

文章编号: 1007-5321(2004) 06-0036-05

基于 Parlay API和代理技术的自适应 上下文感知策略平台

翟晓波, 杨放春

(北京邮电大学 网络与交换国家重点实验室, 北京 100876)

摘要: 首先介绍了自适应业务定义和各种生成技术,然后概述了 Parlay API(应用编程接口)中策略管理.重点描述了增加策略代理和策略代理服务器,并用 Parlay API方式管理;提出了移动终端中的环境感知进程的概念;设计了规则的存储,将自适应业务构建在了下一代移动网络中的业务支撑平台上.最后阐明了方案的优势.

关键词: 上下文感知; 自适应; 代理; 策略服务器

中图分类号: TN915.02 **文献标识码:** A

Adaptive Context-Aware Service Policy Platform Based on Parlay API and Agent Techniques

ZHAI Xiao-bo, YANG Fang-chun

(State Key Laboratory of Networking and Switching, Beijing University of Posts and Telecommunications,
Beijing 100876, China)

Abstract The definition and creation techniques of context-aware service are firstly introduces in this paper. Then Parlay policy management service capability feature (SCF) is simply described. The focus are adding policy agent and policy agent server which are managed by Parlay application programming interface (API), bringing up the idea of context-aware process (CAP) which is running in the mobile terminal, and the design of policies store. This construction moves the adaptive context-aware service to the next generation mobile network service support platform. In the end, the schema advantage of performance and architecture are analyzed.

Key words context-aware; adaptive; agent; policy server

收稿日期: 2004-09-02

基金项目: 国家自然科学基金项目 (90104024, 60125101)

作者简介: 翟晓波 (1977-), 女, 博士生. E-mail: zhaxiaobo@263.net

杨放春 (1957-), 男, 教授, 博士生导师. E-mail: fcyang@bupt.edu.cn

1 自适应上下文感知业务概念简介

上下文 (context)是指一个业务运行过程中所处的环境。上下文按种类可以分为通信环境、物理环境、社会环境、位置信息和用户信息,以及用户的客户化信息几大类。上下文感知业务是指一个带有上下文的正在运行的业务,因为周围物理环境和时间的变化而导致上下文中的参数值发生变化,使得业务对于周围的变化能够感知。上下文感知业务通常根据上下文的改变,依据预先设定好的规则做出某些响应动作,这种满足自适应行为的上下文感知业务通常也叫做自适应业务。自适应业务是上下文感知业务的继承。

2 自适应上下文感知业务生成技术

自适应上下文感知业务主要涉及到以下几种技术:① 位置服务器。设立位置服务器将所有设备对象都进行注册登记,将所有的设备对象变成位置对象。② 代理技术^[1]。采用代理技术构建用户代理、设备代理、资源代理、业务代理。在移动网络中,用户代理、设备代理和业务代理都需要迁移从而改变位置,因此是移动代理。③ 代理服务器。自适应业务系统中设置设备代理服务器、用户代理服务器以及业务代理服务器备份当前小区中所有活动用户和活动设备以及业务的信息,代理服务器中的信息与代理中的信息是同步的。④ 可机器解析的代理语义。文献 [2]中为每一个代理设计了一个主动上下文记忆 (ACM),每个 ACM中包含一个知识库来存放多个移动业务知识 (MSK);并且配备了概念生成器、完整性检查、垃圾收集、评估工具、推论引擎、查询接口等辅助管理工具来管理 MSK。代理之间的通信采用扩展业务协议层 (XSP)来实现。⑤ 传感器。传感器将环境变化的信息收集上来,分析汲取有用信息,传递到用户代理或终端设备代理,再传递到网关,通知应用服务器。有些信息的收集是不需要传感器的,比如时间。⑥ 策略服务器 (PS)。自适应业务如果能够根据上下文的改变而自适应的对业务动作做出响应,就需要和规则相联系起来,即 PS的支持。文献 [3]中策略模型采用的是 IETF中的 PCIM(策略核心信息模型)以及扩展标准。

下面将提出以 Parlay API提供的策略管理业务能力特征 (SCF)来管理 PS。

3 基于 Parlay API方式的策略管理

Parlay API的策略模型秉承了 PCIM的思想。关于 Policy Management SCF的定义如下:
对于策略的存储分为域、组、规则和重用知识库。域和组是规则存储的一些分类,比如电子商务域、预付费组。域和组都是可以自嵌套的,域中还可以有域,组中还可以有组。规则又具有现在是否可以使用、规则用法、优先级、是否强制执行、策略角色、条件列表的类型和动作序列几个属性。事件知识库是用来存储预先定义的条件事件和动作事件,这些事件被在域和组中定义的规则所引用。规则的产生可以是用户的自定义信息,也可能是运营商或者业务提供商对以往业务数据采用数据挖掘技术分析出的比较可靠的规律。Policy TimePeriodCondition用来表示一个规则是由一个或多个策略时间周期联系在一起,根据策略规则的激活与非激活指示时间进程的安排,这个规则对指定的时间的“并”或者“或”是有效的,而除此之外的时间是无效的。

4 将策略代理和 Parlay API引入自适应业务的方案研究

4.1 构建策略代理和策略代理服务器

自适应业务系统中由于用户和设备不断移动可能会频繁用到与己相关的策略,因此在具备了 PS 的基础上,策略平台增加了 2 类设备:① 策略代理 (PA). 每个移动终端一个,存储与该设备和用户相关的业务规则. PA 需要跟随设备移动,因此为移动代理. ② 策略代理服务器 (PAS). 每个小区多个,小区中所有 PAS 信息的总和是小区中所有 PA 信息总和的主备份,每个 PAS 上的信息是小区中运行在一部分终端设备上的 PA 信息.

如图 1 所示,PS 是每个小区一个,存储所有归属网络是该小区的用户设备的业务规则. 当

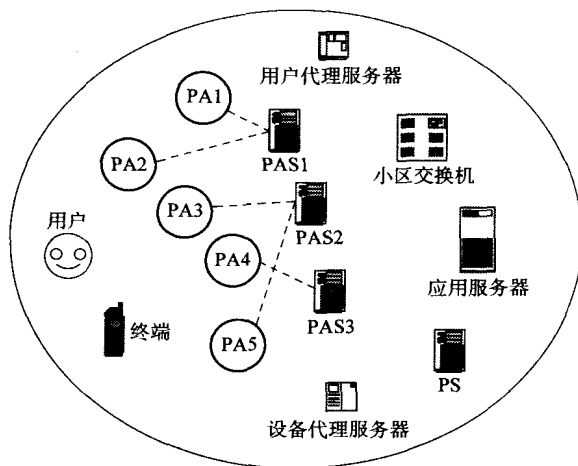


图 1 同小区中的 PA 与 PAS

用户在归属网络中开启设备时,则 PS 把与该用户相关的策略拷贝到 PA 并备份到 PAS. 当用户在外地网络中开启设备时,则外地网络中的服务器会去联系归属网络中服务器,然后找到归属网络中的 PS 的位置,取与该用户相关的策略拷贝到 PA 并备份到该小区的 PAS. 用户随着移动设备迁移到新的小区时,PA 跟随用户设备漫游,在新的小区向新的 PAS 注册,PA 通过注册知道自己备份在哪个 PAS 上. PA 和 PAS 中的信息应当始终保持同步.

PA 是 PS 中与己相关内容的拷贝, PAS 信息又是一个小区中所有 PA 的总

和. 因此最终 PS 的管理方式决定了 PA 和 PAS 的管理方式. 按照第 3 章提出的基于 Parlay API 的策略管理,一个 PS 中包含一个策略库和知识库,一个策略库中包含若干域,知识库中包含若干子库,每个域中定义同种类规则. 策略库和知识库通过 Policy Management API 与应用侧交互.

4.2 移动设备中的环境感知进程

为了控制移动终端的某些能力,在移动终端设备上设计一个驻留进程,叫做环境感知进程 (CAP). 它主要负责接收传感器的信息与 PA 中的规则进行对比,然后对铃声大小、语音大小、背景亮度进行控制. 由于 CAP 所能处理的通用规则与 PA 中的用户自定义规则可能会发生冲突,于是在执行策略之前,CAP 会首先检测 PA 中的事件,条件符合,则上报,由应用服务器去 PAS 中获取相应规则,然后指示软交换执行;如果不符合,再检测移动终端中存储的默认规则,有满足条件的,则执行.

4.3 PS 中业务规则的存储研究

规则在策略服务器中的实际存储方式采用 XML 文件. 分类组中每个用户或设备相应的规则又分别按组存储,同一用户或同一设备的规则存储在一组中. 为了方便事件的分类,将事件增加一个属性来表明属于哪个用户或设备. PA 上的规则只需将与用户和设备相关的组以及事件复制出来. PA 中的规则有 2 个子组——用户组和设备组,仍然要保留所属域名、组名

和事件, PAS中规则仍然按照 PS中的格式存储. 考虑到策略和事件的增、删和改没有策略执行那么频繁,因此只在 PS中进行,重用知识库也只存在于 PS中. PA和 PAS中仅有规则域和事件.

$\langle ? \text{ xml version= " 1.0" encoding= " UTF-8"?} \rangle$	
$\langle \text{Domain} \rangle$	分类域
$\langle \text{DomainName} \rangle \cdots \langle / \text{DomainName} \rangle$	域名
$\langle \text{Property} \rangle \cdots \langle / \text{Property} \rangle$	属性
$\langle \text{Group} \rangle$	分类组
$\langle \text{GroupName} \rangle \cdots \langle / \text{GroupName} \rangle$	组名
$\langle \text{Group} \rangle$	一个用户或一个设备的规则信息
$\langle \text{Name} \rangle \cdots \langle / \text{Name} \rangle$	用户名或设备名
$\langle \text{RuleName} \rangle \cdots \langle / \text{RuleName} \rangle$	规则名
$\langle \text{Condition} \rangle \cdots \langle / \text{Condition} \rangle$	条件 (可多个)
$\langle \text{Event} \rangle \cdots \langle / \text{Event} \rangle$	上报事件
$\langle \text{Action} \rangle \cdots \langle / \text{Action} \rangle$	动作 (可多个)
$\langle \text{Repository} \rangle$	知识库
$\langle / \text{Repository} \rangle$	

5 性能分析

5.1 PA和 PAS数目对系统的影响

策略平台的系统性能与 PA和 PAS的数目有关. 首先,如果 PA越多,代表用户数越多,那么系统占用资源多,业务的平均响应时延也变大,则系统性能与 PA的数目成反比. 其次, PAS的数目分 2种情况来考虑,当 PAS的数目小于一个数值 N 时,那么 PAS的数目与系统性能成正比;当这个数目超过 N 时,过多的 PAS占用了太多的资源,而并没有改善单位业务时延的响应时间,那么 PAS数目与系统性能成反比. 公式表示如下:

$$P = N_u(\text{PAS}) / N_u(\text{PA}), \quad N_u(\text{PAS}) \leq N$$
$$P = 1 / [N_u(\text{PAS}) N_u(\text{PA})], \quad N_u(\text{PAS}) > N$$

多个 PAS接收用户事件响应并上报应用服务器,这是一个多服务窗排队模型 $M/M/n$;后来的用户请求在某个 PAS后排队,系统容量又是一定的,因此为多服务窗混合制排队模型 $M/M/n/m$. 若用户事件按泊松流到达系统,其到达强度为 λ ;有 n 个窗口工作独立,服务时间均为负指数分布,服务强度为 ρ ,假定系统的容量为 m ,即系统最多可容纳 m 个用户的上报事件. 这种模型的平均排队等待队长为

$$L_q = \frac{n^n d^{n-1} p_0}{n!} \cdot \frac{1 - d^{m-n+1}}{1 - d} \cdot \frac{n^n d^{n-1} p_0}{n! (1 - d)^2} [1 - (m - n + 1) d^{n-n} + (m - n) d^{n-n}]$$
$$d = \frac{\lambda}{n}$$

平均用户数为

$$L_s = L_q + n - \sum_{k=0}^{n-1} (n - k) p_k$$

因为 n 个服务窗只有一个排队等候的队列的 $M/M/n$ 排队系统要比 n 个 $M/M/1$ 排队系统有显著的优点^[4],因此上面每个 PA随机注册到某个 PAS的方案要加以改进,让每个 PA在

n 个 PAS 上都有注册. 用户通过上报事件时, 可以随机选择某一个 PAS, 效率会有显著的提高.

5.2 加入策略代理和原系统的对比

加入策略代理和原系统的策略响应时延分别由以下方案的 2 个公式表示.

方案 1 在每个小区设置一个 PS, 然后在小区中增加若干 PAS 存放当前小区所有注册终端的策略. 为每一个移动终端增加一个“随身携带”的策略代理.

$$T_1 = t_1 + I_1 + t'_1$$

式中, t_1 为移动终端将事件上报到 PAS 的时间; I_1 为本地应用服务器处理事件, 并将规则取出的时间; t'_1 为将动作执行到执行终端的时间.

方案 2 在每个小区设置一个 PS, 存放所有归属小区为该小区的移动终端的策略.

$$T_2 = t_2 + I_2 + t'_2$$

式中, t_2 为移动终端将事件上报到 PS 的时间; I_2 为本地应用服务器处理事件, 并将规则取出的时间; t'_2 为将动作传递到执行终端的时间.

由于电磁波的速度非常快, 忽略传输时间, 在传输过程中影响时延的参数仅为经过交换机时的排队时延, 所以时延与经过交换机的次数成正比. 我们把经过一次交换机叫做“一跳”. t_1 , t_2 , t'_1 , t'_2 都依赖于源和目的地址之间的跳数. 显然如果移动终端在外地小区中, $t_2 > t_1$. 外地小区离归属小区越远, 则 t_2 越大. 如果执行动作的作用对象是移动终端, 那么 $t'_2 > t'_1$. 同理, 终端离归属小区越远, 那么 t'_2 越大. 如果终端在归属小区内, $T_1 \geq T_2$. 之所以有大于存在, 是因为还有构建策略代理以及备份 PAS 的时间. 但如果在外地小区, 那么 $T_1 < T_2$. 所以外地小区频繁使用策略时采用方案 1, 而方案 2 适用于用户在归属地.

6 结束语

下一代网络从下至上分为接入层、承载层、网络层和业务层, Parlay API 开放网络结构与下一代网络分层的思想是一致的, 它的目的是开放网络能力, 屏蔽网络细节, 达到平台一致性和业务可移植性. 将 PAS 采用 Parlay API 方式来管理, 符合网络分层的思想.

如果每个小区中只有 PS, 那么漫游到外地网络中的用户和终端还要到归属网络中查找与自己相关的规则, 这样的方案过于依赖网络的连接性, 并且服务器式的工作模式会形成处理的瓶颈. 增加 PA 和 PAS 将策略分布开来, 使得业务的自适应更灵活; 同时减轻某一设备的压力, 也增加了整个网络的可靠性和安全性.

参考文献:

- [1] 胡晓娟, 苏森, 郭乐深, 等. 下一代网络中基于代理技术的 VHE 业务平台 [J]. 北京邮电大学学报, 2004, 27(2): 93-97.
Hu X J, Su S, Guo L S, et al. An agent-based VHE system in NGN [J]. Journal of Beijing University of Posts and Telecommunications, 2004, 27(2): 93-97.
- [2] Theo G K. Adaptive personal mobile communication [D]. Sweden: Department of Microelectronics and Information Technology, Royal Institute of Technology Stockholm, 2001.
- [3] Kun Y, Galis A, Serrat J, et al. Network-centric context-aware service over integrated WLAN and GPRS [A]. 14th IEEE PIMRC 2003 [C]. Beijing: Electronic Industry Press, 2003: 854-858.
- [4] 陆传贵. 排队论 [M]. 北京: 北京邮电大学出版社, 1994: 94-105.
Lu C L. Queuing theory [M]. Beijing: Beijing University of Posts and Telecommunications Press, 1994: 94-105.