KIP-728: Introduce the RESTORING state to the KafkaStreams state machine

Status

Current state: Under Discussion

Discussion thread: *here TODO*

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

Describe the problems you are trying to solve.

Public Interfaces

We will be adding a new state to the KafkaStreams.State:

```
public enum State {
    CREATED(1, 4),                         // 0
    REBALANCING(2, 4, 6),           // 1
    RESTORING(1, 3, 4, 6),            // 2       <--- new state
    RUNNING(1, 3, 4, 6),               // 3
    PENDING_SHUTDOWN(5),    // 4
    NOT_RUNNING,                      // 5
    PENDING_ERROR(7),             // 6
    ERROR;                                   // 7}
```

The allowed transitions to/from this new state are:

1. {REBALANCING, RUNNING} RESTORING
2. RESTORING {REBALANCING, RUNNING, PENDING_SHUTDOWN, PENDING_ERROR}

Thus, this new state effectively sits in between REBALANCING and RUNNING: Streams will always pass through RESTORING following a rebalance, and remain in that state until all state stores have been fully restored from their changelog.

Proposed Changes

Describe the new thing you want to do in appropriate detail. This may be fairly extensive and have large subsections of its own. Or it may be a few sentences. Use judgement based on the scope of the change.

Compatibility, Deprecation, and Migration Plan

- *What impact (if any) will there be on existing users?*
- *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?*

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.