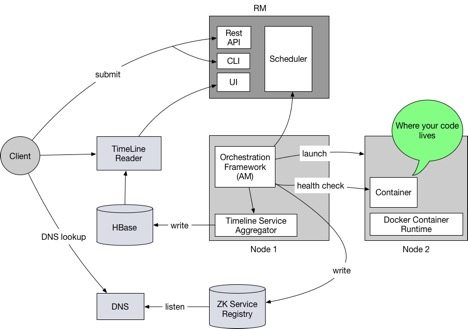
YARN Service Framework简介

Apache Hadoop YARN作为大数据领域通用的资源管理平台，提供复杂的集群资源管理和调度服务，其将通用业务逻辑抽象出来，让更多的计算框架专注于计算本身，开发者使用其提供的接口可以便捷的将任务运行在YARN中。

目前在Hadoop中可以运行多种作业类型，但是主要是MapReduce/Spark这种批处理工作负载，但是YARN社区一直致力于提供运行常驻服务的管理能力，在Hadoop 3.1中，YARN引入了Service Framework框架，用于启动和管理长期运行的服务，如Hive LLAP、HBase和基于容器(Docker)的服务，其让批处理作业和服务之间资源共享成为可能。其主要工作涉及到组件和流程如下图所示：

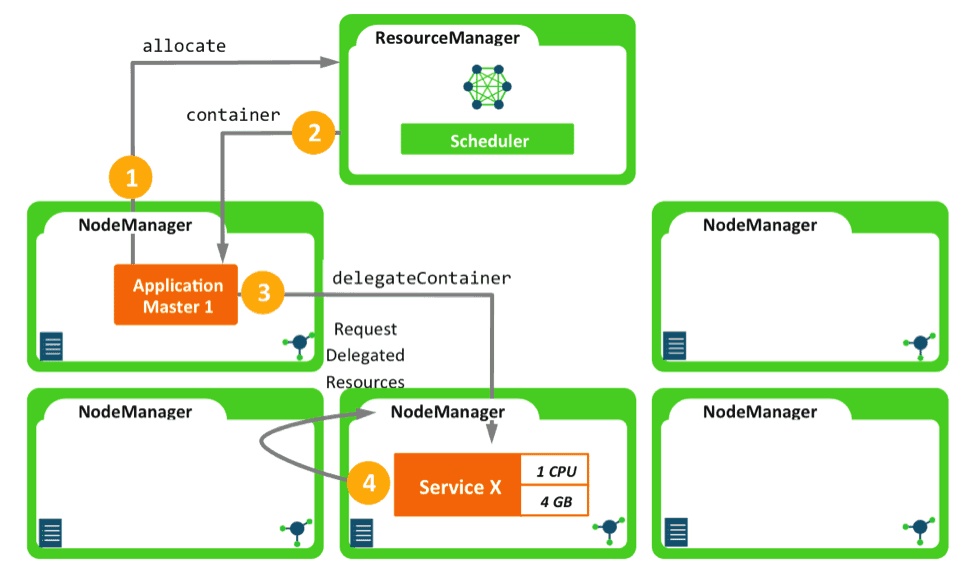


* AM，运行在YARN中，用于容器编排，负责所有服务的生命周期管理。其是Service Framework的核心
* 解析并执行服务定义和配置
* 管理容器生命周期，如Failed容器的自动启动
* 监控各组件之间的健康和状态，确保组件之间按照依赖顺序启动
* 新组件的弹性扩展、组件的升级等

其主要目的是确保服务以用户期望的状态运行。

* RESTful API服务，使用简单的JSON规范，供用户在YARN部署和管理服务。API Server是无状态的，因此可以通过Load Balancer来实现多个实例之间的负载均衡
* YARN DNS服务，通过DNS查找和发现YARN上的服务
* 高级容器编排调度，可以为容器及节点添加标签，支持容器的affinity和anti-affinity调度
* 支持容器和服务的滚动升级
* First Class Support for Docker
* YARN Timeline Service，基于HBase来记录生命周期事件和指标，并为用户提供丰富的分析API来查询应用程序的详细信息
* YARN UI2，提供服务的管理Web界面

在YARN中，Service底层复杂的资源管理、调度等逻辑对于用户是透明的，用户只需要通过CLI或者REST API向YARN中提交规范的JSON定义文件，就可以部署和管理运行在YARN上的服务。系统会自动启动、监听容器和其他需要保持应用程序运行的操作，比如自动重启运行识别失败的容器。YARN Service Framework的工作流程如下：



1. 用户定义JSON格式Service请求，描述服务的规格，如容器内存、CPU核数量、Docker ImageID等
2. 通过YARN CLI或者REST API提交服务
3. RM在接受请求后，启动ApplicationMaster，用于容器编排等功能
4. AM根据用户的资源请求从RM请求资源，然后在分配的NM上启动Container
5. YARN AM会监控容器的健康和执行情况，对容器的故障及异常进行处理。服务生命周期事件和指标写入YARN Timeline Server，同时额外的服务元信息（容器IP及端口）等写入由ZK支持的YARN服务注册表
6. Registry DNS监听Zookeeper中的znode创建和删除，并创建各种DNS记录来提供DNS查询
7. 根据JSON规范和YARN配置中提供的信息，每个容器都会有一个主机名，客户端使用标准的DNS查询容器名从而得到容器IP

参考链接：

https://hortonworks.com/blog/first-class-support-long-running-services-apache-hadoop-yarn/

https://mp.weixin.qq.com/s/HElR-IZS5lGllee8WG6gQA?utm\_medium=hao.caibaojian.com&utm\_source=hao.caibaojian.com