

对等实体执著度语义及其应用

詹 敏,左 春

(中国科学院软件研究所,北京 100080)

摘 要: 对等实体的互操作语义是由心理态度和行为逻辑以及一阶情态语言描述的,它为互操作的实现提供了一个无歧义的参考点.传统的持续目标在语义上存在一定的缺陷,使互操作的发起者缺乏对互操作的有效控制.为了解决这一问题,本文提出了执著度概念,它是指对等实体对实现互操作的承诺程度.基于执著度的执著目标是对持续目标的有效改进,它实现了对不同承诺程度的互操作的支持.

关键词: 对等实体;互操作;持续目标;执著度;执著目标

中图分类号: TP311 **文献标识码:** A **文章编号:** 0372-2112 (2003) 02-0287-03

A Formal Semantics for Degree of Perseverance of Peers and Its Application

ZHAN Min, ZUO Chun

(Institute of Software, Chinese Academy of Sciences, Beijing 100080, China)

Abstract: The semantics for interoperations among peers is expressed in a logic of mental attitudes and action, formalized in a first order modal language, and provides an unambiguous reference point for implementations of interoperations. There exists some defects in the semantics of traditional persistent goal, which make initiator lose their controls of interoperations. In order to solve this problem, this paper brings forward the concept of degree of perseverance, which means the grade of commitment of interoperations. Perseverant goal, based on the concept of degree of perseverance, is an improvement on persistent goal, and realizes supports of different levels of commitment for interoperations.

Key words: peer; interoperation; persistent goal; degree of perseverance; perseverant goal

1 引言

对等实体的互操作是基于对等计算的企业应用集成技术实现异构应用系统信息共享的基本方式,这种互操作从本质上来讲是一系列的消息传递和交互过程.当某个对等实体发送一个消息时,它往往希望消息的接收者能够以一定的方式对该消息做出应答.但是,传统的消息传递机制并没有要求消息中必须包含这种期望,因此不能很好地满足互操作对上下文环境的要求^[1].

人们已经充分认识到了这种不足,并在相关的研究工作中^[2,3]提出了对话机制的概念和用于描述对话的语言.同时,为了更好、更深入地了解这些语言的内在含义,为语言的应用提供一个无歧义的参考点,保证由各独立团队和机构所开发的应用系统能够有效地实现彼此间的信息共享,人们利用心理态度和行为逻辑以及一阶情态语言对这类语言的语义进行了形式化表述^[4,5].

2 互操作语义基础

对等实体是独立、自治的软件实体,人们利用心理态度和行为逻辑以及一阶情态语言对互操作过程中的确信、目标、计划、意图、承诺以及行为等要素进行形式化的表述^[6].在本节以及后续各节中,我们采用如下的符号标记: $p, p_1, p_2, \dots$ 表示互操作的命题; α, β 表示逻辑命题的变量; $x, x_1, x_2, \dots, y, y_1, y_2, \dots$ 表示参与互操作的对等实体.

假定 M 代表某个逻辑模型, σ 代表事件序列, n 为表示事件索引点的整数, v 代表变量,则 M, σ, n, v 代表了一个互操作空间.我们用表达式 $M, \sigma, n, v \models \alpha$ 表示在 M, σ, n, v

所确定的空间中逻辑命题 α 可满足.如果对于所有的 M, σ, n, v, α 都可满足,就称 α 是成立的,记为 $\models \alpha$.

互操作的一些基本语义如下:

- ① $p_1 \leq p_2$: p_1 是 p_2 的首序列;
- ② $p_1 \supset p_2$: p_1 蕴涵 p_2 ;
- ③ $\text{HAPPENS}(x, p)$: p 是将要发生的操作;
- ④ $\text{DONE}(x, p)$: p 是刚刚发生的操作;
- ⑤ $\text{BEL}(x, p)$: x 确信 p 的真实性;
- ⑥ $\text{GOAL}(x, p)$: x 以 p 为互操作的实现目标;
- ⑦ $p?$: 测试语义,表明操作 p 必须在某个操作序列中存在;
- ⑧ p^+ : 重复语义,表明操作 p 至少连续发生一次;
- ⑨ $p_1; p_2$: p_1 与 p_2 形成一个操作序列;
- ⑩ $p_1 \mid p_2$: p_1 与 p_2 至少有一个发生.

与本文论述相关的基本定理如下:

定理 1 $p \supset \text{BEL}(x, p)$

定理 2 $\text{BEL}(x, p) \supset \neg \text{BEL}(x, \neg p)$

利用上述的基本语义,我们接着给出一些与本文相关的扩展语义和运算:

- ① $p \stackrel{\text{def}}{=} \exists a \text{HAPPENS}(x, a); p?$: p 操作最终会发生 (eventually);
- ② $\Box p \stackrel{\text{def}}{=} \neg \neg p$: p 操作总是会发生 (always);
- ③ $\text{LATER}(p) \stackrel{\text{def}}{=} \neg p$: p 操作将在以后发生;
- ④ $\text{BEFORE}(p_1, p_2) \stackrel{\text{def}}{=} \forall c (\text{HAPPENS}(x, c); p_2) \supset \exists a (a \leq c) (\text{HAPPENS}(x, a); p_1)$: p_1 操作在操作 p_2 之前发生;

收稿日期:2002-03-26;修回日期:2002-08-10

基金项目:国家自然科学基金重点项目(No. 19831020)

为了描述对等实体为实现互操作所做的努力,人们在上述语义的基础上定义了一个新的语义:持续目标.具有持续目标的对等实体只有在确信目标已经得到满足或永远无法满足的情况下才会放弃该目标.于是,我们有如下的表述:

$$P - \text{GOAL}(x, p) \stackrel{\text{def}}{=} \text{GOAL}(x, \text{LATER}(p)) \quad \text{BEL}(x, \neg p) \\ \text{BEFORE}(\text{BEL}(x, p) \quad \text{BEL}(x, \square \neg p), \neg \text{GOAL}(x, \text{LATER}(p)))$$

也就是说,持续目标的逻辑含义是:对等实体的目标是实现操作 p , 并且它相信目前 p 操作尚未实现, 对等实体将为实现操作 p 做出持续的努力, 直到它确信 p 已经实现或永远无法实现. 持续目标与普通目标 (GOAL) 的差别在于为实现互操作所做的努力, 普通目标可以随时放弃正在试图进行的互操作, 而持续目标则保证尽最大可能地实现该操作.

持续目标的概念是基础而重要的, 它体现了对等实体为实现互操作所做的承诺. 在持续目标的基础上, 人们定义了一系列用于描述互操作的语义, 如 INFORM, REQUEST, PROXY, CONFIRM 等, 并利用这些语义完成了互操作语法的描述. 以 INFORM 为例, 它语义的形式表述是:

$$\text{INFORM}(x, y, \alpha) \stackrel{\text{def}}{=} \text{BEL}(x, \alpha) \quad \neg \text{BEL}(x, \text{BEL}(y, \alpha)) \quad P - \\ \text{GOAL}(x, \text{BEL}(y, \alpha))$$

对等实体 x 必须首先相信 α 的真实性, 同时它相信 y 不知道 α 的真实性, 并且它的持续目标是最终使 y 确信 α 是真实的, 这就是 INFORM 操作的含义. 与之相应的语法描述如下:

(INFORM

:sender x
:receiver y
:content α
:language java)

3 执著度

在一般情况下, 由持续目标所定义的互操作为实现目标所做的努力是令人满意的: 对等实体将尽可能地实现该目标, 直到确信目标已经实现或永远无法实现为止. 但是, 持续目标在语义上存在一定的缺陷. 假设对等实体 x 向 y 发出互操作请求, 希望通过该互操作向 y 提供某些信息. 于是它设定了一个持续目标用于实现该任务. 但是, 当互操作开始后, x 将发现它对自己的行为失去了控制: 它无法在确信目标实现或永远无法实现之前放弃该目标. 而这种控制对于进行互操作的实际应用系统而言常常是必须的, 它们并不会总要求只能在确信任务已经实现或永远无法实现的情况下终止互操作. 相反, 由于互操作的实现往往同时还受限于系统资源以及任务性质等诸方面的因素, 因此, 持续目标的语义应当做适当的调整, 允许能够根据任务性质/要求的不同为互操作的实现提供不同程度的承诺. 例如, 企业的信息发布系统在向其他应用系统发布信息时, 可能并不介意对方是否能够成功地接收, 因此只需要为这类互操作提供低程度的实现承诺. 而对于企业关键应用之间的互操作, 则可能要求必须尽最大可能地保证它们的成功.

在为实现异构环境下的企业应用集成而设计的 UECM (Universal Enterprise Computing Model) 中, 我们提出了执著度

(degree of perseverance) 的概念, 用于描述对等实体对实现互操作的承诺程度. 一旦对等实体意识到它为实现目标所做的努力已经达到了预设的执著度的要求, 即使此时还无法确信目标是否能够实现, 也将放弃该次互操作.

可以有很多的相关属性用于描述对等实体对目标的执著程度, 如执行时间、尝试次数、占用资源等. 在 UECM 中, 我们采用对目标的尝试次数 n 来描述互操作的执著度. 以下是与执著度相关的一些定义:

定义 1 $\text{ATTEMPT}(x, p) \stackrel{\text{def}}{=} (\text{BEL}(x, \text{HAPPENS}(x, p))) ? ; p$

x 尝试执行 p 操作. 需要注意的是, 这种尝试并不是一种简单的执行承诺, 因为简单的执行承诺可能导致 x 无意识、无计划地去执行 p 操作. 因此, x 必须首先确信它的下一步动作是执行 p , 然后才能真正开始实现 p 操作.

定义 2

$$\text{ATTEMPT}(x, p, n) \stackrel{\text{def}}{=} (\text{BEL}(x, \text{DONE}(x, ((\neg \text{BEL}(x, p) \\ \text{BEL}(x, \square \neg p)))) ? ; \text{ATTEMPT}(x, p))^n)$$

检验 x 为实现 p 操作所付出的努力是否已经满足预先设定的执著度的要求. 在每一次尝试开始的时候, x 首先判断自己是否已经确信 p 操作已经实现或永远无法实现, 如果尚未确信, 那么 x 紧接着将试图执行 p 操作. 当 x 确信这种尝试已经进行完第 n 次时, 就认为自己为实现目标所做的努力已经满足执著度的要求.

为了便于下文的论述, 用 $\text{ATTEMPTIM}(x, p, m)$ 表示 x 为实现 p 操作已经进行了 m 次努力, 即:

$$\text{ATTEMPTIM}(x, p, m) \stackrel{\text{def}}{=} \text{DONE}(x, ((\neg \text{BEL}(x, p) \\ \text{BEL}(x, \square \neg p)))) ? ; \text{ATTEMPT}(x, p))^m)$$

进一步地, 我们定义 $\phi(p, i)$ 如下:

$$\phi(p, i) \stackrel{\text{def}}{=} ((\neg \text{BEL}(x, p) \quad \text{BEL}(x, \square \neg p))) ? ; \\ \text{ATTEMPT}(x, p))^i$$

因此, $\text{SATISFY}(x, p, n)$ 和 $\text{ATTEMPTIM}(x, p, m)$ 的定义可以简化为:

$$\text{SATISFY}(x, p, n) \stackrel{\text{def}}{=} \text{BEL}(x, \text{DONE}(x, \phi(p, n)))$$

$$\text{ATTEMPTIM}(x, p, m) \stackrel{\text{def}}{=} \text{DONE}(x, \phi(p, m))$$

由执著度的定义可以知道, 如果 x 为实现目标所尝试的次数少于执著度要求的次数, 那么我们可以认为目标的执著度没有得到满足, 即推论 1.

推论 1 $\forall m \text{ATTEMPTIM}(x, p, m) \quad (m < n) \supset \neg \text{SATISFY}(x, p, n)$

证明 根据 $\text{DONE}(x, p)$ 的定义, $\text{DONE}(x, \phi(p, m))$ 可以表述为: $\exists m_1, m_1 \leq m$, 有 $M, \sigma, v, m_1 [\phi(p, m)] m$ 成立. 同样地, $\text{DONE}(x, \phi(p, n))$ 可以表述为: $\exists n_1, n_1 \leq n$ 有 $M, \sigma, v, n_1 [\phi(p, n)] n$. 当 $m < n$ 时, 可以得到 $\exists m_1, n_1, m_1 \leq m, m \leq n_1, n_1 \leq n$, 有 $M, \sigma, v, m_1 [\phi(p, m)] m \dots n_1 [\phi(p, n)] n$ 对于任意的 m 都成立. 因此我们得到

$$\forall m \text{DONE}(x, \phi(p, m)) \quad (m < n) \supset \neg \text{DONE}(x, \phi(p, n)).$$

由定理 1, 2 及 ATTEMPTIM 的定义, 有

$\forall m \text{ATTEMPTIM}(x, p, m) \quad (m < n) \supset \text{BEL}(x, \neg$
 $\text{DONE}(x, \phi(p, n))) \supset \neg \text{BEL}(x, \text{DONE}(x, \phi(p, n)))$
 $\supset \neg \text{SATISFY}(x, p, n).$ 证毕.

同样地,如果 x 为实现目标所尝试的次数等于或大于执著度要求的次数,那么我们可以认为目标的执著度已经得到满足,即推论 2.

推论 2 $\forall m \text{ATTEMPTIM}(x, p, m) \quad (m \geq n) \supset \text{SATISFY}(x, p, n)$

4 执著目标

在执著度概念的基础上,对持续目标的语义进行了改进,提出了执著目标的概念,具体定义如下:

$\text{PGOAL}(x, p, n) \stackrel{\text{def}}{=} \text{GOAL}(x, \text{LATER}(p)) \quad \text{BEL}(x, \neg p)$
 $\text{BEFORE}((\text{BEL}(x, p) \quad \text{BEL}(x, \square \neg p))$
 $\text{SATISFY}(x, p, n)), \neg \text{GOAL}(x, \text{LATER}(p)))$

与持续目标的语义相对比,执著目标在它的基础上增加了一个放弃目标的条件:如果互操作的发起者 x 为实现目标所做的努力已经满足执著度的要求,那么 x 将放弃这次互操作.这样做的好处是加强了互操作的发起者对互操作过程的控制:互操作的发起者不仅能够制定互操作的目标,还能够根据实际需求的不同为不同的互操作规定不同的执著程度,从而实现对系统资源更加有效、合理的利用.

推论 3 $P - \text{GOAL}(x, p) \equiv \text{PGOAL}(x, p, \infty)$

证明 由推论 1 知道对于任意 m ,均有 $\neg \text{SATISFY}(x, p, \infty)$ 成立.也就是说, x 为实现目标所做的努力永远无法满足执著度的要求.由 PGOAL 的定义,我们可以得到

$\text{PGOAL}(x, p, n) \equiv \text{GOAL}(x, \text{LATER}(p)) \quad \text{BEL}(x, \neg p)$
 $\text{BEFORE}((\text{BEL}(x, p) \quad \text{BEL}(x, \square \neg p))$
 $\text{SATISFY}(x, p, \infty)), \neg \text{GOAL}(x, \text{LATER}(p)))$

由于 $\text{SATISFY}(x, p, \infty)$ 恒不满足,所以上式可以简化成:

$\text{PGOAL}(x, p, n) \equiv \text{GOAL}(x, \text{LATER}(p)) \quad \text{BEL}(x, \neg p)$
 $\text{BEFORE}((\text{BEL}(x, p) \quad \text{BEL}(x, \square \neg p)),$
 $\neg \text{GOAL}(x, \text{LATER}(p)))$

即 $\text{PGOAL}(x, p, \infty) \equiv P - \text{GOAL}(x, p).$ 证毕.

可见,持续目标只是执著目标的一个特例.以执著度为基础的执著目标从语义上为互操作提供了很大的灵活性.

5 应用

让我们在执著目标语义的基础上重新考察前面讲述的 INFORM 操作.利用执著目标对 INFORM 进行重新定义:

$\text{INFORM}(x, y, \alpha, n) \stackrel{\text{def}}{=} \text{BEL}(x, \alpha) \quad \neg \text{BEL}(x, \text{BEL}(y, \alpha))$
 $\text{PGOAL}(x, \text{BEL}(y, \alpha), n)$

对应的语法描述为:

(INFORM
 $\text{:sender } x$
 $\text{:receiver } y$
 $\text{:content } \alpha$
 $\text{:perseverance } n$
 $\text{:language java})$

对互操作执著度的要求体现在 INFORM 语法的 perseverance 域中,不同的执著度体现了互操作的发起者 x 对目标的

不同实现要求.下面我们将举例说明执著度的应用.

企业 A 的应用系统已经实现了与其他应用系统的互操作. A 的报表发布系统 x 需要定期或不定期地向相关的应用系统发布报表数据.这些应用系统包括企业内部的决策支持系统 y_1 、集团总公司的报表汇总审计系统 y_2 以及对企业 A 运营状况感兴趣的潜在的客户(或股东)的相关应用系统 y_3 等. x 采用预约/发布模型向上述这些应用系统发布数据:应用系统向 x 发出预约报表数据请求,新报表生成后, x 将使用 INFORM 操作将报表数据发往这些应用系统.但是,这些应用系统的重要性并不相同: y_1 是企业 A 的关键应用,必须保证报表数据准确及时的传送; y_2 是上级部门的汇总应用,应当尽可能地保证数据的传输;而 y_3 属于一般应用, x 不需要为数据的成功传输做太高等度的承诺.在 UECM 中,我们利用具有不同执著度的 INFORM 操作成功地实现了对 y_1 、 y_2 、 y_3 的不同发布需求,而在传统的 INFORM 操作中,这点是无法实现的,人们必须为不同的发布需求定义不同的操作.

6 小结

执著度是对等实体实现互操作目标过程的努力程度的量化表示,它有效地加强了互操作的发起者对互操作过程的控制.相对于传统的持续目标,执著目标具有更大的灵活性,能够更好地适应不同互操作的需要.在为实现异构环境下的企业应用集成而设计的 UECM 结构中,我们设计了基于执著目标语义的互操作语法,有效地实现了对不同承诺程度的互操作的支持.

参考文献:

- [1] R Scott Cost, Tim Finin, Yannis Labriou, et al. An agent-based infrastructure for enterprise integration [A]. First International Symposium on Agent Systems and Applications (ASA '99) & Third International Symposium on Mobile Agents (MA '99) [C]. 1999.
- [2] Finin T, Labriou Y. A proposal for a new KQML specification [R]. TR CS-97-03, CSEE Dept, UMBC, Baltimore, MD, 1997.
- [3] FIPA ACL Specification [EB/OL]. <http://www.fipa.org/>, 2000.
- [4] Cohen P R, Levesque H J. Performatives in a rationally based speech act theory [A]. Proc. of the 28th Annual Meeting of the Association for Computation Linguistics [C]. 1990.
- [5] Sadek M D. A study in the logic of intention [A]. In Proceedings of KR '92 conference [C]. Cambridge, MA. 1992. 462 - 473.
- [6] Cohen P R, Levesque H J. Intention is choice with commitment [J]. Artificial Intelligence, 1990, 42(2 - 3): 213 - 262.

作者简介:



詹 敏 男, 1974 年 4 月生于福建闽清, 中国科学院软件研究所博士研究生, 研究方向为大型数据库与网络工程.

左 春 男, 1959 年 4 月于辽宁省, 中国科学院软件研究所研究员, 研究方向为大型数据库与网络工程.