



Twitter Heron Discussion

Ben Ujcich

February 16, 2016

CS 525

Logo source: Twitter, Inc. (https://g.twimg.com/Twitter_logo_blue.png)

Piazza Discussion: Pros

- Interoperability with Storm API
- Assumption that each worker is *not* homogenous
- Debugging
- No single points of failure
- Decoupling topologies from one another in terms of abstraction and implementation
- Practical, real-world use

Piazza Discussion: Cons (1)

- Lack of exactly-once semantics
- Not open source
- Reaction times to backpressure – how to determine low/high marks
- Backpressure from the “source” queues from which the spouts read data
- Other metrics besides throughput, latency, and CPU usage

Piazza Discussion: Cons (2)

- Generalizable beyond Twitter use cases and workloads?
- Improvements at what cost?
- Elasticity
- Assumptions about scalability with k streaming managers and n Heron instances

Own Thoughts

- Efficiency of the network
 - Resource allocation
 - Dedicated bandwidth – is the network a bottleneck?
 - SDN-based approach