

Discussion thread	https://lists.apache.org/thread/oydhcfkco2kqp4hdd1glzy5vkw131rkz
Vote thread	-
JIRA	-
Release	2.0.0-1.17, 2.0.0-1.18

Motivation

As Apache Bahir was moved into the Attic recently, this proposal is about to keep the Kudu connector alive as an external Flink connector to keep it maintainable.

[1] <https://bahir.apache.org/>

Public Interfaces

- Source:
 - [FLIP-27: Refactor Source Interface](#)
- Sink:
 - [FLIP-143: Unified Sink API](#)
- Table:
 - [FLIP-95: New TableSource and TableSink interfaces](#)

Supported Catalog Config Options

Key	Type	Desc	Default	Required
kudu.masters	string	kudu's master server address		Y
property-version	int	Version of the overall property design. This option is meant for future backwards compatibility.	1	

Supported Table Config Options

Key	Type	Desc	Default	Required
kudu.masters	string	kudu's master server address		Y
kudu.table	string	kudu's table name		Y
kudu.primary-key-columns	string	kudu's primary key, primary key must be ordered		
kudu.hash-columns	string	kudu's hash columns		
kudu.replicas	int	kudu's replica nums	3	
kudu.hash-partition-nums	int	kudu's hash partition bucket nums, defaultValue is 2 * replica nums	replica_num * 2	
kudu.max-buffer-size	int	kudu's max buffer size		
kudu.flush-interval	int	kudu's data flush interval		
kudu.operation-timeout	long	kudu's operation timeout		
kudu.ignore-not-found	bool	if true, ignore all not found rows		
kudu.ignore-duplicate	bool	if true, ignore all duplicate rows		
kudu.scan.row-size	int	kudu's scan row size	0	
kudu.lookup.cache.max-rows	long	the max number of rows of lookup cache, over this value, the oldest rows will be eliminated. "cache.max-rows" and "cache.ttl" options must all be specified if any of them is specified. Cache is not enabled as default.	-1	
kudu.lookup.cache.ttl	long	the cache time to live	-1	
kudu.lookup.max-retries	int	the max retry times if lookup database failed	3	

Supported KuduSink Config Options

Heavily overlaps with the Table config options. Only the Kudu "masters" are required, and there are some other optional configs that have default values [1].

[1] [KuduWriterConfig](#)

Proposed Changes

A new flink-connector-kudu repository should be created by the Flink community, similar to ElasticSearch [1]. Migrating the current code keeping the history and noting it explicitly it was forked from the Bahir repository [2].

In its current form, Kudu connector provides a DataStream Source and Sink that is based on the old API, and a Table and Catalog implementation that is based on the new TableSource/TableSink interfaces.

As part of the externalization process, the DataStream connector implementation can be updated to the new Source/Sink APIs.

[1] <https://github.com/apache/flink-connector-elasticsearch>

[2] <https://github.com/apache/bahir-flink>

Compatibility, Deprecation, and Migration Plan

The latest Bahir release was for Flink 1.14 [1], which is not supported anymore, although the code currently contains Flink 1.17 [2].

To avoid any kind of confusion and mark a clean separation from the Bahir connector, the externalized Kudu connector major version should be incremented. The externalized connector should support the two latest available Flink and supported Flink versions.

Any change that is required for adapting the code to the current Flink versions has to be incorporated as part of the externalization process.

[1] <https://github.com/apache/bahir-flink/releases/tag/v1.1.0>

[2] <https://github.com/apache/bahir-flink/blob/4a8f60f03f7caeb5421c7c88fa34a3ba419b61ee/pom.xml#L116>

Test Plan

For now, keeping the already existing tests.

Rejected Alternatives

No rejected alternatives.