

# 一种新的 E-mail 应用模式的研究<sup>\*</sup>

A New Model of E-mail Application

马 勤 陶先平 吕 建

(计算机软件新技术国家重点实验室 南京大学计算机软件研究所 南京210093)

**Abstract** E-mail has been the most mature and widely used way of mail exchanging on Internet. But its ability is far more than this. Virtually, it can be used as a basic facility for data exchanging and automatic processing in various domain applications. From this point of view, a new model of E-mail application is presented in this paper, and based on the new model, we advise a two-layer framework for implementation. In the framework a middleware system EAMS is designed and implemented with well generality, flexibility and expandability. Also meeting notification is discussed as a typical application.

**Keywords** EAMS, EDL, SADF, Agent

## 1 引言

在 Internet 飞速发展的带动下,电子邮件 E-mail 成为了网络环境中异步通讯的基本手段之一,被应用在现实工作、生活中的方方面面,比如:传递消息、发送通知、投递文章、发表新闻、传输文件和共享资源等等。广泛的使用使得邮件量剧增,邮件处理成了相当繁琐的问题。为了减轻人们处理邮件的负担,一些网络软件产品如 Netscape Messenger, Outlook 和 Foxmail 等被开发出来以辅助人们对 E-mail 的处理,但是这些软件产品大多仅提供了部分简单通用的 E-mail 自动处理功能,而对特定领域的特殊请求却缺乏有效的支持,因此都不是那么理想。

分析其中的原因可以看到,通过这些软件使用 E-mail 仍然是一种基于较低的层次的方式,比如用 Netscape Messenger 来发送会议通知,我们目前的做法是:组织一封新邮件,在邮件体中写上通知的内容,逐个写好被通知者的地址,点击“发送”后将会议通知发出;被通知者显式要求 Netscape Messenger 接收邮件,得到邮件后,查看内容发现是一封会议通知,作出决定后再显式地组织一封回复邮件,由 Netscape Messenger 发回给通知发送者。显然,这种直接使用 E-mail,就事论事的方式存在着很多缺陷,最突出的就是 E-mail 被动等待, E-mail 内容需人工理解以及接收到 E-mail 后的处理工作都要由人工完成等。因此,为

了充分发挥 E-mail 的应用潜力,使它不仅能用得广泛,更能用得漂亮,我们提出了一种新的 E-mail 应用模式。

我们认为,在未来的应用中, E-mail 的使用已不再因为它是一种通信手段而仅仅用来作为一种通信手段,它将作为一种通讯的基础设施被上层更完善、自然、智能的应用所屏蔽。换句话说,我们将不再直接使用 E-mail 来实现交互,而是针对不同的应用领域都可以实现一个基于 E-mail 传递和处理的应用系统,从而以一种更高层次的、更自然的且比较标准化的方式来达到目的。各个领域应用系统最终的底层实现仍然是通过 E-mail,只是,所有 E-mail 的收发以及相关处理等工作都由应用系统包揽了,使得用户的概念中根本就不知道 E-mail 的存在,所有 E-mail 的发送、传输和接收的过程也因此都成为透明的了。

目前也有一些面向特定应用领域的专用系统已经被开发出来了,比如 Dee Remaklus Realty 公司开发的符合个人需求的 Buyer's Agent 和 Emmett Travel 公司开发的用于旅游的 E-mail Agent<sup>[4]</sup>等。然而,分析这些软件后我们可以看出,这些软件只对某一特定领域内的某类邮件作专门处理,从而使邮件的处理缺乏通用性和灵活性,同时,也不易扩充其邮件处理能力,不支持邮件处理功能的即插即用。

我们的工作兼顾了通用性和领域针对性两方面的要求,为新的应用模式设计了一种切实可行的实现框

<sup>\*</sup> 本文受江苏省应用基础研究科技项目(BJ99016), 836高科技项目(836-306-ZT02-01-4)和国家杰出青年基金项目(61525204)资助。马 勤 硕士生,研究领域:对象技术,移动 agent 技术,陶先平 讲师,博士生,研究领域:对象技术,移动 agent 技术,吕建 博士,教授,博导,研究领域:形式化方法,对象技术,分布计算技术和构件技术。

架,这个实现框架主要包括两部分:其一是一个通用的中间件技术支持平台,其二是构作在中间件平台之上的针对各领域的典型应用。整个框架具有以下的结构:

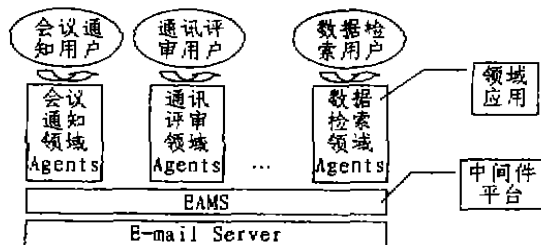


图1 新的 E-mail 应用模式实现框架图

中间件平台 EAMS (E-mail Agent Middleware System) 实现了各应用领域在系统中统一的基础,一方面,它抽取了领域通用的诸如邮件收发、邮件的格式化分拣等功能进行了统一的实现;另一方面,为了实现领域邮件处理的统一,又提供了一种具有良好扩充性和开放性的邮件格式描述及通信协议定义的机制,支持任何领域邮件处理的即插即用。这样,针对各领域的应用开发仅需简单的定义该领域邮件的特定格式和协议,利用 EAMS 的即插即用就可方便地实现领域处理功能到系统的插接,因此,这种实现框架不仅大大地简化了领域程序员的工作,使得他们只需关心领域邮件的处理逻辑而无需考虑各种通用处理的实现,同时也降低了各领域应用维护工作的复杂度。

本文的工作主要包括两个方面。首先,设计并实现了一个中间件平台 EAMS。在 EAMS 中引入了邮件格式化的思想,设计了用于格式化书写邮件正文的邮件描述语言 EDL (E-mail Describe Language) 语言和用于定义通信协议的 SADF (Sample Action Define File) 文件,并用软件 Agent 在各个领域应用中模拟人的行为,代表用户完成对邮件的处理。其次,以会议通知为例构作了一个典型应用,以此介绍 EAMS 针对领域的定制以及领域 Agent 的实现,从而从会议通知这个视角看到了新的 E-mail 应用模式的一个侧面。下面我们首先来介绍一下 EAMS。

## 2 E-mail Agent 中间件平台(EAMS)的设计

EAMS 作为一个领域 E-mail 传输和处理应用的中间件平台,应该具备最基本的邮件收发功能。同时,为了实现各领域的 E-mail 自动处理,还应该提供一套统一的邮件处理机制。在研究的过程中我们认识到:计算机毕竟远没有人的智力,它所能处理的只能是有格式、结构清晰、可以分而治之的信息。要想让它了解信息的含义,唯一的方法就是将信息结构化地表示出来。E-mail 作为 Internet 上的一种广泛使用的信息载

体也不例外,只有人为地给 E-mail 规定一种格式,让它所要表达的信息格式化地体现出来,才有可能将字符串形式的邮件转换为结构化的数据结构交给计算机自动地去处理。

在这个思想的指导下,我们设计了一个格式化书写 E-mail 的语言——EDL 语言。我们知道 E-mail 本身划分为几个部分,已具有了一定的格式,但是其中用自然语言书写的正文部分却是复杂多变,而用 EDL 来书写邮件的正文,就可以实现邮件正文的格式化。

鉴于目前 Web 技术的发展与推广,EDL 语言模仿 HTML 语言的形式,并在其上适当地作了一些调整。一般的邮件体分为两个部分:邮件正文和附件。EDL 又将邮件正文分为两个部分,一般邮件文本部分和领域约定部分,语言的扩展 BNF 描述形式及其文法结构如下:

```
<邮件正文> ::= [<邮件文本>] [<领域约定>];
<邮件正文> ::= <字符> [<字符>];
<领域约定> ::= "[["<领域标记>"]"]" [<领域标记内容>];
```

EDL 作为一种具有良好可扩充性的框架语言,可以适用于任何领域处理要求信息的描述,这一点是通过领域标记和领域标记内容的可扩充能力体现的。针对各个领域和应用需求,都有一些特定的领域标记和领域标记内容,而且,随着新领域的出现或者是新应用需求的提出,还可以向 EDL 中添加定义一些领域标记和领域标记内容,因此 EDL 是一个开放的、不断变化中的语言。

从某种意义上来说,在这里定义的 EDL 并不是一种具体的语言,而只是一种语言的理念,同时给定了最基本的语法。对于各个领域都可以定制各自具体的 EDL。

各个领域的 EDL 都可以用来书写本领域的格式化的 E-mail,但是,要让通信双方能够正确地分解和理解信息并能够作出恰当的自动处理,则还要定义领域的通信协议,为此,我们又在 EDL 的设计基础上提出了用于表示通信协议的 SADF (Sample Action Define File) 文件。所谓通信协议就是要指明如何用 EDL 提供的语言成份来书写 E-mail 以及某种格式的 E-mail 具体该进行什么样的处理,即实现特定格式邮件和相应处理 Agent 之间的关联。SADF 文件是一种格式化的文件,由若干变量和值的二元对组成,具体定义了由 EDL 书写的 E-mail 各个语法部分的要求以及相关联的自动处理 Agent。它的一般形式为:

```
变量名1 = ...../t/t 变量名2 = ...../t/t ..... 变量名13 = ...../t/t
```

EDL 的设计将传统的字符流形式的邮件正文变成了若干个有标记的部分,大大降低了邮件正文分析的复杂度,同时,EDL 的可扩充性和 SADF 文件的引

入实现了各领域 Agent 在 EAMS 上的即插即用,为系统面向各个领域的扩展奠定了基础。

在 EDL 和 SADF 的设计基础之上,我们开发了一个用于辅助 E-mail 自动处理应用系统的中间件平台系统——EAMS(E-mail Agent Middleware System)。EAMS 提供定制领域 EDL 和 SADF 文件的接口,同时支持领域 Agent 的即插即用,居一个中间件平台的地位,为构造领域专门化处理 Agent 的程序员提供了一系列基于 E-mail 处理的服务和设施。其中提供的服务主要有: E-mail 的发送服务;自动获取特定格式的 E-mail,进行预处理组织成格式化文件服务;针对 E-mail 的特定格式启动相应的处理程序(称为处理 Agent)服务。

提供的设施主要有:领域 EDL 和 SADF 的定制;特定格式 E-mail 处理 Agent 的即插即用。

EAMS 主要分为以下几个部分,包括 E-mail 发送模块,用于领域定制的 EDL 和 SADF 定义模块,以及实现邮件的自动收集、分析、预处理和启动相应处理 Agent 的 E-mail 自动探测模块, E-mail 格式匹配模块和 E-mail 处理调度模块。

下面是 EAMS 的功能结构图,可以从中看出 EAMS 的全貌:

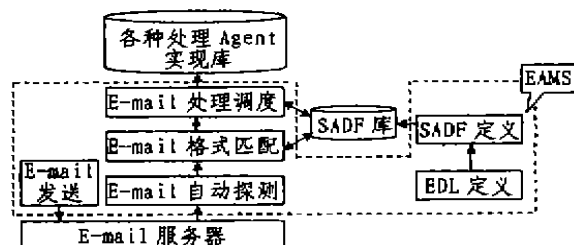


图2 EAMS 功能结构图

这样,只要定义了相关领域的 EDL 和通信协议 SADF,实现了相应的处理 Agent,就可以利用这些服务和设施将 EAMS 作相应领域的定制,完成 Agent 的插接,从而实现这个领域的处理要求。这一方面可以被看作是 E-mail 处理系统的一部分,另一方面也可以和 EAMS 一起被单独出来作为领域专门的应用系统。下面,就以会议通知为例,介绍会议通知领域 EDL 和 SADF 的定制以及处理 Agent 的编制,从中了解如何向会议通知领域扩充 E-mail 处理系统的功能以及新的应用模式在会议通知过程中的外观表现。

### 3 会议通知系统的设计

根据以上的分析和介绍可以知道,在 EAMS 之上实现会议通知领域的功能将主要包括三方面的工作,即领域 EDL 的定义,通信协议的设计和相应处理 A-

gent 的设计。

#### 3.1 领域 EDL 的定义

针对会议通知领域的 EDL 扩充了以下领域标记和领域标记内容,扩充部分的 BNF 如下:

〈领域约定〉::=“[[“M〈领域标记〉”]]”{〈领域标记内容〉}

〈领域标记〉::=Meeting|Feedback|ID|Name|Org|Date|Place|Content

〈领域标记内容〉::=〈字符〉{〈字符〉}

#### 3.2 通信协议的设计

会议通知系统的内部通信协议主要包括两大类邮件:会议通知邮件和回复信息邮件的格式定义和与处理 Agent 的关联,这一节只介绍邮件的格式的设计,与处理 Agent 的关联将在 3.3 节处理 Agent 的设计中给出。

3.2.1 会议通知邮件 具体又分为首发会议通知邮件和重发会议通知邮件两种,特定的邮件格式分别设计如下:

·首发会议通知邮件格式:

```
邮件主题=Meeting1
邮件正文=[[Meeting]]
          [[ID]]〈会议 ID 号〉
          [[Name]]〈会议名称〉
          [[Org]]〈会议组织者〉
          [[Date]]〈会议日期〉
          [[Place]]〈会议地点〉
          [[Content]]〈会议内容〉
```

·重发会议通知邮件格式:

```
邮件主题=Meeting2
邮件正文=[[Meeting]]
          [[ID]]〈会议 ID 号〉
          [[Name]]〈会议名称〉
          [[Org]]〈会议组织者〉
          [[Date]]〈会议日期〉
          [[Place]]〈会议地点〉
          [[Content]]〈会议内容〉
```

3.2.2 回复信息邮件 也将具体区分是确认参加还是要求请假这两种情况,格式设计方法同会议通知邮件,在此略去介绍。

#### 3.3 处理 Agent 的设计

针对以上设计的会议通知领域的邮件格式,处理 Agent 主要包括会议通知发送 Agent 和会议通知接收 Agent,我们以会议通知接收 Agent 为例说明。

会议通知接收 Agent 与会议通知邮件相关联,它的功能包括会议通知的主动接收,报告会议通知到达及回复信息的返回。Agent 在接收到来自某个会议组织者的首发会议通知邮件后,报告被通知者有一个会议通知到达并展示会议的相关内容,被通知者可以选择马上给出回答或者暂时不作决定,一旦被通知者作出决定,Agent 将根据具体的回复决定(参加或是请

假)组织相应的回复邮件并发回给会议组织者。如果接收到的是重发的会议通知邮件,Agent 将提示被通知者已经接收到了来自会议组织者的催促,要求其作出回复。

下面是会议通知系统运作过程的框图,从中可以很清楚地了解到各种邮件格式和其关联处理 Agent 是如何交织来完成整个工作的。

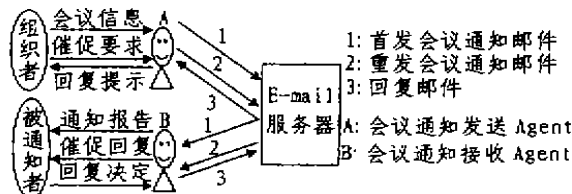


图3 会议通知系统工作框图

### 3.4 管理和智能提醒功能

新的应用模式改进了用 E-mail 来发送会议通知的方式,但是繁多的会议通知信息和回复信息仍需要有效的辅助。因此,针对会议通知这个领域的应用还应该具备对繁多复杂的会议通知进行有效的管理和智能化的提醒的功能。

管理的功能主要体现在对所有已收或已发会议通知的系统记录,有序列举和各种信息的随时待查,这些信息可以包括已收会议通知的相关信息,已做的决定,已发会议通知的相关信息和各位被通知者的回复情况等等。

当收到的会议通知或是组织的会议比较多的时候,就可能会出现遗忘的现象,智能提醒功能针对这一问题,对每个会议都设定了若干个提醒点,一旦到达某个提醒点就会主动提醒用户此会议即将召开,同时展示与本会议相关的各种信息。智能提醒一方面可以提醒组织者和决定参加者准时到会,另一方面又可以督促组织者查询回复情况以及时作出催促的决定。

这样看来,这不仅仅提供了一种更有效、更自然的通过 E-mail 发送会议通知的手段,从某种意义上来说它更像是一个助手,能辅助你完成诸多与会议通知有关的事项,并且做得更好。

**结束语** 本文针对目前 E-mail 的应用现状及 E-mail 本身的应用潜力提出了一种新的 E-mail 应用模式的设想。实现了一个中间件平台 EAMS,并设计了一个邮件格式化语言 EDL。EAMS 是一个为了能够更有效地实现基于 E-mail 处理的应用系统而设计的中间件平台系统,我们在其上以会议通知领域为典型应用说明了新的应用模式。

利用新的应用模式下会议通知领域的功能以 E-mail 为通信途径发送会议通知,不仅克服了现有使用 E-mail 过程中的种种弊端,提供了一种更有效、自然的通信途径,还具有系统化的管理会议信息和智能化的提醒功能。

会议通知领域只是各种各样领域中的一个,文中举例的目的是为了抛砖引玉,我们在其它很多的领域中诸如通讯评审、数据库离线查询、高性能计算资源共享等等都可以构建各自的领域应用,从而以新的应用模式来更好地使用 E-mail 这种通讯手段。

### 参考文献

- 1 Standard for Format of ARPA Internet Text Messages, Ceoacker, RFC822, 1982
- 2 Postel J B. Simple Mail Transfer Protocol, RFC821, 1982
- 3 Rose M Post Office Protocol-Version 3, RFC1460, 1993
- 4 Genesereth M R, Ketchpel S P. Software Agents. Comm. ACM, 1994, 37(7): 48~53
- 5 Maes P. Agents that Reduce Work and Information Overload. Comm. ACM, 1994, 37(7): 30~40
- 6 Lemay L. Web Publishing with HTML, Sams. net Publishing, 1996
- 7 柳杨. 基于 Internet 的 E-mail Agent 中间件系统. [南京大学硕士论文]. 1999

(上接第35页)

**结论** 基于 CBT 组播路由协议的组播密钥分配方法是一种适合于广域网组播的可选组播密钥分配方法,具有很强的可放缩性,每个经过认证的 CBT 传递树上的路由器都可成为组播密钥分配中心 GKDC。本文对其组播密钥分配模型进行了详细分析,指出了其优缺点并从提高这种方法安全性和可用性的角度提出了两种改进措施,克服了原方法存在的一些缺点,具有一定的实用价值和推广意义。

### 参考文献

- 1 Harney H, Muckenhirn C. Group Key Management Protocol(GKMP) Specification, RFC2093, 1997
- 2 Harney H, Muckenhirn C. Group Key Management Protocol(GKMP)Architecture RFC2094, 1997
- 3 Ballardie A, Cain B, Zhang Z. Core Based Tree(CBT Version3) Multicast Routing Protocol Specification. Internet draft, 1998
- 4 Ballardie A. Scalable Multicast Key Distribution, RFC 1949, 1996
- 5 Fenner W. Internet Group Management Protocol, Version 2 RFC2236, 1997