软件过程模型研究开发报告

所谓软件过程模型就是一种开发策略，这种策略针对软件工程的各个阶段提供了一套范形，使工程的进展达到预期的目的。对一个软件的开发无论其大小，我们都需要选择一个合适的软件过程模型，这种选择基于项目和应用的性质、采用的方法、需要的控制，以及要交付的产品的特点。一个错误模型的选择，将迷失我们的开发方向，下面我们介绍几种主流的软件开发模型的基本概念以及其特点、优点、局限性等。

1. 瀑布模型（Waterfall Model）
   1. 基本概念  
      瀑布模型将[软件生命周期](http://www.cnblogs.cc2/view/47193.htm)划分为制定计划、需求分析、软件设计、程序编写、[软件测试](http://www.cnblogs.cc2/view/16563.htm)和运行维护等六个基本活动，并且规定了它们自上而下、相互衔接的固定次序，如同瀑布流水，逐级下落。在瀑布模型中，软件开发的各项活动严格按照线性方式进行，当前活动接受上一项活动的工作结果，实施完成所需的工作内容。当前活动的工作结果需要进行验证，如果验证通过，则该结果作为下一项活动的输入，继续进行下一项活动，否则返回修改。
   2. 特点  
      强调文档前一个阶段的输出就是下一个阶段的输入，文档是个阶段衔接的唯一信息。所以很多开发人员好象是在开发文档，而不是开发软件，因为要到开发的后期才可以看到软件的“模样”。没有迭代与反馈。瀑布模型对反馈没有涉及，所以对变化的客户需求非常不容易适应。瀑布就意味着没有回头路。管理人员喜欢瀑布模型的原因是把文档理解为开发的速度，可以方便地界定不同阶段的里程碑。
   3. 优点
      1. 有利于人员的组织与管理
      2. 有利于软件开发方法和工具的研究
      3. 可以提高大型软件项目开发的质量和效率
   4. 局限性
      1. 难以解决需求不明确的问题
      2. 缺乏灵活性，开发周期长
      3. 易于陷入“阻塞状态”
      4. 反馈时间长，风险大
      5. 可操作性差
   5. 适用范围
      1. 开发人员熟悉项目产品的应用领域
      2. 用户使用环境稳定项目开发过程很少或不需要用户参与的项目
2. 迭代模型（stagewise model）
   1. 基本概念  
      早在20世纪50年代末期，软件领域中就出现了迭代模型。最早的迭代过程可能被描述为“分段模型（stagewise model）”。迭代模型是RUP推荐的周期模型。被定义为：迭代包括产生产品发布（稳定、可执行的产品版本）的全部开发活动和要使用该发布必需的所有其他外围元素。在某种程度上，开发迭代是一次完整地经过所有工作流程的过程：需求、分析设计、实施和测试工作流程。实质上，它类似小型的瀑布式项目。RUP 认为，所有的阶段都可以细分为迭代。每一次的迭代都会产生一个可以发布的产品，这个产品是最终产品的一个子集。
   2. 特点  
      分阶段进行，每个阶段都执行一个传统的，完整的串行过程，其中都包括不同比例的需求分析、设计、编码和测试等活动
   3. 优点
      1. 降低了在一个增量上的开支风险。如果开发人员重复某个迭代，那么损失只是这一个开发有误的迭代的花费。
      2. 降低了产品无法按照既定进度进入市场的风险。通过在开发早期就确定风险，可以尽早来解决而不至于在开发后期匆匆忙忙。
      3. 加快了整个开发工作的进度。因为开发人员清楚问题的焦点所在，他们的工作会更有效率。
      4. 由于用户的需求并不能在一开始就作出完全的界定，它们通常是在后续阶段中不断细化的。因此，迭代过程这种模式使适应需求的变化会更容易
   4. 局限性
      1. 在项目开发早期需求可能有所变化。
      2. 分析设计人员对应用领域很熟悉。
      3. 高风险项目。
      4. 用户可不同程度地参与整个项目的开发过程
      5. 使用面向对象的语言或统一建模语言（Unified Modeling Language，UML）

**具有高素质的项目管理者和软件研发团**

## 三、UP统一过程开发模型

1. 基本概念

统一过程起源于1967年的Ericsson方法，核心思想是将复杂的系统构造成为一组组相互联系的功能块，小的功能块相连形成更大的功能块以构造出完整的系统。尽管对于构成这个系统的每个部分的每个成员来说，可能接触的只是系统的一小部分，也就是说，从底层开发人员来讲，整个系统可能不可理喻的，但是每个人又十分的清楚系统中的小的组件是怎样联系工作的。

1. 特点

“分而治之”的主要思想和基于组件的开发方法的有机结合。可以用二维坐标进行描述，横轴是过程所展开的生命周期特征，体现开发过程的动态结构，用周期、阶段、迭代、里程碑等术语进行描述，纵轴是以内容进行组织的逻辑活动，可以用活动、产物、工作者、工作流等进行描述。

1. 优点

提高了团队的生产力，在软件迭代的开发过程中，需求管理，基于组件的体系结构、可视化软件建模、软件测试等方面，针对关键的开发工作为开发人员提供了十分必要的准则、模板或者说指导，为开发过程提供了较大的通用性。

1. 局限性

统一过程开发模型顾名思义只是实现了一个开发过程的模型，并没有涵盖软件过程的全部内容，同时统一过程模型在遇到多项目的情况的时候，有些力不从心，降低了在开发组织内的大范围重用的可能性。

## 四、敏捷开发

1. 基本概念

敏捷开发是针对传统的瀑布流开发模式的弊端而产生的一种新的开发模型，目标是提高开发效率和响应能力。除了原则和实践，开发模式也是相当重要的。敏捷开发以人为核心、迭代、循序渐进的开发方法。在敏捷开发中，软件项目的每一个子项目都不是一个中间过程，都是一个经过测试、运行的结果。

1. 特点

以人为核心、迭代、循序渐进的开发方法。在敏捷开发中，软件项目的每一个子项目都不是一个中间过程，都是一个经过测试、运行的结果。

1. 优点

敏捷开发的迭代速度十分可观，提高了软件系统迭代的速度，可以极快的的对客户需求进行响应。

1. 局限性
2. 项目的人数不能太多
3. 项目经常发生变更
4. 敏捷开发中沟通十分的重要，尤其是业务团队和技术团队，量翻番应对项目活动的理解应该是一致的，否则会导致整个项目的崩溃。

## XP极限编程

1. 基本概念

极限编程是一种轻量级的、灵巧的、简单的软件工程方法，适用于规模小，需求变化大、质量要求严的项目，它要求以最高的效率和极大的灵活性和最小的代价来满足用户未来的需求。是敏捷开发的代表。

1. 特点

轻量、高效、低风险、柔性、可预测、科学而充满乐趣的软件开发方法。”Kent Beck建议XP应用于规模小、进度紧、需求变化大、质量要求严格的项目。关键词是“沟通，追求有效的沟通；简单，简单的可行方案；反馈，快速有效的反馈；勇气，勇于放弃和重构。”

1. 优点

（1）重视客户的参与；

（2）重视团队合作和沟通；

（3）制定计划前做出合理预测；

（4）让编程人员参与软件功能的管理；

（5）重视质量；

（6）简单设计；

（7）高频率的重新设计和重构；

（8）高频率及全面的测试；

（9）递增开发；

（10）连续的过程评估；

（11）对过去的工作持续不断的检查。

1. 局限性

（1）以代码为中心，忽略了设计；

（2）缺乏设计文档，局限于小规模项目；

（3）对已完成工作的检查步骤缺乏清晰的结构；

（4）质量保证依赖于测试；

（5）缺乏质量规划；

（6）没有提供数据的收集和使用的指导；

（7）开发过程不详细；

（8）全新的管理手法带来的认同度问题；

（9）缺乏过渡时的必要支持。

## 六、Scrum模型

1. 基本概念

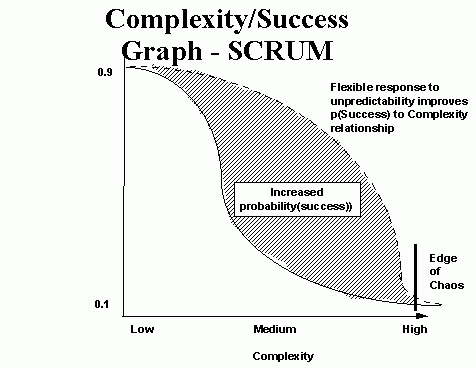
Scrum 是以经验过程控制理论为依据，采用迭代、增量的方法来提高产品开发的可预见性并控制风险。Scrum 的三大支柱支撑起每个经验过程控制的实现。 Scrum是一种轻量级的敏捷开发框架，是一个增量。迭代的开发过程，包含许多迭代周期称为Sprint，每个迭代周期进行总结，探索等。

1. 特点

响应变化，极快的响应变化，随着系统的变化极快的响应客户的需求，团队成员都是多面手，团队关系在一个迭代周期内是固定不变的。有三大支柱：高透明度，管理结果的人看得到哪些影响结果的过程方面；检验，开发过程中的各方面必须做到经常性的检验，以确保及时发现过程中的变化；适应，需要在检验不合格后尽快的调整工作实施。

1. 优点

响应变化，它能够尽快地响应变化。下面的图片使用传统的软件开发模型(瀑布模型、螺旋模型或迭代模型)。随着系统因素（内部和外部因素）的复杂度增加，项目成功的可能性就迅速降低。



1. 局限性

随着项目的复杂的的提高，Scrum开发模型必然无所适从，轻量级的开发限制让Scrum方法无法适用于更多更大的项目，同时极长的迭代周期Sprint也必然让Scrum无法适用于任务紧，时间短的项目中。

**西 安 邮 电 大 学**

**（计算机学院）**

软件过程模型研究开发报告

**专业名称：计算机科学与技术**

**班 级：计科1402**

**学生姓名：王飞**

**学号（8位）：04141056**