

软件工程导论实验



■ 实验目标

- 掌握基于过程的软件工程设计方法和思想。
- 通过一个具体的项目的开发设计来理解传统软件工程设计思想。
- 要求每个学生都能熟练掌握数据流图、软件层次结构图、程序流程图和盒图的绘制。



■ 实验要求

1. 掌握数据流图的设计步骤、层次结构图（HIPO图）的构建方法。
2. 掌握程序流程图、盒图、PAD图、判定表、判定树的绘制。
3. 在软件项目开发过程中尽可能多地运用计算机辅助开发工具（CAD）进行开发。



■ 实验要求

4. 按照给定的参考模板，按组提交详细的开发过程文档（包括图解和文档描述），各模板文档中还未学到的内容可先空着不填。

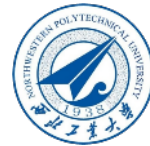
包括如下技术文档：

项目开发计划

需求说明文档（数据要求说明文档）

总体设计文档

详细设计文档



■ 实验要求

5. 培养合作共赢的团队意识，每次实验根据各组提交的文档内容、质量打分，小组得分即个人得分。



■ 实验的组织形式

1. 4人组成一个开发团队，组长轮循，确保每个人都有机会做组长，按组打分（即个人得分）。
2. 各组可在给定的项目中进行选择，也可以自选，但自选的题目需要单独找助教确认题目的难度及功能点是否符合要求。



■ 实验内容

实验1. 组队并确定开发项目名称，根据给定模板完成组队信息表（excel表）及项目开发计划（建议按照给定的国标模板），**确定项目过程模型并说明原因。**

实验2. 根据给定的需求规格说明书模板（建议按照给定的国标模板），完成项目的需求规格说明书。

实验3. 根据给定的总体设计模板（建议按照给定的国标模板），完成项目的总体设计报告。

实验4. 根据给定的详细设计模板（建议按照给定的国标模板），完成项目的详细设计报告。



■ 实验提交

1. 在资源管理器中的地址栏中输入：ftp://10.13.0.10，先双击“课件”，然后双击“软件工程导论”，再双击“实验提交”，最后提交到对应的实验文件夹中即可。
2. 如需二次提交，可将原文件重新命名，在原文件名后加括号，括号里面加上new重新提交到ftp上即可。
3. 访问学院ftp需要借助校园网才能访问，在校外的计算机上需要安装EasyConnect（翻墙软件），可在学校官网中的信息中心下载后安装登录连接即可！学校的VPN服务器地址是<https://vpn.nwpu.edu.cn/>。



■ 相关实验资料的下载

在资源管理器中的地址栏中输入：

ftp://10.13.0.10，先双击“课件”，然后双击“软件工程导论”，选择需要下载的相关文件复制-粘贴到本地机上即可。

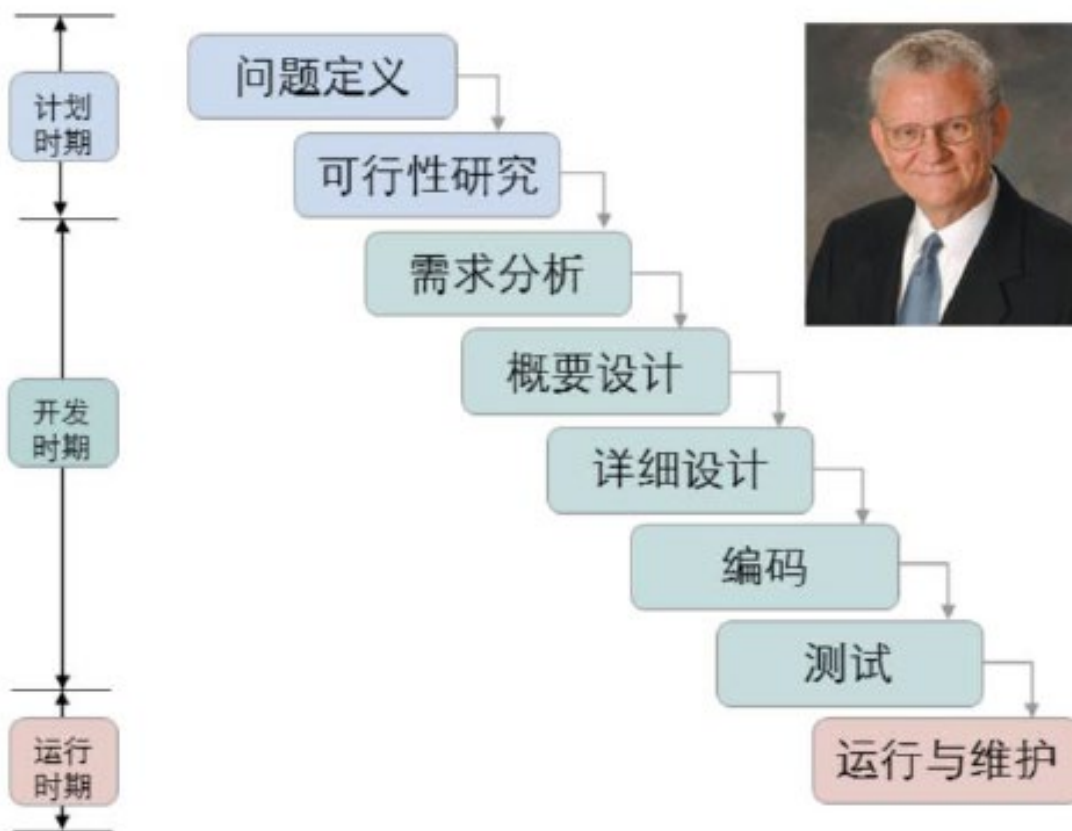


实验课的几点要求：

- 按时上下课。
- 四次实验课，每位同学认真按照四次实验要求完成各自模块，最终成绩为：小组的实验成绩+个人平时成绩（考勤）。
- 每组参照给定的模板文档尽可能地填写其内容，确实填写不了的可先空着。
- 当修完学院后期开设的相关专业课程如：用户界面设计、软件需求工程、数据库系统、软件测试等，各组可以继续完善各文档，并最终完成项目的全部开发设计，确保每个学生都具备完整的项目开发经验，为今后的软件开发工作做好充分的准备。



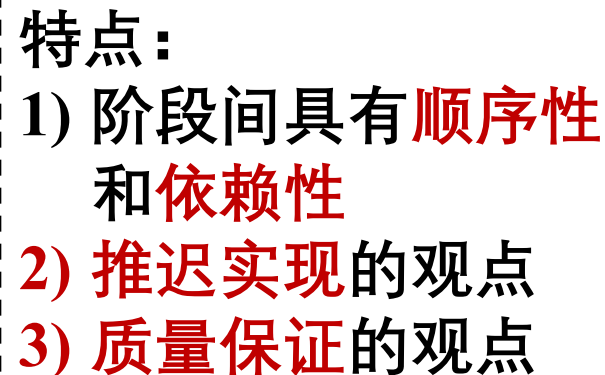
瀑布模型



1970年温斯顿·罗伊斯 (Winston Royce) 提出了著名的“瀑布模型”，直到80年代早期，它一直是唯一被广泛采用的软件开发模型。罗伊斯提出的是一个两次瀑布的“迭代”模型 (Do it Twice)。



- **瀑布模型**也称为**传统生存周期**或**线性模型**。
- 各项活动按**自上而下**，**相互衔接**的**固定次序**，如同瀑布逐级下落，每项活动均处于一个质量环（输入-处理-输出-评审）中。
- 阶段间具有**顺序性**和**依赖性**。
- **推迟实现**的观点。
- **每个阶段必须完成规定的文档**；**每个阶段结束前完成文档审查**。



特点：

1) 阶段间具有顺序性和依赖性

2) 推迟实现的观点

3) 质量保证的观点



➤ 瀑布模型的特点：

- ✓ 瀑布模型严格按照软件生存周期各阶段来进行开发
- ✓ 上一阶段的输出即是下一阶段的输入
- ✓ 规定了各阶段的任务和应提交的成果及文档
- ✓ 每阶段的任务完成后，都必须对其阶段性产品（主要是文档）进行评审，通过后才能开始下一阶段的工作。

文档驱动模型



软件项目或产品选择瀑布模型，必须满足下列条件：

- (1) 在开发时间内需求没有或很少变化。
- (2) 分析设计人员对应用领域很熟悉。
- (3) 低风险项目（对目标、环境很熟悉）。
- (4) 用户使用环境很稳定。
- (5) 用户除提出需求以外，很少参与开发工作。



- 尽管上述条件比较苛刻，但是，软件企业在**开发新产品或新项目**时，往往还是**采用瀑布模型**。
- **系统软件和工具软件**，也常常采用**瀑布模型**。



敏捷模式

敏捷联盟

年2月，在美国犹他州的一个滑雪场，由17个人组成的一个自称为无政府组织的团体出现了。这是17位轻量级软件开发方法的创始人和专家，包括 Kent Beck、Martin Fowler、Robert C. Martin 等等，共同发布了 “The Manifesto for Agile Software Development”（敏捷软件开发宣言）。敏捷终于为软件行业，以及这个行业的一些人所认识、理解和推崇

敏捷宣言



我们正在通过亲身实践以及帮助他人实践，揭示更好的软件开发方法。

通过这项工作，我们认为：

个体和交互 胜过 过程和工具

可以工作的软件 胜过 面面俱到的文档

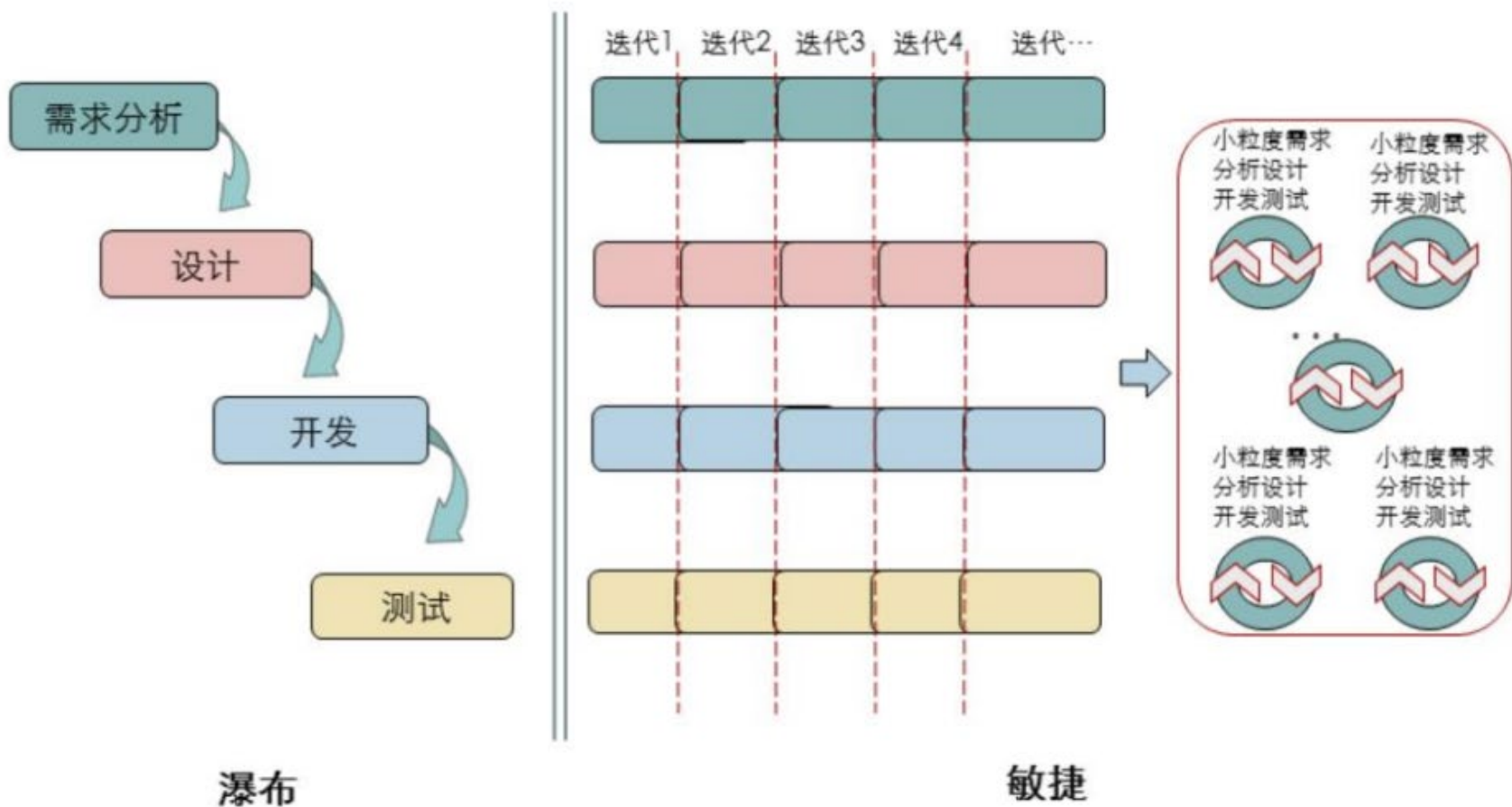
客户合作 胜过 合同谈判

响应变化 胜过 遵循计划

虽然右项也有价值，但是我们认为左项具有更大的价值。



敏捷与瀑布的外在区别





- 从项目的运作角度来看，瀑布模型是一个阶段一个阶段进行，一个阶段的结束是下一个阶段的开始，项目的**需求是**
一次性输入的。
- 而在敏捷方式下，通过将**需求的粒度变小**，在一个迭代中开展需求分析-设计-开发-测试，即一个瀑布模型。
- 因此敏捷方式和瀑布模型的本质区别是输入的**需求粒度**。



敏捷较传统模式更符合软件开发规律

传统开发:



敏捷开发:



- 软件更像一个活着的植物，软件开发是自底向上逐步有序的生长过程，类似于植物自然生长
- 敏捷开发遵循软件客观规律，不断的进行迭代增量开发，最终交付符合客户价值的产品



适应变化，认清“客户是逐步发现真正需求”

美好愿望

- 客户知道自己要的是什么
- 开发人员知道如何开发来满足客户需求
- 在开发过程中需求不会发生变化

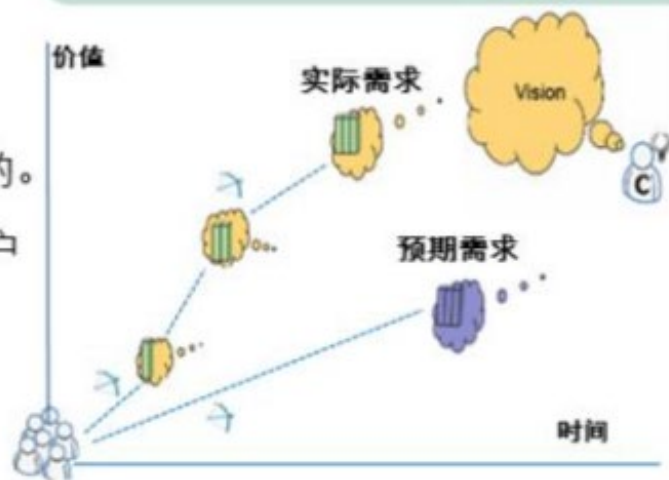
我们认识到



残酷现实

- 客户是逐步发现他真正要的东西
- 开发人员逐步发现如何开发产品满足客户需求
- 在这个过程中随时可能发生变化

- 期望客户一开始就想清楚他们真正要的东西是不现实的。
- 我们应当通过不断的向客户交付可用的产品，启发客户逐步的发现真正的需求。





敏捷软件开发核心—迭代开发

什么是迭代式开发

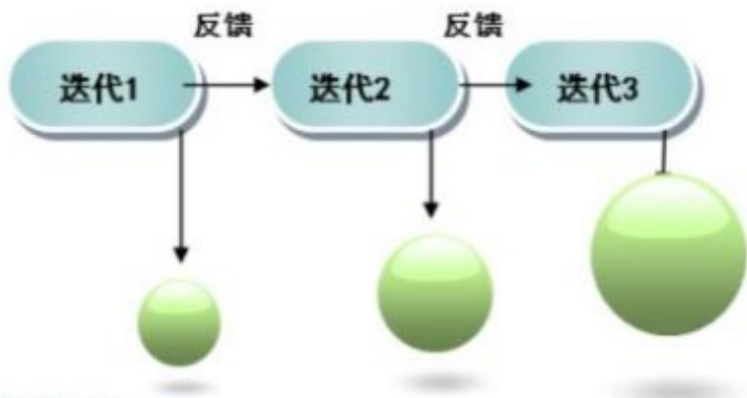
- 迭代开发将整个软件生命周期分成多个小的迭代（一般2-4周），每一次迭代都由需求分析、设计、实现和测试在内的多个活动组成，每一次迭代都可以生成一个稳定和被验证过的软件版本。

迭代式开发的好处

- 通过将高技术风险的需求在早期迭代里实现，有助于尽早暴露问题和及时消除风险
- 通过提供功能渐增的产品，持续从客户获得反馈，根据反馈及时调整，使最终产品更加符合客户的需要
- 通过小批量减少排队，提供更灵活、快速的交付能力
- 平滑人力资源的使用，避免出现瓶颈

迭代式开发的关键要点

- 每一次迭代都建立在稳定的质量基础上，并做为下一轮迭代的基线，整个系统的功能随着迭代稳定地增长和不断完善。
- 每次迭代要邀请用户代表（外部或内部）验收，提供需求是否满足的反馈
- 迭代推荐采用固定的周期（2-4周），迭代内工作不能完成，应当缩减交付范围而不是延长周期



迭代开发是有节奏地小步快跑，但建立在坚实的质量基础上



迭代交付：

- 将一R版本分为多次迭代。
- 通常情况下，一次迭代的持续时间为2至4周。
- 对每张故事卡都作估算。
- 按照客户需求对故事进行排列，并按迭代对故事进行计划。
- 和团队所有成员召开会议，讨论故事，进行估算，达成一致。



项目计划常用的CAD软件：

- 微软的project。
- excel。