Slider技术内幕

——深入解析Slider架构和实现原理

## 第6章 Slider Router设计和实现

### 6.1 概述

通过Slider将Long-live的应用部署到YARN中，例如Apache Tomcat Web Servers或者HBase，应用的Server启动的主机和使用的端口等由YARN根据集群各主机的使用情况进行决策，因此每个应用的server进程的启动主机及端口都是不确定的。

客户端访问应用前，必须能够发现它们哪台主机上运行，并连接它们，尤其是出现故障时，YARN会自动的重启应用或者应用中的某个进程，客户端需要实时感知服务的运行状况并在发生变化时让客户端透明访问。

#### 6.1.1 Slider Router基本职能

整体上讲，Slider Router通过RPC协议与ResourceManager通信，通过JAVA API与ZK等Server进行信息交互，如图6-1所示：



图 6-1 Slider Router各模块及通信

* ApplicatioMasterProtocol，应用程序的ApplicationMaster通过该RPC协议向RM注册、申请资源和释放资源。该协议中，ApplicationMaster扮演RPC Client的角色，RM起到RPC Server的作用，采用了"pull"模型。
* ApplicationClientProtocol，应用程序的客户端通过该RPC协议向ResourceManager提交应用程序、查询应用程序状态和控制应用程序（比如杀死应用程序）等。该协议中，应用程序客户端为RPC Client，而RM为RPC Server。
* SliderRouterProtocol，Slider的客户端通过该RPC协议向Slider Router发送请求，例如查询Slider 启动的应用程序、应用程序的访问Endpoint、配置等，同时可以通过该协议向Slider Router发送应用程序的控制命令（Slider Router接收命令后向YARN提交客户端的请求）
* HTTP 协议，Slider Router启动程序后，客户端通过HTTP协议访问对应的应用程序，首先基于域名服务查找应用程序的启动IP和端口（如果有需要增加反向代理，作为一个转发，增加安全性）
* ZK Java API，应用程序启动后，将应用程序的访问信息注册到ZK中，Slider Router通过查询ZK获取binding信息（例如endpoint、rest、ipc、web ui等），然后根据这些访问信息更新DNS和Nginx的配置，以供客户端访问。

概括起来，Slider Router主要完成以下功能：

* 为客户端提供访问API，接收并处理客户端请求，例如应用程序的访问、管理等
* 与RM交互，启动应用程序，并根据客户端请求对应用程序进行扩容等操作
* 管理DNS及Nginx信息，从ZK中获取应用程序访问信息，更新到DNS及Nginx中，为客户端提供访问信息
* 提供Web UI，供用户实时查看应用实时信息

#### 6.1.2 Slider Router内部架构

### 6.2 SliderRouter各模块分析

### 6.3 SliderRouter DNS信息管理

### 6.4 SliderRouter 反向代理机制

### 6.5 几种常见行为分析

### 6.6 小结