

# 玩转《阿里巴巴开发手册》 P3C插件-云效在线扫描

胜燕 阿里高级开发工程师

## 个人介绍



胜燕(龚胜芳)

阿里巴巴高级开发工程师

2014年加入阿里巴巴，有着丰富的工具平台开发经验，现负责云效公有云测试服务及阿里JAVA规约插件等相关工作。

# 目录

Contents

- 1 怎么让团队代码规范统一？
- 2 真实案例
- 3 在线扫描演示
- 4 快速体验&FAQ

1

# 01 怎么让团队代码规范统一？

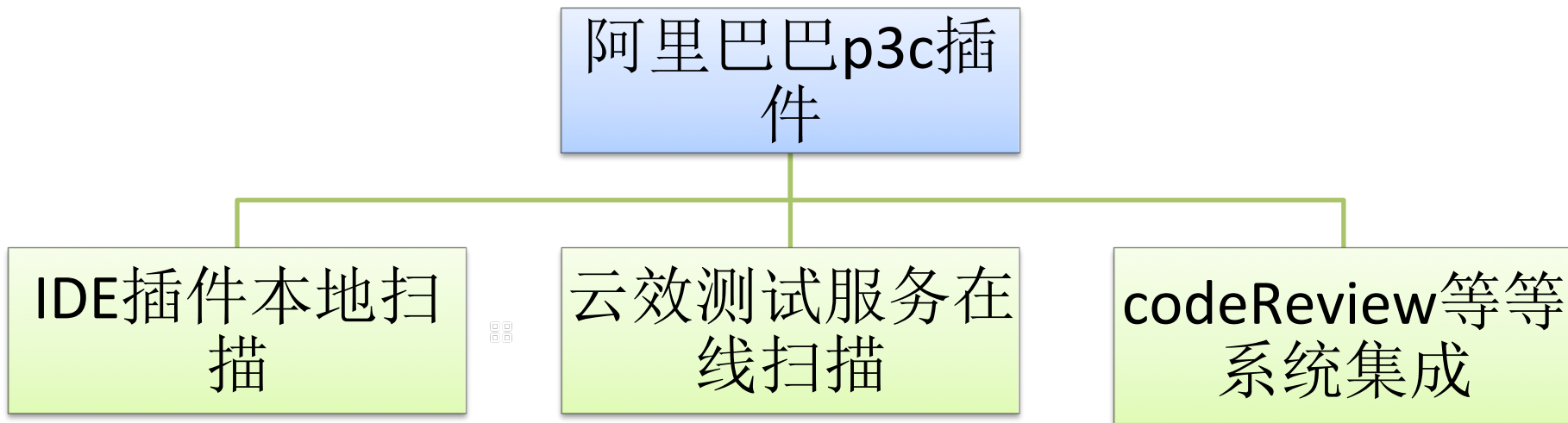
*Part one*

## 怎么让团队代码规范统一？

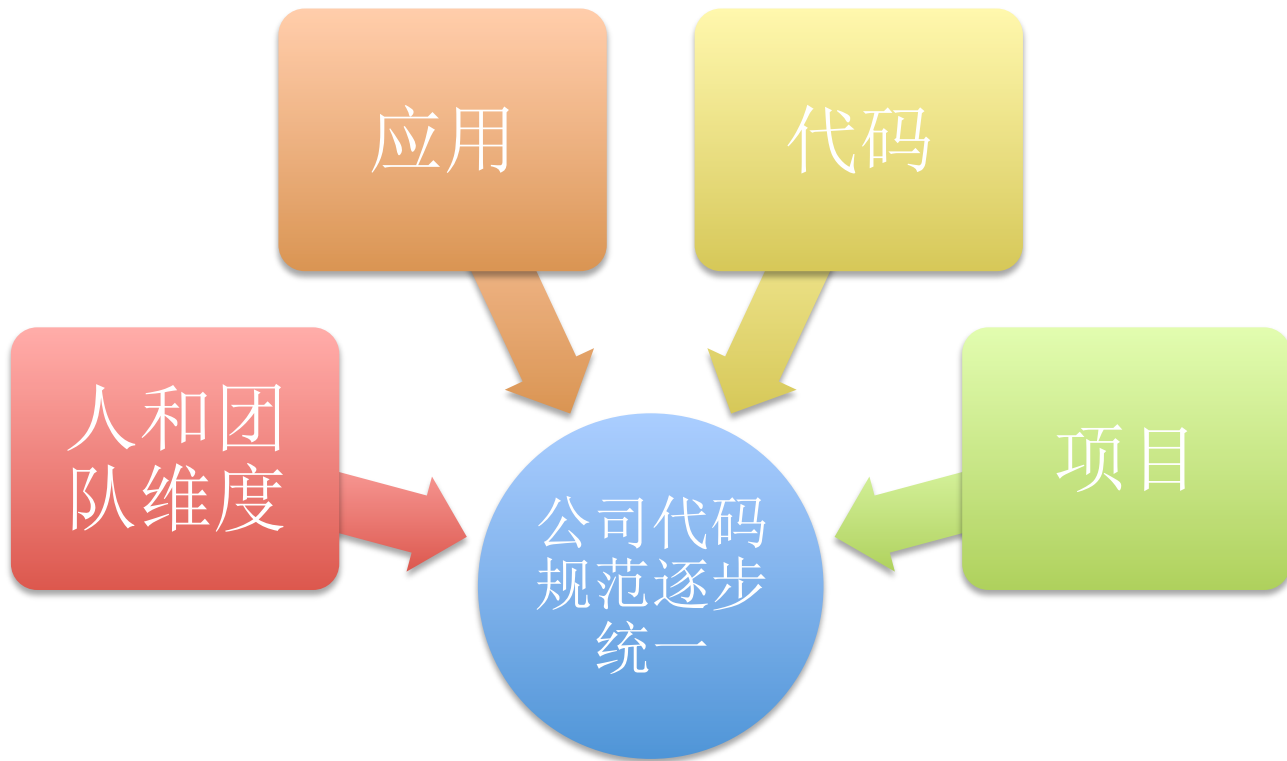
1. 每个程序员都有一些习惯，到底是用4个空格还是2个空格，if到底要不要加括号闭合？ ....
2. 阿里巴巴JAVA开发手册推出来了，先自己熟读，逐条遵守？
3. 将手册内容跟团队内部分享，靠大家自觉达到团队规范统一？
4. 跟公司其他团队分享并遵守规约，达到公司规范统一？



## 工具化落地



## 多维度报表-数据驱动代码质量提升



2

## 02 真实案例

*Part two*



# IDE本地check

```
/** ... */  
package com.alibaba.dubbo.rpc.http.codec;
```

```
import ...
```

```
/**  
 * @author ding.lid  
 */
```

```
public abstract class MapCode
```

```
public String getContentT
```

```
所有的覆写方法，必须加@Override注解。
```

```
反例：getObject()与getObject[]的问题。一个是字母的O，一个是数字的0，加@Override可以准确判断是否覆盖成功。另外，如果在抽象类中对方法签名进行修改，其实现类会马上编译报错。
```

```
public abstract Object de
```

```
public abstract void enco
```

```
public abstract Object[]
```

```
}
```

Rule Detail

Name  
MissingOverrideAnnotationRule

Severity  
Blocker

Message  
所有的覆写方法，必须加@Override注解。

Description  
反例：getObject()与getObject[]的问题。一个是字母的O，一个是数字的0，加@Override可以准确判断是否覆盖成功。另外，如果在抽象类中对方法签名进行修改，其实现类会马上编译报错。

Examples

```
/**  
 * @author cailiang  
 * @date 2016/12/24  
 */  
public class MissingOverrideAnnotationRule extends AbstractEclipseRule {  
    @Override  
    public Object visit(ASTCompilationUnit node, Object data) {  
        return super.visit(node, data);  
    }  
}
```

查看所有实现的规则

```
if (rs.isNull()) {  
    return null;  
} else {  
    return sqlTimestamp;  
}  
  
@Override  
public Object getResult(CallableStatement cs, int columnIndex) throws SQLException {  
    Object sqlTimestamp = cs.getTimestamp(columnIndex);  
}
```

Problems Javadoc Declaration Console Progress Error Log P3C Results

61 Blockers, 293 Criticals, 739 Majors

▼ Blocker (61 Violations)

▶ Map/Set的key为自定义对象时，必须重写hashCode和equals。 (58 Violations)

▶ 所有的覆写方法，必须加@Override注解。 (2 Violations)

▶ 所有的包装类对象之间值的比较，全部使用equals方法比较。 (1 Violations)

▶ Critical (293 Violations)

▶ Major (739 Violations)

清除扫描过和标记

选中规则如果灯亮  
表示该规则支持QuickFix

## 云效在线扫描check

最近构建

历史构建

#1 构建成功 在 2017-11-01 22:38:19 通过 手工

阿里巴巴Java代码规约扫描

成功

查看日志

统计



问题数

统计



Blocker

统计



Critical

统计



Major

File /root/space/117203/source/src/main/java/com/alibaba/acc/AccountService.java

Violation	Error Description	Line
Major	[AlibabaJavaComments.ClassMustHaveAuthorRule] - 【AccountService】缺少包含@author的注释信息	29 - 109
Major	[AlibabaJavaComments.CommentsMustBeJavadocFormatRule] - 字段【loginUsers】必须使用javadoc形式的注释	44 - 44
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【login】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	54 - 69
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【register】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	93 - 104

[Back to top](#)

File /root/space/117203/source/src/main/java/com/alibaba/acc/BookAdminService.java

Violation	Error Description	Line
Major	[AlibabaJavaComments.ClassMustHaveAuthorRule] - 【BookAdminService】缺少包含@author的注释信息	20 - 90
Major	[AlibabaJavaComments.CommentsMustBeJavadocFormatRule] - 字段【clock】必须使用javadoc形式的注释	27 - 27
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【findAll】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	31 - 33
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【findOne】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	36 - 38
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【listMyBook】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	41 - 43
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【saveBook】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	46 - 53
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【modifyBook】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	56 - 72
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【deleteBook】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	75 - 89

[Back to top](#)

File /root/space/117203/source/src/main/java/com/alibaba/acc/BookBorrowService.java

Violation	Error Description	Line
Major	[AlibabaJavaComments.ClassMustHaveAuthorRule] - 【BookBorrowService】缺少包含@author的注释信息	23 - 170
Major	[AlibabaJavaComments.CommentsMustBeJavadocFormatRule] - 字段【clock】必须使用javadoc形式的注释	33 - 33
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【applyBorrowRequest】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	37 - 60
Major	[AlibabaJavaExceptions.TransactionMustHaveRollbackRule] - 方法【cancelBorrowRequest】需要在Transactional注解指定rollbackFor或者在方法中显示的rollback。	63 - 63

# codeReview系统集成check

The screenshot displays a code review page with a sidebar on the left and a main content area. The sidebar includes sections for 'Next Step', 'Author', 'Reviewers', 'Lint', 'Unit', 'Branch', 'Apply Patch', 'Coding Guidelines', 'Test Lab', 'Subscribers', and 'Projects'. The 'Lint' section shows 'No Linters Available' and 'No Unit Test Coverage'. The 'Coding Guidelines' section is highlighted with a red box and lists several critical errors related to naming conventions. The main content area shows a list of code changes with a 'Critical' status indicator. On the right side, there is a list of actions: 'Edit Revision', 'Update Diff', 'Edit Dependencies', 'Edit Manifest Tasks', 'Download Raw Diff', 'Automatically Subscribed', 'Award Token', and 'Flag For Later'.

Accepted All Users

Next Step and 'co: [redacted]

Author [redacted]

Reviewers [redacted]

Lint ★ No Linters Available

Unit ★ No Unit Test Coverage

Branch g [redacted]

Apply Patch ak pa [redacted] 4

Coding Guidelines [redacted] (Critical)

- Critical 常量 `【Regression_Global_Regression_ID】` 命名应全部大写并以下划线分隔 at line 74
- Critical 常量 `【Regression_Manual_Regression_ID】` 命名应全部大写并以下划线分隔 at line 75
- Critical 常量 `【Regression_API_Regression_ID】` 命名应全部大写并以下划线分隔 at line 76
- Critical 方法名 `【SyncRemovedProjectSubscribe】` 不符合lowerCamelCase命名风格 at line 713
- Critical 方法名 `【SyncProjectSubscribe】` 不符合lowerCamelCase命名风格 at line 756
- Critical 变量名 `【Pipeline】` 不符合lowerCamelCase命名风格 at line 141
- Critical 方法名 `【OpenCompanyCodeScan】` 不符合lowerCamelCase命名风格 at line 3020
- Critical 方法名 `【CloseCompanyCodeScan】` 不符合lowerCamelCase命名风格 at line 3063

Test Lab No Testlab Available. (How to use?)

Subscribers [redacted]

Projects None

Test Lab this revision.

added a reviewer: [redacted]

Via Conduit · Mon, Oc [redacted]

## 应用代码质量数据

【各个应用代码扫描情况】

应用名称	问题总数	Blocker	Critical	Major	应用owner	TL
应用名称1	1	0	1	0	应用owner1	TL1
应用名称2	1	0	1	0	应用owner2	TL2
应用名称3	1	0	1	0	应用owner3	TL3
应用名称4	1	0	1	0	应用owner4	TL4
应用名称5	1	0	1	0	应用owner5	TL5
应用名称6	1	0	1	0	应用owner6	TL6
应用名称7	1	0	1	0	应用owner7	TL7
应用名称8	1	0	1	0	应用owner8	TL8
应用名称9	1	0	1	0	应用owner9	TL9
应用名称10	1	0	1	0	应用owner10	TL10
应用名称11	1	0	1	0	应用owner11	TL11
应用名称12	1	0	1	0	应用owner12	TL12
应用名称13	1	0	1	0	应用owner13	TL13
应用名称14	1	0	1	0	应用owner14	TL14
应用名称15	1	0	1	0	应用owner15	TL15
应用名称16	1	0	1	0	应用owner16	TL16
应用名称17	1	0	1	0	应用owner17	TL17
应用名称18	1	0	1	0	应用owner18	TL18
应用名称19	1	0	1	0	应用owner19	TL19
应用名称20	1	0	1	0	应用owner20	TL20

3

## 03 在线扫描演示

*Part three*

# 一键开启扫描

云效-测试服务在线扫描入口: <https://rdc-test.aliyun.com/>

测试任务 / 阿里巴巴Java代码规约扫描

## 新建阿里巴巴Java代码规约扫描服务

\* 任务名称

输入任务名称

0/30

服务



阿里巴巴Java代码规约扫描

\* 代码地址

阿里云代码示例:git@code.aliyun.com:xx/yy.git

代码分支

代码分支

确定



阿里巴巴Java代码规约扫描

### 服务介绍

《阿里巴巴Java开发手册》是阿里巴巴集团技术团队的集体智慧结晶和经验总结，经历了多次大规模一线实战的检验及不断的完善，系统化地整理成册，反馈给广大开发者。手册以码出高效，码出质量为愿景，切实帮助到开发者提升研发效能，遵守编码规范，进而提高系统稳定性。《手册》以Java开发者为中心视角，划分为编程规约、异常日志、单元测试、安全规约、工程结构、MySQL数据库六个维度，明确了作为合格Java工程师的必备技术素养。

# 发布流水线卡点

流水线 /

运行流水线

编辑流水线

回滚

回滚列表

帮助文档

构建

版本: V0.0.1-12  
2分1秒

构建  
已完成

...

约码扫描验证

版本: V0.0.1-12  
2分18秒

阿里巴巴...  
进行中

查看结果 跳过

...

单元测试

未开始

...

部署

未开始

版本名称: V0.0.1-12

分支: master Commit: 6c594a58

构建

节点名称	操作人	操作时间	时长	状态	操作
构建	张三	2017-11-02 10:02:26	2分1秒	SUCCESS	版本信息

构建流程

日志详情

全屏查看日志

编译打包 [INFO] Webapp assembled in [61 msec]

4

## 04 帮助&FAQ

*Part four*



## 帮助文档

1.IDE体验入口:

<https://yq.aliyun.com/articles/224817>

2.云效在线扫描体验入口:

<https://rdc-test.aliyun.com/>

3.另云效-测试服务不仅仅做代码扫描，更多单元测试、API测试等能力，请查看帮助文档:[https://help.aliyun.com/document\\_detail/54836.html](https://help.aliyun.com/document_detail/54836.html)

4. p3c github入口:

<https://github.com/alibaba/p3c>

## FAQ

1. p3c怎么与gradle集成? [参考](#)
2. p3c怎么与maven集成? [参考](#)
3. p3c怎么与Sonar集成? [参考](#)

.....

# THANK YOU



扫码查看更多技术干货



扫码体验云效新品