

飞行器几何数据库设计及其管理软件开发

施永强, 杨青真

SHI Yong-qiang, YANG Qing-zhen

西北工业大学 动力与能源学院, 西安 710072

School of Dynamics and Energy, Northwestern Polytechnical University, Xi'an 710072, China

E-mail: yqshi@nwpu.edu.cn

SHI Yong-qiang, YANG Qing-zhen. Design of aircraft geometry database and development of database management system. Computer Engineering and Applications, 2007, 43(36): 114-116.

Abstract: The key to setting up database of aircraft geometry data is the design of database logical structure based upon the results of aircraft geometry data analysis and study, to reduce or eliminate the redundancy data. The software for aircraft geometry database management system, which is designed through the programme technology of Visual Basic 6.0 Language and SQL Server, has friendly interfaces and the function of database accessing in local area network and reasonable administrative level management, to make the management of aircraft geometry data systematize and enhance security.

Key words: data modal; database management; aircraft; geometry database

摘要: 设计基于局域网的飞行器几何数据库系统软件的关键, 是在对飞行器几何数据分析研究的基础上, 设计合理的数据库逻辑结构, 以消除或减小数据的冗余数据。使用 Visual Basic 6.0 和 SQL Server 编程技术开发出的飞行器几何数据库系统管理软件, 建立了友好的人机交互界面, 实现了局域网访问数据库的功能和合理的分层管理功能, 使得飞行器几何数据的管理系统化, 并强化了数据的安全性。

关键词: 数据模型; 数据库管理; 飞行器; 几何数据库

文章编号: 1002-8331(2007)36-0114-03 **文献标识码:** A **中图分类号:** V221.92; TP317.4

1 引言

在航天飞行器的设计中, 几何外形要通过流场计算才能优化设计, 通常的做法是建立几何图形与做流场计算分开进行。两者所用的数据存储都是以文件的格式分别存放于存储介质上。这就使得数据的存储不仅占用存储空间大, 而且由于文件数量、种类繁多, 而使得数据管理比较困难。对于数据量巨大的航天器设计, 数据的管理尤为艰巨, 也容易造成数据的丢失或者其他安全性问题。

本文要解决的问题就是设计飞行器几何数据库管理系统软件, 使得数据存储集成化、系统化。为了实现数据的共享, 同时出于对数据安全性的考虑, 本管理系统使用了 SQL Server 的数据库管理方式, 采用 Client/Server 结构体系, 实现了局域网内的数据分级共享, 同时也保证了数据的安全性^[1,2]。本管理系统不仅包括数据库本身的相应操作, 如建立、修改、备份、浏览、查询等, 而且包括对使用这个数据库的相应软件的输入/输出数据的管理, 及其与其他工业设计部门的通用软件的接口。此外, 出于对数据库中的冗余数据的考虑, 本文特别为飞行器几何数据设计了节约型数据结构模型, 减少了数据所需要的存储空间。

2 几何模型数据规范

通常几何数据的定义必须在机体直角坐标下采用绝对尺

寸定义, 参考国内外常用软件的定义方法制定的可重复使用天地往返系统的坐标系定义如下: 原点位于机头, x 轴沿机身轴线, y 轴沿翼展右舷, z 轴垂直于 xoy 平面, 并在飞机对称面内指向上方, 符合右手规则。直角坐标的示意图见图 1。

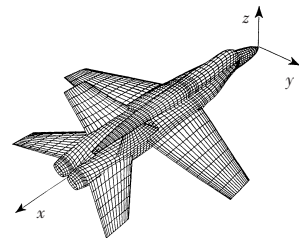


图1 飞行器外形坐标示意图

在确定了坐标之后, 几何表面点的定义按照其对应的直角坐标 (x, y, z) 给出, 一般情况下数据库中只保留绝对坐标。如果需要可以使用相对坐标, 如在指定机翼剖面坐标时可以翼型前缘为相对坐标的原点, 以当地弦长为无量纲参数。

飞行器不同的部件采用不同的剖面定义方法, 剖面的轮廓线均以与其垂直的轴线按顺时针方向由下至上, 或从左到右。根据彼此平行的剖面的法线方向可以将部件剖面分为如下三类:

(1) 剖面的 x 坐标为常数, 即剖面垂直于机体轴线 x , 如机身、雷达罩等, 称机身型部件;

(2)剖面的 y 坐标为常数,即剖面垂直于机体轴线 y ,如机翼和平尾等,称机翼型部件;

(3)剖面的 z 坐标为常数,即剖面垂直于机体轴线 z ,如垂尾等,称垂尾型部件。

对于第一类部件(截面垂直于 x 轴的机身型部件),截面轮廓线给定的方式为:自最下边一点(z 坐标最小)开始沿 y 增加的右手系方向(右手母指指向正 x 方向,其余四指所指方向)逐点依次给出。

对于第二类部件(截面垂直于 y 轴的机翼类部件),截面轮廓线给定的方式为:自下表面最后一点(x 坐标最大)开始沿 x 减小的右手系方向(右手母指指向正 y 方向,其余四指所指方向)逐点依次给出。

对于第三类部件(截面垂直于 z 轴的垂翼类部件),截面轮廓线给定的方式为:自右表面最后一点(x 坐标最大)开始沿 x 减小的右手系方向(右手母指指向正 z 方向,其余四指所指方向)逐点依次给出。

面的定义由剖面轮廓线组部件构成,在建立数据库时按照部件分类建立。这样设计的原因是目前绝大多数气动分析计算软件都采用按部件分块给出原始数据,然后通过前置处理软件生成几何体和表面网格。在数据库中,每一个飞机部件作为独立的数据库采用表面轮廓线数据为相应的数据表,这样可以节约 1/3 的数据存储量。

3 数据库结构

为保证数据库的结构与用户习惯的数据文件存放方式一致(因后者采用的是树型结构),在数据库中遵守以下原则:

(1)每一个飞行器型号设计为一个 SQL 数据设备,相当于一个磁盘分区或一个子目录;

(2)每一个飞行器部件设计一个 SQL 数据库,相当于一个子目录;

(3)每一个截面轮廓线坐标设计为一个 SQL 数据表,相当于一个数据文件。

数据库结构可用图 2 表示。

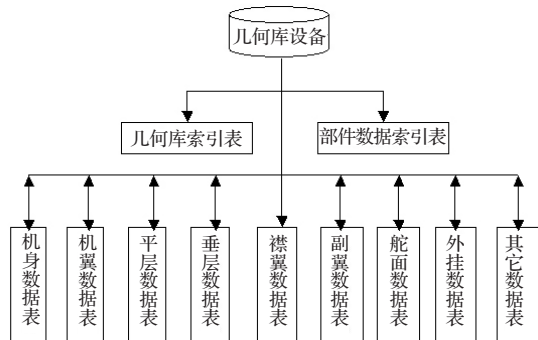


图 2 几何数据库结构示意图

几何库索引表包含几何库的所有宏观信息,比如几何库中几何部件的名称、类型、所包括的剖面数(即表的个数)等基本信息。几何库索引表的字段结构如表 1 所示。

表 1 几何库索引表的字段结构

字段	类型	说明
部件名称	字符	部件的名称如机身、机翼等等
部件类型	字符	表明数据给定方式,如机身型、机翼型等
数据库名	字符	表明该部件在数据库的名称
剖面数	数值	代表剖面个数

建立部件数据索引表的目的是统计和管理数据库中部件的个数、每个部件中剖面的个数以及剖面上数据点的个数,并记录部件的安装信息,如机翼的安装位置、安装迎角、上反角等。这些信息对于不同的部件不完全相同,在给数据库添加数据或从数据库中取出数据时都需要这些信息,因此需要单独建立作为每个部件数据库的一个表。它包括的字段如表 2 所示。

表 2 部件几何数据库索引表的字段结构

字段	类型	含义
剖面序号	数字	表示部件被分解成若干剖面后,相应剖面的编号,其编号规则按模型上的实际位置由前至后(对于机身),或由左到右(对机翼剖面),或由下至上(对垂尾等)。在数据库部件索引表中记录。
剖面数据表名	字符	表示相应剖面数据表的名称。
剖面点数	数值	代表剖面上几何外形坐标点的个数,在数据表中代表剖面数据表的总记录个数
剖面尺度因子	数值	代表剖面坐标在无量纲时所采用的尺度,如机翼剖面的弦长,机翼翼展等。如采用绝对尺寸则此值取 1 或空
剖面安装角	数值	在安装时相对于剖面法线所旋转的角度(单位为度)如采用绝对坐标则此值取 0 度或空
部件安装(上反角)	数值	指整个部件与机身的安装关系,一般指绕机身轴线旋转的角度(单位为度)

机身型数据库可以是机身、短舱、雷达罩、吊挂等轴线平行于 x 轴的部件。在这些部件数据库的数据表中只按剖面建表保存剖面数据点的 y 和 z 坐标,分别作为数据表的数值型字段。 x 坐标作为冗余坐标保存在数据表索引中,每一个数据库中都有一个数据表索引。

机翼型数据库可以是机翼、平尾等截面法线平行于 y 轴的部件。在这些部件数据库的数据表中按剖面建表保存剖面数据点的 x 和 z 坐标,分别作为数据表的数值型字段。 y 坐标作为冗余坐标保存在数据表索引中。

垂尾型数据库可以是垂直尾翼、外挂的挂架等截面法线平行于 z 轴的部件。在这些部件数据库的数据表中按剖面建表保存剖面数据点的 x 和 y 坐标,分别作为数据表的数值型字段。 z 坐标作为冗余坐标保存在数据表索引中。

除上述三种类型的飞行器部件外,还存在其他一些部件,其截面法线与任何轴都不平行,通常需要旋转之后才归为上述某一类部件,这类部件如导弹的弹翼,需要在索引库中给定部件的安装角(上反角)的值。

3 数据库管理系统设计

本数据库管理系统设计采用了 Server/Client 服务模式,实现了数据分级共享。同时,也减轻了服务器的工作量。整个系统如图 3 所示。此外,为了使得数据库得管理适应现在的可视化要求,同时也为了操作的简单方面。本文使用了 Visual Basic 开发了飞行器几何数据库管理系统。数据库的管理部分设计采用了 SQL Server 2000 的内置功能模块,配合飞行器几何数据库的要求设计了数据库服务器的管理、数据库的管理、数据表的管理软件模块。管理的内容涵盖了对服务器、数据库及数据表的所有可能的操作。

在本文数据库管理软件模块设计时考虑了两种主要管理方式^[3]。一种是基于 SQL Server 的通用数据库管理,利用 SQL Server 所带的 SQL NameSpace 对象对数据库进行管理;另一种

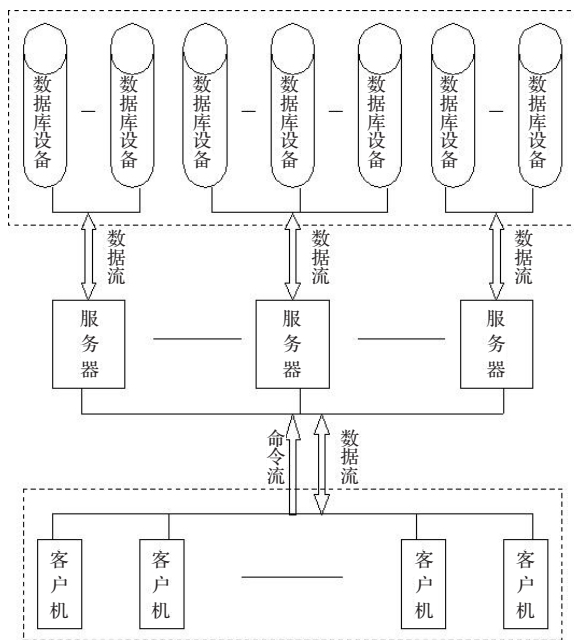


图3 数据库管理系统结构示意图

利用 SQL DMO 对象对数据库进行管理。前一种管理方式可以进行数据库的修改、删除等工作,后一种方式只允许向数据库中增加数据库或对表进行操作,而不能删除数据库。前一种管理方式适用于管理员对数据库进行管理,后一种适用于普通用户。采用 SQL Namespace 方法可以对表进行大多数操作,但对表的数据不能显示和操作,而且对于非管理员用户的安全性要求使得许多功能不能实现。为此,本项目设计了使用 SQL-DMO 对象对数据库和数表进行操作的功能,它使得普通用户可以查看数据库、数据表、表的字段和属性,而不必担心数据被恶意修改,因为用 SQL-DMO 设计了显示数据库信息和创建新

(上接 79 页)

度很大的集散节点,因此该网络具有无标度特性。但虽然平均路径长度较小,但却不具备簇系数大的特点,因此该网络不具有小世界特性。

5 总结及展望

本体的结构具有无标度性,可知本体中的集散节点的数量较少,且集散节点所表示的概念往往是得到相关领域专家认可的在该领域比较重要的概念,本体中其他的概念的衍生和演化也是以这些概念为基础的。在进行语义匹配的过程中,可以根据概念的度分布,衡量其重要程度,优先匹配权重较大的核心概念。今后将分析本体的复杂结构特性对语义检索的意义,以及本体的复杂特性在语义检索中的应用。

(收稿日期:2007年7月)

参考文献:

- [1] <http://www.geneontology.org/>.
- [2] Das A, Wu W, Mcguinness D. Industrial strength ontology management[M]. The Emerging Semantic Web.[S.l.]: IOS Press, 2002.
- [3] Stojanovic L, Motik B. Ontology evolution within ontology editors[C]//

数据表的功能,而不能对其进行任何修改。如此的设计使得数据库的管理和操作分离,保证了数据的安全性。

此外,本文设计了飞行器几何数据库与 CAD 软件接口^[4],使得图形文件的最主要的信息能够出库、入库。这样做不仅便于对数据的管理,而且可以使得数据所占用的存储空间大大减少。管理系统还设计了数据库数据图形文件输出功能,使数据库中点的坐标数据生成图形文件或者生成作图软件可以打开的标准文件,实现了与一些通用作图软件的数据共享。这样就可以实现图形文件与数值计算中的几何离散点文件的互通,并能够在一定程度上达到二者的统一存储,节省了存储空间,也便于对数据的管理。

4 结束语

本文采用 Client/Server 结构体系所设计开发的飞行器几何数据库管理系统具有友好的图形用户界面,操作简单方便,可以实现局域网内的数据管理与操作。使用了两种管理方式,使得数据库的安全性得到了保证。此外,从其数据库结构本身的设计而言,节约了 33% 的存储介质,减小了数据的冗余度。而飞行器几何数据库与 CAD 软件接口设计也减少了数据冗余,从而最大程度上节省存储空间。(收稿日期:2007年9月)

参考文献:

- [1] 欧阳明. Client/Server 体系结构的发展[J]. 计算机系统与应用, 1997(11): 50-53.
- [2] 陈旭, 刘加伶. Client/Server 与 Browser/Server 结构的分析比较[J]. 重庆工学院学报, 2000, 14(2): 100-103.
- [3] (美) Michael Otey, Paul Conte 著. SQL Server 2000 开发指南[M]. 北京: 清华大学出版社, 2002.
- [4] 施永强, 杨青真. 飞行器几何数据库与 CAD 软件数据交换问题研究[J]. 计算机工程与应用, 2006, 42(22): 84-85.
- [5] Proceedings of Onto Web-SIG3 Workshop at the 13th International Conference on Knowledge Engineering and Knowledge Management, 2002: 53-62.
- [6] Yao H N, Orme A M, Eitzkorn L. Cohesion metrics for ontology design and application[J]. Journal of Computer Science, 2005, 1(1): 107-113.
- [7] Kang D Z, Xu B W, Lu J J, et al. A Complexity measure for ontology based on UML [C]// 10th IEEE International Workshop on Future Trends of Distributed Computing Systems, 2004: 222-228.
- [8] Hsi I. Analyzing the Conceptual coherence of computing applications through ontological excavation[D]. PhD Thesis Proposal, 2004.
- [9] Mungall C. Increased complexity in the GO[EB/OL]. <http://www.fruitfly.org/~cjm/obol/doc/go-complexity.html>.
- [10] 杨哲, 张大陆. 基于概念模型的本体复杂性研究[J]. 计算机工程与应用, 2006, 42(10): 19-22.
- [11] Albert R, Barabási A L. Statistical mechanics of complex networks[J]. Review of Modern Physics, 2002, 74(1): 49-97.
- [12] Watts D J, Strogatz S H. Collective dynamics of small-world' networks[J]. Nature, 1998, 393: 440-442.
- [13] Newman M E J, Watts D J. Renormalization group analysis of the small-world network model[J]. Physics Letters A, 1999, 263: 341-346.