

【软考达人】

软考资料免费获取

- 1、最新软考题库
- 2、软考备考资料
- 3、考前压轴题



微信扫一扫，立马获取



6W+ 免费题库



免费备考资料

PC版题库: ruankaodaren.com

论软件设计模式的应用

摘要：

本人 2009 年有幸参加了中国石油集团的高性能数控测井系统项目的开发研制工作。该系统是在当前测井成套测井装备的基础上，为了满足高精度，高性能，高效率的要求开发的测井系统。该系统由井下成套仪器，测井遥测系统，测井地面系统，测井软件系统，测井解释评价系统等子系统组成。本人在其中主要是负责测井软件系统的分析、设计以及部分开发任务。设计模式是前人设计面向对象软件的经验和总结，在软件设计中灵活的使用设计模式可以极大的提高系统的稳定性，可扩展性，以及良好的可维护性。本文描述了在测井软件系统开发过程中，如何分析和发现相关模式，以及如何选择和应用设计模式，特别是介绍了 MVC 模式在软件框架和相关系统模块中的应用和使用效果。在文章的最后，讨论了在实际项目开发中，设计模式应用的有关想法和教训。

正文：

随着当前石油测井技术的发展，为了能更快，更好的得到储层地层信息，解决目前国内测井系统不统一，测井精度不高，效率低下的缺点，2009 年 1 月中国石油集团公司科技局成立了高性能数控测井系统项目，目的是为国内测井行业提供一个从井下到地面以及解释评价的整套测井系统。系统的设计目标是一次测井，取得所有合格资料，并且能保证 60 井次的免维修率。整个系统由井下成套仪器，测井遥测系统，测井地面系统，测井软件系统，测井解释评价系统等子系统组成。我主要是负责测井软件系统的分析，设计和部分开发工作。整个测井软件系统完成三个主要任务：测井数据的采集、测井数据的工程值计算、测井过程的监控。测井数据采集主要是采集井下仪器通过测井遥测系统传输的测井数据，并保证数据的完整性，正确性。测井数据工程值计算主要是把采集的数据根据不同仪器刻度计算方法进行工程值的计算。测井过程监控主要是把计算的测井数据用曲线和图像的方式实时的显示在屏幕和打印成图，由测井操作员进行实时监控。设计模式是前人设计面向对象软件的经验和总结，在软件设计中灵活的使用设计模式可以极大的提高系统的稳定性，可扩展性，以及良好的可维护性。在测井软件系统框架进行分析和设计时，考虑如何提高系统的稳定性、可扩展性和可维护性时，我们采用了 MVC 设计模式。MVC 模式构架包括三个部分：模型（Model）、视图（View）、控制（Control）。模型主要是对系统的数据和逻辑运算的封装。它独立与系统的界面和 I/O。视图把表示模型的数据和逻辑关系用特定的形式展示给用户。控制处理用户和软件之间的交互操作，当模型的数据有所变化时，控制负责通知视图做出相应的更新。模型、视图、控制的相互分离有利于模块之间内聚性的提高，耦合更加松散。一个模型可以对应多个视图，由控制来传播模型的变化从而更新视图。MVC 模式如何在测井软件系统实现，我们主要是从如下四个方面进行：一、分析系统功能，分离功能模型 首先根据系统的主要任务进行系统的模块分解。根据测井软件系统数据采集、数据转换和测井监控三个主要任务，把系统分为三个模块对应于 MVC 模式的三个部分。其中模型（Model）对应于数据的采集和工程值的计算。测井视图（View）对应于测井监控功能。测井模型所要实现的功能包

括：测井数据的采集、数据的刻度计算、数据的存储、数据的操作。测井数据的采集负责硬件平台的初始化，下井仪器的初始化，井下仪器数据的中断响应，数据帧的采集，数据帧的重组等。数据的刻度计算主要是根据不同的仪器实现数据的刻度计算，包括刻度系数表的获取、刻度计算、深度延迟计算等。数据存储主要是原始数据的存储和测井数据的存储。这里我们采用的是测井公用的 XTF 格式做为数据存储格式。数据的操作是视图和模型之间数据交互的接口。它主要是提供数据输入和输出功能。

二、视图的设计与实现 视图主要是提供测井数据的图形显示。通过调用模型中的数据操作方法，提取测井数据，根据不同的测井数据提供曲线、波列、图像等多种表现形式。在本系统的实现中，为了提高数据采集的稳定性和程序的健壮性，采用进程间通讯的方式。就是说视图的实现本身一个独立的程序。它与模型之间的通过 TCP/IP 网络进行通讯。视图主要包括数据源、数据表象对象、绘图打印模块等部分组成。数据源负责得到模型 (Model) 的数据，然后把数据分配给每个数据表象对象。数据表象对象是个有层次的类家族，其基类是绘图类 (CDrawObj)，所有的数据表象包括道 (CDrawTrack)、曲线 (CDrawCurve)、波列 (CDrawWave)、图像 (CDrawImage)，数值对象 (CDrawData) 等都是从其派生的。最后有绘图打印模块提供管理，负责视图的区域更新，数据表象的绘制和打印等功能。

三、控制的设计与实现 控制主要功能是提供用户的输入输出反馈，同时监控模型的数据变化，通知视图进行更新。由于控制和视图的耦合非常的紧密，在架构实现中，控制和视图是在一个应用程序中实现的。控制主要分为井下仪器控制和视图控制两个部分。其中井下仪器控制主要是由操作人员根据视图中的曲线和图像信息，对仪器发出的状态控制命令，以保证测井过程中数据和仪器的安全。视图控制则是操作人员对视图显示参数的调整，包括鼠标的响应和键盘的响应以及用户对测井原始图的特殊要求如道大小，曲线位置的摆放，颜色的调整等。

四、使用可动态添加算法模型 由于每次测井作业中下井仪器串的仪器种类和仪器的数量都是变化的，为了能更好的抽象出实际的测井模型，提高系统的灵活性，在模型中数据刻度计算部分，我们采用的动态添加的方式。我们把不同测井仪器的刻度算法封装到动态连接库，然后根据测井作业的不同，调用不同的仪器动态库中的刻度算法。由于视图和控制与模型之间的松耦合，当用户添加算法模块，视图与控制基本不要修改。

在采用 MVC 模式的软件框架后，整个系统分为两个部分，数据采集管理器 and 数据实时浏览器。数据采集管理器对应于模型 (Model) 的实现，数据实时浏览器对应于视图 (View) 和控制 (Control) 的实现。我们采用的是 Visual C++.net 基于 Window2003 平台来进行系统开发。采用 MVC 模式给我们带来了如下好处：

- 1、由于模型 (Model) 与视图 (View) 和控制 (Control) 之间的松耦合，使得我们非常容易就实现了一个模型运行同时建立多个视图。这在调试仪器时非常有用，当硬件人员调试仪器时直接连接网线就可以一边看仪器一边看数据。不再需要象以前必须到地面系统控制室查看数据了。
- 2、适合多硬件平台的跨接。由于不同的硬件平台上采集数据的方式都不同，有的系统采用的是 PCI 总线，有的是 USB 接入，有的是 ISA 卡接入。由于模型 (Model) 和视图 (View) 的松耦合，当要移植到不同的硬件平台上是我们只有修改相应的模型 (Model)，有可以实现对不同硬件平台的支持。
- 3、良好的可维护性和扩展性。由于采用 MVC 模式，系统模块功能划分明确，代码实现也相对容易。代码的错误不会在系统中扩散，同时由于可以动态添加仪器算法模块，

当用户添加新仪器时，不需要更改系统程序，只有添加仪器动态库 DLL 就可以了。在整个系统的开发中，我们还应用了一些别的模式，有些模式是在进行系统设计时，就考虑到而特意实现的，有些模式是在采用别的方法实现后，效果不太理想，在代码重构时引进的。在应用设计模式进行系统设计和开发后，整个系统各个模块之间逻辑变的相对独立，耦合也很松散，结构的扩展性良好。而且使得代码的重用的程度变好，减少了错误的发生和错误在代码中的扩散。但是在实际应用模式的过程中，我还发现模式应用的经验越丰富，模式应用的就越好。有时在采用何种模式时，有几种模式方案可以采用，但是具体采用那个模式就需要不断的尝试，看看模式是否满足实际的需要。特别要注意的是不能为了设计模式进行设计，也就是过分设计的问题。这样会导致设计过于复杂，偏离程序设计简约够用的基本原则。目前设计模式在软件开发中的应用正引起广大开发人员的注意，各大软件开发商也在软件开发工具中提供了有关设计模式的自动应用的工具，相信设计模式会越来越多应用于软件的设计和开发中。