

KIP-6 - New reassignment partition logic for rebalancing

- [Status](#)
- [Motivation](#)
- [Public Interfaces](#)
- [Proposed Changes](#)
- [Rebalance command](#)
- [Decommission broker command](#)
- [Compatibility, Deprecation, and Migration Plan](#)
- [Rejected Alternatives](#)

The existing re-assign tool requires a lot of manual intervention. The idea is to have fairly balanced consistent result that we can use for partition reassignment.

Status

Current state: *Under Discussion*

Discussion thread: [here](#)

JIRA: [here](#)

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

Motivation

Operational

Public Interfaces

Briefly list any new interfaces that will be introduced as part of this proposal or any existing interfaces that will be removed or changed. The purpose of this section is to concisely call out the public contract that will come along with this feature.

A public interface is any change to the following:

- *Command line tools and arguments*

Proposed Changes

Current implementation produces fair replica distribution between specified list of brokers. Unfortunately, it doesn't take into account current replica assignment.

So if we have, for instance, 3 brokers id=[0..2] and are going to add fourth broker id=3, generate will create an assignment config which will redistribute replicas fairly across brokers [0..3] in the same way as those partitions were created from scratch. It will not take into consideration current replica assignment and accordingly will not try to minimize number of replica moves between brokers.

This should be improved. New output of improved rebalance ("--rebalance" command) algorithm should suite following requirements:

- fairness of replica distribution - every broker will have R or R+1 replicas assigned;
- minimum of reassignments - number of replica moves between brokers will be minimal;

Example.

Consider following replica distribution per brokers [0..3] (we just added brokers 2 and 3):

- broker - 0, 1, 2, 3
- replicas - 7, 6, 0, 0

The new algorithm will produce following assignment:

- broker - 0, 1, 2, 3
- replicas - 4, 3, 3, 3
- moves - -3, -3, +3, +3

It will be fair and number of moves will be 6, which is minimal for specified initial distribution.

The scope of this issue is:

- design an algorithm matching the above requirements;
- implement this algorithm and unit tests;
- test it manually using different initial assignments;

Rebalance command

New command "--rebalance" is proposed to be added.

The usage scenario is same as for "--generate" except new command is called "--rebalance".

Scenario:

1. User generates reassignment configuration by running:
`--rebalance --topics-to-move-json-file topics.json --broker-list 0,1,2 --zookeeper zk:2181`
2. User copies proposed reassignment configuration to reassignment.json file
3. User executes reassignment by running:
`--execute --reassignment-json-file reassignment.json --zookeeper zk:2181`
4. User verifies status of reassignment by running:
`--verify --reassignment-json-file reassignment.json --zookeeper zk:2181`

Decommission broker command

Beside updating implementation of redistribution algorithm, new command is proposed to be added.

The goal of this command is to provide a shortcut for decommissioning (removing) the broker.

Following syntax should be supported:

```
--decommission-broker <broker-id>
```

The implementation should reassign all replicas owned by specified broker to the rest of the brokers using the above modified algorithm. Reassignment should be applied automatically without need to run `--execute` command.

Compatibility, Deprecation, and Migration Plan

To preserve compatibility old "--generate" command (using old algorithm) is still present. New "--rebalance" command (invoking new rebalance algorithm) is added.

- *What impact (if any) will there be on existing users?*
No impact. Old "--generate" command is preserved for compatibility.
- *If we are changing behavior how will we phase out the older behavior?*
- *If we need special migration tools, describe them here.*
- *When will we remove the existing behavior?*
Old "--generate" command will be removed in future releases, when decided it is no longer needed.

Rejected Alternatives

If there are alternative ways of accomplishing the same thing, what were they? The purpose of this section is to motivate why the design is the way it is and not some other way.