

RFC - 20 : handle failed records

RFC - 20 : handle failed records

- [RFC - 20 : handle failed records](#)

Proposers

Approvers

Status

Abstract

Background

Implementation

- Error record
- Errors table
 - Local error table
 - Global error table
 - Configurations
- Write path
- CLI support
- Metrics

Rollout/Adoption Plan

Test Plan

Proposers

- [Raymond Xu](#)

Approvers

- [Vinoth Chandar](#) : [APPROVED/REQUESTED_INFO/REJECTED]
- ...

Status

Current state:

	Current State
UNDER DISCUSSION	
IN PROGRESS	✓
ABANDONED	
COMPLETED	
INACTIVE	

Discussion thread: [here](#)

JIRA: [HUDI-648](#) - Getting issue details... STATUS

Released: 0.6.x

Abstract

Present a proposal for handling failed records in writer path.

Background

To handle failed records properly to facilitate investigation.

Implementation

Error record

Define an avro schema for error records

error record schema

```
{
  "type": "record",
  "namespace": "org.apache.hudi.common",
  "name": "ErrorRecord",
  "fields": [
    {
      "name": "uid",
      "type": "string"
    },
    {
      "name": "ts",
      "type": "string"
    },
    {
      "name": "schema",
      "type": ["null", "string"],
      "default": null
    },
    {
      "name": "record",
      "type": ["null", "string"],
      "default": null
    },
    {
      "name": "message",
      "type": ["null", "string"],
      "default": null
    },
    {
      "name": "context",
      "type": ["null", {"type": "map", "values": "string"}],
      "default": null
    }
  ]
}
```

- `uid`: uuid for the error record
- `ts`: creation unix timestamp for the error record
- `schema`: original schema for the record if any
- `record`: original serialized record in json if any
- `message`: additional message or any string like error stacktrace to be attached
- `context`: kv pairs for any related context info like commitTime, tableName, partitionpath, recordKey, etc

Errors table

Users can configure, based on their preferences, error tables as local or global ones.

Local error table

By default, if error table is enabled, it will be a local error table. Failed records will be written to a local Hudi table alongside with the original Hudi table with a suffix (configurable) like `\_errors`.

Global error table

To write to a global error table, users can configure `hoodie.write.error.table.base.path=<some file system path>` and `hoodie.write.error.table.name=foobar`. If either of these 2 configs were set, error table is set to global mode and `hoodie.write.error.table.suffix` will be omitted.

Configurations

key		default
hoodie.write.error.table.enabled	set to true to activate error table handling feature	false
hoodie.write.error.table.suffix	suffix for local error table name, stored alongside the target table. If the Hudi table is "foo", errored records will be saved to "foo_errors" at the same base dir as configured via `hoodie.base.path`	"_errors"
hoodie.write.error.table.name	error table name	"hoodie_errors"
hoodie.write.error.table.base.path	base path for global error table	same as `hoodie.base.path`

Write path

Start with

- org.apache.hudi.client.HoodieWriteClient#postWrite
- org.apache.hudi.client.HoodieWriteClient#completeCompaction

CLI support

- Consider adding CLI support for easy inspection

Metrics

- Emit a count metric for the number of failed records

Rollout/Adoption Plan

- Use configuration turn on this feature `hoodie.write.error.table.enabled=true`
- Default to false for smooth roll-out

Test Plan

- Functional test cases to cover both local and global cases.