

64-66

图形用户界面 自动生成 软件开发

(17)

图形用户界面的自动生成*)

The Automatic Generation of Graphical User Interface

徐锡山 陈火旺 齐治昌

(国防科技大学计算机系 长沙 410073)

TP311.52

摘 要 This paper opens with a discussion of the importance and difficulties of the automatic generation of graphical user interface(GUI). Then, a model, which is called multi-faceted model, for GUI's dialog control is proposed. According to this model and GUI's characteristics, we build a framework for the specification of dialog control with RSL(RAISE Specification Language). Finally, we discuss how the graphical user interface for a interactive system is generated automatically on the basis of the framework.

关键词 Graphical user interface, Dialog control model, Multi-faceted object, Automatic generation

1 图形用户界面自动生成的意义和研究现状

现代软件产品几乎全部采用交互式的图形用户界面。在图形用户界面大大提高了使用的方便性的同时,也使软件的开发任务变得更为复杂与困难。一项调查表明^[1],用户界面的代码量约占总代码量的48%,在用户界面的设计与实现阶段所花的时间为软件设计与实现总时间的45%~50%,其中34%的系统使用工具箱实现,27%的系统使用UIMS实现,14%的系统使用用户界面构造器实现。由此可见,界面软件已成为软件系统的重要成分,虽然各类用户界面开发工具为界面软件的开发提供了有力支持,界面软件的开发任务仍很艰巨。

在软件工程中,针对瀑布模型的种种缺陷,提出了快速原型方法和进化式软件开发。在快速原型方法及相应的螺旋模型中,用户主要是通过对原型的界面进行操作实验来评估原型并提出改进意见的。因此,快速构造和进化用户界面是快速原型方法的技术前提之一。此外,在软件开发的早期阶段,快速地生成用户界面是进行需求分析的有效途径。

在以上需求的驱动下,图形用户界面开发工具的发展历史呈现为界面开发的自动化程度不断提高的历史,出现了实用的图形用户界面自动生成工具,

这些工具在CASE技术的支持下主要实现目标软件图形用户界面外观布局的自动生成以及与目标软件其他部件的连接,其典型的功能为:

(1)界面编辑功能 图形用户界面一般包括窗口、菜单和对话框。对话框由命令按钮、编辑框、列表框、滚动条等界面元素组成。菜单设计比较简单,CASE工具一般为此提供以下功能:

- 定义菜单项的名称、加速键及其属性(检取标志、可见性、使能性等)。
- 插入、删除菜单项。
- 改变菜单项的级别与位置,等等。

对话框设计工具以可视化的方式让开发人员直观地生成界面元素,确定元素的属性(包括标题、尺寸、位置等)。具体地,系统应为每个界面元素指定一个工具,开发人员用鼠标选中工具后即可在客户区绘出相应界面元素,然后还可以通过选中界面元素对其属性进行定义或调整。系统通常还提供一组对齐工具以方便对话框的布局设计。此外,对话框设计工具一般将所有的界面元素均视为对象,系统提供一张表格以便定义或修改对象的属性。

(2)界面与目标软件其他部件的连接 界面自动生成工具支持开发人员针对每一命令按钮和不是父菜单的菜单项指定欲执行的过程或函数。从现在流行的用户界面生成工具来看,有两种办法实现界

*)本文得到国家863高技术项目基金资助。徐锡山 副教授,研究方向为人机交互、软件开发环境。陈火旺 教授,博导,研究方向为软件工程、人工智能。齐治昌 教授,博导,研究方向为软件工程、计算方法。

面部件与功能部件的连接:

①在界面编辑暂告完成后,由 CASE 工具根据界面的内容,自动生成源代码框架。源代码中以非常显著的标志(注解)说明开发人员应在何处填充相应于命令按钮和菜单项的过程或函数。

②在不退出界面编辑的状态下,由开发人员选取菜单项或命令按钮,当即实现相应的功能代码。

利用上述连接功能,还可将各界面元素按照开发人员的设计构想有机地联系起来。例如,在选中菜单项或命令按钮后弹出特定对话框,以及在选中命令按钮后改变同一对话框中列表框或编辑框的内容。

上述图形用户界面自动生成工具很好地支持了图形用户界面静态布局的自动生成(基于布局描述),而界面的动态行为(或称对话控制)仍需开发人员在一定对话控制框架下编程实现,这种编程的复杂性依赖于对话控制模型和相应的描述语言的复杂性,而且往往要根据系统规定的接口访问界面元素。为支持对话控制的自动生成,研究者们提出了各种对话控制的形式描述工具(如状态转换网络,上下文无关文法),但由于过于繁琐或描述能力的局限,使其难以用于实用系统的界面开发。

近些年,有些研究者研究基于某种数据模型规范或专用说明语言,通过定义正确的选择规则和布局规则实现界面对象的自动选择以及自动布局^{[2],[3]},但未涉及对话控制的规范和生成。

2 实现图形用户界面自动生成

2.1 基本思想

以上我们已经介绍,现有图形用户界面自动生成工具已支持自动生成界面静态布局并可从界面的布局描述生成对话控制框架。在这类对话控制框架下开发者的编程语言则是一般的高级语言(如 C、C++)。因为编程中需要涉及对界面元素的操作,开发者需熟悉界面元素的各种对外接口,而整个编程分别对应界面中的各类元素(如按钮、菜单项),开发者往往对用户界面的整体控制结构和控制机制缺乏理解和认识,这将直接影响到界面软件的质量和开发效率。因此,我们研究图形用户界面自动生成的目标将集中在对话控制的描述和生成。为了克服上述缺点,我们的基本思想是:为图形用户界面对话控制建立统一的模型和描述框架;基于该模型和描述框架支持从布局描述到对话控制的生成;基于该模型和描述框架寻求图形用户界面自动生成的新途径。

2.2 对话控制的多对象模型

我们把所有界面元素的对话控制部件都定义成如图 1 所示的多对象。所谓多对象,是指有多个协议分别对应不同的对象类。对话控制部件以不同的协议分别接收来自最终用户(Act)、应用程序和其它界面元素(Manage)的消息,是典型的多对象。因此,我们把图 1 称为对话控制的多对象模型。

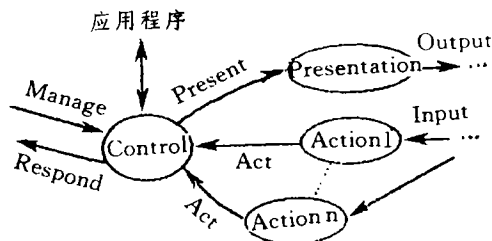


图 1 对话控制的多对象模型

该模型准确地刻画了对话控制的结构和通信关系,为对话控制的实现奠定了基础。

2.3 对话控制的描述框架

基于多对象模型和图形用户界面元素的特点,把对话控制部件分为以下六类:

- 1)命令 对应简单按钮(如 Windows 的 Push-Button)。
- 2)实体 对应需保持一定值的简单界面元素,如多值按钮、滑杆、文本编辑控制。
- 3)单项选择 对应菜单选择和列表选择中任何时候只允许选一项的情况。
- 4)多项选择 对应菜单选择和列表选择中可以同时有多项被选择的情况。
- 5)组合 对应把若干界面元素紧密组合在一起但又不具有独立的视窗功能的界面元素。如 Windows 的 GroupBox、ComboBox。
- 6)视窗 对应窗口、对话框等具有独立视窗功能(即有窗口管理器支持其各种界面操作,如移动、放大等)的界面元素。

我们选用 RAISE 的规范语言 RSL^[4]首先描述对话控制的公共成分,如通信机制、状态转换等,然后把前四类的每一类完全描述为 RSL 的一个 scheme。组合和视窗类依赖于所含的界面元素,无法给出具体的 RSL 语言描述,我们利用 RSL 语言的通道(channel)机制、并发机制以及很强的抽象能力和宽谱描述能力为组合和视窗的描述建立了很好的框架。

2.4 从布局描述生成对话控制框架

布局描述给出了用户界面所含的界面元素及元素的包含关系。根据界面元素和六类对话控制部件的对应关系,首先为对应前4类对话控制部件的界面元素自动生成对话控制部件,在此基础上为对应组合和视窗类对话控制部件的界面元素生成对话控制框架。

2.5 一种图形用户界面的自动生成方法

2.5.1 基本思想 基于布局描述之所以无法自动生成完整的对话控制部件,是因为布局描述没有反映对话控制部件所必需的用户界面动态行为信息及应用接口信息。从软件工程的观点,在用户界面的需求分析阶段,需要进行用户特性分析和任务分析,其结果是用户界面设计的依据。用户特性分析对应界面的友好性,而任务分析对应系统及界面的功能(指所能完成的系统任务)。所以,友好用户界面自动生成的技术前提之一是用户特性分析结果及任务分析结果的形式化。用户的个性差异及复杂的人的因素(Human Factors)使前者的形式化难以实现,而任务分析结果的形式化则是可能的。我们的思路是:寻求任务分析结果的适当形式化表示,基于此,自动生成其图形用户界面。用户界面的人的因素决定了自动生成的结果可以实现界面的功能,但无法保证界面的友好性。为此,可以把自动生成的结果看作图形用户界面的原型,以图形用户界面自动生成技术为核心,构造一个支持图形用户界面快速原型的CASE环境,实现友好图形用户界面的快速生成。

2.5.2 策略和算法

(1)任务分析结果的形式化表示。考虑一个交互系统具有传统的命令语言界面,最终用户通过一组命令模式来实现与系统的交互,每个命令模式规定了命令名称、功能、命令参数及返回值。如果把命令名称看作实现相应功能的函数,命令参数和返回值类型分别看作该函数的定义域和值域,那么,这一函数准确地刻画了实现相应命令模式的所需信息,包括用户输入、系统输出及与应用的接口。按照这一思路,我们可以把任务分析的结果形式地表示为一组RSL的函数(记为URS),每个函数为如下形式:

$$f: I_1 \times \dots \times I_n \rightarrow O$$

其中 f 为函数名,对应一个系统功能, $I_i(i=1, \dots, n)$ 为某一用户输入参数的类型, O 为系统输出值的类型。

实际上,这是通常命令语言界面设计所依赖的界面模型或称界面需求规范。我们将据此来生成图形用户界面。

(2)确定图形用户界面的基本界面元素。基于上述界面需求规范生成图形用户界面的前提是为界面

需求规范中每个函数的每个用户输入参数类型和系统输出值类型确定界面元素。定义用户输入参数的可能类型及所对应的界面元素和对话控制部件类型,任何系统输出值类型均对应实体显示界面元素和实体类对话控制部件。

(3)图形用户界面的生成算法。基于界面需求规范URS,图形用户界面的自动生成算法为:

①对URS中的每一个形如 $f: I_1 \times \dots \times I_n \rightarrow O$ 的函数,定义一个输入窗口和一个输出窗口,输入窗口包含所有 $I_i(i=1, \dots, n)$ 所对应的界面元素且包含一个‘OK’标准按钮,一个‘Cancel’标准按钮;输出窗口包含实体显示界面元素和一个‘Exit’标准按钮(表示关闭窗口)。按多对象模型生成输入窗口和输出窗口及所含界面元素的对话控制部件。

输入窗口的对话控制部件负责接收来自‘OK’和‘Cancel’按钮的选择消息,当接收到‘Cancel’按钮的选择消息时即刻关闭输入窗口,若接收到‘OK’按钮的选择消息,则从各参数对应的对话控制部件中取值并调用应用函数 f ,把函数结果作为消息发送给输出窗口。

输出窗口的对话控制部件负责接收来自相应输入窗口对话控制部件的消息,输出函数结果,当接收到‘Exit’按钮的选择消息则使本窗口消失。

②以URS中所有函数的函数名为内容定义一个命令菜单,生成一个单一选择类型的对话控制部件。

③为所有输入和输出窗口的对话控制部件和命令菜单对话控制部件生成一个上层对话控制部件,该部件通过各对话控制部件的manage通道协调命令菜单和输入、输出窗口的显隐。

结束语 上文所述的思想和方法为图形用户界面的自动生成提供了一条途径,经过RSL语言描述的实践证明是可行的。但是,实际应用系统的图形用户界面开发是非常复杂的,需要把形式方法和图视语言、CASE技术、原型技术相结合,以实现高质量界面的快速生成。

参考文献

- [1] B. A. Myers and M. B. Rosson, Survey on User Interface Programming, CHI'92 Conf. Proc., ACM Press, 1992
- [2] C. Janssen, et al., Generating User Interfaces from Data Models and Dialogue Net Specifications, INTERCHI'93 Conf. Proc., ACM Press, 1993
- [3] C. Wiecha, et al., ITS: A Tool for Rapidly Developing Interactive Applications, ACM Trans. on Information Systems. 8(3), 1990
- [4] C. George, et al., The RAISE Specification Language, Prentice Hall, 1992