

一种问题驱动基于场景的协同式需求诱导方法

张 伟 刘 峰 赵海燕 梅 宏

(北京大学信息科学技术学院软件研究所 高可信软件技术教育部重点实验室 北京 100871)

摘 要 在软件开发中,大多数的利益相关者并不能清晰地把握他们对于将要开发的软件系统的明确需求.文中提出了一种问题驱动基于场景的协同式需求诱导方法,试图在企业信息系统的上下文中为需求获取问题提供一种现实可行的解决方案.该方法的基本思想是:软件系统的需求不应该以一种主观的方式由利益相关者提出;而应该通过一种“问题识别与解决”的过程,逐步引导利益相关者提出具有客观依据的软件需求.即首先识别出组织的业务现状中存在的重要问题;然后,以解决问题为目标,提出相应的软件需求.基于这种思想,文中采用“场景”的概念作为描述组织业务状况的基本单元,并通过利益相关者之间的协同实现对业务现状的建模、对业务现状中问题的识别、对问题解决方案的确定以及对业务期望的建模等活动,从而实现对软件需求的有效获取.

关键词 需求诱导;场景;协同;软件需求

中图法分类号 TP311

A Problem-Driven Scenario-Based Collaborative Requirements Elicitation Method

ZHANG Wei LIU Feng ZHAO Hai-Yan MEI Hong

(Key Laboratory of High Confidence Software Technology of Ministry of Education,

Institute of Software, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871)

Abstract One problem in software development is that most stakeholders can not clearly think out their needs or requirements for the envisioned software system. This paper proposes a problem-driven scenario-based collaborative requirements elicitation method, trying to provide a practical solution to the requirements elicitation problem in the context of enterprise information systems. The basic idea of this method is that software requirements should not be proposed by stakeholders subjectively, but be elicited following a problem-finding-and-solving process. That is, finding problems in the current business states, proposing solutions to resolving these problems, and then eliciting requirements from these solutions. Based on this idea, this method employs the concept of scenarios as the basic unit to describe enterprises' business states, and to help stakeholders declare their requirements in a collaborative way.

Keywords requirements elicitation; scenarios; collaborative; software requirements

1 引 言

在软件开发中,大多数的利益相关者并不能清

晰地把握他们对于将要开发的软件系统的明确需求.这种现象是由于人们对来事物的不可预见性所导致的.在问题域的现状中引入一个新的软件系统,或者对一个已经存在的软件系统进行演化,都会将

收稿日期:2008-07-11. 本课题得到国家“八六三”高技术研究发展计划项目基金(2007AA010301,2006AA01Z156)、国家自然科学基金(60703065,60528006)资助. 张 伟,男,1978年生,博士,讲师,主要研究方向为领域工程与需求工程. E-mail: zhangw@sei.pku.edu.cn. 刘 峰,男,1977年生,硕士研究生,主要研究方向为需求工程. 赵海燕,女,1966年生,博士,副教授,主要研究方向为领域工程与需求工程. 梅 宏(通信作者),男,1963年生,博士,教授,博士生导师,主要研究方向为软件工程及软件开发环境、软件复用及软件构件技术. E-mail: meih@pku.edu.cn.

利益相关者所熟悉的问题域现状迁移至一个新的状况上,而利益相关者并不能在这个新的状况实际发生之前就对其具有透彻的了解,因此,也就无法准确地把握他们在这个新的状况中可能遇到的各种情况以及相应的需求.基于这种原因,如何有效地获取软件需求,即成为软件开发活动面临的一个非常困难的问题.

需求获取问题的困难性在企业信息系统(Enterprise Information System, EIS)的上下文中得到了进一步的加剧.一般而言, EIS 的主要目的是在特定组织目标的导向下,为组织业务过程的实施提供有效的技术支撑平台.为了实现这个目的,企业信息系统需要面对组织内具有不同职能或角色的部门和人员,管理不同部门和人员的各种业务活动以及这些业务活动产生或所需的各种数据,并为不同业务活动之间的协同提供支持.由于业务过程的复杂性和协同性,涉及其中的任何一个个体往往只能从特定的角度对业务过程中的某个局部进行观察,而很难在全局的视角上完整地理解组织的业务状况.相比较面向单个用户的软件系统而言,在 EIS 中,如何有效地识别业务现状中存在的全局性问题,如何确定对这些问题各方所能共同接受的解决方案,如何保证需求获取结果的完整性和一致性,都将变得更加困难.

对于 EIS 的需求获取问题,传统的需求获取方法存在两点较大的局限性:

(1) 利益相关者的被动参与性

在传统方法中,需求获取人员通过对利益相关者的访谈、问卷调查等方式获得相应的软件需求.在这些过程中,需求获取人员往往居于主导地位;而作为需求来源的利益相关者,在大多数情况下只是被动地参与其中.这种需求获取方式的效率和效果严重依赖于需求获取人员对当前问题域的熟知程度,以及对有效需求的敏感程度.实践证明,缺乏利益相关者充分参与的需求获取过程很难保证获取结果的有效性,并会给整个软件开发项目带来风险^{①[1]}.

(2) 需求提出的主观性

由于对将要开发的软件系统缺乏清晰的理解,利益相关者往往会依靠自身的主观感觉提出需求.而由于缺乏必要的客观依据,这种需求并不能真正反映利益相关者的真实需求,并不可避免地导致需求的频繁变动^[2].传统的需求获取方法对于如何在缺乏全局性视图的情况下,保证需求获取结果的客

观性和有效性,还缺乏足够的支持.

针对上述局限,本文提出了一种问题驱动基于场景的协同式需求诱导方法.其主要关注点是如何在 EIS 的上下文中实施有效的需求获取活动.其基本思想是变“以需求获取人员为主导的需求获取模式”为“以利益相关者相互协同为主导的需求获取模式”.其主要技术手段是采用“场景”作为描述业务现状和期望的基本单元;通过利益相关者之间的协同,建立业务现状的全局视图,进而发现业务现状中存在的问题并识别相应的解决方案,从而自然地过渡至期望的业务状况上并捕获相应的软件需求.其实施方式则主要基于一个支持多用户并行操作的协同式场景建模工具.

本文第 2 节将对本文方法的基本思想进行系统地说明;第 3 节定义本文中场景的基本构成成分,并给出现状场景和期望场景的协同式建模过程;第 4 节在工具支持的层面上,对协同式场景建模工具的核心需求进行分析和说明;第 5 节对与本文相关的重要工作进行介绍、分析和对比;第 6 节对全文进行总结,并对将来的工作计划进行简要地说明.

2 基本思想

本节主要介绍本文方法所基于的 3 个立足点:以利益相关者相互协同为主导的需求获取、问题驱动的需求诱导以及基于场景的业务现状与期望建模.这 3 个方面相互联系,共同构成了本文方法的理论基础.

2.1 以利益相关者相互协同为主导的需求获取

针对传统需求获取方法中利益相关者的被动参与性,本文的方法摒弃了传统的以需求获取人员为主导的需求获取模式,而采用了一种以利益相关者相互协同为主导的模式.图 1 给出了这两种需求获取模式的示意图.

在传统模式中,需求捕获人员在整个需求获取活动中居于中心地位,决定着何时何地从何处采用何种手段获取需求.而其它各种利益相关者只是被动地参与其中,其参与的积极性并不能得到显式的保证.

在以利益相关者相互协同为主导的需求获取模式中,需求获取人员不再居于主导地位,而是作为一

① The Standish Group. CHAOS Report 2007: The Laws of CHAOS. <http://www.standishgroup.com>

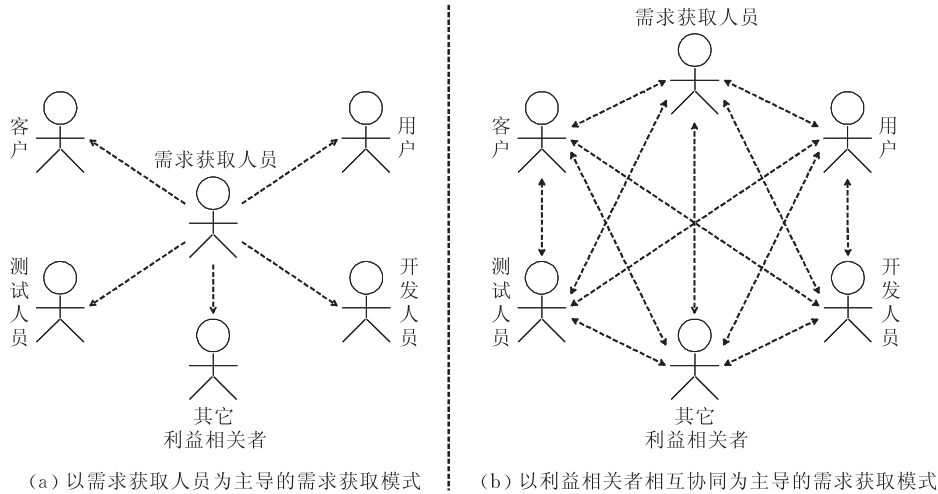


图 1 需求获取模式的转变

种特殊类型的利益相关者,与其它利益相关者以相对平等的身份共同参与到需求获取活动中.其中,任何两个利益相关者之间都存在着相互交流、沟通和协商的信息通道.相比较传统的模式,这种模式对利益相关者积极参与需求获取活动提供了必要的客观基础,并更有利于群体间共识的形成.

2.2 问题驱动的需求诱导

针对传统需求获取方法中需求提出的主观性,本文的方法采用了问题驱动的需求诱导方式.在表象上,软件开发的目的是为了满足一组利益相关者的需求;在更深层次上,其目的则是为了解决现实中存在的问题^[2-3].基于这种观察,我们将“是否存在相关联的问题”作为验证利益相关者所提需求是否具备客观性的标准,即任何一个需求的引入都必须能够解决或部分解决组织业务现状中存在的一个或多个问题.

问题驱动的需求诱导的基本过程如图 2 所示.首先,在组织业务现实状况的基础上,诱导利益相关者发现其中存在的各种问题;然后,进一步诱导利益相关者识别出对这些问题可能的解决方案;最后,针对特定的解决方案,诱导利益相关者建立起组织业务的期望状况,并从中自然地导出相应的软件需求.这里,我们采用“诱导”一词的主要目的是要强调利益相关者在这个过程中的核心作用:诱使利益相关者主动地进行思考,从而自然地引导出他们对于将要开发的软件系统的真正需求.

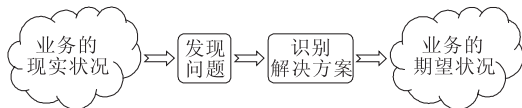


图 2 问题驱动的需求诱导

需要指出的是,“问题驱动的需求诱导”和“以利益相关者相互协同为主导的需求获取”两者之间是紧密联系的.在前者的实施过程中,各种活动都是基于利益相关者之间的相互协同而进行的.协同的目的是为了从更多的视角激发利益相关者进行更为全面和深入的思考.同时,协同也是为了保证各种活动的结果具备必要的群体共识性.

2.3 基于场景的业务现状与期望建模

问题驱动的需求诱导的输入是组织业务的现实状况(以下简称为业务现状),输出则是组织业务的期望状况(以下简称为业务期望).因此,有效地建模业务的现状和期望是决定问题驱动的需求诱导能否成功的必要条件之一.

针对业务现状和期望的建模问题,本文的方法采用了一种分而治之的思想,即将业务的现状和期望分解为一组相对独立的场景,通过对各个场景的建模,建立业务现状和期望的完整视图.一般而言,场景描述了若干实体之间为了完成某个共同的任务而进行的一组动作和交互.以软件系统的边界为基准,通常存在 3 种类型的场景^[4]: (1) 内部场景(由软件系统的内部实体及其之间的交互构成的场景); (2) 边界场景(由软件系统与系统外部实体及其之间的交互构成的场景); (3) 外部场景(完全由软件系统外部的实体及其之间的交互构成的场景).由于主要关注软件需求的获取问题,本文中的场景主要针对后两种类型的场景.

同样,基于场景的业务现状和期望建模也是采用协同的方式进行的.场景为利益相关者之间的协同提供了一个更为具体的上下文,即以完成特定的任务为具体目标,对利益相关者在其中应该实施的动作和交互进行建模.同时,协同式建模也保证了业

务现状建模结果的真实性以及业务期望建模结果的共识性。

3 协同式的场景建模

本节首先给出了本文中场景概念的基本结构，然后在此基础上介绍了采用协同的方式进行现状场景和期望场景建模的基本过程。

3.1 场景的结构

本文中的场景主要由3种成分构成：利益相关者、工作流程和数据（见图3）。其中，场景的利益相关者又分为3种类型：观察者、通信者和参与者。参与者是指直接参与场景执行过程的人员，即参与者将负责执行场景中的特定工作流程。通信者是指那些虽然没有直接参与场景中工作流程的执行，但却可能为其执行提供数据或依赖该场景为其提供数据的利益相关者。观察者是指那些既不是参与者也不是通信者，但仍然与场景存在特定关系的其它利益相关者。例如，对场景的执行或执行结果存在监管责任的利益相关者即是该场景的一类观察者。

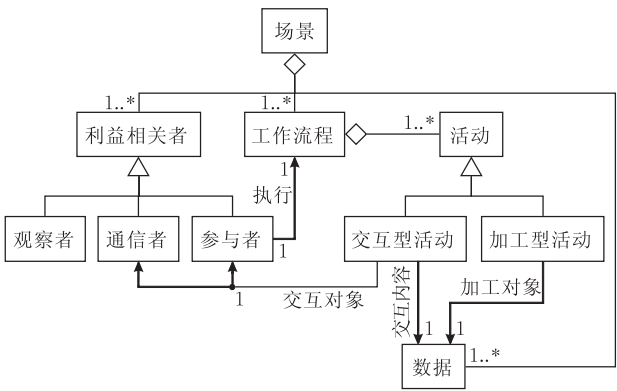


图3 场景的结构

场景还包含了一组工作流程以及其中涉及的一组数据。场景的每个工作流程都将由场景的特定参与者负责执行。工作流程可以进一步被分解为若干活动。本文区分两种类型的活动：加工型活动和交互型活动。前者是指那些会产生数据或对已有数据进行修改的活动；后者是指那些将数据发送给场景的通信者或其它参与者（发送数据型活动），或者从这些利益相关者接收数据的活动（接收数据型活动）。

3.2 现状场景的协同式建模

现状场景建模的目的是形成利益相关者对于组织业务现状的共识，从而尽可能客观地体现出组织业务的真实状况。为了实现这个目的，我们设计了如图4所示的现状场景的建模过程。

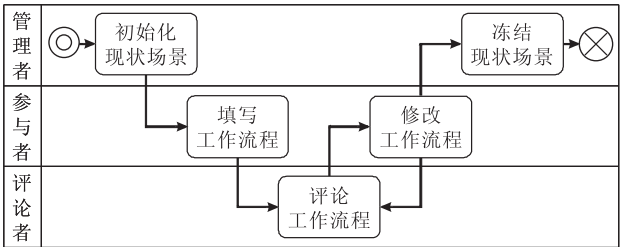


图4 现状场景建模的基本过程

这个建模过程涉及3种建模角色。第1种建模角色是建模过程的管理者，他们负责对一个场景进行初始化，并对一个已经稳定的现状场景进行冻结。第2种建模角色是当前所建模场景的参与者，他们负责填写在当前场景中所执行的工作流程，并根据评论者的反馈对当前的工作流程描述进行修改。第3种建模角色是建模结果的评论者，他们负责对已经填写好的工作流程描述进行评论。由于涉及了3种不同的建模角色，这种建模过程具备了协同的基本前提。具体的协同过程如下所述。

首先，管理者对一个场景进行初始化。初始化的具体内容包括填写场景的名称、场景的各种利益相关者（包括参与者、通信者和观察者）以及场景可能涉及的各类数据等。

在初始化结束后，场景的参与者则可以根据实际的工作经验填写在当前场景中执行的工作流程。一旦工作流程填写完毕后，评论者即可对工作流程的描述进行检查，发现其中存在的二义性、不完整性以及任何与业务现状不相符合的其它问题，并通过添加评论的方式显式地指出这些问题。场景的参与者可以根据这些评论进一步修改或者精化相应的工作流程描述。

在通过多轮的“评论-修改”过程后，现状场景会逐渐趋向一个稳定的状态，即评论者无法或很难从已有的工作流程描述中发现其中存在的与现状的不符合性。在这种情况下，可以认为现状场景的建模结果已经尽可能地逼近了该场景的真实情况。这时，管理者即可以对该场景进行冻结，从而暂时结束当前场景的建模过程。

在场景冻结之后，场景的参与者和评论者将无法修改场景的内容或对场景添加评论。如果确实需要对场景的内容进行修改和评论，则必须由场景管理者对场景解冻后才能进行。

3.3 期望场景的协同式建模

期望场景建模的目的是形成利益相关者对组织业务期望的共识，从而尽可能客观真实地体现出利

益相关者对即将开发的软件系统的需求. 为了实现这个目的,我们设计了如图 5 所示的现状场景的建模过程.

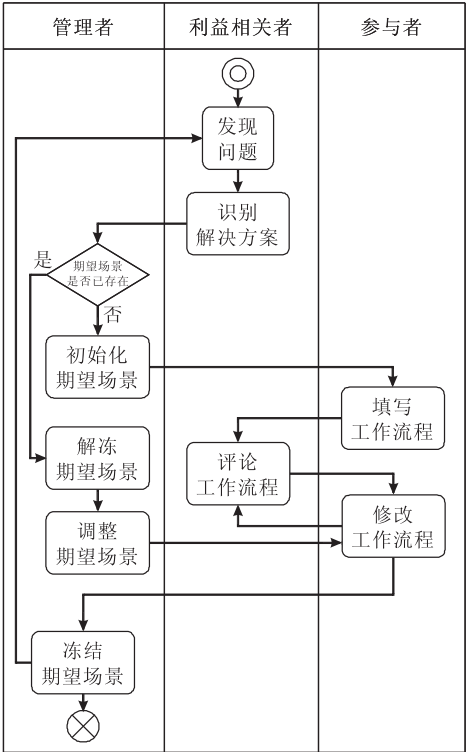


图 5 期望场景建模的基本过程

这个建模过程也涉及 3 种角色:建模过程的管理者、期望场景的参与者以及期望场景的利益相关者. 管理者主要负责对期望场景的初始化、结构调整以及冻结和解冻等活动. 参与者则主要负责填写其需要在当前场景中执行的工作流程,并根据利益相关者的评论对工作流程进行修改或精化. 利益相关者则负责从期望场景的上一版本中发现其中存在的问题,识别相应的解决方案,并在解决方案的指导下对参与者填写的工作流程进行评论. 同样,协同也是该过程的一个重要特点. 具体的协同过程如下所述.

首先,利益相关者在现状场景建模结果的基础上,发现其中存在的问题. 需要指出的是,此处所指的问题与现状场景建模中评论者在工作流程描述中发现的问题存在本质的不同. 后者指的是在工作流程描述中存在的与业务现状的不一致性;对此类问题的解决,可以提高现状场景建模结果与真实场景之间的一致性. 而此处的问题指的是在目前的场景中存在的业务层次上的问题(例如,导致业务流程效率低下的问题,或导致数据冗余的问题等;通常存在于外部场景中)或技术层次上的问题(例如,人机交互过程中的不合理性;通常存在于边界场景中);对

此类问题的解决,可以提升组织实施业务活动的水平和效率,并更好地促成特定组织目标的实现.

在发现问题的基础上,利益相关者需要识别对问题可能的解决方案. 解决方案是指为了解决特定的问题而必须对现状进行的变革的描述. 这种变革包括对现状场景的删除、合并或分割,对新场景的引入,对场景利益相关者和场景数据的修改或设定,对工作责任在场景参与者之间的重新分配等方面.

利益相关者识别出解决方案之后,管理者则可以进行期望场景的初始化,包括对场景名称、场景利益相关者和场景数据的设定等.

场景初始化完毕后,场景的参与者即可根据其理解进行工作流程的填写. 填写完毕后,利益相关者可对根据解决方案对已经填写的工作流程进行检查,发现其中存在的二义性、不完整性以及任何与解决方案不相符合的其它问题,并通过提出问题的方式显式地指出这些问题. 场景的参与者可以根据这些评论进一步修改或者精化相应的工作流程.

在通过多轮的“评论-修改”过程后,期望场景会逐渐趋向一个稳定的状态,即利益相关者无法或很难从当前的工作流程描述中发现其中存在的与解决方案的不符合性. 在这种情况下,可以认为期望场景的建模结果已经尽可能地逼近了解决方案所指向的期望场景. 这时,管理者即可以对该场景进行冻结.

同样,在期望场景冻结之后,场景的参与者和理相关者也将无法修改场景的内容或对场景添加评论. 如果确实需要对场景的内容进行修改和评论,则必须由场景管理者对场景解冻后才能进行.

4 协同式场景建模的工具支持及核心需求

上述协同式场景建模过程的有效性和高效实施必须依赖于有效的软件工具支持. 目前,我们已经完成了支持协同式场景建模过程的软件工具的需求获取和分析活动. 我们在需求获取和分析活动中采用了问题驱动的需求诱导方式. 其中,我们关注的焦点问题是如何在工具支持的层面上尽可能提高上述协同建模过程的高效实施性. 本节将主要对需求分析结果中较为核心的 7 个功能性特征进行介绍.

4.1 工作流程的结构化描述

工作流程描述不仅是场景的核心构成成分之一,也是场景的利益相关者对场景进行评论或从场

景中发现问题的主要依据。工作流程的结构化描述旨在为工作流程提供一种结构化的描述手段,以便于场景的参与者能够清晰有效地描述工作流程以及利益相关者快速准确地理解场景。另外,从 4.3 节可以看到,工作流程的结构化描述也是实现工作流程图形化展示的必要条件。

为了实现工作流程的结构化描述,我们设定了如下的 4 条描述规则:

(1) 对于工作流程中的条件分支结构,使用如下的句式进行描述:“如果”“(条件)”“{”动作序列“}”“否则如果”“(条件)”“{”动作序列“}”[“否则”“{”动作序列“}”]。

(2) 在动作序列描述中,一个语句对应一个动作。动作序列被定义为:(动作“.”| 动作“.”动作序列)。

(3) 对于交互型活动中的发送数据型活动,采用如下的句式进行描述:“将”数据“发送给”角色。

(4) 对于交互型活动中的接收数据型活动,采用如下的句式进行描述:“从”角色“接收”数据。

在这 4 条规则的基础上,软件工具应提供相应的格式化功能以进一步实现场景描述在视觉上的结构化。

4.2 关键字提示与高亮显示

该特征的目的是提高场景建模过程的效率以及提高场景描述和相关信息在视觉上的敏感性,从而提高协同式建模制品的易理解性。关键字提示是指在描述工作流程或对工作流程和场景附加评论和问题的过程中,当建模人员输入一个关键字前缀的文字时,软件工具应自动将所有比配的关键字以列表的形式呈现给建模人员,供建模人员从中选择其所需的關鍵字。关键字高亮显示是指当建模人员从列表中选择了特定的关键字后,该关键字将以不同于非关键字信息的颜色展现在建模制品中。一般而言,应该为不同类型的关键字分别设置不同的高亮颜色,以提高信息的快速辨识性。

协同式场景建模中涉及了 3 种类型的关键字:第 1 种关键字是场景定义中设定的一组利益相关者的名称;第 2 种关键字是场景定义中设定的一组数据的名称;第 3 种关键字是工作流程结构化描述中(参见 4.1 节)规定的一组语法关键字。

4.3 工作流程的图形化展示

在工作流程结构化描述的基础上,软件工具可以进一步实现工作流程的图形化展示特征。该特征的目的仍然是提高工作流程描述的易理解性,从而

便于利益相关者发现其中存在的各类问题。

基于在 4.1 节中建立的结构化描述规则,可以方便地将工作流程的文字描述变换为对应的工作流程图。其中,工作流程描述中的每一个动作对应于工作流程图中的动作图元;动作序列中两个相邻动作的顺序关系对应于工作流程图中两个动作之间的顺序关系图元;条件分支结构对应于工作流程图中的条件分支图元;每一个场景参与者执行的工作流程对应于工作流程图中的一个泳道;每两个匹配的交互型活动在工作流程图中还对应于这两个活动之间的一个数据关系图元。这里,两个匹配的交互型活动是指这两者分别对应于对相同数据的发送型活动和接收型活动,且发送型活动中的数据接收方为接收型活动的执行方,而接收型活动中的数据来源方对应于发送型活动的执行方。

4.4 场景变化的及时通知

该特征的目的是主动地向利益相关推送相应的场景变化信息,以便于利益相关者能够对发生变化的场景及时地进行各种建模活动,从而提高协同建模过程的效率。例如,基于最新添加的评论对场景描述进行及时的修改和精化,基于最新更新的场景描述及时地发现其中存在的业务问题,基于最新提出的业务问题及时地提出相应的解决方案等。

在协同式场景建模过程中,场景变化包含两种类型。一种是场景的构成元素发生了变化。例如,一个场景的利益相关者或数据发生了变化,或者一个场景参与者对应的工作流程发生了变化。另一种是场景关联的评论、问题或问题的解决方案发生了变化。例如,利益相关者对某个场景添加了新的评论或提出了新的问题。

4.5 场景缺陷的自动分析与提示

该特征的目的是通过软件工具自动地分析出场景建模结果中存在的缺陷,并将这些缺陷信息主动地推送给利益相关者,以方便相关人员对这些缺陷进行及时的确认和修复,从而提高协同建模过程的效率。

基于前文中提出的场景结构,场景建模结果中可能会存在如下几种类型的缺陷:

(1) 在场景定义中存在的没有数据被场景中的工作流程描述所引用(造成这种缺陷的原因可能是由于场景的数据定义存在不准确性,或者是因为该场景的工作流程描述存在不完整性)。

(2) 数据不存在生产场景/消费场景(造成这种缺陷的原因可能是由于特定场景的缺失,或者是由

于已有场景的工作流程描述存在错误)。

(3) 场景工作流程描述中的交互型活动涉及的数据接收方或数据来源方并不包含在当前场景的通信者中(造成这种缺陷的原因可能是由于场景的数据定义存在遗漏,或者是因为该场景的工作流程描述存在不准确性)。

(4) 场景工作流程描述中的发送数据型活动/接收数据型活动不存在相匹配的接收数据型活动/发送数据型活动(造成这种缺陷的原因可能是由于特定场景的缺失,或者是由于已有场景的工作流程描述存在错误)。

4.6 场景历史信息的管理

第3节中建立的协同式场景建模过程的一个基本特点是其中包含多种建模角色,且每个角色均有权利对场景的当前建模结果进行修改。多个角色的同时修改往往会导致场景的内容发生频繁的变化。从单一参与者的角度来看,场景内容的频繁变化会对其理解场景带来困惑。因此,需要一种机制使得参与者能够方便地了解场景内容的变化轨迹及变化原因。

该特征的目的是通过记录场景从初始化开始至当前状态经历的所有状态变化,从而帮助协同建模过程的参与者方便地掌握场景的演化轨迹,增强对场景当前建模结果的理解。软件工具应提供多种视角来展示场景的历史信息。例如,可以以场景为单位,或以场景中包含的特定元素为单位,对相关的历史信息进行展示。

4.7 协同建模过程及参与者的性能评估

需要指出的是,第3节中建立的协同式场景建模过程仅仅为利益相关者之间的协同定义了一个基本的流程,其自身并不能保证在具体的流程实例中利益相关者一定会进行充分的协同。因此,在协同建模过程的基础上,还需要一种机制对协同建模过程实例的性能及其中参与者的性能进行评估,以明确在该过程实例中利益相关者之间的协同质量以及特定参与者对这种协同质量的贡献。

该特征的目的在于通过对参与者在协同过程实例中实施的各种活动进行记录,并在此基础上对参与者的性能及过程实例的性能进行综合性的评估,从而明确由该过程实例所产生的制品的可信性。

5 相关工作

等人在1994年提出的模型 Inquiry Cycle Model (ICM)^[5]。ICM模型包含3个基本的活动:需求撰写、需求讨论和需求演化,并通过这3个活动的不断循环实现对需求的获取和分析。虽然ICM模型并没有对需求的具体组织方式进行限制,但Potts等人在实例研究中发现,基于场景的需求建模是一种支持ICM模型的有效技术手段。Potts等人还对需求讨论活动中可能出现的问题的类型进行了细致的分类。但是,Potts等人并没有意识到现状建模对于识别问题的重要作用,因此,ICM模型中并没有现状建模的活动,而是直接基于一个初始的需求文档,进行协同式的需求获取和分析。另外,Potts等人所指场景仅仅局限于边界场景,并不包含业务层面上的外部场景。同时,由于当时技术条件的限制,ICM模型中的协同还是传统的基于集中式会议的协同,缺乏对协同的有效管理和工具支持。

与本文内容密切相关的另一个工作是由德国、英国、法国、比利时学者共同参与的CREWS(Cooperative Requirements Engineering With Scenarios)项目^①。该项目于1996年开始,历时4年,为基于场景的需求工程建立了丰富的理论基础。其中,Jarke等人^[4]从人机交互、软件工程与系统工程以及战略管理等3个研究领域对场景的定义、建模、使用 and 评估等问题进行了系统的阐述。Rolland等人^[6]从内容、形式、目的以及生命周期4个维度出发,建立了场景的分类框架。本文关于场景的理解和分类也是建立在上述两项研究成果的基础之上。CREWS项目的一个重要贡献是其对现状和期望进行了显式的区分,并认为基于场景的现状和期望建模能够更有效地促进利益相关者形成对现状和期望的共识,实现从现状到期望的顺利过渡。这一点和本文基于现状场景的问题发现,以及问题驱动的需求诱导的观点是一致的。另外,CREWS项目还开发了一组支持场景建模与管理的软件工具。需要指出的是,CREWS项目虽然以协同式的需求工程为目标,但在其实际研究中,对协同的关注并不够充分,其采用的协同机制基本还是传统的基于集中式会议的方式。

Hickey等人^[7]针对场景建模的质量问题以及场景描述格式对场景建模质量的影响问题,对3种常用的场景描述格式进行了实例研究。在此基础上,Hickey等人指出,在使用场景的概念进行需求诱导

① CREWS Homepage. <http://sunsite.informatik.rwth-aachen.de/CREWS/>

时,为了提高场景建模结果的完整性和一致性,必须首先建立一个迭代协同式的场景诱导过程,并需要定义一种描述场景的标准模板.本文的方法基本上遵循这样的思想:我们建立了协同式场景建模的基本过程,并定义了场景的结构以及场景中工作流程描述的基本格式.需要指出的是,Hickey 等人的工作并没有显式地区分现状场景和期望场景,也没有意识到问题的识别及解决方案的确定对于提高场景建模质量的正面作用.

Decker 等人^[8]提出了一种轻量级的基于 Wiki 平台的异步协同式需求工程方法.在该方法中,利益相关之间的协同主要是通过对共同感兴趣的页面进行编辑来进行的,即首先由一个发起人建立一个需求主题,并规定该主题的组织结构;在此基础上,对此主题感兴趣的利益相关者即可按照规定的结构对该需求主题的内容进行编辑.同时,利用 Wiki 对网页状态变化的感知能力,利益相关者能够在项目主页上及时地发现那些最新被建立或修改的需求主题,从而实现了与信息变化的及时通知.在这种协同方式的基础上,该方法还定义了一种适用于需求工程的信息组织结构.其中,用况(即本文中的边界场景)来作为组织需求的基本单元.该方法中基于场景的协同式需求工程思想和本文是相同的,但该方法对协同过程的定义还缺乏足够的粒度,其协同过程的实施还主要依赖于利益相关者之间的自发性协调.另外,由于该方法完全基于已有 Wiki 平台的协同机制,因此,在某些方面上还缺乏对协同式需求工程直接和有效的支持.

Geisser 和 Hildenbrand^[9]针对 WinWin^[10]及其扩展版本 EasyWinWin^[11]等需求诱导和分析方法过于复杂且无法适应协同式开发环境的问题,提出了一种在分布式开发环境下工具支持的协同式需求诱导和分析方法 CoREA. CoREA 方法包含两个阶段:协同式的需求诱导;决策支持的需求分析.第一阶段的主要目的是尽可能完整地捕获利益相关者的需求;第二阶段则主要关注如何从捕获到的需求中选择一组适合当前项目上下文的需求集合. CoREA 方法在一种更为一般的层次上定义了协同式需求诱导和分析的过程;本文方法则主要关注如何在企业信息系统的上下文中进行协同式的需求诱导.虽然本文方法中的某些活动已经具有了需求分析的雏形,但与 CoREA 方法中的需求分析相比,还缺乏必要的系统性.

6 总结与后继工作

本文针对企业信息系统的需求获取问题,提出了一种问题驱动、基于场景的协同式需求诱导方法.该方法的基本思想是变传统的“以需求获取人员为主导的需求获取”为“以利益相关者相互协同为主导的需求获取”;使用场景作为描述组织状况的基本手段;并通过从现状场景中发现问题的发现以及对问题解决方案的确定,实现从现状场景到期望场景的过渡.在此基础上,本文给出了场景的具体构成,建立了协同式现状场景建模和期望场景建模的基本过程;并在支持工具的层面上,指出了协同式场景建模工具应该具备的 7 个关键性软件特征.

目前我们已经开发了一个基于 Web 的协同式场景建模工具原型,其中实现了对工作流程的结构化描述、关键词提示与高亮显示、工作流程的图形化展示以及场景历史信息的管理等 4 个关键特征.在后继工作中,我们将继续对这个工具原型进行完善,并选择若干领域进行实例研究;同时,我们还将协同式需求诱导的基础上,对协同式思想在需求工程其它活动中的应用进行研究,并最终建立起系统化的协同式需求工程方法及支持工具.

参 考 文 献

- [1] Emam K El, Quintin S, Madhavji N H. User participation in the requirements engineering process: An empirical study. *Requirements Engineering*, 1996, 1(1): 4-26
- [2] Wiegers K E. *Software Requirements*. 2nd Edition. Redmond, Washington, USA: Microsoft Press, 2003
- [3] Jackson M. *Problem Frames: Analyzing & Structuring Software Development Problems*. Indiana, USA: Addison-Wesley, 2001
- [4] Jarke M, Bui X T, Carroll J M. Scenario management: An interdisciplinary approach. *Requirements Engineering*, 1998, 3(1): 155-173
- [5] Potts C, Takahashi K, Anton A I. Inquiry-based requirements analysis. *IEEE Software*, 1994, 11(2): 21-32
- [6] Rolland C, Achour C B, Cauvet C, RalytéJ, Sutcliffe A, Maiden N, Jarke M, Haumer P, Pohl K, Dubois E, Heymans P. A proposal for a scenario classification framework. *Requirements Engineering*, 1998, 3(1): 23-47
- [7] Hickey A M, Dean D L, Nunamaker J F. Setting a foundation for collaborative scenario elicitation//*Proceedings of the 32nd Annual Hawaii International Conference on System Sciences*. Maui, HI, USA, 1999

[8]

Decker B, Ras E, Rech J, Jaubert P, Rieth M. Wiki-based stakeholder participation in requirements engineering. *IEEE Software*, 2007, 24(2): 28-35

[9]

Geisser M, Hildenbrand T. A method for collaborative requirements elicitation and decision-supported requirements analysis//*Proceedings of the IFIP 19th World Computer Congress, First International Workshop on Advanced Software Engineering*. Boston, USA, 2006: 108-122

[10]

Boehm B, Egyed A, Kwan J, Port D, Shah A, Madachy R. Using the WinWin spiral model: A case study. *IEEE Computer*, 1998, 31(7): 33-44

[11]

Gruenbacher P. Collaborative requirements negotiation with EasyWinWin//*Proceedings of the 11th International Workshop on Database and Expert Systems Applications (DEXA'00)*. London, United Kingdom, 2000: 954-958



ZHANG Wei, born in 1980, Ph. D. , assistant professor. His research interests include domain engineering and requirements engineering.

LIU Feng, born in 1977, master candidate. His re-

search interest focuses on requirements engineering.

ZHAO Hai-Yan, born in 1966, Ph. D. , associate professor. Her research interests include domain engineering and requirements engineering.

MEI Hong, born in 1963, Ph. D. , professor, Ph. D. supervisor. His research interests include software engineering and software development environment, software reuse and software component technologies.

Background

This work is supported by National High Technology Research and Development Program of China under grant No. 2007AA010301 and No. 2006AA01Z156, and National Natural Science Foundation of China under grant No. 60528006 and No. 60703065.

Domain engineering and application engineering are two important activities in software reuse. However, in current software reuse approaches, there still lacks sufficient support to achieve the continual evolution between domain engineering and application engineering. The work in this paper is an important part to achieve the continual evolution between the two activities. This work's functions are to provide an web-based requirements elicitation environment and to achieve the effective requirements elicitation and requirements modeling. After enough software requirements in a specific domain are

elicited and modeled, the domain engineer then can start the domain analysis activity, that is, identifying reusable requirements from these existing requirements for specific applications. After domain reusable requirements are identified, the requirements engineer can carry out the requirements elicitation activity in a more efficient way, based on the reuse of these reusable requirements.

In a narrow sense, the work in this paper belongs to the requirements elicitation activity in requirements engineering. Currently, the authors also have developed a tool prototype to support the method proposed by this paper. In the future research, the authors will investigate the feasibility of this method and the tool by applying them to real applications, and improve them based on the feedback from these feasibility studies.