

[x Dismiss](#)

Join the Stack Overflow Community

Stack Overflow is a community of 7.6 million programmers, just like you, helping each other.
Join them; it only takes a minute:

[Sign up](#)

sharing a BlockingQueue in a storm spout

```
36 if (dev.isBored() || job.sucks()) {  
37     searchJobs({flexibleHours: true, companyCulture: 100});  
38 }  
39 // A career site that's by developers, for developers.
```

[Get started](#)

I am programming a big data application in which two threads running concurrently. Thread A receives data from network and puts them as JSONOBJECT in a BlockingQueue. Thread B, a storm spout, then reads from the BlockingQueue and process them.

I pass the BlockingQueue object to the spout class in the class constructor. The problem I found is that the BlockingQueue in the spout is empty. Could you please let me know how can I solve this problem?

[apache](#) [apache-storm](#)

asked Feb 2 at 17:22



[Ebrahim](#)
6 1

1 Answer

You start a storm application by running some class that builds and configures the topology as a set of objects and then submits that collection of objects (along with the jar file) to the Nimbus server. Some of those objects are instances of the spouts and bolts which are serialized as part of the topology submission. Each instance of the bolt and spout on the cluster is one of these deserialized objects. So all bolts and spouts are constructed when you first start the topology (usually on an edge node) and not on the cluster.

What this means to you is that any objects referenced by the spout during class initialization and object construction are serialized along with the spout instance. This would include the BlockingQueue. Your BlockingQueue is being serialized and distributed to the cluster and it sounds like it's not surviving the trip.

What you want to do is leave the blocking queue variable null in the constructor and instead set the variable in the open() method. When you create the actual queue object you might store it in a public static variable somewhere so that it's available to the spout's open() method.

answered Feb 3 at 3:24



[Chris Gerken](#)
12.8k 5 28 48