection and to disable validation of the SQL query.

- **MongoDB origin enhancements** - You can now use a nested field as the offset field. The origin supports reading the MongoDB BSON timestamp for MongoDB versions 2.6 and later. And you can configure the origin to connect to a single MongoDB server or node.

- **Oracle CDC Client origin enhancement** - The origin can now track changes from a table whose schema has changed, and may continue to change.

Processors

- **Field Type Converter processor enhancement** - You can now configure the processor to convert timestamp data in a Long field to a String. Previously, you had to use one Field Type Converter processor to convert the Long field to a Datetime, and then use another processor to convert the Datetime field to a String.

- **HTTP Client processor enhancements** - You can now configure the processor to use the OAuth 2 protocol to connect to an HTTP service. You can also configure a rate limit for the processor, which defines the maximum number of requests to make per second.

- **JDBC Lookup processor enhancements** - You can now configure the processor to enable auto-commit mode for the JDBC connection. You can also configure the processor to use a default value if the database does not return a lookup value for a column.

- **Salesforce Lookup processor enhancement** - You can now configure the processor to use a default value if Salesforce does not return a lookup value for a field.

- **XML Parser enhancement** - A new Multiple Values Behavior property allows you to specify the behavior when you define a delimiter element and the document includes more than one value: Return the first value as a record, return one record with a list field for each value, or return all values as records.

Destinations

- **New MapR DB JSON destination** - Writes data as JSON documents to MapR DB JSON tables.

- **Azure Data Lake Store destination enhancement -** You can now use the destination in cluster batch pipelines. You can also process binary and protobuf data, use record header attributes to write records to files and roll files, and configure a file suffix and the maximum number of records that can be written to a file.

- **Elasticsearch destination enhancement** - The destination now uses the Elasticsearch HTTP API. With this API, the Elasticsearch version 5 stage library is compatible with all versions of Elasticsearch. Earlier stage library versions have been removed. Elasticsearch is no longer

supported on Java 7. You'll need to verify that Java 8 is installed on the Data Collector machine and remove this stage from the blacklist property in `$SDC_CONF/sdc.properties` before you can use it.

You can also now configure the destination to perform any of the following CRUD operations: create, update, delete, or index.

- **Hive Metastore destination enhancement** - New table events now include information about columns and partitions in the table.

- **Hadoop FS, Local FS, and MapR FS destination enhancement** - The destinations now support recovery after an unexpected stop of the pipeline by renaming temporary files when the pipeline restarts.

- **Redis destination enhancement** - You can now configure a timeout for each key that the destination writes to Redis.

Executors

- **Hive Query executor enhancements:**
  - The executor can now execute multiple queries for each event that it receives.
  - It can also generate event records each time it processes a query.

- **JDBC Query executor enhancement** - You can now configure the executor to enable auto-commit mode for the JDBC connection.

Data Formats

- **Whole File enhancement** - You can now specify a transfer rate to help control the resources used to process whole files. You can specify the rate limit in all origins that process whole files.

Expression Language

- **New record functions** - You can use the following new record functions to determine information about records:
  - `record:fieldAttribute (<attribute name>, <field path>)` Returns the value of the specified field header attribute for the specified field.
  - `record:fieldAttributeOrDefault (<attribute name>, <field path>, <default value>)` - Returns the value of the specified field header attribute for the specified field. If the attribute does not exist or if the field is null, returns the default value.

- **New string functions** - You can use the following new string functions to transform string data:
  - `str:urlEncode (<string>, <encoding>)` - Returns a URL encoded string from a decoded string using the specified encoding format.
  - `str:urlDecode (<string>, <encoding>)` - Returns a decoded string from a URL encoded string using the specified encoding format.

- **New time functions** - You can use the following new time functions to transform datetime data:
  - `time:dateTimeToMilliseconds (<Date object>)` - Converts a Date object to an epoch or UNIX time in milliseconds.
  - `time:extractDateFromString(<string>, <format string>)` Extracts a Date object from a String, based on the specified date format.

    - `time:extractStringFromDateTZ (<Date object>, <timezone>, <format string>)` - Extracts a string value from a Date object based on the specified date format and time zone.

- **New and enhanced miscellaneous functions** - You can use the following new and enhanced miscellaneous functions:
    - `offset:column(<position>)`- Returns the value of the positioned offset column for the current table. Available only in the additional offset column conditions of the JDBC Multitable Consumer origin.
    - `every` function - You can now use the function with the hh() datetime variable in directory templates. This allows you to create directories based on the specified interval for hours.

Fixed Issues

The following table lists some of the known issues that are fixed with this release.

For the full list, click here.

JIRA	Description
SDC-5224	MapR stage libraries should not contain the SDC RPC to Kafka and UDP to Kafka origins. These stages do not support the MapR version of Kafka at this time.
SDC-5195	The Field Renamer produces a null pointer exception because of a null field.
SDC-5038	The Hive Metadata processor doesn't allow columns that start with an underscore.
SDC-4954	The Cassandra destination has a version mismatch on netty jar files.
SDC-4898	Pipeline labels are not listed alphabetically on the Home page.
SDC-4609	A pipeline stops unexpectedly when the Google Bigtable destination encounters non-numeric values for timestamps.
SDC-4608	The Google Bigtable destination encounters a null pointer exception when the stage uses the time associated with the record as the time basis and the specified field doesn't exist.
SDC-4578	The Redis destination should send the record to error instead of stopping the pipeline when it encounters a Jedis exception.
SDC-4462	The Kafka Consumer origin in a cluster pipeline might commit the wrong offsets.
SDC-3115	The Kafka Consumer origin in a cluster pipeline should manage the Kafka offsets instead of Spark Streaming to avoid using Spark checkpoints for restarting.

Known Issues

Please note the following known issues with this release.

For a full list of known issues, check out our JIRA.

JIRA	Description
SDC-5243	After upgrading a Cloudera Manager installation, the Elasticsearch destination might not be successfully upgraded even after completing the required post upgrade tasks for Elasticsearch destinations. Workaround: Delete the previous Elasticsearch destination, and then add a new Elasticsearch destination with the same configured properties.
SDC-5160	When the Field Hasher processor is configured to use Hash to Target and to continue when it encounters a missing field, the processor writes random hashed data to the target field instead of dropping the target field.
SDC-5148	A pipeline with an Elasticsearch destination upgraded to the current release has the Default Operation property set based on the configuration for the Enable Upsert property: - With upsert enabled, the default operation is set to INDEX. - With upsert not enabled, the default operation is set to CREATE which requires a DocumentId. Workaround: Review all upgraded Elasticsearch pipelines to ensure the Default Operation is set to the correct operation.
SDC-5141	Due to a limitation in the Javascript engine, the Javascript Evaluator issues a null pointer exception when unable to compile a script.
SDC-5039	When you use the Hadoop FS origin to read files from all subdirectories, the origin cannot use the configured Hadoop FS User as a proxy user to read from HDFS. Workaround: If you need to use a proxy user to read from all subdirectories of the specified directories, set the HADOOP_PROXY_USER environment variable to the proxy user in libexex/_cluster-manager script, as follows: `export HADOOP_PROXY_USER = <proxy-user>`
SDC-4876	When running Data Collector on Java 8, you might receive “Cannot load driver” errors for stages that use JDBC drivers. Workaround: On the Legacy Drivers tab, configure the Driver Class Name property.
SDC-4212	If you configure a UDP Source or UDP to Kafka origin to enable multithreading after you have already run the pipeline with the option disabled, the following validation error displays: `Multithreaded UDP server is not available on your platform.` Workaround: Restart Data Collector.

SDC-4128	In cluster mode, Data Collector does not generate log files for worker Data Collectors.
SDC-3944	The Hive Streaming destination using the MapR library cannot connect to a MapR cluster that uses Kerberos or username/password login authentication.
SDC-3234	Cluster streaming pipelines that run on YARN use the YARN user instead of the Data Collector user to run executors.
SDC-3133	When you upgrade Data Collector from the RPM package, the environment configuration file `$SDC_DIST/libexec/sdcd-env.sh` is overwritten. Workaround: Back up the `sdcd-env.sh` file before you upgrade.
SDC-2822	If you configure a Kafka Producer destination to write one message per batch, and then use a cluster pipeline to process that data from the Kafka cluster, the cluster pipeline might encounter an out of memory error.
SDC-2586	To process records larger than 1 MB, you must configure the DataFactoryBuilder.OverRunLimit property. However, this property is not configurable in the Data Collector configuration file, `$SDC_CONF/sdc.properties` Workaround: Set the value of DataFactoryBuilder.OverRunLimit property in the SDC_JAVA_OPTS environment variable in the Data Collector environment file, `$SDC_DIST/libexec/sdc-env.sh` or `$SDC_DIST/libexec/sdcd-env.sh` Set the property greater than the largest record you want to process. For example, to process records up to 2 MB, set the property to 2097152 as follows: `SDC_JAVA_OPTS="-DDataFactoryBuilder.OverRunLimit=2097152"`
SDC-2374	A cluster mode pipeline can hang with a CONNECT_ERROR status. This can be a temporary connection problem that resolves, returning the pipeline to the RUNNING status. If the problem is not temporary, you might need to manually edit the pipeline state file to set the pipeline to STOPPED. Edit the file only after you confirm that the pipeline is no longer running on the cluster or that the cluster has been decommissioned. To manually change the pipeline state, edit the following file: `$SDC_DATA/runInfo/<cluster pipeline name>/<revision>/pipelineState.json` In the file, change CONNECT_ERROR to STOPPED and save the file.
SDC-1731	When using the Kafka Consumer or Kafka Producer on HDP 2.3 with Kerberos enabled, set the Kafka broker configuration property `security.inter.broker.protocol` to PLAINTEXT. When enabling Kerberos, HDP 2.3 sets the `security.inter.broker.protocol` property to PLAINTEXTSASL, which is not supported. If the property is not set to PLAINTEXT, when the pipeline starts, validation errors indicate a problem connecting to Kafka.

SDC-891	At this time, writing to error records to file is not supported for cluster mode pipelines. Workaround: Write error records to Kafka or to an SDC RPC pipeline.
SDC-890	For cluster mode pipelines configured to stop on error or to stop upon reaching a memory limit, the Data Collector cannot stop all worker pipelines as expected. Workaround: To stop all pipelines, use the Stop icon in the Data Collector console.

Contact Information

For more information about StreamSets, visit our website: http://streamsets.com/.

Check out our Documentation page for doc highlights, what's new, and tutorials: streamsets.com/docs

Or you can go straight to our latest documentation here:
https://streamsets.com/documentation/datacollector/latest/help

To report an issue, ask for help, or find out about our next meetup, check out our Community page:
http://streamsets.com/community/.

For general inquiries, email us at info@streamsets.com.