

KIP-307 implementation tracking

- [Public Interfaces](#)
 - [Deprecation](#)
 - [Overloaded methods](#)
 - [Name Validation](#)

This page is for marking off proposed interfaces as they're implemented.

Public Interfaces

~~First, we propose to add one new interface `NamedOperation` that can be used to customize stateless operations as well as stateful ones. The objective to create a new class is to keep consistent with the overall API design.~~

```
package org.apache.kafka.streams.kstream;

/**
 * Default interface which can be used to personalized the named of operations, internal topics or store.
 */
public interface NamedOperation<T> extends NamedOperation<T>> {

    /**
     * Sets the name to be used for operation.
     *
     * @param name the name to use.
     * @return an instance of {@link NamedOperation}
     */
    T withName(final String name);
}
```

The `NamedOperation` interface will be implemented/extended by following classes that already exist to configure operations :

- ~~Produced~~
- ~~Consumed~~
- ~~Printed~~
- ~~Joined~~
- ~~Grouped~~
- ~~Suppressed~~

In addition, we propose to add a new static method with the following signature to each of those class **as(final String processorName)**.

(Suppressed is missing it, but we discussed this and decided not to include it)

Printed is missing it, not sure if this is intentional

Deprecation

~~In order to fix some inconsistencies in API, we also propose to deprecate the method **named(String)** from the class `Joined` in favor of new method **as()**.~~

~~In addition, as no configuration classes expose the processor name, we will also deprecate the method `name()` from the class `Joined`. This method should be removed in a future release.~~

Overloaded methods

~~Then, we propose to overload all stateless methods that do not have one of the existing control classes listed above to accept a `NamedOperation` implementation.~~

~~For that we will added a new default class `Named` implementing `NamedOperation` :~~

```

public class Named implements NamedOperation<Named> {

    private static final int MAX_NAME_LENGTH = 249;

    protected String name;

    protected Named(final String name) {
        this.name = name;
        if (name != null)
            validate(name);
    }

    /**
     * Create a Named instance with provided name.
     *
     * @param name the processor name to be used. If {@code null} a default processor name will be generated.
     * @return A new {@link Named} instance configured with name
     */
    public static Named as(final String name) {
        Objects.requireNonNull(name, "name can't be null");
        return new Named(name);
    }

    @Override
    public Named withName(final String name) {
        Objects.requireNonNull(name, "name can't be null");
        return new Named(name);
    }

    ...
}

```

The below tables will resume all new and existing methods :

KStream (16 new methods)

method	Added for this KIP ?	Object/method used for node name	Used for repartition topic name	Used for state store name ?
filter(Predicate, Named)	YES	static Named#as(String)	N/A	N/A
filterNot(Predicate, Named)	YES	static Named#as(String)	N/A	N/A
selectKey(KeyValueMapper, Named)	YES	static Named#as(String)	N/A	N/A
map(KeyValueMapper, Named)	YES	static Named#as(String)	N/A	N/A
mapValues(ValueMapper, Named)	YES	static Named#as(String)	N/A	N/A
mapValues(ValueMapperWithKey, Named)	YES	static Named#as(String)	N/A	N/A
flatMap(KeyValueMapper, Named)	YES	static Named#as(String)	N/A	N/A
flatMapValues(ValueMapper, Named)	YES	static Named#as(String)	N/A	N/A
flatMapValues(ValueMapperWithKey, Named)	YES	static Named#as(String)	N/A	N/A
print(Printed)	NO	static Printed#as(String)	N/A	N/A
foreach(ForeachAction, Named)	YES	static Named#as(String)	N/A	N/A
peek(ForeachAction, Named)	YES	static Named#as(String)	N/A	N/A
branch(Named, Predicate...)	YES	static Named#as(String)	N/A	N/A
through(String, Produced)	NO	static Produced#as(String)	N/A	N/A
to(String, Produced)	NO	static Produced#as(String)	N/A	N/A
to(TopicNameExtractor, Produced)	NO	static Produced#as(String)	N/A	N/A
transform(TransformerSupplier, Named, String...)	YES	static Named#as(String)	N/A	N/A
transformValues(ValueTransformerSupplier, Named, String...)	YES	static Named#as(String)	N/A	N/A

transformValues(ValueTransformerWithKeySupplier, Named, String...)	YES	static Named#as(String)	N/A	N/A
process(ProcessorSupplier, Named, String...)	YES	static Named#as(String)	N/A	N/A
join(KStream, ValueJoiner, JoinWindows windows, Joined)	NO	static Joined#as(final String name)	static Joined#as(final String name)	static Joined#as(final String name)
leftJoin(KStream, ValueJoiner, JoinWindows, Joined)	NO	static Joined#as(final String name)	static Joined#as(final String name)	static Joined#as(final String name)
outerJoin(KStream, ValueJoiner, JoinWindows, Joined)	NO	static Joined#as(final String name)	static Joined#as(final String name)	static Joined#as(final String name)
join(KTable, ValueJoiner, Joined)	NO	static Joined#as(final String name)	static Joined#as(final String name)	N/A
leftJoin(KTable, ValueJoiner, Joined)	NO	static Joined#as(final String name)	static Joined#as(final String name)	N/A
join(GlobalKTable, KeyValueMapper, ValueJoiner, Named)	YES	static Named#as(String)	N/A	N/A
leftJoin(GlobalKTable, KeyValueMapper, ValueJoiner, Named)	YES	static Named#as(String)	N/A	N/A
flatMapTransform(TransformerSupplier, Named named, String... stateStoreNames)	YES	static Named#as(String)	N/A	N/A
flatMapTransformValues(ValueTransformerWithKeySupplier, Named named, String...)	YES	static Named#as(String)	N/A	N/A
flatMapTransformValues(ValueTransformerSupplier, Named, String...)	YES	static Named#as(String)	N/A	N/A

KTable (16 new methods)

method	Added for this KIP ?	Object/method used for node name	Used for repartition topic name	Used for state store name ?
filter(Predicate, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
filter(Predicate, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
filterNot(Predicate, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
filterNot(Predicate, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
mapValues(ValueMapper, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
mapValues(ValueMapperWithKey, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
mapValues(ValueMapper, Named, Materialized)	YES	static Named#as(String)	N/A	static Materialized#as (String)
mapValues(ValueMapperWithKey, Named, Materialized);	YES	static Named#as(String)	N/A	static Materialized#as (String)
suppress(Suppressed)	NO	Suppressed#withName (String)	N/A	N/A
transformValues(ValueTransformerWithKeySupplier, Named, String...)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
transformValues(ValueTransformerWithKeySupplier, Materialized, Named, String...)	YES	static Named#as(String)	N/A	static Materialized#as (String)
groupBy(KeyValueMapper, KeyValue, Grouped)	NO	static Grouped#as(String)	static Grouped#as(String)	N/A
join(KTable, ValueJoiner, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
join(KTable, ValueJoiner, Named, Materialized)	YES	static Named#as(String)	N/A	static Materialized#as (String)
leftJoin(KTable, ValueJoiner, Named);	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
leftJoin(KTable, ValueJoiner, Named, Materialized)	YES	static Named#as(String)	N/A	static Materialized#as (String)
outerJoin(KTable, ValueJoiner, Named);	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
outerJoin(KTable, ValueJoiner, Named, Materialized)	YES	static Named#as(String)	N/A	static Materialized#as (String)
toStream(Named)	YES	static Named#as(String)	N/A	N/A

toStream(KeyValueMapper, Named)	YES	static Named#as(String)	N/A	N/A
---------------------------------	-----	-------------------------	-----	-----

KGroupedStream (6 new methods)

method	Added for this KIP ?	Object/method used for node name	Used for repartition topic name	Used for state store name ?
count(Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
count(Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
reduce(Reducer, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
reduce(Reducer, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
aggregate(Initializer, Aggregator, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
aggregate(Initializer, Aggregator, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)

KGroupedTable (6 new methods)

method	Added for this KIP ?	Object/method used for node name	Used for repartition topic name	Used for state store name ?
count(Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
count(Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
reduce(Reducer, Reducer, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
reduce(Reducer, Reducer, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
aggregate(Initializer, Aggregator, Aggregator, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
aggregate(Initializer, Aggregator, Aggregator, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)

TimeWindowedKStream (6 new methods)

method	Added for this KIP ?	Object/method used for node name	Used for repartition topic name	Used for state store name ?
count(Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
count(Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
aggregate(Initializer, Aggregator, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
aggregate(Initializer, Aggregator, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
reduce(Reducer, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
reduce(Reducer, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)

SessionWindowedKStream (6 new methods)

method	Added for this KIP ?	Object/method used for node name	Used for repartition topic name	Used for state store name ?
count(Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
count(Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
aggregate(Initializer, Aggregator, Merger, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
aggregate(Initializer, Aggregator, Merger, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)
reduce(Reducer, Named)	YES	static Named#as(String)	N/A	(PREFIX + COUNT)
reduce(Reducer, Named, Materialized)	YES	static Named#as(String)	N/A	Materialized#as(String)

Name Validation

User provided node name should follow the same restrictions that ones currently apply to state stores during the create of Materialized instance.

Currently, the Materialized class relies on the static method `Topic#validate`. This method ensure that a provided name only contains legal characters [a-zA-Z0-9._] and have a maximum length of 249.

We propose to copy methods from `Topic#validate` into `Named`. This new method will be used validate both store names and node names. The benefit is to remove a dependency with the core module.

In addition, the Materialized class will throw a `TopologyException` while building the topology in case of a unvalid name instead of `InvalidTopicException`.

Proposed Changes

- Implement the new interface `NamedOperation` and default class `Named`
- Update all parameter class to implement `NamedOperation` : `Produced` `Consumed` `Printed` `Joined` `Grouped` `Suppressed`
- Overload methods `stateless` for classes `KStreams`, `KTables`, `KGroupedStream`, `KGroupedTable`, `TimeWindowedKStream`, `TimeWindowedKTable`
- The processor names specified by developer will be used in place of the static processor prefix. Statics prefixes will still be used if no custom processor name are specified.
- Processor names should follow the same restrictions as the topic names. So legal characters are [a-zA-Z0-9._] and the maximum length of 249.