

# Book1-需求工程导论

## 1. 引入:需求工程与商业模式的链接

1. 需求获取前半段(获取业务需求): 项目前景与范围+涉众分析      商业模式画布
2. 需求获取后半段(获取用户需求): 面谈+原型+观察      商业模式设计
3. 需求分析(获取系统级需求): 利用UML模型细化并精化需求
4. 需求规格说明: 完整描述并记录业务需求、用户需求、系统级需求
5. 需求验证与管理: 常用手段与基线保持      商业模式评估
6. 互联网产品设计与需求工程对比

## 2. 软件生产中的需求问题

### 2.1. 需求问题是当前软件开发的主要问题

1. 需求问题对项目的成败有着比较重要的影响
2. 项目可以划分为
  1. 成功项目
  2. 问题项目
  3. 失败项目

表 1-1 影响成功项目的因素,数据来源于[ Standish 1995]

成功项目的影响因素	影响指数	成功项目的影响因素	影响指数
用户参与	15.9%	员工能力	7.2%
高层管理支持	13.9%	主人翁精神	5.3%
清晰的需求说明	13.0%	清晰的目标和前景	2.9%
正确的项目计划	9.6%	努力工作	2.4%
切合实际的期望	8.2%	其他	13.9%
细化的项目里程碑	7.7%		

表 1-2 影响问题项目的因素,数据来源于[ Standish 1995]

问题项目的影响因素	影响指数	问题项目的影响因素	影响指数
缺少用户输入	12.8%	不切实际的期望	5.9%
不完整的需求说明	12.3%	目标不清晰	5.3%
需求变化	11.8%	不现实的时间要求	4.3%
缺乏高层管理支持	7.5%	新技术的影响	3.7%
技术能力不足	7.0%	其他	23.0%
缺乏资源	6.4%		

表 1-3 影响失败项目的因素,数据来源于[ Standish 1995]

失败项目的影响因素	影响指数	失败项目的影响因素	影响指数
不完整的需求说明	13.1%	缺乏计划	8.1%
缺少用户输入	12.4%	额外的无用功能	7.5%
缺乏资源	10.6%	缺乏 IT 管理	6.2%
不切实际的期望	9.9%	技术能力不足	4.3%
缺乏高层管理支持	9.3%	其他	9.9%
需求变化	8.7%		

## 2.2. 软件的模拟特性

1. 软件的模拟特性来源于**其知识载体的特性**: 软件在运行中表现出来的特性、行为应该和应用的显示情况保持一致。
  1. 比如张三没有借书, 但是软件记录了他借书, 则被认为是不正常的。
2. 软件的冗余功能问题也从另一个侧面很好地反应了它的模拟特性。
  1. 一个软件中接近50%的功能用户从来不使用, 这些冗余功能往往是导致用户不满意和软件不被接受的原因之一。

2. 用户会担忧触发未知或者不确定的功能，“简洁”的魅力

2.2.1. 软件的模拟性的具体指称

- 1. 目的性：软件目标是直接或简介地满足用户的某些目的或者解决用户的某些问题
- 2. 正确性：软件的功能能够保证目标的正确实现
- 3. 现实可理解性：软件系统是在理解其现实环境的基础上，通过影响现实的某些环节，或者改变显示各部分的通信方式，最终达成某些目的或解决某些问题。

2.2.2. 不同类型的软件

表 1-6 不同类型软件的评判标准

软件类别	纯工具型软件		应用型软件
	专业用户	普通用户	
评判标准	功能的复杂性 使用的高效性 技术的先进性	功能的有用性 使用的方便性 技术的可行性	功能的“模拟”性 使用的方便性 技术的可行性
关注点	创新性	有效性	模拟性
示例系统	编程环境 DBMS	Office 语言翻译	MIS EAI

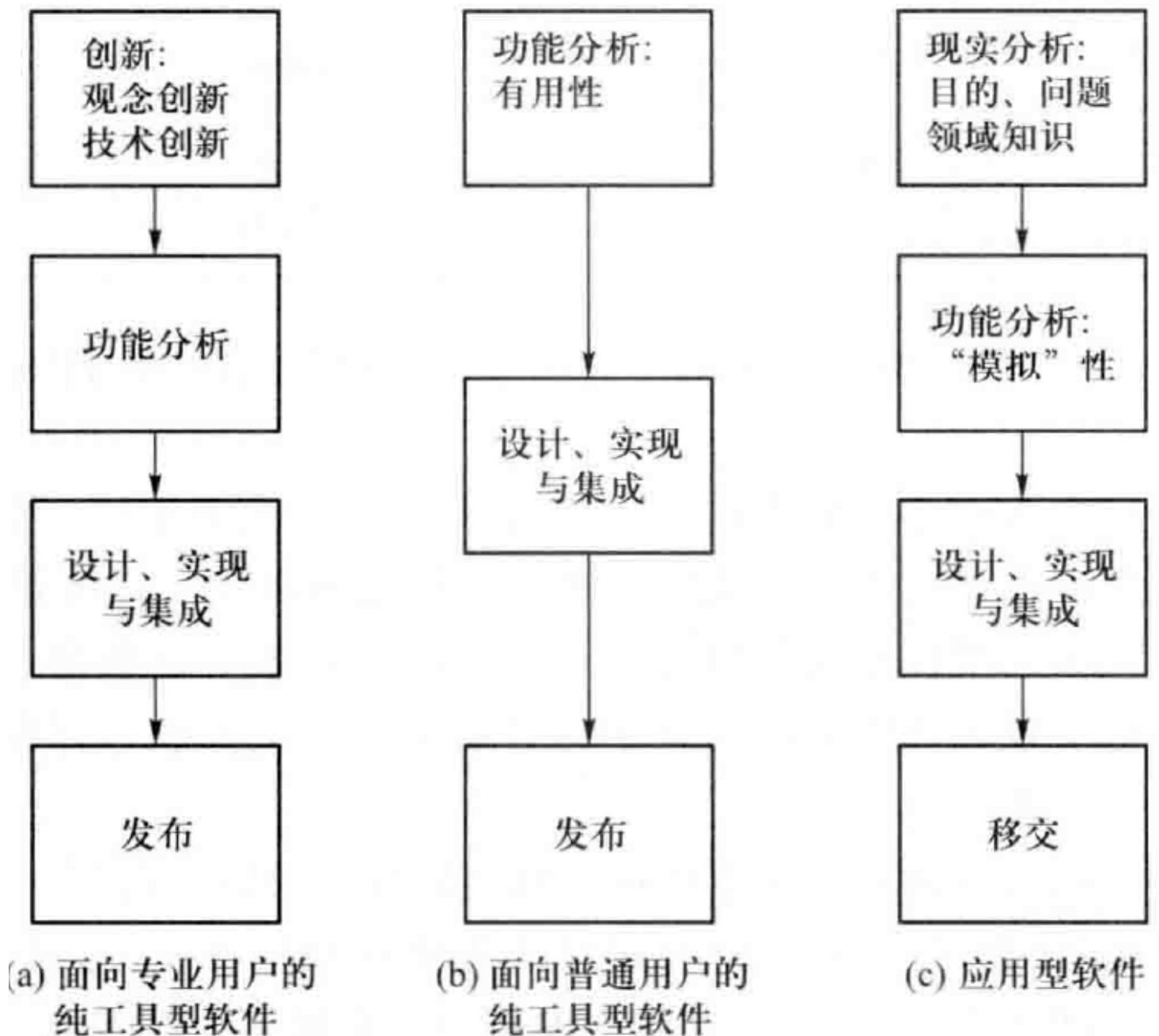


图 1-4 不同类型软件的生产过程

1. 面向专业用户的工具型软件:

1. 首要成功标准: 具有功能的复杂性和使用的高效性
2. 目标: 具有一定的创新优势而进行巧妙的功能安排

2. 面向普通用户的工具型软件:

1. 首要成功标准: 功能的有用性
2. 目标: 进行方案权衡, 寻找一套切实有效的功能配置

3. 应用型软件:

1. 基础是具有模拟性
2. 目标: 发现人们利用软件的原因(目的), 找出需要软件解决的问题。

## 2.3. 需求问题具体原因分析

### 2.3.1. 非技术型和社会性因素重要不足

1. 需求建模与分析需求处理中的**核心活动**，它用一些形式化或半形式化的语言进行**知识描述**。
  1. 只有通过需求与分析活动才能将**混乱、模糊**的用户需求变成**清晰、明确**的软件需求，是获取需求处理活动的**必然后继**，**建立的分析模型**是需求处理中最为重要的成果。
  2. **建模与分析**的理论可以帮助人们系统化的看待问题。
2. 需要考虑的可能导致失败因素包含**组织机构文化、社会背景、商业目标、利益协商**等。
3. 为什么需要重视非技术性和社会性因素
  1. 需求处理的任务角度
    1. 需求处理的主要任务是发现并解决问题。
    2. 需求处理不应该以新系统的功能性和内在特征为主要处理目标，而是应该更集中经历与分析环境的构成、现状和他们将来能与软件达成的期望互动效应。
    3. 关注组织文化、社会背景和系统涉众的目标和利益
  2. 需求处理的手段角度
    1. 建模与分析技术是进行需求处理的主要手段，这些概念本身都是概念性的，不依赖于某些特殊的环境条件。
    2. 不要忽略应用环境中的相关因素。
  3. 需求处理的过程角度
    1. 现实中，因涉众的立场不同而产生的利益冲突的场景十分常见
    2. 面对冲突要能够分析社会原因和组织机构方面的原因，引导涉众进行利益协商，进而建立一个一致、完整的需求模型。

### 2.3.2. 传统需求分析方法的缺陷

1. 传统的需求分析方法都是最先在编程领域取得成功的，后续在设计领域的应用也取得了成功
2. 但是理解现实这个目标是传统分析方法所无法实现的(先天缺陷)
3. 核心：理解现实

### 2.3.3. 软件规模的日益扩大

1. 软件规模的扩大，导致涉众也在扩大，导致相互之间的利益冲突更加剧烈，因此对商业目标和利益协商的处理要求也就更加重要了，需要分别注意各个部分之间的不同考量。
2. 对于以企业为中心，很少有用户可以单独给出全局的解释，需要由需求分析人员结合给出的片段，拼接起来形成一个全局理解，导出需求。

### 2.3.4. 需求问题的高代价性

1. 在维护阶段进行需求修复的代价可能是需求阶段的修复代价的100-200倍

## 3. 需求工程

### 3.1. 需求工程的定义

1. 简单来说，需求工程是所有需求处理活动的总和，它收集信息、分析问题、整合观点、记录需求并验证其正确性，最终反映软件被应用后与其环境互动形成的期望效益。
2. 需求工程是软件工程的一个分支
  1. 它关注于软件系统所应予实现的现实世界目标、软件系统的功能和软件系统应当遵守的约束
  2. 同时它也关注以上因素和准确的软件行为规格说明之间的联系
  3. 关注以上因素与其随时间或跨产品族而演化之后的相关因素之间的联系

### 3.2. 需求工程任务

1. 需求工程必须说明软件系统将被应用的**环境及其目标**，说明用来达成这些目标的**软件功能**，还要说明在设计和实现这些功能时上下文环境对软件完成任务所用的**方式、方法**所施加的限制和约束(做什么，为什么)
2. 需求工程必须将**目标、功能和约束**反映到软件系统中，映射为可行的软件行为，并对软件行为进行准确的**规格说明**
3. 现实世界是不断变化的世界，因此需求工程还需要妥善处理**目标、功能和约束**。同时，为了节省开支和进行需求规格说明的重用，需求工程还需要对目标、功能和约束在软件产品族中的演化和分布情况进行综合考虑与处理。

### 3.3. 基本活动

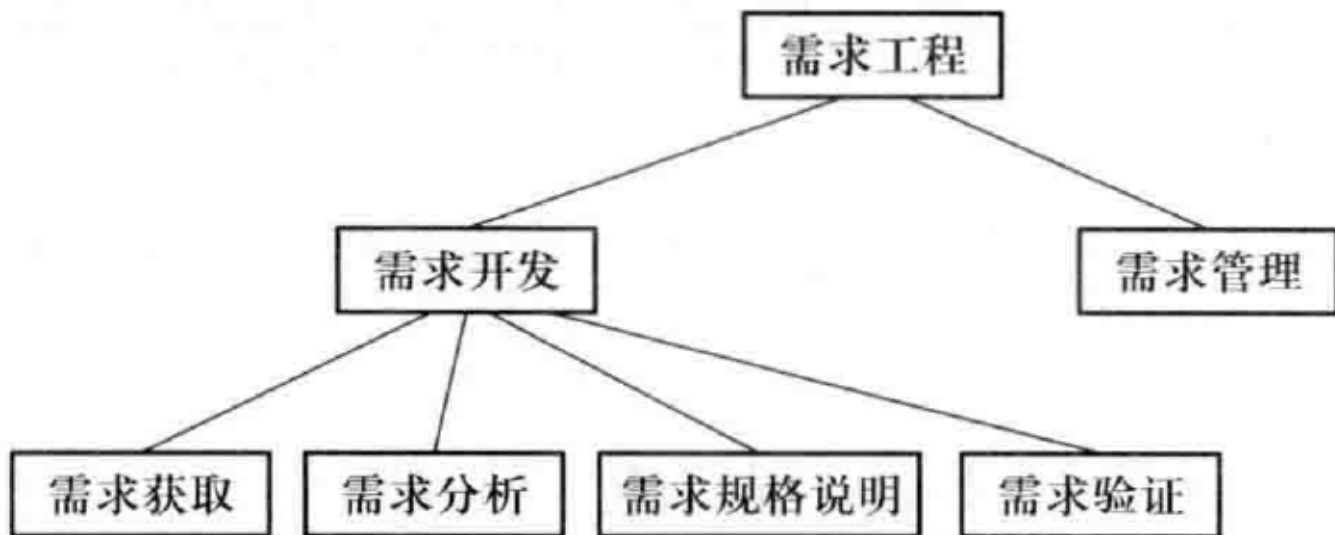


图 1-6 需求工程基本活动

### 需求工程的两方面活动

#### 1. 需求开发

1. 需求获取:目的是从项目的战略规划开始建立最初的原始需求

1. 确定系统涉众
2. 了解现有问题
3. 建立新系统目标
4. 获取为支持新系统目标需要的业务过程细节和具体的用户需求

2. 需求分析:保证需求的完整性和一致性

1. 将需求获取阶段输出的原始需求和业务过程细节出发, 将目标、功能和约束映射为软件行为, 建立系统模型
2. 在抽象的系统模型中进行分析, 标识并修复其中的不一致缺陷, 发现并弥补遗漏的需求。

3. 需求规格说明:需求规格说明的目的是将完整、一致的需求与能够满足需求的软件行为以文档的方式明确地固定下来。

4. 需求验证

1. 首要目的: 保证需求及文档的正确性
2. 另一个目标: 通过检查和修正, 保证文档的完整性和一致性。
3. 需要得到所有的涉众一致同意

2. 需求管理: 对需求开发所建立的需求基线的管理, 在修基线完成后正式开始, 并在需求工程阶段结束后持续存在。

## 3.4. 需求工程与系统工程

1. 系统需求开发又称为需求工程的早期阶段，为了获取整个系统的期望目标，包含功能特征和非功能特征。
2. 系统需求开发的主要目的是获得整个系统的期望目标，包含功能特征和非功能特征。
3. 系统需求开发阶段的需求分析主要是分析系统的成本效率及系统的组织和行政策略,处理互相依赖、冲突、重叠或不一致的涉众需求，检查并弥补需求缺失,检查技术储备、外部系统等环境约束。系统需求开发的结果会写入系统需求规格说明。
4. 系统需求开发阶段获得的需求被分配到软件工程、硬件工程和人力工程部分。

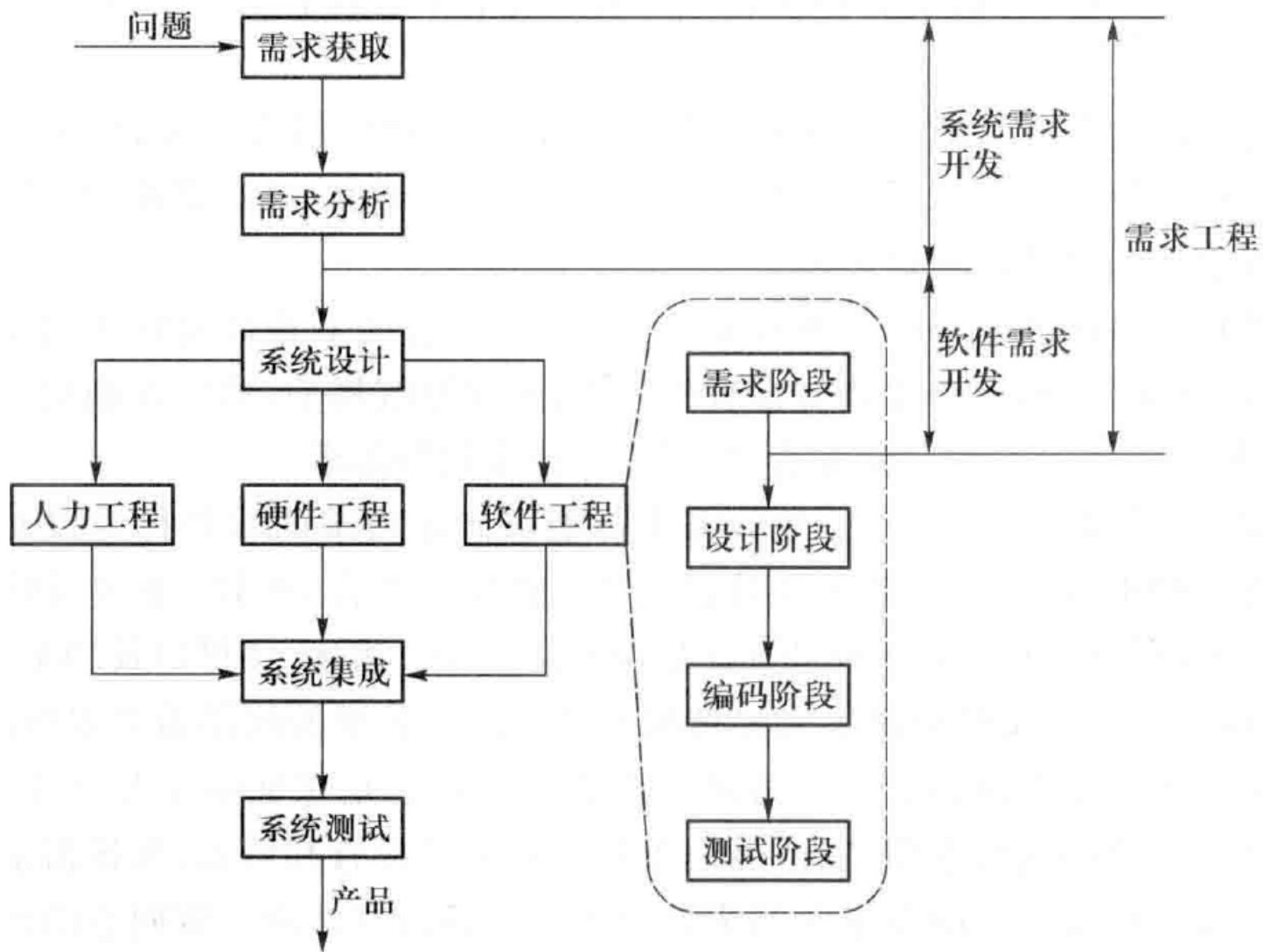


图 1-7 需求工程在系统工程中的位置

### 3.5. 需求工程的重要性

1. 软件开发是一个工程性的问题：利用通用的计算机结构构建一个有用的软件系统来满足人们的目的。
2. 定义问题就是需求工程的任务。



3. 常见的忽视需求工程的例子

- 1. 问题广为人知
- 2. 问题小而简单

4. 文献：

- 1. 1981年，Barry Boehm [Boehm1981]发现项目费用的6%和时间的9-12%被消耗在需求阶段。
- 2. 在20年之后，随着需求工程的发展，[Hofmann2001]发现项目对需求工程的投入也加大了许多：项目工作的15.7%和时间的38.6%被用于进行需求工程
- 3. NASA (U.S. National Aeronutics and Space Administration )提供的数据显示 [Young2002]：当在需求工程当中投入项目总成本的8-14%时，可以极大的降低项目的超支率
- 4. Frederick Brooks[Brooks1987]： "开发软件系统最为困难的部分就是准确说明开发什么。最为困难的概念性工作便是编写出详细技术需求，这包括所有面向用户、面向机器和其它软件系统的接口。同时这也是一旦做错，将最终会给系统带来极大损害的部分，并且以后再对它进行修改也极为困难。"

排序	2010年度		2012年度	
	影响因素	指数	影响因素	指数
1	用户参与	20	高层管理支持	19
2	高层管理支持	15	用户参与	18
3	清晰的业务目标	15	清晰的业务目标	15
4	情感成熟度（Emotional maturity，即项目氛围）	12	情感成熟度（Emotional maturity，即项目氛围）	12
5	最优化(Optimization)	11	最优化(Optimization)	11
6	敏捷过程	11	敏捷过程	9
7	项目管理技能	6	项目管理技能	7
8	有技能的员工	5	有技能的员工	5
9	执行力	3	执行力	4
10	工具与设备	2	工具与设备	1

3.6. 需求工程的复杂性

- 1. 需求工程的复杂性源自于需求处理过程的复杂性
- 2. 复杂性包含
  - 1. 处理范围广泛:需求工程连接现实世界和计算机世界。

1. 理解现实世界
2. 理解计算机世界
3. 连接现实世界和计算机世界：产生互动效应
2. 涉及诸多参与方
3. 处理内容多样：各种类型的功能性和非功能性需求。
4. 处理活动互相交织：四个需求开发活动相互交织
5. 处理结果要求苛刻：包含错误、噪声、不切实际的需求都会导致灾难后果。
3. **问题域、目标、任务、交互**的相互转化(广义的设计)是**创造性的活动**
  1. 每个案例都有其独特性，不可复用，接近于艺术
  2. 需要对问题所在的领域有着深刻的认识
  3. **需要掌握一套设计思维与辅助工具，并多多练习**
4. 编程与设计方面的能力不能直接用于需求分析
  1. 设计和编程都有构建高质量(健壮性、可维护性、适应性等等)软件的共同目标，而且使用相同的概念和组织机制保证了从设计到编程的平滑过渡，所以，结构化与OO思维在设计领域也取得了成功
  2. 但是需求分析除了拥有构建高质量软件的目标之外，还有一个更加重要的目标是理解现实("出圈")中的非技术性和社会性因素
5. **文档撰写、功能验证、基线管理**需要丰富的开发与管理经验：资(nian)深(mai)程序员与新(nian)手(qing)程序员最大的区别

## 4. 需求工程师

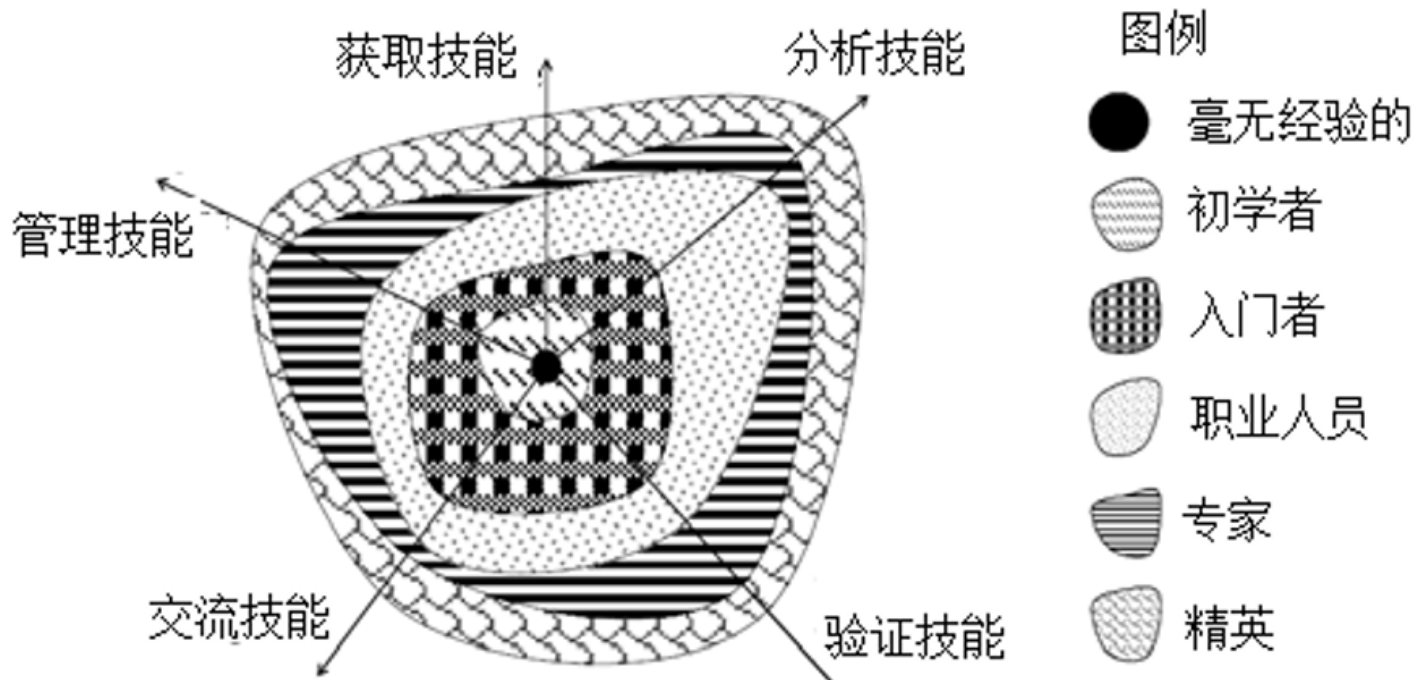
### 4.1. 需求工程师是涉众和开发者之间的桥梁

1. 需求工程师一切工作的核心就是扮演好桥梁作用
2. 面对涉众时，需求工程师就是后续软件开发者的代理，负责设计软件方案以满足涉众的各项需求。
3. 面对后续软件开发时，需求工程师就是涉众的代理，准确地各项需求告知开发者，替涉众跟踪和监控软件开发过程，保护涉众的代理。
4. 需求工程师更应该扮演好涉众代理的角色。

### 4.2. 需求工程师需要具备的技能

1. 精确的表达能力，尤其是文档化能力。
2. 非常好的沟通能力。
3. 抽象建模与分析的能力。

## 4.3. 需求工程师要重视"软技能"



### 4.3.1. 交流能力

1. 包括表述、写作、面谈、团队工作和狭义交流能力等。
2. 狭义交流能力包含
  1. 交谈和提问的技巧
  2. 倾听的技巧

### 4.3.2. 观察技能

1. 观察用户的工作环境和工作过程。
2. 引导用户谈话的过程

### 4.3.3. 抽象分析与问题解决技能

1. 抽象能力：把握问题的重点、核心和本质。
2. 整合全局能力：得到完整和一致的系统需求
3. 系统化思想

### 4.3.4. 写作技能

### 4.3.5. 关系协调和团队工作技能

## 4.4. 需求工程师需要创新

1. 软件系统并不仅仅是模拟现实，还要让现实变得更好。
2. 出色的需求工程师往往还会给出具有飞跃意义的创新：发现潜在需求——涉众自己没有认识到但是事实上非常需要的东西。

## 4.5. 需求开发是团队行为