Mesos:

一个计算机集群资源管理框架，为应用提供计算机集群上的计算资源，如提供一些cpu，提供一些内存

架构：

Mesos-master：

主mesos节点，协调管理Mesos-slave和各种各样的计算framework。

用allocator插件将slave上的资源分配给各个framework。

Framework通过MesosSchedulerDiver接入Mesos

Mesos-slave：

从属Messo节点，管理本节点上的一些mesos-task，并为task分配本节点的资源。

向Mesos-master汇报自己的可用资源及task状态

比如：为各个executor分配资源。

Framework：

计算框架，需要使用mesos资源的各种计算框架，或者需要使用资源的程序。

Framework通过调度器Scheduler向mesos提交任务，与mesos交互

如Hadoop，spark，ElasticSearch等

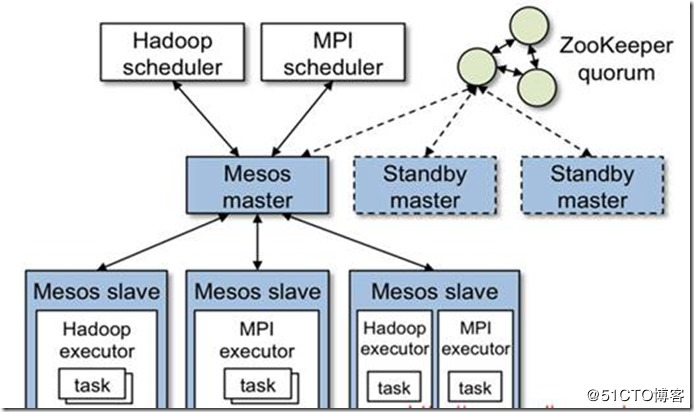
Executor：

执行器，用于启动Framework的task即用来实际运行软件程序。

mesos使用docker作为执行器来创建运行一个任务程序。

Framework task通过MesosExecutorDirver来让Executor以合适的方式来启动task

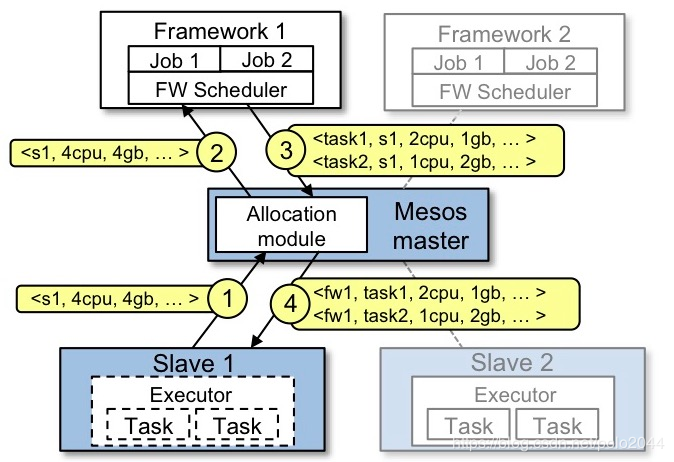
由于一个Master下管理了所有slave，当Master出现故障时，整个系统将失效，master存在单点故障问题。因此使用zookeeper来对Mesos-master进行集群部署，保持各master之间的通信，选举leader、HA



Mesos架构

运行流程：

1. Mesos-slave向mesos-master上报自己所在节点的资源（cpu，memory，storage..）和task状态
2. Mesos-master向Framework调度器描述其下slave的可用资源
3. Framework Scheduler发现有资源可用，向Mesos-master发送需要执行的任务描述，开始调度运行任务
4. Mesos-master收到调度器的任务请求时，向其slave节点派发任务。
5. Slave节点收到task任务后使用Framework Executor执行任务。



Marathon：

是一种Mesos的框架（mesos架构中的那个Framework），负责根据Mesos资源调度任务

使用Marathon来运行程序，可以将应用运行于计算机集群上，而不必知道应用具体运行在那部计算机上。当有程序需要运行在分布式系统上时，需要通过Marathon来运行。如Hadoop可以通过marathon在mesos提供的集群资源上运行。上。

搭建一个Mesos集群

主计算机安装的软件

Mesos-Master，marathon， zookeeper（zookeeper用来对多态Mesos-Master进行集群部署，防止单点故障）

从属计算机安装的软件

Mesos-slave，docker （Executor使用docker来运行Framework task）

Zookeeper

ZooKeeper是一个分布式的，开放源码的分布式应用程序协调服务。

使用zookeeper的集群功能，解决Mesos的单点故障问题。只要部署多个Mesos-master同时配置zookeeper，并使用zookeeper集群来将Mesos-master群进行统一管理。选举master。