

软件工程
第2章 软件生命周期及开发模型

重庆理工大学
计算机科学与工程学院 李梁
qq:1255214405
liliang@cqut.edu.cn
智慧树课号: K3388095
智慧树网址: www.zhihuishu.com

课程号: K3388095



16:45:02 重庆理工大学计算机科学与工程学院 李梁(liliang@cqut.edu.cn) 1

复习预习调查提问?

- 软件工程方法学的三要素是那几个要素?
方法、工具和过程
- 常用的软件开发方法有那两种?
面向过程 (结构化开发方法)、面向对象
- 常用的软件开发模型有那些?
瀑布、原型、螺旋、增加、敏捷。...

16:45:02 重庆理工大学计算机科学与工程学院 李梁(liliang@cqut.edu.cn) 2

1.3 软件工程方法学-软件工程三要素

- 软件工程是把软件看作是一个工程产品,用工程化管理方法
- 软件工程方法学三要素: 方法、工具和过程
 - ✓ 方法: 完成软件开发任务的技术方法, 回答“如何做”问题。提供一系列软件开发技术。包括完成开发过程中各方面任务的方法并用某种特殊的语言或图形来描述。
 - ✓ 工具: 为运用方法而提供的自动或半自动软件工程支撑环境。将多种工具集成在一起可构成计算机辅助软件工程 (CASE) 的软件开发支撑系统。
 - ✓ 过程: 为了获得高质量的软件所需要完成的一系列任务的框架, 规定了完成各项任务的工作步骤, 回答“何时做”问题。将软件工程方法和工具综合起来, 进行软件开发。

16:45:02 重庆理工大学计算机科学与工程学院 李梁(liliang@cqut.edu.cn) 3

1.3 软件工程方法学-基本目标

- 软件工程基本目标: 以工程化思想进行软件开发, 生产高质量和效率的软件。
 - ✓ 高质量是最大程度地满足客户需求、帮助客户获得成功。
 - ✓ 高效率就是以最小的成本获得最好的收益, 即在规定的时间内和规定(资源)预算内, 完成规定的功能的软件开发、维护和服务等任务—三个规定

16:45:02 重庆理工大学计算机科学与工程学院 李梁(liliang@cqut.edu.cn) 4

1.4 开发方法的特点与选择-开发方法

- 通常把在软件生命周期全工程中使用的一整套技术方法的集合称为软件开发方法, 也称为软件工程开发范型。
- 范型: 模型、模式, 就是一套实现具体任务的技术、规范的集合。
- 目前使用得最广泛的软件工程方法分别是面向过程的方法学和面向对象方法学。
- 结构化开发方法 (面向过程方法): 其技术要么面向行为, 要么面向数据
- 面向对象开发方法 (范型): 将对象视作一个融合了数据及在其上操作的行为的、统一的软件组件。数据和行为同等重要

16:45:02 重庆理工大学计算机科学与工程学院 李梁(liliang@cqut.edu.cn) 5

1.4 开发方法的特点与选择-结构化方法

- 结构化方法的特点
 - 按照功能分解的原则;
 - 自顶向下、逐步求精;
 - 直到实现软件功能为止。
 - 简单、实用。
- 结构化方法主要问题
 - 它以功能分解为基础, 而用户的功能是经常改变的, 必然导致系统的框架结构不稳定。
 - 从数据流程图到软件结构图之间的过渡有明显的断层, 导致设计回溯到需求有困难。

16:45:02 重庆理工大学计算机科学与工程学院 李梁(liliang@cqut.edu.cn) 6

1.4 开发方法的特点与选择-面向对象方法

- 面向对象方法学（面向对象范型）
- 特征：将对象视作一个融合了数据及在其上操作的行为的、统一的软件组件。

(1) 把对象作为融合了数据及在数据上的操作的软件构件，即用对象分解取代了传统方法的分解。

(2) 把所有对象都划分成类。

(3) 按照父类与子类的关系，把若干个相关类组织成一个层次结构的系统。

(4) 对象彼此间仅能通过发送消息互相联系。

16:45:02 重庆理工大学计算机科学与工程学院 李黎(liliang@cqu.edu.cn) 7

1.4 开发方法的特点与选择-面向对象方法

- 面向对象范型
- ✓ 优点：
 - 对象的概念符合业务或领域的客观实际，将现实世界问题向面向对象解空间直接映射，实现对现实世界的直接模拟。
 - 以数据为中心，而不是基于对功能的分解
 - 使得软件结构相对稳定，软件的重用性、可靠性、可维护等特性都较好。
- ✓ 缺点：开发复杂大型系统比较困难
- 开发出的软件产品是由许多小的，相对独立的单元（对象）组成的。因此，面向对象范型降低了软件产品的复杂度，从而简化了软件开发与维护工作。

16:45:02 重庆理工大学计算机科学与工程学院 李黎(liliang@cqu.edu.cn) 8

1.4 开发方法的特点与选择-选择

- 开发队伍最熟悉哪种开发方法？
- 提供的软/硬件资源和开发工具？
- 开发方法在计划、组织和管理方面的可行性？
- 对开发项目所涉及领域的知识的掌握情况？

16:45:02 重庆理工大学计算机科学与工程学院 李黎(liliang@cqu.edu.cn) 9

小结

- 软件工程的宗旨是以工程化思想做指导，以高质量和高效率的开发软件。
- 软件工程化思想的核心是把软件看作为一个工程产品。
- 传统软件工程方法学包含结构化范型和面向对象范型。
- 软件工程的研究课题
 - ① 形式语言的研究
 - ② 体系结构与架构技术
 - ③ 质量保证（可靠性研究）
 - ④ 过程控制

16:45:02 重庆理工大学计算机科学与工程学院 李黎(liliang@cqu.edu.cn) 10

小结-开源软件（自学）

- 开源软件，也称为开放源代码软件，是指其源代码对公众开放的软件。这意味着任何人都可以获取、使用、修改和分发这些软件的源代码。开源软件的特点是允许用户自由地对其进行二次开发和分发，同时通常伴随着一个开源许可证，该许可证规定了用户在使用、修改和分发软件时的权利和限制。
- 开源软件与闭源软件相对，后者是指源代码不对外开放，用户只能使用而不能对其进行修改或分发的软件。开源软件的开发和维护通常由全球的开发社区进行，这些社区通过开放协作和透明度来推动软件的发展。
- 开源软件的应用非常广泛，包括操作系统、编程语言、办公软件、图像处理软件等。开源软件的发展不仅促进了技术创新，也促进了全球开发者的交流与合作，同时，由于其源代码的公开性，也使得软件的安全性和可靠性得到了更好的保障。
- 重要的开源社区（微软）：Github（开源仓库：面向开源及私有软件项目的托管平台）
- GitHub于2008年4月10日上线，除了Git代码仓库托管及基本的Web管理界面以外，还提供了订阅、讨论组、文本渲染、在线文件编辑器、协作图谱（报表）、代码片段分享（Gist）等功能。到2022-11-17注册用户已经超过9400万，托管版本数量也是非常之多，其中不乏知名开源项目Ruby on Rails、jQuery、python等。2018年6月4日微软75亿收购
- 常用的18个开源仓库电子书：<https://baijiahao.baidu.com/s?id=1761440118550869203&wfr=spider&for=pc>
- 国内常用开源社区：<https://www.cnblogs.com/shaoyunpeng/p/6026187.html>及<https://m.163.com/dy/article/H4CFRHB705149FJG.html>

16:45:02 重庆理工大学计算机科学与工程学院 李黎(liliang@cqu.edu.cn) 11

小结-常用CASE工具（自学）

- 建模工具：如 Rational Rose / Microsoft Visio / StarUML 等。
- 原型制作工具：Balsamiq Mockups / ExtJS / jQuery / EasyUI / 墨刀 等。
- 集成开发环境：Visual C++ / Visual Studio.NET / My Eclipse 等。
- 数据库设计工具：PowerDesigner / ERWin 等。
- 代码生成工具：Rational Rose（含 C++/Java 等代码生成模块）等。
- 自动化测试工具：Rational RunNer/LambdaTest/ TestComplete 等平台；JUnit / JMeter 等工具。
- 配置管理工具：Git / GitHub / GitEE 等。
- 项目管理工具：Microsoft Project 等。
- AI编程工具：GitHub Copilot、OpenAI Codex、AI Robocode、AlphaCode（Deepmind lab）、AskCodi、CodeT5、Codota、Polycoder、Ponicode、PyCharm、CodeArts Snap、aiXcoder
- Tabnine（AI assistant for software developers | Tabnine：<https://www.tabnine.com/>）基于AI的代码补全工具

16:45:02 重庆理工大学计算机科学与工程学院 李黎(liliang@cqu.edu.cn) 12



第2章 软件生命周期及开发模型

- 2.1 软件开发的认知模型
- 2.2 软件生命周期
- 2.3 瀑布模型
- 2.4 原型模型
- 2.5 螺旋模型
- 2.6 增量模型
- 2.7 其它模型 (自学)

16:45:02

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

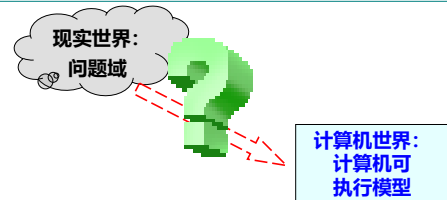
13



2.1 软件开发的认知模型

软件系统的开发过程就是根据**需要解决的问题**，建立一个有软件技术支撑、与解决问题相关的、数据处理的、可运行的**计算机模型**。

关键：将**现实世界**用某种**表达方式**（**标准格式**：如结构化文字、图表）**规范、无异议**进行描述（**抽象**）成软件方案—**模型**



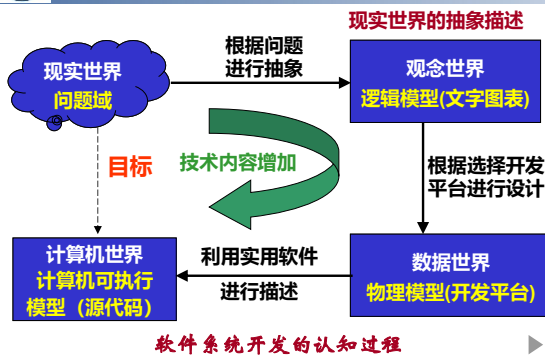
16:45:02

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

14



2.1 软件开发的认知模型



软件系统开发的认知过程

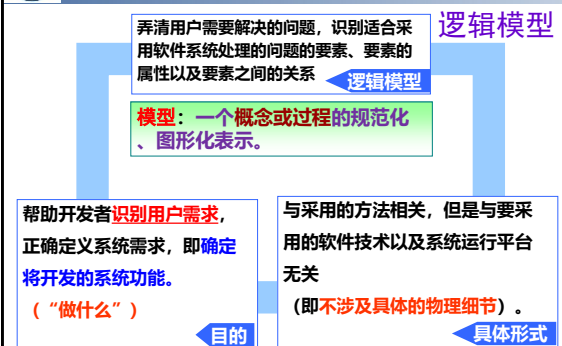
16:45:02

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

15



2.1 软件开发的认知模型



16:45:02

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

16



2.1 软件开发的认知模型

物理模型

在考虑信息技术支撑、人机分工的基础上，按照软件技术的方法、规则将观念世界的系统（即**逻辑模型**），设计**出物理模型**。

物理模型

（“如何做”或“怎么做”）

目的

与采用的方法相关，也与将采用的软件技术以及系统运行平台有关。

具体形式

16:45:02

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

17



2.1 软件开发的认知模型

逻辑和物理的区别

- **逻辑模型**描述系统要“**做什么**”，或者说系统具有哪些功能
- **物理模型**描述系统“**如何做**”，或者说如何实现这些功能

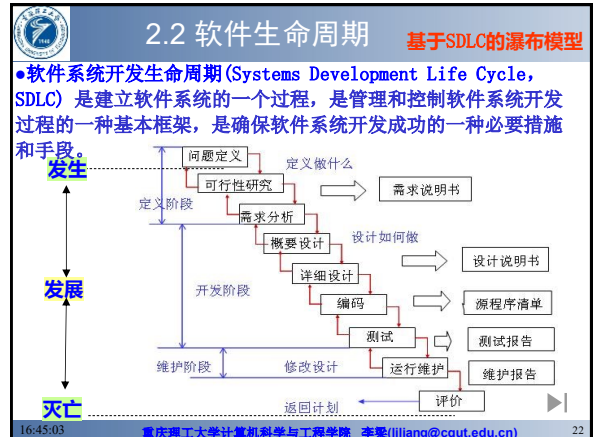
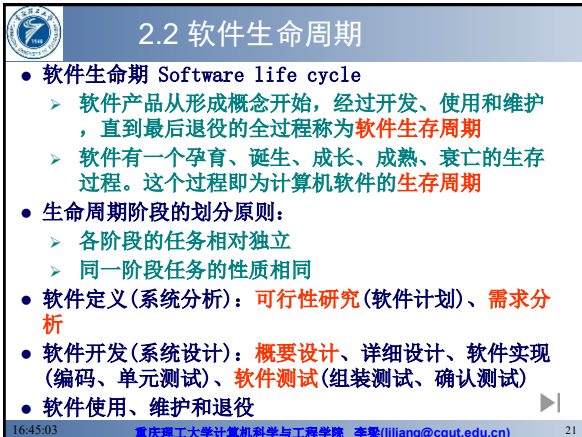
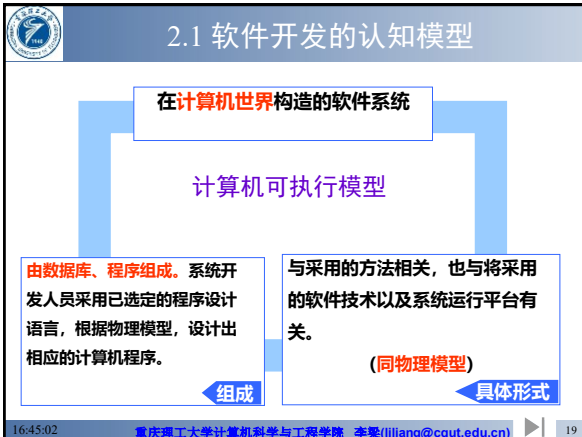
如**读者查询**的逻辑过程主要包括：查什么内容？从什么表（文件）中获取需要查询的信息，读者的逻辑结构如何设计等

而**读者查询**的实现细节：如数据库存放在外存的位置，查询信息输出的设备，读者查询的界面设计等问题则属于“物理”和“如何做”的问题。

16:45:02

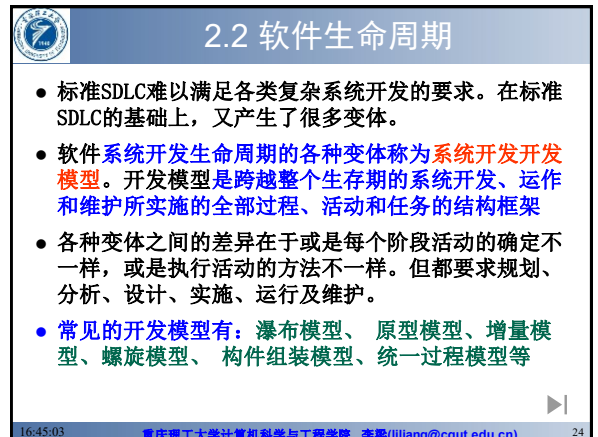
重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

18



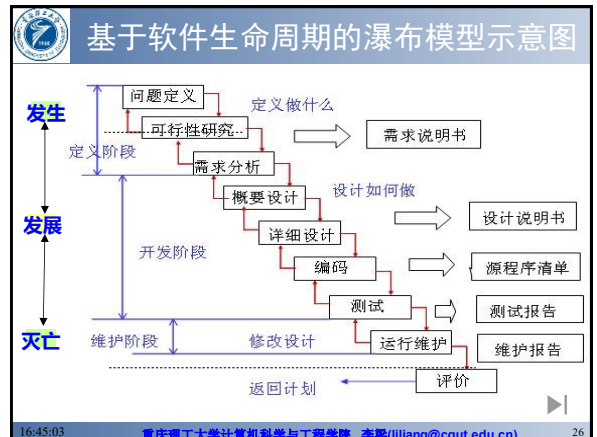
| 阶段 | 关键问题 | 结束标准(任务) |
|----------|--------------------|---|
| 问题定义 | 要解决 问题是什么? | 关于规模和目标的报告书 |
| 可行性研究 | 经济、技术、管理、社会环境是否可行? | 系统的高层逻辑模型; 数据流图; 成本/效益分析 |
| 需求分析 | 系统必须要做什么? | 系统的逻辑模型; 功能、性能、接口、约束。数据流图; 类图; 数据字典, 算法描述 |
| 概要(总体)设计 | 如何解决问题? | 系统结构(子系统划分)及数据结构(共享数据组织)。系统流程图; 层次图和结构图。 |
| 详细设计 | 怎样具体的实现 | 模块设计、数据库设计, UI |
| 编码和单元测试 | 正确的程序模块 | 源程序清单; 单元测试方案和结果 |
| 测试 | 符合要求的软件 | 综合测试方案和结果; 单元测试、集成测试及验收测试完整一致的软件配置 |
| 运行、维护 | 持久地满足用户需要 | 完整准确的维护记录, 日志 |

16:45:03 重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn) 23



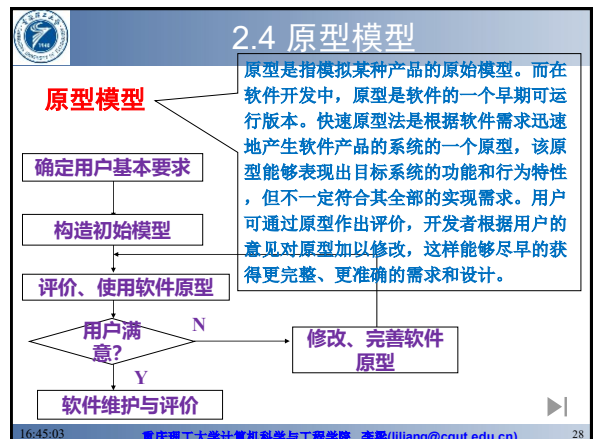
2.3 瀑布模型

- 瀑布模型就是**传统的生命周期模型**（自顶向下结构化开发模型方法）。瀑布模型提供了软件开发的基本框架。
- 瀑布模型将软件生命周期划分为软件计划、需求分析和定义、软件设计、软件实现、软件测试、软件运行和维护这**6个阶段**，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而得名
- 它是一个**软件开发架构**，开发过程是通过**一系列阶段顺序**展开的
- 每个阶段都会产生**循环反馈**
- 各个阶段产生的**文档**是维护软件产品时必不可少的，没有文档的软件几乎是不可维护的。
- ✓ **瀑布模型特点**：顺序性和依赖性、推迟实现、质量保证的观点、是一种线性模型、**强调文档的作用**。是一个**文档驱动模型**！



2.3 瀑布模型（生存周期模型）

- 优点**：奠定了软件工程方法的基础；流水依赖；便于分工协作；推迟现实；文档易修改；有复审质量保证。强迫开发人员采用规范的技术方法；严格地规定了每个阶段必须提交的文档；每个阶段结束前必须正式进行严格的技术审查和管理复审。
- 缺点**：用户需求明确困难；用户看见效果晚；发现错误晚，纠错慢。难于克服系统分析员不懂专业领域的知识，适合于系统要求明确的小系统。在可运行的软件交付用户之前，用户只能通过文档来了解未来的产品是什么样。开发人员和用户缺乏有效的沟通，很可能导致最终开发出的产品不能真正满足用户的需求。

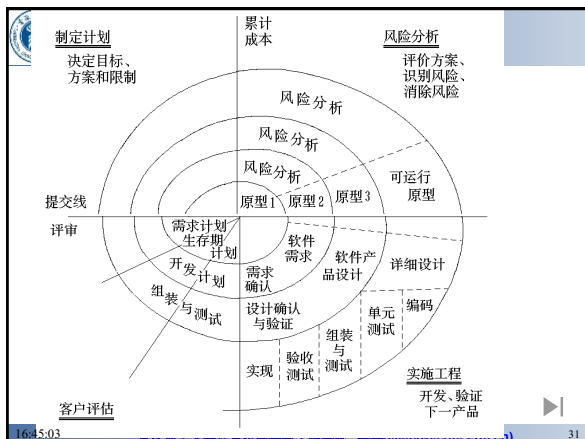


2.4 原型模型

- 快速原型的特性**：
 - ① **“快速”**：尽快向用户提供一个可在计算机上运行的目标系统的模型，以便使用户和开发者尽可能快地达成对目标系统“做什么”的共识。
 - ② **“易于修改”**：如果原型的第一版不是用户所需要的，就必须根据用户的意见迅速地修改它，构建出原型的第二版。
- 原型模型存在的问题**
 - 用户看到的不是**实际运行**的系统。
 - 为了提高原型的开发速度，开发人员很可能采用**不理想**的选择。

2.5 螺旋模型

- 螺旋模型（Spiral Model）**是**迭代开发模型**，瀑布模型 + 快速原型模型。
- 特点**：特别适合于大型复杂的系统，强调了其他模型忽略了的风险分析：
 - ✓ 风险识别
 - ✓ 风险分析
 - ✓ 风险控制
- 构成**：包括需求定义、风险分析、工程实现和评审查等4个阶段。此4个阶段就构成1个周期，每个周期的完成就前进1层，从而构成螺旋上升的**迭代过程**。



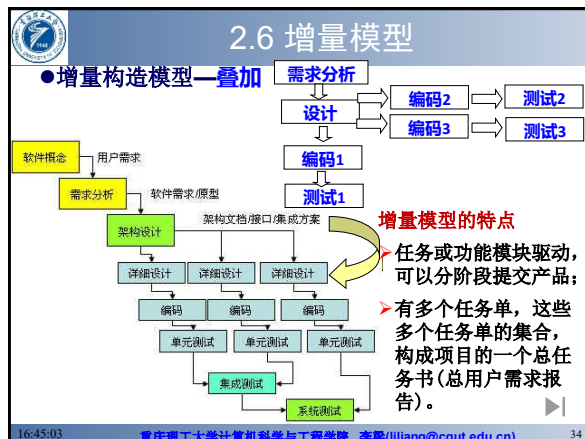
2.5 螺旋模型

✓ 螺旋模型沿着螺旋线旋转（一个螺旋式周期），在四个象限上分别表达四个方面的活动，即：

- **确定目标，制定计划：** 确定软件目标，选择实施方案，确定项目开发的约束、限制条件，选定完成目标的策略；
- **评估方案/风险分析：** 识别、分析评估并排除风险；分析所选方案，考虑如何识别和消除风险，风险角度分析该策略
- **开发实施：** 实施软件开发和验证产品；
- **客户评估，制定下期计划：** 评价本期工作，提出修正建议，制定下步计划螺旋模型是风险驱动的模式。

2.6 增量模型

- 增量模型也称为新增模型，是在项目的开发过程中以一系列的增量方式开发系统。
- **核心思想：** 软件被作为一系列的**增量构件**来设计、实现、集成和测试，每一个构件是由多种相互作用的模块所形成的提供特定功能的代码片段构成。
- ✓ 增量模型融合了线性瀑布模型的基本成份和原型实现模型的迭代特征。
- ✓ 增量模型采用随着日程时间的进展而**交错**的线性序列。每一个线性序列产生软件的一个可发布的“增量”。
- ✓ 增量模型强调每一个增量均发布一个可操作产品。早期的增量是最终产品的“可拆卸”版本，但他们确实提供了给用户服务的功能，并且提供了给用户评估的平台。
- ✓ 将软件产品看作**一组增量构件**，每次设计、实现、集成、测试和交付一块构件，直到所有构件全部实现为止。



2.6 增量模型

增量是**逐块建造**的概念，迭代是**反复求精**的概念。实际上现在很多迭代开发都是**增量&迭代**。

The diagram shows two iterations of building the Mona Lisa painting. In the first iteration, a person is shown adding a piece to the painting, with a thought bubble saying 'Automatic or personal setting'. In the second iteration, the person is shown adding another piece, with a thought bubble saying 'Automatic or personal setting'.

第1章 小结

- **软件：** 程序+数据+文档
- **软件工程：** 软件工程是指导计算机软件开发和维护的一门工程学科，该学科的目的是生产出能按期交付的、在预算范围内的、满足用户需求的、质量合格的软件产品。
 - ✓ 在软件工程领域中通常由具有另一种文化背景的人具有另一种文化背景的人创造产品。
 - ✓ 方法（面向过程、面向对象）、工具和工程
- **软件生命周期：**
 - ✓ 可行性研究（软件计划）、需求分析（是什么）、概要设计（做什么）、详细设计、软件实现（编码、单元测试）、软件测试（组装测试、确认测试）、件使用、维护和退役
- **常用软件四种开发模型：** 瀑布模型、原型模型、增量模型、螺旋模型



小结

- 软件开发模型是指软件开发全部过程、活动和任务的结构框架，能清晰、直观地表达软件开发全过程，明确规定了要完成的主要活动和任务，用来作为软件项目工作的基础。
- 瀑布模型是一种线性模型，文档驱动模型。
- 增量提交模型采用一系列的增量方式开发系统。
- 螺旋模型结合瀑布和快速原型，是一种风险驱动的迭代开发模型
- 目前软件企业中普遍采取的开发与运维方式：**DevOps**
<https://www.zhihu.com/question/58702398> 敏捷开发：开发+运维

16:45:03

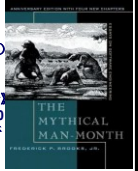
重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

37



小结-软件工程发展过程（自学）

- 1968年NATO的计算机科学家在联邦德国召开国际会议，讨论软件危机问题，正式提出了“软件工程”这一术语。
- 1970年罗伊斯（Royce）提出了著名的瀑布模型（Waterfall Model），直到80年代早期一直是唯一被广泛采用的软件开发模型。后来，又有人针对该模型的不足，提出了快速原型法、螺旋模型等。
- 1975年，布鲁克斯（Brooks）写出了软件工程领域著作《人月神话》（The Mythical Man-Month），作者根据自己在IBM公司System 360家族和OS 360中的项目管理经验，以随笔集的形式对大型软件项目开发提出独到见解。“软件开发被喻为让众多史前巨兽痛苦挣扎，却无力摆脱的焦油坑。”
- 20世纪70年代，结构化软件开发方法学形成。
- 20世纪80年代，面向对象程序设计语言出现。Smalltalk、C++软件重用技术被提出，例如面向对象程序语言的可重用类库等；软件工程标准化工作活跃，如CMU SEI的SW-CMM等。
- 20世纪90年代，随着计算机网络技术的发展，以CORBA为代表的分布式面向对象技术发展和流行；以Java和C++为代表的面向对象技术成为主流；开源软件和技术出现，如Linux出现并产生广泛影响；人机交互技术发展，Windows操作系统为代表的基于窗口的用户界面技术。
- 1995年，GoF写出了设计模式经典著作《Design Patterns: Elements of Reusable Object-Oriented Software》。



16:45:03

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

38



小结-软件工程发展过程（自学）

- 1997年1月，UML 1.0 (Unified Modeling Language 统一建模语言) 发布，它解决了开发人员之间互相交流的问题，它包括用例图、类图、活动图、状态图、时序图、协作图、组件图和部署图等。同时，统一过程提出并细化。
- 传统软件工程（软件工程1.0）主要向土木工程、工业工程学习，吸收其百年实践积累下来的实践方法和经验、沉淀下来的思想。软件工程1.0体现了下列一些特征：
 - 产品化：交付符合质量标准的组件、构件和系统
 - 过程决定结果：流程质量决定产品质量，一环扣一环，相信良好的过程产生良好的产品，关注过程胜过关注人，也非常关注过程评估和过程改进，如CMM（过程成熟度模型）是其典型代表。
 - 阶段性明确，需求评审通过了，才能开始设计；设计评审通过了，才能开始实施（编程），编程结束了再进行测试等，如瀑布模型是其典型代表。
 - 责任明确：角色定义清楚，分工细致；
 - 文档化：强调规范的文档，大量的文档模板；
 - 计划性强：完整的计划，严格控制变更；
 - 建筑工程的框架结构设计，在软件工程上表现为以架构设计为中心。
 - 现代制造业流水线的启发——软件工厂思想。
 - 注重项目管理：围绕项目开展管理工作，包括风险预防、里程碑控制和关键路径法等。
 - 以顾客为中心的全面质量管理。
 - 预防为主，检验为辅，即缺陷预防思想。

16:45:03

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

39



小结-软件工程发展过程（自学）

- 现代软件工程（软件工程2.0 企业实践）
- 受互联网、开源软件运动、敏捷/DevOps开发模式所影响并最终形成的、建立在SaaS（软件即服务）、Cloud（云）之上的软件工程定义为“软件工程2.0”。
- 没有互联网（Internet），就没有Cloud、没有SaaS，我们对软件就不能部署在软件研发公司自己的数据中心，持续交付（CD）就没有意义，因为无法做到将包装盒形式的软件产品持续交付到客户手中，敏捷、DevOps也就难以实施，虽然可以在内部实现持续集成。
- 开源软件运动让我们认识到“软件过程”、“软件管理”不是非常重要，至少不是第一要素，第一要素还是人，其次是软件架构（微服务、serverless等）和代码质量，使软件容易演化（如架构可扩展性、代码的可维护性）。
- 交付高质量的产品，又认可软件的灵活性，希望软件能够具有随需应变的能力，及时进行必要的修改来满足业务的新需求。同时，软件又是一种知识型产品，需要创造性，并依赖每个研发人员的创造力和积极性。所有这些引导人们新的思考，引导人们不断认识软件工程，从而在2001年17位软件开发轻量级流派掌门人联合签署了敏捷宣言：之后逐渐形成了敏捷/DevOps开发模式、精益软件开发模式等，即让软件工程进入2.0时代。



敏捷软件开发宣言

我们一直在实践中探索更好的软件开发方法，身体力行的同时也帮助他人。由此我们建立了如下价值观：

- 个体和互动 高于 流程和工具
- 工作的软件 高于 详尽的文档
- 合作 高于 合同谈判
- 响应变化 高于 遵循计划
- 与他人合作 胜过与机器合作

16:45:03

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

40



小结-软件工程发展过程（自学）

- 2000后，软件开发和运行从单机环境扩展到互联网环境，软件从产品到服务，如微信和Google Docs等；敏捷开发方法被应用到软件开发中；DevOps方法在产业得到广泛应用；智能化软件开发和群体化软件开发不断发展。
- 软件工程2.0的特征可以简单概括为以下几点：
 - SaaS（软件即服务）：软件更多是以一种服务存在；
 - 以人为本：个体与协作胜于流程和工具，充分发挥个人和团队的创造性和潜力；
 - 拥抱变化：敏捷开发或轻量级过程，加速迭代，以不变应万变。
 - 持续性：阶段性不明确，持续构建、持续集成、持续测试、持续交付，以时间换空间，消除市场风险；
 - 融合：强调测试开发融合，开发与运维融合，推崇全栈工程师等。
 - 真正把用户放在第一位：用户、产品经理尽可能参与团队研发过程中，注重用户体验，千人千面；
 - 强调价值交付：只做对用户有价值的的事情，加速价值流的流动；
 - 知识管理：将软件工程纳入知识管理的范畴。
 - 史诗般故事、用户故事、站立会议，让软件研发工作更有趣、更健康。
 - 考虑到它的发展速度，在不久的将来，甚至从今天开始，AI就开始逐渐接手一些软件研发的工作，我们通过将GPT-4+融入到软件研发生命周期中，研发人员的使命发生变化，GPT-4+重新定义了开发人员构建、维护和改进软件应用程序的方式。之后的软件开发会依赖这种全新的语言交流方式（类似chatGPT那样），让这类工具理解研发人员交待的任务、自主完成软件开发，如理解需求、自动生成UI、自动生成产品代码、自动生成测试脚本等。此后，研发团队的主要任务不是写代码、执行测试，而是训练模型、参数调优、围绕业务主题提问或给提示（prompt）。因此，GPT-4开启“软件工程3.0”新时代

16:45:03

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

41



小结-软件工程发展过程（自学）

- 敏捷开发是一种思想或方法论，通过不断迭代开发和增量发布，最终交付符合用户价值的产品。由此可见，敏捷开发本身的目标并不是提升研发效能，而是持续交付价值给客户。
- 敏捷首先是思想，是文化，其次才是实践方法和技术。敏捷思想或者敏捷文化鼓励创新，鼓励大胆尝试，从这一点上来说，真正的敏捷一定会促进研发效能的提升。
- 在精益软件开发中有个湖水岩石效应的经典隐喻，水位代表库存多少，岩石代表问题。水位高的时候，岩石就会被隐藏，即库存多时，设备运转不良、上一环节输出的质量差、停工等待、供应不及时等问题都会被掩盖起来。如果是精益或敏捷开发模式，在制品快速流动，同时限制在制品的数量，没有了临时库存的缓冲，就会出现“水落石出”的局面：生产或开发中的瓶颈都会一一暴露出来。



敏捷开发是一种思想或方法论，通过不断迭代开发和增量发布，最终交付符合用户价值的产品。由此可见，敏捷开发本身的目标并不是提升研发效能，而是持续交付价值给客户。

敏捷首先是思想，是文化，其次才是实践方法和技术。敏捷思想或者敏捷文化鼓励创新，鼓励大胆尝试，从这一点上来说，真正的敏捷一定会促进研发效能的提升。

在精益软件开发中有个湖水岩石效应的经典隐喻，水位代表库存多少，岩石代表问题。水位高的时候，岩石就会被隐藏，即库存多时，设备运转不良、上一环节输出的质量差、停工等待、供应不及时等问题都会被掩盖起来。如果是精益或敏捷开发模式，在制品快速流动，同时限制在制品的数量，没有了临时库存的缓冲，就会出现“水落石出”的局面：生产或开发中的瓶颈都会一一暴露出来。

16:45:03

重庆理工大学计算机科学与工程学院 李黎(lililang@cqu.edu.cn)

42

小结-软件工程发展过程（自学）

软件工程 1.0 软件工程 2.0 软件工程 3.0

1968年 NATO会议 “软件工程”学科诞生 2004年 敏捷宣言发布 2023年 ChatGPT/GPT-4发布

- 软件工程 3.0的体现了下列一些特征：
- 数字化：软件研发平台开始能够理解需求、设计、代码等，软件研发从过去的信息化进入数字化时代；
- AIGC：生成软件（software）的各种ware；验收标准、测试用例、UI、代码、测试脚本等；
- 极致的持续交付：虽然软件工程2.0开始面向CI/CD，但还存在许多障碍，而在软件工程3.0，得益于设计、代码、测试脚本等生成，可以真正实现持续交付，即及时响应客户需求，交付客户所需的功能特性；
- 人机交互智能：软件研发过程就是人与计算机的交互过程；
- 以模型和数据为本：研发人员服务于大模型和大数据平台，包括模型创建、训练、调优、使用等；

作为一个语言模型，GPT-4 在软件工程方面可能会有以下影响：

1. 自动代码生成：GPT-4 可以通过理解程序员的自然语言输入，自动地生成代码。这可能会加速开发过程，提高软件工程的效率。
2. 基于自然语言处理（NLP）的编码辅助：GPT-4 可能会提供基于自然语言处理的编码辅助，通过建议代码片段、函数和 API 来帮助开发人员。这可能有助于减少错误，提高代码质量，提高生产率。
3. 测试和调试：GPT-4 可以通过根据自然语言输入自动生成测试用例，并更快、更准确地识别 bug 的根本原因，从而改善测试和调试。
4. 设计和架构：GPT-4 可以通过提供基于自然语言输入的建议来帮助软件工程师设计和架构软件系统。这可能有助于创建更强大、更高效的软件系统。

16:44:03 重庆理工大学计算机科学与工程学院

软件工程中的一些观念——思考、讨论

- 1、如果我们落后于计划，可以增加更多的程序员来解决。
- 2、如果软件运行较慢，是换一台更快的计算机，还是设计一种更快的算法？ **硬件？软件？**
- 3、有最好软件工程方法还是最好的编程语言？
- 4、编程时是否应该多使用技巧？ **好的算法的地位**
- 5、软件中的错误是否可按严重程度分等级？
- 6、确定是否进行一个开发项目时（假如该项目已通过可行性分析），最先考虑的是否有：**时间、掌握了项目需要的开发技术、对项目的应用领域了解程度、合作者情况**
- 7、我可以同时从事多个项目开发吗？
- 8、我是项目经理，为了保密，必须掌握项目的核心模块源代码、开发时，应选择我最熟悉的工具和软件？

16:45:03 重庆理工大学计算机科学与工程学院 李黎(lililiao@cqust.edu.cn)