

KIP-657: Add Customized Kafka Streams Logo

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

Status

Current state: *Under Discussion*

Discussion thread: TBD

JIRA:



Note, this is a joint proposal from [Sophie Blee-Goldman](#) [guozhang Wang](#) [Matthias Sax](#) [Bruno Cadonna](#) [John Roesler](#) and [Boyang Chen](#)

Motivation

Kafka Streams has been receiving wide adoptions in the community as a robust library to build stream processing frameworks. One of the awkward situation we have been encountering is that when making a technical comparison with other streaming frameworks, Kafka Streams doesn't have a dedicated logo to represent. Using the general Kafka logo was ok, but it doesn't imply the stream processing component and would normally cause more confusions than necessary. Kafka Streams deserves a separate logo which could be very distinguishable as well as have a very strong connection to both ideas of Apache Kafka and stream processing holistically.

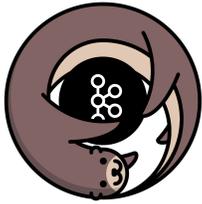
Proposed Changes

We are proposing two designs of the new Kafka Streams logo with a cute otter:

Design A



Design B



Kafka Streams

To decide the final winner, we would like to start a public vote thread on the dev mailing list. The voting thread will be open for one week, and the criteria for the competition would be collecting both binding and non-binding votes at the same time, with the following comparison rule:

- If design A and design B have different binding votes, whoever gets more vote wins.
- If design A and design B have the same number of binding votes, compare the non-binding votes

Later the winner will be used to update the official Kafka website and necessary README files, etc.

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

N/A