

## 前 言

本标准与应用开发者和应用协议开发者提供正确的和有效的使用 T. 120 基础结构的指南, 它为使用 T. 120 服务进行通信的应用提供一个通用的模型, 并定义了 T. 122(MCS)和 T. 124(GCC)服务使用的通用应用模板。使用 T. 120 服务的各应用协议均需要 T. 122(MCS)和 T. 124(GCC)服务。本标准与应用协议实体提供了一个公共的结构, 以确保不同的应用协议实体能在同一会议中共存。不遵循本标准定义的操作的应用协议实体将冒有与其他应用协议相冲突的危险。

本标准等效采用 ITU-T 建议 T. 121(1996 年版)。

本标准由邮电部电信科学研究规划院提出并归口。

本标准起草单位: 邮电部电信传输研究所。

本标准主要起草人: 聂秀英、李守静。

# 中华人民共和国通信行业标准

## 多媒体会议业务的通用应用模板

YD/T 948 1998  
eqv ITU-T T. 121:1996

Generic application temple for multimedia conferencing service

### 1 范围

本标准规定 T. 120 应用的通用模型,以及为实现预期应用行为开发 T. 120 基础结构的指南。为定义公共框架(通用应用模板),本标准提供了 T. 120 应用的通用模型,以形成标准化应用协议的基础。

本标准适用于使用 T. 120 基础结构的基于终端的应用协议。

### 2 引用标准

下列建议和其他参考资料所包含的条文,通过在本标准中引用而构成为本标准的条文。本标准出版时,所示版本均为有效。所有的建议和其他参考资料都会被修订,使用本标准的各方应探讨使用下列建议和其他参考资料最新版本的可能性。

ITU-T 建议 T. 120(1996) 多媒体会议的数据协议

ITU-T 建议 T. 122(1993) 用于音像和视听会议业务的多点通信服务

ITU-T 建议 T. 124(1995) 通用会议控制

ITU-T 建议 T. 125(1994) 多点通信服务协议规范

ITU-T 建议 T. 50(1992) 国际参考字符——7 位编码字符集

### 3 定义

本标准采用下列定义。

3.1 应用资源管理器:提供用于管理 GCC 和 MCS 资源通用功能(性)的应用协议实体部分。

3.2 应用服务元素:提供应用协议特定功能(性)的应用协议实体部分,例如:消息格式和顺序。

3.3 非标准基础会话:无需用户干预,可通过非标准的 APE 确定的应用协议会话。它类似于标准的 APE 使用的标准基础会话。

3.4 专用会话:由会话创建者控制的限制成员资格的应用协议会话。

3.5 公开会话:不限制成员资格的应用协议会话,典型情况是在与协议相关的基础会话已经处于使用状态且用户期望建立新的独立会话时使用。

3.6 登记会话:为通知列席会议者出席了会议,由应用使用的应用协议会话。登记会话使用 T. 124 缺省会话,在该缺省会话中没有会话 ID。

3.7 会话创建者:在新应用协议会话中注册的应用协议实体。会话创建者的概念仅适用于公开和专用会话。

3.8 会话成员:在已经存在的应用协议会话中注册的应用协议实体。会话成员的概念仅适用于公开和专用会话。

3.9 标准基础会话:无需用户干预,可通过标准的 APE 确定的应用协议会话。T. 120 中规定了标准基础会话标识符。

### 4 缩略语

本标准使用下列缩略语。

中华人民共和国邮电部 1998-02-05 批准

1998-05-01 实施

ARM	应用资源管理器
APE	应用协议实体
ASE	应用服务元素
GAT	通用应用模板
GCC	通用会议控制
GCC SAP	GCC 服务接入点
MCS	多点通信服务
MCSAP	MCS 服务接入点
MCU	多点控制单元
PDU	协议数据单元
SAP	服务接入点

5 概述

图 1 示出了 T. 120 系统模型。该模型中,用户应用为在同一会议中与其他节点上它们的对等实体进行通信,使用标准的和非标准的应用协议。用户应用的范围限制在不影响对等用户应用之间通信的本地功能(例如,用户接口)。每一用户应用使用一个或多个应用协议实体(APE),这些应用协议实体采用图 2 所示的互通协议。在此处描述的模型中,APE 可以被进一步划分成两个元素:应用资源管理器(ARM)和应用服务元素(ASE)。ASE 提供特定功能(性)互通,ARM 负责 GCC 和 MCS 资源的通用管理,例如,ASE 可以提供 T. 127 规定的通用文件传送功能(性)或 T. 126 规定的静止图像互换功能(性)。

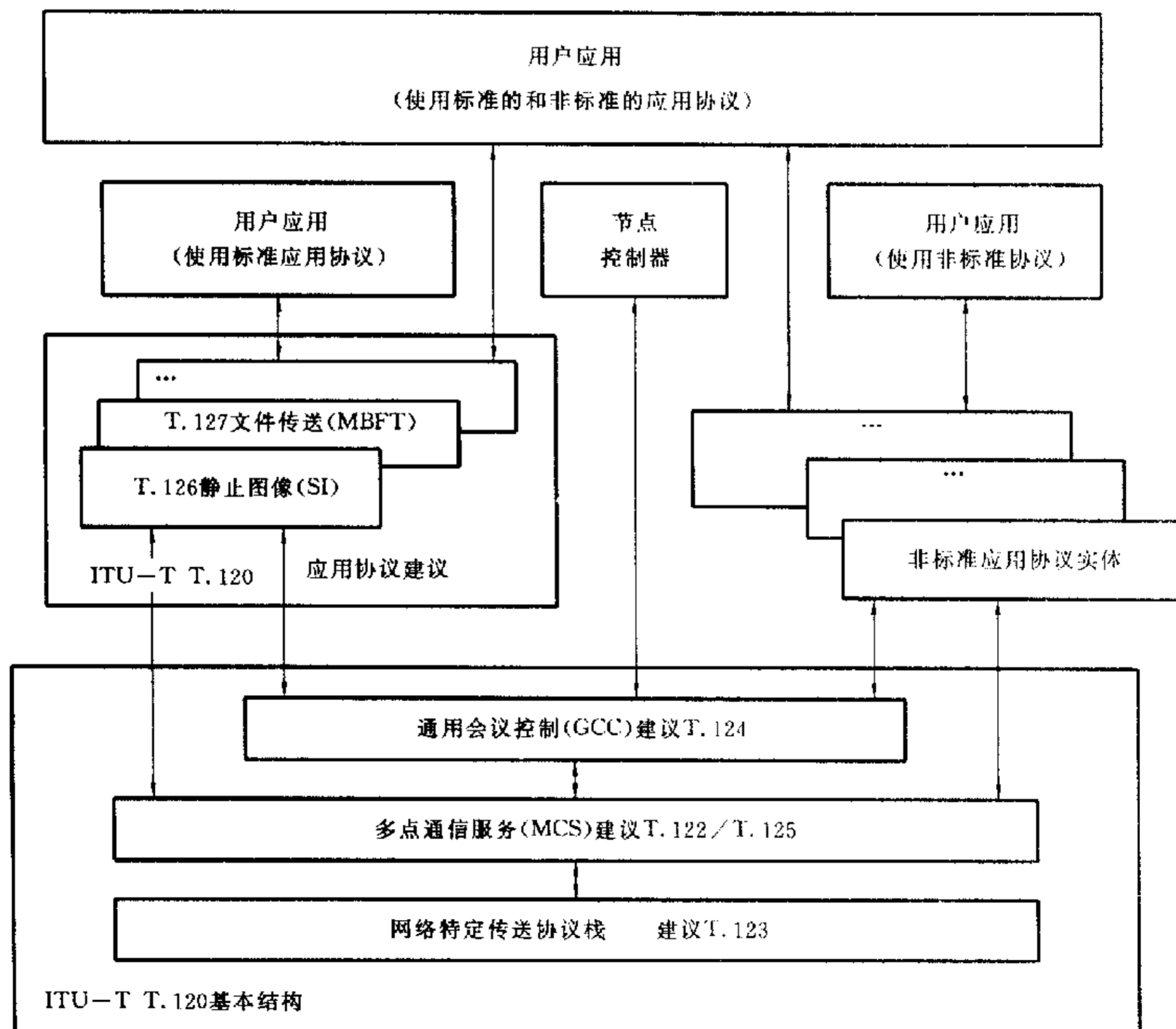


图 1 T.120 系统模型

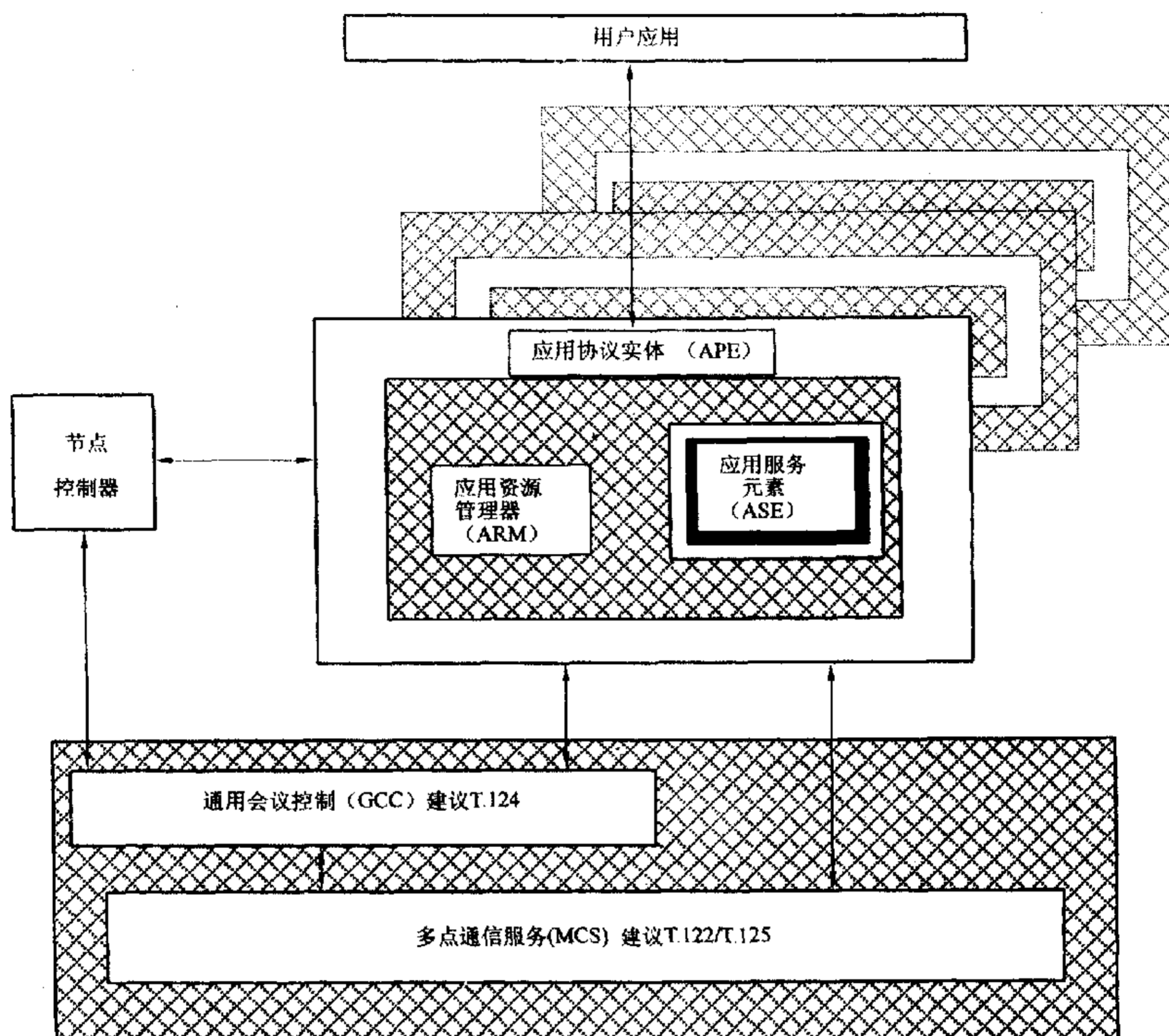


图 2 通用的 T. 120 应用协议实体的模型

将彼此进行通信的对等应用协议实体称为加入了同一个应用协议“会话”。注意，在某个给定站点可能有不止一个用户应用加入同一会话。这就需要应用协议的多种实例，即多个 APE，每一用户应用有一个 APE。也可能有使用同一应用协议的并发的会话。再者，每一加入的用户应用需要它自身的 APE。图 3 示出了一个例子。

一个会话由一组使用公共应用协议彼此通信的对等应用协议实体组成。每个会话有一会话标识符，该标识符的值等于 MCS 信道 ID 的值，该信道 ID 为用于在一组 APE 中传送数据的特有的 ID。在此处所示的模型中，与会话标识符相关联的信道为加入的第一条信道，称其为会话标识符信道（即使并不要求这种情况）。使用多个 APE 的用户应用可以加入多个应用协议会话。每一会话可能有唯一的一个会话标识符，但并不要求这样。即使它们具有相同的会话 ID，不同类型的协议实体加入不同的会话。图 4 中示出了一个例子。对每一应用协议，在没有会话标识符的情况下，可能有单一的应用协议会话，即登记会话。

有下列会话类型。

**登记会话**：可被标准和非标准协议使用。登记会话的目的是使应用在不必要变为激活的情况下，能通知它们出席了会议。这就使得各种应用能够确定在会议中的哪些节点有类似的功能（性），而同时消耗最少的资源。登记会话也允许应用将其应用协议的特定功能（性）作为不可拆装的能力通知会议。在会议期间为接收由其他节点通知的在功能（性）上的任何变化，这些应用在登记会话中基本上要保持已登记状态。在使用特定应用协议的登记会话中，注册的某个应用在会议中使用那个特定应用协议接收有关的所有会话信息。

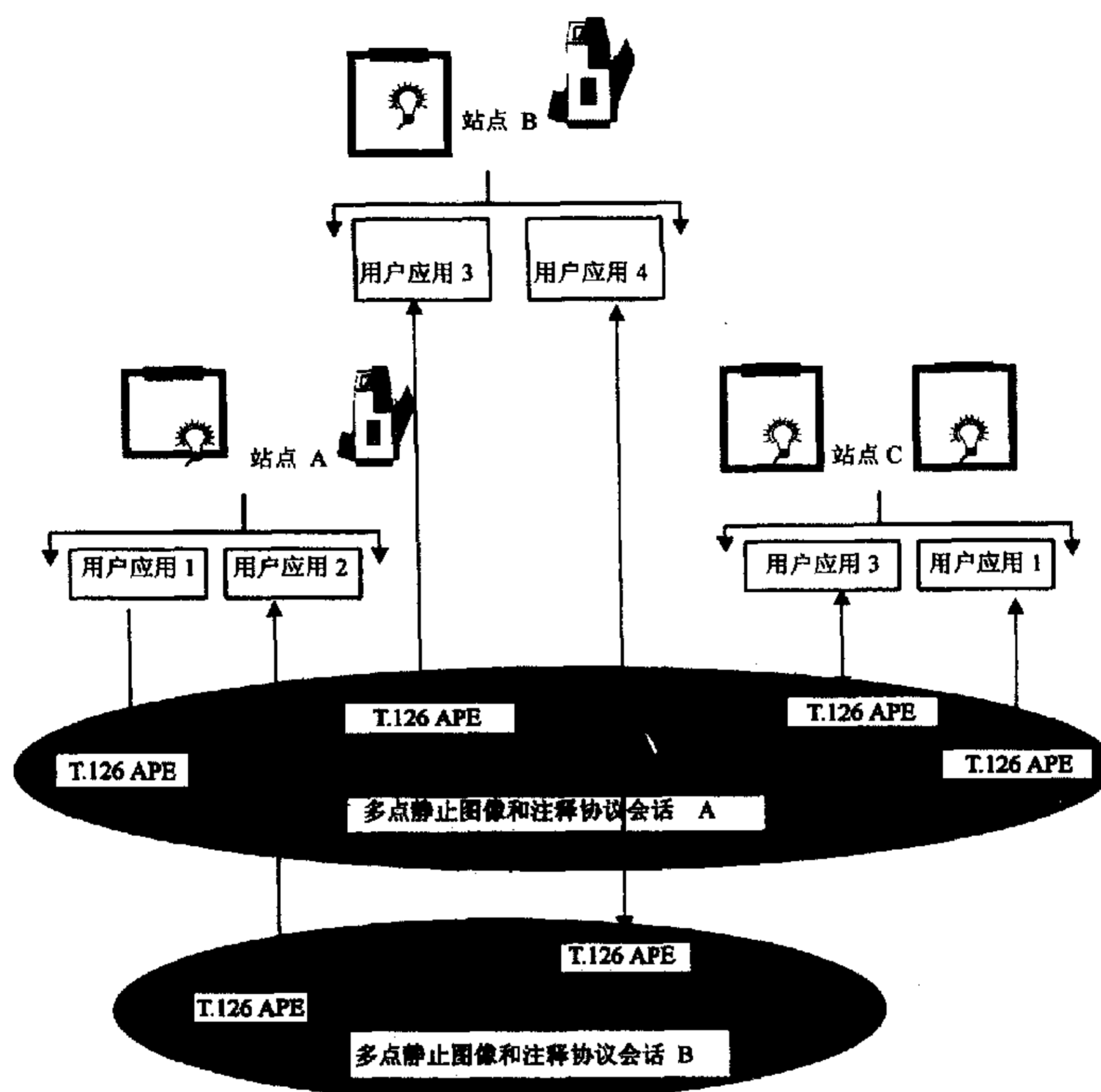


图 3 在一个站点使用同一个协议的多个应用的例子

在实际应用活动中使用的登记会话不是这些模型的一部分,也不允许这样用。这就确保了激活和潜在应用的清楚分界。这种分界是重要的,这是因为以下原因。

- a) 由非激活应用所通知的任何可拆装的能力可能对该会话能力集具有不利影响;
- b) 在由应用活动所产生的登记会话中不再存在的登记处登录项可能会存留到会话结束。

对会议中的每一应用协议,至多能有一个登记会话。利用具有适当应用协议键和会话 ID 所组成的会话键来表征每一应用协议。

**标准基础会话:**被标准化的应用协议使用。接入标准基础会话是不受限的,并且该会话中的所有与会者(即会话成员)具有平等的资格;他们可以随意加入或者离开会话。因为用于标准化应用协议的标准基础会话的会话标识符是预先规定的(在 T. 120 中),所以无需任何用户干预某个应用就可以加入到会话中。一个标准基础会话使用一个静态信道作为该会话 ID 信道。按照应用协议的规定在用于标准基础会话的登记过程中需要标识的资源集可以包含更多的静态或动态信道和令牌。如果需要超过该基本集的附加资源,就可能使用动态信道和令牌。

对于在一个会议中希望使用标准应用协议与所有对等的应用进行通信的某个应用,典型的情况是优先选择标准基础会话。对会议中的每个标准化的应用协议,至多有一个标准基础会话。

**非标准基础会话:**被非标准应用协议使用。非标准基础会话允许使用公共的非标准协议独立地确定同一个会话的各种应用,而无需选择特定会话的用户干预。对于在一个会议中希望使用非标准应用协议与所有对等的应用进行通信的某个应用,典型的情况是优先选择非标准基础会话。该非标准基础会话使用一个分配的(即动态组播)信道作为会话 ID,使用 GCC 应用登记处来显示这个标识。加入非标准基础会话的结果就是加入相应的登记会话。对会议中的每个非标准应用协议,不能有多于一个非标准基础会

话。一旦所有的与会者已经离开了非标准基础会话,该会话就不再存在,如果还需要的话就必须重新创建。

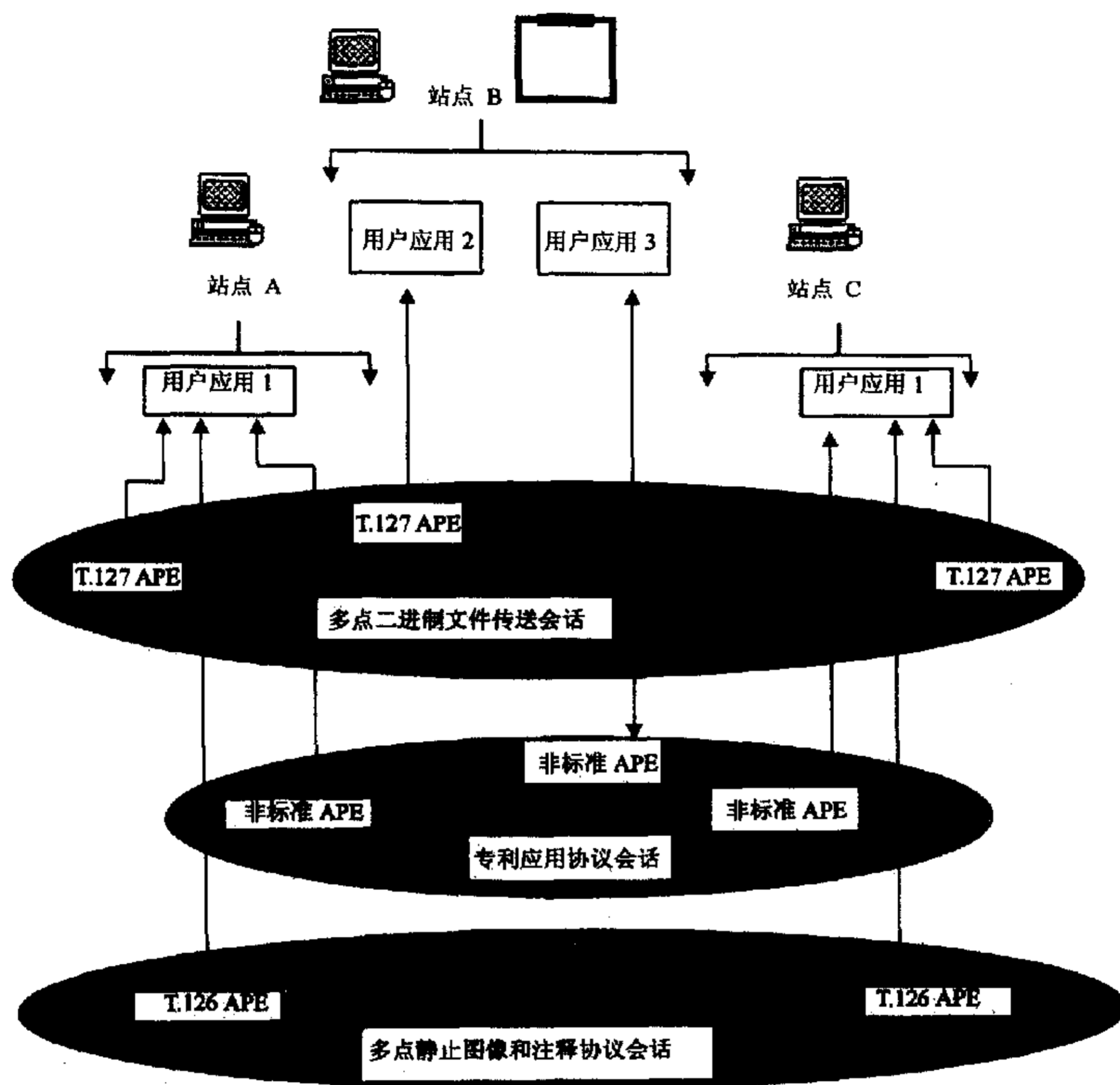


图 4 使用多协议应用的例子

公开会话:可以被标准的或非标准的应用协议使用。当协议的有关基础会话已经在使用中,且用户希望再建立更多的独立会话以便与会议中的所有对等应用进行通信时,典型地使用公开会话。接入到公开会话是不受控的。公开会话有一个指定的创建者,该创建者就是分配组播会话标识符信道并负责登记会话资源的 APE;所有其他与会者都是会话的成员。任何与会者可以随意加入或者离开公开会话;他们可以使用 GCC-Application-Invoke 机制进行选择,以邀请其他的节点来参与会话。在会话的创建者已经离开或甚至拆连以后,公开会话仍可以持续。一旦所有的与会者都离开了公开会话,会话不再存在,因而没有一方能够重新加入。

专用会话:可以被标准的或非标准的应用协议使用。当用户希望把应用协议会话的成员限制在会议参加者的一个选定的子集中时,典型地使用专用会话。专用会话有一个指定的创建者。这就是召集专用会话并且负责登记会话资源的 APE。专用会话的参加者由会话创建者来控制,该会话创建者必须明确地邀请其他应用成为该会话的成员。在会话的创建者已经解散会话或者拆连了以后,专用会话就不再存在了。

### 5.1 用户应用

用户应用指对互操作(例如用户接口)没有直接影响的、而且可能是产品和平台特性的那些任务。用户应用的影响通过它使用的应用协议在其他点体现出来。

用户应用依赖于一个或多个应用协议实体的服务(APE)以便与其他节点的对等用户应用进行通信。它不与 MCS 或 GCC 进行通信;它由几个 APE 来完成。用户应用及其多个 APE 之间的接口是本地

事宜,超出了本标准的范围。

对于用户应用使用的每个 APE,该用户应用规定了宣称的 APE 能力子集。它还选择会话的类型(标准基础、非标准基础、公开、专用或登记)。

用户应用与适当的 APE 进行通信以通过规定会话的应用能力和操作模式发起应用会话。一旦建立了一个会话,所有的应用特定协议的执行均由代表用户应用的 APE 完成。

## 5.2 应用协议实体

应用协议实体(APE)由用户应用来使用,以使得它与具有相似功能(性)的其他用户应用(即为了相同目的利用相同应用协议的用户应用)进行通信,例如,对于要求一般目的的文件传送功能(性)的用户应用必须利用 APE,该 APE 使用了 T.127 定义的协议和 T.127 规定的应用协议键,以保证互操作能力。每个 APE 既可以使用标准应用协议,也可以使用非标准应用协议。

从概念上讲,按图 2 所示 APE 有两个构成体:应用资源管理器(ARM)和应用服务元素(ASE)。ARM 提供通用的功能(性),对许多标准应用协议是公共的,而 ASE 对其相应的应用协议提供特定的功能(性)。

在通用模型中,应用协议实体由下列属性来表征。

- a) 单个应用资源管理器(ARM);
- b) 单个应用服务元素(ASE);
- c) 单个 GCC SAP;
- d) 与节点控制器的相互作用(超出了 T.120 建议的范围);
- e) 一个或多个 MCSAP;
- f) 单个注册的 MCS 用户 ID;
- g) 能力清单(包含可拆装的和不可拆装的能力);
- h) 单个会话 ID(相互通信的对等 APE 共享同一个会话 ID。没有会话 ID 表示 APE 在应用协议的登记会话中注册);
- i) 应用协议键。

## 5.3 应用资源管理器

代表 APE 中的 ASE,应用资源管理器(ARM)负责管理 GCC 和 MCS 资源。ARM 提供如下服务。

- a) 响应来自 GCC 的指示(如允许注册);
- b) 注册具有 GCC 的 APE;
- c) 获得来自 GCC 的句柄;
- d) 连入一个 MCS 域以获得一个用于其 APE 的单一 MCS 用户 ID;
- e) 加入静态信道;
- f) 使用 GCC 登记和 MCS 的标识并加入分配的信道;
- g) 召集专用信道并允许对等 APE 加入这样的信道;
- h) 加入到已允许其 APE 的任何专用信道;
- i) 标识并获得来自 GCC 登记处的令牌;
- j) 使用 GCC 登记处标识并分配参数;
- k) 删除 APE 负责的任何冗余的登记处登录项;
- l) 请求其他节点上的对等 APE;
- m) 处理应用名册报告以确定当前协商应用能力表和对等节点的标识;
- n) 接收 GCC-Conductor-Assign 和 Release 指示;
- o) 通告其 ASE 在组织权状态的变化。

## 5.4 应用服务元素

应用服务元素(ASE)用 ARM 获得的资源向用户应用提供应用协议特定功能(性)。例如,在建议

T.127 中 ASE 提供通用文件传送功能(性)。它的操作与送至它的令牌和信道的类型(即静态的或动态的)及标识无关。ASE 从它的 ARM 获得使用的资源标识。在专用信道上通信的情况下,用户应用必须规定需要与其通信的那些节点的 GCC 节点 ID。

ASE 提供下列服务。

- a) 发送和接收应用协议特定 PDU;
- b) 使用 MCS 捕获和释放令牌并确定令牌状态;
- c) 经过 ARM 加入和离开信道。

表 1 定义了 ARM 和 ASE 构成体之间使用的原语。在本表中没有提及的 GCC 原语由节点控制器处理。

表 1 ARM/ASE 使用的原语

原语	ARM	ASE	节点控制器
GCC-Registry-Register-Channel	✓		
GCC-Registry-Assign-Token	✓		
GCC-Registry-Set-Parameter	✓		
GCC-Registry-Retrieve-Entry	✓		
GCC-Registry-Delete-Entry	✓		
GCC-Registry-Monitor	✓		
GCC-Registry-Allocate-Handle	✓		
GCC-Conference-Roster-Inquire	✓		✓
GCC-Conductor-Assign	指示		✓
GCC-Conductor-Release	指示		✓
GCC-Conductor-Please			✓
GCC-Conductor-Give			✓
GCC-Conductor-Inquire	✓		✓
GCC-Conductor-Permission-Ask	指示		✓
GCC-Conductor-Permission-Grant	指示		✓
GCC-Conductor-Permission-To-Enroll	✓		
GCC-Application-Enroll	✓		
GCC-Application-Roster-Report	✓		✓
GCC-Application-Roster-Inquire	✓		✓
GCC-Application-Invoke	请求		✓
MCS-Attach-User	✓		
MCS-Detach-User	✓		
MCS-Channel-Join	✓		
MCS-Channel-Leave	✓		
MCS-Channel-Convence	✓		



表 1(完)

原语	ARM	ASE	节点控制器
MCS-Channel-Disband	✓		
MCS-Channel-Admit	✓		
MCS-Channel-Expel	✓		
MCS-Send-Data		✓	
MCS-Uniform-Send-Data		✓	
MCS-Token-Grab		✓	
MCS-Token-Inhibit		✓	
MCS-Token-Give		✓	
MCS-Token-Please		✓	
MCS-Token-Release		✓	
MCS-Token-Test		✓	

### 6 通用应用模板

将通用应用模板(GAT)设计成对标准和非标准应用协议均提供通用资源管理功能(性)。因此,它具有一个应用资源管理器(通用 ARM),但没有应用服务元素(ASE)。GAT 的示意图 5。不同的应用协议规范详述了它们各自的 ASE 操作,并标识出 ASE 哪些服务需要由通用 ARM 提供。它们不需描述 ARM 的操作,而应参考本标准。通用 ARM 提供了 5.3 描述的所有服务。

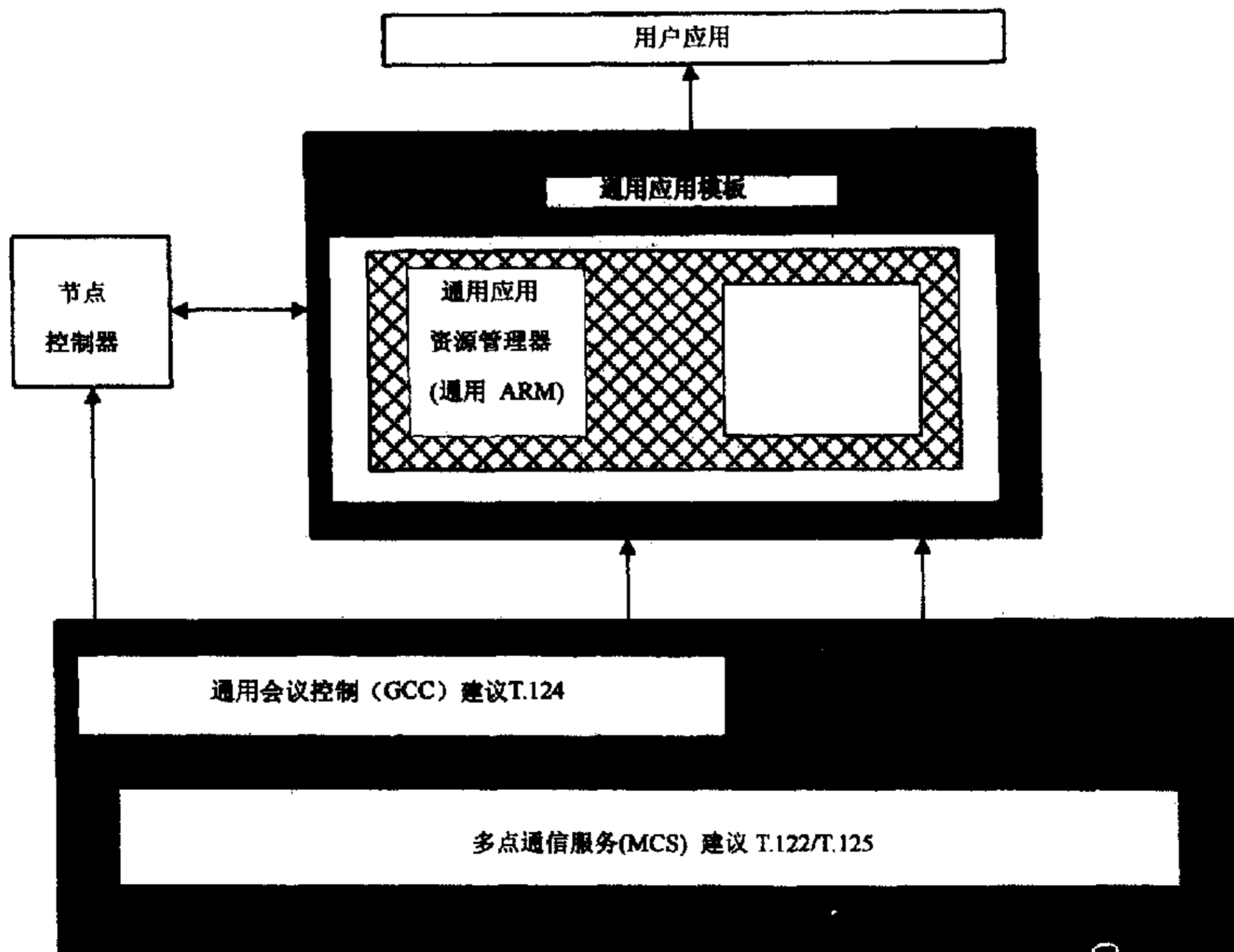


图 5 通用应用模板

下面描述通用 ARM 提供的功能(性)描述。

### 6.1 初始化

ARM 代表用户应用负责注册它的应用协议实体,并负责获得 ASE 需要的必不可少的资源。对于使用的每个应用协议,用户应用必须对相应 ARM 规定如下参数。

- a) 它是否愿意注册(如果用户应用不愿意注册就不需要更多的参数了);
- b) 应用协议键;
- c) 会话类型(登记的、标准基础的、非标准基础的、公开的或专用的);
- d) 会话 ID(如果声明通过登记会话来支持协议,在非标准基础会话中注册或者创建新的公开或专用会话,就省略该参数);
- e) 所需的动态令牌清单;
- f) 所需的分配信道;
- g) 所需的专用信道清单(作为选项,被邀请到那些信道的节点);
- h) 它是否能够作为应用会话组织者(如果在一个站点有多于一个应用加入到同一个会话中,本地 GCC 提供者应确定哪一个将成为组织者);
- i) 支持的应用能力清单(按在应用协议规范中的定义)。可拆装的能力对于登记会话应当被省略,且对于所有其他的会话应当包括。

对于用户应用所使用的每个应用协议,表 2 规定了传送到 ARM 的参数。

### 6.2 注册

为了参加一个会议,对于每个协议,用户应用需要一个 APE(由一个 ARM 和一个 ASE 组成)。创建 APE 的方法是本地的事情,超出了本标准的范围。每个 APE 必须建立一个 GCC SAP,以允许它在那个节点上与 GCC 提供者进行通信。如何实现这一点是本地的事情。类似地,每个 APE 必须建立一个 MCSAP,以允许它与其本地 MCS 提供者进行通信。

当节点加入一个会议,或者 APE 建立了一个 GCC SAP,接着加入一个或多个会议的时,本地 GCC 提供者通过发布一个 GCC-Application-Permission-to-Enroll 指示,将授予/撤消标志设置成授予,通告在那个节点上的所有 APE。对于用户应用所使用的每个协议,相应的 ARM 接着应当发出一个 GCC-Application-Enroll 请求,而不管用户应用是否希望在那个时候注册。如果该用户应用不希望注册 APE,相应的 ARM 应当规定会议 ID 并将 GCC-Application-Enroll 请求中的注册/未注册标志设置成未注册。不需要其他参数。除非因收到 GCC-Application-Permission-to-Enroll 指示将授予/撤消标志设置成撤消而将许可撤消,用户应用可以命令它的 ARM(一个或多个)在任何时候注册。

用户应用可能希望在决定是否参加一个现有的会话或者创建一个新会话之前接收所有进行中的会话信息。按 6.2.1 中描述的,通过命令其 ARM(一个或多个)注册为非激活,在登记会话中就可以做到这一点。

如果用户应用希望告知其他节点处的对等应用它出席了会话,但是不希望立即参与会话,它可以在登记会话中命令它的 ARM 注册为非激活而无需 MCS 用户 ID。这使得用户应用能宣布支持一个或多个协议,而不消耗 MCS 资源。

表 2 APE 参数

参 数	描 述
enroll/un-enroll	若用户应用希望立即注册,本参数就设置成 ENROLL,如果用户应用希望撤消注册,本参数就设置成 UN-ENROLL。如果设置成 UN-ENROLL,就不需要更多的参数了
Applicationprotocol-key	应用协议键标识出在应用协议会话中所用的应用协议。具有同一目的的多应用协议会话利用相同的应用协议键来标识的(但是有不同的会话 ID)。一个应用协议键既可以是属于一个建议、标准或非标准协议的一个 ASN.1 目标识别符,也可以是一个利用建议 T.124(像 H.221 一样使用了 T.35 国家代码)的编码协定的非标准标识符

表 2(续)

参 数	描 述
Session Type	<p>本参数能够具有以下 5 个值之一。</p> <p>登记: 登记会话允许用户应用宣告他出席了会议, 并且还提供了一种确定非标准基础会话的方式。不需要会话 ID。</p> <p>标准基础: 该值表示 ARM 应当使用由应用协议键和会话 ID 参数组成的会话键进行登记。它应当按照使用应用协议规范中定义的并按照建议 T. 120 中的分配, 作为资源的基本集的预先定义的静态信道和静态令牌。该会话类型只是在标准化的应用协议中有效。</p> <p>非标准基础: 该值表示 ARM 应当首先在登记处会话中注册以确定通过与 GCC 登记处协商, 分配给非标准基础会话的会话 ID。如果会话 ID 已不存在, 就由 ARM 分配一个会话 ID。接着该 ARM 利用经由应用协议键和会话 ID 参数组成的会话键在非标准基础会话中注册。非标准基础会话的成员可以由 GCC 登记来确定令牌和信道 ID。该会话类型只是对非标准应用协议有效。</p> <p>公开: 该值表示 ARM 应当利用由应用协议键和会话 ID 参数组成的会话键进行注册。如果省略了会话 ID 参数, 就由 ARM 分配一个会话 ID 参数。所有的信道和令牌资源都是动态的, 并由公开会话的创建者分别利用 MCS-Channel-Join 和 GCC-Registry-Assign-Token 机制来分配。公开会话的成员可以经由 GCC 登记来确定令牌和信道 ID。</p> <p>专用: 该值表示 ARM 应当利用由应用协议键和会话 ID 参数组成的会话键注册。如果省略了会话 ID, 就由 ARM 分配一个会话 ID。所有的信道和令牌资源都是动态的, 并由专用召集者 ARM 分别利用 MCS-Channel Convene 和 GCC-Registry-Assign-Token 机制来分配。创建会话的 ARM 可以允许在节点处对等 APE 加入信道, 其 GCC 节点 ID 存在于 admitList 协议参数中。</p> <p>专用成员 ARM 在试图加入之前必须等待专用 MCS 信道的召集者的许可</p>
SessionID	<p>本参数用于区分由同一个 MCS 域中可能同时存在的该协议的多个会话使用的资源。</p> <p>如果用户应用希望参与一个标准基础会话或者参加一个现有的公开或专用会话, 就必须规定会话 ID。如果该用户应用希望创建一个新的公开或专用会话, 参与一个非标准基础会话, 或者该用户应用希望在登记会话中注册而不消耗 MCS 资源, 就省略会话 ID。除了登记会话外, 由于在会议域内保证会话 ID 是唯一的, 作为启动信道所分配的 MCS 信道 ID 被用作会话 ID</p>
AdmitList	<p>如果会话类型 = 专用和省略的会话 ID</p> <p>则对应于节点的 GCC 节点 ID 清单, 在该节点, 对等 APE 被允许加入专用的被召集的信道。</p> <p>否则</p> <p>参数值省略</p>
ResourceList	<p>这是用户应用所需的一个资源(信道、令牌、登记参数和句柄)的清单。</p> <p>如果会话类型 = 标准基础</p> <p>则资源清单包含由 APE 加入的静态信道的 MCS ID 和由 APE 使用的静态令牌。它可有选择地包含一个资源 ID 清单以便经由 GCC 登记确定动态信道和令牌的 MCS ID。它还可以规定所需的登记处理的数量, 并包含一个用于登记参数和那些(合适的)参数值的资源 ID 清单。</p> <p>如果会话类型 = 公开、专用或者非标准基础</p> <p>则资源清单包含一个资源 ID 清单, 该资源 ID 清单经由 GCC 登记确定动态信道和令牌的 MCS ID。它还可以规定所需的登记句柄的数量, 并包含一个用于登记参数和那些(合适的)参数值的资源 ID 清单。如果省略了会话 ID(即 APE 是会话召集者), ARM 就负责分配和登记在资源清单中规定的所有信道和令牌。如果已有会话 ID(即 APE 是一个会话成员), ARM 就负责收回所有的信道和令牌。在这两种情况下, 资源 ID 和相关联的 MCS 信道 ID 或者令牌 ID 都由 ARM 传递到它的 ASE。</p> <p>如果会话类型 = 登记</p> <p>参数值省略</p>

表 2(完)

参 数	描 述
组织者操作能力	在组织者模式下,若 APE 有能力成为会话组织者则设置
不可拆装的能力清单	由应用协议规定的且由 ASE 支持的任何不可拆装的能力
应用能力清单	由应用协议规定的且由 ASE 支持的任何可拆装的能力
期望的能力清单	<p>如果会话类型=专用和省略的会话 ID 表明必须被调用的 APE 所支持的任何应用能力,允许专用会话的创建者为那个会话规定一个最小能力集。</p> <p>否则 参数值省略</p>

当应用决定变为激活时,每个 ARM 必须采用一个 MCS 用户 ID。使用 GCC-Application-Permission-to-Enroll 指示中包含的会议 ID 作为域选择符,通过向 MCS 提供者发出 MCS-Attach-User 请求,ARM 获得了一个新的 MCS 用户 ID。ARM 在接收到一个成功的 MCS-Attach-User 证实后,作为响应,该 ARM 应当加入到通过发布 MCS-Channel-Join 请求所指示的用户 ID 信道,并向相应的 ASE 提供用户 ID。

接下来的操作取决于会话类型。

注:对于用户应用所使用的每个 APE,需要重复描述的过程。注意到模型模板的一个共同特征(登记会话模板除外)是会话 ID 信道是启动信道,它的类型决定了会话类型,并且它总是在 APE 的初始序列期间加入。

为会话注册和加入启动信道而定义的规程必须遵循描述的每个会话类型所规定的顺序。如果应用协议不需要使用任何更多的资源(信道、令牌或登记参数),ARM 就可以省略非激活的注册操作。在应用协议使用令牌、登记参数或者附加动态信道的地方,本标准不需要分配或标识那些资源的顺序。如果应用协议使用 GCC 句柄,可以在应用注册处理期间和随后的会话持续期间的任何时间分配 GCC 句柄。

为了将初始化时间缩短到最小,通过交错进行每个信道所需的操作允许以并行方式标识多个信道。例如,会话创建者可以选择召集所有所需的信道,然后登记所有那些信道,而不是分别召集和登记每个信道。然而对于每个给定的信道,必须遵循本标准中规定的操作序列。当其 ARM 接收到一个 GCC-Application-Roster-Report 指示(在其中有实体 ID)时,APE 只能看作会话的一部分。APE 在接收到这样一个名单报告指示之前不应向会话提交数据。然而,APE 应当准备缓冲处于会话中的注册激活与接收名单报告(在其中有 APE 实体 ID)之间用于会话而接收到的所有数据。这是因为另一个 APE 可能已收到了包含新与会者的名单报告指示并在相应的名册报告之前已向会话提交了到达新与会者的数据。应用协议可以规定如何处理该缓冲数据。

### 6.2.1 登记会话

如果用户应用希望告知另一节点上的对等用户它出席了会话,但不希望立即参与会话,则它可以命令其 ARM 在登记会话中注册,而不需要 MCS 用户 ID。这使得用户应用能宣布支持一个或多个应用协议而不消耗 MCS 资源。它还为被通告的每个应用协议接收有关处理进程中的所有会话信息。每个 ARM 发出一个 GCC-Application-Enroll 请求,将注册/未注册标志设置成注册,并将激活/非激活标志设置成非激活,规定相应协议的会话键,而不需要会话 ID。见图 6。

### 6.2.2 标准基础会话

在加入其用户 ID 信道后,ARM 应当按相应的应用协议规范的规定加入静态信道。一旦已收到了加入这些信道的肯定证实,ARM 应通过向 GCC 提供者发出一个具有表 3 所规定的参数的 GCC-Application-Enroll 请求将该 APE 注册为非激活。

在接收到具有相应于其 APE 的登录项的 GCC-Application-Roster-Report 指示之后,ARM 应当分配协议规范规定的任何动态指配信道。ARM 应当为每个这样的信道发出一个 GCC-Registry-Retrieve-Entry 请求原语,以确定它是否已经被登记。如果在 GCC-Registry-Retrieve-Entry 证实中返回的结果参数是“成功”,那么 ARM 应当试图通过发出一个 MCS-Channel-Join 请求加入到登记事项参数中指示

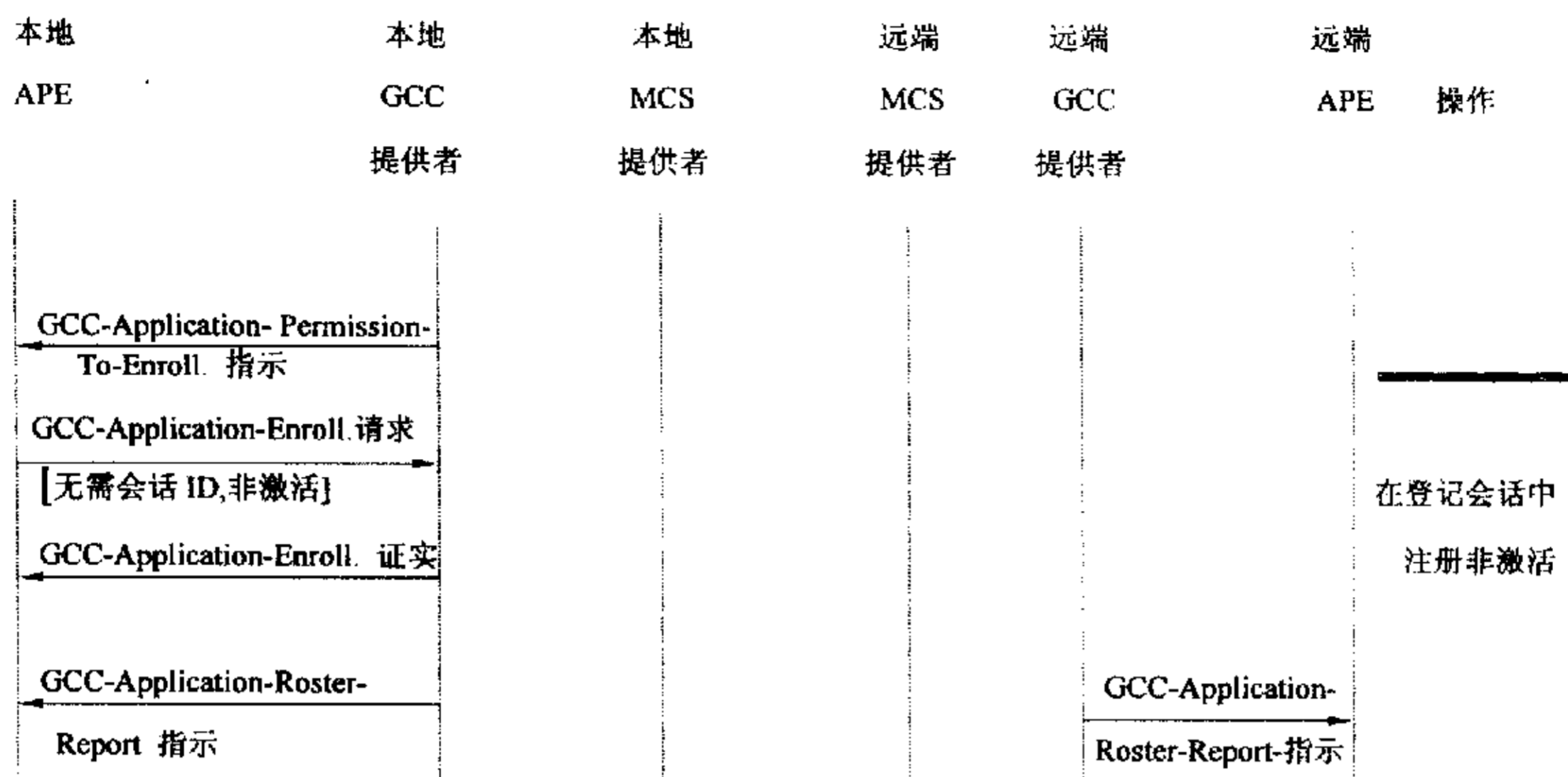


图 6 在登记会话中注册以宣布支持协议的

信道。如果在 GCC-Registry-Retrieve-Entry 证实中返回的结果参数是“登录项未找到”，ARM 就应当试图通过发出一个具有信道 ID=0 的 MCS-Channel-Join 请求，而分配一个新信道。如果是成功，被返回的 MCS-Channel-Join 证实就包含被分配的信道 ID，ARM 将试图通过发出一个 GCC-Registry-Register-Channel 请求在标准基础会话中登记该信道 ID。接收到了 GCC-Registry-Register-Channel 证实后，ARM 应当检查该结果参数。如果结果是“成功”，ARM 没有进一步的动作。如果结果是“索引已经存在”，那么说明该信道已经被另一个 APE 登记了，其信道 ID 在登记事项参数中提供。在这种情况下，ARM 将首先通过发出一个包含在 MCS-Channel-Join 证实中返回的信道 ID 的 MCS-Channel-Learn 请求而脱离先前已分配的信道。接着它将通过发出一个 MCS-Channel-Join 请求而加入到登记事项参数中指示的信道。

如果应用协议命令 APE 必须在与它的对等实体进行交互作用之前标识出任何动态令牌，那么该 ARM 可以经由 GCC 登记来确定它们的标识。对于应用协议规定的每个令牌资源 ID，ARM 应当利用表 7 中规定的参数向 GCC 提供者发出 GCC-Registry-Assign-Token 请求。如果从 GCC-Registry-Assign-Token 证实返回的结果参数是“成功”或者“索引已存在”，那么包含在该返回证实原语的登记事项的令牌 ID 被用作对应于登记键中使用的资源 ID 令牌的令牌 ID。ARM 应当将该资源 ID 和相关联的令牌 ID 传递给它的 ASE。

如果应用协议需要一个 APE，以便在与它的对等交互作用之前分配或者标识出登记参数，那么 ARM 应当向 GCC 提供者发出一个 GCC-Registry-Set-Parameter 请求。

一旦会话所需的全部资源均已标识出，这时 ARM 就应当通过发出 GCC-Application-Enroll 请求而登记激活。激活/非激活标志应设置成激活，会话 ID 应规定成会话键部分，启动信道应规定成静态的，同时应提供能力表。见图 7 和图 8。

### 6.2.3 非标准基础会话

登记会话可以用来提供接入到非标准基础会话。在加入到其用户 ID 信道之后，ARM 应当通过发出一个 GCC-Application-Enroll 请求，将激活/未激标志设置成非激活，并通过规定会话键没有会话 ID 而注册在该登记会话中。

在接收到具有对应于其 APE 登录项的一个 GCC-Application-Roster-Report 指示之后，ARM 应通过首先发出一个具有信道 ID=0 的 MCS-Channel-Join 请求来试图创建非标准基础会话。如果是成功的，则返回的 MCS-Channel-Join 证实就包含分配的信道 ID，ARM 将试图在登记会话中登记该信道 ID 作为非标准基础会话的启动信道。ARM 应当利用由会话键(没有会话 ID)和资源 ID“BASE”组成的登记键来发出一个 GCC-Registry-Register-Channel 请求，该资源 ID“BASE”按照建议 T.50 用 4 个连续

的八位组编码。收到了 GCC-Registry-Register-Channel 证实之后,该 ARM 应当检查结果参数。

如果结果是“索引已经存在”,那么非标准基础会话就已经被另一个 APE 登记了,其会话 ID 在登记事项参数中提供。在这种情况下,ARM 应当通过发出一个包含有 MCS-Channel-Join 证实中返回的信道 ID 的 MCS-Channel-Leave 请求而首先从先前分配给它的信道中离开。ARM 应当通过遵循 6.2.4 中描述的公开会话成员的过程,担当起会话成员的角色,并在现有的基础会话中注册。该过程如图 9 所示。如果登记事项参数指向一个不再存在的非标准会话(即会话的启动信道不再存在),那么加入 GCC-Registry-Register-Channel 中标识的启动信道的企图就可能失败;在这样的环境下,ARM 可能通过发出一个 GCC-Registry-Delete-Entry 请求,同时使用与以前相同的登记键来试图重新使用这个并不存在的会话。在收到了相应的 GCC-Registry-Delete-Entry 证实之后,ARM 应当接着检查结果参数,如果结果是“成功”,则 ARM 可以重复非标准基础会话创建过程。

如果 GCC-Registry-Register-Channel 证实的结果参数是“成功”,那么 ARM 应当承担该非标准会话创建者的能力,见图 10。该 ARM 应当在新的基础会话中注册非激活,如表 3 所规定。

表 3 GCC-Application-Enroll 请求的参数

参 数	内 容
会议 ID	由 GCC-Application-Permission-to-Enroll 指示提供
会话键	由应用协议键和会话 ID 组成
应用用户 ID	由 MCS-Attach-User 证实提供
激活/非激活	当 ARM 已经标识了那些作为注册过程的部分需要标识的资源集,而且加入了协议所需的全部信道时,则为激活 当标识之前已注册了那些作为注册过程的部分需要标识的资源集时,对登记会话以及标准基础、非标准基础、公开或者专用会话,则为非激活
组织者操作能力标志	如果在组织者模式下 APE 能够成为会话组织者就设置该标志。如果激活/非激活标志设置成非激活,就不一定要设置该标志
启动信道	规定下列会话类型的可选参数
	会话类型      启动信道
	登记            省略的
	标准基础      静态
	非标准基础    动态组播
	公开            动态组播
专用            动态专用	
不可拆装的能力清单	由应用协议规定并由 ASE 支持的任何不可拆装的能力
应用能力清单	由应用协议规定并由 ASE 支持的任何可拆装的能力 若激活/非激活标志设置成非激活,则省略之
注册/未注册	ENROLL

如果协议规范规定使用任何附加信道,那么 ARM 应当向每个这样的信道发出 GCC-Registry-Retrieve-Entry 请求原语以确定它是否已经登记。用于登记这些资源的登记键应当包含非标准基础会

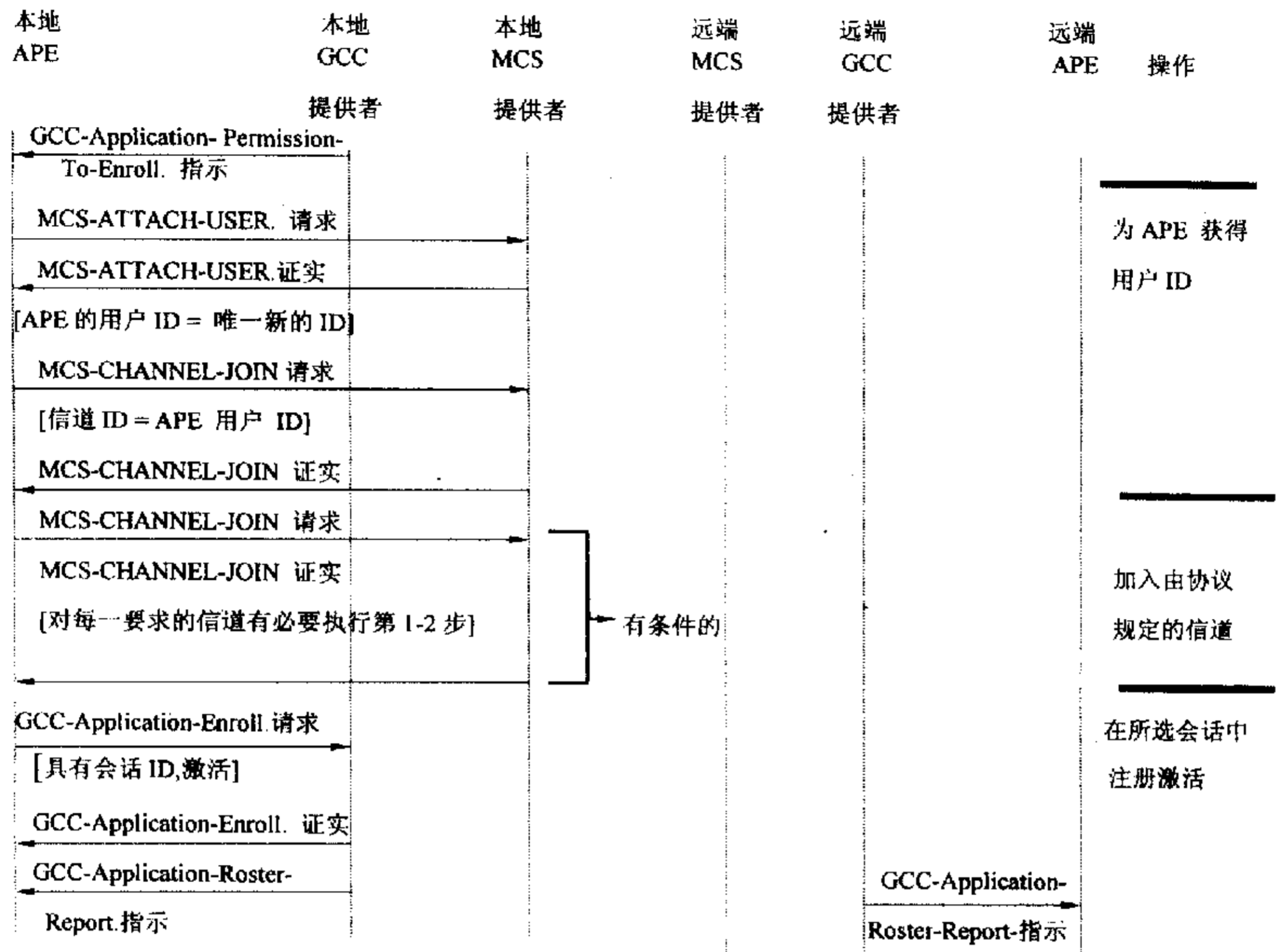


图 7 标准基础会话协议初始化顺序示例的第一种情况:仅使用静态资源

话的会话 ID 作为会话键的一部分。如果 GCC-Registry-Retrieve-Entry 证实中返回的结果参数是“成功”,那么 ARM 应当试图通过发送一个 MCC-Channel-Join 请求而加入到登记事项参数中指定的信道。如果 GCC-Registry-Retrieve-Entry 证实中返回的结果参数是“未找到登录项”,那么 ARM 应当试图通过发送一个 MCC-Channel-Join 请求,使信道 ID=0 而分配一个新的信道。返回的 MCC-Channel-Join 证实如果是“成功”,则包含分配的信道 ID,ARM 将试图通过发出一个 GCC-Registry-Register-Channel 请求在非标准基础会话中登记该分配的信道 ID。在接收到 GCC-Registry-Register-Channel 证实之后,ARM 应当检查结果参数。如果其结果是“成功”,ARM 就不应执行进一步的动作。如果其结果是“索引已经存在”,那么该信道已经被另一个 APE 登记了,它的信道 ID 在登记事项参数中提供。在这种情况下,ARM 应当通过发出一个包含有 MCC-Channel-Join 证实中返回的信道 ID 的 MCC-Channel-Leave 请求,而首先从先前分配给它的信道中离开。该 ARM 应当通过发出一个 MCC-Channel-Join 请求而加入到登记事项参数所指定的信道。

如果应用协议委托 APE 在与它的对等实体交互作用之前必须知道令牌的标识,则 ARM 可以经由 GCC 登记确定它们的标识。对于应用协议规定的每个令牌资源 ID,ARM 应当利用表 7 中规定的参数向 GCC 提供者发出一个 GCC-Registry-Assign-Token 请求。如果从 GCC-Registry-Assign-Token 证实中返回的结果参数是“成功”或者“检索已经存在”,那么包含在返回的证实原语中的登记项的令牌 ID 就被用作在登记键中使用资源 ID 所对应的令牌的令牌 ID。ARM 应当向它的 ASE 传送该资源 ID 和相应的令牌 ID。

如果应用协议要求 APE 在与它的对等实体交互作用之前分配或者标识出登记处参数,那么 ARM 就应向 GCC 提供者发出一个 GCC-Registry-Set-Parameter 请求。

一旦会话需要的所有资源均被标识出,ARM 将发出一个包含有表 3 规定的参数的 GCC-Application-Enroll 请求而注册激活。

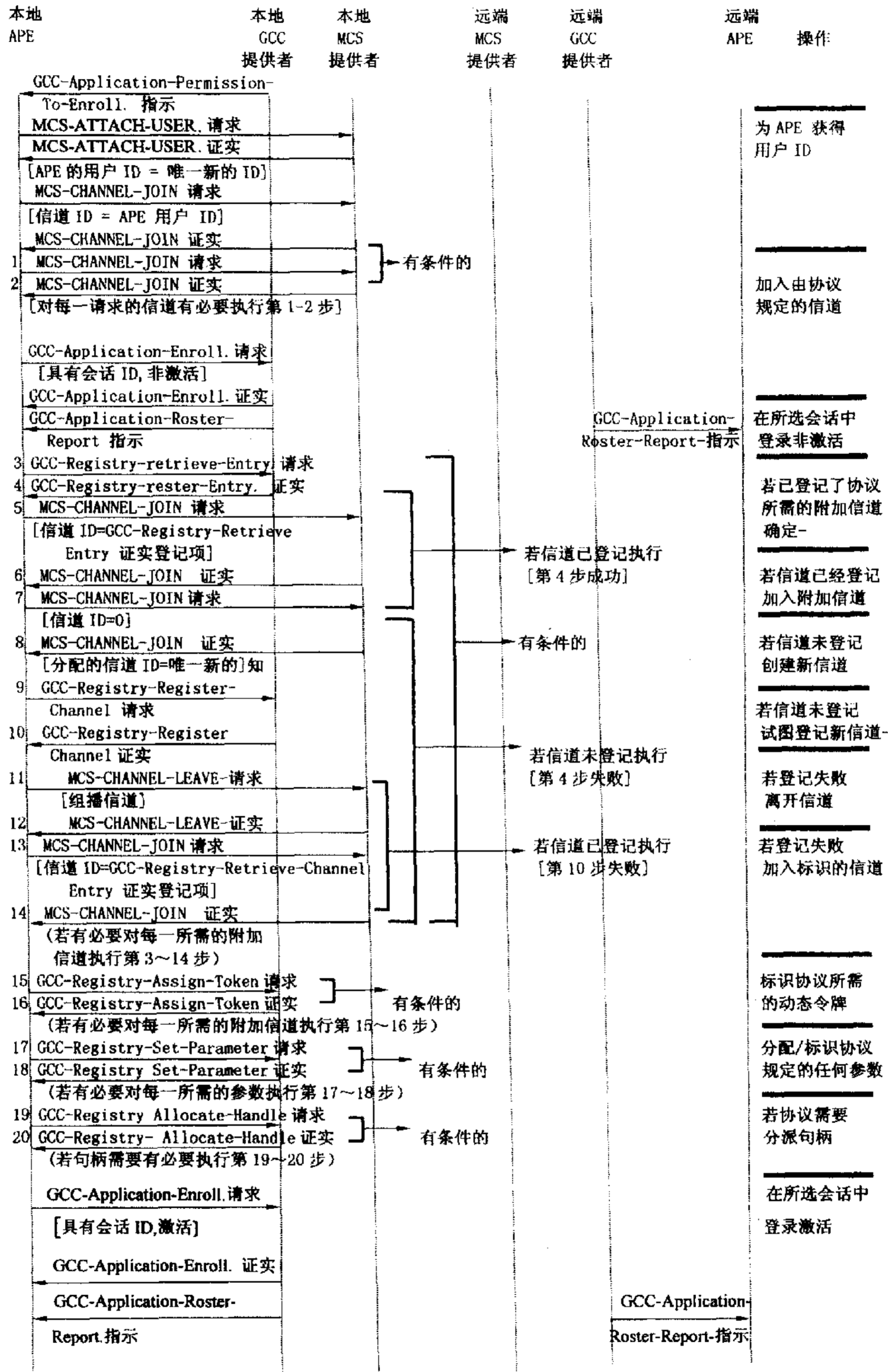


图 8 标准基础会话协议初始化顺序示例的第二种情况:使用静态和动态资源



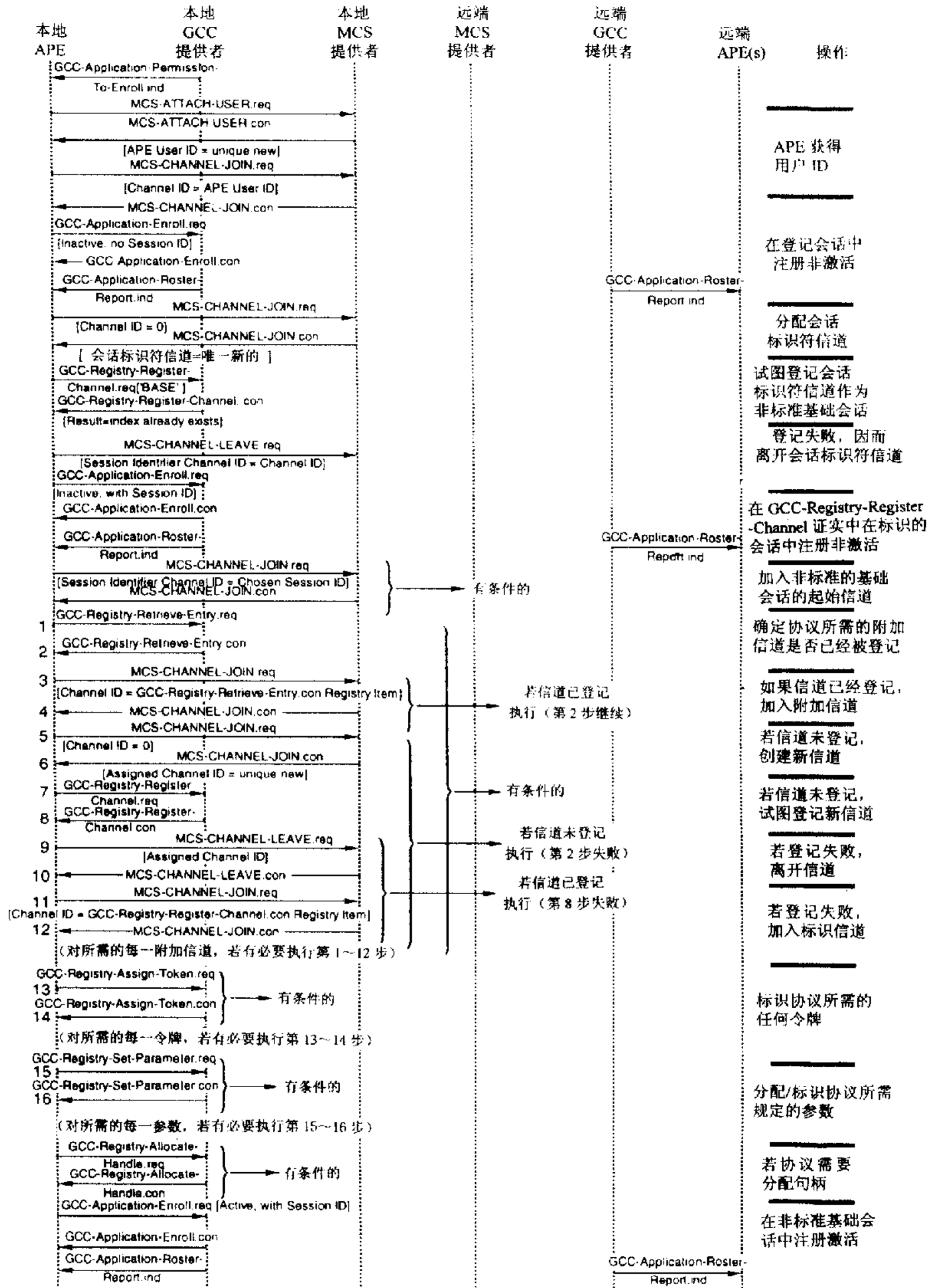


图 9 非标准基础会话协议初始化顺序示例第一种情况:基础会话已经登记

### 6.2.4 公开会话

在加入到其用户 ID 信道之后,ARM 应检查会话 ID 参数,若该参数存在,就加入一个现存的公开会话(作为一个公开成员)中,或者,若省略了该参数,就创建一个新的公开会话(作为一个公开创建者)。见图 11 和图 12。

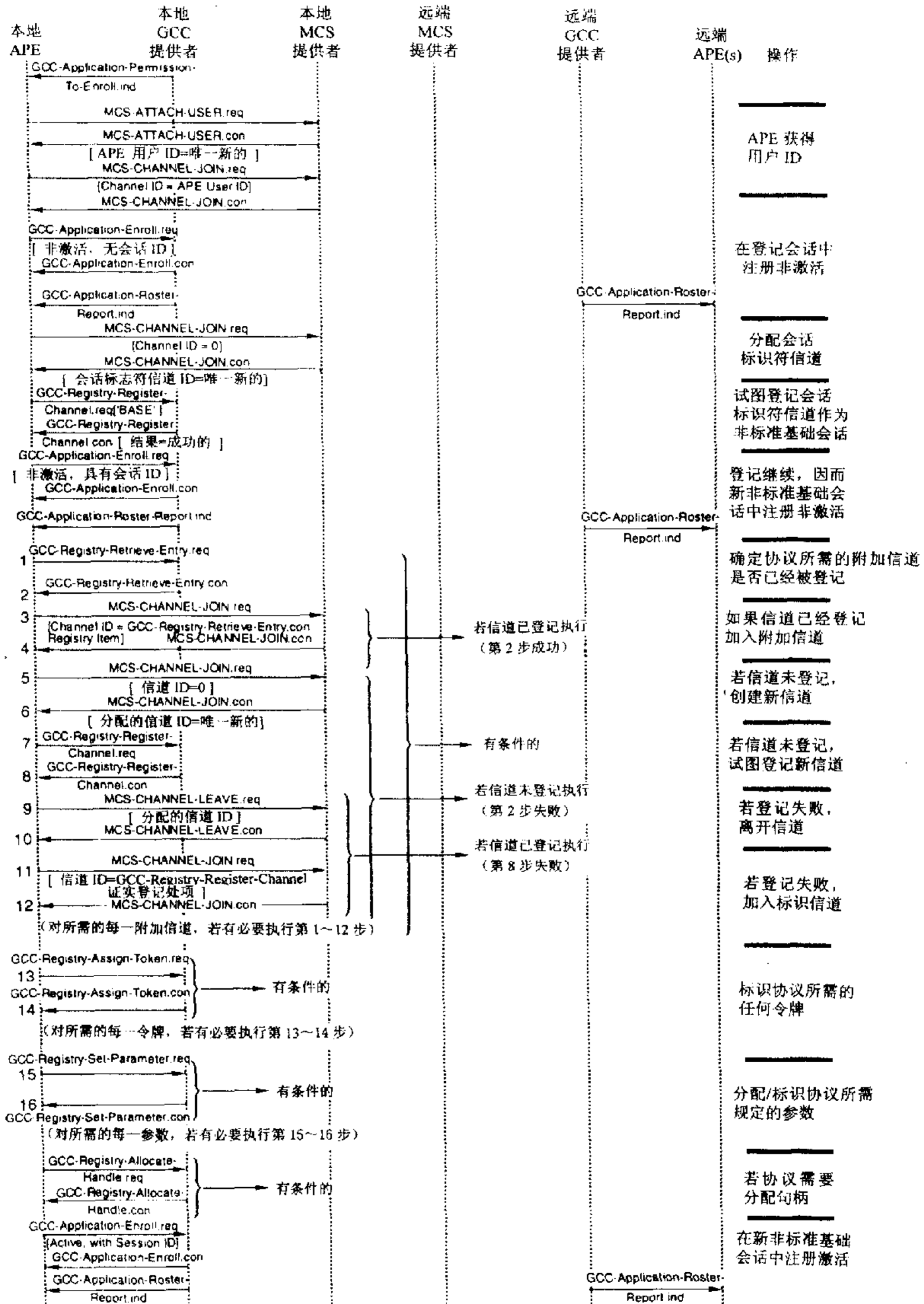


图 10 非标准基础会话协议初始化顺序示例第二种情况:基础会话未登记

如果会话 ID 参数存在,ARM 应在指定的会话中进行注册。进行的注册是通过发出一个将激活/非激活标志设置成非激活的 GCC-Application-Enroll 请求,并且规定具有请求会话 ID 的会话键来实现。然后 ARM 发出一个 MCS-Channel-Join 请求,规定所选会话的会话 ID 作为信道 ID 参数。

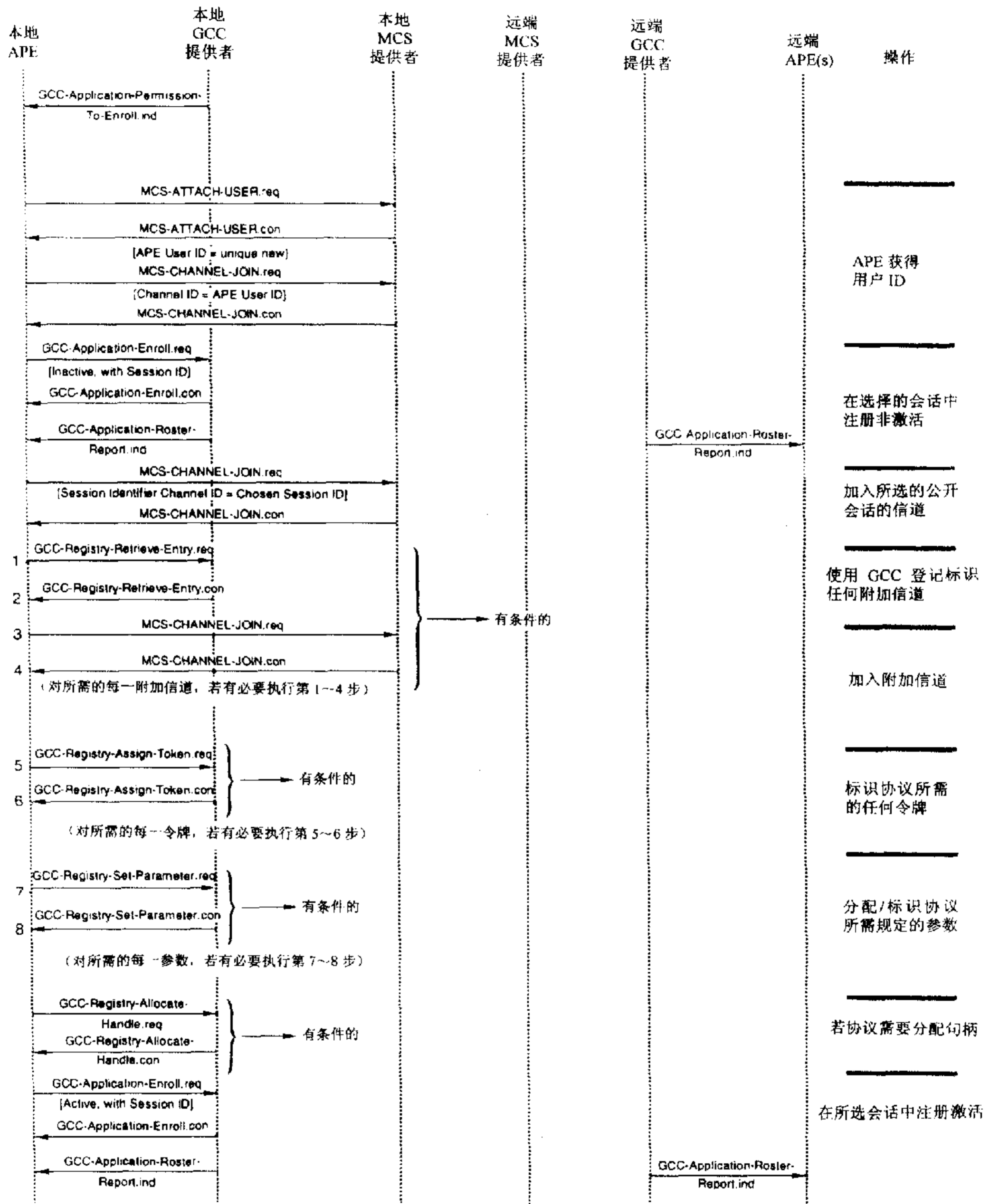


图 11 公开会话协议初始化顺序示例第一种情况：会话成员

如果在 APE 能够被认为是会话中一个激活的与会者之前应用协议要求多个信道加入, ARM 则应试图分别利用 GCC 登记和 MCS 的服务来标识并加入到这些信道中。当应用会话处于进程之中时, 可以按需要加入更多的信道, 同样, 如果应用协议委托 APE 在与它的对等实体交互作用之前必须知道令

牌或登记处参数的标识,那么 ARM 可以经由 GCC 登记来确定它们的标识。

一旦会话需要的所有资源均已标识出来,这时 ARM 应通过发出一个具有表 3 中规定的参数 GCC-Application-Enroll 请求而在所选的会话中注册激活。

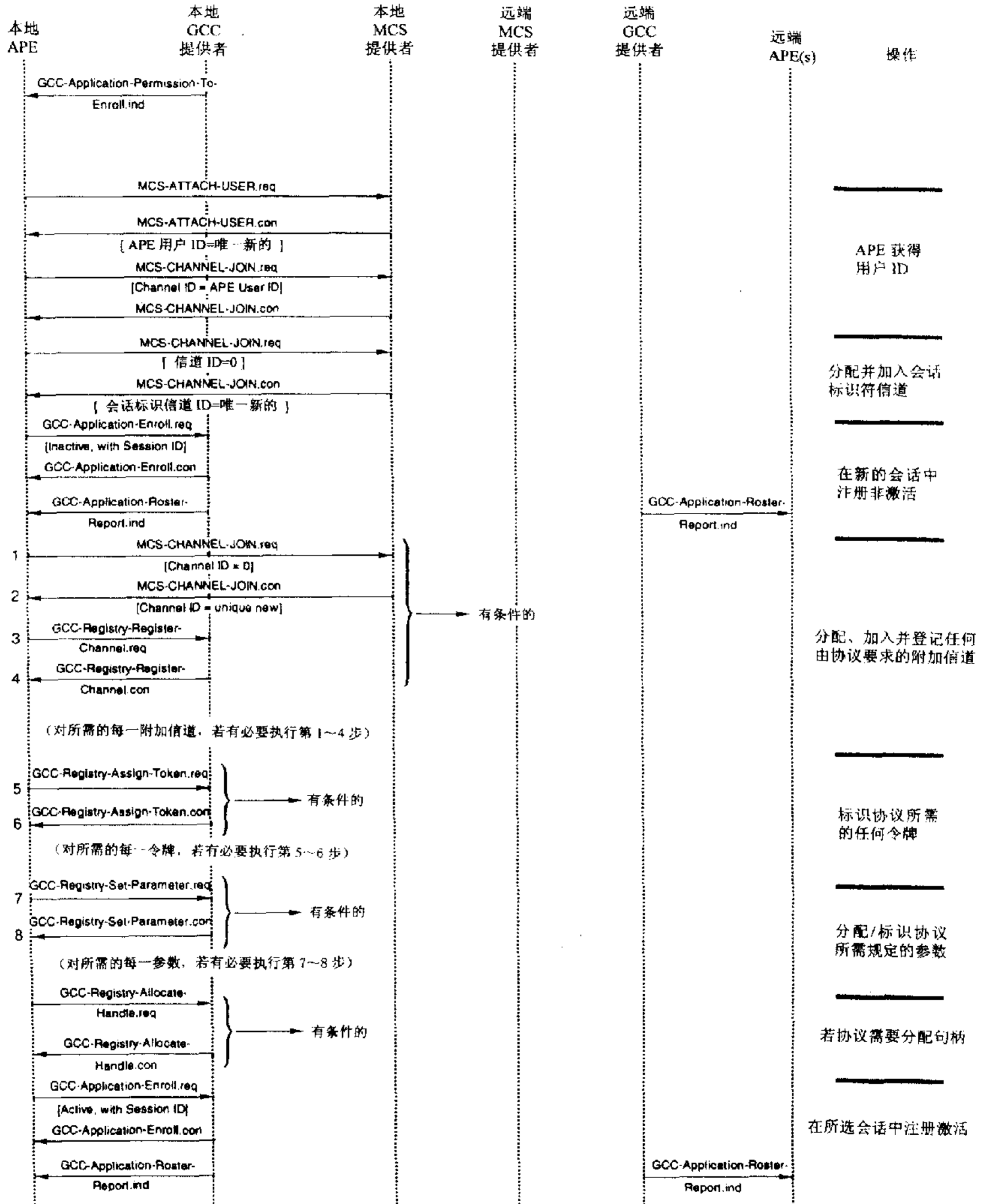


图 12 公开会话协议初始化顺序示例第二种情况:会话创建者

如果省略了会话 ID 参数,ARM 应在新的公开会话中注册。ARM 将通过发出具有 ID=0 的 MCS-Channel-Join 请求首先给会话分配一个分配的信道。如果成功,返回的 MCS-Channel-Join 证实包含分

配的信道 ID。这个信道 ID 也被用作 APE 的会话 ID。然后 ARM 通过发出具有表 3 中规定的参数的 GCC-Application-Enroll 请求,在该会话中注册非激活。

获得作为公开会话注册过程之一部分需要进行标识的资源集的责任由创建该会话的 ARM 承担。如果任何附加信道、登记处参数或者令牌是由协议规范来委托的,那么 ARM 应当在必要时利用 GCC 来对其进行分配和登记。用于登记这些资源的登记键应包含作为该会话键的会话 ID。注意直到它已经接收到了包含对应于它自身的 APE 登录项的 GCC-Application-Roster-Report 指示,ARM 才试图使用 GCC 登记处。

一旦所有的资源都被成功地分配,ARM 就通过发出包含有表 3 中规定的参数的 GCC-Application-Enroll 请求来注册激活。

### 6.2.5 专用会话

在加入到其用户 ID 信道之后,ARM 应检查会话 ID 参数,以确定若该参数存在,就参加一个现存的专用会话(作为一个专用成员)。或者,若省略了该参数,就创建一个新的会话(作为专用会话召集者)。见图 13。

如果会话 ID 参数存在,通过用激活/非激活标志设置成非激活的,发出 GCC-Application-Enroll 请求,以及通过用必要的会话 ID 规定会话键 ARM 应在所指定的会话中注册非激活。然后 ARM 应等待,直到它从召集专用会话的节点处的对等 ARM 接收到了 MCS-Channel-Admit 指示为止。接下来该 ARM 通过发出 MCS-Channel-Join 请求试图加入到在这个原语中指示的信道,如果信道加入成功,则通知相应的 ASE。信道 ID 被用作会话 ID。

如果在 APE 能被认为是一个会话的激活参加者之前,应用协议需要将要被加入的多个信道,那么 ARM 将等待被允许到那些信道并加入那些信道。当应用会话在进行时,可以按照需要加入到更多的信道。同理,如果应用协议管理 APE 在与它的对等端相互作用之前必须知道令牌或登记参数的标识,那么 ARM 可以经由 GCC 登记确定它们的标识。

接着 ARM 将发送一个具有表 3 中规定的参数的 GCC-Application-Enroll 请求登记激活。

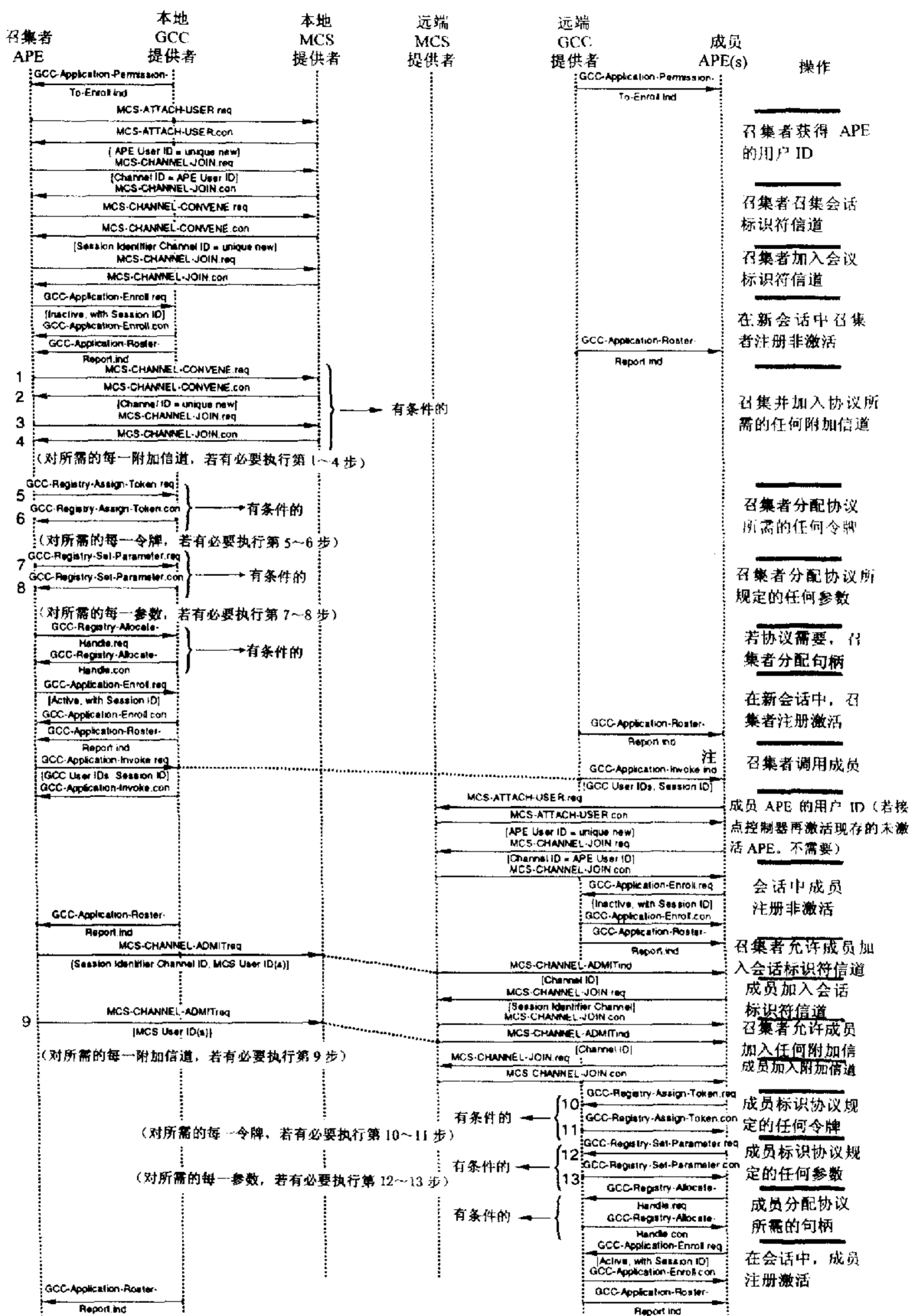
如果省略了会话 ID 参数,那么 ARM 应试图创建一个新的专用会话。该专用会话召集者 ARM 应通过发出一个 MCS-Channel-Convene 请求首先召集一个专用信道。如果成功,则返回的 MCS-Channel-Convene 证实包含分配信道的 ID。该信道 ID 也就成为会话 ID。接着 ARM 将通过发送一个 MCS-Channel-Join 请求加入这个信道。然后该 ARM 通过发出一个具有表 3 中规定的参数的 GCC-Application-Enroll 请求,在该会话中注册非激活。

获得那些需要标识出来作为专用会话登记处理之一部分的资源集的责任由创建该会话的 ARM 承担。如果任何附加信道、登记参数或者令牌是由协议规范来托管的,那么 ARM 应当在必要时利用 GCC 来对其创建和登记。用于登记这些资源的登记键应包含作为会话键的一部分的会话 ID。注意到允许专用会话的成员确定每个信道的功能可以不必登记各专用信道。如果应用协议委托召集者向会话成员发出 MCS-Channel-Admit 请求原语的次序,该信息可能被更有效地运送。因而,专用信道的登记并不是预先受阻止的。

一旦所有的资源已被成功地分配完,ARM 就应通过发出具有表 3 中规定的参数的 GCC-Application-Enroll 请求来注册会话中的激活。

接下来通过发出一个 GCC-Application-Invoke 请求,ARM 应规定被邀请或 NULL(表示会议中的所有节点均被邀请)的 GCC 节点 ID 清单作为目的地节点参数,来试图邀请其他节点参加到该专用会话中。ARM 还规定了动态专用信道作为应用协议登录中的启动信道。

这时召集会议的 ARM 必须等待,直到它收到了包含被邀请到专用会话的各个 APE 的 MCS 用户 ID 的 GCC-Application-Roster-Report 指示为止。应预计到 APE 在一个或多个被邀请的节点可以不被激活。接下来该 ARM 应为启动信道发出 MCS-Channel-Admit 请求,将被邀请的各 APE 的 MCS 用户 ID 规定作为 MCS 用户 ID 清单。被邀请者的清单由用户应用提供。然后根据协议的规定,对任何附加信道均重复上述过程。



注：将 GCC-Application-Invoke 指示送到节点控制器

图 13 专用会话协议初始化顺序示例

### 6.3 形成登记处键

为了确定经由 GCC 登记处的动态令牌或信道的标识,ARM 必须形成一个包含当前会话的会话键和资源 ID 的登记处键。资源 ID 是一个 ASN.1 OCTETC STRING,它描述需要 ID 的特定信道或者令牌的。信道和令牌的资源 ID 的结构将在应用协议规范中规定。

### 6.4 能力协商

APE 使用了应用注册机制进行能力协商。GCC-Application-Enroll 请求原语的应用能力清单参数被用于规定由本地 APE 支持的能力清单。

包含在 GCC-Application-Enroll 请求中的应用能力清单由通告作为 APE 所支持的那些能力清单组成。能力协商过程的结果在 ARM 的表面上由从 GCC 提供者接收到的 GCC-Application-Roster-Report 指示所产生的。应用名单报告包括在指定会议中用于对等端 APE 的应用名单,即,各个 APE 已经指定了相同的会话键,应用名单包括用于对等 APE 已注册的节点的清单。对于每个节点,该清单包含那个节点的 GCC 用户 ID,以及那个节点对等 APE 的应用用户 ID。应用名单还包含一个事例编号、指示从上次事例以来新节点是否加入的标志、指示从上次事例以来节点是否是移动的标志、指示从上次事件以来应用能力清单是否已更新的标志,以及因此产生新应用能力清单。在最新登记 APE 的情况下,由于清单的先前事例对该 APE 是无效的,因此应用能力清单总是要更新。

在首次注册时,ARM 应忽略接收到的不包含 APE 的(即在应用名单中没有登录,该应用名单中有本地节点的 GCC 用户 ID 以及 GCC-Application-Enroll 证实中指定的本地 APE 的应用协议实体 ID) GCC-Application-Roster-Report 指示。一旦收到了包含本地 APE 的名单,该 APE 当时就被认为是会议的一部分,并且可以着手检查名单,以确定如何进行。

接收到作为 GCC-Application-Roster-Report 指示一部分的应用能力清单是与所有注册的对等 APE 的可拆装的应用能力清单相对应的。也就是说,该清单包括任何对等 APE 发出的每个能力的登录项。对于每个登录项,它包括能力 ID,已通告作为它们登记程序的一部分的该能力的对等 APE 的数量(包括本地的 APE)、以及在 MIN 或 MAX 类别能力的情况下,在通告该能力的所有对等端 APE 中参数的最小或最大值。对于每个能力,用于确定能力交换结果的规则是由应用协议来规定的。

APE 在会议中注册的任何时候,该 APE 可以从 GCC 提供者收到附加的 GCC-Application-Roster-Report 指示,表示名单的内容已经改变。这可能是由于会议进行中注册了新的对等 APE 或对等 APE 离开了会议或者对等 APE 已经修改了它们的注册信息。

在某个时候,如果本地 APE 需要指明它的应用能力清单中发生了变化,它就可以重新登记。其作法是向 GCC 提供者重新发出将注册/未注册标志设置成注册的 GCC-Application-Enroll 请求,并更新应用能力清单,同时,通常在登记请求中包含其他参数。这样做的结果是应用名单发生潜在的变化,使得会议中的本地 APE 以及所有其他对等 APE 接收到一个 GCC-Application-Roster-Report 指示。

### 6.5 离开会话

希望离开会话的用户应用将有关信号送给 ARM。在从会话中解除注册之前,ARM 负责删除它可能已创建的任何 GCC 登记登录项。假设会话中的其他参加者不再需要它们。确定这个的方法由应用协议来规定。ARM 通过发出一个将注册/未注册标志设置成未注册同时不需要其他参数的 GCC-Application-Enroll 请求而离开会话。

如果在某个时刻,ARM 接收到了其中不再包含它(即它的实体 ID 不存在)的 GCC-Application-Roster-Report 指示,ARM 应立即发出一个 MCS-Detach-User 请求,以便从特定的会议中拆连。此时不再认为 APE 在会议中注册,但是可以通过重复其初始化步骤而试图在会议中重新注册该 APE。

在某个时刻,如果 ARM 接收到了 GCC-Application-Permission-To-Enroll 指示,其授予/撤消标志设置成撤消,那么它应立即发送一个 MCS-Detach-User 请求,以便从特定的会议中拆连。此时不再认为 APE 在会议中登记,并且不应试图重新注册。

如果 ARM 是专用会话的一个成员,并且收到的会话中不再包含会话创建者(即,它的实体 ID 不存在)的 GCC-Application-Roster-Report 指示,则认为该会话已停止存在,ARM 应立即发出一个 GCC-Application-Enroll 请求,将注册/未注册标志设置成未注册,以便从特定的会话中拆除连接。不需要其

他参数。

## 7 MCS 资源和 GCC 服务综述

本章提供了使用 MCS 资源(信道和令牌)和 GCC 登记处的综述。这些主题的全部处理在建议 T.122、T.125 和 T.124 中给出。注意完成会话资源初始设置的责任由会话创建者承担,如果协议规范中允许这样的话,任何一个会话成员都可以在任何时间需要附加的资源。

应用服务元素(ASE)应这样设计,它们的操作尽可能与可用于会话的资源类型(静态或动态)无关。

### 7.1 信道

MCS 信道用于在一个会议中分发数据。MCS 信道被分成两种类型:静态和动态,后一类有 3 种变体,即用户 ID 信道、分配的信道和专用信道(见 T.122 § 8.4)。ARM 通过分别使用信道加入和信道离开服务而加入和离开其用户应用所需的信道。

#### 7.1.1 静态信道

这是一种具有范围为 1~1 000MCS 信道 ID 的信道。静态信道被标准化的应用协议(例如 T.126 和 T.127)使用。静态信道被分配给 T.120 中它们相应的协议,每个静态信道有一个在相应的应用协议建议中所描述的特定目的。一个标准化的应用协议可以使用一个或多个静态信道。对于每个标准化的应用协议,其中的一个信道应被设计成起始信道;这个信道的 MCS 信道 ID 被用作协议的标准基础会话的会话 ID。

静态信道只可以用在相应的标准化应用协议的标准基础会话中。典型的情况是参加这个会话的所有 APE 加入静态信道,因而使用它们向该会话广播数据。

ARM 应检查资源清单参数,以确定它是否需要加入任何的静态信道。对于资源清单中标识的每个静态信道,ARM 应发出一个规定的适合信道 ID 的 MCS-Channel-Join 请求。如果成功的话,ARM 应向它的 ASE 传送信道 ID。

#### 7.1.2 动态信道

这是一种具有范围为 1 001~65 535MCS 信道 ID 的信道。动态信道形成可用于一个会议内所有会话的信道资源库。动态信道的功能不是预定的,而是由 APE 在发出了一个动态信道请求时确定的。动态信道有 3 种变体:用户 ID 信道、分配的信道和专用信道。

#### 7.1.3 用户 ID 信道

这是一种由单个的 APE 加入的 MCS 信道。一个 MCS 用户 ID 信道唯一地标识一个 APE,而且只可用于发送到该 APE 的数据。当 APE 进行注册激活的时候,就需要用户 ID。为了获得唯一的用户 ID,ARM 应当向 MCS 提供者发出一个 MCS-Attach-User 请求,使用包含在 GCC-Application-Permission-To-Enroll 指示中的会议 ID 作为域选择符。当响应时,接收到了成功的 MCS-Attach-User 证实,ARM 就应通过发出一个 MCS-Channel-Join 请求而加入到指定用户 ID 信道。

#### 7.1.4 分配的信道

这是一种可用于向一个会话内的所有与会者广播数据的 MCS 信道。加入到这些信道是没有规律的,即任何 APE 可以在任何时间加入或者离开一个分配的信道。如果所有的 APE 离开了一个分配的信道,那么它就不再存在,它的信道 ID 可以重新分配作其他的用途,这是一个潜在的缺陷。见图 15。

分配的信道可以用于标准基础会话、非标准基础会话或者公开会话。建议在专用会话中不使用分配的信道,这是因为如果知道了分配的信道的存在,那么任何 APE 都能加入该信道。

公开会话或非标准基础会话的起始信道是一个分配的信道。

如果在已经建立了一个会话之后需要另外的分配的信道,ARM 则应通过发出一个具有 ID=0 的 MCS-Channel-Join 请求原语首先试图创建一个新的分配的信道。如果成功的话,返回的 MCS-Channel-Join 证实包含了用户分配的信道的分配信道 ID。ARM 可以通过发出一个具有表 4 中给出参数的 GCC-Registry-Register-Channel 请求而试图登记该分配的信道。在接收到相应的 GCC-Registry-Register-Channel 证实之后,ARM 应检查结果参数。如果是成功,该 ARM 就应将其分配的信道 ID 的 ASE 通告给该分配的信道。如果结果是“索引已经存在”,就表明另一个 ARM 已经用相同的登记处键登记了一个



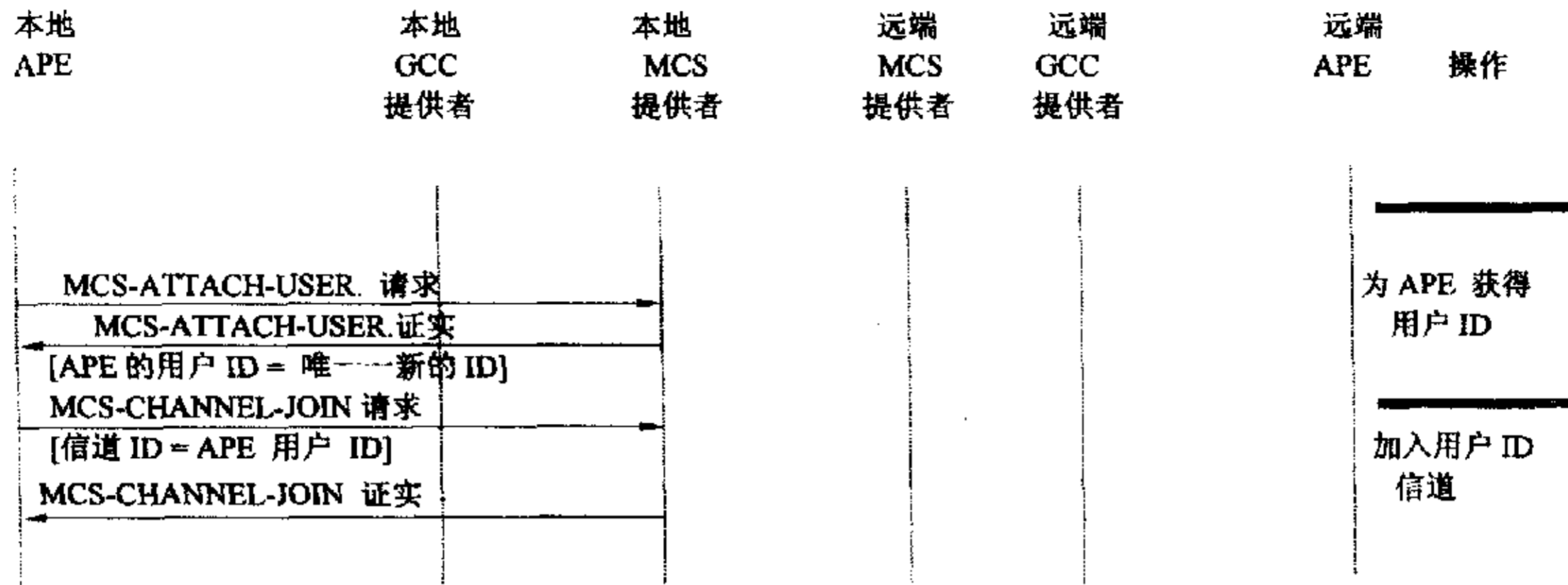


图 14 获得一个 MCS 用户 ID 信道

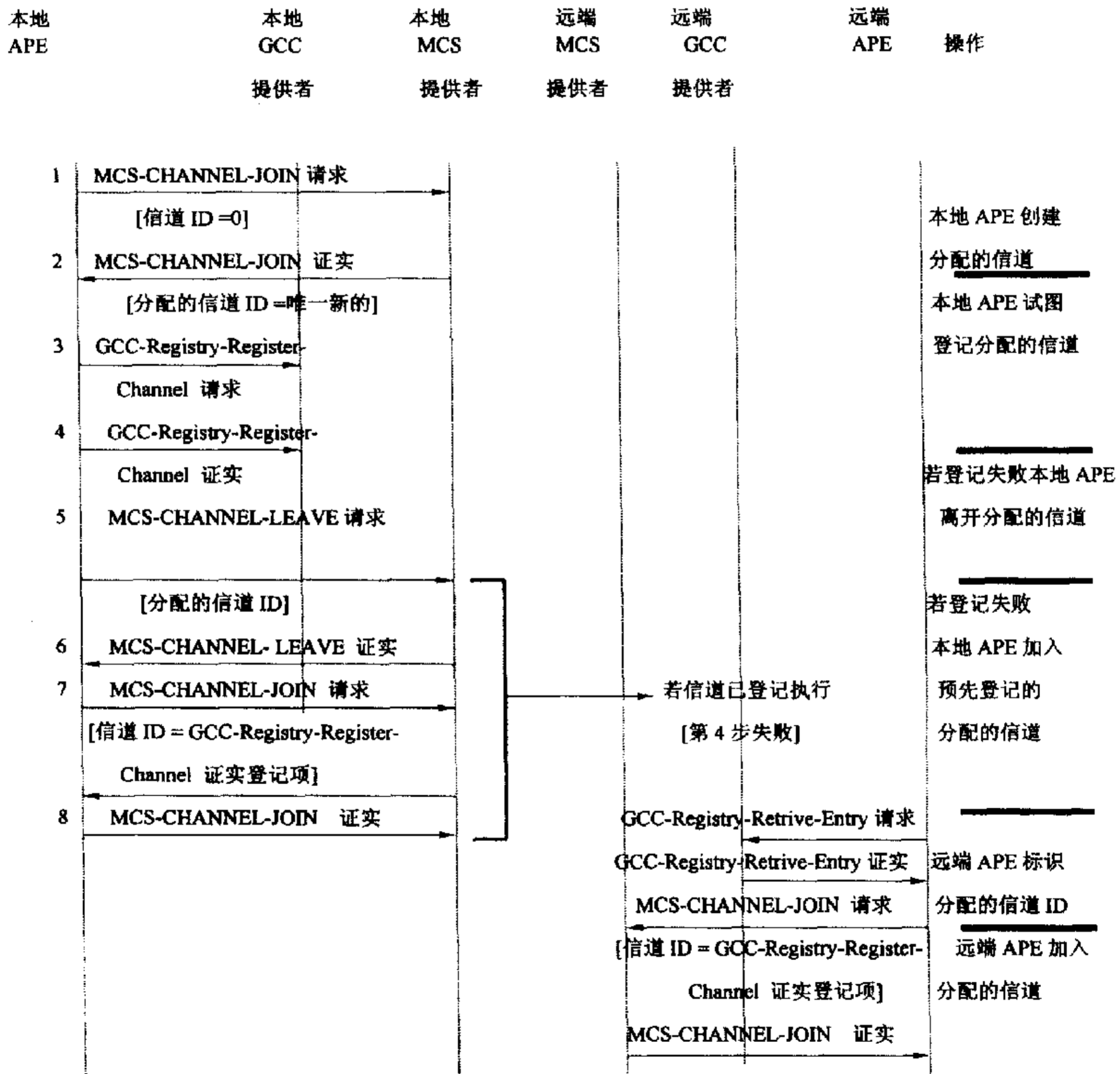


图 15 创建和登记一个分配的信道

分配的信道。在这种情况下,该 ARM 应通过发出一个 MCS-Channel-Leave 请求而离开它先前已经创建的这个信道。然后它应发出一个 MCS-Channel-Join 请求试图加入由另一个 ARM 登记的分配的信道,同时,规定把在 GCC-Registry-Register-Channel 证实的登记处项参数中返回的信道 ID 作为信道 ID。如果继续,应向它的 ASE 传送资源 ID 和相应的信道 ID。如果上述尝试失败,而且结果是“没有这样的信道”,ARM 则可以重新尝试信道创建过程。在所有其他情况下,ARM 应向 ASE 表明它不能分配信道了,见表 5。

表 4 用于 GCC-Registry-Register-Channel 请求的参数

参 数	内 容
会议 ID	由 GCC-Application Permission-To-Enroll 指示提供
登记处键	如 6.3 中描述所形成的注册键
信道 ID	在 MCS-Channel-Join 证实中返回的信道 ID

表 5 用于 GCC-Registry-Pertrive-Entry 请求的参数

参 数	内 容
会议 ID	由 GCC-Application-Permission-To-Enroll 指示提供
登记处键	如 6.3 中描述所形成的登记处键

### 7.1.5 专用信道

专用信道是一个具有一组授权用户的 MCS 信道。它可以用来向一个会话的与会者的选定子集分发数据。加入到一个专用信道是由召集该信道的 APE 来控制的。该 APE 被命名为信道管理者并具有下列特权。

- a) 允许其他 APE 加入到专用信道的特权;
- b) 从专用信道中驱逐其他 APE 的特权;
- c) 解散专用信道的特权。

专用信道可以用于标准基础会话、非标准基础会话、公开会话或者专用会话中。

专用会话的起始信道是一个专用信道。

如果召集用户应用希望从专用信道的成员中去掉一个或多个 APE,那么召集者 ARM 应发出一个 MCS-Channel-Expel 请求,规定要驱逐的 ASE 的 MCS 用户 ID 作为 MCS 用户 ID 的清单。

一旦在召集节点的用户应用指出它已没有更多的信道用作专用信道,召集者 ARM 将通过发出一个 MCS-Channel-Disband 请求,规定专用信道作为信道 ID 而解散该信道。当前加入到该专用信道的所有 ARM 这时接收一个 MCS-Channel-Expel 指示,通知它们专用信道不再能够有效使用。

如果在已经建立了一个会话之后需要附加的专用信道,ARM 则应遵循为召集者或者合适的成员所描述的程序进行,见图 16。

### 7.1.6 使用 MCS 信道的潜在问题

a) 当 APE 加入或者离开一个信道的时候,MCS 不向信道的各个成员提供任何指示。这在专用信道的使用中是个特殊的问题,这是因为专用信道管理者还没有证实被邀请的 APE 是否已经加入到该专用信道。对于各个 APE,一个可能的工作任务(如在 T.127 中实施的那样)就是通过向信道管理者的用户 ID 信道发送一个 PDU 以证实它们已经加入了专用信道;

b) 由于全部加入的用户都离开时这些分配的信道不再存在,建议在使用分配的信道期间要谨慎。当这种情况出现时没有提供指示。既然 MCS 能够重新使用动态信道,APE 就不应试图去使用一个分配的信道,除非确定了它的存在和当前使用情况。用于分配信道的 GCC 应用登记处中的一个登录项的存在并不确保该信道存在或者由登记处所指明的使用仍然有效;

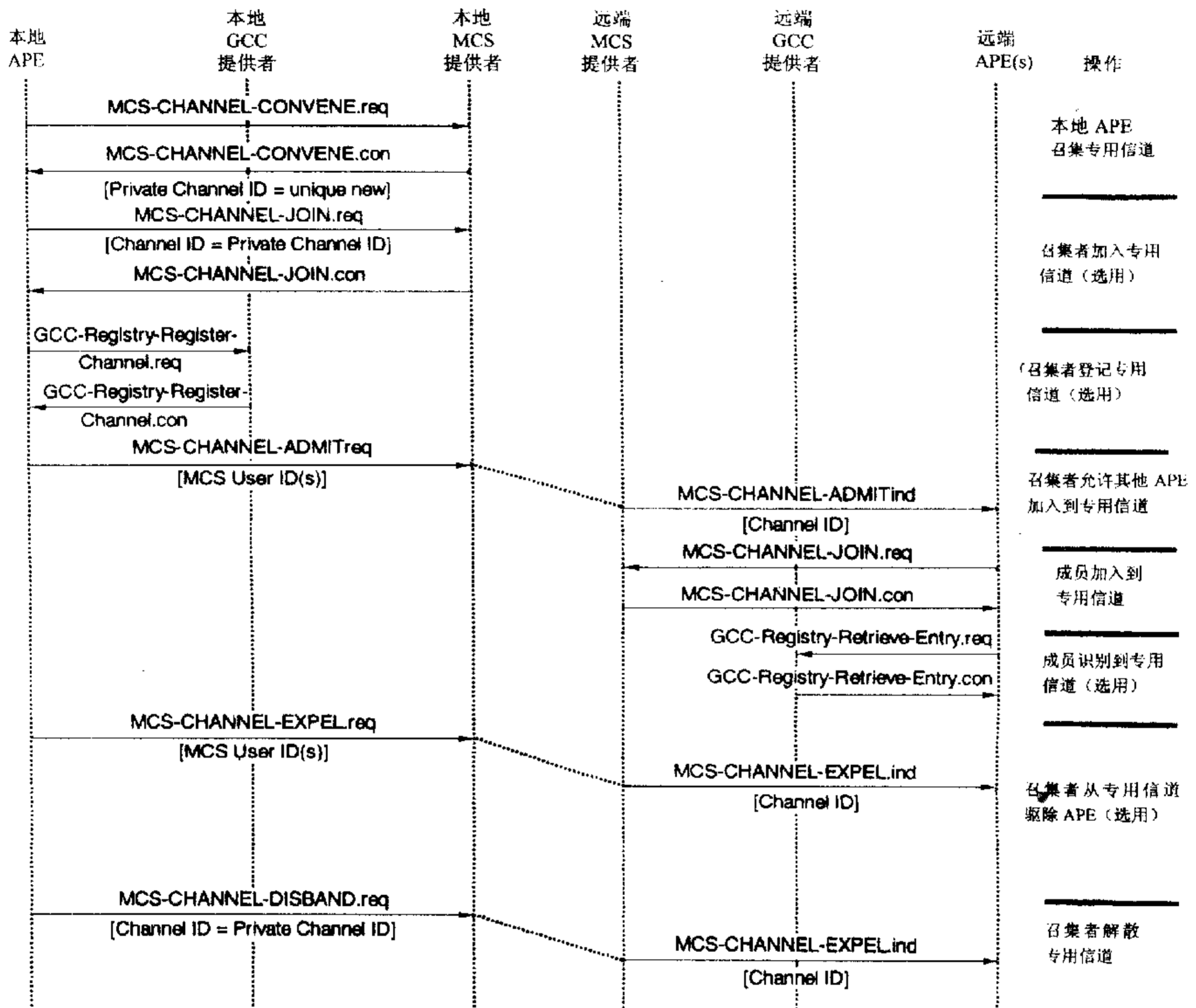


图 16 专用信道的管理

c) 由于在传送数据之前 MCS 可能驱逐了一些信道成员,当信道管理者离开、解散或者驱逐信道中的成员时,专用信道上处于传送中的数据可能没有到达它们设定的目的地。MCS 令牌禁止服务的使用允许信道成员向信道管理者发送事务处理完成的信号,从而使信道管理者能避免由于信道成员的过早驱逐而造成偶然的数据损失。在事务处理可能开始并且必须释放事务处理完成的那个令牌之前,每个信道成员需要禁止一个指定的令牌。应用协议必须确保在传送开始之前所有的信道成员已经禁止了相关的令牌。另一方面,应用协议可以在信道管理者和各个信道成员之间规定一个 PDU 交换,以确保在专用信道解散之前所有的数据都被传送了;

d) 如果在这种数据传送之前发送方 APE 拆连,则信道中处于传送之中的数据就不能到达它们设定的目的地。通过使用 c) 中描述的补救方法,发送 APE 可以避免这一点。

### 7.1.7 小结

标准化的应用协议的标准基础会话使用静态信道广播数据。

每个激活的 APE 需要一个用户 ID 信道。该信道可以用来仅向那个 APE 发送数据。

分配的信道用于在任一会话中广播数据。

专用信道用于在专用会话中广播数据。它们也可以在有选择性地分发数据的任何会话中使用。

对于一般使用,可用于非标准应用协议的信道的选择是在分配的信道和专用信道之间进行的。本小节对每种信道的优点进行比较,以便于应用协议开发者作出适当的选择,见表 6。

## 专用信道

- 由信道管理者确定的,已知信道工作的时间;
- 有规则地加入;
- 信道管理者离开会话时不再存在。

## 分配的信道

- 不知道信道工作的时间;
- 无规则地加入;
- 创建者已离开会话之后仍然能存在。

表 6 MCS 信道的使用

信道类型	会话类型			
	标准基础	非标准基础	公开	专用
静态	✓	?	?	?
用户 ID	✓	✓	✓	✓
分配的	✓	✓	✓	*
专用的	✓	✓	✓	✓
注 ✓ 允许的 ? 不允许的 * 不建议的				

## 7.2 数据传送

有两种可用于分发数据的机制,即发送数据和一致排序发送数据。两种技术都向加入到一个 MCS 信道的全部 APE 传送数据。

“发送数据”经过最短的可能路由向加入到某个 MCS 信道的每个 APE 提供数据传送,因此它具有最小的网络业务量和等待时间。在每个 APE 上接收数据的次序可以不同,并且数据的发送者不会收到该数据的复本。

“一致排序发送数据”提供数据的顺序传递,即它确保加入到某个 MCS 信道的每个 APE 按照相同的次序接收数据。如果数据的发送者加入到了该信道,它则会收到该数据的复本。

为了接收 MCS 信道上的数据,APE 必须加入到那个信道。APE 可以向任何 MCS 信道发送数据,而不考虑它是否加入了那个信道,但专用信道是例外,APE 在试图向专用信道发送数据之前必须经许可加入到该信道。

## 7.3 令牌

令牌提供在一个会议中实现唯一的接入资源的方法。例如,为了避免不同用户向一个设备(如摄像机)发送相冲突的命令,需要严格限制在任意给定的情况下每一个用户对那个设备的接入。这可以通过如下方法来实现,将令牌和设备结合起来并委托相应的应用协议,只有当前令牌持有者才被许可向该设备发出控制命令。

令牌的管理操作是由一定数量的 MCS 服务来支持的。

令牌捕获服务允许一个用户唯一地持有一个特定的令牌。各个用户可以在任何时候使用令牌测试服务以确定令牌的状态,并可以用令牌请求服务从持有者那里请求该令牌。该令牌持有者可以用令牌给出服务向另一个特定的用户传递令牌的控制或者通过令牌释放服务将令牌返回到一般可用的状态。令牌释放和令牌给出服务允许令牌在两个用户之间传递,而不可能被第三个用户捕获。

令牌还可以通过令牌禁止服务用于在多个用户间进行事件协调。各个用户可以独立地禁止和释放

同一个令牌。例如,如果它需要知道什么时候全部用户都已完成了接收并处理了大宗的文件传送,全部用户则会在开始操作时都禁止同一个令牌,而且在每个单独的用户已完成该过程之后要释放该令牌。为了避免竞争情况,有必要确保在检测令牌是否已变成未禁止之前,全部用户均已禁止了该令牌。任何用户都可以根据意愿检测该令牌,以确定它是否为自由的,意味着全部用户都完成了上述处理。

令牌可以是静态的也可以是动态的。MCS 对所有令牌进行同样的处理,静态和动态令牌的区别是在 GCC 中产生的,GCC 使用登记处来分配动态令牌。全部令牌操作是由 ASE 使用 MCS 令牌服务来完成的。

### 7.3.1 静态令牌

静态令牌是具有范围在 1~16 383 内的 MCS 令牌 ID 的令牌。静态令牌被标准化应用协议(例如 T.126 和 T.127)使用。静态令牌被分配到建议 T.120 中它们相应的协议,按照相应的应用协议建议中的描述,每个静态令牌有一个特定目的。一个标准化的应用协议可以使用一个或者多个静态令牌。

静态令牌只可以用在相应的标准化应用协议的标准基础会话中。

ARM 不涉及静态令牌的使用。

### 7.3.2 动态令牌

动态令牌是一种具有范围在 16 384~65 535 内的 MCS 令牌 ID 的令牌。动态令牌形成一个会议中可用于所有会话的令牌资源库。动态令牌的功能不是预定的,而是在 GCC 分配时确定的。动态令牌可以用在任意种类的会话中,见图 17。

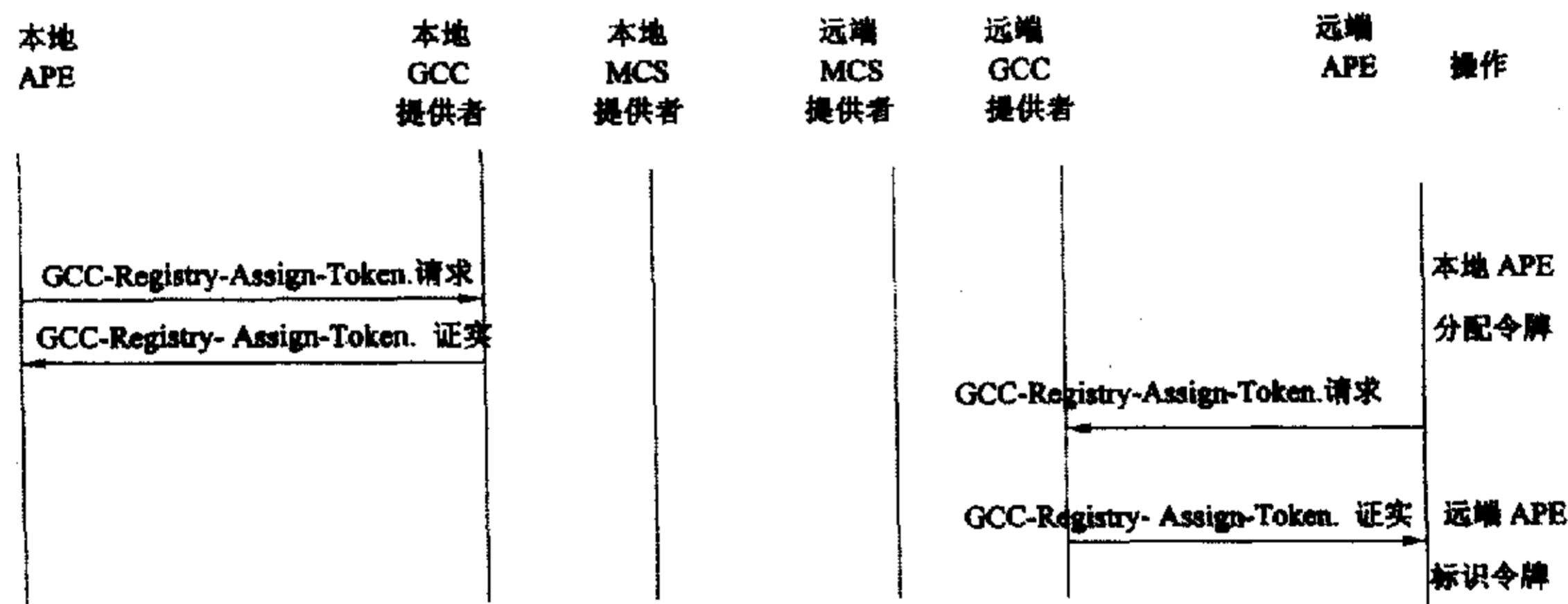


图 17 分配和标识动态令牌

ARM 应检查资源清单参数,以确定它是否需要标识用于任何动态令牌的令牌 ID。不管 ARM 是会话召集者还是会话成员,可以使用同样的方法。对于资源清单中规定的每个令牌资源 ID,ARM 应使用表 7 中规定的参数向 GCC 提供者发出一个 GCC-Registry-Assign-Token 请求。如果该 GCC-Registry-Assign-Token 证实返回的结果参数是“成功的”或者“索引已经存在”,那么包含在返回的证实原语的登记处项中的令牌 ID 就被用作登记键中所使用的相应于资源 ID 的令牌的令牌 ID。ARM 应向其 ASE 传递资源 ID 和相关联的令牌 ID。

注意在作出了请求之前 GCC-Registry-Assign-Token 返回了令牌 ID(不考虑令牌是否存在),就没有必要使用 GCC-Registry-Retrieve-Entry 来标识令牌。

遵循由有关的应用协议规范所规定的限制,会话期间所需的附加动态令牌,可以由该会话中的任意参与者分配。

表 7 用于 GCC-Registry-Assign-Token 请求的参数

参 数	内 容
会议 ID	由 GCC-Application-Permission-To-Enroll 指示提供
登记处键	如 6.3 中的描述形成的登记处键

#### 7.4 应用名单

ARM 负责监视 GCC-Application-Roster-Report 指示,并通告其 ASE 的任何变化。应用名单登录项报告包含了与在其中无效注册 APE 的应用协议会话相对应的部分名单。在没有会话 ID 而注册的 APE 非激活的情况下,应用名单报告包含具有相同应用协议键的全部应用键相对应的部分名单。

当在会议中注册 APE 的任何时候,它都可以接收附加的 GCC-Application-Roster-Report 指示,将名单内容中的变化通告它。这可能是由于新的对等 APE 在会议中注册、对等 APE 离开会议或者对等 APE 修改了它们的注册信息。

一个用户应用可以通过指示合适的 ARM 重新注册而在任何时间改变它的应用能力清单。该 ARM 然后发出一个将注册/未注册标志设置成注册的 GCC-Application-Enroll 请求,同时包括修正的应用能力清单,并按照包含在初始激活的注册提供所有其他参数。这可能导致可拆装的能力集的变化,在这种情况下,会话中的全部 APE 会接收到一个 GCC-Application-Roster-Report 指示。

对于出现于变换时刻的任何事务处理可拆装的激活的能力集中任何变化的影响,将由各自的应用协议予以规定。

#### 7.5 应用登记处

应用登记处是在顶 GCC 提供者处所保持的一个集中化的数据库,可以用它来允许各个 APE 独立地确定要用于特定应用协议会话的各个资源(例如动态信道和令牌)。登记处内容的意义由各自的应用协议来规定。

应用登记处提供以下方面的机制。

- 登记信道;
- 分配令牌;
- 全范围的分配唯一的句柄,或者单独地分配或者作为一个句柄块进行分配;
- 建立用于会话的公共参数;
- 改变并删除登记处登录项;
- 收回登记处登录项;
- 监视登记处登录项产生的变化(包括登录项的删除)。

利用标识与该登录项相关联的应用协议会话的唯一键来标识每个登录项。该键还包含由应用协议定义的一个资源 ID 以描述该会话中登录项的使用。

每个登记处登录项有一个指明的拥有者,即创建该登录项的 APE。如果该登录项的拥有者离开了会议,那个登录项就成了不被拥有的。被拥有的登录项只能由它们各自的拥有者删除,不被拥有的登录项可以被任意的 APE 删除。信道和令牌登录项不能被修改。被拥有的参数登录项可以由它的各自的拥有者修改。无拥有者的参数登录项可以由任意一个 APE 修改,这个 APE 就成了该登录项的新拥有者。登录项要维持会话的整个过程,而且登录项拥有者必须注意以保证在不再需要时删除之。这一点在登记会话的情况下尤其正确,在那时,除非被拥有者明确地删除多余的登录项很可能会无限制地存留下去。注意在某些情况下为了离开 APE 以删除它们创建了的登录项,可能是不合适的或者不可能的。

APE 与应用登记项之间的全部通信是由 ARM 来完成的。

#### 7.6 组织权

当会议登记或者离开组织模式时,ARM 负责通告它的 ASE 和用户应用以及组织模式中的任何变化。组织模式中的其他操作由用户应用或者 ASE 来直接完成。

用户应用应经由其本地节点控制器向它的本地节点发出请求,以成为 GCC 组织者、传送组织权或者放弃组织权。

会话组织者是,在组织节点处的 APE,在那个会话的应用名单中已设置了组织操作标志。在组织节点的节点控制器在 APE 缺席时代理会话组织者。

如果 APE 担当了会话组织者的角度,其 ASE 应仲裁来自起作用的对等 ASE 请求。这样一些请求及其相关联的响应将要利用应用协议规定的 PDU 来发出。这些可以用来指定会话中每个 ASE 的不同特权。

如果节点控制器成为会话组织者,它只可以由节点基授予或者拒绝在一个节点上起作用的许可。这给出了许多 ASE 的方向控制。ASE 可以经由它的本地节点控制器请求许可起作用,反之发出一个 GCC-Conductor-Permission-Ask。如果组织者节点控制器发出了一个在“节点授予许可清单”中包括请求节点的 GCC-Conductor-Permission-Grant,那么在那个节点的全部 ASE 可以起作用。

组织权的详细实施由每个应用协议规范来规定。

### 7.7 远端调用

如果一个用户应用希望动态调用在另一个节点处的对等应用并邀请它加入该应用会话,那么用户应用应经由 ARM 来做到这些。该用户应用必须提供对于每个应用协议希望用到的下列信息。

- 会话键;
- 启动信道类型(静态和动态组播、动态专用或动态用户标识符);
- 最小能力集;
- 要邀请到应用会话的节点清单。

这时 ARM 发出一个 GCC-Application-Invoke 请求,规定被邀请节点的 GCC 节点 ID 清单、启动信道类型、注册其中的应用协议会话的清单以及对每个应用协议期望的(即最小的)能力集。

GCC-Application-Invoke 指示被传送到负责确定激活哪个 APE 的节点控制器。

## 8 用户应用指南

本章提供了使用 T. 120 基础结构的指南,以保证始终不变的和可预计的应用行为。

### 8.1 用户应用的标识

在某些环境下,为能确定哪些节点主持一个特定的用户应用或者在一个远端节点处可得到的用户应用的范围,用户应用的标识可能是有用的。可能需要它用于本地的和远端的用户应用的调用。APE 可以通过在 GCC-Application-Enroll 请求中包含一个非标准的不可拆装的能力参数,作为可选项标识出它的用户应用。通过经由其 APE 咨询应用名单,一个用户应用可以因此检测在其他节点处存在一个特殊的用户应用。接收 GCC-Application-Invoke 指示的节点控制器可以选择咨询用于合适的会话应用名单中的用户应用能力登录项,以确定在本地激活一个用户应用之前在其他节点处哪个用户应用是激活的。

注:如果知道由哪个应用通告的不可拆装的能力参数,那么节点控制器将只能检测某个给定的用户应用的存在。

非标准应用协议键只能用作通告支持非标准应用协议。由于它阻止了与使用相同应用协议用于相同目的的其他用户应用互操作,不允许用通告一个特定的用户应用的存在注册一个非标准应用协议键。

### 8.2 可拆装和不可拆装的能力的使用

一个 APE 可以在 GCC-Application-Enroll 请求中使用不可拆装的能力范畴,以标识它的用户应用并通告它的应用协议特定能力。

为了保证互操作性,如果需要标识哪个特定的对等 APE 支持某个给定的能力,在 GCC-Application-Enroll 请求中使用不可拆装的能力参数是合适的。另外,它可以用来向会议通告资源(例如可用于远端控制的摄像机)的有效性。如果 APE 只需要确定那些支持特定能力的对等 APE 的数量,那么它应使用可拆装的能力。为了避免使不必要的应用名单过大,一定要建议将不可拆装的能力的数量保持在绝对最小的程度。

### 8.3 会话上下文

如果一个会议有特定应用协议的多个会话,有必要区分那些会话,以便使用户能够选择参加最合适的会话。典型情况是,一个会话可能在该会话的参与者所确定的指定内容中使用。例如,一个 T. 126 白板会话可以用于提出的一个主题,另一个可以用于提出的第二个主题。这些会话的不同主题允许用户有区别地去参加。

GCC 应用登记处参数可以作为选项,用来描述一个特定应用协议会话的内容。资源 ID(根据建议 T. 50 由“CONTEXT”这个字的 7 个字母编码成连续的八位组串而构成)用于此目的。登记处参数的内容是一个用户可读的位串,运用分组编码规则(基本一致变体)将该可读位串编码成受限的 BMP 串。生

成的编码位串按顺序放在 OCTECT STRING 中,在这个顺序中对每个八位组串,先导位放在最高特征位的位置,而结尾位放在最低特征位的位置。

#### 8.4 选择参与一个会话

如果一个用户仅希望与用于支持相同目的的不同应用协议的任何节点进行通信,那么最合适的会话是。

——用于标准化应用协议的标准基础会话(见 8.3);

——用于非标准化应用协议的非标准基础会话。

基础会话预期用于在支持特定协议的全部终端之间进行一般目的的通信(例如 T.127 标准基础会话应当用作一般目的的文件传送)。

注: APE 可以在一个基础会话中安排注册而无需用户介入,因此允许基于标准协议或非标准协议的用户应用自动地开始加入到一个会议中。

如果一个给定的应用协议有多个激活的会话出现在一个会议中,那么可能需要用户介入。该用户需要足够多的信息,以便选择参加最合适的会话,或者决定是否有必要建立一个新的会话。

这样的决定可能是基于如下的一个或几个标准。

- a) 会话内容(例如讨论的主题);
- b) 参与会话中的哪些节点;
- c) 会话的协商能力集。

#### 8.5 控制会话的使用

会话的创建者可能希望对一个会话加以限制,以确保较小能力的 APE 不会在会话中偶然地注册并有可能将会话能力降低到会话创建者认为是不可接受的程度。如果一个用户希望对一个会话施加最小的能力集,那么应该使用动态专用会话。会话创建者规定 GCC-Application-Invoke 请求中要求的能力准则。

如果一个用户希望指出用于特定内容的会话,那么应当使用动态会话,且会话内容(用户可读的文本串)应在用于该会话的 GCC 应用登记处中提供。

#### 8.6 添加新的与会者到一个会话

专用会话的召集者可以使用 GCC-Application-Invoke 请求原语,以便邀请另外的节点(例如后面的与会者)在任何时间参加该会话。标准基础会话、非标准基础会话或者公开会话中的任何与会者可以利用 GCC-Application-Invoke 请求在任何时间邀请新的节点到相应的会话中。如果在标识的会话中的那个节点上已经有一个激活的 APE,那么接收到 GCC-Application-Invoke 指示的节点控制器可以选择不理睬。

#### 8.7 应用协议的扩展、修改和重新利用

对于一个特定的用途,应用协议是可以被扩展、修改或剪切的工具。为了确定这些改变之后产生了新的独有的协议还是仅仅扩展了现有协议,有必要决定那些符合已改变应用协议的 APE 是否打算与那些符合原有的应用协议的 APE 互通,如果打算互通,那么已修订的协议则能够被认为是现有协议的扩展,否则它就是一个新的应用协议并且必须分配一个独有的应用协议键和资源。

如果 APE 提供了对一个应用协议的专有的扩展,以致它打算能够与符合原有协议的那些 APE 互通,那么它应以原有协议的应用协议键进行注册,并按照 8.2 中规定的指南,利用可拆装或不可拆装的能力通告它的增强功能。

当应用协议用作一个独有的目的时,或者以如下方式进行修改的时候,即符合修订协议的 APE 不打算能与那些符合原有协议的 APE 互通时,该修订协议就被认为是一个新的应用协议,并且必须给它指定一个与原有协议不同的独有应用协议键。原有应用协议所使用的任何静态信道和令牌不能被修订的协议使用。如果以某种方式限制一个应用协议的使用,那么上述这些规则同样适用。例如,使用 T.127 以外的方法(PDU 及程序)传送文字处理机应用之间的文件的那些 APE,如果它们不打算与一般目的



的文件传送互通,则不应使用 T.127 应用协议进行注册,也不应使用留给建议 T.127 用的静态信道和令牌。

类似地,如果为了实现某些功能(性)的协作(例如重新利用 T.126 和 T.127 方法的事务卡应用协议)需要一些应用协议,如果各组成部分的 APE 只打算与协同完成相同目的的其他 APE 互通,就需要新的应用键。

当应用协议被组合以完成特定的功能时,更好的方法是通过符合各部分的协议来定义一个新应用协议。这就导致通过合并现已存在的 APE 而创建新 APE。可能有必要解决新 APE 使用的应用协议之间的功能 ID 和 GCC 登记处资源 ID 的冲突。

另一个方法是保留预先存在的 APE 作为独有的组成部分,每个 APE 具有一个新的应用协议键以便与它原有的协议键区别开来。但是要注意,在这种情况下有必要规定一种办法来联合各应用协议会话,这些会话协同工作以实现共同的目的,以便使那些用户应用能够确定它们的要注册的 APE 的会话的正确组合。为了避免信息的复合,建议相关联的会话清单保持在多个会话中的一个(称之为根会话)的应用登记处中。由应用协议开发者来确定哪个应用协议将用作根会话。最合适的一个选择是功能(性)最接近于与预期的共同目的相匹配的这个应用协议。

---