

# 软件工程学科导论

## Introduction to Software Engineering

### 第4课 软件需求工程概要

王念滨

软件学院 大数据与智能计算课题组

2020年3月

## 二、课程安排

学习内容

软件工程学科概述

软件开发与软件开发管理

软件项目管理概要

软件需求工程概要

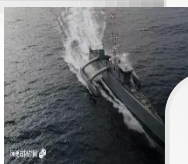
软件系统架构概要

软件设计原则与设计模式概要

软件测试与维护要点

软件工程教育与职业发展





## 第4课 软件需求工程概要





01

软件需求的意义和任务

02

需求的层级要点

03

需求开发纪要

04

需求开发最佳实践

05

需求管理纪要

06

需求管理最佳实践





**目录页**  
CONTENTS PAGE

**01**

软件需求的意义和任务

**02**

需求的层级要点

**03**

需求开发纪要

**04**

需求开发最佳实践

**05**

需求管理纪要

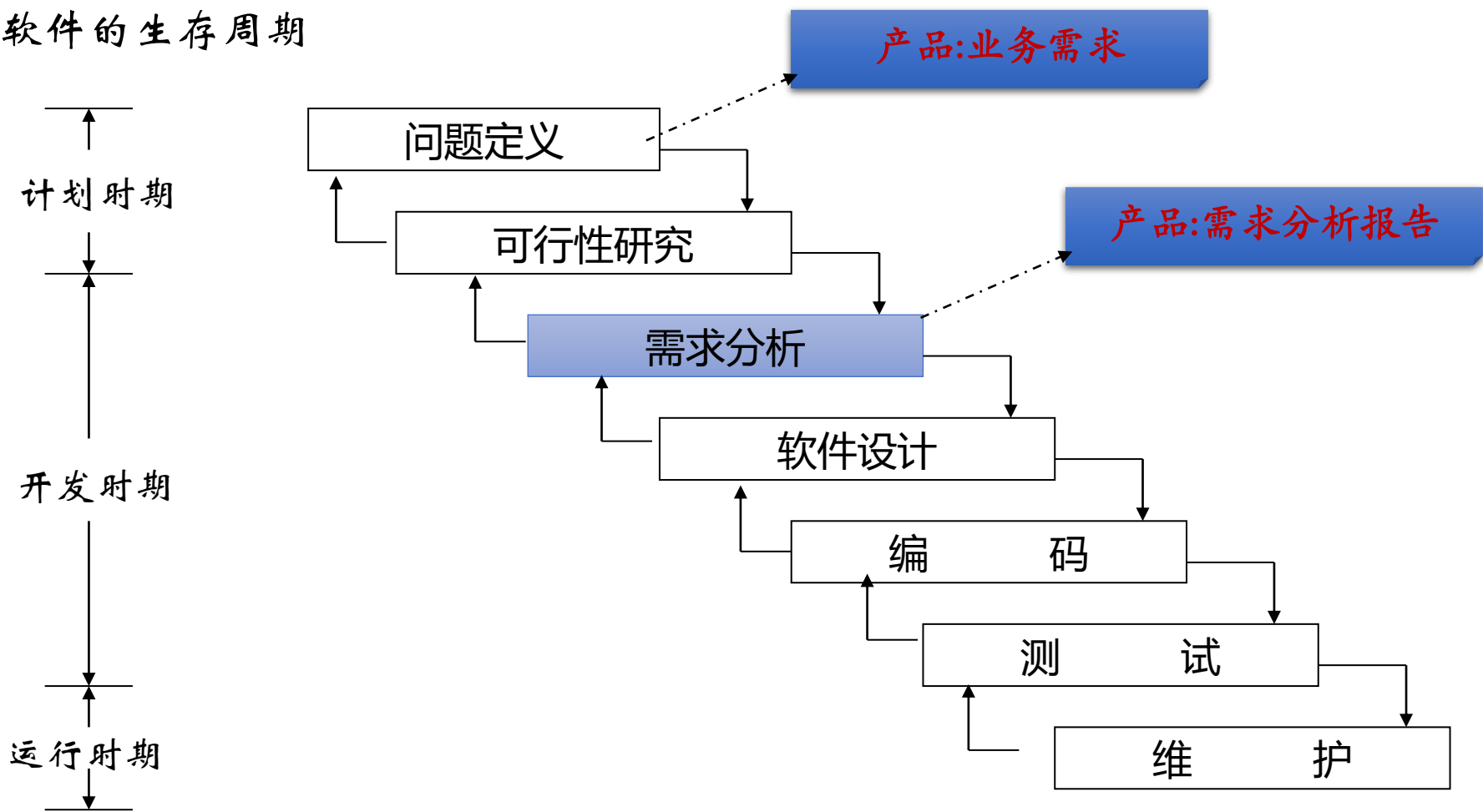
**06**

需求管理最佳实践



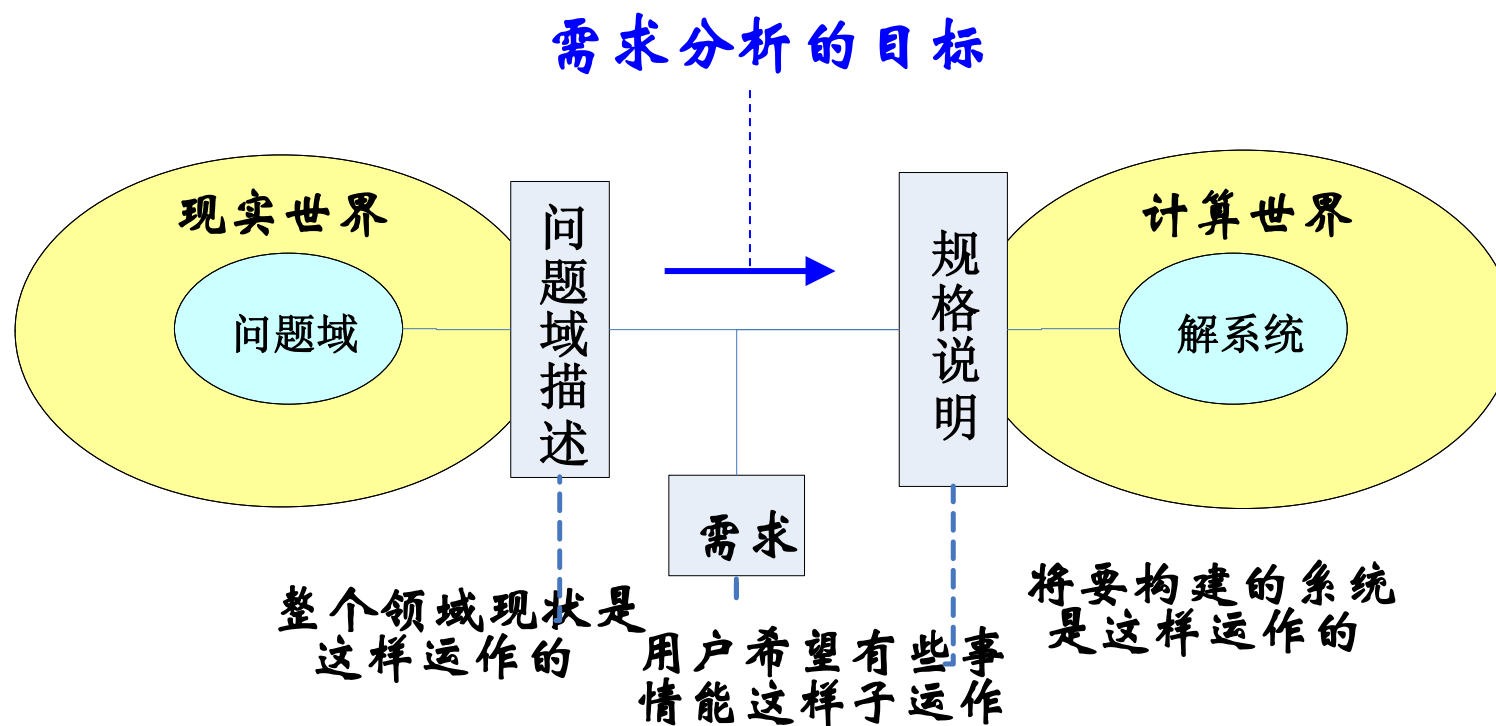
# 01 软件需求的任务与意义

## 软件需求在软件生命周期中的位置与产品



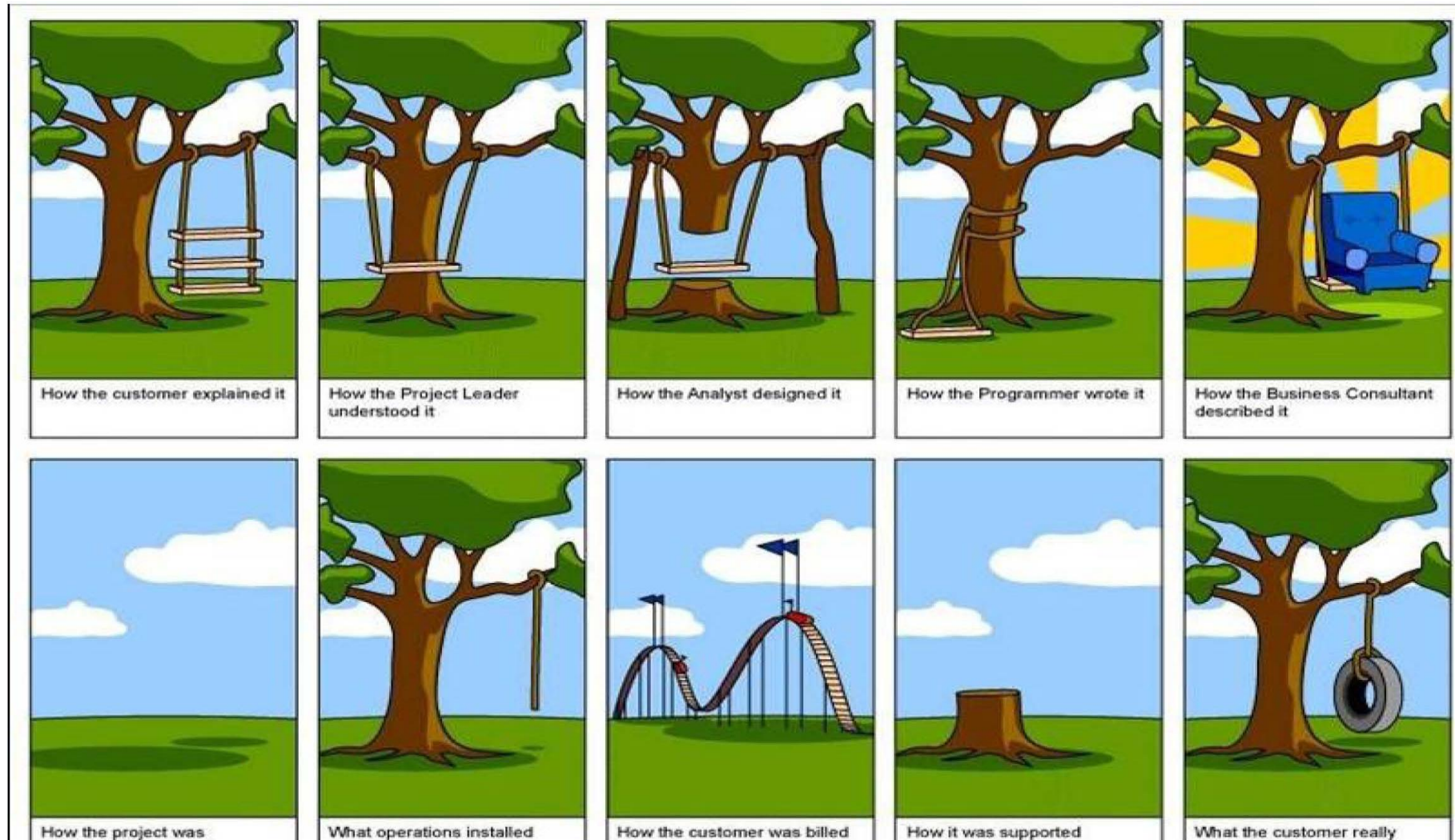
## 01 软件需求的任务与意义

### 软件需求及分析的目标



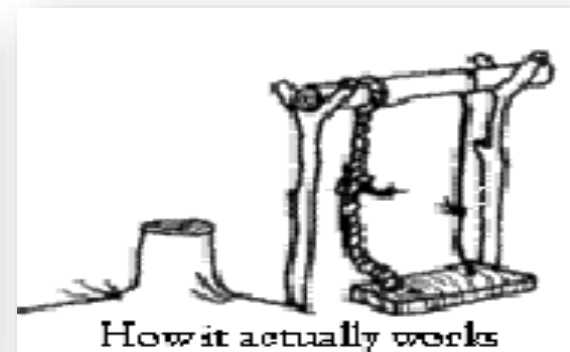
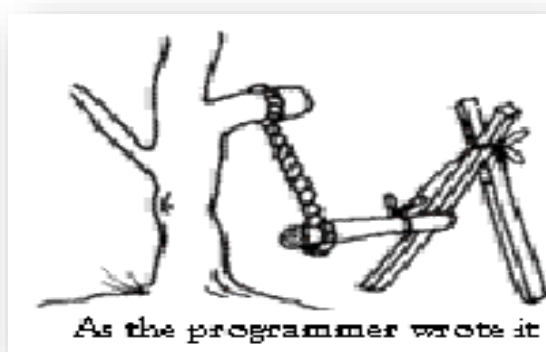
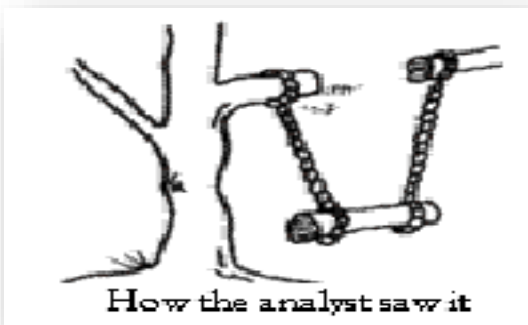
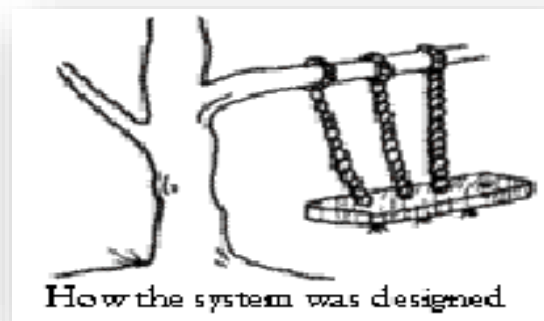
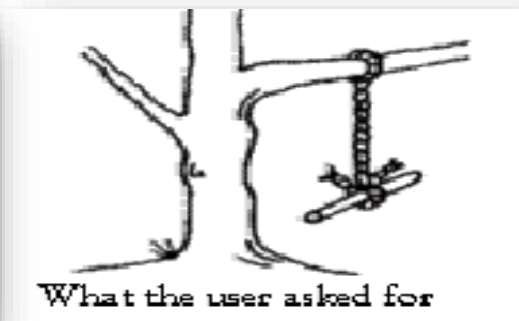
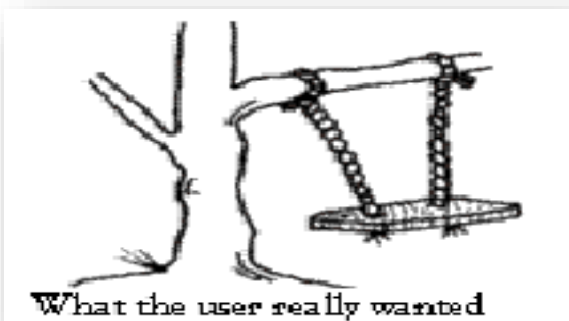
# 01 软件需求的任务与意义

现实中有关需求的最大问题-错误理解用户需求



## 01 软件需求的任务与意义

现实中有关需求的最大问题-错误理解用户需求



## 01 软件需求的任务与意义

### 需求错误的高昂代价



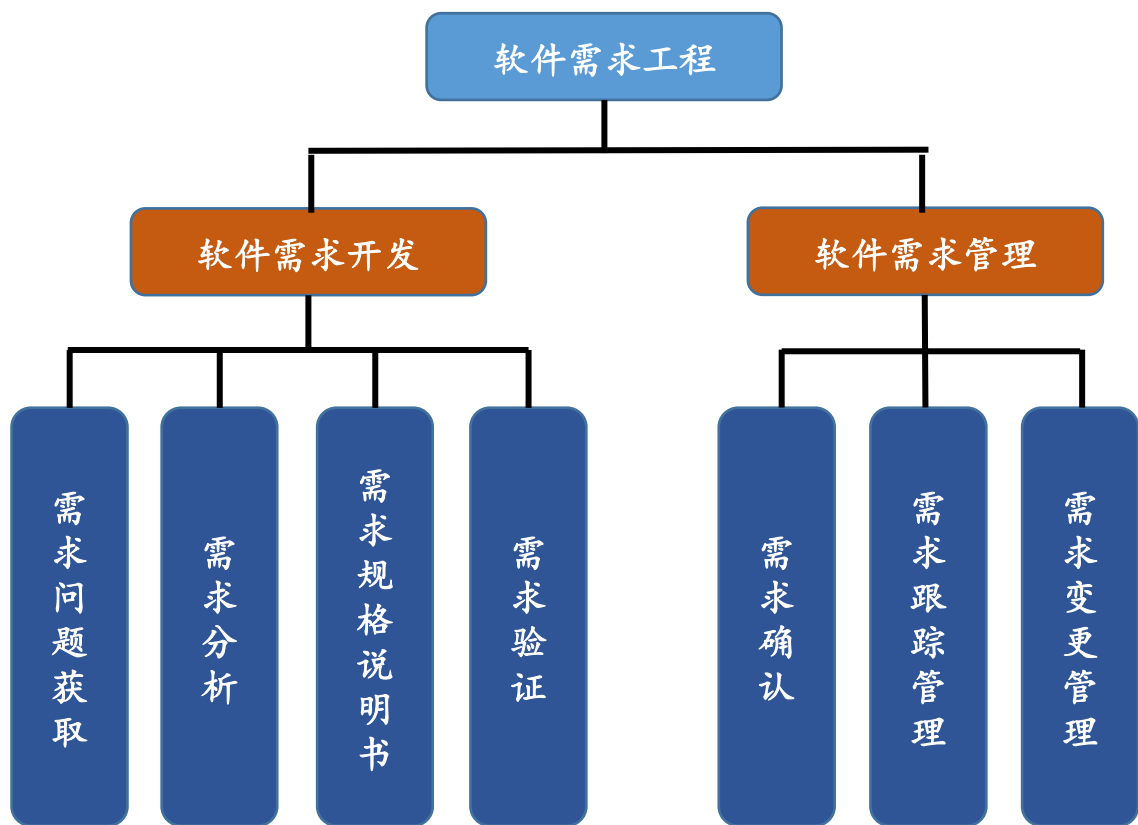
一般仅将需求称为工程，主要原因在于（如左图）

如果在需求阶段只需要花费1个时间单位就能够改正的错误，如果推迟到设计阶段改正就需要5个时间单位，如果推迟到测试阶段就可能达到20-50个时间单位，如果推迟到运行与维护阶段就可能需要花费200个时间单位



# 01 软件需求的任务与意义

## 软件需求要做什么？



需求是用户在特定场景下为了得到某种服务及功能而提出的要求，它是产品最终要达到的目的。

可以是用户使用过程中的不满，也可以是对一堆数据分析和市场调查后得出的结论。

需求分析在产品中占有重要的地位，决定着产品做出后被用户接纳的程度。开展需求分析的主要方法是调研和数据。

# 01 软件需求的任务与意义

## 软件需求的重要意义-对大型软件的感性认识（1）

属性 \ 软件名称	Postgres95	XXX省地方税务局管理信息系统
软件类型	系统软件、数据库系统软件	应用软件，管理信息系统
代码量	450, 000	500, 000
编程语言	C	C++, PB, SQL, LOTUS, JSP
支撑环境	UNIX	WINDOWS
开发周期	约5年	1.5年
参加人员	POSTGRES 项目是由防务高级研究项目局（DARPA），陆军研究办公室（ARO），国家科学基金（NSF），以及 ESL, Inc 共同赞助的，加州大学伯克利分校计算机系开发	项目总负责1人 项目经理 3人；系统架构师 2人 系统分析员3人；SYBASE DBA 2人 高级程序员8人；程序员 20-25人 用户方参与人员10-15人
文档		文档总量 约120万字 项目论证报告 5万字 需求分析 25万字 概要设计 10万字 详细设计 40万字 数据库设计 10万字 测试方案 5万字 系统培训教材15万字 系统使用手册 10万字

# 01 软件需求的任务与意义

## 软件需求的重要意义-对大型软件的感性认识（2）

Oracle代码量大约是postgres的10倍, 450万行源代码  
Windows大约是Oracle的5倍, 2250万行源代码



	Exchange2000	Windows2000
项目 经理	25人	约250人
开发 人员	140人	约1700人
测试 人员	350人	约3200人

## 01 软件需求的任务与意义

### 软件需求的重要意义-软件危机(1)

#### 软件危机的主要表现

- 对软件开发成本和进度的估计常常不准确。开发成本超出预算，实际进度比预定计划一再拖延的现象并不罕见。
- 用户对“已完成”系统不满意的现象经常发生。
- 软件产品的质量往往靠不住。Bug一大堆，Patch一个接一个。
- 软件的可维护程度非常之低。
- 软件通常没有适当的文档资料。
- 软件的成本不断提高。
- 软件开发生产率的提高赶不上硬件的发展和人们需求的增长。

# 01 软件需求的任务与意义

## 软件需求的重要意义-软件危机(2)

### 软件生产和成本的原因

- 软件维护费用急剧上升，直接威胁到计算机应用的扩大
- 软件生产技术进步缓慢
- 软件系统的复杂性不断增长，复杂性成为系统设计和开发最大的障碍
- 软件开发生产率的提高赶不上硬件的发展和人们需求的增长。

概括来说，软件危机包含两方面问题：

如何开发软件，以满足不断增长，日趋复杂的需求；  
如何维护数量不断膨胀的软件产品。

## 01 软件需求的任务与意义

### 软件需求的重要意义-软件危机的原因

#### 软件危机的原因

##### 客观

软件本身的特点

- 逻辑部件规模庞大

系统  
网络  
数据库  
应用

##### 主观

不正确的开发方法

- 忽视需求分析
- 错误地认为软件开发=程序编写
- 轻视软件维护

## 01 软件需求的任务与意义

### 软件需求的任务-软件需求的基本定义

#### 需求的定义

IEEE(美国电气电子工程师学会)软件工程标准词汇表(1997年)将需求定义为:

- (1) 用户解决问题或达到目标所需的条件或能力。
- (2) 系统或系统部件要满足合同、标准、规范或其它正式规定文档所需具有的条件或能力。
- (3) 一种反映上面(1)或(2)所描述的条件或能力的文档说明。

需求就是以一种**清晰、简洁、一致且无二义性**的方式对一个待开发系统中各个有意义陈述方面的一个集合。

# 01 软件需求的任务与意义

## 软件需求的任务-软件需求的基本定义

### 需求定义

该阶段的基本任务是回答“系统必须做什么？”；

提交详细的需求分析文档；

该阶段是系统开发的基础，质量和成败的关键；

需要审查和验证。

# 01 软件需求的任务与意义

## 软件需求分析的任务

### 需求分析的任务

需求分析的任务就是解决“做什么”（What to do, do what）的问题.就是要全面理解用户的各项要求，并正确地表达所接受的用户需求。需求分析之所以重要，因为它具有策略性、方向性和决策性。

需求阶段产生的问题在整个开发工程中将具有指数级的复杂性。

需求不仅是开发商的问题，更重要的是它是用户的问题。用户要表达自己的需求，开发者要理解用户的需求。

## 01 软件需求的任务与意义

### 软件需求是决定软件开发是否成功的关键因素

- 1 需求分析可以帮助开发人员真正理解业务问题
- 2 需求分析是估算成本和进度的基础
- 3 需求分析可以避免建造错误系统，减少不必要的浪费
- 4 软件规格说明有助于开发人员与客户在“系统应该做什么”问题上达成正式契约
- 5 需求分析形成软件开发的基线，有助于管理软件的演化和变更
- 6 软件需求是软件质量的基础，为系统验收测试提供标准

## 01 软件需求的任务与意义

### 软件需求的重要意义-需求是关键因素

- 需求分析是估算成本和进度的基础
  - 需求分析构建了系统的业务模型，根据业务模型分析可以对项目成本做大略的估算。
  - 一般在需求分析完成后，项目管理人员都需要也可能对开发成本做一个大略的估计。
  - 具体方法是按照业务模块，根据投入人力资源情况，满足用户时间节点需求，采用“人月”度量大致的项目开发成本。

# 01 软件需求的任务与意义

## 软件工程领域著名书籍 人月神话



Frederick P. Brooks （弗雷德里克 布鲁克斯）

图灵奖获得者，他“对计算机体系结构、操作系统和软件工程作出了里程碑式的贡献”。

IBM System/360系统之父。担任360系统项目经理。

第16章 没有银弹-软件工程中的根本和次要问题

第17章 再论“没有银弹”



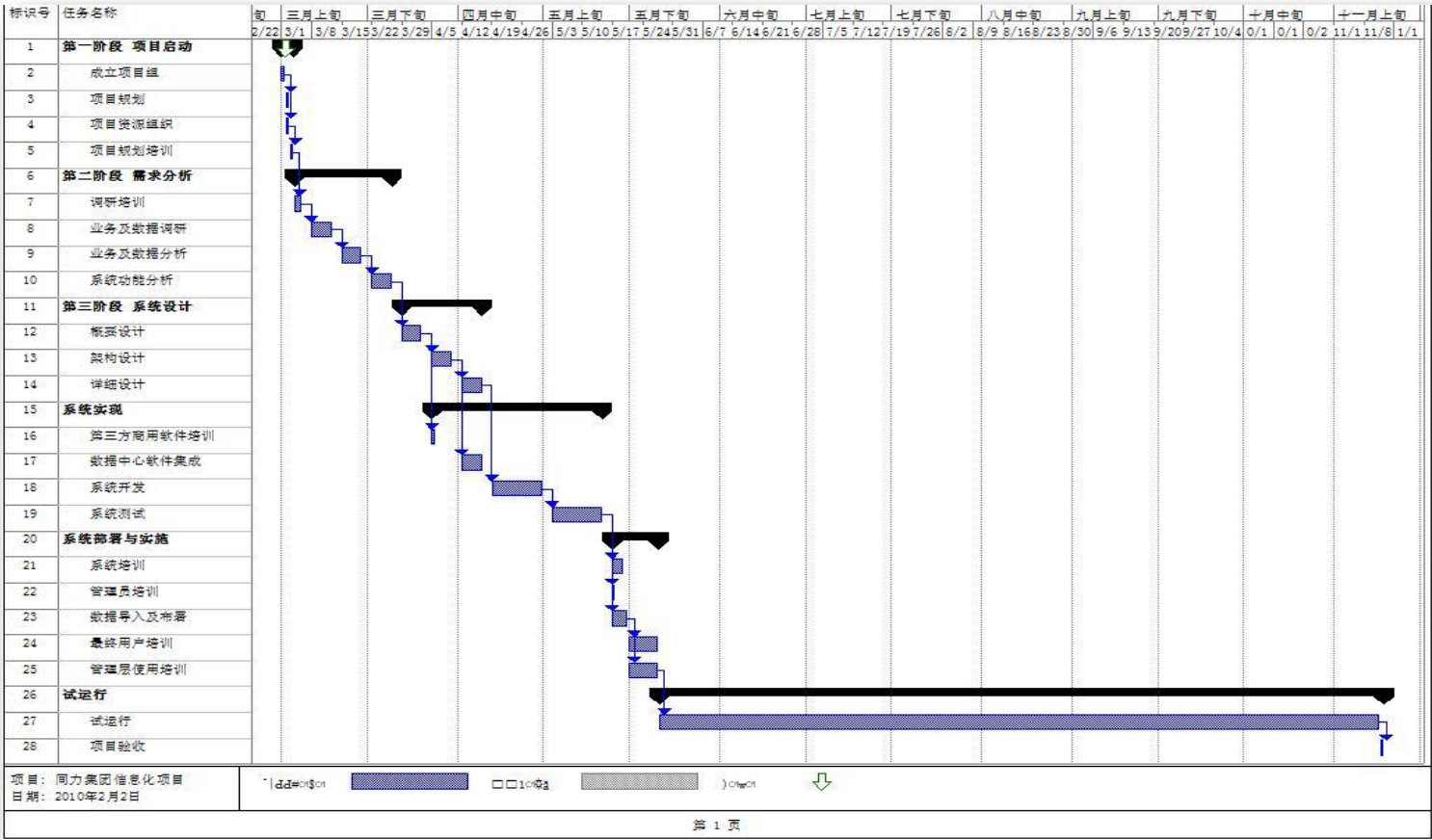
银色子弹（英文：Silver Bullet），或者称“银弹”“银质子弹”，指由纯银质或镀银的子弹。在欧洲民间传说及19世纪以来哥特小说风潮的影响下，银色子弹往往被描绘成具有驱魔功效的武器，是针对狼人等超自然怪物的特效武器。后来也被比喻为具有极端有效性的解决方法，作为杀手锏、最强杀招、王牌等的代称。

# 01 软件需求的任务与意义

软件需求的重要意义-需求是关键因素

需求分析是估算成本和进度的基础

进度安排示例——甘特图



# 01 软件需求的任务与意义

## 需求错误的高昂代价

需求错误占全部缺陷的三分之一。说明需求错误是系统软件开发中最重要的错误。

假如在编码阶段发现需求错误，则需要根据修改的需求重新进行设计；

假如是在测试阶段发现需求错误，则需要根据修改的需求重新进行设计并重新编码；

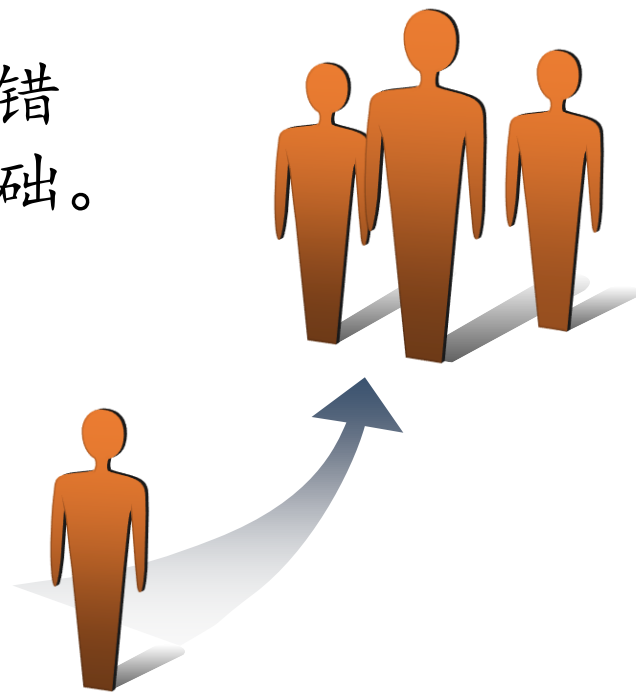
假如是在维护阶段发现的需求错误，就需要根据修改的需求重新进行设计，重新编码，并再次进行测试。

**结论：一定要尽量消除或者减少需求阶段产生错误的可能。**

## 01 软件需求的任务与意义

软件需求的任务：采用正确的方法和工具，准确、完整地获取用户的需要。

软件需求的意义：尽量消除或减少需求阶段产生错误的可能性，是保证软件项目按时保质完成的基础。





**目录页**  
CONTENTS PAGE

01

软件需求的意义和任务

02

需求的层级要点

03

需求开发纪要

04

需求开发最佳实践

05

需求管理纪要

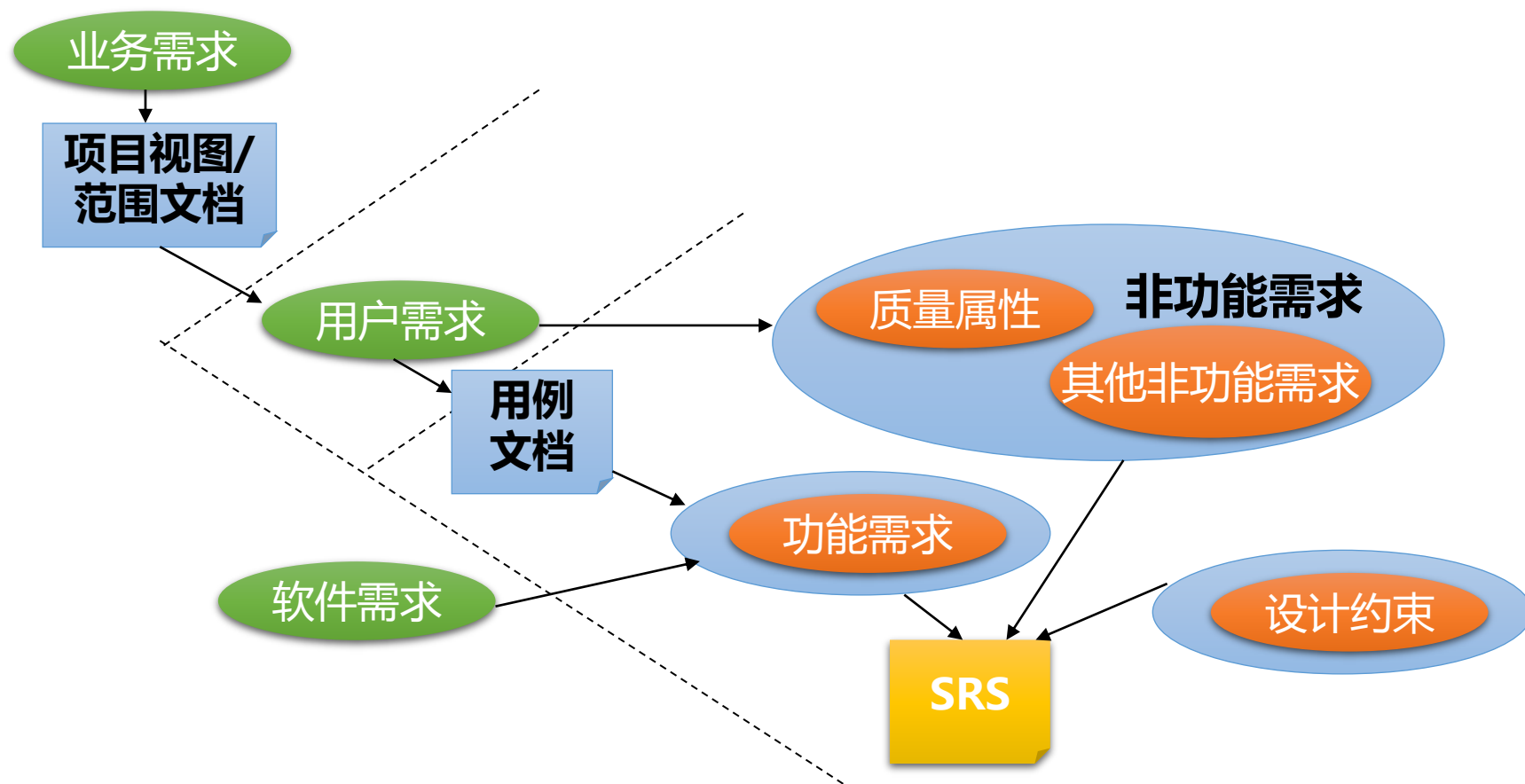
06

需求管理最佳实践



## 02 软件需求的层次要点

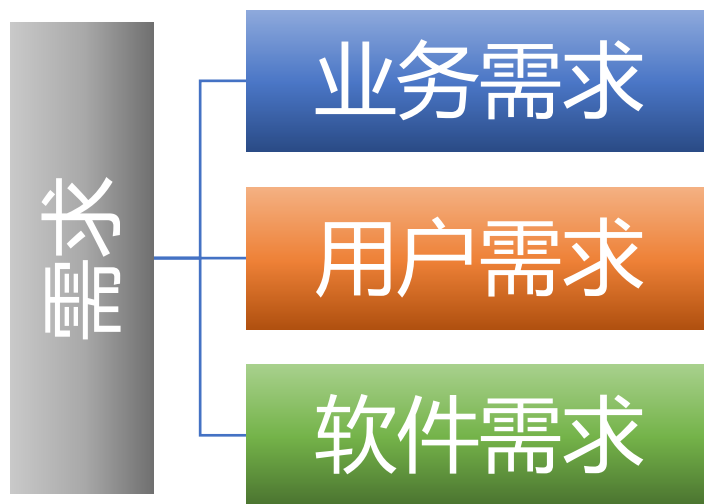
### 需求的三个层次及关系



## 02 软件需求的层次要点

### 需求的三个层次

在实际工作中，编写需求规格说明书时，一般会考虑以下几个名词：业务需求、用户需求、软件需求。实际上这几个名词正好反映了需求的三个层次



## 02 软件需求的层次要点

### 需求的三个层次-业务需求

- 需求的三个层次及关系
  - 业务需求
    - 业务需求反映企业/组织对软件系统的高层次目标需求，也就是说软件需求的建设目标。通常这一目标体现在两个方面。
    - 问题：解决企业/组织运作过程中遇到的问题，例如销售下降、物资供应问题、用户投诉、客户流失率居高不下等。
    - 机会：抓住外部环境变化（业务、技术）所带来的机会，以便为企业带来新的发展，例如电子商务，网上银行、协同工作等。

## 02 软件需求的层次要点

### 需求的三个层次-业务需求

- 需求的三个层次及关系
  - 业务需求
    - 业务需求理解
      - 系统建立的战略出发点，表现为高层次的目标（Objective），它描述了组织为什么要开发系统
      - 为了满足用户的业务需求，需求工程师需要描述系统高层次的解决方案，定义系统应该具备的特性（Feature）
      - 参与各方必须要对高层次的解决方案达成一致，以建立一个共同的前景（Vision）
      - 特性说明了系统为用户提供的各项功能，它限定了系统的范围（Scope）

## 02 软件需求的层次要点

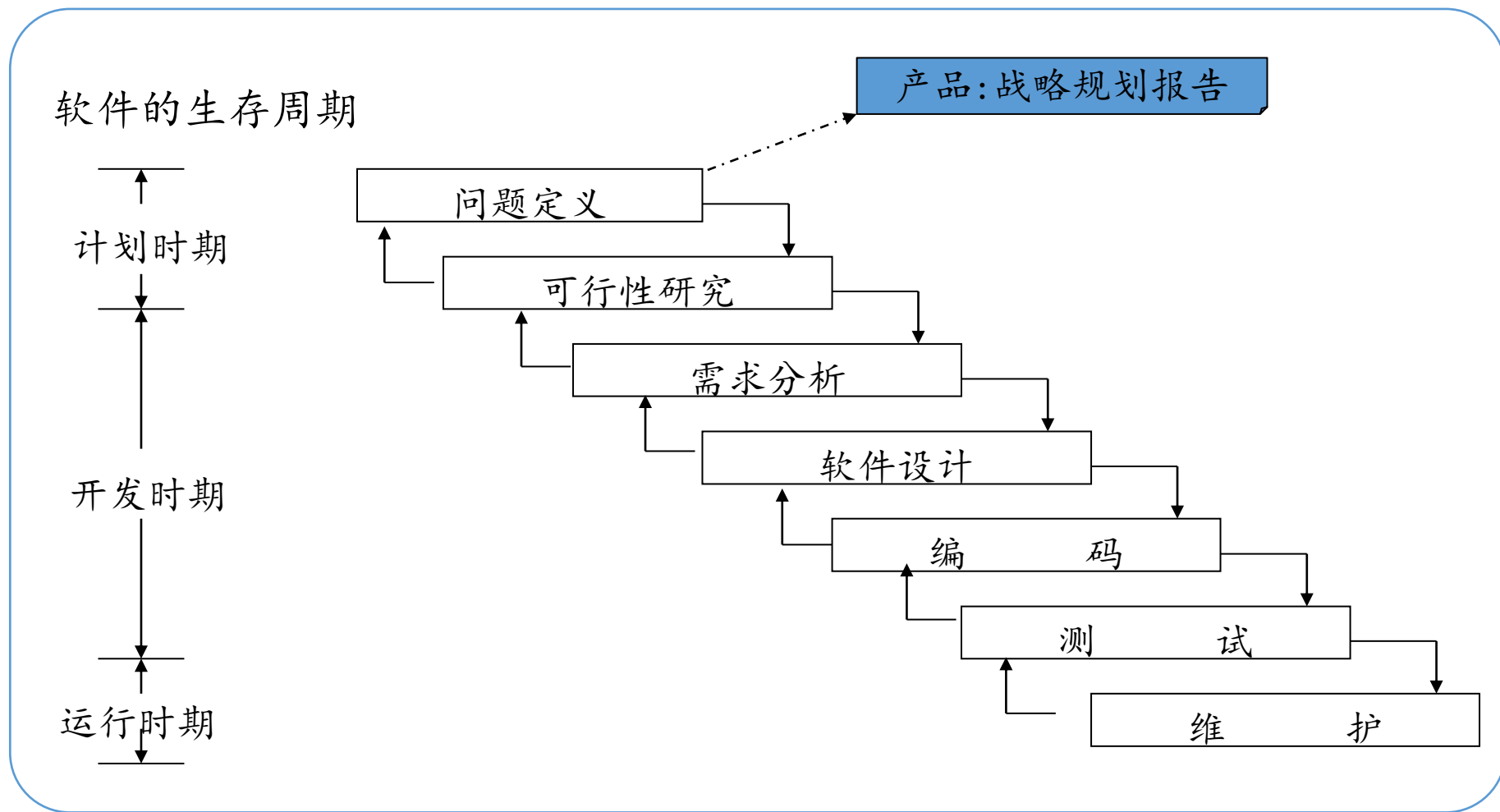
### 需求的三个层次-业务需求

- 需求的三个层次及关系
  - 业务需求
    - 业务需求作用意义
      - 业务需求的提出人通常是企业/组织的高层管理人员。业务需求是彻底从业务角度描述的，是指导软件开发的高层需求。明确地定义业务需求，将给整个团队指出努力的方向，这整个开发活动将有积极的意义
    - 业务需求建立的时间
      - 业务需求往往是在项目立项阶段整理完成的，通常体现在战略规划报告，或者**立项建议报告**中。

## 02 软件需求的层次要点

### 需求的三个层次-业务需求

#### 业务需求—高层次需求



# 02 软件需求的层次要点

## 需求的三个层次-业务需求

### 实例1-广东地税业务需求

第二章	广东地税业务战略方向	7
一、	广东地税组织结构	7
二、	广东地税业务战略	8
(一)	广东地税面临的挑战	9
(二)	广东地税业务战略要点	10
三、	广东地税业务战略和“大集中”工程的关系	12
(一)	广东地税业务战略对“大集中”工程的要求	12
(二)	“大集中”工程的目标以及业务需求	12

## 02 软件需求的层次要点

### 需求的三个层次

#### 广东地税业务战略要点

- 1 依法治税
- 2 为纳税人服务
- 3 加强内部监控，促进
- 4 增强税收系统的决策支持能
- 5 降低税收成本，提高税收工作效率

在目前税收环境尚不完善，纳税人的纳税意识有待加强的实际情况下，有必要在依托现代信息系统的基础上加强对税源的监控，为真正的应收尽收、依法征税打下基础，也为稽查工作提供数据来源。值得指出的是，依法治税也意味着稽查工作在相当长一段时间内仍然是税务机关的工作重点。

## 02 软件需求的层次要点

### 需求的三个层次-业务需求

#### 广东地税业务战略要点

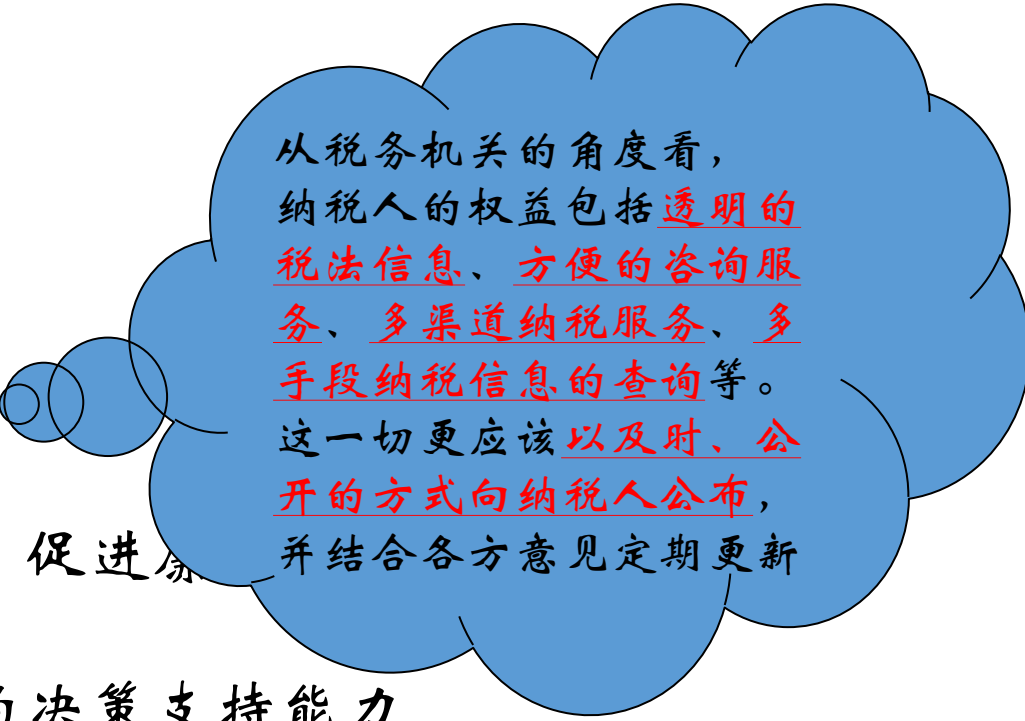
1 依法治税

2 为纳税人服务

3 加强内部监控，促进发展

4 增强税收系统的决策支持能力

5 降低税收成本，提高税收工作效率



从税务机关的角度看，纳税人的权益包括透明的税法信息、方便的咨询服务、多渠道纳税服务、多手段纳税信息的查询等。这一切更应该以及时、公开的方式向纳税人公布，并结合各方意见定期更新

## 02 软件需求的层次要点

### 需求的三个层次-业务需求

#### 广东地税业务战略要点

1 依法治税

2 为纳税人服务

3 加强内部监控，

4 增强税收系统的决策支持

5 降低税收成本，提高税收工作效率

除了加强内部监督与廉政教育之外，更重要的是从制度上和管理手段上尽可能地减少税收执法的随意性以及滥用权力现象的发生。采取征、管、查分离的方式。内部监控的加强往往涉及机构与岗位职能、职业操守准则、考核机制、业务流程、信息系统等多方面的因素，只有综合采用多种管理手段，才能收到好的效果。

## 02 软件需求的层次要点

### 需求的三个层次

#### 广东地税业务战略要点

- 1 依法治税
- 2 为纳税人服务
- 3 加强内部监控，促
- 4 增强税收系统
- 5 降低税收成本，提高税

完整、准确的税收信息对确定税收工作的重点、合理调配资源等决策工作有着非常重要的作用。在反映经济运行状况的各类数据中，税收数据是最为及时、也是相对全面和准确的信息资源，长期以来一直受到政府有关部门的高度重视。税务部门应努力提高税收数据的质量，并深入开展分析和研究工作，为政府宏观经济决策提供更有力的支持。目前全国税务系统在提供准确、及时的税收数据，支持决策方面的能力还很有限。应率先作出有益的积极尝试，让税收信息能够真正为管理和决策提供服务。

## 02 软件需求的层次要点

### 需求的三个层次-业务需求

#### 广东地税业务战略要点

1 依法治税

2 为纳税人服务

3 加强内部监管

4 增强税收系统的支持能力

5 降低税收成本，提高税收工作效率

纳税成本对政府税收政策的制定有重要影响，因此有必要考虑试行收集有关数据，建立相应系统来核算纳税成本、以便为决策支持提供依据，这同时也是为纳税人服务的另一种体现。针对税收工作的效率，仍有很大潜力可挖。如：通过信息技术手段而实现更智能化的稽查选案、更多的申报和缴纳渠道、更快速准确的税收统计分析等

## 02 软件需求的层次要点

### 需求的三个层次-用户需求

- 需求的三个层次及关系

- 用户需求

- 用户需求是指描述用户使用软件需要完成什么任务，怎么完成的需求。通常是在业务需求定义的基础上通过用户访谈、调查，对用户使用的场景进行整理，从而建立用户角度的需求。用户需求是需求捕获的结果。

## 02 软件需求的层次要点

### 需求的三个层次-用户需求

- 需求的三个层次及关系
  - 用户需求
    - 用户需求理解
      - 执行实际工作的用户对系统所能完成的具体任务的期望，描述了系统能够帮助用户做些什么
        - ✓ 直接用户
        - ✓ 间接用户
    - 对所有的用户需求，都应该有充分的问题域知识作为背景支持。

## 02 软件需求的层次要点

### 需求的三个层次-用户需求

- 需求的三个层次及关系
  - 用户需求
    - 用户需求理解
      - 用户需求是从用户角度描述的系统功能需求和非功能需求，通常只涉及系统的外部行为，而不涉及系统的内部特性。
    - 用户需求的描述
      - ✓原则：应该易于用户的理解。一般不采用技术性很强的语言，而是采用自然语言和直观图形相结合的方式  
进行描述。
      - ✓问题：自然语言表达容易含糊和不准确。

## 02 软件需求的层次要点

### 需求的三个层次-用户需求

- 需求的三个层次及关系

- 用户需求

- 用户需求理解

- 实例：图书馆用户需求

- ✓用户可以通过Internet 随时查询图书信息和个人借阅情况，并可以快捷地查找和浏览所需要的电子资料。

- 分析：上述需求描述包含了三个不同的需求

- ✓用户可以通过Internet 随时查询图书信息。

- ✓用户可以通过Internet 随时查询个人借阅情况。

- ✓用户可以通过Internet 快捷地查找和浏览所需要的电子资料。

- 问题：

- ✓“随时”和“快捷”是对系统功能的约束，十分模糊。

## 02 软件需求的层次要点

### 需求的三个层次-用户需求

- 需求的三个层次及关系

- 用户需求

- 用户需求的特点

- 1 零散：用户会提出不同角度、不同层次、不同粒度的需求，并且通常是以一句话的方式提出的。例如，在电信行业，对资费快要用完的用户，可以根据用户的要求通过短信方式将欠费和即将欠费的信息发送给相关用户。
      - 2 存在矛盾：由于用户往往处于企业/组织的不同层面，难免会出现盲人摸象的情况，从而导致需求的片面性，甚至在不同用户之间会持有不同的观点。

**需要对用户需求进行分析、整理，从而获得比较准确的需求说明**

## 02 软件需求的层次要点

### 需求的三个层次-软件需求

- 需求的三个层次及关系
  - 软件需求（系统需求）
    - 用户对软件系统行为的期望，一系列的行为联系在一起可以帮助用户完成任务，满足业务需求
    - 软件需求可以**直接映射为系统行为**，定义了系统中需要实现的功能，描述了开发人员需要实现什么
    - 将用户需求转化为软件需求的过程是一个复杂的过程
      - 首先需要分析问题领域及其特性，从中发现问题域和计算机系统的共享知识，建立系统的知识模型；
      - 然后将用户需求部署到系统模型当中，即定义系列的系统行为，让它们联合起来实现用户需求，每一个系统行为即为一个系统需求。
      - 该过程就是需求工程当中最为重要的需求分析活动，又称建模与分析活动。

## 02 软件需求的层次要点

### 需求的三个层次-软件需求

- 需求的三个层次及关系

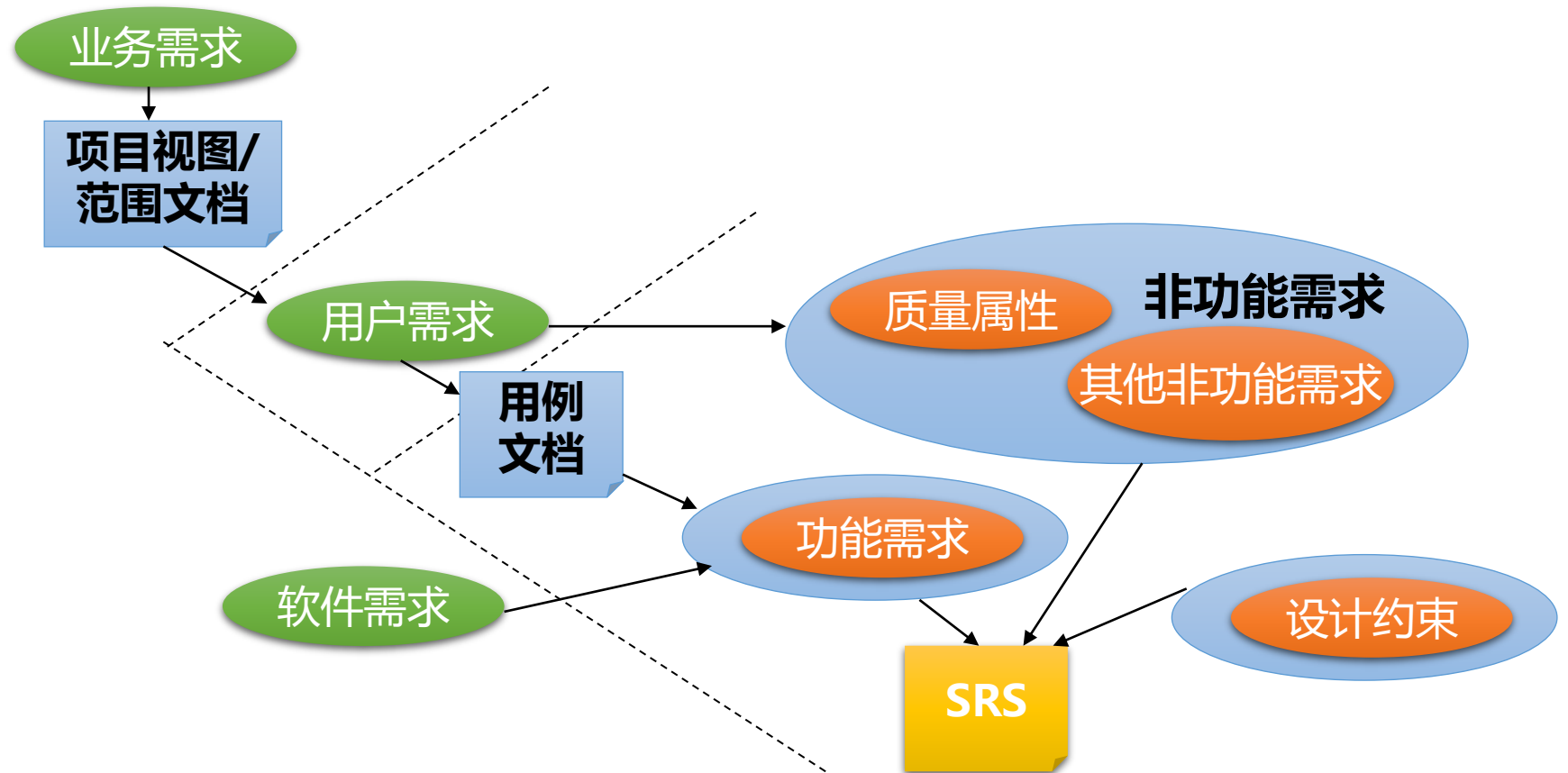
- 软件需求

- 如前所描述，业务需求具有高层次，比较抽象。用户需求具有零散、存在矛盾的情况。
- 需求分析人员需要按照业务需求的基本要求和指导原则，对用户需求进行分析、整理、提炼等工作，从而生成指导开发的、更精确的软件需求。

#### 需求文件

REQ001 系统应该有姓、中间名首字母和名等字段。  
REQ002 系统应该显示名字如果存储的个人资料中已有一个。  
REQ003 系统应该要求姓名是完整的。  
REQ004 系统应该有职位或头衔字段。  
REQ005 系统应该要求头衔是完整的。  
REQ006 系统应该显示职位或头衔如果存储的个人资料中已有一个。  
REQ007 系统应该有电子邮件地址字段。  
REQ008 系统应该有备用的电子邮件地址字段。  
REQ009 系统应该显示电子邮件地址如果存储的个人资料中已有一个。  
REQ010 系统应该显示备用电子邮件地址如果存储的个人资料中已有一个。  
REQ011 系统应该要求电子邮件地址是完整的。  
REQ012 系统应该要求备用的电子邮件地址是完整的。  
REQ013 系统应该具有白天电话号码的字段。  
REQ014 系统应该显示电话号码如果存储的个人资料已有一个。  
REQ015 系统应该要求电话号码是完整的。  
REQ016 系统应该在验证电话号码字段中所有字符是数字当用户退出该字段时。  
REQ017 系统应该显示错误消息如果在电话号码字段不是所有字符都是数字。  
REQ018 系统应该有传真号码的字段。  
REQ019 系统应该要求传真号码是完整的。  
REQ020 系统应该显示传真号码如果存储的个人资料已有一个。  
REQ021 系统应该验证在传真号码字段中的所有字符是数字当用户退出该字段时。  
REQ022 系统应该显示错误信息如果在传真号码字段里不是所有字符都是数字。  
REQ023 系统应该有街道地址的两个字段。  
REQ024 系统应该要求街道地址字段是完整的。  
REQ025 系统应该显示地址如果存储的个人资料已有一个。  
REQ026 系统应该有城市的字段。  
REQ027 系统应该要求城市字段是完整的。  
REQ028 系统应该显示城市如果存储的个人资料已有一个。  
REQ029 系统应该有状态的字段。  
REQ030 系统应该显示状态如果存储的个人资料已有一个。  
REQ031 系统应该要求状态字段是完整的。  
REQ032 系统应该有邮政编码的字段。  
REQ033 系统应该显示邮政编码如果存储的个人资料已有一个。  
REQ034 系统应该要求邮政编码字段是完整的。

## 02 软件需求的层次要点



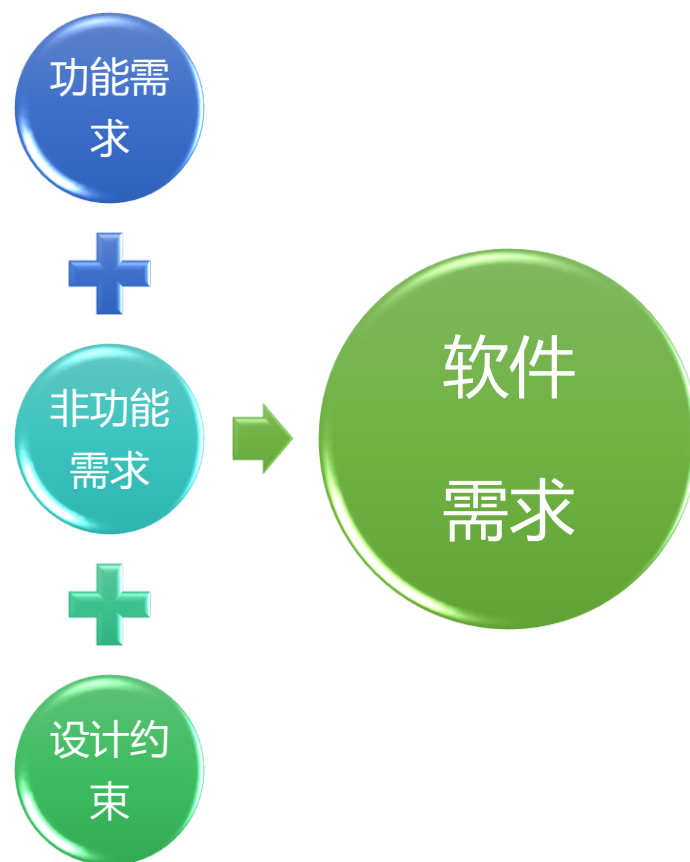
业务需求是软件定义的产物

用户需求是需求获取（捕获）的结果

软件需求是需求分析和建模的综合。

## 02 软件需求的层次要点

- 需求的三个层次及关系
  - 软件需求
    - 软件需求的三种类型



## 02 软件需求的层次要点

- 需求的三个层次及关系

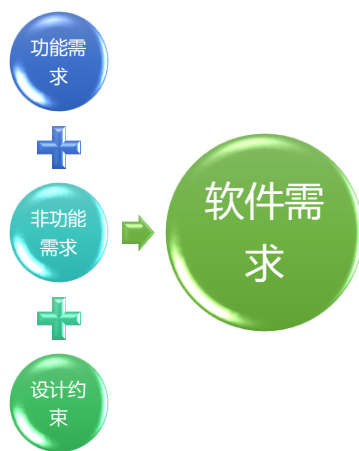
- 软件需求

- 功能需求

- 描述系统应该提供的功能或服务，通常涉及用户或外部系统与该系统之间的交互，一般不考虑系统的实现细节。

- 实例：图书馆系统功能需求

- 用户可从图书资料库中查询或选择其中的一个子集。
        - 系统可提供适当的浏览器供用户阅读电子文献。
        - 用户每次借阅图书应该对应一个唯一的标识号，它被记录到用户的帐户上。



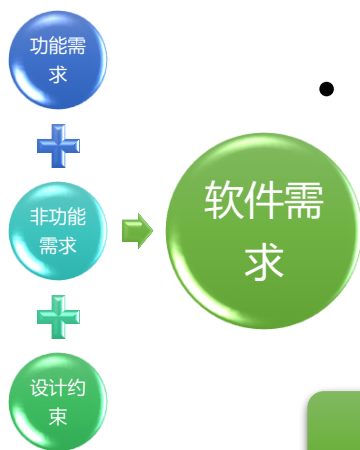
## 02 软件需求的层次要点

- 需求的三个层次及关系

- 软件需求

- 功能需求理解

- 对于功能需求而言，最为关键的是任何对其进行组织，否则一句话、一句话地描述就会显得十分零散，而且难以保证开发人员逐一满足这些需求。
      - 传统的需求开发方法中，通常会以软件系统----子系统----模块----子模块的层次结构来组织。其问题是采用该方法更多地是从程序的结构来梳理需求，问题是可能将用户的使用场景割裂开来。
      - 现代需求理论更强调需求分析人员从用户的角度，将系统理解为一个黑盒子，从使用角度来整理需求，不管是RUP或者是XP 方法都是如此。当然也不是说这样的方式就不存在问题，比如可能存在组织混乱的情况。



功能需求的要点在于如何组织

## 02 软件需求的层次要点

- 需求的三个层次及关系

- 软件需求

- 非功能需求

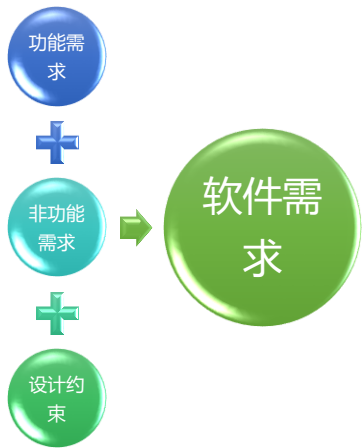
- 从各个角度对系统的约束和限制，反映了应用对软件系统质量和特性的额外要求，例如响应时间、数据精度、可靠性、开发过程的标准等。

- 实例：图书馆系统非功能需求

- 系统应在20 秒之内响应所有的请求。

- 系统每周7 天、每天24 小时都可以使用。

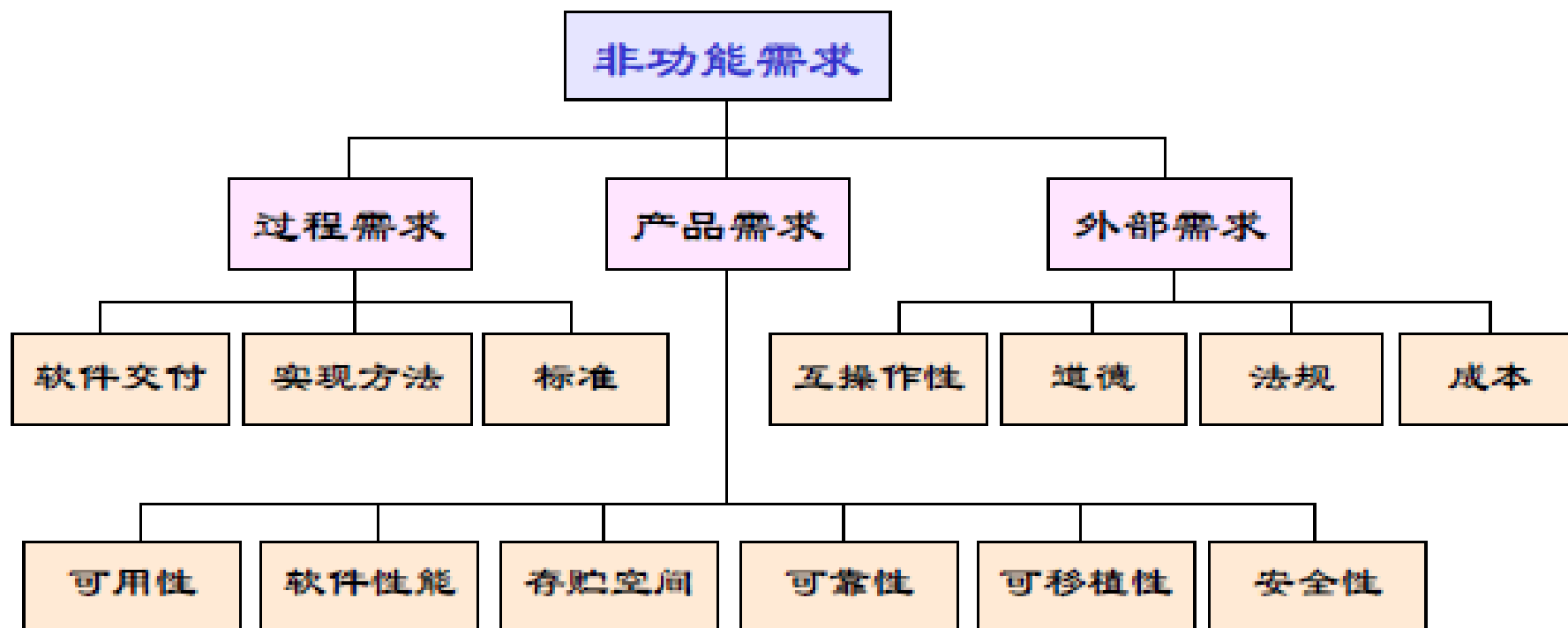
- 对于一个没有经验的用户而言，经过两个小时的培训就可以使用系统的所有功能。



## 02 软件需求的层次要点

- 需求的三个层次及关系
  - 软件需求
    - 非功能需求

非功能需求可以考虑的组织方式



## 02 软件需求的层次要点

- 需求的三个层次及关系

- 软件需求

- 非功能需求理解

非功能需求一定要注意保证信息的有效传递和注意其局部性

- 非功能需求常见的问题主要有：信息传递的无效性；忽略了非功能需求的局部性。
- 信息传递的无效性：许多需求规格说明书中，会通过一个名为设计原则的章节来说明非功能需求，列出诸如高性能、高可靠性、高可用性、高可扩展性等要求。实际应用中，多数开发人员根本就不去考虑这些问题。因为对这样的非功能描述没有判定标准，由此这类传递的信息是无效的。
- 忽略了非功能需求的局部性：通常可以看到的非功能需求，比如“系统响应时间小于5秒”的描述，但当用户查询的是年度统计数据时，这样的要求根本无法实现。由此开发人员仍然不会将这类问题加以仔细考虑，最终将这些要求变成摆设。

功能需求



非功能需求



设计约束



软件需求

## 02 软件需求的层次要点

- 需求的三个层次及关系
  - 软件需求
    - 非功能需求

### 性能需求

- 速度（Speed），系统的响应时间，例如PR2. 3. 3-1。
  - PR2. 3. 3-1：所有的用户查询都必须在10秒内完成。
- 容量（Capacity），系统所能存储的数据量，例如PR2. 3. 3-2。
  - PR2. 3. 3-2：系统应该能够存储至少10万条销售记录。
- 吞吐量（Throughput），系统在连续的时间内完成的事务数量，例如R2. 3. 3-3。
  - PR2. 3. 3-3：解释器每分钟应该至少解析5000条没有错误的语句。
- 负载（Load），系统可以承载的并发工作量，例如PR2. 3. 3-4。
  - PR2. 3. 3-4：系统应该允许200个用户同时进行正常的工作。
- 实时性（Time-Critical），严格的实时要求，例如PR2. 3. 3-5。
  - PR2. 3. 3-5：监测到病人异常后，监控器必须在0.5秒内发出警报。

## 02 软件需求的层次要点

- 需求的三个层次及关系
  - 软件需求
    - 非功能需求

### 质量属性

- ❖ 系统为了满足规定的及隐含的所有要求而需要具备的要素称为质量
- ❖ 质量属性是为了度量质量要素而选用的特征
- ❖ 质量模型就是能够为质量需求的描述和评价提供工作基础的特征集及特征之间的联系
- ❖ 质量属性的重要性
  - 对设计的影响很大
  - 对越复杂的系统越为重要
  - 真实的现实系统中，在决定系统的成功或失败的因素中，满足非功能属性往往比满足功能性需求更为重要。

## 02 软件需求的层次要点

- 需求的三个层次及关系
  - 软件需求
    - 非功能需求

### 质量属性的开发

- ❖ 用户并不能明确地提出他们对产品质量的期望
  - 并不了解软件系统的开发过程，也就无从判断哪些质量属性会在怎样的程度上给设计带来多大的影响，也无法将他们对软件系统的质量要求细化成一组组的可量化的质量属性
- ❖ 需求工程师
  - 质量属性大都是和功能需求联系在一起的，因此需要对照软件的质量属性检查每一项功能需求，尽力去判断质量属性存在的可能性
    - 形容词和副词通常意味着质量属性的存在
  - 对于一些不 and 任何功能需求相联系的全局性质量属性，需求工程师要在碰到特定的实例时意识到它们的存在

## 02 软件需求的层次要点

- 需求的三个层次及关系
  - 软件需求
    - 非功能需求
      - 一般在软件开发过程中可以将非功能需求划分为：

性能需求，质量属性，对外接口等

## 02 软件需求的层次要点

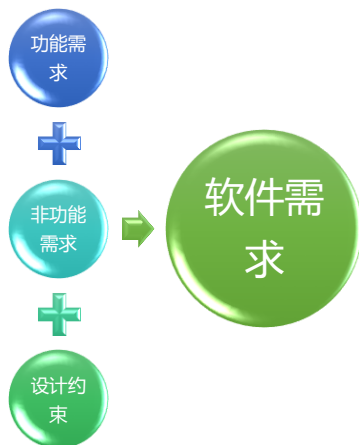
- 需求的三个层次及关系
  - 软件需求
    - 非功能需求

### 对外接口

- ❖ 解系统和其他系统之间的软硬件接口
  - 接口的用途
  - 接口的输入输出
  - 数据格式
  - 命令格式
  - 异常处理要求
- ❖ 用户界面
  - 利用专门的人机交互设计文档记录

## 02 软件需求的层次要点

- 需求的三个层次及关系
  - 软件需求
    - 设计约束
      - 一般包括非技术因素决定的技术选型问题，以及预期的软硬件环境，预期的使用环境等。
  - 设计约束非常重要，不要认为是可用可无的。



## 02 软件需求的层次要点

- 需求的三个层次及关系

- 软件需求

- 设计约束的理解

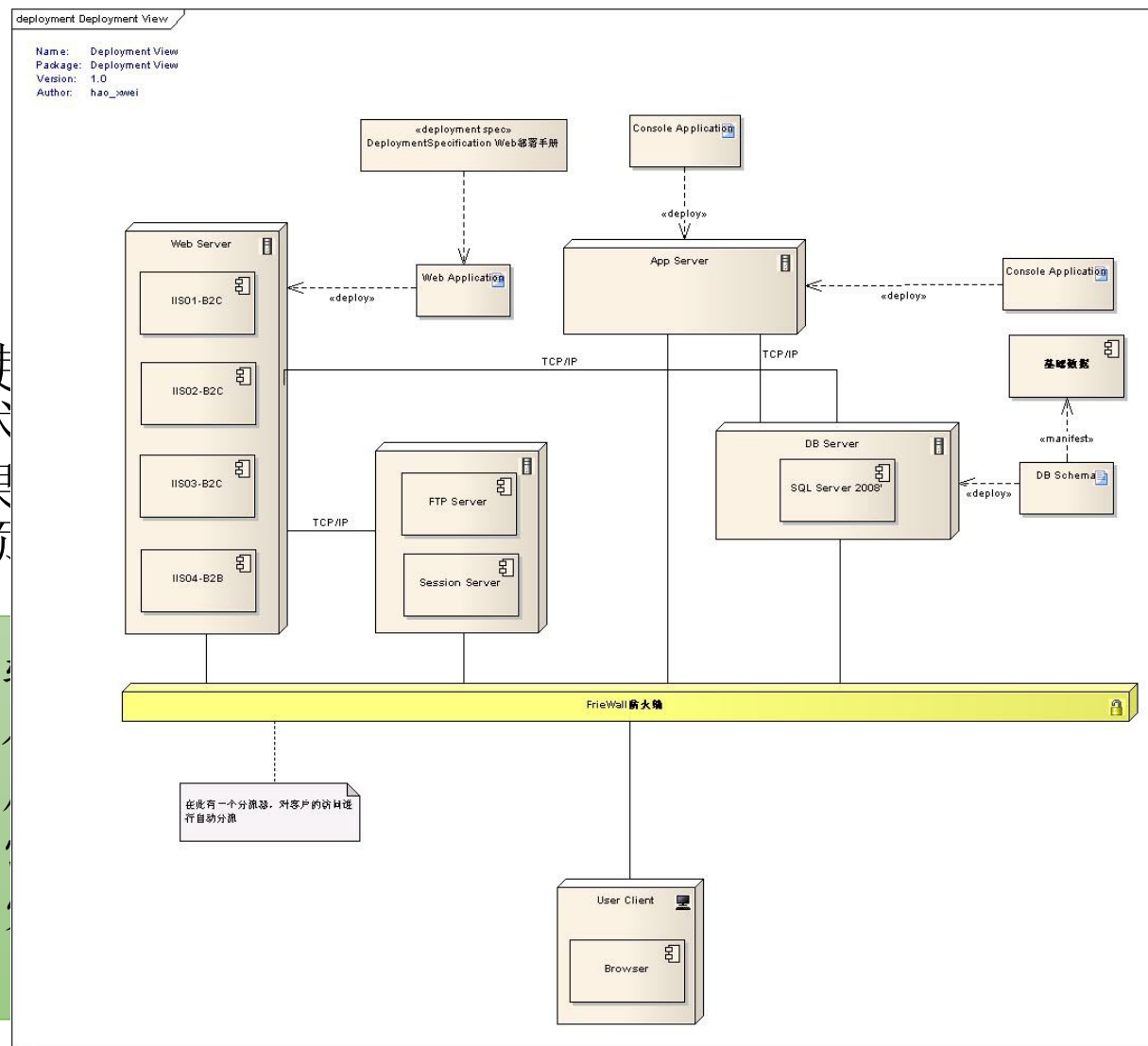
- 非技术因素决定的技术选型：对于软件开发而言，有些技术不是由技术团队决定的，而是会受到企业/组织实际情况的影响。
      - 例如：必须采用具有自主知识产权的数据库系统，系统开发必须使用J2EE技术等。

## 02 软件需求的层次要点

- 需求的三个层次及关系
  - 软件需求
  - 设计约束的理解

- 预期的软硬件环境和使  
定架构、选择实现技术  
硬件环境的影响，如果  
项目带来不必要的麻烦

例如：黑龙江省地方税务局征管系统县（区）应用良好。但在所级应用存单位地处边远，电脑配置低、网络速后不得不专门编写了所级软件。说明应该将这些预期的软硬件环境都描述部署图方式。





01

软件需求的意义和任务

02

需求的层级要点

03

需求开发纪要

04

需求开发最佳实践

05

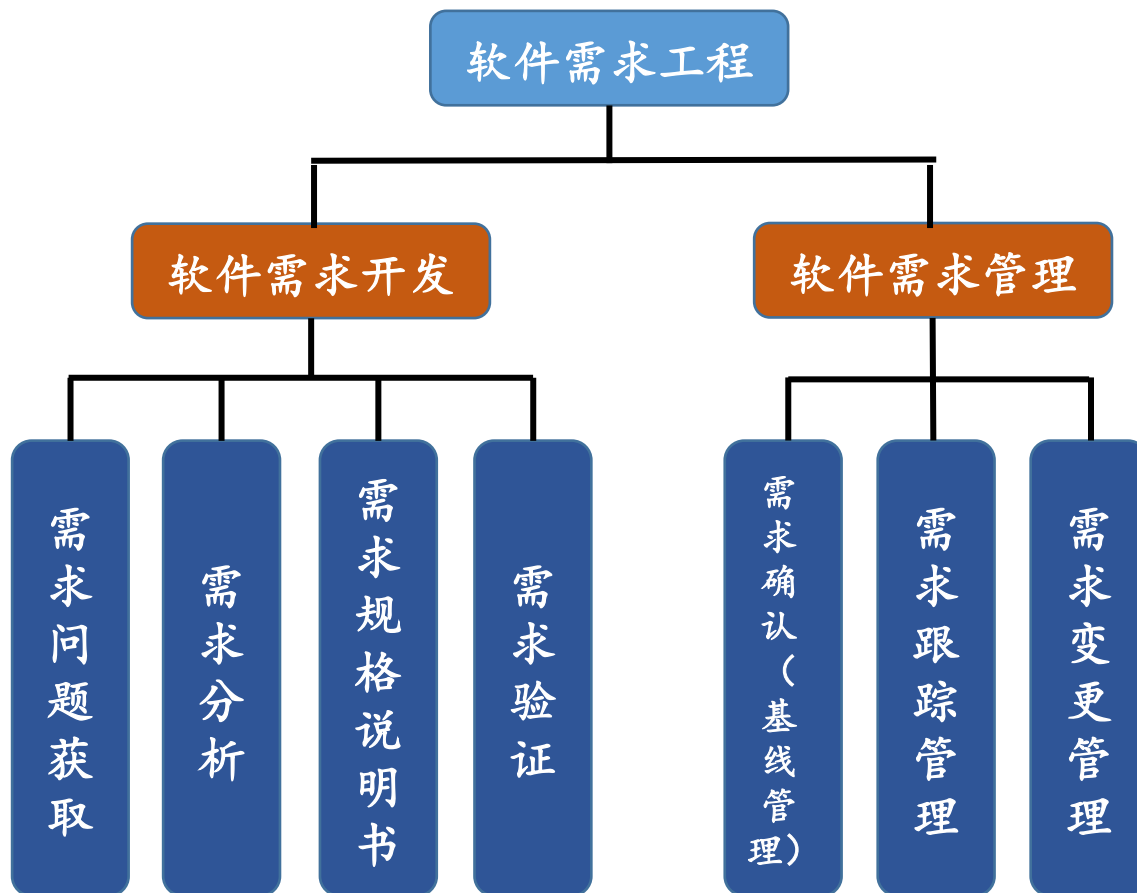
需求管理纪要

06

需求管理最佳实践

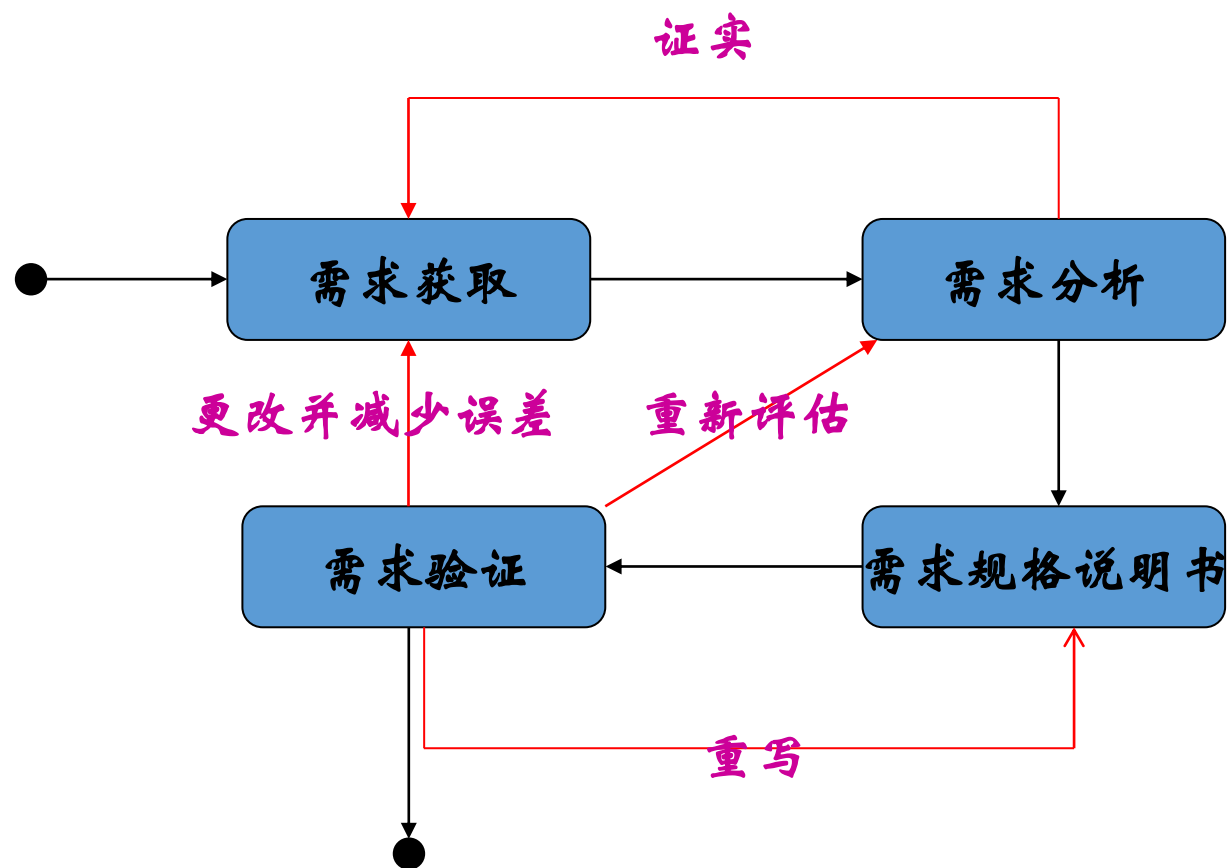


## 03 需求开发纪要



## 03 需求开发纪要

### 需求开发过程



## 03 需求开发纪要

- 良好的软件需求应该具备的特质
  - 完整性、真实性、优先级、技术早期介入

## 03 需求开发纪要

- 良好的软件需求应该具备的特质
  - 完整性
    - 需求的完整性简单的说就是需求没有遗漏。其表现为需求变更中新需求所占的比例为零。当然这只是理想情况，一般我们在软件开发工程中需求变更不超过10%就相当不错了。
  - 实现完整性的有效方法：
    - 用户才是验证完整性的合适人选；
    - 需求完整性存在不同的层面上。

## 03 需求开发纪要

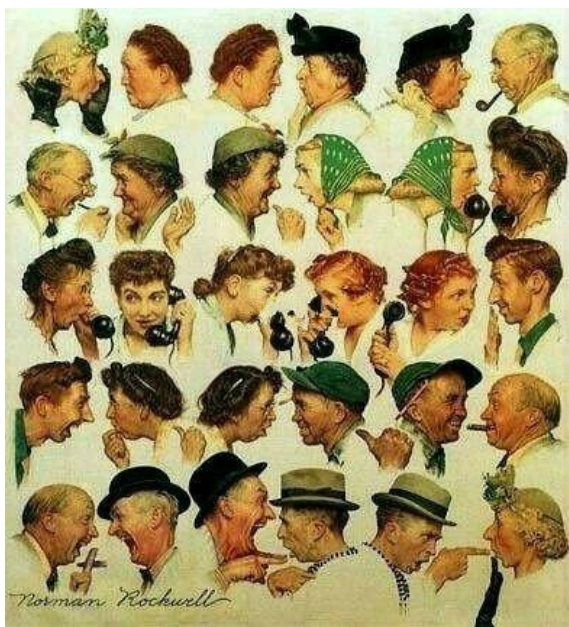
- 良好的软件需求应该具备的特质
  - 完整性的理解
    - 需求分析人员就像是在为一个人画像，如果将视角放得太近，想要将人画完整是不可能的。正确的做法是先后退到适当的距离，把整个轮廓画出来，然后走近一些，将不同的组织部分分解开，最后再走到跟前，将细节画出来。
    - 具体说就是在需求构建时先描述宏观部分（比如确定主题域的划分，并让高层进行验证。分析标识出来的主题域是否能够达到目标所需涉及的范围。然后针对主体域进行分析，找到其流程和实体，让中层对其进行验证；最后走向操作层，对细节进行描述并验证）

## 03 需求开发纪要

- 良好的软件需求应该具备的特质

- 真实性

- 需求的正确性和无歧义性是一组相关的要求。指的是确保在信息传递的过程中不失真。
  - 正确性：要使需求确保正确，就需要找到正确的人来验证。分层验证是可以存取的有效手段。
  - 无歧义性：不同背景的人在传递时加入了不同理解而导致歧义。由此需要建立一些消除歧义性的方法和手段。例如建立系统术语表等。



强化验证是保证真实性的关键

## 03 需求开发纪要

- 良好的软件需求应该具备的特质
  - 真实性



## 03 需求开发纪要

- 良好的软件需求应该具备的特质
  - 优先级
    - 优先级要分层次。层次的划分一般要从业务角度考虑

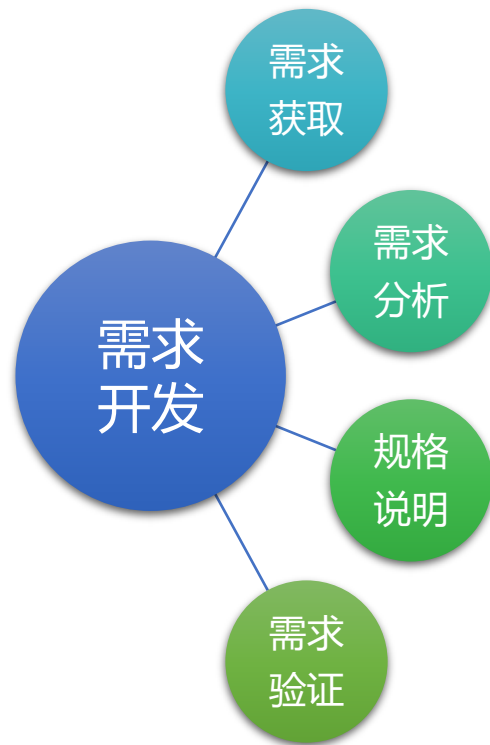
## 03 需求开发纪要

- 良好的软件需求应该具备的特质
  - 关于技术早期介入
    - 需求规格说明书的内容从用户来；需求规格说明书谁看？当然是技术团队。所以在需求规格说明书构建过程中，应该让技术团队了解相关需求，并分析存在什么问题，缺少什么信息等。是改进需求规格说明书的主要方法。

## 03 需求开发纪要

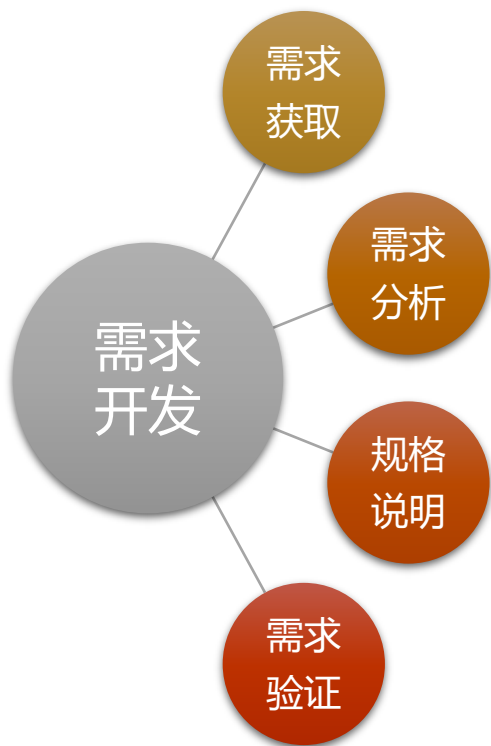
### ❖ 需求工程的基本活动

- 需求工程的活动主要包括需求开发和需求管理两个方面的工作。需求开发是实现的目标，需求管理是保证需求开发得以有序实现的基础和保障。
- 需求开发主要包括需求获取、分析、规格说明和验证。



## 03 需求开发纪要

### ❖ 需求工程的基本活动



- ❖ 需求获取的目的是从项目的战略规划开始建立最初的原始需求，需要研究系统将来的应用环境，确定系统的涉众，了解现有的情况，建立新系统的目标，获取为支持新系统目标而需要的业务过程细节和具体的用户需求；
- ❖ 需求分析的目的是保证需求的完整性和一致性。以需求获取阶段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，然后在抽象后的系统模型中进行分析，标识并修复其中存在的不一致问题，发现并弥补遗漏的需求。



01

软件需求的意义和任务

02

需求的层级要点

03

需求开发纪要

04

需求开发最佳实践

05

需求管理纪要

06

需求管理最佳实践



## 04 需求开发最佳实践

### 需求开发最佳实践

- 1 甲方说不清怎么办？
- 2 乙方不要以己度人？
- 3 获取和挖掘需求的要点？

## 04 需求开发最佳实践

### 需求开发最佳实践

#### 1 甲方说不清怎么办？

##### (1) 项目中唯一不变的是变化，学会拥抱变化

人类有一个共同的特点就是，总是用自己之前见过的东西来描述一个还没有加过的东西（自己希望的、想要的）。结果需求开发人员感觉已经了解了用户需求，其实未必。

因此，不断地沟通和反馈。每一次变更，离用户的需求越近。

##### (2) 不要试图仅从技术方面讨论问题

项目是一个业务过程，项目经理是业务层面的管理者，项目是业务导向的。

##### (3) 要给用户看到“样板房（DEMO）”

用以引导用户需求，使其思路控制在执行的范围内，还可以通过逆向反馈，推动客户的需求确认，尽可能避免理解上的偏差。

## 04 需求开发最佳实践

### 需求开发最佳实践

#### 2 乙方不要以己度人？

(1) 需求由用户为主，优劣由用户定义，谨防投射效应

每个干系人的目标都不尽相同，项目经理要谨防投射效应的负面影响，不要以自己的认知和好坏定义项目需求和目标。

(2) 投射效应，以己度人，难以真正了解用户

投射响应 (Projection effect)：在认知他人时，人们总是不自觉地把己的意愿强加到别人身上，也就是以己度人的认知障碍，在生活中，善良的人总是认为全世界的任都是善良的。

投射响应有三种表现形式：相同投射（你妈觉得你冷）；愿望投射；情感投射。

## 04 需求开发最佳实践

### 需求开发最佳实践

#### 3 获取和挖掘需求的要点？

##### (1) 业务专家的介入是有价值的

获取及挖掘需求时，要善于使用业务专家，这有利于实现业务价值。

##### (2) 区分想要的和需要的

想要的趋向于用户想象的解决方案，需要的则与业务相关，区分二者的一个最佳实践是，用户愿意花钱的是真正需要的，用户不愿意花钱的是想要的。

##### (3) 获取需求的注意问题

汉语词汇的多义性使得了解非常具有挑战性；国人的含蓄增加了需求获取的难度。



01

软件需求的意义和任务

02

需求的层级要点

03

需求开发纪要

04

需求开发最佳实践

05

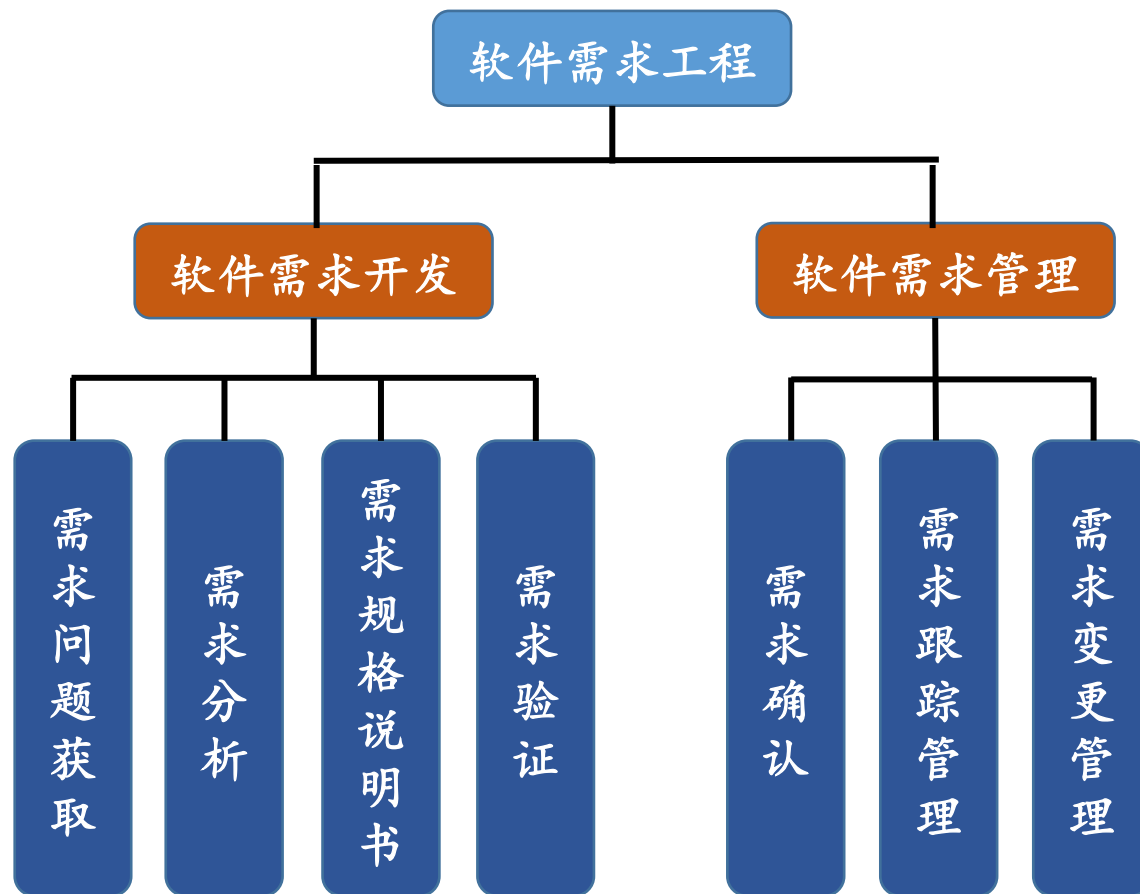
需求管理纪要

06

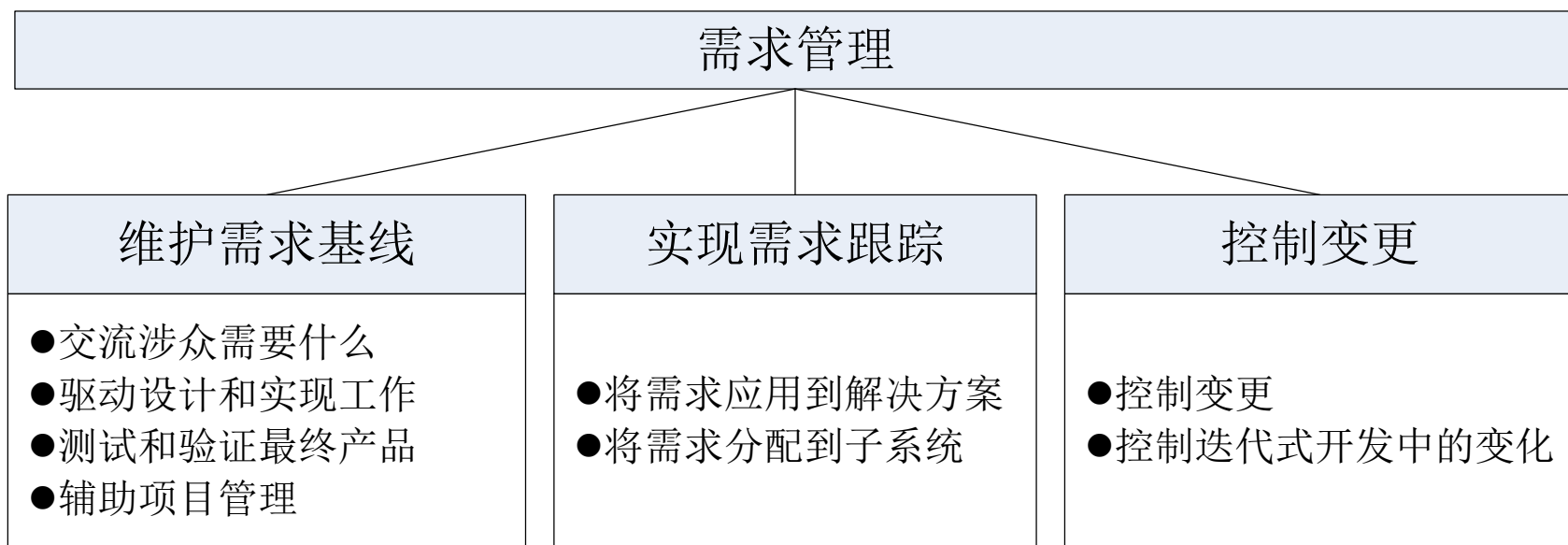
需求管理最佳实践



## 05 需求管理纪要



## 05 需求管理纪要



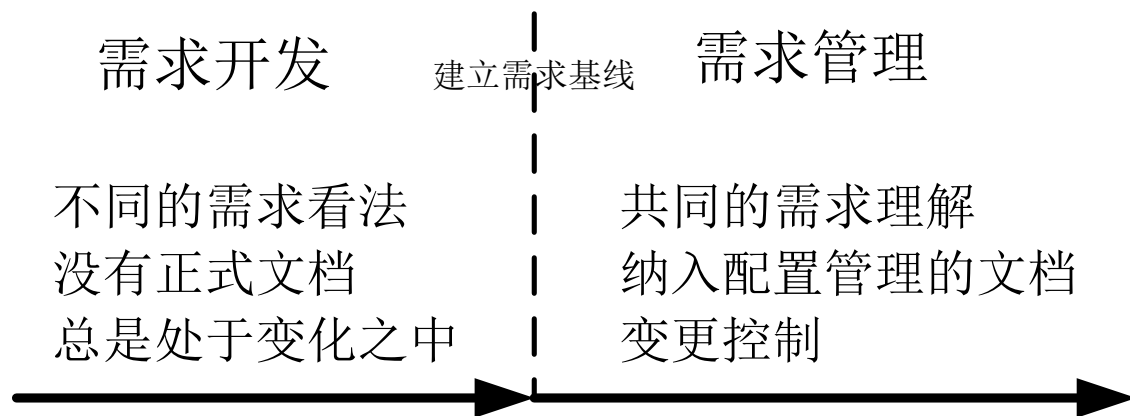
## 05 需求管理纪要

- 需求工程的基本活动
  - **需求管理**是对需求开发所建立的需求基线的管理，它在需求基线完成之后正式开始。并在需求工程结束之后继续存在，在设计、测试、时限等后继的软件系统开发中保证需求作用的持续、稳定发挥。用以监督并管理需求实现和变更。

## 05 需求管理纪要

### 需求基线

- 已经通过正式评审和批准的规格说明或产品，它可以作为进一步开发的基础，并且只有通过正式的变更控制过程才能修改它
- 是被明确和固定下来的需求集合，是项目团队需要在某一特定产品版本中实现的特征和需求集合



## 05 需求管理纪要

### 需求基线—描述内容

- 标识符（ID），为后续的项目工作提供一个共同的交流参照。
- 当前版本号（Version），保证项目的各项工作都建立在最新的一致需求基础之上。
- 源头（Source），在需要进一步深入理解或者改变需求时，可以回溯到需求的源头。
- 理由（Rational），提供需求产生的背景知识。
- 优先级（Priority），后续的项目工作可以参照优先级进行安排和调度。
- 状态（Status），交流和具体需求相关的项目工作状况。
- 成本、工作量、风险、可变性（Cost、Effort、Risk、Volatility），为需求的设计和实现提供参考信息，驱动设计和实现工作。
- 需求创建的日期；
- 和需求相关的项目工作人员，包括需求的作者、设计者、实现者、测试者等；
- 需求涉及的子系统；
- 需求涉及的产品版本号；
- 需求的验收和验证标准；
- ...

## 05 需求管理纪要

### 需求基线—配置管理

- 标识配置项
  - 递增数值，例如1, 2, ...x;
  - 层次式数值编码，例如1.1.1, 1.2.1, ...x.y.z;
  - 层次式命名编码，例如Order.Place.Date, Order.Place.Register, ...Task.Step.Substep
- 版本控制
  - 每一条单独的需求需要进行版本控制
  - 相关的需求文档也需要进行版本控制
- 变更控制
- 访问审计
  - 记录和审计访问的情况
- 状态报告
  - 反映需求基线的成熟度（变化的幅度越大，成熟度越低）、稳定性（改变的次数越多，稳定性越差）等

# 05 需求管理纪要

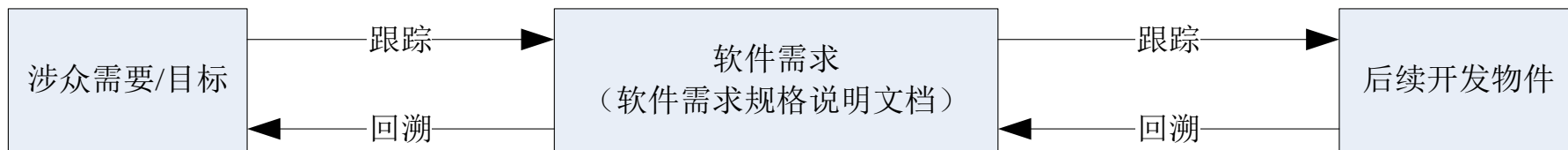
## 需求基线—状态维护

状态	定义
已 提 议 (Proposed)	该需求已被有相应权限的人提出
已 批 准 (Approved)	该需求已经被分析，它对项目的影响已进行了估计，并且已经被分配到某一特定版本的基线中。关键涉众已同意包含这一需求，软件开发团队已承诺实现这一需求
已 实 现 (Implemented)	实现这一需求的系统组件已经完成了设计和实现。这一需求已经被跟踪到相关的设计元素和实现元素
已 验 证 (Verified)	已在集成产品中确认了这一需求的功能实现是正确的。这一需求已经被跟踪到相关的测试用例。这一需求目前可以被认为是已完成了
已 删 除 (Deleted)	已批准的需求又从需求基线中取消了。要解释清楚为什么要删除这一需求，以及是谁决定删除的
已 否 决 (Rejected)	需求已被提议，但并不在下一版本中实现它。要解释清楚为什么要否决这一需求，以及是谁决定否决的

## 05 需求管理纪要

### 需求跟踪

- 避免在开发过程或者演化过程中与需求基线不一致或者偏离的风险



- 前向跟踪是指被定义到软件需求规格说明文档之前的需求演化过程
  - 向前跟踪到需求：说明涉众的需要和目标产生了哪些软件需求
  - 从需求向后回溯：说明软件需求来源于哪些涉众的需要和目标
- 后向跟踪是指被定义到软件需求规格说明文档之后的需求演化过程
  - 从需求向前跟踪：说明软件需求是如何被后续的开发物件支持和实现的
  - 回溯到需求的跟踪：说明各种系统开发的物件是因为什么原因（软件需求）而被开发出来的

## 05 需求管理纪要

### 需求跟踪—用途

- 需求的后向跟踪可以帮助项目管理者：
  - 评估需求变更的影响；
  - 尽早发现需求之间的冲突，避免未预料的产品延期；
  - 可以收集没有被实现的需求，并估算这些需求需要的工作量；
  - 发现可以复用的已有组件，从而降低新系统开发的时间和精力；
  - 明确需求的实现进度，跟踪项目的状态。
- 需求的后向跟踪可以帮助客户和用户：
  - 评价针对用户需求的产品的质量；
  - 可以确认成本上没有（昂贵的）镀金浪费；
  - 确认验收测试的有效性；
  - 确信开发者的关注点始终保持在需求的实现上。

## 05 需求管理纪要

### 需求跟踪—用途

- 需求跟踪中针对具体需求的设计方案选择、设计假设条件以及设计结果等信息可以帮助设计人员：
  - 验证设计方案正确的满足了需求；
  - 评估需求变更对设计的影响；
  - 在设计完成很久之后仍然可以理解设计的原始思路；
  - 评估技术变化带来的影响；
  - 实现系统组件的复用；
- 需求跟踪信息还可以帮助维护人员：
  - 评估某一个需求变化时对其他需求的影响；
  - 评估需求变化时对实现的影响；
  - 评估未变化需求对实现变更的允许度。

# 05 需求管理纪要

## 需求跟踪—方法（需求跟踪矩阵）

用户需求	功能性需求	设计组件	实现组件	测试用例
UC-28	Catalog.query.sort	Class catalog	Catalog.sort()	Search.7 Search.8
UC-29	Catalog.query.import	Class catalog	Catalog.import() Catalog.validate()	Search.12 Search.13 Search.14

## 05 需求管理纪要

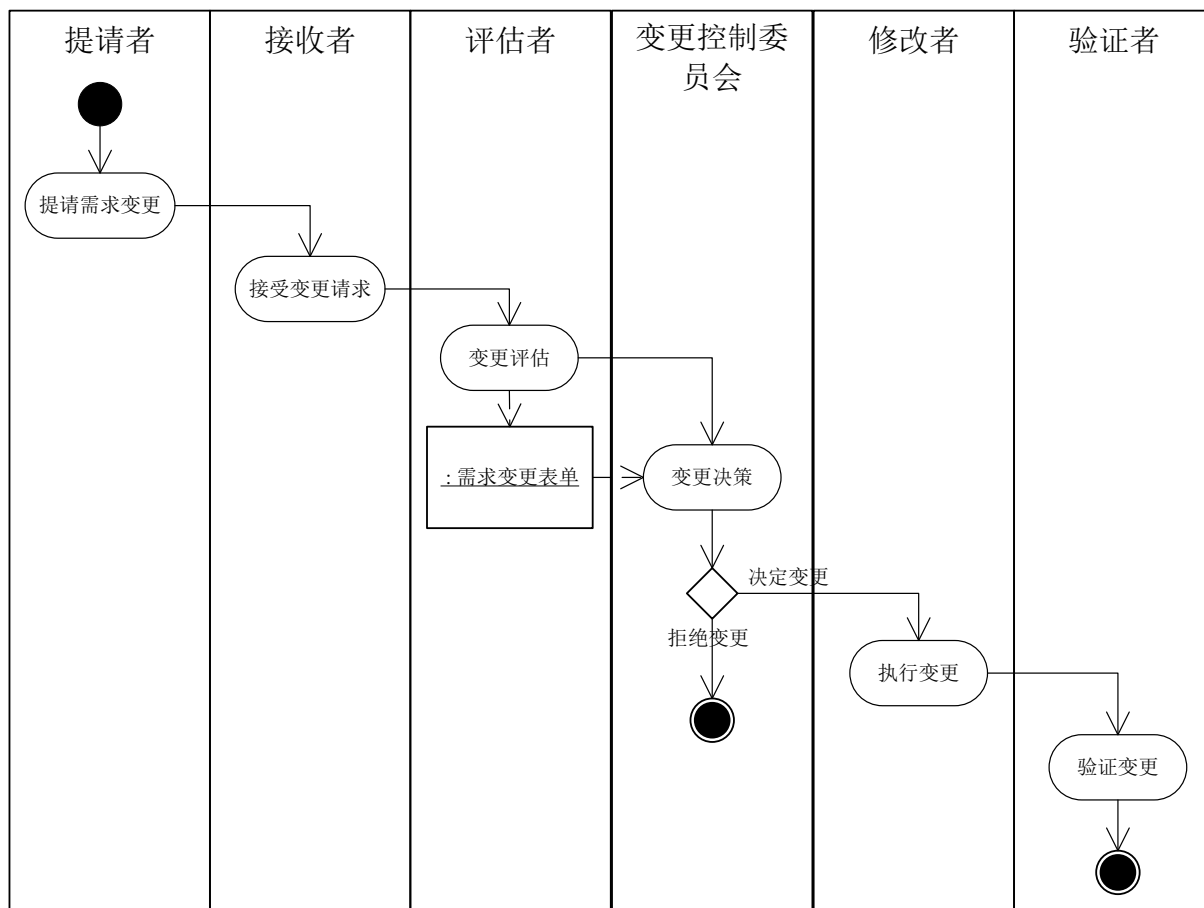
### 需求变更控制

- 需求的变化是正当和不可避免的
  - 问题发生了改变
  - 环境发生了改变
  - 需求基线存在缺陷
- 用户变动
- 用户对软件的认识变化
- 相关产品的出现

## 05 需求管理纪要

### 需求变更控制-过程

- 以可控、一致的方式进行需求基线中需求的变化处理，包括对变化的评估、协调、批准或拒绝、实现和验证





目录页  
CONTENTS PAGE

01

软件需求的意义和任务

02

需求的层级要点

03

需求开发纪要

04

需求开发最佳实践

05

需求管理纪要

06

需求管理最佳实践



## 06 需求管理最佳实践

### 1 谨防虚假共识

在20世纪60年代,美国人哈维根据自己生活中的一次经历描述了阿比勒尼悖论:他和太太以及岳父岳母在40摄氏度的高温下坐在位于德克萨州科勒曼城的家中的门廊里。科勒曼距离德州的另一个城市阿比勒尼大约53英里。四个人在酷暑里尽可能的减少活动,喝柠檬水,看着风扇懒洋洋的转,偶尔玩玩多米诺牌。过了一会儿,岳父建议开车去阿比勒尼,去那儿的一家餐厅吃饭。哈维作为女婿觉得这个主意很疯狂,但看不出任何反对的必要,因此他附和了该提议,太太和岳母也附和了该提议。

四个人上了没有空调的大众轿车,冒着尘暴驱车去阿比勒尼。他们在那的餐厅吃了一顿乏味的午餐,然后回到科勒曼,筋疲力尽,并且燥热难当。大家对这次经历普遍不满意。

直到他们到家后,才发现没有一个人真的想去阿比勒尼他们只是附和,因为他们认为其他人会盼着去。哈维将此命名为:“阿比勒尼悖论”,并且认为组织中也会出现类似的荒唐现象,即组织采取的行动往往与真正的意图相悖,不能达到理想的结果。

## 06 需求管理最佳实践

### 需求管理最佳实践

#### 1 谨防虚假共识

避免阿比勒尼悖论的方法：

- (1) 鼓励所有成员的参与，赋予那个成员自由发言的权利。
- (2) 促进团队成员与上级部门之间的沟通和互动。
- (3) 如果某些员工仍对公开表达自己的观点比较犹豫，请让他（她）抽取书面或网络的形式。
- (4) 在团队后组织实施任何决策钱，有必要进行最后的审核，以确保达成的不是“虚假”的共识。
- (5) 让成员们有时间分析情况，并能提出批判性建议，避免让他们“屈服”于仓促的决定。

## 06 需求管理最佳实践

### 2 需求定义（管理）最佳实践-鱼骨图与帕累托图

#### 背景

问题分析的五个步骤：

- 1、在问题上达成共识
- 2、理解根本原因—分析问题背后的问题
- 3、确定相关人员和用户
- 4、定义解决方案的界限
- 5、确定加在解决方案上的约束

2 需求定义（管理） 最佳实践-鱼骨图与帕累托图

问题分析就是理解真实世界中的问题和用户需求，并提出满足解决这些问题的解决方案的过程。因此需求定义首要工作是要把解决的问题提出来，并且得到所有人的共识。

找准开发新系统要解决的问题是项目成功的关键，即对问题进行了正确的定义，意味着成功解决了一半。

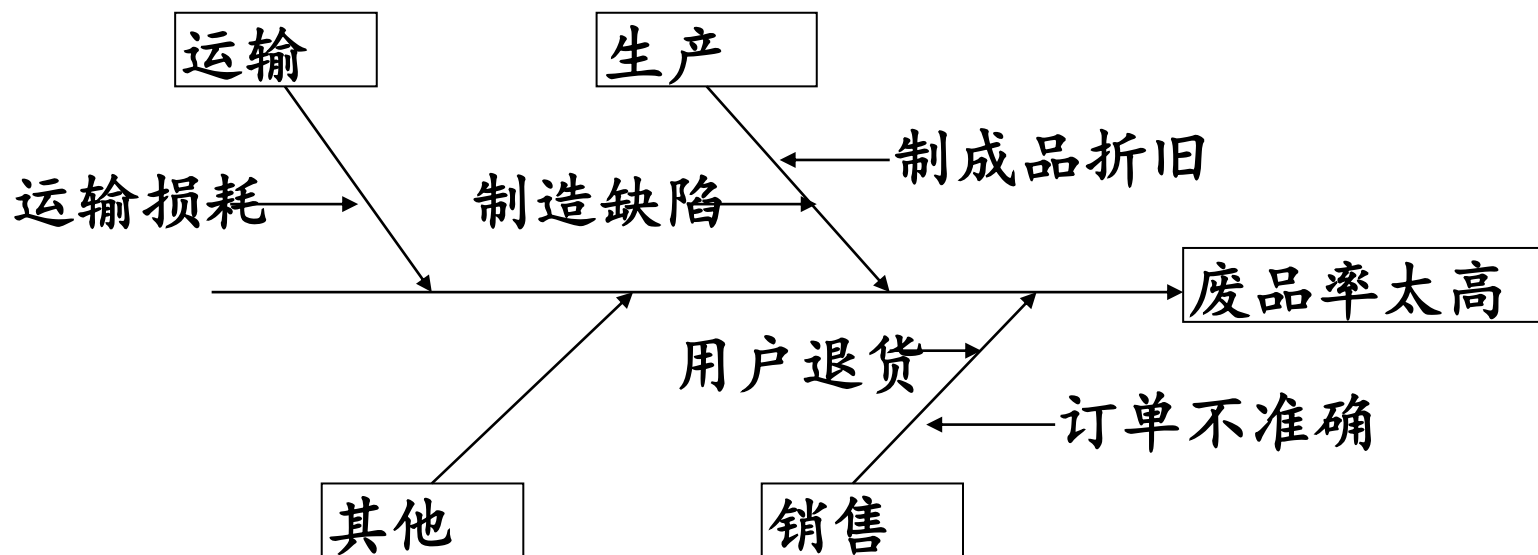
要让大家达成共识，采用统一的表格写出问题就是很有效的手段。RUP提供的描述模板如下：

写作项目	说明
问题	描述存在的问题，或要解决的问题
影响	该问题影响了哪些人群
结果	该问题对这类人群产生了什么影响
优点	预期什么样的解决方案，它应该具备什么优点

### 2 需求定义（管理）最佳实践-鱼骨图与帕累托图

分析问题背后的问题，即寻找问题的本源——**定性分析**。

**鱼骨图分析方法**：对每个问题绘制一张鱼骨图，找出问题的所有潜在原因。  
该方法是一种定性分析方法。



鱼骨图也叫因果分析图。上图中，“运输”、“生产”、“销售”等称为**原因类型**，“运输损耗”、“制造缺陷”、“订单不准确”等称为**原因**。

2 需求定义（管理） 最佳实践-鱼骨图与帕累托图

帕雷托分析方法：对每个问题绘制一张表，找出所有潜在原因对问题的影响程度。该方法是一种定量分析法。

原因	数量	比列
制造缺陷	29	4.83%
制成品折旧	45	7.50
订单不准确	317	52.83%
用户退货	82	13.67%
运输耗损	107	17.83%
其他	20	3.34%
合计	600	100%

### 2 需求定义（管理）最佳实践-鱼骨图与帕累托图

鱼骨图分析相当于寻找解决问题的靶子，而帕累托分析则相当于瞄准目标的准确性。

通过分析问题背后的原因，可以判断哪些原因可以通过信息系统来解决，从而使得系统的目标更加科学，需求的范围更易于确定。