

IMS 客户端技术标准及软件特性分析

IMS 是基于 SIP (session initiation protocol, 会话起始协议) 的系统, 它为多媒体服务提供了一整套标准体系架构。作为日趋成熟的标准体系, IETF、3GPP、OMA (open mobile alliance, 开放移动联盟) 等国际标准组织都在定义和完善 IMS 标准。IMS 技术允许运营商能更好地控制业务层, 能更快地集成和开展 IMS 多媒体服务, 并减少网络投资和运营开销, 所以运营商都很重视 IMS 技术。同时, IMS 技术也能给用户带来统一的用户体验, 用户将会获得更多质量和安全都有保障的 IMS 服务。IMS 的提出, 顺应了通信网络技术融合与业务融合发展的趋势, 它将在未来通信网络中发挥重要作用。

当前的 IMS 技术工作主要集中在探讨 IMS 网络上, 而忽视了对 IMS 客户端的研究, 然而, IMS 客户端才是最终用户享受 IMS 技术带来的诸多成果的最直接的表现方式。当前还没有统一的对 IMS 客户端的定义, 但根据作者的理解, 可以将 IMS 客户端定义为一个软件包 (包括了驱动程序、协议栈、各种引擎、应用程序、人机界面等), 并可运行在多种终端上 (如移动终端、固定终端、PDA、台式机、笔记本电脑等), 可在 IMS 网络架构下提供多种实时与非实时 IMS 业务 (如 VoIP、视频电话、呈现、即时消息、会议、组管理、一键通、协同工作、文档共享等) 和统一的用户体验, 并且符合 IETF、3GPP、OMA、JCP (Java community process, Java 标准制定组织) 等国际标准组织所规定的 IMS 相关规范。

对 IETF、3GPP、OMA 和 JCP 等国际标准组织的相关 IMS 规范的研究是开发 IMS 客户端软件的基础。通过对这些标准的研究, 便于理解相关标准之间的关系, 从而总结出 IMS 客户端的基本需求, 这将有助于描绘出 IMS 客户端的软件架构以及今后技术路线图的研究, 为将来 IMS 客户端软件开发与具体实现 做好准备工作。

2、IMS 客户端标准分析和架构参考

在此首先分析包括 IETF、3GPP、OMA 和 JCP 在内的标准组织与 IMS 客户端相关的规范, 然后基于这些研究, 给出了 IMS 客户端的软件架构参考设计图。

2.1 IETF 中 IMS 客户端相关规范

IETF 定义了一整套基础协议包括 SIP、SDP (会话描述协议)、RTP/RTCP (实时传送协议/实时控制协议)、SCTP (流控制传输协议) 和 XCAP (XML 配置接入协议) 等, 作为 IMS 客户端的基本协议簇。SIP 用于两个或者多个 IP 节点间会话的建立、维护和拆除, 可以运行在可靠的传输层 (如 TCP 和 SCTP) 上或者非可靠的传输层 (如 UDP) 上。SIP 的扩展很多, 比如 SIP 消息类型的增加 (如 Update、Refer、Publish、Notify 等)、Simple、SIP 信令压缩、用于 3GPP 的私有包头扩展、认证和安全机制等。在实现 IMS 客户端时, 这些 SIP 扩展的部分都应当有所考虑。SDP 是一种应用层协议, 用来描述媒体会话能力、媒体格式、媒体流地址和端口等信息。RTP 是用于 端到端传递实时数据的协议, RTCP 用于实时数据的服务质量监控。XCAP 允许用户上传信息到 XCAP 服务器, 通过 HTTP 更改、增加和删除存储在服务器上的 XML 文档。XCAP 复用了 HTTP 中的 Get、Put 和 Delete 方法来获取、更改/增加和删除 XML 文档。通过一套巧妙的方法, 将 XML 文档的 存储路径和文档中的条目、元素和属性映射到 HTTP 中的 URL 路径。目前, XCAP 在 IETF 中仍处于草案阶段。

SIP 及其扩展、SDP、RTP/RTCP 和 XCAP 都是实现 IMS 客户端最重要的基础协议。

2.2 3GPP 中 IMS 客户端相关规范

如图 1 所示, 3GPP 中描述的 IMS 客户端 (UE) 通过两个参考点访问 IMS 网络, 即 Gm 和 Ut 参考点, 其他 IMS 网络节点对 IMS 客户端都是不可见的。IMS 客户端通过 Gm 参考点连接到 IMS 网络, 它对应的节点是代理呼叫会话控制功能 (P-CSCF), 所有的 SIP 消息都必须经过 P-CSCF。这些 消息用于注册过程 (如 Register)、会话控制过程 (如 Invite) 和事务处理过程 (如 Message)。Ut 参考点是 IMS 客户端和应用服务器 (application server, AS) 之间的交互点。用户可以通过它安全地管理和配置存储在 AS 上与网络服务相关的信息。XCAP 可以作为该参考点的协议。

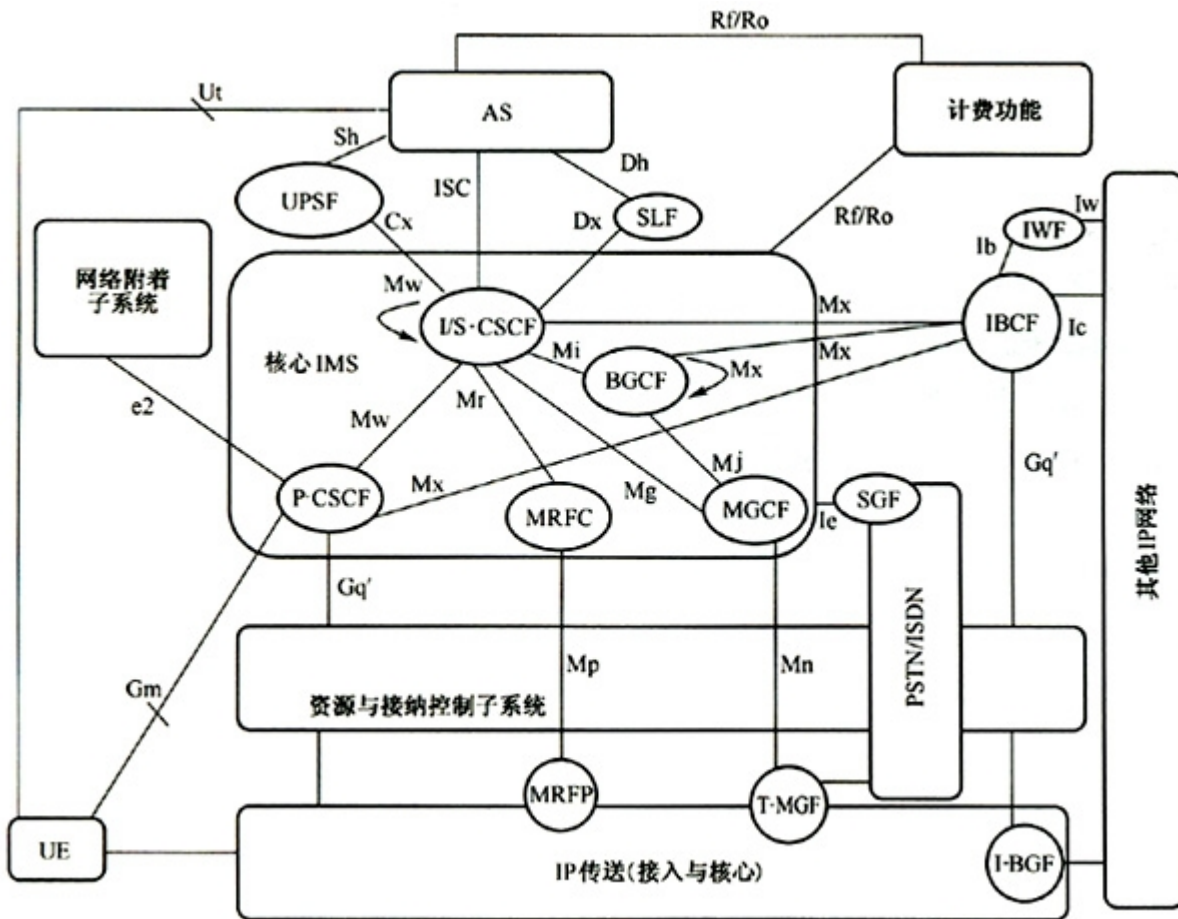


图1 3GPP 中 IMS 网络和 IMS 客户端间的接口

3GPP 中也定义了一些服务所需的 IMS 架构和功能，如呈现、即时消息、组管理、会议等服务。

2.3 OMA 中 IMS 客户端相关规范

OMA 主要定义移动服务规范，以确保运营商之间和终端之间端到端服务的互连性。OMA 提出了一系列基于 IMS 的服务应用，每种应用都包含了客户端的功能列表、协议要求、与应用服务器之间的交互等。

OMA 中呈现和可用性工作组定义了 Presence Simple 服务。呈现功能是许多 IMS 应用的基础。IMS 客户端既是呈现者也是观察者。呈现者是信息源，提供呈现信息给呈现服务器；观察者则请求获取关于呈现者的呈现信息。呈现服务器存储订阅者和产生呈现信息改变通知。呈现信息包括网络信息、用户当前的状态，也包括用户终端的能力等。有些呈现信息是网络侧提供的，如用户是否已经注册；有些是呈现者提供的，如呈现者设置的通信偏好。呈现者的状态只能被已授权的观察者看到，因此当某个观察者想订阅某个用户的状态信息时，需要呈现者的确认，呈现者有权拒绝观察者的订阅请求。观察者一般通过一个资源列表订阅一组呈现者的呈现服务，由资源列表服务器再向呈现服务器逐个订阅呈现信息，这样能够减少 IMS 客户端的负担和网络负载。在协议方面，呈现者通过 Publish 方法发布自己当前的状态，观察者通过 Subscribe 订阅呈现服务，呈现服务器通过 Notify 通知观察者其订阅用户的状态信息改变，呈现者也可以通过 Subscribe 订阅能获取其呈现信息的观察者列表。资源列表和呈现服务授权是通过 XCAP 实现的。每个资源列表和呈现服务授权都是一个单独的 XML 文档，IMS 客户端可以通过 XCAP 生成和修改这些文档。IMS 客户端需要一个友好的人机界面，同时需要实现相应的 SIP 消息类型扩展和 XCAP，才能给用户提供一个完整的呈现服务。

OMA 中消息工作组定义了 IM Simple 服务，它允许实时地交换用户之间的即时信息。IMS 中消息分为直接消息和基于会话的消息。直接消息是通过 IMS 客户端直接发送和接收消息实现的 (RFC 3428)，它适用于像短信这样单独的短消息通信。基于会话的消息是通过 Invite 发起 MSRP (message session relay protocol) 信道协商，所有消息通过 MSRP 建立的信道传送，它适合于交互式的文本会话，如聊天。IM 服务一般和呈现服务结合起来使用。通过呈现服务，用户可以将自己的好友分成不同组，并能实时地看到好友的信息。用户可以根据好友的状态发送即时消息。IMS 客户端可以实现简单的 IM 服务，如只是通过消息方法进行在线即时通信，也可以增加更复杂的功能，如聊天室、会议聊天、消息历史存储、延迟消息等功能的支持。

OMA 中移动一键通 (push to talk over cellular, PoC) 工作组定义了一键通服务。提供 PoC 服务的 IMS 客户端能实现基于分组交换、半双工的 VoIP 方案。它用 SIP 作为信令，用 RTP 传输语音数据，同时它需要复用呈现和组管理功能来实现 PoC 服务。PoC 应该是现实世界中第一个基于 IMS 的应用，因为 Presence 和 IM 应用最初是基于 XMPP (可扩展消息和呈现协议)，后来又是基于 IMPS (即时消息和呈现业务) 协议实现。

OMA 中呈现和可用性工作组还定义了 XML 文档管理服务。用户可以通过 IMS 客户端定位、存取和处

理可被其他的服务引擎所存取的用户和服务相关信息，存储和处理以 XML 文档形式保存在网络上的服务相关的数据，也可以通过 SIP 来订阅和通知文档变更。该服务集成了其他 IMS 服务中的 XML 文档管理功能。XML 文档管理功能包括：共享 XML 文档、呈现 XML 文档、资源列表 XML 文档、即时消息 XML 文档、PoC XML 文档管理等。

OMA 还成立了一个名叫融合 IP 消息的新工作组。具有这种功能的 IMS 客户端将对短信、彩信、即时消息、移动、一键通等这些传统的消息方式进行整合。这些传统的消息方式都是基于 IP 支持固定和移动网络传输，基于呈现服务支持多媒体，并且与一个统一的地址簿集成，能保持一致的用户体验，其具体的技术方案还在制定之中。

2.4 JCP 中 IMS 客户端的相关规范

JCP 是主要的 Java 标准组织，JSR (Java specification request) 则定义了 Java 应用程序需调用的应用编程接口 (API)。

JSR164 规范提供给 Java 开发者基于 Simple 协议栈的一套标准 API，用以开发基于呈现服务的 Java 程序。JSR165 规范也提供给 Java 开发者基于 Simple 协议栈的一套标准 API，用以开发基于即时消息服务的 Java 程序。而 JSR180 规范提供给 Java 开发者基于 SIP 协议栈的一套标准 API，这套 API 屏蔽了 SIP 的许多实现细节，开发者不需要对 SIP 有非常详细的了解就能开发出基于 SIP 的诸多应用程序。

JSR281 规范使应用开发者能很容易地开发出可以和 IMS 系统集成的应用程序，此规范以统一的高层 API 方式向用户提供 IMS 的功能。这些 API 最大限度地隐藏了 IMS 实现细节，抽象了下层技术，同时提供给开发者最大的灵活性。其 API 中至少支持 3 种类型的功能：高级 IMS 功能、PoC 服务和组列表管理服务。JSR281 规范目前没有涉及在 JSR164 和 JSR165 中已经定义了的呈现服务和即时消息服务。此规范还在制定过程中。

2.5 IMS 客户端的软件架构

通过对于 IMS 客户端相关规范的研究与分析，可以看出 IETF 提供了 IMS 客户端所需要的协议部分，包括详细的 SIP 信令消息交互，服务参数协商、媒体流的建立、XML 文档的交互等。3GPP 和 OMA 提供了 IMS 客户端所需要的服务引擎，与不同应用服务器之间的交互方式以及如何接入到 IMS 网络等。JCP 提供了一整套 IMS 客户端上 Java 应用程序所需的标准 Java 应用编程接口。由此可以总结归纳出 IMS 客户端软件架构参考，具体参见图 2。

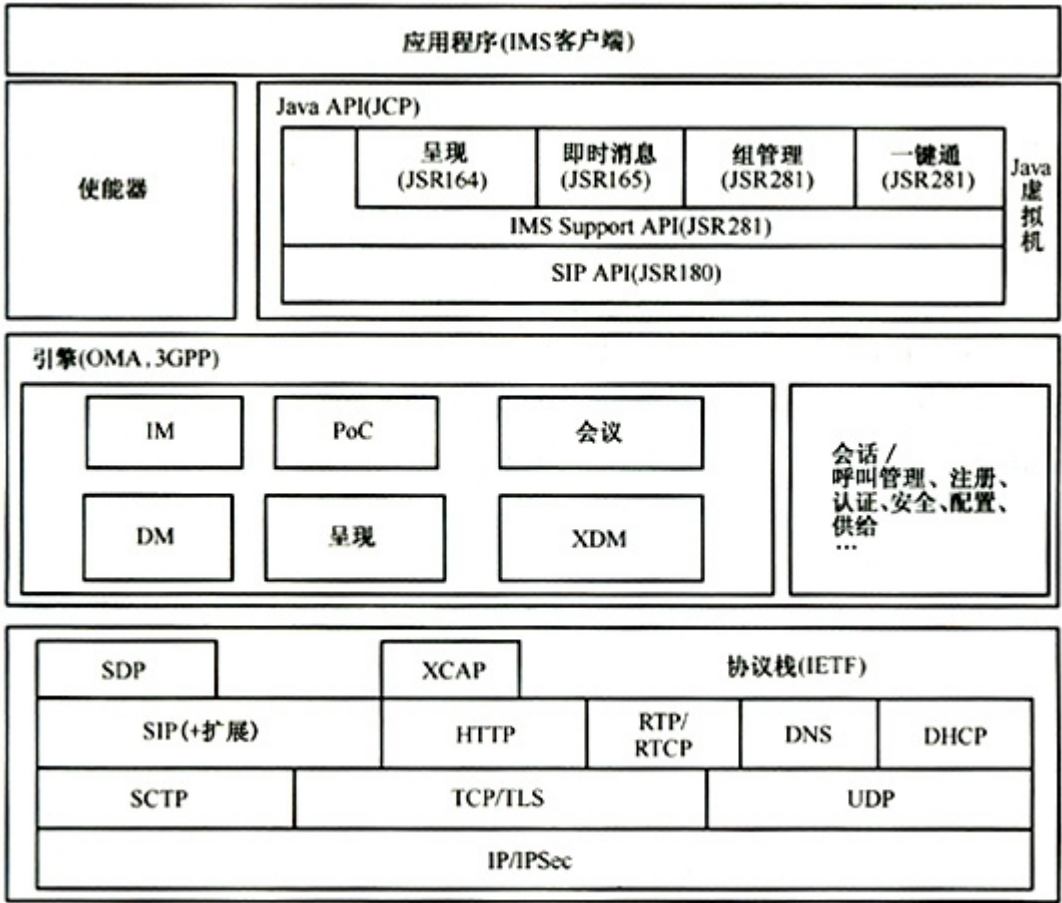


图 2 IMS 客户端软件架构参考

IMS 客户端软件架构主要包括了：

(1) 协议栈

IMS 客户端的底层是协议栈部分。它们大都是基于 IETF 标准的，包括 SIP、SDP、HTTP、XCAP、RTP/RTCP、DNS、DHCP 等。其中用于接入 3GPP 定义的 IMS 网络所要

求的那些 SIP 扩展部分也必须支持。

(2) 引擎/使能器

服务引擎是提供应用编程接口给上层应用程序或者第三方应用开发的关键部分。根据其所提供服务的不同,可以包括不同的引擎,比如呈现引擎、即时消息引擎等。这些引擎主要是在 OMA 和 3GPP 中定义的,其中一些共同的部件包括会话/呼叫管理、注册、认证、安全、配置、供给等。

(3) Java 应用编程接口

这些应用程序接口被上层的 Java 应用程序所使用。Java 应用程序给用户提供了可以下载的更丰富且与操作系统无关的 IMS 应用。

(4) 应用层/图形界面

应用程序给用户提供了 GUI 界面。GUI 界面应当足够的友好和方便,这样才能更好地展现 IMS 服务和应用。

3、IMS 客户端区别于一般 SIP 客户端的特性

通过研究可以发现,IMS 客户端和一般的 SIP 客户端有许多不同之处,它相比一般的 SIP 客户端而言需要支持更多的功能,也更加复杂,对于 IMS 终端的要求也更高。其中关键的一点是 IMS 客户端必须符合 IMS 相关规范,才能够接入到 IMS 网络。为用户提供一系列的 IMS 服务。

(1) SIP 扩展

IMS 客户端必须支持 SIP 扩展部分的有关规范,特别是 3GPP 所要求的那些 SIP 包头扩展部分,这样才能访问 IMS 网络。而一般 SIP 客户端只需要支持 RFC3261。

(2) 认证机制

IMS 标准中定义了不同的认证机制,如 HTTP 摘要 (RFC2617)、IMS-AKA (RFC 3310 和 3GPP TS 33.203) 和 pre-IMS 认证 (3GPP TR 33.878) 等。IMS 客户端需要支持更安全的认证方式 (如 IMS-AKA) 才能保证 IMS 终端和 IMS 网络之间的安全访问。

(3) IPSec

IPSec 在 IP 层上提供了多种安全机制,用于保证用户客户端和安全网关之间的安全通信。在 IMS 客户端和 P-CSCF 之间建立一个安全的 IPSec 通道,能确保 IMS 客户端安全地接入到 IMS 网络中,这个通道是在 IMS 注册过程中建立起来的,而一般 SIP 客户端不需要支持这种特性。

(4) 包压缩功能

SIP 包压缩能改善服务质量,特别是在无线环境下大大缩短呼叫建立时间。通过压缩网络和传输协议中的包头,能更有效地利用带宽,对 SIP/SDP 消息的压缩也提高了无线资源利用率。IMS 客户端一般都是通过移动无线方式接入 IMS 网络的,所以包压缩的功能是必须的。而一般 SIP 客户端是通过宽带接入,所以不需要支持这个特性。

(5) 前提条件下的 QoS 保证

前提条件下的 QoS 保证是指在会话建立过程中,必须在确保双方端到端的服务质量所需的媒体资源得以预留后,才能成功地建立起会话。比如在视频呼叫建立中,该机制用以验证会话中是否已经获得恰当的端到端服务质量。但是,这种机制比较复杂,延长了会话建立的时间。因此,仅在必要的时候,IMS 客户端才会打开这种机制。

(6) 发现机制

P-CSCF 是 IMS 客户端访问 IMS 网络惟一的接入点,所有从 IMS 客户端来的 SIP 信息都必须经过 P-CSCF。所以,在 SIP 信息发送前,IMS 客户端必须知道 P-CSCF 的地址。该地址不是预先配置好的,而是 IMS 客户端通过发现机制而获得的。这些机制包括基于 OTA (空中下载) 供给、基于 GGSN (gateway GPRS support node, GPRS 网关支持节点) 和基于 DHCP 的 P-CSCF 发现机制,除非是手工地配置 P-CSCF 信息,否则 IMS 客户端必须支持这个功能。

(7) IPv4/v6 的支持

一般 SIP 客户端只支持 IPv4,但是 3GPP 最初规定 IMS 客户端应当支持 IPv6。如果 IMS 核心网是 IPv4 和 IPv6 双栈,只支持 IPv4 的 IMS 客户端也能接入到这样的 IMS 网络中。

(8) ISIM 卡的支持

IMS 客户端通过 ISIM (IMS subscriber identity module) 卡中的信息来认证和注册到 IMS 网络。ISIM 卡中包括了用户的私有身份、公共身份、家乡域、密钥等与认证和注册相关的重要信息。如果是 USIM (universal subscriber identity module) 卡,也可以通过相关的算法推导出类似信息。但是 IMS 终端种类是多样性的,对非 IMS 移动终端,ISIM 卡的支持不是必须的,可以通过其他方式实现 IMS 网络认证和注册。

(9) CS 域和 IMS 的结合应用

3GPP 中定义了 CSI (combining CS bearer with IMS),即电路交换 (circuit switch, CS) 域和 IMS 的结合应用。IMS 客户端间语音呼叫仍然使用 CS 域,同时利用分组交换 (packet switch, PS) 域传送非实时媒体流。这样能保证语音质量,提高频谱利用率,解决了目前通过 GSM/UMTS 传送 IP 语音包而造成的语音质量下降的问题。CSI 的第一阶段不涉及网络侧,主要是 IMS 客户端间交换终端能力,保持 CS 域和 PS 域的同时通信。但是这种服务需要 IMS 终端支持双传输模式 (dual transfer mode, DTM) (如果是 GERAN 接入) 或者是 MultiRAB (multiple radio access bearer) 能力 (如果是 UTRAN 接入),这样才能同时建立 PS 域会话和 CS 域通话。

(10) 语音无缝切换

语音控制连续性 (voice call continuity, VCC) 是 3GPP 提出的解决 CS 域通话和 IMS 域会话之间的语音无缝切换的标准。支持 VCC 服务的 IMS 客户端和呼叫连续控制服务器配合,能保证用户进入和离开

家庭或者办公室里的 WLAN (无线局域网) 时仍然能保持 IMS 域或 CS 域语音呼叫的连续性。但是这种服务要求 IMS 终端具备多种无线接入能力, 如 GSM/WLAN 双模终端就具备这样的物理条件。

4、IMS 客户端软件开发中需注意的问题

通过对 IMS 客户端相关标准与技术的研究, 以下几点被认为是在 IMS 客户端软件开发中应当注意的方面:

(1) 符合标准及协议的一致性

IMS 客户端软件开发应当遵照相关标准组织的协议与规范进行, 特别是协议层的一致性, 需要严格按照 IETF 中的规定去解析和组织 SIP 包头。但是, 如果还没有提出相关的标准或者标准还没有完全被定义好, 一些私有的解决方案也是可行的, 因为标准总会存在一定的滞后。对 SIP 包头和携带的文档一些域进行私有定义以及通过 XCAP 中交互的 XML 文档中一些字段的私有定义, 可以实现一些 IMS 服务的创新。

(2) 保证与 IMS 网络和终端的互联互通性

IMS 客户端软件应当和不同的 IMS 网络提供商的应用服务器以及其他的 IMS 客户端软件进行互联互通测试, 从而保证客户端具有良好的互连性。IMS 客户端的复杂性决定了 IMS 客户端间互联互通的重要性。不同的 IMS 客户端可以支持不同的特性, 但是应当保持相同功能特性间的互通。比如具备 CSI 的 IMS 客户端仍然可以和不具备 CSI 的 IMS 客户端进行 PS 域的会话连接。

(3) 保持系统的可扩展性

IMS 客户端的功能和特性还在不停地变化与演进中, 因此, 应当确保 IMS 客户端软件架构设计中的可扩展性和灵活性, 以方便新的特性和引擎的加入。如果 IMS 客户端软件架构合理, 当有新的协议和引擎加入时, 只需增加相应的功能模块, 而不需要对已有的功能模块做较大的改动就可以增加新的 IMS 服务。

(4) 实现软件性能优化

由于手机上的 CPU、内存、电池等资源都是有限的, IMS 客户端软件中的关键部分应当注意实现性能上的优化, 如对内存的分配机制、电源管理、XML 文档解析器算法优化等。

(5) 提供软件平台的开放性

IMS 客户端软件应该能够提供相关的应用编程接口给第三方软件开发者。由于 IMS 服务是多样性的, IMS 客户端提供的这些接口会有助于更多的软件开发人员更快地开发出更多创新的 IMS 应用程序。IMS 客户端软件在提供接口的开放度和灵活性将有所权衡。JCP 中的 JSR281 为 IMS 客户端软件的 API 开发提供了一个很好的参考。

(6) 具有操作系统无关性

IMS 客户端软件应尽量保持与操作系统的无关性, 这样软件会很容易地被移植到其他的操作系统, 如 Windows Mobile、Symbian、Linux 或者一些专有的操作系统。这需要在软件架构设计中将与系统相关的部分尽可能地分离出来。比如 IMS 客户端中的引擎和协议栈部分应尽量保持系统无关性, 但是人机界面部分一般在不同的系统中都需要重新实现。系统无关部分调用相同的消息通信、内存分配、文件管理、信号管理等应用编程接口, 然后根据不同的操作系统重新编写这些 API。这种方法能很好地解决 IMS 客户端的软件移植问题。

(7) 支持传输层无关性

由于手机上的 CPU、内存、电池等资源都是有限的, IMS 客户端软件中的关键部分应当注意实现性能上的优化, 如对内存的分配机制、电源管理。IMS 客户端应当支持不同的传输方式, 如 GPRS、xDSL、Wi-Fi、WiMax 等接入方式, 并尽量保持接入方式的无关性, 但是不同的接入方式也会直接影响到 IMS 客户端的行为。比如通过 GPRS 接入, 就存在主和从 PDP (Packet data protocol) 上下文激活问题、在 PDP 上下文激活时获得 P-CSCF 地址问题、SIP 包压缩问题等。如果是通过 Wi-Fi 接入, 就不存在这些问题。如果 IMS 终端是双模, 其接入方式发生转换时也会对 IMS 客户端产生影响。在设计 IMS 客户端软件时应当适当考虑这些情况。

5、结束语

目前, 业界在 IMS 客户端的实际产品开发方面较之 IMS 网络要滞后一些, 但仍然已取得许多成果, 如爱立信已经推出了基于爱立信移动平台的 IMS 客户端, 实现了 weShare (语音和多媒体共享业务); 美国 Ecricio 公司推出了手机 IMS 框架软件, 集成多种 IMS 功能, 并提供了 IMS 软件开发包。随着 IMS 网络测试和今后 IMS 网络部署的展开, 可以预见, IMS 客户端逐渐会成为开发和研究的热点。

随着 IMS 应用的增加和丰富, IMS 客户端软件会变得越来越复杂, 对 IMS 终端的要求也会更高。比如对多线程和多任务的需要, 这要求 IMS 终端是一个智能终端, 比较低端手机可能不支持这样的特性。如果 IMS 客户端支持 CSI, IMS 终端就必须支持 DTM 模式或者具备 MultiRAB 能力。如果 IMS 客户端支持 VCC 或者一些固定移动网络融合服务, IMS 终端必须是一个多模终端, 包含多个无线空中接口。如果 IMS 客户端必须支持 IPSec 和包压缩, IMS 终端可能需要更强的 CPU/DSP 和更多的内存来处理复杂运算, 因此, 来自芯片制造商对 IMS 终端中的某些特性的硬件支持将有助于 IMS 终端的性能增强。

IMS 客户端中仍有大量的课题有待研究。在 IMS 客户端协议栈中 SIP 和 XCAP 都是基于文本的信令协议, 需要大量的文本解析工作, SIP 和 XML 解析器的性能和效率变得尤为重要, 因此如何优化解析器算法就是一个需要解决的课题。IMS 客户端的安全和认证机制也是比较复杂的, 不同的接入方式有完全不同的安全和认证要求, 同时上层各种 IMS 应用也有不同的服务级的安全要求, 如何整合和实现这些功能也是需要解决的问题。IMS 客户端的用户设备能力管理也是很重要的, 这些能力包括设备能力、网络能力和用户服务属性等, 这些能力可以是预设的, 可以是存储在网络侧的, 也可以是通过会话协商获得的。IMS 客户端的复杂性和多样性决定了 IMS 客户端的一致性测试和互联性测试是今后要面临的重大课题, 互联性没有很好地解决将会影响 IMS 技术和网络的发展。

随着 IMS 技术和应用的日渐成熟与推广, 对 IMS 客户端相关技术以及软件的设计实现方式等课题的深入

研究，将会对有关设备生产商及电信运营商等具有重要的参考借鉴意义。

参考文献

- 1 IETF RFC 3261.SIP: session initiation protocol, Jun 2002
- 2 IETF RFC 2327.SDP: session description protocol, Apr 1998
- 3 IEFT draft-ietf-simple-xcap.The extensible markup language (XML) configuration access protocol (XCAP) , May 2006