

Finding Topic and Partition Leader

Finding the Topic and Partition Leader data for a cluster

Finding all the Topics, Partitions, Partition Leaders and Replicas for a cluster requires using a TopicMetadataRequest in the SimpleConsumer.

The following code will connect to a Broker and print out all the Topics, Partitions, which Broker is the leader and which Brokers are replicas for the Partitions.

By passing an empty List<String> to the TopicMetadataRequest you are asking Kafka to return data on all Topics.

```
import kafka.javaapi.TopicMetadataRequest;
import kafka.javaapi.consumer.SimpleConsumer;

import java.util.ArrayList;
import java.util.List;

public class MetadataDump {
    public static void main(String[] args) {
        kafka.javaapi.consumer.SimpleConsumer consumer = new SimpleConsumer("broker1.test",
            9092,
            100000,
            64 * 1024, "test");

        List<String> topics2 = new ArrayList<String>();
        TopicMetadataRequest req = new TopicMetadataRequest(topics2);
        kafka.javaapi.TopicMetadataResponse resp = consumer.send(req);
        List<kafka.javaapi.TopicMetadata> data3 = resp.topicsMetadata();
        for (kafka.javaapi.TopicMetadata item : data3) {
            System.out.println("Topic: " + item.topic());
            for (kafka.javaapi.PartitionMetadata part: item.partitionsMetadata() ) {
                String replicas = "";
                String isr = "";
                for (kafka.cluster.Broker replica: part.replicas() ) {
                    replicas += " " + replica.host();
                }
                for (kafka.cluster.Broker replica: part.isr() ) {
                    isr += " " + replica.host();
                }
                System.out.println( "    Partition: " + part.partitionId() + ": Leader: " + part.leader().
                    host() + " Replicas:[" + replicas + "] ISR:[" + isr + "]" );
            }
        }
    }
}
```

To find the details for a specific Topic, add the name to the List<String> passed to the TopicMetadataRequest:

```
List<String> topics2 = new ArrayList<String>();
topics2.add("my-topic");
TopicMetadataRequest req = new TopicMetadataRequest(topics2);
kafka.javaapi.TopicMetadataResponse resp = consumer.send(req);
```