

电力仿真系统图形编辑器面向对象的设计与实现

董朝霞¹, 陈青华¹, 范斗²

(1. 华中科技大学水电与数字化工程学院, 湖北 武汉 430074; 2. 河南省电力公司, 河南 郑州 450052)

摘要: 阐述电力仿真系统图形编辑器面向对象的设计方法, 并给出其在 UNIX/X Window 下的实现过程, 所介绍的方案已成功应用于实际电力仿真系统。

关键词: 图形编辑器; 面向对象; 图元; 电力仿真

中图分类号: TM743

文献标识码: B

文章编号: 1003-4897(2002)06-0040-03

1 引言

现代电力仿真系统均采用全图形方式显示及操作^[1], 包括电力系统潮流图、厂站图、分区系统图等。其中的图形是二维的示意图, 该图使用电力工程师能接受的标准符号(本文中称为图元)表示电力系统各组成部分。由于仿真系统中图形直接面向操作人员和受训人员, 因而要求图形简洁、布局合理、操作性好、界面友好。图形编辑器正是用来生成这种图形的专用 CAD 工具。

面向对象(OO)技术自出现以来受到广泛关注。面向对象的开发强调从问题域的概念到软件程序和界面的直接映射^[2], 心理学的研究也表明, 把客观世界看成是许多对象更接近人类的自然思维方式, 而且对象比函数更稳定。面向对象的开发也支持软件工程实践中的信息隐藏、数据抽象和封装, 并且面向对象开发的软件更易于修改、扩充和维护。面向对象的开发鼓励重用, 包括软件的重用和分析、设计、建模的重用, 借鉴成熟的开发模式可以减小复杂系统所面临的风险。

2 系统分析与面向对象设计

2.1 系统功能和结构分析

系统的功能设计框图如图 1 所示:

电力仿真系统中的图形可分为背景和前景两大部分。背景以静态图符为主, 由点、线、圆、折线、文本等基本图符组成。背景图不具备电力系统的任何特点, 因而可用于绘制省网中的行区、地理图等图形。前景是随系统状况改变而变化的图符或动态数据, 例如隔离开关随开合应显示不同状态。前景又可以称为元件层, 是由表示实际设备的抽象图符组成, 并且被定义了相应的动态属性。图形编辑

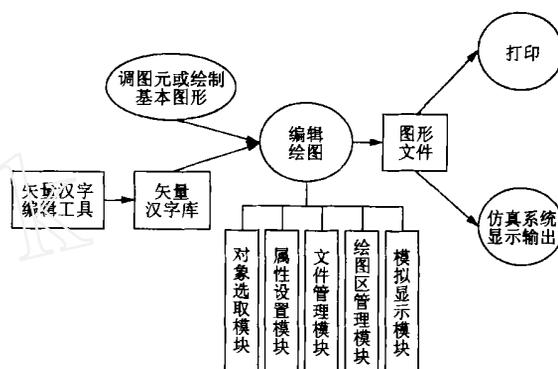


图 1 图形编辑器设计框图

器最终生成的是待驱动的静态画面, 将该画面与实时数据绑定才可获得反映实际状况的动态画面。

电力仿真系统中的图元已被作为行业标准和国家标准固定下来, 每个图元代表的意义是不可改变的。从软件工程的角度出发, 形成图元库对于系统的可扩展性具有重要意义。目前图元库可以采用两种技术: 一是图元库与代码库相结合, 即对图元进行分类, 每类图元的每个功能对应一段代码; 二是提供图元的编辑工具供用户定制图元以形成图元库。方法一的可扩展性差, 当增加新图元时要改动大量代码, 同时增加了程序出错的可能, 因而适用于专用的图形工具。采用方法二增加和编辑图元时以图元的整体为单位进行处理, 通过合理设计图元库文件和索引方法, 可以有效地管理图元并可方便地扩展。图上的多个图元可以组合和解除组合, 组合后作为一个整体编辑处理。

系统采用单线程结构, 局部放大功能采用子窗口显示, 其他功能均显示在主窗口中。局部放大时用户在主窗口拖动一个矩形框(视口), 被选中的图元将在子窗口按设定的倍数放大显示。用户可以用鼠标或键盘移动主窗口中的视口, 此时子窗口中的显示同时变化。

系统利用分层机制支持绘制复杂的图形并可以有选择地查看,一般将图上不同类型的数据放在不同的图层中。系统中设置测试和编辑两种工作模式。编辑模式下可以任意绘制和更改图形,测试模式用于测试动态数据等前景图图元的位置、范围显示是否正常,图形整体布局是否合理,并且测试模式下可以预览图层的分布情况。

反映电力元件的动态图元除了可以定义与数据库关联的属性外,还可定义设备元件的各种性质,如设备的铭牌参数等。这些属性在仿真系统中一般不出现,用户可以查询各个属性的值以便更好地了解现场的设备情况。

2.2 类和对象设计

根据以上分析将图形编辑器分为应用层和图形对象层两个层次,其类结构如图 2 示:

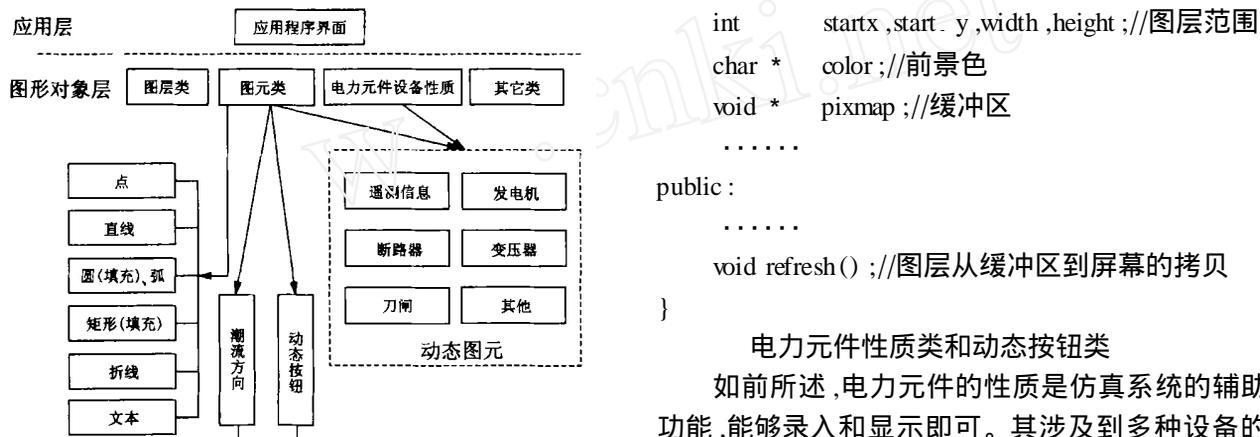


图 2 图形编辑器类结构图
图元类和图层类

显然图元类是最基本的类,其它如静态图符或动态图元均是它的子类。将各类图元(包括静态图符)的公共属性和方法进行封装来定义图元类 CObject,可以充分体现面向对象开发的继承性和封装性的优点,提高开发效率。下面简要列出 CObject 类的部分属性和方法。

```
class CObject {
protected:
    short type; //图元类别号
    short layer; //图元所属图层号
    char * color; //颜色
    float scale; //显示比例
    short isSelected; //标记是否选中
    int start_x, start_y; //图元范围矩形起点
    int width, height; //图元范围矩形的宽和高
    .....
```

```
public:
    .....
```

```
void display(); //图元的显示
Boolean isHit(); //判断图元是否选中
void move(); //图元的移动
void dynamic(); //图元的动态定义
    .....
```

图层类是系统中最重要的一类,该类的简要描述如下:

```
class CLayer {
protected:
    char * name; //图层名称
    short layer_num; //图层编号
    Boolean isVisible; //是否可见
    int start_x, start_y, width, height; //图层范围
    char * color; //前景色
    void * pixmap; //缓冲区
    .....
```

```
public:
    .....
```

```
void refresh(); //图层从缓冲区到屏幕的拷贝
}
```

电力元件性质类和动态按钮类

如前所述,电力元件的性质是仿真系统的辅助功能,能够录入和显示即可。其涉及到多种设备的多种属性,如发电机的各种标么值和限值,变压器的电阻、电抗、电纳和档位等,在此不做详细介绍。

按钮作为标准界面组件之一,一般是不需要开发者自己绘制的。但电力仿真及 EMS 系统中考虑到图形布局的灵活性和系统的方便扩展性,图中往往在绘图区而不是用户界面放置按钮。鉴于此在图形编辑器的设计中要求用二维矢量图形实现三维按钮的效果,系统中的按钮定义为以下几类:

- Hyper Button :调用某一特定画面,用于画面的切换;
- Hyper Region :调用线路对端画面,只显示标签文字不显示按钮外形,类似 Web 超链接功能;
- Control Button :控制该图形不同图层之间的切换;
- Resource Button :选择画面的数据源(采集或模型);
- Execute Button :执行某一特定功能(如显示报警记录等)。

这些按钮类均继承自抽象按钮类 class CButton, 其中定义了按钮按下和弹起时的处理方法供重载。

3 系统的实现

目前 UNIX 系统中广泛使用 X 窗口系统作为用户界面的显示模型和窗口模型, 本系统开发中用户界面部分采用 Motif^[3] 组件库, 绘图功能利用 Xlib 的绘图函数实现^[4], 编程语言为 DEC C++。系统的硬件平台为 DEC Alph XP1000 工作站, 操作系统为 Digital Unix 4.0F。

3.1 数据组织和图元组的实现

图形数据在内存中以链表的形式保存, 图元以所在的图层来分类, 同一层的图元形成一个链, 这样便于图元的增删和删除后的恢复。显示有大量图元组成的复杂图形时, 不可见的图层不参与显示, 可以显著提高显示函数的执行速度。链表结构如下:

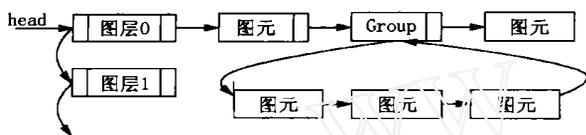


图3 图元链表结构

实现中采用嵌套的链表结构来表示图元组。

图3中 Group 标志即为组标志。

3.2 图形打印功能

由于 UNIX 系统中并未提供图形的打印驱动, 本系统中打印功能用两种间接方式实现: 转换为 Windows bmp 位图格式送局域网 PC 机 (Windows NT) 打印输出; 由 X Window 应用程序 xwd 捕捉程序窗口后转换为 PostScript 打印机识别的格式在 UNIX 环境下打印输出。

3.3 图元的拾取

图元拾取功能是所有交互式绘图平台都必须具备的, 同时也是其他编辑功能如移动、缩放、复制、对齐等的基础和前提, 其精度和效率对整个系统影响很大。本系统实现中提供点拾取和框拾取两种方法。点拾取是用户在绘图区单击鼠标时进行的, 拾取成功则高亮显示被选中的图元, 若用户在绘图区拖动一个矩形框则进行框拾取。一组图元中的任意

一个被选中则全组图元均被选中。

以点拾取为例, 拾取算法为了提高效率, 图3所示链表中若某一层图元不可见则该图层上的所有图元均不参加比较判断。判断某个图元是否被选中时首先进行初选, 方法是判断鼠标位置是否位于 (start_x, start_y) 为左上点, width 为宽度高度到达绘图区下端的矩形内, 对通过初选的图元进行进一步的比较。一旦图元选取成功则终止比较以避免无谓的计算。

3.4 层切换的快速实现

由多层图元组成的图形进行层切换时, 一般的方法是清除绘图区上要关闭图层的图元, 然后重画新图层的所有图元。当图形比较复杂时执行多次 X 绘图函数显然效率不高。本系统实现中每个图层都有一个绘图缓冲区 pixmap, 绘图时图元画在绘图区的同时也画在相应层的缓冲区中。执行层的切换时首先清除绘图区, 然后利用 X 函数 XCopyArea 实现所有可见层到绘图区的映射。实践证明这种方法可以明显改善大图的层切换效率。

4 结束语

本文介绍的图形编辑器的面向对象设计与实现方法已成功运用于河南电网调度员培训仿真系统 (河南 DTS) 中, 其具有界面友好、操作简便、设置灵活等特点。实践证明电力仿真软件开发中面向对象思想的运用保障了系统的可扩展性, 增强了系统的可维护性, 提高了工作效率, 具有较高的实用价值和广泛的应用前景。

参考文献:

- [1] 张力平. 电网调度员培训模拟 (DTS) [M]. 北京: 中国电力出版社, 1997.
- [2] 宛延阁. C++ 语言和面向对象程序设计 [M]. 北京: 清华大学出版社, 1998.

收稿日期: 2001-10-12; 修回日期: 2002-04-05

作者简介: 董朝霞 (1964 -), 女, 副教授, 研究方向为电力系统仿真, 计算机监控; 陈青华 (1977 -), 男, 硕士研究生, 研究方向为电力系统仿真。

Design and implementation of graphic editor based on object-oriented method in power systems simulation

DONG Zhao-xia¹, CHEN Qing-hua¹, FAN Dou²

(1. Hydropower & Information Engineering Dep. HUST, Wuhan 430073, China;

2. Henan Power System Dispatching Center, Zhengzhou 450052, China)

Abstract: This paper explains the design of graphic editor based on object-oriented method in power systems simulation. The implementation is also discussed under the UNIX/X Window environment. The application has demonstrated the effectiveness of the proposed method.

Keywords: graphic editor; object-oriented; model; power systems simulation