

# Ch. 3 软件开发中的规范

2018 / 4

# 目录

- **3.1**开发规范的重要性
- **3.2**文档规范
- **3.3**编程规范
- **3.4**流程规范
- **3.5**敏捷



# 规则的重要性

以下是一段代码

```
__ $ _ $ _ "OK" $ _ $ _ $ SS $ _
```

# 另一段代码

```
/*
```

وفيما يلي وصف وظيفة هذا الكود

هو تحقيق وظيفة واحد يترجم السلسلة الرقمية

画像フォーマットのファイルパス

```
*/
```

```
void fun()
```

```
{
```

```
}
```



# 目录

- 3.1 开发规范的重要性
- 3.2 文档规范
- 3.3 编程规范
- 3.4 流程规范
- 3.5 敏捷

# 为什么要有文档

- 理解问题本身是有分歧的
- 语言本身会有歧义
- 形成文档是达成共识的过程
- 文档记录达成的共识
- 文档记录达成共识的过程
- 需求会经常发生变更



# 需求发生变更的原因

- 需求会逐渐更明确
- 随着产品研究深入，会有进一步的使用需求产生
- 需求整理过程中，对原有业务进行了重新规划
- 需求人员能力不足
- .....

# 文档规范

- 规范文档的种类
- 规范文档的格式
- 规范文档的内容
- 规范文档的内容变动
- 规范文档的管理办法



# 软件开发过程中涉及到的文档

- 规范性文档
- 流程性文档
- 描述性文档

# 一些规范性文档

- 开发文档规范
- 代码编写规范
- 服务器操作规范
- .....



# 文档本身的规范

- 文档的命名规则
- 文档的编号规则
- 文档的行文规则

# 流程性文档

- 可行性分析说明书
- 需求说明书
- 设计说明书
- 测试要点说明
- .....



# 描述性文档

- 软件操作手册
- .....

# 目录

- 3.1 开发规范的重要性
- 3.2 文档规范
- 3.3 编程规范
- 3.4 流程规范
- 3.5 敏捷



# 为什么要有编程规范

- 每个人写代码都有自己的风格
- 避免一些常见的错误
- 特殊规则避免行业特殊的问题
- 质量标准统一

# 一般都规范哪些

- 命名
- 注释
- 常见写法
- 行业相关规范
- .....



# 一些参考

- Google
- Microsoft
- Facebook
- 阿里巴巴

# Google – C++



The screenshot shows a web browser displaying the Google C++ Style Guide in Chinese. The browser's address bar shows the URL: `zh-google-styleguide.readthedocs.io/en/latest/google-cpp-styleguide/naming/#`. The page has a blue header with the text "Google 开源项目风格指南" and "latest". Below the header is a search bar labeled "Search docs". A dark grey sidebar on the left contains a table of contents for the "C++ 风格指南 - 内容目录", with "7. 命名约定" (Naming Conventions) selected and highlighted. The main content area on the right shows the start of section "7. 命名约定", which discusses the importance of naming management and consistency rules. Below this, the sub-section "7.1. 通用命名规则" (General Naming Rules) is visible, starting with a "总述" (Summary) section that mentions naming functions, variables, and files.

← → ↻ ⓘ zh-google-styleguide.readthedocs.io/en/latest/google-cpp-styleguide/naming/#

🏠 Google 开源项目风格指南  
latest

Search docs

Google 开源项目风格指南 (中文版)

☐ C++ 风格指南 - 内容目录

- 0. 扉页
- 1. 头文件
- 2. 作用域
- 3. 类
- 4. 函数
- 5. 来自 Google 的奇技
- 5. 其他 C++ 特性
- ☐ 7. 命名约定
- 8. 注释
- 9. 格式
- 10. 规则特例
- 11. 结束语

Docs » C++ 风格指南 - 内容目录 » 7. 命名约定

## 7. 命名约定

最重要的一致性规则是命名管理. 命名的某个名字代表的含义: 类型, 变量, 函数, 有些命名规则.

命名规则具有一定随意性, 但相比按个人习惯, 则总归是规则.

### 7.1. 通用命名规则

#### 总述

函数命名, 变量命名, 文件命名要有描述性.

#### 说明

尽可能使用描述性的命名, 别心疼空间, 别用缩写.



# 目录

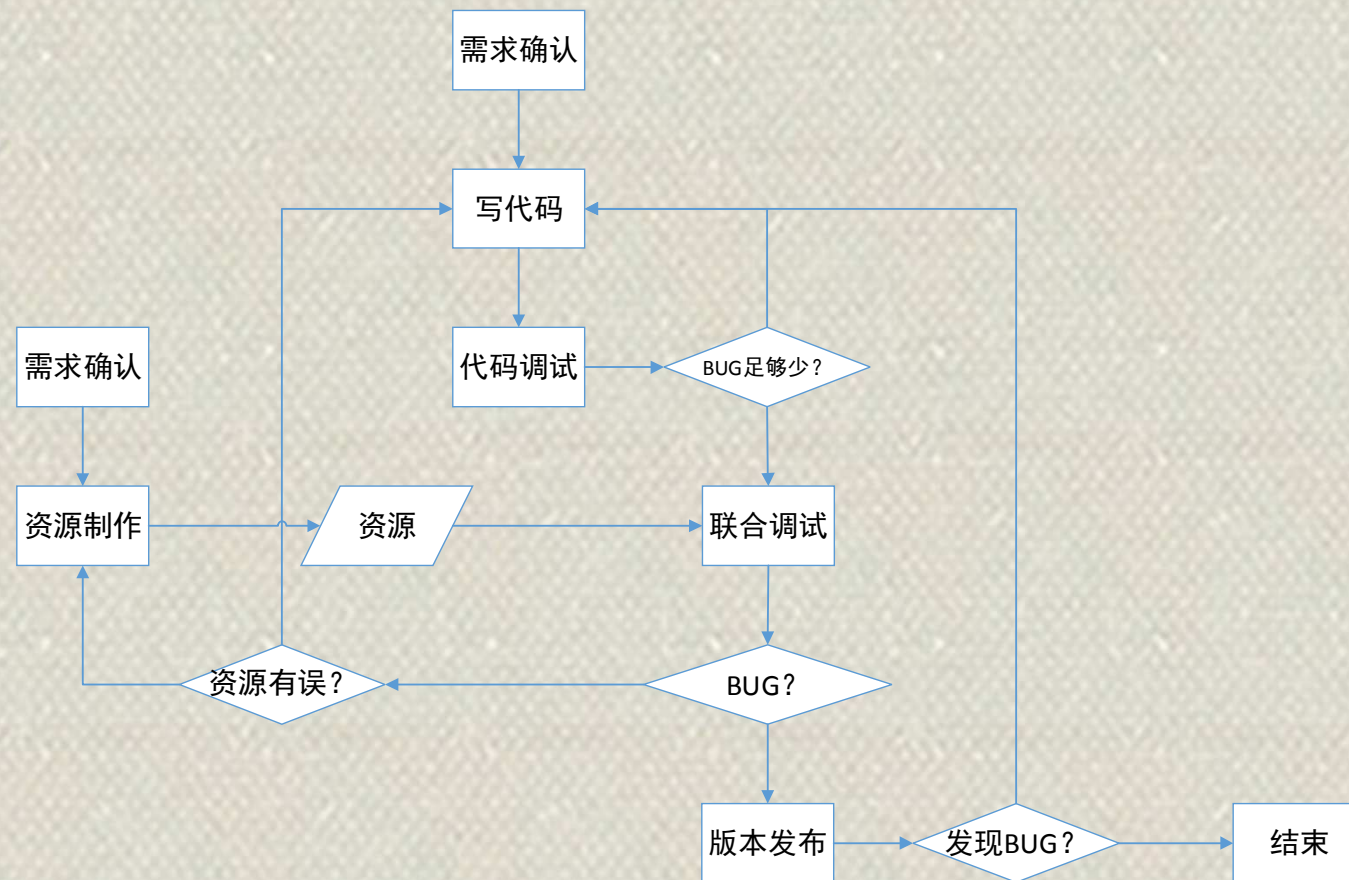
- 3.1 开发规范的重要性
- 3.2 文档规范
- 3.3 编程规范
- 3.4 流程规范
- 3.5 敏捷

# 流程规范的内容

- 定义角色分工
- 工作衔接处理说明
- 项目流程说明
- 项目进度说明和进度规划



# 哪些流程












# 怎样进行流程管理

- 用人监督 PM
- 分工记表
- 用流程软件



# 一些流程化的方法

 WBS 表.xlsx	2012/8/13 16:48
 项目变更管理表.xlsx	2012/8/13 16:50
 项目风险管理表.xlsx	2012/8/13 16:50
 项目沟通计划表.xlsx	2012/8/13 16:50
 项目会议纪要.xlsx	2012/8/13 16:50
 项目进度计划表.xlsx	2012/8/13 16:49
 项目任务书.xlsx	2012/8/13 16:49
 项目总结表.xlsx	2012/8/13 16:50
 项目组成员表.xlsx	2012/8/13 16:48

# 一些流程软件

- Bugzilla: 开源的缺陷跟踪系统（Bug-Tracking System），它可以管理软件开发中缺陷的提交（new），修复（resolve），关闭（close）等整个生命周期





# 一些流程软件

- **Jira** :被广泛应用于缺陷跟踪、客户服务、需求收集、流程审批、任务跟踪、项目跟踪和敏捷管理等工作领域。



# 其他产品

- **Axosoft**
- **Redmine**
- **TestTracker**
- 腾讯**TAPD**
- 禅道



# 目录

- 3.1 开发规范的重要性
- 3.2 文档规范
- 3.3 编程规范
- 3.4 流程规范
- 3.5 敏捷

# 敏捷开发

- 以人为核心、迭代、循序渐进的**开发方法**



# 一些特点

- 快速应对需求变化
- 迭代式开发
- 及时修正问题

# 工作形式

- **XP**
- **Scrum**
- **Kanban(看板)**



# XP -- ExtremeProgramming

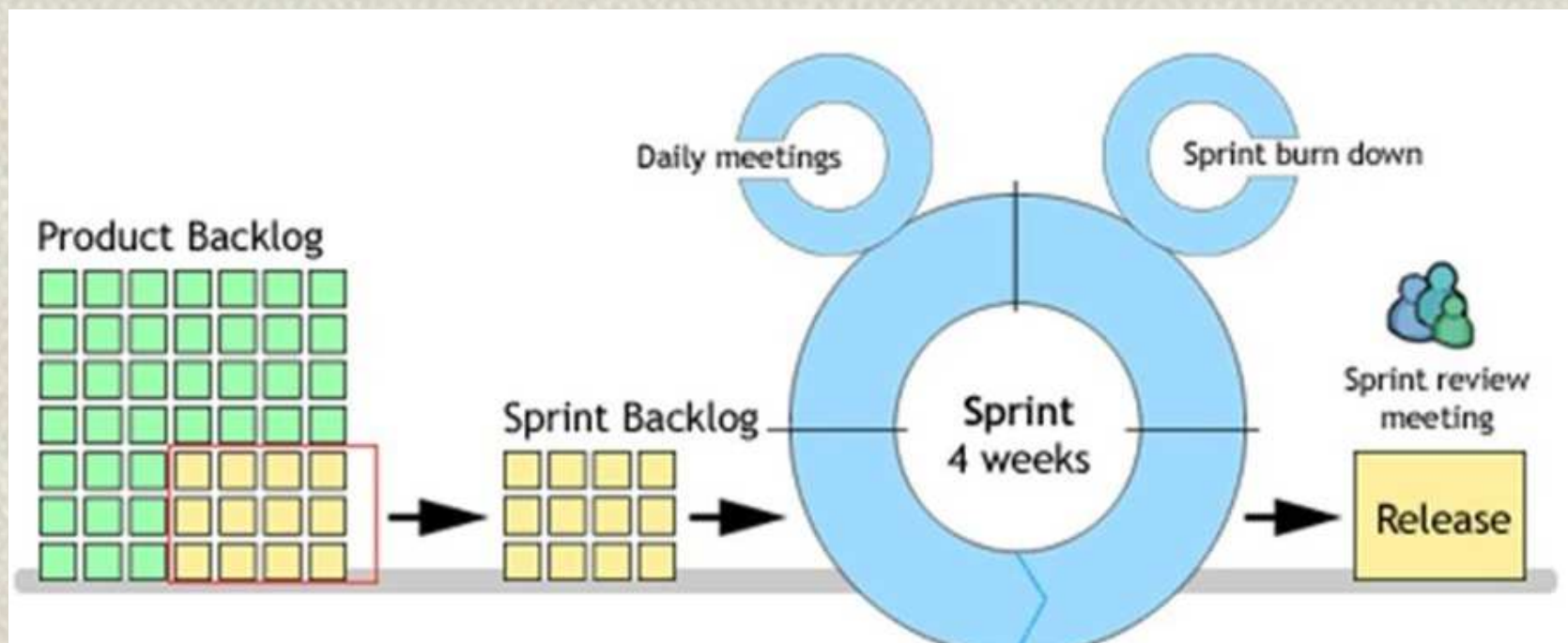
- 适合小团队
- 突出了人在软件开发过程中的作用
- 直接编程比什么（文档、架构）都有效
- 沟通、简单、反馈、尊重和勇气
- 测试驱动开发
- 结对编程
- 严格遵守流程

# Scrum

- 强调项目的管理
- 划分出角色：**Product Owner**、**Scrum Master**、**Scrum Team**
- 通过管理，对任务根据需要加以灵活调整



# Scrum 开发模型

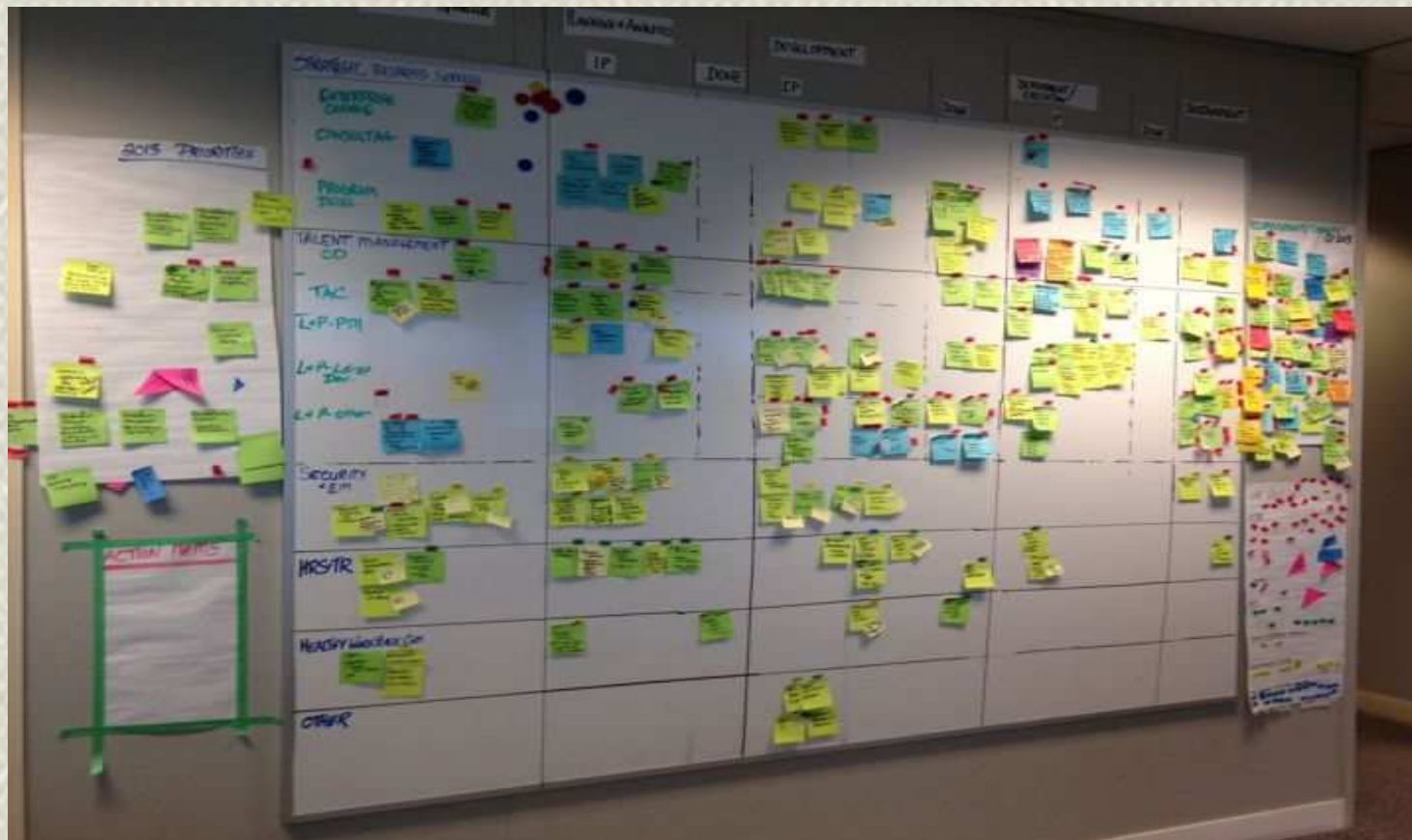


# Kanban

- 主要的目的是做过程改进
- 寻找瓶颈和短板



# 一个看板



# 对比

- **XP**是开发方法论
- **Scrum**是管理
- **Kanban**是手段



Q & A