

文章编号 :1001-2486(2001)02-0061-05

基于构件的决策支持系统中模型组织集成框架 IFMO^{*}

赵新昱, 陈文伟, 张维明, 陈卫东

(国防科技大学人文与管理学院, 湖南长沙 410073)

摘要 决策支持系统模型组织和表示是影响模型和建模效果的一个基本问题,也是实现模型管理的前提。在研究国内外建模技术以及开发 CS-DSSP 决策支持系统开发平台^{[1][2]}的基础上,本文提出了一种基于构件的模型组织集成框架,包括模型构件化描述、构件化框架、集成框架体系结构等。IFMO 提出了一种把先进建模技术应用到 DSS 中的方法。

关键词 构件 框架 决策支持系统 模型组织集成框架

中图分类号 :TP301 文献标识码 :A

IFMO : Integrative Framework for Model Organizations in Component Based Decision Support Systems

ZHAO Xin-yu, CHEN Wen-wei, ZHANG Wei-ming, CHEN Wei-dong

(College of Humanities and Management, National Univ. Of Defense Technology, Changsha 410073, China)

Abstract Model organization and description are a basic problem in decision supporting systems. At the same time, they are also the preconditions in model management. After studying the integrated modeling and design and development of CS-DSSP (Client/Server based decision supporting systems platform), an integrative framework for model organization, called IFMO is introduced, which includes component model description, model component framework, architecture of integrative framework etc. IFMO provides a method that implements a new modeling technique in DSS.

Key words component; framework; decision support systems; integrative framework for model organization

随着软件技术的发展和更新,计算机辅助决策的深度和范围也随之扩大,以模型技术为核心的决策支持技术系统研究不断得到充实和提高,特别是近几年,构件技术的发展扩大了对建立决策支持系统的需求。构件技术采用多层客户/服务器的计算模型,充分利用网络上的资源,达到模型的共享与复用,从而使单机、集中式的决策支持系统向网络、分布式系统转化。通过对构件模型的分布提高系统的性能,通过系统互连和互操作可以提高决策支持系统的协作能力,通过模块化技术可以提高系统的可伸缩性,通过构件模型共享和复用技术可以提高系统的性能价格比。基于以上考虑,模型构件化以及基于构件的模型组织越来越得到人们的重视。

1 构件与框架定义

构件是封装了它的设计和实现,而仅向外部提供接口的相对独立的可重用的软件单元^{[3][4]}。所谓接口是一套用于说明构件提供服务的操作集合,它着重于给定服务的行为而不是服务的实现,服务的实现由构件内部的实体具体完成。接口给出了一组操作的名字,操作用来完成一个系统或子系统的逻辑行为。

一个构件同一个小微型的应用程序类似,都是已经编译链接好的二进制代码,应用程序由多个这样的构件搭建而成。在需要对应用程序进行修改时,只需将构成此应用程序的某个构件用新版本替换即可,可见,构件在应用程序中完全是动态的。

构件的运行和管理需要操作系统的支持,构件是一种客户机/服务器方式的对象模型。当一个客户需要构件服务器对象的一些服务时,构件库核心服务负责生成该构件的服务器对象并且在客户对象和

* 收稿日期 2000-09-26
基金项目 国家自然科学基金资助项目(79670019)
作者简介 赵新昱(1973-)男,博士生。

服务器对象之间建立初始连接。此后,构件库就不再参与客户对象和服务器对象之间的工作了。这两个对象可以自由地进行直接通信。

框架提供对于相似问题的一种统一的解决方案,框架的最终目标是能够动态地组装构件,实现软件的“即插即用”。领域内的应用程序由两部分组成:不变的部分和可变的的部分。在同一领域内,不变的部分可以被许多程序共享,变化的部分是使一个应用程序在其领域内成为与众不同的那一部分。框架就是把握领域问题内全部问题集解决方案的不变的部分,领域问题开发人员利用框架,再向框架添加变化部分,进而完成所在领域内应用的开发。

在决策支持系统中,无论是传统的 SC 结构(Spragure 和 Carlson 提出的“对话、模型、数据”三部件结构)还是后来提出的以模型服务器实现模型部件的特定的 DSS 结构,模型部件都是极其重要的部件,包括模型管理系统和具体的模型,模型管理系统是不变的部分,模型随着开发具体的决策支持系统而相应地变化,是可变的的部分。模型管理系统一般分为模型信息管理、模型描述、模型的使用与控制、模型的集成等几个部分。以框架形式实现模型构件的静态与动态管理,以接口形式实现决策模型规范化服务,提高了系统的开发效率,增强了模型复用性,实现模型的共享和复用。由于构件支持分布式应用,构件技术和模型技术相结合,为建立分布式决策支持系统提供前提。

2 模型构件化描述

传统的决策模型有四种表示法^[6] 程序表示法、数据处理法、智能知识模型表示法和数学方程表示法。针对构件建模的特点和要求,根据融合程序表示法和知识表示法的思想,我们提出模型的构件化描述方法,它由四部分组成:说明部分、接口部分、输入输出规范化部分和实现部分,其结构如图 1 所示。其中,构件说明部分是二级抽象部分,是辅助决策者的视图,它采用非形式化、自然语言的形式对构件的特征做出更为详细、准确的描述,主要记录构件的版本说明、构件的开发环境、构件的运行环境、构件的使用环境、构件功能的自然语言描述、构件的可复用信息。这样形成了构件的文档,适合于决策人员理解。构造部分和参数部分是一级抽象部分,是模型使用者的视图,该部分给出模型的参数、模型构件的接口、模型的数据以及模型运行后结果输出等信息。在这一层上,决策开发人员可以利用已有的构件模型进行搭建决策支持系统。模型底层是具体模型实现部分,即按照一定的构件规范对具体构件模型进行编制,后两部分适合于编程人员使用和理解模型。

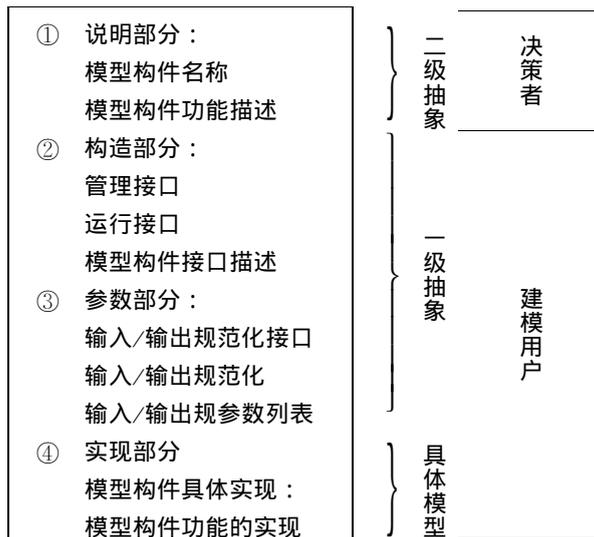


图 1 模型的构件化结构

Fig.1 Structure of model component

3 模型构件化框架

经过模型构件化描述,可以看出,构造部分和参数部分是决策模型共性部分,模型构件化框架主要思想是通过框架技术实现决策支持系统中模型的共性部分,提取各个模型的公共信息,对模型进行统一表达,以方便模型管理、运行、组合等。模型构件化框架在模型构件化的基础上,采用构件技术实现决策支持系统中模型操作的共性部分,包括两个方面的主要内容:一个方面为模型数据处理,另外一个方面为模型运行。

1) 模型数据处理

在模型规范化的基础上,利用构件技术实现对模型数据的处理。通过输入/输出规范化接口实现对模型数据描述文件^[8]的操作,这样,模型之间的组合运行也同样利用模型数据描述文件实现。模型构件化框架数据处理部分涉及到两个构件:数据库操作构件和模型规范化构件。数据库操作构件提供对数据库的操作,包括三个接口:数据库连接接口,数据库操作接口(插入、删除和修改)以及数据库查询接口。模型规范化构件包括三个接口:模型说明处理接口,模型数据处理接口以及数据关系处理接口。两个构件通过聚合实现二者之间的关系,如图2的模型构件化框架数据处理部分所示。

模型说明处理接口(IModelDsp)对模型名称、模型类型、模型描述等模型基本信息进行处理,模型数据处理接口(IModelData)主要对模型参数、模型输入/输出数据进行处理,对话数据处理接口(IDlgData)主要功能在模型运行时动态调整模型属性或数据。以上三个接口中包含的方法如图2所示:

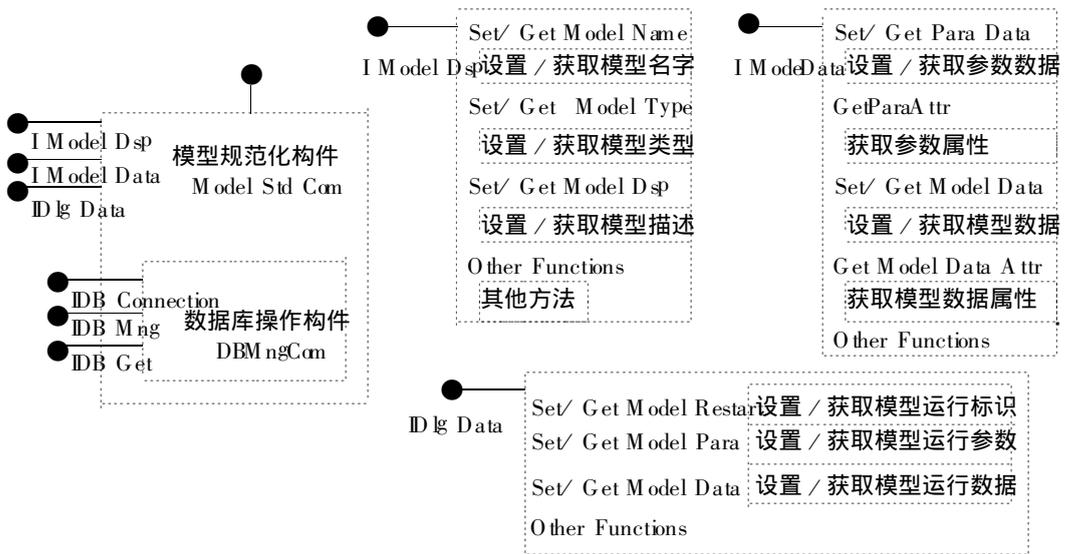


图2 模型构件化框架及接口说明

Fig.2 Model component framework and description of model standardization component interface

DBMngCOM 构件是对数据库操作构件 OLE DB 进行包装得到的,因为模型对数据库操作要求快速、简单、有效,所以对 OLE DB 接口进行简化处理得到三个接口 IDBConnection, IDBMsg 和 IDBGet。

2) 模型运行

模型构件化框架的另一面是模型的运行,模型的运行通过调用接口实现。一个模型具有若干接口,每个接口代表模型的某个属性和功能。其他模型或应用程序可以设置或调用这些属性和功能来进行特定的逻辑处理。模型和应用程序的连接是通过其接口实现的。负责集成的开发人员无需了解模型功能是如何实现的,只需创建模型对象并与其接口建立连接。在保证接口一致性的前提下,可以调换构件模型、更新版本,也可以把模型安插在不同的应用系统中。决策支持系统模型多种多样,但模型的运行只有几种类别,模型构件化后,模型的运行一般包括模型创建、表现、模型数据匹配和模型的运行。模型运行框架的主要功能就是在模型构件化的基础上,实现模型的动态管理,即通过模型动态管理构件运行模型,该构件包括三个接口:模型创建和表现接口,模型数据匹配接口和模型运行接口。其中模型运行接口是在编制每个具体模型时继承的,里面的细节在编制具体模型时实现,即每个模型都有 IModel-

Run 接口 , 其中具有方法 RunModel , 模型的运行就是调用该方法。这样 , 屏蔽模型的实现细节 , 保证在调用模型时有统一的标准。接口中包含的具体方法如图 3 所示 :

IModelCreate 接口实现模型的创建和表现 , IModelRun 接口实现模型的统一运行 , IDataMatch 接口实现模型数据类型匹配判断 , 主要有三种匹配关系 : 数据和参数匹配 , 数据和数据匹配以及参数和参数匹配。

决策支持系统中的模型由不变的部分和可变的部分组成。提取模型的共性信息 , 即模型不变的部分 , 通过模型构件化框架实现 ; 模型具体实现是可变的部分 , 根据具体模型的不同有不同的实现方法。在把模型应用到决策支持领域后 , 利用模型构件化框架 , 可以用一种统一的方式进行模型参数和数据的处理、模型的创建和运行。

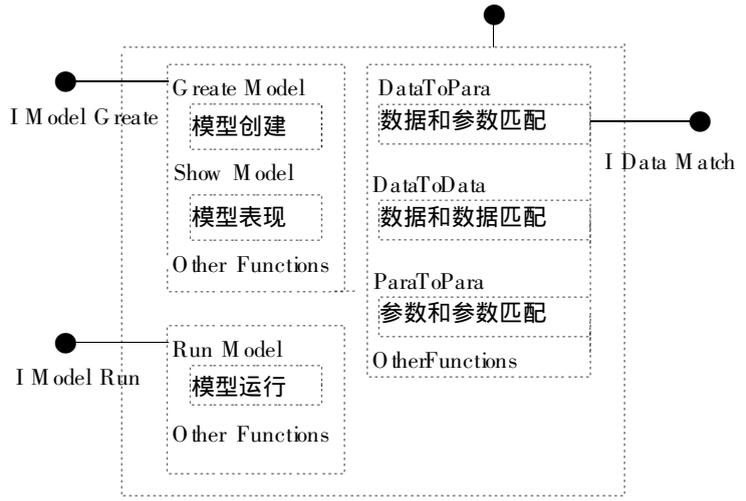


图 3 模型运行构件接口说明

Fig.3 Description of model running component iinterface

4 模型组织集成框架 IFMO

建立 DSS 系统 , 模型不论采取何种表达方式 , 都需要模型库和模型库管理系统的支持。基于构件的管理系统包括 : 模型构件的提取部分 , 能自动在操作系统中或网络中提取所需构件 ; 静态管理部分 , 对构件增加、删除、修改 ; 动态管理部分 , 对模型构件创建、运行和删除。在基于构件的决策支持系统中 , 模型组织集成框架 (Integrative Framework for Model Organizations : IFMO) 建立在模型构件化基础之上 , 构件管理层的模型库管理系统通过构件的方式来实现 , 这样 , 在网络上的每台计算机都可以共享管理层构件 , 应用管理层构件提供的接口进行搭建决策支持系统。

模型组织集成框架 (IFMO) 包括模型信息获取构件模型管理构件和构件库组成。

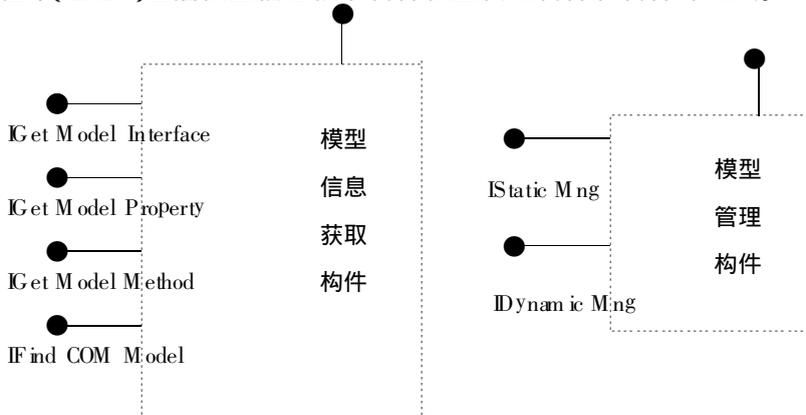


图 4 模型信息获取构件和模型管理构件接口说明

Fig.4 Description of model information pick-up component and model management component

模型信息获取构件由四个接口组成 (如图 4 所示)。获取构件模型接口 (IGet Model Interface) 主要功能是取得构件模型接口个数、判断接口名称、接口是否存在 , 获取构件模型属性 (IGet Model Property) 主要功能是获取模型属性个数 , 判断构件模型属性是否存在、构件属性名称以及属性类型 ; 获取构件模型方法 (IGet Model Method) , 主要功能是获取模型构件的方法个数 , 判断模型方法是否存在 , 获取方法名称、方法中参数个数、参数名字 , 参数类型等 ; 查找构件 (IFindCOMModel) , 主要功能是查找操作系统中登记构件 , 判断构件是否存在 , 获取构件唯一标识等信息。

模型管理构件由两个接口组成(如图4所示),分别是静态管理接口(IStatic Mng)和动态管理接口(I-Dynamic Mng)。模型的静态管理主要建立构件模型库、对模型的识别信息和描述信息进行统一的管理,包括模型的类别、标识、位置以及其他信息等。通过模型库和模型字典库结合注册表对模型进行有效的管理,利用构件实现模型管理,将模型信息与模型实现分离开,使模型管理方便、简单。模型静态管理接口主要功能是对模型的增加、删除、修改和查询,这些操作是针对模型库的。动态管理接口负责模型运行过程中属性的赋值和方法的调用。由于采用自动化接口实现模型构件,所以通过 Invoke 函数即实现。

构件库的实现主要借助于数据库技术,由于决策支持系统是建立在网络平台上,所以网络上的每个结点都有一套构件库管理系统。构件字典库存储模型信息,包括模型唯一标识、模型类别、模型关键字、模型功能描述、模型接口、模型属性、模型方法等信息。构件可以从注册表中得到模型唯一标识。通过模型信息获取构件、得到构件接口、属性和方法,利用模型管理构件对模型信息进行入库管理,方便查询和检索。通过模型构件化框架操作模型数据、参数,通过模型管理构件完成模型的运行。基于模型构件化的决策支持系统体系结构如图5所示。

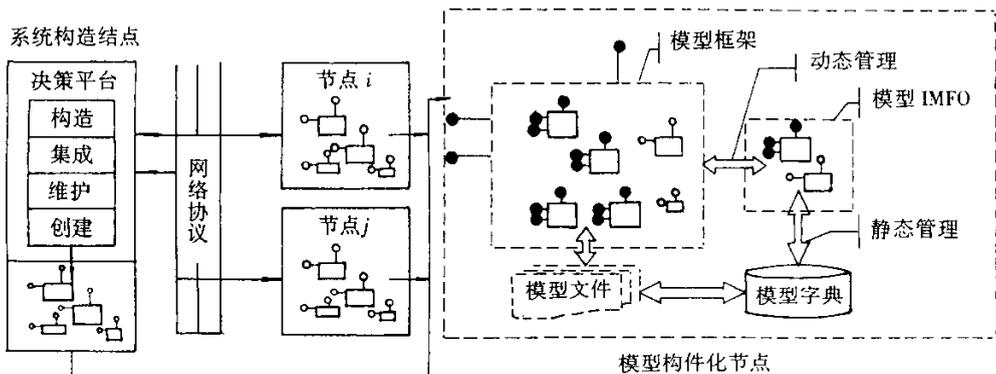


图5 基于模型构件化的决策支持系统体系结构

Fig.5 Architecture of decision support systems based on model component

基于模型构件化的决策支持系统由系统构造节点和模型构件化节点组成,系统构造节点包括决策平台和模型构件化节点中的内容,系统构造节点集成网络中各个节点所提供的模型构件,进行构造决策支持系统。模型构件化节点应用模型构件化框架技术建立构件模型,通过模型组织集成框架管理模型,使用模型信息获取构件,完成模型信息获取、信息存储、查询、修改则通过模型管理构件,进而建立模型字典库和模型库,以便被系统构造节点使用和共享。

5 结束语

基于构件的决策支持系统中模型组织集成框架 IFMO 是 CS - DSSP 决策支持系统平台中模型服务器的基础上进一步研究而设计的,扩充了模型部件,建立了模型构件库,为模型共享和复用提供了一条有效的途径。

参考文献:

- [1] 陈文伟. 决策支持系统及其开发[M]. 北京:清华大学出版社,南宁:广西科学技术出版社,2000.
- [2] 陈文伟. 智能决策技术[M]. 北京:电子工业出版社,1998.
- [3] 王肇东,沙一鸣,尤晋元. 可视编程模型[J]. 计算机科学,1999,26(1):75-78.
- [4] 李景峰,刘西洋,陈平. 一种可重用构件模型——类属构件[J]. 计算机科学,1999,26(9):83-86.
- [5] 赵新昱,肖卫东,张维明等. 构件的动态提取、创建和表现[J]. 计算机工程,1999,25(5):13-14.
- [6] 戴超凡,邓苏,刘青宝. 基于COM的可视化集成建模环境的设计与实现[J]. 计算机工程与应用,2000,36(6):4-5.
- [7] 赵新昱,郭兵兵,陈文伟. 构件属性、方法的自动提取和调度[J]. 计算机工程,2000,26(11).
- [8] 赵新昱,陈文伟,黄金才等. DSS中广义模型服务器规范化研究与实现[J]. 小型微型计算机系统,2000(6):595-599.

