KIP-321: Update TopologyDescription to better represent Source and Sink Nodes

Status

Current state: *"Accepted"*

Discussion thread: here

JIRA: *KAFKA-6966*

Release*: 2.1*

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

Motivation

The original intention of **TopologyDescription** was to allow users to perform runtime checking. However, large number of users have opted to print or parse the textual representation of the Topology for convenience instead. The problem is the current **TopologyDescription** relies too heavily on String to represent the Source and Sink Nodes. By returning the underlying object and overriding **toString()**, users can still leverage **TopologyDescription** for runtime checks while also preserving the human readable representations of the Topology.

Public Interface

The underlying object for a Source node in **TopologyDescription** is set of topics or a pattern to match source topics on. Similarly, the underlying object of a Sink node is either the final topic if it is statically determined or the **TopicNameExtractor** used to the determine the topic.

```
public interface TopologyDescription {
        // Other interfaces and variables within TopologyDescription are not shown here.

        interface Source extends Node {

                Set<String> topicSet()

                   Pattern topicPattern();
        }

        interface Sink extends Node {
            String topic();

                // Add abstract method to return the TopicNameExtractor class in situations where dynamic 
routing is used.
                // Otherwise, return null.
            TopicNameExtractor topicNameExtractor();
        }
}
```

Proposed Changes

Add the above interface change and add the following to **InternalTopologyBuilder.java**.

```
public final static class Source extends AbstractNode implements TopologyDescription.Source {
        // Other methods and variables are not shown here


        @Override
        public Set<String> topicSet() {
            return topics;
        }

        @Override
        public Pattern topicPattern() {
            return topicPattern;
        }
}


public final static class Sink extends AbstractNode implements TopologyDescription.Sink {
        // Other methods and variables are not shown here


        // Output the topic name if dynamic routing is not used. Otherwise, output the toString value of the TopicNameExtractor.
        @Override
        public TopicNameExtractor topicNameExtractor() {
            if (topicNameExtractor instanceof StaticTopicNameExtractor) {
                return null;
                 } else {
                return topicNameExtractor;
                }
        }
}
```

Compatibility, Deprecation, and Migration Plan

Deprecating **topics()** because it is no longer needed with the addition of **topicSet()**. In order to get the same functionality, a user can just call **topicSet().toString()**.

Rejected Alternatives