

基于 OOD 的软件适应性设计方法^①

Method of Designing Application Software Adaptability Based on OOD

赵立臻 (肇庆学院 教育技术与计算机中心 广东 肇庆 526061)

杨帆 (肇庆学院 计算机科学与软件学院 广东 肇庆 526061)

摘要: 针对应用软件的适应性问题,分析影响应用软件适应性的主要因素,简要介绍 OOD 设计方法,将变化的应用需求要素抽象为对象,并以应用主题为索引组织存储。在此基础上,通过增添需求变更设置服务和改进应用软件的主控服务流程,提供用户直接变更和系统间接处理应用需求的能力,形成了一种基于 OOD 的软件适应性设计方法。

关键词: OOD 适应性 应用主题 数据对象 格式对象

应用软件的适应性是系统设计者为了规范系统设计、提高应用系统的通用性和灵活性、降低系统维护和系统升级的复杂度、减少相应工作量而赋予应用软件的一种技术指标,直接反映应用系统对不同用户相似应用需求的适应能力,是应用软件设计的关键内容之一。近年来人们逐渐采用面向对象设计——OOD^[1]方法来开发应用软件系统,但 OOD 的应用大多停留在采用面向对象的桌面设计和编程工具实现特定用户需求的“直接”设计层面,这种设计的优点是快速实现用户需求,随之带来的缺点是不能适应多种需求,软件的适应性较差。影响应用软件适应性的因素有很多^[2],其中不同的数据项、输入格式、输出格式等是主要因素。

本文针对应用软件的适应性问题,结合 OOD 的设计思想,提出了一种借助数据库技术实现的基于 OOD 的应用软件适应性设计方法。

1 OOD 方法简介

OOD 是近年来逐渐形成的一种面向对象的系统设计方法,它建立在 OOA^[3]的基础之上,运用面向对象的概念和思想进行系统设计,所使用的基本概念主要有对象、属性、服务、类、消息、关联、主题等,基本设计方法包括抽象、封装、继承、分类、聚合、关联、消息通信、粒度控制、行为分析等。通过“抽象”方法,可完成

对问题本质的共同特征的提取,将单一问题概念化为具体的“对象”,其中,对问题静态特征的抽象形成对象的“属性”,对问题动态特征的抽象形成对象的“服务”。将服务和属性结合为一个不可分的系统单位并尽可能隐藏内部细节的操作就是“封装”;把具有相同属性和服务的对象划分为一类,用类作为这些对象的抽象描述;通过继承,可以由一般类产生特殊类,特殊类的对象拥有一般类的全部属性和服务,这些属性和服务只需在一般类中一次性地进行定义即可自动地、隐含的被继承到特殊类中。

依据 Coad P 和 Yourdon E 提出的 OOD 模型^[4](如图 1),OOD 包括问题域设计、人机交互设计、控制驱动

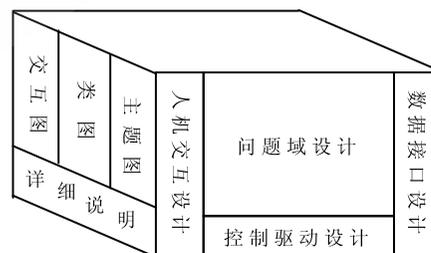


图 1 OOD 模型

设计和数据接口设计四部分内容,每一部分都是由类图、交互图、主题图和详细说明来表达。具体地说,由于每个设计部分的设计结果都是通过对象、对象的内

① 基金项目:肇庆学院自然科学基金项目(0614)

部特征以及对象间的关系来表达的,所以,每个设计部分都以类图作为必不可少的基本模型,并通过详细说明进行全面、确切而详细的描述。在此基础上,可用交互图表示对象之间的交互情况,当每个设计部分的对象类较多时,可以划分主题并用主题图来表示。

在具体的应用软件开发过程中,问题域设计主要完成对用户功能需求和大部分系统性能需求的设计,是应用系统设计的核心内容。人机交互设计主要完成应用系统的人机界面设计,而控制驱动和数据接口设计分别完成应用系统的控制流程和数据存储及管理服务。

2 软件适应性的设计思想

(1) 采用 OOD 方法,将一个应用软件需求中的所有数据项以及输入/输出格式中的相关要素抽象为对象,定义相应的属性和服务,将属性符号化后形成存储模型。

(2) 将同一用户需求中的所有数据项对象归纳为一个主题^[5],称为数据项主题,同一输入格式和输出格式中的各类对象也分别归纳为相应主题,分别称为输入格式主题和输出格式主题,这样三个主题分别定义了一种应用需求的数据项和相应的输入输出格式,各种主题用主题编号或主题名称来标识。

(3) 依据存储模型,分别建立不同主题的数据表(数据项主题表、输入格式主题表、输出格式主题表)和主题索引表,以主题标识为索引将各对象分别存储于各自主题的数据表中。这些数据表用户不能直接访问,属系统数据表^[6]。

(4) 针对数据项主题表,设计数据项设置服务,提供用户自行更改数据项的能力,并可指定默认数据项主题;针对输入/输出格式主题表,设计输入输出格式设计服务,提供 I/O 格式用户自定义能力,并可指定默认格式主题,所需数据项来源于数据项主题表中的某个主题。用户自定义能力,均通过“继承”、添加新属性和保存为新的主题来完成,这样可在提供适应性的同时,提供易用性和实用性。

(5) 分别设计数据项控制服务和输入输出格式控制服务,它们不“直接”处理具体的数据项和输入/输出格式。数据项控制服务根据数据项主题索引表中的默认数据项主题标识访问数据主题表,按主题标识

读出所有对象的属性值,据此创建或访问对应的用户数据表。输入输出格式控制服务根据格式主题索引表中的默认格式主题标识访问格式主题表,按主题标识读出所有对象的属性值,据此处理对应的用户输入/输出格式。从而提供支持多种需求(用户数据项以及输入输出格式)的选择与处理能力,实现应用系统的适应性,如图 2。

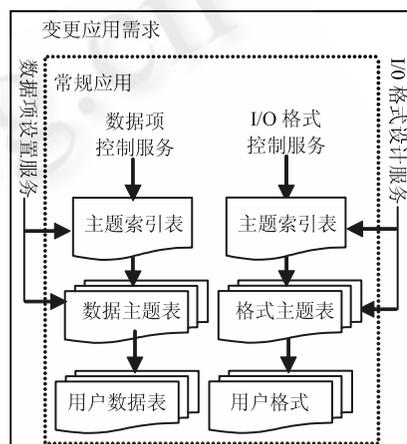


图 2 数据项及输入输出格式的变更与控制

3 对象定义与主题存储

将不同用户需求中的数据项和输入输出格式,用 OOD 方法抽象定义为各类不同的对象,再划分为不同主题,并以主题标识为索引存储所有对象属性是应用软件适应性设计的基础。

3.1 数据项对象

应用软件系统的数据项可用名称、类型、长度三个基本参数来定义,采用 OOD 后,这些参数就是数据项对象的基本属性,从具体数据项到一般对象类的抽象过程如图 3 所示。其中,Di 是符号化后的对象标识,Aij

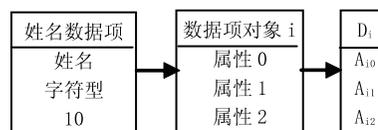


图 3 数据项对象的抽象与符号化

是对象 Di 的第 j 个属性。对于一个包含 M 个数据项的具体应用需求来说,可将各数据项抽象表示为 D₀ ~ D_{m-1} 共 m 个对象,组成与该应用需求相对应的应用主题。不同主题用不同的主题编号(名称)唯一标识,



图 4 对象及其应用主题划分

形成数据项主题图(如图4实线部分),其中N为主题个数。

在基于数据库技术的应用环境中,利用图4实线部分所示的主题与对象,对应创建主题索引表和主题数据表,通常,设计者可将应用软件的一般应用需求当作基础数据,抽象表示为基础对象加以存储。当应用需求变化时,允许用户以基础对象为依据,“继承”产生新对象或新主题,新主题定义新的应用需求。

性,如图5。输入输出格式中包含提示信息、输入/输出数据项、格式线、图片等四种不同要素,遵照OOD方法,这些要素可分别抽象为不同类型的对象,同一格式中的所有对象可划分为一个主题,称为输入(或输出)主题,新的输入输出格式将通过“继承”基础输入/输出格式主题来实现,不同的输入/输出格式将用不同的主题来定义,格式对象定义与主题划分与数据项对象类似,但要在数据项对象的基础上明确定义相应的格式属性,见图4的虚线部分。

需要说明的是,“对象类型”不仅可定义提示信息、输入/输出数据项、格式线、图片等四种不同类型的对象,而且还可考虑具体编程工具的控件特征,进行详细定义,比如,采用Visual Basic.net工具时,对象类型可按表1来定义,这样输入方式和图片输出的问题都很容易解决。

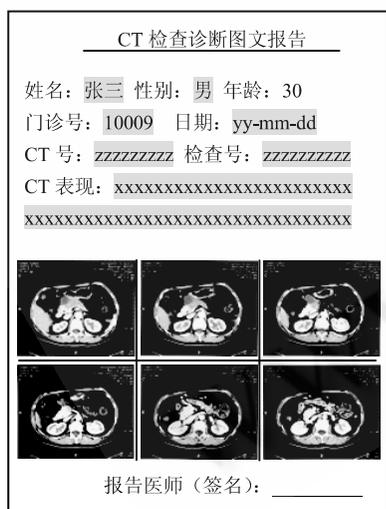


图 5 输入/输出格式样例

表 1 对象类型对应表

| 对象类型 | 类型编号 | 对应控件 | 说明 |
|------|------|------------|------------------|
| 提示信息 | 1 | Label | |
| 文本数据 | 2 | Textbox | |
| 选择文本 | 3 | Listbox | |
| 大文本 | 4 | textbox | Multiline = 1 |
| 图片 | 5 | picturebox | |
| 格式线 | 6 | picturebox | ForeColor = 线的颜色 |

3.2 输入/输出格式对象

输入输出格式中的数据项来源于上述主题数据表中,不同格式的数据项来源于主题数据表中的不同主题,与数据项对象的不同之处在于,输入输出格式数据项对象还需要定义与位置、大小、风格等因素相关的属

4 结束语

以上方法采用OOD思想,将可变的用户需求抽象

为对象和应用主题,并通过主题表存储对象属性,流程控制服务通过访问相应主题表来间接处理用户数据和输入输出格式,从而使应用系统具备相应的适应性。另外,数据项设置以及输入输出格式设计服务,允许用户通过对各类基础数据项或基础模板的“继承”,快速设计出所需的新数据项和新输入输出格式,简单易用。这样,应用系统的适应性设计就不会以牺牲系统易用性和灵活性为代价。该方法不仅有效解决了应用软件的通用性问题,还改善了应用软件的易用性和实用性,可广泛应用于以数据记录为处理单位的各类应用软件的开发中,如医用 PACS 系统、图文界面的人事、科研、商品等各类管理系统,若采用 Web 数据库技术,还可应用于各类 Web 环境的应用软件开发。

参考文献

- 1 绍维忠,杨芙清. 面向对象的系统设计. 北京:清华大学出版社,2003:1-36.
 - 2 王莉,刘厚泉,刘迎春,陆亚萍. 可定制办公自动化系统的设计与实现. 计算机工程与设计,2006(27):2085-2087.
 - 3 绍维忠,杨芙清. 面向对象的系统分析. 北京:清华大学出版社,1998:29-53.
 - 4 Coad P, Yourdon E. Object-Oriented Design. Englewood Cliffs, NJ: Prentice-Hall, 1991.
 - 5 张凤英,邹咸林. 基于领域模型的面向对象分析及 UML 建模. 计算机应用与软件,2004.(6).
 - 6 孙仁鹏. 基于系统表的通用动态查询系统设计与实现. 计算机工程与设计,2007,(23):5819-5822.
- 1 绍维忠,杨芙清. 面向对象的系统设计. 北京:清华