

FIPA 规范: Agent 系统的开放标准

FIPA Specifications: Open Standards for Agent-Based Systems

张德同 周明全 耿国华

(西北大学计算机科学系 西安 710069)

Abstract FIPA is an international organization formed by companies and academic institutions, with the aim of producing software standards for heterogeneous and interacting Agents and Agent-based systems. FIPA specifications includes five parts: Abstract Architecture, Agent Message Transport, Agent Management, Agent Communication, and Agent Applications. Each FIPA specification is in a life cycle, and it will finally reach the standard state or obsolete state. This paper provides a concise analysis and introduction of the current FIPA specifications, together with some Agent platforms conforming to the FIPA specifications.

Keywords FIPA, Agent-based systems, Interoperating, FIPA specifications, Agent platform

1. 引言

FIPA (The Foundation for Intelligent Physical Agents) 是一个由活跃在 Agent 领域的公司和学术机构组成的国际组织,其目标是为异质的 Agent 和 Agent 系统之间能够互操作而制订相关的软件标准。^[1]

FIPA 规范的制订过程是开放的,它的每一个会员组织都有权对规范提出自己的意见和建议,在充分讨论的基础上达成一致。FIPA 组织本身也是开放的,任何公司、政府机构和国际组织都可以申请加入 FIPA。FIPA 与 OMG (Object Management Group) 等标准化组织也保持着密切的工作关系。

所有 FIPA 规范都处在一个生命周期中,整个生命周期包括以下五个状态:

- 预备状态:由技术委员会(TCs)提出的原始规范;
- 实验状态:预备状态的规范需要送交 FIPA 体系结构委员会(FAB),经审查批准转为实验状态。
- 标准状态:如果实验状态的规范在一个以上遵从 FIPA 规范的 agent 平台上成功实现,并且得到 FIPA 体系结构委员会和 FIPA 各成员的认可,这项规范就转为标准状态。
- 反对状态:如果由于技术的变化或 FIPA 其它规范的变化,使得某项 FIPA 规范没有必要继续存在,则转为反对状态。
- 废止状态:处于反对状态的规范经过六个月的过渡期,转为废止状态。

下面结合 FIPA 公布的最新规范,对 FIPA 规范的各组成部分做简要介绍。

2. FIPA 规范的组成部分

FIPA 规范包括五大部分,即抽象体系结构、Agent 消息传输、Agent 管理、Agent 通讯和基于 Agent 的应用。

2.1 抽象体系结构

FIPA 抽象体系结构规范[FIPA00001]是贯穿整个 FIPA 规范的核心规范,其目标是实现互操作性和可重用性。在制订规范的过程中,借鉴了面向对象技术、分布式计算模型、已实现的各种 Agent 系统以及其它重要的软件系统,如 Java、CORBA、DCOM 等。

抽象过程包括两方面:一是找出体系结构的基本元素,二是找出基本元素之间的基本关系。FIPA 共抽象出 24 种基本元素,5 组基本关系。

FIPA 抽象出的基本元素包括:Agent、Agent 通讯语言、内容、内容语言、目录项、目录服务、信封、消息、编码表达法、编码转换服务、消息传输服务、传输描述等。对体系结构的每个基本元素,都从以下四个方面进行说明:摘要、与其它基本元素的关系、行为、背景描述。只有服务性元素才有行为。

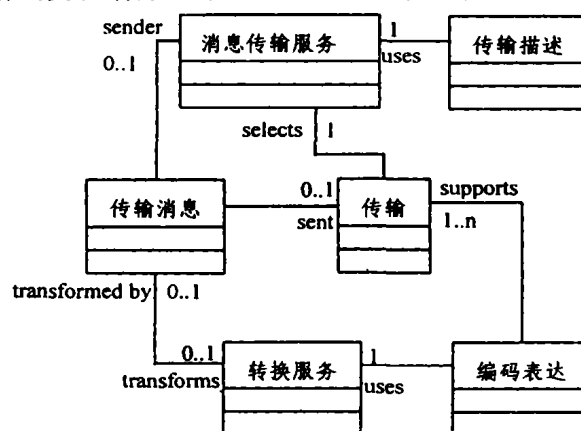


图1 UML-消息传输元素

例如,FIPA 对目录服务的说明主要如下:

• 摘要 目录服务是一个共享信息库,Agent 可以在这里发布它们的目录项,也可以在这里搜索感兴趣的目录项。目录项中包含 Agent 的名字、地址等属性值。

• 与其它基本元素的关系

Agent 可以在一个目录服务中注册它的目录项。
Agent 可以在一个目录服务中修改它的注册目录项。
Agent 可以在一个目录服务中删除它的注册目录项。
Agent 可以在一个目录服务中查询感兴趣的目录项。

• 行为

(1)注册 注册时,Agent 向目录服务提供自己的目录项,目录服务验证参数的合法性后进行注册。如果注册成功,目录服务将返回一个表示成功的行为状态,否则返回一个表示失败的行为状态。

(2)修改 修改时,Agent 向目录服务提供一个与待修改

目录项具有相同 Agent 名的目录项,目录服务验证 Agent 的权限和参数的合法性后,搜索相应的目录项。找到相应的目录项后,用新目录项中的值替换旧目录项中的值。如果新目录项中某个属性的值为空值,则删除旧目录项中的相应属性。如果修改成功,目录服务将返回一个表示成功的行为状态,否则返回一个表示失败的行为状态。

(3)删除 删除时,Agent 向目录服务提供一个与待删除目录项具有相同 Agent 名的目录项,目录服务验证 Agent 的权限和参数的合法性后,搜索并删除相应的目录项。如果删除成功,目录服务将返回一个表示成功的行为状态,否则返回一个表示失败的行为状态。

(4)查询 查询时,Agent 向目录服务提供一个目录项作为搜索模式,目录服务验证参数的合法性后,搜索相应的目录项。如果存在与搜索模式完全匹配的目录项,则查询成功。查询成功时,目录服务将返回一个表示成功的行为状态和一组满足搜索模式的目录项,否则返回一个表示失败的行为状态。

·背景描述

目录服务可以用多种方法实现,如像 X.500 这样的通用方案,或某种 Agent 系统的专用机制。典型的目录服务使用某种分层结构或联邦模式。目前,多数目录系统如 LDAP、X.500 和 NIS 还不能提供足够灵活的搜索服务。有可能利用 Prolog 或 LISP 这种允许复杂嵌套结构的语言,实现类似中介的复杂搜索服务系统。

FIPA 抽象出的 5 组基本关系为:Agent 关系、传输消息关系、目录项关系、消息元素关系、消息传输元素关系。每组基本关系都用 UML(Unified Modelling Language)^[2]图表示,图 1 给出了消息传输元素关系的 UML 图示。

从这些基本元素和基本关系,能派生出各种可能的具体体系结构。由于这些具体的体系结构共享同一抽象体系结构,所以能够实现互操作。

2.2 Agent 消息传输

FIPA Agent 消息传输规范主要处理不同网络传输协议之间的消息传输和表达,包括有线和无线环境。在消息传输层,一个消息由信封和消息体组成。信封包含具体的传输要求,以及其它供消息传输服务(MTS)使用的支持信息。信封由一组参数定义,包括:to,from,date 以及 acl-representation 等。消息体包含具体的消息内容,通常用 FIPA ACL 表达。与 FIPA 规范兼容的 Agent 平台上,都有相应的消息传输服务 MTS。MTS 由 Agent 通讯频道 ACC 提供,ACC 是 Agent 平台上的通讯实体。

Agent 消息传输的参考模型如图 2 所示。

该模型提供以下支持功能:

- 对 Agent 平台中消息传输服务 MTS 的支持[FIPA00067]。

- 支持多种消息传输协议,如 OMG IIOP(Internet Inter-ORB Protocol)[FIPA00075]、HTTP[FIPA00084]以及 WAP(Wireless Application Protocol)[FIPA00076]。

- 支持与特定消息传输协议对应的信封表达法,如与 HTTP 对应的 XML 编码方法[FIPA00085],与 WAP 对应的高效二进制编码方法[FIPA00088]。

- 支持 FIPA ACL 表达法,如串编码方法[FIPA00070],XML 编码方法[FIPA00071],以及高效二进制编码方法[FIPA00069]。

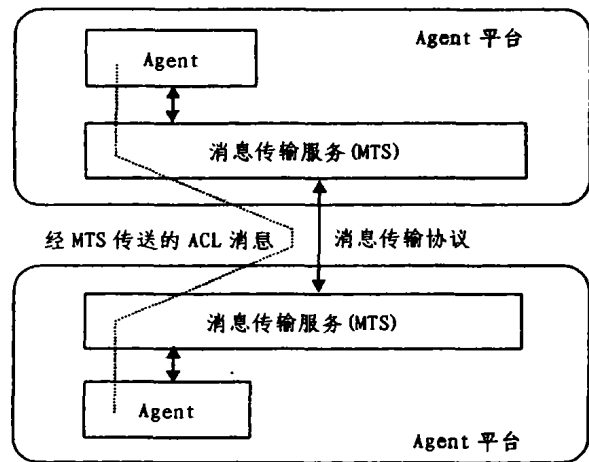


图 2 Agent 消息传输参考模型

利用 FIPA 支持的消息传输协议,可以实现异质 Agent 平台间的消息传输。除了上述跨平台的消息传输协议,还可以支持 Agent 平台的内部消息传输协议(如 Java RMI),不同协议之间的转换工作由消息传输服务 MTS 完成。

2.3 Agent 管理

FIPA Agent 管理规范[FIPA00023]提供 FIPA Agent 存在和运作的框架,为 Agent 创建、注册、定位、通讯、迁移和引退,建立了一个逻辑参考模型,如图 3 所示。

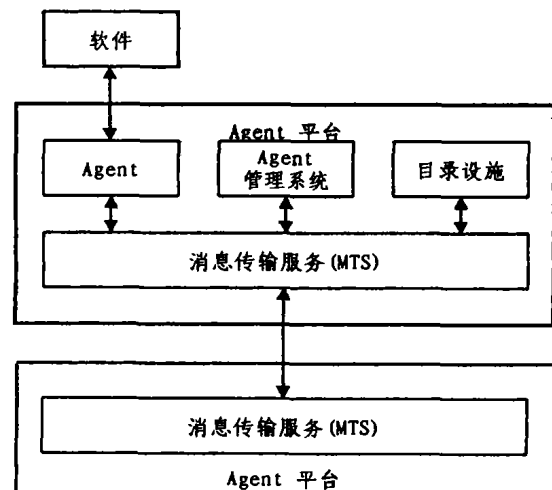


图 3 Agent 管理参考模型

Agent 管理参考模型主要涉及以下几方面:

- 软件 指所有可以通过 Agent 存取的、可执行指令集合。通过获取软件,可以增加新的服务、通讯协议、安全协议、协商协议、迁移支持工具等。

- Agent 管理系统 每个 Agent 平台上只能有一个 Agent 管理系统,负责 Agent 的命名、定位和控制服务。每个 Agent 必须在一个 Agent 管理系统中注册以便得到一个有效、唯一的 Agent 标识(AID)。AID 用一种灵活的、可扩展结构表示,其中包括社会名、传输地址等。Agent 管理系统维护着一个 AID 目录,用于 Agent 管理。

- 目录设施 Agent 可以在目录设施中注册它们的服务或查询其它 Agent 提供的服务。在一个 Agent 平台上可能有多个目录设施,以联邦的形式运作。

- 消息传输服务(MTS) 是默认的跨平台 Agent 消息传输机制[FIPA00067]。在进行消息传输时,MTS 会与 Agent

管理系统和目录设施进行交互,以便获取有关信息。

• **Agent 平台** Agent 平台由硬件、操作系统、Agent 支撑软件、FIPA Agent 管理部件 (Agent 管理系统、消息传输服务 MTS 和目录设施) 以及 Agent 构成。

通过消息缓冲、重定向和代理服务, FIPA Agent 管理规范与 FIPA Agent 消息传输规范对膝上型电脑和个人数字助理也能提供支持。

2.4 Agent 通讯

在 FIPA 中, Agent 之间的通讯是基于一种语义通讯模型, 这种模型具有丰富的预定义语义, 并且易于被 Agent 理解。FIPA Agent 之间的通讯基础是通讯动作 (communicative acts), 也称为述行语 (performatives), 是一种基于言语动作理论 (speech act theory) 的通讯技术。通讯动作是一组动词 (或动词短语), 这些动词会告诉接收到消息的 Agent, 应该在何种上下文中解释消息的内容。目前, FIPA 共定义了 22 个通讯动作, 包括 Call for Proposal、Propose、Accept Proposal、Inform、Confirm、Request、Agree、Refuse、Propagate、Query If、Cancel、Not Understood 等 [FIPA00037]。每个通讯动作都从以下五个方面予以说明: 摘要、消息内容、描述、形式模型、示例。例如, 通讯动作 Query If 的有关说明如下:

• 摘要

用 Query If 可以给其它 Agent 发送一个命题, 询问该命题是否为真。

• 消息内容

一个命题。

• 描述:

发送 Agent 不知道该命题的真值, 并且相信另一个 Agent 会把该命题的真值告诉它, 则可利用 Query If 要求接收 Agent 将命题的真值告诉它。

• 形式模型

```
(i, query-if (j, φ)) =
  (i, request (j, (j, inform-if (i, φ))))
  FP:  $\neg \text{Bifip} \wedge \neg \text{Uifip} \wedge \neg \text{Bi Ij Done}((j, \text{inform-if} (i, \phi)))$ 
  RE: Done ((j, inform(i, φ)) | (j, inform(i,  $\neg \phi$ )))
```

• 示例

Agent i 询问 Agent j 是否已经在领域服务器 d1 上注册:

```
(query-if
  :sender
    (Agent-identifier: name i)
  :receiver
    (set (Agent-identifier: name j))
  :content
    ((registered (server d1) (Agent j)))
  :reply-with r09
)
```

Agent j 回答没有在领域服务器 d1 上注册:

```
(inform
  :sender
    (Agent-identifier: name j)
  :receiver
    (set (Agent-identifier: name i))
  :content
    ((not (registered (server d1) (Agent j))))
  :in-reply-to r09
)
```

通讯动作是 Agent 通讯语言 ACL 的重要基础。FIPA ACL [FIPA00061] 是在 KQML^[3] 的基础上发展而来的。由 FIPA ACL 表达的消息包括通讯动作、发送者、接收者、消息内容、交互协议等。其中消息内容由内容语言表达。目前 FIPA 内容语言库中有 FIPA 语义语言 [FIPA00008]、FIPA 约束选择语言 [FIPA00009] 等四种内容语言。

另外, 还有一组处理 Agent 之间连续会话过程的交互协

议 [FIPA00025-FIPA00036], 其中每个协议对应一种典型会话模式。当前的交互协议库中共有 11 种交互协议, 包括: 请求交互协议、查询交互协议、合同网交互协议、中介交互协议、提议交互协议等。

2.5 基于 Agent 的应用

FIPA 提出了几个与 Agent 应用有关的规范, 包括移动应用支持规范、Agent 软件集成规范、个人旅行助理规范、视听娱乐与广播规范、网络管理与供应规范、个人助理规范、消息缓冲服务规范等。其中 Agent 软件集成规范主要解决非 Agent 软件与 Agent 系统的集成问题; 消息缓冲服务规范主要用于无线连接等间断性连接设备上的 Agent 平台, 当无法与一个 Agent 平台建立连接时, 可以向消息缓冲服务 (MBS) 申请消息缓冲; 其它几个规范讨论了典型的 Agent 应用。

3. FIPA 规范的具体实现

目前已经实现了许多符合 FIPA 规范的 Agent 平台, 下面是十个开放源代码的 Agent 平台^[4]:

- Agent Development Kit
- April Agent Platform
- Comtec Agent Platform
- FIPA-OS
- Grasshopper
- JACK Intelligent Agents
- JADE
- JAS (Java Agent Services API)
- LEAP
- ZEUS

这些 Agent 平台可以自由下载并进行修改, 其中 FIPA-OS 是第一个开放源代码的 Agent 平台, 已经被下载数千次, 世界各地的开发者对它进行了大量的改进和多次升级, 已经发布了十多个正式版本。在这些 Agent 平台上, 已经和正在开发许多大型 Agent 应用项目^[5,6]。

结束语 FIPA 为异质的 Agent 和 Agent 系统之间互操作制订了相关的软件规范。所有 FIPA 规范都处在一个生命周期中, 最后或者成为 FIPA 标准, 或者被废止。我们简要分析介绍了最新 FIPA 规范的五大部分, 以及若干符合 FIPA 规范的 Agent 平台。

最后, 我们对所有为 FIPA 规范做出贡献的人们表示衷心的感谢。

参 考 文 献

- 1 The Foundation for Intelligent Physical Agents. <http://www.fipa.org/>
- 2 OMG Unified Modelling Language Version 1.1. Object Management Group, 1999. <http://www.omg.org/uml/>
- 3 Finin T, Labrou Y, Mayfield J. KQML as an Agent Communication Language. In: Software Agents, Bradshaw, J (Ed), MIT Press, 1997
- 4 Publicly available agent platforms. <http://www.fipa.org/resources/livesystems.html>
- 5 Burg B. Foundation for Intelligent Physical Agents. Official FIPA presentation, Lausanne, 2002. <http://www.fipa.org/docs/output/f-out-00111/>
- 6 Burg B. Towards the Deployment of an Open Agent World. 2001. <http://www.fipa.org/docs/input/f-in-00033/>