

# 一种用户目标驱动的交互设计方法

张立臣, 毛庆, 冯德民, 王映辉

(陕西师范大学 计算机学院, 陕西 西安 710062)

**摘要:**面向对象的软件工程是面向对象技术在软件工程领域的全面运用,提高了软件的可修改性和可维护性,但软件的可用性并未得到有效解决。提出了以用户目标为中心,用户目标驱动功能的开发和界面交互设计的思想;描述了交互设计与面向对象设计的结合方法;划分了交互设计师和建模人员的职责。在保证软件可维护性的同时提高了软件的可用性。

**关键词:**可用性; 人机交互; 用户目标驱动; 交互设计

**中图分类号:** TP311.11

**文献标识码:** A

**文章编号:** 1005-3751(2006)01-0193-03

## An Approach of User-Target Driven Interaction Design

ZHANG Li-chen, MAO Qing, FENG De-min, WANG Ying-hui

(College of Computer Science, Shaanxi Normal University, Xi'an 710062, China)

**Abstract:** Object-oriented software engineering is the engineering in which the object-oriented technique is widely used. It increases the software revisability and maintainability. Otherwise, the software usability is not effectively solved. The method of user-target driven interaction design is produced with focus on the user. The integrated method of interaction design and the OO design is described. The responsibility of interaction engineer and the modeling engineer is discussed. Assuring the software maintainability, the software usability is boosted at the same time.

**Key words:** usability; human-computer interaction; user-target driven; interaction design

### 0 引言

软件工程的主要目标是解决“软件危机”,控制软件的复杂性以及由此产生的高开发和维护费用。面向对象的软件工程是面向对象技术在软件工程领域的全面运用,它填平了从问题域到计算机领域之间的鸿沟<sup>[1]</sup>。用户需求的不断变化和基础设施的更新使得对软件进行修改和维护成为必然,而面向对象技术的优势是有助于开发易于理解、易于实现、易于修改和维护的软件;再加上近年来设计模式、软件体系结构、UML、基于构件的开发方法等技术的出现,使得采用面向对象的软件工程方法开发的软件具有较高的可修改性和可维护性。但是面向对象的软件工程并没有很好地解决软件的可用性(Usability)问题。操作复杂,培训时间长,软件虽然功能强大但很难使用且易出错,在实际运行中有的已经造成了巨大的破坏<sup>[2]</sup>。

可用性是人机交互(HCI, Human-Computer Interaction)系统设计中的一个需要重点考虑的方面<sup>[3,4]</sup>,它关系到人机交互能否达到用户的预期目标,以及实现这一目标的效率与便捷性。可用性是指特定的用户在特定环境下

使用产品并达到特定目标的效力、效率和满意的程度<sup>[3]</sup>。可用性并不仅与用户界面有关,而是蕴涵更广泛的内涵,可从5个方面去理解可用性,这5个方面集中反映了用户对产品的需求,从它们的英文表达上被归纳为5个“E”<sup>[5]</sup>,即有效性(Effective)、效率(Efficient)、吸引力(Engaging)、容错能力(Error Tolerant)和易于学习(Easy to Learn)。当可用性上升到工程的层次时,就成为可用性工程。

面向对象的软件开发方法从问题域和系统责任出发,强调用面向对象的观点去分析客观世界和用户需求<sup>[1]</sup>。以功能建模为中心,从中发现对象、识别对象、抽象出类并在各个层面上考察系统是当前面向对象的主要方法。虽然软件界面可能是由GUI专家设计的,但这并没有从根本上解决软件的可用性问题;加之软件与用户的交互过程一般是由建模人员设计的,虽然建模人员具有很强的逻辑思维,但他们不知道哪些功能是用户经常使用的,哪些功能是不经常使用的;哪些需要非常强的交互支持,哪些可以弱化交互机制。他们通常都以一致的方式对待这些功能,所以使得所设计的交互过程对用户的适用性较差。

要从根本上解决软件的可用性问题,应该以用户为中心,按照用户的思维方式设计用户软件交互过程和界面。为此,文中提出了一种提高软件可用性的方法,该方法首先是描述用户的目标,由目标驱动功能的开发和界面的设计,之后利用面向对象技术抽取功能和界面对象,最后由

收稿日期:2005-04-05

**作者简介:**张立臣(1979—),男,河北邢台人,硕士研究生,研究方向为软件工程与模式识别;冯德民,教授,硕士生导师,研究方向为软件工程、分布式集成技术与计算理论;王映辉,博士,副教授,硕士生导师,研究方向为可视化、模式识别和软件演化技术。

程序员设计和实现软件界面。

## 1 目标驱动功能的交互设计

为了提高软件的可用性,进行交互设计应该以用户为中心,设计的交互过程要满足用户的需求,但最重要的是要满足用户的目标。用户的目标和用户要完成的功能或任务是不同的,目标是一个终结条件,然而功能或任务是达到目标所必需的一个中间进程<sup>[2]</sup>。

### 1.1 用户目标的描述

交互设计师经过与用户座谈,或对工作环境中的用户进行观察,分析用户的工作情况及个体特征;分析用户的目标和任务,以及为达到目标通常采用的方法;分析竞争产品及其交互界面,了解其优缺点。然后交互设计师按照各个用户目标和任务的不同对用户进行分类,抽象出每一类用户的目标,对每一类用户用一个用户模型来表示。每个用户模型定义了大量的细节,诸如姓名、性别、年龄、职业、文化程度、性格特点、期望目标等等。其中的关键细节是用户的目标:实际目标和个人目标。用户的实际目标往往与软件要完成的功能是相近的,用户可以根据自己所学的领域知识而不必经过复杂的培训学习就可掌握软件的操作方法,完成用户日常的功能,比如记录订单。但是,交互设计要突出用户的个人目标,比如尽量减少对话框,避免所有的任务都要用户确认而使用户厌烦等等。抽象用户模型的过程如图 1 所示。

对抽象出的每一个用户模型都要进行详细的交互设计和界面设计。只有满足了用户的目标,设计的软件才能保证可用性,所以用户模型的个人目标和实际目标决定了交互设计中的功能和界面的设计。

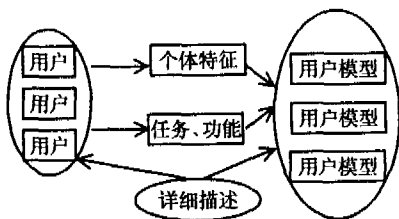


图 1 抽象用户模型及目标

### 1.2 目标驱动功能的开发

建模人员通过 use case 对功能进行抽象,并用自然语言进行详细描述。交互设计师也通过 use case 对功能进行抽象,并且用详尽的语言描述用户与软件的交互过程。他们通过详细描述软件的外在表现,从用户的角度刻画系统。以是否满足用户的目标来检验软件是否达到可用性,描述的功能和交互过程是否达到可用性。

在描述 use case 的时候,交互设计师要强调从用户的角度来观察系统,对 use case 的描述要满足用户的目标。用户与软件的交互过程中,多个 use case 可能要进行沟通,以协作的方式完成用户的功能,交互设计师还要详细描述 use case 间的协作关系,使软件的外在表现形式更接近用户日常业务工作的方式,保证以用户更容易操作和理

解的方式实现用户的需求。为了更好地满足用户的目标,交互设计师必须区分对待用户的不同功能,对用户经常使用的功能或 use case,要尽可能详细地描述其交互过程;对其它一些不经常使用的功能,描述性可以不强,也可以采纳建模人员的方案。目标驱动功能的开发过程如图 2 所示。

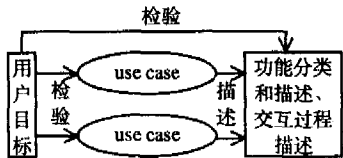


图 2 目标驱动功能的开发过程

### 1.3 界面设计

由于人机界面是系统与用户直接接触的部分,它给予用户的影响和感受最为明显,所以人机界面质量之优劣对于一个软件系统是否能获得成功具有至关重要的作用。人机交互界面设计所要解决的问题是如何设计人机交互界面和系统,以有效地帮助用户完成任务,而且尽量使用户在交互过程中获得愉悦的心情<sup>[4]</sup>,所以,是否满足用户的目标,用户是否感到满意就成为检验界面设计优劣的标准。Gould, Boies 和 Lewis 于 1991 年提出了以用户为中心设计(User-centered Design)的 4 个重要原则<sup>[6]</sup>:及早以用户为中心、综合设计、及早并持续性地进行测试、反复式设计。

根据软件与用户的交互描述情况,交互设计师也开始对界面精心设计。每一种用户模型都要有一致的唯一的界面布局,以便该类用户能在使用过程中触类旁通,举一反三。交互设计师要详细说明界面的布局,使设计的界面与交互过程中的描述一致,最终满足用户的目标。对于界面设计目前有一些技术和技巧<sup>[7-9]</sup>,比如保证日常使用的功能和任务所需要的控制与数据突出地放在界面上,而把所有其它不经常使用的功能放到次要位置上,让它们离开用户正常的视野。交互设计师还可以详细说明在哪些地方程序员不能私自添加按钮、对话框等等。

以用户为中心、用户目标驱动功能的开发与界面设计的交互设计过程保证了交互设计可以满足用户的目标,使用户感到满意,最终提高软件的可用性。

## 2 交互设计师与建模人员的职责划分

由于现实世界是复杂的,并且交互设计师是代表用户的,其描述语言和图形都是用户易于理解的,不适合程序员的逻辑思维,所以,仅仅依据交互设计的结果进行编程实现是非常复杂和困难的,还需建模人员对交互设计结果进行面向对象的抽象和设计。交互设计师通过描述产品的外在表现,通过大量的细节和高度的精确性,详细说明用户与产品的交互过程,与此同时,具体的实现问题则留给了建模人员和程序员。建模人员用面向对象的观点对交互设计文档和界面进行建模和抽象,并在各个层面上描

述系统,最后由程序员进行编程实现。

交互设计的结果不仅仅影响人机交互部分,还可能影响到系统的其它设计部分。因为交互设计是由用户的目标决定功能的,所以交互设计师对单个 use case 或 use case 与 use case 之间的协作的描述可能与建模人员的描述不一致。这种不一致需要通过协商解决。一般情况下,交互设计的结果是交互设计师通过分析用户的目标得到的,设计结果更符合最终用户的需求,所以交互设计师通常对整个软件项目负责,拥有更大的发言权。

交互设计文档和其它设计文档作为设计的结果,共同对后续的过程产生影响。在进行面向对象的编程时,程序员必须以此作为编程的参考,要充分实现设计文档的内容。程序员不能单方面地更改设计文档。在测试阶段,测试人员除了要进行面向对象的测试之外,尤其要测试软件与用户的交互是否和交互设计文档中所详细描述的过程一致,并且要检查软件的界面是否与交互设计文档对界面的描述保持一致。

### 3 结 论

软件的可用性是软件成功的重要因素,提高软件可用性是软件向来追求的目标之一。从软件建模角度来看,面向对象的软件开发技术对人机交互部分的设计过多关注于面向对象的抽象,其开发过程基本上是 use case 驱动的,通过分析阶段产生的 use case 来指导人机交互部分的设计,对交互部分的设计和实现也是基于建模和编程人员的逻辑思维,这样使设计出的软件的可用性无法保证。

交互设计从过程上为获得软件的可用性提供了一种

途径,以用户目标驱动功能的开发和界面的设计来满足用户的需求。文中提出了以目标驱动功能的交互的设计思路,并对交互设计师和建模人员的职责进行了探讨,使得交互设计和面向对象设计进行了较好的结合。由于交互设计师是代表用户的,其描述语言和图形都是用户易于理解的,而建模人员运用面向对象的观点对交互设计进行抽象和设计,在保证软件可用性的基础上又提高了软件的可修改性和可维护性。

### 参考文献:

- [1] 邵维忠,杨美清.面向对象的系统设计[M].北京:清华大学出版社,2003.
- [2] Cooper A. 软件开发的创新思维[M].刘瑞挺,刘强,程岩等译.北京:电子工业出版社,2003.
- [3] ISO 9241-11. Guidance on Usability[S]. 1998.
- [4] 孟祥旭,李学庆.人机交互技术:原理与应用[M].北京:清华大学出版社,2004.
- [5] Michael J A, Mazur B. Content and Complexity: Information Design in Technical Communication[M]. Erlbaum: [s. n.], 2003.
- [6] Gould J D, Boies S J, Lewis C. Making usable, useful, productivity-enhancing computer applications[J]. Communications of the ACM, 1991, 34(1): 74-85.
- [7] David R P, Alan M. Graphical User Interface Design and Evaluation[M]. London: Prentice Hall, 1995.
- [8] Dermot B. Structured User-interface Design for Interaction Optimisation[M]. New York, NY: Prentice Hall, 1993.
- [9] 董士海,王 坚,戴国忠,等.人机交互和多通道用户界面[M].北京:科学出版社,1999.

(上接第 192 页)

```
Course.xml");
OutputStreamWriter outWriter = new OutputStreamWriter(outStream);
((XmlDocument)doc).write(outWriter, "GB2312");
outWriter.close();
outStream.close();
```

此算法是根据扫描算法得来的,该算法在此不作详细介绍。DOM 的处理过程如下:首先,将相关的 XML 数据文档以 DOM 树的形式读入内存,在循环体内,从课程优先级最高值(10)开始扫描,逐个找出,然后将优先级高的课程排在效率较高的课时(此处效率由高到低分别是第一节(上午),第二节(上午),第三节(下午)),这样逐一进行排课,最后将向相应的课程的 XML 文档数据插入到所关联的教师当中。当以上信息自动执行完毕后,将存于内存中的教室、课程相关联的 XML 文档写入 roomCourse.xml 文件中,作为下一模块的信息基础。

### 4 结束语

文中以 XML 作为信息描述和存储的方法,并且通过

XML Schema 来定义相关的信息模型,在数据的处理方面采用了 XML DOM 作为信息抽取、修改的基本方法。这样使系统的信息描述精确,使系统的设计更加简洁。但该系统,由于 DOM 本身的特点决定,在操作大的信息文档时候,会造成内存的占用过大。所以,该系统在系统信息库中的数据不是很大的情况下会取得很好的效率。当信息库比较大的时候,可以考虑采用结合 SAX 的方法来提取、修改数据。

### 参考文献:

- [1] Birbeck M, Martin D. XML 高级编程[M].裴剑锋译.北京:机械工业出版社,2002.
- [2] Carlson D. Modeling XML Vocabularies with UML(Part I, II, III)[EB/OL]. <http://www.xmlmodeling.com>, 2001.
- [3] W3 Consortium. XML Schema Part 0: Primer, Recommendation[EB/OL]. <http://www.w3.org/TR/2001/REC-xmlschema-0-20010502/>, 2001.
- [4] Hunter D, Watt A, Rafter J, et al. Beginning XML[EB/OL]. <http://www.wrox.com/>, 2004.
- [5] Harold E R. 用 Java 处理 XML(影印版)[M].北京:科学出版社,2004.