

一种模型驱动的交互式信息可视化开发方法^{*}

任磊⁺, 王威信, 周明骏, 滕东兴, 马翠霞, 戴国忠, 王宏安

(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 100190)

A Model Driven Development Method for Interactive Information Visualization

REN Lei⁺, WANG Wei-Xin, ZHOU Ming-Jun, TENG Dong-Xing, MA Cui-Xia, DAI Guo-Zhong, WANG Hong-An

(Laboratory of Human-Computer Interaction and Intelligent Information Processing, Institute of Software, The Chinese Academy of Sciences, Beijing 100190, China)

+ Corresponding author: E-mail: renlei05@ios.cn

Ren L, Wang WX, Zhou MJ, Teng DX, Ma CX, Dai GZ, Wang HA. A model driven development method for interactive information visualization. *Journal of Software*, 2008,19(8):1947-1964. <http://www.jos.org.cn/1000-9825/19/1947.htm>

Abstract: Wide-Spread deployment for interactive information visualization is difficult. Non-Specialist users need a general development method and a toolkit to support the generic data structures suited to tree, network and multi-dimensional data, special visualization techniques and interaction techniques, and well-known generic information tasks. This paper presents a model driven development method for interactive information visualization. First, an interactive information visualization interface model (IIVM) is proposed. Then, the development method for interactive information visualization based on IIVM is presented. The Daisy toolkit is introduced, which includes Daisy model builder, Daisy IIV generator and runtime framework with Daisy library. Finally, an application example is given. Experimental results show that Daisy can provide a general solution for development for interactive information visualization.

Key words: information visualization; HCI (human-computer interaction); user interface; model; development method; toolkit

摘要: 设计与实现面向领域应用的交互式信息可视化软件十分困难,缺乏统一的开发方法与支撑工具箱,为非专家用户提供对层次、网络、多维等数据类型的统一支持,对各种可视化技术与交互技术的统一支持,以及对信息可视化任务的统一支持.针对此问题,提出了一种模型驱动的交互式信息可视化开发方法 Daisy.首先,提出了交互式信息可视化界面模型 IIVM(interactive information visualization interface model);然后,提出了基于 IIVM 的交互式信息可视化开发方法 Daisy,讨论了该方法的两个核心技术:IIVM 建模与描述文件生成方法、系统自动生成方法.同时,

* Supported by the National Natural Science Foundation of China under Grant No.60673195 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant No.2007AA04Z113 (国家高技术研究发展计划(863)); the National Basic Research Program of China under Grant No.2006CB303105 (国家重点基础研究发展计划(973)); the National Science and Technology Supporting Program of China under Grant No.2006BAF01A44 (国家科技支撑计划)

Received 2008-01-08; Accepted 2008-04-18

给出了 Daisy 工具箱,包括 Daisy 建模工具、Daisy 系统自动生成工具以及运行时框架与组件库.最后,给出了该开发方法与工具箱的应用实例.实例表明,该方法能够为交互式信息可视化开发的统一支撑方法问题提供一种有效的解决方案.

关键词: 信息可视化;人机交互;用户界面;模型;开发方法;工具箱

中图法分类号: TP391 文献标识码: A

信息可视化技术通过对抽象信息提供交互式的可视化表示形式,能够增强人们对复杂信息的认知能力^[1],成为人们解释现象、发现规律、辅助决策的强有力工具^[2].围绕着信息可视化中数据类型的分类^[3],研究者提出了大量信息可视化技术,如一维、二维、三维、多维、层次、网络(图)、时序等数据类型的信息可视化技术^[2].其中,层次、网络、多维等信息可视化技术成为研究热点^[4].围绕着信息可视化中的任务分类^[3],研究者提出了大量交互技术,如动态过滤(dynamic query)^[5]、全局+详细(overview+detail)^[6]、平移+缩放(panning+zooming)^[6]、焦点+上下文(focus+context)及变形(distortion)^[7]、多视图关联协调(multiple coordinated views)^[8]等技术.随着信息可视化技术的发展,商务智能、生命科学等应用领域对于构建交互式信息可视化软件的需求日益增长^[4,9].然而,交互式信息可视化软件的设计与实现仍然十分困难^[4,9,10].对于此类软件中涉及的各种可视化技术、交互技术,只有信息可视化研究领域的专家用户才能掌握,并且需要对各种算法进行繁琐、重复的开发,这一问题阻碍了信息可视化的应用^[4,9,10].不具备信息可视化领域专家经验的用户(非专家用户)需要一套统一的开发方法,能够简单、快速地实现特定领域的交互式信息可视化应用.该方法需要能够提供对层次、网络、多维等数据类型的统一支持,对各种可视化技术与交互技术的统一支持,对信息可视化任务的统一支持^[4,9].研究者对此已进行了探索,在这些研究中,一部分^[11-16,17]仅针对某一种数据类型,而未能对层次、网络、多维等数据类型提供统一的支撑;一部分^[18-21]仅提供对于多视图关联协调机制的支持;一部分^[4,9,10,22]虽然提供了对各种数据类型可视化技术与交互技术的支持,但是未提供对多视图关联协调机制的高层支持.

模型驱动式软件开发是指对实际问题进行高层的抽象建模,并经过模型转换直至生成可执行代码的方法^[23].基于界面模型的软件开发方法是面向非专家用户的模型驱动开发的重要方法之一^[24,25].本文将模型驱动开发的思想用于交互式信息可视化软件的统一开发方法的研究,提出了交互式信息可视化界面模型(interactive information visualization interface model,简称IIVM);基于IIVM提出了交互式信息可视化的开发方法Daisy**,对该方法的两个核心技术:IIVM建模与描述文件生成方法、系统自动生成方法进行论述,并提供了支撑工具箱;最后将其应用于制造业的交互式信息可视化软件开发.实例表明,Daisy能够提供对层次、网络、多维等数据类型的统一支持,对各种可视化技术与交互技术的统一支持,对信息可视化任务的统一支持.目前可视化算法库支持 Treemap^[26],DOITree^[27],CirclePacking^[28,29],RadialGraph^[30],ScatterPlot^[31],Fisheye Menu^[32],Nested Circle Fisheye^[33]等算法.

1 相关研究

信息可视化源于计算机图形学、用户界面等研究领域的促进与发展,研究者针对 7 种数据类型提出了大量信息可视化技术,例如,层次信息可视化技术中的 Treemap,DOITree,HyperbolicTree,ConeTree 等;多维信息可视化技术中的 ScatterPlot,Parallel coordinates 等;网络信息可视化中的 RadialGraph 等^[2].针对信息可视化的主要任务,即总览(overview)、缩放(zoom)、过滤(filter)、详细查看(details-on-demand)、关联(related)等^[3],研究者提出了大量交互技术,主要包括动态过滤^[5]、全局+详细^[6]、平移+缩放^[6]、焦点+上下文及变形^[7]、多视图关联协调^[8]等技术.伴随着各种可视化技术与交互技术的发展,研究者提出了一系列方法与工具,用于支持某一种数据类型的交互式信息可视化软件的开发.其中,用于网络可视化的包括 Pad++^[11],JAZZ^[12],Piccolo^[13],Graphviz^[14],GVF^[15],Pajek^[16]等方法与工具;用于多维可视化的以 Polaris^[17]为代表.上述方法与工具仅针对某一种特定的数

** 我们将基于 IIVM 的交互式信息可视化开发方法命名为“Daisy”,取其英文词意“雏菊”,相应的工具箱称为“Daisy 工具箱”.

据类型,而我们提出的 Daisy 提供对层次、网络、多维等数据类型的统一支持。

用户在对某一个目标信息概念实体(conceptual entity)^[8]进行可视化分析时,需要将该概念实体转换为一个信息多面体(multi-facet),将其分解为具有关联的多个信息侧面(facet),每个信息侧面是与目标信息概念实体相关的信息,通过一种可视化技术呈现于一个视图中,为目标概念实体的分析提供具有语义关联的多角度支持^[18]。多视图关联协调的用户界面能够改善用户对可视化信息的认知^[8]。用于支持多视图关联协调软件开发的方法与工具主要包括 Snap-Together^[19]、GeoVISTA^[20]、PRISMA^[21]等。上述多视图关联协调系统的开发方法主要关注对多视图关联协调机制的实现,未能提供支持常用可视化技术与交互任务的集成框架。我们提出的 Daisy 在支持多视图关联协调机制的同时,围绕着主要信息可视化任务,提供对各种常用可视化技术与交互技术的统一支持机制。

与 Daisy 目标较为接近的包括 IVCI^[22]、InfoVis^[9]、Prefuse^[4]、ManyEyes^[10]。它们都旨在提供通用开发方法与工具平台,用于快速构建交互式可视化应用。IVCI 本质上是一个可视化算法包,而并未提供一套通用方法与工具箱。InfoVis 面向层次、网络、多维数据类型提供了基于关系表的统一支持,并且提供了对 Treemap、node-link diagram、ScatterPlot 等可视化算法以及对动态过滤的支持。InfoVis 将各种可视化与交互技术封装为界面组件库,在创建新应用时灵活性不高^[4],也没有提供对多视图关联协调机制的支持。Prefuse 面向层次、网络、多维数据类型提供了基于节点和边的统一支持,提供了对 Treemap、Hyperbolic tree、node-link diagram、ScatterPlot 等可视化算法以及 Fisheye Menu 变形算法的支撑,提供了对总览、缩放、过滤、详细查看等任务的支持。Prefuse 对各种算法仅进行了一定程度的封装,提供了比 InfoVis 更为灵活的方法与工具。然而 Prefuse 并未提供一个集成开发环境,也未对多视图关联协调机制提供高层支持,在构建领域应用时仍然需要用户对其组件库进行学习与再开发。ManyEyes 面向 Internet 用户,提供了一个用于快速构建简单可视化网页的网站。但其构建的可视化应用都较为简单,缺乏灵活性与可扩展性,也未提供对多视图关联协调机制的支持,因此不能支持复杂的应用。IVCI、InfoVis 及 Prefuse 均基于 Card 等提出的信息可视化参考模型^[1],以数据流动中的一系列映射指导具体应用的软件开发^[34]。该模型适合于对较低层次的软件结构与程序开发方法进行指导,然而对领域应用的开发而言,仍然需要大量低层次的程序开发工作。与上述方法与工具相比,Daisy 在支持层次、网络、多维等数据类型的多种可视化算法与交互技术的基础上,提供了对多视图关联协调机制的灵活支撑。

2 IIVM

2.1 IIVM概述

在界面模型驱动的软件开发方法中,完备的用户界面模型需要对 5 个组成元素即任务(user task)、用户(user)、领域对象(domain)、表征(presentation)、对话(dialog)及其映射关系进行描述^[24]。其中,任务、用户及领域对象元素属于界面模型的抽象组成元素(abstract element),表征、对话元素属于界面模型的具体组成元素(concrete element),具体组成元素构成了可运行的用户界面,界面模型驱动的软件开发即界面模型中的抽象组成元素与具体组成元素之间的映射问题^[25]。我们根据文献[24]中的界面模型构建交互式信息可视化界面模型 IIVM。

如图 1 所示,IIVM 由一系列模型与映射关系组成。主要包括信息模型(information model,简称 IM)、任务模型(task model,简称 TM)、用户模型(user model,简称 UM)、可视化表征模型(visual presentation model,简称 VM)、对话模型(dialog model,简称 DM)。IM 对应着上述完备界面模型中的领域对象模型,用于对领域应用中的信息及其统一的数据模型进行描述。信息被描述为一系列信息概念实体,每个信息概念实体包含多个信息侧面及其关联。每个信息侧面被描述为数据节点和关联的集合,作为层次、网络、多维等数据的统一数据模型。IM 是对信息可视化参考模型^[1]中数据表(DataTable)的扩展。TM 对领域应用中的交互任务进行描述。交互任务被描述为一系列原子任务的序列集合,用于对主要交互任务即总览、缩放、过滤、详细查看、关联等进行描述。UM 对领域应用中有不同的信息需求的用户角色进行描述,作为交互式信息可视化的个性化用户界面的支撑。VM 对界面中 3 种类型的可视化表征元素进行描述,包括对视图容器、可视结构(visual structure)^[1]、交互控件(动态过滤条和视

觉属性图例)的描述,VM 是对信息可视化参考模型^[1]中可视结构与视图(view)的扩展.DM 对用户在完成任时与可视化表征之间的交互方式(鼠标拖放、键盘按键等)进行描述.

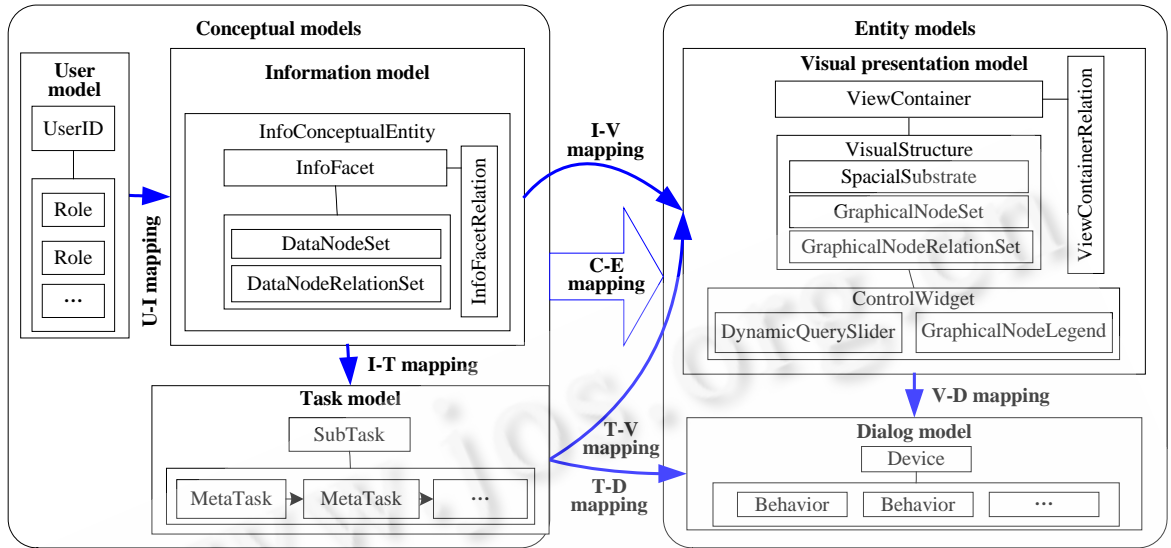


Fig.1 IIVM

图 1 交互式信息可视化界面模型

VM 和 DM 描述的是用户最终可以直接访问的可运行界面,即界面模型的具体组成元素,我们称其为实体模型(entity model).IM, TM 及 UM 描述的是用户通过 VM 和 DM 间接访问的界面,即界面模型的抽象组成元素,我们称其为概念模型(conceptual model).例如,当用户拖动(DM)动态过滤条对 ScatterPlot 图(VM)进行直接的交互操作时,用户是通过 ScatterPlot 图形节点的变化间接访问图形所表征的信息(IM),并且通过动态过滤条的鼠标拖动这一对话方式间接地完成过滤任务(TM).IIVM 所描述的问题是:对于给定用户 u 、信息 i 与任务 t ,找到对应的界面可视化表征 v 和交互对话方式 d ,使用户 u 可以直接通过可视化表征 v 和交互对话方式 d 来完成对信息 i 的任务 t ,也即概念模型到实体模型的 C-E 映射问题(C-E mapping).如图 1 所示,为了对 C-E 映射问题进行描述,我们对 UM-IM 映射(U-I mapping)、IM-TM 映射(I-T mapping)、IM-VM 映射(I-V mapping)、TM-VM 映射(T-V mapping)、TM-DM 映射(T-D mapping)、VM-DM 映射(V-D mapping)进行描述.UM-IM 映射用于对领域应用中不同用户角色可访问的信息进行描述.IM-TM 映射用于描述各个信息侧面需要的交互任务.IM-VM 映射用于描述信息模型与 3 类可视化表征的映射.TM-VM 映射用于描述交互任务中包含的可视化表征.TM-DM 映射用于描述交互任务对应的物理交互设备对话方式.VM-DM 映射用于对可视化表征可接收的物理交互设备对话方式进行描述.在 C-E 映射中存在两类映射约束问题:一类是 IM-VM 映射中的布局算法约束,即布局算法类型对空间基映射与图形节点视觉属性映射的约束;另一类是 TM-VM 映射与 TM-DM 映射中的原子任务唯一性约束.

2.2 IIVM组成元素描述

2.2.1 信息模型

信息模型由领域信息概念实体集合组成.每个信息概念实体作为一个信息多面体,由具有关联关系的信息侧面组成.每个信息侧面由数据节点集合及数据节点关联集合组成,对层次、网络、多维等数据进行统一描述.

定义 1(数据节点(DataNode,简称 DN)). $DN = \langle DNID, Attribute_1, Attribute_2, \dots, Attribute_n \rangle$.其中, $DNID$ 为数据节点标识, $Attribute_i$ 为数据属性, $Attribute_i = \langle AttrName_i, MetaData_i \rangle$, $AttrName_i$ 为第 i 个属性名; $MetaData_i$ 为元数据类型, $MetaData_i \in \{nominal, ordinal, quantitative\}$, $nominal$ 为名词种类型, $ordinal$ 为顺序型, $quantitative$ 为数值型^[1].

定义 2(数据节点关联(DataNodeRelation,简称 DNR)). $DNR=(SourceDNID,TargetDNID,Direction)$.其中, $SourceDNID$ 和 $TargetDNID$ 为源、目的数据节点标识; $Direction \in \{direct,undirect\}$,表示有向与无向.

定义 3(信息侧面(InfoFacet,简称 IF)). $IF=(IFID,DNSet,DNRSet,DataSource)$.其中, $IFID$ 为信息侧面标识, $DNSet$ 是数据节点集合, $DNRSet$ 为数据节点关联集合. $DNSet$ 可对层次、网络、多维等数据节点集合进行统一描述. $DNRSet$ 可对层次和网络类型节点的关系进行描述,对于多维型, $DNRSet=\emptyset$.因此, IF 可对层次、网络、多维等数据进行统一描述. $DataSource$ 为数据源指针,可指向格式化文本文件、XML文件、关系数据库等.

定义 4(信息侧面关联(InfoFacetRelation,简称 IFR)). 设 $SourceAttributeS$ 与 $TargetAttributeS$ 为信息侧面 $SourceIFID$ 和 $TargetIFID$ 中的数据属性集合,设 $PRS=SourceAttributeS \cap TargetAttributeS$,若 $PRS \neq \emptyset$,则对于 $Attribute \in PRS$,称信息侧面 $SourceIFID$ 和 $TargetIFID$ 关于 $Attribute$ 的关联为 $IFR=(SourceIFID,TargetIFID,Attribute,Direction)$.其中, $Direction \in \{direct,undirect\}$.

定义 5(信息概念实体(InfoConceptualEntity,简称 IC)). $IC=(IFSet,IFRSet)$.其中, $IFSet$ 为信息侧面集合, $IFRSet$ 为信息侧面关联集合.

定义 6(信息模型(IM)). $IM=(ICSet)$.其中, $ICSet$ 为信息概念实体集合.

2.2.2 可视化表征模型

可视化表征模型对界面中3种类型的可视化表征元素进行描述,主要包括可视结构;视图容器及关联;交互控件,如动态过滤条、视觉属性图例等.我们将Card等定义的可视结构中的图形标记^[1]分为两种类型:图形节点(GraphicalNode,简称GN)与图形节点关联(GraphicalNodeRelation,简称GNR),从而可以为信息模型与可视化表征模型之间的映射提供支撑.根据Bertin提出的视网膜变量(retinal variable)^[1],取常用的形状(shape)、颜色(color)、大小(size)、方向(orientation)、纹理(texture)等作为图形节点的视觉属性,给出如下定义:

定义 7(图形节点). $GN=(GNID,Label,Location,Shape,Color,Size,Orientation,Texture)$.其中, $Label$ 是图形节点的文本显示标签, $Location$ 是图形节点的布局位置.

定义 8(图形节点关联). $GNR=(SourceGNID,TargetGNID,Direction)$.其中, $SourceGNID$ 和 $TargetGNID$ 是源和目的图形节点标识, $Direction \in \{direct,undirect\}$.

定义 9(可视结构). $VS=(VSID,SpatialSubstrate,GNSet,GNRSet)$.其中, $GNSet$ 是图形节点集合, $GNRSet$ 是图形节点关联集合. $SpatialSubstrate$ 为空间基, $SpatialSubstrate=(AxisSet)$, $AxisSet$ 是轴 $Axis$ 的集合, $Axis=(AxisID,AxisType,DNAttribute,ValueRange)$, $AxisType \in \{NoAxis,NominalAxis,OrdinalAxis,QuantityAxis\}$,其中, $NoAxis$ 表示无轴,用于层次和网络类型数据的可视结构. $NominalAxis,OrdinalAxis$ 及 $QuantityAxis$ 表示名词种类型、顺序型及数值型3种轴^[1],用于多维数据的可视结构. $DNAttribute$ 是轴对应的数据节点属性. $ValueRange$ 是轴刻度值集合.

定义 10(视图容器(ViewContainer,简称 VC)). $VC=(VCID,Location,Size,VSID)$.其中, $VCID$ 是视图容器标识, $InitLocation$ 和 $InitSize$ 是视图容器的布局位置与大小, $VSID$ 是可视结构标识.

定义 11(视图容器关联(ViewContainerRelation,简称 VCR)). $VCR=(SourceVCID,TargetVCID,DNAttribute,Direction)$.其中, $SourceVCID$ 和 $TargetVCID$ 是源和目的视图容器标识; $DNAttribute$ 是关联属性,与信息侧面关联属性相对应.

定义 12(动态过滤条(DynamicQuerySlider,简称 DQS)). $DQS=(DQSID,VSID,DNAttribute,ValueRange,Min,Max)$.其中, $DQSID$ 是动态过滤条标识, $VSID$ 是该DQS控制的可视结构标识, $DNAttribute$ 是用于过滤的数据节点属性, $ValueRange$ 是 $DNAttribute$ 属性值范围,Min和Max是属性值过滤范围最小值和最大值.

定义 13(视觉属性图例(GraphicalNodeLegend,简称 GNL)). $GNL=(GNLID,RetinalProperty,VSID,DNAttribute,Type,ValueSet)$.其中, $RetinalProperty$ 是视觉属性, $RetinalProperty \in \{Shape,Color,Size,Orientation,Texture\}$; $VSID$ 是该GNL对应的可视结构标识; $DNAttribute$ 是该GNL对应的数据节点属性; $Type$ 是该视觉属性的连续性类型, $Type \in \{Continuous,Discontinuous\}$;ValueSet为 $RetinalProperty$ 的取值范围.

定义 14(可视化表征模型). $VM=(VSSet,VCCRSet,ControlWidgetSet)$.其中, $VSSet$ 是可视结构集合; $VCCRSet=$

($VCSet, VCRSet$), 是视图容器集合($VCSet$)与视图容器关联集合($VCRSet$); $ControlWidgetSet = \langle DQSSet, GNLSet \rangle$, 是动态过滤条集合($DQSSet$)和视觉属性图例集合($GNLSet$).

2.2.3 任务模型

任务模型对子任务集合、子任务的原子任务组成及序列进行描述.用于对各种信息可视化任务描述.

定义 15(子任务(SubTask, 简称 ST)). $ST = \langle STID, MetaTask_1, MetaTask_2, \dots, MetaTask_n \rangle$. 其中, $STID$ 是子任务标识; $MetaTask_i$ 是执行 ST 过程中的第 i 项原子任务, 具有原子性, 不可再分.

定义 16(任务模型). $TM = \langle SubTaskSet, MetaTaskSet \rangle$. $SubTaskSet$ 是子任务集合, $MetaTaskSet$ 是原子任务集合. 对于每个信息侧面的交互式信息可视化任务域, 我们预定义:

- $STIDSet = \{ Overview, Pan, Zoom, FilterByAttrDynamicQuery, FilterByRetinalProperty, KeywordSearch, Detail, RelateToFacet, FocusPlusContext \}$. 即对信息侧面的总览、平移、缩放、通过数据节点属性值的动态过滤、通过图形节点视觉属性值的过滤、关键字搜索、查看节点详细信息、信息侧面关联过滤、焦点和上下文浏览.

- $MetaTaskSet = \{ SelectFacet, SelectNode, LayoutInFacet, ShiftInFacet, ScaleInFacet, SetAttrRange, FilterByAttrRange, SelectRetinalProperty, FilterByRetinalValue, SetKeyword, SearchByKeyword, BrowseNodeDetail, FilterInRelatedFacet, SetDistortRadius, FocusDistort \}$. 即选择信息侧面、选择数据节点、总览选中的信息侧面、平移选中的信息侧面、缩放选中的信息侧面、设置数据节点属性值范围、通过属性值范围动态过滤、选择视觉属性、通过视觉属性值过滤、设置关键字、关键字搜索、浏览选中节点详细信息、在关联信息侧面中过滤、设置焦点变形半径、焦点变形.

- $SubTaskSet = \{ \{ Overview, SelectFacet, LayoutInFacet \}, \{ Pan, SelectFacet, ShiftInFacet \}, \{ Zoom, SelectFacet, ScaleInFacet \}, \{ FilterByAttrDynamicQuery, SelectFacet, SetAttrRange, FilterByAttrRange \}, \{ FilterByRetinalProperty, SelectFacet, SelectRetinalProperty, FilterByRetinalValue \}, \{ KeywordSearch, SelectFacet, SetKeyword, SearchByKeyword \}, \{ Detail, SelectFacet, SelectNode, BrowseNodeDetail \}, \{ RelateToFacet, SelectFacet, SelectNode, FilterInRelatedFacet \}, \{ FocusPlusContext, SelectFacet, SetDistortRadius, FocusContextDistort \} \}$.

2.2.4 用户模型

用户模型对领域应用中的用户标识及角色分类进行描述.

定义 17(用户模型). $UM = \langle UserIDSet, RoleSet, UserRoleMappings \rangle$. 其中, $UserIDSet$ 是用户标识集合; $RoleSet$ 是特定领域应用中的用户角色集合; $UserRoleMappings$ 是用户标识与角色的映射集合, $UserRoleMappings \subseteq UserIDSet \times RoleSet$.

2.2.5 对话模型

对话模型对物理交互设备的交互行为进行描述.

定义 18(对话模型). $DM = \langle DeviceSet, BehaviorSet, DBMappings \rangle$. 其中, $DeviceSet$ 是物理交互设备集合, 如鼠标、键盘、笔等; $BehaviorSet$ 是交互行为集合, 如单击、拖放等; $DBMappings$ 是物理交互设备与交互行为的映射集合, $DBMappings \subseteq DeviceSet \times BehaviorSet$.

2.3 IIVM映射关系描述

2.3.1 UM-IM 映射

UM-IM 映射描述用户模型与信息模型的映射. f_{UI} 是映射 $RoleSet \rightarrow ICset$. 形式上, $f_{UI} = \langle Role-ICMappings \rangle$. $Role-ICMappings \subseteq RoleSet \times ICset$, 是 $RoleSet$ 与 $ICset$ 的映射集合. f_{UI} 描述了不同用户角色可访问的信息概念实体集合.

2.3.2 IM-TM 映射

IM-TM 映射描述信息模型与任务模型的映射. f_{IT} 是映射 $IFset \rightarrow SubTaskSet$. 形式上, $f_{IT} = \langle IF-STMappings \rangle$. $IF-STMappings \subseteq IFset \times SubTaskSet$, $IFset$ 是信息侧面集合, $SubTaskSet$ 是子任务集合. f_{IT} 描述了各个信息侧面对应的交互式信息可视化任务集合.

2.3.3 IM-VM 映射

IM-VM 映射描述信息模型与可视化表征模型的映射,共包括 3 个子映射: f_{IV1} , f_{IV2} 及 f_{IV3} .

f_{IV1} 是映射 $ICset \rightarrow VCCRset$,描述信息概念实体与视图容器及关联集合的映射.形式上, $f_{IV1} = \langle IF-VCMappings, IFR-VCRMappings \rangle$.其中, $IF-VCMappings \subseteq IFset \times VCset$,是 $IFset$ 与 $VCset$ 的映射集合; $IFR-VCRMappings \subseteq IFRset \times VCRset$,是 $IFRset$ 与 $VCRset$ 的映射集合.

f_{IV2} 是映射 $IF \rightarrow VS$,描述信息侧面与可视结构的映射. f_{IV2} 包括 3 个子映射,分别描述信息侧面与可视结构中图形节点的布局映射、空间基映射及图形节点的视觉属性的映射.

形式上, $f_{IV2} = \langle IF-GNNRMMapping, IF-SSMapping, IF-RPMMapping \rangle$.其中:

$IF-GNNRMMapping$ 描述了信息侧面与图形节点的布局位置的映射. $IF-GNNRMMapping = \langle LayoutAlgorithmTpye, LayoutAlgorithmTpye \rangle$ 是布局算法类型. $LayoutAlgorithmTpye \in \{Treemap, DOITrees, CirclePacking, RadialGraph, ScatterPlot, ParallelCoordinates, \dots\}$.

$IF-SSMapping$ 描述信息侧面与空间基的映射. $IF-SSMapping = \langle \langle Attribute_1, Axis_1 \rangle, \dots, \langle Attribute_n, Axis_n \rangle \rangle$. $Attribute_i$ 是第 i 个轴 $Axis_i$ 对应的数据节点属性.

$IF-RPMMapping$ 描述信息侧面与图形节点的视觉属性的映射. $IF-RPMMapping = \langle \langle Attribute_1, RetinalProperty \rangle, \dots, \langle Attribute_n, RetinalProperty \rangle \rangle$. $Attribute_i$ 是视觉属性 $RetinalProperty$ 对应的数据节点属性. $RetinalProperty \in \{Shape, Color, Size, Orientation, Texture, \dots\}$.

f_{IV3} 是映射 $DN \rightarrow ControlWidget$,描述数据节点属性与交互控件的映射. f_{IV3} 包括两个子映射,分别描述数据节点属性与动态过滤条及视觉属性图例的映射. $f_{IV3} = \langle Attr-DQSMMapping, Attr-GNLMMapping \rangle$.其中, $Attr-DQSMMapping$ 描述数据节点属性与动态过滤条的映射, $Attr-DQSMMapping = \langle Attribute, DQS \rangle$; $Attr-GNLMMapping$ 描述数据节点属性与视觉属性图例的映射, $Attr-GNLMMapping = \langle Attribute, GNL \rangle$.

2.3.4 TM-VM 映射

TM-VM 映射描述任务模型与可视化表征模型的映射. f_{TV} 是映射 $MetaTaskSet \rightarrow VPTYPESet$.形式上, $f_{TV} = \langle MT-VPMappings \rangle$. $MT-VPMappings \subseteq MetaTaskSet \times VPTYPESet$,其中, $MetaTaskSet$ 是原子任务集合; $VPTYPESet$ 是可视化表征基本元素类型集合, $VPTYPESet = \{GraphicalNode, ViewContainer, DynamicQuerySlider, GraphicalNodeLegend, \dots\}$. f_{TV} 描述了任务模型中的原子任务与可视化表征模型中的可视化表征元素类型(可视结构图形节点、视图容器、动态过滤条、视觉属性图例等)的映射关系.

2.3.5 TM-DM 映射

TM-DM 映射描述任务模型与对话模型的映射. f_{TD} 是映射 $MetaTaskSet \rightarrow DBMappings$.形式上, $f_{TD} = \langle MT-DBMappings \rangle$. $MT-DBMappings \subseteq MetaTaskSet \times DBMappings$,其中, $MetaTaskSet$ 是原子任务集合, $DBMappings$ 是物理交互设备的交互行为集合. f_{TD} 描述了各个原子任务对应的物理交互设备的交互行为.

2.3.6 VM-DM 映射

VM-DM 映射描述可视化表征模型与对话模型的映射. f_{VD} 是映射 $VPTYPESet \rightarrow DBMappings$.形式上, $f_{VD} = \langle VP-DBMappings \rangle$. $VP-DBMappings \subseteq VPTYPESet \times DBMapping$. f_{VD} 描述了对可视表征中的图形节点、视图容器、动态过滤条、视觉属性图例等可视化表征元素可进行的交互行为.

2.4 C-E映射约束描述

在概念模型与实体模型间的映射关系中,存在两类映射约束:一类是 IM-VM 映射中 f_{IV2} 产生的布局约束,即布局映射对空间基映射与图形节点视觉属性映射的约束;另一类是 TM-VM 映射 f_{TV} 与 TM-DM 映射 f_{TD} 中的原子任务唯一性约束.

2.4.1 布局约束

形式上,布局约束 $ConstLayout = \langle LayoutAlgorithmTpye, AxisConst, RetinalConst \rangle$.其中, $LayoutAlgorithmTpye$ 是布局算法类型; $AxisConst$ 是由 $LayoutAlgorithmTpye$ 产生的轴映射约束, $AxisConst \in \{NoAxis, NominalAxis, OrdinalAxis, QuantityAxis\}$; $RetinalConst$ 是由 $LayoutAlgorithmTpye$ 产生的视觉属性映射约束; $RetinalConst \subseteq$

{Shape,Color,Size,Orientation,Texture}.

2.4.2 原子任务唯一性约束

形式上,原子任务唯一性约束 ConstMetaTask=<MetaTask,VPTypConst,DBConst>.其中,MetaTask 是原子任务;VPTypConst 是 MetaTast 对应的可视化表征基本类型约束,VPTypConst∈ {GraphicalNode,ViewContainer,DynamicQuerySlider,GraphicalNodeLegend,...};DBConst 是 MetaTask 对应的设备交互行为约束.对 MetaTask_i 和 MetaTask_j,若 i≠j,则<VPTypConst_i,DBConst_i>≠<VPTypConst_j,DBConst_j>.

3 基于IIVM的交互式信息可视化开发方法

3.1 基于IIVM的交互式信息可视化开发过程与Daisy工具箱概述

基于 IIVM 的交互式信息可视化开发过程如图 2 所示.图中纵向分为两层,上层为开发过程(development process),下层为支撑工具箱(toolkit).图中横向将整个开发过程分为两个阶段:建模(modeling)阶段与系统自动生成(automatic generation)阶段.IIVM 是整个过程的核心,建模阶段围绕 IIVM 生成描述文件,系统自动生成阶段以 IIVM 描述文件为基础,通过解析自动生成交互式信息可视化系统(IIV system).

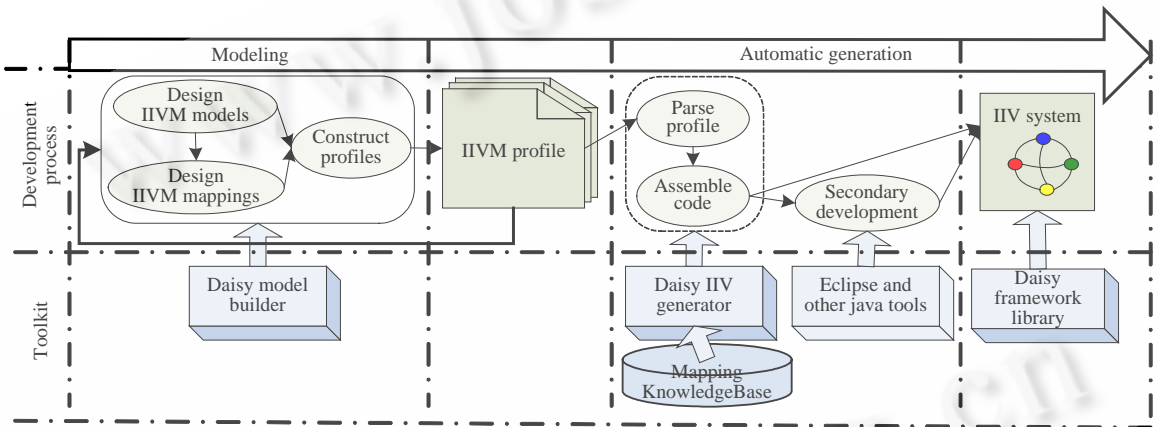


Fig.2 IIVM-Based process of interactive information visualization development

图 2 基于 IIVM 的交互式信息可视化开发过程

建模阶段的目标是为交互式信息可视化的领域应用构建 IIVM 并生成 IIVM 描述文件.构建模型的过程是一个迭代过程,首先对 IIVM 中的概念模型即 IM, TM, UM 以及实体模型 VM, DM 建模,然后对概念模型间的映射(UM-IM 映射与 IM-TM 映射)、概念模型到实体模型的映射(IM-VM 映射、TM-VM 映射及 TM-DM 映射)、实体模型之间的映射(VM-DM 映射)建模.最后生成基于 XML 的 IIVM 描述文件.对 IIVM 描述文件可以进行迭代设计,从而支持模型的复用与扩展.系统自动生成阶段的目标是生成系统原型代码并可进行二次开发,生成可运行的交互式信息可视化系统.IIVM 描述文件基于声明性的语言,仅仅给出了软件系统的抽象描述,具体实现需要映射到相应的实现技术^[25].这一过程首先对 IIVM 描述文件进行解析,基于映射知识库(mapping KnowledgeBase)将 IIVM 描述文件映射到相应的实现技术,基于系统框架程序模板对组件进行组装并生成代码,生成原型系统.对于生成的代码可以作进一步的二次开发,以弥补 IIVM 描述能力的不足,从而支持生成系统的可扩展性.在整个开发过程中,主要有以下工具提供支持:Daisy 建模工具(Daisy model builder)用于支持模型构建与描述文件生成.Daisy 系统生成工具(Daisy IIV generator)用于系统自动生成,基于映射知识库对 IIVM 描述文件进行解析并嵌入系统框架程序模板生成代码.Eclipse 等 Java 工具用于对已生成代码的二次开发.Daisy 框架组件库用于对交互式信息可视化系统提供底层组件库支持.

3.2 IIVM建模及描述文件生成方法

图 3 为 IIVM 建模及生成 IIVM 描述文件的过程.开发者首先对 IIVM 的 5 个组成模型建模,然后对 3 种类型(概念模型间的映射、概念模型到实体模型的映射、实体模型间的映射)共 6 个子映射及约束建模,最后生成描述文件.由于 XML 具有平台无关性、自描述性、易于标准化等特点,我们选择 XML 作为 IIVM 描述文件语言.开发者可以使用 Daisy 建模工具对 IIVM 描述文件进行迭代设计,对专家用户而言,可以直接使用 XML 编辑工具进行编辑.如图 4 所示为 IIVM 描述文件的 XML 模式结构图,图中每个矩形框表示 XML 文档中的一个节点元素,下部的小方框内的数字表示此类节点可能出现的次数范围,后部的减号表示已展开,加号表示未展开.

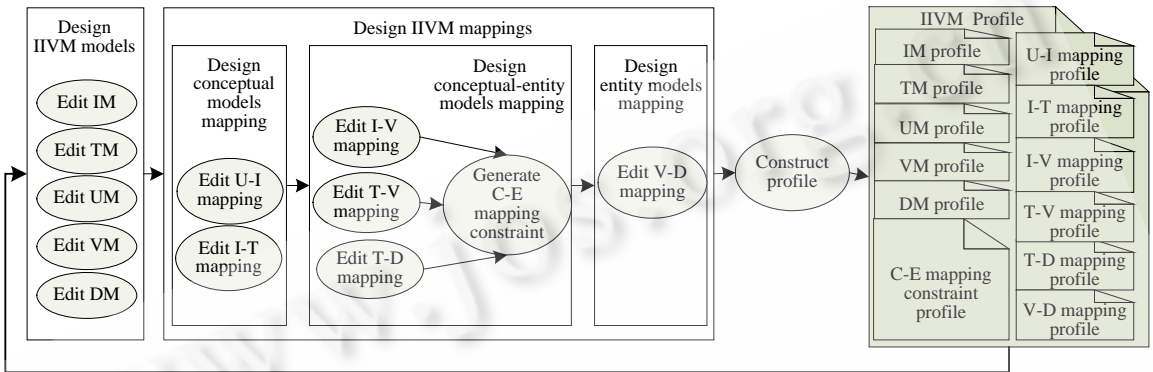


Fig.3 Modeling IIVM and construct profiles
图 3 IIVM 建模与构造描述文件

IIVM 的 5 个组成模型的建模方法如下:(1) IM 建模.IM 中数据节点及关联集合为数据源提供统一的数据模型,开发者首先将来源于不同数据源的原始数据转换为 IM 中数据节点与数据节点关联的集合,形成统一的数据模型.对原始数据转换的方法不是本文的研究重点,可参见文献[35].然后,将数据节点及关联集合形成信息侧面,将多个具有数据属性关联的信息侧面形成一个信息多面体即信息概念实体.将多个信息概念实体形成 IM.IM 描述文件模式如图 5 所示.(2) TM 建模.针对交互式信息可视化的常用任务,Daisy 预定义了子任务集合与原子任务集合.子任务集合包括对信息侧面的总览、平移、缩放、通过数据节点属性值的动态过滤、通过图形节点视觉属性值的过滤、查看节点详细信息、信息侧面关联过滤、焦点和上下文浏览、关键字搜索等.开发者可以对子任务、子任务的原子任务组成及序列进行编辑和扩展,形成 TM,描述文件模式如图 6 所示.(3) UM 建模.对领域应用的用户角色进行分类,并建立用户标识与用户角色的映射关系,UM 描述文件模式如图 6 所示.(4) VM 建模.根据信息概念实体包含的信息侧面集合建立对应的视图容器集合;建立可视结构标识集合,每个可视结构默认选择网络型;建立动态过滤条与视觉属性图例的标识集合.VM 描述文件模式如图 7 所示.(5) DM 建模.Daisy 预定义了常用物理设备的交互行为,如鼠标双击、鼠标拖放、键盘按键等.设计者对物理设备与交互行为进行配对,并可以对其进行扩展,如加入笔手势等,DM 描述文件模式如图 6 所示.

如图 3 所示对 5 个组成模型建模后,需要对 3 种类型共 6 个映射及其约束进行建模.

首先,对概念模型间的映射(UM-IM 映射与 IM-TM 映射)进行建模.(1) UM-IM 映射.建立用户角色 *Role* 与信息概念实体 *IC* 的一对多映射关系,描述文件模式如图 8 所示.(2) IM-TM 映射.建立各个信息侧面 *IF* 与子任务 *ST* 的一对多映射关系,描述文件模式如图 8 所示.通过上述概念模型间的两个映射,描述不同用户角色可以对哪些信息侧面执行哪些子任务的概念语义.

然后,对概念模型与实体模型间的映射(IM-VM 映射、TM-VM 映射及 TM-DM 映射)进行建模,并建立映射约束.(1) IM-VM 映射需建立 f_{IV1} , f_{IV2} 及 f_{IV3} 三个映射,描述文件模式如图 9 所示.对于 f_{IV1} ,针对每个信息概念实体,建立其各个信息侧面 *IF* 与视图容器 *VC* 的一对一映射,并根据信息侧面关联 *IFR* 建立与视图容器关联 *VCR* 的一对一映射.对于 f_{IV2} ,针对信息侧面 *IF*,通过布局算法类型 *LayoutAlgorithmType* 建立与图形节点集合之间的布

局映射;通过 *IF* 中数据节点属性 *Attribute* 与轴 *Axis* 的一对一映射,建立空间基映射;通过 *IF* 中数据节点属性 *Attribute* 与视觉属性 *RetinalProperty* 的一对一映射,建立图形节点的视觉属性编码映射.对于 f_{IV3} ,建立 *IF* 中数据节点属性 *Attribute* 与动态过滤条 *DQS* 的一对一映射,并建立 *IF* 中数据节点属性 *Attribute* 与视觉属性图例 *GNL* 的一对一映射.(2) *TM-VM* 映射.建立原子任务 *MetaTask* 与可视化表征基本类型 *VPTYPE* 的一对一映射关系,描述文件模式如图 8 所示.(3) *TM-DM* 映射.建立原子任务 *MetaTask* 与设备交互行为 *DBMapping* 的一对一映射关系,描述文件模式如图 8 所示.(4) 建立 *C-E* 映射约束,即 *ConstLayout* 与 *ConstMetaTask*.根据 *IM-VM* 映射中的布局算法类型 *LayoutAlgorithmType*,建立对空间基的轴映射约束 *AxisConst*,并建立对视觉属性编码的映射约束 *RetinalConst*.根据 *TM-VM* 与 *TM-DM* 映射,建立原子任务 *MetaTask* 与可视化表征类型及设备交互行为的一对一约束,即 *VPTYPEConst* 与 *DBConst*.描述文件模式如图 10 所示.

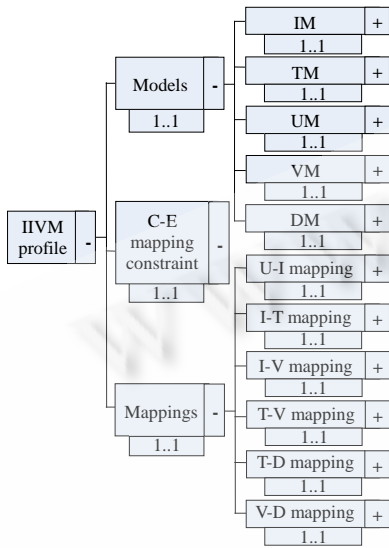


Fig.4 Overview of IIVM XML schema

图 4 IIVM 的 XML 模式概览

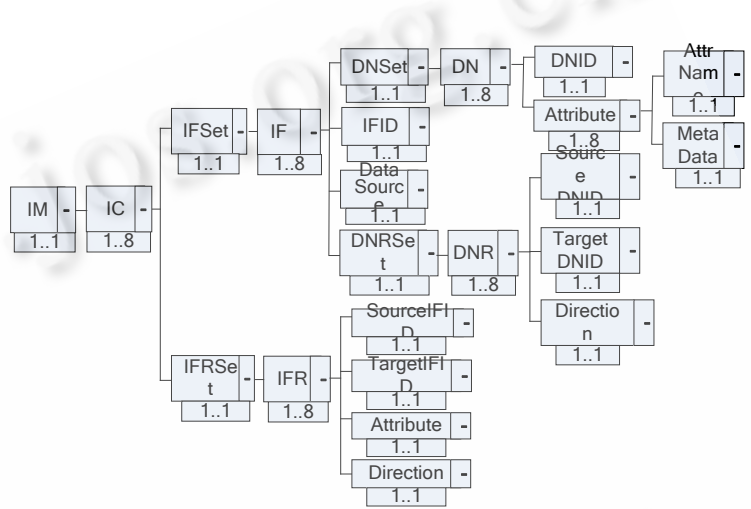


Fig.5 IM XML schema

图 5 IM 的 XML 模式

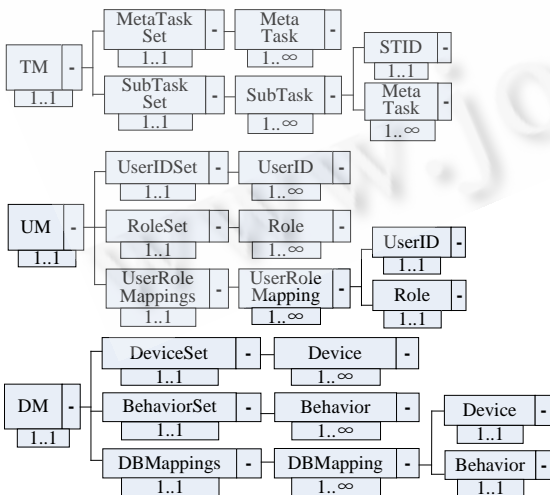


Fig.6 TM, UM and DM XML schema

图 6 TM,UM 以及 DM 的 XML 模式

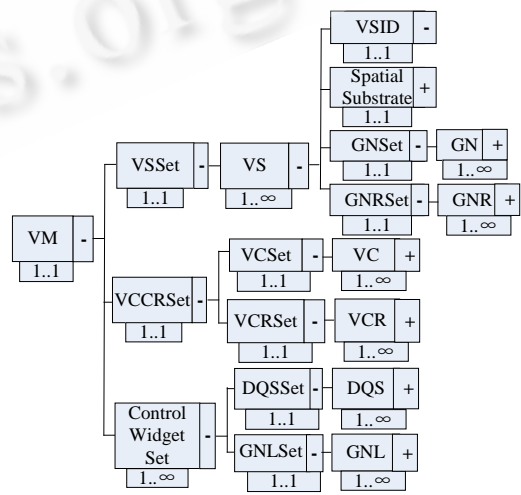


Fig.7 VM XML schema

图 7 VM 的 XML 模式

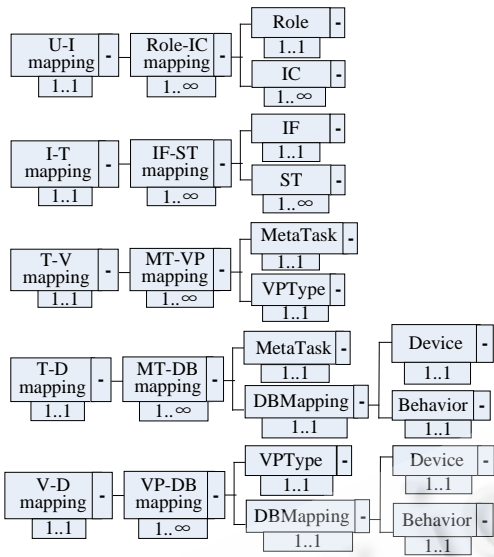


Fig.8 U-I,I-T,T-V,T-D and V-D mapping XML schema

图 8 U-I,I-T,T-V,T-D 以及 V-D 映射的 XML 模式

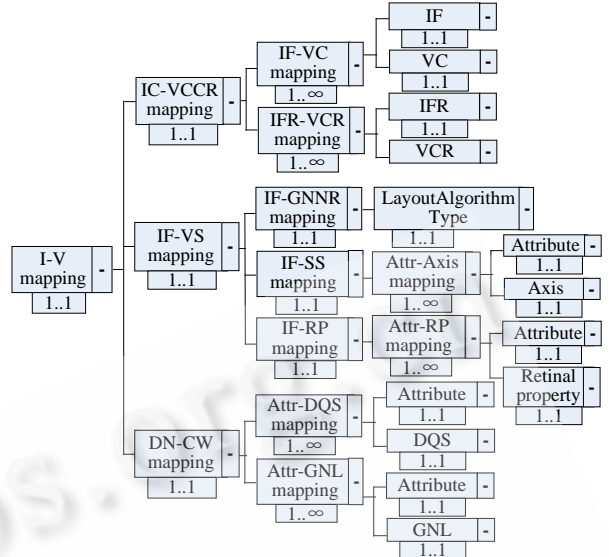


Fig.9 I-V mapping XML schema

图 9 I-V 映射的 XML 模式

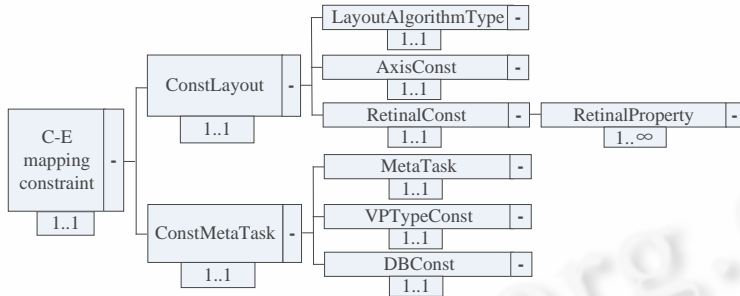


Fig.10 C-E mapping constraint XML schema

图 10 C-E 映射约束的 XML 模式

最后,建立实体模型间的映射,即 VM-DM 映射.建立可视化表征基本类型 *VPTYPE*(图形节点、视图容器、动态过滤条、视觉属性图例等)与物理设备的交互行为 *DBMapping* 的一对一映射,描述文件模式如图 8 所示.Daisy 预定义了一系列映射,如可视结构图形节点的鼠标点击、鼠标双击、鼠标悬停、鼠标拖放等,视图容器的鼠标点击、鼠标双击、鼠标拖放等,动态过滤条的鼠标拖放等,视觉属性图例的鼠标点击等.设计者可以根据 TM-VM 映射和 TM-DM 映射对 VM-DM 映射进行扩展.通过构建实体模型间的映射,描述了各种基本可视化表征对设备交互行为的响应事件.在对 IIVM 的 5 个组成模型与 6 个映射建模完成之后,通过描述文件语言规范构造器生成 IIVM 描述文件.

3.3 基于IIVM的系统自动生成方法

在系统自动生成阶段,Daisy 基于映射知识库对 IIVM 描述文件进行解析,将抽象描述文件映射到具体的信息数据组件、可视化组件及控制组件,然后生成 3 种类型的代码,将生成的代码嵌入系统框架.系统框架结构与底层组件库如图 11 所示.图的上部表示用户通过交互设备与系统界面中的图形元素进行对话.图的中部表示系统运行时的内部框架结构,其中包括运行时流程(以实线箭头表示)、运行时功能处理模块(以椭圆表示)以及运

运行时数据结构(以矩形表示),在框架结构虚线上部标出了开发者定义的部分,即生成代码对应的位置.图的下部为 Daisy 组件库,共包括 4 类组件库,为运行时框架提供支撑(以虚线箭头表示).

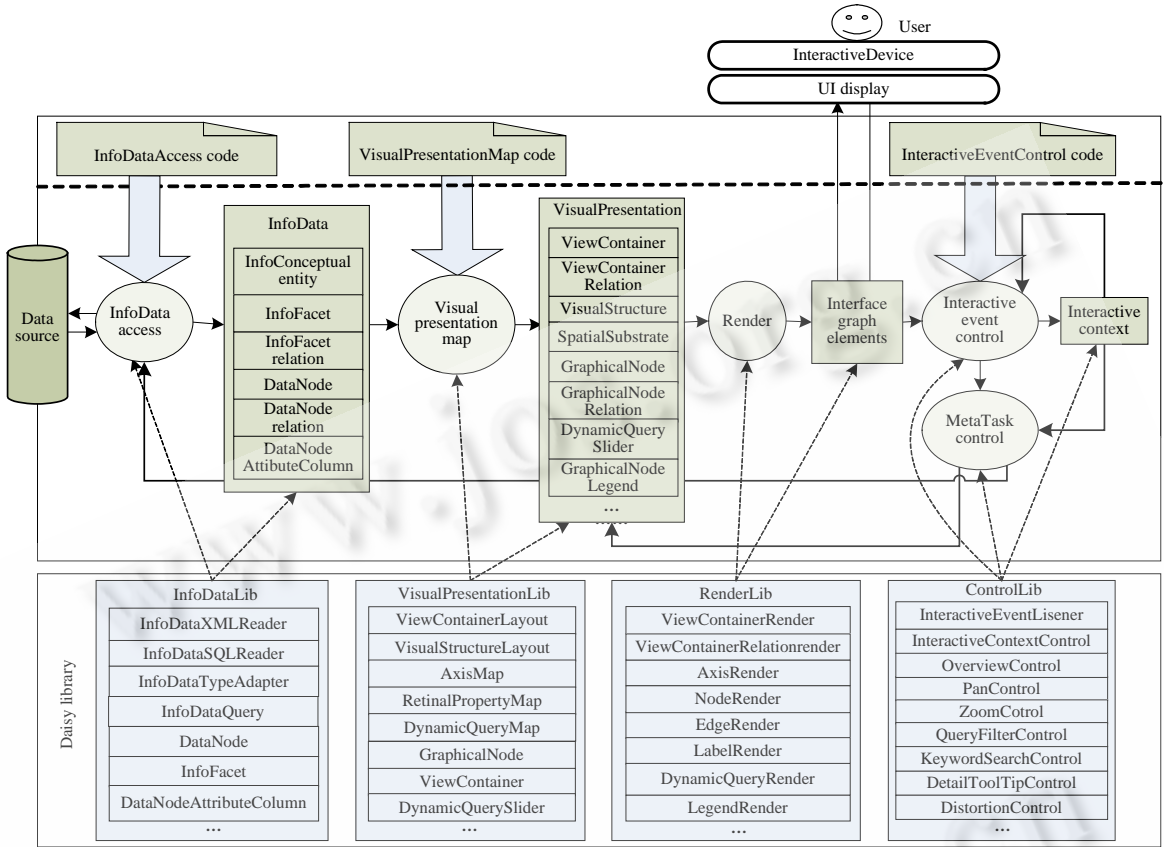


Fig.11 IIV system framework

图 11 系统框架结构

系统框架中包括 5 类功能处理模块与 4 类数据结构.功能处理模块包括交互事件控制模块(InteractiveEventControl)、原子任务控制模块(MetaTaskControl)、信息数据访问模块(InfoDataAccess)、可视化表征映射模块(VisualPresentationMap)及绘制模块(Render).数据结构包括信息数据(InfoData)、可视化表征数据(VisualPresentation)、图形元素数据(InterfaceGraphElements)及交互上下文数据(InteractiveContext).运行时组件库包括 4 类:控制组件库(ControlLib)、绘制组件库(RenderLib)、信息数据组件库(InfoDataLib)及可视化表征组件库(VisualPresentationLib).系统框架的循环处理流程开始于用户通过交互设备(InteractiveDevice)对界面(UI display)中图形的交互动作.系统首先通过交互事件控制模块对交互事件对应的交互任务进行分类,分类时根据当前的交互上下文数据,其中暂存着当前的交互任务执行状态,如已选中的界面可视化表征数据集合等.每当一个原子任务执行后,交互上下文数据将被更新.交互事件控制模块对交互任务分类后,将原子任务的执行控制交由原子任务控制模块.原子任务控制模块将执行两类原子任务.一类不涉及信息数据的读取,仅针对界面可视化表征的图形元素的变换任务(如平移、缩放、变形等).对于此类任务,原子任务控制模块调用控制组件库中相关功能组件,对交互上下文中暂存的可视化表征数据集合进行变换,将变换后的可视化表征数据重新绘制(render)成图形元素数据,并输出到界面中进行显示.另一类涉及信息数据的读取,如动态过滤、关键字搜索等,原子任务控制模块调用控制组件库中相关功能组件,将控制交由信息数据访问模块.信息数据访问模块调用信息数据组

件库中相关功能组件,对信息数据集合进行访问并生成信息数据的数据结构(包括信息概念实体、信息侧面、信息侧面关联、数据节点、数据节点关联、数据节点属性列等).信息数据访问模块将控制交由可视化表征映射模块,可视化表征映射模块调用可视化表征组件库中的相关功能组件,将信息数据生成可视化表征数据(包括视图容器、视图容器关联、可视结构、空间基、图形节点、图形节点关联、动态过滤条、视觉属性图例等).然后,再由绘制模块生成图形元素数据输出到界面显示.用户在界面显示中接收到反馈,一个循环处理流程结束.随着用户进一步的交互行为再进入下一次循环.

系统框架需要底层 Daisy 组件库的支撑,如图 11 所示,共包括 4 类组件库.(1) 控制组件库.该库为原子任务控制模块与交互上下文数据结构提供支撑.主要包括用于交互事件监听的组件 `InteractiveEventListener`、用于交互上下文数据管理的组件 `InteractiveContextControl`、用于在视图容器中呈现整体可视结构的组件 `OverviewControl`、用于在视图容器中平移可视结构的组件 `PanControl`、用于在视图容器中缩放可视结构的组件 `ZoomControl`、用于在视图容器中过滤可视结构的组件 `QueryFilterControl`、用于在视图容器中对可视结构变形的组件 `DistortionControl`.(2) 绘制组件库.该库为绘制模块和界面图形元素数据结构提供支撑.主要包括用于绘制视图容器及关联的组件 `ViewContainerRender`、绘制可视结构中的轴的组件 `AxisRender`、绘制具有视觉属性(形状、颜色、大小等)的图形节点的组件 `NodeRender`、绘制图形节点连接边的组件 `EdgeRender`、绘制文本标签的组件 `LabelRender`、绘制动态过滤条的组件 `DynamicQueryRender`、绘制视觉属性图例的组件 `LegendRender`.(3) 信息数据组件库.该库为信息数据访问模块与信息数据结构提供支撑.主要包括用于读取 XML 文件的组件 `InfoDataXMLReader`、用于读取关系数据库的组件 `InfoDataSQLReader`、用于数据类型转换(将 `Nominal`, `Ordinal`, `Quantity` 类型转化为 `string`, `float` 等类型)的组件 `InfoDataTypeAdater`、用于信息数据集合检索的组件 `InfoDataQuery`,以及用于数据节点及关联、信息侧面及关联、信息概念实体、属性列等数据结构的组件.(4) 可视化组件库.该库为可视化表征映射模块与可视化表征数据结构提供支撑.主要包括用于视图容器布局的组件 `ViewContainerLayout`、可视结构布局的组件 `VisualStructureLayout`、轴映射组件 `AxisMap`、图形节点视觉属性映射组件 `GraphicalPropertyMap`、动态过滤条映射组件 `DynamicQueryMap`,以及用于视图容器及关联、可视结构、图形节点及关联、动态过滤条的数据结构的组件.

系统自动生成阶段的主要目标是针对上述系统框架中 3 个功能处理模块(信息数据访问模块、可视化表征映射模块及交互事件控制模块),生成 3 种类型的代码.系统自动生成阶段的流程如图 12 所示.IIVM 描述文件是对交互式信息可视化系统的声明性的抽象描述,需要将其映射为系统能够理解的具体实现.映射知识库中描述了映射规则,用于将 IIVM 描述文件的语言翻译为对应的 Daisy 组件名称、方法、参数等程序代码语言.Daisy 系统生成工具根据映射知识库对 IIVM 描述文件进行解析.解析主要分为 3 个方面:(1) 对 IM,UM 及 UM-IM 映射的解析.通过对用户角色、信息概念实体集合及二者映射关系的解析,映射至信息数据组件库,用于生成领域信息概念实体集合、信息概念实体包含的信息侧面及关联集合、信息侧面中数据节点及关联集合的构造代码.(2) 对 IM,VM,IM-VM 映射的解析.主要通过对信息模型与可视化表征模型的子映射的解析,映射至可视化组件库,用于生成视图容器集合、视图容器中的可视结构、可视结构中的空间基、图形节点及关联集合、图形节点中的图形属性等的构造代码.(3) 对 TM,VM,DM,TM-VM 映射、TM-DM 映射及 VM-DM 映射的解析.主要通过对原子任务与设备交互行为之间映射关系的解析,映射至控制组件库,生成对各种可视化表征的交互事件响应的处理代码.Daisy 系统生成工具将 3 种类型的生产代码分别嵌入到系统框架 Java 程序中.开发者可以使用 Eclipse 等工具对其进行二次开发,对系统功能进行扩展,也可以直接生成交互式信息可视化系统原型.

4 应用实例

我们将 Daisy 应用于制造业的交互式信息可视化原型系统开发中,表 1 列出了其部分需求,其中包括用户角色(user role)、信息概念实体(InfoConceptualEntity)、信息侧面(InfoFacet)、数据属性(data attributes)、数据类型(data type)、可视化算法(visualization)、交互任务(InteractionTask)等.由表 1 可见,该应用实例需要满足以下主要需求:(1) 能够对层次、网络、多维等数据类型进行统一处理,如表 1 中 Data type 一栏涉及网络型(network)、

层次型(tree)与多维型(multi-dimension);(2) 能够支持面向层次、网络、多维等数据类型的主流可视化技术,如表 1 中 Visualization 一栏列出了 Radial graph,Circle packing,DOI tree 及 Scatter plot 等可视化技术;(3) 能够涵盖各种信息可视化任务,并提供相关交互技术,如表 1 中 InteractionTask 一栏列出了常见的信息可视化任务.

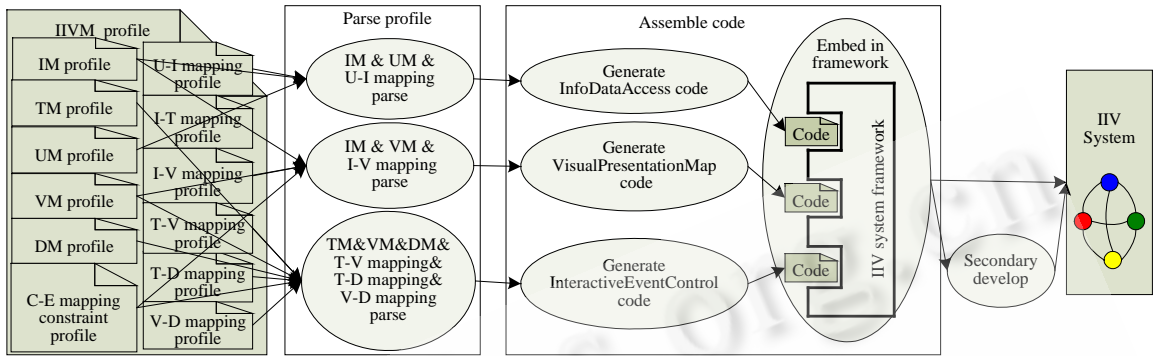


Fig.12 Automatic generation

图 12 自动生成

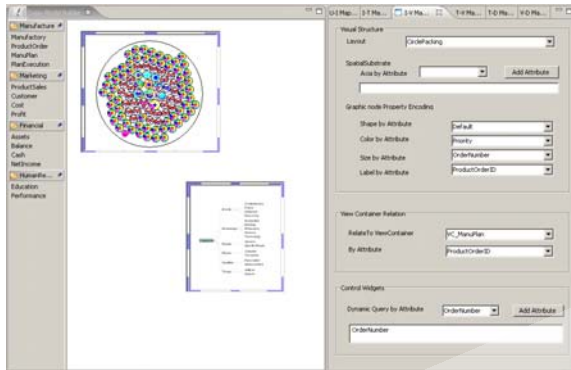
Table 1 Requirement of an application

表 1 应用实例的需求

| User role | InfoConceptual-Entity | InfoFacet | Data attributes | Data type | Visualization | Interaction task |
|-------------------|-----------------------|----------------|--|-----------------|----------------|--|
| Manu manager | Manufacture | Manufactory | ManufactoryID, OverallProductRate, Manager | Network | Radial graph | Overview, Pan, Zoom, FilterByAttrDynamicQuery, KeywordSearch, DetailRelateToFacet, FocusPlusContext |
| | | Product order | ProductOrderID, ProductID, OrderNumber, Priority, ManufactureID | Tree | Circle packing | Overview, Pan, Zoom, FilterByAttrDynamicQuery, FilterByRetinalProperty, KeywordSearch, Detail, RelateToFacet, FocusPlusContext |
| | | Manu plan | PlanID, ProductOrderID, PlanNumber, FinishDate | Tree | DOI tree | Overview, Pan, Zoom, KeywordSearch, Detail, RelateToFacet, FocusPlusContext |
| | | Plan execution | ExecID, PlanID, ProductID, Date, FinishNumber, IsExpire, Quality | Multi-Dimension | Scatter plot | Overview, Pan, Zoom, FilterByAttrDynamicQuery, FilterByRetinalProperty, KeywordSearch,Detail |
| Marketing | Product sales | ... | ... | ... | ... | ... |
| | | ... | ... | ... | ... | ... |
| Financial manager | Financial | ... | ... | ... | ... | ... |
| ... | ... | ... | ... | ... | ... | ... |

我们使用 Daisy 工具箱构建上述应用实例.首先,使用 Daisy 建模工具建立 IIVM 并生成 XML 描述文件.如图 13 所示为使用 Daisy 建模工具为 IIVM 建模的界面.界面主要分为 3 部分,左侧面板为信息概念实体树视图,开发者可以在此对信息概念实体及信息侧面进行增、删、改等操作;界面中央为可视化表征预览区,开发者可以预览信息侧面对应的视图及可视化效果;右侧一组选项卡为 IIVM 编辑区.图 13(a)右侧选项卡为开发者对信息侧面 ProductOrder 的 I-V mapping 进行编辑的界面.其中,开发者选择了 CirclePacking 算法,设置了颜色和大小等视觉属性对应的数据属性,设置了动态过滤条对应的数据属性,并建立了与信息侧面 ManuPlan 的关联.图 13(b)是 I-T mapping 编辑界面,开发者可以对每个信息侧面对应的信息可视化任务建立关联,并对每个任务预定义的原子任务序列进行编辑.图 13(c)是 U-I mapping 编辑界面,开发者可以在此建立用户标识与角色,并为用户角色与信息概念实体建立关联.图 13(d)~图 13(f)为 T-V mapping,T-D mapping 及 D-V mapping 编辑界面.

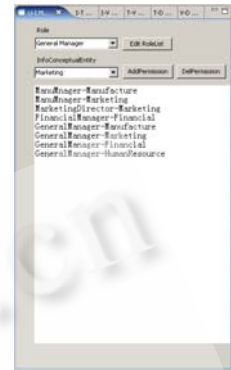
开发者在建模过程中不需要关注各种技术实现细节,不需要书写代码,只需要通过 Daisy 建模工具的编辑界面对各个子模型及映射进行编辑即可.Daisy 建模工具生成的 IIVM 的 XML 描述文件片段如图 13(g)所示.



(a) Daisy model builder with I-V mapping editor
(a) Daisy 建模工具与 I-V 映射编辑器



(b) I-T mapping editor
(b) I-T 映射编辑器



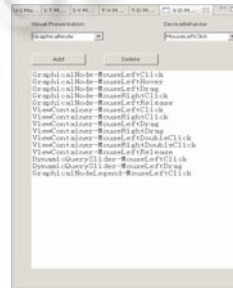
(c) U-I mapping editor
(c) U-I 映射编辑器



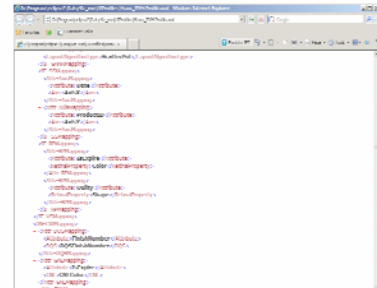
(d) T-V mapping editor
(d) T-V 映射编辑器



(e) T-D mapping editor
(e) T-D 映射编辑器



(f) V-D mapping editor
(f) V-D 映射编辑器



(g) XML profile of IIVM
(g) IIVM 的 XML 描述文件

Fig.13 Modeling IIVM with Daisy model builder
图 13 使用 Daisy 建模工具建立 IIVM

基于 IIVM 的 XML 描述文件,开发者通过 Daisy 系统生成工具构建的原型系统如图 14 所示.图中显示的是用户角色 ManuManager 登录系统后的界面,由图可见信息概念实体 Manufacture 的 4 个信息侧面对应的 4 个视图的可视化效果.界面左侧面板是该用户角色权限范围内的信息概念实体的树结构视图.界面右侧是信息概念实体的可视化区域,其中包含了系统自动生成的可视化表征,即具有关联的多个视图容器、视图容器中的可视化结构、视图容器下方的交互控件等.图 14(a)主要显示了用户通过表 1 需求中所列的任务(如总览、平移、缩放、通过数据节点属性值的动态过滤、查看节点详细信息等)对信息侧面 Manufacture 对应的 RadialGraph 图进行分析.然后,将节点拖入与之关联的信息侧面 ProductOrder 所在的 CirclePacking 图进行过滤,如图 14(b)所示,用户对 CirclePacking 图执行需求中所列的任务.图 14(b)显示了焦点和上下文浏览任务形成的鱼眼变形效果.图 14(c)显示了对信息侧面 ManuPlan 的 DOITree 图分析后,将节点拖入与之关联的信息侧面 PlanExecution 所在的 ScatterPlot 图进行过滤,然后,用户在 ScatterPlot 图中执行需求中的各项任务.由图 14(c)中可见针对 ScatterPlot 自动生成的动态过滤条以及关于颜色和形状的视觉属性图例.

实例表明:Daisy 方法与工具箱能够对层次、网络、多维等数据类型进行统一处理,如实例中对 4 个信息侧面涉及的网络、层次、多维等数据类型的统一支持;能够支持主要界面可视化表征,即视图容器、可视化结构、交互控件的可视映射,如实例中的 4 个相互关联的视图容器、4 种可视结构映射算法(Radial Graph,Circle Packing,DOITree 及 ScatterPlot)、动态过滤条及视觉属性图例控件;能够支持各种信息可视化任务,如实例中涉及的总览、平移、缩放、通过数据节点属性值的动态过滤、通过图形节点视觉属性值的过滤、查看节点详细

信息、信息侧面关联过滤、焦点和上下文浏览、节点关键字搜索等.开发者在 Daisy 方法与工具箱的支持下构建领域应用时,不需要关注各种信息可视化技术的实现细节,不需要具备信息可视化领域的专家经验,因而可以为非专家用户开发领域应用提供一种统一的解决方案.

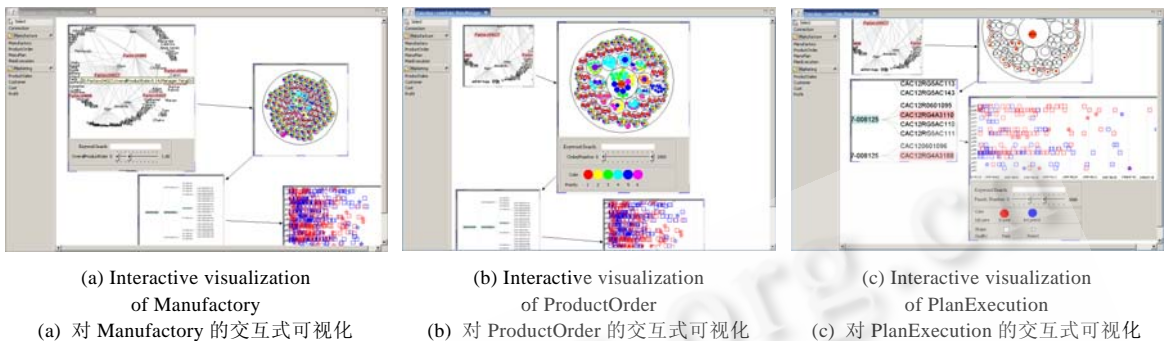


Fig.14 The system prototype generated by Daisy IIV generator

图 14 Daisy 系统生成工具生成的系统原型

5 结 论

本文针对交互式信息可视化软件的统一开发方法问题,采用模型驱动的软件开发思想,提出了交互式信息可视化界面模型 IIVM;提出了 IIVM 驱动的交互式信息可视化开发方法 Daisy.我们讨论了该方法的两个核心技术,即 IIVM 建模及描述文件生成方法与基于 IIVM 的系统自动生成方法,同时,提供了该方法的支撑工具箱.Daisy 为交互式信息可视化软件开发提供了以下支持:(1) 面向层次、网络、多维等数据类型提供了统一的支持机制;(2) 对各种可视化技术与交互技术的统一支持,面向交互式信息可视化软件的界面可视化表征(视图容器、可视化结构及交互控件)提供了映射机制的支持,其中可视化算法库已涵盖了针对层次、网络、多维等数据类型的几类典型算法;(3) 面向信息可视化的常用任务,提供了对总览、平移、缩放、通过数据节点属性值的动态过滤、通过图形节点视觉属性值的过滤、节点关键字搜索、查看节点详细信息、信息侧面关联过滤、焦点和上下文浏览及变形等任务的统一支持机制;(4) 面向非专家用户,通过 Daisy 工具箱能够快速构建交互式信息可视化软件的领域应用.我们给出了将 Daisy 应用于制造业的开发实例.实例表明,该方法能够为交互式信息可视化软件的开发提供一种统一的解决方案.下一步的研究工作是对 Daisy 的自适应可视化方法的研究.

致谢 在此,我们向对本文的工作提出宝贵意见的评审专家表示衷心的感谢.

References:

- [1] Card SK, Mackinlay JD, Shneiderman B. Readings in Information Visualization: Using Vision to Think. San Francisco: Morgan Kaufmann Publishers, 1999.
- [2] Wang WX, Dai GZ. Information visualization. Advances in Information Science, 2006,6:80-118 (in Chinese with English abstract).
- [3] Shneiderman B. The eyes have it: A task by data type taxonomy for information visualizations. In: Regina S, ed. Proc. of the IEEE Symp. on Visual Languages. Los Alamitos: IEEE Computer Science Press, 1996. 336-343.
- [4] Heer J, Card SK, Landy JA. Prefuse: A toolkit for interactive information visualization. In: Wendy K, Shumin Z, eds. Proc. of the SIGCHI Conf. on Human Factors in Computing Systems (CHI 2005). New York: ACM Press, 2005. 421-430.
- [5] Shneiderman B. Dynamic queries for visual information seeking. IEEE Software, 1994,11(6):70-77.
- [6] Hornbæk K, Bederson BB, Plaisant C. Navigation patterns and usability of zoomable user interfaces with and without an overview. ACM Trans. on Computer-Human Interaction, 2006,9(4):362-389.

- [7] Furnas GW. A fisheye follow-up: Further reflections on focus+context. In: Grinter R, Roden T, Aoki P, Cutrell E, Jeffries R, Olson G, eds. Proc. of the SIGCHI Conf. on Human Factors in Computing Systems. New York: ACM Press, 2006. 999–1008.
- [8] Baldonado MQ, Woodruff A, Kuchinsky A. Guidelines for using multiple views in information visualization. In: Vito DG, Stefano L, Laura T, eds. Proc. of the Working Conf. on Advanced Visual Interfaces (AVI 2000). New York: ACM Press, 2000. 110–119.
- [9] Fekete JD. The InfoVis Toolkit. In: Matt W, Tamara M, eds. Proc. of the IEEE Symp. on Information Visualization (InfoVis 2004). Washington: IEEE Computer Society Press, 2004. 167–174.
- [10] Viegas FB, Wattenberg M, van Ham F, Kriss J, McKeon M. Many eyes: A site for visualization at Internet scale. IEEE Trans. on Visualization and Computer Graphics, 2007,13(6):1121–1128.
- [11] Bederson BB, Hollan JD, Perlin K, Meyer J, Bacon D, Furnas GW. Pad++: A zoomable graphical sketchpad for exploring alternate interface physics. Journal of Visual Language and Computing, 1996,7(1):3–31.
- [12] 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). New York: ACM Press, 2000. 171–180.
- [13] Bederson BB, Grosjean J, Meyer J. Toolkit design for interactive structured graphics. IEEE Trans. on Software Engineering, 2004, 30(8):535–546.
- [14] Graphviz. <http://www.research.att.com/sw/tools/graphviz/>
- [15] Marshall MS, Herman I, Melancon G. An object-oriented design for graph visualization. Software: Practice and Experience, 2001, 31(8):739–756.
- [16] Batagelj V, Mrvar A. Pajek: Analysis and visualization of large networks. In: Michael J, Petra M, eds. Graph Drawing Software. Heidelberg: Springer-Verlag, 2003. 77–103.
- [17] Stolte C, Tang D, Hanrahan P. Polaris: A system for query, analysis and visualization of multi-dimensional relational databases. IEEE Trans. on Visualization and Computer Graphics, 2002,8(1):1–14.
- [18] Hetzler B, Whitney P, Martucci L, Thomas J. Multi-Faceted insight through interoperable visual information analysis paradigms. In: Graham W, John D, eds. Proc. of IEEE Symp. on Information Visualization (InfoVis'98). Los Alamitos: IEEE Computer Science Press, 1998. 137–144.
- [19] North C, Shneiderman B. Snap-Together Visualization: A user Interface for coordinating visualizations via relational schemata. In: Vito DG, Stefano L, Laura T, eds. Proc. of the Working Conf. on Advanced Visual Interfaces (AVI 2000). New York: ACM Press, 2000. 128–135.
- [20] Takatsuka M, Gahegan M. GeoVISTA studio: A codeless visual programming environment for geoscientific data analysis and visualization. Computers and Geosciences, 2002,28(10):1131–1144.
- [21] Godinho PIA, Meiguins BS, Meiguins ASG, Carmo RMC, Carcia MB, Almeida LH, Lourenco R. PRISMA—A multidimensional information visualization tool using multiple coordinated views. In: Ebad B, Remo A, Georges G, eds. Proc. of the 11th Int'l Conf. on Information Visualization (IV 2007). Washington: IEEE Computer Science Press, 2007. 23–32.
- [22] Borner K, Zhou Y. A software repository for education and research in information visualization. In: Banissi E, Khosrowshahi F, Sarfraz M, eds. Proc. of the 5th Int'l Conf. on Information Visualisation (IV 2001). Los Alamitos: IEEE Computer Society Press, 2001. 257–262.
- [23] Favre L. Foundations for MDA-based forward engineering. Journal of Object Technology, 2005,4(1):129–153.
- [24] Puerta A, Eisenstein J. Towards a general computational framework for model-based interface development systems. In: Mark M, Pedro S, Christoph GT, eds. Proc. of the 4th Int'l Conf. on Intelligent User Interfaces (IUI'99). New York: ACM Press, 1999. 171–178.
- [25] Paterno F. Model-Based Design and Evaluation of Interactive Applications. Heidelberg: Springer-Verlag, 2000.
- [26] Bederson BB, Shneiderman B. Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies. ACM Trans. on Graphics, 2002,21(4):833–854.
- [27] Heer J, Card SK. DOITrees revisited: Scalable, space-constrained visualization of hierarchical data. In: Costabile MF, ed. Proc. of the Working Conf. on Advanced Visual Interfaces (AVI 2004). New York: ACM Press, 2004. 421–424.
- [28] Wang WX, Ming CY, Wang HA, Dai GZ. Visualization of hierarchical information based on Venn diagrams. Chinese Journal of Computers, 2007,30(9):1632–1637 (in Chinese with English abstract).

- [29] Wang WX, Wang H, Dai GZ, Wang H. Visualization of large hierarchical data by circle packing. In: Grinter R, Roden T, Aoki P, Cutrell E, Jeffries R, Olson G, eds. Proc. of the SIGCHI Conf. on Human Factors in Computing Systems (CHI 2006). New York: ACM Press, 2006. 517–520.
- [30] Yee KP, Fisher D, Dhamija R, Hearst MS. Animated exploration of dynamic graphs with radial layout. In: Keith A, Steven R, Pak CW, eds. Proc. of IEEE Symp. on Information (InfoVis 2001). Los Alamitos: IEEE Computer Science Press, 2001.43–50.
- [31] Becker RA, Cleveland WS. Brushing scatterplots. Technometrics, 1987,29(2):127–142.
- [32] Bederson BB. Fisheye menus. In: Ackerman M, Edwards K, eds. Proc. of the 13th Annual ACM Symp. on User Interface Software and Technology (UIST 2000). New York: ACM Press, 2000. 217–225.
- [33] Ren L, Wang WX, Teng DX, Ma CX, Dai GZ, Wang HA. Fisheye view for visualization of large tree by packing nested circles. Journal of Computer-Aided Design & Computer Graphics, 2008,20(3):298–303 (in Chinese with English abstract).
- [34] Heer J, Agrawala M. Software design patterns for information visualization. IEEE Trans. on Visualization and Computer Graphics, 2006,12(5):853–860.
- [35] Griebisch J, Arafat O, Koska W. xAnVi—An extensible visualization and analysis framework. In: Ebad B, Remo A, Anna U, eds. Proc. of the Conf. on Information Visualization (IV 2006). Washington: IEEE Computer Society, 2006. 45–52.

附中文参考文献:

- [2] 王威信,戴国忠.信息可视化.情报学进展,2006,6:80–111.
- [28] 王威信,明春英,王宏安,戴国忠.基于 Venn 图的层次信息可视化.计算机学报,2007,30(9):1632–1637.
- [33] 任磊,王威信,滕东兴,马翠霞,戴国忠,王宏安.面向海量层次信息可视化的嵌套圆鱼眼视图.计算机辅助设计与图形学学报,2008,20(3):298–303.



任磊(1979—),男,北京人,博士生,主要研究领域为人机交互技术,信息可视化.



马翠霞(1975—),女,博士,助理研究员,CCF 高级会员,主要研究领域为人机交互技术,草图计算,上下文感知交互技术.



王威信(1970—)男,博士,助理研究员,主要研究领域为人机交互技术,信息可视化,CAD&CG.



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.



周明骏(1982—),男,博士生,CCF 学生会会员,主要研究领域为人机交互技术.



王宏安(1963—),男,博士,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,实时数据库.



滕东兴(1973—),男,博士,副研究员,CCF 高级会员,主要研究领域为人机交互技术,虚拟现实技术,计算机辅助设计.