

软件工程

第二部分 结构化方法学

需求分析

- 3.1 概述
- 3.2 需求分析目标与任务
- 3.3 需求获取方法
- 3.4 需求建模及其工具
- 3.5 需求描述与评审
- 3.6 需求管理

需求分析

1

教学要求

- 理解需求分析的目标与任务
- 掌握不同的需求分析的方法与常用工具
- 运用不同的需求分析方法与工具, 实施小型软件系统的需求分析

需求分析

2

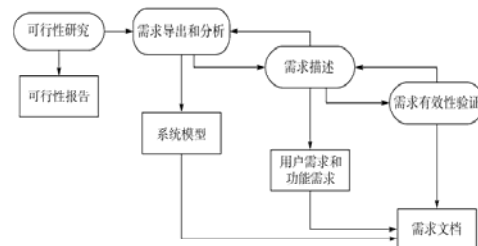
3.1 概述

- 基于计算机的系统的系统元素包括硬件、软件、人、数据库、文档和过程。
- 一般的需求定义:
指明系统必须实现什么的规格说明, 描述了系统的行为、特性或属性, 是在开发过程中对系统的约束。
- 需求工程包括需求分析与需求管理两部分。

需求分析

3

▪ 软件需求分析过程:



需求分析

4

3.2 需求分析目标与任务

- 需求分析是一项软件工程活动, 两个主要的目标是: **建立分析模型**和**规格说明**。具体说来:
 - ✓ 清楚地理解所要解决的问题, 完整地获取用户要求;
 - ✓ 刻划出软件的功能和性能;
 - ✓ 指明软件与其他系统元素的接口;
 - ✓ 建立软件必须满足的约束。

需求分析

5

- 软件需求分析是一项软件工程活动。包括如下需求分析的任务:

- ✓ 需求获取
- ✓ 需求建模
- ✓ 规格说明
- ✓ 需求验证

需求分析

6

需求获取

- ✓刻划出软件的功能、性能、可靠性与可用性、出错处理、未来可能的需求;
- ✓指明软件与其他系统元素的接口;
- ✓建立软件必须满足的约束。

需求分析

7

需求建模

- ✓需求分析通常要建立分析模型,包括数据模型、逻辑模型和行为模型。
- ✓这些模型为软件设计人员提供了可被翻译成数据、体系结构、接口和处理过程设计的模型。

需求规格说明

- ✓需求规格说明为开发人员和用户提供软件开发完成时质量评价的依据。

需求分析

8

需求验证

- ✓需求分析研究的对象是用户的要求。
- ✓必须全面理解用户的各项要求,准确表达被接受的用户要求。
- ✓只有经过确切描述的软件需求才能成为软件设计的基础。

需求分析

9

3.3 需求获取

- 需求获取是在问题及其最终解决方案之间架设桥梁的第一步。
- 需求获取的目的是:清楚地理解所要解决的问题,完整地获得用户的要求。
- 获取需求的一个必不可少的结果是对项目中描述的客户需求的普遍理解。一旦理解了需求,分析者、开发者和客户就能探索出描述这些需求的多种解决方案。

需求分析

10

1. 软件需求的层次

- 业务需求 反映了组织或客户对系统、产品高层次的目标要求,它们一般在项目视图和范围文档中给予说明。
- 用户需求 描述用户使用软件需要完成哪些任务,它们可通过使用实例图或脚本说明加以阐明。
- 功能—非功能需求 定义了开发者必须实现的软件功能,而非功能需求如表所示:

需求分析

11

产品要求	性能要求	实时性;其他时间要求,如响应时间、处理时间、包传送时间等;资源配置要求;精确度、处理量等要求
	可靠性要求	有效性;数据完整性
	安全保密要求	安全性;保密性
	运行要求	使用频度、运行期限;控制方式;对操作员要求
	物理要求	系统的规模等
过程要求	开发类型	实用性开发或试验性开发
	项目估算	开发工作量估计
	开发方法	质量控制标准;里程碑和评审;验收标准
	优先顺序	权衡各种质量目标要求,排定优先实现次序
	可维护性	可理解性、可测试性、可修改性、可移植性

需求分析

12

2. 需求获取应遵循的原则

- 抽象和分解是在人们认识世界和改造世界的长期实践中总结出来的行之有效的原则，在需求获取的过程中需遵循的三个原则：
 - a. **分解**：捕获问题空间的**整体-部分**关系。如问题/子问题分解；
 - b. **抽象**：捕获问题空间的**一般化-特殊化**关系。如问题的不同变型；
 - c. **投影**：捕获问题空间的**多维视图**。即从不同角度考察。

需求分析

13

3. 需求获取的步骤

- 软件开发项目和组织文化的不同，对于需求开发没有一个简单的、公式化的途径。
- 信息系统的需求获取通常包括如下9个步骤：
 - 1) **定义项目的视图和范围** 包括组织结构图、各部门的岗位/角色列表。
 - 2) **确定用户类** 包括人员/责任矩阵。
 - 3) **确定目标系统的业务工作流程** 包括物流、资金流、信息流，建立业务工作流程模型。

需求分析

14

- 4) **运用需求获取技术开发反映主要业务规则的数据流图并设置优先级。**
- 5) **收集来自用户的质量特性信息和其他非功能需求** 将性能、安全性、可靠性等需求和其他设计约束结合业务规则，形成功能需求。
- 6) **分类在数据流图中涉及的数据** 包括数据的组成和数据之间的关系。
- 7) **详细拟订数据流图的规格说明** 建立功能模型，并进行审查，用以澄清需求获取的参与者对需求的理解。

需求分析

15

- 8) **开发并评估界面原型** 设想输入设备、输出设备、显示风格、显示方式、输出格式等，建立接口规范和信息流传输规则。
- 9) **从功能描述中开发概念测试用例** 用测试用例来验证数据流图、功能需求和原型。

需求分析

16

4. 需求整理与表达的方法

- 采用**穷举方法**可以避免遗漏。
- 采用**归纳方法**，通过对各种情况进行综合分类可以使问题条理化。
- 采用**抽象方法**，可以发现问题实质，抓住问题的主要矛盾，忽略次要矛盾。
- 需求整理可以多种手段共用，如组织结构图、业务流程图、多叉树、关系矩阵、文字叙述、表格、图形等。需求描述包括组织结构与岗位定义、业务流程、处理规则、数据项、功能以及上述5个方面的关系。

需求分析

17

5. 需求获取技术的基本特征

- 需求获取可能是软件开发中最困难、最关键、最易出错及最需要交流的方面。表现在：
 - ✓ **需求的不稳定性**：在整个软件生存周期内软件需求会随着时间的推移发生变化；
 - ✓ **需求的不准确性**：用户和开发人员的认识会随着使用系统实现业务流程的实践逐步提高，一开始不可能设想得面面俱到。
- 需求获取只有通过有效的**客户/开发者的合作**才能成功。

需求分析

18

- 好的需求获取技术，对于规范需求获取活动，高效准确地获取需求定义，是十分重要的。
- 好的需求获取技术，应具有如下基本特征：
 - ✓提供便于沟通的工具，如易于理解的语言和直观的图表；
 - ✓提供定义系统边界（交互）的方法；
 - ✓提供支持抽象的机制，如“分解”、“映射”等；
 - ✓鼓励分析员使用面向问题的术语思考问题，编写文档；
 - ✓为分析员提供多种可供选择的解决方案；
 - ✓适应需求的变化。

需求分析

19

- 需求获取技术的关键点在于：

a. 深入浅出

- ✓需求获取要尽可能全面、细致。
- ✓获取的需求是个全集，系统真正实现的是个子集。分析时的调研内容并不都纳入到新系统中，目的在于以后的扩充。

需求分析

20

b. 以流程为主线

- ✓在与用户交流的过程中，应该用流程将所有的内容串起来。如信息、组织结构、处理规则等。这样便于交流沟通。
- ✓流程描述有宏观，也有微观。既要强调总体的业务流程、全生存周期的业务流程，又要对流程细化，有分支的业务流程。

需求分析

21

6. 常用的需求获取技术

1. 访谈

最早开始使用的获取用户需求的技术，也是迄今为止仍然广泛使用的需求分析技术。访谈有两种基本形式：

- ✓正式访谈时，系统分析员将提出一些事先准备好的具体问题。
- ✓在非正式访谈中，分析员将提出一些用户可以自由回答的开放性问题，以鼓励被访问人员说出自己的想法。

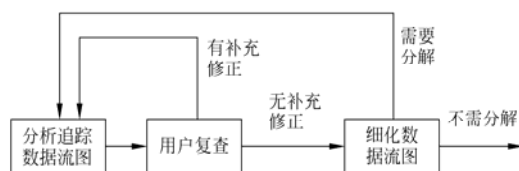
访谈中经常使用调查表、情景分析技术。

需求分析

22

2. 面向数据流自顶向下求精(结构化分析)

可行性研究得到目标系统的高层数据流图，需求分析则需将数据流和数据存储定义到元素级。



面向数据流自顶向下求精过程

需求分析

23

3. 简易的应用规格说明技术

一种面向团队的需求收集法。提倡用户与开发者密切合作，共同标识问题，提出解决方案要素，商讨不同方案并指定基本需求。典型过程如下：

- 首先进行初步的访谈，通过用户对基本问题的回答，初步确定待解决问题的范围和解决方案。
- 然后开发者和用户分别写出“产品需求”，并在集体会议上讨论。与会者分成小组，将意见一致的议题写成小型规格说明。
- 最后，由一名或多名与会者根据会议成果起草完整的软件需求规格说明书。

需求分析

24

4. 快速原型化方法

快速原型化方法是最准确、最有效、最强大的需求分析技术。

构建原型的要点：实现用户看得见的功能，省略目标系统的“隐含”功能。

构建原型应具备“快速”和“容易修改”等特性。

构建原型常采用“第四代技术”、“可重用的软件构件”、“形式化规格说明和原型环境”等技术与工具。

需求分析

25

3.4 需求建模及其工具

- 需求建模是为了分析需求，以确定项目的确切需求。
- 需求建模遵循三个原则：
 - ✓ **划分**：描述需求的整体-部分关系；
 - ✓ **抽象**：描述需求的一般化-特殊化关系；
 - ✓ **投影**：描述需求的多维视图；
- 定义系统模型要区分逻辑模型和物理模型。

需求分析

26

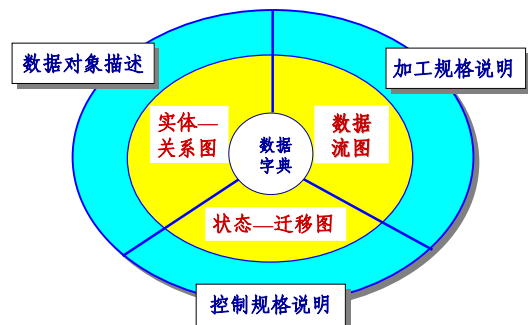
3.4.1 结构化分析方法

- 结构化分析方法最初只是着眼于数据流，自顶向下，逐层分解，建立系统的处理流程，以数据流图和数据字典为主要工具，建立系统的逻辑模型。
- 扩充后，将建模技术扩展到**数据建模**、**功能建模**和**行为建模**，以实体-关系图、数据流图和控制流图、状态-迁移图工具，数据字典为核心，从不同视点建立系统的分析模型。

需求分析

27

结构化分析的分析模型



需求分析

28

数据建模

- 需求分析阶段，按照用户的观点描述数据对象和它们之间的关系。建立的模型是概念性数据模型，又称信息模型。
- 描述数据模型的工具：**实体-关系图**
- 数据模型包括三种互相关联的信息：
 - ✓ **数据对象**，
 - ✓ **描述对象的属性**，
 - ✓ **描述对象间相互连接的关系**。

需求分析

29

实体-联系图

- **数据对象**是对软件必须理解的复合信息的抽象。仅有单个值的事物(例如，宽度)不是数据对象。数据对象可以是外部实体、事物、行为、事件、角色、单位、地点或结构等。表示为**矩形框**。
- **属性**定义了数据对象的性质。根据对所要解决的问题的理解，确定特定数据对象的一组合适的属性。表示为**椭圆形或圆角矩形**。
- **关系**是数据对象彼此之间相互连接的方式，也称为联系（1:1；1:N；M:N）。表示为**直线**。

需求分析

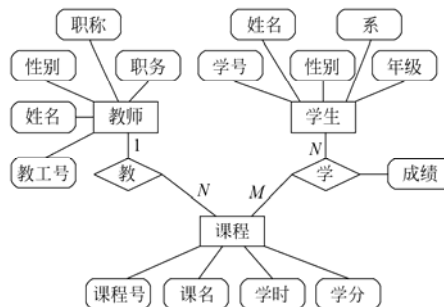
30

数据建模举例

- 例如，在教学管理中，一个教师可以教授零门、一门或多门课程，每位学生也需要学习几门课程。因此，教学管理中涉及的对象有学生、教师和课程。

需求分析

31

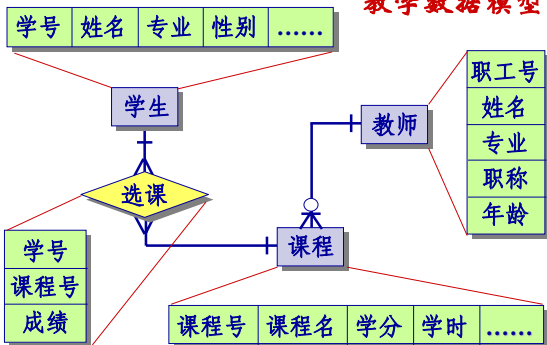


教学数据模型

需求分析

32

教学数据模型



需求分析

33

基数:一位教师

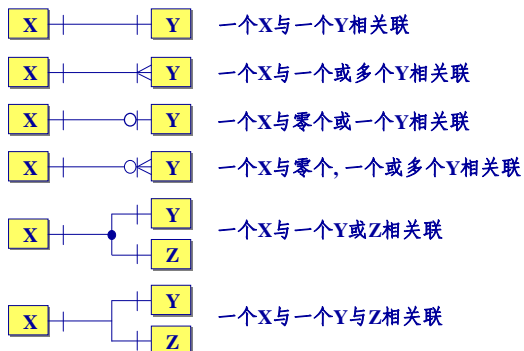
基数:多位学生



- 实例的关联有三种：一对一 (1:1)；一对多 (1:m)；多对多 (n:m)。
- 实例的关联称为“基数”，基数表明了“重复性”。

需求分析

34

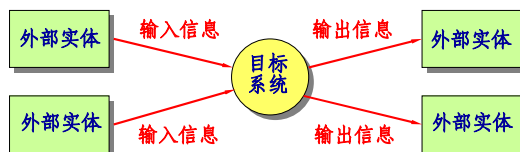


需求分析

35

功能建模和数据流图

- 结构化分析方法仅讨论数据流建模，目标系统被表示成如图所示的数据变换流程图。系统的功能体现在核心的数据变换中。



需求分析

36

行为建模

- 行为建模给出需求分析方法的所有操作原则，但只有结构化分析方法的扩充版本才提供这种建模的符号。
- 数据流图不描述时序关系，控制和事件流通过行为模型描述。
- 描述系统行为的工具：状态图。

需求分析

37

状态图

状态图通过描述系统的状态，以及引起系统状态转换的事件来表示系统的行为。

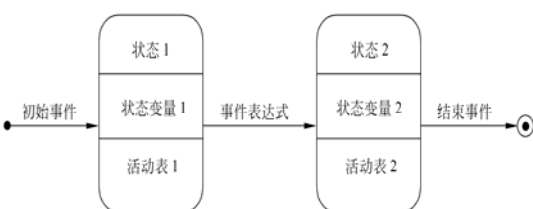
- 状态
 - 任何可以被观察到的系统行为模式，它状态规定了系统对事件的响应方式。既可以做动作，也可以改变系统本身的状态。一张状态图中只能有一个初态，终态则可以有0至多个。
- 事件
 - 事件是在某个特定时刻发生的事情，它是对引起系统做动作或状态转换的外界事件的抽象。

需求分析

38

状态图表示法

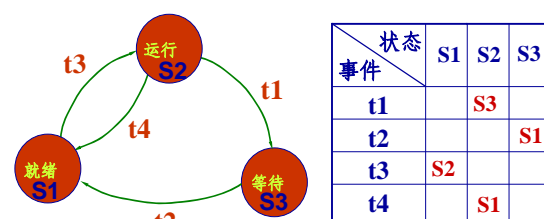
初态用实心圆表示，终态用一对同心圆表示，中间状态用圆角矩形表示。状态图中两个状态之间带箭头的连线称为状态转换，箭头指明了转换方向。



需求分析

39

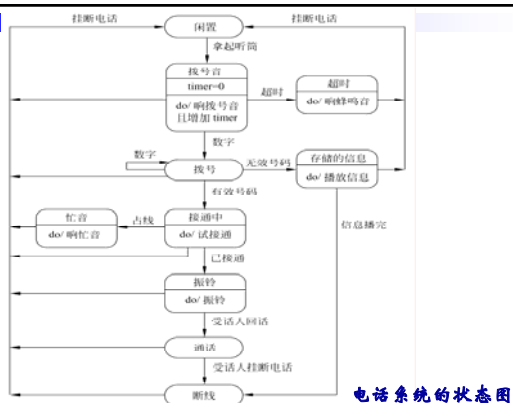
状态图举例



处理器分配的进程状态迁移

需求分析

40



电话系统的状态图

需求分析

41

Petri网

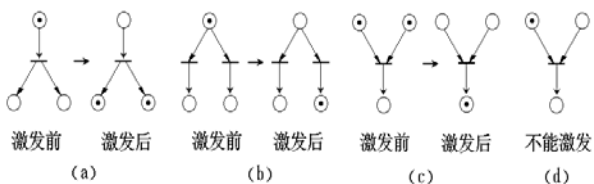
- Petri网已广泛地应用于硬件与软件系统的开发中，它适用于描述相互独立、协同操作的处理系统，也就是并发执行的系统。
- Petri网简称PNG (Petri Net Graph)，它有两种结点：
 - ✓ 位置：符号“○”，表示系统状态。
 - ✓ 转移：符号“|”，表示系统中的事件。
 - ✓ 有向边“→”表示向转移的输入，或从转移的输出。

需求分析

42

✓ 标记，或称令牌 (token)，是表明系统当前处于什么状态的标志。

■ Petri网可能的变化有：



需求分析

43

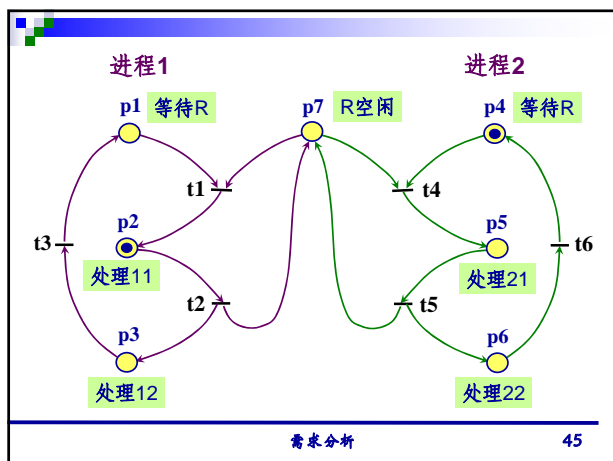
■ 例如，处理两个进程PR1和PR2的同步问题（此时两个进程共用一个资源R）：

进程 得到资源 占用资源运行 释放资源 不用资源运行



需求分析

44



需求分析

45

3.4.2 快速原型化方法

■ 一种有效驾驭风险的技术。通过原型：

- ✓ 可以增进软件者和用户对系统服务需求的理解，使比较含糊的具有不确定性的软件需求（主要是功能）明确化。
- ✓ 可以容易地确定系统的性能，确认各项主要系统服务的可应用性，确认系统设计的可行性，确认系统作为产品的结果。
- ✓ 有的原型可以直接成为产品，有的略加修改就可成为最终系统的一个组成部分。

需求分析

46

1. 原型分类

- 1) 探索型：
目的是要弄清对目标系统的要求，确定所希望的特性，并探讨多种方案的可行性。
- 2) 实验型：
这种原型用于大规模开发和实现之前，考核方案是否合适，规格说明是否可靠。
- 3) 进化型：
这种原型的目的在于改进规格说明，而是将系统建造得易于变化，在改进原型的过程中，逐步将原型进化成最终系统。

需求分析

47

2. 原型使用策略

- 软件原型支持需求工程的两项活动：
 - ✓ 需求获取
 - ✓ 需求有效性验证
- 其他用途：
 - ✓ 用户培训
 - ✓ 系统测试
- 原型开发主要分类：
 - ✓ 进化式原型开发
 - ✓ 抛弃式原型开发

需求分析

48

1) 进化式原型开发

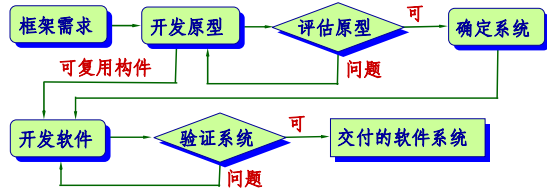


- 基本思路是：先给出一个系统的最初实现，让用户去使用和评价，不断进行细化和改善，经过多次这样的反复过程后形成最终的完善的系统。

需求分析

49

2) 抛弃式原型开发



- 基本思路是：原型的根本作用是弄清楚需求和为风险评估提供补充信息。通过评估后，原型被抛弃，重新规划和实施系统的开发。

需求分析

50

3. 原型开发技术

- 1) 可执行规格说明
- 2) 基于场景的设计
- 3) 自动程序设计
- 4) 专用语言
- 5) 可复用的软件
- 6) 简化假设

需求分析

51

3.4.3 其它需求分析工具

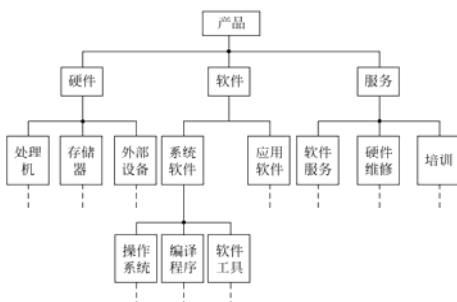
■ 层次方框图

用树形结构的一系列多层次的矩形框描绘数据的层次结构。树形结构的顶层是一个单独的矩形框，它代表完整的数据结构，下面的各层矩形框代表这个数据的子集，最底层的各个框代表组成这个数据的实际数据元素(不能再分割的元素)。

需求分析

52

■ 层次方框图例子



需求分析

53

■ Warnier图

信息层次结构的另外一种图形工具。

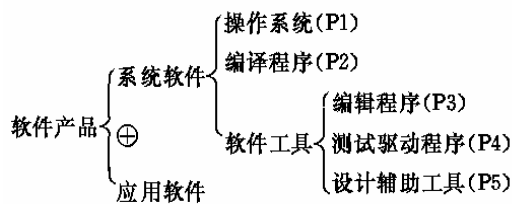
Warnier图也用树形结构描绘信息，比层次方框图提供了更丰富的描绘手段。它可以指出一类信息或一个信息元素是重复出现的，也可以表示特定信息在某一类信息中是有条件地出现的。

可以将Warnier图转变成软件设计的工具。

需求分析

54

■ Warnier图举例



需求分析

55

■ IPO图

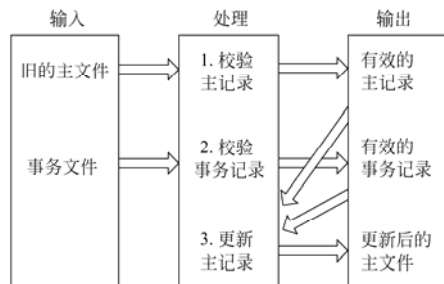
IPO图是输入、处理、输出图的简称，IBM公司发展完善起来的一种图形工具，能够方便地描绘输入数据、对数据的处理和输出数据之间的关系。

左边框中列出有关的输入数据，中间框列出主要的处理，在右边的框内列出产生的输出数据。处理框中列出处理的次序暗示了执行的顺序，用类似向量符号的粗大箭头清楚地指出数据通信的情况。

需求分析

56

■ IPO图举例



主文件更新的IPO图表示

需求分析

57

■ Z语言

Z是应用得最广泛的形式化语言，已经在许多软件开发项目中成功地运用。它具有如下一些特点：

- ✓ 易于发现用Z写的规格说明的错误；
- ✓ 增强了规格说明的精确性，减少了模糊性、不一致性；
- ✓ 可以严格地验证规格说明的正确性；
- ✓ 可以降低软件开发费用。

需求分析

58

■ Z语言描述的形式化规格说明含有下述4个部分：

- ✓ **给定的集合、数据类型及常数**: 初始化集合就是不需要详细定义的集合，这种集合用带方括号的形式表示。
- ✓ **状态定义**: 含有一组变量说明和一系列限定变量取值范围的谓词。
- ✓ **初始状态**: 抽象的初始状态是指系统第一次开启时的状态。
- ✓ **操作**: 操作的谓词部分，包含了一组调用操作的前置条件，以及操作完全结束后的后置条件。

需求分析

59

■ Z语言举例

集合 {Button}

说明	S
谓词	

格Button_State

Button_State
floor_buttons, elevator_buttons : P Button
buttons : P Button
pushed
floor_buttons ∧ elevator_buttons == Φ
floor_buttons ∪ elevator_buttons == buttons

初始状态 Button_Init = Button_State | pushed = Φ

操作Push_Button

Push_Button
△ Button_State
button? : Button
((button? ∈ buttons) ∧ ((button? ∈ pushed) ∧ (pushed' == pushed ∨ {button?})) ∨ ((button? ∈ pushed) ∧ (pushed' == pushed)))

需求分析

60

3.5 需求描述与评审

- 又称需求规格说明或需求规约。
- 需求定义的主要目的是分析需求草稿和模型，解决其中存在的二义性和不一致性，系统地准确地表达系统需求，形成需求规格说明。包括
 - ✓ 系统应提供的功能和服务；
 - ✓ 非功能需求；
 - ✓ 系统开发或运行的限制条件；
 - ✓ 与系统互连的其他系统的信息。

需求分析

61

1. 需求规格说明的原则

- 软件需求规格说明的基本原则 (1979, Balzer & Goldman提出):
 - ① 功能与实现分离，描述要“做什么”而不是“怎样实现”。
 - ② 要求使用面向处理的规格说明语言，从而得到“做什么”的规格说明。
 - ③ 如果目标软件只是一个大系统中的一个元素，那么整个大系统也包括在规格说明的描述之中。
 - ④ 规格说明必须包括系统运行的环境。

需求分析

62

- ⑤ 系统规格说明必须是一个认识的模型，而不是设计或实现的模型。
- ⑥ 规格说明必须是可操作的。
- ⑦ 规格说明必须容许不完备性并允许扩充。
- ⑧ 规格说明必须局部化和松散的耦合。当信息被修改时，只要修改某个单个的段落，能够很容易地加入和删去一些段落。

需求分析

63

需求规格说明的内容与格式

- 通常用自然语言完整、准确、具体地描述系统的数据要求、功能需求、性能需求、可靠性和可用性要求、出错处理需求、接口需求、约束、逆向需求以及将来可能提出的要求。
- 形式化方法描述用户对软件系统的需求，可以消除用自然语言书写的软件需求规格说明书中可能存在的~~不一致、歧义、含糊、不完整及抽象层次混乱~~等问题。

需求分析

64

基于IEEE 830 改写的规格说明模板格式:

- | | |
|----------------|--------------|
| a. 引言 | b. 综合描述 |
| a.1 目的 | b.1 产品的前景 |
| a.2 文档约定 | b.2 产品的功能 |
| a.3 预期的读者和阅读建议 | b.3 用户类和特征 |
| a.4 产品的范围 | b.4 运行环境 |
| a.5 参考文献 | b.5 设计和实现的限制 |
| | b.6 假设和依赖 |

需求分析

65

- | | |
|---------------|---------------|
| c. 外部接口 | e. 其他非功能需求 |
| c.1 用户界面 | e.1 性能需求 |
| c.2 硬件接口 | e.2 基本设施需求 |
| c.3 软件接口 | e.3 安全性需求 |
| c.4 通信接口 | e.4 软件质量属性 |
| d. 系统特性 | e.5 业务规则 |
| d.1 说明和优先级 | e.6 用户文档 |
| d.2 激励 / 响应序列 | f. 其他需求 |
| d.3 功能需求 | 附录A: 词汇表 |
| | 附录B: 软件需求分析模型 |
| | 附录C: 待确定的问题 |

需求分析

66

需求验证

- 又称需求评审。需求评审的目的是确保需求编写正确。通常需要从**一致性、完整性、现实性和有效性**四个方面验证需求的正确性。
- 审查的内容包括：
 - 1) 系统定义的目标是否与用户的要求一致；
 - 2) 需求分析阶段提供的文档资料是否齐全；
 - 3) 文档中的所有描述是否完整、清晰、准确反映用户要求；

需求分析

67

- 4) 与所有其它系统成分的重要接口是否都已经描述；
- 5) 主要功能是否已包括在规定的软件范围之内，是否都已充分说明；
- 6) 软件的行为和它必须处理的信息、必须完成的功能是否一致；
- 7) 设计的限制条件是否符合实际；
- 8) 是否考虑了开发的技术风险；
- 9) 是否详细制定了检验标准，它们能否对系统定义是否成功进行确认；

需求分析

68

验证需求的方法与工具

- ✓ **一致性** 自然语言书写的规格说明书只能依赖人工技术审查验证。当需求规格说明书是用形式化的需求陈述语言书写时，则存在自动化验证工具。
- ✓ **完整性与有效性** 需要用户的密切合作，比较现实的方法是开发原型系统验证。
- ✓ **现实性** 参照以往开发类似系统的经验，分析用现有的软、硬件技术实现目标系统的可能性；或通过仿真与性能模拟进行辅助分析。

需求分析

69

3.6 需求管理

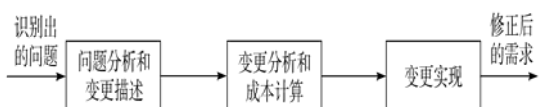
- 需求管理就是对系统**需求变更**了解和控制的过程。它包括一组帮助项目组在任何阶段、任何时候去**标识、控制和跟踪**需求的活动。
- 需求标识应能唯一确定每个软件需求。常用的需求标识方法包括：
 - ✓ **序列号** 例如SRS-13。该方法不提供任何相关需求在逻辑上或层次上的区别，不提供任何有关需求内容的信息。
 - ✓ **层次化编码** 例如3.2.4.3。不提供任何有关需求内容的信息。改进的方法：需求主要部分层次化+文字代码

需求分析

70

■ 变更管理(或变更控制)规定了如何处理需求变更的策略和过程。

- ✓ **问题分析和变更描述** 应如何提交一个需求变更请求？
- ✓ **变更分析和成本分析** 如何分析需求变更对范围、进度和成本的影响？
- ✓ **变更实现** 如何批准或驳回需求变更？如果批准了变更，该变更如何实现？



需求分析

71

- 当变更发生的时候，必须追踪这些变更对其它需求和系统设计的影响。
- 可跟踪性反映了发现相关的需求的能力。三类可跟踪性需要维护，可采用**需求跟踪表**来记录。

需求	属性						
	优先级	状态	成本	难度	稳定度	追溯目标	追溯来源
需求1	2						
需求2	1						
...
需求n	3						

需求分析

72

小结

- 需求分析的目标是清楚地理解所要解决的问题，完整地获取用户要求，结果是建立需求分析模型和软件需求规格说明。
- 需求分析是发现、求精、建模、规格说明和复审的过程。常用的需求获取技术包括访谈，面向数据流自顶向下逐步求精，简易的应用规格说明技术，快速原型化方法。
- 传统的软件工程方法学中，结构化分析技术通常建立数据模型、功能模型和行为模型。

使用实体-联系图建立数据模型，使用数据流图建立功能模型，使用状态图建立行为模型

- 软件需求规格说明书，经过严格评审并得到用户确认之后，作为需求分析阶段的最终成果。通常主要从一致性、完整性、现实性和有效性等方面复审软件需求规格说明书。
- 需求工程包括需求分析与需求管理。需求管理是对系统需求变更了解和控制的过程，包括标识、控制和跟踪需求的活动。