

# 1. 配置管理概述

## 1.1. 配置管理定义

软件配置管理 (SCM, Software Configuration Management)

定义

- 记录软件产品的演化过程
- 得到精确的产品配置
- 最终保证产品的完整性、一致性、可溯性、可控性

主要功能

- 版本管理
- 变更管理

## 1.2. 配置项

软件配置项

- SCI : software configuration item
- 受控于软件配置管理的款项
- 在系统周期中对一个系统中的配置项进行标识和定义，一个特定的、可视化的工作产品集
- 配置项可以看作oop 中的类，版本看作实例

常见配置项

- 软件项目计划、需求分析结果、软件需求规格说明书、设计规格说明书、源代码清单、测试规格说明书、测试计划、测试用例与实验结果、可执行程序、用户手册、维护文档。

## 1.3. 基线

定义

- 提供了软件生存期各个开发阶段的特定点，标志了开发过程一个阶段的结束和里程碑

- 一个配置项形成并通过审核就形成基线
- 基线修改需要按照正式程序执行

## 1.4. 配置控制委员会

软件配置控制委员会 (SCCB, Software Configuration Control Board)

责任:

- 评估变更
- 批准变更申请
- 在生存期内规范变更申请流程
- 对变更进行反馈
- 与项目管理层沟通

## 1.5. 配置管理作用

- 没有变更管理，项目就会无限放大
- 软件配置管理是软件项目管理的重要内容，也是保证软件质量的重要手段。它能够对软件开发过程进行有效管理和控制，从而实现软件产品的完整性、一致性、可控性，使产品极大程度地与用户需求相吻合。它能够控制、记录、追踪对软件的修改并形成规范文档，方便日后维护和升级，更重要的是能够保护代码资源，积累软件财富，提高软件重用率。

# 2. 配置管理过程

## 2.1. 配置项标识、跟踪

- 将软件项目中需要控制的部分拆分成 SCI
- 配置项被唯一标识
- 建立配置项的对应关系，便于跟踪和版本控制，实现数字化的管理

## 2.2. 配置管理环境建立

- 建立配置管理库
  - 存储所有基线配置项和相关文件内容的系统
  - 是在软件产品的整个生存期中建立和维护软件产品完整性的主要手段

## 2.3. 基线变更管理

- 基线修改受到控制，经过 **SCCB** 授权，按程序控制并记录基线修改的过程
- 基线变更系统
  - 变更请求
  - 变更评估
    - 变更分类
    - 变更的影响分析
  - 变更批准/拒绝
  - 变更实现

## 2.4. 配置管理审计

作用

- 作为变更控制的补充手段

分类

- 配置管理过程审计
  - 是否按照流程执行
- 基线审计
  - 保证软件工作产品的完整性和一致性，SCI 正确构造并实现

## 2.5. 配置状态统计

- 被批准的配置项
- 变更请求的数量
- 配置项的所有请求的变化状态
- 配置项所有被批准的变更实现状态
- 配置管理系统以及SCCB 在运作中发生异常的次数

## 2.6. 配置管理计划

1. 人员职责（确定 SCCB 等）
2. 配置项定义
3. 基线定义
4. 版本控制（说明配置管理工具）
5. 定义变更控制系统

## 3. 敏捷项目的配置管理

### 3.1. 全面配置管理

#### 特征

- 敏捷项目的一个重要特征是持续交付，因此配置管理是重要的要素
- 敏捷需要全面配置管理，即所有相关产物都要进行配置管理

#### 基本要求

- 代码和构建产物配置管理
- 应用配置管理
- 环境配置管理

### 3.2. 分支管理策略

- 基于分支的开发
  - 开发都在分支进行，可能有多个并行分支，快要上线才合并到主干
- 基于主干的开发（敏捷项目采用）
  - 所有提交到主干上，提交后自动触发持续集成进行验证和快速反馈
  - 持续交付更倾向使用基于主干的开发模式

## 4. 配置管理工具

- git
  - 分布式版本控制软件
- ClearCase
- Firefly
- SVN