

KIP-312: Add Overloaded StreamsBuilder Build Method to Accept java.util.Properties

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

Status

Current state: *Accepted*

Discussion thread: [here](#)

Voting Thread: [here](#)

JIRA: [here](#)

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

Motivation

With the desire to make topology optimization optional via configuration, we introduced a new config option in KIP-295. The status quo now for building a Kafka Streams topology with the DSL is Kafka Streams writes the physical plan as the user makes calls on the DSL.

Once the user has finished building the topology with the DSL, the next step is to create a `KafkaStreams` instance passing in `Topology` and `java.util.Properties` instances as constructor arguments.

Users call the `StreamsBuilder.build()` method, returning the `Topology` instance. With the topology optimization in mind, we no longer build the physical plan immediately, but it's during the `StreamsBuilder.build()` call that Kafka Streams makes and optimizes the physical plan of the topology.

However, at this point, there is no access to the `StreamsConfig` object from the `StreamsBuilder`, which is necessary to determine if optimization is to be applied. Additionally, any further optimizations based on user configs will need access to configs set by the user.

Public Interfaces

Currently the `StreamsBuilder` class has a build method:

Existing StreamsBuilder Build Method

```
/**
 * Returns the {@link Topology} that represents the specified processing logic.
 *
 * @return the {@link Topology} that represents the specified processing logic
 */
public synchronized Topology build()
```

This KIP proposes to add an **additional** overloaded `StreamsBuilder.build` method

New StreamsBuilder Method

```
/**
 * Accepts user defined configs for a Kafka Streams app that maybe used
 * to turn on optimization.
 *
 * @param props {@link java.util.Properties} representing user configs
 * @return the {@link Topology} that represents the specified processing logic
 */
public synchronized Topology build(Properties props)
```

Proposed Changes

The added method will provide the needed configuration values when building the topology, namely whether we should optimize or not. In future releases, it will also allow for determining if the latest optimizations should be applied vs. optimizations from the previous version. Also, this will allow for other optimizations based on the configs. Using the older `StreamsBuilder.build` method taking no parameters will be an indication to perform no optimization and the Kafka Streams physical plan will be written as is.

Compatibility, Deprecation, and Migration Plan

- Since this is adding a method, there is no expected impact on existing users. However users wishing to use optimization will need to update their code to use the method
- There is no plan at the moment to deprecate the existing no-arg build method

Rejected Alternatives

N/A