

KIP-245: Use Properties instead of StreamsConfig in KafkaStreams constructor

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

Status

Current state: "Accepted" (v1.2.0)

Discussion thread: No discussion thread: [Vote for KIP-245: Use Properties instead of StreamsConfig in KafkaStreams constructor](#)

JIRA: [here](#)

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

Motivation

Currently, KafkaStreams constructor has overloads that take either Properties or StreamsConfig a parameters.

Because StreamsConfig is immutable and is created from a Properties object itself, the constructors accepting StreamsConfig are not useful and adds only boiler plate code. Thus, we should deprecate those constructors in order to remove them eventually.

Public Interfaces

```
// Old API (using StreamsConfig)
public KafkaStreams(final Topology, final StreamsConfig config)
public KafkaStreams(final Topology, final StreamsConfig config, final KafkaClientSupplier clientSupplier)
public KafkaStreams(final Topology, final StreamsConfig config, final Time time)

// New API (using Properties)
public KafkaStreams(final Topology, final Properties props)
public KafkaStreams(final Topology, final Properties props, final KafkaClientSupplier clientSupplier)
public KafkaStreams(final Topology, final Properties props, final Time time)
public KafkaStreams(final Topology, final Properties props, final KafkaClientSupplier clientSupplier, final
Time time)
```

Proposed Changes

We suggest to deprecate all the constructors using StreamsConfig, and use java.util.Properties instead.

Compatibility, Deprecation, and Migration Plan

- *We are not removing any existing API, so this change should be backward compatible. All new APIs are straightforward which don't require excessive comments.*

Rejected Alternatives

Not applicable.