

# KIP-442: Return to default max poll interval in Streams

- [Status](#)
- [Motivation](#)
- [Proposed Changes](#)
- [Compatibility, Deprecation, and Migration Plan](#)
- [Rejected Alternatives](#)

## Status

**Current state:** *Accepted*

**Discussion thread:** <https://www.mail-archive.com/dev@kafka.apache.org/msg96520.html>

**JIRA:** [KAFKA-6399](#) - Getting issue details...

**PR:** <https://github.com/apache/kafka/pull/6509>

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

## Motivation

Streams previously used an "infinite" default `max.poll.interval.ms` `Consumer` config. The reasoning was that we didn't call `poll()` during restore, which can take arbitrarily long, so our maximum expected interval between poll calls was infinite. Since 1.0, we do call `poll` during restore, so we no longer need the infinite default, and setting a reasonable limit here can help to resolve situations in which a particular thread gets stuck for a while and Streams stops making progress.

## Proposed Changes

We want to remove the override and instead fall back to the `ConsumerConfig`-defined default of **five minutes**.

## Compatibility, Deprecation, and Migration Plan

The only problem I foresee is that existing applications may currently take longer than five minutes between calls to poll in the steady state. Think: low-volume, but high-latency computations. These applications are leaning on the current Streams-defined default of "max int" millis. Upon updating Streams, they would start to see timeouts leading to rebalances if they don't override the `max.poll.interval.ms` config. The fix for them would be to set the config to something reasonable for their application, which would be a runtime fix.

## Rejected Alternatives

In the ticket, we discussed even shorter defaults of 30s or 1m, but this would put even more applications at risk for spurious timeouts.