

面向可用性评估的协同工作系统建模

梁路 滕少华 孙为军

(广东工业大学计算机学院 广州 510006)

摘要 在开发以使用为中心的协同工作系统中,越来越要求在早期就能发现并修正可用性方面的缺陷,这就要求评估者对被评估系统的使用上下文有清晰的理解,进而要求在协同工作任务模型上得到充分的支持。然而,现有的界面模型和面向单用户的可用性评测技术却不能有效地覆盖最基础的协作原子行为和团队协作的上下文信息,于是从面向评估的角度提出了一种新的协同工作系统描述模型,在模型上应用快速评估方法中的遍历技术,通过迭代地模拟真实环境中系统的使用状况,可以实现低成本的早期可用性评估。实验表明,该方法较几种传统方法有一定的优势。

关键词 协同工作建模,早期可用性评估,可用性缺陷

中图分类号 TP311 **文献标识码** A

Collaborative Work System Modeling Method towards Usability Evaluation

LIANG Lu TENG Shao-hua SUN Wei-jun

(Faculty of Computer Science, Guangdong University of Technology, Guangzhou 510006, China)

Abstract In the development of the usage-centered collaborative work system, it is urgently required that usability problems can be found and corrected in the early stage, causing the evaluators to have clear understanding on the usage contexts of the system. This work further needs to be supported by an effective collaborative task model. However, neither the existing interface models nor the single-user-oriented usability inspection techniques could well work on the basic-level of collaboration units and the use-context of team work. Therefore, this paper proposed a new collaborative work system model as viewed of usability evaluation, on which a discounting usability inspection-walkthrough approach is applied. This method is able to achieve low-cost usability evaluation in the early stage of software development cycle to detect usability problems by iteratively simulating the use situation in real-world. At last, an experiment showed that the approach has some advantage than several traditional techniques.

Keywords Collaborative system modeling, Early usability evaluation, Usability problem

随着协同工作系统应用的不断增长,越来越多的实践结果表明,许多系统存在着严重的可用性问题,导致很多从业人员对该类系统进行工作抱有担忧和困惑(与真实的面对面的协作相比),使得在系统开发早期就发现并修正可用性缺陷的需求变得越来越迫切。早期可用性评估需要依赖有效的协同工作任务模型,而现有的多种任务分析方案均无法全面地覆盖产生可用性问题的多种因素。另一方面,传统的面向单用户应用系统的可用性检查技术也需要得到合适的改进。

要强调的是,我们并不关心系统级的协同任务和相关资源的描述,因为为协同工作系统建模的目的并不是提供任何具体的用以支持系统设计和开发的手段,即采用什么具体的技术来实现协作任务,而是使用用户级(评估者评估系统时就是系统的用户)的建模方法,因此我们把这种模型称为“面向评估”的模型。通过分析多用户的团队行为、单用户的独立行为和协同工作的原子行为,描述影响用户任务的外在环境因素,本文得出了一种面向早期可用性评估的协同工作系统建模方法,在其上应用快速检查方法 DUI(Discount Usability

Inspection)中的遍历技术,可以大大减少评估成本,并能取得很好的效果。

1 协同工作系统可用性评估现状

可用性是指在特定的使用环境中,用户有力地、有效地、主观满意地使用系统完成特定目标的程度。可用性缺陷是指潜藏于操作过程、外观设计或系统组织中的问题,它们可能会导致用户在完成某些任务时出现迷惑不解、操作错误、进展缓慢或无法完成任务等情况。可用性评估包含了一系列对系统用户界面可用性的方方面面进行评测并确定缺陷的方法,它是整个设计过程的一个重要组成部分。

一直以来,协同软件产品在实施前都缺乏有效而又高性价比的可用性评估手段,这也是造成其在使用过程中常常不能让用户满意的原因之一。经验表明,协同工作系统的可用性缺陷主要表现在^[1-3],高端层面上,系统常常缺乏对受社会、政策和组织特征影响的上下文信息的充分考虑和模拟;低端层面上,系统常常缺乏对协作活动中普遍发生的最基本协作

到稿日期:2009-08-25 返修日期:2009-11-23 本文受国家自然科学基金(60673135),广东省自然科学基金(9451009001002777)资助。

梁路(1980-),女,博士,讲师,CCF会员,主要研究方向为可用性工程、协同软件工程等,E-mail:lianglu@gdut.edu.cn;滕少华(1962-),男,教授,CCF高级会员,主要研究方向为协同软件、网络信息安全等;孙为军(1975-),男,博士,主要研究方向为软件工程方法学等。

行为充分和匹配的支持。虽然近些年来,在研究怎样理解、表示和建模协作任务,模型又怎样被用于系统开发中来支持协同工作等方面已经取得了一定的成果,但至今仍没有一种建模和分析方案能更好地适用于协同工作系统可用性的评估。原因之一是,影响多用户协作行为的许多因素都是高度可变的,它们在不同的团队中可能表现得不一样,因此它们应当在软件系统开发的早期就得到解决。然而矛盾的是,在早期缺乏真实世界的用户和使用环境的情况下又很难对系统进行评测。

针对这一问题,目前 HCI 领域界提出的解决办法之一是快速低耗费地对界面设计的可用性进行评测的检查技术(Usability Inspection Method)^[4]。这种技术不需要使用真实用户和功能化的系统原型,它致力于模拟协作行为的低层次特性。另一种解决办法是使用面向可用性的工程模型,这类模型对人能否在给定的设计中很好地完成任务提供了定量的预测,且预测结果能够用来代替真实的用户数据,因此它们可以被用于快速迭代的评估过程中。另外,不同于单纯的用户测试,工程模型还能捕获界面设计的一些本质,获得设计经验,在未来可以反复使用。然而,现有的工程任务建模方案或是无法表示真实世界的协作行为内在的柔性和可变性,或是在太广的层面上分析协作行为而不能在实践上完全适用于协同工作系统。

2 底层的协作元行为

影响协同工作系统可用性的一个重要方面是系统能在什么程度上支持团队有效地、高效地和令人满意地协作。在文献[5,6]中提出了一种观点:CSCW 应用系统的一部分可用性缺陷问题并非由社会或组织等使用上下文相关的因素产生,而是因为缺乏充分和匹配的方法来检查系统能否支持协作过程的基本活动。这就是说,理想的任务模型要能合适地表达和描述协作环境下合作行为的基本元素。然而在社会科学和 CSCW 研究中,群体工作的传统理论却未能提出相应的、细分的概念用于评估特定的界面设计。这些基本概念称之为“协作元行为”,代表了团队成员在完成共享任务时必须普遍执行的最基本的动作和互作用^[5]。

对协作元行为的定义来源于协作发生时的环境(即共享的工作空间)供给和约束。在共享空间的协作中普遍存在一系列基本的、核心的交互行为,它们与团队文化、协作成员的个人特性或具体的协作任务类型等上下文信息无关。这些元行为包括与其他协作者进行交流、跟踪他人的活动、协商共享工具的使用权、向他人传递对象或工具等。从元行为的层次对协作进行分析,有助于在评估过程中确定某些面向协作特性的清晰的、可度量的可用性目标。也就是说,协作元行为级的分析方法将在面向评估的任务模型中起重要的作用。

经过 HCI 专家不断地总结和归纳^[7-10],主要有 7 类协作元行为:直接通信(explicit communication)、隐式通信(implicit communication)、行为合作(coordination of action)、计划(planning)、监视(monitors)、帮助(assistance)和保护(protection),它们还可以合并为两大类型:通信行为和合作行为,如表 1 所列。但是,对于高级的协作活动,如计划、争论和推理等,由于难以将它们完全从更小的、低层的机械活动的角度进行分解,因此协作元行为级的分析方法就显得不够完善,这

也是这种观点的不足之处。

表 1 协作元行为类型

分类	协作元行为
通信	直接通信 语言信息,书面信息,姿态信息,指示信息(如:指着对象进行描绘),暗示信息(如:拉开抽屉暗示存取东西)
	隐式通信 简单意识(如:观察别人在干什么),对象反映的行为信息(如:观察对象的变化时反映出的信息),肢体反映的行为信息(如:观察人的动作而推测其从事的活动),反馈行为信息(如:对他人言语指示做出的反馈行为),聆听行为
合作	共享访问 (对合作空间对象的使用) 获取资源,储备资源,保护资源 (注:资源包括工具、物体、空间和实践等)
	传递 传递对象(物理地/口头地),存放对象(物理地/标注地) (注:“对象”不仅指代客观实体,还指代非实体信息。)

3 协同工作系统描述模型的定义和框架

使用可用性检查方法^[4]可以将评估提前到设计开发的早期阶段,有助于更早地发现和纠正严重的缺陷。然而,这种方法需要建立在评估者对团队任务的理解之上。由于评估过程并不发生在真实的工作环境中,因此只能通过清晰的协同软件任务模型把评估者和检查方法联系起来,而有效的模型需要能覆盖最基础的协作原子行为和团队协作的上下文信息(影响可用性的两方面因素)。

在为面向评估的协同工作系统建模时,我们以场景(一种用于可用性测试或快速的可用性检查工程的评估工具^[11,12])为单元进行分析,并借用了认知理论中的概念“资源”,并赋予它定义:能够在交互协作的步骤中指导动作的信息集合,在交互环境中,这个信息集合是能够清晰定义的。我们定义模型中有 7 种资源:上下文、工作空间、任务、目标、事件、用户、角色,并得到图 1 所示的描述框架。

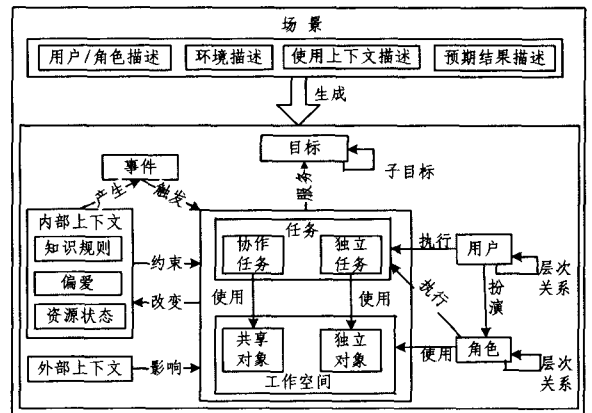


图 1 以场景为单元的协同工作系统模型

1) 上下文描述交互过程某一时刻的系统状态,根据状态的内部表征属性和外部表征属性,将上下文分为“外部上下文”和“内部上下文”。

外部上下文是高层的使用上下文,描述了影响协作交互方式的间接资源,在任务中不对这些资源进行操作,协作的结果也不会改变这些资源的状态。外部上下文中的资源主要包括社会的、政策的、团队组织特性方面的信息。

内部上下文描述了任务执行过程中直接涉及的资源,它是一个三元组 (R, KR, B) 。其中, R (Resource)描述直接资源

的状态集合,任务的执行结果可以改变部分资源的状态;KR (Knowledge Rule)描述了对交互方式起到约束作用的知识规则^[13];B(Bias)即偏爱,是协作者根据当前环境、知识和感知、认识的结果而对动作做出的有倾向性的选择,它可以直接决定执行任务的方式,如当组织特性决定该团队倾向于采用松耦合的方式完成团体目标时,在多条可选的协作通路上,协作者则偏向于选择彼此联系不紧密的途径。

2) 工作空间是工作环境中包含的所有对象的统称。在一定环境下与任务相关的所有物体都被称为对象,对象既可以是具体的(如共享的文件和设备),也可以是概念的(如手势、密码、签名、留言等),包括共享对象和独立对象两类;共享对象允许被协作成员共享地访问;独立对象则包括个体成员的私有空间和拥有物,它只能被特定的成员访问。

3) 目标是用户预期的结果,团队工作的目标是公共的、共享的目标。目标还被分解成子目标,每个子目标是整体目标的一个方面或一个阶段。

4) 任务是模型的核心部分,描述了用户完成目标过程中发生的一系列动作,以及各动作之间的相互关系(主要指依赖关系和时序关系)。团队活动可分解成两部分:任务工作,即为了完成任务而进行的活动;合作工作,即为了以团队合作的方式完成任务而进行的活动。以这种观点为基础,便得到了子任务的两种类型:独立子任务和协作子任务,一个团队任务可以由独立子任务和协作子任务的不同组合构成。独立子任务是松耦合式任务,描述了把团队看成是一个整体来完成目标时的任务步骤,同时它单方面构成了个体任务。协作子任务是紧耦合式任务,它将基于协作元行为来进行描述。

5) 事件描述了触发任务的条件,它由内部上下文的状态变迁而引发。

6) 用户描述了所有对任务进行操作的主体,不仅指人(个人或团队),有时也可能是系统。用户是一个二元组(属性,描述),其中属性是用户的个体特征,如语言、技能、知识等;描述是对每个属性的具体描述。

7) 角色描述了用户的类别,不同类别的用户分配不同的任务,通常以用户的职能来划分。当用户指代人时,多个用户可能会履行同一个角色,反之一个用户也可能同时扮演多个角色。基于任务分配的用户和角色之间的关系是团队特征的基础。角色之间具有层次关系,这种层次关系反映了团队的结构和文化。另外,在团队的描述中还应包括任务职责从一个角色到另一角色的分配或撤销。

4 模型的主要构成

根据前面讨论过的协同工作系统的可用性缺陷的主要来源,模型主要致力于3方面的突破:(1)对影响团队协作行为的高层次的上下文信息的描述和建模;(2)引入协作元行为的概念,从最基本的协作和交互动作的角度,自底向上构建团队协作任务;(3)统一的建模过程。于是,它主要包含上下文模型和任务模型两个部分。

4.1 上下文模型

资源“上下文”包括内部上下文和外部上下文。内部上下文已经用三元组的结构清晰地表示,因此这里的模型对象是外部上下文。外部上下文表示了真实环境中的语境信息的不同方面(如用户关系信息、物理环境信息和组织关系等),这些

方面依据特定使用状况的不同而不同,同时还描述了这些方面因素怎样影响协作任务,并通过层次关系视图的方式理解它们相互之间是怎样关联的。

我们定义了3种抽象元素来构建视图:组件、组件关系和视图规约。外部上下文的每个方面都统一用这3种抽象元素来构建,它们可以半形式化地组织高端层面的语境概念,于是增强了处理这些概念的灵活性。表2列举了几个上下文信息的构成。

表2 用抽象元素构建上下文对象举例

对象	组件	组件关系	视图规约
组织关系	成员的角色	角色的层次关系	如:矩形对应角色,矩形的排列对应角色的职能关系等
环境编排	元素,元素的位置,元素的形状	位置关系	如:元素的命名规则,符号表示等
...

注释:表2中,清晰的组织关系模型表明了为什么某个用户的任务类型与他人不同,而这些职能的差异性又如何影响用户的表现。在一个团队中,这种隐含分布的角色关系能在一定程度上辅助成员和谐地合作;环境编排模型用于模拟真实环境的场面(如机械控制系统),它可以模拟真实图像的类型、内容和位置等信息,帮助确定界面元素和资源的空间关系等。

4.2 任务模型

任务建模过程使用了GOMS(Goal Operation Method Selection)模型的思想,把人机交互的过程看成解决问题的过程,人使用软件系统是有目的的,一个大的目的可以被细分为许多小的目的。为了完成每个小目的,又有不同的动作和方法可供选择,而动作和方法之间的关系又受到上下文信息的影响。在任务的表示方面,我们提出了任务层次分析结构图进行图形化表示。它可能不是严格的树状结构(一个协作场景可能对应多棵任务树),其基本思想是从语义上来确定交互描述和分析交互中的高层模型,而不是具体的执行细节。

通过现实观察收集到的数据是模型的基础,任务建模包含以下几个主要步骤:

1) 首先要把观察数据组合为构成整体协作应用的一系列场景单元。场景描述中包含任务描述、上下文描述和其他相关描述,场景的划分主要依赖于不同的合作团队的特征和任务的特征。

2) 其次要确定协作场景。为明确团队的协作任务,第一步要根据团队中的不同角色确定该角色发生的独立场景活动,并确定这些场景的顺序关系。然后对比每个角色的场景序列,就可以确定在哪里发生了这些角色的协作交互行为,即交叉的场景就是协作场景。

3) 在协作场景中,运用自底向上的建模方法,从协作元行为的角度,对多角色的协作过程进行分解,一级级地定义基于元行为类型的协作子任务和其对共享对象的操作。同时,为达到协作目标可能有多种任务执行途径,各通路上的任务类型有顺序、分支、可选和迭代等。而在时序上,两个并发任务之间也存在交替、选择、同步、允许/带信息的允许和禁止等关系,因此在模型中引用了LOTOS算符对它们进行表示,以满足对协作行为的柔性的需求。于是,自上而下的看,便得到了清晰分层的协作任务结构图。

4) 接着,在其他的独立场景中使用简单的任务树来描述单独用户的行为序列关系。

这样,便得到了一个个图形化的任务模型。最后应用快速可用性评估技术的遍历流程^[14]实施评估过程。如果同步使用启发式准则^[4]进行约束,则可以取得更好的效果。

5 实验与结果

为了在实践上验证这种基于模型的评估方法,我们将其应用于澳洲医疗卫生组织 HealthCare 与 Wollongong 大学 HCI 实验室合作开发的一个关于社区医疗的协同工作系统 CPAT,它涉及病人、护理人员、医生、医疗主管等多种用户的分布式协同工作。同时,我们还组织评估小组分别采用其他两种检查技术:启发式评估^[15]和可用性测试^[15],在实验中对三者进行比较。参与评估的人员主要来自 Wollongong 大学 HCI 实验室的可用性检查(UI)专家和同学。

实验中,把可用性缺陷的严重程度划分成 1~9 等 9 个级别(1 表示最不严重,9 表示最严重)。在执行每种方法确定了系统的可用性缺陷后,专家对这些缺陷的严重程度进行鉴定并给出严重性分值。然后,对分别在 3 种评测方法中得到的缺陷的数量、平均严重程度值、总严重性指标(缺陷严重性分值的总和)和收益/代价(总严重性指标除以总耗时,总耗时包括培训耗时和执行耗时)等进行了比较,如表 3 所列。

表 3 3 种评测方法的实验比较

指标	基于本模型的早期评估	启发式评估	可用性测试
参与评估人员	5 位普通评估员	3 位 UI 专家	2 位 UI 专家和 6 位用户
缺陷数量	59	35	45
平均严重程度值	3.45	3.40	4.72
总严重性指标	204	126	212
收益/代价	4.7	3.3	2.2

结果表明,协作遍历法在收益代价率方面有一定的优势,在同样的时间内能发现更多的缺陷,而且不需要涉及 UI 专家,所以成本较低;传统的启发式评估在各个指标上都比较低,说明它的确不适应协同工作系统的评估;可用性测试虽然可以较好地发现严重的缺陷,但它不能用于早期评估,而且相当昂贵和耗时。显然,使用协作遍历技术对 CPAT1.0 进行可用性评估比其他两种方法更合适。

结束语 可用性评估在协同工作系统的演变周期中扮演了重要的角色,然而传统的面向单用户行为的评测方法却无法适应动态的、分布的、面向多用户的协同工作的需求。本文将面向可用性的工程模型应用到检查技术中,提出了一种面向评估的协同工作系统描述模型。它的基础是由上下文模型、任务模型和快速可用性评估的遍历技术,其中我们不仅集中覆盖了影响可用的协同工作系统的低层面协作特性,还在一定程度上考虑了部分高层面的团队协作环境因素。这种评估方法的可行性在实践中已得到了一定的证明,但样本数量还不够,未来还需要在实践中不断改进。

致谢 忠心地感谢本文作者在澳大利亚 Wollongong 大

学访问交流期间,得到的可用性工程小组对作者的构思提出了宝贵的意见和建议,感谢参与验证实验的所有 HCI 专业人员,感谢对整理实验数据做出贡献的人员。

参考文献

- [1] Grudin J. Groupware and Cooperative Work : Problems and Prospects[C]//The Art of Human Computer Interface Design. Boston:Academic Press,1990
- [2] Gutwin C,Greenberg S. The Mechanics of Collaboration; The Effects of Workspace Awareness Support on the Usability of Real-time Distributed Groupware [J]. ACM Transactions on Computer-Human Interaction,1999,6(3):243-281
- [3] Steves M, Morse E, Gutwin C, et al. A comparison of usage evaluation and inspection methods for assessing groupware usability [C]// International ACM SIGGROUP Conference on Supporting Group Work. US:ACM Press,2002:125-134
- [4] Nielsen J, Mack R. Usability Inspection Methods (Chapter 2: Heuristic Evaluation) [M]. John Wiley and Sons, NY: ACM Press,1994,62-85
- [5] Gutwin C,Greenberg S. The mechanics of collaboration; Developing Low Cost Usability Evaluation Methods for Shared Workspaces[C]// Proceedings of 9th IEEE METICE Workshop on Collaborative Enterprise. Gaithersburg, MD:NIST,2000:14-16
- [6] Johnson H,Hyde J. Towards Modeling Individual and Collaborative Construction of Jigsaws Using Task Knowledge Structures (TKS) [J]. ACM Transactions on Computer-Human Interaction,2003,10(4):339-387
- [7] Tang J. Findings from observational studies of collaborative work[J]. Int. J. Man-Machine Studies,1991,34(2):143-160
- [8] Bekker M,Olson J,Olson G. Analysis of gestures in face-to-face design teams provides guidance for how to use groupware in design [C]// Proceedings 1995 Symposium on Designing Interactive System. Michigan:ACM Press,1995:157-166
- [9] Clark H. Using Language [M]. Cambridge:Cambridge University Press,1996
- [10] Baker K,Greenberg S,Gutwin C. Heuristic evaluation of groupware based on the mechanics of collaboration[C]//Little M,Nigay L,eds. Engineering for Human-Computer Interaction,LNCS Vol 2254. Springer,2002:123-139
- [11] Carroll J M. Introduction to the special issue on "Scenario-Based System Development" [J]. Interacting with Computers,2001,13(1):41-42
- [12] 王丹力,华庆一,戴国忠. 以用户为中心的场景设计方法研究 [J]. 计算机学报,2005,28(6):1043-1047
- [13] 王常青,等. 基于分布式认知理论的扩展资源模型[J]. 软件学报,2005,16(10):1717-1725
- [14] Eriback A L,Höök K. Using Cognitive Walkthrough for Evaluating a CSCW Application[C]// Proceedings' 94 of Computer Human Interaction. Boston:ACM Press,1994:91-93
- [15] Nielsen J. Usability Engineering [M]. Boston:Academic Press,1993