

以交互为中心的 Post-WIMP 界面模型^{*}

秦严严⁺, 田 丰, 王晓春, 戴国忠

(中国科学院 软件研究所 人机交互技术与智能信息处理实验室,北京 100080)

An Interaction-Centered Hierarchical Post-WIMP User Interface Model

QIN Yan-Yan⁺, TIAN Feng, WANG Xiao-Chun, DAI Guo-Zhong

(Laboratory of Human-Computer Interaction and Intelligent Information Processing, Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

+ Corresponding author: Phn: +86-62540451 ext 8031, Fax: +86-10-62540451, E-mail: yanyan02@iscas.cn, <http://iel.iscas.ac.cn>

Qin YY, Tian F, Wang XC, Dai GZ. An interaction-centered hierarchical Post-WIMP user interface model. *Journal of Software*, 2006,17(4):691-702. <http://www.jos.org.cn/1000-9825/17/691.htm>

Abstract: Following the great improvement in interaction devices and software technologies, lots of novel interaction technologies which are based on Post-WIMP user interface are studied. It is crucial that the designer chooses the interaction component or the interaction technology properly and efficiently to adapt to the various interaction devices and environments in the user interface designing. According to the analysis of the interaction process and the design level of the Post-WIMP user interface, an interaction-Centered Hierarchical Post-WIMP User Interface Model is presented. It is described and analyzed from the aspects of the user, task, device, behaviors and entity in interaction. Moreover, a Post-WIMP user interface auto-generate tool is designed and implemented based on the model, to help the designers to insert those novel interaction technologies into the system flexibly, and to develop and evaluate the prototype quickly and efficiently. Experiments show that, the model can guide the process of user interface design, the implement of prototype, and the evaluation of the result of design.

Key words: Post-WIMP user interface; UI design; hierarchical model; interaction-centered

摘 要: 随着硬件设备和软件技术的发展,国内外开展了大量基于 Post-WIMP 界面的新型交互技术的研究。面对多种交互设备以及使用环境,在界面设计的过程中,根据上下文选择适当的交互组件或技术,并有效地进行组合与评估,成为构造 Post-WIMP 界面的关键问题。将界面设计与应用语义分离,使设计者能够灵活地置换各种交互技术。通过对 Post-WIMP 界面交互过程与界面设计层次的分析,建立起以交互为中心的分层 Post-WIMP 界面模型,将交互的各个层次进行分离。在模型的基础上,描述一个了 Post-WIMP 界面生成工具。借助该工具,设计者能够在设计过程中方便地引入新的交互技术,并能在最终的软件系统中加以灵活应用,从而可以快速、有效地进行界面原型实现和迭代评估。应用实例表明,Post-WIMP 模型的建立以及生成工具的实现,有利于设计者确定设

^{*} Supported by the National Natural Science Foundation of China under Grant No.60503054 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant No.2002CB312103 (国家重点基础研究发展规划(973)); the Key Innovation Project from Institute of Software "Research on Pen-Based Interactive System and Its Applications in Personal Office Environment", the Chinese Academy of Sciences (中国科学院软件研究所创新基金重大项目)

Received 2005-09-14; Accepted 2005-10-02

计方案及对方案的整体评估.

关键词: Post-WIMP 界面;界面设计;分层模型;以交互为中心

中图法分类号: TP391 文献标识码: A

随着个体用户处理信息的指数级增长,信息在网络以及移动计算设备等硬件上的跨设备、跨平台分布以及用户对计算机应用的技能、需求与期望的扩大,目前仍处于主流地位的 WIMP 界面形态已经无法满足界面设计、软件体系结构以及界面交互技术等方面的新的需要^[1].Post-WIMP 界面概念被提出以后,迅速出现了多种新型界面交互风格、交互设备及与之对应的交互技术.这些交互技术的产生极大地提高了人机交互的带宽,同时也为系统的设计与实现带来了相应的问题.

用户信息、使用环境、任务领域等知识资源在很大程度上影响着对用户交互的设计与实现.环境带来的任务需求多样化,导致在设计方案中多种交互技术的混合使用.这种情况往往会带来界面复杂性与交互复杂性的非线性增长,从而极大地增加了用户的认知负担^[2].在满足任务需求多样化的基础上,如何能够保证灵活的界面设计、快速的原型构造和方便的可用性评估,成为建立 Post-WIMP 界面的关键问题.另一方面,界面的分布性、直接操纵与实时反馈等特征的引入,极大地增加了界面控制和通信的复杂程度,给系统结构的修改与维护带来了困难^[3].

综上所述,Post-WIMP 界面设计的核心在于确定用户完成任务的交互方式和交互过程.为了满足 Post-WIMP 界面设计的要求,应将界面以交互为中心分为相互独立的层次,在各个层次内部保证一种“可置换性”,使实现细节不会影响到设计方案的改动,即在设计阶段保持界面交互和应用语义的分离.这就需要—个从较高层次对 Post-WIMP 界面进行抽象的层次模型,以指导界面的设计和开发过程.因此,本文通过对 Post-WIMP 界面的交互过程与设计过程的分析,提出了以交互为中心的分层 Post-WIMP 界面模型.利用该模型,进行 Post-WIMP 界面设计中设备、界面实体以及行为的抽象描述,从词法、语法和语义不同层次指导界面设计.界面模型为 Post-WIMP 界面的设计、原型实现及评估提供了完整的框架及界面描述.下面,分别就建模分析、模型的层次结构、基于该模型设计并实现的界面生成工具以及利用该工具进行设计开发的相应实例展开讨论.

1 相关工作

1.1 Post-WIMP界面研究

研究表明,计算机硬件设备的发展与计算机的广泛普及暴露出传统 WIMP 界面的诸多缺陷.Nielsen 认为:在下一代用户界面中,用户将会更关注于任务本身,而非如何操纵计算机来完成任任务^[4];van Dam 在分析 WIMP 界面缺陷的同时,也提出了 Post-WIMP 的概念,用以提高人机交互的带宽,使交互过程更为自然^[5].从 20 世纪 90 年代起,以笔式用户界面、多通道用户界面等为代表的 Post-WIMP 界面迅速成为了国内外研究的热点^[6].

国内外多名学者通过对 Post-WIMP 界面与传统 WIMP 界面的研究,总结了 Post-WIMP 界面在交互方面与 WIMP 界面的不同^[5,7,8](见表 1).

Table 1 The interaction characteristic between WIMP user interface and Post-WIMP user interface

表 1 WIMP 界面与 Post-WIMP 界面交互特征比较

Interaction characteristic	User interface	WIMP user interface	Post-WIMP user interface
Continuous		Discrete event	Discrete event+continuous event
Process sequence		Step-By-Step	Synchronous
Device		Keyboard+mouse	Multi-Modal device
Time sequence		Offline process events without time-sequence	Online process events with time-sequence, Real-Time feedback
Probabilistic interaction		Precise tokens	Probabilistic input

在交互特征与界面范式等 Post-WIMP 界面研究的基础上,国内外多家机构研究开发了许多新型交互组件

或交互技术,如 Translucent Patch, Tool Glasses, ZUI, Pie Menu^[9-12].

然而从实用性的角度来看,尽管单一的交互技术能够起到减轻用户认知负担的作用,但在实际系统中,要将这些交互技术按照任务需求结合在一起,往往会导致界面复杂性及交互复杂性的非线性增长,反而极大地增加了用户的认知负担.这也是多种新型交互技术无法得到真正实用的主要原因.本文通过建立以交互为中心的层次 Post-WIMP 界面模型,用于保证交互组件与交互行为的独立性,为各种交互技术提供嵌入系统的框架描述.

1.2 基于模型的界面生成方法

基于模型的方法和工具通过向交互式系统设计提供一种更高层次的方式,进行交互模型和体系结构模型的连接.Brad Myers 认为:基于模型的界面设计方法的一个重要目标是提供了设备无关性,这在 Post-WIMP 界面多种交互设备与交互技术并存中显得尤为重要^[13].目前,在对 Post-WIMP 界面的研究中,基于模型的设计方法及工具得到了大量的应用:Beaudouin-Lafon 将界面交互组件以及相应的交互动作抽象为交互模型^[14],基于该模型设计了一个可嵌入 Post-WIMP 交互组件的图形用户界面体系结构并进行了相应实现^[15];Erwin Cuppens 和 Karin Coninx 通过对 3D 虚拟交互环境中交互任务的建模,实现了一个支持多设备的 3D 虚拟环境生成工具 CoGenIVE^[16];Jullien Bouchet 设计了框架结构 ICARE,使用概念组件模型(conceptual component model)描述界面中的交互组件,并利用代码生成技术进行多通道用户界面的快速建立^[17].

在基于模型的设计方法得到大量应用的同时,我们注意到:多数研究的建模都是在设备或界面交互组件的层次上进行,而仅仅将交互行为本身视为交互组件的上下文状态转换.Foley^[18]提出:人机交互技术可以定义为使用物理输入/输出设备在对话过程中执行一个任务的方式.对于相同的任务,可以存在许多不同的交互方式.这种交互方式的不同,不仅体现在交互设备、交互组件的差异上,也可以表现为具有不同的交互行为.实际上,将交互行为组件化,与交互组件进行分离,有利于软件重用.交互行为的组件化,使设计者能够在系统设计时,为同一交互组件选择不同的交互行为,使界面的设计更加灵活.

本文通过对 Post-WIMP 界面的建模,不仅将设备和界面交互实体进行了分离,同时还将交互行为单独作为一个层次进行描述,并提供了交互行为与交互组件之间的连接框架.使设计者在设计阶段不用关心交互行为的具体实现,从而方便地将交互行为嵌入到系统设计方案中.

2 Post-WIMP 界面模型

用户使用软件进行工作的目的是完成某个特定目标(goal).其过程是:用户与软件的设备硬件载体进行交互,完成任务,并从设备上得到任务完成的感知反馈.

Norman 提出的界面交互框架模型给出了交互的 4 个主要部分及它们之间的转化关系^[19],如图 1 所示.用户将完成的目标任务与领域知识相结合,转化为对系统界面的动作行为序列,形成界面输入.输入设备得到输入信息后,将其根据当前系统的状态解释为系统理解的交互行为,并由界面实体执行该行为所对应的任务操作,给出相应的结果反馈.最后,用户通过观察来评估系统的执行结果是否满足自己的期望.

从设计过程来看,van Dam^[20]将 Post-WIMP 用户界面设计分为概念层(conceptual level)、功能/语义层(functional/semantic level)、顺序/语法层(sequencing/syntactical level)以及绑定/词法层(binding/lexical level)等层次,并提出了相应的设计步骤.其中:概念层和功能层面向特定应用,形成界面的应用语义;连续层和绑定层面向界面组成,包括设备输入/输出、界面外观等.从软件的设计阶段来看,分析过程(analyze)用于确定需求信息、用户信息以及软件使用的环境信息等;在形式化过程中定义设计的目标,即确定任务;在综合(synthesize)设计过程中,设计者需要根据真实世界的交互方式确定软件交互隐喻、领域对象信息、交互设备、交互对象以及交互行为.

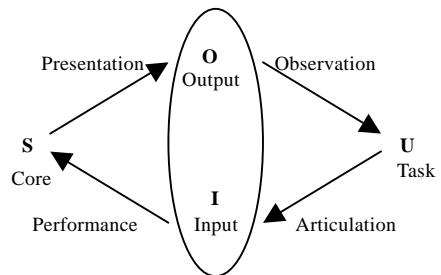


Fig.1 The interaction process model

图 1 交互过程模型

从交互过程和设计过程来看,Post-WIMP 界面设计应该由用户出发,根据领域知识以及任务目标,确定交互的各个组成部分.界面设计的目的是从多种交互技术中选择出一种符合用户习惯、满足任务需要的组合.其中,交互技术的选择是根据交互的词法、语法和语义分层次进行的.

2.1 模型的建立

Beaudouin 认为:交互模型是一系列指导界面设计的原则、规律和属性,用于描述如何将各种交互技术有意义和一致地合并到一起,并能从用户的角度来定义交互的 look and feel^[1].交互模型的属性可以被用于某些特定交互设计的评估.为了在 Post-WIMP 界面设计的同时,能够更好地满足用户的可用性需求、体现界面的交互特征,本文通过上述对交互及设计过程的分析,进行界面模型的建立.

由于每个阶段的设计都是一个反复迭代、反复尝试的过程,因此,在设计过程中需要一个与界面设计层次对应的分层界面模型描述支持,以提高界面设计的效率.同时,为设计方案的原型实现、测试及评估提供良好的框架环境.我们根据以上的界面层次划分以及设计过程,设计了与之对应的模型层次结构.

如图 2 所示,Post-WIMP 界面模型分为几个层次:User model 提供了用户的基本信息;Task model 则用于描述特定应用的任务信息以及完成任务的相关领域知识;Device model, Entity model 和 Behavior model 统一为交互模型,是界面模型的核心.

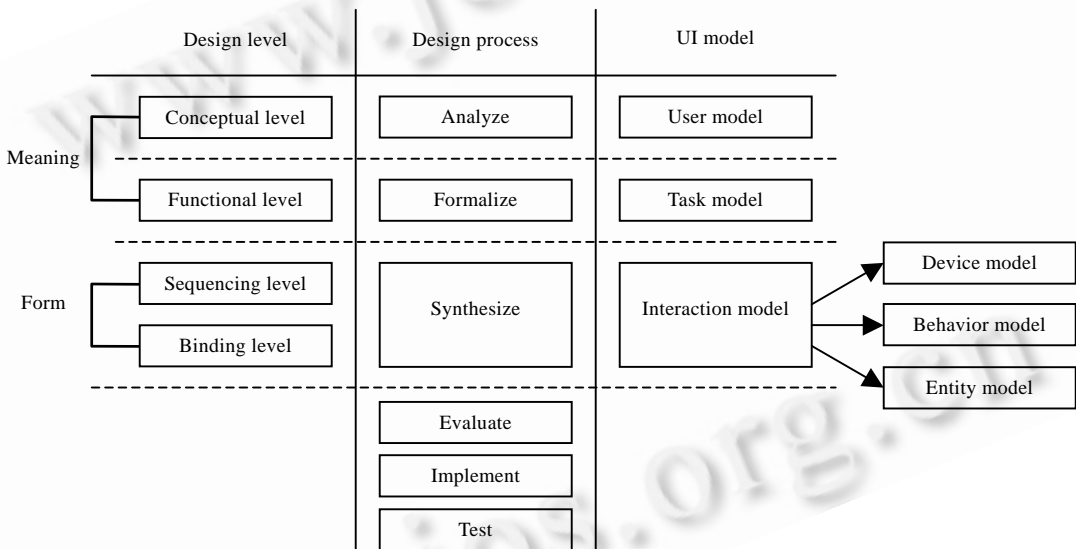


Fig.2 The structure of Post-WIMP user interface model

图 2 Post-WIMP 界面模型结构

2.2 用户模型(user model)

用户模型用于提供用户的基本信息.描述用户的信息可以根据设计用途分为 3 类:一是用户的个人信息,如用户的姓名、电话、邮件地址等.由于 Post-WIMP 界面在移动设备或无处不在环境中的大量应用,使得系统在完成用户任务的过程中,往往需要通过用户基本信息或当时环境条件来主动获取上下文信息.用户模型对用户基本信息的描述,提供了提取上下文环境信息的数据;二是应用人群信息.该信息描述了面向特定领域的用户信息,如用户在完成任务时常处的环境、用户在任务过程中的地位等.这类信息提供用户完成任务流程、操作、环境时的相关信息给设计者,使设计人员能够根据用户工作的实际情况选择界面的设计方案,并使最后的设计结果真正满足用户的可用性需求;三是用户与系统的连接信息.由于 Post-WIMP 界面交互的多维性特征,很多系统中的交互源并不一定是使用者自身,也有可能是其他软件系统或智能代理.此类描述信息被用于设计者了解该类用户与本系统之间的关联协议、数据输入/输出方法等,以便进行系统功能设计.用户模型描述以属性-文字值配对描述为主,如职业-教师等.

2.3 任务模型(task model)

任务模型用于描述用户在使用系统过程中所要完成的目标以及完成目标所处的环境.该模型所描述的信息为界面设计提供了不同层次的指导.对于设备层,任务模型提供的环境信息使设计者在了解任务环境的同时,能够初步确定系统所选用的硬件设备;对于实体层,任务将被分解为子任务,从而与界面实体提供的各项服务(services)进行映射,以确定设计中选择的界面实体是否可以满足(compatible)应用任务的要求.如图 3 所示.

Post-WIMP 界面的交互多维性导致系统往往会出现任务处理的同步进行.同样,在任务模型中也允许多个相同任务同时出现.因此,界面实体是否能够满足任务的需要,也 与 该 实 体 是 否 能 够 具 有 提 供 同 步 服 务 的 能 力 有 关,如 图 3 中 Entity1 支持同时提供两个 Service1 来分别满足 Sub-Task1.此时,才能够确定 Entity1 的确可以满足 Task1 的要求.

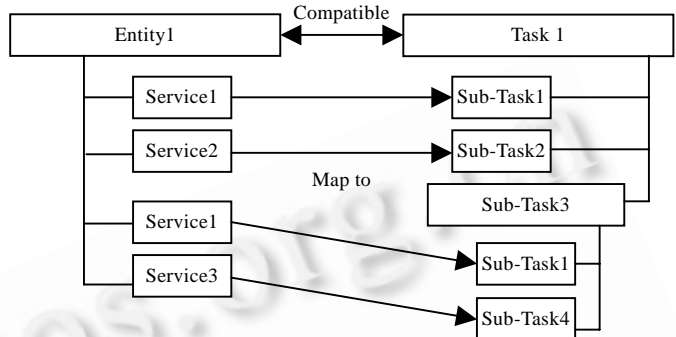


Fig.3 The map between tasks and entity services

图 3 任务与界面实体服务的映射关系

2.4 设备模型(device model)

Erwin 认为:交互式系统一方面向着多设备(手持设备和嵌入式设备)系统发展;另一方面,则进化成为多种交互技术融合的交互虚拟环境^[21].因此,将设备与抽象的交互实体及应用任务分离出来,对于 Post-WIMP 界面的设计是有必要的.

硬件设备是用户与系统进行交互的接口,不同的设备将会产生不同类型的交互信息,存在不同的信息呈现方式以及不同的界面实体布局.Post-WIMP 界面的发展过程中出现了多种交互设备.设备模型描述了设备所提供的功能、适用范围等信息,为软件界面设计提供指导(描述设备所产生和接受的消息、参数以及界面布局和信息呈现方式等人机工效学(ergonomics)指南).该模型描述主要是以功能手册和设计指南的方式给出.

2.5 交互行为模型

用户在操作界面实体完成任务时,通过不同的交互行为进行(如图 4 所示).完成不同的任务可以使用相同的交互行为.同样,不同的交互行为也可以用于实现相同的任务.在一个三维场景中实现 Select 任务,设计者可以

设计鼠标的点击、数据手套的抓取以笔的圈选或是点击行为进行.而用笔划圈的行为,在系统中可以被解释为物体的选中操作,也可以是创建一个圆或球体的手势.设计者可以根据用户习惯、选用设备、完成任务的效率等指标选择完成任务的交互行为.

从 Post-WIMP 界面交互任务的产生方式来看,由于 Post-WIMP 界面的交互设计使界面的交互过程从面向对象的过程朝着面向用户/任务的过程发展.因此,系统会自动根据用户当前的上下文状态进行用户交互动作的解释.交互行为组成的基本单位是由界面实体传递的交互事件,多个交互事件构成交互动作,多个交互动作形成交互行为,行为之间也允许有嵌套的结构.以笔交互中的手势为例,一个两笔交互的叉号手势的生成首先分解为两个笔划动作,其中每一个笔划动作

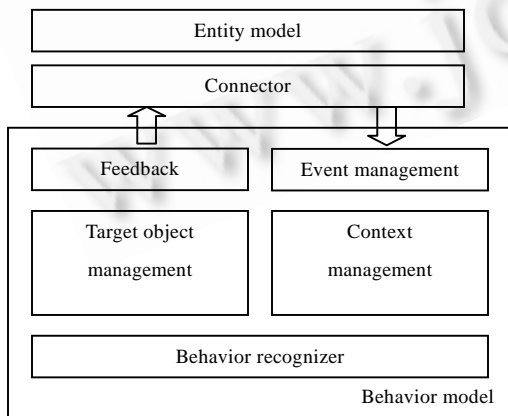


Fig.4 Post-WIMP UI behavior model

图 4 Post-WIMP 用户界面行为模型

都分解为一个笔按下事件,多个笔移动事件和一个笔抬起事件.这些事件通过一定的词法逻辑形成交互动作,交互动作根据给定的语法结构形成交互行为.

如图 4 所示,交互行为的结构主要分为事件管理、反馈、上下文管理、目标对象管理、行为识别等模块.

交互行为的内部结构及实现被封装在上下文管理以及目标对象管理模块中,通过连接器与界面实体进行信息传递.设计者将不必关心交互行为的具体实现,而只需在设计过程中指定交互行为的发生状态以及交互行为所对应的任务语义,即可将交互行为嵌入到界面设计及系统原型中.

当界面实体向交互行为发送事件时,由事件管理模块将统一结构的事件转化为行为所能理解的消息结构,进入到行为的上下文状态转换中.根据行为的语法结构组织并将其发往行为识别模块,以确定行为产生的可能.在行为形成的过程中,往往要利用到外界的上下文信息或进行语法的反馈,反馈模块负责与界面实体进行信息交流.最主要的 3 种交流信息类别是反馈(feedback)、信息请求(require)以及识别结果(recognize result).

2.6 界面实体模型

界面实体是用户通过设备来完成工作的工具,是交互任务的执行者.实体将目标任务分解映射到自身的服务,以设备所传递的事件信息作为输入,判断当前发生的交互行为,分析交互行为所对应的交互任务和应进行的服务.同时改变系统运行状态,进行内在数据管理,并给出任务完成的用户感知反馈.用户进行交互的过程就是用户通过设备在实体上操纵领域对象的过程如图 5 所示.

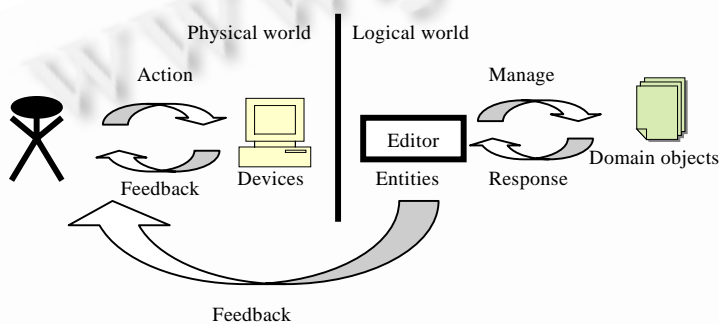


Fig.5 The interaction process between users and domain objects

图 5 用户与领域对象的交互过程

从对领域对象操作的直接程度来看,界面实体可以分为 3 类:直接操作实体、间接操作实体和反馈实体.直接操作实体是领域对象的载体,如放置文本的文本编辑器、承载三维实体的窗口等;间接操作实体的行为操作目标仍然是领域对象,然而却是通过影响直接操作实体来进行任务的完成,如改变文字属性的对话框、功能菜单等;而反馈实体并不

会改变系统内部的数据,它的作用主要是改变界面的反馈效果,如 Fish Eyes,语音输出控件等.

在界面设计过程中,面向特定设备满足同一任务的界面实体存在多种选择.而 Post-WIMP 界面交互行为的多样性,使得面对不同的用户或任务,在同样的界面实体上往往存在不同的交互行为选择.为了使界面设计更加灵活有效、实现实体与设备之间的独立性,实体与交互动作之间的分离成为关键问题.

如图 6 所示,界面实体模型主要包括事件处理、上下文信息管理、反馈、行为管理、服务管理以及数据管理模块.在系统运行时,由设备发送的消息经由连接层转化为统一的界面事件,并发送到界面实体的事件管理模块.通过上下文信息管理模块,根据当前状态与行为管理模块中的行为信息,判断由设备传来的事件是否会引起某个交互行为(该行为的描述在设计阶段已经登记在界面实体的行为管理模块,交互行为可以是界面实体缺省或是由用户定制的).当系统确定事件触发了交互行为时,事件管理模块将该事件及其后同源事件发送到行为模型的输入部分进行交互行为解释,并在行为结束时从交互模型得到行为类型和参数判定.由于 Post-WIMP 交互存在模糊性,常常使得实体无法确定发生的交互行为类别.因此,在行为管理模块判断发生交互行为类别或交互对象的二义性后,将给出相应的决策方案或是交给上下文模块进入用户决策状态.交互行为类型被确定后,界面实体将根据服务管理模块中交互行为与服务的映射情况决定应当提供的服务.服务具体体现在实体自身属性管理、子实体管理与目标对象管理之中.服务的反馈同样以事件的形式发送到连接器,最终传递到设备层.反馈事件既包括面向用户的感知反馈,也包括向其他界面实体发送的信息.

界面实体模型与设备模型之间通过统一的事件结构进行通信,利用连接器将不同的设备信息(如鼠标、键

盘、语音、笔)转化为标准的参数化结构事件,将设备与界面实体分离.同样,实体与交互之间根据统一的事件描述结构进行连接,从而实现界面实体与交互行为之间的分离.

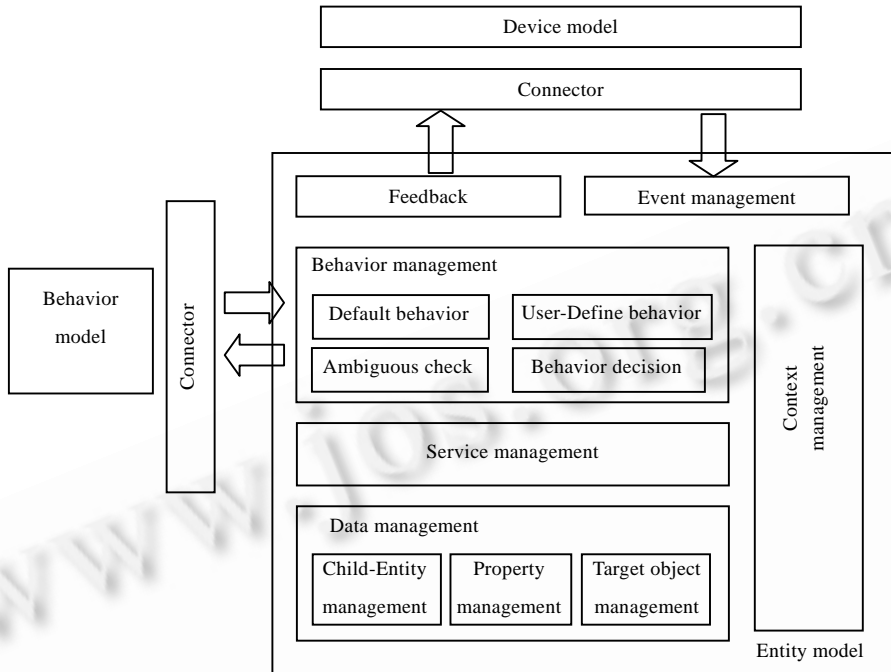


Fig.6 Post-WIMP UI entity model

图 6 Post-WIMP 用户界面构件实体模型

3 基于界面模型的 Post-WIMP 界面设计

3.1 Post-WIMP界面设计原则

交互设计的过程是由用户及用户的目标任务出发,确定系统的硬件载体、界面实体及其布局、用户与系统界面的交互方式、系统内部的数据管理以及各部分之间的连接.进行界面设计时,在交互过程中各部分之间作出组合方案选择是在方便用户使用和满足应用需求的前提下进行的.因此,界面模型中用户模型和任务模型的主要作用是为界面设计提供良好的指导.如图 7 所示.

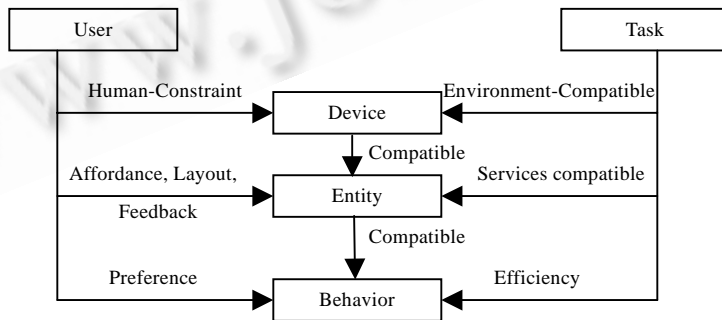


Fig.7 The principle of the choices between the interaction model

图 7 交互模型组合方案选择准则

从用户的角度来看,界面设计中的设备(device)选择首先要满足人机工效学原理,以确保选用设备可以被用

户使用,尤其是应用面向特殊人群时,Post-WIMP 界面中交互的隐性特征,使设备上提供用户显式信息的实体(entity)需要指示用户如何进行操作,给出交互的提示(affordance);使用环境或使用设备不同(如白板与 Tablet PC 等),需要界面提供合理的实体布局(layout)和信息呈现方式;用户在交互过程中对产生结果的反馈要求,也使得界面实体需要提供不同层次上的实时反馈(real-time feedback),满足 Post-WIMP 界面的实时特征.行为动作(behavior)的选择与用户完成任务的偏好相关,如在进行文本编辑时的手势、进行语音交互的关键词设置等.

从任务的角度看,设备的选择需要满足交互任务发生的环境,如是否处于移动状态、是否处于强噪音场所.而设备上的界面实体应当能够提供用户完成任务所需的服务,如用户需要进行文字编辑,承载文字的界面实体就应该提供相应的文字编辑服务.而对于 Post-WIMP 交互的多维性特征,也要求界面设计中提供可以同时进行不同信息处理的功能实体.由于同一任务可以有不同的行为完成,因此,行为设计选择的标准主要是完成任务的效率.在认知心理学上,有多种较为成熟的评估方法进行交互效率的衡量.然而,最终的设计方案是由完成某一任务所有交互动作组合的效率来决定的.

3.2 交互模型中各层次在设计中的连接

交互模型的层次结构将交互设备、交互行为以及界面实体相互分离,以保证同层次组件的可置换性.在设计中,设计者需要明确各层次之间的连接方式,包括消息转换与信息传递机制等,以便在系统运行时实现连接.

在设计中,设备模型与界面实体模型之间主要存在两方面的连接:一是输入/输出信息的格式转换,将设备模型对输入/输出信息的描述统一转换为规定结构的事件描述,转换的规则可以由设计者或开发者给出;二是设备模型对界面实体的布局等静态属性的限制或指南,将引导设计者对界面实体的选择方案.

交互行为模型和界面实体模型的连接是保证实体和交互行为对话独立性的核心.设计者将交互行为引入到界面设计中,需要明确几个关键问题:

(1) 交互行为的目标(which),即明确用户动作所操作的目的对象.在设计阶段,设计者必须要指定用户做出该行为时的目标对象,即为行为模型中的目标对象(target object)赋值,赋值的结果将由界面实体中的行为管理模块保存,并在界面运行时赋予具体含义;

(2) 交互行为的发生时间(when),即交互动作开始进行的系统状态描述.描述是在界面实体的行为管理模块中进行的.在运行时,通过实体的上下文管理模块判断是否满足该状态,并开始向所指定的行为组件发送交互事件.与传统的 WIMP 界面行为不同,在 Post-WIMP 界面中,交互行为往往没有明确的开始或结束状态.如在眼动输入的系统,交互行为自始至终都保持在进行状态.此时,只能通过交互行为的识别器来判断交互是否发生.交互行为的发生时间就是系统初始化完毕、开始接收设备事件信息的时间;

(3) 交互行为的发生结果(what),用户进行交互的目的是完成某个具体任务.由于应用任务在系统设计时被映射为界面实体的服务,因此,交互行为也被对应到界面实体的特定服务上.设计阶段,这种映射关系将被保存在界面实体的行为管理模块中.运行时,则将行为所对应的结果体现为服务管理模块中的特定服务;

(4) 交互行为的产生过程(how),即交互行为的组成以及形成过程.形成过程将由交互行为的内部结构以及在行为进行中的反馈得以体现.从 Post-WIMP 界面交互连续性特征来看,交互行为所产生的效果不仅仅是在全部行为的语法结构形成后才体现出来,在交互行为的形成过程中,同样也会引起特定的反馈.如在笔交互界面的文字编辑软件 GestureWord 中,文字的选中手势在系统还没有得到笔抬起事件时,通过对用户操作的增量式理解,动态地进行了手势作用范围内文字的选中服务.在面向界面设计的行为模型中,内部结构将不需要进行描述,行为进行中的反馈以及行为识别的结果将形成标准的界面实体事件,反馈回相应的界面实体中.

下面,以笔式文本编辑软件中多折线(draw_zigzagline)及圈选(draw_circle)手势行为为例,说明在运行时的数据流程.如图 8 所示.其中,圈选手势行为的交互行为 XML 描述如下:

(Behavior description)

```
<Name value="Circle_Gesture_Behavior"/>
```

```
<Type value="Pen_Gesture"/>
```

```
<SrcEntity value="TextEditor1"/>
```



```

<TargetObject value="Text"/>
<TargetService value="TextSelect">
<Recognizer value="Pen_Gesture_Recognizer"/>
<Action>
  <StartEvent value="Pen_Down"/>
  <ContinuousEvent value="Pen_Move"/>
  <EndEvent value="Pen_Up"/>
</Action>
</Behavior description>

```

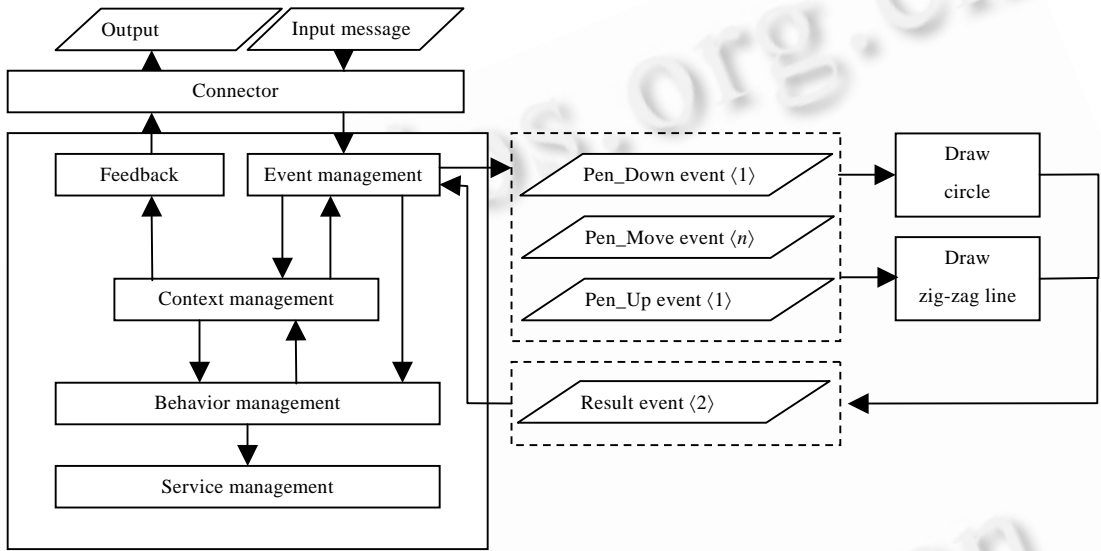


Fig.8 The connection between the behavior model and the entity model in Post-WIMP UI model

图8 Post-WIMP用户界面交互行为模型与实体模型的连接

通过改变行为的 XML 描述,用户可以在设计时指定行为作用的对象、所对应的服务以及行为产生和结束的事件等,从而有效地将行为描述加入到整体设计中。

当笔式文本编辑实体接收到笔输入信息后,由连接器转化为统一的实体事件,根据当前的上下文状态查找行为管理模块中可以开始或正在进行的行为,并将其需要的事件类型发送到该行为中。Draw_Circle 和 Draw_ZigzagLine 都是以 Pen_Down 事件开始并以 Pen_Up 事件结束。在上下文管理模块确定 Pen_Down 发生时,由事件管理模块分别向两个行为实体发送相应的事件。在 Pen_Up 事件发生后,两个行为实体将根据已经接收到的事件进行词法结构的识别,将识别结果构成标准的结果事件(result event)返回到文本编辑实体的事件管理模块中,并由行为管理模块和上下文管理模块共同进行决策,决定系统将进行的交互行为结果。

3.3 Post-WIMP界面生成工具

在建立界面模型的基础上,我们设计并实现了一个 Post-WIMP 界面生成工具,该工具以 Post-WIMP 界面模型为基础,采用以文档为中心的构件技术进行界面生成。用户界面的生成分为设计(design)阶段与运行时(runtime)阶段,两个阶段之间采用基于 XML 的界面描述文档进行连接。

界面描述文档分为界面场景树描述、单一场景描述以及界面实体描述 3 个层次。界面场景树从系统全局的角度描述界面场景之间的关系;单一场景描述反映了该场景的界面布局,包括组成场景的界面实体类型及实体之间的包含关系;界面实体描述反映界面实体的属性与特征,包括实体静态属性和动态行为描述。界面描述文档都基于声明性的语言,仅仅给出了软件系统的抽象描述,具体实现仍然通过映射文档对应到相应的实现技术^[22]。

由于设备模型与界面实体模型的数据转换细节不在设计阶段体现,因此,界面设计阶段的主要目标是确定界面实体、交互行为组件及其之间的连接.用户界面的设计阶段与界面描述文档的层次对应,分别对场景树、单一场景(如图 9(a)所示)与界面实体(如图 9(b)所示)进行设计.我们提供了一个界面设计工具,在界面的制作过程中,设计者可以通过该工具根据用户意见或系统需要进行不同界面实体或实体行为的设计尝试(如图 9(c)所示).设计完成后,界面设计工具能够自行生成界面描述文档(如图 9(d)所示).同时,工具支持界面描述文档在代码级上的二次开发,用户可以针对实体不同事件进行 C++代码的填写.在运行时,生成工具自动将代码生成成为类文件并编译执行,充分保证了界面设计的灵活性和系统的可扩展性.



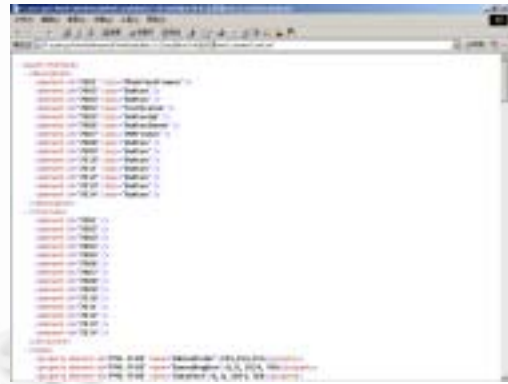
(a) The design of the single scenario
(a) 单一场景的设计



(b) The design of the style of entity
(b) 界面实体的风格设计



(c) The connection between the behavior and the entity
(c) 交互行为与界面实体连接



(d) The XML description of the user interface
(d) 基于XML的界面描述文档

Fig.9
图9

在设计过程中,设计者可以选择不同的界面实体,设计工具的属性表显示提供了实体的静态属性以及实体可接收的事件.如设计者为实体设计了新的交互行为,则可以通过实体事件确定交互行为的开始状态、目标对象及对应服务.图 9(c)中,设计者将实体接收的 LButtonDown 事件作为行为的初始条件,该事件将作为多折线手势以及圈手势两个交互行为的开始.同时,设计者通过填写 XML 结构来描述行为的目标对象与对应服务,以完成对交互行为的接入.

界面的运行时阶段通过界面解释工具进行.解释工具在系统初始化时,根据界面描述文档构建各个场景;根据映射文档确定界面实体及行为组件的类型、创建相应实体或行为组件;并依据描述文档配置界面布局、界面实体的静态属性以及行为组件与界面实体的连接.在系统运行时,显示工具动态地将设备发送的消息统一为界面事件并传送给界面实体的连接层,由系统上下文管理器决定消息的发送目的,形成交互行为,完成交互任务.

4 应用实例的设计和评估

我们通过界面生成工具进行了应用系统的开发.笔式会议系统是一个面向电子办公领域的笔式用户界面软件,用户通过笔与界面进行交互.笔式界面是 Post-WIMP 界面的一种重要形态.由于纸笔交互的自然性,导致交互的连续性与隐性特征在笔式用户界面中得到了更多的体现.手势(gesture)是笔式用户界面中主要的人机交互方式之一,是将人手持笔进行交互时,笔在“纸面”移动的轨迹记录下来,并且赋予每个具有一定特征的手势以一定的交互含义而形成的交互方式.在笔式会议系统中存在大量的手势交互.如何让用户自然地、符合日常书写习惯地进行交互,完成工作;如何在用户的使用过程中降低用户的记忆负担,都是手势行为设计的关键问题.通过界面生成工具,设计者可以快速地将不同手势描述嵌入到系统,并从系统全局的角度进行整体手势设计方案的评估.如图 10 所示.



Fig.10

图 10

进行系统设计后,我们就不同的设计方案与北京师范大学心理学院合作,对会议系统进行了专家评估.评估结果显示:评估者对不同设计方案中的易学性、易用性等可用性评测结果有很大差异,达到显著性水平.使用界面生成工具,极大地提高了界面设计的效率,加快了原型设计和评估过程的迭代速度.

5 总 结

相对于传统的 WIMP 界面,Post-WIMP 界面以其自然直接的交互、较大的交互带宽以及对多种环境的适应能力,成为目前人机交互的研究热点.本文通过对 Post-WIMP 界面交互过程与设计过程的分析,建立了以交互为中心的层次 Post-WIMP 界面模型.该模型充分考虑了 Post-WIMP 界面的交互特征,使设计者在设计过程中不必关心设备实体与交互行为的连接实现,能够灵活地进行组件选择,进行快速原型构建,并进行不同设计方案的可用性评估.在模型的基础上,我们设计并实现了面向 Post-WIMP 界面的界面生成工具,并通过该工具进行了相应的应用实例的设计与开发.从设计过程可以看出:基于界面模型的生成工具可以协助设计者迅速完成界面静态布局和动态行为的设计,并为系统原型的评估提供了良好的框架环境.

Post-WIMP 界面模型是随着软硬件技术的发展而不断更新完善的.随着在人机交互领域研究的不断深入,我们将进一步扩充和完善该模型,并对模型中各个部分制订更为详细的描述规范.相应地,我们也将对界面生成工具进行改进,以使设计者能够利用该工具更快地进行界面设计、原型创建与方案评估.

References:

- [1] Beaudouin-Lafon M. Designing interaction, not interfaces. In: Costabile MF, ed. Proc. of the Conf. on Advanced Visual Interfaces, AVI 2004. Gallipoli: ACM Press, 2004. 15–22.
- [2] Hua QY. A 3D visual paradigms of the object-oriented system and its implement. Chinese Journal of Computers, 1997,20(9): 775–781 (in Chinese with English abstract).
- [3] Hartson RH. User-Interface management control and communication. IEEE Software, 1989,26(1): 62–70.
- [4] Nielsen J. Noncommand user interfaces. Communications of the ACM, 1993,36(4):83–99.
- [5] van Dam A. Post-WIMP user interface. Communications of the ACM, 1997,40(2):63–67.

- [6] Dong SH, Wang J, Dai GZ. Human-Computer Interaction and Multi Model User Interface. Beijing: Science Press, 1999 (in Chinese).
- [7] Jacob RJK, Deligiannidis L, Morrison S. A software model and specification language for non-WIMP user interfaces. ACM Trans. on Computer-Human Interaction, 1999,6(1):1-46.
- [8] Green M, Jacob R. Software architectures and metaphors for non-WIMP user interfaces. Computer Graphics, 1991,25(3):229-235.
- [9] Kramer A. Translucent patches: Dissolving windows. In: Szekely P, ed. Proc. of the UIST'94: ACM SIGGRAPH Symp. on User Interface Software and Technology. New York: ACM Press, 1994. 121-130.
- [10] Bier E, Stone M, Pier K, Buxton W, De Rose T. Toolglass and magic lenses : The see-through interface. In: Whitton MC, ed. Proc. of the ACM SIGGRAPH. New York: ACM Press, 1993. 73-80.
- [11] Bederson BB, Meyer J, Good L. Jazz: An extensible zoomable user interface graphics toolkit in Java. In: Ackerman M, ed. Proc. of the ACM Symp. on User Interface Software and Technology: UIST 2000. San Diego: ACM Press, 2000. 171-180.
- [12] Kurtenbach G, Buxton W. Issues in combining marking and direct manipulation techniques. In: Rhyne JR, ed. Proc. of the UIST'91. New York: ACM Press, 1991. 137-144.
- [13] Myers B, Hudson SE, Pausch R. Past, present and future of user interface software tools. ACM Trans. on Computer-Human Interaction, 2000,7(1):3-28.
- [14] Beaudouin-Lafon M. Instrumental interaction: An interaction model for designing Post-WIMP user interfaces. In: Turner T, ed. Proc. of the ACM CHI 2000. New York: ACM Press, 2000. 446-453.
- [15] Beaudouin-Lafon M, Lassen M. The architecture and implementation of CPN2000, a Post-WIMP graphical application. In: Ackerman M, ed. Proc. of the ACM UIST 2000. New York: ACM Press, 2000. 181-190.
- [16] Cuppens E, Coninx K. CoGenIVE: Code generation for interactive virtual environments. In: CHI 2005 Workshop on The Future of User Interface Design Tools. 2005. <http://hci.stanford.edu/srk/chi05-ui-tools/>
- [17] Bouchet J, Nigay L. ICARE: A component-based approach for the design and development of multimodal interfaces. Extended Abstracts CHI 2004. 2004. 1325-1328.
- [18] Foley JD, van Dam A, Feiner SK, Hughes JF. Computer Graphics: Principles and Practice in C. 2nd ed. Reading: Addison-Wesley, 1995.
- [19] Norman DA. The Psychology of Everyday Things. New York: Basic Books, 1988.
- [20] van Dam A. User interface I: Introduction to Computer Graphics. 2000. <http://www.cs.brown.edu/courses/cs123/lectures/Introduction.pdf>
- [21] De Boeck J, Cuppens E, De Weyer T, Raymaekers C, Coninx K. Multisensory interaction metaphors with haptics and proprioception in virtual environments. In: ACM Int'l Conf. Proceeding Series; Vol. 82. Proc. of the 3rd Nordic Conf. on Human-Computer Interaction. 2004. 189-197.
- [22] Hua QY, Wang H, Muscogiuri C, Niederée C, Hemmje M. A UCD method for modeling software architecture. In: Dai GZ, ed. Proc. of the APCHI. 2002,2:729-743.

附中中文参考文献:

- [2] 华庆一. 一个面向对象系统的三维可视范型及实现. 计算机学报, 1997, 20(9): 775-781.
- [6] 董士海, 王坚, 戴国忠. 人机交互和多通道用户界面. 北京: 科学出版社, 1999.



秦严严(1977 -),男,北京人,博士生,主要研究领域为人机交互技术,笔式计算.



王晓春(1976 -),男,博士生,主要研究领域为人机交互技术,笔式计算.



田丰(1976 -),男,博士,主要研究领域为人机交互技术,虚拟现实.



戴国忠(1944 -),男,研究员,博士生导师,CCF高级会员,主要研究领域为人机交互技术,计算机图形学.