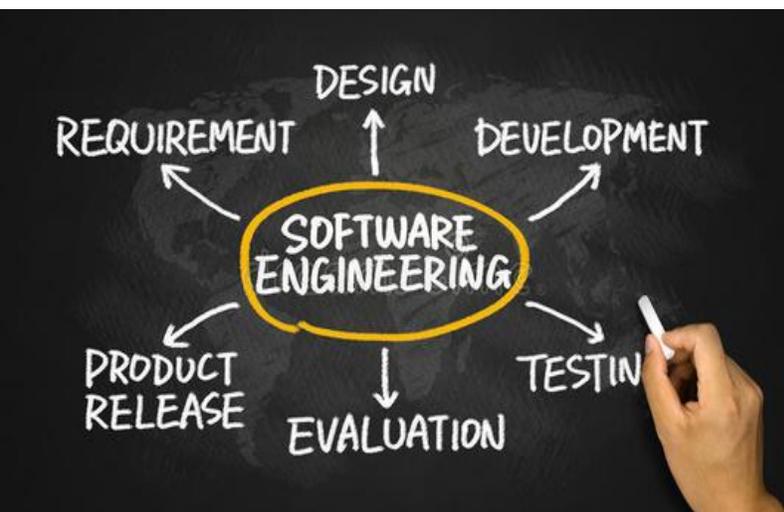




软件工程基础

—— 第21章 软件配置管理



计算机学院 孟宇龙

21.1 软件配置管理概述

21.2 SCM中心存储库

21.3 SCM过程

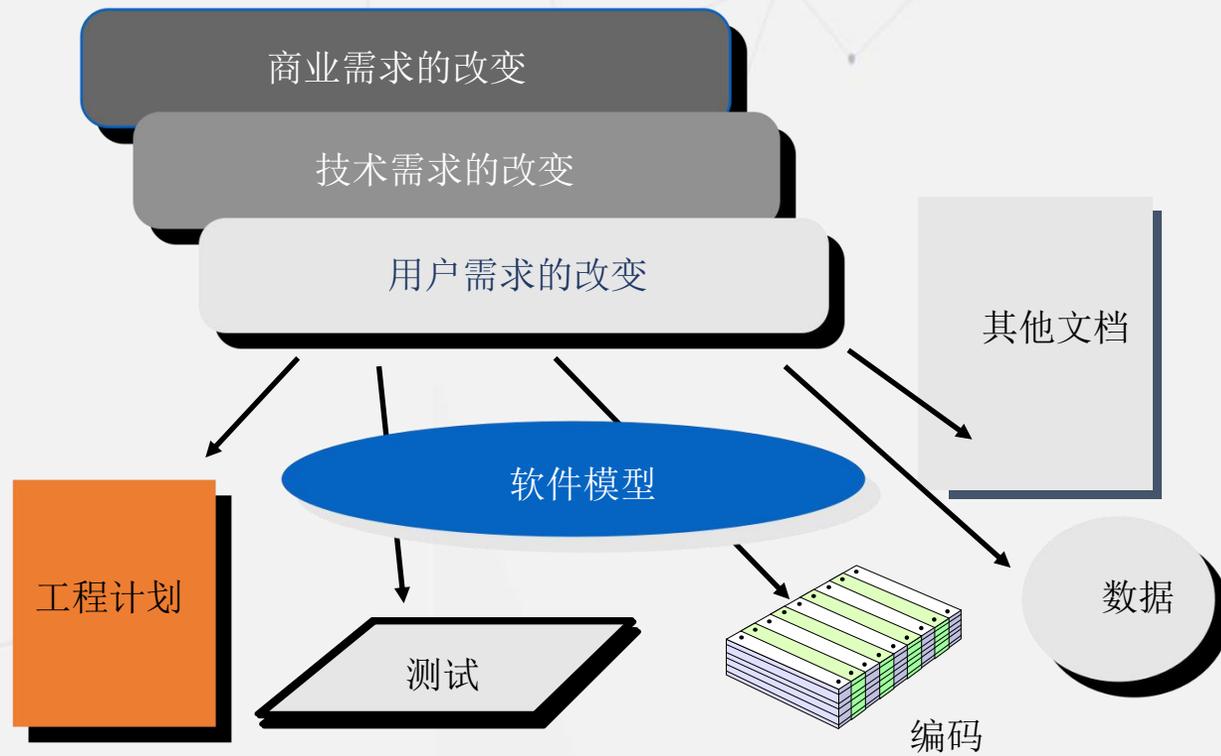
关键概念

- 基线
- 变更控制
- 配置审核
- 配置管理
- 配置对象元素
- 内容管理
- 标识
- 影响管理
- 中心存储库
- SCM过程
- 软件配置项
- 状态报告
- 版本举例

不管你处在系统生存周期的什么阶段, 系统都可能发生变更, 并且在整个生存周期中将会不断地提出变更的要求。

Bersoff, et al, 1980

这些变更是什么？



- **变更是不可避免的**
- **软件配置管理也称为变更管理，是一组管理变更的活动**

它通过下面的方式来管理变更：

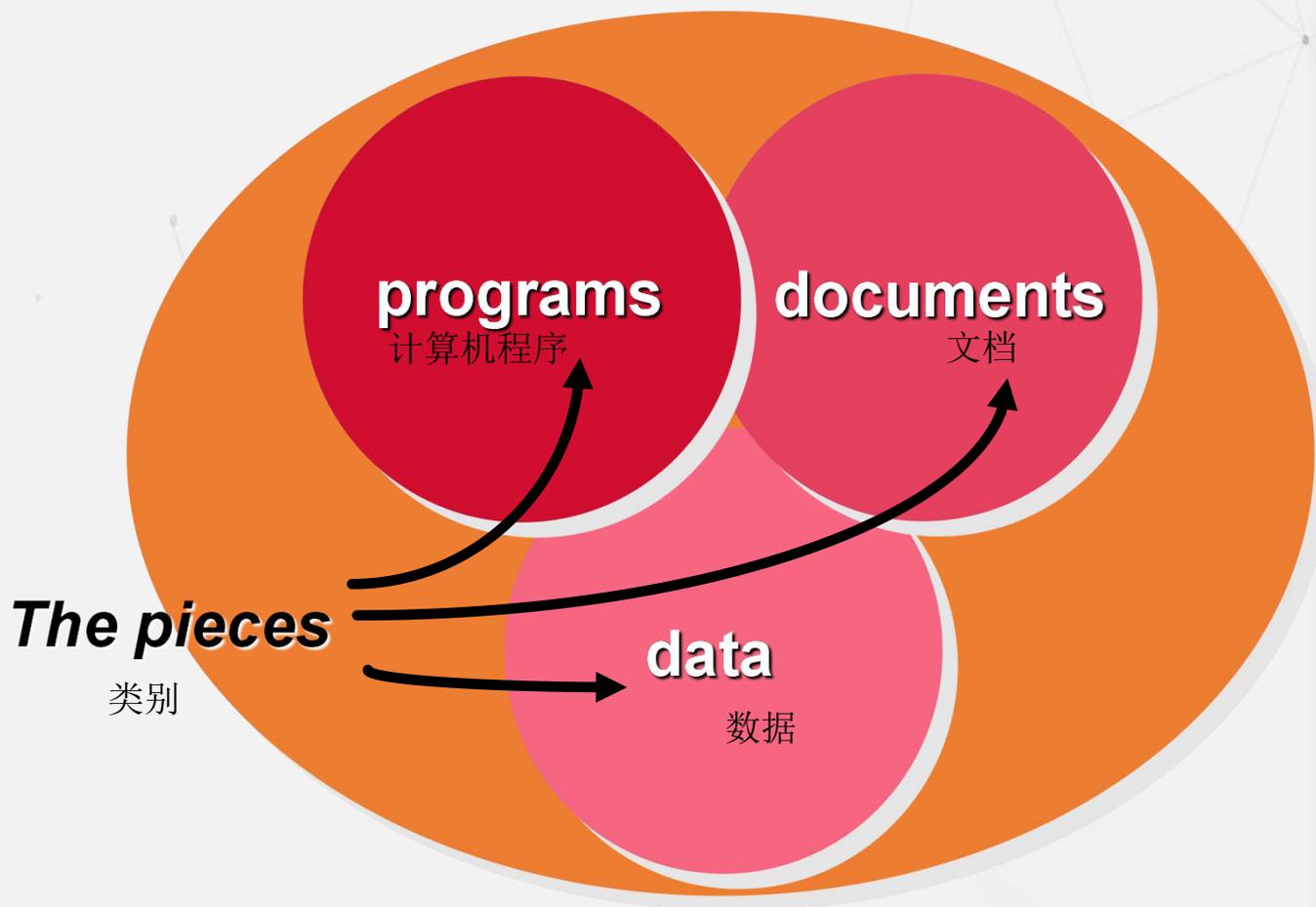
识别可能发生变更的工作产品，建立这些工作产品间的关系，指定管理这些工作产品的不同版本的机制，控制所施加的变更，审核和报告所发生的变更

- **协调软件开发以最大限度地减少混乱的技术称为配置管理**
- 配置管理是对软件开发团队正在构建的软件的修改进行标识、组织和控制的技术，其目标是使错误量减少的最小，并使得生产率最高

21.1 软件配置管理概述

- **软件配置管理（SCM）是在整个软件过程中应用的一种普适性活动**
- SCM活动用于：
 - （1）标识变更
 - （2）控制变更
 - （3）保证恰当地实施变更
 - （4）想其他可能的相关人员报告变更

2.1.1 软件配置管理概述



SC :

软件过程中产生的所有信息项总称为软件配置 (SC)

SCI : 软件配置项目

或小或大的命名信息元素

SCM : 软件配置管理

2.1.1 软件配置管理概述

- 四种基本的变更源：
 - (1) 新的业务或市场条件
 - (2) 新的客户需求
 - (3) 企业变化
 - (4) 预算或进度的限制

21.1.1 SCM场景

典型的配置管理工作场景包括：

负责软件小组的**项目经理**

负责CM规程和方针的**配置管理员**

负责开发和维护软件产品的**软件工程师**以及使用软件产品的**客户**

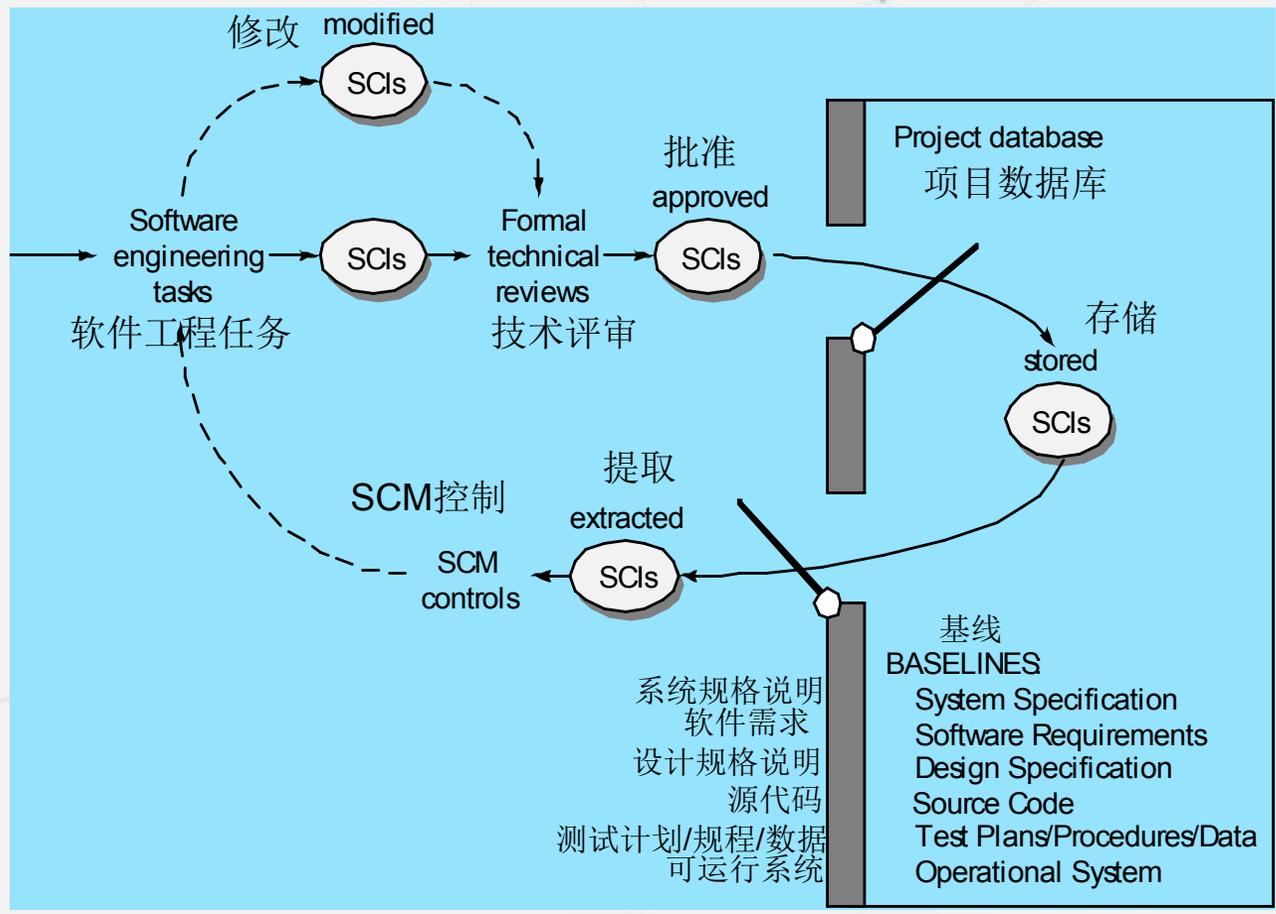
21.1.2 配置管理系统的元素

配置管理应具备的4个元素：

- 构件元素
- 过程元素
- 构建元素
- 人员元素

21.1.3 基线

- IEEE (IEEE标准610.12-1990) 是这样定义基线的：
 - 已经通过正式评审和批准的规格说明或产品，它可以作为进一步开发的基础，并且只有通过正式的变更控制规程才能修改它。
- 基线是软件开发中的里程碑，其标志是正式技术评审中已经获得批准的一个或多个软件配置项的交付。

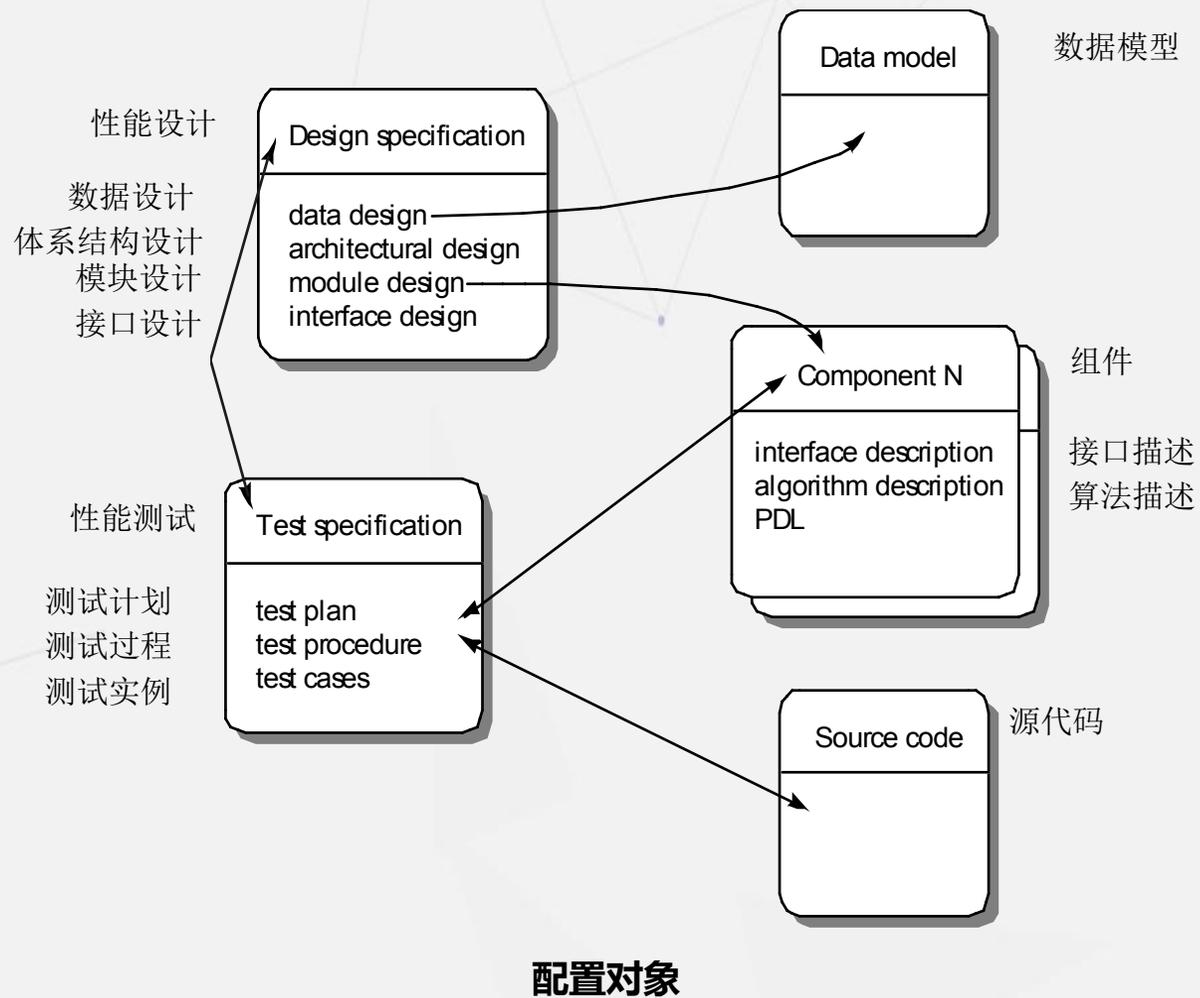


基线化的SCI和项目数据库

21.1.4 软件配置项

- 软件配置项是在软件工程过程中创建的信息
- 例如，一个测试用例、一份文档、一个程序构件、已经确定版本的工具、.....
- 实际中，将SCI组织成**配置对象**
- **配置对象**具有一个名称和多个属性，通过关系与其他配置对象关联

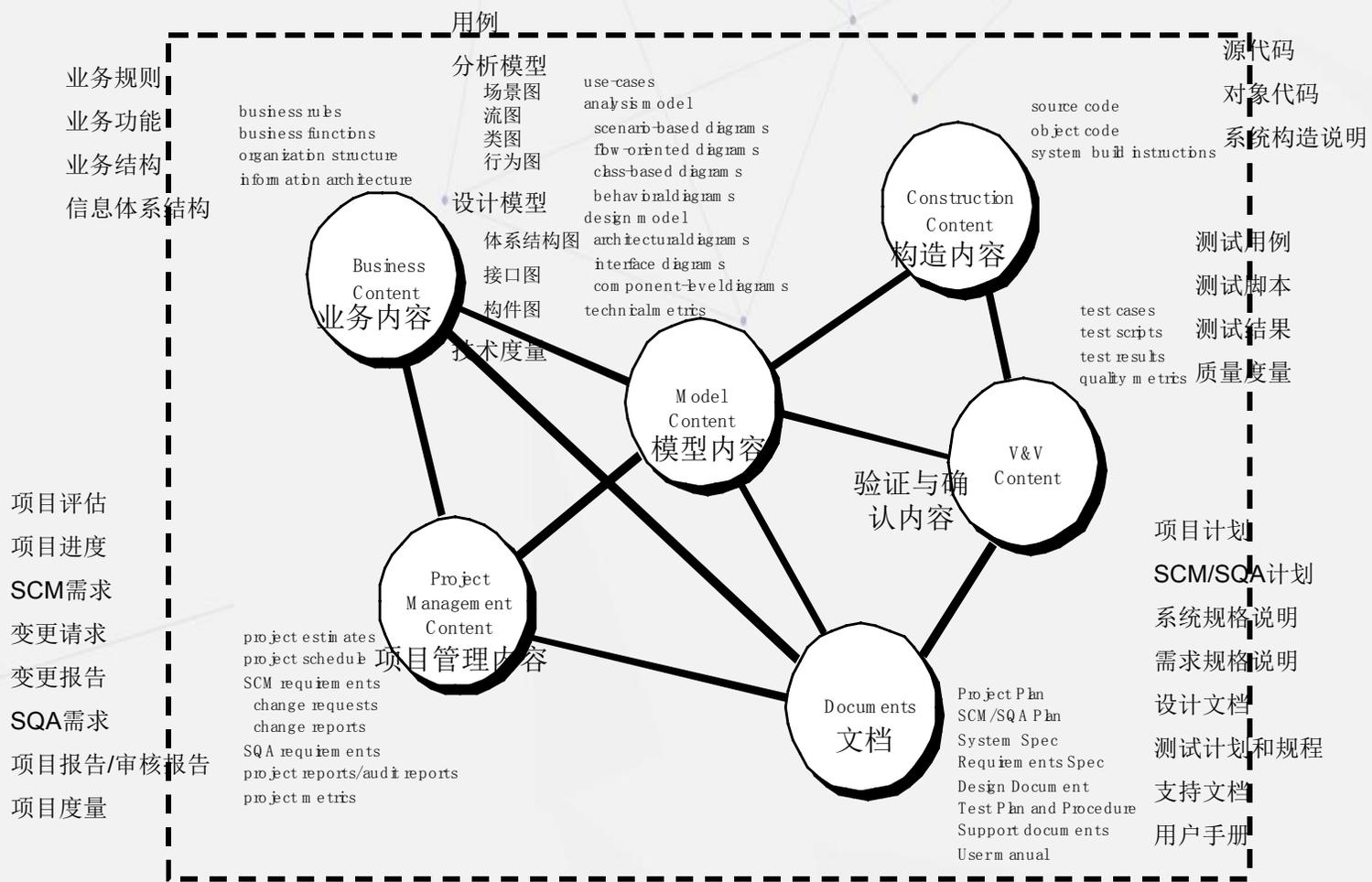
21.1.4 软件配置项



21.2 SCM中心存储库

- SCM中心存储库是一组机制和数据结构，它使软件团队可以有效地管理变更。
- 中心存储库具有以下功能[For89]：
 - 数据完整性
 - 信息共享
 - 工具集成
 - 数据集成
 - 方法执行
 - 文档标准化

21.2.1 一般特征和内容



中心存储库的内容



版本控制

保存所有这些版本，以便有效地管理产品发布，并允许开发者在测试和调试过程中可以返回到早先的版本。



依赖性跟踪和变更管理

中心存储库要管理所存储的配置对象之间的各种关系。



需求跟踪

- 跟踪由特定需求规格说明产生的所有设计构件、架构构件以及可交付产品。



配置管理

跟踪表示特定项目里程碑或产品发布的一系列配置。版本管理提供了所需的版本，而链接管理负责跟踪其相互依赖性。



审核跟踪

了解变更是在什么时候、什么原因以及由谁完成等信息。

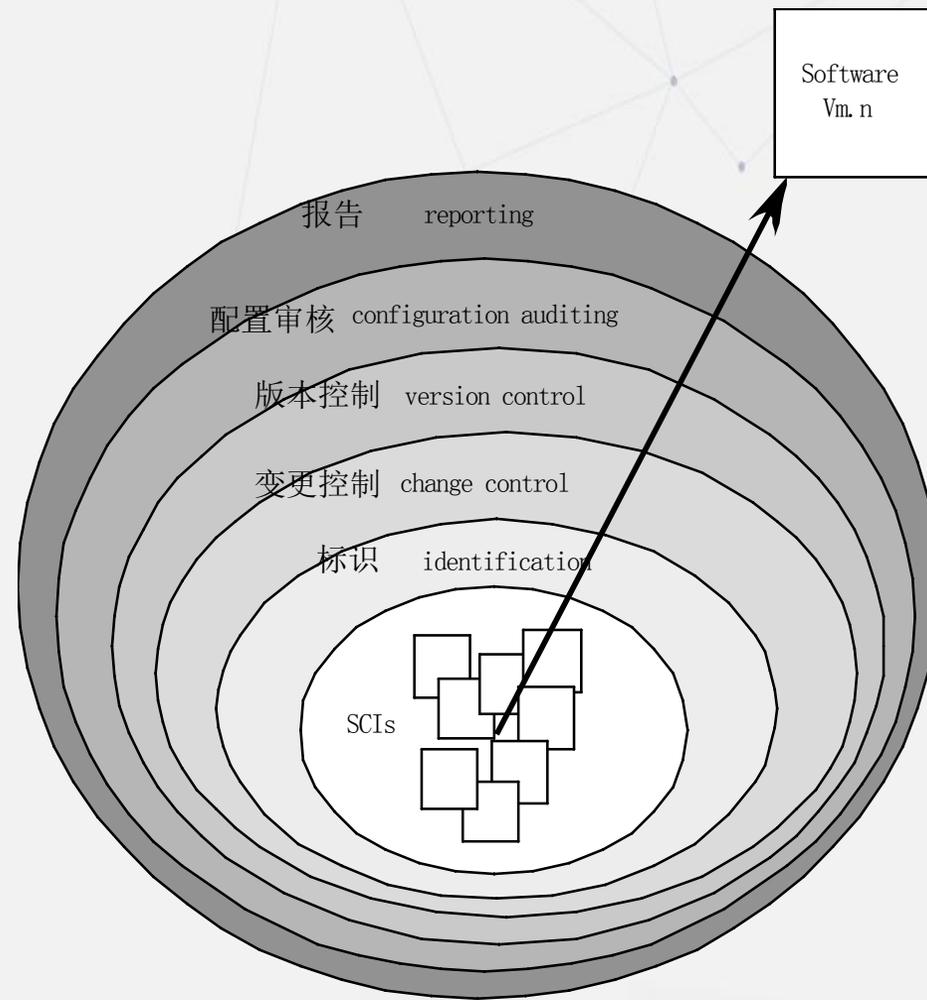


- **构件元素** ——是一组具有文件管理系统 (如, 数据库) 功能的工具, 使我们能够访问和管理每个软件配置项。
- **过程元素** ——是一个动作和任务的集合, 它为所有参与管理、开发和使用计算机软件的人员定义了变更管理 (以及相关活动) 的有效方法。
- **构造元素** ——是一组自动软件构造工具, 用以确保装配了正确的有效构件 (即, 正确的版本) 集。
- **人员元素** ——为有效实施SCM, 软件团队使用的一组工具和过程特性 (包括其他CM 元素)。

提出以下问题 ...

- 软件团队应该如何标识软件配置的离散元素?
- 组织应该如何管理程序 (及其文档) 的多个已有版本, 从而使变更能够高效地进行?
- 组织应该如何 在软件发布给客户之前和之后控制变更?
- 应该由谁负责批准变更并给变更确定优先级?
- 我们如何保证能够正确地完成变更?
- 应该采用什么机制去评价那些已经发生了的变更?

21.3 SCM过程



SCM 过程的层次

21.3.1 SCM中的对象标识

- 为了控制和管理软件配置项，必须对每个配置项单独命名，然后用面向对象方法进行组织
- 包括：基本对象和聚合对象
- 每个对象都具有一组能够唯一标识它的独特特征：名称、描述、资源表及“实现”

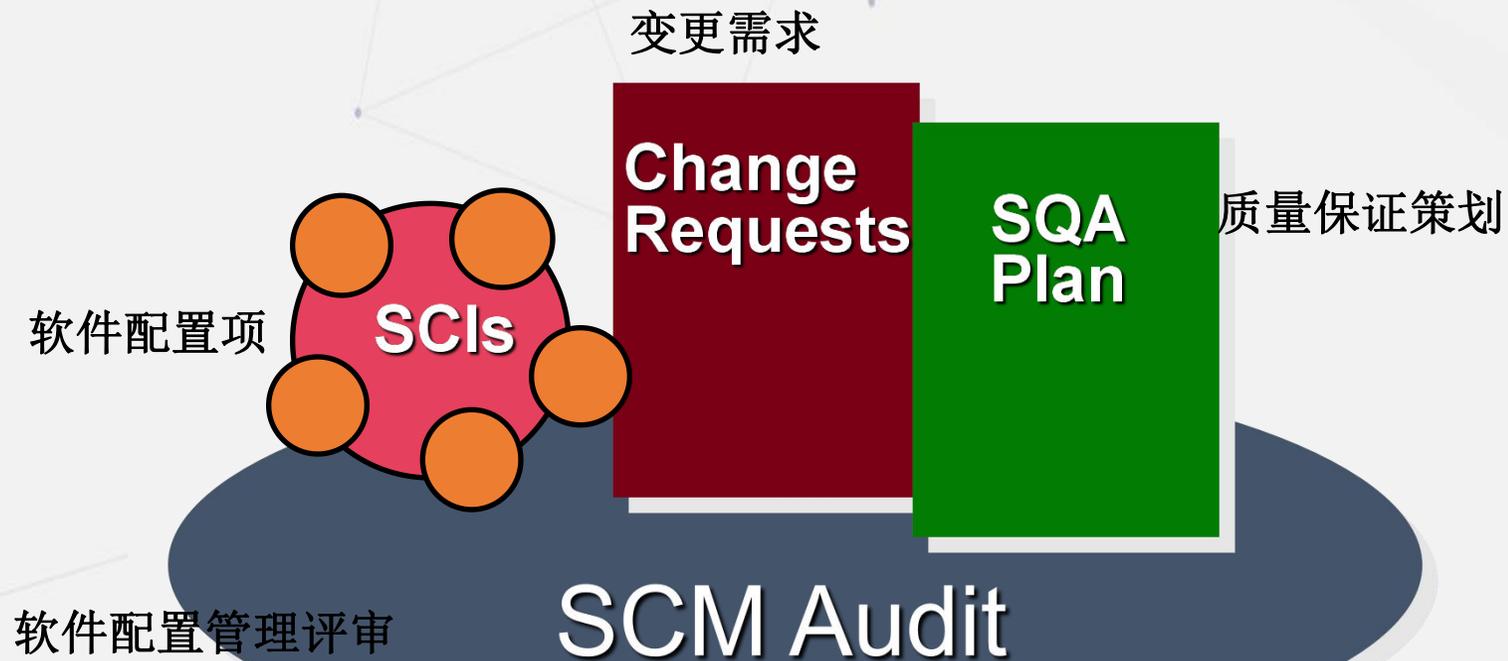
21.3.2 版本控制

- 版本控制结合了规程和工具，可以用来管理在软件过程中所创建的配置对象的不同版本。
- 版本控制系统实现或者直接集成了4个主要功能：
 - 存储所有相关配置对象的 **项目数据库 (中心存储库)**；
 - 存储配置对象所有版本(或能够通过与其先前版本间的差异来构造任何一个版本)的 **版本管理功能**；
 - 使软件工程师能够收集所有相关配置对象和构造软件特定版本的制作功能；
 - **问题跟踪** (也叫做错误跟踪) 使团队能够记录和跟踪与每个配置对象相关的所有重要问题的状态。

21.3.3 变更控制



21.3.5 配置审核



21.3.5 状态报告

