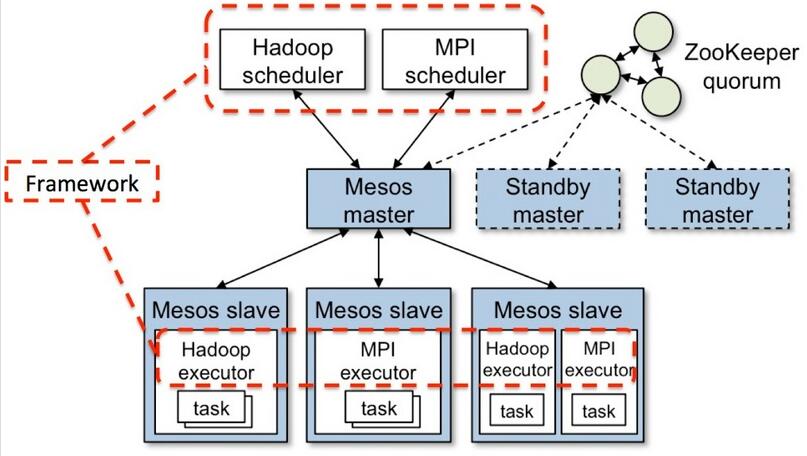
# Mesos

1. mesos介绍



Mesos-master：协调全部的slave，并确定每个节点的可用资源，聚合计算跨节点的所有可用资源的报告，然后向注册到Master的Framework发出资源邀约。

Mesos-slave：向master汇报自己的空闲资源和任务的状态，负责管理本节点上的各个mesos-task，在framework成功向Master申请资源后,收到消息的slave会启动相应framework的exexutor

Framework：Hadoop，Spark等，通过MesosSchedulerDiver接入Mesos

Executor：执行器，用于启动计算框架中的task

Apache Mesos是一款开源集群管理软件。Mesos实现了两级调度架构，可以管理多种类型的应用程序。

第一级调度是Master的守护进程，管理Mesos集群中所有节点上运行的Slave守护进程。集群由物理服务器或虚拟服务器组成，用于运行应用程序的任务，比如Hadoop和MPI作业。第二级调度由被称作Framework的“组件”组成。Framework包括调度器（Scheduler）和执行器（Executor）进程，其中每个节点上都会运行执行器。

Mesos master实际上是一个全局资源调度器，采用某种策略将某个slave上的空闲资源分配给某一个framework，各种framework通过自己的调度器向Mesos master注册，以接入到Mesos中。而Mesos slave主要功能是汇报任务的状态和启动各个framework的executor

Mesos由始至终只做一件事情，就是分布式集群资源的分配。任务的调度和执行由每个计算框架(Framework)自己完成,这样可以容易的实现mesos的扩展性和稳定性。

MesosSchedulerDriver

每个framework包含“scheduler”和“executor”两部分。framework通过运行scheduler来与mess进行交互，本质上scheduler就是一个进程。scheduler启动之后，首先会去连接mess以使用mesos的相关资源，在scheduler运行的过程中，它会向mess请求资源，如果资源满足要求，scheduler会向mess发起运行executor的请求来完成相关任务。

#运行C语言写的test framework

./mesos-0.28.0/build/src/test-framework --master=127.0.0.1:5050

#运行java写的test framework，该test framework的源码位置mesos-0.28.0/src/examples/java/

./mesos-0.28.0/build/src/examples/java/test-framework 127.0.0.1:5050

#运行python写的test framework，源码位置：./mesos-0.28.0/src/examples/python/，

./mesos-0.28.0/build/src/examples/python/test-framework 127.0.0.1:5050

进入到./mesos-0.28.0/src/examples/python/目录，可以发现该test framework包含两个文件:

test\_framework.py   test\_executor.py

其中test\_framework.py就是framework的scheduler，test\_executor.py是framework的executor

我们直接运行：python test\_framework.py  127.0.0.1:5050

