

软件过程

董渊

(System Software & Software Engineering)

Department of Computer Science & Technology

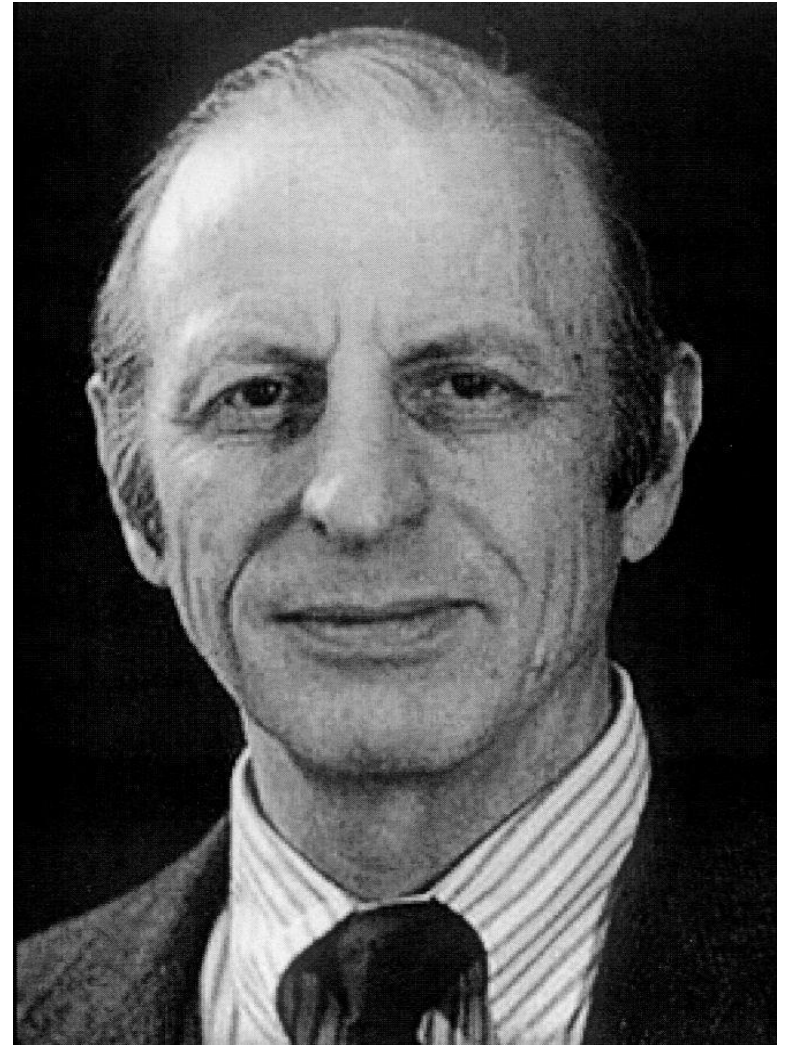
Tsinghua University

The Process Premise

“The quality of a software system is governed by the quality of the process used to develop and evolve it.”

-- **Watts Humphrey**

Capability Maturity Model – CMM
Managing the Software Process - SPM



内容概要

- 软件生存周期
- 软件过程的基本概念
- 软件过程模型
 - 基本模型
 - RUP
 - Agile 方法

软件生存周期

- 软件孕育、诞生、成长、成熟、衰亡的生存过程
 - 制定计划
 - 需求分析和定义
 - 软件设计
 - 程序编写
 - 软件测试
 - 运行/维护

制定计划

- 确定总目标
- 可行性研究
 - 探讨解决方案
 - 资源、成本、效益、进度评估
- 实施计划

- 软件生存周期
 - 制定计划
 - 需求分析和定义
 - 软件设计
 - 程序编写
 - 软件测试
 - 运行/维护

需求分析和定义

● 目的

- 描述软件的功能和性能
- 确定软件设计的限制和软件同其它系统元素的接口细节
- 定义软件的其它有效性需求

软件生存周期

制定计划

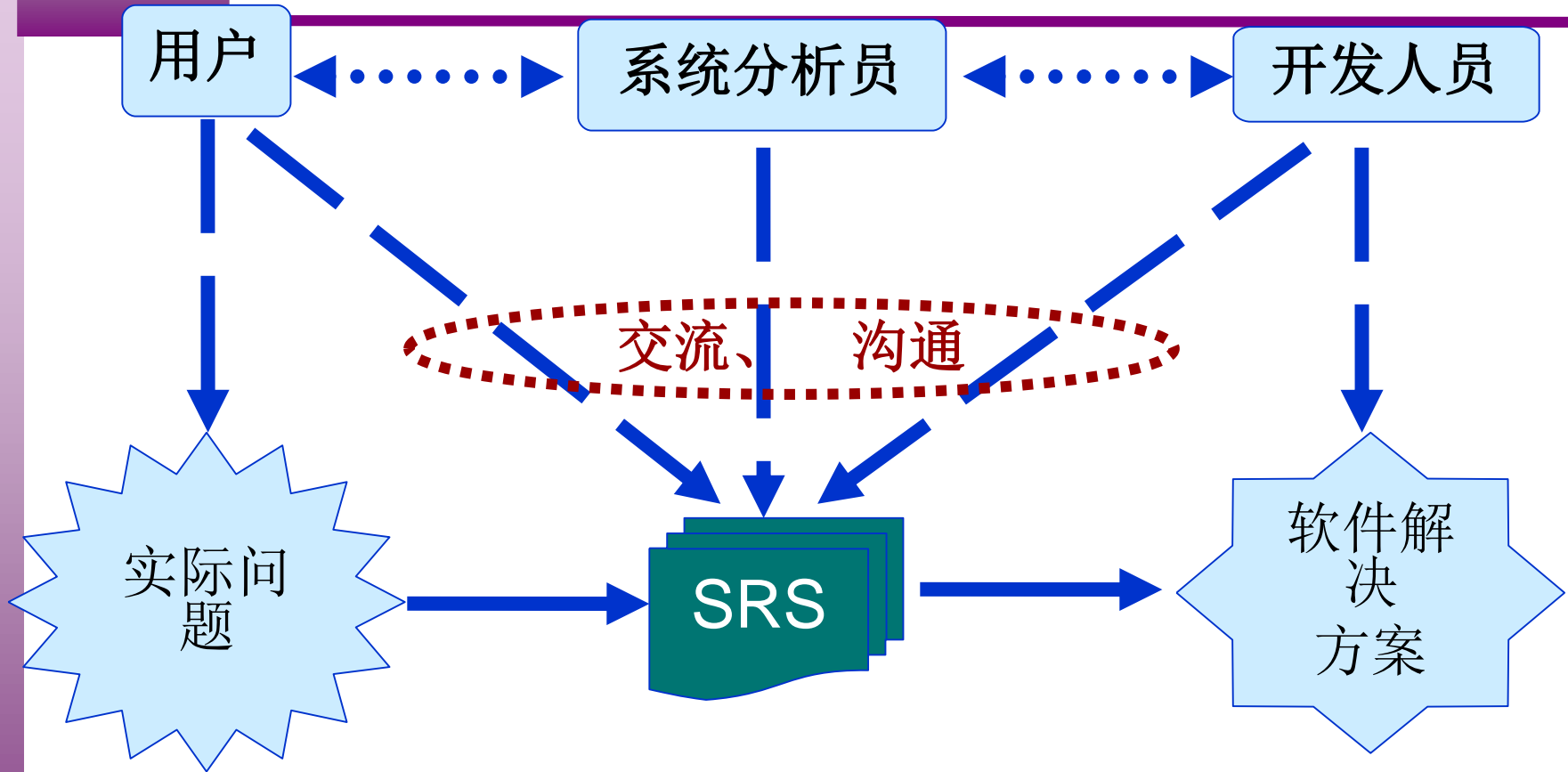
需求分析和定义

软件设计

程序编写

软件测试

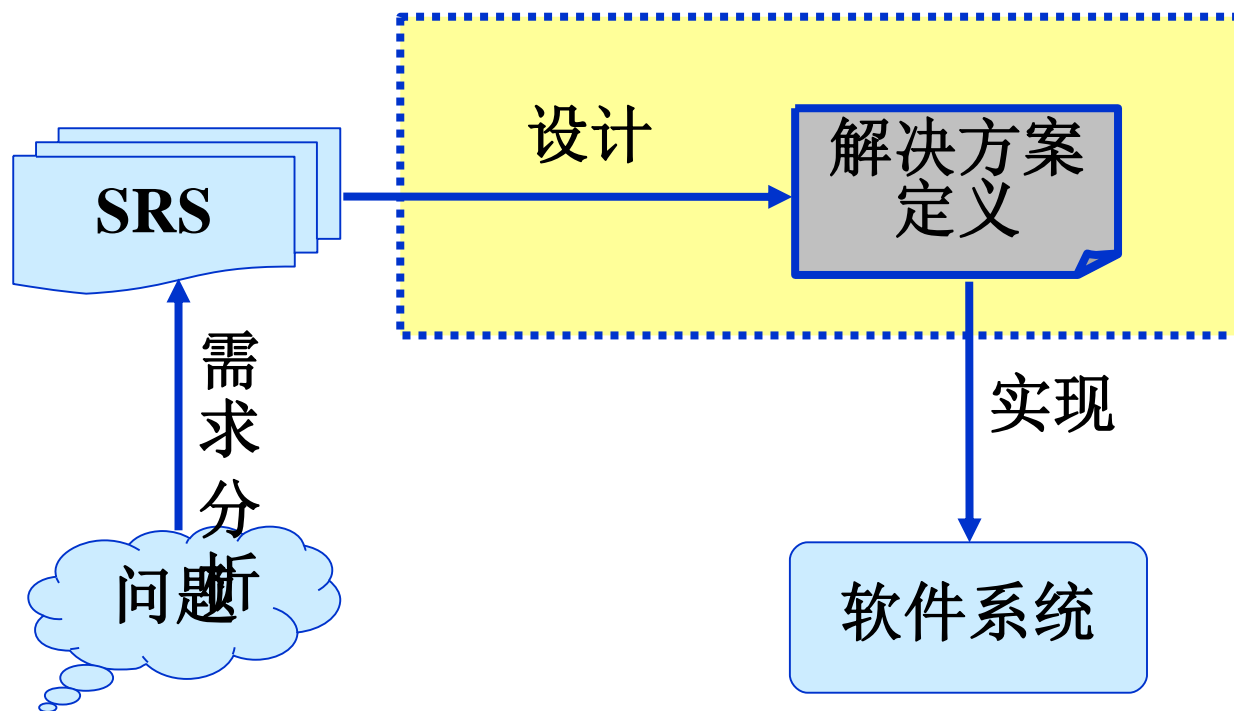
运行/维护



He who asks a question is a fool for five minutes;
he who doesn't ask a question remains a fool forever.

软件设计

- 软件工程的技术核心



软件生存周期

制定计划

需求分析和定义

软件设计

程序编写

软件测试

运行/维护

程序编写

- 将软件设计转换成计算机可接受的、以某种特定语言表示的程序代码
- 结构良好、清晰易读、与设计相一致
- 程序设计语言、程序设计风格

软件生存周期

制定计划

需求分析和定义

软件设计

程序编写

软件测试

运行/维护

软件测试

- 质量保证的重要手段。
 - 以最少的时间和人力，系统地找出软件中潜在的各种错误和缺陷。
- 分类
 - 基本分类：黑盒/白盒
 - 按测试阶段划分：单元/集成/系统
 - 按测试目的划分：功能/性能/安全
 - 按测试对象划分：C/S测试，GUI测试，实时软件测试

软件生存周期

制定计划

需求分析和定义

软件设计

程序编写

软件测试

运行/维护

软件维护

- 在系统进入运行后对系统所做的修改
- 分类
 - 改正性维护
 - 适应性维护
 - 完善性维护
 - 预防性维护

软件生存周期

制定计划

需求分析和定义

软件设计

程序编写

软件测试

运行/维护

Software in Natural

- 软件 —— 具体化的知识，资本，“知本”
- 软件开发 —— 社会的学习过程，认识
- 认识过程：理解、描述、验证、改进

内容概要

- 软件生存周期
- 软件过程的基本概念
- 软件过程模型
 - 基本模型
 - RUP
 - Agile 方法

软件过程的基本概念

“软件过程也称为软件生存周期过程或软件过程组，是指软件生存周期中的一系列相关过程。过程就是活动的集合，活动是任务的集合，任务则起到把输入加工成输出的作用。活动的执行可以是顺序的、迭代的（重复的）、并行的、嵌套的或是有条件引发的。”

❖ 软件过程的含义

个体含义：

软件或系统在生存周期中某一类活动的集合

整体含义：

软件过程的总体

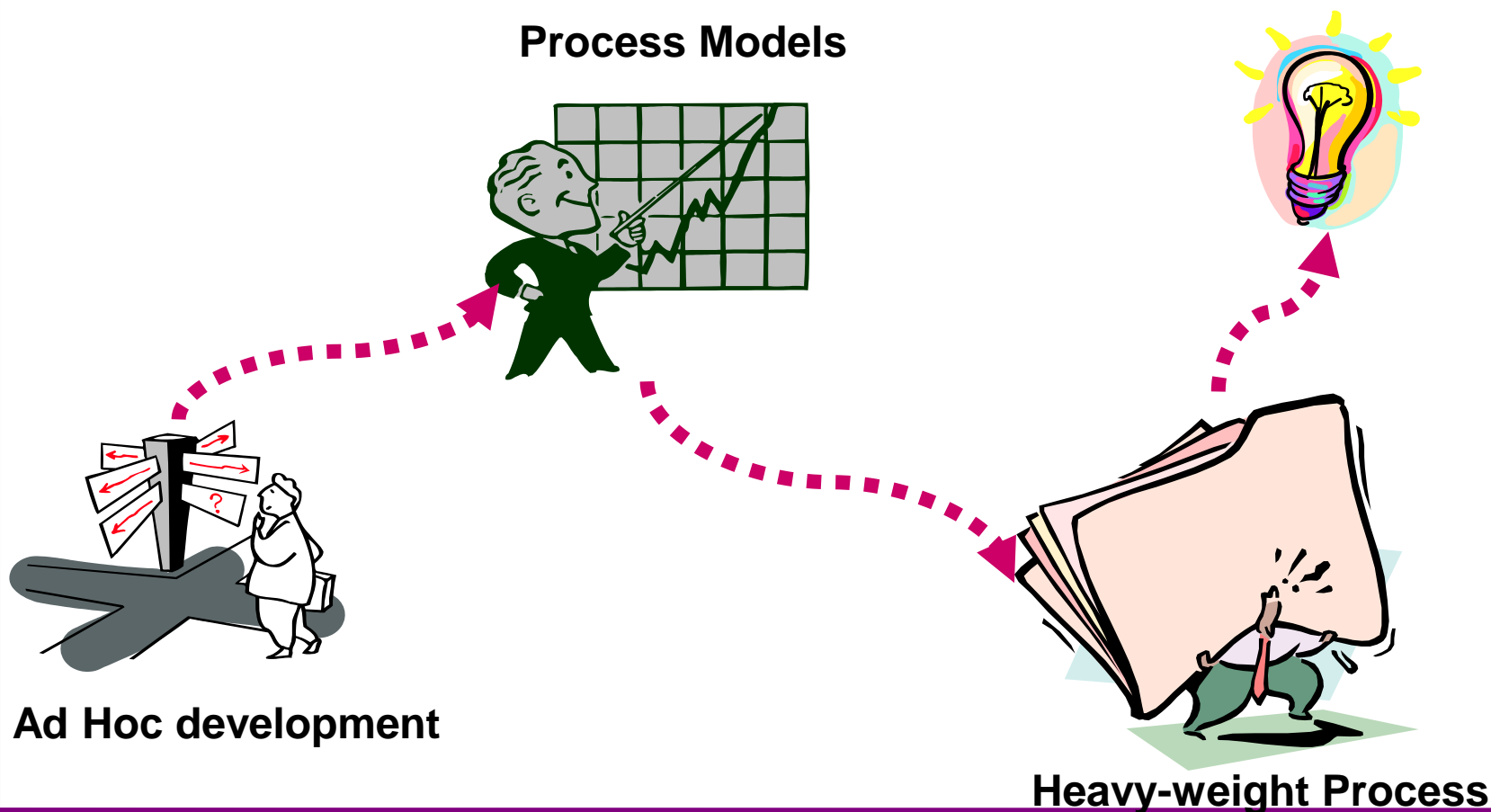
工程含义：

应用软件工程原则、方法来构造软件过程模型，并结合软件的具体要求进行例化，以及在用户环境运作

软件过程的发展

Agile (Light-weight) Process

Process Models



内容概要

- 软件生存周期
- 软件过程的基本概念
- 软件过程模型
 - 基本模型
 - RUP
 - Agile 方法

基本模型

- 瀑布模型 (Waterfall Model)
- 原型模型 (Prototyping Model)
- 增量模型 (Incremental Model)
- 螺旋模型 (Spiral Model)
-

1. 最佳软件开发实践

- 为解决软件开发中的问题，需要以一种更好的、迭代的、可预测的方式开发软件产品，R U P (By Rational, IBM) 提出软件开发中的最佳实践。
- 最佳实践包括：
 - 1) 迭代式软件开发；
 - 2) 需求管理；
 - 3) 基于构件的软件体系结构；
 - 4) 建立软件可视化模型；
 - 5) 不断验证软件质量；
 - 6) 控制变更。

RUP (Rational Unified Process)

- 广泛用于各种面向对象项目
- 统一的描述工具，UML(Unified Modeling Language)
- 遵循太多的规则、编写太多的文档，往往阻碍整体的开发进度。
- “重量级”过程
- 可裁剪、可定制
- Rational Unified Process软件（ROSE）

敏捷方法

- 2001年，Kent Beck和其他16位知名软件开发、软件工程文章作家以及软件咨询师（被称为敏捷联盟）共同签署了“敏捷软件开发宣言”。该宣言声明：
我们正在通过亲身实践和帮助他人以实践的方式揭示更好的软件开发之路，通过这项工作，我们认为：
个体和交互胜过过程和工具
可工作软件胜过宽泛的文档
客户合作胜过合同谈判
响应变化胜过遵循计划
亦即，虽说上述右边的各项很有价值，但我们认为左边的各项具有更大的价值。
- 一份宣言通常和一场即将发生的除旧立新的政治运动相联系。从某些方面来讲，敏捷开发也确实就是一场运动。

敏捷方法

- Light-weight process / Heavy-weight process:
 - “重载过程” 有大量的文档要求和严格的执行纪律，使得软件人员负担过重，难以适应快节奏的软件开发要求。
- Martin Fowler “新方法论”
 - 需求不可预期，要等到软件问世后才会清楚，因此好的方法应能够灵活适应需求的变化
 - Just Enough, Good Enough, 过程管理不是目的，应采用合理的管理策略
 - “人”的因素，发挥人员的创造力和潜能

- 自由软件开发 (**Bazaar**)
 - GNU and GPL
 - Copyright vs CopyLeft
 - 没有担保
 - 利益相关者
 - 开发人员/测试者/用户, 自身的需求
 - 不限制商业用途
 - Linux
 - 通过网络联系, 遍布世界的开发者
 - 并行开发/测试和修改
 - 短周期系统发布

小结

- 软件生存周期包括制定计划、需求分析和定义、软件设计、程序编码、软件测试、软件维护六个基本活动。
- 软件过程是软件开发活动的集合。软件活动可以顺序、重复、迭代等以不同的次序执行，构成不同的软件过程。过程模型是软件过程的抽象描述。
- 不同的软件过程模型具有不同的特点，适用于不同的开发环境，同一软件项目可针对解决的问题采用不同的过程模型。

小结（续）

- “重量级”过程改进强调全面严格的管理、控制
- “轻量级”过程改进强调人的因素，以求灵活、快速应对各种环境变化。
- 对于提高软件可靠性和可维护性，过程与技术同等重要，过程能力成熟度是软件企业的工程能力及可持续发展性的一个重要标志。
- 关于大作业：真正的需求！！！！

作业与课外阅读

❖ 作业（10Points）

- 浏览一个自由软件项目的发展情况，讨论其过程模型及改进情况。（<1000字中文）
- 小组作业，公开发布于课程网站小组页面
- 9月27日参加课堂讨论
- 查找资料、完成作业、网上实验 / 讨论，<10页Slides

● 新方法论， Martin Fowler

- <http://www.martinfowler.com/articles/newMethodology.html>

● 大教堂和市集， The Cathedral and the Bazaar

- <http://www.catb.org/~esr/writings/cathedral-bazaar/>
- <http://www.linuxforum.net/books/c&b.html>

❖ 大作业选题

- What: 软件解决什么问题？主要功能是什么？
- Who: 预期用户是谁？
- Why: 为什么需要？现有相关软件是什么？
- 小组“大作业项目题目”和初步方案（<800字中文）
- 小组作业，公开发布于课程网站小组页面
- 10月9日课堂时间邀请责任助教小组讨论
- <10页Slides