Storm核心组件

了解Storm的核心组件对于理解Storm原理非常重要，下面介绍Storm的整体，然后介绍Storm的核心。

Storm集群由一个主节点和多个工作节点组成。主节点运行一个名为”Nimbus”的守护进程，工作节点都运行一个名为“Supervisor”的守护进程，两者的协调工作由ZooKeeper来完成，ZooKeeper用于管理集群中的不同组件。

每一个工作节点上运行的Supervisor监听分配给它那台机器的工作，根据需要启动/关闭工作进程，每一个工作进程执行一个Topology的一个子集；一个运行的Topology由运行在很多机器上的很多工作进程Worker组成。那么Storm的核心就是主节点（Nimbus）、工作节点（Supervisor）、协调器（ZooKeeper）、工作进程（Worker）、任务线程（Task）。

1. 主节点Nimbus

主节点通常运行一个后台程序 —— Nimbus，用于响应分布在集群中的节点，分配任务和监测故障，这类似于Hadoop中的JobTracker。

Nimbus进程是快速失败（fail-fase）和无状态的，所有的状态要么在ZooKeeper中，要么在本地磁盘上。可以使用kill -9来杀死Nimbus进程，然后重启即可继续工作。

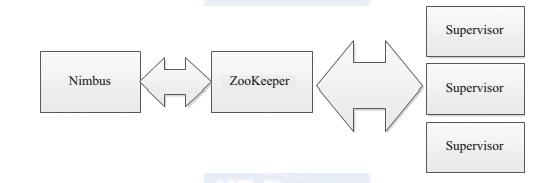
1. 工作节点Supervisor

工作节点同样会运行一个后台程序 —— Supervisor，用于收听工作指派并给予要求运行工作进程。每个工作节点都是Topology中一个子集的实现。而Nimbus和Supervisor之间的协调则通过ZooKeeper系统。

同样，Supervisor进程也是快速失败（fail-fast）和无状态的，所有的状态要么在ZooKeeper中，要么在本地磁盘上，用“kill -9”来杀死Supervisor进程，然后重启就可以继续工作。

1. 协调服务组件ZooKeeper

ZooKeeper是完成Nimbus和Supervisor之间协调的服务。Storm使用ZooKeeper协调集群，由于ZooKeeper并不用于消息传递，所以Storm给ZooKeeper带来的压力相当低。在大多数情况下，单个节点的ZooKeeper集群足够胜任，不过未了确保故障恢复或者部署大规模Storm集群，可能需要更大规模的ZooKeeper集群。Nimbus、Supervisor与ZooKeeper的关系如图所示。



1. 其他核心组件

Storm的组件不止上面的，还有一些组件也是Storm的核心，缺一不可。下面简单介绍Worker和Task。

1. 具体处理事务进程Worker

运行具体处理组件逻辑的进程。

1. 具体处理线程Task

Worker中的每一个Spout/Bolt线程称为一个Task。在Storm 0.8之后，Task不再与物理线程对应，同一个Spout/Bolt的Task可能会共享一个物理线程，该线程称为Executor。