

Kafka 0.9 Unit and Integration Tests

- Unit Test
 - kafka.common:
 - kafka.clients.producer
 - kafka.server
- Integration Test
 - admin
 - api
 - config
 - replication (with broker hard/soft failures and restart, network partition scenarios)
 - compression *
 - logging
 - util (?)

Pre-0.8 the integration tests span over core/src/test, some of them resides in the unit.kafka, some others in other.kafka. In 0.9 we want to have a cleaner coverage of integration tests along with the unit tests.

Unit Test

For each low-level component of kafka server and client that is self-functional, such as bufferpool/sender of producer, memoryrecords of common, replica/partition-manager of server, we will have a corresponding unit test class in the test/unit/<package-name>/<component-name>.

kafka.common:

- config: config properties can be read/write correctly, default value can be retrieved, config errors can be detected.
- protocol: schema fields can be put/get by pos/name correctly, schema/arrays can be serialized/de-ser correctly.
- record:
 - record fields can be read/write correctly, record functions (size, isValid, checksum) works correctly.
 - memory-records functions (append with/wo compression, etc) works correctly.
- requests: request serialization/de-ser correctly, and non-null fields are defined.
- metadata:
 - cluster/topic-partition/partition functions works correctly.
 - serializer works properly.
- network:
 - networksend/receive functions work correctly.
 - bytbuffer/send/receive functions work correctly.
 - selector functions work correctly (may need some MockChannels).

kafka.clients.producer

- partitioner: logical correctness.
- recordsend: functions work correctly (may need MockCallback).
- bufferpool: logical correct under multi-thread, failure cases.
- metadata: logical correct under multi-thread, failure cases.
- recordaccumulator: functional correct for read/drain, etc; logical correct for append under multi-thread.
- recordbatch: succeed/fail cases of functions (tryAppend, etc)
- sender: processReadyPartitions function, handleXX functions, and close function correctness.

kafka.server

- log: log recover and cleaning logical correctness
- MORE...

Integration Test

For integration tests we will categorize the following modules, and for each module we will do integration tests for different scenarios (normal success cases, failure cases with different configs, etc).

" * " indicate tests that we can start in parallel with producer code stabilization.

admin

- Create topic
- Adding partitions to topic
- Alter topic configs (retention, etc)
- Alter topic replication factor
- Partition reassignment
- Preferred leader election

api

Producer: *

- Blocking and non-blocking send behavior, with different message size (including too-large message size)
- Send behavior with null key or null value
- Send with specific partition key(s)
- New topic auto-creation

Consumer:

- Commit and fetch offset behavior
- Poll behavior with/wo timeout, different fetch size
- Rebalance behavior
- Topic change capture behavior

config

- Invalid values *

replication (with broker hard/soft failures and restart, network partition scenarios)

- Replica consistency and ISR correctness
- Controller failover behavior
- Coordinator failover behavior
- No message-loss with ack > 1 *

compression *

- compress/decompress behavior with different batch size
- decompress/re-compress in mirror-maker (?)

logging

- String encoder
- Appender property

util (?)