**YARN Service Discovery**

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，其执行流程如下图所示：

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

1. 通过YARN CLI或者REST API提交服务
2. RM在接受请求后，启动ApplicationMaster，用于容器编排等功能
3. 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
4. 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
5. 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
6. 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

YARN Service Framework启动服务后，服务实例运行在集群中任意节点，其IP和端口等信息动态分配，对用户是不可知的，并且由于系统的auto-scaling,failures等因素，服务运行实例数量也是动态变化，因此客户端需要服务发现机制。在YARN Service Framework中，引入了YARN Registry，将服务实例注册到Zookeeper中，Registry DNS监听注册的服务实例，通过DNS查询来提供给用户使用。

下面介绍其使用。

1. YARN Registry
2. YARN DNS
3. Service Discovery使用示例

https://www.jianshu.com/p/ece3e0ffc70c

https://issues.apache.org/jira/browse/YARN-4757

https://hadoop.apache.org/docs/r3.1.1/hadoop-yarn/hadoop-yarn-site/yarn-service/ServiceDiscovery.html

https://hadoop.apache.org/docs/r3.1.1/hadoop-yarn/hadoop-yarn-site/yarn-service/RegistryDNS.html