

# 开发事务处理专家系统\*

## ——快速原型化方法\*\*

王国中

(郑州工学院数力系)

**摘 要:** 本文把人工智能、专家系统引入事务处理专家系统的开发。重点介绍了开发事务处理专家系统的快速原型化方法。最后,给出了开发事务处理专家系统的技术要点。

**关键词:** 人工智能, 专家系统, 快速原型化方法, 事务处理专家系统

当前,计算机应用的两大主流是:事务处理和工业过程控制。在事务处理领域,计算机的应用已经取得了很大成就,各种管理数据库、灵活多功能的电子报表处理和文字处理,以及丰富多彩的图形生成系统等,使人们从大量繁重的工作中解脱出来。但是,由于事务处理的复杂性、模糊性、不确定性、无结构性等因素,在事务决策、管理方面不能很好地模拟人类专家的思维决策,因而,现有的一些事务处理系统无论在功能还是性能上不尽人意。

把人工智能和知识工程用于开发事务处理软件、开发事务处理专家系统是日前国际、国内一个很有前途的方向。下面详细介绍开发事务处理专家系统的快速原型化方法,最后给出开发事务处理专家系统的技术要点。

## 1 事务处理和人工智能

事务处理专家系统是人工智能、传统的事务处理设施和过程性程序设计语言三者的紧密集成。可用下表来说明人工智能和事务处理的不同特点和不同方法。

一般事务处理系统存在下列缺陷:

- ①系统用户界面不友好,非专业人员使用系统非常困难。
- ②只能进行程序化决策,即对方案已知,目标非常明确并且可以定量判断进行决策。但是,对于非程序化决策,特别是中、高层决策人员的非程序化决策,系统无能为力。
- ③模糊性是非程序化决策的重要特点,而且是带有本质意义的特点,一般来说,事务

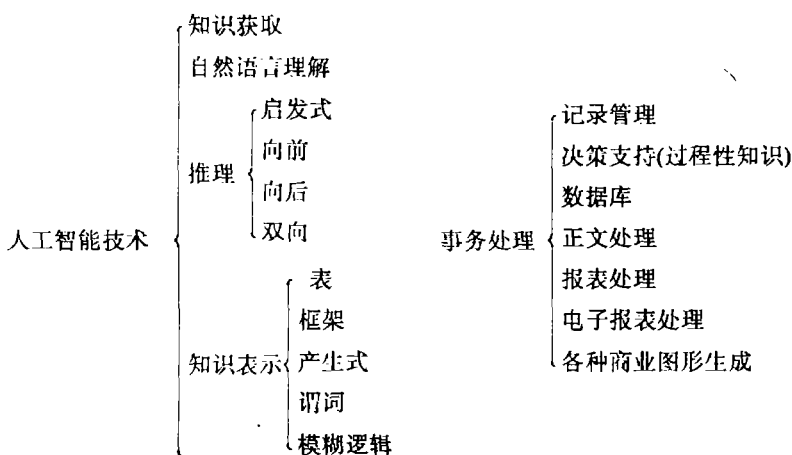
---

\* 管理专家系统

\*\* 收稿日期: 1990.04

处理系统无法处理这种情况,也不能处理都是用自然语言描述的难以定量确定的因素和目标问题。

④ 现有的管理系统,功能简单、水平不高,只能做简单的单项管理。



要让计算机代替人们进行事务处理,执行决策,就要模拟人们处理问题的推理过程,也就是总结人们中优秀代表处理同类问题的经验并在计算机中实现,这样的系统就是专家系统。专家系统是一个智能程序,它能对那些需要专家知识才能解决的应用难题提供专家水平的解答。而且专家系统可以集多个专家的知识 and 经验成一个系统,从而克服某个专家解决问题的主观性和片面性。在管理系统中引入人工智能、专家系统,能使系统应用广、效率高。提高管理决策的科学性。给社会带来更大的经济效益。

如何快速、高效、经济开发一个事务处理专家系统是摆在我们面前的一项艰巨而又重要的任务。现代化的开发技术和管理方法以及功能性强的开发环境对开发事务处理专家系统起到至关重要的作用。下一节详细介绍开发事务处理专家系统的快速原型化方法。

## 2 快速原型化方法

原型化方法 (PROTOTYPING) 是开发一个软件的快速有效的方法,它是在现有开发软件的基础上,找出适合用户需求的软件原件,快速开发成型,然后给予用户,根据用户需求,进一步修改、完善,直至用户满意。它要求集成化功能性强的软件开发环境和现代化的软件管理技术。

对于事务处理专家系统这种特殊的软件,在其开发过程中使用快速原型化方法,既容易又有效,因为专家系统的重要组成部分之一是规则集,而规则集是模块化的知识。开发者可以先针对问题的一个子集,快速生成一个小的规则集,来试验系统的可行性。同时,许多开发工具提供了各种推理机制(例如,PROLOG 语言提供了向后推理机制)开发者无需自己编写推理程序,只需选用开发工具中的有关功能即可。选用不同的推理策略可得不同的快速原型,经比较后决定在最终产品中究竟使用什么策略。

另外,各单位的决策者,对某个项目(软件)的功能、性能、周期、费用等不十分了

解。如果开发者一开始就拿出一个快速原型系统,形象地显示给决策者,就可促使决策者下决心投资开发。

图 1 是快速原型开发事务处理专家系统的总框图。

下面对图 1 中的工作流程逐一说明。

#### 系统分析与定义:

确定开发事务处理专家系统的总体目标,给出其系统的功能、性能、可靠性以及用户接口方面的设想。估计开发成本、资源需求,效益和进度。开发者应尽可能理解该专家研究领域及术语,今后的用户及咨询环境、开发工具、所支持的知识表示方式。开发者在进行充分调查研究、广泛征求用户和领域专家的基础上制订出详尽、准确的问题定义及功能说明书。

在初始的快速原型系统 1 的系统分析与定义阶段,可能不尽详细和准确,可以先进入下一阶段的开发工作,等快速原型系统 1 已经成型,再根据用户需求进入下一个快速原型系统的开发。直至开发出软件产品,详见图 1。

#### 设计与编码:

设计工作包括规则集的设计、用户接口设计、模块划分、模块间的接口设计、每一模块间的内部实行算法。在原型化 1 阶段,开发者可以针对具有代表性的实例设计规划。在此规则集的基础上,进行编码设计、测试、维护直至一个原型系统成型。

#### 测试与维护:

测试阶段包括专家测试和用户测试。系统的规则集经设计被编译成内部形式后,专家与开发人员可以在一起测试。测试时,实际可以用系统分析与定义时专家所给出的具有代表性的实例,若测试结果与专家给出的结果一致可进入开发的下一步,若不一致,可以通过推理和解释机制的自动跟踪找出原因所在,经调用编辑器重新修改规则集后,重新编译,再次测试。

用户测试主要是测试其用户接口是否友好,提出改进用户界面的方案。

软件维护是指在软件投入使用一段时间后,及时发现开发时期前几个阶段的遗留问题,及时修改错误;为适应环境的修改和扩充,这些维护工作可以在软件运行一段时间后,集中一些要修改的内容,统一进行修改。

在软件原型系统,各个阶段都有明确的工作内容,按时间顺序严格执行。前一阶段的任务未完成,决不能往下阶段执行。每一阶段的末尾要进行阶段复审,若发现问题必须暂停向下进行,返回,找出问题,并加以解决。图 1 的有向箭头就反映了这一过程。事务处理专家系统的研制过程是渐进的过程,使知识系统开始时能模拟人并接近人类的性能。

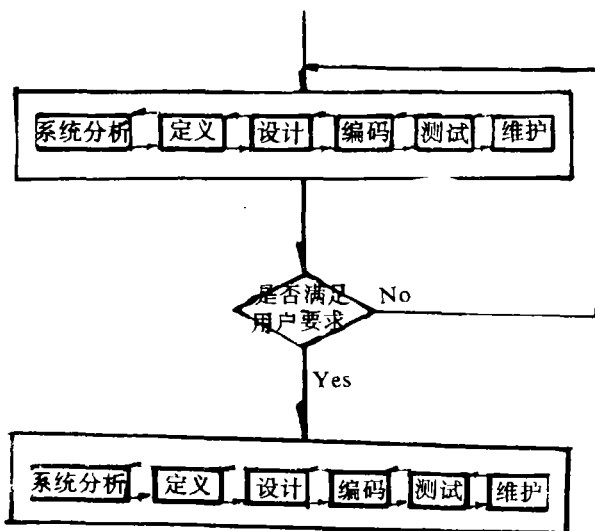


图 1

3 技术要点

上面介绍了快速开发事务处理专家系统的方法。科学的项目管理手段, 集成化功能性强

的开发环境, 是开发方法得以实现的保证。  
快速原型化事务处理专家系统的管理工作包括组织多层次的开发人员。一般, 在原型化系统, 2~3 人最为适宜, 这样可以减少不必要的通信, 提高开发效率。以后随着系统的进一步扩大, 根据项目需要, 可以增加不同层次的开发人员。项目管理工作还包括项目的计划、进度、质量、经费、配置等。

开发环境包括硬件环境和软件环境, 快速原型专家系统开发人员必须有适合项目开发需要的各种硬件资源。软件环境要求集成化的软件结构环境。在中国, 软件的汉化特别重要。所有软件间的通信都通过一个活动的数据字典, 见下图。

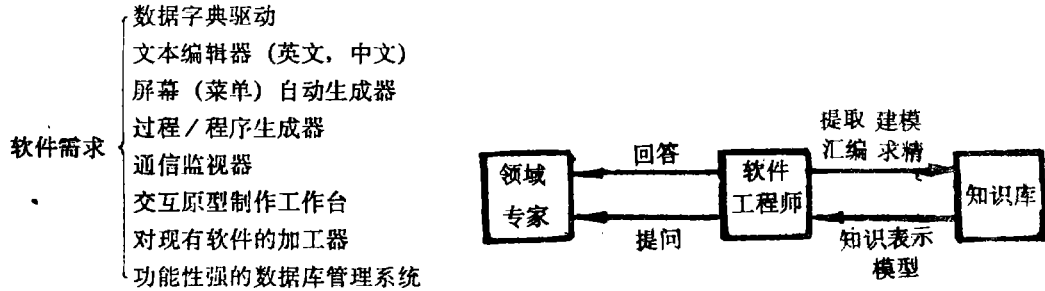


图 2

经验表明, 功能性强、集成化的软件环境使得事务处理专家系统的原型得以快速生成。

事务处理专家系统中丰富、正确的知识是系统应用广泛的一个重要因素, 而知识的获取是一个“瓶颈”问题。知识工程师要快速、正确地从事务处理领域专家那里获得知识并存储于计算机中, 必须采用正确的方法和手段。图 2 介绍一个知识获取模型。

知识工程师的主要任务见下表。

名称	解释	内容	方法
提取	知识获取	概念和规则	向专家提问, 或从书本
建模	知识系统设计	知识表示	形式逻辑, 语义网, 分层框架, 规划过程, 模糊逻辑
汇编	知识存贮	知识库和推理机	向前, 向后, 双向推理
求精	知识求精	已修改的概念和规则	运行系统, 与领域专家共同修改概念和规则

5 结束语

事务处理领域的知识往往是不确定的和模糊的, 利用模糊数学、置信度、概率统计等数学工具可以帮助工程师把事务处理中的模糊知识、不确定信息存储于知识库中, 事务处理专家系统可以方便地使用知识库中的知识解决实际问题, 有关详细内容在另一篇待发表

文章“事务处理专家系统的知识表示”中介绍。

### 参 考 文 献

- (1) Bonczek, R.H.; C.W.Holsapple; and A.B.Winston. Foundations of Decision Support Systems. New York, Academic Press, 1981
- (2) Guru Reference Manual, Vol.1. Lafayette, Ind. MDBS, Inc. 1985
- (3) 李卫华, 周祥和. IBM-PC机编译型PROLOG语言. 武汉大学出版社, 1987

## Development Management Expert System —Rapid Prototyping Methods

Wang Guozhong  
(ZhengZhou Institute of Technology)

**Abstract:** In this paper, we introduce expert system into management system. The paper discusses rapid prototyping methods to develop mangement expert system. At last, we give some techniques of developing management expert system.

**Keywords:** AI, rapid prototyping method, management expert system