Measures

软件复用指南

麦哲思科技（北京）有限公司

主　题

- 软件复用的基本思想
- IEEE 1517
- 软件复用的经验与教训
- 小　结

Measures

软件复用的基本思想

复用的基本类型（1/3）

Measures

- **代码复用**
  - 一种最常见的复用形式
  - 最常见的情况复用代码被共享为公共类或是函数库或是过程
  - 优点：
    - 大量减少重复代码的出现，降低了开发和维护成本
  - 缺点：
    - 作用范围仅限于程序的编写而且会造成程序结构紧密耦合
- **模版复用**
  - 一种典型文档说明性复用。它通常被实践于运用一系列规定格式管理手工书写的文档，建模模型以及程序代码
  - 优点：
    - 文档模版形式的最大的好处在于对于手工书写文档的统一性和质量有了较好的保证
  - 缺点：
    - 使用者常常会因为个人的需要而修改文档规格造成混乱

复用的基本类型（2/3）

Measures

- **构件复用**
  - 构件通常是在某一个方面的能够高效解决问题的对象集合。
  - 构件具有的模块独立性使得它能够方便的插入到应用程序中，而此特性使得构件级复用的适用范围大大超过代码级复用和继承级复用。其次，广泛存在的基础平台例如Win32和Java平台都为第三方厂方开发和销售他们的构件提供了便利。

- **框架复用**
  - 开发人员在框架的基础上进行开发，只需要完成应用程序最终20%的部分而另外80%的部分已经由框架完成了。
  - 目前，在保险，人力资源，制造业，银行和电子商务软件开发中已有一些成熟的框架模型
  - 框架对业务领域重点提出了相应的解决办法，并将那些复杂的需要花费时间进行开发验证的复杂逻辑进行了良好的封装。
  - 框架复用也有自身的缺点，框架的复杂程度给使用者的学习和使用带来了一定的难度。

复用的基本类型（3/3）

Measures

- **交付物复用**
  - 包括使用以前创建的用例，标准文档，模型，过程方法和计划以及应用程序等。
- **模式复用**
  - 利用已被反复实践的规则解决通用性问题。
  - 模式的复用的思想是将复用代码背后的思想进行了抽象从而形成一种应用规则。它是一种高层次的复用机制，其生存期远远超过了描述它的语言甚至是规范它的面向对象思想。
  - 模式复用提供了更高层次的复用和跨平台，跨语言的特性。
- **领域构件复用**
  - 某个领域方面的构件常常是一系列关系紧密，完成具体功能的商业对象组合。
  - 大量已有的，关系紧密的商业功能在许多的应用程序中都会用到。任何领域对象都应该被设计为可复用的，因为领域对象可以高效的融入商业功能中并为今后的管理和复用提供了基础。

软件复用的益处

Measures

- 产品投放市场的时间：减为原来的1/2到1/5
- 缺陷密度：降低为原来的1/5到1/10
- 维护成本：减少为原来的1/5到1/10
- 整体开发成本：降低大约15%到75%
- 公司的复用率
  - 日本软件工厂的复用率已接近50%
  - HP公司的复用水平最高达83%
  - AT&T在电信运行系统中达40%至92%
  - 爱立信在数百个与不同客户有关的配置中达到90%
  - 摩托罗拉在编译器中达到了85%

复用的成本

Measures

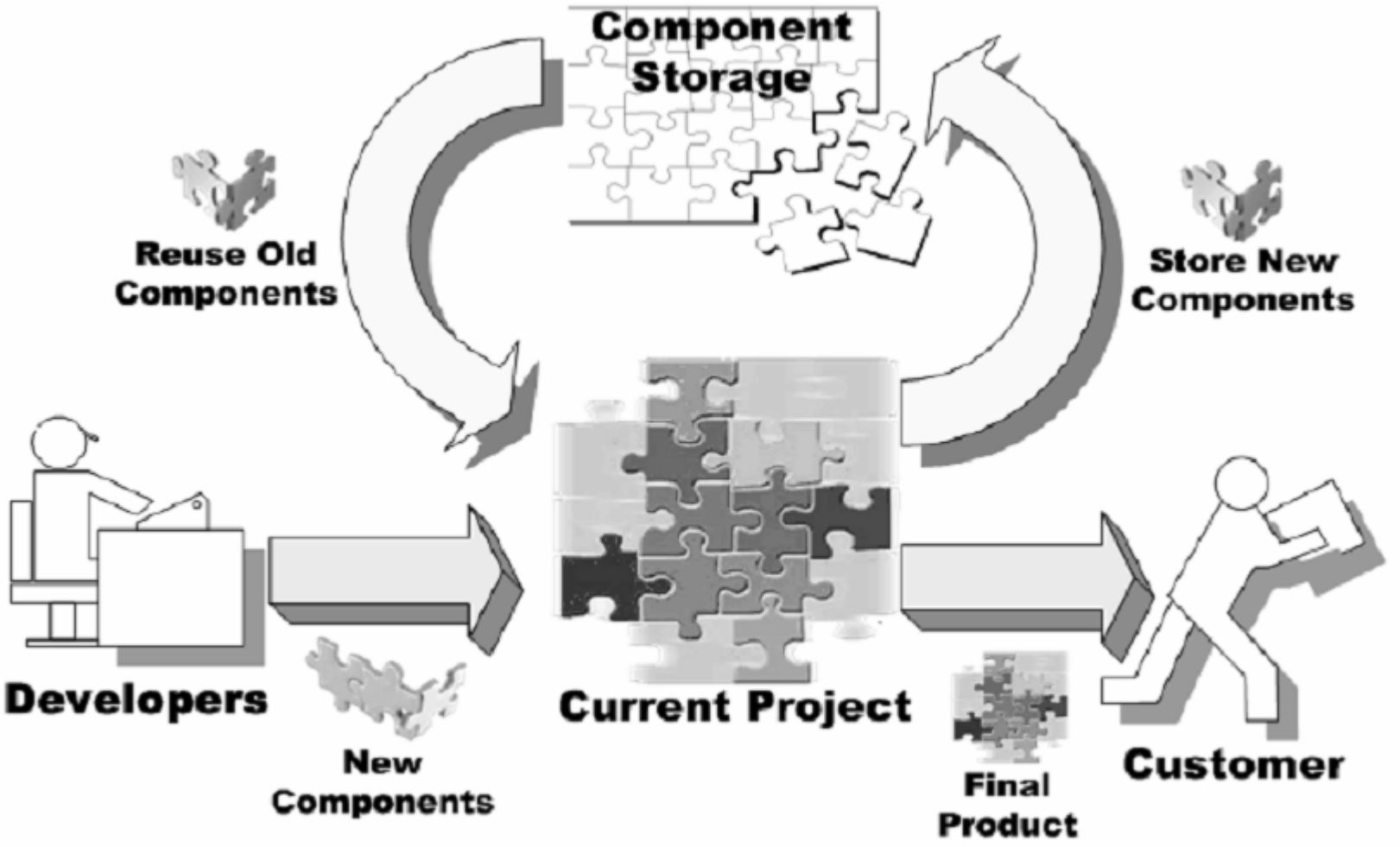
- 组件必须在应用系统项目中使用三到五次，可以收回最初创建和以后支持该组件所支出的成本
- 创建和支持可复用组件的成本是为单个应用系统实现类似组件成本的1.5到3.0倍
- 使用一个可复用组件的成本只是从头开发开发新组件成本的四分之一
- 需要要2到3个产品周期，一般要三年左右，才能收到显著的复用效益。

复用的障碍

Measures

- 很多人认为理论上可行，实践上困难
- 缺乏项目组之间的沟通
- 应用领域多变
- 缺少公司级的资金支持
- 构件的维护责任不明确、维护及时性差
- 人们不愿意使用别人开发的可复用构件
- 对软件开发的特点不了解

基于复用的软件开发


Component
Storage
Reuse Old
Components
Store New
Components
Developers
New
Components
Current Project
Final
Product
Customer

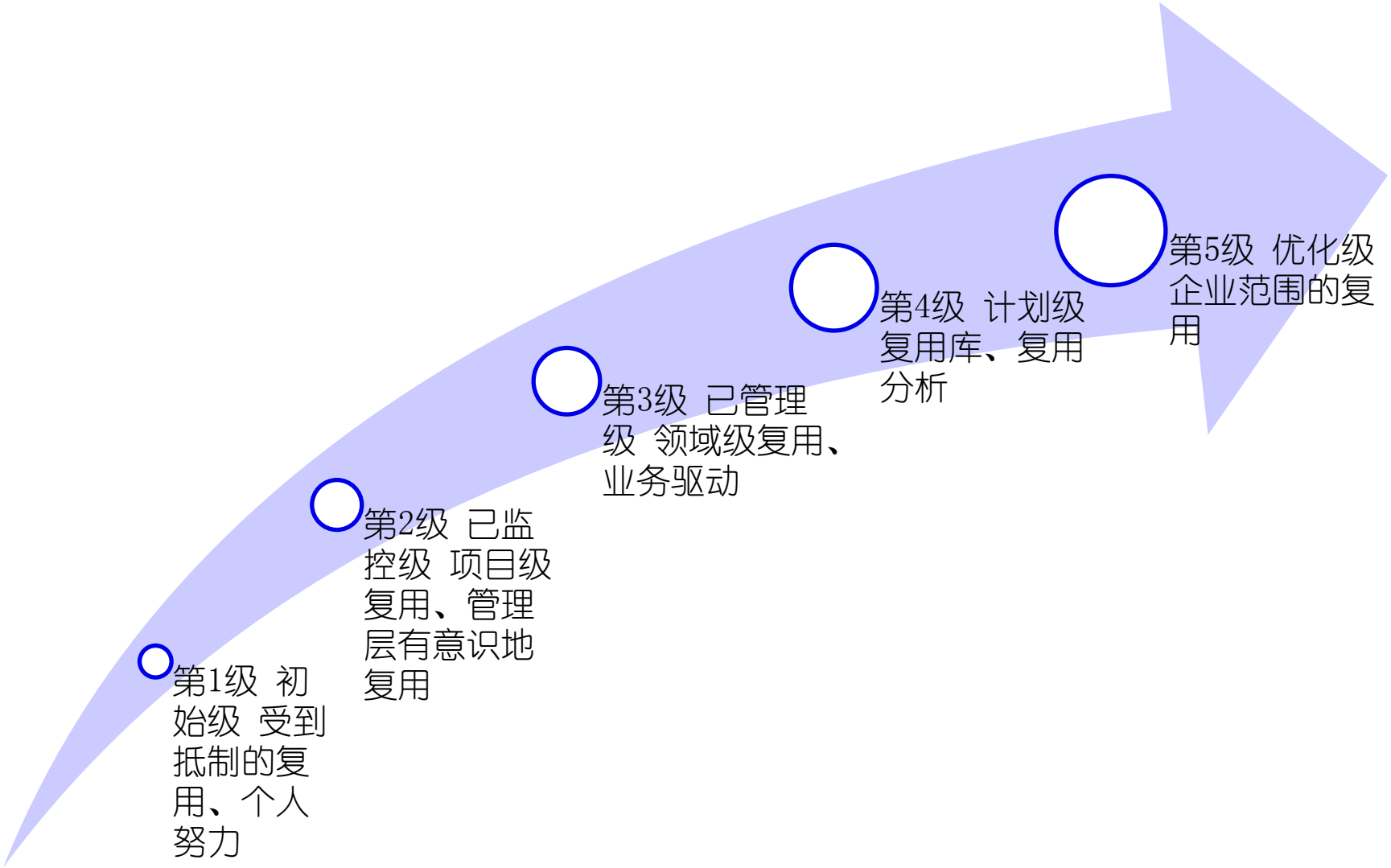
系统级复用

Measures

- 系统级复用是指按照良好定义的可重复过程进行复用的实践活动。
- 复用的实施从项目级别提升到企业级别，并进行了规范化

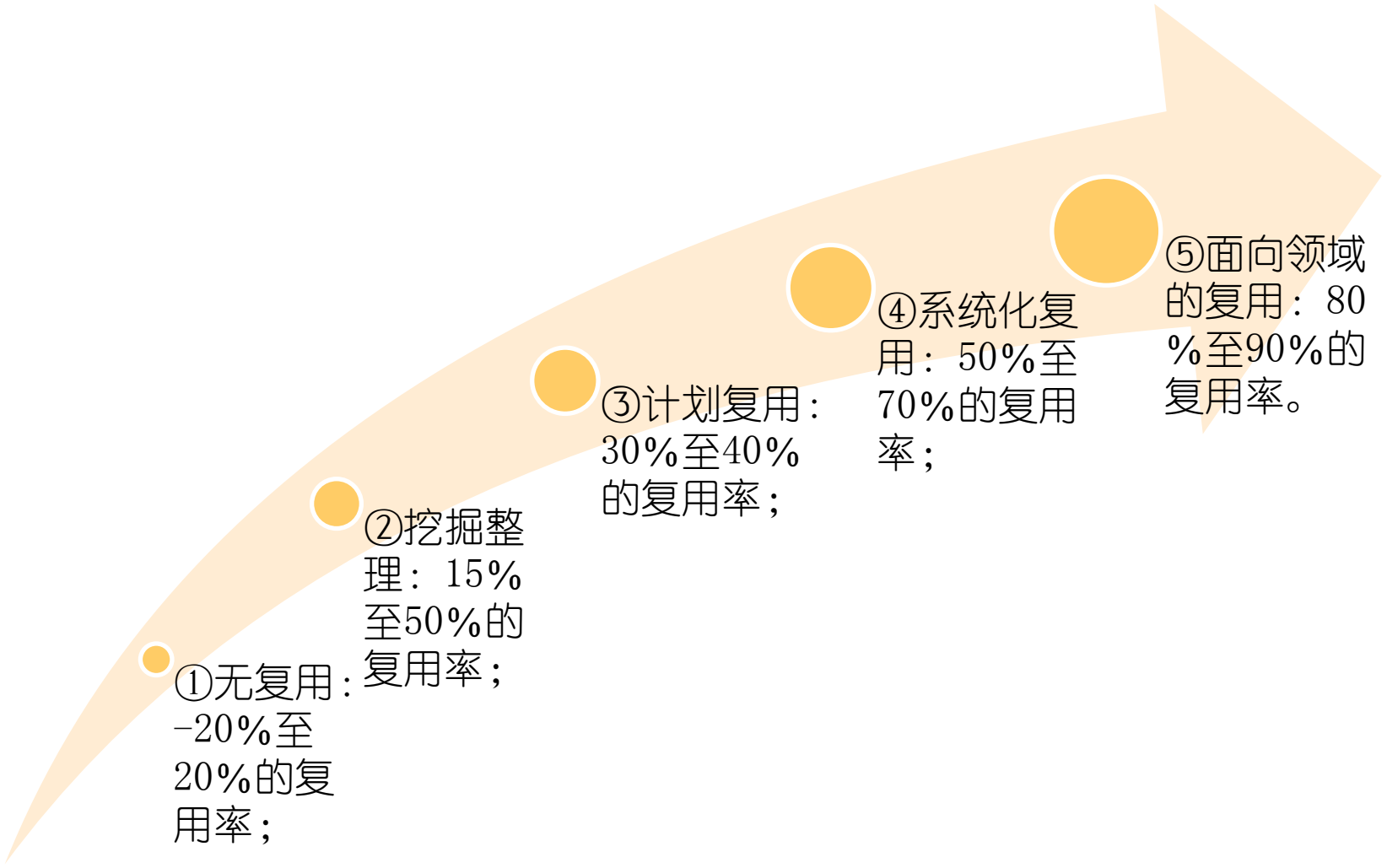
系统级复用的能力

Measures



HP的软件复用成熟度模型

Measures



非系统的复用

Measures

- 复用是无计划的，是软件生命周期隐含的副产品
- 复用是事后的想法，软件部件不是为复用而设计的，有时候软件需求为了屈从于已有的软件部件要妥协
- 有可能会有系统性能问题
- 有可能导致维护的复杂性

Measures

IEEE 1517

IEEE 1517标准

Measures

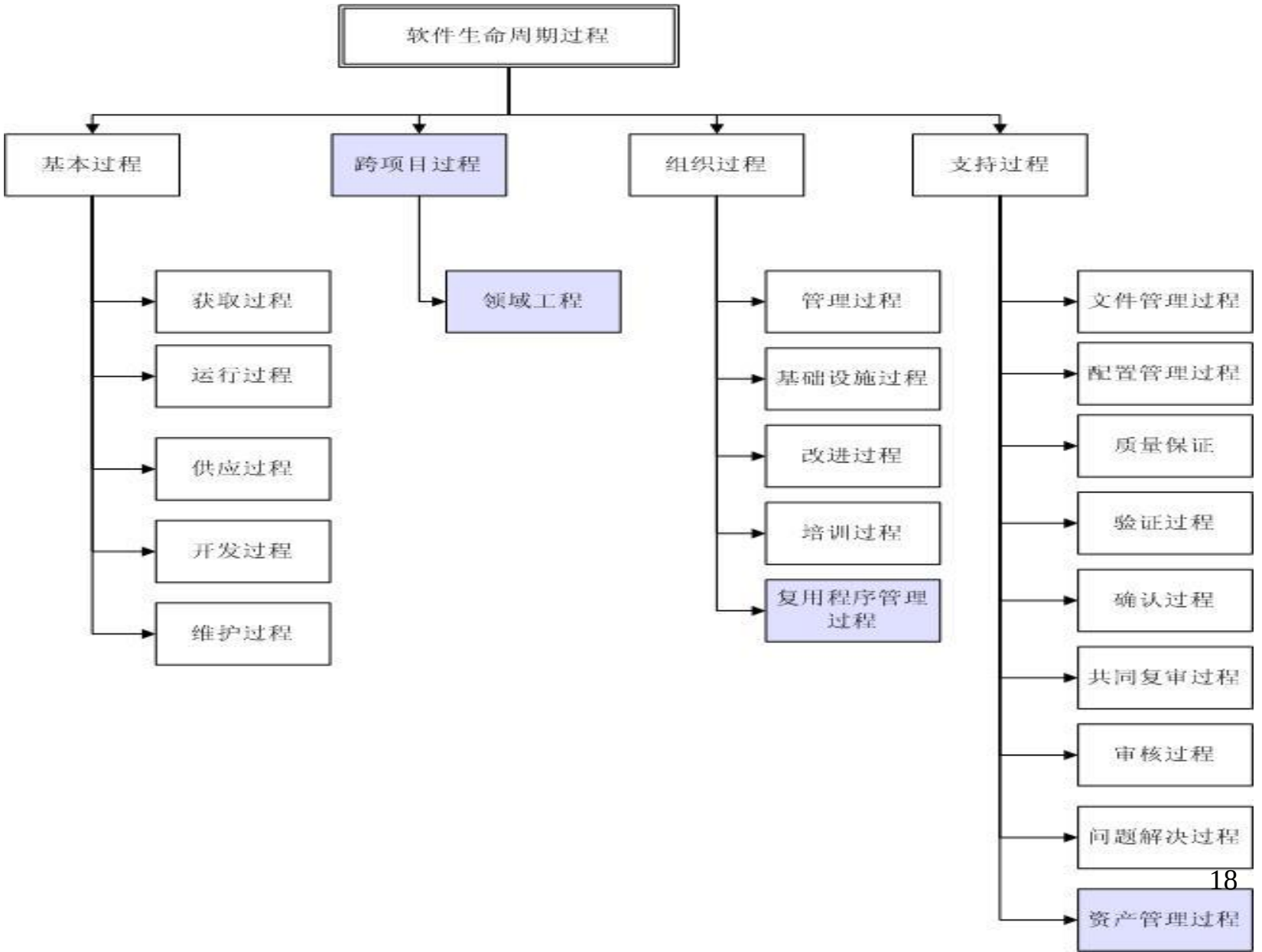
- IEEE 1517标准---信息技术-软件生命周期过程-复用过程标准
- 1999年7月发布,2004年重申
- IEEE 1517的宗旨;
  - 在软件生命周期中建立实践复用的框架
  - 指明开发或维护软件应用和系统时，支持系统复用实践所需要的过程、活动和任务的最小集合
  - 定义复用过程需要的输入和产生的输出
  - 解释如何在ISO/IEC和IEEE/EIA 12207标准软件生命周期框架中集成过程、活动和任务
  - 改进和阐明软件生产者和软件消费者之间就复用过程和复用术语的沟通
  - 提升和控制开发维护软件产品的软件复用实践
- 该标准并不坚持某个特有的严格的软件生命周期

IEEE1517标准的应用

Measures

- IEEE1517标准是为那些愿意获取、提供或开发软件应用和系统或者可复用资产的经理及技术人员所写的。IEEE1517标准适用于：
  - 软件和软件服务的获取
  - 资产的获取
  - 运用CBD（基于构件的开发）方法的软件应用和系统的提供、开发、操作和维护
  - 资产的提供、开发、管理和维护
  - 在组织或企业级别建立系统级复用程序和构件策略
- 在标准中可复用的软件构件、文档、指南等称为资产。

IEEE1517复用标准框架



在软件复用过程中涉及的角色与组织

Measures

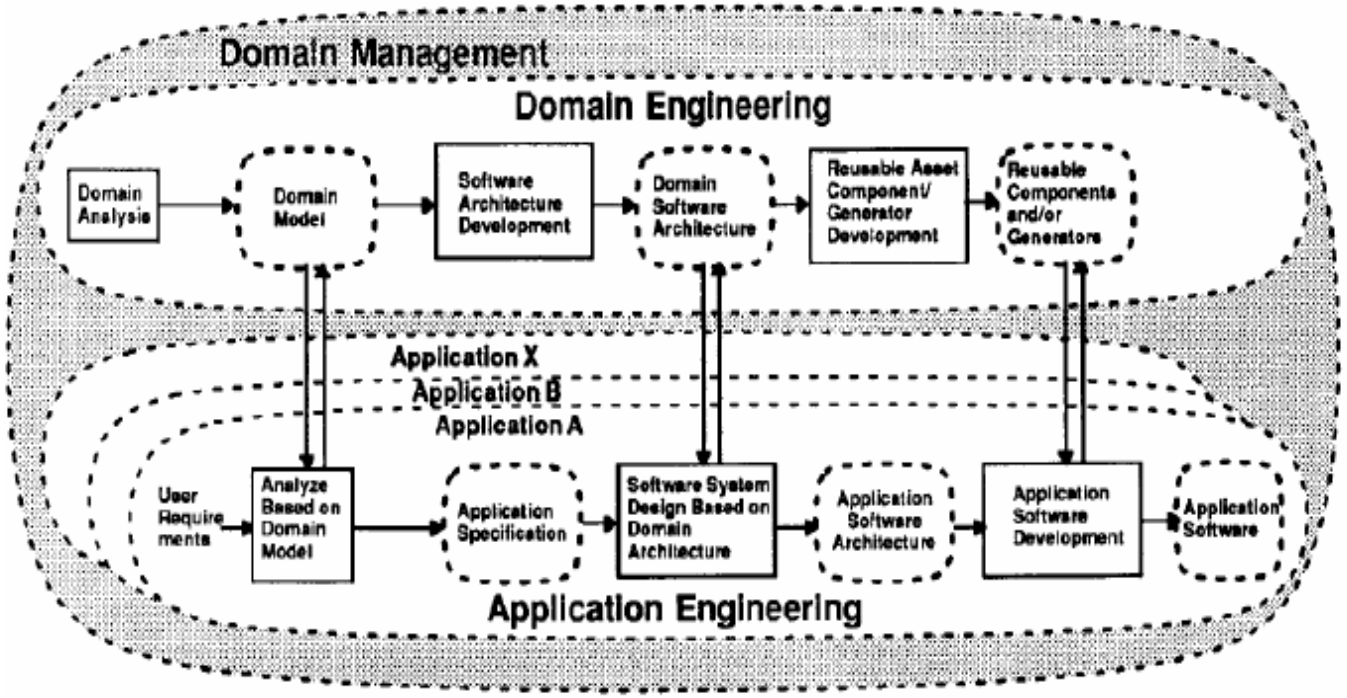
- 复用发起人：出资人，保证复用的资源等承诺
- 复用指导委员会：解决组织层面的复用支持，推广问题
- 复用程序管理者：复用战略项目的项目经理
- 复用程序支持小组：复用程序的执行者
- 复用带头人：复用的积极参与者
- 资产管理员：复用资产的管理人员
- 领域专家：负责领域建模
- 领域工程师：领域模型的设计人员
- 资产复用者：应用系统的开发人员
- 资产提供者：资产的开发人员或者采购人员

Measures

领域工程过程

领域工程

- 一个为一类系统、子系统或应用程序定义范围（即领域的定义）、阐明结构（即领域构架）、构建资产（如需求、设计、软件代码和文档）的基于复用的方法。领域工程可以包含下列活动：领域定义、领域分析、开发领域架构、领域实现

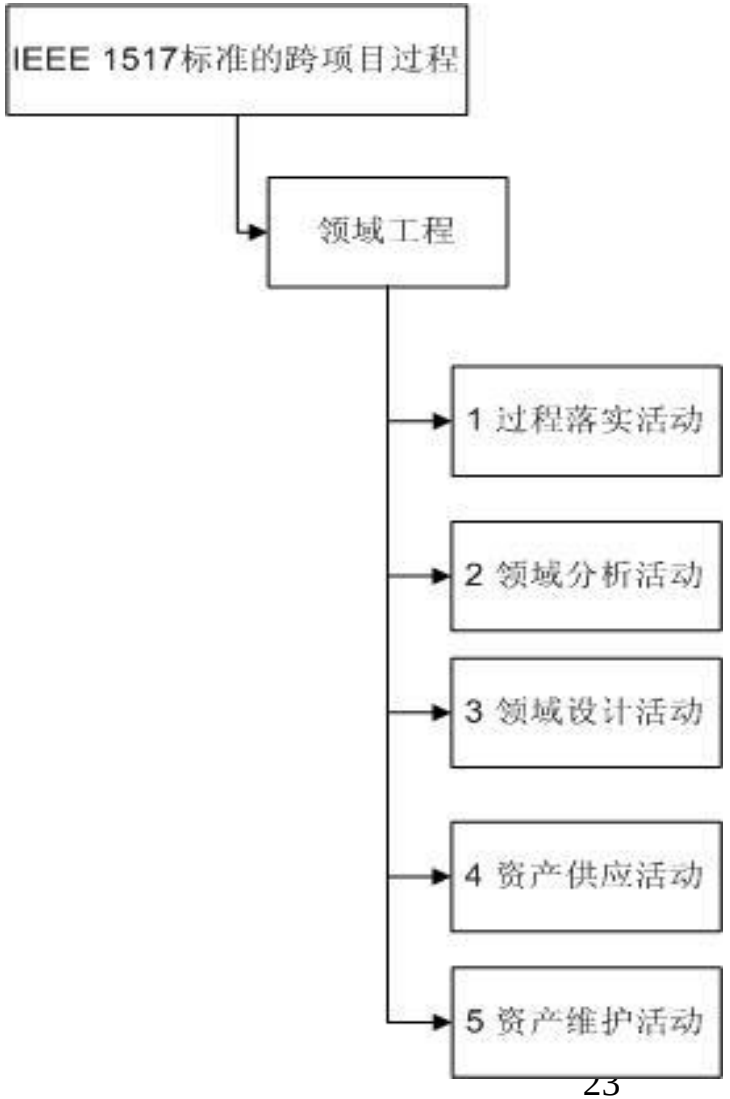


领域

- 一个问题空间
- 通过公共的属性刻画了一组系统，这些公共的属性可以被组织成一族可复用的资产
  - 公共属性
    - 公共的设计
    - 公共的性能或服务
    - 公共的技术
    - 公共的体系结构
- 如：
  - 飞行器的维护
  - 一个产品线
- 组织并不计划在每一个领域中实施复用，并且所有的领域也不具有相同的复用潜力

领域工程过程

- 制定领域工程计划
- 定义领域、领域词汇表和领域模型
- 提出领域架构和资产设计规约
- 开发或获得领域资产
- 维护领域资产



领域工程过程

Measures

活动名称	活动描述
过程落实	建立、文档化和执行领域工程计划 选择领域模型和领域构架的表现形式
领域分析	定义领域边界和其他领域的关系 表示领域软件开发人员的需求 构建、文档化、分类、评估和提交领域模型 构造、评价和提交领域词汇表
领域设计	建立、文档化、分类、评价和提交领域架构 开发、文档化和评估资产规约
资产供应	开发、文档化、分类、评估和提交资产
资产维护	分析资产修改的要求，选择资产修改的方案并使该方案获得批准，实施资产的修改，提交修改后的资产

领域工程活动的结果

Measures

- 1.1 领域工程计划
- 1.2 领域模型的表现形式
- 1.3 领域体系结构的表现形式
- 1.4 沟通过程
- 2.1 领域边界
- 2.2 开发者需求文档
- 2.3 领域模型
- 2.4 领域词汇表
- 2.5 领域分析评估文档
- 3.1 领域构架
- 3.2 资产规约
- 3.3 领域设计评估文档
- 4.1 资产
- 4.2 资产评估文档
- 5.1 资产修改实现可选方案
- 5.2 资产修改实现批准
- 5.3 资产管理员通告
- 5.4 修改后的资产

领域工程计划

Measures

- 该计划定义了执行领域工程的资源和过程。
- 该计划包括实施领域工程的
  - 标准
  - 方法
  - 工具
  - 活动
  - 责任

领域模型与领域架构的表现形式

- 领域模型：领域分析的产物，提供了领域需求的表示。
  - 领域模型标识和描述了领域中的数据结构、信息流、功能、约束和控制。
  - 领域模型描述了领域中不同系统之间在需求上的共性和特性。
  - 表现形式如：ER图、数据流程图、状态转移图、结构图、类模型、用例模型、交互模型
- 领域架构：领域中系统的一般性结构或设计。
  - 领域架构包括那些满足领域模型中所规约的需求的设计。
  - 领域架构是设计的文档，相反，领域模型是需求的文档。
  - 领域架构：
    - 1）能够通过一些少量的调整来生成领域中的软件系统设计；
    - 2）为单个软件系统提供配置资产的框架。
  - 表现形式如：过程交互模型、模块结构图

领域分析

- 领域分析的定义
  - A）对领域中系统的分析，以发现它们之间的共性和特性
  - B）一个过程，通过这个过程来标识、捕捉和组织开发软件系统时需要使用的信息，从而使得这些信息能在创建领域中的新系统时被复用
  - C） A)和B）中的过程的结果
- 领域分析活动对领域中已有的和预期的软件产品的特征进行分析、抽象，并对其建模，以确定这些产品在什么地方是相同的（共性），以及什么地方是不同的（特性）。这些信息被领域分析的过程中产生的一组领域模型所捕获。
- 领域模型的目的：
  - 辅助理解领域的关键共性元素和这些元素之间存在的关系
  - 定义领域词汇表，建立对领域的共同理解
  - 捕获领域中的关键共性和特性特征、性能、概念和功能

领域边界

- 基于功能、特征、属性和性能应该包括在这个领域中，哪些不应该包括在这个领域中。
- 上下文模型、数据流模型或对象模型可以被用来表示领域的边界。
- 数据流模型或结构图可以用来表示领域之间的关系。
- 用来表示领域边界的模型应该分别与所选的表示领域模型和领域架构的表现形式相同

领域词汇表

- 领域词汇表用以提供描述重要领域概念和领域中相似或通用资产之间的关系所需的术语
- 领域词汇表标识领域中软件产品之间存在的共性的基础
- 领域词汇表能使领域工程师和软件开发人员使用相同的语言进行交流
- 领域词汇表：
  - 概念
  - 关键词
  - 名词
  - 动词
- 领域词汇表应在创建领域模型的迭代过程中不断精华

领域分析活动-分类和文档化领域模型

Measures

- 8.1.2.5 领域工程师应该分类和文档化领域模型
- 分类：易于从复用库中查询和提取资产的资产组织方式
- 文档：
  - 资产名称：符合组织的命名规范
  - 资产功能简述：资产是做什么的
  - 使用资产的基础设施，包括技术环境需求、性能约束和法律约束
  - 测试，包括测试计划、测试用例、测试脚本、测试数据集和测试工具
  - 资产的质量（例如错误率）
  - 改善资产的建议
  - 资产使用的历史记录
  - 资产的分类信息
  - 复用的条件
  - 复用的方法

领域设计活动

Measures

- 领域设计活动的目标是生成领域构架和领域资产的设计规约。领域构架为使用资产构建软件产品提供了一个通用的、一般性的框架。如果领域模型可以被看做是理解领域共性的定义层面的设备，那么领域构架可以被看做是使用资产构建领域中的软件产品时使用的一种实现层面的媒介。
- 领域构架是将资产组装成软件产品的“胶水”。

领域构架

- 在领域构架中定义了构件的接口，以指导设计领域资产和软件产品
- 当创建这个领域构架时，做出了许多与系统运行相关的体系结构层面上的决定。当在领域软件产品的开发中复用构架时，就不需要再做这样的决定了。这样就可以加快开发过程，并能消除构建冗余软件部件的现象。
- 一个领域模型可能对应于不同的多个领域构架。如果领域中的软件产品需要不同的目标环境，那么为该领域开发多个领域构架是必要的。
- 领域构架文档
  - 为复用构架所做的适配调整信息，指导复用消费者为构建特殊的软件产品修改或特殊化领域构架
  - 什么时候可以复用构架，描述设计领域构架所需的特殊的硬件目标环境和特殊的开发工具
  - 领域构架的分类信息，使用领域资产管理员指名的分类模式对领域构架进行分类的信息

资产设计规约

- 资产设计规约的信息
  - 资产的功能
    - 资产期望它的客户提供什么
    - 资产能为客户产生什么
    - 资产的性能特性
    - 资产需要的共性和特性范围
    - 对资产所需目标环境的假设
  - 使用资产的限制
- 识别资产的准则
  - 资产被用于构建或维护某领域中的软件产品的可能的次数
  - 可以复用该资产的软件产品在商业战略上的重要性
  - 在这些软件产品中可能存在的相似性和差异性
  - 这些差异对资产复用潜力的影响，以及复用该资产的收益
  - 资产在复用中适应可能的差异和捕获可能的相似性的能力
  - 验证资产的复用性和整体质量的容易程度
  - 生成/再工程/获取资产的成本
  - 为了偿还其成本，资产需要被使用的次数
  - 使用资产提供的商业效益
  - 使资产适合领域构架的容易程度

资产供应活动

- 软件产品：一组计算机程序、过程、可能相关的文档和数据
- 资产：一种特殊的软件产品，资产具有多次使用的能力

开发或获取资产的一般性指南

Measures

- 尽量复用而不是新创造一个资产
- 符合命名规则
- 使用一致的设计风格
- 使用来自复用库/目录的模板建立新的资产，确保资产符合标准的格式，并确保其完整性
- 使用对象技术
- 限制资产之间的交互。资产应该“高内聚，低耦合”
- 了解组织的资产验证标准，生成资产时要尽可能符合这些标准
- 使每一个资产既满足当前需求又满足未来资产的需求
- 使资产尽可能具有鲁棒性，具有处理错误和异常条件的能力
- 使资产尽可能高效
- 遵循组织的编程、用户界面、数据库、测试和文档标准
- 文档化资产，使对资产不熟悉的人能够理解资产
- 捕获与设计相关的决策，建立资产设计与文档和代码的关联关系，提供所有可能和资产一起使用的相关资产的链接

资产维护活动

Measures

- 过程落实活动
- 问题和修改分析活动
- 修改实施活动
- 移植活动
- 软件引退活动

Measures

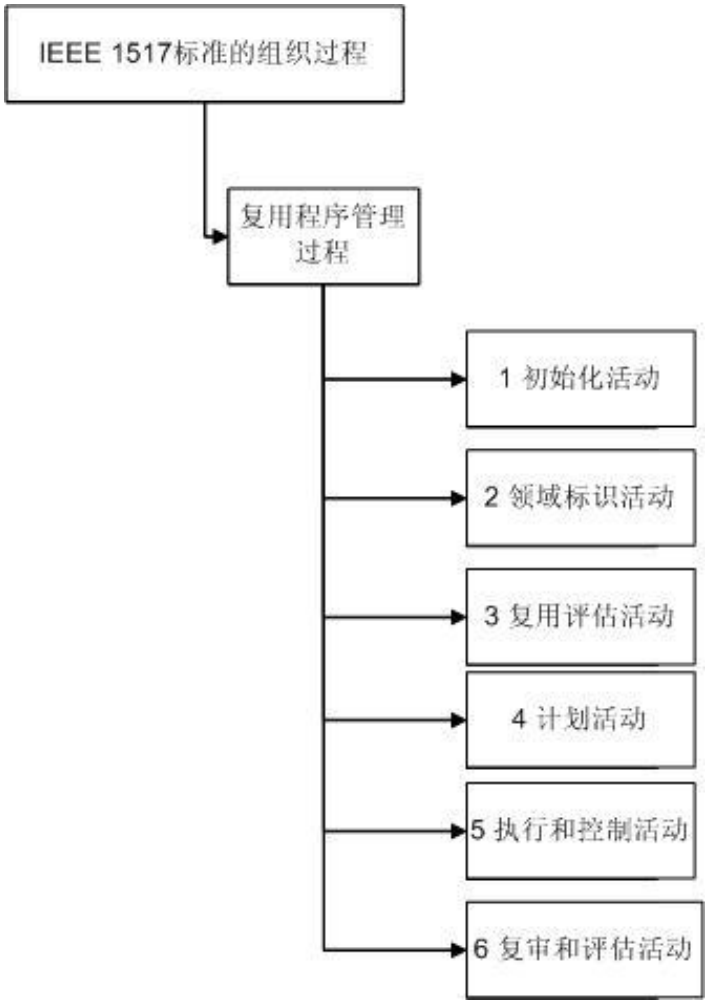
复用程序管理过程

复用程序管理过程

- 程序(program)
  - 实现特定目标的执行计划，是被遵循的活动执行的进度、工序等，是一组为某一特定目标、计划好的、相关的活动、工序
- 复用程序管理员
  - 负责建立、管理和支持系统复用实践的角色

复用程序管理过程

- 定义组织的复用策略
- 实施复用评估来决定组织的能力以进行系统级复用的实践
- 定义和落实复用程序实施计划
- 为复用程序建立一个管理上的和组织级别的支持机构
- 监控、评估和改进复用程序



复用管理过程中的活动

Measures

初始化	建立复用策略 指定复用发起人 标识复用程序的参与者/角色 建立复用指导机构 建立复用程序支持机构
领域标识	标识、评估和精化组织的复用领域
复用评估	评估组织实施系统级复用的能力 评估在组织的领域中实施复用的潜力 建议精化组织的复用策略和复用程序实施计划 改进组织的复用基础设施
计划	建立、文档化、评估、复审、批准和维护组织的复用程序实施计划
执行和控制	执行组织的复用程序以实施计划中定义的活动 监控复用程序的进度 报告复用程序的问题 重新确认管理层对复用程序的支持
复审和评估	定期评估组织复用程序的成果 报告复用程序的经验和教训 为改进组织的复用程序提出建议性的修改意见

复用程序

- 一个组织的复用程序应该通过建立该组织的复用策略而初始化，这个复用策略包括组织进行复用的目的、意图、目标和范围。复用程序应该包含下列元素：
  - 复用的发起人
  - 复用的基础设施（硬件、软件、工具、技术、标准、规则和实施复用所需的其他设备）
  - 复用的资金和其他资源
  - 复用程序的支持机构
  - 复用所需的交流、反馈和通告机制
- 复用的目标
  - 技术目标
    - 提高质量
    - 提高生产率
  - 商务目标
    - 缩短上市时间
    - 增加客户满意度
- 注意复用的成本
  - 创建一个复用构件将比创建单次使用的软件成分要多花费25%到13倍的成本
  - 除非复用程序拥有充足的资金支持，否则复用不会成功

复用发起人

- 复用发起人应该是管理层中的个人或者小组（级别越高越好），一个可能的选择是CIO。
- 复用发起人的职责
  - 建立共同的复用视角
  - 向组织介绍复用，鼓励并批准复用的实施
  - 决定企业如何处理任何需要为复用做出的、组织和文化上变化
  - 批准、通过、推广复用程序，并且为复用程序提供资源
  - 通过确定复用帮助企业达到商业目标的方式，来确保其作为一个商业策略而获得成功

复用程序的参与者

- 复用带头人 (reuse champion)
  - 负责对复用进行调查和试验，并向组织推广复用概念的个人或小组
  - 选择：
    - 被管理层和软件专业开发人员尊重的
    - 擅于沟通的
    - 在复用中训练有素的
    - 对复用感兴趣的个人或组织
- 复用程序管理员
  - 为组织执行战略上的复用计划
  - 负责定义、实施复用程序，并为正在进行的复用程序提供支持
  - 管理复用程序日常的细节
  - 为复用的基础设施分配职责
  - 度量和报告复用程序对组织造成的影响

复用指导机构

- 该机构的任务包含如下内容：
  - 调查组织中复用的实施情况
  - 标识组织中可能存在复用机会的范围
  - 分配组织中的复用职责
  - 重新定义组织的奖惩措施和文化以支持和鼓励复用
- 复用指导机构的成员
  - 复用发起人
  - 软件开发经理
  - 运行经理
  - 维护经理
  - 复用带头人
  - 复用程序管理员

复用程序支持机构

- 复用程序支持机构的职责应该包括以下内容：
  - 参与建立和实施复用程序实施计划
  - 标识、文档化复用策略，并向所有的复用程序参与者转达复用策略
  - 推广复用实践，鼓励促进复用的软件文化
  - 在现在或未来的软件项目中找到实施复用的机会
  - 建立和维护复用的基础设施
  - 为实践复用的软件项目提供复用的咨询支持
- 复用程序支持机构的目的是使得组织中对复用的实践变为主动的行为。复用程序支持机构是将那种民间自发的纸上谈兵式的复用转变为正规的，被组织官方认可的系统级复用的方法

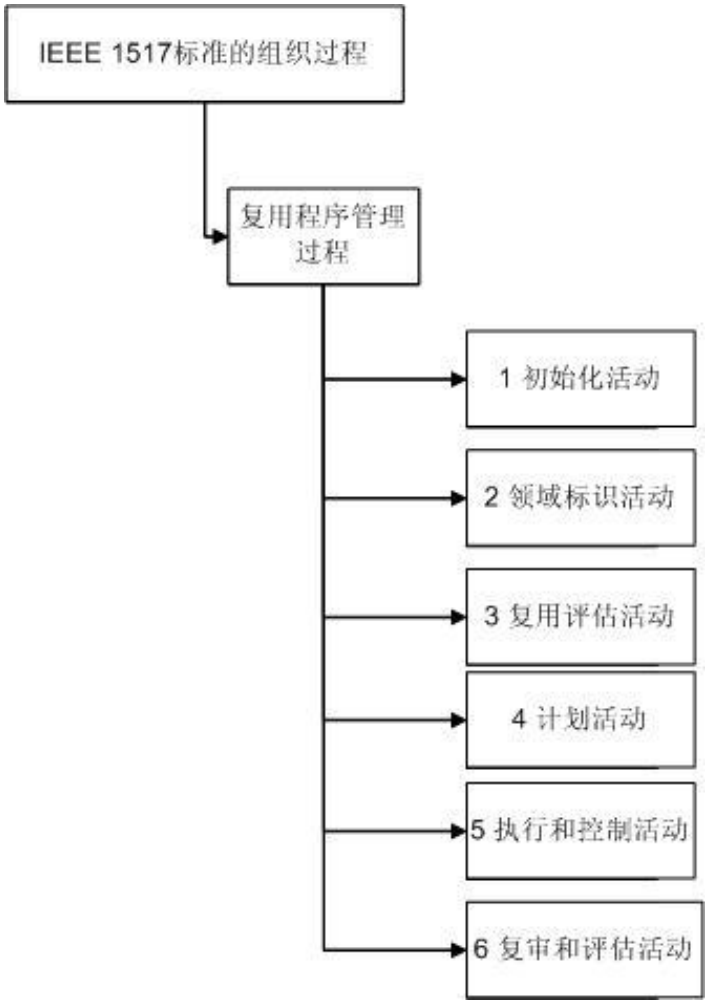
复用程序实施计划的内容

Measures

a) 发行的日期和状态
b) 范围
c) 发行的组织
d) 参考文献
e) 正式批准的授权
f) 被计划的活动和任务
g) 宏观参考（该计划涉及的政策或者法律）
h) 微观参考（其他计划或本计划详细阐述的任务描述）
i) 进度表
j) 预算
k) 资源和它们的分配
l) 义务和权利
m) 风险
n) 质量控制措施
o) 成本
p) 角色之间的接口
q) 环境/基础设施（包括安全需求）
r) 培训
s) 术语表
t) 变化过程和历史记录

复用程序管理过程的活动成果

- 1.1 复用策略
- 1.2 复用发起人
- 1.3 复用程序参与者和角色
- 1.4 复用指导机构
- 1.5 复用程序支持机构
- 2.1 领域
- 2.2 领域评估文档
- 3.1 复用能力评估文档
- 3.2 复用评估文档
- 3.3 复用建议文档
- 4.1 复用程序实施计划
- 4.2 复用程序实施计划评估文档
- 5.1 复用程序实施计划调整文档
- 5.2 复用程序问题记录
- 6.1 复用程序结果



Measures

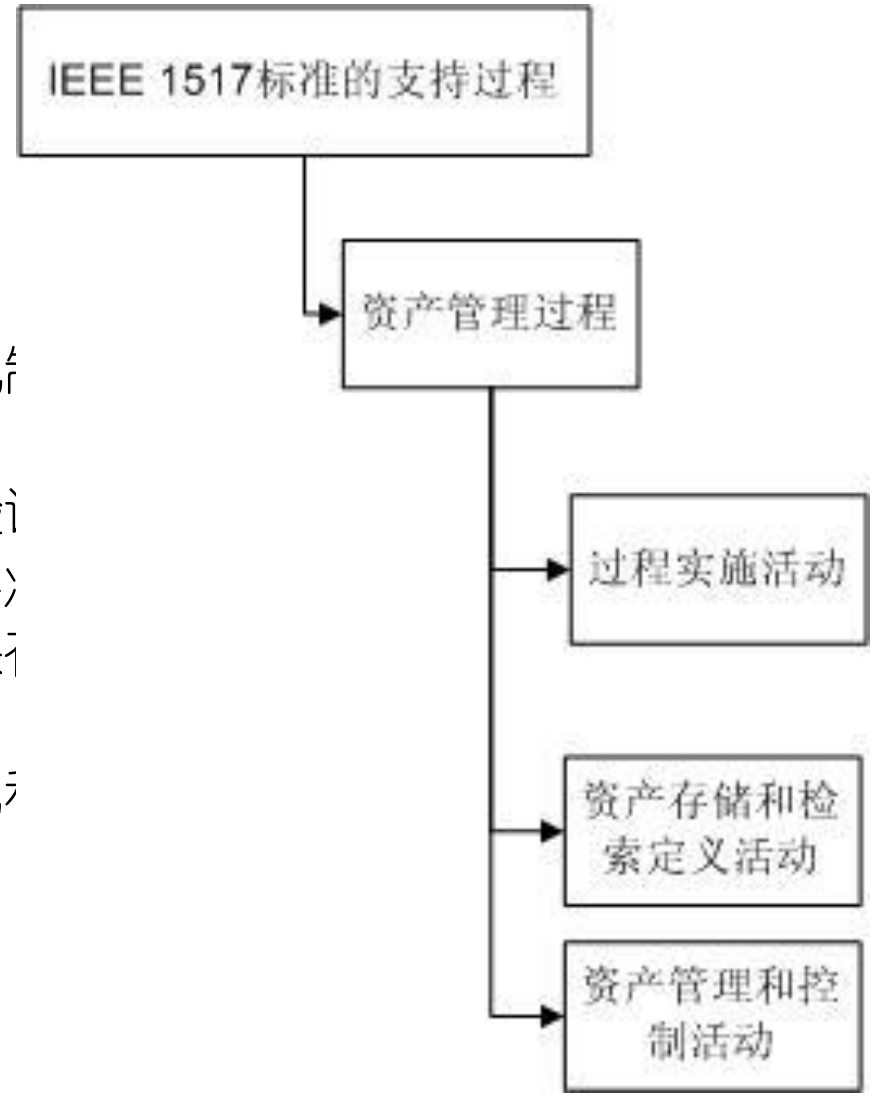
资产管理过程

资产管理过程

- 资产
  - ——一个诸如设计、规约、源代码、文档或者测试过程指南等的项目，它被设计为可以在多种上下文环境中使用。
- ……一切贯穿资产生命周期的管理和技术过程，如标识、详细说明、验证、分类和建立基线等，跟踪资产的修改记录、变迁记录和版本信息，记录和报告资产的状态，控制资产的存储和操作，向资产的复用消费者发送资产，以及删除资产等。

资产管理过程

- 建立和实施资产管理计划
- 事实和维护资产保存和检索机
  库）
- 建立和维护资产分类模式和验
- 评估和认可新的后选资产，并
  新版本或新的版本加入资产保
  构中
- 管理资产存储、跟踪使用情况
  决问题



资产管理活动列表

Measures

活动的名称	活动的描述
过程落实	制定、文档化和复审一个资产管理计划
资产存储和检索定义	定义、实现、复审和维护一个资产存储和检索机制 开发、文档化和维护一个资产分类模式
资产管理和控制	评估资产 接收资产并将其放入资产存储和检索机制中进行管理 分类资产 进行资产配置管理 跟踪和报告资产的使用 监控、记录资产修改需求和问题报告，并且向领域工程师和资产使用者报告 从资产存储和检索机制中引退资产

资产管理计划

Measures

- 资产管理计划应该包含下列内容：
  a) 定义资产存储和检索机制的需求
  b) 定义资产存储和检索机制
  c) 把资产存储和检索机制作为软件生命周期中的一个完整的部分建立起来
  d) 为负责管理与维护资产存储和检索机制的组织命名
  e) 定义资产的接收、认证和引退过程
  f) 定义资产管理员和其他成员（如开发者、维护者、领域工程师）之间的关系
  g) 促进资产存储和检索机制的使用
  h) 定义资产管理沟通机制
  i) 定义资产分类模式

复用库小组的职责

Measures

- 建立资产存储和检索机制
- 建立资产存储和检索机制的访问方法
- 决定哪些资产可以存储在资产存储和检索机制中
- 管理资产存储和检索机制
- 定义资产消费者的界面接口

资产管理过程的活动成果

1.1 资产管理计划

1.2 资产管理计划的评估报告

2.1 资产存储和检索机制

2.2 资产分类模式(可选)

2.3 资产存储和检索机制的评价报告

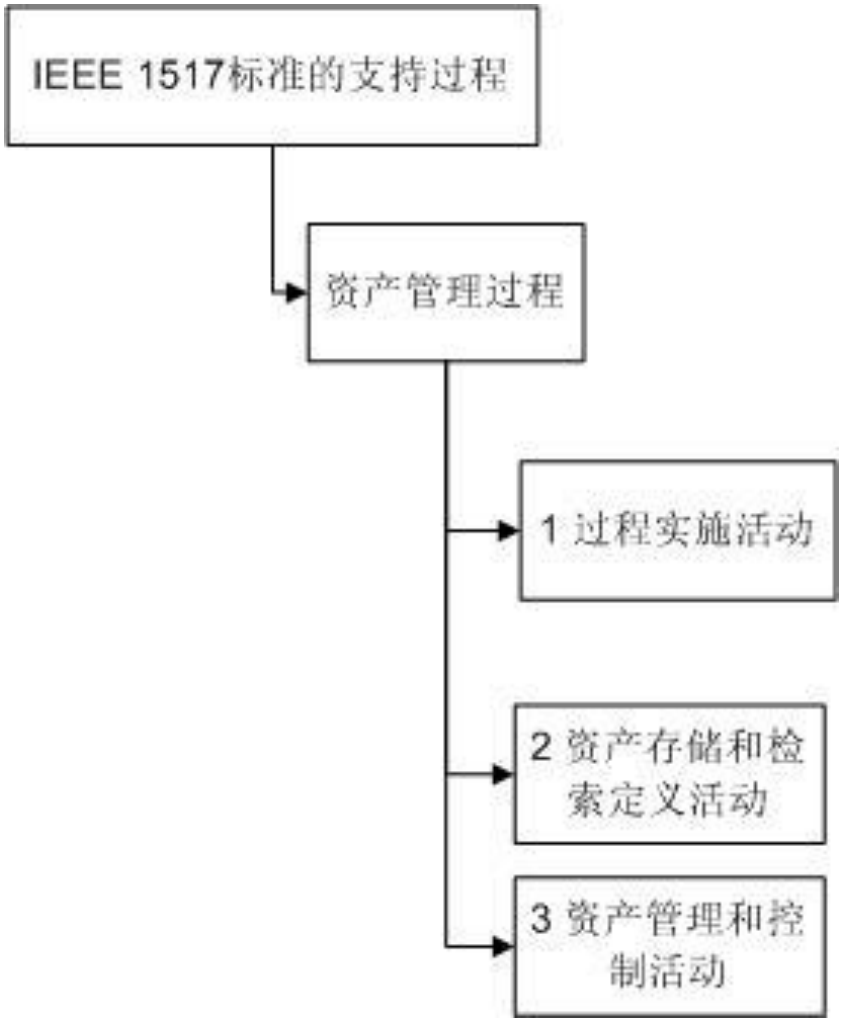
3.1 资产的评估报告

3.2 已分类的资产

3.3 资产使用报告

3.4 资产修改需求和问题报告

3.5 资产通告



Measures

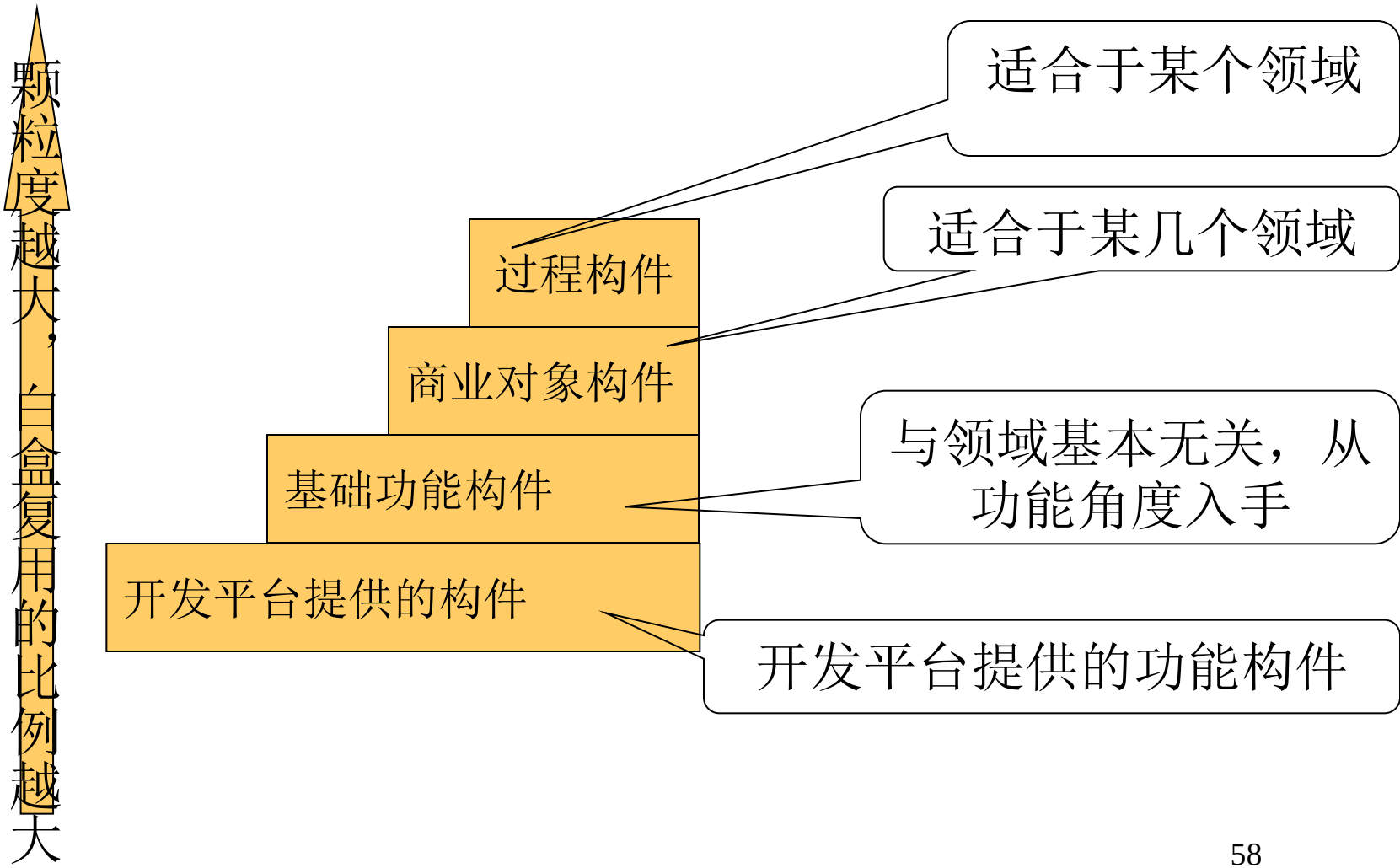
软件复用的经验与教训

软件复用成功的最佳实践

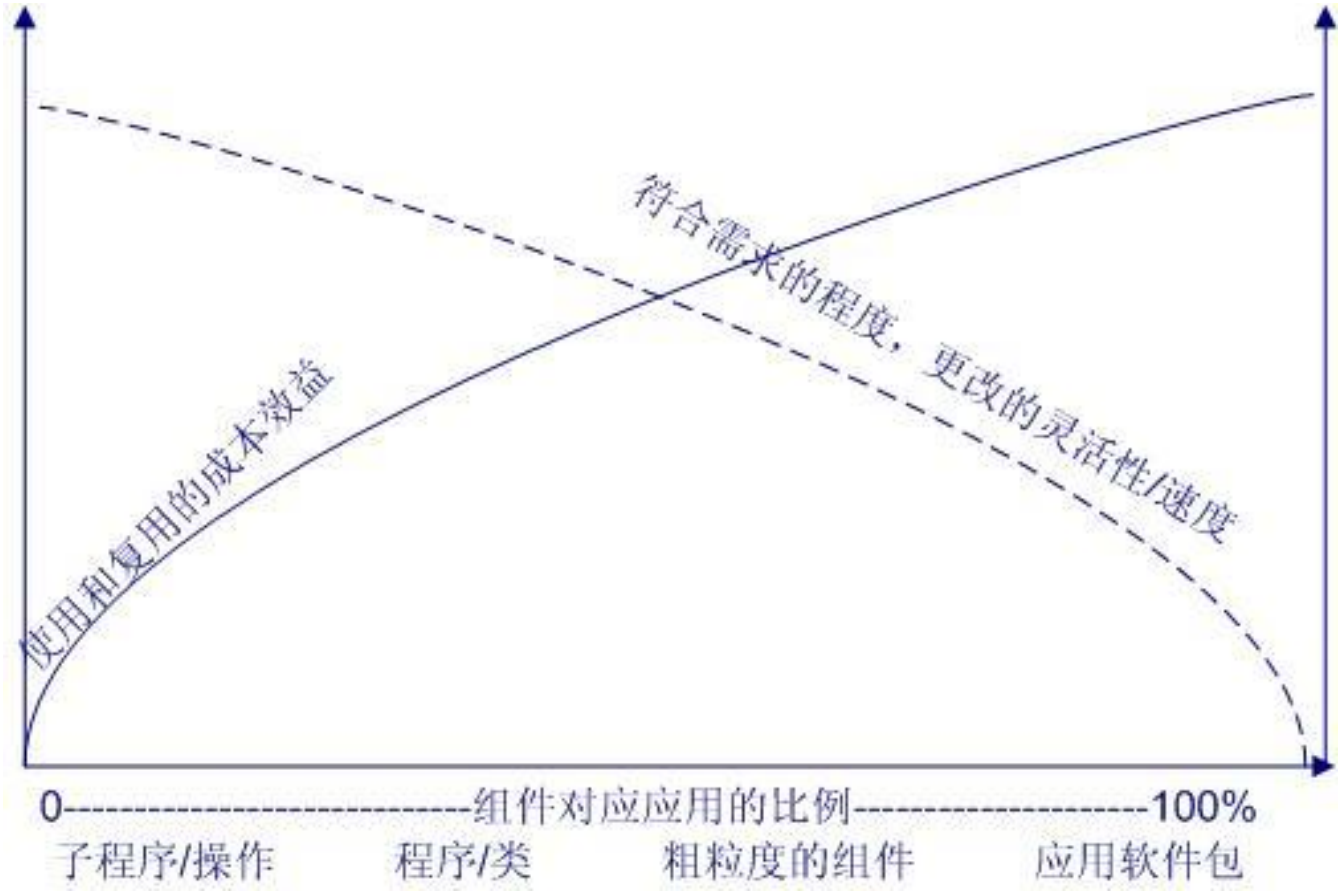
Measures

- 管理层的长期支持与组织的大量投资
- 设计良好的体系结构
- 通过设计可复用的小系统实现大系统
- 设计开发组织的结构以匹配系统体系结构
- 在技术评审中发现可以复用的构件
- 建立软件复用的榜样
- 要在实际工作环境中创建和改进可复用组件
- 规划并渐进地修改系统框架、开发过程和机构以满足复用的需要
- 专门的组件开发团队、管理团队
- 仅有对象或组件技术是不够的
- 重点考虑高回报应用系统中的复用
- 复用需要涉及到软件组织的奖励机制.
  - 鼓励复用构件的使用者, 构造者
- 通过指标度量复用进展，并优化复用程序

复用的层次


颗粒度越大，白盒复用的比例越大
过程构件
商业对象构件
基础功能构件
开发平台提供的构件
适合于某个领域
适合于某几个领域
与领域基本无关，从功能角度入手
开发平台提供的功能构件

复用的颗粒度与效益、灵活性的关系

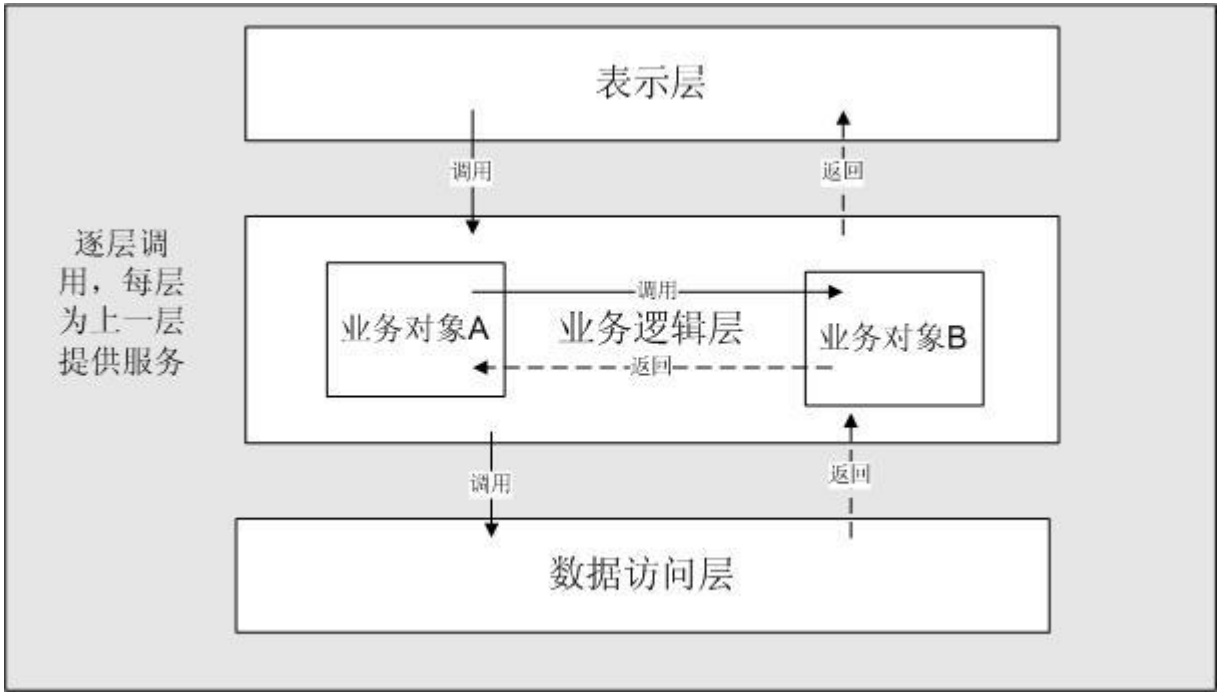


可复用构件的分类

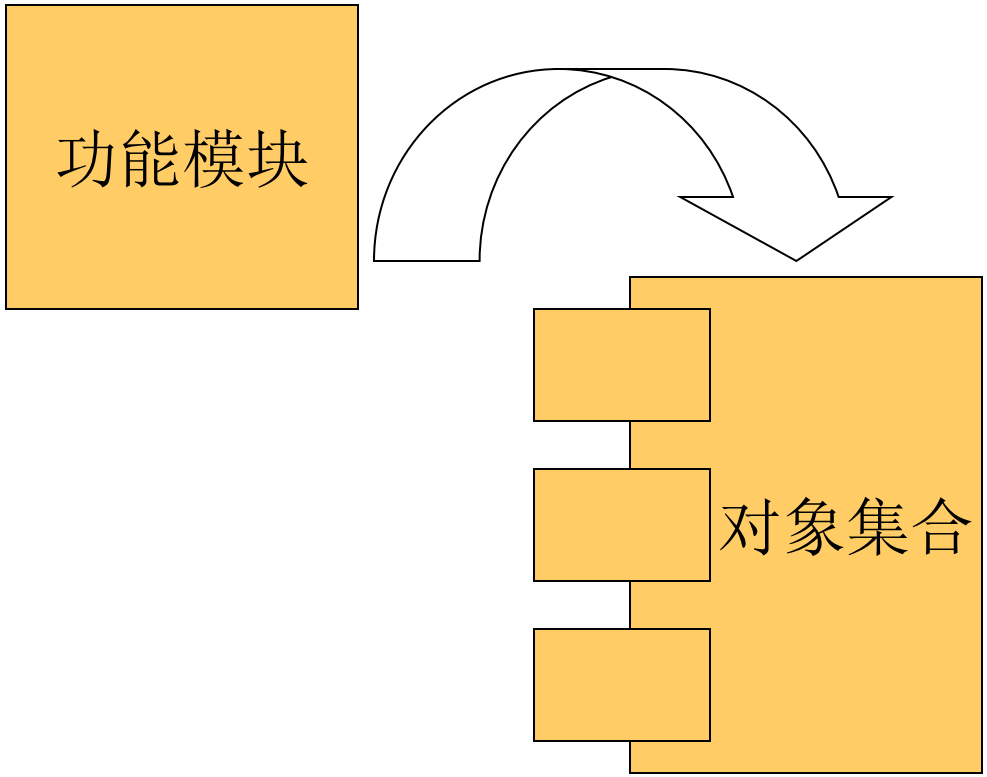
Measures

- 与数据库系统的相关性
- 与表结构的相关性
- 与存储方式的相关性
- 与开发工具的相关性
- 与标准的相关性
- 与业务领域的相关性
- 所属于的层次
- 数据的私有性
  - 没有私有的永久数据
  - 有私有的永久数据
  - 有共享的数据

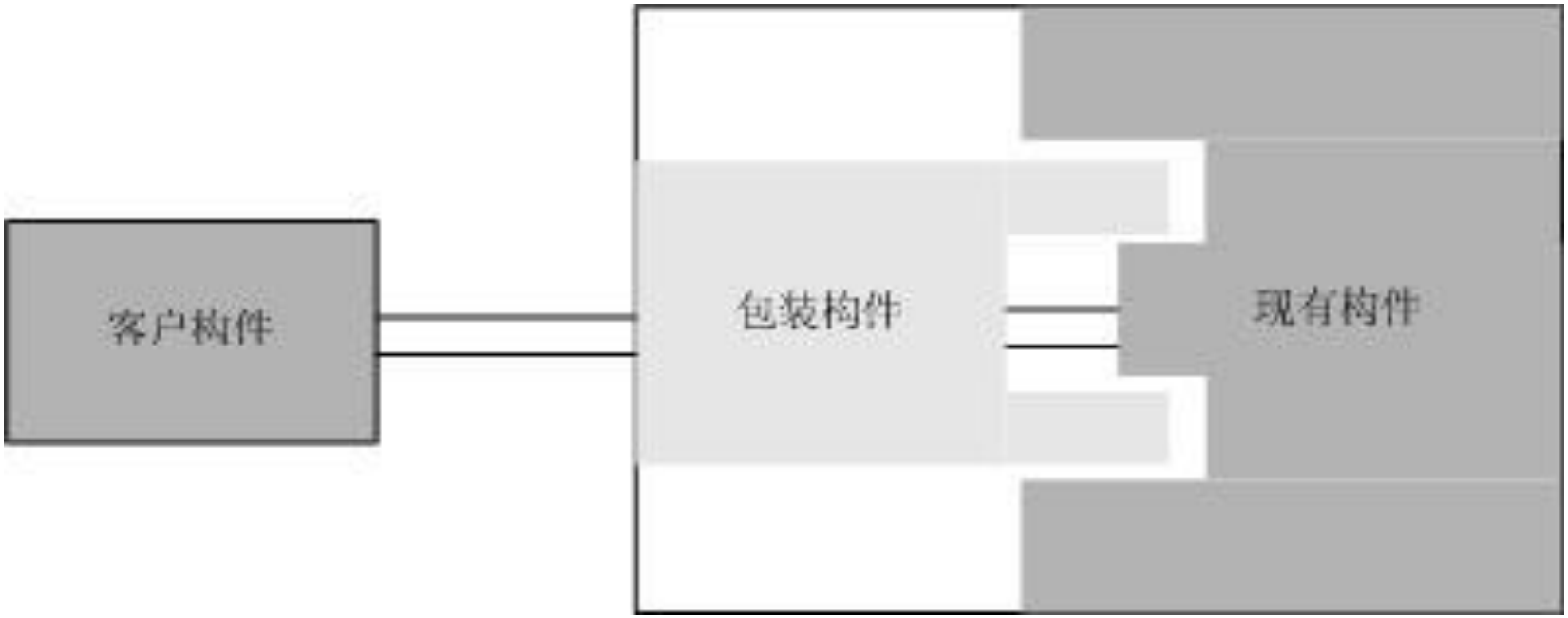
复用的策略：采用分层思想



复用策略：采用封装的思想实现

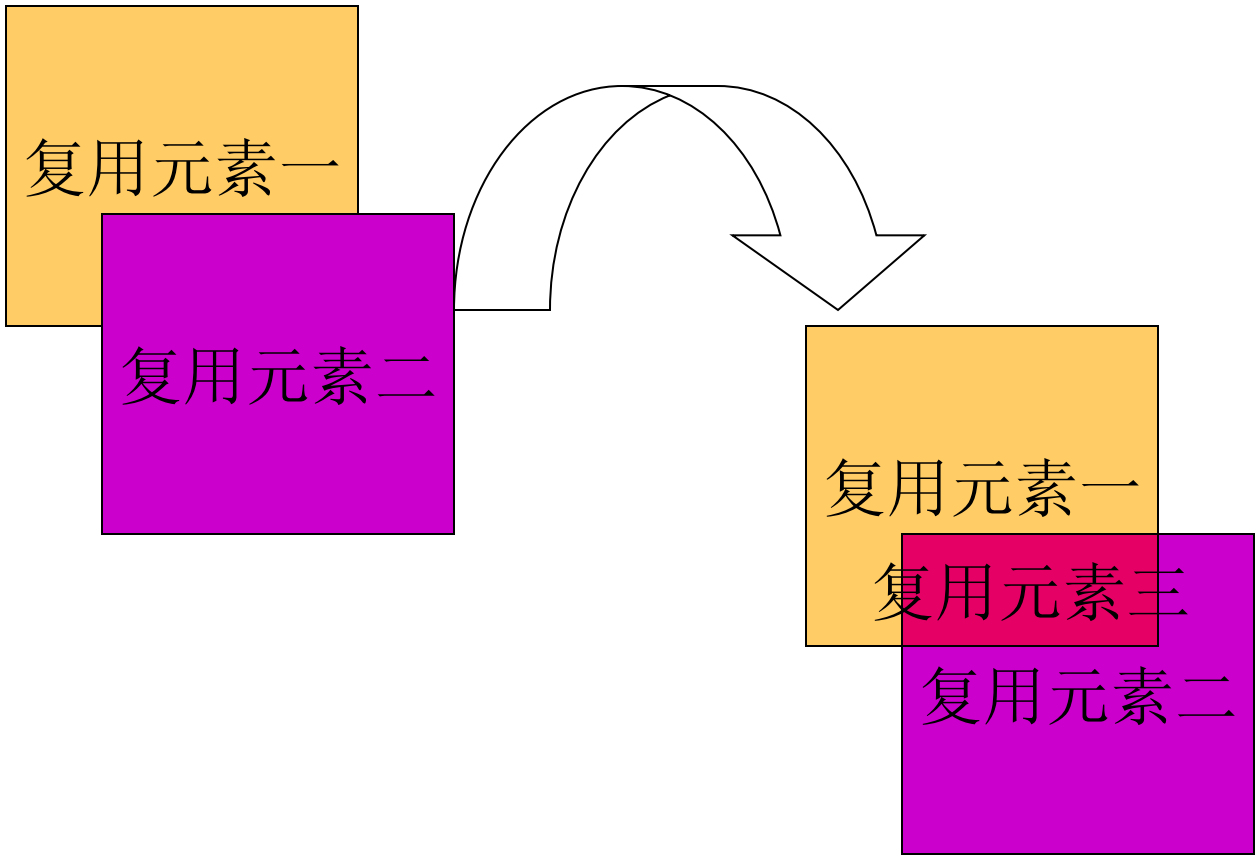


复用策略:包装的方法

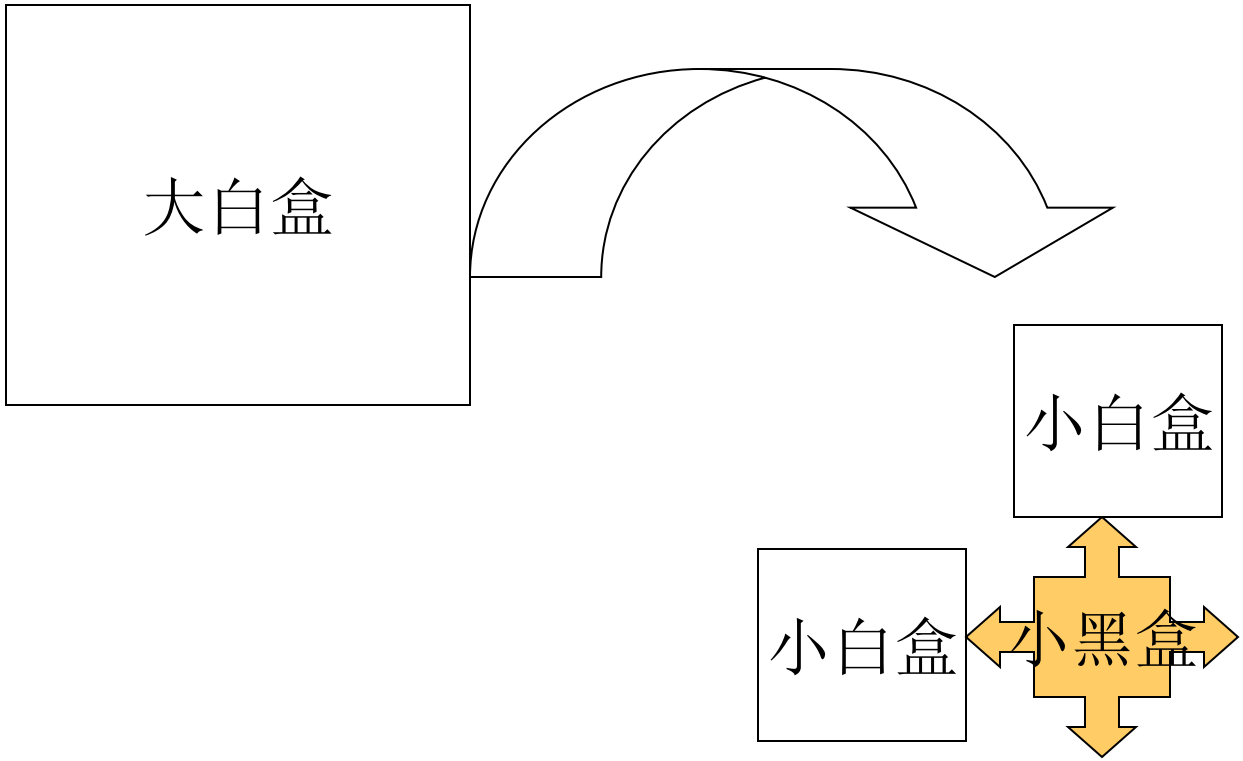


复用策略:适配器

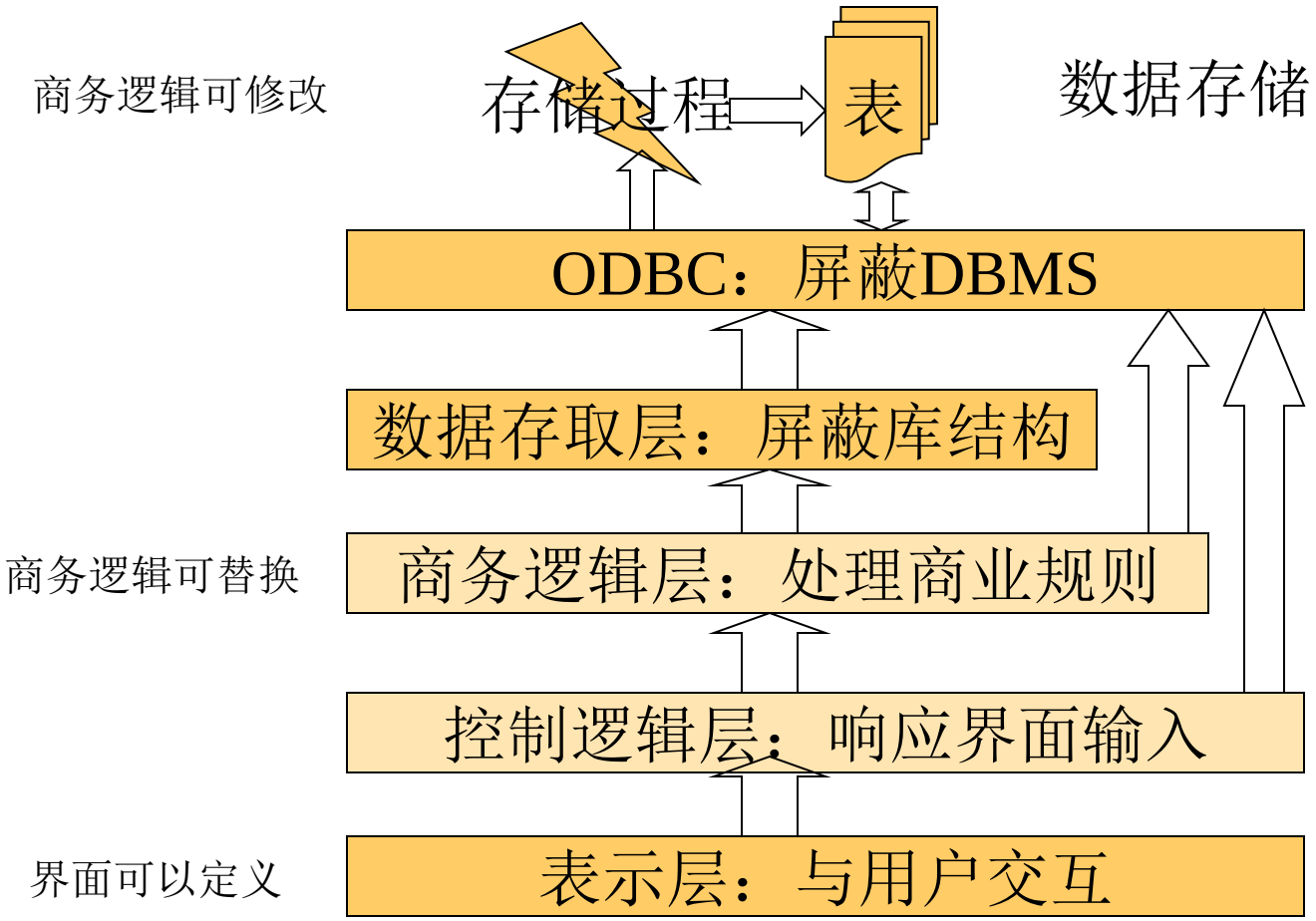
复用策略：将原来的复用元素提取更复用的元素



复用策略：将原来的白盒复用拆分



复用策略：合理地安排变化的位置



复用策略：采用设计模式与分析模式

Measures

- Pattern是在特定环境下对特定问题的解决方案
- Pattern的描述方法：
  - 问题
  - 环境
  - 解决方案
  - 效果

GOF模式列表

目的 / 范围	创建型	结构型	行为型
类	工厂方法	适配器(类)	解释器 模板方法
对象	抽象工厂 生成器 原型 单例	适配器(对象) 桥接 组合 装饰 外观 享元 代理	职责链 命令 迭代器 中介者 备忘录 观察者 状态 策略 访问者

GoF主要由Erich Gamma、Richard Helm、Ralph Johnson和John Vlissides等收集整理，其中包含了23种设计模式，并基于两条准则将这些模式组织成不同的分组。

信息系统中常见的可复用构件

系统管理	权限管理
	用户管理
	用户登录
	操作日志
	数据日志
	数据库的备份与恢复
	驱动菜单
打印显示输出	一般打印
	套打构件
	报表构件
	图形分析构件
数据录入	单表字典录入
	多表字典录入
	单据录入

查询输出	条件查询
	统计分析
	查询格式的构件
	帐本查询
数据管理	内码机制
	删除异常处理机制
	加解锁的机制
其他构件	日历
	计算器
	编码帮助
	联机帮助
	Todo Lists
	会议管理

复用组件的前提

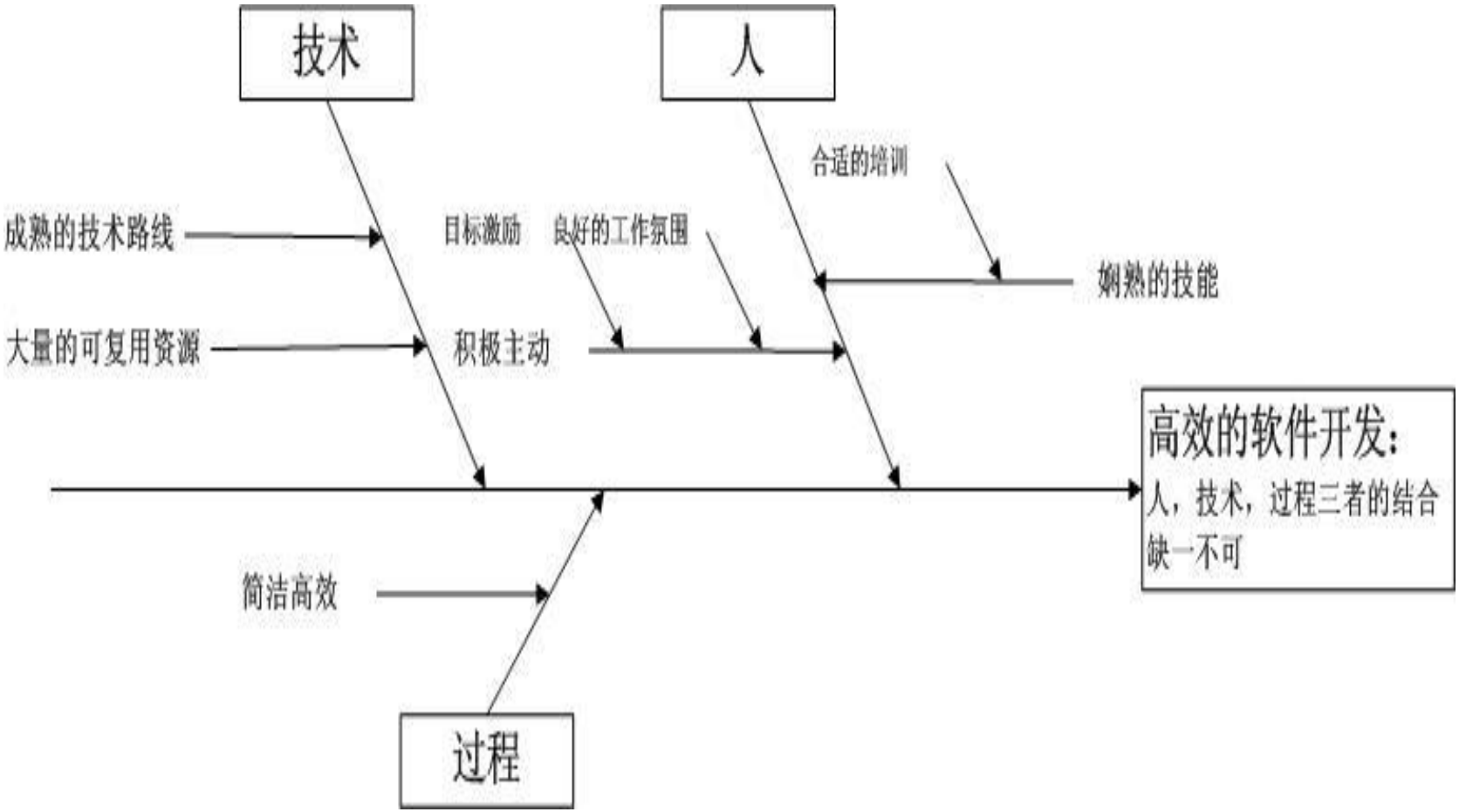
Measures

- 代码组件经过了单元测试与系统测试
- 代码组件经过了代码审查
- 代码组件经过了现场测试

小结

- 软件复用的基本思想
- IEEE1517软件复用标准
- 软件复用的经验与教训

过程、技术和人


技术
人
合适的培训
成熟的技术路线
目标激励
良好的工作氛围
娴熟的技能
大量的可复用资源
积极主动
高效的软件开发：
人，技术，过程三者的结合
缺一不可
简洁高效
过程

Measures

谢　谢！