Big Data

Storm Topology & Management

Storm Topology & Management

❖ Apache Storm Topology Operations

- Storm Transactional Phases
  - Commit phase
    - Any number of Blots can participate in the Commit phase, which are called Committers
    - Strong Ordering
      - Commits are ordered based on how the batches were admitted

Storm Topology & Management

❖ Apache Storm Topology Operations

- Storm Transactional Phases
  - Processing phase
    - Strom default tuple tree completion time is 30 s
    - If 30 s expires then Spout re-executes the process of the tuple tree again
    - At least once guarantee of processing
      - Exactly once semantics
        - If a failure occurs, then that tuple/batch is processed again
        - If tuple/batch succeeds then move on to the next tuple/batch

Storm Topology & Management

❖ Apache Storm Topology Operations

- Storm with **Mesos**
  - Mesos can be used to manages the cluster Topology distribution of Task and processing Core assignments
  - Mesos can enable different Storm Topologies to be executed independently without interfering with other Topologies running within the same cluster

Storm Topology & Management

❖ Storm Management Commands

- Jar
  - Jar is used to submit a topology to the cluster
  - Jar process execution order
    1. Topology jar file will be uploaded to the Nimbus

```
storm jar topology_jar topology_class [arguments]
```

Storm Topology & Management

❖ Storm Management Commands

- Jar
  - Jar process execution order
    2. Nimbus will distribute topology tasks to the Supervisors through the help of ZooKeeper
    3. Jar will run the **`main( )`** based on the topology class and **`arguments`** specified
    4. Storm activates the Topology and starts the processing within the cluster

Storm Topology & Management

❖ Storm Management Commands

- Deactivate
  - Stop streaming of tuples from the Spout(s)
  - Storm UI can be used to deactivate a topology
- Activate
  - Activate or resume to stream tuples from the Spout(s)
  - Storm UI can be used to activate a topology

Storm Topology & Management

❖ Storm Management Commands

- Spout Deactivate, Activate & Close Process
  - Spout can be Deactivated using "**`deactivate( )`**"
  - Serialization will be conducted to save the processing state information and tuples
  - To resume processing, Activate using "**`activate( )`**" can be used
  - Spout closing (with "**`close( )`**") will first deactivate the Spout and then close it

Storm Topology & Management

❖ Storm Management Commands

- Kill
  - Kill command (i.e., "**`kill( )`**") is used to terminate a topology in process
  - Kill process
    1. Serialization of the Spout and Bolts status and results will be conducted and the serialized files will be saved
    2. Spouts of the Topology will be deactivated
    3. Bolts will be stopped and the saved states will be cleaned up

Storm Topology & Management

❖ Storm Management Commands

- Rebalance
  - Used to reconfigure a topology
  - Rebalance will change the process conducted in the Spouts and/or Bolts
  - Rebalance will redistribute the task among the Spouts and Bolts in a cluster
    - Example: Used when an additional Supervisor node is added to the cluster

Storm Topology & Management

❖ Storm Management Commands

- Rebalance
  - Rebalance enables a nearly seamless swap of the topology
  - Having to kill and then resubmit a new topology would take at least tens of seconds or a few minutes to do

Storm Topology & Management

❖ Storm Management Commands

- Rebalance
  - Changes can be made to the number of workers (using the **"-n"** option) and executors (using the **"-e"** option)

    - Example use of the rebalance program options

```
storm rebalance topology_name [ -w wait_time] [ -n worker_count]
                              [-e component_name=executor_count] ...
storm rebalance key-topology -w 10 -n 20 -e first-spout=5 -e second-bolt=7
```

Storm Topology & Management

❖ Storm Management Commands

- Open
  - Internal initialization activities are opened using **open( )**
  - Open based operations
    - Collector
      - Collector is used when sending data into the cluster to be processed by the Bolts
    - Configuration
    - Task information
    - Connecting to a data source
    - etc.

Big Data

Reference

References

- Apache Storm, http://storm.apache.org
- Nathan Marz, "ETE 2012 - Nathan Marz on Storm," https://www.youtube.com/watch?v=bdps8tE0gYo&t=542s, Feb. 15, 2012.
- Wikipedia, https://en.wikipedia.org
- edureka!, "Understanding Spout In Apache Storm | Edureka," https://www.youtube.com/watch?v=5kiZs1a8UPM, Oct. 10, 2014
- https://www.webopedia.com/TERM/E/ETL.html
- Sean T. Allen, Matthew Jankowski, "Storm Applied: Strategies for real-time event processing", Apr. 12, 2015.
- Jonathan Leibiusky, Gabriel Eisbruch, Dario Simonassi, "Getting Started with Storm: Continuous Streaming Computation with Twitter's Cluster Technology",Sep. 17, 2012.
- Flavio Junqueira, Benjamin Reed, "ZooKeeper: Distributed Process Coordinationt", Dec. 5, 2013.

References

- **Image source**
  - By FreeStockTips (Own work) [CC BY-SA 4.0 (https://creativecommons.org/licenses/by-sa/4.0)], via Wikimedia Commons
  - https://upload.wikimedia.org/wikipedia/commons/8/80/New_York_Stock_Exchange_trading_floor_on_Wall_Street%2C_New_York%2C_New_York_LCCN2011634218.tif
  - https://upload.wikimedia.org/wikipedia/commons/b/bd/USB-thumb-drive-16-GB.jpg