Cloudera DataFlow for Data Hub 7.2.6

Cloudera DataFlow for Data Hub Release Notes

Date published: 2020-05-01
Date modified: 2020-12-11

https://docs.cloudera.com/

Legal Notice

Contents

What's New in Cloudera DataFlow for Data Hub 7.2.6

Cloudera DataFlow for Data Hub 7.2.6 includes components for Flow Management, Streaming Analytics, and Streams Messaging. Learn about the new features and improvements in each of these components.

What's New in Flow Management

Learn about the new Flow Management features in Cloudera DataFlow for Data Hub 7.2.6.

This release introduces the Technical Preview of the following cluster definitions for Flow Management in CDP Public Cloud:

- Flow Management Light Duty for Google Cloud Storage
- Flow Management Heavy Duty for Google Cloud Storage

These cluster definitions support installing Flow Management clusters running Apache NiFi and Apache NiFi Registry.

Note: Google Cloud Storage Flow Management clusters are available for Technical Preview. Cloudera encourages you to explore these technical preview features in non-production environments and provide feedback on your experiences through the *Cloudera Support Portal*.

Related Information

Cloudera Support Portal

What's New in Streams Messaging

Learn about the new Streams Messaging features in Cloudera DataFlow for Data Hub 7.2.6.

Kafka

Remove usage of non-FIPS compatible algorithms in Kafka

Murmur3 hashing is introduced for the log cleaner when it builds offset maps. In addition, to the introduction of the new hashing algorithm, the default algorithm used is also changed to Murmur3. The previous default was MD5. If required, MD5 can still be used by adding cloudera.log.cleaner .hashing.algorithm=MD5 to the Kafka Broker Advanced Configuration Snippet (Safety Valve) for kafka.properties property in Cloudera Manager.

Schema Registry

There are no new features for Schema Registry in this release.

Streams Messaging Manager

CSD support to configure caching in SMM authorizer

SMM request processing is sped up by introducing an authorization cache. The default TTL of the cache is 30 seconds and it is configurable in Cloudera Manager. Setting the TTL to 0 disables the cache entirely.

The affected versions are Cloudera Manager 7.2.4 and higher and CDH 7.2.1 and higher, Cloudera Manager 7.3.0 and higher and CDH 7.1.6 and higher.

SMM automatically configures SRM in Cloudera Manager

SMM automatically configures the SRM connection based on a service dependency. Manual configuration options are removed. This feature affects Cloudera Manager versions 7.2.3 and higher

and CDH versions 7.2.3 and higher, Cloudera Manager versions 7.3.0 and higher and CDH versions 7.1.6 and higher.

Streams Replication Manager

SRM is available in Data Hub and CDP Public Cloud

SRM can now be provisioned in CDP Public Cloud with Data Hub. The default Streams Messaging cluster definitions are updated to include SRM. For more information, see Streams Messaging cluster layout and Creating your first Streams Messaging cluster.

SRM high availability mode

You can now deploy SRM in high availability mode. For more information, see Enable high availability for Streams Replication Manager.

Replication-specific Kafka Connect REST servers

SRM can now run multiple drivers in the same cluster (high availability). To make this possible, the SRM driver role now deploys a Kafka Connect REST server for each replication that you set up and configure. These REST servers ensure communication between the different instances of the driver role and make replication with multiple drivers in a single cluster possible.

If required, you can configure these REST servers in Cloudera Manager with the Streams Replication Manager's Replication Configs property and two specific prefixes. For more information, see Configuring replication specific Kafka Connect REST servers .

The grace and retention period as well as the collection frequency of SRM Service role metrics are configurable

Configuration properties related to the SRM Service role's metric processing are added. These properties give users the ability to configure the grace and retention periods as well as the collection frequency of SRM Service role metrics. The grace and retention periods can be configured directly with the following Cloudera Manager properties:

- SRM Service Metrics Grace Period (streams.replication.manager.service.streams.metrics.grace)
- SRM Service Metrics Retention Period (streams.replication.manager.service.streams.metrics.retention)

Metric collection frequency can be configured through Streams Replication Manager's Replication Configs with the metrics.period property. The metrics.period property can only be configured on a replication level. For example:

```
primary->secondary.metrics.period=60
```

What's New in Streaming Analytics

There are no new features for Streaming Analytics in Cloudera DataFlow for Data Hub 7.2.6.

Component Support in Cloudera DataFlow for Data Hub 7.2.6

Cloudera DataFlow for Data Hub 7.2.6 includes the following components.

Flow Management clusters

- Apache NiFi 1.11.4
- Apache NiFi Registry 0.5.0

Streams Messaging clusters

- Apache Kafka 2.5.0

- Schema Registry 0.9.1
- Streams Messaging Manager 2.1.0
- Streams Replication Manager 1.0.0

Streaming Analytics clusters

- Apache Flink 1.10

Supported NiFi Extensions

Apache NiFi 1.11.4 ships with a set of Processors, Controller Services, and Reporting Tasks, most of which are supported by Cloudera Support. Review the supported extensions and avoid using any unsupported extensions in your production environments.

Supported NiFi Processors

Apache NiFi 1.11.4 ships with a set of Processors, most of which are supported by Cloudera Support. You should be familiar with the available supported Processors, and avoid using any unsupported Processors in production environments.

Additional Processors are developed and tested by the Cloudera community but are not officially supported by Cloudera. Processors are excluded for a variety of reasons, including insufficient reliability or incomplete test case coverage, declaration of non-production readiness by the community at large, and feature deviation from Cloudera best practices.

Supported processors

- AttributesToCSV
- AttributesToJSON
- Base64EncodeContent
- CalculateRecordStats
- CaptureChangeMySQL
- CompressContent
- ConnectWebSocket
- ConsumeAMQP
- ConsumeAzureEventHub
- ConsumeEWS
- ConsumeGCPubSub
- ConsumeJMS
- ConsumeKafka
- ConsumeKafka_0_10
- ConsumeKafka_1_0
- ConsumeKafka_2_0
- ConsumeKafkaRecord_0_10
- ConsumeKafkaRecord_1_0
- ConsumeKafkaRecord_2_0
- ConsumeMQTT
- ConsumeWindowsEventLog
- ControlRate
- ConvertAvroSchema
- ConvertAvroToJSON
- GeoEnrichIPRecord
- GetAzureEventHub
- GetAzureQueueStorage
- GetCouchbaseKey
- GetFile
- GetFTP
- GetHBase
- GetHDFS
- GetHDFSFileInfo
- GetHDFSSequenceFile
- GetHTMLElement
- GetHTTP
- GetIgniteCache
- GetJMSQueue
- GetJMSTopic
- GetKafka
- GetMongoRecord
- GetSFTP
- GetSolr
- GetSplunk
- GetSQS
- GetTCP
- GetTwitter
- HandleHttpRequest
- PutDistributedMapCache
- PutDruidRecord
- PutDynamoDB
- PutElasticsearch
- PutElasticsearch5
- PutElasticsearchHttp
- PutElasticsearchHttpRecord
- PutElasticsearchRecord
- PutEmail
- PutFile
- PutFTP
- PutGCSObject
- PutGridFS
- PutHBaseCell
- PutHBaseJSON
- PutHBaseRecord
- PutHDFS
- PutHive3QL
- PutHive3Streaming
- PutHiveQL
- PutHiveStreaming
- PutHTMLElement
- PutInfluxDB
- PutJMS

- ConvertAvroToORC
- ConvertAvroToParquet
- ConvertCharacterSet
- ConvertCSVToAvro
- ConvertJSONToAvro
- ConvertJSONToSQL
- ConvertRecord
- CreateHadoopSequenceFile
- CryptographicHashAttribute
- CryptographicHashContent
- DeleteAzureBlobStorage
- DeleteAzureDataLakeStorage
- DeleteByQueryElasticsearch
- DeleteDynamoDB
- DeleteElasticsearch5
- DeleteGCSObject
- DeleteGridFS
- DeleteHBaseCells
- DeleteHBaseRow
- DeleteHDFS
- DeleteS3Object
- DeleteSQS
- DetectDuplicate
- DistributeLoad
- DuplicateFlowFile
- EncryptContent
- EnforceOrder
- EvaluateJsonPath
- EvaluateXPath
- EvaluateXQuery
- ExecuteGroovyScript
- ExecuteInfluxDBQuery
- ExecuteProcess
- ExecuteScript
- ExecuteSQL
- ExecuteSQLRecord
- ExecuteStreamCommand
- ExtractAvroMetadata
- ExtractGrok
- ExtractHL7Attributes
- ExtractImageMetadata
- ExtractText
- FetchAzureBlobStorage
- FetchDistributedMapCache
- FetchElasticsearch
- FetchElasticsearch5
- FetchElasticsearchHttp
- FetchFile
- FetchFTP
- FetchGCSObject
- HandleHttpResponse
- HashAttribute
- HashContent
- IdentifyMimeType
- InvokeAWSGatewayApi
- InvokeHTTP
- InvokeScriptedProcessor
- JoltTransformJSON
- JoltTransformRecord
- JsonQueryElasticsearch
- ListAzureBlobStorage
- ListDatabaseTables
- ListenHTTP
- ListenRELP
- ListenSyslog
- ListenTCP
- ListenTCPRecord
- ListenUDP
- ListenUDPRecord
- ListenWebSocket
- ListFile
- ListFTP
- ListGCSBucket
- ListHDFS
- ListS3
- ListSFTP
- LogAttribute
- LogMessage
- LookupAttribute
- LookupRecord
- MergeContent
- MergeRecord
- ModifyHTMLElement
- MonitorActivity
- Notify
- ParseCEF
- ParseEvtx
- ParseSyslog
- PartitionRecord
- PostHTTP
- PublishAMQP
- PublishGCPubSub
- PublishJMS
- PublishKafka
- PublishKafka_0_10
- PublishKafka_1_0
- PublishKafka_2_0
- PublishKafkaRecord_0_10
- PublishKafkaRecord_1_0
- PublishKafkaRecord_2_0
- PutKafka
- PutKinesisFirehose
- PutKinesisStream
- PutKudu
- PutLambda
- PutORC
- PutParquet
- PutRecord
- PutRiemann
- PutS3Object
- PutSFTP
- PutSNS
- PutSolrContentStream
- PutSolrRecord
- PutSplunk
- PutSQL
- PutSQS
- PutSyslog
- PutTCP
- PutUDP
- PutWebSocket
- QueryCassandra
- QueryDatabaseTable
- QueryDatabaseTableRecord
- QueryElasticsearchHttp
- QueryRecord
- QuerySolr
- QueryWhois
- ReplaceText
- ReplaceTextWithMapping
- ResizeImage
- RetryFlowFile
- RouteHL7
- RouteOnAttribute
- RouteOnContent
- RouteText
- ScanAttribute
- ScanContent
- ScanHBase
- ScrollElasticsearchHttp
- SegmentContent
- SelectHive3QL
- SelectHiveQL
- SplitAvro
- SplitContent
- SplitJson
- SplitRecord
- SplitText
- SplitXml
- TagS3Object

- FetchGridFS
- FetchHBaseRow
- FetchHDFS
- FetchParquet
- FetchS3Object
- FetchSFTP
- FlattenJson
- ForkRecord
- GenerateFlowFile
- GenerateTableFetch
- GeoEnrichIP
- PublishMQTT
- PutAzureBlobStorage
- PutAzureDataLakeStorage
- PutAzureEventHub
- PutAzureQueueStorage
- PutBigQueryBatch
- PutBigQueryStreaming
- PutCassandraQL
- PutCloudWatchMetric
- PutCouchbaseKey
- PutDatabaseRecord
- TailFile
- TransformXml
- UnpackContent
- UpdateAttribute
- UpdateCounter
- UpdateRecord
- ValidateCsv
- ValidateRecord
- ValidateXml
- Wait
- YandexTranslate

Supported NiFi Controller Services

Apache NiFi 1.11.4 ships with a set of Controller Services, most of which are supported by Cloudera Support. You should be familiar with the available supported Controller Services, and avoid using any unsupported Controller Services in production environments.

Additional Controller Services are developed and tested by the Cloudera community but are not officially supported by Cloudera. Controller Services are excluded for a variety of reasons, including insufficient reliability or incomplete test case coverage, declaration of non-production readiness by the community at large, and feature deviation from Cloudera best practices.

- AvroReader
- AvroRecordSetWriter
- AvroSchemaRegistry
- AWSCredentialsProviderControllerService
- AzureStorageCredentialsControllerService
- AzureStorageCredentialsControllerService_v12
- CassandraSessionProvider
- CouchbaseClusterService
- CouchbaseKeyValueLookupService
- CouchbaseMapCacheClient
- CouchbaseRecordLookupService
- CSVReader
- CSVRecordLookupService
- CSVRecordSetWriter
- DatabaseRecordLookupService
- DatabaseRecordSink
- DBCPConnectionPool
- DBCPConnectionPoolLookup
- DistributedMapCacheClientService
- DistributedMapCacheLookupService
- DistributedMapCacheServer
- DistributedSetCacheClientService
- DistributedSetCacheServer
- DruidTranquilityController
- ElasticSearchClientServiceImpl
- ElasticSearchLookupService
- ElasticSearchStringLookupService
- FreeFormTextRecordSetWriter
- GCPCredentialsControllerService
- GrokReader
- HBase_1_1_2_ClientMapCacheService
- HBase_1_1_2_ClientService
- HBase_1_1_2_ListLookupService
- HBase_1_1_2_RecordLookupService
- HBase_2_ClientMapCacheService
- HBase_2_ClientService
- HBase_2_RecordLookupService
- Hive3ConnectionPool
- HiveConnectionPool
- HortonworksSchemaRegistry
- JMSConnectionFactoryProvider
- JndiJmsConnectionFactoryProvider
- JsonPathReader
- JsonRecordSetWriter
- JsonTreeReader
- KafkaRecordSink_1_0
- KafkaRecordSink_2_0
- KeytabCredentialsService
- KuduLookupService
- ParquetReader
- ParquetRecordSetWriter
- PrometheusRecordSink
- RedisConnectionPoolService
- RedisDistributedMapCacheClientService
- RestLookupService
- ScriptedLookupService
- ScriptedReader
- ScriptedRecordSetWriter
- ScriptedRecordSink
- SimpleDatabaseLookupService
- SimpleKeyValueLookupService
- SimpleScriptedLookupService
- SiteToSiteReportingRecordSink
- StandardHttpContextMap
- StandardProxyConfigurationService
- StandardRestrictedSSLContextService
- StandardS3EncryptionService
- StandardSSLContextService
- Syslog5424Reader
- SyslogReader
- VolatileSchemaCache
- XMLReader
- XMLRecordSetWriter

Supported NiFi Reporting Tasks

Apache NiFi 1.11.4 ships with a set of Reporting Tasks, most of which are supported by Cloudera Support. You should be familiar with the available supported Reporting Tasks, and avoid using any unsupported Reporting Tasks in production environments.

Additional Reporting Tasks are developed and tested by the Cloudera community but are not officially supported by Cloudera. Reporting Tasks are excluded for a variety of reasons, including insufficient reliability or incomplete test case coverage, declaration of non-production readiness by the community at large, and feature deviation from Cloudera best practices. Do not use these features in your production environments.

- AmbariReportingTask
- ControllerStatusReportingTask
- MetricsEventReportingTask
- MonitorDiskUsage
- MonitorMemory
- PrometheusReportingTask
- QueryNiFiReportingTask
- ReportLineageToAtlas
- ScriptedReportingTask
- SiteToSiteBulletinReportingTask
- SiteToSiteMetricsReportingTask
- SiteToSiteProvenanceReportingTask
- SiteToSiteStatusReportingTask

Unsupported Features in Cloudera DataFlow for Data Hub 7.2.6

Some features exist within Cloudera DataFlow for Data Hub 7.2.6 components, but are not supported by Cloudera.

Unsupported Flow Management features

There are no unsupported Flow Management features in Cloudera DataFlow for Data Hub 7.2.6.

NiFi

There are no updates for this release.

NiFi Registry

There are no updates for this release.

Related Information

Cloudera Community Forum

Unsupported Streams Messaging features

Some Streams Messaging features exist in Cloudera DataFlow for Data Hub 7.2.6, but are not supported by Cloudera.

Kafka

The following Kafka features are not ready for production deployment. Cloudera encourages you to explore these features in non-production environments and provide feedback on your experiences through the *Cloudera Community Forums*.

- Only Java and .Net based clients are supported. Clients developed with C, C++, Python, and other languages are currently not supported.
- While Kafka Connect is available as part of Runtime, it is currently not supported in CDP Public Cloud. NiFi is a proven solution for batch and real time data loading that complement Kafka's message broker capability. For more information, see Creating your first Flow Management cluster.
- The Kafka default authorizer is not supported. This includes setting ACLs and all related APIs, broker functionality, and command-line tools.

Schema Registry

There are no updates for this release.

Streams Messaging Manager

There are no updates for this release.

Streams Replication Manager

There are no updates for this release.

Related Information

Cloudera Community Forum
Creating your first Streams Messaging cluster

Unsupported Streaming Analytics features

Some Streaming Analytic features exist in Cloudera DataFlow for Data Hub 7.2.6, but are not supported by Cloudera.

Flink

The following Flink features are not ready for production deployment. Cloudera encourages you to explore these features in non-production environments and provide feedback on your experiences through the *Cloudera Community Forums*.

- SQL Client

Related Information

Cloudera Community Forum

Apache Patch Information in Cloudera DataFlow for Data Hub 7.2.6

The following sections list patches in each Cloudera DataFlow in Data Hub component, beyond what was fixed in the base version of the Apache component.

NiFi patches

This release provides Apache NiFi 1.11.4 and these additional Apache patches.

- NIFI-7422 and NIFI-7452 - Improvements for Atlas lineage with AWS S3 and Azure ADLS
- NIFI-7436 - Performance improvements on analytics predictions
- NIFI-7683 and NIFI-7994 - ReplaceText issue
- NIFI-7707 - Google Cloud PubSub issue
- NIFI-7968 and NIFI-7954 - Kerberos relogin issues

NiFi Registry patches

This release provides Apache NiFi Registry 0.5.0. There are no additional Apache patches.

Known Issues in Cloudera DataFlow for Data Hub 7.2.6

You must be aware of the known issues and limitations, the areas of impact, and workaround in Cloudera DataFlow for Data Hub 7.2.6.

Known Issues in Flow Management

Learn about the known issues in Flow Management, the impact or changes to the functionality, and the workaround.

Technical Service Bulletins

TSB 2022-580: NiFi Processors cannot write to content repository

If the content repository disk is filled more than 50% (or any other value that is set in nifi.properties for nifi.content.repository.archive.max.usage.percentage), and if there is no data in the content repository archive, the following warning message can be found in the logs: "Unable to write flowfile content to content repository container default due to archive file size constraints; waiting for archive cleanup". This would block the processors and no more data is processed.

This appears to only happen if there is already data in the content repository on startup that needs to be archived, or if the following message is logged: “Found unknown file XYZ in the File System Repository; archiving file”.

Upstream JIRA

- NIFI-10023
- NIFI-9993

Knowledge article

For the latest update on this issue see the corresponding Knowledge article: TSB 2022-580: NiFi Processors cannot write to content repository

Known Issues in Streams Messaging

Learn about the known issues Streams Messaging, the impact or changes to the functionality, and the workaround.

Kafka

Learn about the known issues and limitations in Kafka in this release:

Known Issues

Topics created with the kafka-topics tool are only accessible by the user who created them when the deprecated --zookeeper option is used

By default all created topics are secured. However, when topic creation and deletion is done with the kafka-topics tool using the --zookeeper option, the tool talks directly to Zookeeper. Because

security is the responsibility of ZooKeeper authorization and authentication, Kafka cannot prevent users from making ZooKeeper changes. As a result, if the --zookeeper option is used, only the user who created the topic will be able to carry out administrative actions on it. In this scenario Kafka will not have permissions to perform tasks on topics created this way.

Use kafka-topics with the --bootstrap-server option that does not require direct access to Zookeeper.

Certain Kafka command line tools require direct access to Zookeeper

The following command line tools talk directly to ZooKeeper and therefore are not secured via Kafka:

- kafka-reassign-partitions

None

The offsets.topic.replication.factor property must be less than or equal to the number of live brokers

The offsets.topic.replication.factor broker configuration is now enforced upon auto topic creation. Internal auto topic creation will fail with a GROUP_COORDINATOR_NOT_AVAILABLE error until the cluster size meets this replication factor requirement.

None

Requests fail when sending to a nonexistent topic with auto.create.topics.enable set to true

The first few produce requests fail when sending to a nonexistent topic with auto.create.topics.enable set to true.

Increase the number of retries in the producer configuration setting retries.

Custom Kerberos principal names cannot be used for kerberized ZooKeeper and Kafka instances

When using ZooKeeper authentication and a custom Kerberos principal, Kerberos-enabled Kafka does not start. You must disable ZooKeeper authentication for Kafka or use the default Kerberos principals for ZooKeeper and Kafka.

None

KAFKA-2561: Performance degradation when SSL Is enabled

In some configuration scenarios, significant performance degradation can occur when SSL is enabled. The impact varies depending on your CPU, JVM version, Kafka configuration, and message size. Consumers are typically more affected than producers.

Configure brokers and clients with ssl.secure.random.implementation = SHA1PRNG. It often reduces this degradation drastically, but its effect is CPU and JVM dependent.

OPSAPS-43236: Kafka garbage collection logs are written to the process directory

By default Kafka garbage collection logs are written to the agent process directory. Changing the default path for these log files is currently unsupported.

None

Limitations

Collection of Partition Level Metrics May Cause Cloudera Manager’s Performance to Degrade

If the Kafka service operates with a large number of partitions, collection of partition level metrics may cause Cloudera Manager's performance to degrade.

If you are observing performance degradation and your cluster is operating with a high number of partitions, you can choose to disable the collection of partition level metrics.

Important: If you are using SMM to monitor Kafka or Cruise Control for rebalancing Kafka partitions, be aware that both SMM and Cruise Control rely on partition level metrics. If partition level metric collection is disabled, SMM will not be able to display information about partitions. In addition, Cruise Control will not operate properly.

Complete the following steps to turn off the collection of partition level metrics:

1. Obtain the Kafka service name:
   a. In Cloudera Manager, Select the Kafka service.
   b. Select any available chart, and select Open in Chart Builder from the configuration icon drop-down.
   c. Find $SERVICENAME= near the top of the display.

      The Kafka service name is the value of $SERVICENAME.
2. Turn off the collection of partition level metrics:
   a. Go to HostsHosts Configuration.
   b. Find and configure the Cloudera Manager Agent Monitoring Advanced Configuration Snippet (Safety Valve) configuration property.

      Enter the following to turn off the collection of partition level metrics:

      ```
      [KAFKA_SERVICE_NAME]_feature_send_broker_topic_partition_ent
      ity_update_enabled=false
      ```

      Replace [KAFKA_SERVICE_NAME] with the service name of Kafka obtained in step 1. The service name should always be in lower case.
   c. Click Save Changes.

Schema Registry

There are no known issues in Schema Registry in this release.

Streams Messaging Manager

Learn about the known issues in Streams Messaging Manager in this release:

CDPD-19495: SMM UI does not show producer data on topics page

In the SMM UI, the topics page, the topic profile pages, and the broker profile pages consistently show 0 for producer messages.

Workaround: For the real producer metrics, check the aggregated REST API responses. The real producer metrics are within the producerIdToOutMessagesCount field.

Streams Replication Manager

Learn about the known issues and limitations in Streams Replication Manager in this release:

Known Issues

MM2-163: SRM does not sync re-created source topics until the offsets have caught up with target topic

Messages written to topics that were deleted and re-created are not replicated until the source topic reaches the same offset as the target topic. For example, if at the time of deletion and re-creation there are a 100 messages on the source and target clusters, new messages will only get replicated once the re-created source topic has 100 messages. This leads to messages being lost.

None

CDPD-14019: SRM may automatically re-create deleted topics

If auto.create.topics.enable is enabled, deleted topics are automatically recreated on source clusters.

Prior to deletion, remove the topic from the topic whitelist with the srm-control tool. This prevents topics from being re-created.

```
srm-control topics --source [SOURCE_CLUSTER] --target [TARGET_CL
USTER] --remove [TOPIC1][TOPIC2]
```

CDPD-13864 and CDPD-15327: Replication stops after the network configuration of a source or target cluster is changed

If the network configuration of a cluster which is taking part in a replication flow is changed, for example, port numbers are changed as a result of enabling or disabling TLS, SRM will not update its internal configuration even if SRM is reconfigured and restarted. From SRM's perspective, it is the cluster identity that has changed. SRM cannot determine whether the new identity corresponds to the same cluster or not, only the owner or administrator of that cluster can know. In this case, SRM tries to use the last known configuration of that cluster which might not be valid, resulting in the halt of replication.

The internal topic storing the configuration of SRM can be deleted. After a restart SRM will re-create and re-populate it with the configuration data loaded from its property file. The topic is hosted on the target cluster of the replication flow. The topic name is: mm2-configs.[*SOURCE_ALIAS*].internal. However, changing a replicated cluster's identity is generally not recommended.

CDPD-22094: The SRM service role displays as healthy, but no metrics are processed

The SRM service role might encounter errors that make metrics processing impossible. An example of this is when the target Kafka cluster is not reachable. The SRM service role does not automatically stop or recover if such an error is encountered. It continues to run and displays as healthy in Cloudera Manager. Metrics, however, are not processed. In addition, no new data is displayed in SMM for the replications.

1. Ensure that all clusters are available and are in a healthy state.
2. Restart SRM.

CDPD-22389: The SRM driver role displays as healthy, but replication fails

During startup, the SRM driver role might encounter errors that make data replication impossible. An example of this is when one of the clusters added for replication is not reachable. The SRM driver role does not automatically stop or recover if such an error is encountered. It will start up, continue to run, and display as healthy in Cloudera Manager. Replication, however, will not happen.

1. Ensure that all clusters are available and are in a healthy state.
2. Restart SRM.

CDPD-23683: The replication status reported by the SRM service role for healthy replications is flaky

The replication status reported by the SRM service role is flaky. The replication status might change between active and inactive frequently even if the replication is healthy. This status is also reflected in SMM on the replications tab.

None

Limitations

SRM cannot replicate Ranger authorization policies to or from Kafka clusters

Due to a limitation in the Kafka-Ranger plugin, SRM cannot replicate Ranger policies to or from clusters that are configured to use Ranger for authorization. If you are using SRM to replicate data to or from a cluster that uses Ranger, disable authorization policy synchronization in SRM. This can be achieved by clearing the Sync Topic Acls Enabled (sync.topic.acls.enabled) checkbox.

SRM cannot ensure the exactly-once semantics of transactional source topics

SRM data replication uses at-least-once guarantees, and as a result cannot ensure the exactly-once semantics (EOS) of transactional topics in the backup/target cluster.

Note: Even though EOS is not guaranteed, you can still replicate the data of a transactional source, but you must set isolation.level to read_committed for SRM's internal consumers. This can be done by adding *[***SOURCE CLUSTER ALIAS***]->[***TARGET CLUSTER ALIAS***]*.consumer.isolation.level=read_committed to the Streams Replication Manager's Replication Configs SRM service property in Cloudera Manger.

SRM checkpointing is not supported for transactional source topics

SRM does not correctly translate checkpoints (committed consumer group offsets) for transactional topics. Checkpointing assumes that the offset mapping function is always increasing, but with transactional source topics this is violated. Transactional topics have control messages in them, which take up an offset in the log, but they are never returned on the consumer API. This causes the mappings to decrease, causing issues in the checkpointing feature. As a result of this limitation, consumer failover operations for transactional topics is not possible.

Known Issues in Streaming Analytics

There are no known issues for Streaming Analytics in Cloudera DataFlow for Data Hub 7.2.6.

Fixed Issues in Cloudera DataFlow for Data Hub 7.2.6

Fixed issues represent selected issues that were previously logged through Cloudera Support, but are addressed in the current release. These issues may have been reported in previous versions within the Known Issues section; meaning they were reported by customers or identified by Cloudera Quality Engineering team.

Review the list of issues that are resolved in Cloudera DataFlow for Data Hub 7.2.6.

Fixed Issues in Flow Management

Review the list of Flow Management issues that are resolved in Cloudera DataFlow for Data Hub 7.2.6.

NIFI-7422 and NIFI-7452 - Improvements for Atlas lineage with AWS S3 and Azure ADLS

Support for aws_s3_pseudo_dir and adls_gen2_directory in Atlas reporting task.

NIFI-7436 - Performance improvements on analytics predictions

For a given FieldValue, you can obtain a String of a logical path for that node to the root.

NIFI-7683 and NIFI-7994 - ReplaceText issue

ReplaceText, when scheduled to run with multiple Concurrent Tasks, and using a Replacement Strategy of "Regular Expression" or "Literal Replace" no longer results in content being corrupted due to sharing a single buffer (byte[]) between threads.

NIFI-7707 - Google Cloud PubSub issue

When you generate an avro event from a record and publish it using the PublishPubSub processor, the content-type of the file no longer changes from avro/binary to UTF-8.

NIFI-7968 and NIFI-7954 - Kerberos relogin issues

Support for:

- PutHDFS and PutParquet processors can use a KeytabCredentialsService.
- Kerberos ticket renewal.

Fixed Issues in Streams Messaging

Review the list of Streams Messaging issues that are resolved in Cloudera DataFlow for Data Hub 7.2.6.

Kafka

Learn about the fixed issues in Kafka in this release:

CDPD-16683: Support consumer offset sync across clusters in MM 2.0 (backport KAFKA-9076)

This is a backported improvement, see KIP-545 and KAFKA-9076 for more information.

CDPD-17921: advertised.listeners should allow duplicated ports (backport KAFKA-10478)

The advertised.listeners Kafka property now accepts duplicated ports.

OPSAPS-57907: The Kafka metric collector adapter generates high CPU load

This issue is now resolved.

OPSAPS-58319: Kafka metrics may have been wiped after a Kafka cluster restart

Empty responses are now properly handled when topic names are fetched. Kafka metrics will no longer be wiped.

Schema Registry

Learn about the fixed issues in Schema Registry in this release:

OPSAPS-58397: Make the Schema Registry hashing algorithm configurable

Added new option to Schema Registry configuration where the users can change the hashing algorithm used to generate schema fingerprints. The default value is MD5.

OPSAPS-58157: Schema Registry Swagger page does not work due to CSP violation

Schema Registry's Swagger page now correctly renders and the browser does not report a Content Security Policy violation error.

OPSAPS-58153: Schema Registry role log is not visible through CM UI

In versions before Cloudera Manager 7.2.3, Schema Registry logs were not displayed in the Cloudera Manager UI. Now, the Schema Registry log format was changed to make it consistent with the log format of other CDP components. Schema Registry Server role logs are now correctly displayed in Cloudera Manager.

CDPD-18345: Schema Registry fails to start on a cluster with TLS enabled and multiple SANs in the certificate

Schema Registry can now start on a cluster when TLS is enabled and there are multiple Subject Alternative Names in the certificate.

CDPD-17969: Cannot fork schema

Previously, schema versions could not be forked when Ranger authorization was enabled. This fixes the bug and allows creating branches.

CDPD-17849: Remove usage of non-FIPS compatible algorithms in Schema Registry

Added configuration option in both client and server to change the hashing algorithm used for generating schema fingerprints. The default value is MD5 but can be changed to SHA-2 or other algorithms.

CDPD-16851: updateSchemaMetadata() allows changing schema type to an invalid value which breaks adding new versions to a schema

When updating a SchemaMetadata, type parameter is now validated.

CDPD-16812: SR - read boolean properties from kafka configs

Schema Registry can read and cast String values as Boolean values.

CDPD-11076: Schema Registry API swagger doc is incorrect/not up to date

Schema Registry API swagger documentation has been updated, descriptions completed to have better understanding about endpoints and parameters have been also completed and this way more accurate searches can be implemented.

Streams Messaging Manager

Learn about the fixed issues in Streams Messaging Manager in this release:

CDPD-16215: SMM showing inaccurate Producer Messages Count in multiple places

If a producer was inactive for a few minutes it would be emptied from the Kafka broker cache. In that case, the producer's messages count entity would start from 0 in ServiceMonitor's database and show incorrect values for the "Messages" fields where that producer is shown.

CDPD-16438: SMM does not handle sum() metrics correctly

Single Point metrics (metrics that are a single timeStamp - Value pair such as sums, avgs etc) were showing data that might have been related to another timeSpan. So for instance when querying for 30 minutes that 6-hourly data was shown. The problematic queried timespans and the corresponding shown timespans were the following:

- 6 hours -> 6 hours, 1 hour, 30 mins
- 2 days -> 24 hours, 2 days

CDPD-17889: SMM UI uses the "latestOutMessagesCount" field from the response for showing the number of messages at the ProducerDetail page

If a producer is inactive for a few minutes, and gets evicted from the Kafka broker cache, the metrics will restart from 0 in ServiceMontior's database and will show inaccurate data at the messages field at the ProducerDetail page.

OPSAPS-58488: SMM is missing from Ranger users for SchemaRegistry

SMM cannot complete Schema Registry related API operations because it is not whitelisted for access in Ranger.

Streams Replication Manager

Learn about the fixed issues in Streams Replication Manager in this release:

CSP-956: Topics or groups added to white or blacklists are not returned when using srm-control --list

The srm-control tool and the SRM driver are now able to read the full white and blacklist from the srm-control.<alias>.internal configuration topic.

CSP-462: Replication failing when SRM driver is present on multiple nodes

Replication no longer fails when the driver is present on multiple nodes. Running SRM in high availability mode is now possible.

CDPD-18300: SRM resolves configuration provider references in its internal configuration topic

Configuration provider references are no longer resolved. Sensitive information is no longer exposed this way.

Fixed Issues in Streaming Analytics

There are no fixed issues for Streaming Analytics in Cloudera DataFlow for Data Hub 7.2.6.