



Web需求工程

Qiuyan Huo 霍秋艳
qyhuo@mail.xidian.edu.cn
Software Engineering Institute



Web需求工程

- Web需求特性
- Web需求获取
- Web需求表示
- Web需求确认与验证
- Web需求工具
- 总结与展望

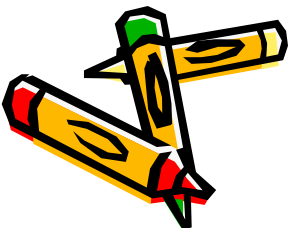


Web需求工程概述

- 需求活动是软件开发过程的重要组成部分，它始于软件项目开发的早期，是整个软件开发过程的入口
 - 处理不好需求会导致需求获取不完全、不正确、不一致等问题，甚至导致Web应用失败
- 软件需求包括业务需求、用户需求、功能需求三个层次
- 需求工程是为了保证系统满足所有利益相关者的目标、需要和期望而进行的活动
- Web应用的需求工程的活动一般分为需求获取、需求表示、需求分析、需求确认与验证四部分



WEB需求特性



Web需求特性

- 多学科性
 - 需要不同领域的专家的共同参与，如网络安全专家、系统架构师、软件可用性专家、美工设计师、数据库专家、业务领域专家、Web应用开发人员等
- 利益相关者未知
 - 在获取需求的时候，Web应用的很多利益相关者无法确定



Web需求特性

- 不断变化的需求和约束
 - 需求和约束常常处于变更之中，很难稳定
- 未知的软、硬件环境
 - 用户软、硬件环境多种多样，有许多因素需要考虑
- 质量控制
 - 质量主要包括性能、可用性、安全性等方面



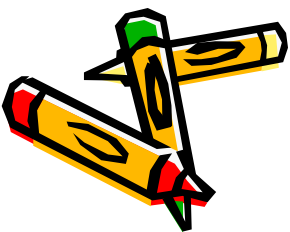
Web需求特性

- 用户界面的可用性
 - 良好的用户界面是用户方便使用Web应用的最重要因素
- 内容的质量
 - 内容是Web应用吸引用户的关键，同时也是传统的需求工程方法经常忽略的内容



Web需求特性

- 开发人员缺乏经验
 - Web应用中使用的很多技术都很新颖
- 严格的预算与交付日期
 - 大多数Web应用的开发者都是规模较小的团队，因此开发预算通常较为紧张，且开发周期一般较短
- 与商业目标紧密相关
 - 不仅需要考虑提供用户所需的内容、迎合用户的目标，也必须能够满足Web应用自身的商业目标



Web应用需求的分类

- **功能需求**：Web应用有什么用、需要做什么、能够为用户提供什么功能、能解决哪些问题
- 功能需求常用UML中的**用例图**进行描述
- 功能需求具体细分：
 - 数据需求：确定Web应用的信息存储和管理方法
 - 界面需求：定义Web应用如何与各种不同类型的用户交互
 - 导航需求：让用户可以方便地在Web应用的不同频道与页面之间跳转
 - 个性化需求：描述Web应用在用户或环境变化时适应的能力
 - 事务性需求：Web应用必须进行的内部处理（后台处理）



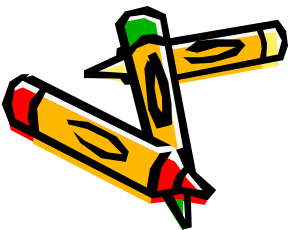
Web应用需求的分类

- **质量需求**：服务的质量以及Web应用的安全性、性能及可用性等方面
 - 系统性能、可靠性、可用性、效率、可维护性、可移植性等
 - 需要考虑网络带宽、网络延时等的影响

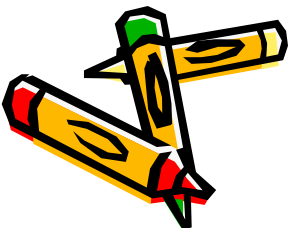


Web应用需求的分类

- **系统环境需求**：Web应用如何嵌入到一个已有的环境中:如何与遗留系统、中间件等外部组件交互
- **项目约束**：项目预算、进度、技术限制、项目所采用的标准、开发技术等
- **发展需求**：需要对Web应用未来的发展有前瞻性



WEB需求获取



Web需求获取

- 集中在用户希望Web应用能够完成的任务上，而不应过于关注Web应用的用户界面
- 需求来源包括文档、遗留系统、问卷调查、面谈等形式
- Web需求获取
 - 需求准备
 - 需求获取方法
 - 需求获取原则
 - 敏捷需求获取



需求准备

- 了解用户为什么会使用这个Web应用
- 用户使用Web应用的特点：
 - 只是扫描(浏览)页面：不会过多地关注Web应用的细节，不希望花费太多时间和精力去阅读那些他们不需要的内容
 - 不做最佳选择：满意即可
 - 不追根究底：只要Web应用能正常使用即可



示例：不同利益相关者的期望不同

- Web应用应该在2014-6-31在线可用。客户约束
- Web应用必须应该至少能允许3000个用户并发访问。客户的质量目标
- 使用Java EE作为软件开发平台。开发人员的预期技术
- 所有用户数据应该被安全提交。客户的质量目标
- 不同用户组可以为用户界面设置不同的布局。用户的质量目标
- 用户能在3分钟内找到他想要的产品。用户的可用性目标



需求获取方法

- 面谈
 - 过程包含四个步骤：识别面谈的利益相关者、面谈准备、进行面谈、将面谈结果记录为文档
 - 要求负责面谈的人拥有丰富的经验，同时要求负责面谈的人能够选择出最适合的人进行访问



需求获取方法

- 用例建模
 - 常采用UML用例图的形式描述
 - 用例是参与者（不仅仅是人，也包含其他软件系统）与系统的交互，代表了系统为其参与者所执行的有价值的操作，用于表达系统的功能需求和行为
 - 目的是理解用户如何对系统进行操作，因此应该从参与者的角度来编写用例



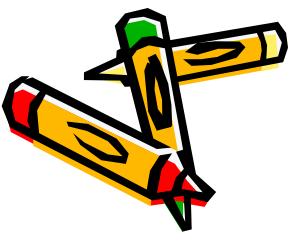
需求获取方法

- 头脑风暴
 - 组织团队会议，利用参会者的相互交流产生思维碰撞
 - 参会者一般不发表反对意见，以免影响会议的自由气氛
- 素描与故事板
 - 绘制各个用户界面（页面）的示意图，让用户对系统的表现形式有感性了解
- 问卷调查和调查表
 - 设计问卷调查表有一定难度



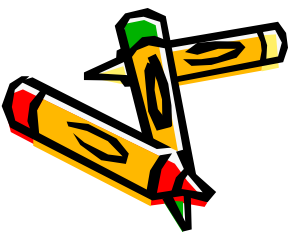
需求获取方法

- 联合应用开发
 - 由IBM于1977年提出，通过一连串的合作研讨会来获取用户需求。耗时，但获取的需求质量较高
- 原型化
 - 先开发出一个系统的原型（只设计页面不实现功能或实现简单的功能）可以让用户在直观上了解系统的展现情况
 - 不仅可以用于需求的获取，还可以用于需求的确认与验证



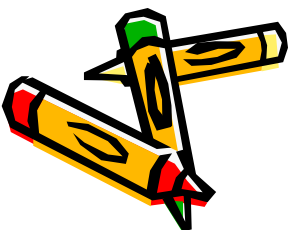
需求获取原则

- 识别目标用户
 - Web应用必须有明确的目标用户群
- 全面考虑各利益相关者的意见
 - 应充分考虑所有的利益相关者的目标，时刻注意解决和协调产生的冲突和矛盾
- 了解系统所处的环境
 - 需要对系统所处的大环境进行分析



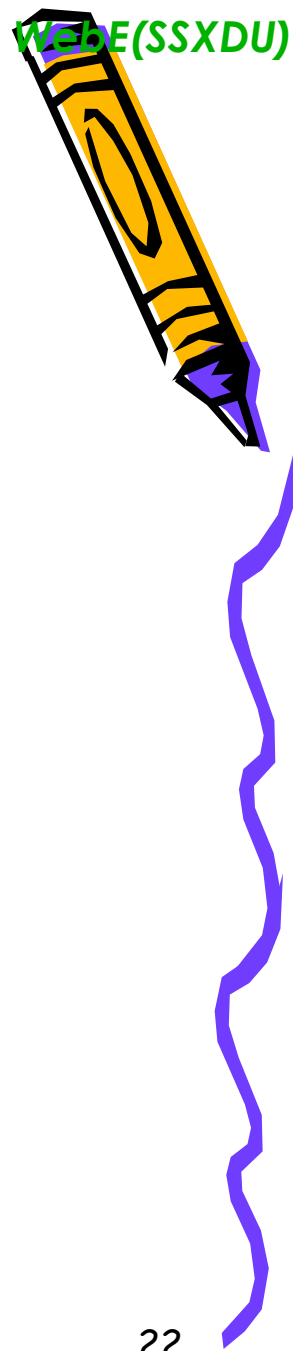
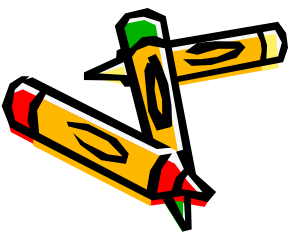
需求获取原则

- 明确调查内容
 - 需要调查用户的实际情况，明确用户需求。常见的调查内容包括：功能需求、性能的要求和可靠性的要求维护的要求、风格及美工要求等
- 进行市场调研
 - 获取Web应用背景资料，并分析其他相似Web应用的性能和运行情况
 - 调研的重点应该放在主要竞争对手的Web应用上



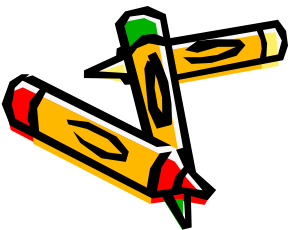
简单的新闻系统

- (1) 登录与注册
- (2) 新闻提交
- (3) 新闻审核
- (4) 新闻发布
- (5) 新闻管理
- (6) 新闻一览
- (7) 设置新闻的评论权限
- (8) 用户管理
- (9) 栏目管理
- (10) 日志管理
- (11) 系统管理



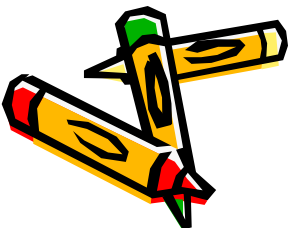
敏捷需求获取

- 敏捷方法日益受到软件业界的重视
 - 敏捷方法对软件需求的变化采取更为主动积极的处理方法，更重视人的作用，适合中小规模的开发团队，很符合Web应用的开发特点
- 敏捷开发方法中一般采用故事板、用例建模、原型化等方法进行需求建模



敏捷需求获取的建议

- 明确权利与义务
 - 利益相关者之间相互理解、相互尊重，各方面都清楚各自的职责并承担自己的义务，如有疑问协商解决
 - 开发者需要让客户了解需求具有不同的重要性和优先级以及忽视需求的危害性
- 结对学习
 - 开发者必须向客户学习其业务，深入客户的工作环境，了解客户的需求
 - 极限编程的思想采用“结对”策略让一个客户和一个开发者结对，两人一起工作，共同讨论需求

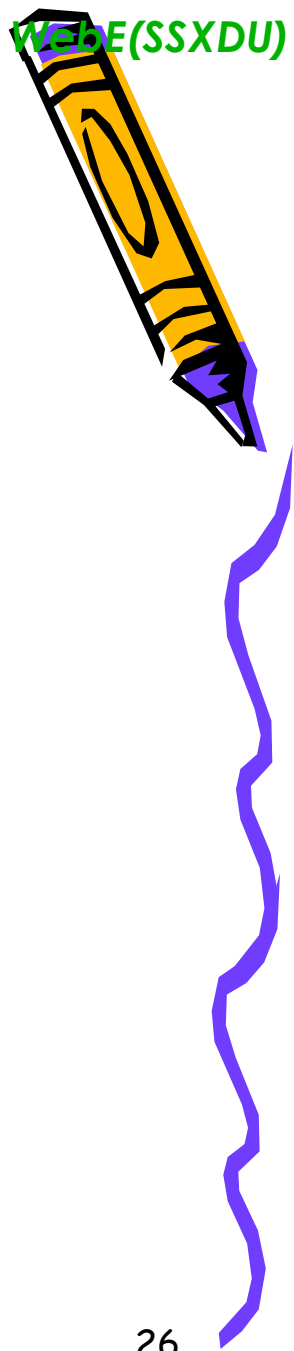


WEB需求表示



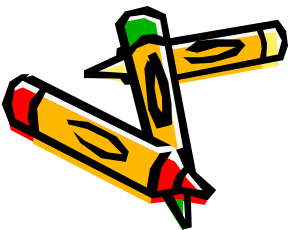
Web需求表示

- Web需求分析
- Web需求表示
- Web需求管理



Web需求分析

- 包括提取、分析和审查收集到的需求，以确保所有的利益相关者都明白其含义并找出其中的错误、遗漏或不足
- 目的在于得到高质量和具体的需求，经历相关的一些活动，遵循的一些原则



Web需求分析活动

- 绘制系统关联图
 - 关联图用于定义Web应用与其外部实体间的边界和接口的简单模型
- 创建用户界面原型
 - 开发一个用户界面原型可以使Web应用具体化
- 分析需求可行性
 - 在成本和性能的约束下，分析每项需求实施的可行性，明确与每项需求实现相联系的风险
- 确定需求的优先级别
 - 以优先级为基础确定Web应用的各个版本将包括哪些特性或哪类
- 为需求建立模型
 - 常用的模型可分为结构化需求分析方法和面向对象需求分析方法两类
- 创建数据字典
 - 数据字典应避免出现二义性



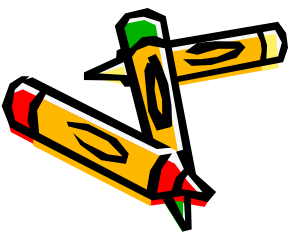
Web需求分析原则

- 注意需求描述用语
 - 开发者需要了解客户使用的业务术语
- 了解客户业务
 - 有助于开发人员设计出真正满足客户需求并达到期望的优秀软件
- 描述产品的非功能特性
 - 分析人员通过询问和调查了解客户所要的非功能特性，分析特性之间的影响，在性能代价和所提出解决方案的预期利益之间做出权衡
- 评估需求变更代价
 - 如果用户需要对需求进行变更，则开发人员应通过分析给出一个真实可信的评估
- 客户参与
 - 客户应该尽可能的参与到整个需求工程的过程中



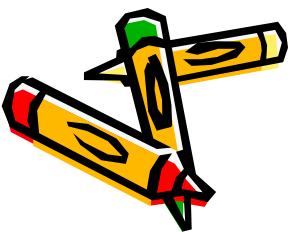
Web需求表示

- 需求故事
 - 需求故事是以口语化的方式描述需求所包含的内容，通常用于开发者和用户之间的交互
 - 例：极限编程中的用户故事（user stories）
- 条目化需求
 - 以自然语言编写，包含多项具体的需求
 - 例：IEEE/EIA J-STD-016标准中的数据项描述（DID，data item description）



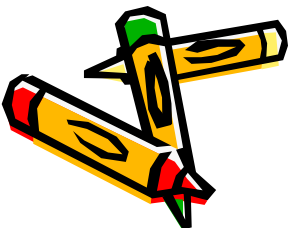
Web需求表示

- 格式化规格说明
 - 格式化规格说明采用严格定义的语法，但仍然允许用自然语言进行描述
 - 例：UML中的用例描述
- 正规需求规格说明
 - 采用形式化规定的语法和语义
 - 很少使用正规需求规格说明

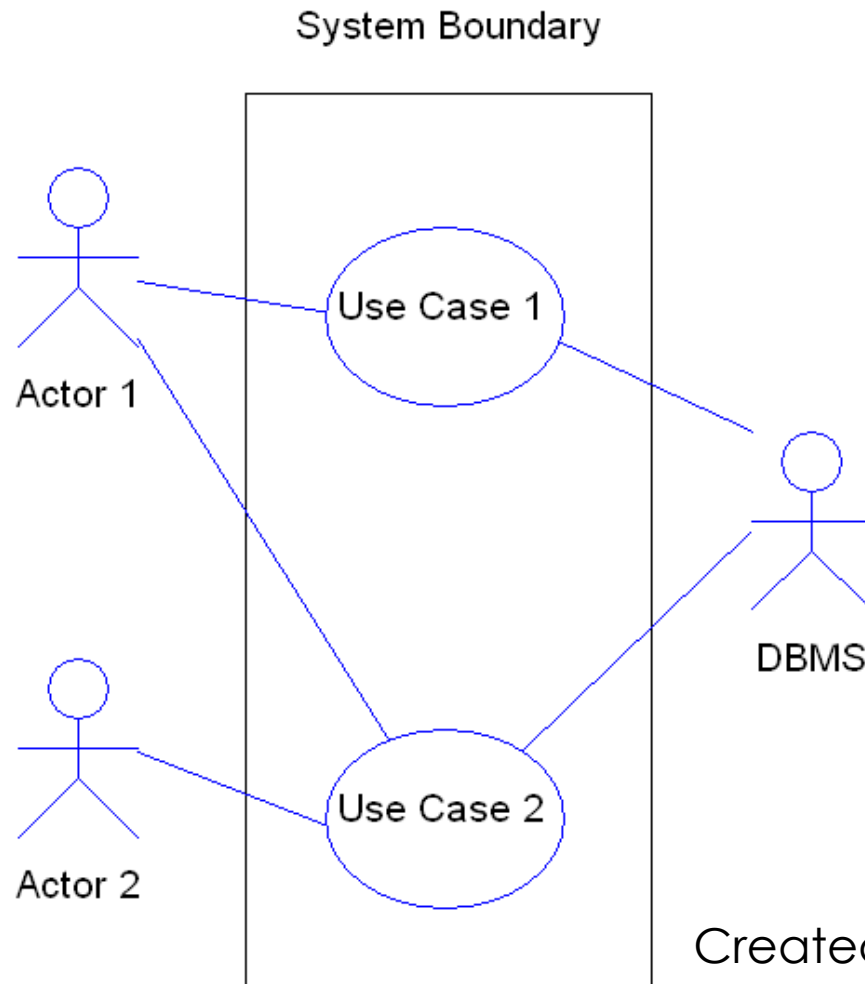


Web需求表示

- 需求故事、格式化规格说明等方法比较适合Web应用开发
 - 开发者的重点应该放在如何把需求向不了解Web开发的用户解释清楚
- 对于Web应用而言，需求分析阶段需要编写**软件需求规格说明**作为整个需求分析活动的结果性文档
 - 不仅是系统测试和用户文档的基础，也是项目规划、设计和编码的基础



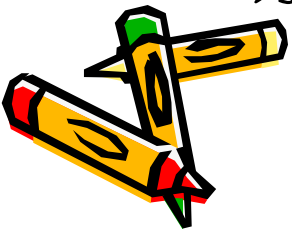
Web需求表示：USE CASE DIAGRAM



Created with PowerDesigner

Web应用需求规格说明的内容

- Web应用功能；
- 初步的Web应用用户界面；
- Web应用运行的软、硬件环境；
- Web应用性能要求；
- 确定Web应用维护的要求；
- 确定Web应用空间租赁要求；
- Web应用页面总体风格及美工效果；
- 主页面及次页面大概数量；
- 管理及内容录入任务分配；
- 各种页面特殊效果及其数量；
- 项目交付时间及进度安排表；
- 明确项目完成后的维护责任。



Web需求管理

- Web需求管理的意义
 - 可以更好地控制复杂的系统：有助于系统开发人员清晰理解Web应用的功能和完成时间的要求，也能将项目资源在需求优先级和影响的基础上进行合理的分配，需求变更的影响也能够得到很好的处理
 - 提高软件产品的质量和客户的满意度
 - 降低项目成本：需求分析阶段所产生的错误，其更正代价是最大的
 - 促进沟通：促进用户与开发者的沟通
 - 使项目的过程更加符合标准：业界最常用的CMM就将需求管理作为软件质量改进的第一步



Web需求管理：变更管理

- 在Web应用需求基线已经确定之后，又要添加新的需求或进行较大的需求修改
- 有好处也有坏处
- 需求变更处理不好的后果：
 - 影响Web应用质量及开发进度
 - 影响文档和代码的一致性
 - 影响开发人员与用户的合作关系



需求变更的常见流程

- 确定需求的基线
 - 通常会以用例作为需求基线，在用例确认之后的任何需求改变，都需要经过需求变更流程
- 项目经理接收到需求变更的要求
 - 需求变更的提出者可以是项目中的任何相关人员
- 项目经理评估需求变更
 - 项目经理可以召集相关人员讨论需求变更的合理性、可行性，实施的代价以及对项目的影响
- 记录需求变更
 - 需求变更确认后由专人将需求变更记录下来，将需求变更的原因等信息通知给项目中所有成员
- 确认需求变更
 - 需求分析人员修改项目文档中的相关内容，开发人员修改代码中的相关部分
- 需求冻结
 - 在项目开发到一定程度时会进入需求冻结阶段，不再接收需求的变更。



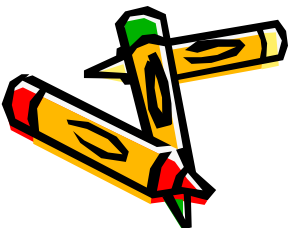
Web需求管理：版本控制

- 对开发过程中代码和文档的版本进行记录和保存，通过合适的规则来创建新版本，删除旧版本，合并已有版本等
 - 通常采用SVN、CVS、Git等版本控制工具来进行需求规格说明的版本控制



Web需求管理：状态跟踪

- 在整个开发过程中，跟踪每个需求的状态是需求管理的一个重要的方面。通常可将需求的状态归类到如下九种状态中的一种：
 - Open（开放）
 - Analyzed（已分析）
 - Reviewed（已评审）
 - Resolved（已编码）
 - Passed（已完成）
 - Unresolved
 - Closed（已结束）
 - Cancel（已取消）
 - Failed（需修正）



Web需求管理：需求跟踪

- 通过比较需求文档与后续工作成果之间的对应关系，确保项目是依靠需求文档进行开发的
- 确保当需求发生变化时不遗漏受到影响的系统元素，从而保证需求变更的正确实施
- 目的是建立与维护“需求—设计—编码—测试”之间的一致性，确保所有的工作成果符合用户需求



Web需求管理：需求跟踪

- 需求跟踪包括：
 - 建立与维护需求跟踪表：记录软件需求与各个软件工程活动的阶段结果之间的映射关系
 - 一致性检查：当发现存在不一致时，应记录在需求跟踪报告中，并通报相关责任人
 - 消除不一致：消除不一致后，应提醒项目管理者更新需求跟踪表



WEB需求确认与验证



Web需求确认与验证

- 验证需求是否正确、是否合理、是否存在着与实际情况不符的问题，并解决这些问题
- 是一个反复迭代的过程，在整个Web应用的开发过程中可能需要进行多次
- “确认与验证->修正需求->再次确认与验证->再次修正需求.....” 的反复过程



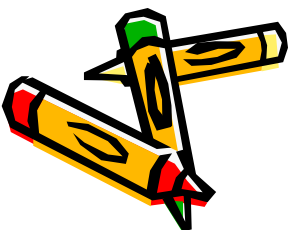
合格的需求规格说明书的特点

- 正确
 - 需求规格说明对描述能反映出用户的真正要求
- 完整
 - 需求规格说明应该包括软件要完成的全部任务
- 一致/无二义
 - 需求规格说明对需求的描述不能存在矛盾之处
- 可修改
 - 需求规格说明的格式和组织方式应保证易于进行后续的修改
- 可跟踪/可验证
 - 每项需求都能与其对应的来源、设计、源代码和测试用例联系起来，每项需求都应该有明确的方法来验证其是否已经实现



需求确认与验证的目标

- 软件需求规格说明正确描述了预期的系统行为和特征
- 需求是完整的和高质量的
- 任何对需求的看法是一致的
- 需求为继续进行产品设计、构造和测试提供了足够的基础



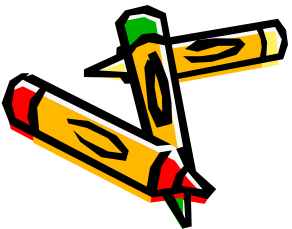
需求确认与验证常用的形式

- 评审和走查 (Review and Walkthrough)
 - 评审分为客户评审和同行评审两类
 - 内容包括：评审Web应用的目的和目标、运行环境、是否准确地反应了客户的需要和意图、可能存在的风险
- 审计 (Audit)
 - 在审查文档的过程中执行的检查，将结果与过程开始之初预先定义的列表进行比较



需求确认与验证常用的形式

- 需求跟踪矩阵
 - 可以检测需求中的不一致和未覆盖的问题
 - RTM分为纵向跟踪矩阵和横向跟踪矩阵两类
 - RTM的优点：
 - 在需求变更、设计变更、代码变更、测试用例变更时，RTM是目前最有效的进行变更波及范围影响分析的工具。
 - RTM可以验证需求是否已得到实现



需求确认与验证常用的形式

- 原型验证

- 系统分析员借助白板、白纸等沟通方式，帮助用户清楚表述需求。然后，开发一个用户界面原型，以便用户确认需求
- 必须让用户明白他们观察到的只是一个原型而不是最终的系统。开发人员需要分清原型和真实应用之间的区别，对真实的应用不能一味沿用原型的开发方式



需求确认与验证常用的形式

- 模型验证
 - 当系统需求已经采用某种结构化或形式化的表示机制来刻画的时候，可以采用已有的模型验证技术来进行需求的确认与验证
 - 模型验证的目的：证明每个独立的模型是自包含的、证明模型是内容一致和外部一致的、证明模型精确地反映了系统需求相关者的真实需求
- 需求建模测试
 - 包括把需求转换成图形模型或形式化语言模型
 - 通常需要借助一定的CASE工具



WEB需求工具



Web需求工具

- 需求工具可以分为以数据库为核心和以文档为核心两大类
 - 以数据库为核心的需求管理工具如CaliberRM和DOORS，它们把所有的来自不同文档中的需求属性和跟踪能力信息存储在数据库中
- 以文档为核心的需求管理工具如RequisitePro，它允许选择将整份需求文档存储在数据库中，加强了以文档为核心的处理方法的能力



商用需求管理工具

- CaliberRM
 - Borland公司的一款基于Web 的支持多人协作的需求定义和管理工具
- RequisitePro
 - IBM公司Rational产品线中的需求和用例管理工具
- DOORS
 - IBM公司Rational产品线中的多平台的需求管理解决方案
- TestTrack RM
 - Seapine公司的一款需求管理产品
- DevSpec
 - TechExcel公司的一款需求定义与需求管理工具

<http://makingofsoftware.com/resources/list-of-rm-tools>

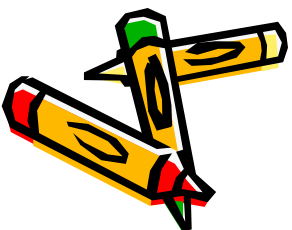


开源需求管理工具

- Trac
 - 基于Web的软件项目开发跟踪工具，它最大的特点是支持众多的插件，对Wiki也有很好的支持
- Redmine
 - 基于Web的项目开发管理工具，用Ruby on Rails 框架开发。支持多项目、多数据库、多语言。对项目开发中的文档管理、进度管理、权限控制、成员交流等都有很好的支持

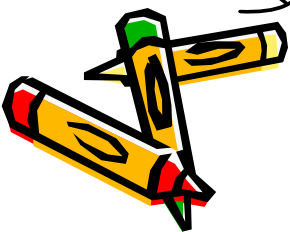


总结与展望



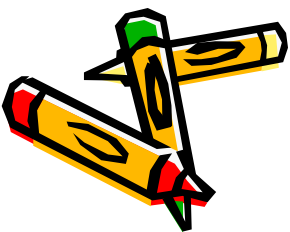
总结

- 需求定义的好坏直接影响了开发者对Web应用的理解程度，是影响一个Web应用成功的重要因素
- 通过了解本章的内容，Web开发者能够更好的获取到有用的Web系统信息，最终开发出满足用户需求的Web产品
- Web应用需求的特性，使得从需求获取、表示和所使用工具等方面均有众多差异，面临诸多挑战



展望

- 针对Web应用开发，其需求工程未来的发展趋势主要包括：
 - 开发和使用系统的边界越来越模糊
 - 需求和架构之间的集成越来越紧密
 - 新的支持分布式需求工程的工具逐步出现
 - Internet的开放性使得Web应用中使用的各种开发系统的需求工程越来越重要



Task3 – 团队

- Web项目需求文档
 - 根据本章需求内容，完成项目需求规格说明书

