Trident

public class TridentWordCount {

public static class Split extends **BaseFunction** {

@Override

public void execute(TridentTuple tuple, TridentCollector collector) {

String sentence = tuple.getString(0);

for (String word : sentence.split(" ")) {

collector.emit(new Values(word));

}

}

}

public static StormTopology buildTopology(LocalDRPC drpc) {

**FixedBatchSpout** spout = new FixedBatchSpout(new Fields("**sentence**"), 3, new Values("the cow jumped over the moon"),

new Values("the man went to the store and bought some candy"),

new Values("four score and seven years ago"),

new Values("how many apples can you eat"),

new Values("to be or not to be the person"));

spout.setCycle(true);

**TridentTopology** topology = new TridentTopology();

**TridentState** wordCounts = topology.**newStream**("spout1", spout)

.parallelismHint(16)

.**each**( new Fields("**sentence**"), new Split(), new Fields("**word**") )

.groupBy(new Fields("word"))

.persistentAggregate(new MemoryMapState.Factory(),

new **Count()**, new Fields("**count**"))

.parallelismHint(16);

topology.newDRPCStream("words", drpc)

.each(new Fields("args"), new Split(), new Fields("word"))

.groupBy(new Fields("word"))

.stateQuery(wordCounts, new Fields("word"), new **MapGet**(), new Fields("count"))

.each(new Fields("count"), new **FilterNull**())

.project(new Fields(**"word", "count"**));

return topology.build();

}

public static void main(String[] args) throws Exception {

Config conf = new Config()

;

conf.setMaxSpoutPending(20);

if (args.length == 0) {

try (LocalDRPC drpc = new LocalDRPC();

LocalCluster cluster = new LocalCluster();

LocalTopology topo = cluster.submitTopology("wordCounter", conf, buildTopology(drpc));) {

for (int i = 0; i < 100; i++) {

System.out.println("DRPC RESULT: " + drpc.execute("words", "cat the dog jumped"));

Thread.sleep(1000);

}

}

}

else {

conf.setNumWorkers(3);

StormSubmitter.submitTopologyWithProgressBar(args[0], conf, buildTopology(null));

}

}

}