目录

[方案说明 2](#_Toc471981209)

[传统方案 2](#_Toc471981210)

[优化一：docker运行master和slave 2](#_Toc471981211)

[优化二：kubernetest管理slave 3](#_Toc471981212)

[附录1 当时的工作思路 5](#_Toc471981213)

[附录二 slave动态连接master 7](#_Toc471981214)

集群方案

朱学涛

## 方案说明

一种Jenkins的可伸缩部署方式。

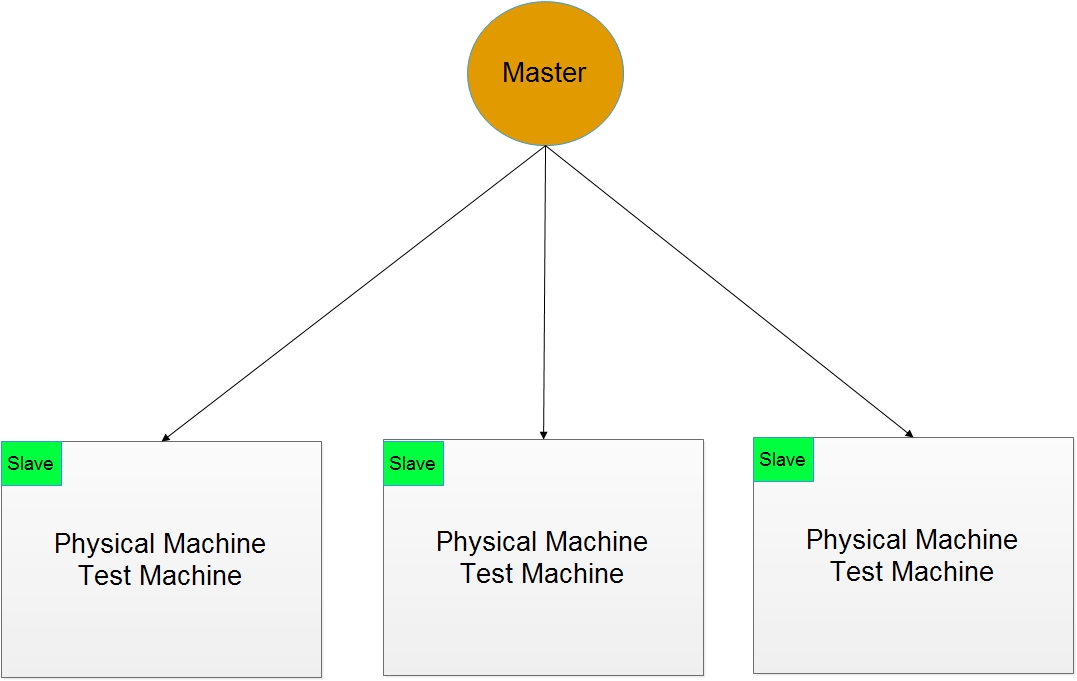
一种是基于swarm的部署方式，另外一种是基于kubernetes的部署方式。两种方式都是以docker为基础，都需要用到docker进行通信和中转。

本方案是对传统方案时进行的优化。

## 传统方案

Jenkins本身是支持集群化的，主要的运行模式为master-slave模式，其中master为整个系统的控制节点，slave节点负责具体的项目编译测试等工作。

当项目增加后，此方案中需要大量的物理节点作为slave，一旦节点遭到破坏，需要大量工作进行人为重建，而且大量的slave节点在线，会使master的管理效率低下，影响Job任务的执行。



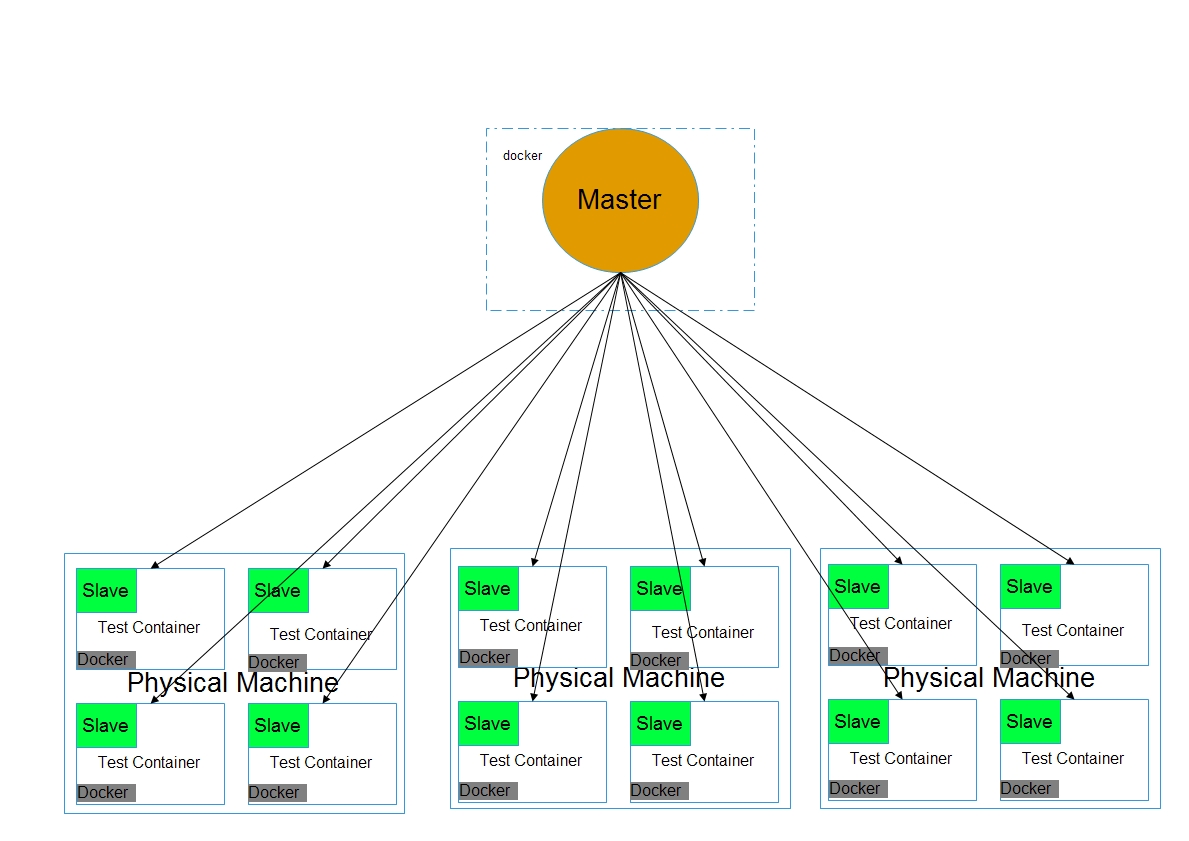
## 优化一：docker运行master和slave

在docker上搭建master和slave，可解决一旦某个节点被破坏，可快速恢复。

master和slave在docker中的安装、运行、配置与物理机上没有区别，这里不再赘述。

实施难点：

1. docker的私有仓库的搭建——Pass团队应该已解决
2. docker image的制作，在master和slave安装配置完成后，最好制作镜像，上传上私有仓库中，以备恢复或添加slave节点时使用



## 优化二：kubernetes管理slave

docker同样可进行集群，使用管理工具kubernetes。通过kubernetes调度slave，可解决上文提到的传统集群方案中存在第二个问题，不再需要大量闲置的slave同时在线，当有任务需要执行时，动态的启动slave，执行完成后再进行销毁。

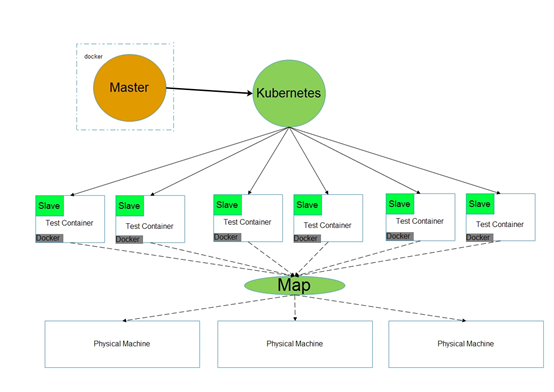
具体实施中，可建立少量slave节点长期在线（最好分布在多个物理机上），以便提高Job任务的响应时间，当任务并发增大时再动态的增slave节点。

实施难点：

1、搭建docker集群——paas团队应该已完成

2、slave节点动态的连接master，附录中将讲到，实现不了这点，将不能动态的增减slave节点

3、jenkins和kubernetes集成，即插件jenkins kubernetes plugin的使用



## 附录1 当时的工作思路

清楚代码库目录结构

1.开发分支 用于每日构建，控制研发过程中的代码质量

2.发布分支 用于考核，ADP的部分考核数据来源

整理出两类Job的配置

1.代码库地址、目录结构

2.开发者名单，项目经理（海尔、外包公司），用于报告的接收

3.maven与ant (java与其它)

4.git与svn

5.jdk版本

6.默认定时每天零晨1点执行

7.pom与sonar

8.build前后是否要执行外部脚本

持续集成功能规则（17Q2之前 java系）

1.下载代码 git svn

2.静态分析 sonar

3.编译 maven

4.单元测试 junit

5.打包 maven

整理sonar规则（整理核心规则，后期逐步扩充）

1.核心规则：无异议的，

2.

jenkins集群搭建

1.第一种方案：jenkins+docker，再由swarm或kubernetes管理docker集群 当有build任务时，由docker启动一个jenkins slave进行执行，完成后，slave被销毁

优点：节省资源，slave节点易维护，扩充时

缺点：部署麻烦，kubernetes或swarm有运维成本

2.第二种方案：jenkins 使用jenkins的slave工具创建slave节点

优点：部署简单，扩充简单，只需要扩充新slave节点

缺点：无任务时，节点空闲

倾向第一种

实施计划

1.master物理机（或docker虚拟机）+几台slave（docker虚拟机）

2.搭建docker私有仓库，制作master和slave的docker image

3.swarm或kubernetes管理docker集群（Jenkins的Docker插件与kubernetes插件）

4.docker-in-docker和pipeline

jenkins与jira的集成

1.以jira的插件形式把jenkins对使用者隐藏

2.查看build历史、触发发布build、查看某次build结果详情、具体配置

## 附录二 slave动态连接master

Jenkins master连接slave有两种模式，一直是master主动发起连接，主要通过ssh来进行连接，另外一种是slave主动连接master，使用的jnlp协议。

在docker中运行jenkins slave两种模式都支持，但此方案中只能用jnlp来进行连接。

使用docker的镜像jenkinsci/jnlp-slave/

详细说明见：<https://hub.docker.com/r/jenkinsci/jnlp-slave/>