

# 第3章

# 需求分析

### 【本章简介】

本章主要介绍需求分析的任务和步骤、需求分析方法和需求分析规格说明,并在结构化分析方法中详细介绍各种方法的使用场景。通过本章学习,读者可以学习到软件工程中软件需求的分析要求。

### 【知识导图】



### 【学习目标】

- 了解软件需求的定义和要求。
- 掌握软件需求分析的步骤。
- 掌握结构化分析的方法。
- 能够依照国标文档,结合需求案例分析,了解并掌握撰写需求报告的能力。



### 趣味小知识

纳米机器人的设计与制造,与我们普遍认知的“机器人”大不相同。现今的机器人研究方向多以模仿人类的行为为主,也就是所谓的人型机器人,以精密的机械结构来执行双脚走路,并且能对周遭环境变化做出立即且适当的反应,而这些动作须要有强大的侦测感知装置及高度协调的神经网络作后盾。

## 3.1 软件需求分析概述

### 3.1.1 软件需求分析的目的

软件需求分析是软件生存周期中重要的一步,也是最关键的一步。只有通过软件需求分析,才能把软件的功能和性能的总体概念描述为具体的软件需求规格说明,进而建立软件开发的基础。

软件需求分析的目的是解决现实世界的问题,这也是它最基本的特点。因此,软件需求分析是用来解决某个具体问题的。举个例子,使用某款软件的用户可能在使用途中遇到一些问题,例如,公司或组织里的业务流程问题,或者控制某种设备等问题。而在实际中改正软件中存在的缺点、完善用户的功能、理清业务流程、纠正设备等都是非常复杂的。因此,软件需求分析是一个非常重要且复杂的流程,这些需求可能来自一个组织中不同层次的人。另外,在做需求分析时还要考虑软件的运行环境。

软件需求的一个重要特性就是它们是可验证的。验证某个软件的需求非常困难,代价很大。例如,开发呼叫中心模拟软件就必须验证吞吐量需求。软件需求和软件质检人员必须保证软件是可验证的。

需求除了表现出来的行为属性外,还有其他属性。常见的例子包括优先级,它使得资源有限的情况下保证开发的正常进行,使项目进展能被监测。一般的软件需求应该非常明确,以便于判断是否符合软件的要求,方便软件开发周期中的管理。

### 3.1.2 软件需求分析要素

#### 1. 软件需求分析涉及的内容与要素

软件需求分析涉及的内容与要素较多,主要包括以下 4 方面。

(1) 在功能方面,需求包括系统要做什么;相对于原系统,目标系统需要进行哪些修改;目标用户有哪些;以及不同用户需要通过系统完成何种操作等。

(2) 在性能方面,需求包括用户对于系统执行速度、响应时间、吞吐量和并发度等指标的要求。

(3) 在运行环境方面,需求包括目标系统对于网络设置、硬件设置、温度和湿度等周围环境的要求,以及操作系统、数据库和浏览器等软件配置的要求。

(4) 在界面方面,需求涉及数据的输入/输出格式的限制及方式、数据的存储介质和显示器的分辨率要求等问题。

#### 2. 软件需求分析的分类分析方法

软件需求分析可以采用下面的分类分析方法。

##### (1) 产品和过程需求。

产品参数和过程参数有明显的区别。产品参数是待开发软件的要求,例如,软件可以验证一个学生在选一门课之前是否满足选课条件。过程参数本质上是对软件开发的一个限制,例如,要求一个软件使用 Ada 语言编写。这有时被称为过程需求,许多软件都暗含过程需求,验证技术的选择就是一个典型的例子;另一个例子是使用严谨的分析方法(如形式验证方法),它可以用来减少导致低可靠性的错误。过程需求可能是直接由开发组织、客户或

者第三方(如安全管理者)提出的。

#### (2) 功能性和非功能性需求。

功能需求用来描述系统应该做什么,即为用户和其他系统完成的功能、提供的服务。例如,客户登录、邮箱网站的收发邮件、论坛网站的发帖留言等。

非功能性需求是指必须遵循的标准:外部界面的细节、实现的约束条件、质量属性等。非功能需求限定了选择解决问题方案的范围,如运行平台、实现技术、编程语言和工具等。

这些需求的例子如下。

- ① 硬件、软件和将遵照的通信接口。
- ② 必须服从公司标准的用户界面。
- ③ 将被坚持的报告格式。
- ④ 过程限制,如 ISO 9000 等。
- ⑤ 基础设施造成的硬件限制。

#### (3) 量化需求。

应该尽可能清楚地陈述软件需求,避免主观判断的、含糊的及不能验证的软件需求。例如,“呼叫中心的软件要求增加吞吐量、减少错误可能性”,这个需求可量化为“呼叫中心的软件必须增加 20% 的吞吐量,在任何营业时间出现一个致命错误的可能性应低于  $1 \times 10^{-8}$ ”等。不管是功能性需求还是非功能性需求,都应该如此,而不能简单地陈述为“软件应该是可靠的,软件应该是用户友好的”这种形式。

### 3.1.3 系统需求分析要素

系统意味着为了完成某一目标而相互作用的元素的组合。这些元素包括硬件、软件、固件、人、信息、技术、设施、服务和其他系统工程、国际委员会定义的元素。

在一个包含软件组件的系统中,软件需求来源于系统需求。在有些文献中也称系统需求为用户需求。系统需求包括用户需求、其他投资者的需求(如认证机构)和无法确定的人力资源的需求。

需求分析的任务不是确定系统如何完成工作,而是确定系统必须完成哪些工作,也就是对目标系统提出完整、准确、清晰、具体的要求。

(1) 确定目标系统的具体要求。需求分析阶段要确定目标系统的具体要求。

① 确定系统的运行环境要求。系统运行时的硬件环境要求,如外存储器种类、数据输入方式、数据通信接口等;软件环境要求,如操作系统、汉字系统、数据库管理系统等。

② 确定系统性能要求。如系统所需要的存储容量、安全性、可靠性、期望的响应时间(即从终端输入数据到系统后,系统在多长时间内可以有反应并输出结果,这对于实时系统来讲是关系到系统能够被用户接受的重要因素)等。

③ 确定系统功能要求。确定目标系统必须具备的所有功能、系统功能的限制条件和设计约束。

④ 确定接口需求。接口需求描述系统与其环境通信的格式。常见的接口需求有用户接口需求、硬件接口需求、软件接口需求、通信接口需求等。

(2) 建立目标系统的逻辑模型。需求分析实际上就是建立系统模型的活动。

模型是为了理解事物而对事物做出一种抽象、无歧义的书面的描述。模型由一组图形符

号和组成图形的规则组成。建模的基本目标如下。

- ① 描述用户需求。
- ② 为软件的设计奠定基础。
- ③ 定义一组需求,用以验收软件产品。

模型分为数据模型、功能模型和行为模型。为了理解和表示问题的信息域,建立数据模型;为了定义软件的功能,建立功能模型;为了表示软件的行为,建立行为模型。在分析过程中,可用层次的方式来细分这三个模型,以得出软件实现的具体细节。

## 3.2 需求分析的原则与步骤

### 3.2.1 需求分析的原则

需求分析的原则如下。

- (1) 必须能表达和理解问题的数据域和功能域。其中,数据域包括数据流、数据内容和数据结构。
- (2) 自顶向下逐层分解问题。
- (3) 要给出系统的逻辑视图和物理视图。
- (4) 逻辑视图给出软件要达到的功能和要处理数据之间的关系,而不是实现的细节。
- (5) 物理视图给出处理功能和数据结构的实际表示形式。

### 3.2.2 需求分析的一般步骤

遵循科学的需求分析步骤可以使需求分析工作更高效,其一般步骤如图 3-1 所示。



图 3-1 需求分析的一般步骤

- (1) 获取需求,识别问题。

开发人员从功能、性能、界面和运行环境等多个方面识别目标系统要解决哪些问题,要满足哪些限制条件,这个过程就是对需求的获取。开发人员通过调查研究,要理解当前系统的工作模型和用户对新系统的设想与要求。

此外,在获取需求时,还要明确用户对系统的安全性、可移植性和容错能力等其他要求。例如,多长时间需要对系统做一次备份,系统对运行的操作重启系统允许的最长时间是多少等。

获取需求是需求分析的基础,为了能有效地获取需求,开发人员应该采取科学的需求获取方法。在实践中,获取需求的方法有很多种,如问卷调查、访谈、实地操作、建立原型系统等。

- ① 问卷调查法:采用调查问卷的形式是进行需求分析的一种方法。

通过对用户填写的调查问卷进行汇总、统计和分析,开发人员可以得到有价值的信息。采用这种方法时,调查问卷的设计很重要。一般在设计调查问卷时,要合理地控制开放式问题和封闭式问题的比例。开放式问题的回答不受限制,自由灵活,能够激发用户的思维,使

他们能尽可能地阐述自己的真实想法。但是,对开放式问题进行汇总和分析的工作会比较复杂。封闭式问题的答案是预先设定的,用户从若干答案中进行选择。封闭式问题便于对问卷信息进行归纳与整理,但是会限制用户的思维。

② 访谈:通过开发人员与特定的用户代表进行座谈,进而了解用户的意见,是最直接的需求获取方法。

为了使访谈有效,在进行访谈之前,开发人员要首先确定访谈的目的,进而准备一个问题列表,预先准备好希望通过访谈解决的问题。在访谈的过程中,开发人员要注意态度诚恳,保持虚心求教的姿态,同时还要对重点问题进行深入的讨论。由于被访谈的用户身份可能多种多样,开发人员要根据用户的身份特点进行提问,给予启发。当然,进行详细的记录也是访谈过程中必不可少的工作。访谈完成后,开发人员要对访谈的收获进行总结,澄清已解决的和有待进一步解决的问题。

③ 实地操作:开发人员以用户的身份直接参与到现有系统的使用过程中,亲身实践以更好地了解用户需求。

为了深入地了解用户需求,有时候开发人员还会以用户的身份直接参与到现有系统的使用过程中,在亲身实践的基础上,更直接地体会现有系统的弊端,以及新系统应该解决的问题。通过实地操作得到的信息会更加准确和真实,但是这种方法比较费时间。

④ 建立原型系统:当用户本身对需求的了解不太清晰的时候,开发人员通常采用建立原型系统的方法,对用户需求进行挖掘。

原型系统就是目标系统的一个可操作的模型。在初步获取需求后,开发人员会快速地开发一个原型系统。通过对原型系统进行模拟操作,开发人员能及时获得用户的意见,从而对需求进行明确的分析。利用原型系统获取需求的方法示意图如图 3-2 所示。

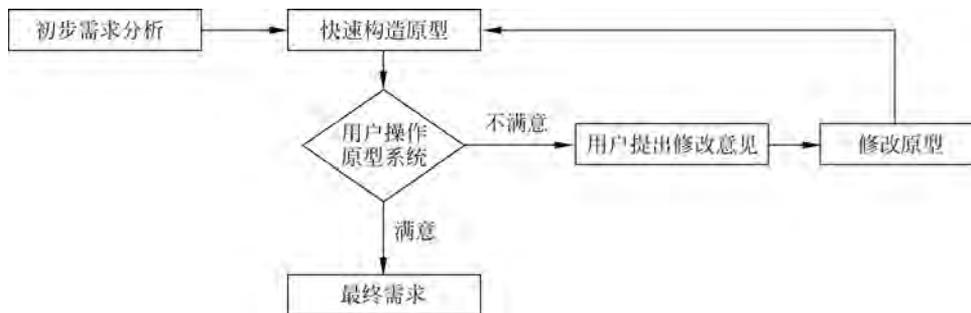


图 3-2 利用原型系统获取需求

### (2) 分析需求,建立目标系统的逻辑模型。

在获得需求后,开发人员应该对问题进行分析抽象,并在此基础上从高层建立目标系统的逻辑模型。模型是对事物高层次的抽象,通常由一组符号和组织这些符号的规则组成,常用的模型图有数据流图、E-R图、用例图和状态转换图等,不同的模型从不同的角度或不同的侧重点描述目标系统。绘制模型图的过程,既是开发人员进行逻辑思考的过程,也是开发人员进一步认识目标系统的过程。

### (3) 将需求文档化。

获得需求后要将其描述出来,即将需求文档化。对于大型的软件系统,需求阶段一般会

输出以下 3 个文档。

- ① 系统定义文档(用户需求报告)。
- ② 系统需求文档(系统需求规格说明书)。
- ③ 软件需求文档(软件需求规格说明书)。

对于简单的软件系统而言,需求阶段只需要输出软件需求文档(即软件需求规格说明书)即可。软件需求规格说明书主要描述软件的需求,从开发人员的角度对目标系统的业务模型、功能模型和数据模型等内容进行描述。作为后续的软件设计和测试的重要依据。需求阶段的输出文档应该具有清晰性、无二义性和准确性,并且能够全面和确切地描述用户需求。

#### (4) 需求验证。

需求验证是对需求分析的成果进行评估和验证的过程。为了确保需求分析的正确性、一致性、完整性和有效性,提高软件开发的效率,为后续的软件开发做好准备,需求验证的工作非常必要。

在需求验证的过程中,可以对需求阶段的输出文档进行多种检查,如一致性检查、完整性检查和有效性检查等。同时,需求评审也是在这个阶段进行的。



视频讲解

## 3.3 结构化分析方法

结构化分析方法是强调开发方法的结构合理性以及所开发软件的结构合理性的软件开发方法。

结构是指系统内各个组成要素之间的相互联系、相互作用的框架。结构化分析方法提出了一组提高软件结构合理性的准则,如分解与抽象、模块独立性、信息隐蔽等。针对软件生存周期各个不同的阶段,有结构化分析(SA)和结构化程序设计(SP)等方法。它一般利用图形表达用户需求,使用的手段主要有数据流图、数据字典、结构化语言、判定表以及判定树等。结构化分析的步骤如下。

- (1) 分析当前的情况,做出映应当前物理模型的 DFD。
- (2) 推导出等价的逻辑模型的 DFD。
- (3) 设计新的逻辑系统,生成数据字典和基元描述。
- (4) 建立人机接口,提出可供选择的目标系统物理模型的 DFD。
- (5) 确定各种方案的成本和风险等级,据此对各种方案进行分析。
- (6) 选择一种方案。
- (7) 建立完整的需求规约。

本节关于各种结构化分析方法绘图所使用的软件为 ProcessOn。ProcessOn 隶属于北京大麦地信息技术有限公司,是一款专业在线作图工具和分享社区。它支持流程图、思维导图、原型图、网络拓扑图和 UML 等多种类型图的绘制。



视频讲解

### 3.3.1 数据流图

#### 1. 数据流图的概念

数据流图(Data Flow Diagram, DFD)从数据传递和加工角度,以图形方式来表达系统的逻辑功能、数据在系统内部的逻辑流向和逻辑变换过程,是结构化分析方法的主要表达工

具及用于表示软件模型的一种图示方法。

数据流图是结构化分析方法中使用的工具,它以图形的方式描绘数据在系统中流动和处理的过程。由于它只反映系统必须完成的逻辑功能,所以它是一种功能模型。在结构化分析方法中,数据流图是需求分析阶段产生的结果。数据流图从数据传递和加工的角度,以图形的方式刻画数据流从输入到输出的移动变换过程。

## 2. 数据流程图中的主要元素

(1) →: 数据流。数据流是数据在系统内传播的路径,由一组成分固定的数据组成。如订票单由旅客姓名、年龄、单位、身份证号、日期、目的地等数据项组成。由于数据流是流动中的数据,所以必须有流向,除了与数据存储之间的数据流不用命名外,数据流应该用名词或名词短语命名。

(2) □: 数据源或宿(“宿”表示数据的终点),代表系统之外的实体,可以是人、物或其他软件系统。

(3) ○: 对数据的加工(处理)。加工是对数据进行处理单元,它接收一定的数据输入,对其进行处理,并产生输出。

(4) =: 数据存储。表示信息的静态存储,可以代表文件、文件的一部分、数据库的元素等。

## 3. 在单张数据流图时,必须注意遵循原则

(1) 一个加工的输出数据流不应与输入数据流同名,即使它们的组成成分相同。

(2) 保持数据守恒。也就是说,一个加工所有输出数据流中的数据必须能从该加工的输入数据流中直接获得,或者说是通过该加工能产生的数据获得。

(3) 每个加工必须既有输入数据流,又有输出数据流。

(4) 所有的数据流必须以一个外部实体开始,并以一个外部实体结束。

(5) 外部实体之间不应该存在数据流。

## 4. “销售管理系统”案例

下面将结合“销售管理系统”案例按步骤分层作图。

(1) 画出系统的输入/输出,即先画顶层数据流图。

顶层流图只包含一个加工,用以表示被开发的系统,然后考虑该系统有哪些输入数据流、输出数据流,其作用是表明被开发系统的范围以及它和周围环境的数据交换关系。图 3-3 所示为“销售管理系统”的顶层图。

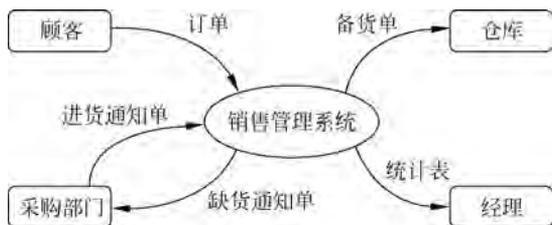


图 3-3 “销售管理系统”顶层图

(2) 画出系统内部,即画下层数据流图。

不再分解的加工称为基本加工。一般将层号从 0 开始编号,采用自顶向下、由外向内的原则。画 0 层数据流图时,分解顶层流图的系统为若干子系统,决定每个子系统间的数据接

口和活动关系。

可以用下述方法来确定加工。

① 在数据流的组成或值发生变化的地方应该画出一个加工图形,用以表示这一变化过程,也可以根据系统的功能决定加工图。

② 确定数据流的方法:用户把若干数据当作一个单位来处理(这些数据一起到达、一起处理)时,可以把这些数据看成一个数据流。

③ 关于数据存储:对于一些以后某个时间要使用的数据,可以组织成为一个数据存储来表示。

图 3-4 所示即为“销售管理系统”的 0 层图。

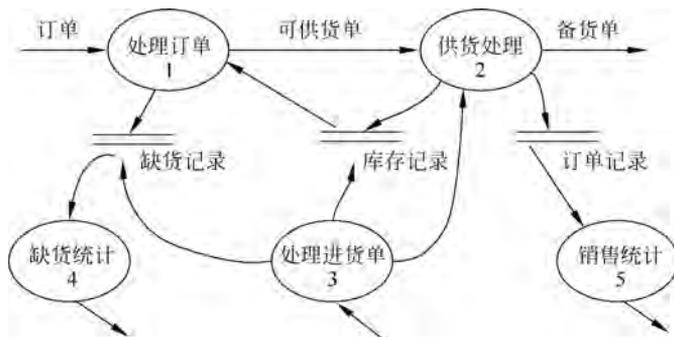


图 3-4 “销售管理系统”0 层图

(3) 画出加工的内部。

把每个加工看作一个小系统,把加工的输入/输出数据流看成小系统的输入/输出流。于是可以像画 0 层图一样画出每个小系统的加工的数据流图。

(4) 画子加工的分解图。

对第(3)步分解出来的数据流图中的每个加工,重复第(3)步的分解过程,直到图中尚未分解的加工都是足够简单的(即不可再分解)。至此,得到一套“销售管理系统”分层数据流图(1 层图),分别如图 3-5(a)~图 3-5(d)所示。

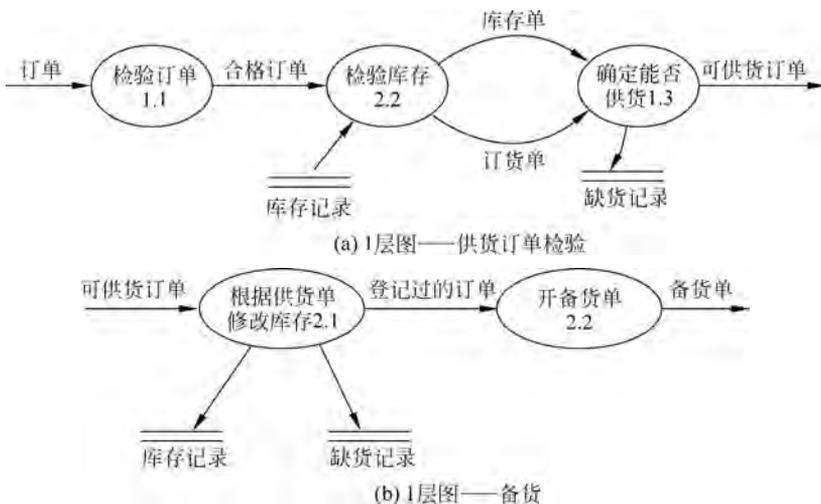
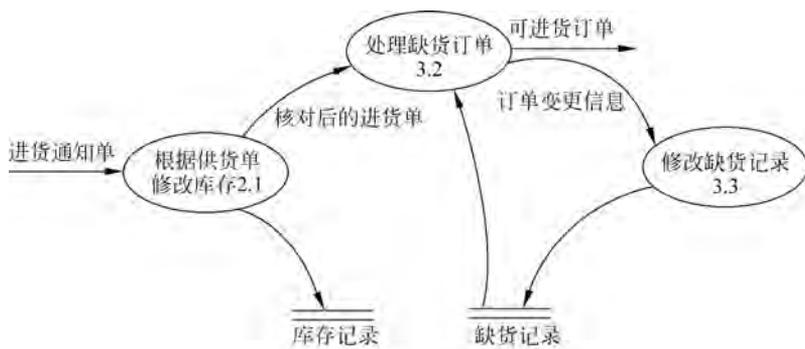
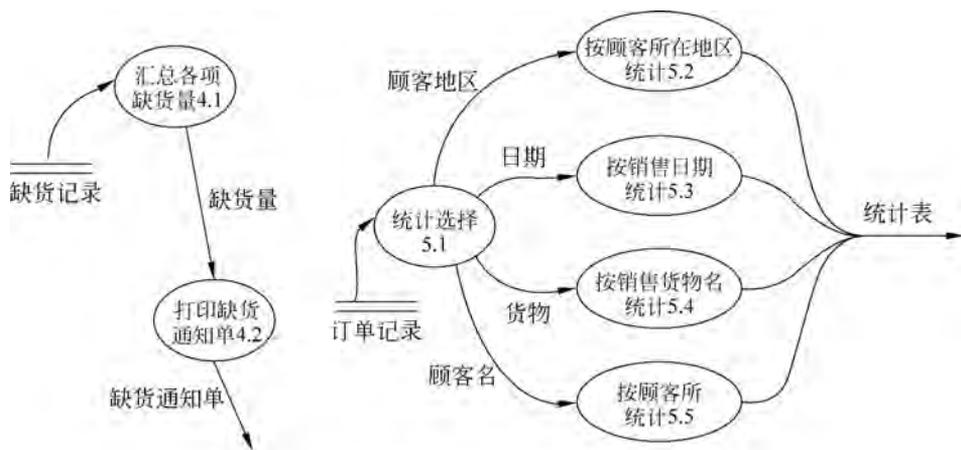


图 3-5 “销售管理系统”1 层图



(c) 1层图——处理缺货记录



(d) 1层图——缺货通知与订单统计

图 3-5 (续)

(5) 对数据流图和加工编号。

对于一个软件系统,其数据流图可能有许多层,每一层又有许多张图。为了区分不同的加工和不同的数据流子图,应该对每张图进行编号,以便于管理。

- ① 顶层图只有一张,图中的加工也只有一个,所以不必为其编号。
- ② 0层图只有一张,图中的加工号分别是 0,1.0 或者 2.0...
- ③ 子图就是父图中被分解的加工号。
- ④ 子图中的加工号由图号、圆点和序号组成,如 1.12、1.3 等。

(6) 注意事项。

① 命名。不论是数据流、数据存储还是加工,恰当的命名都可以使人们易于理解其含义。

② 画数据流而不是控制流。数据流反映系统“做什么”,不反映“如何做”,因此,箭头上的数据流名称只能是名词或名词短语,整个图中不反映加工的执行顺序。

③ 一般不画物质流。数据流反映能用计算机处理的数据,并不是实物,反映出此加工数据的来源与加工的结果。

④ 每个加工至少有一个输入数据流和一个输出数据流,反映出此加工数据的来源与加工的结果。

⑤ 编号。如果一张数据流图中的某个加工分解成另一张数据流图时,则上层图为父图,直接下层图为子图。子图及其所有的加工都应编号。

⑥ 父图与子图的平衡。子图的输入/输出数据流同父图相应加工的输入/输出数据流必须一致,此即父图与子图的平衡。

⑦ 局部数据存储。如果某层数据流图中的数据存储不是父图中相应加工的外部接口,而只是本图中某些加工之间的数据接口,则称这些数据存储为局部数据存储。

⑧ 提高数据流图的易懂性。注意合理分解,要把一个加工分解成几个功能相对独立的子加工,这样可减少加工之间输入/输出数据流的数目,增加数据流图的可理解性。

### 3.3.2 数据字典

数据字典是描述数据信息的集合,是对系统中使用的所有数据元素/数据流图中包含的所有元素的定义的集合;是为了描述在结构化分析过程中定义对象的内容时,使用的一种半形式化的工具;是在软件分析和设计的过程中,给程序设计与实现相关人员提供关于数据描述信息的方法。通过数据字典可以查询任何在项目中无法理解的各种数据,从而最大程度地消除开发人员或不同开发小组之间的歧义和交流不畅问题。

数据字典的组成如表 3-1 所示。

表 3-1 数据字典的组成

组 成	说 明
数据项	数据项是不可再分的数据单位。对数据项的描述通常包括以下内容:数据项描述={数据项名,含义说明,别名,数据类型,长度,取值范围,取值含义,与其他数据项的逻辑关系},若干数据项可以组成一个数据结构
数据结构	数据结构反映了数据之间的组合关系。一个数据结构可以由若干数据项组成,也可以由若干数据结构组成,或由若干数据项和数据结构混合组成
数据流	数据流是数据结构在系统内传输的路径。对数据流的描述通常包括以下内容:数据流描述={数据流名,含义说明,数据流来源,数据流去向,组成:{数据结构},平均流量,高峰期流量}
数据存储	数据存储是数据结构停留或保存的地方,也是数据流的来源和去向。对数据存储的描述通常包括以下内容:数据存储描述={数据存储名,含义说明,编号,流入的数据流,流出的数据流组成:{数据结构},数据量,存取方式}
处理过程	数据字典中只需要描述处理过程的说明性信息,通常包括以下内容: 处理过程描述={处理过程名,含义说明,输入:{数据流},输出:{数据流},处理:{简要说明}}

### 3.3.3 实体-关系图

#### 1. 实体-关系图概念

实体-关系图(Entity-Relationship Diagram, E-R 图)是指提供了表示实体、属性和关系的方法,用来描述现实世界的概念模型。E-R 方法是实体-关系方法(Entity-Relationship Approach)的简称,是描述现实世界概念结构模型的有效方法。

通常使用 E-R 图来建立数据模型,可把用 E-R 图描绘的数据模型称为 E-R 模型。E-R 图中包含实体(即数据对象)、关系和属性 3 种基本成分,通常用矩形框代表实体,用连接相

关实体的菱形框表示关系,用椭圆形或圆角矩形表示实体(或关系)的属性,并用直线把实体(或关系)与其属性连接起来。

人们通常就是用实体、关系和属性这3个概念来理解现实问题的,因此,E-R模型比较接近人的习惯思维方式。此外,E-R模型使用简单的图形符号表达系统分析员对问题域的理解,不熟悉计算机技术的用户也能理解它,因此,E-R模型可以作为用户与分析员之间有效的交流工具。

#### (1) 实体(Entity)。

具有相同属性的实体具有相同的特征和性质,用实体名及其属性名集合来抽象和刻画同类实体;在E-R图中用矩形表示,矩形框内写明实体名,如学生张三丰、学生李寻欢都是实体。如果是弱实体的话,在矩形外面再套实线矩形。

#### (2) 属性(Attribute)。

实体所具有的某一特性,一个实体可由若干个属性来刻画。在E-R图中用椭圆形表示,并用无向边将其与相应的实体连接起来,如学生的姓名、学号、性别都是属性。如果是多值属性的话,在椭圆形外面再套实线椭圆。如果是派生属性,则用虚线椭圆表示。

#### (3) 关系(Relationship)。

数据对象彼此之间相互连接的方式称为关系,也称为联系。联系可分为以下3种类型。

##### ① 一对一联系(1:1)。

例如,一个部门有一个经理,而每个经理只在一个部门任职,则部门与经理的联系是一对一的。

##### ② 一对多联系(1:N)。

例如,某校教师与课程之间存在一对多的联系“教”,即每位教师可以教多门课程,但是每门课程只能由一位教师来教。

##### ③ 多对多联系(M:N)。

例如,学生与课程间的联系(“学”)是多对多的,即一个学生可以学多门课程,而每门课程可以有多个学生来学。联系也可能有属性。例如,学生“学”某门课程所取得的成绩,既不是学生的属性,也不是课程的属性。由于“成绩”既依赖于某特定的学生又依赖于某门特定的课程,所以它是学生与课程之间的联系“学”的属性。

## 2. 案例:图书借阅管理系统

数据库要求提供以下服务。

(1) 可随时查询书库中现有书籍的品种、数量与存放位置。所有各类书籍均可由书号唯一标识。

(2) 可随时查询书籍借还情况,包括借书人单位、姓名、借书证号、借书日期和还书日期。

(3) 约定任何人可借多种书,任何一种书可为多个人所借,借书证号具有唯一性。

(4) 当需要时,可通过数据库中保存的出版社的电报编号、电话、邮编及地址等信息向相应出版社增购有关书籍。约定:一个出版社可出版多种书籍,同一本书仅为一个出版社

出版,出版社名具有唯一性。

满足上述需求的 E-R 图如图 3-6 所示。

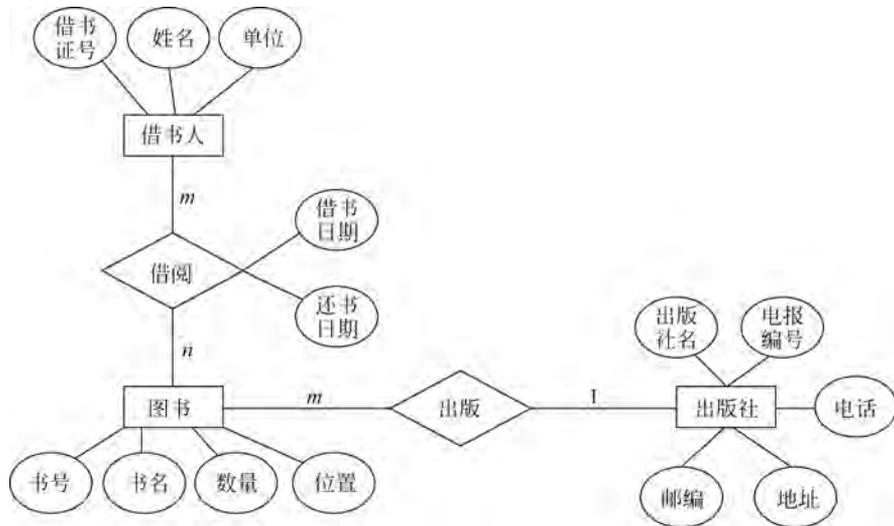


图 3-6 图书馆借书系统 E-R 图

转换为等价的关系模式结构如下。

借书人(借书证号,姓名,单位)

图书(书号,书名,数量,位置,出版社名)

出版社(出版社名,电报编号,电话,邮编,地址)

借阅(借书证号,书号,借书日期,还书日期)

### 3.3.4 层次方框图

层次方框图是一种用多层次的矩形树状结构描述数据的层次结构。树状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。

例如,某计算机公司全部产品的数据结构如图 3-7 所示。这家公司的产品由硬件、软件和服务三类产品组成,硬件产品分为处理机、存储器和外围设置,软件产品分为系统软件和应用软件,服务产品分为软件服务、硬件维修和培训等。

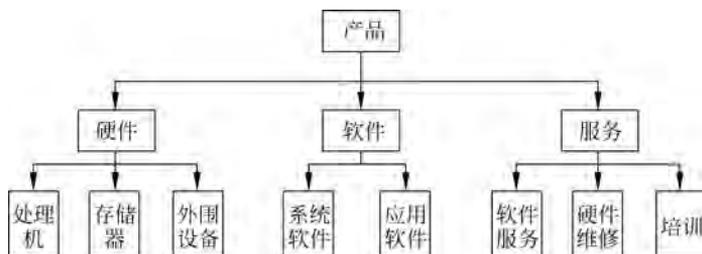


图 3-7 计算机公司产品的数据结构

随着结构的精细化,层次方框图对数据结构也描述得越来越详细,这种模式非常适合需求分析阶段的需要。系统分析员从对顶层信息的分类开始,沿图中每条路径反复细化,直到确定数据结构的全部细节为止。

### 3.3.5 Warnier 图

Warnier 图的作用和层次方框图的作用基本相同,也用树状结构来描绘数据结构。只不过 Warnier 图的描述手段更多,它还能指出某一类数据或某一数据元素重复出现的次数,并能指明某一特定数据在某一类数据中是否是有条件的出现。在进行软件设计时,从 Warnier 图入手,能够很容易转换成软件的设计描述。

图 3-8 是用 Warnier 图描绘软件产品的例子,并说明了这种图形工具的用法。图中的花括号用来区分数据结构的层次,在一个花括号中的所有名字都属于同一类信息。例如,操作系统、编译程序和软件工具都在系统软件花括号中,都属于系统软件类,而编辑程序、测试驱动程序和设计辅助工具都在软件工具花括号中,都属于软件工具类;异或信息表明一类信息或者一个数据元素在一定条件下才出现,而且在这个符号上、下方的两个名字所代表的信息只能出现一个。例如,一个软件产品可以是系统软件或应用软件,不可能既是系统软件又是应用软件;在一个名字下面(或右边)的括号中的数字(P1、P2、P3、P4 和 P5)表明了这个名字所代表的信息类(或元素)在这个数据结构中出现的次数。

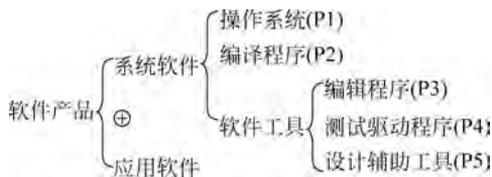


图 3-8 软件产品的 Warnier 图

### 3.3.6 IPO 图

IPO 图是输入加工输出(Input Process Output)图的简称。在系统的模块结构图形成过程中,产生了大量的模块,在进行详细设计时开发者应为每一个模块写一份说明。IPO 图就是用来说明每个模块的输入、输出数据和数据加工的重要工具。

IPO 图使用的基本符号少而简单,因此很容易掌握。它的基本形式是在左边的框中列出有关的输入数据,在中间的框中列出主要的处理,在右边的框中列出产生的输出数据。处理框中列出了处理的顺序,但是用这些基本符号还不足以精确描述执行处理的详细情况。图 3-9 是一个主文件更新的 IPO 图。



图 3-9 主文件更新的 IPO 图

## 3.4 实战案例——撰写机票预订系统需求分析报告

62



### “机票预订系统”项目的需求分析报告

#### 1. 引言

##### (1) 编写目的。

开发机票预订系统软件,能够适应当今社会并提高生产效率,使订票、售票、取票业务更加方便高效。该系统软件要易学易用,便于管理。

##### (2) 项目背景。

随着社会发展的不断进步和航空事业的壮大,以及人们消费水平逐渐提高,乘坐民航的消费者越来越多,机票预订系统也开始影响着人们日常生活和出行,并且变得越来越重要。而原有的系统随着航空公司载客量的迅猛增长和人们对便捷性要求的提高,已经无法满足需求。原有的系统不仅效率比较低,而且在安全性、准确性等方面有很多不足。

为了实现航空公司以及旅游行业的现代化管理,进一步提高工作效率,方便旅客,需要开发一个机票预订系统。该系统需要具有完整的存储、查询、核对、打印机票的功能。在这个系统中,旅客或工作人员通过机票预订系统查询,为旅客安排航班,打印取票通知和账单,旅客在飞机起飞的前一天凭取票通知和账单交款取票,系统校对无误即可打印机票给旅客。

##### (3) 参考资料。

国家标准文档(详见本章附件)。

#### 2. 任务概述

##### (1) 目标。

机票预订系统的总目标:在计算机网络、数据库和先进的开发平台上,利用现有的软件,配置一定的硬件,开发一个具有开放体系结构的、易扩充的、易维护的、具有良好人机交互界面的机票预订系统,实现航空公司的机票销售自动化的计算机系统,为企业的决策层提供准确、精细、迅速的机票销售信息。

本机票预订系统实现后能够大大提高航空公司的机票预订服务效率,降低售票服务中的错误发生率,减少信息交流的烦琐过程及其带来的开销。

##### (2) 用户特点。

使用本系统的最终用户可以定位为所有计算机使用者,尤其以旅游人员为主。需要软件系统操作简单,界面友好,对用户的教育水平和技术水平应该几乎没有任何要求,只要会用计算机进行常规操作的用户均可使用。

##### (3) 假定和约束。

普通管理员,只能对数据库(航班库和客户库)中的信息进行查询操作;系统维护人员,可以根据具体需要进行适当的数据管理(增、删、改、查)。

客户只能对航班信息库中的内容进行查询操作,客户进入页面之后在不进行登录的情

况下只能进行航班信息查询操作,预订机票必须先注册登录提交自己的基本信息;系统会根据管理员和客户的各种操作做出相应的返回信息提示。

### 3. 机票预订系统业务描述

(1) 系统业务流程图描述。

首先分析本系统总的业务流程图,如图 3-10 所示。机票预订系统的主要业务为订票业务、取票业务和退票业务,其业务流程图分别如图 3-11~图 3-13 所示。

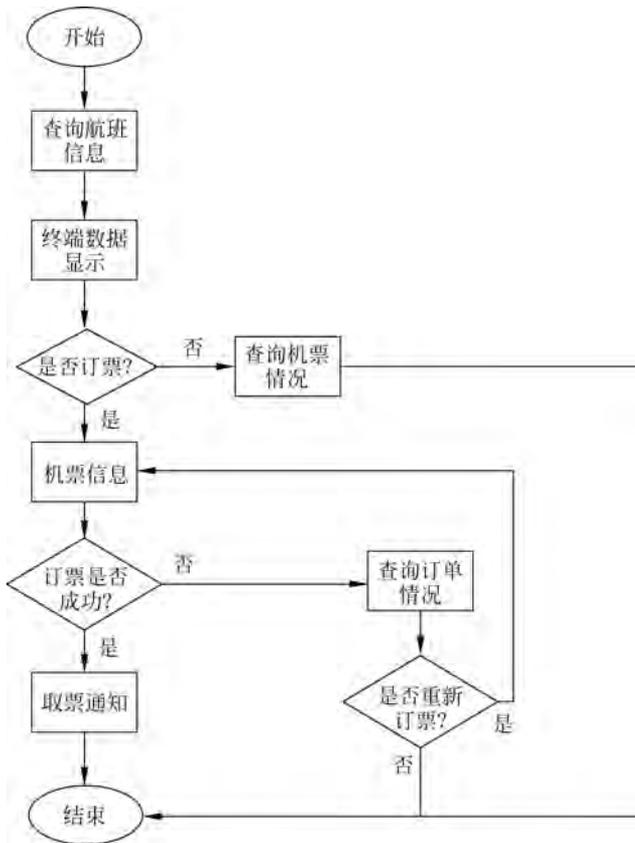
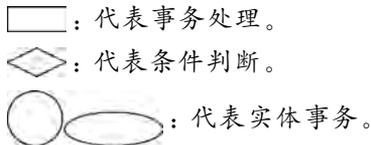


图 3-10 业务流程图

① 订票业务。根据旅客提出的要求(航班号、订票数额)查询该航班票额情况。若有余票,则为客户办理订票手续输出座位号;若已满员或余票少于订票额,则登记排队候补。

② 取票业务。根据取票通知书,打印机票,交给顾客。

③ 退票业务。根据客户提供的情况(日期、航班),为客户办理退票手续,然后查询该航班是否有人排队候补,首先询问排在首位的客户,若所退票额能满足他的要求,则为他办理订票手续,否则依次询问其他候补的客户。

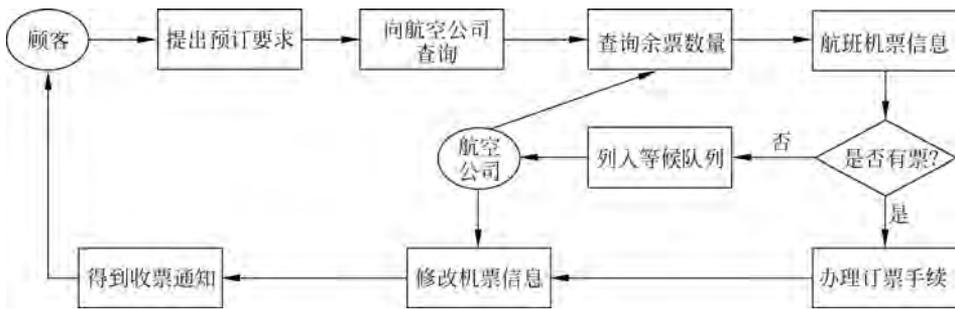


图 3-11 订票业务流程图

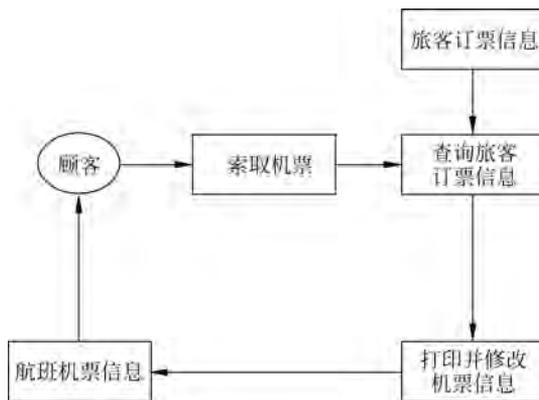


图 3-12 取票业务流程图

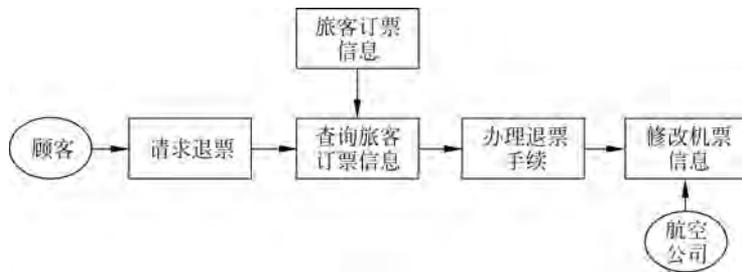


图 3-13 退票业务流程图

(2) 机票预订系统的数据需求。

机票预订系统的数据需求包括如下 3 点。

① 数据录入和处理的准确性和实时性。数据的输入是否准确是数据处理的前提,错误的输入会导致系统输出不正确和不可用,从而使系统的工作失去意义。在系统中,数据的输入往往是大量的,因此,系统要有一定的处理能力,保证迅速地处理数据。

② 数据的一致性与完整性。系统的数据可以是共享的,在不同的用户订票页面中,机票信息等资源数据都来源于同一个原始数据库,所以如何保证这些数据的一致性,是系统必须解决的问题。要解决这一问题,要有一定的人员维护数据的一致性,在数据录入处控制数据的去向,并且要求对数据库的数据完整性进行严格的约束。对于输入的数据,要为其定义

完整性规则,如果不能符合完整性约束,则系统应该拒绝该数据。

③ 数据的共享与独立性。整个机票预订系统的数据是被相关使用者共享的。然而,从系统开发的角度看,共享会给软件设计和软件调试带来困难。因此,应该提供灵活的软件配置,并可通过人工干预的手段进行系统数据的交换。这样,不仅能保障各子系统的独立运行,也能借此增强系统的健壮性。

(3) 机票预订系统数据流图。

首先分析系统总的流程图,如图 3-14 所示。整个系统的数据流程图比较复杂,本案例例仅以系统中的订票、取票两项业务为例,展开这两项业务的数据流图,分别如图 3-15 和图 3-16 所示。

- : 代表事务处理。
- : 代表实体事务。
- : 代表数据存储(如数据库)。
- : 代表数据流通。

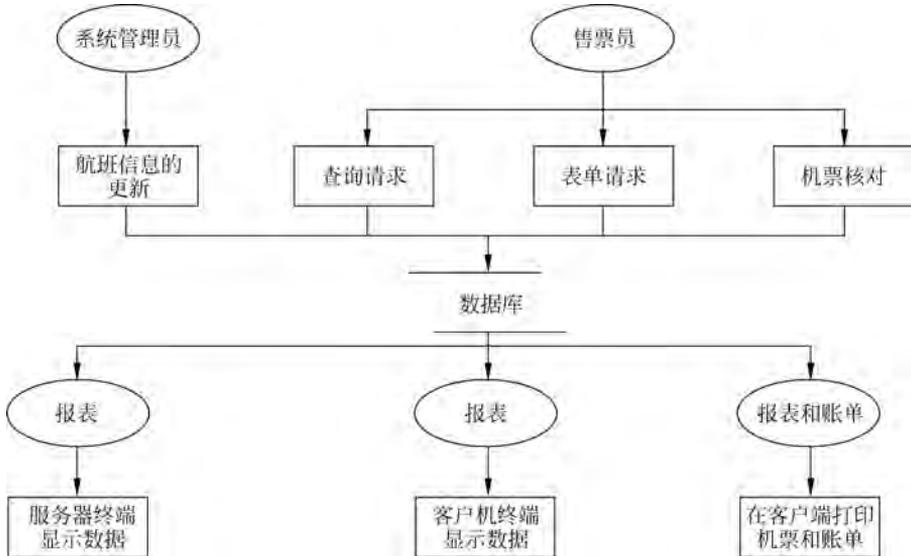


图 3-14 系统数据流图

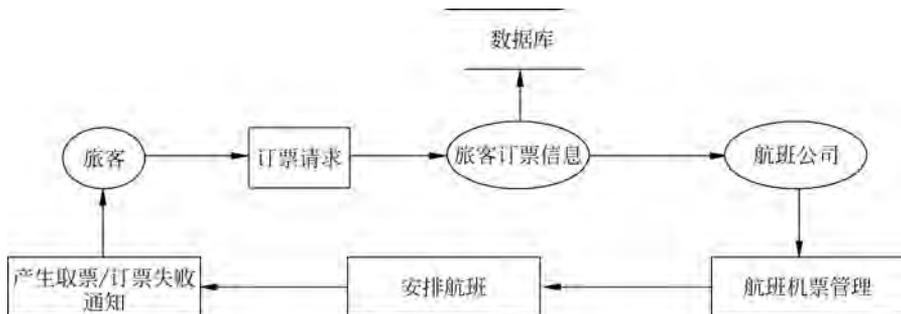


图 3-15 订票业务数据流图

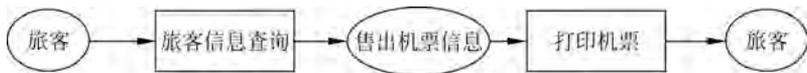


图 3-16 取票业务数据流图

(4) 机票预订系统数据字典。

数据字典是用来规范描述数据具体内容的工具,也是对数据汇总分析的一个总结。一般来说,可为每个数据建立一张二维表。在本系统中,需要分别为旅客信息、旅客订票信息、候补旅客信息、航班机票信息、取票通知和售出机票信息等建立数据字典,如图 3-17 所示。

名字:航班机票信息 别名: 描述:旅客的航班信息,根据旅客的旅行时间和目的地确定 定义:航班机票信息=旅客机票时间+旅客机票班次 位置:传输到旅行社端(Client端)
名字:取票通知 别名: 描述:旅客领取机票的凭证 定义:取票通知=旅客姓名+领票时间 位置:输出到打印机
名字:售出机票信息 别名: 描述:旅客的航班机票的信息 定义:售出机票信息=旅客的航班机票 位置:输出到打印机
名字:旅客订票记录 别名: 描述:已订票的旅客在航空公司的记录 定义:旅客订票记录=旅客姓名+航班机票信息 位置:输出到航空公司端(Server端)

图 3-17 数据字典示例

(5) 机票预订系统的逻辑模型。

系统的逻辑方案是指在对现行系统进行分析和优化的基础上,确定新系统的目标、信息流程、总体结构、功能模型以及拟采用的管理模型和信息处理方法等。机票预订系统的逻辑模型如图 3-18 所示。

4. 机票预订系统的功能要求

(1) 功能划分。

根据可行性研究的结果和客户的要求,分析现有情况及问题,采用 Client/Sever 结构,将机票预订系统划分为两个子系统:客户端子系统、服务器子系统。

(2) 功能描述。

下面分析各个子系统的功能需求。

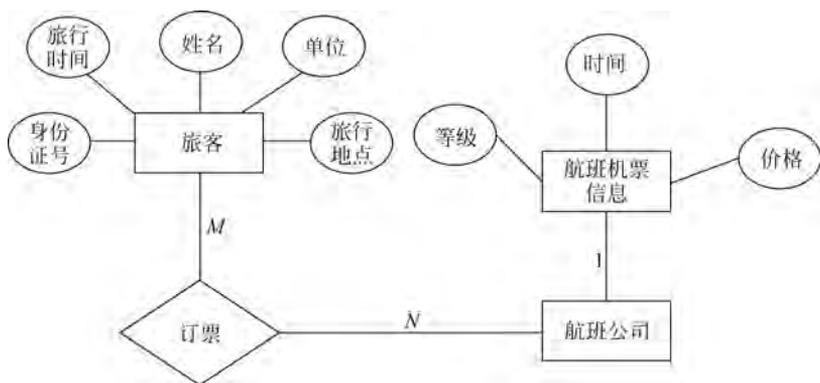


图 3-18 机票预订系统的逻辑模型

客户端子系统的功能要求可以分为以下 6 部分。

① 旅客信息的输入和统计：旅客输入订票信息，这部分功能是客户端子系统的基本部分，这个功能是以以后各个部分的基础。系统要求既能够从其他子系统中共享一部分信息，又有方便的操作界面供手工输入旅客信息。这部分要求对输入的数据进行简单的统计，供航空公司进行查询和宏观调控。

② 旅客信息的存储：将旅客的信息存储到数据库系统中，以备以后的取票确认以及查询。

③ 机票信息的传递及接收：将旅客所需的机票信息由客户端网络传到航空公司的服务器上，并且接收航空公司返回的航班信息，然后存储起来。

④ 取票通知及账单的生成和打印：把已存储的从航空公司返回的航班机票信息打印出来，连同生成的账单一起交给旅客。

⑤ 给已经订票的旅客打印机票：根据旅客的取票通知及账单，经确认无误，在接收旅客的付款后把机票打印出来交给旅客。

⑥ 机票销售情况的核算：这一功能是在上一功能的基础上，对机票销售额进行单项核算，得到销售情况并把核算结果作为企业报表输出。

服务器端的功能要求：通过计算机网络将客户端与服务器的数据库相连，将从客户端得到的信息进行处理，实现航班查询、机票生成、销售统计、综合信息查询等子系统。以计算机成本核算为中心，实现销售业务的计算机自动化，为航空公司降低成本、提高销售额和经营决策提供及时精确的依据。

在客户端系统的功能实现上，可以分为以下 6 部分。

① 接收由客户端发回的所需机票信息：通过网络接收机票信息并存入服务器的数据库中。

② 生成航班信息：根据所需机票信息(时间、地点)，在数据库中查询并得到正确的航班信息(价格、时间、等级)，分配所需的机票数并在数据库中做出已售出的标记。

③ 传递航班信息到客户端：把得到的航班信息通过网络传回客户端。

④ 接收反馈信息：对反馈信息进行分析，把已经售出的机票进行统计，对被旅客所退掉的机票要进行数据库的恢复。

⑤ 给已经订票的旅客打印机票：根据旅客的取票通知及账单，经确认无误，在接收旅

客的付款后把机票打印出来交给旅客。

⑥ 销售额的分析和管理的：要求包括对销售的机票进行分析，这一工作是在前面的基础上，以计算机为工具，对机票预订系统的功能和目标进行扩充。它以财务管理学为理论基础，以辅助决策为目标，以机票销售数据为中心，广泛采用统计学、运筹学等分析方法，对销售信息进行深层加工，建立反映不同航班需求的模型，提供管理所需财务信息。这一要求是机票预订系统的最高目标，在通过系统运行各种辅助决策信息和数据基础上实现这一目标。

### 5. 机票预订系统的性能要求

为了保证系统能够长期、安全、稳定、可靠、高效地运行，机票预订系统应该满足以下需求。

#### (1) 系统处理的准确性和及时性。

系统处理的准确性和及时性是系统的必要性能。在系统设计和开发过程中，要充分考虑系统当前和将来可能承受的工作量，使系统的处理能力和响应时间能够满足企业对信息处理的需求。

机票预订系统的查询功能对于整个系统的功能和性能完成举足轻重。作为系统的很多数据来源，机票数量和时间又影响企业的决策活动，其准确性很大程度上决定了机票预订系统的成败。在系统开发过程中，必须采用一定的方法保证系统的准确性。

#### (2) 系统的开放性和系统的可扩充性。

机票预订系统在开发过程中，应该充分考虑系统以后的可扩充性。例如，订票系统中订票方式的改变(网上订票)，用户查询的需求也会不断地更新和完善。所有这些都要求系统提供足够的手段进行功能的调整 and 扩充。而要实现这一点，应通过系统的开放性来完成，即系统应是一个开放系统，只要符合一定的规范，就可以简单地加入和减少系统的模块，配置系统的硬件。通过软件的修补、替换完成系统的升级和更新换代。

#### (3) 系统的易用性和易维护性。

机票预订系统是直接面对使用人员的，而使用人员往往对计算机并不是非常熟悉。这就要求系统能够提供良好的用户接口，易用的人机交互界面。要实现这一点，就要求系统尽量使用用户熟悉的术语并提供向导帮助的界面：针对用户可能出现的使用问题，要提供足够的在线帮助，缩短用户对系统熟悉的过程。

机票预订系统中涉及的数据是航空公司相当重要的信息，系统要提供方便的手段供系统维护人员进行数据的备份、日常的安全管理、系统意外崩溃时数据的恢复等工作。

#### (4) 系统的标准性。

系统在设计开发使用过程中涉及很多计算机硬件、软件，所有这些都符合主流国际、国家和行业标准。例如，在开发中使用的操作系统、网络系统、开发工具都必须符合通用标准，如规范的数据库操纵界面、作为业界标准的 TCP/IP 网络协议及 ISO 9002 标准所要求的质量规范等。同时，在自主开发本系统时，要进行良好的设计工作，制定行之有效的软件工程规范，保证代码的易读性、可操作性和可移植性。

#### (5) 系统的先进性。

目前计算系统的技术发展相当快，作为机票预订系统工程，应该保证系统在以后某个时期仍旧是先进的，在系统的生命周期中尽量做到系统的先进，充分完成企业信息处理的要求而不至于落后。这一方面可通过系统的开放性和可扩充性，不断完善系统的功能完成。另

一方面,在系统设计和开发的过程中,应在考虑成本的基础上尽量采用当前主流并先进且有良好发展前途的产品。

(6) 系统的响应速度。

机票预订系统在日常处理中的响应速度为秒级,达到实时要求,以及实时反馈信息。在进行统计分析时,原则是保证工作人员不会因为进度问题而影响工作效率。

## 6. 机票预订系统的运行要求

机票预订系统中的各个子系统的硬件和软件的配置如下。

(1) 服务器端子系统的运行要求。

系统软件: Windows NT Server。

数据库管理系统: SQL Server 2010。

硬件要求: Pentium II 800 以上,256MB RAM,40GB HD。

(2) 客户端子系统的运行要求。

系统软件: Windows NT Workstation 及以上版本。

数据库管理系统: SQL Server。

硬件要求: Pentium 450 以上,128MB RAM,10GB HD。



## 本章小结

本章首先介绍了需求分析的概念,将需求分析拆分为分析用户需求、建立需求原型、分析系统需求和进行需求验证等;接着,详细介绍了结构化分析建模,所谓建模,就是对问题所做的一种符号抽象,并对各个方法的图做了细致介绍;最后,以一个案例实战来帮助读者巩固知识。



## 知识拓展

需求分析师,是一个类似于技术翻译的工作。需求分析师们将公司业务部门所给予的客户需求进行业务规则、业务范围、业务流程等方面的技术分析后,把这些需求输出成开发工程师看得懂的语言,如常见的 UML、需求规格说明书等。然后在遵守这些基本项目流程要求的基础上,将需求通过软件工程师来得以实现,满足客户的需求。



## 体息一会儿

人工神经网络是一种应用类似于大脑神经突触连接的结构进行信息处理的数学模型。在工程与学术界也常直接简称为神经网络或类神经网络。神经网络是一种运算模型,由大量的结点(或称神经元)和结点之间的相互连接构成。每个结点代表一种特定的输出函数,称为激励函数(Activation Function)。每两个结点间的连接都代表一个对于通过该连接信号的加权值,称之为权重,这相当于人工神经网络的记忆。网络的输出则依网络的连接方式、权重值和激励函数的不同而不同。而网络自身通常都是对自然界某种算法或者函数的逼近,也可能是对一种逻辑策略的表达。

它的构筑理念是受到生物(人或其他动物)神经网络功能的运作启发而产生的。人工神经网络通常是通过一个基于数学统计学类型的学习方法(Learning Method)得以优化,所以人工神经网络也是数学统计学方法的一种实际应用,通过统计学的标准数学方法能够得到大量的可以用函数来表达的局部结构空间。另一方面,在人工智能学的人工感知领域,通过数学统计学的应用可以来做人工感知方面的决定问题(也就是说,通过统计学的方法,人工神经网络能够类似人一样具有简单的决定能力和简单的判断能力),这种方法比起正式的逻辑学推理演算更具有优势。

### 【本章附件】

以下为国标(GB/T 8567—2006)所规定的软件需求说明书内容要求。

## 软件需求说明书

### 1 引言

#### 1.1 编写目的

说明编写这份软件需求说明书的目的,指出预期的读者。

#### 1.2 背景

说明:

- a. 待开发的软件系统的名称。
- b. 本项目的任务开发者。
- c. 系统还使用 XX 提供的 XX 数据。

#### 1.3 定义

列出本文件中用到的专门术语的定义和外文首字母组词的原词组。

#### 1.4 参考资料

列出用得着的参考资料,如:

- a. 本项目的经核准的计划任务书或合同、上级机关的批文;
- b. 属于本项目的其他已发表的文件;
- c. 本文件中各处引用的文件、资料,包括所需用到的软件开发标准。

列出这些文件资料的标题、文件编号、发表日期和出版单位,说明能够得到这些文件资料的来源。

### 2 任务概述

#### 2.1 目标

本系统的开发意图、应用目标、作用范围以及其他应向读者说明的有关该开发的背景材料。解释被开发软件与其他有关软件之间的关系。如果本软件产品是一项独立的软件,而且全部内容自含,则说明这一点。如果所定义的产品是一个更大的系统的一个组成部分,则应说明本产品与该系统中其他各组成部分之间的关系,为此可使用一张方框图来说明该系统的组成和本产品同其他各部分的联系和接口。

#### 2.2 用户的特点

列出本软件的最终用户的特点,充分说明操作人员、维护人员的教育水平和技术专长,以及本软件的预期使用频度。这些是软件设计工作的重要约束。

#### 2.3 假定和约束

列出进行本软件开发工作的假定和约束,例如经费限制、开发期限等。

### 3 需求规定

#### 3.1 对功能的规定

用列表的方式(例如 IPO 表即输入、处理、输出表的形式),逐项定量和定性地叙述对软件所提出的功能要求,说明输入什么量、经怎样的处理、得到什么输出,说明软件应支持的终端数和应支持的并行操作的用户数。

#### 3.2 对性能的规定

##### 3.2.1 精度

说明对该软件的输入、输出数据精度的要求,可能包括传输过程中的精度。

##### 3.2.2 时间特性要求

说明对于该软件的时间特性要求,如:

- a. 响应时间、更新处理时间的要求;
- b. 数据的转换和传送时间的要求;
- c. 解题时间的要求等。

##### 3.2.3 灵活性

说明对该软件的灵活性的要求,即当需求发生某些变化时,该软件对这些变化的适应能力,并对于为了提供这些灵活性而专门设计的部分应该加以标明。如:

- a. 运行环境及操作方式上的变化;
- b. 精度和有效时限的变化;
- c. 同其他软件的接口的变化;
- d. 计划的变化或改进。

#### 3.3 输入/输出要求

解释各输入/输出数据类型,并逐项说明其媒体、格式、数值范围、精度等。对软件的数据输出及必须标明的控制输出量进行解释并举例,包括对硬拷贝报告(正常结果输出、状态输出及异常输出)以及图形或显示报告的描述。

#### 3.4 数据管理能力要求

说明需要管理的文卷和记录的个数、表和文卷的大小规模,要按可预见的增长对数据及其分量的存储要求做出估算。

#### 3.5 故障处理要求

列出可能的软件、硬件故障以及对各项性能而言所产生的后果和对故障处理的要求。

#### 3.6 其他专门要求

如用户单位对安全保密的要求,对使用方便的要求,对可维护性、可补充性、易读性、可靠性、运行环境可转换性的特殊要求等。

### 4 运行环境规定

#### 4.1 设备

列出运行该软件所需要的硬设备。说明其中的新型设备及其专门功能,包括:

- a. 处理器型号及内存容量;
- b. 外存容量、联机或脱机、媒体及其存储格式,设备的型号及数量;
- c. 输入及输出设备的型号和数量,联机或脱机;

d. 数据通信设备的型号和数量;

e. 功能键及其他专用硬件。

#### 4.2 支持软件

列出支持软件,包括要用到的操作系统、编译(或汇编)程序、测试支持软件等。

#### 4.3 接口

说明该软件同其他软件之间的接口、数据通信协议等。

#### 4.4 控制

说明控制该软件的方法和控制信号,并说明这些控制信号的来源。

### 【第3章网址】

