

## 需求体系结构:概念与模型\*)

Requirements Architecture: Its Concept and Model

8 | -85, 封四

周永林 潘云鹤

TP311.5

(浙江大学人工智能研究所 杭州 310027)

**Abstract** This paper introduces a new concept: solution model which differs from software requirements specification. It is problem-oriented, and is an abstract information-processing model. This paper also introduces a concept: requirements architecture, gives a modeling approach for requirements architecture and a case study. Requirements architecture as solution model is more attractive than object-oriented analysis model and is expected to overcome some drawbacks existing in object-oriented analysis.

**Keywords** Requirements, Solution model, Implementation model, Requirements architecture model (RAM)

## 1 需求分析:从面向产品到面向问题

九十年代,需求受到越来越多的重视,专门研究需求的领域——需求工程<sup>[1]</sup>成为研究热点。人们希望通过需求工程的研究更好地把握客户的要求,生产出更加令人满意的软件产品。然而事实是,需求工程虽然取得了一些成果但困难重重,需求工程的理论与实践也有相当大的距离<sup>[2,3,4]</sup>。我们认为,出现这种状况的根本原因,主要由于“需求”概念本身存在缺陷。本文提出了一种面向问题的需求概念:需求体系结构,通过实例给出了需求体系结构模型的构造方法。

“需求”是面向产品和客户的,它注重目标产品的功能和产品的一些细节,然而,决定产品功能和产品实现的因素非常多,包括主客观两方面的因素(如领域知识、软件实现技术以及人的主观要求),每一因素的变化都会引起需求的变化。可以说,需求具有无限的可变性。

尽管引起需求变化的因素很多,但大致可以按可变性大小划分为三类:①问题的客观状况和总体解决方案;②软件实现技术;③对界面、操作方式等

软件产品细节的要求。其中第三类因素可变性最大,在当时技术可实现的范围内,它完全取决于人们的主观要求;第二类因素的可变性取决于软件技术的发展速度;第一类因素可变性相对最小,它涉及软件应用领域,以及软件与软件应用环境之间的关系,它在很大程度上取决于当时的客观现实。如果能够把不同种类因素对需求的影响限制在软件过程的不同阶段,并表达在不同的模型之中,那么对需求的可变性将更易于把握。

## 1.1 面向问题的需求

许多需求工程的研究者和实践者们已经认识到了传统需求概念的不足,并提出了一些新的观点<sup>[1,3]</sup>,大致可以概括为下列几点:

- 系统可以使用标准构件库来建造,管理需求的核心是既要考虑对产品的预见性又要考虑问题情境(context)。

- 软件界已经意识到,有关需求的概念必须扩展,以把系统所处的环境包括在内。

- 面对应用领域不同特点和结构,需求要采取一种面向问题的方法。

- “一个面向现实世界的需求规范比只面向目标

\*)本项目得到国家自然科学基金,攀登计划,863智能计算机主题,国家教委博士点基金资助,周永林 博士生,主要研究领域为软件工程、信息与决策支持系统、管理学,潘云鹤 院士、校长,主要研究领域为智能CAD、认知科学、信息与决策支持系统。

系统的需求规范更有效和更具表达力”，“要有一种新的需求规范，它比现在的功能规范更加面向世界，我们希望把这种需求规范看成是建立了一个世界模型”<sup>[1]</sup>。

·不可能获得一个完全的需求，需求不完全性的真正挑战是如何决定不完全性的种类和水平，且这种不完全性是开发者可以承受的。我们需要能确定适当界限的技术和工具——哪些东西必须在需求中解决，哪些可以留到开发的后期阶段解决。

在信息系统(IS)研究领域，一个基本的问题是如何确定信息需求。J. Iivari 和 R. Hirschheim 区分了关于信息需求的三种不同观点<sup>[2]</sup>：客观论，主观论和共观论(intersubjective)。客观论者强调信息需求的非个人性如组织、用户任务或需要通过软件系统加以模型化的那部分客观现实，主观论者强调信息需求完全由用户个人特性决定。一个值得注意的趋势是，越来越强调组织的重要作用。一些研究者希望通过分析组织结构和行为主体来建立一种社会-技术(Socio-technical)系统<sup>[3]</sup>；另一方面，由于组织(特别是企业)为适应激烈的市场竞争而不断进行重组，这需要软件也能适应这种变化，为此软件设计必须采取一种基于组织结构的全局观点，并采用基于模型的方法<sup>[4]</sup>；此外，现代 IS 不再是某种单一的数据处理工具，而要求 IS 能“无缝”嵌入组织并提供全面信息、通讯、控制、协同工作的一体化系统。这需要对组织结构有更深入的了解，而且软件结构要在一定程度上能够与组织结构相适应。

软件开发的实践也表明，软件失败常常是由于对需要解决的问题理解不够，解决方案与问题不匹配或从设计到实现没能很好地过渡<sup>[10]</sup>。

所有这些都表明，需求应采取一种客观论和面向问题(世界)的观点，而且这种观点应贯穿于整个软件过程。也就是说，要想通过软件来解决现实问题，必须对现实问题有深入的理解。深入理解问题自然是为了更好地了解客户对软件产品的需求，但更重要的是，它可以为软件未来的演化提供一个较为稳定的框架，因为在引起需求可变性的三类因素中，问题的客观状况和总体解决方案是可变性最小的一类。对现实问题的理解，不能停留在自然语言的描述上，而要建立更精确描述问题客观状况和总体解决方案的模型。至于引起需求变化的第二、第三类因素可以在软件实现时再行考虑。

## 1.2 面向问题域的需求：求解模型

软件工程是要研究如何为现实问题提供一个自

动化的解决方案并加以实施和维护，软件过程本质上是一个问题求解的过程<sup>[11]</sup>。这一过程涉及两个不同的领域：一是问题域，即现实问题所处的领域(软件应用领域)；二是实现域，即建造软件所涉及的领域(计算域)。这是两个完全不同的领域，它们的不同点体现在两个方面。第一，概念模式不同，问题域使用的是客观世界中的概念如人、组织、设备、使用中的软件等，而实现域使用的是软件世界的概念模式如模块、数据结构、对象等；第二，可变性不同，问题域涉及客观事物，变化相对较小，而人们对软件产品行为的需求具有无限可变性，产品的技术实现具有多样性。

软件需求规范(SRS)及据此完成的各种设计模型如模块结构图、程序流程图等都是属于实现域的模型，称为实现模型。传统需求分析从一开始就要建立一个完全、一致和无二义的软件实现模型，这是相当困难的，即使建立起来，据此生产出的产品往往又不能满足客户真正的需要。因此，需求分析的目的不应是获得 SRS，而应是一个更加面向问题域的、从总体上把握问题解决方案的抽象模型：求解模型。求解模型是一种面向现实世界的模型，它并不关注软件实现方式，而是着重问题域的理解，着重描述问题的结构特征、问题的总体解决方案以及问题结构如何决定对功能的需要，用以表达决定需求的第一类因素。实现模型涉及软件实现方式，用以表达决定需求的第二、三类因素。图1表示了求解模型与实现模型在软件过程中的作用。

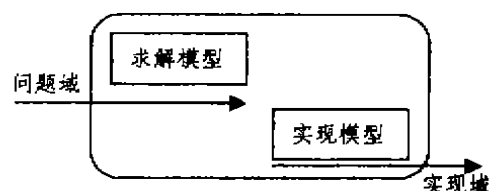


图1 软件过程中的求解模型与实现模型

把问题域与实现域分开，并分别建立模型，有望获得下列好处：①把引起需求变化的不同因素表达在不同的模型之中，克服需求多变性给软件生产带来的困难；②求解模型为实现模型的建立提供了一个较为稳定的框架，而且，注重问题域结构特征的表达为新的软件生产观念和软件生产方式(如软件体系结构的设计、基于构件的软件生产以及软件的复用和演化)提供了概念基础。

传统的软件过程没有严格区分问题域与实现

域。面向产品的“需求”，迫使人们在分析阶段就作出某些设计上的决策和选择，把问题域知识混杂到实现域的实现细节之中。此外，“功能决定论”使人们把主要精力放在软件功能的实现上，对软件结构考虑较少，所生产出的产品往往缺乏灵活性和适应性，导致了需求多变性与目标软件系统封闭性之间难以克服的矛盾。

## 2 求解模型的表示：需求体系结构模型

求解模型是一个面向问题域的模型。它注重问题域的理解，表达问题的结构特征和问题的总体解决方案。建立求解模型的过程本质上是对问题的理解不断深化的过程。求解模型要表达问题的结构特征和总体解决方案，并试图最终把这些信息反映到实现模型和软件产品之中，以便建立一种能容易地适应环境变化的“柔性系统”，“柔性系统”要求系统的功能分布要与环境相一致，即与环境有一定的对应关系。要做到这一点，软件结构必须映射一部分环境结构，即软件系统中的构件必须有一部分与环境中的实体有某种对应关系。这样，环境变化只需修改与之对应的软件构件即可实现软件系统的演化，软件也就具有了更大的适应环境的能力。这要求我们在软件工程中从“黑箱方法”转向“白箱方法”，从“功能决定论”转向“结构决定论”，即在软件过程中，要充分考虑问题域的结构特征，并把这种结构信息贯穿于软件生产的全过程。

面向对象分析(OOA)被认为是一种完全不同于功能分析的分析范式，有望克服结构化分析的不足<sup>[2]</sup>。然而，到目前为止已经提出了许多各不相同的 OOA，它们之间存在较大的差异，还没有一个统一公认的建模方法和语言<sup>[3]</sup>。这些 OOA 的共同弱点在于：

- 在对象的层次上不能很好地定义功能特性<sup>[4]</sup>。

- OOA 将客观世界看成“刚性的”对象，不能组织和抽象出更高层次的构建单元，对于复杂的系统，对象之间复杂的静态(数据)和动态(行为)关系使人难以把握问题的实质和总体结构。

- OOA 到 OOD 仍难以平滑过渡<sup>[4]</sup>。

因此，OOA 及其表示作为求解模型并不令人满意。为了克服 OOA 的不足，我们提出了一个面向 Agent 的分析和建模方法(AOA)<sup>[15]</sup>。AOA 分析的结

果是需求体系结构模型(RAM)。RAM 作为一种求解模型，有望克服 OOA 中的一些不足，RAM 认为，现实世界中存在着各种各样的实体，其中有些实体具有主动的行为能力，有些则没有。前者抽象为 Agent，后者抽象为对象，Agent 和对象都是封装的实体，通过接口和联结与外界建立静态的交互通道，通过事件和活动进行动态交互。RAM 包括：实体结构图；定义 Agent 及对象之间静态结构关系；Agent 状态图；定义每一 Agent 的动态行为模式；约束。定义与问题总体解决方案有关的限制和要求如安全性等。

下面简要描述需求体系结构模型的构成。

### 2.1 实体结构图

Agent 是行为主体，在一定状态下，它能对某些外界事件作出响应，即能够从事某些活动、进行状态转换或产生新的事件。生物体、人类的各种组织(如企业组织、政府组织)、控制系统中的控制器、软件系统等都可以看成是具有主动行为能力的实体。对象是指那些具有一定属性、没有主动行为能力的实体，这类实体是被操作者或被处理者。一般地，对象由三部分组成：对象名、对象体和触发器。对象名唯一地标识一个对象。对象体是一个对象所封装的所有数据的集合，它可以是属性值的集合，对象的集合，也可能是多个数据库的集合，对象之间可能具有某种关系如组装、分类等，也可能完全没有关系，无论它们之间有没有关系，都可以封装在一个单一的对象中。触发器定义对象的哪些状态变化会引起事件的产生。触发器是可选择的部分，它用于描述事物重要的状态变化，如闹钟在设定的时间可以响铃，水在冰点结成冰等。在图形表示中，Agent 用椭圆表示，对象用矩形表示。

实体结构图定义 Agent 及对象之间静态通讯通道的联结关系。Agent 或对象都是封装的实体，通过接口与外界联结起来。接口只表明了实体与环境交互的通道，并没有指明与环境中的哪个或哪些实体交互。因此，仅仅定义了实体以及实体的接口是不够的，还应定义实体之间如何通过联结建立起通讯通道。联结是实体之间交互的通道。Agent 或对象所要发送的事件或施加的活动首先通过自己的接口，然后通过这些联结通道传送到与之联结的 Agent 或对象。

Agent 与 Agent 之间可通过事件联结或函数联

结建立通讯通道。事件联结是一种单向通道, Agent 发出事件后,继续从事自己的任务,至于这一事件能不能通过事件接口传出去,事件联结将它传给谁,事件的产生者是不关心的,因此,事件联结是一种异步通讯通道。函数联结是一种双向通道, Agent 发出事件(函数调用)后,处于等待状态,直到这一函数调用返回结果,才继续自己的任务,因此,这是一种同步通讯通道。函数联结可以看成是一类特殊的事件联结。

Agent 与对象之间通过事件联结或活动联结建立联系。对象状态的变化,可能会引起触发器产生事件,这一事件可以通过事件联结传递给相关的 Agent,活动联结把活动的施者(Agent)和活动的受者(对象)联系起来。

图2表示了作为 Agent 的“银行”和“客户”,以及作为对象的“服务指南”之间的静态结构关系。其中带阴影的椭圆表示 Agent 类,具有同样接口和行为模式的一组 Agent;“银行帐户”封装在“银行”之内,表示只有封装它的 Agent(银行)能对它施加活动,其他外部 Agent(客户)不能直接对“银行帐户”进行操作,只能通过与“银行”的联结来获得有关“银行帐户”的信息。银行与客户之间有两个联结:一个是事件联结,一个是函数联结。事件联结“@寄对帐单(客户地址,对帐单)”定义了银行与客户群之间的通讯通道,银行所产生的能与“@寄对帐单(客户地址,对帐单)”匹配的事件将通过这一通道直接发往客户。函数联结“@查询帐户(帐号,密码) return(帐户信息)”建立了客户查询通道,客户可以通过这一通道查询帐户并得到返回结果。银行还可通过活动联结修改或查阅“服务指南”(双向),客户只能查阅“服务指南”(单向)。“服务指南”是一个不带触发器的对象。

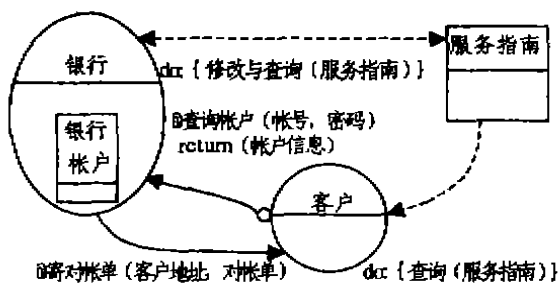


图2 通过联结而成的实体结构图

## 2.2 状态图

• 84 •

一个 RAM 由若干并行、分布的 Agent 和对象组成,模型的整体行为由每一 Agent 的行为以及 Agent 之间的交互决定。因此,有必要对每一 Agent 的动态行为模式加以描述,我们采用状态图来定义 Agent 的动态行为模式。状态图是 D. Harel 在 1987 年提出的一种图形语言,用以描述复杂系统的行为。我们在 D. Harel 的基础上<sup>[17]</sup>,结合 J. Rumbaugh 等的表示<sup>[18]</sup>和我们的需要,对状态图的表示作了一定的扩展。状态图用以描述 Agent 在哪些事件下,进行哪些状态转换,从事哪些活动以及要产生哪些事件。

### 2.3 约束

约束表达了问题解决方案所受到的无法通过其他方式体现的限制和要求,传统的分析方法把约束都叫作“非功能需求”。然而,有些约束与问题的总体解决方案有关,有些则与软件实现有关。RAM 只考虑与问题总体解决方案有关的约束,而这些约束中,又有一些可以在实体结构图和 Agent 状态图中反映,这里所说的约束是指那些与问题的总体解决方案有关,又不能在实体结构图和 Agent 状态图中反映的约束。

为了进一步说明我们的方法,我们引用 G. Booch 在文[19]中的一个问题。这个问题的描述如下:

“一些自由飘浮的海上浮标,向空中及过往航船提供导航与天气数据。浮标上有一系列传感器,通过它们收集气温、水温、风速以及浮标所在地点的数据。每一浮标上的风速与温度传感器数目不等,有些可改装成其他类型的传感器以备日后之需。每一传感器还装备了无线电发射机(用来发射天气与地点信息以及 SOS 消息)和一个无线电接收机(用来接收过往船只的请求),某些浮标上还配有红灯,过往船只寻找航向时可以打开它,如果航海者能到达这一浮标,他或她还可以用手打开浮标上的一个开关以发出 SOS 信号。每一浮标上的软件必须:

- 保持当时的风速、温度、和地点信息,风速每 30 秒读一次,温度与地点信息每 10 秒读一次;风速与温度值以动态平均值保存。

- 每 60 秒发送一次当时的风速、温度和地点信息。

- 根据过往船只的请求,发送过去 24 小时风速、温度、和地点信息;这一发送优先于定时发送。

- 根据过往船只的请求,打开或关闭红灯。
  - 当航海者打开紧急开关时,连续发送 SOS 信号;这一消息的发送优先于其他任何消息的发送并不停地发送直到有过往船只重新设置为止”。
- 这一实例的 RMA 如图3、图4、图5所示。

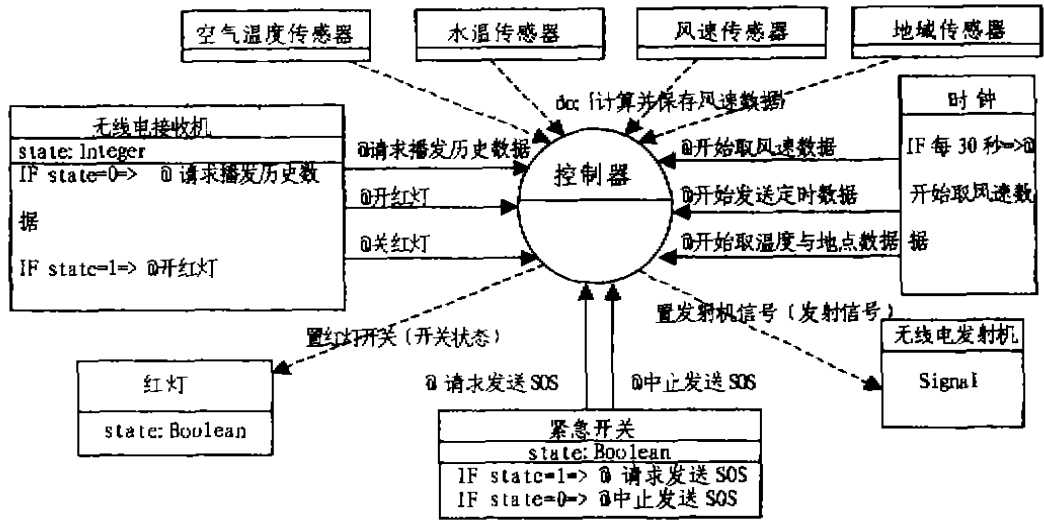


图3 实例的实体结构图

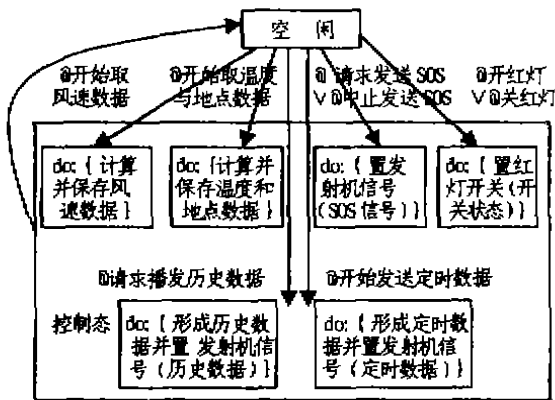


图4 控制器状态图

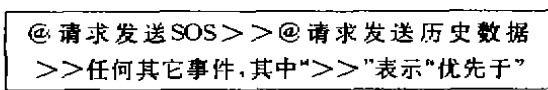


图5 实例中的约束

方式的出现也的确提出了这种要求。现有的面向对象分析作为求解模型还存在一定的不足,需求体系结构作为求解模型则能够克服这些不足。当然,任何一种方法和思路都有其适用的范围,AOA 分析方法和需求体系结构建模,特别适合于复杂系统(如并行分布系统)和人机混合系统的分析和建模,至于有些只涉及算法级的问题(如快速富立叶分析、数据加密软件),进行 AOA 分析是没有必要的。因为,一个算法一旦被发明或设计出来,一般不会有太多的变化,它的软件实现相对简单,人们只关心算法的功能和性能,至于算法在什么平台用什么语言实现,具有什么样的界面和软件结构等并不十分重要。人们可以不断改进算法,一般不会考虑维护或复用或演化实现某一算法的软件。大型软件(特别是大型应用软件)的情况正好相反,很多情况下,问题没有任何变化,但解决问题的软件要求(如界面、平台、操作方式)则可能有非常大的变化。大型软件是一种软件资产,需要不断地加以维护、复用或演化,才能保持或增加其价值。

**结束语** 求解模型的提出,有望克服传统需求概念的不足,改变传统的软件生产过程。事实上,面向对象技术、软件体系结构设计、基于体系结构的软件生产、软件的复用和演化等新的生产观念和生

随着面向 Agent 软件工程<sup>[20]</sup>,特别是面向 Agent 编程(AOP)的发展,我们希望 AOA 也能运用到算法级的问题。

(下转封四)

(上接第85页)

## 参 考 文 献

- 1 McDermud J A. Requirements analysis; Orthodoxy, fundamentalism and heresy. Requirements Engineering, Academic Press Ltd., 1994. 17~40
- 2 Siddiqi J. Challenging Universal Truths of Requirements Engineering. IEEE Software, March 1994
- 3 Siddiqi J, et al. Requirements Engineering the Emerging Wisdom. IEEE Software, March 1996
- 4 Pohl K. The Three Dimensions of Requirements Engineering: A Framework and Its Application. Information Systems, 1994, 19(3): 243~258
- 5 Hsia P, et al. Status Reports. Requirements Engineering. IEEE Software, 1993(November): 75~79
- 6 张家重, 徐家福. 需求工程研究新进展. 计算机研究与发展, 1998, 35(1)
- 7 Borgida, et al. Knowledge Representation as the Basis for Requirements Specifications. Computer, 1985 (Apr.), 82~91
- 8 Iivari J, et al. Analyzing Information Systems Development. Information Systems, 1996, 21(7): 551~572
- 9 David A. Taylor Business Engineering with Object Technology. John Wiley & Sons, Inc, 1995
- 10 Shaw M, Garlan D. Software Architecture. Perspectives on an Emerging Discipline. Prentice-Hall, Inc., 1996
- 11 Blum B I. Software Engineering. A Holistic View. Oxford University Press, New York, 1992
- 12 Coad P, Yourdon E. Object-oriented analysis. Yourdon Press, 1990
- 13 Iivari J. Object-orientation as structural, functional and behavioral modeling: a comparison of six methods for object-oriented analysis. Information and Software Technology, 1995, 37(3)
- 14 Hoydalsvik G M, Sindre G. On the Purpose of Object-oriented Analysis. ACM OOPSLA' 93. 240~255
- 15 周永林, 潘云鹤. 面向 agent 的分析与建模. 计算机研究与发展, 1999, 36(4): 410~416
- 16 Harel D. On Visual Formalisms. CACM, 1988, 31: 51
- 17 Harel D. Statecharts: A Visual Formalism For Complex Systems. Science of Computer Programming, 1987, 8: 231~274
- 18 Rumbaugh J, et al. Object-Oriented Modeling And Design. Prentice-Hall, Inc., 1995
- 19 Brooch G. Object-Oriented Development. IEEE Trans. on Software Engineering, 17(2)
- 20 樊晓聪, 等. 面向 agent 软件工程 (I) (II). 计算机科学, 1998, 25(3), 25(4)

---

## 计算机科学

(1974年1月创刊)

第26卷第8期 (月刊)

1999年8月25日出版

中国标准刊号: ISSN 1002-137X  
CN50-1075/TP

定价: 7.50元 国外定价: 5美元

邮发代号: 78-68

发行范围: 国内外公开

主管单位: 国家科学技术部

主办单位: 国家科技部西南信息中心

编辑出版: 《计算机科学》杂志社

重庆市渝中区胜利路132号 邮政编码: 400013

电话: (023) 63500828 传真: (023) 63502473

主 编: 朱宗元

印刷者: 国家科技部西南信息中心印刷厂

总发行处: 重 庆 市 邮 政 局

订购处: 全 国 各 地 邮 政 局

国外总发行: 中国国际图书贸易总公司 (北京399信箱)

国外代号: 6210M