

软件复用导引

爱立信上海研发

参考书

-) 软件复用:结构、过程和组织
 - 雅各布森 (Jacobson, I.) 等著
 - 译者: 韩柯
 - 机械工业出版社, 2003





- 软件复用技术:在系统开发过程中 考虑复用
 - Carma McClure 著
 - 译者: 廖泰安 安志远 沈升源
 - 机械工业出版社, 2003
 - SOFTWARE REUSE: METHODS MODELS AND COSTS
 - By Ronald J. Leach, 2011



内容



- > 软件复用的基本概念
- , 软件复用的实施: 在软件过程中复用
- > 软件复用技术综述



软件复用的基本概念



软件复用的提出



,软件危机问题

-60年代的"软件危机"使程序设计人员明白难于维护的软件成本是极其高昂的,当软件的规模不断扩大时,这种软件的综合成本可以说是没有人能负担的,并且即使投入了高昂的资金也难以得到可靠的产品。

> 软件复用的主要思想

- 将软件看成是由不同功能部分的"组件"所组成的有机体,每一个组件 在设计编写时可以被设计成完成同类工作的通用工具,这样,如果完 成各种工作的组件被建立起来以后,编写一特定软件的工作就变成了 将各种不同组件组织连接体来的简单问题,这对于软件产品的最终质 量和维护工作都有本质性的改变。

软件复用导引 | Page 6

初期强调节省成本 提高软件质量

阻碍: training, costs, technical difficulties, and psychological resistance Costs that are explicitly reuse-related include acquisition, analysis, component selection, measurement, certification, and maintenance costs.

什么是软件复用



- > 《软件复用技术:在系统开发过程中考虑复用》:
 - 软件复用(software reuse)是一种预先构造好的,<u>为复用目的</u>而设计的<u>软件构</u> 件来建立或者组装软件系统的<u>过程</u>。
-)《软件复用:结构、过程和组织》
 - 系统化的软件复用概念相当简单,开发尺寸合适的组件系统,并复用这些组件,然后扩展"组件系统"的思想,使其超出单纯的代码范畴,扩展到需求,分析模型,设计和测试。软件开发过程的所有阶段都可"复用"。
- > 《SOFTWARE REUSE: METHODS, MODELS, AND COSTS》
 - In general, the term "software reuse" refers to a situation in which some software is used in more than one project.
 - The term "software artifact" is frequently used in this context. A software artifact
 is some component of a software system at some phase of the software life
 cycle.
 - › software artifact: 工件, 软件单元资产

什么能被复用



- › 软件开发过程的提交物, 一切皆可复用
 - 体系架构
 - 需求
 - 开发环境
 - 设计
 - 算法
 - 模块
 - 用例
 - 代码
 - 接口协议
 - 接口转换系统
 - 组件/构件
 - 测试分析/用例/数据
 - 文档
 - COTS (Commercial Off-the Shelf)
 - . . .

软件复用导引 | Page 8

COTS (Commercial Off-the Shelf): 可以采购到的具有开放式标准定义的接口的软件或硬件产品

软件复用的常见问题



- > 如何开发可被重用的代码?
- > 如何定位/发现合适的复用组件?
- > 有哪些工具可以帮助软件复用? 如何使用?
- > 为什么有时候项目的质量很难与预期一致?
- > 为什么有时候很难区分由于软件复用所节省的开销大小?
- > 与不复用的项目相比,为什么有时候基于软件复用的开发项目会有更长的开发周期?
- > 为了利用复用,需要软件开发过程做出哪些修改?
- > 为了实现复用,需要软件经济模型做出哪些修改?

软件复用导引 | Page 9

构件要达到一定的数量,才能支持有效的复用。而储备大量构件需要有很高的投入和长期的积累。

发现合用构件的困难。当构件达到较大数量(比如几千个)时,使用者要从中 找到一个自己想要的构件,并断定它确实是自己需要的,不是一件轻而易举的 事。

软件开发是一种创造性工作,长期从事这个行业的人们形成了一种职业习惯: 喜欢自己创造而不喜欢使用别人的东西。

复用的困难并不是惟独在软件复用中才能遇到的。为何软件复用显得特别困难?

- 由于对"生产"二字的理解与定位不同,使人们将脑力劳动密集的软件开发与其他产品生产中基本上不包含脑力劳动的批量制作相类比,并讨论其复用问题。
- 对于什么算是一个产品, 衡量标准也大不相同。
- 软件复用所关注的焦点总是随着软件开发水平的提高而上升到更高的抽象层次。这种情况在其他工业产品生产中是不太常见的。
- 物质产品的生产受到人类制造能力的限制。软件却没有这种限制,只要人的大脑能想得到的问题,都可能要求软件去解决。

软件复用与软件工程



- **,软件复用**是一种计算机软件工程**方法和理论**。
- **,软件工程**的发展是计算机程序设计<u>工业化</u>的体现。
- › 软件工程 (Software Engineering)
 - 研究和应用如何以系统性的、规范化的、可定量的过程化方法去开发和维护软件,以及如何把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法结合起来的学科。
 - › 软件需求 (Software requirements)
 - › 软件设计 (Software design)
 - > 软件建构 (Software construction)
 - › 软件测试 (Software test)
 - › 软件维护与更新(Software maintenance)
 - > 软件构型管理 (Software Configuration Management, SCM)
 - → 软件工程管理(Software Engineering Management)
 - → 软件开发过程 (Software Development Process)
 - → 软件工程工具与方法(Software Engineering Tools and methods)
 - 软件质量 (Software Quality)
- 基于组件的软件工程(Component-based software engineering, 简称CBSE)或基于组件的开发(Component-Based Development, 简称CBD)
 在组件对象模型的支持下,通过复用已有的构件,软件开发者可以"即插即用"地快速构造应用软件。这样不仅可以节省时间和经费,提高工作效率,而且可以产生更加规范、更加可靠的应用软件。

软件开发过程

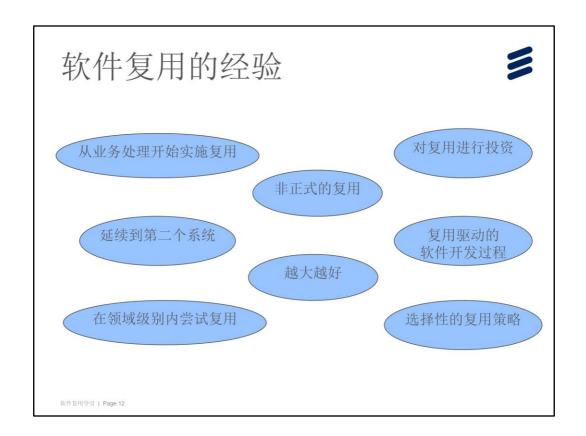


- > 软件开发过程是随着开发技术的演化而随之改讲的。
- > 从早期的瀑布式(Waterfall)的开发模型到后来出现的螺旋式的迭代(Spiral)开发,以致最近开始兴起的敏捷软件开发(Agile),他们展示出了在不同的时代软件产业对于开发过程的不同的认识,以及对于不同类型项目的理解方法。
 - -瀑布式(Waterfall):最典型的预见性的方法。严格遵循预先计划的需求、分析、设计、编码、测试的步骤顺序进行,步骤成果作为衡量进度的方法
 - 迭代(Spiral): 开发工作可以在需求被完整地确定之前启动,并在 一次迭代中完成系统的一部分功能或业务逻辑的开发工作,再通过客 户的反馈来细化需求,并开始新一轮的迭代。
 - 敏捷软件开发(Agile): 应对快速变化的需求的一种软件开发能力。

软件复用导引 | Page 11

敏捷软件开发原则:

个体和互动: 高于 流程和工具。**工作的软件**: 高于 详尽的文档。**客户合作**: 高于 合同谈判。**响应变化**: 高于 遵循计划。



软件复用成功实施的关键



- > 管理者的支持
 - 企业文化(信任)
 - 持续管理(日常指导,周期性调整,多项目协调)
 - 度量
- > 复用支持组织
 - 推广,培训
 - 解决矛盾
 - 建立信任,确保质量
- > 复用库
 - 高质量内容, 完备文档
 - 分类管理,容易找到
- > 复用驱动的方法
 - 引入开发每一过程
 - 体系结构确保跨应用系统复用

软件复用导引 | Page 13

努力获得整个领域需求, 以建立体系结构

软件复用失败的因素



- > 工程
 - 缺乏清晰的标识
 - 缺少能复用的组件
 - 潜在可复用的组件缺乏灵活性
 - 缺乏执行复用过程的工具

> 过程

- 缺乏鼓励复用的机会
- 没有定义复用角色
- 缺乏复用的评审

)组织

- 需要跨项目复用
- 缺乏信任和共享

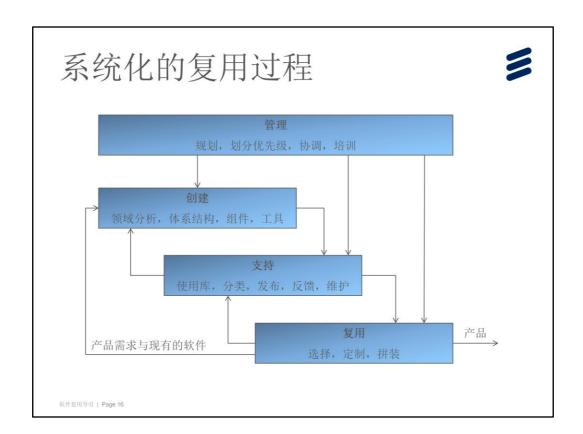
>管理

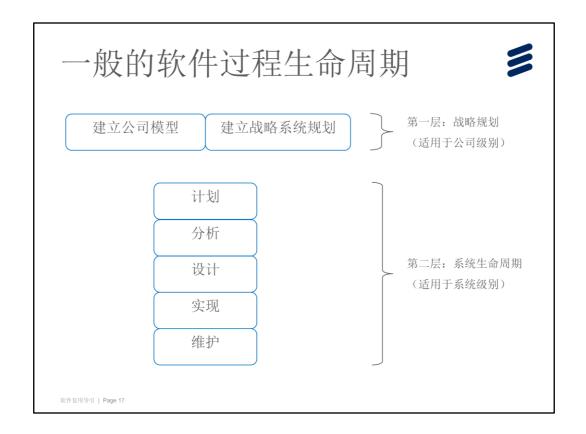
- 需要投入
- 需要建立机制,以提供组织和资金支持

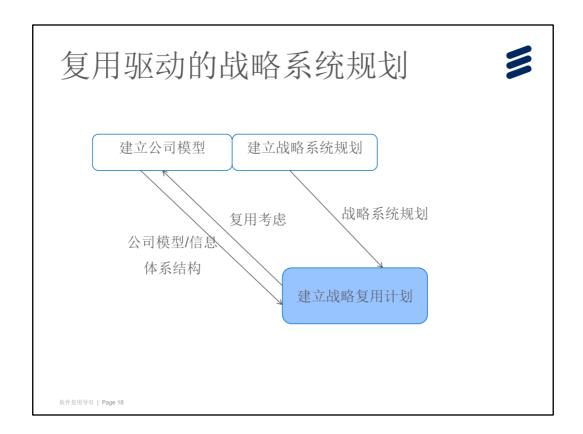


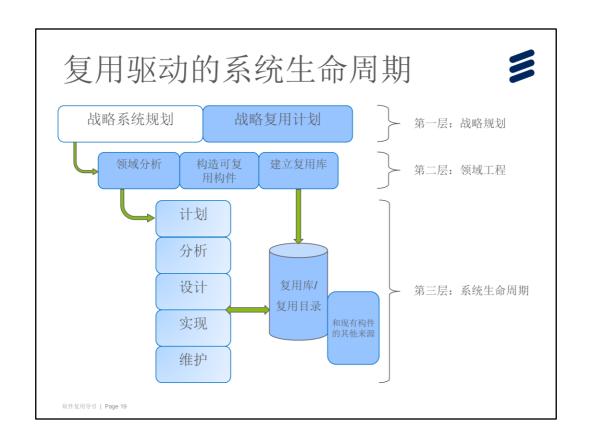
软件复用的实施

在软件过程中复用









领域工程 (战略规划和系统开发之间)

一个识别的建立领域中开发系统所需要适用的可复用构件的过程 将复用实施划分为多个领域

应用者复用微型生命周期:

发现,选择,更改/特殊处理,集成,推荐

生产者复用微型生命周期:

识别候选,领域分析,建立/准备,封装,交付/入库

体系结构跨应用系统

组件工程师开发满足特定需求, 又适合体系结构的组件

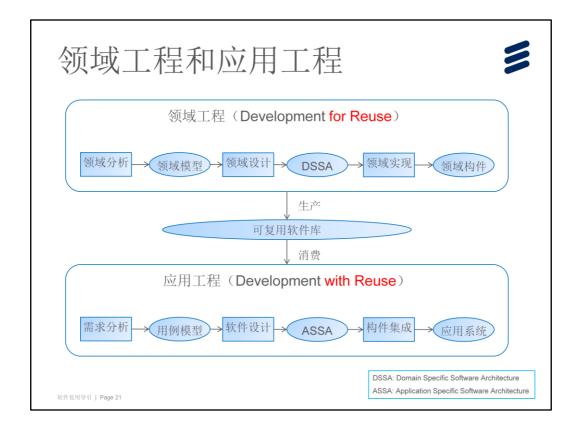
应用系统工程师需要把组件,体系结构和应用系统的特有需求结合到一起

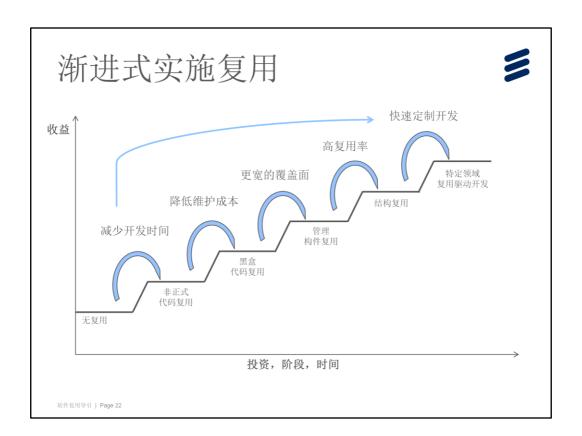
领域工程



- > 领域是指一组具有相似和相近软件需求的应用系统所覆盖的功能区域。
- > 特定领域的软件复用活动相对容易取得成功
 - 领域的内聚性: 领域知识逻辑上的紧密相关性
 - 领域的**稳定性:** 在一定时间内,领域知识不会发生剧烈的变化
-)纵向领域(vertical domain)
 - 业务领域: 医学记录系统、证券管理系统、订单处理系统
- > 横向领域(horizontal domain)
 - 功能分类:数据库系统、工作流系统、GUI库

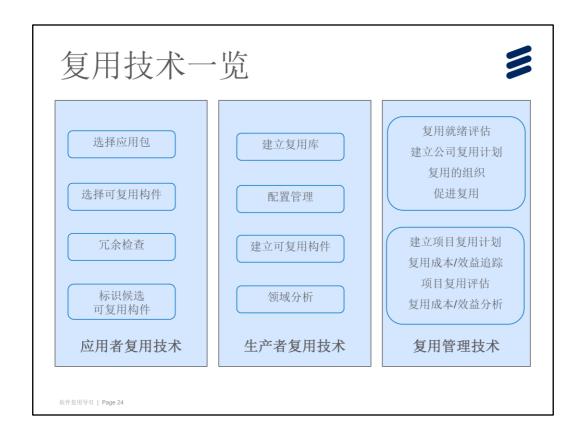








软件复用技术综述



应用者复用技术



- , 选择应用包
 - 整体复用, 越大越好, 最快
- ,选择可复用构件
 - 容易找到, 高质量
-) 冗余检查
 - 共性分析,产生通用构件
- >标识候选可复用构件
 - 精选复用策略

生产者复用技术



-)建立复用库 -组织分类
- > 配置管理
 - 变更维护
-)建立可复用构件
 - -一般化,标准化,自动化,验证化,文档化
- > 领域分析
 - 发现通用体系结构

复用管理技术



- ,公司级复用管理技术
 - 复用就绪评价
 - 建立公司复用计划
 - 复用组织
 - 促进复用
 - > 文化
- ,项目级复用管理技术
 - 项目复用计划
 - 复用成本/效益分析
 - > 事先估计
 - 项目复用评价
 - > 反馈
 - 复用成本/效益追踪
 - > 事后度量

小结 Why 软件复用的概念 软件复用的成效关键 复用驱动的软件过程 领域工程 从How 应用者复用技术 生产者复用技术 复用管理技术 发用管理技术

