

doi: 10.3969/j.issn.1006-1576.2011.08.026

一种基于数据库和面向对象的软件复用技术

赵建平¹, 赵建辉², 顾培¹, 施斌¹, 段慧芬¹

(1. 中国卫星海上测控部技术部, 江苏 江阴 214431;

2. 中国人民解放军 63999 部队, 北京 100094)

摘要: 为满足航天测控任务软件高密度多型号试验任务的需求, 以及解决软件规模、复杂度提高引起的软件开发问题, 提出一种试验信息分析软件复用技术。将面向对象技术和数据库技术通过“类虚拟机”机制相结合, 将配置于数据库的独立的分析项、分析步骤、字段处理方法和信息帧处理方法同最终的软件构件相映射, 实现了试验信息分析软件的复用。实践证明: 该软件复用技术有助于加快软件开发进度及提高软件的可靠性和易维护性。

关键词: 软件复用; 面向对象; 软件构件; 数据库; 虚拟机

中图分类号: TP311.11 **文献标志码:** A

Software Reuse Technology Based on Database and Object-Oriented

Zhao Jianping¹, Zhao Jianhui², Gu Pei¹, Shi Bin¹, Duan Huifen¹

(1. Technology Department, Satellite Maritime Tracking & Controlling Department of China, Jiangyin 214431, China;

2. No. 63999 Unit of PLA, Beijing 100094, China)

Abstract: In order to meet the high-density and multi-model task demand of space instrumentation and command software, and to solve the software development problems caused by the complexity and the scale of software, a new software reuse technology applied in the experiment information analysis software (EIAS) is proposed. The new software reuse combines the object-oriented technology with the database technology by mechanism of similar virtual machine, and maps the configuration of independent analysis items, analysis steps, field-processing methods and frame-processing methods in the database to final software components to realize the reuse in experiment information analysis software. The result shows that the new software reuse helps to speed up progress of software development and to improve the software reliability and maintainability.

Keywords: software reuse; object oriented; software component; database; virtual machine

0 引言

在航天测控系统中, 中国卫星海上测控部指挥所接收指控中心传送的各类数据信息和计算结果, 实时监视显示任务过程, 并以十六进制码实时记录各类数据信息。试验信息分析软件主要功能是实现信息的正确处理以及半自动和全自动化的分析流程, 及流程中数据的图形化显示, 完成实时任务事后的数据分析。为了简化试验信息分析流程的定制, 提高试验信息分析软件的代码复用, 降低开发风险, 提高软件的易维护性, 笔者采用基于数据库和面向对象的试验信息分析软件复用技术。

1 试验信息分析软件复用技术的构成

试验信息分析软件的复用技术分解为 3 方面的内容: 1) 基于数据库和面向对象的多架构视图体系结构设计; 2) 面向对象的试验信息处理与分析构件及复用流程; 3) 基于“类虚拟机”的数据处理与分析项解析技术。

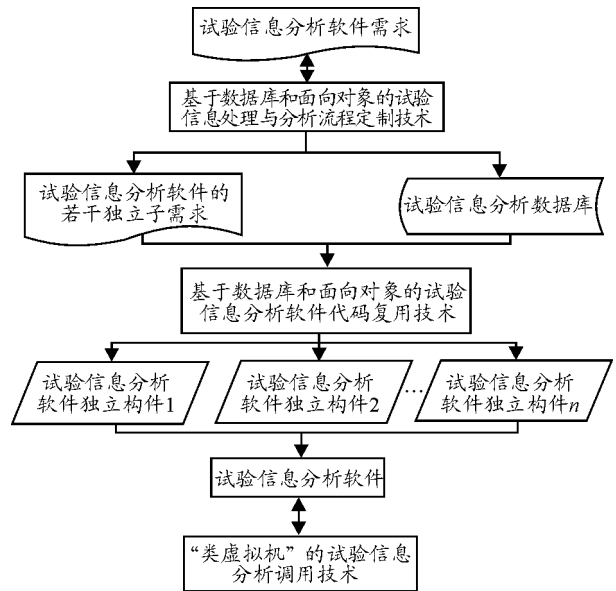


图 1 试验信息分析软件复用技术构成及其相互关系

基于数据库和面向对象的多架构视图体系结构设计描述了试验信息处理与分析流程的定制, 为面

收稿日期: 2011-04-23; 修回日期: 2011-05-20

作者简介: 赵建平(1974—), 女, 福建人, 硕士, 工程师, 从事软件工程及计算机应用研究。

面向对象的试验信息处理与分析构件及复用流程提供必需的设计成果; 面向对象的试验信息处理与分析构件及复用流程则为试验信息处理与分析流程的定制提供了代码实现; 基于“类虚拟机”机制的分析项解析技术将前2种技术的成果有机结合起来, 形成统一整体^[1]。3方面技术相辅相成, 构成了软件的复用技术, 其相互关系如图1。

2 基于数据库和面向对象的多架构视图体系设计

分别从用户和开发方的视角出发, 试验信息分析软件采用逻辑架构视图、开发架构视图和数据视图进行体系结构设计。

在支持软件复用方面, 面向对象方法能够从软件开发生命周期的需求分析阶段就开始发挥作用^[2], 可分析设计一个可复用的系统架构。这个架构可在不同应用系统中复用, 并通过扩充不同的详细设计来体现系统之间的差异, 能产生较大的复用价值^[1]。面向对象的分析设计中, 有以下3种架构:

1) 逻辑架构

逻辑架构关注功能, 不仅包括用户可见的功能, 还包括为实现用户功能而必须提供的“辅助功能模块”, 关注行为或职责的划分。

试验信息分析软件采用3层体系结构设计, 分为接口层、业务层和表示层, 接口层负责对数据库和记盘文件的访问, 业务层负责初始化以及数据分析处理, 表示层负责人机交互处理, 见图2。

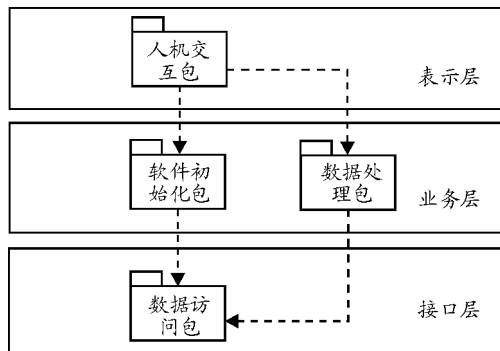


图2 软件层次分解结构图

各层功能划分如下:

① 接口层用于实现对数据库和记盘文件的访问, 主要是实现对数据库的存取访问以及记盘文件的读取。

② 业务层用于实现软件初始化和数据处理与分析。软件初始化通过读取配置文件, 完成对数据库连接参数、文件服务器路径等系统参数的配置;

实现各类数据处理分析功能, 包括记盘文件处理类、各类数据处理分析类、打印类等。

③ 表示层用于提供主窗口显示界面和各种对话框操作, 主要包括主框架、配置对话框类等。

2) 开发架构

开发架构着重考虑开发期质量属性, 如可扩展性、可重用性等, 关注点是软件模块的实际组织方式, 涉及源程序文件、配置文件、目标文件、第三方库文件等。

笔者从软件开发的早期阶段——分析与设计阶段就把软件复用作为一个重点问题来考虑:

① 在试验信息分析软件的分析阶段, 采用面向对象的分析方法, 将完整的试验信息处理与分析流程进行高度的总结、归纳与抽象, 将一个完整的试验信息处理与分析需求进行分解, 提炼出若干独立的处理与分析步骤。

② 将分解后的独立需求子项进行分类, 存储在定义好的不同数据表中:

A. 对不同类的试验信息分析流程, 利用存储在数据表中的这些独立的分析模块进行任意组装, 就可以实现流程的灵活定制, 使信息分析流程更加模块化、直观化, 从而在面向对象的分析阶段为软件复用提供了框架。

B. 试验信息由起飞零点、遥外测弹道数据、火箭分离点参数、轨道根数等构成。根据对信类帧各字段处理后再进行组合或再处理组合, 可抽象归纳出不同帧处理方法: 普通帧处理方法、起飞时间帧处理方法、瞬时站址数据帧处理方法、轨道根数帧处理方法、遥控指令帧处理方法等。

对试验信息帧处理以及分析流程的归纳、抽象与提炼使得试验信息分析软件在面向对象的设计阶段与实现阶段的软件复用成为可能。

3) 数据架构

数据架构着重考虑数据需求, 关注持久化数据的存储方案。

① 数据库设计

将试验信息处理与分析流程进行抽象和提炼, 可有效地支持开发与维护阶段的软件复用。使用数据库技术, 将面向对象的分析与设计结果进行分类存储, 可使整个试验信息分析软件开发周期各个阶段的成果一一分解映射。因而, 数据库成为连接面向对象的分析、面向对象的设计以及面向对象的编程的三者之间的纽带。

面向对象的分析以及数据库的设计如图3。

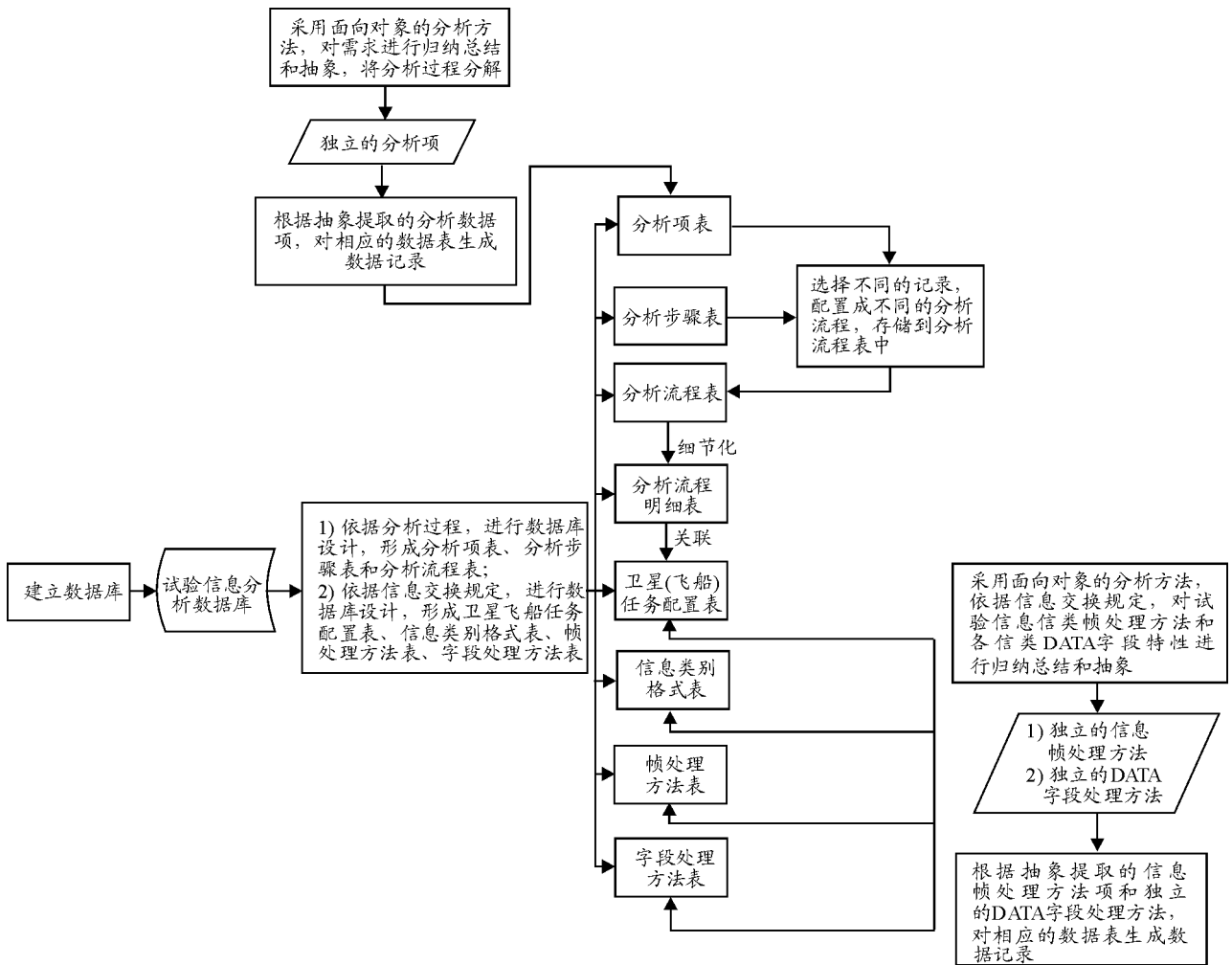


图 3 面向对象的分析和数据库的设计

采用面向对象的分析方法，一方面对需求进行归纳、总结和抽象，将需求分解，提炼出独立的数据分析需求子项；另一面依据信息交换规定，对试验信息各信类帧处理方法和各信类 DATA 字段特性进行归纳总结和抽象，提炼出独立的信息帧处理方法和独立的 DATA 字段处理方法。

建立试验信息分析数据库并进行数据库设计，一方面建立与分析流程相关的数据项表，如分析数据项表，分析项表和分析流程表；另一方面建立与数据处理相关的数据项表，如帧处理方法表、字段处理方法表、各信息类别格式表、卫星(飞船)任务配置表；其中卫星(飞船)任务配置表是数据处理过程与试验信息分析流程的纽带，该表装订了某型号任务的所有需要处理的信息帧的信类 B 码、帧处理方法编码、以及对应的信息类别格式表名称。

建立、修改和删除分析数据项、分析步骤以及分析流程的时候，只需在数据库中进行操作即可，

或开发一个试验信息数据库管理模块进行操作即可，而不需要修改程序。同样，数据帧处理过程中建立、修改和删除卫星(飞船)任务的信息类别、增加相应的帧处理方法和信息类别格式也只需要在数据库中进行操作即可，或开发一个试验信息数据库管理模块进行操作即可，而不需要修改程序。即使某型号任务出现新的特别的信息类别，只需在数据库中进行相应增加记录和增加该信息格式表的操作，并设计实现对应的新的帧处理类与对象即可。

② 试验信息处理与分析流程定制

为最大限度地提高分析数据项的复用性，提高软件构件的抽象程度，试验信息分析数据库采用了关系型数据库，同时建立了多个数据表，包括分析项类别表、分析项表、分析流程表、分析步骤类别表以及分析流程明细表、卫星(飞船)任务配置表、字段处理方法表、帧处理方法表、信息类别格式表。数据表的建立及其相互关系有效地描述了试验信息

处理与分析流程。

3 面向对象的试验信息处理与分析构件及复用流程

对于在面向对象的分析阶段中所有归纳、抽象和提炼出来的独立的试验信息分析流程需求和独立的字段处理方法和信息帧处理方法需求, 在面向对象的设计阶段用类作为其抽象描述, 这些独立试验信息处理与分析流程需求的属性、行为及其对外关系等也都是通过对象、类、封装、继承、关联等^[3]来表示的, 采用面向对象的编码技术将这些抽象出来的对象和类进行代码实现, 形成可复用的软件构件, 并与数据库中存储的信息处理与分析项形成映射, 从而将经过组装的分析流程代码化, 形成多个独立试验信息处理与分析软件构件。其中每一个分析流程中嵌套了对所需分析信息的原码数据帧的处理过程。

当进行软件编码的时候, 可以调用这些独立的软件构件, 加速软件的实现, 提高开发效率。具体流程如下:

1) 采用面向对象的编码技术, 对每一个分析项

及每一个字段处理项和信息帧处理项进行独立的编码。形成独立的软件构件, 这些构件要与特定的型号任务记录信息处理与分析过程剥离, 使其能够方便地被复用到其它的型号任务记录信息数据处理与分析中;

2) 在分析数据库中建立独立的分析项及字段处理项和信息帧处理项记录, 并与这些独立的软件构件相对应;

3) 系统启动时, 根据选择的分析的信息类别和分析项, 在数据库中提取该分析项下的所有分析步骤及数据处理方法, 形成分析队列, 其中分析队列元素里嵌套着试验信息的数据处理过程;

4) 顺序执行分析队列中的分析内容, 所有的分析步骤均与软件构件一一对应, 所以这种方法建立的队列可以任意组装和修改。

5) 分析队列里每个队列元素嵌套着试验信息的数据处理过程, 每个分析项与数据处理过程之间通过数据库中分析流程明细表和卫星(飞船)任务配置表相关联, 帧处理方法和字段处理方法均与软件构件一一对应。

试验信息处理与分析构件及复用流程如图 4。

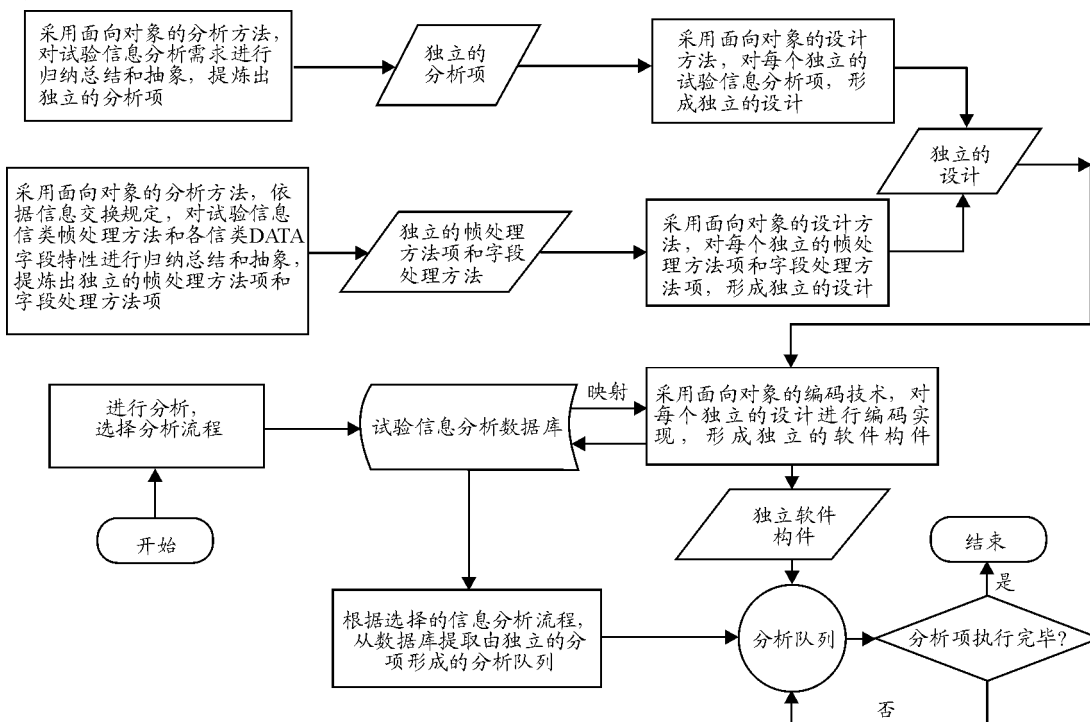


图 4 面向对象的试验信息处理与分析软件构件及复用

4 基于“类虚拟机”的分析调用

如图 5, 试验信息分析软件采用了类似于“虚拟机”机制的一种调用方法, 通过对分析项信息的

解释, 将各种独立的构件在程序执行层面上与数据库的记录映射关联, 将基于数据库和面向对象的试验信息分析软件复用技术应用起来^[1]。

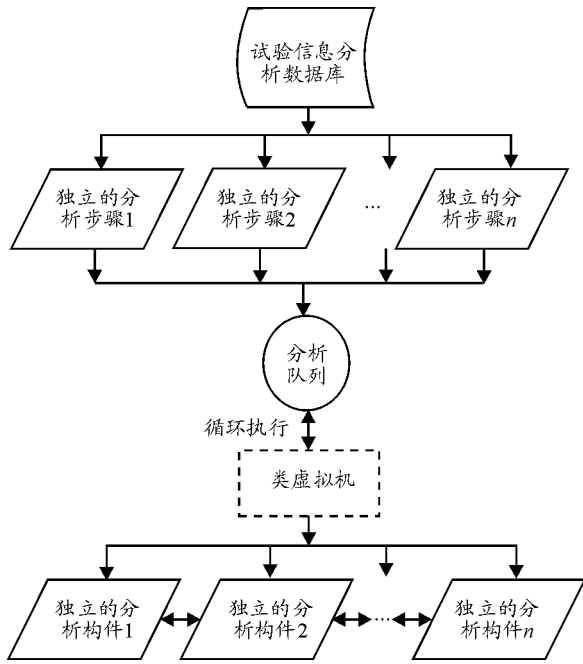


图 5 基于“类虚拟机”的分析调用

1) 在程序启动之初,就将数据库中的记录提取出来,形成嵌套着试验信息数据处理过程的分析队列,存储在内存中;

2) 在主程序中设立一个类似于“虚拟机”的解释线程,并将其常驻内存,通过检索分析队列中的记录和每个记录的类别,从而进行相应软件构件的调用,并通过将对象的信息传递到相应的软件构件中,输出相应的处理与分析结果。

当任务准备阶段,试图:

① 对分析项中某类信息帧处理方法或字段处理方法修改或增加时,只需对数据库中定义的卫星(飞船)任务配置进行修改或增加记录,并修改或增加相应的该信息类别格式表。

② 对某个分析内容进行修改的时候,只需要对数据库中定义的分析步骤信息进行修改即可;程序执行的时候,会根据修改过的信息重新形成分析队列,再利用“虚拟机”进行遍历执行,实现对分析流程的修改。

另外,虽然每个分析步骤都是解释执行的,但由于各分析模块的信息均采用了高效的数据结构定

义,且所有的操作都在内存中执行,因此,不会影响程序的执行速度。

5 结论

实际证明,采用基于数据库和面向对象的软件复用技术实现的试验信息分析软件具有以下优点:

1) 软件可维护性高。在面向对象的设计与实现中使用了模式方法,如观察者模式、策略模式等^[4],因此将新的字段处理功能、新的信息帧处理功能、以及新的分析项功能加到软件系统中去或对已有的功能进行修改成为一件很容易的事。

2) 代码复用性高。相同类型的分析项共用相同的代码对象,相同类型的字段和信息帧共用相同的代码对象,不用重复编码,形成独立的软件构件,而且还可用于其他试验信息解算和分析软件,实现试验信息分析软件的代码复用。

3) 通过对数据库的灵活配置,在不用修改代码的情况下,实现分析流程的任意组装,满足试验信息分析流程灵活配置的需要;在不用修改代码或只维护少量代码的情况下,实现信息数据处理过程满足某型号任务的需要。

4) 软件可修改性高。只需修改独立的软件构件或增加软件构件,即可实现对所有分析流程及数据处理方法的增加,提高软件维护效率。

5) 重复错误出现概率低。由于相同的分析项使用同一个软件构件,相同的字段处理项和信息帧处理项使用同一个软件构件,所以重复错误出现的概率低。

参考文献:

[1] 孙海峰,陈迪,解月江,等. 基于数据库和面向对象的火箭地面测发控软件复用[J]. 航天测控, 2010(2): 79-83.

[2] 顿海强,庄雷. 面向对象与软件复用技术研究[J]. 计算机应用研究, 2002, 19(3): 42-44.

[3] 阎宏. JAVA 与模式[M]. 北京: 电子工业出版社, 2008.

[4] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. 软件设计模式[M]. 北京: 电子工业出版社, 2007.