

运用 PowerDesigner

设计实时帐务处理数据库

文章介绍了如何运用数据模型构造软件PowerDesigner进行帐务处理系统数据库的设计及帐务系统实时处理的实现思想、实现方法。



PowerDesigner 设计特色

PowerDesigner 是 Sybase 公司推出的基于客户 / 服务器体系结构的一组图形化的数据库建模设计工具。它的六个模块为应用系统的开发提供了一个完整的集成化的开发环境, 支持系统开发过程中从需求分析、概念结构设计、物理结构设计到对象和组件生成的所有阶段。它采用结构化的方法设计和建立数据库与数据仓库, 其建模过程分为三个阶段 (如图 1 所示), 首先是处理分析模型的设计阶段(Process Analyst Model(PAM)), 此阶段是数据库设计的需求分析阶段, 也是系统开发过程中的一个关键工作阶段, 其目标是对现实世界要处理的对象, 即面向现实世界建模。PAM主要是利用数据流程图来描述信息系统中所用到的基本数据以及对这些数据的处理, 将组成系统的各个处理过程以一个树状的层次结构模型表示, 以此作为下一阶段 CDM 和 PDM 设计的基础。第二是概念数据模型设计阶段(Conceptual Data Model(CDM))。实体关系图正是概念模型一种最流行的表达方式, 此阶段把现实世界中需要保存的信息抽象成信息世界中的实体和关系产生实体关系图。这一阶段可以为组织所需的高质量的应用提供坚实的基础。第三是物理数据模型设计阶段, 即面向具体的数据库管理系统建模。此阶段对物理数据库结构进行设计, 加入概念模型中未考虑的因素: 触发器的运用、存储过程的定义、与应用程序的良好连接以及主键、外键、视图、索引等, 所以是面向机器的。PowerDesigner 会自动地根据具体数据库的特征生成物理数据模型(Physical Data Model(PDM))。在此基础上, 设计人员可根据需要修改物理模型。

PowerDesigner 可用于数据仓库的设计和实现; 支持多种数据库(如SQL Server,DB2,Sybase,Oracle,Informix等), 并能产生多种数据库的物理数据模型, 可自动根据物理数据模型生成产生数据库(包括建表、索引、主键、外键、触发器、存储过程)的SQL语言和完整的文档资料, 同时可以实现不同数据库之间的相互转换及把已存在的数据库的内容逆向生成概念的和物理的数据模型, 从 CDM 中导入对象到 PAM 中, 可同步更新 CDM 和 PAM 中相同的对象; 支持团对同时设计, 提高设计效率, 并可针对模型特定部分进行设计, 使设计与实际保持一致、数据模型与应用相互独立; 另外PowerDesigner还可为物理数据模型的对象定义扩展属性, 利用扩展属性在4GL用户界面和PDM之间传递信息, 实现数据库设计与用户界面的接口; 对企业的模型信息还可用只读的图形化方式访问。

运用 PD 生成帐务处理数据库

利用 PowerDesigner 生成数据库可分为四个步骤: (1) 处理分析模型设计; (2) 概念模型设计; (3) 物理模型设计; (4) 自动生成 SQL 语句或将结果移到 PB 和 DBMS 中。由于 PowerDesigner 可将概念模型和物理模型逆向转换, 也可从 CDM 中导入对象到 PAM 中, 以同步地更新 CDM 和 PAM 中相同的对象, 并检查 CDM 中建立的对象是否都用到了 PAM 中, 进而对 PAM 做进一步的修改, 故我们在设计帐务处理系统的数据库时首先利用PD的ProcessAnalyst(PA)设计PAM, 然后将物理模型设计出来, 物理模型设计的主要内容就是表、列、主键、外键、触发器、视图、存储过程, 然后由它自动产生概念模型和文档资料。

1. 物理模型 (PDM)的设计及实现思想

帐务处理系统主要是通过采集记帐凭证的数据产生帐簿、报表和各种辅助帐的核对、核销以及进行相关财务分析的数据。在 PowerDesigner 环境中, 选择新建 PDM 和 DBMS 后, 利用放置于面板中的 Tool 工具箱中的工具

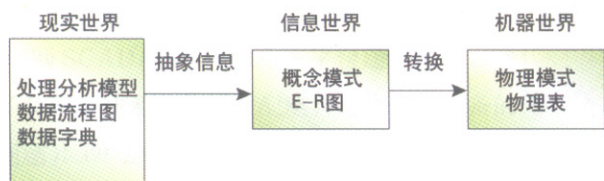


图 1 PowerDesigner 建模过程

建立凭证主表、凭证明细表、部门科目编码表、方向编码表、内外部银行主、明细表、现金流量项目编码表等各种核算和管理主、明细表，并建立主明、细表间的关联，关联后主表中的主键会自动成为明细表的外键，与数据库连接后会自动据主、外键生成索引。主表一般用于存放同类数据的相同部分内容，不同部分写入明细表，其中凭证主表用于存放一张凭证中公共部分字段的内容，明细表用于存放同一张凭证中的每一条科目（即每张凭证网格中的内容），两个表以凭证流水号、部门编码关联，凭证流水号按部门每月更新，这样可以减少数据的冗余，便于查询。对写入表中的数据均不能被删除，以出错标志标识，为审计部门留下充分的审计线索。每条记录在写入凭证主、明细表的同时要判断该科目是否需要辅助管理，需要，那么该科目还要写入辅助管理表中。如：银行科目每月都要与银行对帐，那么，这条科目在写入凭证主、明细表的同时还要将该科目写入外部银行对帐表中。本人在进行数据模型设计时，力求做到（1）最小定义。用尽可能少的实体描述尽可能多的对象；（2）适应性。当系统功能或内部结构发生变化时，尽可能只作最小的修改就能适应新的要求，大大降低维护工作量；（3）完备性。尽可能的满足用户要求；（4）字段定义清晰，能为应用程序设计人员和用户了解；（5）模型易于实现。具体模型部分如下：



图2 帐务处理系统部分物理模型图

本系统在实现上突出“实时处理”。模型中的凭证主表和凭证明细表既是记帐凭证文件，同时又是帐簿，即记帐过程实时进行。其凭证明细表包含了本会计期间的本期发生、年累计金额，期末余额可根据本月的累计金额减上

月的累计金额计算求得。科目的内容为企业所有发生的会计明细科目，其内部关系为：一级科目的数额等于下属二级科目数额之和，二级科目数额为下属三级科目数额之和，依此类推。只要有新的科目写入（分录只能做在底级科目），就会在凭证明表中逐笔写入该科目的发生额、年累计金额，根据凭证明细表内部结构及关系，同时将金额累计到上级及上上级直到总帐科目的发生额、年累计金额中。在处理上一般先从最后一级开始，逐级累计到一级为止。例如：某条记录 kmbm=“5210101”，假设这是一个三级科目（科目编码规则为树状），在处理时，先将5210101科目作为一条新记录写入凭证主、明细表中，登记发生额，同时读出部门科目编码表中该科目最大累计金额，据此计算出年累计金额，并将此金额更新部门科目编码表中的科目累计金额；然后，又将52101科目作为一条新记录写入凭证主、明细表中，登记发生额、年累计金额；最后将521作为一条新记录写入凭证主、明细表中，登记发生额同时计算出年累计金额。这样，对于一个三级科目，在凭证明细表中实际写入的是三条记录，三条记录以明细分支号区别。凭证查询时只提取无明细分支号和分支号最小的记录，这才是该凭证的实际分录。实现时充分利用触发器和存储过程，定义存储过程可以在PowerDesigner环境下，选Dictionary/Triggers and Procedures/list procedures,

在弹出的窗口中选择“new”定义新的存储过程。

2. 运用 PowerDesigner 生成触发器

触发器是存储过程的一个特殊形式。当对某表或某列进行插入、修改、删除操作时起作用。可以利用触发器来实现参照完整性。不同的DBMS支持参照完整性触发器的方法不同，所以参照完整性触发器基于特定的数据库存在。例如 SQL Server，其具体的程序略。定义触发器涉及到触发器模板（Trigger template）、模板项（Template item）、触发器（Trigger）、触发器脚本（trigger script）。对触发器模板、模板项和触发器可定制。但如果对触发器模板的进行改动将会影响到当前模型或所有模型；对触发器的改动将会影响到当前模型的某个表。■

参考文献

- 1 王珊、刘怡 SYBASE PowerDesigner 原理和使用高级指南，中国水利水电出版社，1999
- 2 侯志平、徐常胜 珠联璧合—PowerBuilder 与数据库配合开发技术，北京晓通网络数据库研究所。