# Centos+jenkins+php持续集成

# 借鉴文档

Jenkins+php+phing检查代码打包部署

https://dzone.com/articles/continuous-integration-php

https://testerhome.com/topics/9185

http://shashikantjagtap.net/php-continuous-integration-template-using-composer-jenkinsci/

https://jenkins.io/solutions/php/

https://github.com/SebastianHanke/Jenkins-build-examples

https://www.future-processing.pl/blog/php-code-analysis-on-jenkins/

# 相关插件

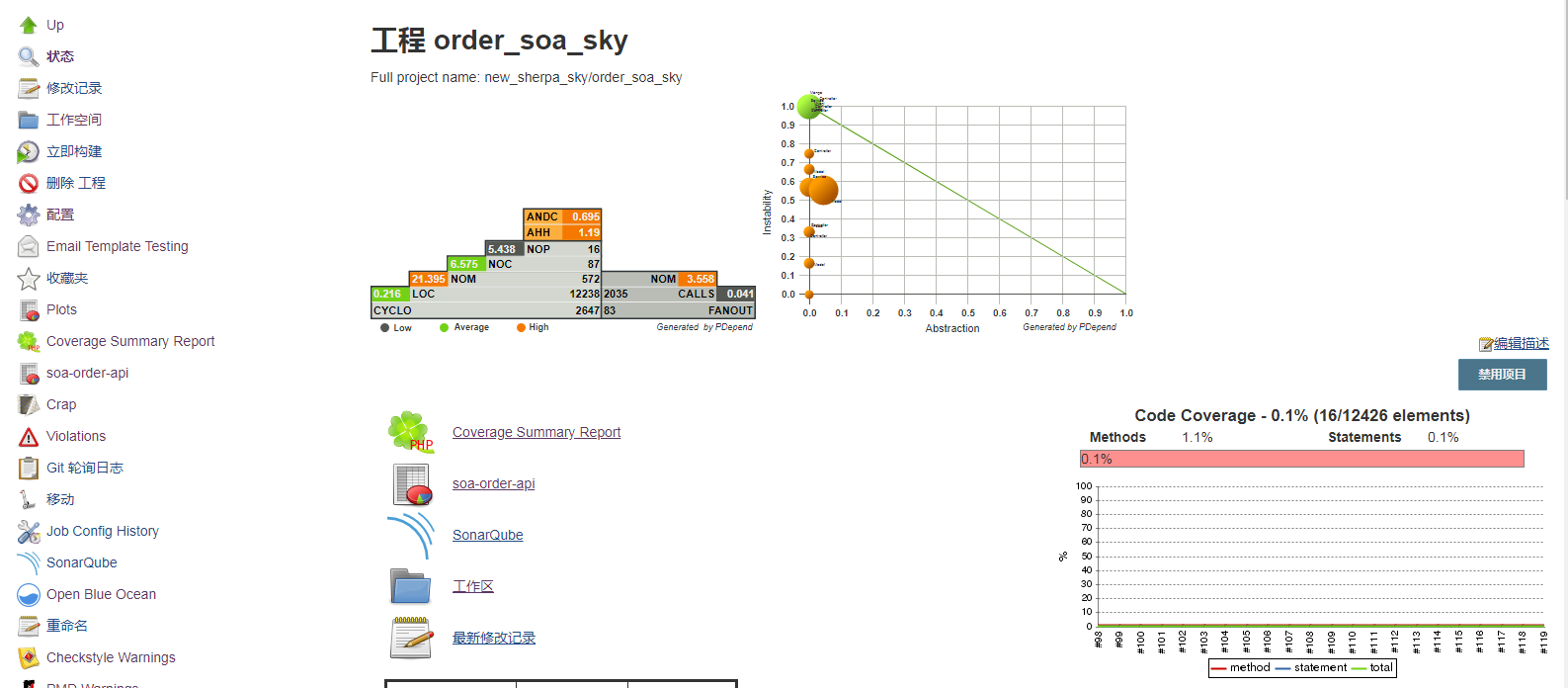
* 1. Git
  2. Phing/Ant:使用Phing或Apache Ant 对PHP项目做自动化构建
  3. CheckStyle:使用PHP CodeSniffer进行代码风格检查的工具。用于检查PHP代码是否有违反一组预先设置好的编码标准的一个PEAR包，内置了ZEND,PEAR的编码风格规则
  4. Clover PHP:使用phpunit进行单元测试的工具，可以被xdebug扩展用来生成代码覆盖率报告，并且可以与phing集成来自动测试
  5. DRY:使用PHPCPD(php copy paste detector)来发现项目中的重复代码
  6. HTML Publisher:用来发布phpdox文档
  7. JDepend:使用PHP Depend分析php中静态代码，用来检查项目中的代码规模和复杂程度
  8. Plot:使用phploc来统计php项目规模大小的工具，可以统计php的项目代码行数
  9. PMD:使用phpmd(php mess dector),对基于pdepend的结果进行分析，一旦项目超过了pdepend中各具体指标的规定，将发出警告信息.
  10. Violations：按照代码缺陷严重性集中显示pwd静态代码分析的结果
  11. xUnit:使用JUnit的格式来输出phpunit的日志文件
  12. crap4j
  13. publish over ssh 推送代码包
  14. warnings
  15. greenballs

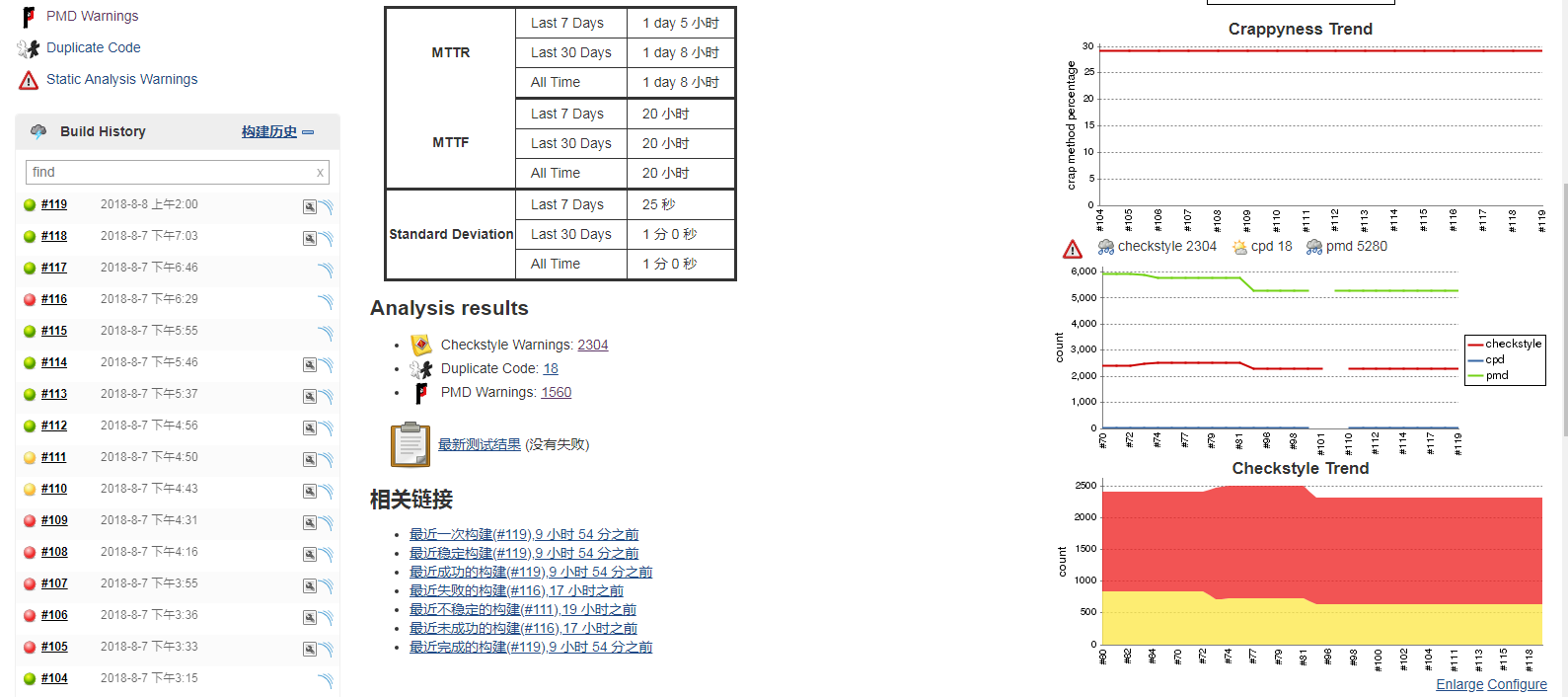
php相关

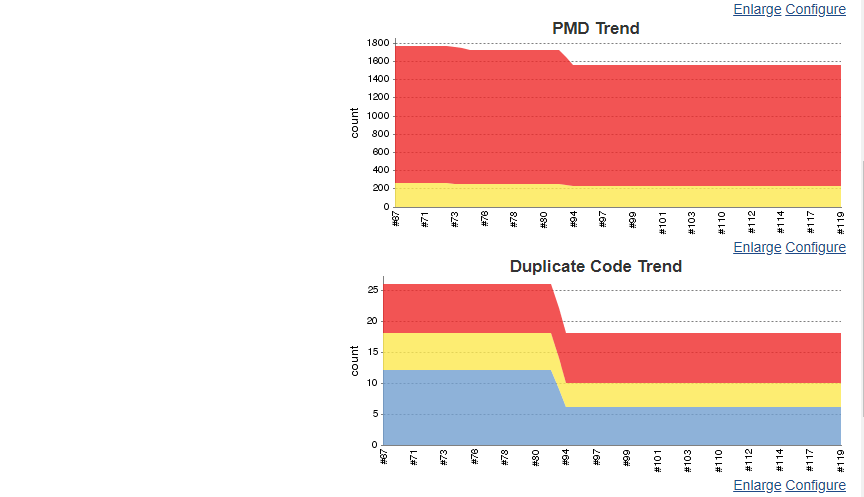
**PHPUnit** php测试框架   
**PHP\_CodeSniffer** 代码格式检测   
**PHPLOC** 统计php的项目代码行数   
**PHP\_Depend** 它可以用来检查你的PHP项目中的代码规模和复杂程度   
**PHPMD** （PHP Mess Detector）能够探测PHP源代码中一些潜在的问题。如：可能存在的Bug；未达最佳标准的代码；过于复杂的语法；从未使用过的参数、方法、属性。   
**PHPCPD**（Copy/Paste Detector）PHPCPD是一个在代码中寻找类似模式的工具，使用它是为了在代码库中识别代码在何处被复制或粘贴。这是常规构建过程中一个非常有用的工具，它会帮助程序员分析代码，以避免在代码库中重复调用函数。   
**phpDox**一个PHP项目的文档生成器（不限于API文档）

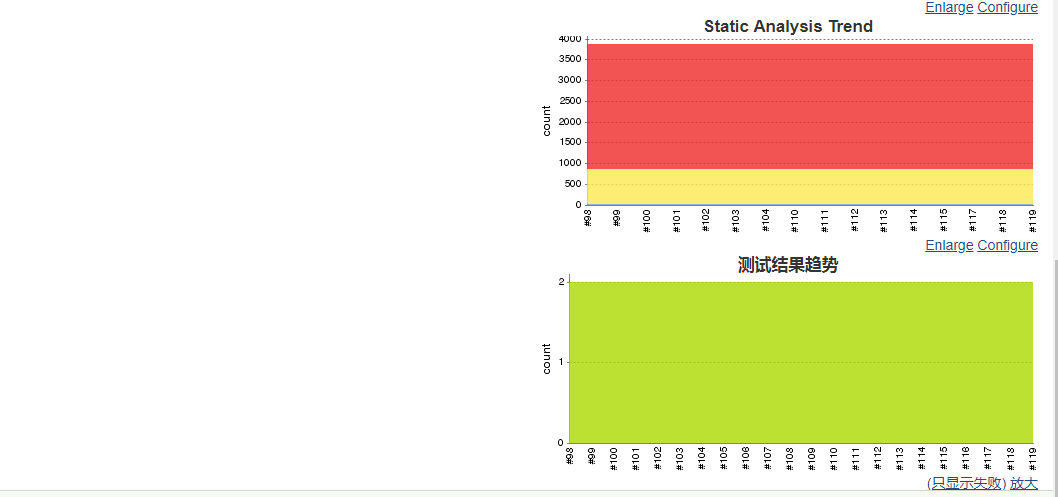
# 详细配置

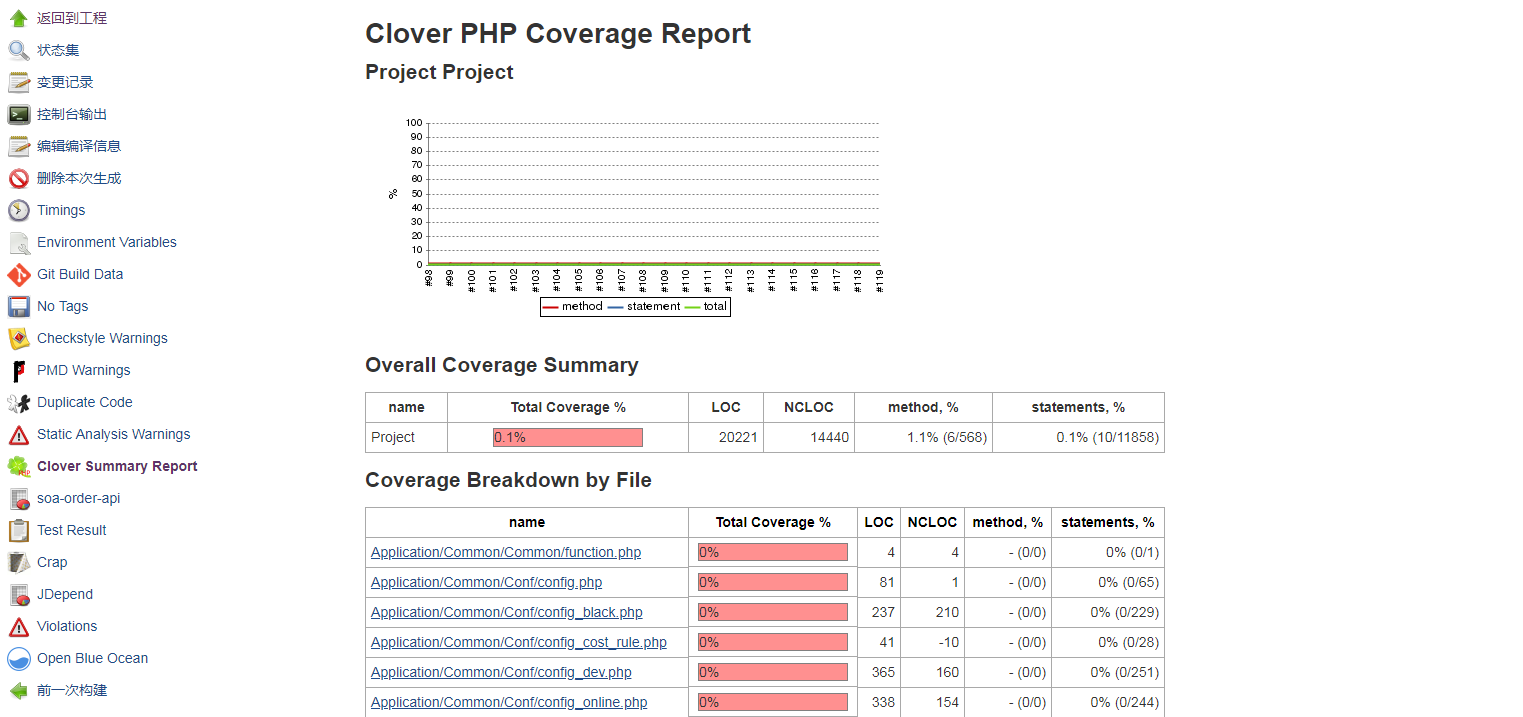
## 最终结果

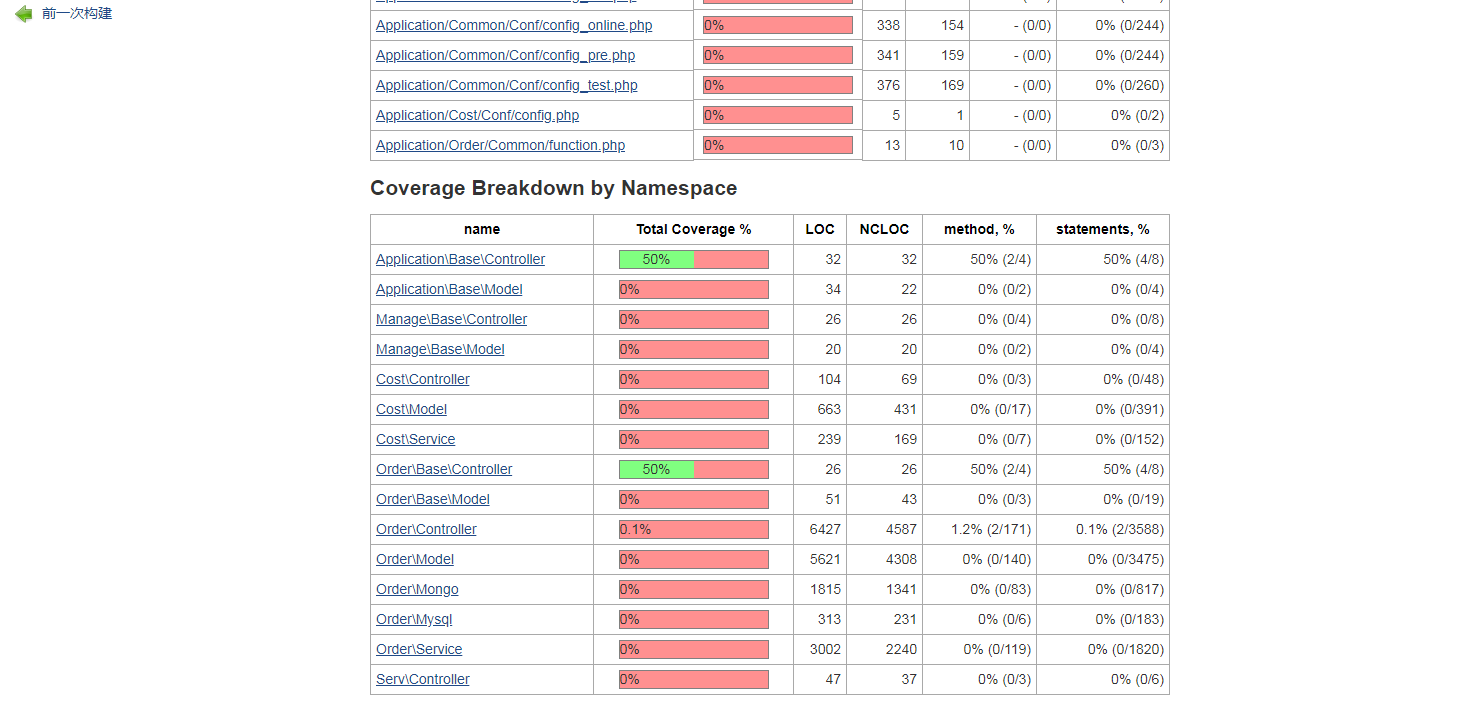


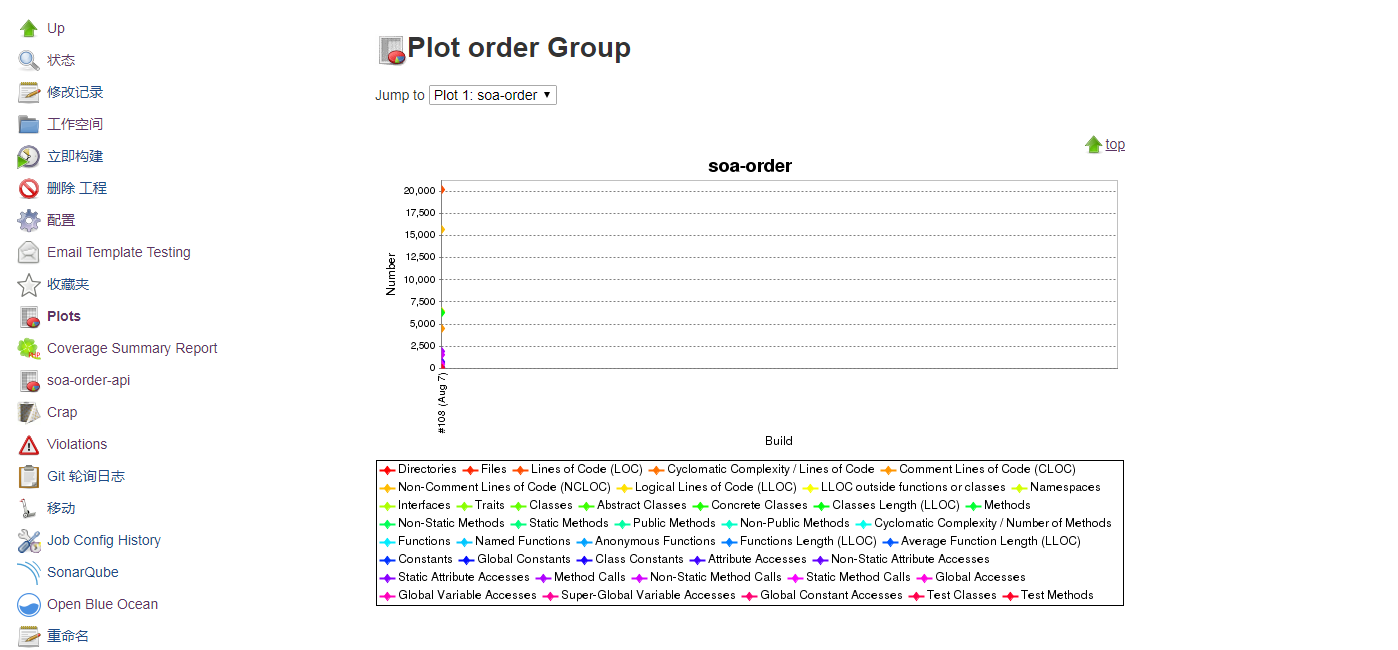


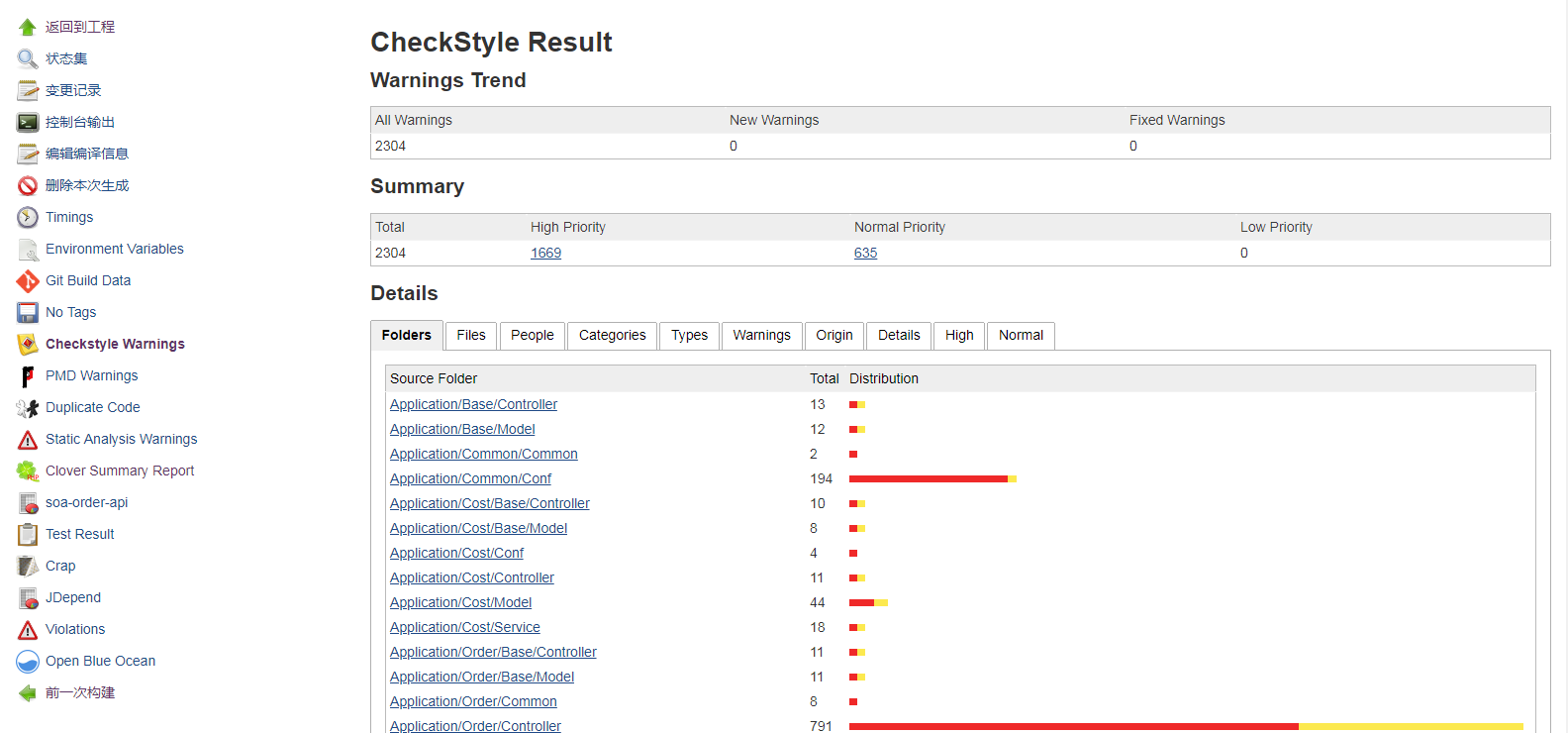


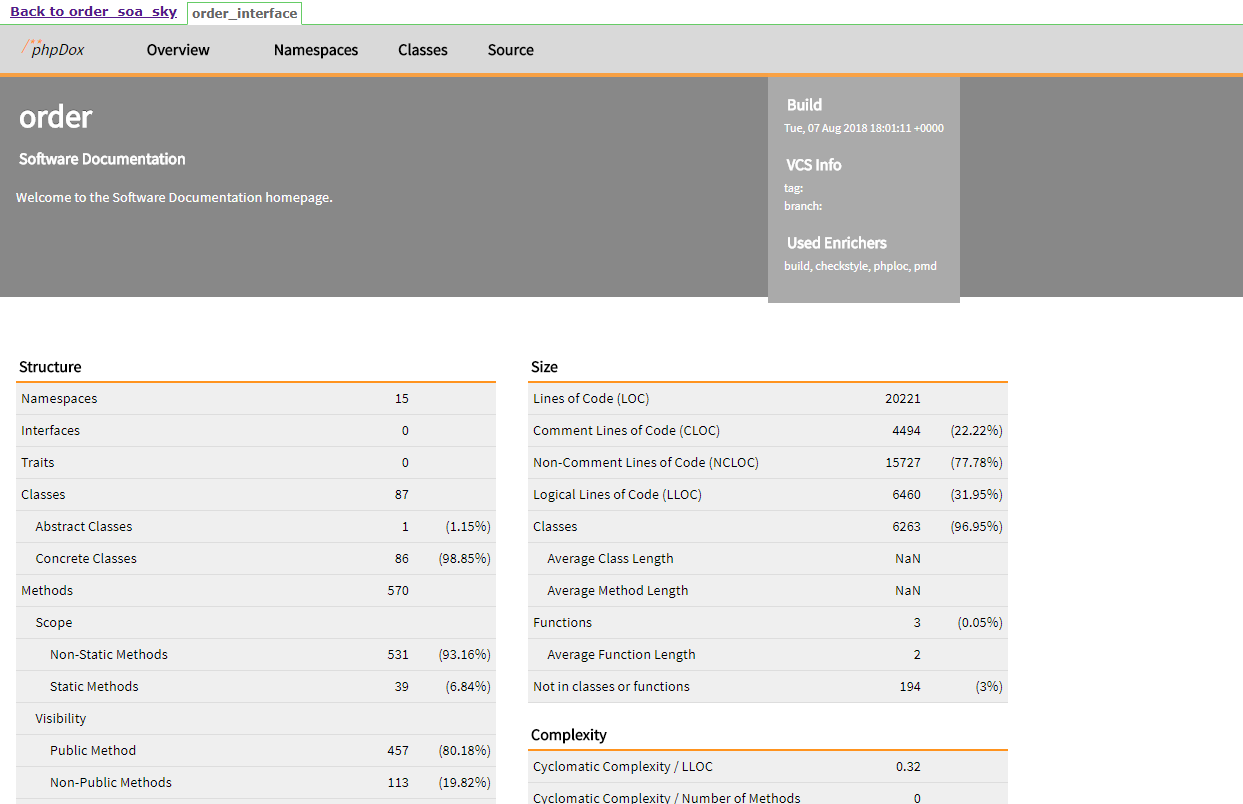






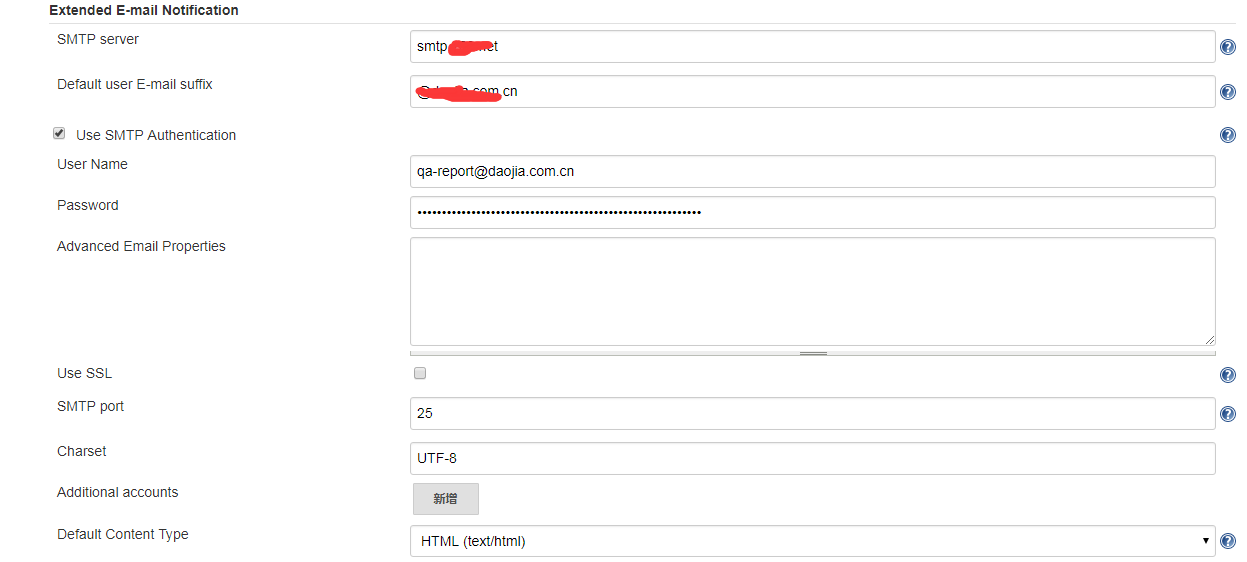






## 详细配置

### 系统管理->配置Extended E-mail Notification





<hr/>

(本邮件是程序自动下发的，请勿回复！)<br/><hr/>

项目名称：$PROJECT\_NAME<br/><hr/>

构建编号：$BUILD\_NUMBER<br/><hr/>

git版本号：${GIT\_REVISION}<br/><hr/>

构建状态：$BUILD\_STATUS<br/><hr/>

触发原因：${CAUSE}<br/><hr/>

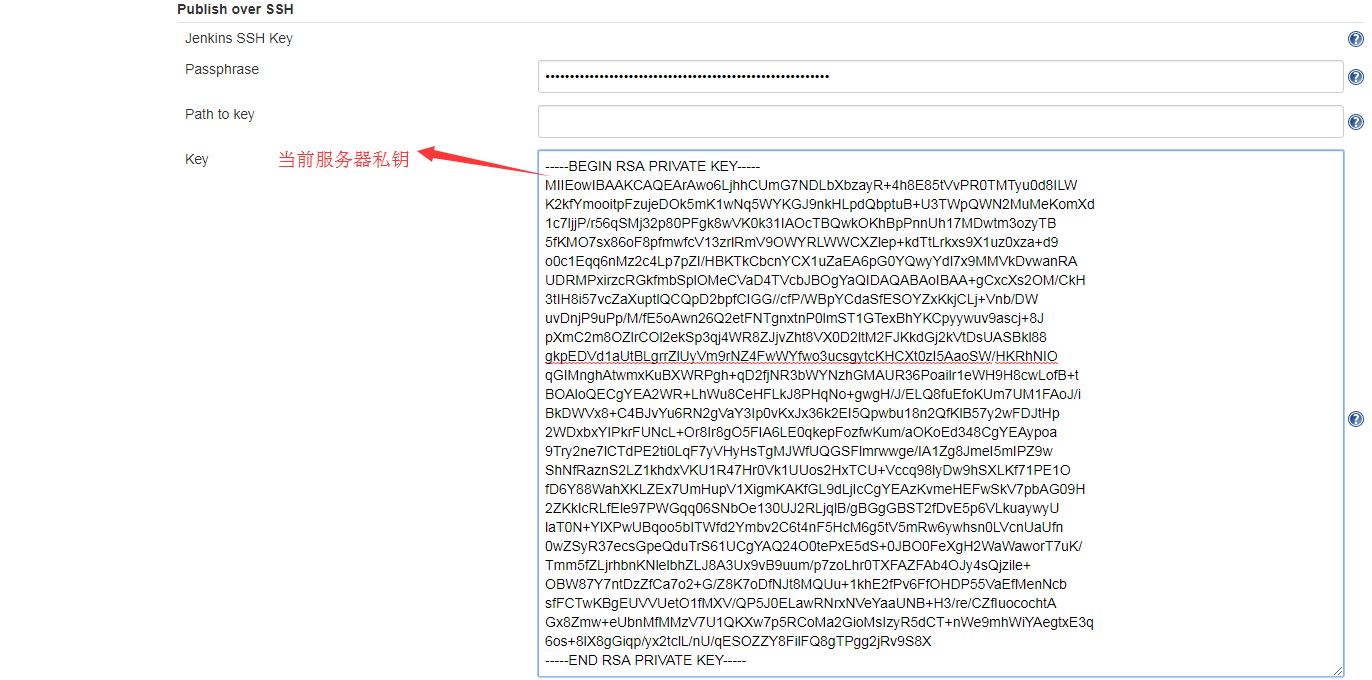
构建日志地址：${BUILD\_URL}console

构建地址：$BUILD\_URL<br/><hr/>

失败单测：${FAILED\_TESTS} <br/><hr/>

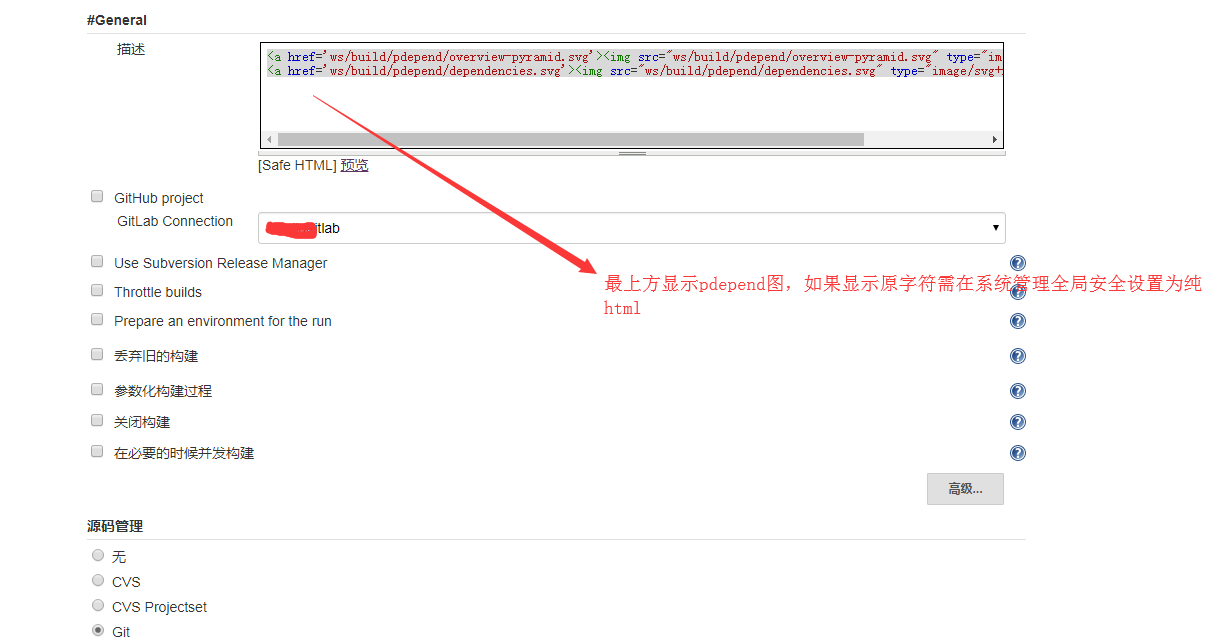
变更集:${JELLY\_SCRIPT,template="html"}<br/><hr/>

### 系统管理->配置publish over ssh



### 

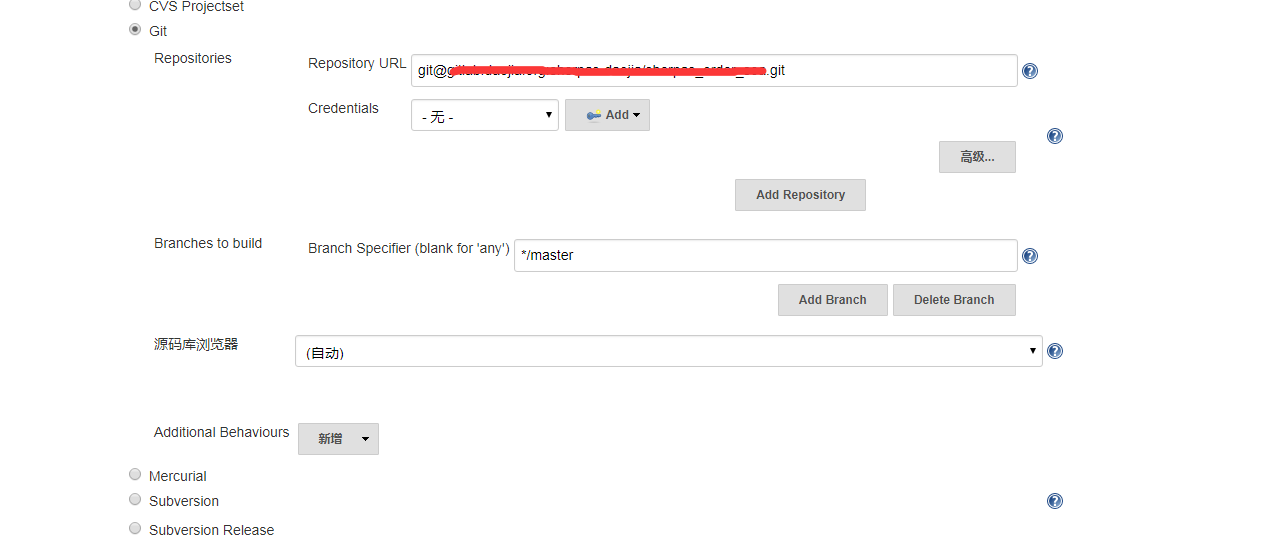
### 项目配置

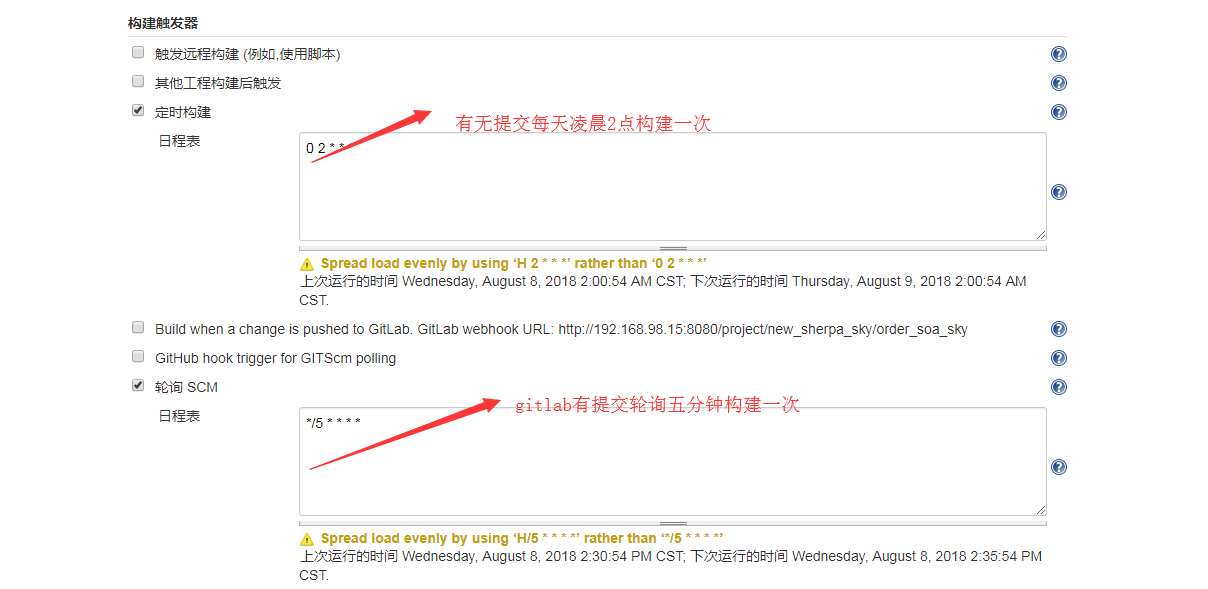


<a href='ws/build/pdepend/overview-pyramid.svg'><img src="ws/build/pdepend/overview-pyramid.svg" type="image/svg+xml" /></a>

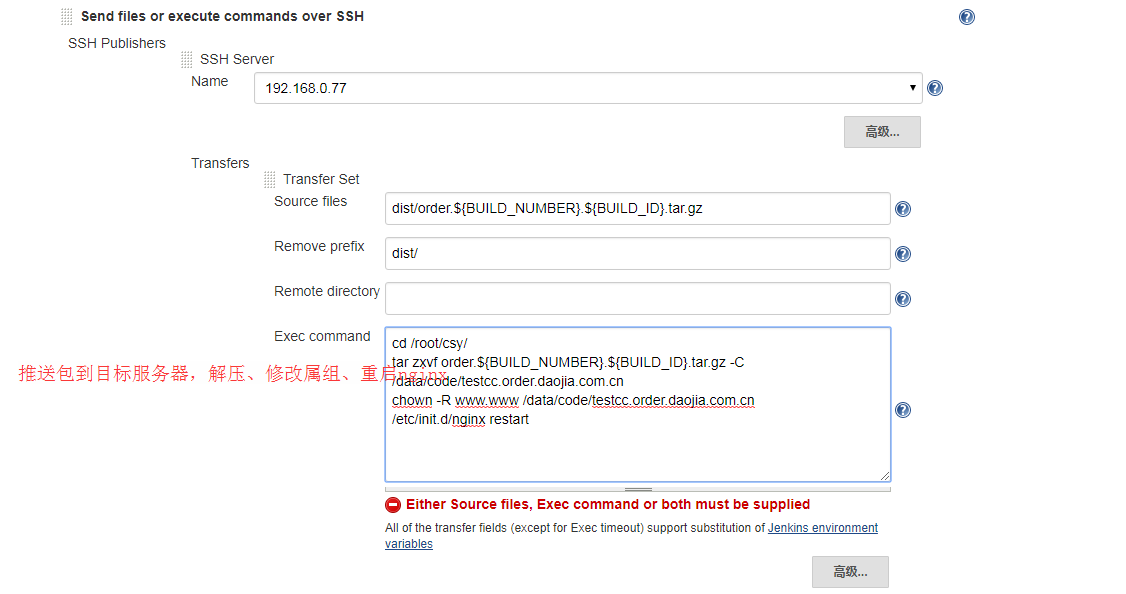
<a href='ws/build/pdepend/dependencies.svg'><img src="ws/build/pdepend/dependencies.svg" type="image/svg+xml" /></a>

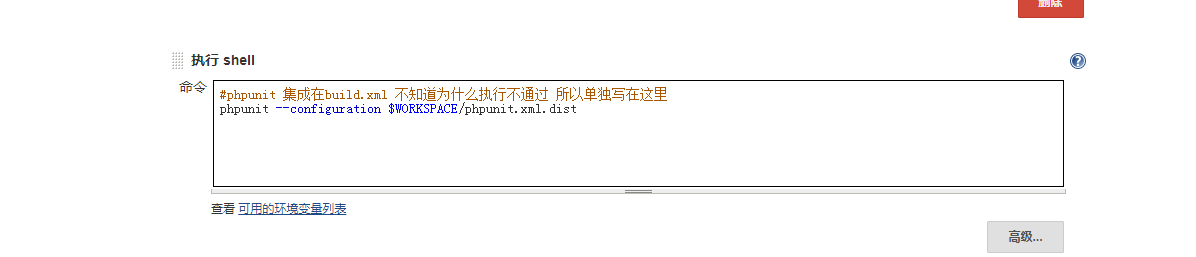
Git配置



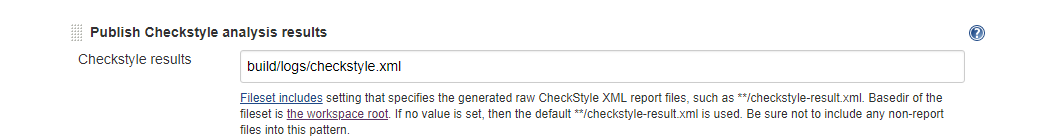


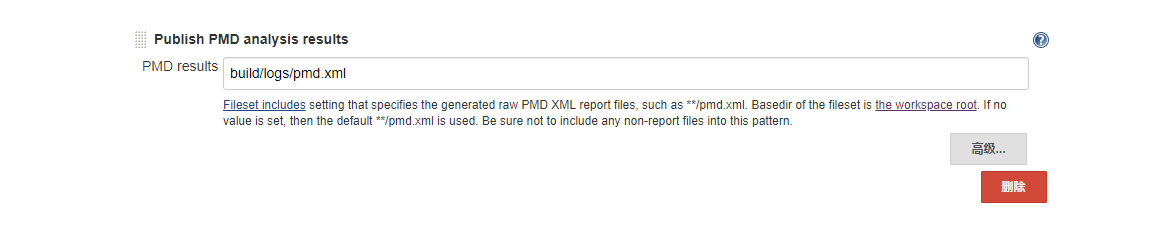


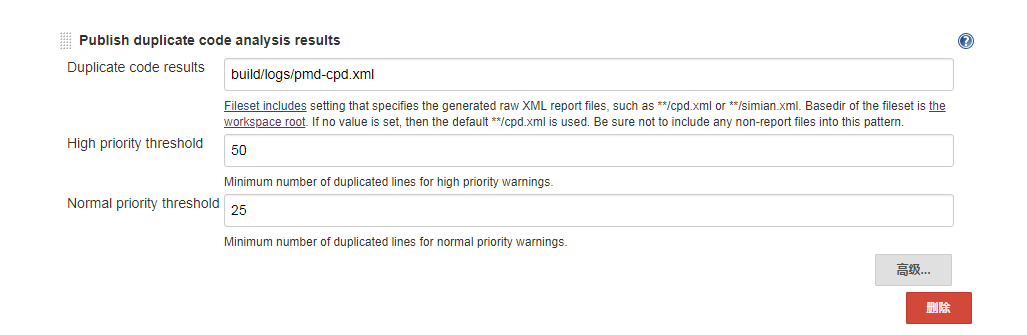




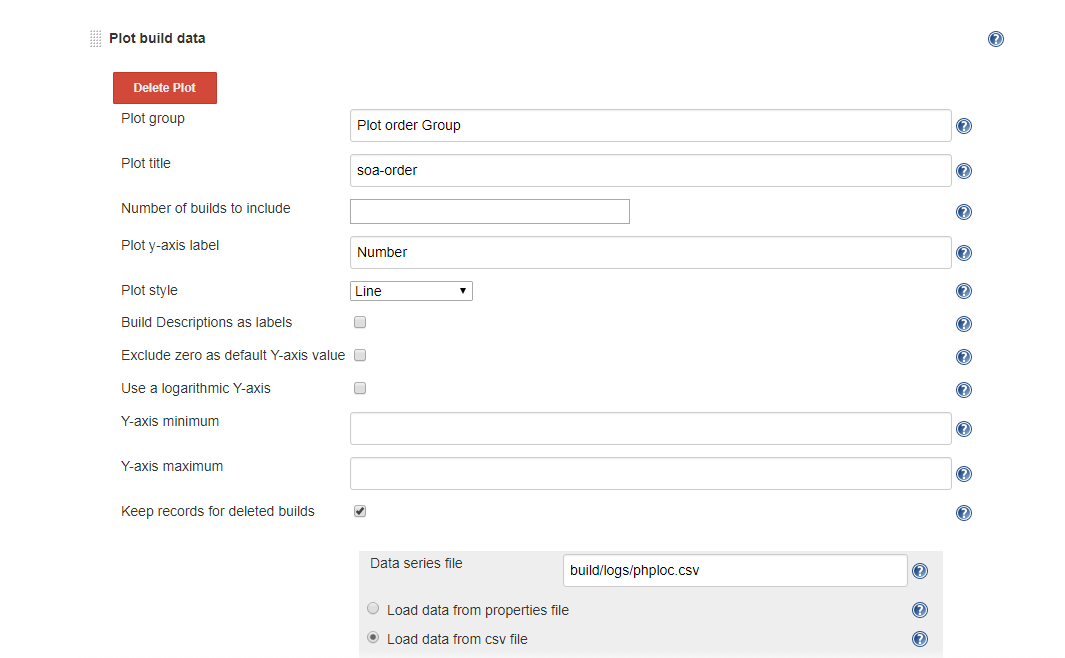
构建后操作

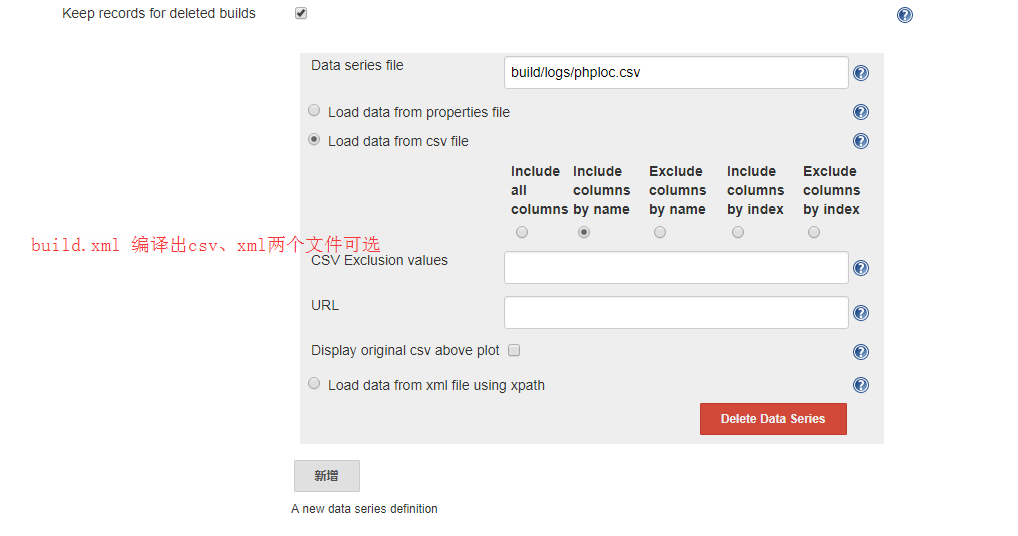


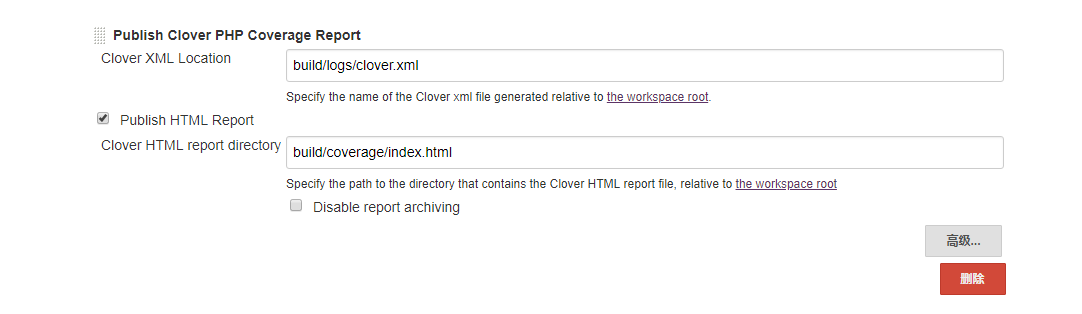


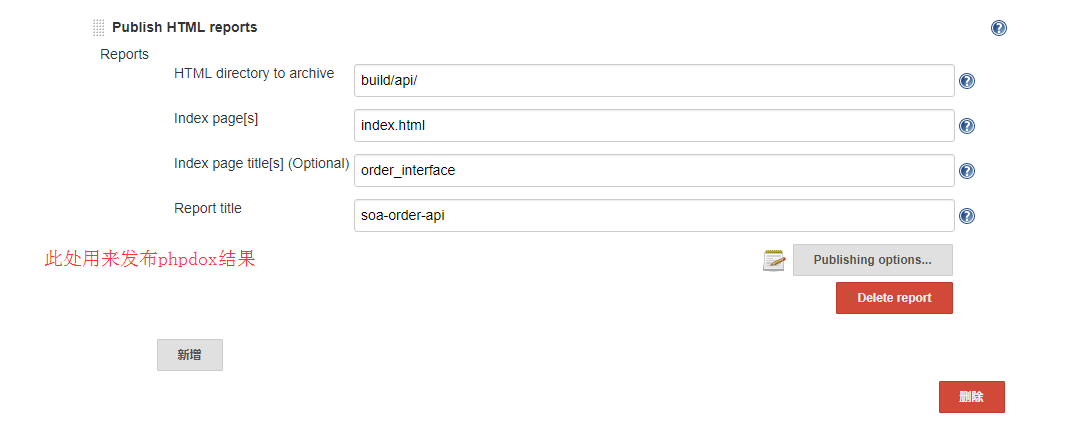


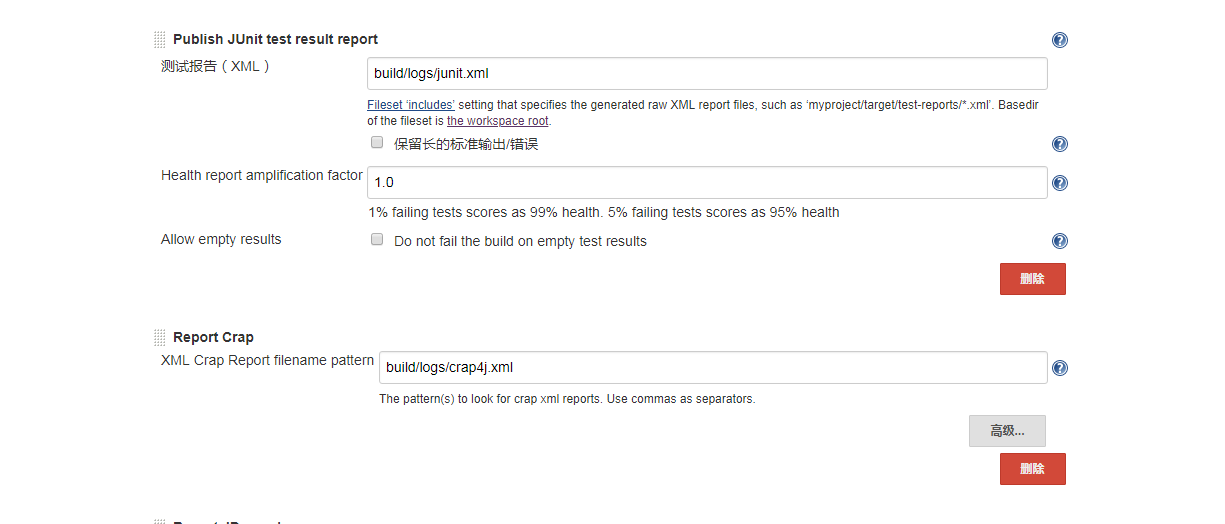


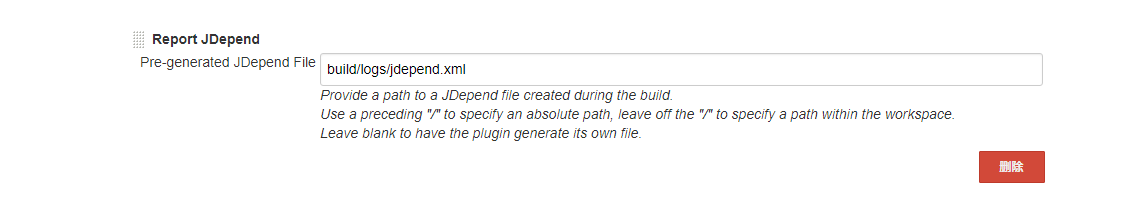


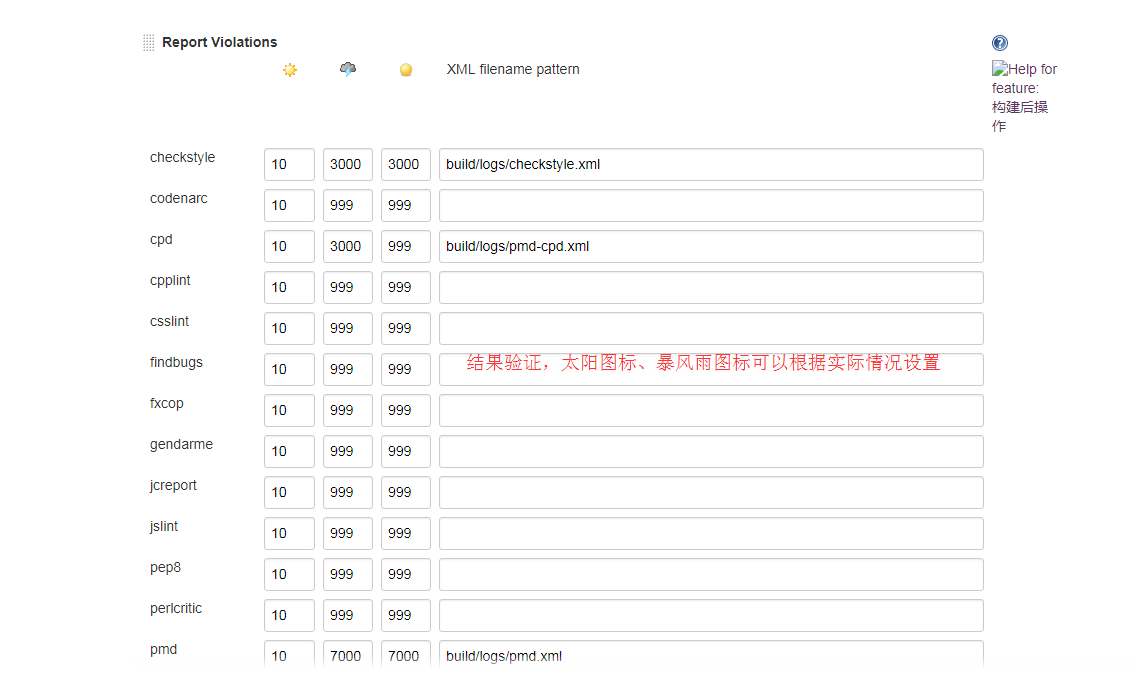


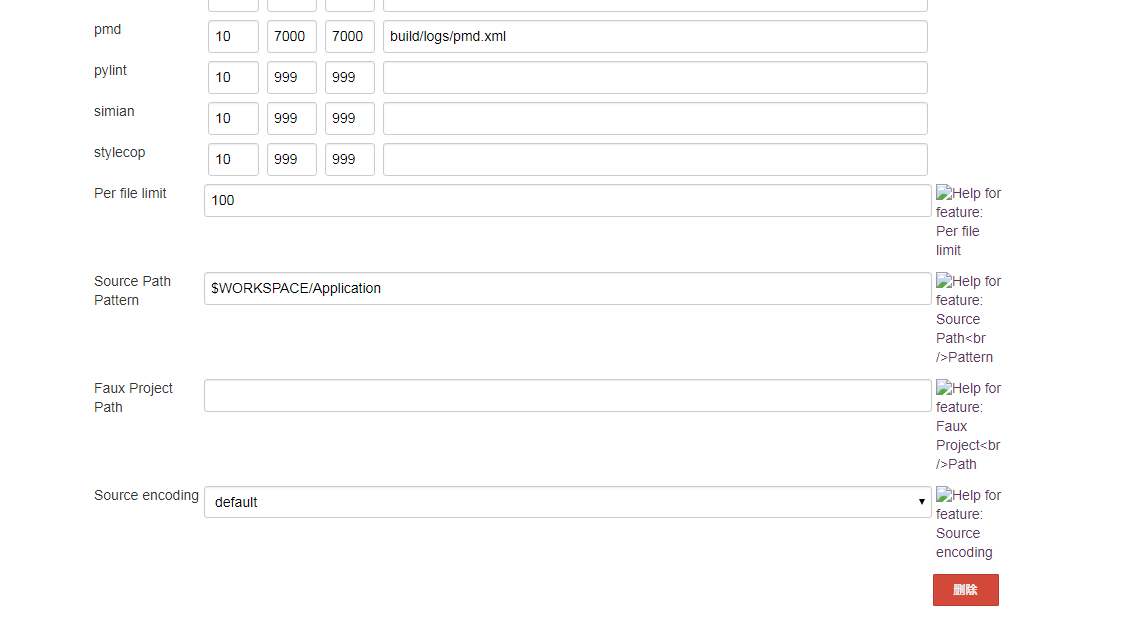




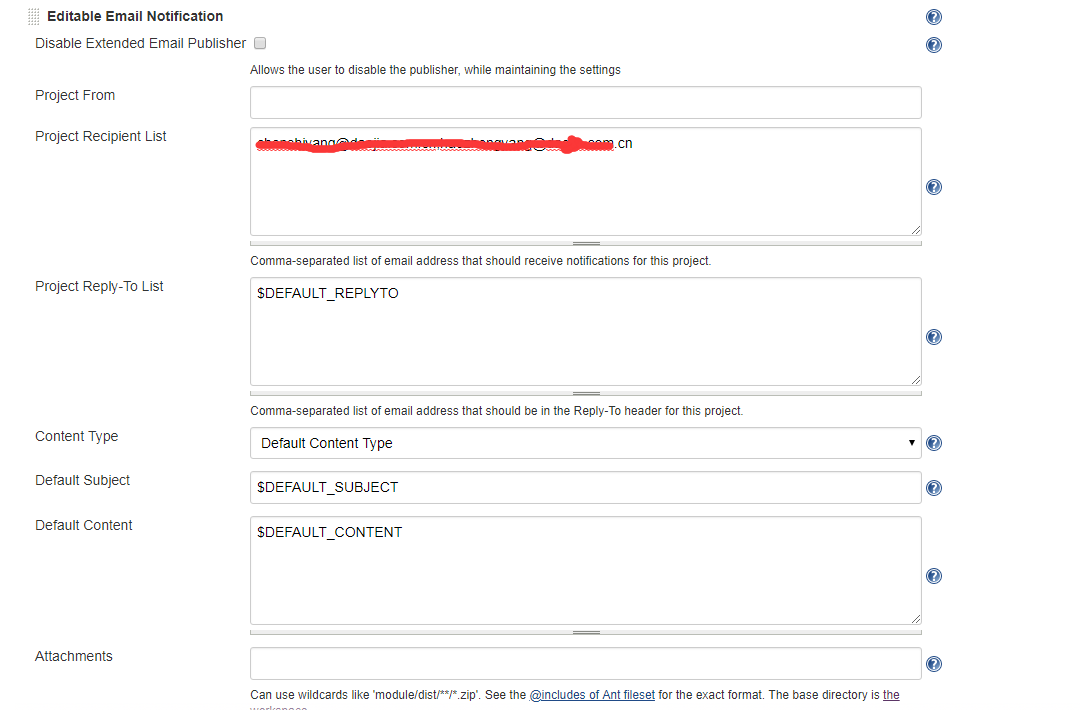


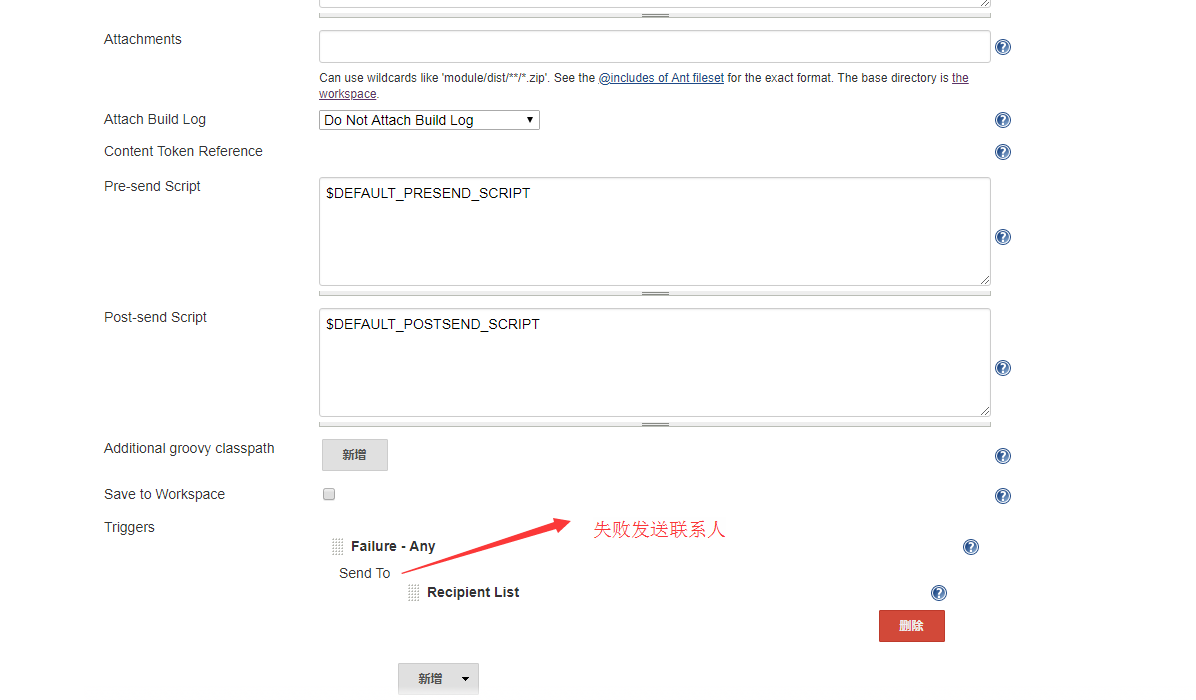






邮件配置





# 错误解决

如果配置发布phpdox 网页有报错

要解决该问题，方式也比较简单，就是修改Content Security Policy的默认配置。

修改方式为，进入Manage Jenkins->Script console，输入如下命令并进行执行。

System.setProperty("hudson.model.DirectoryBrowserSupport.CSP", "")

# 图表解释

## Depend



大小和复杂性

NOP 软件包

NOC 类

NOM 方法

LOC 有效代码行数

[CYCLO](https://pdepend.org/documentation/software-metrics/cyclomatic-complexity.html) 代码复杂度

ANDC派生类平均值；比如有10个类，若  ANDC 值为0.5，意味着平均每两个类，派生自另一个类

AHH继承层次(深度)平均值；比如有10个类，若 AHH 值为1，可以理解为，5个类继承于另外5个类；或是5个类继承于一个基类

耦合

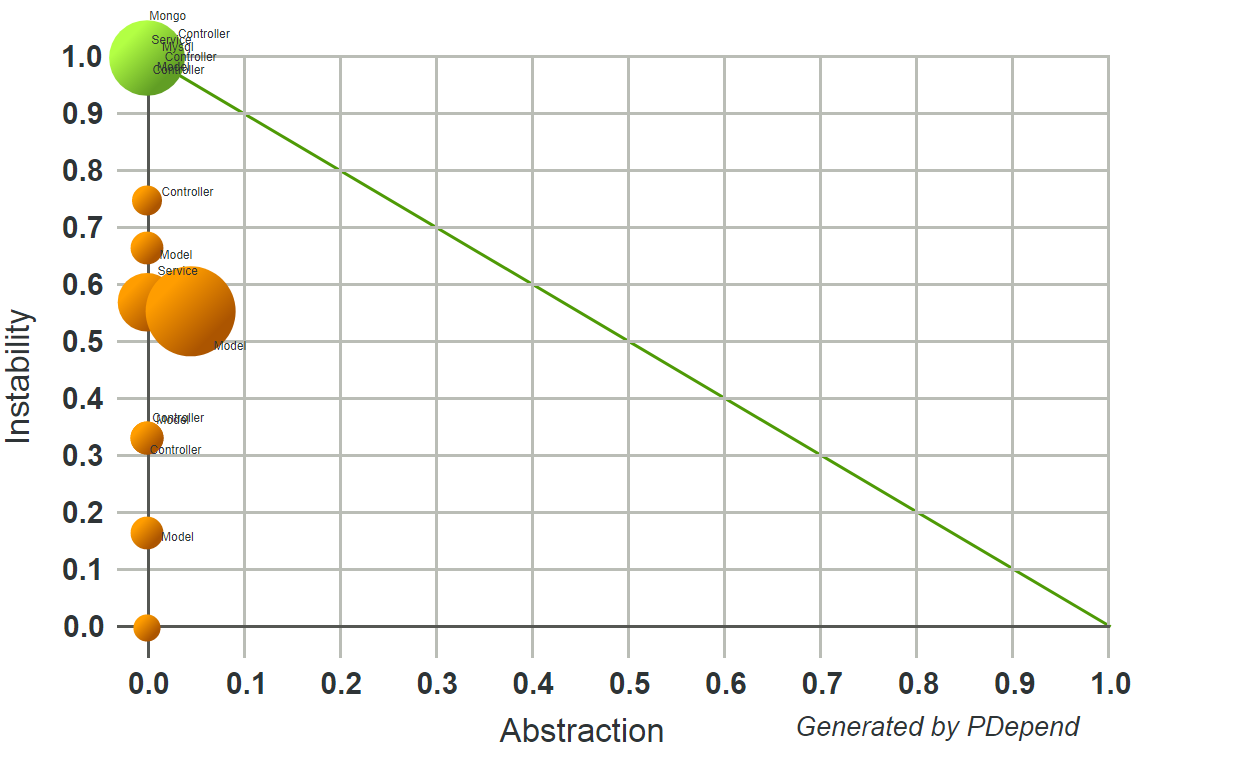
CALLS 调用次数、此度量标准计算不同的函数和方法调用的数量。区别意味着函数或方法体内的同一个方法调用只计算一次

FANOUT该扇出提供了关于类和接口引用类型的信息。它只计算那些不属于同一继承分支的类型

* NOC/NOP：值偏高，说明对于 classes 数量而言，packages 偏少。
* LOC/NOM：值偏高，说明对于代码行数而言，methods 偏少

参考值

| **Metric** | **Low** | **Average** | **High** |
| --- | --- | --- | --- |
| CYCLO/LOC | 0.16 | 0.20 | 0.24 |
| LOC/NOM | 7 | 10 | 13 |
| NOM/NOC | 4 | 7 | 10 |
| NOC/NOP | 6 | 17 | 26 |
| CALLS/NOM | 2.01 | 2.62 | 3.2 |
| FANOUT/CALLS | 0.56 | 0.62 | 0.68 |
| ANDC | 0.25 | 0.41 | 0.57 |
| AHH | 0.09 | 0.21 | 0.32 |



Y轴表示不稳定性，x轴表示抽象性

Ca - Afferent Couplings:指一个 package 内的抽象类或接口。若变化会影响其他模块

Ce - Efferent Couplings:被该 package 依赖的外部 package 的数目。即，被其他模块影响

Instability：ce/ca+ce ，值在0至1之间，指数越小，越稳定

Abstractness：ac/ac+cc ，值在0至1之间，若值为1，说明包中只有抽象类和接口

越靠近绿色斜线，Instability 与 Abstractness 平衡程度越好

详见：https://pdepend.org/documentation/handbook/reports/abstraction-instability-chart.html

## Duplicate Code

重复代码

## PMD warning

PMD是一种开源分析Java代码的工具，通过静态分析获知代码错误，该工具可以做到检查Java代码中是否含有未使用的变量、是否含有空的抓取块、是否含有不必要的对象等等

## Crap

Java检查代码复杂度、垃圾代码等