

天火分布式 workflow 调度系统

蒲雨露

2019/05/29

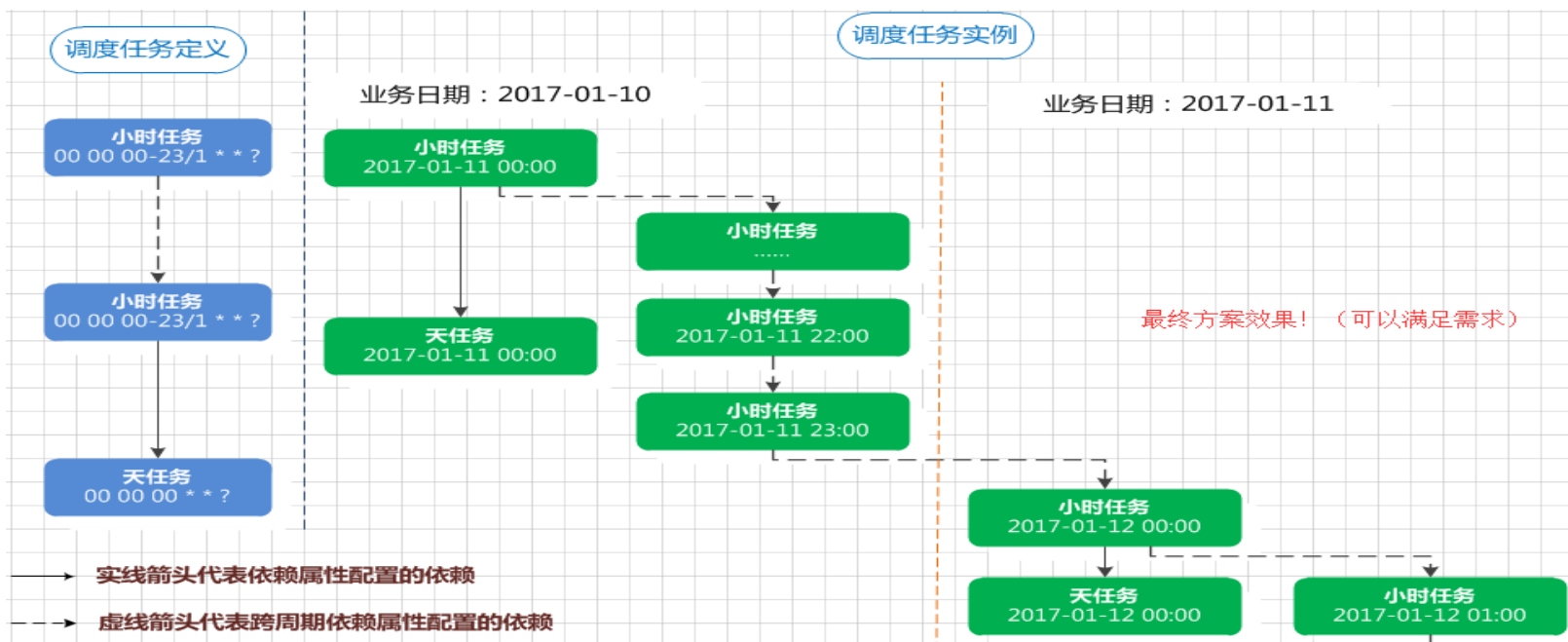
议程

- 背景
- 相关产品
- 系统功能
- 系统架构
- 组件介绍
- 系统使用

背景

业务需求

1. 系统需要统计截止到每小时的业务数据增量，然后在最后一个小时的数据汇总完成后需要一个任务进行一整天的汇总。



相关产品

Azkaban

- Windows, Linux
- 基于java
- 基于http
- 文件定义依赖
- dag
- 部署麻烦

天火

- Windows, Linux
- 基于java
- 基于rpc
- xml定义依赖
- dag
- 部署简单
- 性能高
- 资源隔离

Airflow

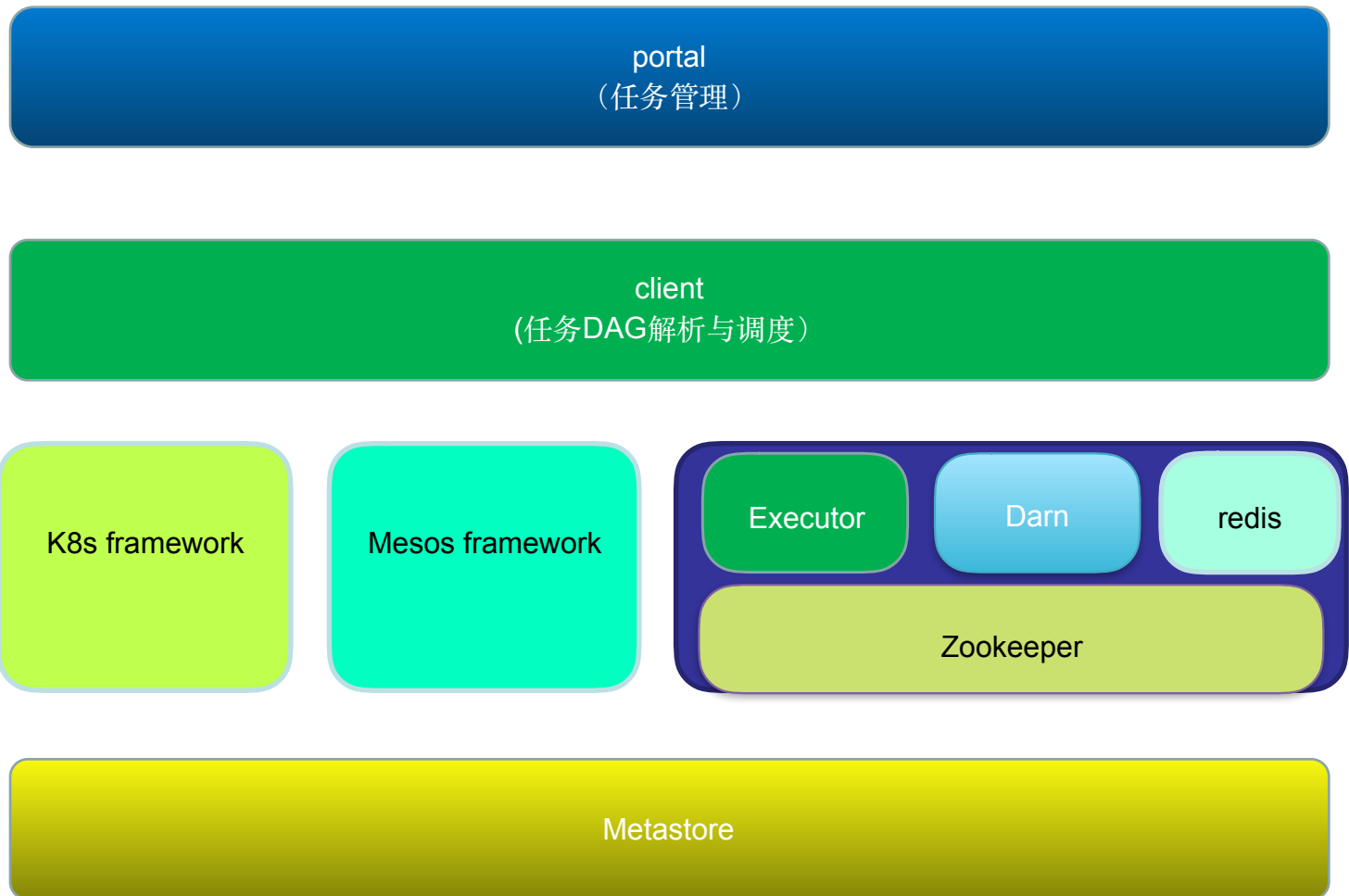
- Linux
- 基于python
- 基于celery, MQ等
- 代码定义依赖
- dag
- 大量任务性能比较低

系统功能

系统目前主要有如下功能：

- 自定义DAG
- 分布式部署
- 基于真实资源
- 资源隔离
- 支持大规模任务调度
- 节点掉线自动发现
- 任务失败后自动重试
- 任务失败后自动告警
- 可以通过页面调度，停止，查看任务详情和依赖等等
- 既可以执行普通的工作流又可以调度定时任务
- 可以调度大数据任务
- 基于容器化的调度

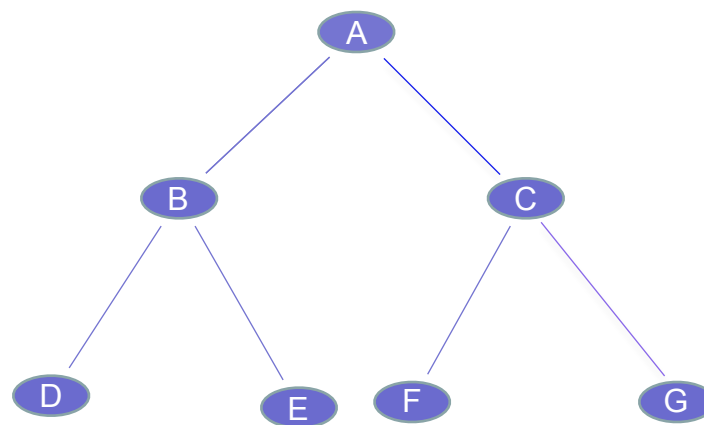
系统架构



组件介绍

Client

- 任务DAG解析
- 根据配置选择最适合机器
- 提交任务
- 任务重试



任务无环图

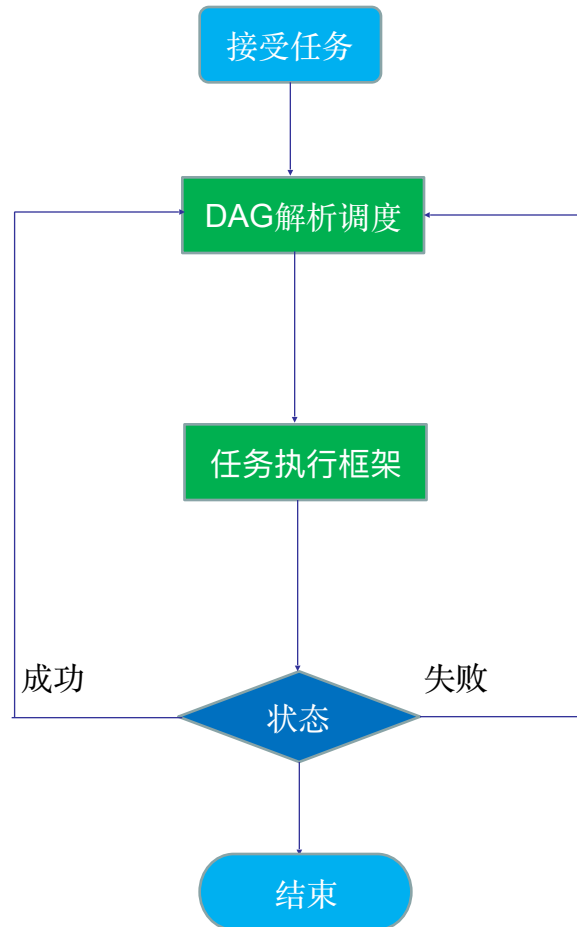
```
<?xml version="1.0"?>
- <jobplan>
  <job depends="" jobId="a"/>
  <job depends="a" jobId="b"/>
  <job depends="a" jobId="c"/>
  <job depends="b" jobId="d"/>
  <job depends="b" jobId="e"/>
  <job depends="c" jobId="f"/>
  <job depends="c" jobId="g"/>
</jobplan>
```

xml定义任务依赖

组件介绍

Client

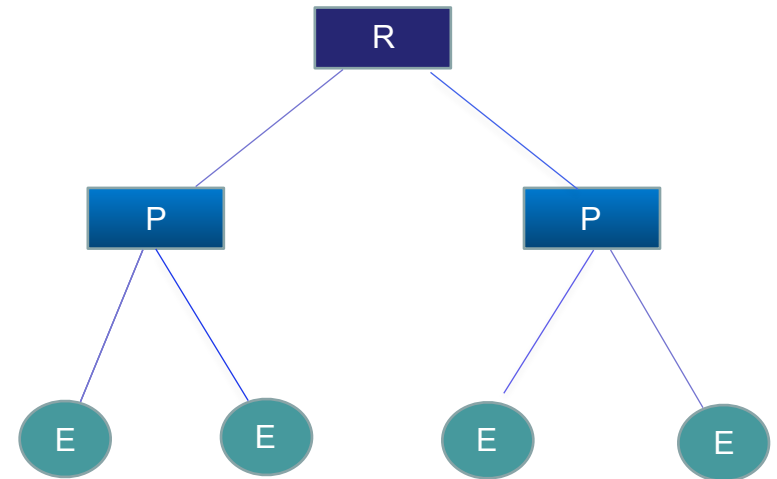
•流程图



组件介绍

Executor

- 注册节点到zookeeper上
- 写资源到redis上
- 启动jetty server供用户查看任务详情
- 执行client端提交的任务
- 更新任务状态
- 任务告警
- 释放资源



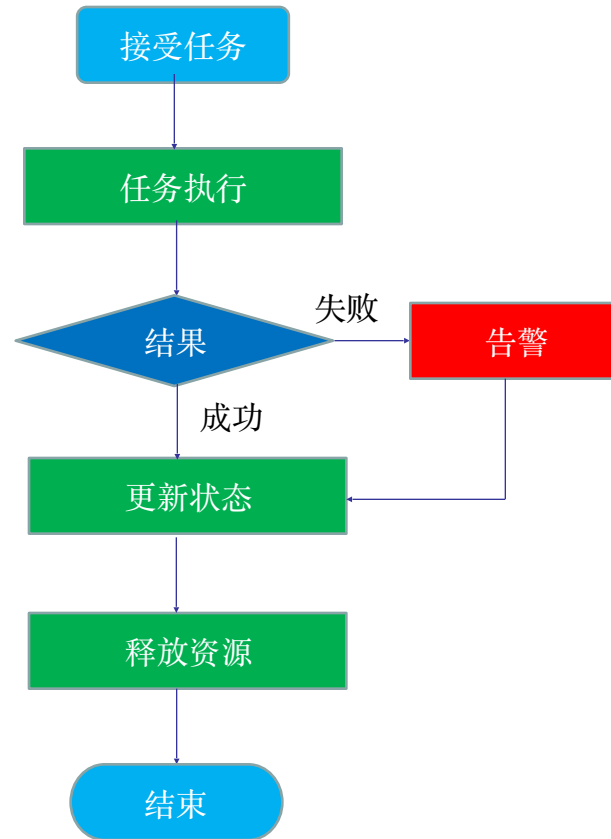
资源在zookeeper中结构

R:root ----- P:pool ----- E:executor

组件介绍

Executor

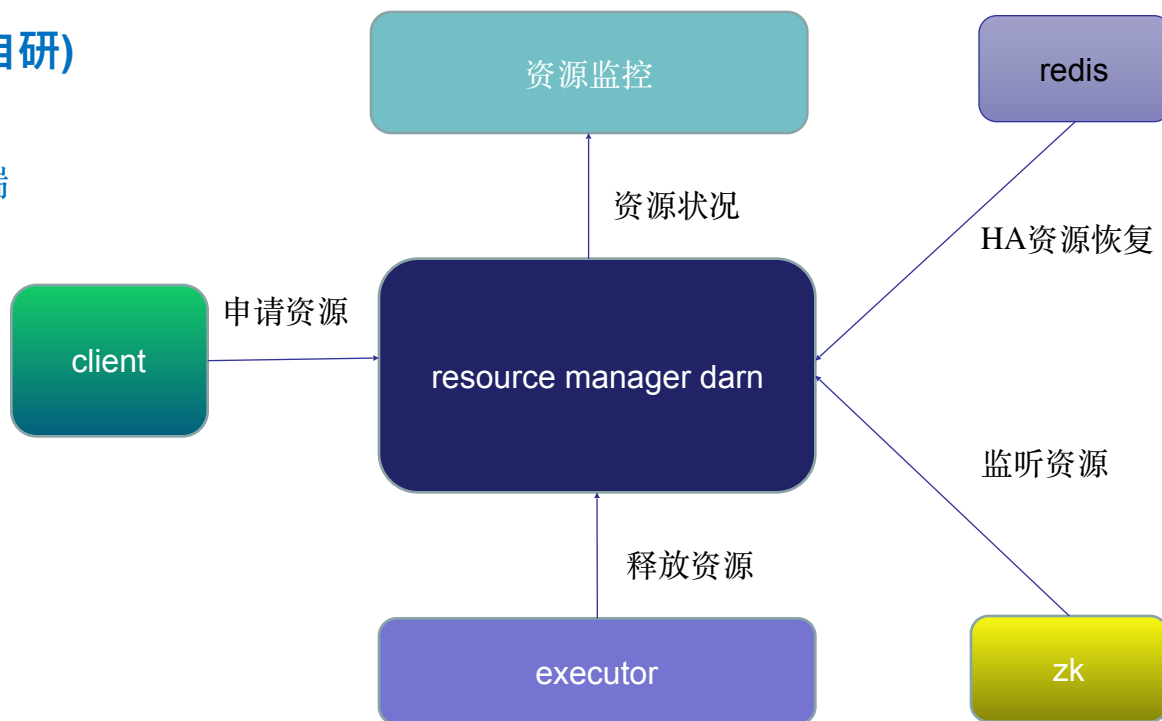
•流程图



组件介绍

Resource manager darn (自研)

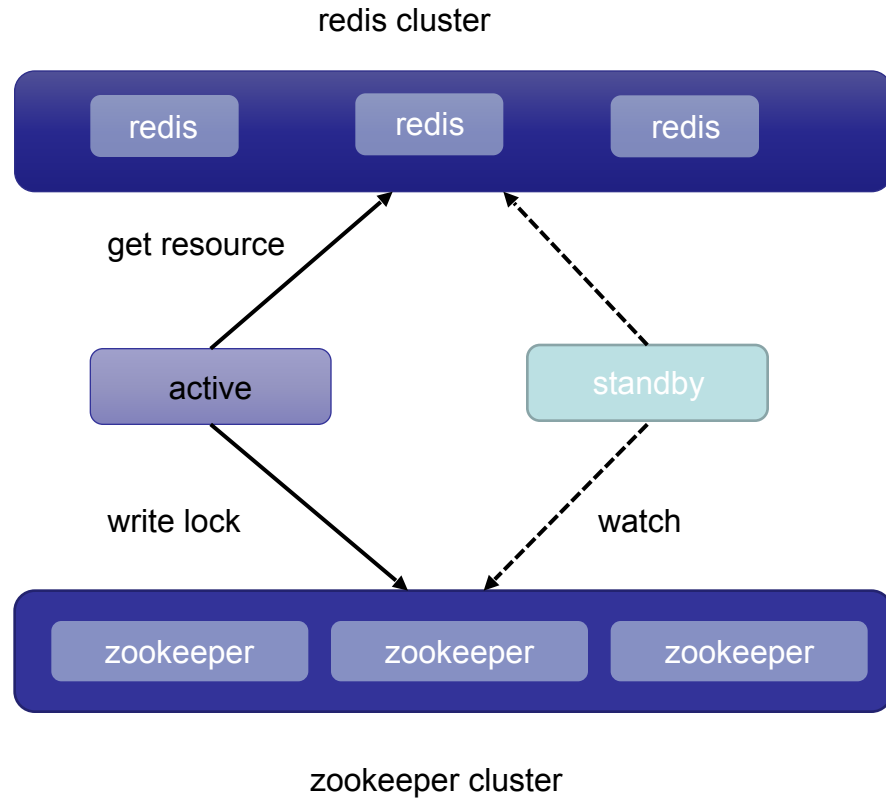
- 监听节点变化及时更新资源列表
- 根据策略提供最优机器给client端
- 对资源进行管理
- 暴露资源metric信息给用户



组件介绍

Resource manager(darn)

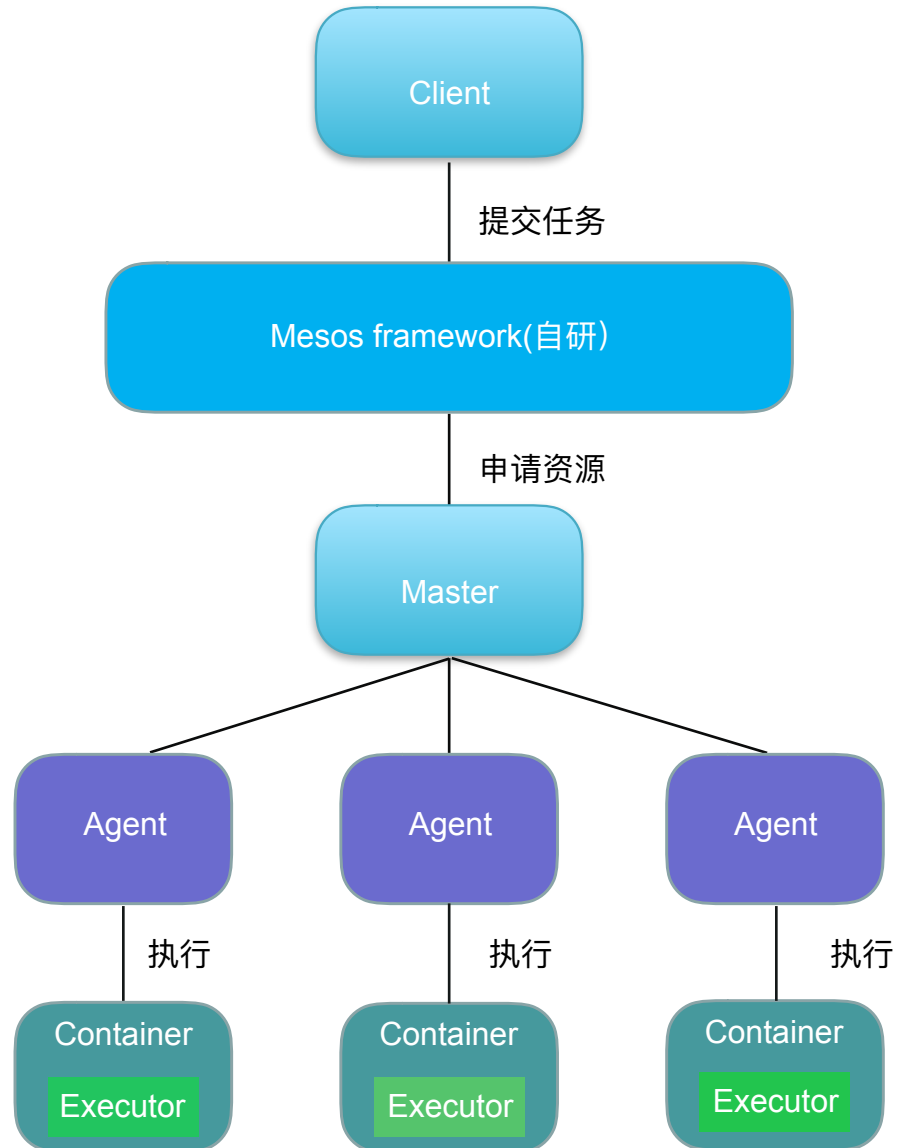
- HA



组件介绍

Resource manager mesos

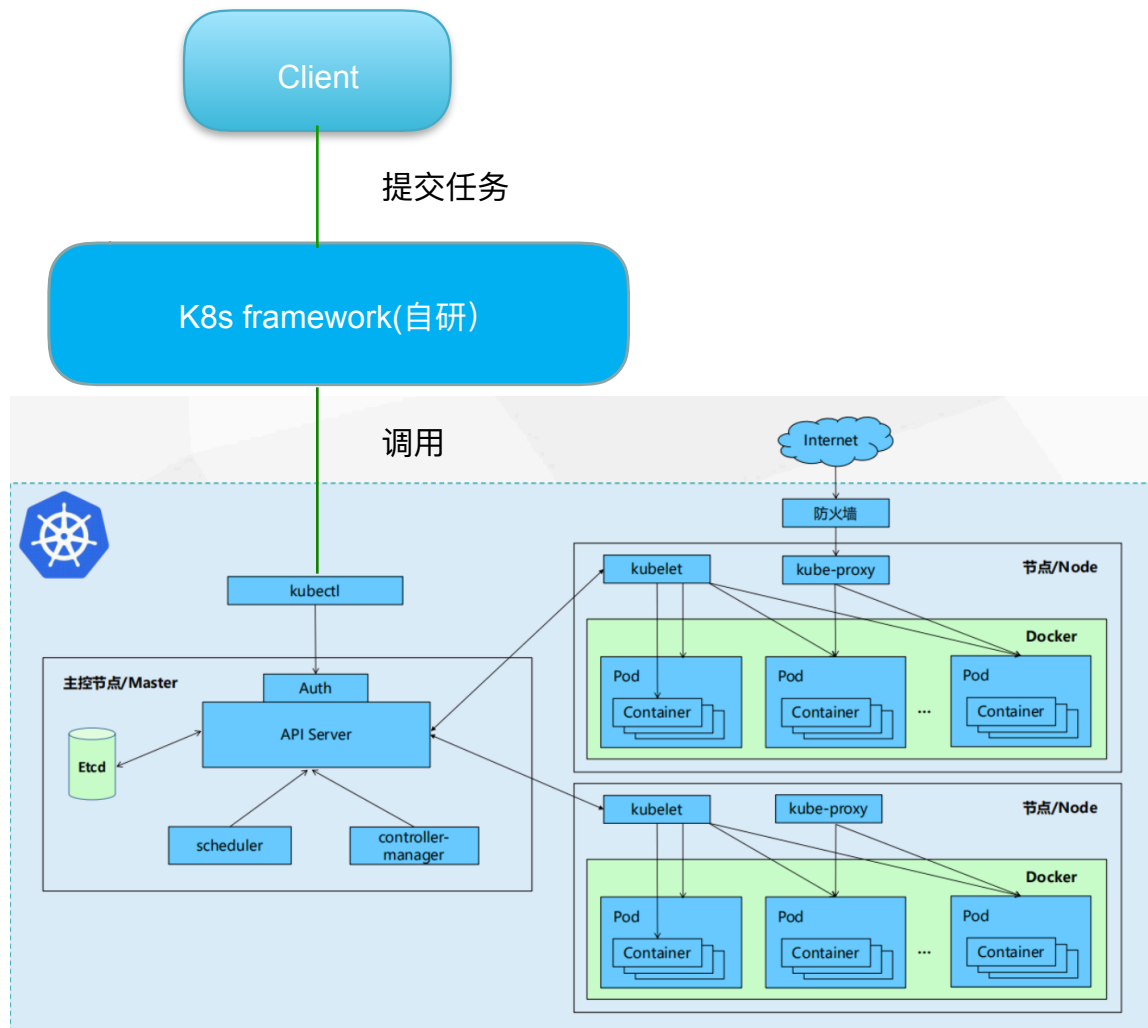
- 资源调度
- 任务执行
- 容器化



组件介绍

Resource manager k8s

- 资源调度
- 任务执行
- 容器化



系统使用

1.创建任务

创建任务

任务名称*	<input type="text"/>	任务描述	<input type="text"/>
任务执行时间	<input type="text"/>	任务资源池*	<div>请选择 ▼</div>
任务重试次数	<input type="text"/>	任务执行策略	<div>请选择 ▼</div>
任务类型*	<div>请选择 ▼</div>	告警邮件	<input type="text"/>
任务资源	<input type="text"/>	任务负责人*	<input type="text"/>
任务命令*	<div></div>		
<div>提交 取消</div>			

系统使用

2.上传依赖

Job Archive

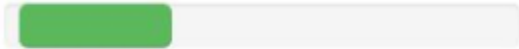


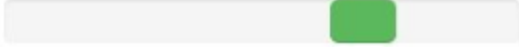
选取文件 未选择文件

Cancel

Upload

系统使用

3.任务列表

Name	Type	Timeline	Start Time	End Time	Elapsed	Status	Details
o2o_get_file_ftp1	command		2016-01-20 10:44 52s	2016-01-20 10:44 52s	0 sec	Success	Details
o2o_get_file_ftp2	command		2016-01-20 10:44 52s	2016-01-20 10:44 52s	0 sec	Success	Details
o2o_up_2_hdfs	command		2016-01-20 10:44 52s	2016-01-20 10:44 52s	0 sec	Success	Details
o2o_clean_data	command		2016-01-20 10:44 52s	2016-01-20 10:44 52s	0 sec	Success	Details