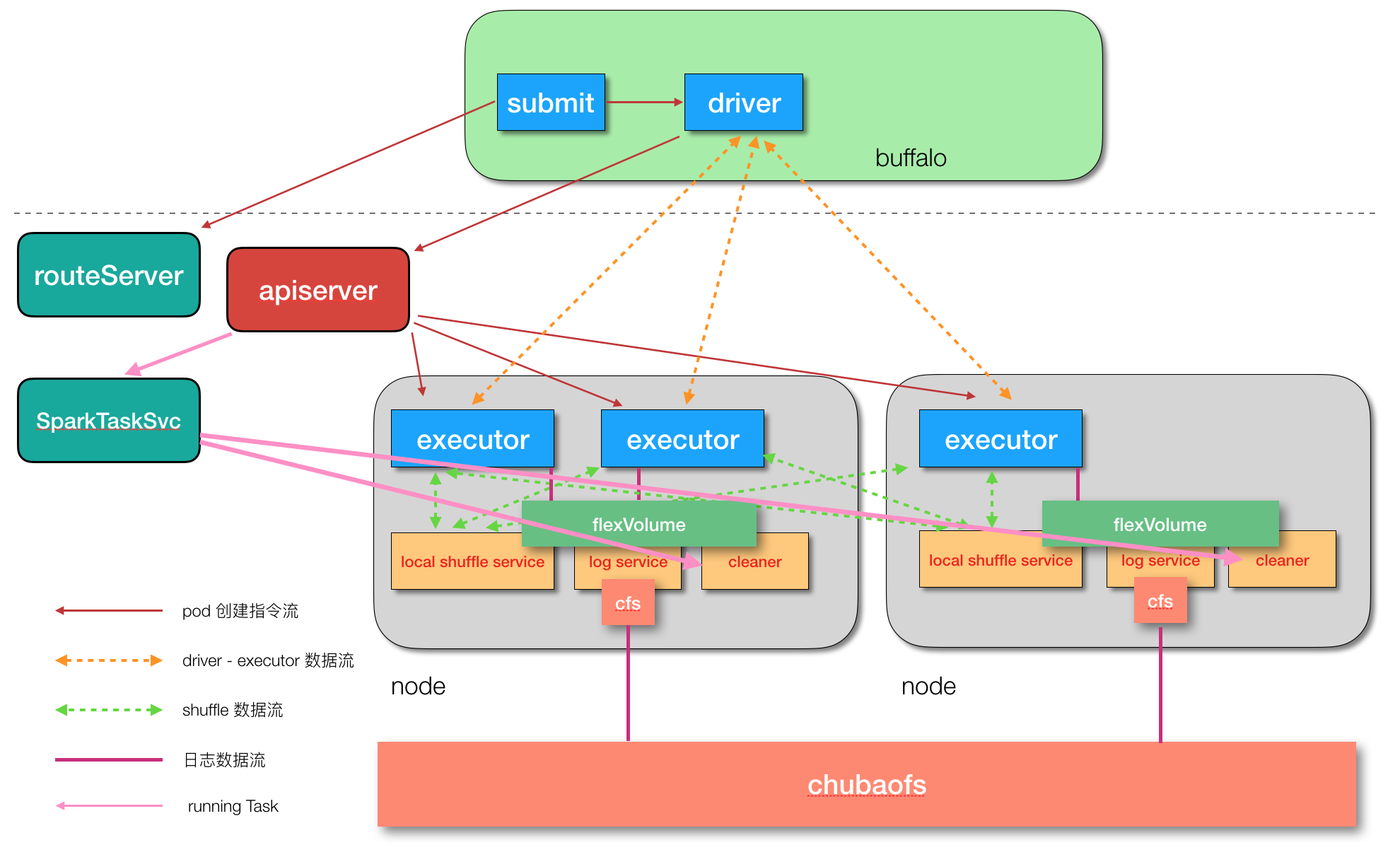
spark on jdos 相关辅助功能、组件设计

为支持spark程序能够平稳顺利的运行在jdos上，不仅需要对spark自身作出修改，同时也需要一些辅助功能和组件来实现 k8s 平台上不具备的能力，本文档主要介绍了这些相关辅助组件的设计及部署等信息。

为使spark能够顺利运行到Kubernetes集群上，需要以下辅助功能：

1. Kubernetes 集群发现功能，根据任务的集群、集市、队列为spark-submit 提供Kubernetes的配置（kubeconfig）；
2. shuffle 数据的存储及删除功能，支持local shuffle service，并且在任务结束后能够立即删除已经无用的shuffle 数据；
3. 日志的存储、管理及访问功能，提供日志存储、查看的功能。
4. 另外，因为jdos不支持hostpath，需要他们提供一个 shared flexVolume 的功能，就是支持一个 flexVolume 在同一个机器上，可以对多个Pod可见。

一次任务的执行过程为：

1. 在node上以daemonset 的方式运行 local-shuffle-service，local-shuffle-service会挂载一个 shared-flexVolume（shared-flexVolume 可以为同一台机器上的不同Pod提供一个可以共享的存储Volume）；
2. 在node上以daemonset 的方式运行一个 logservice，logservice 会同时挂载上述的 shared-flexVolume 以及 cfs 存储，spark 任务运行时，会将日志直接重定向到 shared-flexVolume 的一个目录，等任务运行完之后，会将 日志 rsync 到cfs 存储上，logservice 会同时提供一个 webserver 功能，提供日志文件的访问，该webserver 会先尝试访问 shared-flexVolume 的日志位置，未找到时再继续尝试 cfs 的目录位置；
3. spark-submit 获取到<集群，集市，队列>的信息，并以其为参数，向routeServer服务申请Kubernetes 访问配置，配置为一个完整的 kubeconf 配置文件，其中包含了apiserver 地址和用户认证配置（jdos为用户名和密码）；
4. spark-submit 获取到Kubernetes 访问配置后，创建出driver，driver用获取到的Kubernetes去Kubernetes 集群创建 Executor；
5. Executor 会挂载一个 shared-flexVolume，用于保存中间的shuffle数据（与local-shuffle-service共享位置）和临时的日志数据（临时是因为在任务结束后，日志会被移动到 cfs 上去）；

**router server(autoconfig)**

一个用于提供k8s集群访问配置的http server，根据所传参数返回对应集群的kubeconfig配置文件，避免了认证文件在 buffalo集群上的分发部署，以及后续可以基于此做任务的负载均衡。

代码位置：<https://git.jd.com/bag/spark-on-jdos/blob/master/autoconfig>

访问示例：

（<http://autoconfig.spark.bdp.jd.local:8611/v1/jdos/route?cluster=10k&market=mart_scr&queue=bdp_jmart_dwm_union.bdp_jmart_dwm_normal>）

或

（http://autoconfig.sparkonjdos.svc.bcc-ht1.jd.local:8611/v1/jdos/route?cluster=10k&market=mart\_scr&queue=bdp\_jmart\_dwm\_union.bdp\_jmart\_dwm\_normal）

$ curl -q 'http://autoconfig.sparkonjdos.svc.bcc-ht1.jd.local:8611/v1/jdos/route?cluster=10k&market=mart\_scr&queue=bdp\_jmart\_dwm\_union.bdp\_jmart\_dwm\_normal' | jq -cr '.configData'

apiVersion: v1

clusters:

- cluster:

insecure-skip-tls-verify: true

server: https://k8s-api-ht07.n.jd.local:6443

name: jdos-ht7

contexts:

- context:

cluster: jdos-ht7

namespace: sparkonjdos

user: jdos-ht7-client

name: jdos-ht7

current-context: jdos-ht7

kind: Config

preferences: {}

users:

- name: jdos-ht7-client

user:

password: 9V53mry7uNeQZF8fAcgVQg==

username: sparkoffline

部署位置：

HT1 k8s 集群， kubectl get pods -n sparkonjdos

**SparkTaskSvc**

因为jdos的磁盘资源较小，为了节省资源，在spark任务结束时，需要能够及时的将之前产生的临时数据和日志进行清理和和转移，就需要有一个接口来查询当期集群正在运行的任务有哪些，这样就可以清理掉那些不在运行中的任务数据和日志。

代码位置：<https://git.jd.com/bag/spark-on-jdos/tree/master/assistant>

访问示例：

$ curl 'http://sojas-apis.sparkonjdos.svc.ht7.n.jd.local/v1/apps'

{“cleaner\_switch":"on"} //无正在运行的任务

部署位置：

该任务是部署在 jdos 的集群上的：

<http://console.jdos.jd.com/#/sparkonjdos/sojas/group/>

**LogService**

logservice 是一个提供日志服务的简单http server，在spark entrypoint 中，将日志访问的环境变量设置为 本服务的访问地址（按照一定的pattern），从sparkui上访问日志时，会向logservice 发送http请求，拉取日志，如果是正在运行的任务，日志是直接重定向到 flexVolume的，logservice会从 flexVolume上查找对应的文件，如果没有找到，则说明该任务可能已经结束，日志文件被同步到了cfs上面，logservice会继续从 cfs 上查找该日志文件，如果还没有找到，则返回 404.

代码位置：

<https://git.jd.com/bag/spark-on-jdos/tree/master/logserver>

部署位置：

以daemonset 的方式部署在 spark 运行的集群上： <https://git.jd.com/bag/spark-on-jdos/blob/master/sparkcleaner/deploy/spark-assistant.ds.yaml>

**cleaner**：

cleaner 负责在 spark 任务运行结束后，将临时的shuffle数据进行删除，或者将日志数据备份到cfs 后再删除。

代码位置：<https://git.jd.com/bag/spark-on-jdos/tree/master/sparkcleaner>

部署位置：

以daemonset 的方式部署在 spark 运行的集群上： <https://git.jd.com/bag/spark-on-jdos/blob/master/sparkcleaner/deploy/spark-assistant.ds.yaml>