

# KIP-143: Controller Health Metrics

The contents of this KIP were authored by [Jun Rao](#).

- [Status](#)
- [Motivation](#)
- [Public Interfaces](#)
  - [Controller Metrics](#)
  - [Partition Metrics](#)
- [Proposed Changes](#)
- [Compatibility, Deprecation, and Migration Plan](#)
- [Rejected Alternatives](#)
- [Future work](#)

## Status

**Current state:** *Adopted*

**Discussion thread:** [here](#)

**JIRA:**

 Unable to render Jira issues macro, execution error.

Please keep the discussion on the mailing list rather than commenting on the wiki (wiki discussions get unwieldy fast).

## Motivation

Ensuring that the Kafka Controller is healthy is an important part of monitoring the health of a Kafka Cluster. However, the metrics currently exposed are not sufficient for reliably detecting issues like slow progress or deadlocks. We propose a few new metrics that will solve this issue. Even though KAFKA-5028 will potentially fix existing deadlocks, there will still be known (and potentially unknown) issues that can cause slow or no progress so these metrics will still be useful.

## Public Interfaces

All of the following will be added via the Yammer metrics library like most of the broker metrics. Retrieving a metric value will not acquire any Controller locks (which was an issue in the past).

### Controller Metrics

(1) kafka.controller:type=KafkaController,name=ControllerState

type: gauge

value: the state the controller is in, i.e. the event that is currently being processed. Some actions like partition reassignment may take a while and include many events (potentially interleaved with other events), but that doesn't change the fact that at most one event is processed at a time.

Valid states (events comprising that state in brackets):

- 0 - idle
- 1 - controller change (Startup, ControllerChange, Reelect)
- 2 - broker change (BrokerChange)
- 3 - topic creation/change (TopicChange, PartitionModifications)
- 4 - topic deletion (TopicDeletion, TopicDeletionStopReplicaResult)
- 5 - partition reassignment (PartitionReassignment, PartitionReassignmentIsrChange)
- 6 - auto leader balance (AutoPreferredReplicaLeaderElection)
- 7 - manual leader balance (PreferredReplicaLeaderElection)
- 8 - controlled shutdown (ControlledShutdown)
- 9 - isr change (IsrChangeNotification)

For each state, there's a timer with the rate and time with 2 exceptions: BrokerChange (currently tracked as LeaderElectionRateAndTimeMs) and ControlledShutdown (tracked via RequestQueueTimeMs for the the ControlledShutdown request).

(1). kafka.controller:type=ControllerStats,name=ControllerChangeRateAndTimeMs  
type: timer  
value: rate and latency for the controller change state

(2). kafka.controller:type=ControllerStats,name=TopicChangeRateAndTimeMs

type: timer  
value: rate and latency for the controller to create new topics

(3). kafka.controller:type=ControllerStats,name=TopicDeletionRateAndTimeMs  
type: timer  
value: rate and latency for the controller to delete topics

(4). kafka.controller:type=ControllerStats,name=PartitionReassignmentRateAndTimeMs  
type: timer  
value: rate and latency for the controller to reassign partitions

(5). kafka.controller:type=ControllerStats,name=AutoLeaderBalanceRateAndTimeMs  
type: timer  
value: rate and latency for the controller to auto balance the leaders

(6). kafka.controller:type=ControllerStats,name=ManualLeaderBalanceRateAndTimeMs  
type: timer  
value: rate and latency for the controller to manually balance the leaders

(7) kafka.controller:type=ControllerStats,name=IsrChangeRateAndTimeMs

type: timer  
value: rate and latency for the controller to manually balance the leaders

## ControllerChannelManager Metrics

We also want to know the size of the queue in ControllerChannelManager:

(9) kafka.controller:type=ControllerChannelManager,name=TotalQueueSize

type: gauge

(10) kafka.controller:type=ControllerChannelManager,name=QueueSize,brokerId=10

type: gauge

## Partition Metrics

Since quite a few jiras reported continuous errors due to "'Cached zkVersion 54 not equal to that in zookeeper, skip updating ISR". It would be useful to measure the occurrences of failed ISR update in ZK.

(13) kafka.cluster:type=Partition,name=FailedIsrUpdatesPerSec

type: meter

## Proposed Changes

We will add the relevant metric type to one of KafkaController, ControllerStats, ControllerChannelManager or Partition as specified in the Public Interfaces section.

## Compatibility, Deprecation, and Migration Plan

We are introducing new metrics so there is no compatibility impact.

## Rejected Alternatives

1. Don't add these metrics: it's currently difficult to detect these issues, they impact cluster health and the overhead of the proposed metrics is low.
2. Use Kafka metrics instead of Yammer metrics: most of the broker metrics use Yammer Metrics so it makes sense to stick with that until we have a plan on how to migrate them all to Kafka Metrics.

## Future work

1. KAFKA-5028 introduced a queue for Controller events. It would be useful to have a gauge for the queue size and a histogram for how long an event waits in the queue before being processed. However, we are in the process of making additional changes to improve the handling of soft failures and there's a possibility that the controller queue could be replaced by a broker queue for all ZK communication. We will see how that develops before deciding which metrics should be exposed. In the meantime, the ControllerState and other metrics should provide enough information to issue an alert if the Controller is not healthy.