Compaction pooling

Concept:

Compaction requests and workers can be assigned to pools. A worker assigned to a specific pool will only process compaction requests in that pool. Workers and compaction requests without pool assignment are implicitly belong to the *default pool.* The pooling concept allows fine tuning of processing compaction requests. For example it is possible to create a pool name 'high priority compaction', assign some frequently modified tables to it, and dedicate a set of workers to this pool. As a result, the compaction requests for these tables will be immediately picked up by the dedicated workers, even if there are several other compaction requests (enqueued earlier) in the default queue.

Pool assignment

Compaction requests can be assigned to pools in three different ways.

Automatic pool assignment

Databases, tables and partitions can be assigned to compaction pools through the

```
hive.compactor.worker.pool={pool_name}
```

Database/Table property. If the property is set on Database level, it applies to all tables and partitions. The pool also can be assigned on a table/partition level, in this case it overrides the Database level value (if set).
If any of the above is set, it is used by the *Initiator* during the creation of the compaction requests.

Manual pool assignment

The compaction request also can be assigned to a pool by using the ALTER TABLE COMPACT command (E.g. manual compaction). If provided, this value overrides the *hive.compactor.worker.pool* value on any level.

```
ALTER TABLE COMPACT table_name POOL 'pool_name';
```

Implicit pool assignment

Tables, partitions and manual compaction requests without specified pool name are implicitly assigned to the default pool.

Pool timeout

If a compaction request is not processed by any of the labeled pools within a predefined period, it falls back to the default pool. The timeout can be set through the

```
hive.compactor.worker.pool.timeout
```

configuration property. This approach is to cover the following use cases:

- The request is accidentally assigned to a non-existent pool. (E.g.: Typo in the pool name when issuing an ALTER TABLE COMPACT command.
- Typo in the DB or table property used by the initiator to create compaction requests.
- A HS2 instance is stopped due to a scaledown or schedule, and its pending compaction requests still should be processed.

The timeout can be disabled by setting the configuration property to 0.

Labeled worker pools

The labeled worker pools can be defined through the

```
hive.compactor.worker.{poolname}.threads={thread_count}
```

configuration setting. Please note that in this case the configuration key is also dynamic.

Default pool

The default pool is responsible for processing the non-labeled and timed-out compaction requests. On a cluster-wide level, at least 1 worker thread on at least one node should be assigned to the default pool, otherwise compaction requests may never be processed.

Worker allocation

The already existing *hive.compactor.worker.threads* configuration value holds the maximum number of worker threads. The worker allocation happens as follows:

- Labeled pools are initialized in a sequential manner with random ordering.
- Each pool decreases the number of available workers by its own worker count.
- If the number of assignable workers is less than the configured one, the pool size will be adjusted (In other words: if the requested pool size is 5 but there are only 3 workers remaining, than the pool size will be decreased to 3).
- If the number of assignable workers is 0, the pool won’t be initialized.
- All remaining workers not used up by the labeled pools, are assigned to the default pool.

The worker allocation can be configured per HS2 instance.