

Universidad Politécnica de Madrid  
Departamento de Ingeniería de Sistemas Telemáticos



**POLITÉCNICA**

**TESIS DOCTORAL**

**CONTRIBUCIÓN AL SOPORTE DE APLICACIONES  
MULTIMEDIA MULTIUSUARIO DE ALTA  
INTERACTIVIDAD EN EL SUBSISTEMA IP  
MULTIMEDIA**

Autor

**Alberto Hernández Ortiz**

*Ingeniero de Telecomunicación*

Director

**Enrique Vázquez Gallo**

*Catedrático de Universidad del Dpto. de Ingeniería de Sistemas  
Telemáticos. Universidad Politécnica de Madrid*

2010





**POLITÉCNICA**

**Título de la Tesis:** *Contribución al soporte de aplicaciones multimedia multiusuario de alta interactividad en el Subsistema IP Multimedia*

**Autor:** Alberto Hernández Ortiz

**Director:** Enrique Vázquez Gallo

Tribunal nombrado por el Excmo. y Magfco. Sr. Rector de la Universidad Politécnica de Madrid, el día \_\_\_ de \_\_\_\_\_ de 2010

**Presidente:** \_\_\_\_\_

**Vocal:** \_\_\_\_\_

**Vocal:** \_\_\_\_\_

**Vocal:** \_\_\_\_\_

**Secretario:** \_\_\_\_\_

**Suplente:** \_\_\_\_\_

**Suplente:** \_\_\_\_\_

Realizado el acto de defensa y lectura de Tesis el día \_\_\_ de \_\_\_\_\_ de 2010 en la E.T.S. de Ingenieros de Telecomunicación de Madrid.

**Calificación:**

EL PRESIDENTE

LOS VOCALES

EL SECRETARIO

---

*Este trabajo ha sido financiado parcialmente por el Proyecto AmIVital CENIT 2007-2010 subvencionado por el Centro para el Desarrollo Tecnológico Industrial (CDTI) perteneciente al Ministerio de Ciencia e Innovación de España.*

*Este trabajo ha sido financiado parcialmente por el Ministerio de Ciencia e Innovación de España a través del proyecto CASERTEL-NGN (TSI2005-07306-C02-01).*

*Adicionalmente, el autor es beneficiario del Programa Nacional de Formación de Personal Investigador (ref. BES-2006-12803) promovido por el Ministerio de Ciencia e Innovación.*

# Resumen

Las aplicaciones multimedia multiusuario de alto grado de interactividad (HIMMA) están en auge actualmente gracias a la evolución de las redes de acceso fijas y móviles, las mejoras en los terminales de usuario y a la propia oferta de aplicaciones existente. Aplicaciones como las de trabajo colaborativo, juegos multijugador, tele-enseñanza o telemedicina fomentan la comunicación entre usuarios en distintos ámbitos profesionales y de entretenimiento, tanto en entornos fijos como en movilidad.

Estas aplicaciones requieren generalmente un conjunto de funciones comunes como son la gestión de usuarios (alta, baja, almacenamiento de perfil, creación de grupos), información sobre servicios (presencia, predisposición a participar en actividades, etc.) y gestión de las comunicaciones (establecimiento de sesiones multiusuario, notificación de eventos, soporte de múltiples dispositivos, gestión del estado, calidad de servicio en la red, etc.). Todas estas funciones deben proporcionarse por infraestructura específica de la aplicación en caso de desplegarse en una red que sólo proporcione conectividad IP (como la propia Internet). Esta situación conduce a desarrollos habitualmente propietarios o específicos que dan lugar a deseconomías de escala y dificultan la interoperabilidad entre aplicaciones y, por tanto, entre usuarios, problema que constituye la motivación principal de la Tesis.

El Subsistema IP Multimedia (IMS) está definido por el 3GPP como arquitectura estándar para la provisión de servicios Todo-IP en redes móviles y ha sido adoptado por otros foros de estandarización para entornos de telefonía fija, redes de cable, fibra e inalámbricas. La arquitectura de IMS está basada en el establecimiento de sesiones mediante el protocolo SIP del IETF y cuenta con un conjunto de servicios básicos, denominados habilitadores, entre los que se incluyen gestión de grupos, presencia, mensajería, etc.

En esta Tesis se aborda la adecuación de IMS como plataforma para la provisión de aplicaciones multimedia multiusuario de alto grado de interactividad, estudiando las

aplicaciones multiusuario provistas de forma por estándar por IMS (videoconferencia, Pulsar para Hablar) y proponiendo como contribución básica un marco genérico de soporte a aplicaciones HIMMA basado en los estándares de IMS que facilita el desarrollo de este tipo de aplicaciones. Este marco de soporte se aprovecha de las posibilidades de extensión de IMS, a través del concepto de habilitador, y comprende dos elementos fundamentales: un habilitador ligero para la gestión de aplicaciones HIMMA y un middleware para clientes que ofrece una interfaz Java de alto nivel que facilita el desarrollo de aplicaciones HIMMA sin infraestructura específica aprovechando las capacidades de la red IMS.

Finalmente, se presenta la aplicabilidad de las contribuciones anteriores a dos tipos de escenarios relevantes para las aplicaciones HIMMA: juegos multijugador en red y servicios profesionales.

**Palabras clave:** comunicaciones multimedia, aplicaciones interactivas, IMS, SIP, juegos, HIMMA, multimodo, multidispositivo

# Abstract

Highly interactive multimedia multiparty applications (HIMMA) are gaining interest in the TIC market thanks to the evolution of fixed and radio access networks, improvements in user terminals and the broad range of existing applications. Applications such as collaborative work, multiplayer networked games, e-learning and e-health encourage communication among users in a variety of professional and entertainment spheres, both in fixed and mobile environments.

These applications usually require a set of common functions such as user management (creation of records, storage of the user profile, creation of groups, etc.), information services (presence, willingness to participate in activities, etc.) and communications management (establishment of multiparty sessions, event notification, support for multiple devices, state management, quality of service in the network, etc.). Should the network provide only IP connectivity (like the Internet), all the previous functions must be provided by application-specific infrastructure deployed in the network. This situation usually leads to custom developments of the required functions, where different providers duplicate the same functionality. It also generates scale diseconomies and makes interoperability among applications difficult and, thus, among users.

The IP Multimedia Subsystem (IMS) is defined by 3GPP as a standard architecture for the provision of All-IP services in mobile networks and has been adopted by other standardization bodies for fixed telephony, cable, fiber optic and wireless networks. The IMS architecture is based on the SIP signaling protocol to create and maintain communication sessions. It also includes a set of basic services, namely service enablers, including group management, rich presence, messaging and so on.

This PhD thesis addresses the suitability of the IMS as a platform for the provision of highly interactive multimedia multiparty applications and proposes as global contribution a generic standards-based framework to ease the development and

deployment of HIMMA applications. This framework makes the most of the extensibility of IMS architecture and protocols through the concept of service enabler, comprising two core elements: a Lightweight Conferencing Enabler (LCE) to manage HIMMA applications, and a HIMMA middleware which provides developers a high-level Java interface to ease the development of HIMMA applications without application-specific infrastructure.

Finally, we present the applicability of the previous contributions to two types of relevant scenarios: multiplayer networked games and professional services.

**Keywords:** Multimedia communication, interactive applications, IMS, SIP, gaming, HIMMA, multimodal, multidevice



# Agradecimientos

La finalización de la Tesis es, sin duda, un hito muy importante en el camino y en estas líneas quiero dar las gracias a todos los que me habéis acompañado haciéndolo posible. Elegir el orden de los agradecimientos siempre es un problema, pero no puedo empezar por otras personas sino por mis padres Antonio y Carmen que son los que me animaron en primera instancia a iniciar esta etapa doctoral y los que han estado siempre a mi lado en todo momento ofreciéndome consejo, alivio y fuerza.

Agradecer por supuesto a mi director de Tesis, Enrique, haber puesto toda su experiencia y tiempo a mi disposición para guiar adecuadamente este trabajo y conseguir encajar todas las piezas del puzzle de ideas que han dado lugar a esta Tesis. Comentario extensible a todos mis compañeros del grupo RSTI-UPM, dirigido por Julio -y al que agradezco especialmente la confianza depositada año tras año-, porque no sólo han contribuido a un inmejorable ambiente de trabajo sino que han colaborado activamente en la Tesis aportando constructivas sugerencias, comentarios, ideas y, además, han puesto todo en su mano para que en la recta final pudiera centrarme en la redacción de esta memoria.

Resistiré la tentación de dar nombres concretos porque la lista sería muy extensa, pero tampoco me olvido de los colegas de otros grupos y organismos con los que he participado en proyectos, congresos, clases, etc. y que han sido fuente de ideas, de los proyectistas que han pasado por el laboratorio (algunos actuales compañeros de trabajo y amigos), ni del personal técnico y de secretaría del Departamento.

Y, cómo no, tampoco me olvido de mis amigos y amigas más cercanos, que ya sabéis quiénes sois y el inestimable apoyo y compañía que me habéis brindado estos años.

¡Gracias a todos!



# Índice general

Resumen .....	i
Abstract.....	iii
Agradecimientos .....	v
Índice general.....	vii
Índice de figuras .....	xiii
Índice de tablas.....	xvii
Índice de listados.....	xix
Siglas .....	xxi
1. Introducción.....	1
1.1 Contexto.....	1
1.2 Motivación.....	4
1.3 Objetivos .....	12
1.4 Fases de trabajo y metodología .....	14
1.5 Proyectos relacionados .....	16
1.5.1 Acción COST 290 (Wi-QoST) .....	16
1.5.2 CASERTEL-NGN .....	16
1.5.3 AmIVital .....	16
1.5.4 VISION.....	17

1.6	Estructura de la memoria .....	18
2.	Estado del arte .....	21
2.1	Aplicaciones multimedia multiusuario de alta interactividad (HIMMA) ...	21
2.1.1	Aplicaciones y servicios de juego en red .....	21
2.1.2	Aplicaciones y servicios profesionales .....	23
2.2	El Subsistema IP Multimedia (IMS).....	26
2.2.1	Arquitectura general de IMS .....	26
2.2.2	Servicio de conferencias audiovisuales multimedia .....	29
2.2.3	Servicio Pulsar para Hablar (Push to Talk).....	29
2.3	Modelos de comunicaciones multiusuario .....	30
2.4	Gestión de documentos XML (XDM 2.0) .....	34
2.4.1	Presencia .....	36
2.4.2	Lista compartida.....	36
2.4.3	Perfil compartido.....	36
2.4.4	Política compartida .....	37
2.4.5	Grupo compartido.....	37
2.5	Tecnologías y arquitecturas de soporte a aplicaciones HIMMA en el IMS .	39
2.5.1	IMS Client Platform (JSR 281 y JSR 325) .....	39
2.5.2	Java SipServlets (JSR 289).....	41
2.5.3	Plataforma de juegos de la Open Mobile Alliance .....	42
2.5.4	Propuestas de soporte a aplicaciones HIMMA.....	46
2.5.5	Soporte de aplicaciones HIMMA en plataformas IMS comerciales.....	49
2.6	Conclusiones .....	50
3.	Arquitectura de soporte de aplicaciones HIMMA en el IMS.....	53
3.1	Introducción.....	53

3.2	Arquitectura propuesta .....	55
3.2.1	Criterios de diseño .....	55
3.2.2	Descripción de la arquitectura.....	55
3.3	Ventajas de las extensiones propuestas.....	58
3.4	Relación entre el habilitador LCE y el middleware HIMMA .....	61
3.4.1	Comparación de funcionalidad.....	61
3.4.2	Uso conjunto .....	62
3.5	Conclusiones .....	62
4.	Habilitador ligero de gestión de actividades (LCE) .....	65
4.1	Introducción.....	65
4.2	Arquitectura .....	66
4.3	Modelos de datos.....	69
4.3.1	Modelo de descripción de la actividad .....	70
4.3.2	Modelo de estado de la actividad .....	70
4.4	Gestión de actividades multiusuario con LCE.....	81
4.4.1	Descripción de la actividad.....	81
4.4.2	Unión a una actividad.....	81
4.4.3	Búsqueda de actividades.....	82
4.4.4	Suscripción a eventos de la actividad.....	83
4.4.5	Finalización de la actividad .....	83
4.5	Funciones avanzadas del LCE.....	85
4.5.1	Pausa y reanudación de actividades.....	85
4.5.2	Transferencia de sesión con estado.....	86
4.5.3	Soporte multidispositivo/multimodo.....	89
4.6	Implementación y pruebas.....	91

4.7	Conclusiones .....	93
5.	Middleware de soporte a aplicaciones HIMMA.....	95
5.1	Introducción.....	95
5.2	Arquitectura .....	97
5.3	Modelo de datos .....	100
5.3.1	Modelo de participante como documento de presencia.....	100
5.3.2	Modelo de datos de contacto del middleware .....	102
5.3.3	Ejemplo de documento de presencia generado por el middleware....	103
5.4	Gestión de actividades.....	105
5.4.1	Descripción de la actividad.....	105
5.4.2	Unión a una actividad.....	108
5.4.3	Búsqueda de actividades.....	114
5.4.4	Suscripción a eventos.....	114
5.4.5	Finalización de la actividad .....	115
5.5	Implementación y pruebas.....	115
5.5.1	Prueba de registro en IMS.....	117
5.5.2	Prueba de presencia .....	118
5.5.3	Prueba de establecimiento de sesiones de actividad distribuida .....	118
5.6	Conclusiones .....	119
6.	Aplicación a distintos entornos .....	121
6.1	Servicios de juego en red multijugador.....	121
6.1.1	Funcionalidad de las plataformas de videojuegos comerciales.....	125
6.1.2	Presentación y análisis del caso de uso .....	135
6.1.3	Conclusiones .....	160
6.2	Servicios profesionales.....	164

6.2.1	Sala de espera virtual personalizada .....	166
6.2.2	Videoconferencia orientada a grupos.....	169
6.2.3	Compartición de imágenes con anotaciones (pizarra virtual) .....	171
6.2.4	Conclusiones .....	173
6.3	Consideraciones sobre escalabilidad .....	175
7.	Conclusiones y trabajos futuros .....	183
7.1	Contribuciones.....	184
7.2	Trabajos futuros.....	188
7.2.1	Contribución a estándares.....	188
7.2.2	Adaptación a nuevos estándares de IMS .....	189
7.2.3	Aplicación a otros proyectos: LiLa (Library of Labs) .....	189
7.2.4	Composición de servicios IMS / Web.....	191
7.2.5	Gestión de redes vehiculares en redes IMS-LTE.....	192
	Referencias .....	193





# Índice de figuras

Fig. 1. Situación actual de desagregación y duplicidad funcional en servicios de juegos en red .....	5
Fig. 2. Mal aprovechamiento de los recursos de red en el servicio de presencia .....	6
Fig. 3. Actores del mercado de juegos multijugador.....	10
Fig. 4. Entorno unificado de aplicaciones HIMMA con servicios comunes provistos por IMS .....	13
Fig. 5. Fases de trabajo de la Tesis.....	14
Fig. 6. Volumen de usuarios mundial de plataformas de juego en red destacadas ..	23
Fig. 7. Torre de protocolos de H.323.....	24
Fig. 8. Esquema general de la arquitectura IMS .....	28
Fig. 9. Arquitectura de IMS por capas.....	29
Fig. 10. Diferentes modelos de establecimiento de una conferencia con múltiples usuarios.....	31
Fig. 11. Arquitecturas de conferencia XCON y DCON.....	32
Fig. 12. Distribución de contenidos VS aplicaciones interactivas.....	34
Fig. 13. Arquitectura del gestor de documentos XML de OMA.....	35
Fig. 14. Esquema de un grupo compartido.....	38
Fig. 15. Arquitectura de la plataforma para clientes de IMS.....	41
Fig. 16. Primera versión de la plataforma de juegos de OMA.....	43

Fig. 17 Segunda versión de la plataforma de juegos de OMA.....	44
Fig. 18 Utilización de IMS en el Servicio de Juego de OMA .....	45
Fig. 19. Interacción entre módulos de la plataforma de juego [40] .....	47
Fig. 20 Arquitectura de soporte a aplicaciones de comunidad de Araki et al. [44] ...	48
Fig. 21 Arquitectura del caso de estudio de juegos de Chunyan et al. ....	49
Fig. 22 Marco general de soporte a aplicaciones HIMMA en el IMS.....	57
Fig. 23. Flexibilidad y mayor integración de las actividades con IMS gracias a LCE	59
Fig. 24. Simplificación del proceso de búsqueda de actividades por el cliente .....	61
Fig. 25. Arquitectura del habilitador ligero de actividades LCE.....	67
Fig. 26. Arquitecturas soportadas por LCE .....	69
Fig. 27. Modelo de conferencia SIPING (RFC 4575).....	71
Fig. 28. Gestión de estado centralizada en servidor de juegos (caso cliente-servidor) .....	74
Fig. 29. Gestión de estado centralizada en un supernodo (caso P2P).....	75
Fig. 30. Gestión de estado distribuida con soporte potencial de servidor de juegos (caso P2P).....	76
Fig. 31. Modelo de información de estado de actividad multiusuario distribuida ...	80
Fig. 32. Establecimiento de una actividad multiusuario basada en servidor central	84
Fig. 33. Transferencia de estado entre dispositivos del mismo usuario .....	87
Fig. 34. Transferencia de sesión entre dispositivos de distintos usuarios.....	88
Fig. 35. Almacenamiento y recuperación de estado a través de XDMS .....	89
Fig. 36. Diagrama de decisión de adaptación de aplicaciones.....	90
Fig. 37. Diagrama de bloques de implementación de LCE .....	92
Fig. 38. Comparativa de niveles de abstracción de HIMMA API, JSR 281 y JSR 180	96
Fig. 39. Middleware de soporte a aplicaciones HIMMA dentro de la arquitectura lógica de IMS.....	97

Fig. 40. Modelo de dos capas del middleware de soporte a aplicaciones HIMMA...	98
Fig. 41. Encaminamiento de mensajes SIP en el terminal basado en IARI.....	108
Fig. 42. Ejemplo mecanismo de descubrimiento de supernodos .....	110
Fig. 43. Toma de decisiones ante una petición de incorporación a una actividad...	112
Fig. 44. Relación de las clases del middleware con la biblioteca ICP.....	116
Fig. 45. Establecimiento de actividad de tres usuarios con supernodo con middleware HIMMA.....	119
Fig. 46. Situación actual de desagregación y duplicidad funcional en servicios de juegos en red .....	122
Fig. 48. Caracterización de las funciones de juego multiusuario y alternativas de despliegue en IMS.....	124
Fig. 49. Escenario de análisis de un sistema de entretenimiento multijugador actual .....	131
Fig. 50. Consumo de ancho de banda de subida por jugador en distintos juegos de Xbox 360.....	133
Fig. 51. Comparativa de arquitecturas de red habituales en juegos multijugador populares.....	134
Fig. 52. Fases habituales de interacción con una plataforma de juegos.....	137
Fig. 53. Fase de elección de dispositivo.....	138
Fig. 54. Síntesis de capacidades de dispositivos de juego .....	140
Fig. 55. Registro con autenticación en IMS.....	142
Fig. 56. Elementos de IMS relacionados con el perfil de usuario .....	142
Fig. 57. Fase de elección de la actividad.....	144
Fig. 58. Elementos relacionados con la tarificación de servicios en IMS.....	146
Fig. 59. Fase de elección de compañeros de juego.....	148
Fig. 60. Elementos relacionados con la gestión de grupos en IMS.....	149
Fig. 61. Ejemplo de almacenamiento de contactos en el XDMS .....	150

Fig. 62. Ejemplo de recuperación de contactos del XDMS .....	151
Fig. 63. Ejemplo del uso de XDMS y 3PCC para creación de torneos .....	154
Fig. 64. Mejora del proceso de búsqueda de usuarios en IMS con mecanismo de reputación.....	155
Fig. 65. Establecimiento de sesión de juego de dos jugadores.....	157
Fig. 66. Elementos de IMS relacionados con conferencias audiovisuales .....	158
Fig. 67. Servicio de soporte al modo fantasma.....	160
Fig. 68. Creación y búsqueda de partidas con LCE.....	163
Fig. 69. Agrupación funcional de los nodos de control de las comunicaciones .....	164
Fig. 70. Sala de espera virtual .....	168
Fig. 71. Videoconferencia AmIVital.....	170
Fig. 72. Aplicación de LCE a pizarra virtual .....	172
Fig. 73. Establecimiento de una sesión de pizarra virtual con RFC 4483 .....	173
Fig. 74. Esquema actual de acceso a experimentos remotos en LiLa .....	190

# Índice de tablas

Tabla 1. Información específica de estado de aplicación necesaria para distintas actividades.....	72
Tabla 2. Criterios de asignación de supernodos .....	111
Tabla 3. Síntesis de características generales de plataformas de juego en red .....	127
Tabla 4. Síntesis de características de comunicación directa .....	128
Tabla 5. Síntesis de características de comunidad o sociales .....	129
Tabla 6. Elementos de información de presencia enriquecida.....	152
Tabla 7. Síntesis del impacto de consultas al XDMS impuestas por el marco de soporte a aplicaciones HIMMA.....	179



# Índice de listados

Listado 1. Ejemplo de búsqueda de actividades con Xquery.....	83
Listado 2. Ejemplo de documento RPID generado por el middleware.....	104
Listado 3. Ejemplo de REGISTER con indicación de aplicaciones soportadas.....	107
Listado 4. Estructura de ejemplo de una aplicación que utilice el middleware HIMMA .....	117
Listado 5. Ejemplo de mensaje PUT para crear una lista de amigos en el gestor de grupos .....	150





# Siglas

3GPP	Third Generation Partnership Project
AAA	Authentication, Authorization and Accounting
AJAX	Asynchronous JavaScript And XML
API	Application Programming Interface
AS	Application Server
ATM	Asynchronous Transfer Mode
B2B-UA	Back-to-Back User Agent
BGCF	Border Gateway Control Function
CDF	Charging Data Functions
CDR	Charging Data Record
CGF	Charging Gateway Function
CSCF	Call State/Session Control Function
CSCW	Computer-Supported Cooperative Work
DRM	Digital Rights Management
DSL	Digital Subscriber Line
ETSI	European Telecommunications Standards Institute

GLM	Group List Management
GS	Games Services
GSCF	Group Session Control Function
GSM	Global System for Mobile communication
HICE	Highly Interactive Computing in Education
HIMMA	Highly Interactive Multimedia Multiuser Applications
HIWA	Highly Interactive Web Applications
HSS	Home Subscriber Server
HTTP	HyperText Transfer Protocol
ICP	IMS Client Platform
I-CSCF	Interrogating CSCF
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IP	Internet Protocol
ISC	IMS Service Control
J2ME	Java 2 Micro Edition
JSR	Java Specification Request
LCE	Lightweight Conferencing Enabler
LTE	Long Term Evolution (3GPP)
MCU	Multipoint Control Unit
MEGACO	Media Gateway Control Function
MGIF	Mobile Games Interoperability Forum
MGW	Media Gateway
MMORPG	Massively multiplayer online role-playing game
MRF	Media Resource Function

MRFC	Media Resource Function Controller
MRFP	Media Resource Function Processor
MSISDN	Mobile Station International ISDN Number
NGN	Next Generation Network
OCS	Online Charging System
OMA	Open Mobile Alliance
OSA	Open Service Access
P2P	Peer to peer
P4P	Proactive network Provider Participation for P2P
P-CSCF	Proxy CSCF
PDA	Personal Digital Assistant
PGM	Presence and Group Management
PIDF	Presence Information Data Format
PoC	PTT over Cellular
PTT	Push To Talk
RFC	Request For Comments
RTCP	Real Time Control Protocol
RTP	Real-time Transport Protocol
SCIM	Service Capability Interaction Manager
S-CSCF	Serving CSCF
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SIPPING	SIP Proposal Investigation
TCP	Transmission Control Protocol
TIC	Tecnologías de la Información y las Comunicaciones

TISPAN	Telecoms and Internet converged Services and Protocols for Advanced Networks
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
UGC	User Generated Content
UGS	User Generated Services
UMTS	Universal Mobile Telecommunication System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UTRAN	UMTS Terrestrial Radio Access Network
VoIP	Voice over IP
WiFi	Wireless Fidelity (ver WLAN)
WIMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WWW	World Wide Web
XCAP	XML Configuration Access Protocol
XCON	Centralized Conferencing
XDM	XML Document Management
XDMS	XDM Server
XML	Extensible Markup Language

# 1. Introducción

## 1.1 Contexto

Uno de los conceptos clave en la evolución reciente tanto de la informática como de las nuevas redes de comunicaciones es el de **multimedia**, entendido como la capacidad de procesar diversos tipos de medios de forma simultánea, habitualmente audio, vídeo y texto.

El acceso a contenidos multimedia y la disponibilidad de nuevas redes de banda ancha permiten aquello que los sistemas clásicos de difusión como la televisión, principal fuente de consumo de contenidos en el hogar, nunca han podido ofrecer: **interactividad**. La interactividad implica que el usuario tiene cierto papel protagonista en la prestación del servicio, bien sea en poca medida como podría ser elegir uno de los finales alternativos de una película, o en mayor medida como podría ser la participación en un videojuego o una videoconferencia.

La tendencia es que el usuario ya no es un simple consumidor de contenidos o servicios sino que, gracias a su facilidad de interacción, puede convertirse en generador de los mismos. Prueba de ello son los sistemas de blogs (como *Wordpress*) o microblogs (como *Twitter*) donde los usuarios escriben su propio diario, enciclopedias como la *Wikipedia* creada por y para usuarios, sitios de intercambio de vídeos y contenido multimedia como *Flickr* o *YouTube*, las cada vez más famosas redes sociales como *Facebook* o *Tuenti*, o videoconsolas como *Playstation 3* o *Xbox 360* que apuestan por los juegos multijugador en red, donde los propios usuarios sustituyen la inteligencia artificial que tradicionalmente controlaba a los jugadores con los que se enfrentaba cada usuario.

En general, aquellos servicios que, como los ejemplos citados, tienen las interacciones entre sus usuarios como un elemento fundamental, requieren una serie de funciones básicas que den soporte al registro de usuarios, la gestión de contactos, el

establecimiento de grupos, etc. Este es uno de los aspectos relevantes para la Tesis. Dentro del marco general de prestación de servicios en los que los usuarios tienen un papel predominante, conviene destacar una serie de servicios y aplicaciones multimedia, objeto de esta Tesis, que contempla tanto la participación de varios usuarios simultáneamente, es decir, **multiusuario**, como un **alto grado de interactividad**.

Algunos ejemplos indicados anteriormente, como los blogs, ofrecen baja interactividad pues, en un caso habitual, el usuario empieza a escribir un comentario y al cabo de unos minutos lo termina y envía por la red, recibiendo la consiguiente página de confirmación. La interacción en este caso consiste únicamente en escribir un texto y navegar por enlaces. En cambio, los servicios de alto grado de interactividad, también denominados “**intensivos en comunicaciones**” (del inglés *communication intensive*), permiten una mayor inmersión en los servicios. En ellos, las interacciones entre usuarios son más directas y se establecen normalmente en el contexto de sesiones de comunicación en las que múltiples usuarios pueden participar activamente mientras dura la sesión. Las sesiones tienen un estado asociado, que puede ser complejo. Las acciones de un participante pueden modificar el estado y, típicamente, sus efectos deben ser percibidos por los demás miembros de la sesión en un tiempo corto. Por ejemplo, los servicios que componen una aplicación de aprendizaje en red son de alta interactividad, como podría ser una pizarra compartida donde cada trazo pintado por un usuario modifica el estado de la pizarra y se refleja en las pantallas de los observadores de la misma. Las aplicaciones de trabajo colaborativo constituyen otro ejemplo similar. Así mismo, los juegos en red habitualmente son de alta interactividad porque varios usuarios interactúan simultáneamente modificando el escenario del juego y sus acciones tienen repercusión en otros jugadores que estén en dicho escenario.

La característica de “alta interactividad” se suele asociar a requisitos de prestaciones, por ejemplo a cumplir unos límites de retardo aceptables para la aplicación considerada. El cumplimiento de determinados requisitos de retardo o, más en general, de QoS (Quality of Service) dependerá, en general, de múltiples factores como la capacidad disponible en el acceso de cada usuario a la red, el adecuado dimensionamiento de la misma, la existencia de mecanismos de discriminación entre flujos de tráfico con diferentes necesidades de QoS, reserva de recursos, etc. así como de mecanismos que permitan a las aplicaciones solicitar la calidad que necesitan en cada momento, de la capacidad de los servidores, etc. Sin embargo, el propósito de esta Tesis no es tratar cuestiones de QoS, tema sobre el que existen otros trabajos, sino centrarse en otras funciones clave que son igualmente necesarias para las aplicaciones multimedia multiusuario de alta interactividad, en particular las funciones

relacionadas con la gestión de sesiones: establecimiento de sesiones multiusuario, notificación de eventos de sesión, soporte de terminales con diferentes características, gestión de estado (visibilidad del estado, pausa y reanudación, transferencia a otro terminal), etc. Las funciones que se acaban de mencionar, y que se definirán oportunamente a lo largo del trabajo, constituyen un segundo aspecto relevante para la Tesis.

En resumen, la Tesis se centra en los dos grupos de funciones introducidos en los párrafos anteriores, gestión de usuarios y gestión de sesiones, en aplicaciones multimedia multiusuario de alta interactividad.

Como se verá, las propuestas de este trabajo se basan en el uso de la arquitectura de red normalizada conocida como Subsistema IP Multimedia (IMS), aprovechando las funciones básicas que IMS ofrece para dar soporte a las aplicaciones. Entre éstas, se encuentra precisamente el control de la QoS que necesitan las aplicaciones. Sin embargo, el uso de IMS en la Tesis no se reduce al soporte de QoS, sino que se ha seguido el criterio general de tratar de aprovechar lo más posible todas las funciones útiles que ofrece IMS, complementándolas con funciones específicas propuestas para mejorar el soporte de las aplicaciones multiusuario objetivo de la Tesis.

Al maximizar las funciones que se delegan en elementos del IMS, que forman parte de la infraestructura de red provista por los operadores, se busca reducir el número de funciones a implementar en servidores o elementos propios de las aplicaciones y mejorar su escalabilidad. El desarrollo de las aplicaciones se simplifica, reduciendo su coste y contribuyendo a evitar la aparición de soluciones propietarias incompatibles entre sí.

Al tener una arquitectura abierta, IMS permite, respetando los protocolos definidos en los estándares, añadir nuevos módulos habilitadores que soporten las funciones específicas necesarias. Que las tareas básicas de gestión de usuarios y de gestión de sesiones se hagan de una forma estándar y compatible entre aplicaciones distintas, en este caso con el soporte de IMS, no solo facilita el desarrollo de las aplicaciones, sino que beneficia también a los usuarios, los cuales no necesitan registrarse y utilizar varias plataformas distintas para poder acceder a los servicios que presta cada una. Es decir, se trata de combatir tanto la duplicación innecesaria de funciones como la segregación de los usuarios en grupos cerrados ligados a plataformas particulares. Además, los desarrolladores pueden concentrar su esfuerzo en mejorar los aspectos propios de su aplicación que sean más innovadores y atractivos para los usuarios, diferenciándose de los competidores.

De aquí en adelante se referirán frecuentemente las aplicaciones multimedia multiusuario de alta interactividad que dan título a la Tesis por su acrónimo HIMMA (*Highly Interactive Multimedia Multiuser Applications*). El acrónimo es una propuesta original, pues si bien otros autores mencionan acrónimos similares como HIWA (*Highly Interactive Web Applications*) [1] o HICE (*Highly Interactive Computing in Education*) [2], ninguno de ellos implica necesariamente ser multimedia y multiusuario, características fundamentales en el ámbito de esta Tesis.

Los puntos siguientes desarrollan las ideas avanzadas en este apartado y profundizan en la motivación y objetivos de la Tesis. En ellos se presenta la forma en que se suelen implementar actualmente las aplicaciones HIMMA sobre redes IP, los problemas planteados y las propuestas originales de este trabajo encaminadas a resolverlos. Como se ha dicho, con estas contribuciones se pretende mejorar el soporte estándar que las redes IP ofrecen a las aplicaciones HIMMA, para simplificar el desarrollo de las mismas, abaratar su despliegue, facilitar la oferta de nuevos y mejores servicios y, en definitiva, enriquecer la experiencia de los usuarios.

## 1.2 Motivación

Actualmente, las aplicaciones multimedia multiusuario de alta interactividad (HIMMA) introducidas previamente están, en su mayor parte, ofrecidas a través de Internet (para el público en general, como pudieran ser los juegos en red) o mediante redes IP privadas bien físicas o virtuales (en ámbitos corporativos, típicamente con aplicaciones de trabajo colaborativo). En cualquiera de estos casos, la red se utiliza únicamente para el transporte de datos a nivel IP, por lo que cada aplicación debe implementar el resto de funciones habituales de este tipo de aplicaciones: gestión de perfiles de usuario, autenticación, tarificación, comunicación multimedia, mensajería, gestión de la movilidad, etc.

Tomando como ejemplo las plataformas de juego en red encontradas actualmente en el mercado, aunque el ejemplo es generalizable a cualquier otro tipo de aplicaciones, todas ellas tienen servidores específicos desplegados para dar soporte a todas las funciones necesarias, como muestra la Fig. 1, incluyendo el registro de usuarios, la gestión de listas de amigos, actualización del estado de presencia, etc., además de las propias funciones específicas de los juegos (clasificaciones, creación de torneos, etc.). El despliegue y mantenimiento de estas infraestructuras específicas por parte de cada proveedor de servicios de juegos supone un gran esfuerzo técnico y económico a la hora de ofrecer un servicio de calidad (máxime teniendo en cuenta que Internet es una



red *best-effort*), al tiempo que dificulta la interoperabilidad entre los usuarios de cada plataforma y obliga a los desarrolladores a multiplicar los esfuerzos de implementación [3].

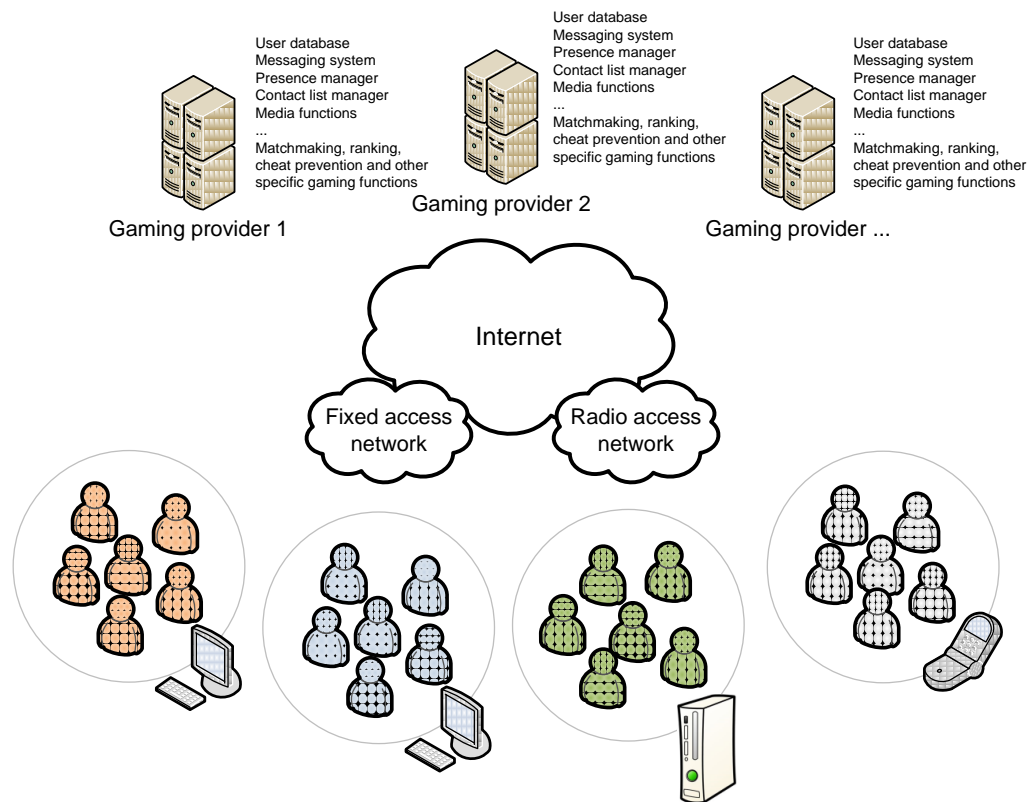


Fig. 1. Situación actual de desagregación y duplicidad funcional en servicios de juegos en red

Por otro lado, al no contar con otro servicio más que conectividad IP en las redes de datos disponibles para los usuarios hoy día, las aplicaciones multiusuario habitualmente requieren infraestructura específica y se diseñan conforme a una arquitectura cliente-servidor. Alternativamente, existen aplicaciones basadas en topologías P2P donde los propios usuarios tienen instalado software que actúa como cliente y servidor simultáneamente, si bien requieren igualmente uno o más nodos estables denominados servidor de arranque o *bootstrap*.

Si es un servidor centralizado el que soporta la aplicación y está alejado de los usuarios hay un mal aprovechamiento de los recursos de red. La Fig. 2 ilustra este caso con un servicio muy habitual de cualquier aplicación HIMMA: la información de presencia. Los servidores de programas de mensajería instantánea con información de presencia como MSN Messenger se encuentran en los Estados Unidos, de modo que el mensaje de presencia de un usuario de España a otros del mismo país va a seguir un recorrido como el mostrado en la Fig. 2, atravesando el Atlántico hasta en tres ocasiones cuando todos los usuarios involucrados están en un radio de escasas centenas de kilómetros.

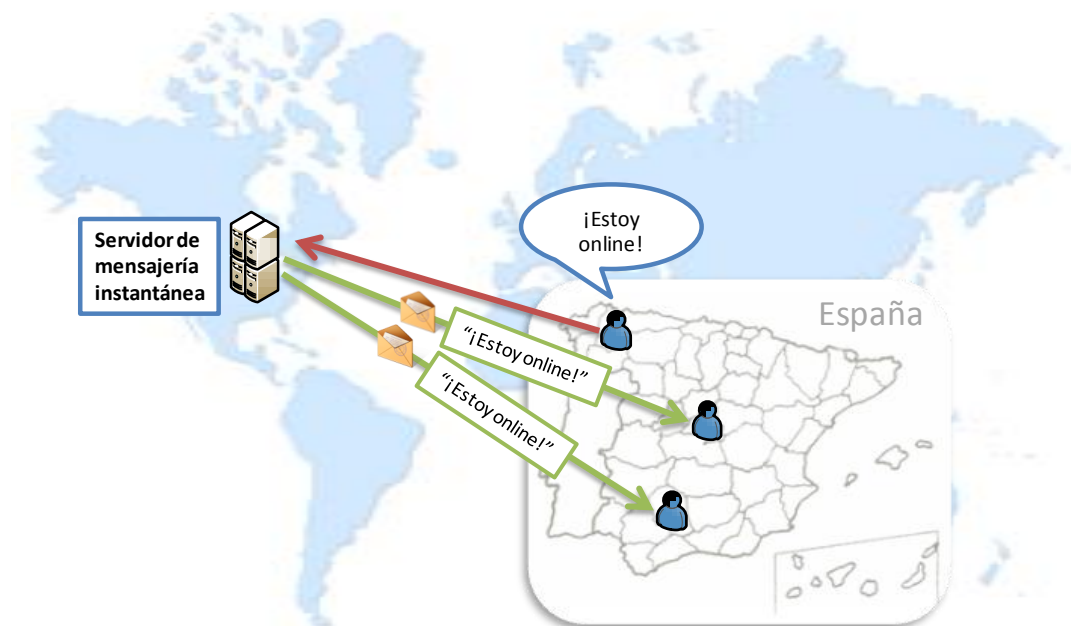


Fig. 2. Mal aprovechamiento de los recursos de red en el servicio de presencia

El Subsistema IP Multimedia (IMS) [4], estandarizado por el 3GPP como la arquitectura Todo-IP para provisión de servicios multimedia en UMTS, ya comparte la visión de la red como algo más que un medio de transporte, dotándola de una serie de servicios básicos, denominados habilitadores o *enablers*, que las aplicaciones pueden utilizar, como la gestión de información de presencia, mensajería, gestión de grupos o gestión de sesiones multimedia, redundando en una menor infraestructura específica necesaria por parte de los desarrolladores y proveedores de aplicaciones. Además, otra ventaja de IMS derivada de ser una arquitectura desplegada en un entorno de operador es que permite optimizar los recursos de red, así como incluir aspectos de calidad de servicio (QoS) no presentes en Internet. Por ello, IMS tiene el potencial de reducir el coste de las aplicaciones HIMMA al tiempo que permite ofrecer a los usuarios una mejor experiencia tanto en funcionalidad como en prestaciones y, a los desarrolladores, una mayor facilidad para crear aplicaciones avanzadas.

En el periodo de 2002 a 2006 IMS se ha ido perfilando como una actividad de estandarización creciente en el 3GPP y se ha planteado como la propuesta clave para el despliegue de nuevos servicios convergentes IP en el mercado de telecomunicación tanto en el entorno móvil (a través del propio 3GPP), como fijo (a través del trabajo del grupo ETSI TISPAN [5]).

La adopción y despliegue de soluciones IMS por parte de los operadores ha pasado por diferentes etapas, desde el optimismo inicial hasta fases de escepticismo sobre la conveniencia de desplegar esta nueva infraestructura. A las dudas de los operadores ha contribuido el éxito entre el público de aplicaciones de Internet como las redes sociales o aplicaciones de comunicación, que deben soportar las redes de los operadores de comunicaciones sin que estos obtengan beneficios. Sin embargo, aunque inicialmente las aplicaciones de Internet se consideraban un factor negativo para el desarrollo de IMS, actualmente se consideran una oportunidad para ofrecer servicios convergentes de mayor valor añadido para el usuario. Detrás de todo esto, subyace la preocupación de los operadores por el impacto de unas u otras soluciones en los futuros modelos de negocio.

No obstante, desde 2007 son numerosos los grandes operadores que manifiestan que no existe una alternativa clara a IMS como solución que permita un entorno para el despliegue de servicios Todo-IP controlado, robusto y fiable. El informe [6] indica que operadores como ATT, Deutsche Telekom, France Telekom, Telefónica y Verizon entre otros ya han desplegado soluciones, en uno u otro grado, basadas en IMS. Del mismo modo, empresas como Sun, Nokia, Siemens o Ericsson disponen en su catálogo de soluciones IMS, tanto hardware como software, y lo consideran una gran apuesta, aportando incluso herramientas de desarrollo (Sun Sailfin, Ericsson SDS, etc.), y fomentando la creación de comunidades de desarrolladores que empujen el despliegue de IMS mediante la creación de servicios innovadores.

Recientemente, a finales de 2009, ligado al impulso recibido por la nueva tecnología de acceso móvil LTE (*Long Term Evolution*), las principales compañías de telecomunicación del mundo han anunciado [7] el soporte de IMS como tecnología de núcleo de red que permita la interoperabilidad de los servicios prestados en redes LTE. Entre las compañías involucradas se encuentran AT&T, Orange, Telefónica, TeliaSonera, Verizon (que prevé ser pionera en 2011 con el despliegue comercial de redes móviles LTE/IMS), Vodafone, Alcatel-Lucent, Ericsson, Nokia Siemens Network, Nokia, Samsung y Sony Ericsson, por lo que están representados operadores, suministradores y fabricantes de terminales.

El diseño de aplicaciones innovadoras (la ansiada búsqueda de la *killer application*<sup>1</sup>) y la demostración de las ventajas que proporciona IMS a los actores involucrados, son probablemente los principales escollos que tiene que superar IMS para asentarse en el mercado. Independientemente del trabajo de investigación que hay alrededor de IMS,

---

<sup>1</sup> Dícese de un servicio o aplicación tan necesario o deseado que demuestra de forma determinante el valor de la tecnología que lo soporta, provocando un aumento notable de la adopción de dicha tecnología.

los propios documentos de especificación de IMS se centran principalmente en cómo ofrecer los servicios que ya se prestan hoy día con redes basadas en circuitos (mensajería, llamadas de voz, vídeo y conferencias), y no es hasta la *Release 8*, cuya especificación pasó a estado de *congelada* en diciembre de 2008, cuando se empieza a pensar en servicios de auténtico valor añadido para el usuario, como las comunicaciones enriquecidas (*rich communications*) que permiten incluir, a la vez que el audio o el vídeo, servicios como compartición de ficheros e imágenes, por ejemplo. A la hora de plantear las contribuciones de la Tesis en el ámbito de soporte de aplicaciones multimedia multiusuario de alta interactividad en el IMS, merece la pena distinguir los dos grandes grupos de aplicaciones de este tipo que existen actualmente: las orientadas a la colaboración en el trabajo y otros ámbitos profesionales en general (aprendizaje, asistencia sanitaria,...) y las orientadas al juego en red y entretenimiento en general.

Ambos grupos de aplicaciones tienen en común una participación de múltiples usuarios con una alta interactividad, si bien en el caso de los juegos en red estas características están en ocasiones más acentuadas por contar con elementos competitivos y servicios de comunidad. Así, mientras que las aplicaciones de colaboración en el trabajo suelen estar limitadas a un grupo de personas estable, acotado y de confianza, los juegos en red suelen gozar de mayor flexibilidad (dependiendo del tipo de juego, obviamente) al incorporar nuevos jugadores a la partida o crear torneos de forma dinámica donde los jugadores no tienen por qué conocerse. La interactividad es también un punto cada vez más potenciado en las aplicaciones de juego en red, máxime en juegos de acción donde los jugadores están constantemente realizando movimientos y, actualmente, llegan al punto de fomentar la participación entre usuarios mediante la integración de aplicaciones orientadas inicialmente al trabajo en grupo como la pizarra electrónica, la videoconferencia o el intercambio de ficheros.

Los juegos multijugador en red y su creciente número de funciones orientadas a la colaboración e interacción de la comunidad de juego son, por tanto, un excelente tipo de aplicaciones HIMMA a analizar en profundidad, con la ventaja añadida de poder extrapolar los resultados obtenidos a otro tipo de aplicaciones HIMMA como los citados de trabajo colaborativo, asistencia sanitaria (*e-health*) o aprendizaje en red (*e-learning*).

Por otro lado, no hay que perder el punto de vista de mercado, donde las aplicaciones de entretenimiento interactivo y, en particular, los juegos en red, pueden ser clave para acelerar la adopción de IMS entre el público. En la actualidad, la industria del videojuego, apoyada por los avances tecnológicos en cuanto a representación gráfica y conectividad, vive su mayor apogeo y empieza a liderar el

sector del entretenimiento por delante de la industria audiovisual tradicional [8]. La entrada en la denominada era de la alta definición [9] ha significado un impulso en la calidad visual de los juegos, que contribuyen a sumergir en mayor medida al usuario dentro de los universos virtuales que representan. De forma paralela, la generalización de las redes de banda ancha, tanto fijas como móviles, incentiva la incorporación de funciones de valor añadido a las plataformas de juego, permitiendo establecer partidas multijugador con audio y vídeo en tiempo real, así como fomentar la participación de los usuarios incorporando funciones de comunidad como indicación del estado de presencia, mensajería instantánea, búsqueda de usuarios, establecimiento de torneos, rankings, recomendaciones de juegos, etc.

Algunos de los servicios o plataformas de juego en red actuales más conocidos son Battle.net [10] (utilizado en juegos de rol de la firma *Blizzard Entertainment* como Diablo, Warcraft o Starcraft, que se estima que cuenta con más de 20 millones de usuarios<sup>2</sup>), Xbox Live [11] (asociado principalmente a la consola de videojuegos Microsoft Xbox 360, que cuenta con más de 20 millones de usuarios de su servicio en red [12]), PlayStation Network Platform (asociada a las consolas de Sony PS2, PS3 y PSP) y Wi-Fi Connection (para las consolas Nintendo Wii y DS).

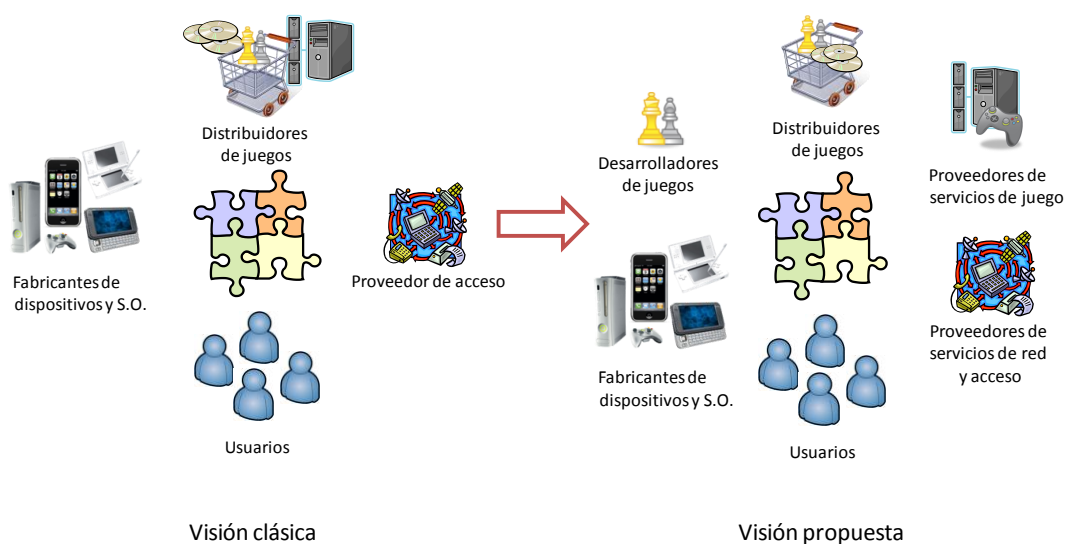
Además, existe un conjunto de casi 300 millones de usuarios habituales ([13] y [14]) de los denominados mini-juegos web (o “juegos Flash” porque habitualmente están desarrollados con esta herramienta), que actualmente no suelen contar con funciones multijugador en red y constituye un mercado que puede tener gran crecimiento si se aprovechan las características de las redes de próxima generación como IMS para ofrecer a los usuarios un valor añadido, así como para incentivar al resto de actores involucrados para desarrollar y desplegar aplicaciones de entretenimiento innovadoras para estos jugadores.

La identificación de los actores involucrados en la provisión de juegos multijugador en red nos va a servir a lo largo de este apartado como ejemplo para motivar las propuestas de la Tesis, enfocadas al valor que ofrecen a cada actor y las mejoras que introducen en los mecanismos de interacción entre ellos. No obstante, la mayoría de las consideraciones que se van a hacer son extrapolables a otras aplicaciones.

---

<sup>2</sup> Los datos de número de usuarios de Battle.net están obtenidos de la Wikipedia (<http://en.wikipedia.org/wiki/Battle.net>). En general, las cifras de usuarios de plataformas de juego en red son estimaciones y hay que considerar que una misma persona pueda tener varios usuarios registrados así como que esos usuarios no estén activos (la creación de usuarios es gratuita y el usuario ha podido dejar de usarla, pero sigue contabilizándose).

La *Open Mobile Alliance* (OMA), en el informe de evolución de juegos para móviles [15][15], creado dentro del grupo de trabajo *Games Services*, presenta una identificación de actores, que nos servirá de punto de partida. Los actores identificados por OMA son los siguientes: **usuarios, distribuidores de juegos, operadores móviles, vendedores de sistemas operativos y fabricantes de dispositivos**. Esta visión del mercado, representada gráficamente en la parte izquierda de la Fig. 3, responde a un modelo clásico de provisión de servicios en red donde los fabricantes de dispositivos y sistemas operativos proporcionan el terminal de usuario, el operador de red la conectividad IP y los distribuidores, de juegos en este caso, proporcionan el software y todos los servicios e infraestructura (servidores) necesarios para que los usuarios finales puedan jugar.



**Fig. 3. Actores del mercado de juegos multijugador**

En la actualidad, el modelo de provisión de servicios más popular en Internet se basa en los *mashups*, esto es, aplicaciones basadas en la composición de varios servicios prestados por terceros y accesibles a través de un API. Por ejemplo, una aplicación turística puede recomendar al usuario sitios cercanos interesantes, utilizando para ello el servicio de localización del operador móvil, un servicio de representación de mapas como *Google Maps* y una base de datos de restaurantes y cines como la que proporciona el sitio web *11870.com*.

Aplicando este modelo basado en la composición de servicios a la provisión de juegos en red proponemos un escenario como el presentado a la derecha en la Fig. 3, donde además del distribuidor de juegos se distinguen otros dos roles que extienden los identificados por OMA y que son: los **desarrolladores**, que aparecen explícitamente

y pueden ser incluso usuarios avanzados (de acuerdo a la corriente de *User Generated Content/Services*), y los **proveedores de servicios de juego**. Por otro lado, si admitimos un entorno de red de próxima generación o NGN, agnóstico en el acceso, donde la red ofrece más servicios además de la propia conectividad IP, entonces el rol de operador fijo/móvil se convierte en **proveedor de servicios de red y acceso**.

Las contribuciones de la Tesis están orientadas a soportar, desde el punto de vista técnico, el cambio de visión señalado en la Fig. 3, proporcionando mecanismos y reglas de diseño que aprovechen las ventajas de IMS y ofrezcan un valor añadido a los actores involucrados respecto del escenario actual de prestación de servicios. En concreto, los **beneficios esperados** para cada uno de los actores, extrapolables a otros tipos de aplicaciones multimedia multiusuario de alta interactividad, son los siguientes:

- **Usuarios.** Ven facilitado el acceso a las distintas aplicaciones y la interacción con sus amigos o contactos en general al prescindir de procesos de registro y alta en diversas plataformas de juego con la consiguiente segmentación de sus contactos. Al mismo tiempo, fomenta la competitividad entre los proveedores de servicios de juegos y desarrolladores, pues los usuarios dejan de estar ligados a una plataforma concreta, obteniendo aplicaciones con funciones multiusuario de valor añadido. La convergencia de servicios y el acceso ubicuo les permite acceder a las aplicaciones en cualquier momento y lugar, adaptándose el servicio o las capacidades del dispositivo en uso. La posibilidad de garantizar la calidad de servicio permite a los usuarios interactuar con la aplicación manteniendo una calidad de experiencia satisfactoria.
- **Desarrolladores de juegos/aplicaciones.** El coste de introducir funciones multiusuario en las aplicaciones puede ser reducido o incluso nulo por no necesitar desplegar infraestructura específica para su soporte, pues con IMS se pueden resolver los mecanismos básicos que precisa este tipo de aplicaciones sin desarrollos específicos en servidores de aplicación. De este modo, los pequeños desarrolladores o incluso usuarios con los conocimientos técnicos necesarios pueden entrar a competir en el mercado más fácilmente y sin intermediarios. Además, si la facturación de los servicios se realiza a través del operador de red y se generalizan los micro-pagos, los usuarios encontrarán menos dificultades para adquirir las aplicaciones al no tener que mostrar sus datos de facturación a terceros, incrementando el nivel de ventas.
- **Proveedores de servicios de juegos/aplicaciones.** Pueden centrarse en su lógica de negocio al delegar en la arquitectura IMS de los operadores las funciones básicas de comunicación entre usuarios, ofreciendo servicios

innovadores que enriquezcan los juegos y atraigan a desarrolladores y usuarios.

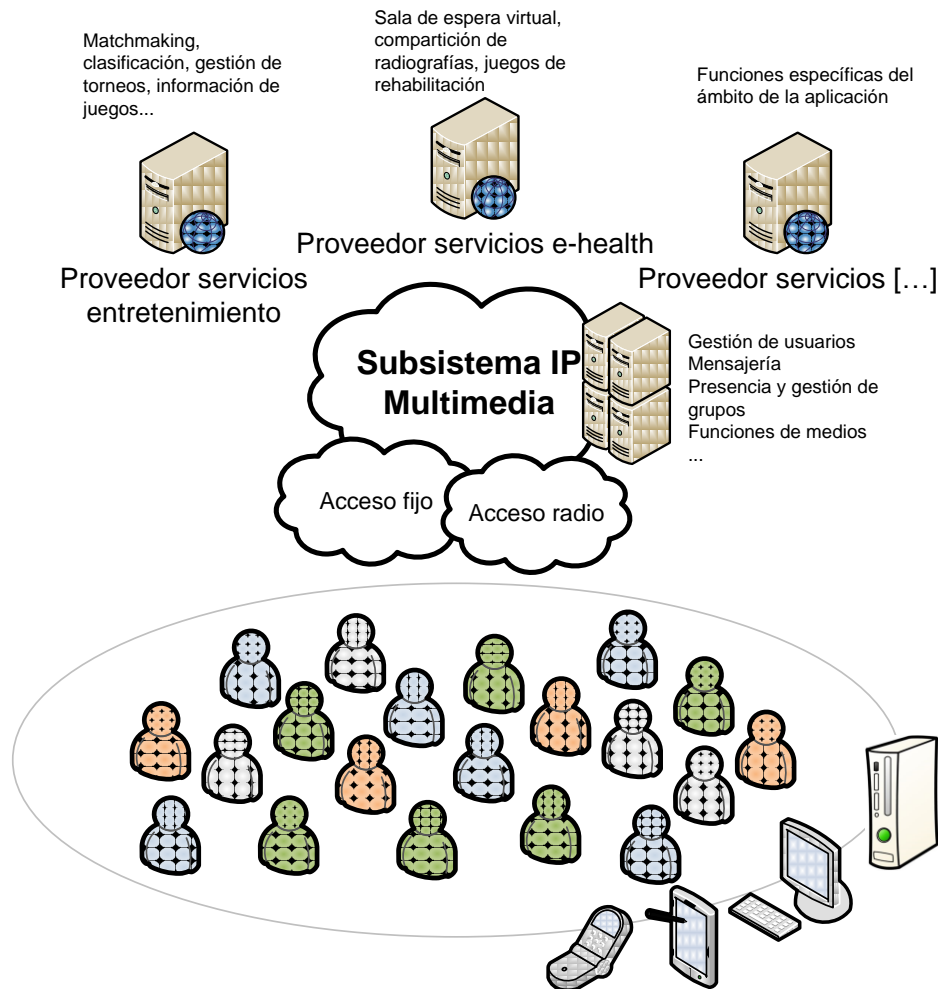
- **Distribuidores de juegos/aplicaciones.** Se benefician de los nuevos canales de distribución no físicos y el previsible incremento en la oferta de aplicaciones por las ventajas identificadas para los desarrolladores.
- **Proveedores de servicios de red y acceso.** Mayor control y optimización de la utilización de los recursos de red gracias a la orientación a sesiones de IMS. Incremento de beneficios debido a comisiones por facturación de servicios ofertados por terceros.
- **Fabricantes de dispositivos y S.O.** La introducción de nuevos mecanismos de interacción con las aplicaciones y el acceso ubicuo a las mismas anima a los usuarios a renovar o adquirir nuevos terminales con mejores prestaciones, entendidas no sólo como mayor capacidad de proceso sino como mejora de otros aspectos como autonomía, movilidad, usabilidad, etc.

### 1.3 Objetivos

Las motivaciones expresadas en el apartado anterior, a través del análisis de la situación actual del escenario de provisión de aplicaciones multimedia multiusuario de alta interactividad, nos permiten enunciar los objetivos de la Tesis.

En particular, en el apartado anterior se presentaron los problemas planteados por la situación actual de desagregación tanto de usuarios como servicios, ver Fig. 1, fruto de la utilización de las redes únicamente como medio de transporte y la consiguiente replicación de funciones por cada proveedor. Ante esta situación, el objetivo global de la Tesis es **proponer mecanismos y reglas de diseño que contribuyan a mejorar el soporte estándar que las redes IP ofrecen a las aplicaciones HIMMA, en un entorno convergente como el mostrado en la Fig. 4, basado en la arquitectura IMS con habilitadores originales para el soporte de aplicaciones HIMMA definidos en la Tesis.**





**Fig. 4. Entorno unificado de aplicaciones HIMMA con servicios comunes provistos por IMS**

De esta propuesta se pueden derivar ventajas para los desarrolladores de aplicaciones, usuarios, proveedores de servicios de red y otros actores participantes en el mercado identificados previamente, ver la Fig. 3, que pueden resumirse en:

- Disminución del coste de desarrollo de aplicaciones (protocolos estándar para todos los usuarios, infraestructura propia reducida, servicios básicos ya dados por la red)
- Facilidad de acceso a las distintas aplicaciones y mejor experiencia de uso para los usuarios
- Externalidad de red acentuada al no segregar la base de usuarios en plataformas
- Mayor facilidad para el ofrecimiento de servicios y contenidos premium (mejor oferta para el usuario, mayor confianza de los usuarios si es el operador el que factura, incremento de ventas por los proveedores y distribuidores)

Para ello se plantean como **objetivos parciales** los siguientes:

1. Profundizar en el análisis del estado de la técnica en lo relativo a las aplicaciones HIMMA y su soporte en redes IP
2. Definir un marco general para la provisión de aplicaciones HIMMA en entornos IMS, incluyendo la propuesta de módulos habilitadores específicos para dichas aplicaciones
3. Mostrar la aplicación de las propuestas de la Tesis en casos de uso de aplicaciones HIMMA profesionales y de entretenimiento

### 1.4 Fases de trabajo y metodología

A partir de los objetivos identificados anteriormente se han establecido las fases de trabajo representadas en la Fig. 5

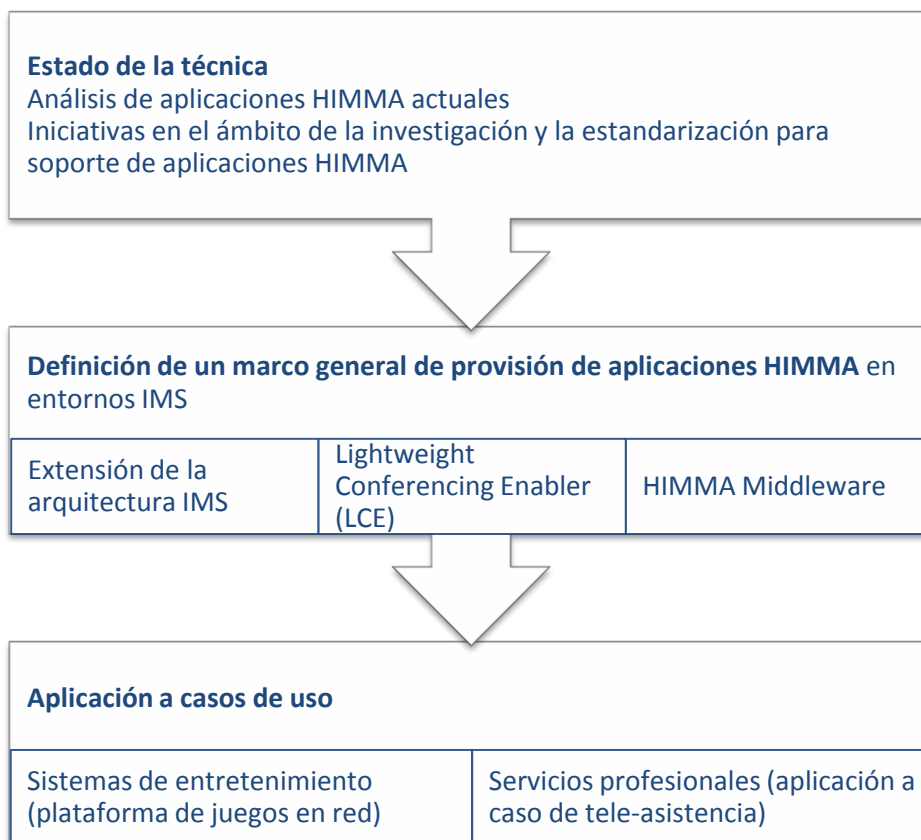


Fig. 5. Fases de trabajo de la Tesis

En primer lugar se sitúa el estudio del estado de la técnica, centrado en el análisis de aplicaciones multimedia multiusuario de alta interactividad (HIMMA) disponibles actualmente, así como las iniciativas de soporte para este tipo de aplicaciones en IMS.

La segunda fase consiste en la definición de un marco general de provisión de aplicaciones HIMMA en entornos IMS. Este marco genérico responde a la motivación de la Tesis de extender la arquitectura IMS, manteniendo el uso de tecnologías estándar, para ofrecer un mejor soporte a aplicaciones HIMMA. Se identificarán las extensiones siguiendo dos métodos o enfoques complementarios: extender la funcionalidad del lado servidor mediante el concepto de *habilitador* y extender la funcionalidad del lado de cliente mediante un *middleware*. El denominado habilitador ligero de gestión de actividades (LCE) sigue el primer enfoque y se encarga de gestionar actividades multiusuario y mantener su estado, mientras que el middleware HIMMA sigue el segundo enfoque y se encarga de proporcionar un conjunto de funciones que facilitan el desarrollo de aplicaciones HIMMA en entornos IMS sin necesidad de infraestructura adicional a la provista por IMS.

Para la definición de los elementos anteriores se utilizan diagramas de secuencia para ilustrar la comunicación con los nodos de IMS, diagramas de clase para el diseño software y esquemas de las estructuras XML utilizadas.

Los conceptos presentados en el LCE y el middleware se implementan y validan de forma práctica mediante tecnologías estándar de desarrollo de aplicaciones en IMS. Concretamente, los desarrollos se implementan en lenguaje Java, utilizando el estándar JSR 289 (*SipServlets API*) para el gestor ligero de actividades (LCE) y JSR 281 (*IMS Services API*) para el middleware HIMMA. En ambos casos, los desarrollos se prueban en una arquitectura IMS emulada (Ericsson SDS con servidor XDMS 1.0 emulado) y el servidor de aplicaciones abierto Sun Sailfin.

En la última fase se estudia la aplicación del marco propuesto de soporte a aplicaciones HIMMA a dos escenarios relevantes (servicios de juegos y de asistencia sanitaria), evaluando de forma crítica las ventajas que aporta a cada uno de ellos y en qué medida cumplen las expectativas planteadas en el apartado anterior. En el primer escenario se han seleccionado una serie de casos de uso relevantes a través del análisis de las fases sucesivas de la experiencia de un jugador durante la interacción con una plataforma de juego en red. En el escenario de asistencia sanitaria se estudia la aplicación práctica de los conceptos presentados en el proyecto AmIVital (ver apartado siguiente).

## 1.5 Proyectos relacionados

El trabajo de investigación de la Tesis está relacionado con diversos proyectos de investigación en los que ha participado el autor y que se enmarcan en el ámbito de redes de próxima generación (NGN). Las siguientes secciones, ordenadas cronológicamente, recogen una breve descripción de cada proyecto relacionado.

### 1.5.1 Acción COST 290 (Wi-QoST)

La acción COST 290, denominada Wi-QoS (*Traffic and QoS Management in Wireless Multimedia Networks*), se ha desarrollado durante los años 2004 a 2008 con el objetivo general de mejorar los aspectos de calidad de servicio, gestión de tráfico y tarificación de los servicios multimedia de las futuras redes móviles de comunicación.

En concreto, la participación durante los inicios del proyecto dentro del grupo de trabajo número 2, dedicado a “movilidad e interoperabilidad”, sirvió como primera toma de contacto con IMS como red multimedia de próxima generación con soporte de calidad de servicio, seguida después por otros proyectos más directamente relacionados con las contribuciones de la Tesis.

### 1.5.2 CASERTEL-NGN

CASERTEL-NGN (“Caracterización de nuevos servicios de telecomunicación desde el punto de vista de la calidad de servicio e impacto en las reglas de diseño de redes de próxima generación”) es un proyecto del Plan Nacional I+D+i del Ministerio de Educación y Ciencia (ref. TSI2005-07306-C02-01), desarrollado durante los años 2005 a 2008.

El objeto del proyecto es caracterizar nuevos y futuros servicios con requisitos de calidad de servicio, y el trabajo de la Tesis relacionado con el mismo se ha centrado en el análisis del tráfico de red de una de las plataformas de servicios de juegos más utilizada actualmente (*Xbox Live*).

### 1.5.3 AmIVital

AmIVital (“Entorno personal digital para la salud y el bienestar”) es un proyecto del Programa nacional Ingenio 2010 (2007-2010), cuyo objetivo general es el desarrollo de una nueva generación de tecnologías y herramientas TIC para el modelado, diseño, operación, e implementación de dispositivos y sistemas de inteligencia ambiental

(AmI), cuyo fin es la provisión de servicios y soportes personales de la vida independiente, el bienestar y la salud.

Se trata de construir un espacio tecnológico que facilite el desarrollo del concepto europeo del AAL (*Ambient Assisted Living*), mediante el diseño de aplicaciones y modelos de negocio para un sector emergente de gran futuro, dirigido a cubrir necesidades sociales de primer orden. Se pretende implementar una serie de sistemas, que sirvan para desarrollar aplicaciones de soporte a la vida independiente y a la movilidad de las personas mayores; monitorización y control de personas con dolencias crónicas; prestación de servicios de integración social y laboral por medios accesibles; ayuda a familiares y apoyo informal a la vida dependiente; prestación de servicios para fomentar el estilo de vida saludable -incluidos los que faciliten la práctica deportiva-; herramientas de gestión de toda esta constelación de servicios para el entorno profesional, y la integración de proveedores de servicios independientes.

AmIVital no sólo trata de desarrollar productos concretos de utilización inmediata, sino configurar una plataforma tecnológica, que incluya componentes normalizados (dispositivos, redes y software), y permita la creación sencilla de servicios, adaptados a las diferentes necesidades de cada persona y sus situaciones concretas.

El trabajo de investigación llevado a cabo en la Tesis tiene relación directa con este proyecto, concretamente en los aspectos de arquitectura de red, donde se ha propuesto utilizar la arquitectura multimedia IMS, potenciar el uso de sus habilitadores básicos de servicios y diseñar habilitadores avanzados con el fin de facilitar la creación de las aplicaciones de soporte al bienestar contempladas en el marco del proyecto. En concreto, el habilitador ligero de gestión de actividades (LCE) especificado y desarrollado en el ámbito de la Tesis es una pieza clave para los servicios de videoconferencia, pizarra compartida y sala de espera virtual, presentes, entre otros, en el caso de uso de teleconsulta que se describe en la sección 6.2. Asimismo, el plan de trabajo de AmIVital incluye el soporte para el ofrecimiento de juegos, en principio orientados al entrenamiento mental, rehabilitaciones o simples divertimentos para los usuarios objetivo (habitualmente mayores de edad).

#### **1.5.4 VISION**

VISION (“Comunicaciones de vídeo de nueva generación”) es un proyecto perteneciente al Programa nacional Ingenio 2010 (2007-2010), que tiene como objetivo desarrollar una nueva generación de sistemas de comunicación que permitan transmitir la sensación de presencia real, de manera que las personas separadas por

grandes distancias perciban la sensación de estar físicamente reunidas en un mismo lugar.

El desarrollo de los sistemas de comunicación con sensación de presencia, contribuirá en gran medida a la reducción del uso del transporte y de los desplazamientos reales, sustituyéndolos por comunicaciones virtuales que ofrezcan sin embargo las mismas sensaciones que un encuentro presencial. Estas nuevas capacidades en los sistemas de comunicación abren las puertas a nuevas formas de relación tanto profesional como personal. Posibilitará una mejora en las relaciones sociales y familiares ya que se favorecerán los encuentros inter-personales incluso entre personas separadas por grandes distancias.

Para conseguir este objetivo, el proyecto se centra en la investigación de las tecnologías que permitan el desarrollo de los nuevos sistemas de comunicación, entre ellas:

- Tecnologías de captura de la realidad
- Tecnologías avanzadas de análisis y procesado vídeo
- Tecnologías avanzadas de análisis y procesado de audio
- Tecnologías avanzadas de comunicación
- Tecnologías de presentación de la realidad
- Factores Humanos

El trabajo de investigación de la Tesis relacionado con este proyecto se enmarca, al igual que en el proyecto AmIVital, en los aspectos de arquitectura de red y soporte de aplicaciones multimedia. En este caso, los sistemas con sensación de presencia son especialmente exigentes en recursos de red, al precisar la gestión conjunta de múltiples flujos de vídeo, elevado ancho de banda y muy baja latencia, además de servicios que ayuden a mejorar la interacción y comunicación entre los usuarios.

## 1.6 Estructura de la memoria

El resto de la memoria está estructurado como sigue. El Capítulo 2 presenta el estado de la técnica relativo a la provisión de aplicaciones multimedia multiusuario de alta interactividad. Se introduce igualmente la arquitectura IMS, el grado de soporte que ofrece a aplicaciones HIMMA de forma estándar y las iniciativas en el ámbito de la investigación para mejorar el soporte a este tipo de aplicaciones en entornos IMS.

El Capítulo 3 presenta un marco genérico de soporte a aplicaciones HIMMA basado en la arquitectura IMS y los principios de diseño que han guiado su definición. La arquitectura describe nuevos elementos lógicos en la red, concretamente el habilitador

ligero de actividades o LCE (*Lightweight Conferencing Enabler*) en la parte de servidor, y el middleware de soporte aplicaciones HIMMA en la parte cliente.

Los Capítulos 4 y 5 especifican, respectivamente, el habilitador LCE y el middleware de soporte a aplicaciones HIMMA que fueron introducidos en el Capítulo 3, detallando la funcionalidad y los flujos de mensajes involucrados en cada uno.

El Capítulo 6 presenta la aplicación de las contribuciones anteriores a dos casos de uso que sirven de validación de los conceptos presentados. En primer lugar, la aplicabilidad de la arquitectura extendida propuesta para el ofrecimiento de un servicio de entretenimiento basado en el juego en red. En segundo lugar, la aplicación de las contribuciones a servicios profesionales, tomando como ejemplo un escenario de asistencia médica remota. Adicionalmente se presenta un conjunto de consideraciones sobre la escalabilidad de la solución propuesta.

Finalmente, en el Capítulo 7 se presentan las conclusiones y líneas de continuación de este trabajo.





## **2. Estado del arte**

Este capítulo presenta el estado de la técnica relativo a la provisión de aplicaciones multimedia multiusuario de alta interactividad, centrándose en dos conjuntos de aplicaciones significativas: las orientadas al entretenimiento interactivo, en particular los juegos en red, y las orientadas a la colaboración en entornos profesionales, como trabajo colaborativo, educación a distancia, asistencia sanitaria remota, etc. Seguidamente, se introduce la arquitectura IMS, el grado de soporte que ofrece a aplicaciones HIMMA de forma estándar y las iniciativas en el ámbito de la investigación para mejorar el soporte a este tipo de aplicaciones en entornos IMS.

### **2.1 Aplicaciones multimedia multiusuario de alta interactividad (HIMMA)**

#### **2.1.1 Aplicaciones y servicios de juego en red**

La industria del videojuego lidera actualmente el mercado del entretenimiento gracias a los avances tecnológicos en calidad gráfica y conectividad, incluso superando la industria audiovisual tradicional ([8] y [16]). La denominada era de la alta definición ha impulsado la calidad visual de los juegos, contribuyendo a mejorar la experiencia de los jugadores dentro de los mundos virtuales donde juegan. Adicionalmente, los accesos de banda ancha han permitido la introducción de servicios de valor añadido en las plataformas de juegos en red. Los juegos pueden enriquecerse con audio y vídeo en tiempo real. Además, los jugadores tienen acceso a servicios de comunidad como estado de presencia de los amigos, mensajería, funciones de búsqueda de partidas, campeonatos en red, clasificaciones, etc. Las facilidades de interactividad y comunicación entre grupos de usuarios que permiten las redes de banda ancha se pueden incorporar también en otros servicios de entretenimiento como la televisión, pasando del servicio tradicional de difusión de programas a servicios de televisión

interactiva social, donde, por ejemplo, un grupo de amigos que están en sitios diferentes pueden comunicarse entre sí mediante voz, texto, etc. mientras ven un programa de televisión en sus terminales móviles.

Centrándonos en los juegos en red como caso más relevante, las plataformas disponibles en el mercado actualmente ofrecen sus servicios apoyándose en la conectividad IP provista por Internet. Los denominados servidores de juegos proporcionan todas las funciones requeridas, incluyendo registro de usuarios, almacenamiento de listas de amigos, gestión del estado de presencia, etc. La infraestructura de un proveedor de servicios de juegos puede llegar a ser muy costosa, especialmente para los juegos más exigentes (p.ej. juegos multijugador masivos y juegos muy interactivos). Por otro lado, la incompatibilidad entre plataformas y consolas de juego impide la interacción entre jugadores de diferentes sistemas de juego y multiplica el coste de desarrollo de nuevos juegos, como ya han expresado importantes desarrolladoras [3].

Los juegos multijugador en red y plataformas de juego en red más conocidos del mercado son:

- *World of Warcraft* (WoW) para PC, el principal juego de rol en red multijugador masivo (MMORPG) con más de 10 millones de usuarios, representando aproximadamente un 60% del mercado de este tipo de juegos.
- *Xbox Live*, la plataforma que soporta las funciones de red de la consola Microsoft *Xbox 360* y los juegos *Games for Windows Live* para ordenador, y cuenta con alrededor de 30 millones de usuarios;
- *Sony Playstation Network* (PSN), proporciona las funciones de red para las consolas *Sony Playstation 2*, *Playstation 3* y *Playstation Portable* (PSP);
- *Nintendo Wi-Fi Connection* (WFC), soporta las funciones en red para los juegos de la consola *Nintendo Wii* y la familia de consolas portátiles DS (*DS*, *DS Lite* y *DSi*).

Atendiendo al volumen de usuarios cabe destacar, según datos de la consultora comScore [14] un conjunto de más de 250 millones de usuarios que juegan habitualmente a los denominados juegos Web (o juegos Flash, pues suelen estar desarrollados con esta herramienta). Estos juegos empiezan a incorporar funciones multijugador y, del mismo modo que ocurre con los juegos para móviles, es un mercado con gran potencial.

La Fig. 6 muestra el volumen de usuarios (en millones) en el mundo de los juegos o plataformas de juego mencionadas anteriormente. Las cifras para los tres sistemas de entretenimiento doméstico (PS3, Xbox 360 y Wii) de la generación actual y las dos

consolas portátiles (PSP y DS) se refieren a los millones de unidades vendidas según *vgchartz.com*, fuente habitual de los medios. Por supuesto, no implica que todos los usuarios hagan uso de las funciones de red disponibles en las citadas consolas. Por otro lado, el número de usuarios de juegos para móviles (no necesariamente multijugador), es una extrapolación aproximada a partir del informe de comScore [17], que destaca que sólo un 8% de los usuarios de telefonía de USA utiliza su móvil para jugar, y del informe de [18] que recoge el volumen mundial de terminales multimedia (ligado a la capacidad de ejecutar juegos).

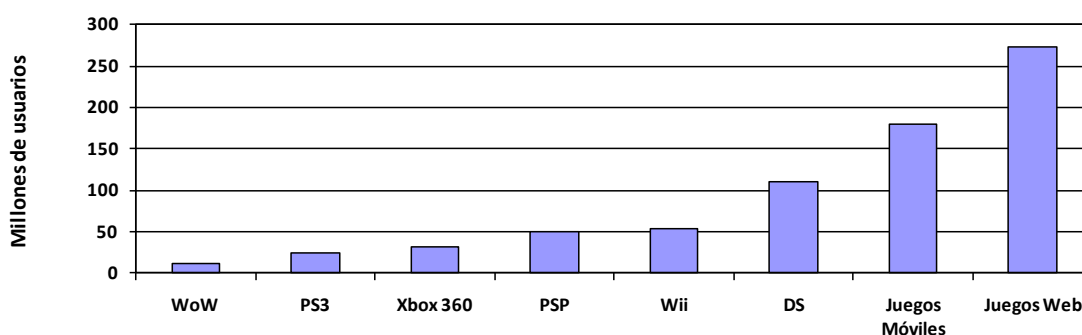


Fig. 6. Volumen de usuarios mundial de plataformas de juego en red destacadas

Las redes de próxima generación pueden enriquecer todo tipo de juegos con servicios de valor añadido y facilitar el acceso a los jugadores. Del mismo modo, puede facilitar a los desarrolladores la creación de nuevos juegos y servicios, potenciando nuevos modelos de negocio basados en micro transacciones, descarga de contenidos y servicios de suscripción, modelos que se consideran clave para el crecimiento de este sector según numerosos actores [19].

### 2.1.2 Aplicaciones y servicios profesionales

Las aplicaciones y servicios multimedia multiusuario de alta interactividad en el ámbito profesional vieron su impulso con las soluciones de gestión colaborativa de proyectos a mediados de los años 90, motivado principalmente por la globalización y la mayor distribución geográfica de los equipos de trabajo que conlleva. Así, las herramientas de soporte al trabajo en grupo de forma remota han ido evolucionando e incorporan diversas funciones como mensajería instantánea, conferencia de audio y vídeo, pizarra electrónica, gestión de documentos, edición simultánea de documentos, sistemas de encuestas, etc.

### 2.1.2.1 Soluciones basadas en ITU-T T.120 y H.323

La recomendación T.120 de ITU-T agrupa un conjunto de protocolos y servicios de comunicación y aplicación que proporcionan soporte para comunicaciones de datos multipunto en tiempo real. Las aplicaciones más conocidas de trabajo colaborativo como *Lotus Sametime*, *Microsoft Netmeeting* o las soluciones de videoconferencia de *Polycom* y *Cisco* se basan en estas recomendaciones para ofrecer servicios de compartición de aplicaciones, conferencias de audio y vídeo, pizarra electrónica y otras funciones, garantizando al tiempo cierta interoperabilidad (normalmente los fabricantes hacen extensiones propietarias a las recomendaciones por lo que no todas las funciones interoperan correctamente).

Algunos servicios especificados por estas recomendaciones incluyen **pizarra virtual** (T.126), **transferencia de ficheros multipunto** (T.127), **compartición de aplicaciones** (T.128), **mensajería instantánea** (T.134) y **control remoto de dispositivos** (T.136) entre otros.

Es frecuente que las aplicaciones que ofrecen estos servicios utilicen además los protocolos de señalización para el control de sesiones multimedia en redes de paquetes definidos en el conjunto de recomendaciones H.323 de ITU-T. En concreto, H.323 recoge un conjunto de protocolos – ilustrados en la Fig. 7 – para el establecimiento y control de llamadas, transporte multimedia y gestión de ancho de banda para comunicaciones de audio, voz y datos punto a punto y punto a multipunto.

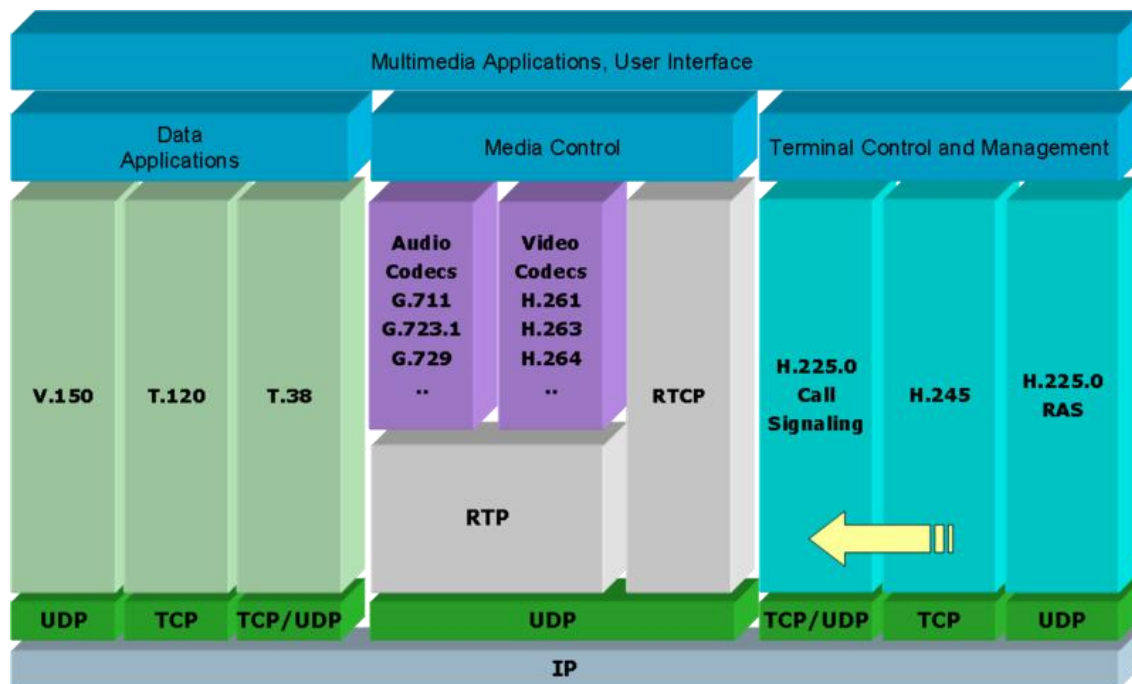


Fig. 7. Torre de protocolos de H.323

Históricamente, las aplicaciones orientadas al uso profesional hacen uso de las recomendaciones de H.323 para el establecimiento de comunicaciones entre múltiples usuarios, ofreciendo los servicios de datos especificados en las recomendaciones T.120. Actualmente, sin embargo, la tendencia en este tipo de aplicaciones tiene una doble vertiente: la de aplicaciones de escritorio basadas en el protocolo de inicio de sesión SIP definido por IETF, como *Ekiga*, y la de aplicaciones ligeras basadas en web (siguiendo habitualmente el paradigma de “*cloud computing*” donde las comunicaciones son totalmente propietarias), como las soluciones colaborativas de *Google* analizadas en este mismo apartado.

Las soluciones de comunicación de los fabricantes tradicionales siguen basándose en ofrecer arquitecturas (servidores y hardware/software cliente) H.323 para garantizar la interoperabilidad con los sistemas antiguos y reutilizar el conocimiento de la tecnología, si bien en los últimos cinco años se observa cada vez más que se destaca la compatibilidad con SIP como ventaja competitiva. Al mismo tiempo, también se apuesta por el modelo de SaaS (*Software as a Service*) donde el proveedor de los servicios de comunicación mantiene la infraestructura necesaria, proporciona una interfaz web a los mismos y cobra en función de su utilización.

### 2.1.2.2 Soluciones basadas en web

Las soluciones de trabajo colaborativo basadas en web permiten la visión, creación y edición de distintos tipos de información tales como documentos, hojas de cálculo, presentaciones, diagramas, calendarios, mapas y otros contenidos a través de un navegador web.

Google Docs es una de las más conocidas y permite la colaboración en tiempo real con otros usuarios. Se integra además con otras soluciones de Google como Talk para intercambiar mensajes instantáneos o el servicio de videollamada (que requiere un plugin en el navegador). Recientemente se anunciaba la herramienta Google Wave, una plataforma de colaboración web con servidor de software libre que permite la creación de temas de discusión donde los usuarios pueden añadir texto, vídeos, imágenes y otros contenidos (mapas, encuestas, etc.), añadir anotaciones, enlazar otros temas o revisar el historial de cambios en tiempo real.

Centrado en la edición de documentos existen otras herramientas basadas en web con objetivos similares a Google Docs como Zoho, de la empresa estadounidense AdventNet u Office Live de Microsoft. Igualmente, existen portales web de ámbitos más específicos que proporcionan herramientas de colaboración para gestión de proyectos, edición de código fuente, tele-educación, gestión de agendas, etc.

## 2.2 El Subsistema IP Multimedia (IMS)

El objetivo general de la Tesis es contribuir al soporte de las aplicaciones HIMMA como las mencionadas anteriormente, centrándose en entornos de Red de Próxima Generación (*Next Generation Network* o NGN) y, más concretamente, en uno de sus principales exponentes: el **Subsistema IP Multimedia o IMS** [4] (*IP Multimedia Subsystem*).

El IMS es la propuesta de 3GPP [20] (*Third Generation Partnership Project*) para la creación de una red multimedia *Todo IP* capaz de prestar tanto los servicios tradicionales (telefonía, principalmente) como nuevos servicios multimedia de valor añadido (videoconferencia, televisión bajo demanda, mensajería instantánea, etc.). Dado que está basado en estándares de Internet como SIP [21] (*Session Initiation Protocol*) o RTP [22] (*Real Time Protocol*), la implementación del IMS se ve favorecida por la madurez del conocimiento de tecnologías de Internet. Cabe destacar que el IMS constituye un estándar ampliamente aceptado en la industria (especialmente por los operadores móviles), si bien su disponibilidad comercial para el público es todavía escasa.

El IMS fue introducido por el *3rd Generation Partnership Project* (3GPP) en las especificaciones de la Release 5 de UMTS (*Universal Mobile Telecommunication System*), como componente clave para la provisión de servicios multimedia en redes móviles 3G evolucionadas. No obstante, la arquitectura IMS es independiente de la red de acceso y existen diversos esfuerzos de estandarización, no sólo por parte de 3GPP, para garantizar la convergencia de redes manteniendo IMS como núcleo de gestión de servicios. Así, las siguientes revisiones de la especificación de IMS definen la integración con redes fijas, gracias al trabajo de ETSI TISPAN (*Telecoms and Internet converged Services and Protocols for Advanced Networks*), y nuevas tecnologías inalámbricas como Wimax y LTE (*Long Term Evolution*).

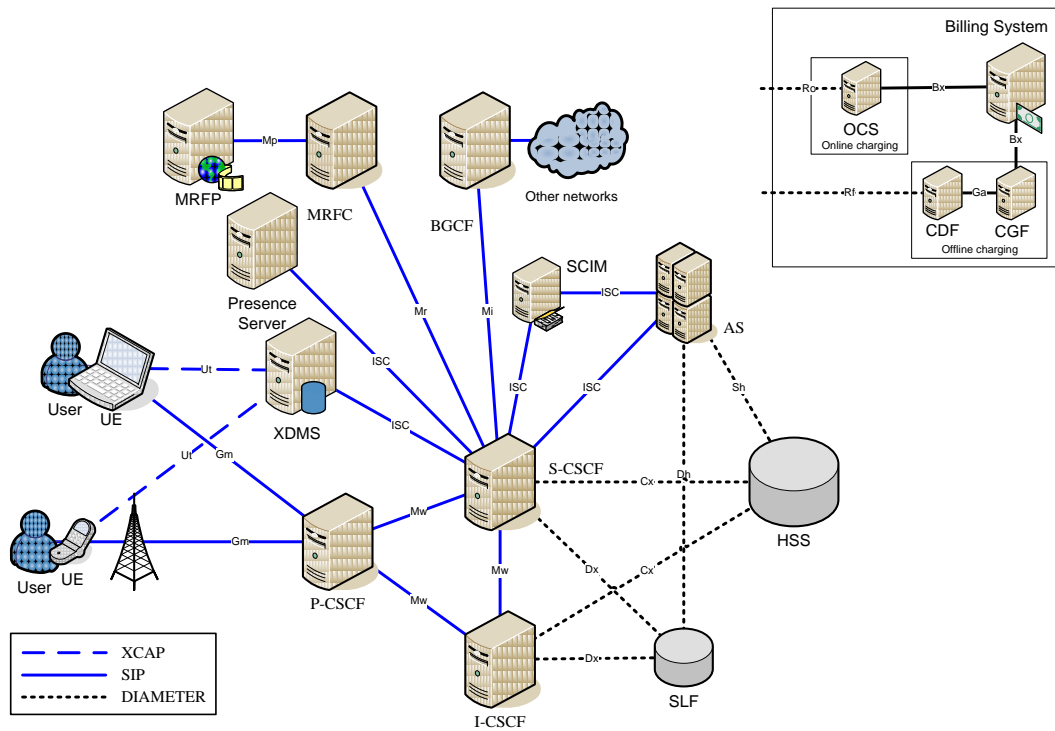
### 2.2.1 Arquitectura general de IMS

Los operadores que desplieguen una arquitectura IMS deben contar al menos con el denominado “núcleo IMS” (*IMS core*) formado por el *Home Subscriber Server* (HSS) y los intermediarios de señalización, que reciben el nombre de *Call State Control Function* (CSCF). El HSS es la base de datos de perfiles de usuario, credenciales de autenticación y condiciones de servicio, accesible mediante interfaces basados en el protocolo DIAMETER. Por su parte, los CSCF son servidores de señalización SIP (*Session Initiation Protocol*) que soportan las funciones de registro, localización y

encaminamiento definidas en el estándar. El estándar define tres tipos de CSCF, especializados en distintas tareas:

- P-CSCF (*Proxy-CSCF*): Servidor proxy especializado en el diálogo directo con los terminales, constituyendo la puerta de entrada de los usuarios al IMS. La dirección del P-CSCF la obtiene el terminal de usuario al solicitar el acceso al IMS.
- I-CSCF (*Interrogating-CSCF*): Proporciona funciones de localización en el IMS mediante su capacidad para realizar consultas a la base de datos central (HSS). Constituye el punto de contacto para alcanzar a los usuarios (propios o *roamers*) de una red IMS. El I-CSCF puede, opcionalmente, facilitar la ocultación de detalles en sesiones IMS entre operadores.
- S-CSCF (*Serving-CSCF*): Especializado en cursar las peticiones de registro y mantener el estado de las sesiones SIP de los usuarios del IMS. Así mismo, es el elemento donde se identifican los servicios a ejecutar, sea en el propio S-CSCF o en un servidor de aplicaciones externo. Cada usuario registrado en el IMS tiene asignado un S-CSCF.

Existen otros elementos que proporcionan funcionalidad adicional, como el servidor de gestión de documentos XML (XDMS, clave para las funciones de presencia y gestión de grupos), el servidor de tratamiento de medios (*Media Resource Function* o MRF), el servidor de aplicaciones (para funciones específicas de cierto servicio) y diversas pasarelas para interconexión con otras redes, todos ellos mostrados en la Fig. 8 y que pueden ordenarse lógicamente en capas como ilustra la Fig. 9.



**Fig. 8. Esquema general de la arquitectura IMS**

Conforme avance la exposición iremos profundizando en los aspectos específicos de la arquitectura que sean relevantes en la solución propuesta en esta Tesis, indicando en todo momento si los componentes y mecanismos de interacción forman parte del estándar (siendo susceptibles de estar disponibles en cualquier entorno IMS), si son componentes específicos de la solución añadidos a la arquitectura o si son propuestas de modificación o ampliación de la señalización estándar para soportar servicios de valor añadido.



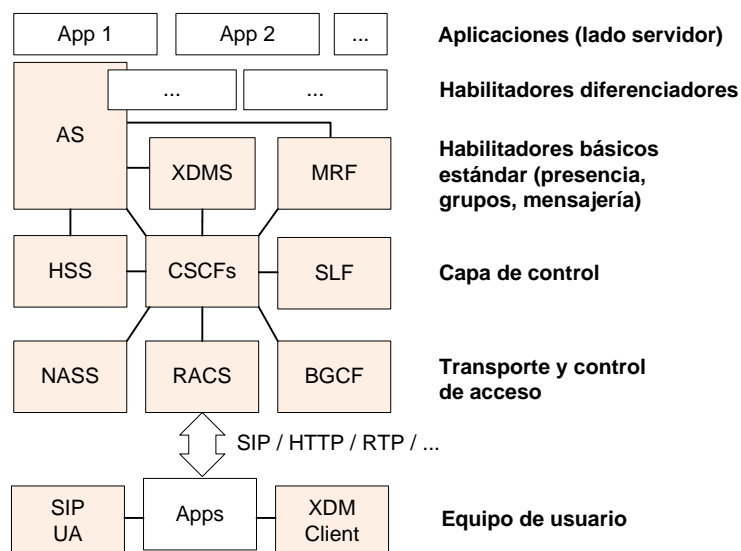


Fig. 9. Arquitectura de IMS por capas

### 2.2.2 Servicio de conferencias audiovisuales multimedia

El servicio de conferencias multimedia IMS permite comunicaciones de voz, vídeo o texto full-duplex entre varios usuarios. Este servicio se presta de forma estándar de manera centralizada acorde a la especificación de 3GPP [23]. El nodo central es el MRF (Media Resource Function) que incluye tanto funciones de señalización SIP (MRF Controller) como de tratamiento de medios audiovisuales (MRF Processor). El controlador es un foco (*focus*) de señalización SIP, mientras que el procesador realiza las mezclas del audio y vídeo empaquetado en los flujos RTP de los participantes con el fin de crear y distribuir el mosaico de vídeos típicos de una videoconferencia con múltiples usuarios.

En un escenario típico, el usuario que quiera crear una conferencia envía un INVITE a la dirección de la factoría de conferencias (una URI conocida), la cual responderá con una URI única que identifique a la conferencia creada. El resto de usuarios deben establecer sesión de audio o vídeo con esa URI única para ser incluidos en la conferencia.

### 2.2.3 Servicio Pulsar para Hablar (Push to Talk)

El servicio de pulsar para hablar, *Push To Talk Over Cellular*, también conocido por sus acrónimos PTT o PoC, es un servicio de tipo *walkie-talkie* que permite comunicaciones de voz half-duplex uno a uno o uno a varios. Está estandarizado por la Open Mobile Alliance (OMA) y sigue una arquitectura cliente/servidor. Los servidores de PoC incluyen las funciones de establecimiento de sesión, control de sala y

distribución de audio. Los clientes deben soportar la especificación de PoC, incluyendo los siguientes protocolos: SIP para el establecimiento de sesión, TBCP (Talk Burst Control Protocol, específico de PoC) para el control de sala y RTP para el transporte de medios. Además, para crear y gestionar grupos de usuarios de PoC es necesario el protocolo XCAP.

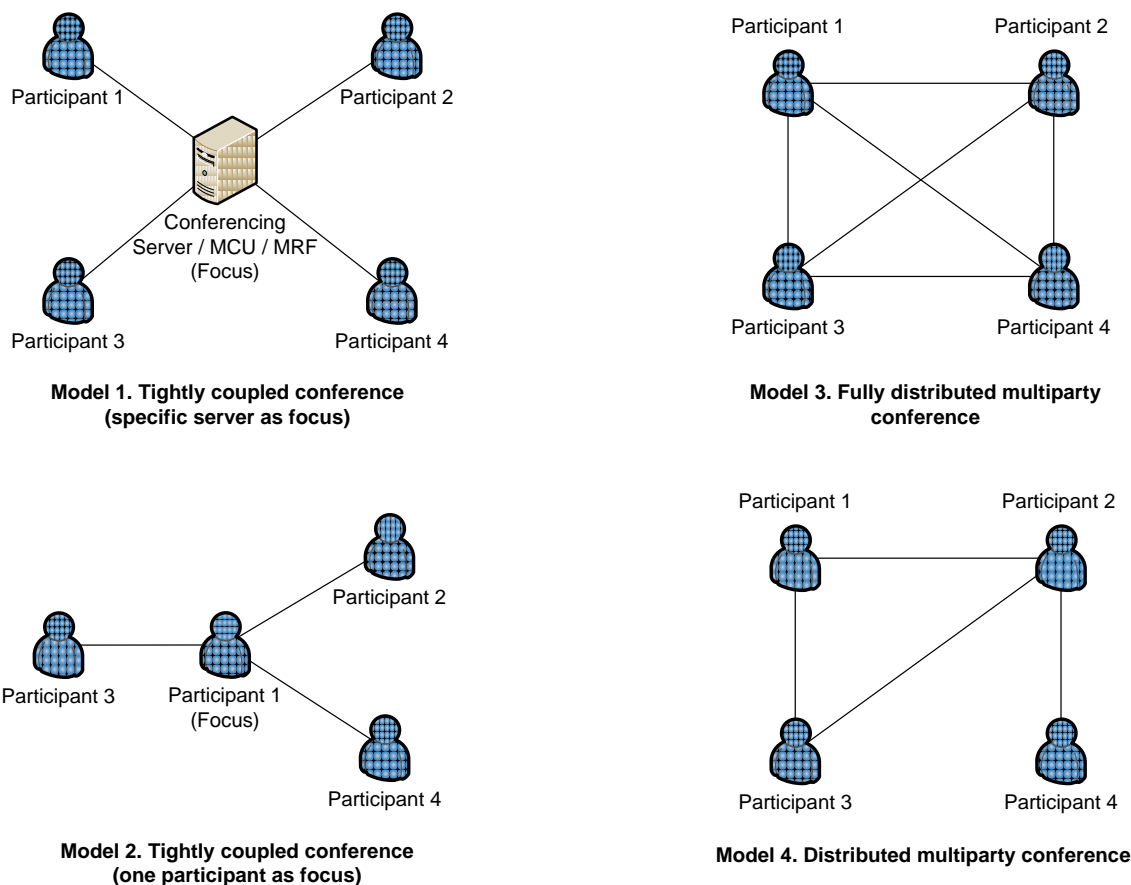
Para establecer una comunicación PoC uno a uno, el usuario que inicia la sesión invita directamente al otro usuario. El intermediario S-CSCF, que inspecciona las cabeceras de los mensajes SIP para aplicar las reglas de encaminamiento definidas, enviará el mensaje a través del servidor PoC. En comunicaciones uno a varios, el primer paso es crear un grupo PoC en el servidor XDM especificando el conjunto de participantes y/o los criterios para permitir participar a usuarios que no estén en la lista. La comunicación puede ser iniciada de forma automática por el servidor PoC, que enviará las peticiones de inicio de sesión pertinentes a todos los usuarios del grupo. Los usuarios que conozcan la URI de servicio de esa comunicación PoC también pueden añadirse iniciando sesión con ella siempre y cuando sean aceptados por cumplir las políticas de acceso del grupo.

### 2.3 Modelos de comunicaciones multiusuario

El protocolo de inicio de sesión SIP, adoptado por IMS como protocolo de señalización para el establecimiento de sesiones multimedia, modela únicamente sesiones entre dos entidades SIP, no existiendo el concepto de “sesión multiusuario”. La concepción de sesión es similar al de llamada en el servicio telefónico tradicional, que permite a un usuario llamar a otro marcando su número de teléfono: si el usuario desea establecer una llamada con múltiples usuarios ha de marcar un número especial perteneciente a un gestor de conferencias y solicitar a los participantes que llamen a ese mismo número, encargándose el gestor de conferencias de mezclar el audio de los distintos usuarios para que todos puedan hablar entre ellos.

Aunque en SIP las sesiones pueden ser de cualquier tipo, pues son independientes del tipo de medio, el uso más extendido es para el establecimiento de llamadas de voz sobre IP (VoIP) y video, tanto entre dos usuarios como entre múltiples usuarios. Para el establecimiento de conferencias entre múltiples usuarios (siguiendo la recomendación RFC 4353[24]), la arquitectura habitual es del tipo cliente-servidor donde todos los usuarios se comunican con el denominado “foco” de la conferencia, que puede ser un servidor específico (como el MRF de IMS, que desempeña el rol de foco en las conferencias multimedia), o puede ser un propio participante de la conferencia cuyo dispositivo incorpore la funcionalidad requerida (p.ej. mezcla de audio y vídeo para

conferencias audiovisuales), de acuerdo a los modelos 1 y 2, respectivamente, presentados en la Fig. 10.



**Fig. 10. Diferentes modelos de establecimiento de una conferencia con múltiples usuarios**

Sin embargo, cada vez es más habitual recurrir a soluciones basadas en arquitecturas P2P para mejorar la escalabilidad, rendimiento y fiabilidad de las aplicaciones multimedia multiusuario, al tiempo que redundan en un menor coste al prescindir de infraestructura específica de soporte. El modelo 2 mostrado en la Fig. 10 ya prescinde de servidores específicos, pero la dependencia de todos los usuarios con el foco central, responsable de mantener el estado de la conferencia, le resta fiabilidad y escalabilidad. Los modelos 3 y 4 presentan topologías más descentralizadas que no dependen de un único participante para mantener el estado de la aplicación a costa de aumentar la complejidad en las relaciones entre los participantes, que ya no tienen que establecer una única sesión con el foco sino gestionar un conjunto de sesiones con el resto de participantes.

Dentro del ámbito de la estandarización, el grupo de trabajo XCON (*Centralized Conference*) del IETF aborda en [25] la definición de un marco de gestión de conferencias centralizada que extiende la mencionada RFC 4353 para contemplar protocolos de señalización alternativos a SIP y más tipos de contenido además de audio y vídeo, así como proporcionar mecanismos de control de sala (muy orientado a conferencias donde sólo un usuario o grupo de usuarios tienen la palabra en cierto momento). Por otro lado, la Universidad de Nápoles ha presentado el modelo DCON (*Distributed Conference*), [26] y [27], en vías de estandarización, que define un mecanismo para la creación de conferencias basadas en la orquestación de “islas” (conjuntos de usuarios) gestionadas por servidores del tipo propuesto por XCON, consiguiendo mejores prestaciones en conferencias con decenas de usuarios. El modelo propuesto por DCON, presentado en la Fig. 11 al lado del modelo XCON, se denomina modelo en cascada, y aunque es distribuido al contar con varios servidores coordinados entre ellos, no es P2P y requiere infraestructura específica en los servidores para su soporte.

Los esfuerzos de estandarización expuestos anteriormente en el marco de conferencias multiusuario están muy orientados a aplicaciones de voz y vídeo, aplicables habitualmente a la creación de reuniones virtuales o la impartición de clases a distancia, donde cierto usuario tiene el rol de moderador y permite asignar turnos de participación a cada usuario, de tal forma que el audio y vídeo generado por ese usuario llegue a todos los demás. En este sentido, las arquitecturas de creación de conferencias como los citados XCON o DCON pretenden resolver el problema de distribución de contenido (voz o vídeo) entre un conjunto de usuarios, añadiendo una capa de control que permita aplicar ciertas políticas a esa distribución.

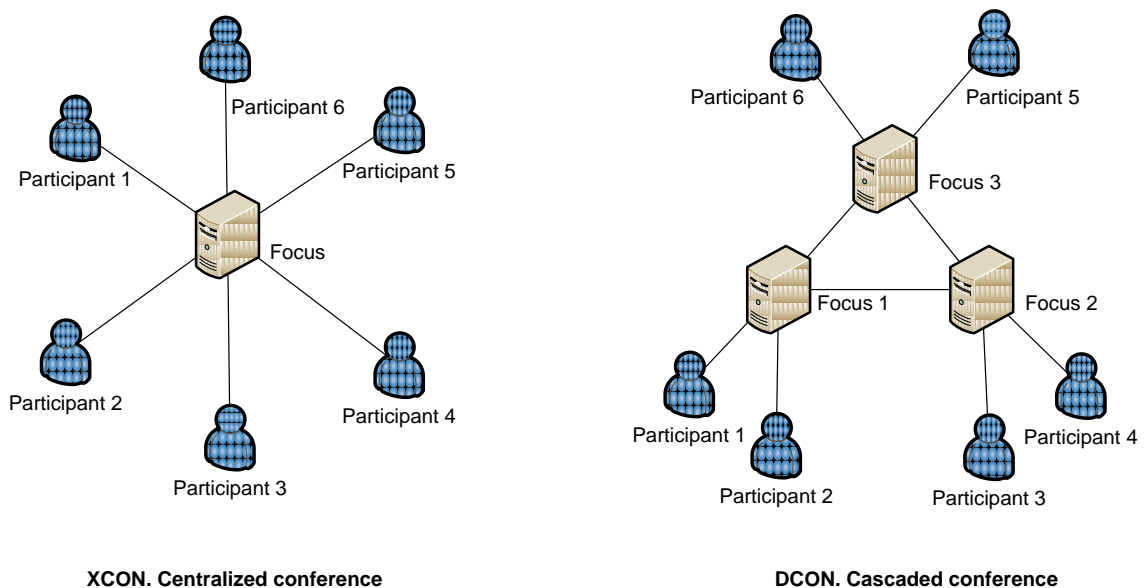


Fig. 11. Arquitecturas de conferencia XCON y DCON

Las conferencias multimedia tienen, sin embargo, una peculiaridad que impide generalizar su arquitectura de soporte: la actividad de cada usuario no tiene efecto directo en las actividades de los demás. En el fondo, los sistemas de conferencia intentan replicar el contenido audiovisual generado por un usuario a todos los demás, recurriendo a mezclas de audio y vídeo para ganar ancho de banda, pero sin mayor interacción entre los contenidos; una conferencia a cuatro, por ejemplo, puede estar representada como un mosaico compuesto por cuatro vídeos en miniatura, uno de cada usuario. En este caso, las actividades de cada usuario sólo afectan a su parte del mosaico, no afecta a los demás.

En una aplicación interactiva como un juego de ajedrez, por indicar el ejemplo más sencillo, existe un tablero de juego, compartido por los jugadores, y las acciones de un jugador condicionan las acciones que puede efectuar el otro. Por ejemplo, si en un movimiento el primer jugador se come un caballo del segundo, este último no podrá mover ese caballo porque ya no existe en el tablero de juego. Las aplicaciones multimedia multiusuario de alta interactividad, en general, precisan el mantenimiento y sincronización del denominado estado de la aplicación, presentando una problemática añadida respecto de las conferencias multimedia audiovisuales.

La comparación es igualmente válida con los servicios de provisión de televisión por IP (IPTV), que han ganado gran atención en el campo de la investigación durante los últimos años y ya se empiezan a ver despliegues comerciales como Imagenio o Jazztelia TV en el caso de España. Como se muestra en la Fig. 12, en un servicio de distribución de vídeo en tiempo real (esquemas 1 y 2 de la figura), la fuente de vídeo se retransmite a los espectadores, bien mediante envío unicast a cada espectador o con el apoyo de redes multicast, y éstos únicamente reproducen el contenido recibido.

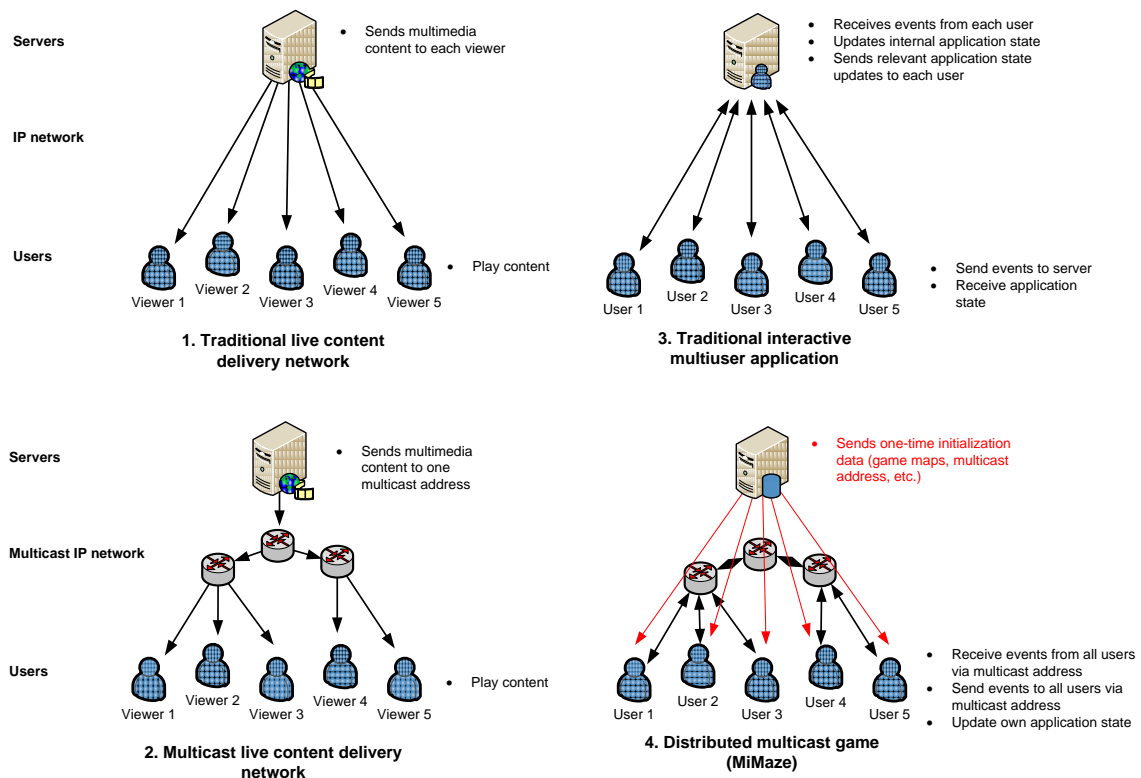


Fig. 12. Distribución de contenidos VS aplicaciones interactivas

Sin embargo, en una aplicación multiusuario interactiva (esquemas 3 y 4), es necesario almacenar el estado de la actividad, procesar el conjunto de eventos (acciones realizadas) generado por los usuarios y actualizar el estado, enviando las actualizaciones oportunas a cada participante para que se mantenga la coherencia en la percepción del estado por parte de cada usuario.

## 2.4 Gestión de documentos XML (XDM 2.0)

La arquitectura de gestión de documentos XML (XDM) es una iniciativa de OMA (*Open Mobile Alliance*) para proporcionar a los habilitadores de servicio y, en general, a cualquier aplicación que requiera manejar datos en formato XML, un conjunto común de protocolos, convenciones de acceso a datos, clasificación de aplicaciones y entidades funcionales. Servicios como la gestión de presencia, pulsar para hablar (*push to talk*), registro de conversaciones de texto, gestión de listas de contactos, etc. utilizan XML como formato de intercambio de información. Todos estos servicios son clave en IMS y han sido adoptados por 3GPP como servicios estándar, de ahí la importancia de la arquitectura de gestión de documentos XML definida por OMA.

La especificación XDM comprende el uso de un conjunto común de protocolos para que los usuarios autorizados (denominados *Principals*) puedan crear, modificar, suscribirse a cambios, borrar y buscar documentos XML. En concreto, adopta el protocolo XCAP (*XML Configuration Access Protocol*) definido en la RFC 4825 [28] para gestionar el contenido de los documentos y el protocolo SIP –concretamente el paquete de notificación de eventos definido en la RFC 3265 [29] – para gestionar las suscripciones a cambios en los mismos. Además, la especificación de XDM define funciones de búsqueda utilizando el lenguaje XQuery [30] y establece los requisitos para la gestión de documentos XML entre dominios.

La figura Fig. 13 muestra la arquitectura XDM 2.0 completa. Los clientes de XDM (XDMC), que se encuentran en los equipos de usuario o en los servidores de aplicación, acceden a documentos XML almacenados en el servidor XDM (XDMS). El proxy de agregación (AP) y los intermediarios de búsqueda proporcionan interfaces tanto a redes locales como remotas, constituyendo el punto de entrada en un escenario multidominio.

Cabe destacar que los documentos gestionados por XDM no tienen un formato arbitrario: deben estar asociados a un identificador de aplicación único (*Application Usage Identifier - AUID*) y estar construidos conforme al modelo requerido por dicha aplicación. Algunos tipos de aplicación son presencia, listas compartidas, perfiles compartidos, políticas compartidas y grupos compartidos.

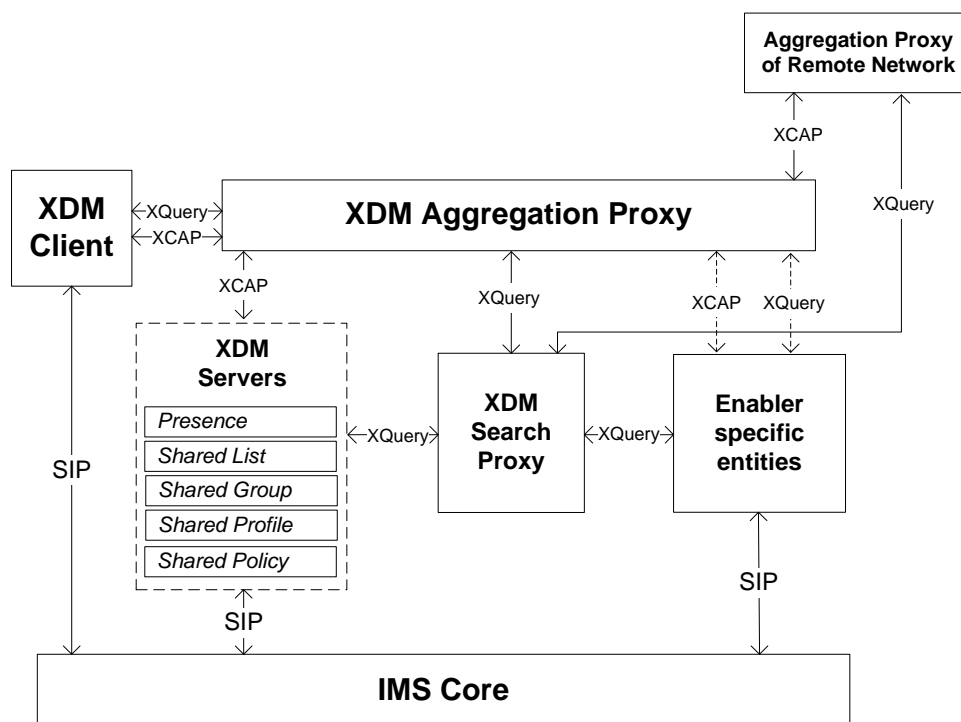


Fig. 13. Arquitectura del gestor de documentos XML de OMA

### 2.4.1 Presencia

El servicio de presencia permite publicar información acerca del estado actual de un usuario (lugar, actividad, etc.), dispositivos disponibles y preferencias (p.ej. teléfono móvil para llamadas de voz y un equipo de salón para video llamadas) o incluso servicios en uso en un momento dado.

La información de presencia se combina habitualmente con la lista de contactos o agenda para mostrar la disponibilidad del usuario, si bien las aplicaciones innovadoras que se desarrollen podrían utilizar esta información sobre dispositivos y preferencias de servicio para ofrecer un valor añadido al usuario. Por ejemplo, un servicio de teleconsulta médica basado en IMS, donde el usuario disponga de un monitor de corazón portátil para hacer la lectura del electrocardiograma en su domicilio, podría actualizar el estado de presencia del usuario automáticamente al colocarse los sensores para mostrarse como no disponible y redirigir las llamadas al buzón de voz mientras dure la lectura para evitar distracciones que influyan en la medida o hagan que deba repetirse.

### 2.4.2 Lista compartida

Una lista compartida define un conjunto de URIs de usuario y atributos asociados. Se utiliza habitualmente para almacenar la agenda de contactos en la red (concretamente en el servidor XDM) y compartirla/sincronizarla con todos los dispositivos conectados del usuario. Además, puede utilizarse para crear listas personales con el fin de agrupar contactos, como por ejemplo una lista de amigos que les gusta el golf, y usar las listas como entradas para aplicaciones que ofrezcan servicios a grupos (p.ej. un servicio avanzado de videoconferencia que, a partir de la lista, coordine el establecimiento de las sesiones necesarias para agregar a todos los usuarios a una videoconferencia).

### 2.4.3 Perfil compartido

El perfil compartido es un documento XML que comprende diversos atributos personales como la edad, hobbies, idiomas, país, etc. Las entidades autorizadas (usuarios o aplicaciones) pueden usar esta información para ofrecer un servicio personalizado o forzar políticas de servicio (p.ej. no permitir el acceso a personas menores de 18 años a un chat público). Por supuesto, XDM contempla mecanismos de seguridad y privacidad para controlar el acceso a esta información.



#### 2.4.4 Política compartida

Un documento de políticas compartidas consiste en un conjunto de reglas de acceso al servicio. Las reglas pueden incluir condiciones (p.ej. la URI es “sip:golfservice@provider”) y acciones (p.ej. permitir el acceso a mi lista de amigos de golf).

Cabe destacar que la arquitectura XDM sólo permite almacenar y gestionar las políticas de acceso, pero no forzarlas. Es responsabilidad de la aplicación o servicio (p.ej. el gestor de listas compartidas o una aplicación de terceros) consultar el documento de políticas asociado y evaluar las condiciones expuestas.

#### 2.4.5 Grupo compartido

Un grupo compartido (*Shared Group*) es un conjunto predefinido de usuarios (*Users*) acompañado de un conjunto de políticas y atributos. La comunicación entre estos usuarios se denomina sesión de grupo (*Group Session*).

El concepto de grupos compartidos es posiblemente la adición más importante a la arquitectura 2.0 de XDM en lo relativo a aplicaciones multiusuario. En XDM 1.0 existe un concepto similar denominado grupo PoC y asociado a la aplicación de pulsar para hablar (PTT/PoC), que permite definir un conjunto de usuarios y ponerlos en comunicación tipo *walkie-talkie*. En XDM 2.0, sin embargo, se ha generalizado el concepto para extenderlo a cualquier tipo de aplicación, de modo que cualquier servicio o aplicación pueda describir una comunicación multiusuario en un formato estándar. Por supuesto, PoC es una de estas aplicaciones pues la definición de grupos compartidos es compatible con la ya definida de grupos PoC<sup>3</sup>.

No obstante, el documento XML que describe un grupo compartido únicamente define sus características de una forma estándar, pero los servidores XDM de grupos compartidos no realizan ninguna gestión de sesiones ni almacenan el estado de la conferencia. Tampoco fuerzan las políticas descritas. Todas estas tareas deben ser implementadas por una entidad lógica externa denominada *Group Session Controlling Function* (función de control de sesiones de grupo), que típicamente corresponde a la aplicación multiusuario que haga uso de la arquitectura XDM para la gestión de grupos compartidos. Citando a OMA [31], <<la función de control de sesiones de

---

<sup>3</sup> De hecho, para mantener la compatibilidad con las aplicaciones PoC definidas en el marco de XDM 1.0, el espacio de nombres por defecto en XDM 2.0 para los grupos compartidos es urn:oma:xml:poc:list-service, aunque el documento no tiene por qué estar definiendo realmente un grupo para la aplicación de PoC.

grupo está implementada en un servidor de aplicaciones y proporciona gestión centralizada de las sesiones de grupo, lo que incluye la aplicación de las políticas de grupo>>.

La Fig. 14 muestra la estructura de un documento que describe un grupo compartido. Cada grupo está identificado por su URI de servicio (*list-service URI*) e incluye el conjunto de miembros que forman actualmente el grupo (*list*) y las políticas que aplican al establecimiento de sesiones de grupo (*ruleset, age-restriction, sessionactive-policy, qoe, supported-services, etc.*). Adicionalmente, permite indicar que el grupo aparezca en los procesos de búsqueda, de forma que usuarios externos puedan encontrar el grupo mediante consultas de búsqueda y potencialmente unirse a la sesión de grupo.

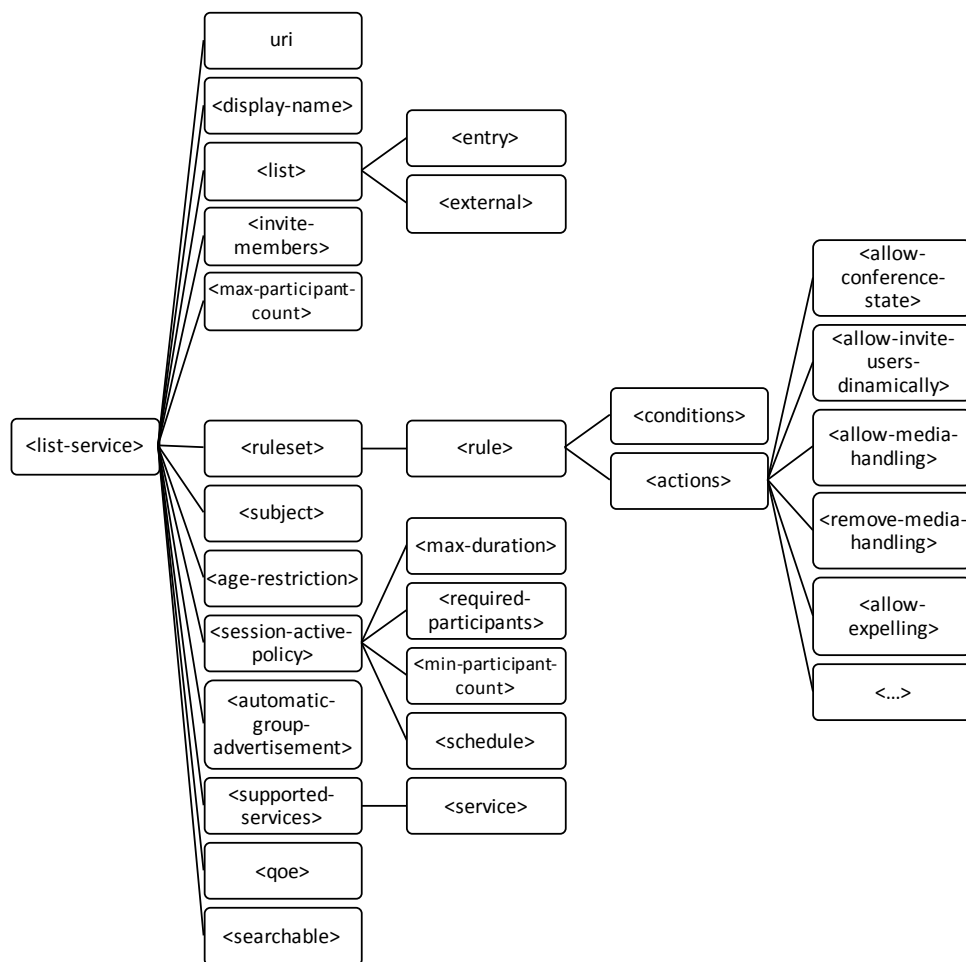


Fig. 14. Esquema de un grupo compartido

## 2.5 Tecnologías y arquitecturas de soporte a aplicaciones HIMMA en el IMS

En las siguientes secciones se analizan distintas propuestas de soporte a aplicaciones HIMMA en el IMS, incluyendo tanto tecnologías de desarrollo de relevancia en esta Tesis como iniciativas en el ámbito de la investigación y la estandarización, algunas orientadas exclusivamente al soporte de juegos en red y otras de ámbito más general.

En concreto, se presentan iniciativas desde el campo de la estandarización, como es el caso de las interfaces Java para desarrollo de aplicaciones en IMS (JSR 281, JSR 325 y JSR 289) y la plataforma de juegos propuesta por la Open Mobile Alliance. Adicionalmente, se presentan los trabajos de investigación más relacionados con la Tesis y, finalmente, se revisa el soporte a aplicaciones HIMMA existente en las soluciones comerciales de IMS encontradas en la industria actualmente.

### 2.5.1 IMS Client Platform (JSR 281 y JSR 325)

*IMS Client Platform* (ICP) es una plataforma desarrollada por Ericsson para fomentar la creación de aplicaciones IMS en los dispositivos, e incluye las herramientas de desarrollo, bibliotecas y emuladores de terminales, así como una interfaz de programación Java que aísla al desarrollador de los protocolos involucrados en IMS.

La interfaz de programación de ICP se convirtió en propuesta de estándar Java para el acceso a servicios IMS desde dispositivos en 2005 y estaba liderada por Ericsson y BenQ. Inicialmente quedaba recogida bajo la JSR (*Java Specification Request*) 281, aunque finalmente la funcionalidad completa de la plataforma para clientes IMS se dividió en dos especificaciones:

- **JSR 281** (*IMS Services API*) [32], aprobada como estándar de Java en julio de 2008 y que incluye los aspectos más básicos como registro en la red, comunicaciones de voz o vídeo y gestión de aplicaciones instaladas.
- **JSR 325** (*IMS Communication Enablers*) [33], aprobada como estándar de Java en enero de 2010, que proporciona un API para el acceso a un conjunto de habilitadores comunes en IMS que fueron descartados en la JSR 281: principalmente presencia, gestión de grupos y mensajería instantánea.

La plataforma de clientes IMS no define una arquitectura de red en cuanto a que sólo persigue acelerar el desarrollo de las aplicaciones que hagan uso de las funciones básicas que ya proporciona IMS. Igualmente, no se centra en ningún ámbito de trabajo como los juegos en red o el trabajo colaborativo, si bien la implementación de esta plataforma es uno de los elementos que Ericsson ha utilizado en la plataforma

multijugador para móviles que se expone en la sección 2.5.5 y es base del middleware HIMMA propuesto en el Capítulo 5 como una de las contribuciones de la Tesis.

Las características de la plataforma (combinando JSR 281 y JSR 325) son las siguientes:

- Acceso a la funcionalidad del núcleo de IMS (accesibilidad, seguridad, control de la política de calidad de servicio).
- Soporte de *Push-to-Talk over Cellular* (Pulsar para Hablar, PoC o PTT)
- Soporte de Gestión de Listas de Grupos (GLM – *Group List Management*)
- Todos los aspectos del servicio están estandarizados, como en el OMA PoC:
  - Características relacionadas con el usuario.
  - Arquitectura.
  - Gestión de protocolos.
  - Interfaz del nivel de aplicación.
- El desarrollador no tiene que manejar mensajes SIP en la aplicación pues la especificación es de más alto nivel.

La arquitectura software de un cliente de IMS desarrollado con estas APIs es como la mostrada en la Fig. 15, donde pueden distinguirse las siguientes capas lógicas:

- **Plataforma software del dispositivo.** Incluye las pilas de los distintos protocolos involucrados en la prestación del servicio: SIP, SDP, RTP, etc.
- **Capa de implementación.** Implementa las funciones ofrecidas por las distintas APIs de Java, incluidas las propias de la JSR 281 y JSR 325.
  - Los terminales móviles deben incluir soporte dentro del sistema operativo para los servicios básicos estandarizados en el IMS.
  - Para el acceso a servicios combinados (aquellos compuestos por varios servicios básicos del IMS) así como aquellos que no estén estandarizados, la especificación define mecanismos de descarga e integración en el API de JSR 281, asegurando así la flexibilidad necesaria para utilizar cualquier servicio.
- **Interfaces de programación.** La capa de interfaces contempla tanto las interfaces estándar de Java (Core API en la figura) como las propias de JSR 281 y JSR 325 (Service API en la figura).
- **Capa de aplicación.** Es aquella compuesta por las aplicaciones Java creadas por el desarrollador y que hacen uso de las distintas interfaces provistas en la capa inferior.

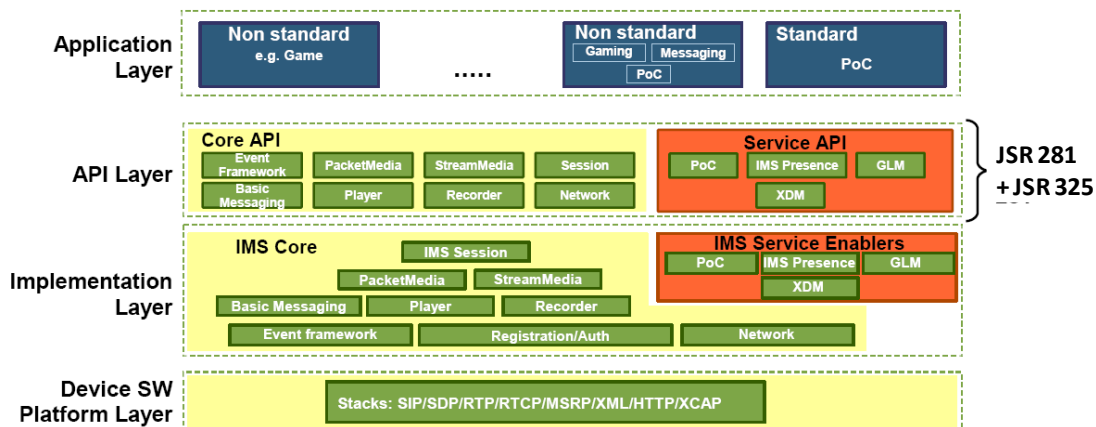


Fig. 15. Arquitectura de la plataforma para clientes de IMS

El hecho de que estas dos APIs, JSR 281 y JSR 325, hayan sido aprobadas dentro del proceso de estandarización de la plataforma Java, muestra el interés de la industria en el desarrollo de aplicaciones para redes IMS. El comité de expertos que ha participado durante el proceso de aprobación de estas propuestas incluye miembros pertenecientes a operadores (AT&T, China Mobile, Orange, T-Mobile, etc.) y fabricantes (Ericsson, Motorola, NEC, Siemens, LG, Samsung, Sony-Ericsson, Nokia, etc.) con presencia internacional, lo que sin duda supone una ventaja añadida para facilitar su adopción en comparación con propuestas independientes.

## 2.5.2 Java SipServlets (JSR 289)

Los Servlets son módulos de código Java que se ejecutan en un servidor de aplicaciones (de ahí el nombre de Servlets, en comparación con los Applets en el lado de cliente) y que responden a las peticiones de los clientes desencadenando acciones o devolviendo cierto contenido.

La especificación JSR 289 [34] presenta una plataforma que permite simplificar la gestión de señalización SIP – en concreto evita preocuparse de gestionar los sockets de comunicación y de la sintaxis propia del protocolo – y crear aplicaciones convergentes HTTP/SIP. Un servlet SIP se ejecuta en un contenedor de servlets – el servidor de aplicaciones en IMS – y procesa señalización SIP. Como cualquier componente Java, los servlets pueden ser cargados y ejecutados en cualquier servidor de aplicaciones SIP siempre y cuando respeten el API definida. Los servlets SIP interactúan con los clientes SIP mediante el intercambio de mensajes de petición y respuesta que se lleva a cabo a través del contenedor de servlets.

La tecnología Java SipServlet se ha convertido en un estándar *de facto* para el desarrollo de aplicaciones en IMS y los principales fabricantes de servidores de

aplicación tienen productos compatibles con la especificación JSR 289, como el servidor libre *Sun Glassfish/Sailfin* o el servidor de aplicaciones *BEA WebLogic SIP Server*.

Sin embargo, hay que tener en cuenta que los SipServlets no constituyen una interfaz completa para el acceso a los servicios de IMS, únicamente facilitan las relacionadas con tratamiento de señalización SIP. Así, los SipServlets constituyen una pieza de apoyo básica a aplicaciones HIMMA pero es necesaria lógica adicional en el servidor que gestione las interacciones entre los usuarios y con otros habilitadores (p.ej. el XDMS para gestión de grupos o presencia, que no sólo tiene interfaz SIP sino XCAP).

### 2.5.3 Plataforma de juegos de la Open Mobile Alliance

Open Mobile Alliance (OMA) es una organización formada en junio del 2002 por más de doscientas compañías representando a los operadores móviles, proveedores de terminales y de red, compañías de tecnologías de la información y proveedores de contenidos líderes de todo el mundo.

El grupo de trabajo *Games Services* de OMA continúa el trabajo originado por el MGIF (*Mobile Games Interoperability Forum*) y propone el desarrollo de una especificación detallada de servicios de juego para facilitar la interoperabilidad, conseguir una experiencia de juego rica y establecer los mecanismos para el despliegue de juegos.

Hasta la fecha, se han propuesto dos especificaciones de la plataforma de juegos:

- Plataforma de juegos 1.0 (OMA Gaming Platform 1.0)
- Interfaz cliente/servidor para servicios de juegos (OMA Game Services Client/Server Interface)

La segunda es, cronológicamente, la continuación del trabajo de la primera, y podría considerarse la versión 2.0 de la plataforma de juegos. Los cambios sustanciales introducidos en la interfaz cliente/servidor respecto de la versión 1.0 hace que las dos especificaciones sean incompatibles entre sí.

La primera especificación de interfaz para plataformas de juegos de OMA [35] está dirigida a gestionar los aspectos relacionados con la portabilidad e interoperabilidad en el espacio de los juegos móviles.

La plataforma permite a los desarrolladores de juegos producir y desplegar juegos móviles que sean portados más fácilmente entre múltiples plataformas de juegos y redes inalámbricas, y puedan ser ejecutados en diferentes terminales móviles.

Nótese que la *Gaming Platform* de OMA no es realmente una plataforma de juegos como tal sino una interfaz para que los distintos juegos puedan hacer uso de las funciones de la plataforma que los soporte de forma estándar (la Fig. 16 [35] pone de manifiesto este aspecto).

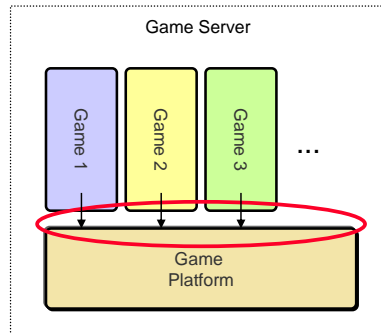


Fig. 16. Primera versión de la plataforma de juegos de OMA

La especificación contempla una arquitectura, un modelo de objetos y un conjunto de interfaces de programación en lenguaje Java para acceder a las siguientes funciones:

- Gestión de la sesión de juego
- Conectividad
- Facturación
- Puntuaciones y gestión de competiciones
- Temporizadores
- Registro de usuarios

La continuación del trabajo en la *Gaming Platform (GP) 1.0* dio lugar a la definición del denominado interfaz cliente/servidor para servicios de juegos, que comprende los siguientes puntos:

- **GP 2.0 Server Framework.** Es la continuación directa del trabajo de GP 1.0, centrándose en el mismo ámbito: las comunicaciones entre los servidores de juego y la plataforma de juegos.
- **GP 2.0 Client/Server Protocol:** Es una nueva interfaz que se centra en la comunicación entre un cliente de juego en un terminal móvil y la plataforma de juego. El protocolo definido puede ejecutarse encima de varias capas de

transporte, como por ejemplo HTTP, SOAP o UDP, en función de las capacidades del terminal y la red. Los detalles técnicos del cliente y el servidor son los cubiertos por el documento Client/Server Interface [36].

La Fig. 17 [37] muestra las dos interfaces definidas en la versión 2.0 de los servicios de juegos de OMA. Puede identificarse fácilmente la nueva interfaz añadida comparándola con la Fig. 16, perteneciente a la versión 1.0.

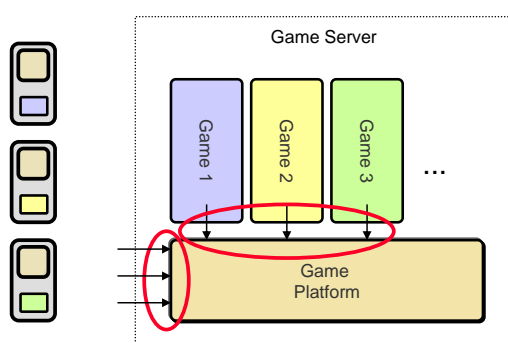


Fig. 17 Segunda versión de la plataforma de juegos de OMA

El nuevo conjunto de interfaces proporciona acceso estandarizado a las siguientes funciones:

- Inicio de sesión y registro
- Persistencia
- Creación de juego y *Matchmaking* (organización de partidas)
- Acciones o movimientos del juego
- Mensajería
- Puntuaciones
- Otras

Adicionalmente, se especifican los requisitos de comunicación dentro del juego que tiene que cumplir *Game Services 2.0* [38], entendiendo comunicación como comunicación personal entre los jugadores. Esta comunicación puede darse mientras el juego está en curso o pausado y puede ser en forma de voz o mensajes de texto o multimedia entre algunos o todos los jugadores de la sesión. Además, la comunicación entre jugadores y no jugadores también puede ser considerada.

Los servicios mínimos que tiene que ofrecer la plataforma, en lo relativo a la comunicación entre usuarios, para que cumpla la especificación de OMA son:

- **Mensajería.** Es obligatorio el soporte de:
  - Mensajes de texto entre un grupo particular de personas.
  - Mensajes de texto entre todos los jugadores de un juego.
  - Mensaje de texto para pausar/continuar un juego.



- **Seguridad.** La mensajería en una sesión de juego requiere el mismo grado de protección de privacidad que para el de mensajería instantánea.
- **Facturación.** Los usuarios reciben notificaciones del cargo de la mensajería entre ellos.
- **Administración y configuración.** Se pueden usar soluciones estándar de la industria para filtrar el contenido de los mensajes.
- **Usabilidad.** Aunque el servicio de mensajería debería ser independiente del servicio de juegos, la plataforma de juegos debería evitar que el usuario sea consciente y experimente una integración perfecta del juego y la mensajería.
- **Interoperabilidad.** Debe haber un API común para que los implementadores de juegos accedan y usen las características de los apodos (*nicknames*) y las comunicaciones dentro del juego. El API para mensajes de texto debería soportar caracteres internacionales.
- **Privacidad.** Los usuarios son capaces de ver las propiedades de otros jugadores sólo de acuerdo a la política definida por el usuario, por ejemplo como se define en el juego. Especialmente, el número de teléfono móvil (MSISDN - *Mobile Station International ISDN Number*) de los jugadores es confidencial.

Dado que el Subsistema IP Multimedia (IMS) ya proporciona algunas de las funciones definidas en la interfaz para plataformas de juego propuesta por OMA, como los aspectos de seguridad, gestión de suscriptores y control de sesiones, OMA ha definido[39] una serie de habilitadores de servicio, mostrados en la parte central de la Fig. 18.

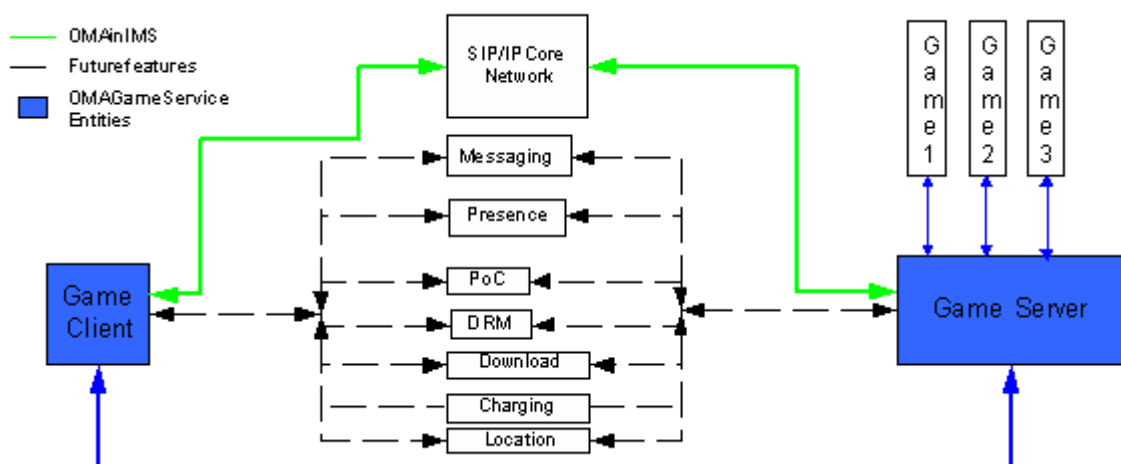


Fig. 18 Utilización de IMS en el Servicio de Juego de OMA

Los componentes habilitadores contemplados por OMA y mostrados en la Fig. 18 son:

- **El núcleo de red SIP/IP.** Proporciona servicios de seguridad, registro, autenticación y autorización.
- **Mensajería.** Permite el intercambio de mensajes de texto entre jugadores.
- **Presencia.** Permite el control del estado de los jugadores.
- **PoC.** Permite el intercambio de mensajes de voz entre jugadores.
- **DRM (Digital Rights Management).** Habilita la gestión de derechos de autor.
- **Descargas.** Permite la descarga de contenidos por el usuario.
- **Facturación.** Permite integrar los cobros relativos al juego en el sistema de facturación del operador, evitando la necesidad de desplegar un sistema de facturación paralelo por parte del desarrollador del juego.
- **Localización.** Permite la creación de servicios basados en la posición del usuario.

Como indica la Fig. 18, sólo están definidas las interfaces que habilitan los servicios del núcleo de red (principalmente aspectos de autenticación y registro) mientras que el resto de funciones identificadas se indican como trabajo futuro.

Cabe destacar el marcado carácter cliente/servidor que sigue teniendo la iniciativa, pues la utilización de los habilitadores de IMS se reduce a establecer sesiones multimedia con el servidor de juego y no aprovecha el potencial de gestión de usuarios y dispositivos que ofrece la red.

#### 2.5.4 Propuestas de soporte a aplicaciones HIMMA

El trabajo de Akkawi et al. [40] es uno de los primeros en tratar la adecuación de la arquitectura IMS para soportar servicios de juego en red y propone una plataforma de juegos para móviles en el IMS. El componente principal de esta plataforma es el denominado *Game Focus* (punto central de juegos), un servidor central que hace de intermediario con todos los jugadores. No obstante, los jugadores utilizan SIP para establecer una sesión con servidores de juego tradicionales que implementan todas las funciones, no explotando la capacidad de IMS a través de sus habilitadores como los de

gestión de grupos o presencia enriquecida. La Fig. 19 muestra los distintos componentes de la plataforma propuesta y las relaciones entre ellos.

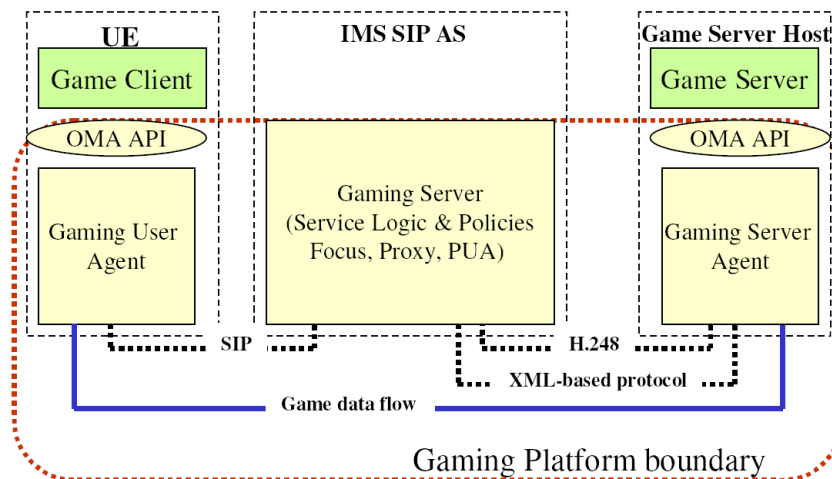


Fig. 19. Interacción entre módulos de la plataforma de juego[40]

El componente central de la plataforma es el *Focus*, que reside en el servidor de aplicaciones SIP. El *Focus* es responsable de la gestión de sesiones y mantiene una relación de señalización con cada jugador. Tiene acceso a las políticas del juego, las cuales contienen información de autorización de los jugadores, ante las cuales el *Focus* actúa en consecuencia. El *Focus* proporciona funcionalidad de proxy para acceder a la información de presencia del juego (utilizando las extensiones estándar de mensajería de SIP) y para acceder a las capacidades de mensajes IMS soportadas por el juego. Cada juego ejecutado en el terminal de usuario incluye tanto la parte específica del juego (*Game Client*) como la parte de acceso a los servicios de la plataforma (*Gaming User Agent*), las cuales se comunican a través de la interfaz estándar definida por OMA. De este modo, al desarrollador del juego se le oculta el manejo de sesiones SIP.

Balakrishnan y Sadasivan [41] presentan los juegos interactivos en movilidad como una aplicación definitiva (*killer application*) para potenciar las redes y servicios móviles. Además, destacan las ventajas que ofrecería una arquitectura IMS de cara a aportar una solución de soporte a estas aplicaciones que fuera independiente de la red de acceso y de los terminales.

Singh y Acharya presentan en [42] el diseño de un prototipo que utiliza el protocolo SIP para proveer contextos de juego en juegos multijugador. El prototipo, además de permitir las funciones básicas como permitir a los jugadores hablar entre ellos para coordinar jugadas de equipo y actividades, soporta comunicaciones en tiempo real

entre jugadores basadas en contextos compartidos, como la misma situación física o habitación dentro del entorno de juego. Es decir, es capaz de gestionar de forma dinámica y transparente las sesiones de audio (conferencias) basándose en el comportamiento del jugador y su situación dentro del entorno de juego con el fin de recrear el espacio sonoro del mismo (voces lejanas o cercanas, ruido de fondo, etc.). Puede encontrarse más información sobre esta aplicación de SIP a juegos multijugador en la patente [43].

Araki et al. [44] describen una plataforma para proporcionar servicios de entretenimiento en IMS basada en el habilitador de gestión de grupos ofrecido por XDMS y una implementación propia del nodo de orquestación de servicios SCIM (*Service Capability Interaction Management*), ilustrado en la Fig. 20. Los autores proponen un escenario de ejemplo basado en un juego con arquitectura cliente-servidor, si bien la solución es aplicable a otro tipo de aplicaciones.

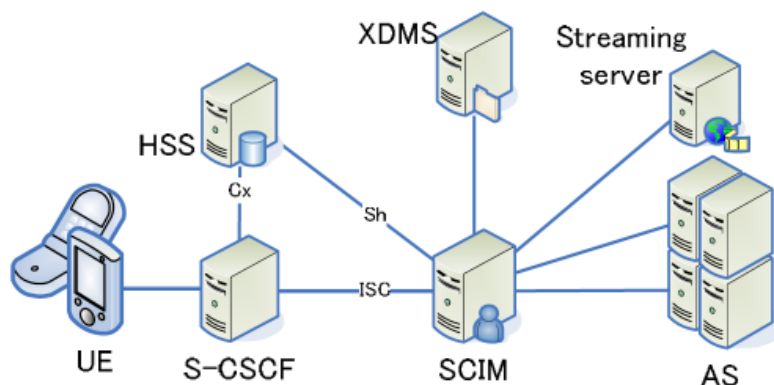


Fig. 20 Arquitectura de soporte a aplicaciones de comunidad de Araki et al. [44]

Rashid et al. [45] muestran algunas implicaciones prácticas de SIP e IMS en la evolución de las aplicaciones móviles, principalmente orientadas a aspectos de desarrollo. En este sentido, destacan el dominio de Java como tecnología de desarrollo de aplicaciones móviles y cómo se adapta al desarrollo de aplicaciones SIP e IMS.

Neumann [46] y Baughnman [47] discuten aspectos generales de los juegos sin servidores específicos (peer-to-peer), principalmente relacionados con los retos que supone la gestión de la información de estado de forma distribuida y los procedimientos para evitar trampas en el juego. Estos artículos presuponen que la red no ofrece nada más que conectividad IP, por lo que hay aspectos que no aprovechan el entorno IMS, que está orientado a sesiones y se beneficia de un conjunto de habilitadores reutilizables.

El artículo de Chunyan et al. [48] es una de las iniciativas más recientes publicadas en cuanto a soporte de aplicaciones HIMMA en el IMS. Los autores han escogido un caso de estudio de un juego en red multijugador donde compiten dos equipos, si bien

es extrapolable a otro tipo de aplicaciones. Proponen una prueba de concepto, ilustrada en la Fig. 21, que consta de un API de alto nivel para el establecimiento de la partida, basándose en las JSR 289 y JSR 309 de Java así como en un API de control de conferencias que presentaron Belqasmi et al. en [49] como extensión al servicio de Pulsar para Hablar (PTT). Los usuarios ejecutan un software específico del juego basado en la plataforma para clientes IMS (ICP).

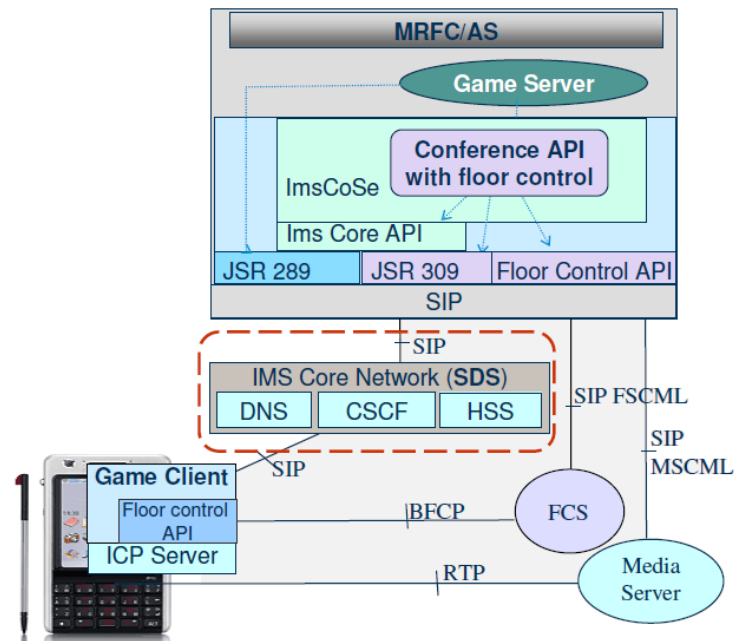


Fig. 21 Arquitectura del caso de estudio de juegos de Chunyan et al.

### 2.5.5 Soporte de aplicaciones HIMMA en plataformas IMS comerciales

Las soluciones IMS disponibles actualmente en el mercado implementan las especificaciones de 3GPP, por lo que no cuentan con soporte específico para aplicaciones HIMMA más allá de los servicios estándar identificados en 2.2. Los servidores de aplicación en estas soluciones suelen basarse en tecnología Java y hacer uso de la tecnología Java SipServlet vista en 2.5.2. No obstante, algunos fabricantes han realizado demostraciones tecnológicas de sus soluciones IMS con aplicaciones HIMMA.

Ericsson, por ejemplo, realizó la primera demostración de un juego multijugador para móviles en el IMS [50]. La solución propuesta incluye la utilización de ICP (expuesto en 2.5.1) para acceder a los servicios de IMS y se demostró con un juego de cartas en la que los usuarios podían estar jugando y, a la vez, acceder a la información

de presencia de otros jugadores, compartir datos, intercambiar mensajes instantáneos e intercambiar mensajes de voz (mediante el servicio PTT).

La plataforma IMS de Alcatel-Lucent incluye un entorno de distribución de aplicaciones multimedia denominado MiViewTV™. Está orientada a ofrecer servicios de televisión IP (IPTV) principalmente, pero también incluye una aplicación denominada *Multiparty Video Gaming* [51] que permite a los operadores desplegar juegos multijugador con soporte para teléfonos y ordenadores, integrando voz, vídeo y datos.

La plataforma *Gaming Capability* [52] de Inova Gaming, permite a los operadores ofrecer juegos multijugador como un servicio a sus usuarios. Reside en el entorno IMS del operador (que estará suministrado por otro fabricante) y hace uso exclusivo de SIP como su protocolo para inicio de sesión durante la jugada y la interacción entre los jugadores.

También para entornos de operador, si bien no IMS, encontramos la plataforma Nokia SNAP [53]. La plataforma de Nokia soporta juegos en red multijugador en entornos de operador (no es accesible desde Internet) y consta de infraestructura de servidor y bibliotecas del lado cliente. Proporciona algunos servicios básicos para los juegos como el registro de usuarios, establecimiento de listas de amigos, rankings y algunos mecanismos de gestión del estado del juego. Es relevante porque está desplegada en múltiples operadores a lo largo del mundo pero no es una solución para entornos IMS. De hecho, la arquitectura IMS ya puede proporcionar de forma estándar gran parte de la funcionalidad que ofrece esta plataforma.

Finalmente, *Neutron*, de la compañía Exit Games [54] es una plataforma de provisión de servicios para juegos en red tanto para operadores como para ámbitos de Internet que facilita el desarrollo de juegos gracias a una biblioteca cliente multiplataforma (hay implementaciones para Android, iPhone, Symbian, Windows Mobile, etc.). Según el fabricante, se ha añadido soporte para interoperar con la red IMS del operador con el fin de reutilizar la agenda de contactos y establecer comunicaciones multimedia en los operadores que presten estos servicios a través de una infraestructura IMS.

## 2.6 Conclusiones

En este capítulo dedicado al estado de la técnica se han repasado diversas tecnologías y propuestas relacionadas con aplicaciones multimedia multiusuario de alta interactividad (HIMMA) en el IMS.

En primer lugar (2.1) se han identificado dos grupos de aplicaciones HIMMA relevantes, juegos en red y aplicaciones profesionales. Se han analizado las principales plataformas que proporcionan servicios multijugador en red en la actualidad, por considerarse un caso de estudio representativo de aplicaciones HIMMA. El análisis de su funcionalidad ha dado lugar a una síntesis original del conjunto de características presentes en las plataformas de juego en red actuales (ver 2.1.1). Igualmente, se han estudiado algunas soluciones de aplicaciones profesionales conocidas y las tecnologías que utilizan.

A continuación se han introducido los conceptos básicos de la arquitectura IMS (2.2) y los servicios que ofrece de forma estándar, que permiten presentar la adecuación de esta arquitectura para el soporte de aplicaciones HIMMA y servirá de punto de partida en el siguiente capítulo para identificar las extensiones apropiadas para mejorar su soporte.

También se han repasado los modelos de comunicaciones multiusuario habituales (2.3), recordando que IMS únicamente facilita un servicio de audio y vídeo conferencia de manera centralizada (concretamente a través del nodo MRF o del servidor PoC en el caso del servicio Pulsar para Hablar), no disponiendo de un marco genérico para soporte de aplicaciones HIMMA.

El análisis de la arquitectura IMS y modelos de comunicaciones multiusuario soportados se complementa con el estudio del gestor de documentos XML (XDM) en la sección 2.4, un nodo de especial importancia dentro de la arquitectura de IMS pues soporta las funciones de gestión de grupos, presencia, etc. que resultarán clave para las contribuciones de la Tesis.

Finalmente, se han recogido las principales propuestas de soporte de IMS relativas al soporte de aplicaciones HIMMA en general y de juegos en red en particular, destacando el uso que hacen de las funciones provistas por IMS. Tras el análisis, cabe resaltar las siguientes ideas:

- Recientemente se ha aprobado el estándar Java JSR 281 para el acceso a servicios IMS desde plataformas cliente Java y la JSR 325 que la complementa. No son exclusivas de juegos en red y son aplicables a cualquier otro ámbito, si bien no incluyen soporte específico para aplicaciones multimedia multiusuario de alta interactividad – sólo contemplan una interfaz de programación para el acceso a servicios básicos como mensajería, presencia o gestión de grupos. Por el lado de servidor únicamente es de relevancia la JSR 289 para la gestión de sesiones SIP, no

existe un API equivalente a la JSR 281/JSR 325 para acceso a servicios en IMS.

- Varios autores comparten la visión de la Tesis, considerando IMS como arquitectura idónea para el soporte de aplicaciones HIMMA gracias a la posibilidad de componer aplicaciones basándose en habilitadores, hacer uso de tecnologías estándar y soportar entornos fijos y móviles.
- En el caso de juegos en red, la Open Mobile Alliance (OMA) empezó a marcar pasos en la estandarización de interfaces de programación exclusivos para plataformas de juegos, considerando al igual que otros autores la reutilización de funciones provistas por el IMS.

Las plataformas de juego multijugador en red, que soportan miles de usuarios simultáneos y constituyen actualmente la mejor referencia de aplicaciones multimedia multiusuario de alta interactividad, cuentan con un gran número de funciones destinadas a enriquecer la experiencia del juego y la colaboración entre usuarios, donde tanto *Battle.net* como *Xbox Live* destacan claramente. No obstante, las pruebas de concepto de plataformas de juegos para el IMS destacan por su sencillez al incorporar únicamente gestión de usuarios y mensajería de texto con juegos de baja interactividad, poniendo de manifiesto la necesidad de un mayor trabajo de investigación en este campo para aprovechar el potencial de las Redes de Próxima Generación como IMS.

Hay que considerar el enfoque de la Tesis que es proporcionar un marco genérico para el soporte de aplicaciones HIMMA, sin centrarse en aspectos que puedan ser muy específicos de determinado ámbito, como es el caso de la detección de trampas en juegos, algoritmos inteligentes para clasificar usuarios o el mantenimiento de estados muy complejos como los requeridos por los juegos de rol multijugador masivos (MMORPG y similares).



# 3. Arquitectura de soporte de aplicaciones HIMMA en el IMS

## 3.1 Introducción

La arquitectura de soporte a aplicaciones multimedia multiusuario de alta interactividad (HIMMA) responde al objetivo principal de la Tesis que es proponer mecanismos y reglas de diseño que contribuyan a mejorar el soporte estándar que las redes IP ofrecen a las aplicaciones HIMMA, en un entorno convergente basado en la arquitectura IMS con habilitadores originales para el soporte de aplicaciones HIMMA definidos en la Tesis.

El Subsistema IP Multimedia o IMS revisado en la sección 2.2 constituye una arquitectura de red muy apropiada para el despliegue de aplicaciones HIMMA al contar con distintos habilitadores de servicio, soporte de sesiones multimedia, calidad de servicio (QoS), etc. Ya vimos que en IMS se definen un conjunto de aplicaciones multiusuario como las de conferencia de audio y vídeo, o el servicio de voz *Push to Talk* (PTT, pulsar para hablar), sin embargo son aplicaciones finales asociadas a tipos de medio concretos (audio, vídeo) y ofrecidas de forma centralizada, no siendo aplicables a otro tipo de aplicaciones multiusuario.

Para dotar al IMS de una capa de gestión de aplicaciones HIMMA se define en el ámbito de esta Tesis un marco general de provisión de aplicaciones HIMMA en entornos que cuenten con la arquitectura estándar IMS para la provisión de servicios multimedia. Este marco genérico responde a la motivación de la Tesis de extender la arquitectura IMS, manteniendo el uso de tecnologías estándar, para ofrecer un mejor soporte a aplicaciones HIMMA. Las extensiones se adelantaron en el capítulo introductorio y siguen dos enfoques: extendiendo la funcionalidad del lado servidor mediante el concepto de *habilitador* y extendiendo la funcionalidad del lado de cliente

mediante un *middleware*. El habilitador ligero de gestión de actividades (LCE) sigue el primer enfoque y se encarga de gestionar actividades multiusuario y mantener su estado, mientras que el *middleware* HIMMA sigue el segundo enfoque y se encarga de proporcionar un conjunto de funciones que facilitan el desarrollo de aplicaciones HIMMA en entornos IMS sin necesidad de infraestructura adicional a la provista por IMS.

En el Capítulo 4 y el Capítulo 5 se detallan respectivamente el diseño y funcionalidad del habilitador LCE y el *middleware* HIMMA, así como los detalles de implementación y pruebas correspondientes. Este capítulo se centra en la definición del marco general de provisión de aplicaciones HIMMA en el IMS y las decisiones de diseño involucradas en la definición del mismo.

### **Nota sobre terminología**

Para evitar confusión con la terminología en sucesivos apartados, denominaremos *sesión* a la relación de comunicación entre dos entidades, de acuerdo al protocolo SIP y *actividad* (en vez de “sesión multiusuario”, concepto no existente en SIP o IMS) a la relación de señalización entre las distintas entidades involucradas en la interacción de los distintos usuarios que participan en una aplicación. Una actividad multiusuario en SIP se compone, por tanto, de un conjunto de sesiones entre distintos participantes o nodos en función de la topología.

Los estándares de SIP, en concreto la RFC 4353 [24] sobre aplicaciones multiusuario, denominan “*conference*” a una comunicación entre múltiples participantes con sus sesiones asociadas, por lo que el habilitador LCE es estrictamente hablando un gestor de conferencias. Sin embargo, el término “conferencia”, especialmente en español, tiene connotaciones sobre los tipos de medios involucrados al asociarse a llamadas telefónicas o conversaciones con audio y vídeo entre múltiples personas, derivadas de los escenarios clásicos de telecomunicación. El habilitador LCE es independiente de los tipos de medio y, por ende, aplicable a cualquier escenario multiusuario, por lo que nos referiremos al LCE como gestor de actividades para evitar confusión con el significado tradicional de “conferencia”.

## 3.2 Arquitectura propuesta

### 3.2.1 Criterios de diseño

A la hora de definir el marco general de provisión de aplicaciones HIMMA en IMS y la extensión de su arquitectura, en caso necesario, se han tenido en cuenta los siguientes criterios de diseño:

- Posibilidad de **definir, establecer y gestionar actividades multiusuario**, sin necesidad de conocer los detalles de las sesiones SIP involucradas.
- Posibilidad de **descubrir (buscar) actividades** existentes.
- Independiente de tipos de medios (*media agnostic*)
- Compatible con **topologías descentralizadas** (a nivel de transporte, se entiende, ya que la señalización de sesiones SIP en IMS siempre es centralizada)
- **Compatible con aplicaciones de conferencia estándar** que hacen uso del paquete de eventos de conferencia definido en la RFC 4575.
- **Interfaces sencillas**. Las extensiones de protocolos o estructuras de datos se harán de acuerdo a los estándares y manteniendo en lo posible la compatibilidad con clientes que no soporten las extensiones. En el caso de definir interfaces de programación, las APIs tendrán un alto nivel de abstracción para facilitar el desarrollo de aplicaciones.
- **Reutilización de la funcionalidad provista por los habilitadores** disponibles de forma estándar en IMS: gestión de grupos, presencia, gestión de documentos XML, etc.
- Soporte de **aplicaciones multimodales** (accesibles mediante navegador web, cliente estándar SIP o aplicación personalizada)
- Soporte **multidispositivo**
- Soporte de **pausa y reanudación** de actividades
- Soporte de **transferencia de sesión con estado**
- **Escalable y multidominio**

### 3.2.2 Descripción de la arquitectura

La extensión de la arquitectura, que se ilustra de forma detallada en la Fig. 22, y la consiguiente implementación de funciones de soporte a aplicaciones HIMMA, contemplan dos enfoques:

- **Extender por el lado del servidor**, añadiendo un habilitador (**LCE - *Lightweight Conferencing Enabler***) que permita manejar actividades multiusuario definidas en XML y guardadas en el XDMS (*XML Document Managment Server*). La definición, propuesta por un usuario o por el organizador de la actividad, incluye el tipo de actividad, si es centralizada (y en qué servidor) o distribuida, condiciones para los participantes, etc. Otros usuarios pueden buscar en el XDMS actividades de su interés y unirse si cumplen las condiciones de participación. Alternativamente, un usuario puede ser invitado por otro (con SIP REFER) para unirse a una actividad. Para unirse, usan SIP enviando en el mensaje SIP INVITE la URI de la actividad. El INVITE se encamina al habilitador LCE que se encarga de analizar el perfil del solicitante para comprobar que cumple las condiciones de participación y, en caso afirmativo, establecer las sesiones necesarias, ya sea con el servidor o con los otros usuarios según la actividad sea de tipo centralizado o distribuido, respectivamente. El LCE se encarga igualmente de actualizar el estado de la conferencia (participantes activos, medios utilizados, etc.), que guarda en otro documento XML dentro del XDMS al que los usuarios pueden suscribirse para recibir notificaciones del cambio de estado. Con este enfoque, la aplicación se comunica mediante los protocolos XCAP/Xquery y SIP (SUBSCRIBE) con el XDMS y con SIP (INVITE, etc.) con el LCE. A su vez, el LCE presenta una interfaz XCAP con el XDMS y otra SIP con el servidor de aplicación o con otros usuarios.
- **Extender por el lado del cliente**. En este enfoque, las funciones de gestión de usuarios y gestión de sesiones para aplicaciones HIMMA se implementan en el lado del cliente mediante un **middleware (HIMMA-MW)**, sin introducir infraestructura o habilitadores adicionales en el IMS. Como el LCE, el middleware permite definir actividades y guardarlas en el XDMS, pero en este caso las actividades se almacenan como parte de la información de presencia del usuario y las posibilidades de descripción son más limitadas. Por ejemplo, no se puede establecer condiciones de participación para otros usuarios. Básicamente, el usuario indica la actividad que está haciendo, o que quiere iniciar, y solo sus contactos puede solicitar unirse. Con el middleware, si se define una actividad solo será visible como parte de la información de presencia del usuario, que no se trata de información pública susceptible de búsquedas por parte de cualquier otro usuario. El middleware establece las sesiones necesarias entre participantes normalmente siguiendo una topología descentralizada, que es el caso para el que está pensado, si bien puede usarse también con servidores de soporte desplegados por el desarrollador manteniendo el API. El middleware proporciona un API Java,

aislando al desarrollador del uso interno de XCAP y SIP para comunicarse con las entidades estándar de IMS en las que se apoya (servidor de presencia y CSCFs).

La Fig. 22 ilustra el marco general de soporte a aplicaciones HIMMA en el IMS propuesto en la Tesis, incluyendo las extensiones propuestas y sus interfaces a los demás elementos de la red.

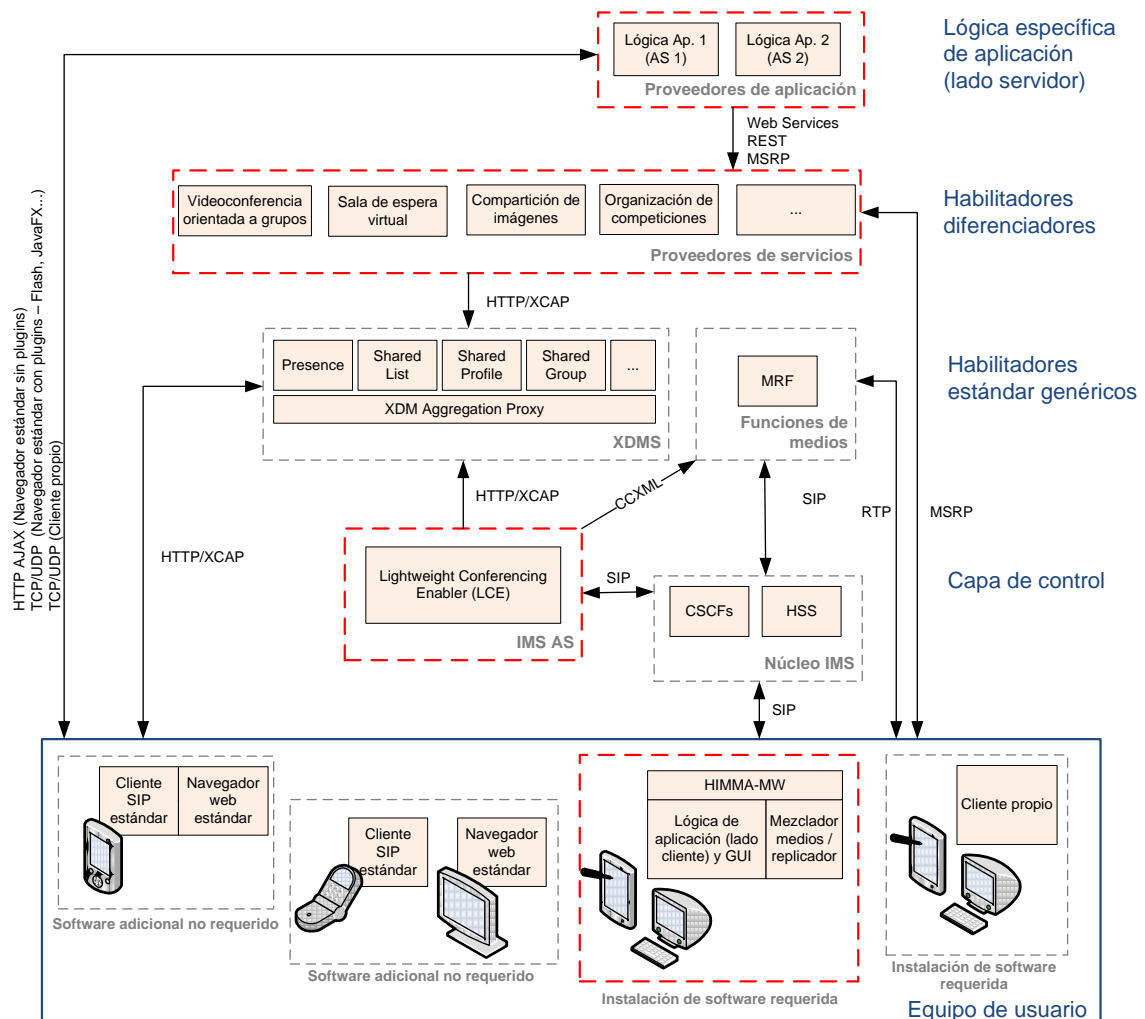


Fig. 22 Marco general de soporte a aplicaciones HIMMA en el IMS

La arquitectura está ordenada por capas de forma lógica, siguiendo la misma filosofía de diseño de IMS que se repasó en el Estado del Arte (sección 2.2.1):

- **Aplicaciones y software en los equipos de los usuarios.** Se contemplan diversas opciones a la hora de acceder a las aplicaciones por parte de los usuarios en función de las capacidades del dispositivo: que cuenten únicamente con clientes estándar de SIP (teléfonos IP) y navegador web, que

dispongan de un cliente específico o que hagan **uso del middleware HIMMA**. En caso de usar el middleware HIMMA, el cliente utiliza el middleware para acceder a las funciones de gestión de la actividad multiusuario.

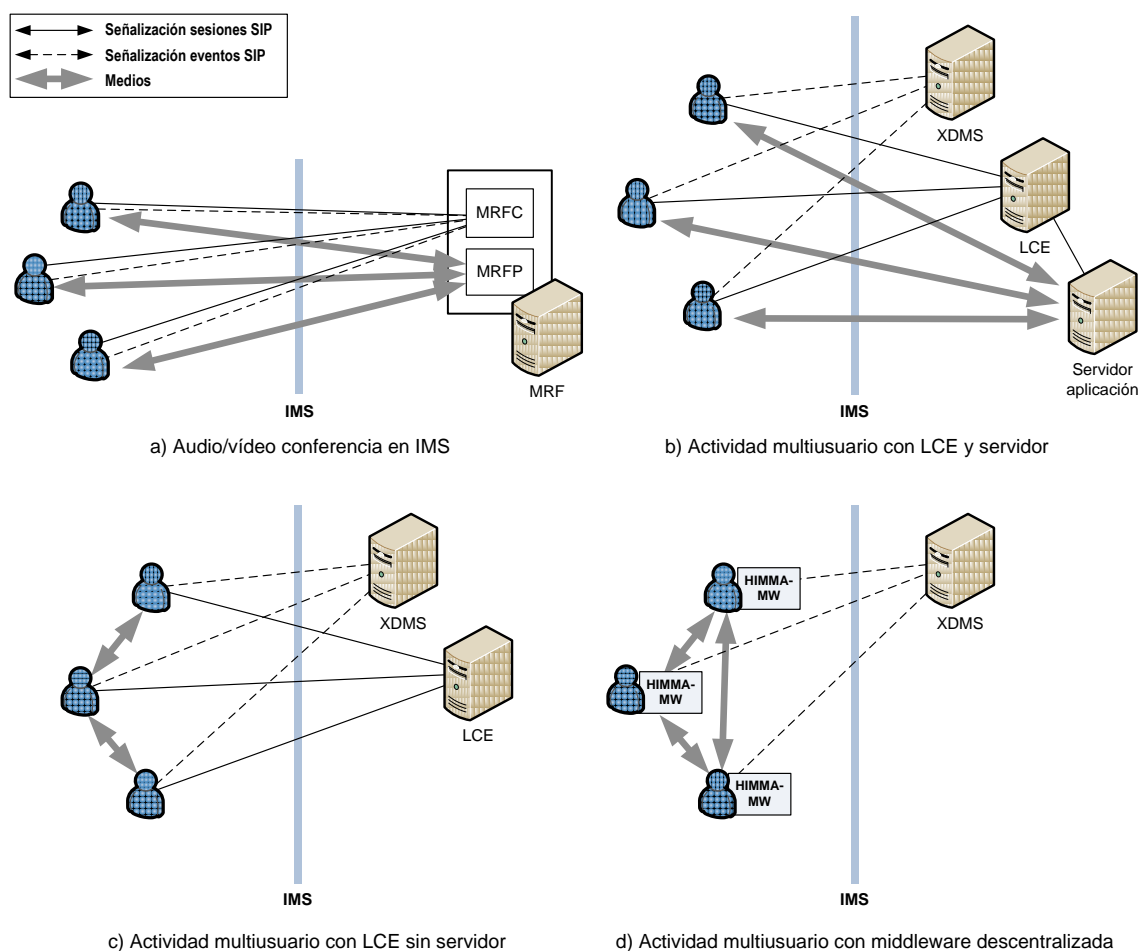
- **Capa de control de IMS**, incluye los intermediarios de señalización SIP (CSCF) y la base de datos de usuarios (HSS). El **habilitador LCE propuesto desde el punto de vista lógico pertenece a esta capa** (podría considerarse un “CSCF para sesiones multiusuario”).
- **Habilitadores estándar** (disponibles en cualquier red IMS), tales como gestión de grupos, mensajería, presencia, etc. cuya funcionalidad se reutiliza en gran medida tanto en el middleware como en el LCE, siguiendo el paradigma de composición de aplicaciones basadas en habilitadores.
- **Habilitadores diferenciadores** (servicios habitualmente de ámbito específico que pueden no encontrarse en cualquier red IMS). Estos habilitadores son servicios ofrecidos por el operador o por terceros que sirven igualmente para componer aplicaciones, aunque no son aplicaciones finales en sí mismas. Los **habilitadores diferenciadores pueden hacer uso de los servicios del habilitador LCE** y de otros habilitadores para ofrecer su funcionalidad, que a su vez será expuesta a otras aplicaciones o habilitadores.
- **Lógica específica de aplicación**, es la aplicación final que se ofrece al usuario: una videoconferencia, un juego, una consulta médica, etc. Tiene lógica específica de esa aplicación pero se apoya en la funcionalidad proporcionada por los habilitadores, ya sean de ámbito específico o genérico.

### 3.3 Ventajas de las extensiones propuestas

Tanto el habilitador ligero de actividades LCE como el middleware HIMMA facilitan el desarrollo y despliegue de aplicaciones HIMMA aprovechando al máximo las ventajas de la arquitectura IMS. Las aplicaciones multiusuario estándar de IMS están destinadas al establecimiento de conferencias de audio y vídeo y no resultan de utilidad para otro tipo de aplicaciones que no dependan del tratamiento de estos tipos de medio concretos. Además, centralizan la gestión de señalización SIP para el establecimiento de sesiones y notificación de eventos, por lo que no aprovechan las posibilidades que ofrece XDMS como la gestión de suscripciones o la agregación de notificaciones.

En la Fig. 23 se ilustra la comparación entre una actividad multiusuario sin las extensiones propuestas (una audio/vídeo conferencia), y distintas posibilidades que

ofrece tanto LCE como el middleware HIMMA. El caso “a)” de una conferencia estándar audiovisual, aplicación ya provista por IMS a través del nodo MRF, es representativa de cualquier aplicación HIMMA que no haga uso de los conceptos y extensiones presentados en la Tesis: un servidor central desarrollado para la aplicación se encarga de gestionar la señalización SIP para el establecimiento de sesiones, los eventos de conferencia (bien de forma estándar mediante notificaciones SIP o mediante un protocolo propietario) y la propia información de la aplicación.



**Fig. 23. Flexibilidad y mayor integración de las actividades con IMS gracias a LCE**

En los casos “b)” y “c)” de la figura se muestra la distribución funcional propuesta en la Tesis para una actividad multiusuario, tanto con servidor de aplicación (caso “b)”) como sin servidor (caso “c)”). En ambos casos, la gestión de la actividad corre a cargo del LCE, que consulta la descripción de la actividad en el XDMS y sólo cursa sesiones que cumplan las condiciones de participación en la actividad. De este modo, se simplifica el servidor de la aplicación, liberándolo de la gestión de las comunicaciones involucradas. El LCE almacena el estado de la actividad en el XDMS,

de modo que los usuarios interesados en una actividad pueden suscribirse y recibir notificaciones automáticamente sin intervención directa del servidor de la aplicación. Para ciertos tipos de aplicación, el LCE permite la posibilidad de prescindir del servidor de aplicación sin pérdida de funcionalidad –con lo que se reduce el coste de despliegue-, facilitando el descubrimiento y participación en actividades de aplicaciones basadas en tecnologías P2P donde el software de los usuarios puede trabajar como cliente y como servidor.

Finalmente, el caso “d)” presenta un escenario del middleware HIMMA, que permite prescindir tanto del servidor de aplicación como de extensiones del lado servidor como es el habilitador LCE, con la contrapartida de tener que instalar el middleware en los terminales de los usuarios.

Además de la flexibilidad para el desarrollo de aplicaciones basadas o no en servidores centrales, destaca la independencia del tipo de medio y la adhesión a estándares. La arquitectura planteada define extensiones para soportar la nueva funcionalidad (nuevos atributos XML, por ejemplo), pero no propone nuevos protocolos y trata de reutilizar al máximo los servicios y funcionalidad ya disponible por habilitadores como el XDMS. La independencia de los tipos de medio garantiza que cualquier aplicación puede beneficiarse de las funciones del LCE y del middleware, y la adhesión a estándares garantiza la correcta integración de las actividades en el entorno IMS sean del proveedor que sean, habilitando búsquedas, suscripciones a eventos, etc.

La Fig. 24 ilustra el beneficio de adherirse a tecnologías estándar a través de un ejemplo de búsqueda de partidas de un juego a las que poder unirse. En un escenario típico en que el proveedor de juegos ofrece su propio servidor de registro y búsqueda de usuarios (caso “a”)), el usuario tendría que buscar partidas disponibles en cada uno de los servidores de búsqueda (lo cual implica también conocer cuáles son y probablemente utilizar protocolos distintos para cada uno). Sin embargo, almacenando las descripciones de la partida (actividad) como documento XML en el XDMS, el usuario puede hacer una búsqueda directamente a través de consultas estándar al XDMS y, cuando encuentre una, unirse a través del LCE que gestionará las sesiones SIP involucradas con el servidor o con otros usuarios en función de la topología requerida por la aplicación.



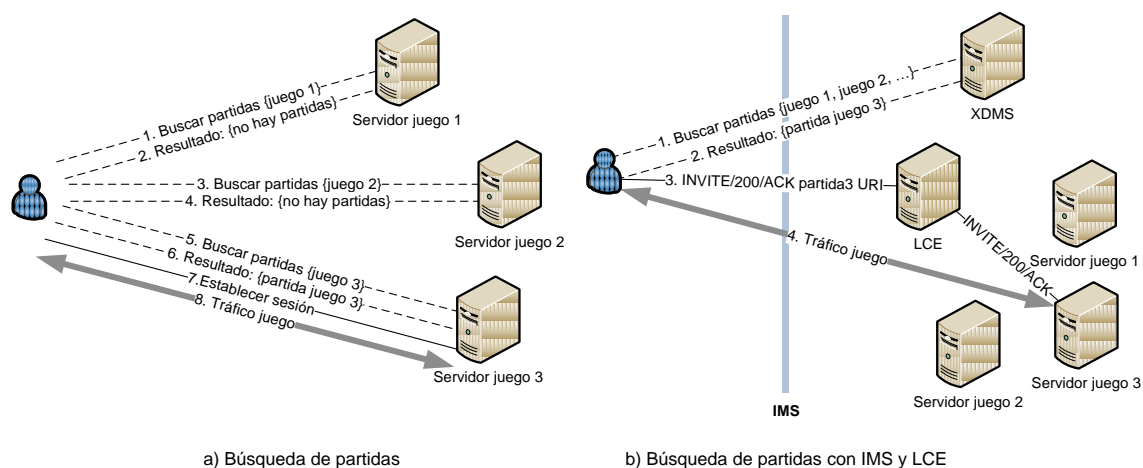


Fig. 24. Simplificación del proceso de búsqueda de actividades por el cliente

### 3.4 Relación entre el habilitador LCE y el middleware HIMMA

El habilitador ligero de actividades LCE y el middleware HIMMA están enfocados a facilitar el desarrollo y despliegue de aplicaciones HIMMA desde dos puntos de acción diferenciados: extendiendo la arquitectura de IMS por el lado de servidor y extendiendo la funcionalidad de los clientes a través de un middleware. Naturalmente, los requisitos de gestión de sesiones y estado dan lugar a solapamientos en la funcionalidad del LCE y el middleware HIMMA.

#### 3.4.1 Comparación de funcionalidad

El habilitador LCE ofrece una capa de gestión de actividades multiusuario mediante la extensión de la arquitectura de IMS desde el lado de servidor. Concretamente, el LCE es capaz de gestionar las sesiones SIP requeridas por la aplicación y de interactuar con el XDMS para crear y actualizar el estado de la actividad, permitiendo que los usuarios autorizados puedan descubrir la actividad y suscribirse a sus cambios. Dada su condición de habilitador es una aplicación que se ejecuta en el servidor de aplicaciones del dominio IMS que tenga desplegado LCE. Al pertenecer al lado de servidor puede disfrutar de ventajas a la hora de acceder al XDMS y manipular documentos XML, siempre y cuando se defina como *“trusted application”* (aplicación de confianza), condición que el middleware nunca podrá cumplir por estar instalado en el lado del usuario. Gracias a esta característica, el LCE puede, por ejemplo, comprobar las condiciones de acceso a una actividad para cursar o no la petición de inicio de sesión de un usuario a esa actividad.

El middleware, por otra parte, es un software que se ejecuta en el terminal de los usuarios, expone una interfaz Java de alto nivel y abstrae al desarrollador de la señalización involucrada para ofrecer funciones de gestión de actividades multiusuario sin soporte adicional en la red IMS (es decir, no presupone la existencia del LCE). Las funciones relativas al descubrimiento de actividades se implementan añadiendo información adicional al documento de presencia de cada usuario, de modo que sus contactos puedan conocer qué actividades están expuestas por ese usuario (recordemos que el middleware está orientado a aplicaciones sin servidor).

### 3.4.2 Uso conjunto

El LCE, apoyándose en la funcionalidad de XDMS, ya implementa la gestión de actividades utilizando el formato de documento XML *Shared Group* definido por OMA, por lo que el middleware podría prescindir de su mecanismo de descubrimiento de actividades si el entorno IMS donde se ejecuta dispone del LCE. De modo que, en su lugar, se haría uso de las funciones provistas por LCE para la búsqueda de actividades y unión a las mismas.

El uso conjunto del middleware HIMMA y el habilitador LCE, modificando convenientemente el middleware para interactuar con el LCE en entornos que dispongan de él, ofrecería la ventaja para los desarrolladores de contar con un API Java de muy alto nivel. Este mismo API podría utilizarse igualmente en el lado de servidor como interfaz para el desarrollo de aplicaciones en el lado de servidor que requieran funciones de gestión de actividades multiusuario y no deseen tratar directamente con los protocolos SIP y XCAP.

## 3.5 Conclusiones

En este capítulo se ha introducido una arquitectura de soporte a aplicaciones HIMMA basada en el Subsistema IP Multimedia, que hereda las ventajas inherentes de la arquitectura IMS (basado en estándares, orientado a sesiones multimedia, rica en habilitadores, etc.) y la extiende aportando una capa de soporte a aplicaciones multimedia multiusuario de alta interactividad.

Las extensiones propuestas se plantean tanto desde el lado servidor, mediante la introducción de un habilitador de soporte a actividades multiusuario (LCE), como desde el lado cliente, mediante un middleware (HIMMA-MW) basado en Java que facilita el desarrollo de aplicaciones HIMMA sin necesidad de soporte en el lado de servidor.

Los siguientes capítulos profundizan en los detalles de diseño, protocolos de comunicación empleados y extensiones propuestas, así como aspectos de implementación y validación del habilitador LCE (Capítulo 4) y el middleware HIMMA (Capítulo 5). La aplicabilidad a distintos entornos se presenta en el Capítulo 6.



## 4. Habilitador ligero de gestión de actividades (LCE)

### 4.1 Introducción

Como se indicó en la sección 2.2, el Subsistema IP Multimedia (IMS) pretende proporcionar un conjunto de servicios más allá de la conectividad IP, intentando evitar problemas de interoperabilidad y duplicación de funcionalidad entre aplicaciones. Sin embargo, como vimos en 2.2.2, los servicios de comunicación multiusuario definidos en IMS son realmente aplicaciones finales enfocadas a proporcionar comunicación de audio y vídeo de forma centralizada.

El habilitador ligero de gestión de actividades (*Lightweight Conferencing Enabler* o LCE), presentado en la sección 3.2, busca proporcionar un marco genérico de soporte a aplicaciones multimedia multiusuario de alta interactividad independiente del tipo de medio y de la topología. Por tanto, el habilitador propuesto puede utilizarse no sólo en aplicaciones de audio o vídeo sino en otro tipo de aplicaciones como juegos en red, aplicaciones de trabajo colaborativo o aplicaciones específicas de otros ámbitos que requieran la gestión de actividades multiusuario en IMS de forma sencilla.

La filosofía del habilitador LCE es ser lo más sencillo y ligero posible para ofrecer un valor añadido a las aplicaciones que lo usen y no distanciarse de los protocolos y arquitectura estándar de IMS. De este modo, el habilitador no define nuevos protocolos de comunicación, aunque sí extiende los grupos compartidos (*shared groups*) de OMA descritos en 2.4.5 y propone reglas de diseño para fomentar la interoperabilidad entre aplicaciones.

El habilitador puede implementarse con la tecnología Java SipServlet y desplegarse en un servidor de aplicaciones. Es ligero porque toda la información de estado necesaria se almacena en el XDMS, con lo que no sufre problemas de escalabilidad<sup>4</sup> siempre y cuando el XDMS esté bien dimensionado. Por otro lado, el habilitador delega el tratamiento de medios a los clientes o a servidores específicos (un servidor de juegos, p.ej., o un mezclador de vídeo), por lo que el habilitador no realiza operaciones exigentes en CPU o memoria que pudieran saturar el servidor de aplicaciones.

## 4.2 Arquitectura

La arquitectura del habilitador ligero de actividades, LCE, se presenta en la Fig. 25. Como ocurre con otros habilitadores de servicio, la lógica del habilitador está desplegada en el servidor de aplicaciones y cuenta con interfaces hacia los nodos de IMS requeridos. Concretamente, el habilitador LCE requiere un núcleo de IMS estándar (CSCFs y HSS) y un entorno OMA XDM 2.0 que cuente con un servidor XDMS con soporte para grupos compartidos (especificación *Shared Group 1.0*).

La arquitectura a tan alto nivel es similar a la de aplicaciones como pulsar para hablar (PoC, que también emplea XDMS para la gestión de usuarios), si bien hay diferencias significativas. La más relevante es que el habilitador es independiente del tipo de medios y permite a las aplicaciones utilizar sus propios tipos de medios (p.ej. datos de un juego en red) así como topologías (centralizada o distribuida) como se muestra en la Fig. 26. Conviene recordar en este punto que tanto los servicios de pulsar para hablar como el servicio de conferencias estándar de IMS están ligados a voz, vídeo o mensajería textual, y el procesamiento de medios corre a cargo de los servidores de PoC y MRF respectivamente de manera centralizada.

---

<sup>4</sup> La técnica habitual para garantizar la escalabilidad de un servicio es poder incrementar el número de instancias o copias del servidor de aplicación que lo sustenta, de modo que las peticiones de los usuarios se repartan entre dichas instancias. En una aplicación multiusuario, sin embargo, es posible que peticiones SIP de usuarios que participan en la misma actividad sean tratadas por instancias distintas del servidor de aplicaciones, de ahí la importancia de que la información que haya que mantener por parte de cada instancia pueda ser compartida por el resto. En el caso de LCE esta información se almacena en el XDMS, permitiendo al propio LCE escalar convenientemente.

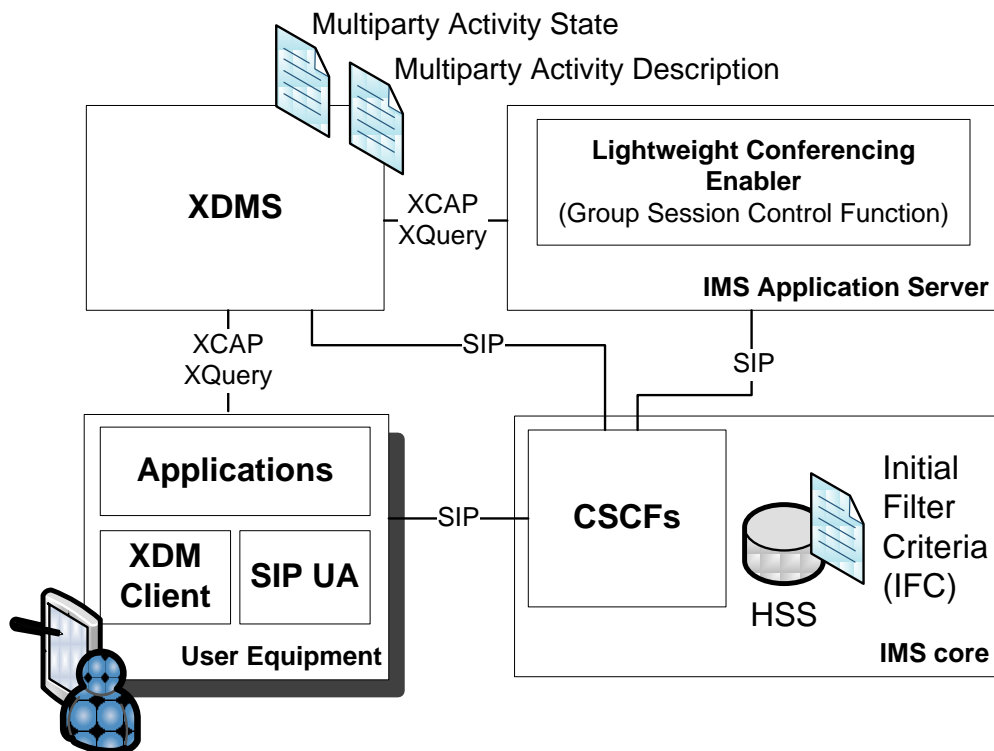


Fig. 25. Arquitectura del habilitador ligero de actividades LCE

El habilitador proporciona el descubrimiento de actividades multiusuario y fuerza la aplicación de las políticas definidas para las mismas, de modo que sólo los usuarios que cumplan con los requisitos establecidos por el organizador de la actividad puedan participar (esto es, establecer las sesiones necesarias). Teniendo en cuenta este aspecto, el habilitador LCE no actúa necesariamente como “foco” (en la terminología de comunicaciones multiusuario de SIP), sino como un intermediario avanzado.

Por otro lado, y al contrario que en las soluciones basadas en servidores PoC o MRF, el habilitador no gestiona directamente las suscripciones al estado de la conferencia. El paquete de eventos estándar de SIP para modelar el estado de una conferencia o comunicación multiusuario se especifica en la RFC 4575[55]. El modelo, definido en XML, incluye información sobre los participantes actuales, tipos de medio involucrados, descripción de la conferencia, etc. Tanto en conferencias de tipo PoC (pulsar para hablar) como basadas en MRF (audio y vídeo), los participantes se suscriben al servidor PoC o al MRF para recibir notificaciones cada vez que haya cambios en la conferencia (p.ej. un usuario ha entrado). El habilitador LCE sigue la estrategia de almacenar la información de estado en el servidor XDM, delegando en el mismo la gestión de suscripciones y cambios. Al fin y al cabo, una de las características de la arquitectura XDM es gestionar las suscripciones a documentos XML y el modelo de grupo compartido ya incluye información sobre la conferencia (ver Fig. 14), por lo

que resulta más natural almacenar el estado de la conferencia o actividad en el XDMS que en los propios servidores de aplicación, especialmente cuando el documento que describe la actividad ya está almacenado en el XDMS. Además, para aplicaciones que no dependen exclusivamente de un único servidor es necesario que un nodo independiente gestione el estado de la conferencia o actividad (p.ej. un juego multiusuario donde el tráfico de voz se mezcla en el MRF, un servidor de juegos proporciona los datos del entorno virtual y además los usuarios establecen una malla para intercambiar la información sobre posiciones en el entorno minimizando la latencia).

El habilitador LCE reutiliza el modelo de grupo compartido de OMA y propone la adición de un nuevo tipo de documento que extiende el modelo de estado de conferencias de SIP, de modo que la información de la actividad y el estado estén almacenados en el XDMS y se gestionen de forma homogénea. De este modo, el habilitador ligero de gestión de actividades actualiza el estado de la actividad en el XDMS mediante el protocolo XCAP. Por ejemplo, si un usuario se añade a una actividad, el habilitador LCE incluirá su URI en el documento de grupo compartido que la describe. Automáticamente, el XDMS notificará al resto de participantes (que previamente se han suscrito a los cambios del grupo compartido que define la actividad).



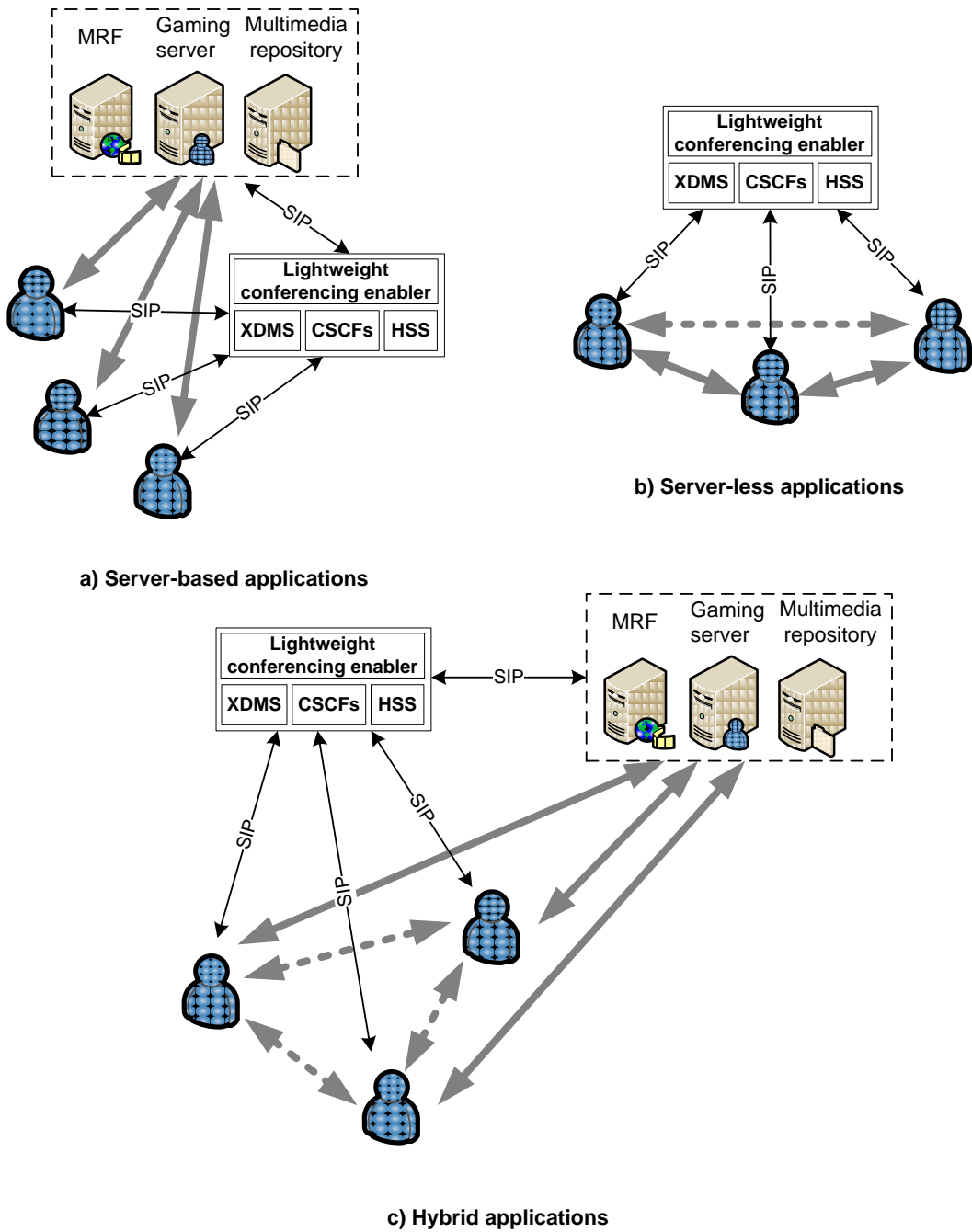


Fig. 26. Arquitecturas soportadas por LCE

### 4.3 Modelos de datos

El habilitador ligero de gestión de actividades LCE trabaja principalmente con dos tipos de información: la descripción de la actividad y el estado de la misma. Ambos están modelados como documentos XML para su almacenamiento y gestión en el servidor XDM.

### **4.3.1 Modelo de descripción de la actividad**

El modelo de descripción de la actividad se rige por la definición del modelo *Shared Group 1.0* de OMA, repasado en 2.4.5, con el fin de asegurar la integración con otras aplicaciones que hagan uso de esa información.

### **4.3.2 Modelo de estado de la actividad**

El modelo de estado de la actividad es una extensión del modelo de conferencias definido en la RFC 4575 y que constituye el estándar seguido por la solución de audio y vídeo conferencia de IMS basada en MRF. El modelo de estado de conferencia definido en la RFC 4575 permite ofrecer información sobre la conferencia a los usuarios que participan en la misma. Este modelo está representado en la Fig. 27 e incluye información básica como la descripción de la conferencia, su URI, el número de participantes y los tipos de medios utilizados.

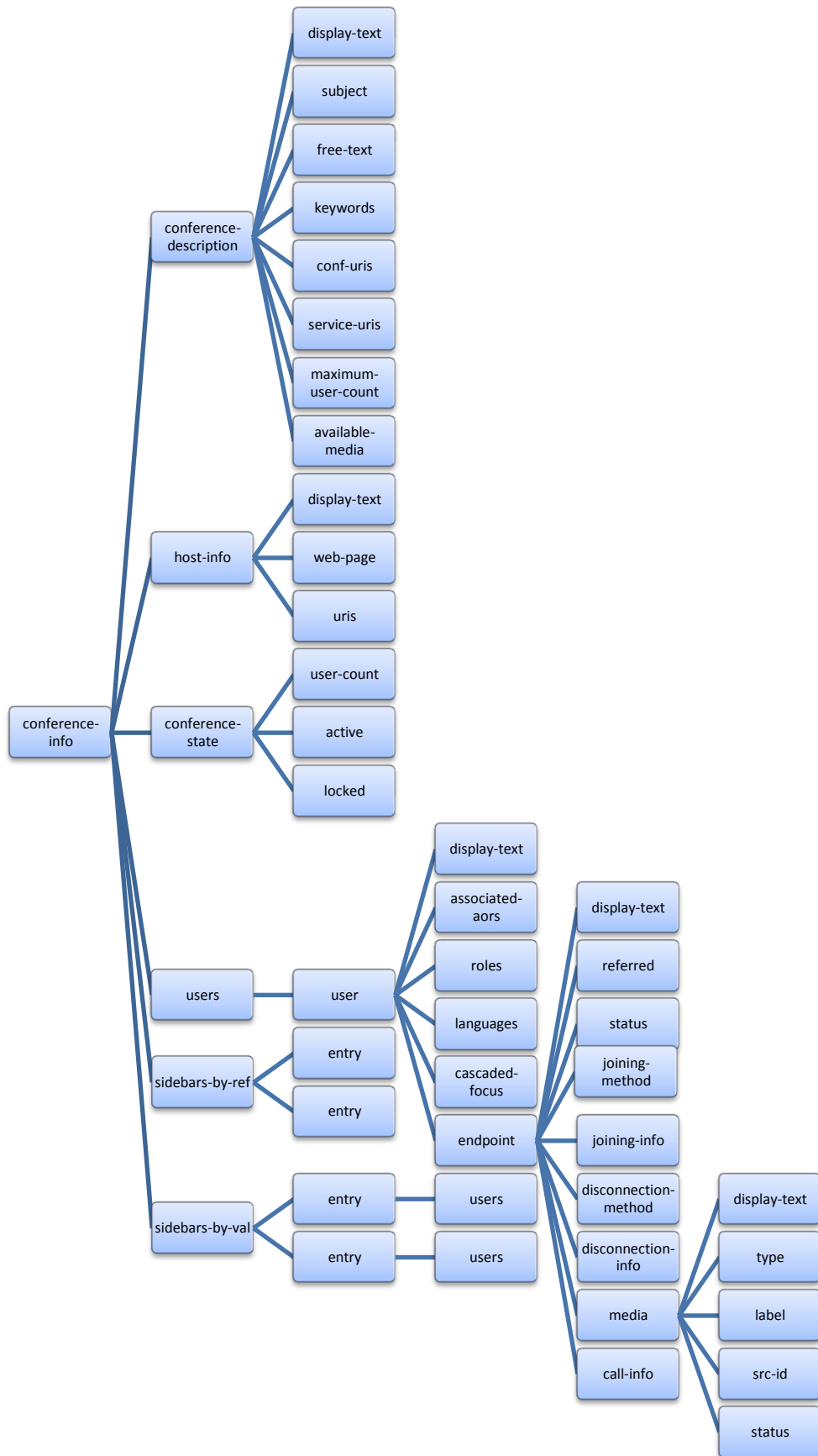


Fig. 27. Modelo de conferencia SIPCONF (RFC 4575)

Además de la información general sobre la conferencia o actividad, cada aplicación requiere el mantenimiento de cierta información de estado específica del tipo de actividad. El modelo de estado de conferencias de SIP y, por ende, de IMS, está pensado para aplicaciones de voz y vídeo conferencia donde no hay más estado que la identificación de los interlocutores que están hablando. En la Tabla 1 se indican algunos ejemplos de actividades interactivas con dos o más usuarios y la información de estado específica de la actividad que es necesario mantener desde un punto de vista funcional. Se excluye, por tanto, la información necesaria en cualquier actividad independientemente de su tipo, como los identificadores de los participantes y de la propia actividad, las características de las sesiones SIP involucradas en la misma y topología empleada, políticas de calidad de servicio y si la actividad está iniciada o ha finalizado.

Entre las actividades indicadas se encuentran aplicaciones tradicionales del mundo de las telecomunicaciones como las llamadas de voz y vídeo y las conferencias audiovisuales. La llamada es el caso más básico y carece de información de estado específica ya que toda la información necesaria se encuentra en el estado de la propia sesión SIP. Las conferencias, por su parte, además del estado de las sesiones involucradas que definen los tipos de medios y direcciones, es necesario mantener información de planificación si es una conferencia programada e información sobre los roles de los participantes y permisos en escenarios donde exista un moderador.

En otras actividades donde los usuarios participan actuando sobre un entorno compartido, como es el caso de la pizarra electrónica o cualquier juego de los que se indican, es preciso mantener la información que refleje el estado de ese entorno para que la actividad transcurra correctamente.

**Tabla 1. Información específica de estado de aplicación necesaria para distintas actividades**

Actividad	Información específica de estado de aplicación
<b>Llamada de voz y/o vídeo</b>	Ninguna. El estado de la sesión SIP entre los dos participantes es suficiente
<b>Conferencia audiovisual programada o moderada</b>	<p>Información de planificación (hora de inicio y finalización, participantes admitidos, contraseña de acceso), si está programada</p> <p>Información de control de sala (rol de cada participante - moderador, creador, etc. - , identificador del turno actual, solicitudes de turno), si está moderada</p>
<b>Pizarra electrónica</b>	<p>Contenido actual de la pizarra (formato mapa de bits o vectorial, con historial de cambios)</p> <p>Estado de la interfaz de usuario de cada participante: color escogido, trazo, fuente, área de visualización, opciones de zoom, etc.</p>

<b>Ajedrez, 4 en raya, parchís...</b>	Piezas o fichas de cada participante Situación de las fichas en el tablero de juego Turno actual Historial de movimientos Ganador de la partida si ha finalizado
<b>Juego de cartas Las siete y media, póker, UNO®</b>	Baraja de cartas Cartas sobre la mesa Cartas ocultas de cada participante Cartas visibles de cada participante Turno actual Ganador del juego
<b>Juego de carreras automovilísticas</b>	Identificación del circuito (localización, número de vueltas, tiempo a batir) Vuelta actual y tiempo de vuelta de cada participante Posición actual y vector de movimiento de cada coche Ganador de la carrera
<b>Acción en primera persona (First person shooter o FPS)</b>	Identificación del mapa o terreno de juego, dirección de descarga si aplica Estado de los objetos del mapa Información de cada participante: energía, munición, armas, arma seleccionada, muertes producidas, posición y movimiento, acción en curso
<b>Juego de rol multijugador masivo</b>	Información de estado similar a FPS al recrear espacios físicos Además, la actividad se desarrolla de forma permanente (no tiene inicio o fin) y los participantes entran y salen de ella esporádicamente. Si se desea persistencia (un participante regresa con el estado que tenía cuando la dejó) es preciso almacenar el estado de todos los participantes que alguna vez han accedido a la actividad.

Entre las actividades planteadas encontramos distintos tipos de interacción con el entorno compartido, ya sea por turnos, como es el caso de los juegos de tablero presentados, o de interacción simultánea en tiempo real, como los juegos de carreras, de disparos o de rol. La participación simultánea de todos los usuarios supone el reto de actualizar en tiempo real el estado del entorno, pero no afecta al modelo de estado en sí, que es independiente de cómo o cuándo se haya generado.

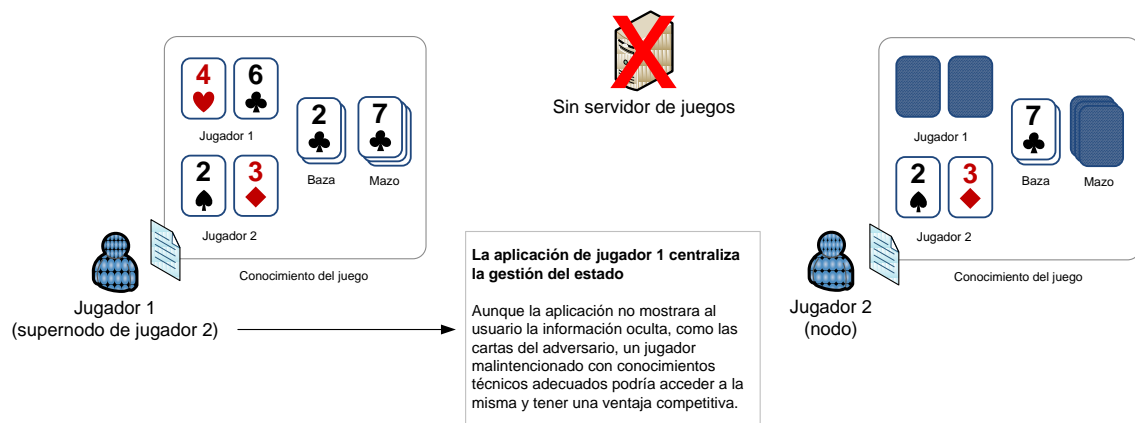
Los escenarios en los que la información de estado se mantiene y gestiona por un único nodo central son los más sencillos cuando este nodo es un servidor externo a los participantes, pues toda la información se procesa en el servidor y éste envía la información que precise cada participante en cada momento. En la Fig. 28 se muestra

un ejemplo de juego de cartas típico donde toda la información de estado se encuentra en el servidor y los participantes sólo conocen la parte correspondiente en función de las reglas de la actividad (p.ej. un participante no puede ver las cartas de los adversarios ni saber cuál es la siguiente carta de la baraja).



Fig. 28. Gestión de estado centralizada en servidor de juegos (caso cliente-servidor)

En los juegos multijugador actuales es frecuente recurrir a soluciones basadas en P2P en la medida de lo posible para disminuir el coste del despliegue de infraestructura específica, atendiendo a topologías totalmente descentralizadas o centralizadas en los usuarios que ejercen el papel de supernodo. Uno de los inconvenientes de las alternativas P2P es la exposición potencial de información de estado a participantes que no deberían conocerla. Si en vez de usar un servidor de juegos externo para mantener el estado como indicaba la Fig. 28, delegamos la gestión de estado en un participante que ejerce de supernodo, ese participante tendrá acceso potencial a información que debería permanecer oculta, como muestra la Fig. 29.



**Fig. 29. Gestión de estado centralizada en un supernodo (caso P2P)**

Evidentemente, la información sólo corre riesgo de ser desvelada en caso de que haya participantes malintencionados que tengan los conocimientos técnicos adecuados. Una aplicación de un juego de cartas no mostrará en pantalla las cartas del adversario al jugador, sin embargo un usuario malintencionado podría modificar la aplicación (que está ejecutándose en su dispositivo) para desvelarlas o capturar y modificar el tráfico de red intercambiado entre los participantes. Las plataformas de juego actuales se basan en la denominada *security through obscurity*, es decir, prevenir el efecto de usuarios malintencionados ocultando los mecanismos utilizados. Por ejemplo, en la consola Xbox 360, el protocolo de comunicaciones entre el hardware de las consolas se desconoce. Además, el propio hardware sólo ejecuta código firmado por el fabricante, evitando así la modificación del software o la creación de software intermedio que permita extraer la información deseada por el usuario malicioso.

La solución para minimizar la infraestructura necesaria para la actividad y garantizar que la información de estado está distribuida conforme al conocimiento que puede tener cada participante, la encontramos en la Fig. 30.

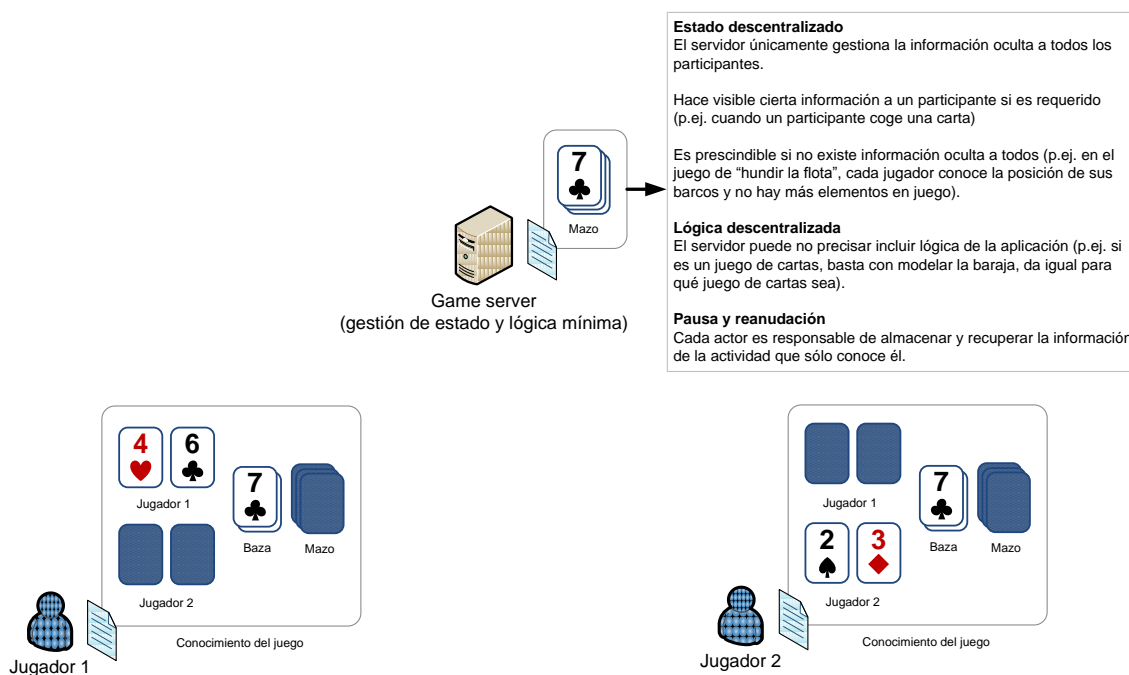


Fig. 30. Gestión de estado distribuida con soporte potencial de servidor de juegos (caso P2P)

Con esta nueva configuración, cada participante sólo conoce aquella información visible por él, lo que incluye sus propios elementos ocultos al resto y los elementos visibles por todos. Por otro lado, el servidor de juegos únicamente ha de gestionar la información oculta para todos los participantes, que en el caso de un juego de cartas es el orden y composición de la baraja utilizada. En este escenario se ahorra infraestructura específica, pues se ha simplificado el papel del servidor de juegos, sin embargo se precisa introducir mecanismos de seguridad apropiados que impidan hacer trampas por parte de los jugadores aprovechándose de la ocultación de información (cartas propias, p.ej.) o, al menos, identificarlas para no confiar en ese jugador.

Nótese que estas restricciones relacionadas con la gestión de estado no están presentes en actividades donde toda la información está a la vista por todos los participantes, como ocurre en una conferencia audiovisual donde todos se escuchan y saben lo que ha dicho el ponente, o en juegos de tablero como el ajedrez o las 4 en raya donde todas las piezas o fichas están a la vista. Los juegos de carreras automovilísticas o de tenis tampoco requieren ocultación de información porque la posición de los coches suele ser conocida, así como de la bola y los jugadores en un juego de tenis.

Sin embargo, actividades como el juego de hundir la flota, donde el juego consiste en adivinar la posición de los barcos enemigos en el tablero, o los juegos de disparos,



donde los adversarios no deben conocer la munición restante del jugador, p. ej., sí requieren la ocultación de información entre jugadores en un entorno distribuido.

Otros juegos, como los ya citados juegos de cartas o que se basen, en general, en el azar y en situaciones desconocidas para todos los participantes involucrados, requieren gestionar información oculta para todos y, por tanto, es necesario el apoyo de un tercero de confianza, habitualmente en forma de servidor de juegos imparcial y controlado por el desarrollador de la actividad.

Por ello, a la hora de crear un modelo de estado distribuido no hay que contemplar el estado como un todo sino como un conjunto de informaciones que deben ser distribuidas respetando la visibilidad de cada una entre los participantes. Identificamos así los siguientes tipos de información que componen el estado:

- **Información visible para todos los participantes de la actividad.** Incluye atributos de objetos visibles en el entorno compartido, atributos visibles de cada jugador (indumentaria, avatar, etc.) y cualquier tipo de información que pueda y deba ser conocido por todos los participantes.
- **Información oculta para todos los participantes de la actividad.** Incluye información que ningún participante de la actividad puede conocer. No aplica en actividades donde la información la generan íntegramente los usuarios, como en una conferencia de audio o vídeo, pero sí en juegos de azar o, en general, donde los jugadores no puedan conocer ciertos datos del entorno.
- **Información visible para uno o varios participantes de la actividad y oculta al resto.** Incluye información de carácter privado como participante dentro de la actividad, como las cartas propias en un juego de póker o la munición restante en un juego de disparos, cuyo conocimiento por parte de los demás participantes conllevaría una desventaja competitiva en un juego o la exposición de información no deseada a los mismos.
- **Información visible para uno o varios participantes de la actividad, sin importar el resto.** Incluye información que debe ser conocida por uno o varios participantes pero que su exposición a los demás no resulta relevante, como por ejemplo las preferencias de interfaz de usuario en una pizarra electrónica (área de visualización, zoom, paleta de colores seleccionada, etc.).

El modelo de datos de conferencia propuesto para XCON, disponible en [56], no incluye información de estado de la conferencia más allá de si está programada, iniciada o ha finalizado, centrandó su atención en la organización de las distintas

sesiones, flujos de medios involucrados y topología subyacente de los nodos mezcladores que intervienen en la conferencia. En un entorno descentralizado, las características de las comunicaciones entre los participantes sólo resultan de relevancia para los participantes involucrados, y éstos pueden inferir dicha información del estado de las sesiones que mantienen asociadas a la actividad.

Por ello, proponemos el modelo de estado genérico para actividades descentralizadas mostrado en la Fig. 31, que tiene la particularidad de estar instanciado por cada participante. El modelo incluye la siguiente información:

- **Identificación de la actividad.** Incluye el identificador de la aplicación dentro de la red IMS, así como el identificador único de la actividad multiusuario y de la conferencia asociada, en su caso.
- **Marca de tiempo.** Incluye tanto la hora universal a la que se refiere la información de estado como el tiempo relativo a la actividad (p.ej. tiempo desde que se inició un juego).
- **Identificación del organizador.** Información del participante que inició la actividad o del organizador si estaba programada.
- **Política de participación.** Si la actividad está abierta a todos los participantes, está restringida a invitaciones provenientes de los participantes activos o si sólo se permite la participación a usuarios conocidos por los participantes.
- **Lista de participantes activos conocidos.** Lista de usuarios participantes identificados por su URI o por un identificador de grupo en el gestor de grupos. Nótese que en actividades que representan mundos virtuales abiertos puede ser inviable conocer a todos los usuarios de ese mundo, en cuyo caso la lista de participantes conocidos incluye los participantes “cercaños” o “influyentes” dentro del ámbito de la actividad.
- **Lista de participantes pasados conocidos.** Lista de participantes que han formado parte de la actividad alguna vez.
- **Información de planificación.** Si la actividad está creada, planificada, en curso, parada, finalizada o cancelada. Incluye marcas de tiempo de los eventos relevantes y comentarios textuales que apliquen (razón por la que se ha cancelado una actividad, p.ej.).
- **Elementos de información específica de estado (*state-slices*).** Contiene la información específica sobre la actividad conocida por el participante, teniendo en cuenta los distintos tipos de información de estado posibles en un escenario descentralizado y los requisitos de ámbito y visibilidad exigidos por cada uno que se expusieron en este apartado. Puede existir, y

de hecho es frecuente, más de un elemento de información de este tipo en la instancia del modelo de estado de actividad.

Este modelo de estado es imprescindible para soportar funciones avanzadas como la transferencia de sesión o la pausa y reanudación, que sólo son posibles de forma general si existe un mecanismo para almacenar y recuperar el estado de la actividad incluso en distintos terminales.

Los elementos de información específica de estado (denominados en la Tesis *state-slices*) pueden ayudar a la aplicación a simplificar su lógica o mejorar su escalabilidad al contar con un mecanismo de almacenamiento del estado proporcionado por la red IMS y con soporte para suscripciones y notificaciones de cambios.

Hay que tener en cuenta que puede no ser viable almacenar y actualizar toda la información de estado directamente en el XDMS si se cuenta con requisitos estrictos de tiempo real, como por ejemplo un juego de disparos o un mundo virtual donde las acciones han de comunicarse en décimas de segundo, o una pizarra virtual donde se quiera ver cada trazo según se dibuja (en contraposición a ver el trazo una vez finalizado). En el caso de la pizarra, por ejemplo, tiene sentido almacenar los trazos ya pintados como información de estado en el XDMS, sin embargo la comunicación en tiempo real de cómo se va formando ese trazo se debe considerar tráfico de la actividad y se gestionará por los medios que proporcione la aplicación. El habilitador LCE en este tipo de casos facilita el establecimiento de las sesiones necesarias en función de si la actividad requiere un servidor de soporte o es descentralizada, así como la pausa y reanudación o transferencia a otro dispositivo.

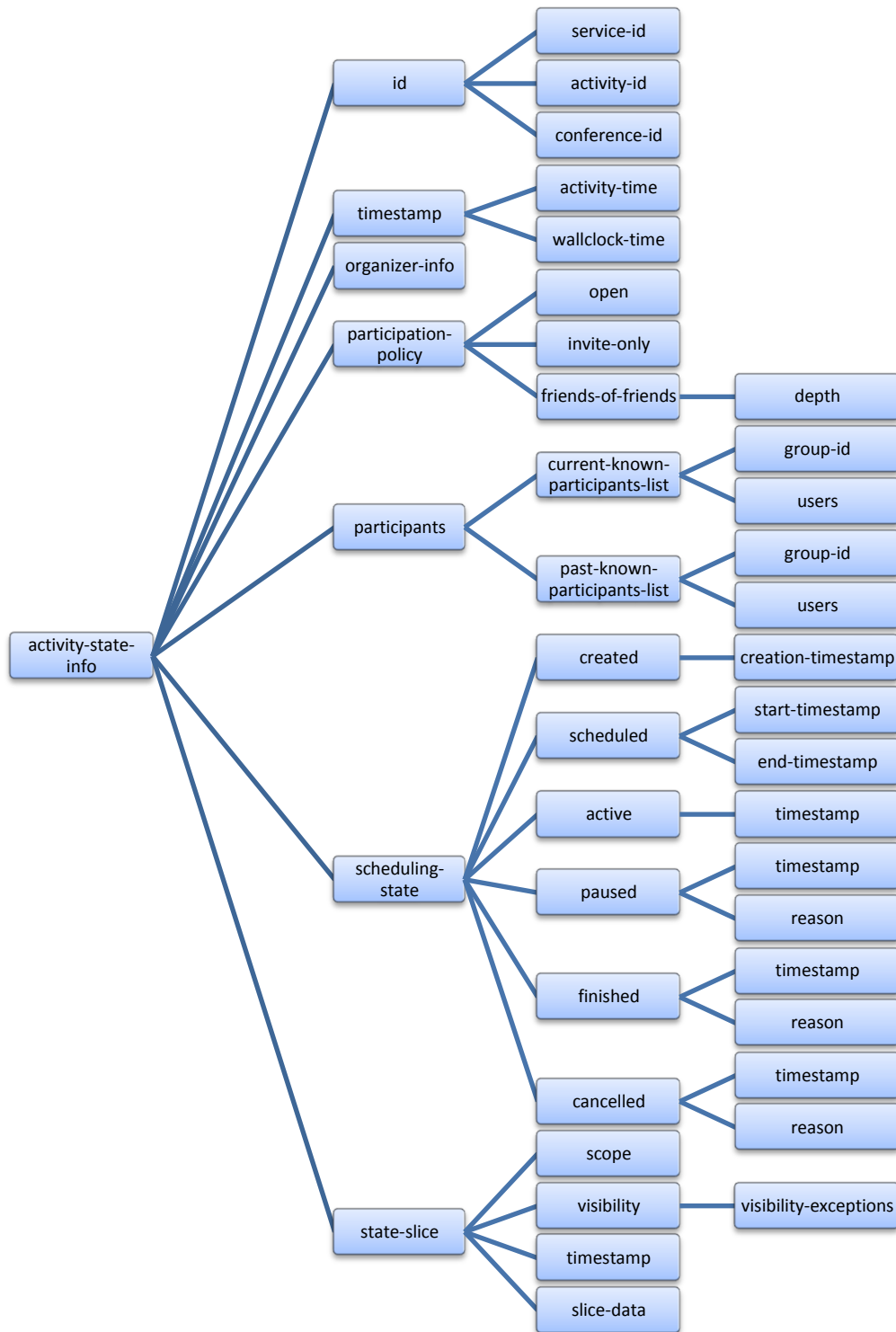


Fig. 31. Modelo de información de estado de actividad multiusuario distribuida

## 4.4 Gestión de actividades multiusuario con LCE

### 4.4.1 Descripción de la actividad

El documento de descripción de la actividad debe almacenarse en el XDMS y se crea bien por un usuario que vaya a participar o por el organizador de la actividad. Incluye el tipo de actividad, si es centralizada (y qué servidor usa) o distribuida, condiciones para los participantes, etc. Se basa en el estándar de *Shared Group* de OMA.

El objeto de este documento es doble. En primer lugar, configurar la actividad de cara a que el habilitador LCE pueda gestionarla. En segundo lugar, que otros usuarios puedan buscar en el XDMS actividades de su interés y unirse si cumplen las condiciones de participación.

### 4.4.2 Unión a una actividad

La unión a una actividad implica el establecimiento de una o varias sesiones SIP, en función de la topología requerida por la actividad. El primer mensaje de INVITE del usuario tiene que ir dirigido a la URI única que identifica la actividad y que se asigna automáticamente al crear su documento de descripción en el XDMS. Alternativamente, puede establecer sesión con la URI de factoría de la actividad a la que desee acceder (de modo similar al mecanismo estándar de creación de conferencias audiovisuales en IMS). Esta segunda alternativa es la presentada en el paso 1 de la Fig. 32, en caso de conocer la URI directamente por el documento de descripción de la actividad, los pasos 1 a 6 de la figura no serían necesarios.

El usuario que ha creado la actividad ya conoce la URI de la misma y puede unirse en caso de querer participar. Los demás usuarios disponen de varias vías para conseguir establecer la primera sesión:

- Descubrimiento de la URI por páginas web, foros, correo electrónico, etc. Habitual en caso de actividades organizadas que quieran promocionarse.
- Descubrimiento de la URI por medio de búsqueda en el XDMS, supuesto que la actividad es pública. En este caso el usuario se comunica mediante el protocolo estándar XCAP/Xquery con el XDMS para buscar actividades que cumplan sus requisitos. El XDMS devolverá un conjunto de descripciones de actividad que se ajusten a las condiciones expresadas en la consulta y la URI única de cada una.

- Un participante invita al usuario a unirse a una actividad (con el método SIP REFER). En ese caso no es necesario que el usuario conozca explícitamente la URI de la actividad porque ya se incluye en el mensaje de REFER.

Cualquier de los casos anteriores resulta en el envío de un mensaje SIP INVITE dirigido a la URI de la actividad. El INVITE se encamina al habilitador LCE a través de los CSCF tras la inspección de la URI. El LCE entonces se encarga de obtener el descriptor de la actividad del XDMS, así como el perfil del usuario y analizarlo para comprobar que el usuario cumple las condiciones de participación y cursar la sesión.

El LCE se encarga igualmente de actualizar el estado de la actividad (participantes activos, medios utilizados, etc.), que guarda en el documento de estado XML dentro del XDMS al que los usuarios pueden suscribirse para recibir notificaciones del cambio de estado.

### 4.4.3 Búsqueda de actividades

La búsqueda de actividades sigue los mecanismos estándar establecidos por OMA para la búsqueda de documentos en el XDMS. Para que una actividad pueda aparecer en los resultados de búsqueda, el atributo *searchable* del documento de descripción de la actividad debe tener valor verdadero.

Las consultas de búsqueda se realizan mediante el protocolo XCAP/Xquery y en función de la implementación deben ajustarse a plantillas de búsqueda predefinidas en el servidor XDM (p.ej. que sólo pueda buscarse documentos especificando el tipo de aplicación y las condiciones de participación).

El resultado de la consulta al XDMS es una lista de URIs de servicio, con las que el usuario puede unirse a la actividad – supuesto que cumpla las condiciones de participación, que no se comprueban en el momento de la búsqueda – iniciando el proceso de establecimiento de sesión con dicha URI.

A continuación, en el Listado 1 se muestra un ejemplo de consulta de actividades (se excluyen las cabeceras del mensaje HTTP POST de acuerdo a XCAP), para una búsqueda de partidas a un juego de ajedrez o de damas:

**Listado 1. Ejemplo de búsqueda de actividades con Xquery**

```
xquery version "1.0";
declare default element namespace "urn:oma:xml:poc:list-service";
declare namespace ext="urn:oma:xml:xm: extensions";
for $g in collection org.openmobilealliance.groups/users/group/list-
service[ext:searchable="true"]
where [$g/ext:subject="ChessGame" or $g/ext:subject="PokerGame"]
return <group>{$g/@uri} {$g/display-name} {$g/ext:subject}</group>
```

La respuesta es un conjunto de resultados especificando la URI de servicio, la descripción dada por el organizador y el tipo de juego.

#### 4.4.4 Suscripción a eventos de la actividad

Según el estándar de creación de conferencias multiusuario de SIP, dado que el protocolo SIP está orientado a sesiones uno a uno, las aplicaciones multiusuario se suscriben a la información de eventos de conferencia del servidor con el que han establecido sesión con el fin de obtener información actualizada sobre los participantes en la misma y la propia conferencia. En el caso de actividades gestionadas por el habilitador LCE, la información sobre el estado de la actividad se crea en el XDMS por el LCE al incorporarse el primer participante en el XDMS y se actualiza conforme haya cambios en la misma.

Dado que las sesiones se establecen a través del LCE, que actúa de intermediario, las peticiones de suscripción se realizan al mismo y éste se encarga de redirigirlas al documento dentro del nodo XDMS que almacena el estado de la actividad conforme al modelo especificado en 4.3.2.

#### 4.4.5 Finalización de la actividad

La actividad finaliza cuando su información de planificación (*scheduling-status*) tenga el valor *finished* (finalizada). Al finalizar la actividad, se marcará su atributo *searchable* a falso si no lo estuviera y se borrará el documento de descripción y el de estado.

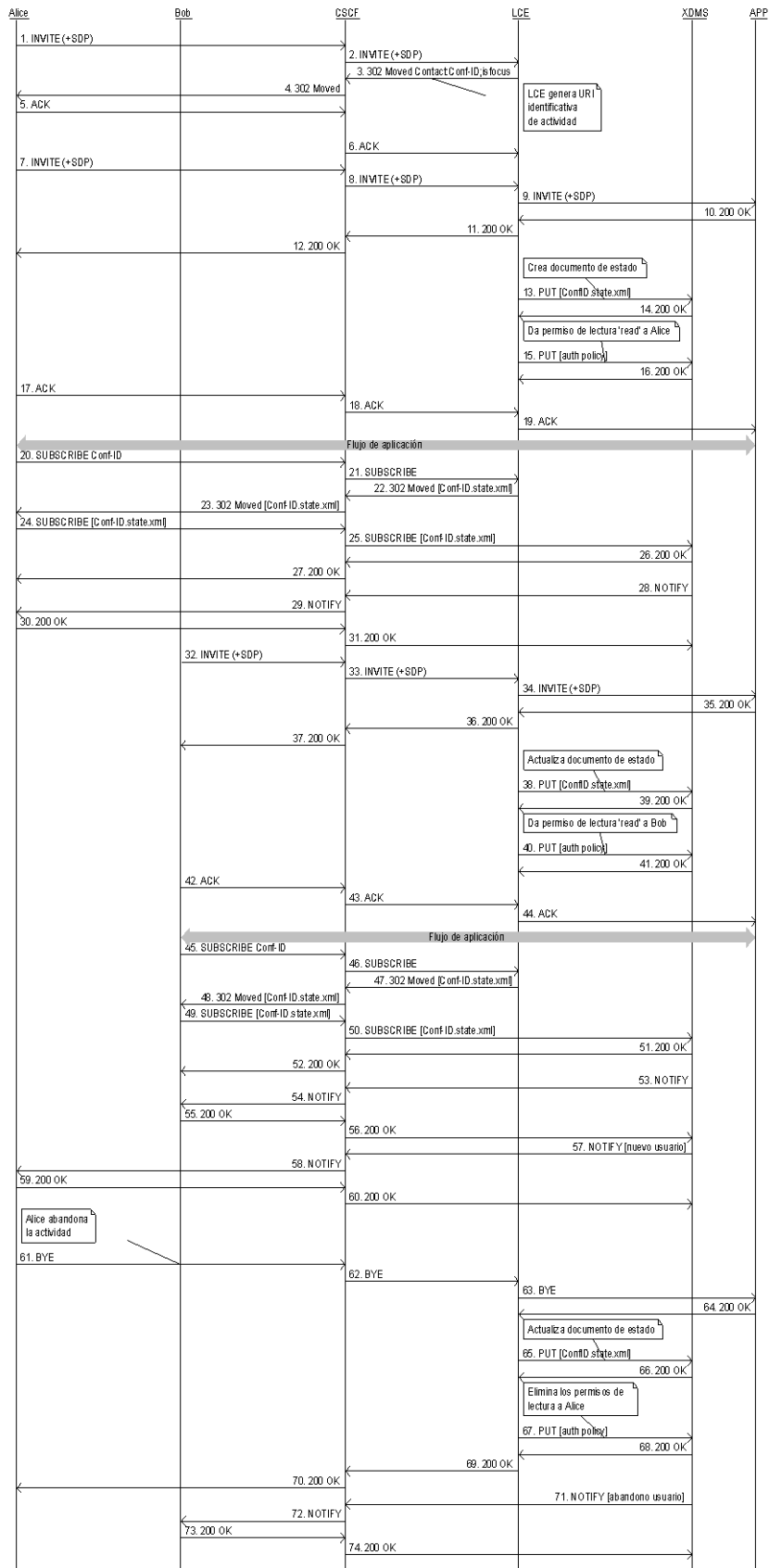


Fig. 32. Establecimiento de una actividad multiusuario basada en servidor central



## 4.5 Funciones avanzadas del LCE

### 4.5.1 Pausa y reanudación de actividades

La pausa y reanudación de actividades es una práctica habitual en aplicaciones que se ejecutan en local, donde la aplicación guarda su estado en algún medio de almacenamiento del equipo y permite recuperarlo en otro momento. Este mecanismo resulta especialmente útil en actividades que requieren mucho tiempo para ser finalizadas, como la edición de un documento o la finalización de un juego de estrategia, por ejemplo.

En el caso de actividades multiusuario, sin embargo, no siempre encontramos la posibilidad de detener la actividad y reanudarla más adelante. Las aplicaciones web multiusuario, por ejemplo, lo pueden permitir por su naturaleza centralizada donde el servidor web debe mantener el estado de la aplicación ya que los protocolos habituales como HTTP o TCP carecen de estado. En otro tipo de aplicaciones, como los juegos en red, lo habitual es que las partidas multijugador se inicien y finalicen en una misma sesión porque no implementan mecanismos auxiliares de pausa y reanudación. Una excepción son los juegos de rol multijugador masivos (MMORPG), donde el participante juega en un mundo virtual que está disponible siempre y las funciones de persistencia permiten que se conecte y desconecte de la actividad almacenando en un servidor las características del jugador dentro del mundo.

En redes IMS, el principal exponente de actividad multiusuario en el ámbito de la estandarización es la conferencia multimedia, que ya hemos descrito anteriormente, prestado de forma centralizada a través del nodo MRF que se encarga de gestionar la señalización y la mezcla de medios, ejerciendo los roles de foco y mezclador respectivamente. Si, en el transcurso de una conferencia, los participantes precisan postponer la conversación, deberán finalizar la conferencia en curso e iniciar una nueva en otro momento, volviendo a invitar a todos los participantes. La ausencia de un mecanismo de pausa y reanudación no tiene demasiado impacto en este caso, pero imaginemos que la conferencia se complementa con un servicio de pizarra electrónica: al haber creado una conferencia nueva, la pizarra asociada a esa conferencia estaría en blanco.

### 4.5.2 Transferencia de sesión con estado

La transferencia de sesión es un mecanismo habitual en entornos convergentes donde el acceso a los servicios se produce desde múltiples redes y dispositivos, de modo que una actividad iniciada en cierto dispositivo o red pueda continuarse en otro dispositivo o red.

Si bien en IMS existe el concepto de transferencia de sesión, heredado del método REFER de SIP definido en la RFC 3515 [57], únicamente consiste en cambiar uno de los extremos de la comunicación por otro, sin tener en cuenta la información previa que se hayan intercambiado los extremos a nivel de aplicación.

Esta limitación se hace evidente, por ejemplo, en una partida de ajedrez donde un jugador comienza la partida en su terminal móvil y, al llegar a su casa, transfiere la sesión al set-top-box de su salón. Aplicando el mecanismo de transferencia de sesiones presente en IMS, la nueva sesión SIP estaría correctamente establecida entre el equipo del salón y el otro jugador, pero en ningún momento se ha transferido el contenido del tablero de juego al equipo del salón, es decir, no se transfiere el estado de la aplicación. En telefonía y otras aplicaciones de voz o vídeo esto no es un problema, al fin y al cabo no tiene sentido volver a escuchar toda la conversación previa tras transferir la llamada.

Soluciones como [3] destacan este problema pero están más centradas en resolver la transferencia del contexto de red (especialmente los aspectos de QoS) en un entorno de múltiples redes de acceso, más que en transferir el estado de la propia aplicación, que se supone un problema específico a resolver por la misma.

Con el fin de evitar replicar funcionalidad en las distintas aplicaciones, proponemos un mecanismo genérico de almacenamiento y restauración del estado de la aplicación que aprovecha al máximo los recursos provistos por la arquitectura IMS y aporta una serie de ventajas a las aplicaciones que lo implementen.

La solución parte de que cada aplicación es capaz de generar, preferiblemente en formato XML y siguiendo el modelo de estado definido en 4.3.2, un contenido con la información de aplicación necesaria para reconstruir el estado de la aplicación (p.ej. la posición de las fichas del tablero de ajedrez y si es turno propio o del oponente).

#### 4.5.2.1 Transferencia de una sesión activa desde un dispositivo del usuario a otro

Alice está jugando una partida de ajedrez con Bob desde el teléfono móvil y, al llegar a casa, selecciona en su terminal la transferencia de sesión a su equipo multimedia de salón. El primer paso, como muestra la Fig. 33, es establecer una sesión entre el teléfono de Alice y el equipo de salón para transferir el estado de la partida, previamente generado como documento XML. Posteriormente, tiene lugar la transferencia a nivel de sesión SIP, mediante el citado método REFER, que incluye implícitamente una suscripción al estado de la transferencia para que el usuario que tiene que establecer la comunicación con el nuevo dispositivo avise al antiguo del éxito de la transferencia.

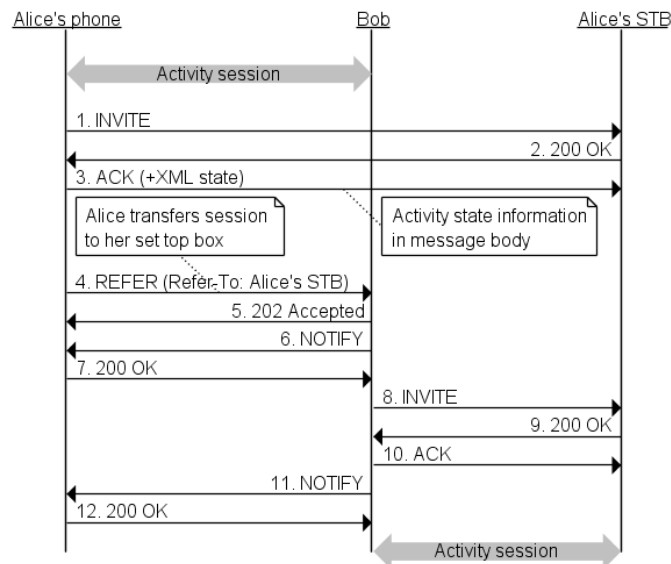


Fig. 33. Transferencia de estado entre dispositivos del mismo usuario

#### 4.5.2.2 Reanudación en un dispositivo de una sesión interrumpida en el mismo u otro dispositivo (caso P2P)

En este caso no se trata de una transferencia de sesión realmente, sino de una reanudación de la misma por haberse interrumpido bien voluntaria o involuntariamente por uno de los participantes.

El mecanismo consiste en que la aplicación que detecta el fin de una sesión almacena el estado de la misma en la memoria persistente del dispositivo, como muestra la Fig. 34 y, al recibir una petición de inicio de sesión proveniente de la misma identidad de usuario, procede a adjuntar la información de estado en la respuesta de confirmación.

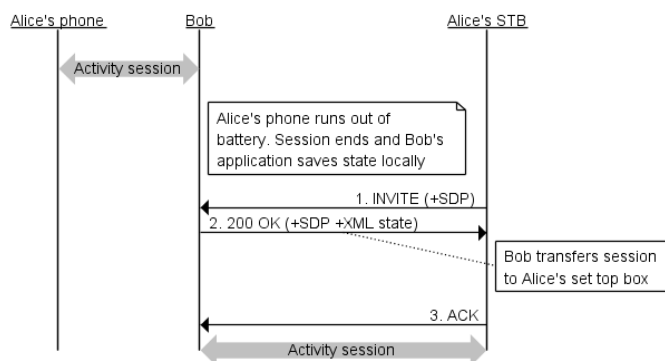


Fig. 34. Transferencia de sesión entre dispositivos de distintos usuarios

#### 4.5.2.3 Reanudación en un dispositivo de una sesión interrumpida en el mismo u otro dispositivo (caso centralizado)

Este caso es similar al anterior con la salvedad de que la información de estado no se recupera desde otro participante sino que se ha almacenado previamente en el servidor de documentos XML de IMS, pudiendo recuperarla en cualquier momento y desde cualquier dispositivo.

Volviendo al caso de la partida de ajedrez, esta alternativa permitiría a Alice recuperar el estado del tablero en otro dispositivo para pensar en su siguiente jugada sin necesidad de que el oponente esté conectado en ese momento.

El procedimiento en este caso es el mostrado en la Fig. 35. En primer lugar, la aplicación almacena el estado en el XDMS mediante una petición de tipo PUT. Posteriormente, a la hora de recuperarlo, la aplicación, aun estando ejecutándose en otro dispositivo, puede realizar una búsqueda en el XDMS utilizando el identificador de la aplicación (ajedrez, por ejemplo) y mostrar a Alice una lista con las distintas partidas de ajedrez que tiene pendientes, con lo que Alice elegiría aquella en la que Bob aparece como oponente. El mecanismo es equivalente a elegir una partida guardada en un juego, con la salvedad de que está almacenado en servidores de la red.

Con una nueva consulta al XDMS se obtiene el documento XML que define el estado de la partida seleccionada.

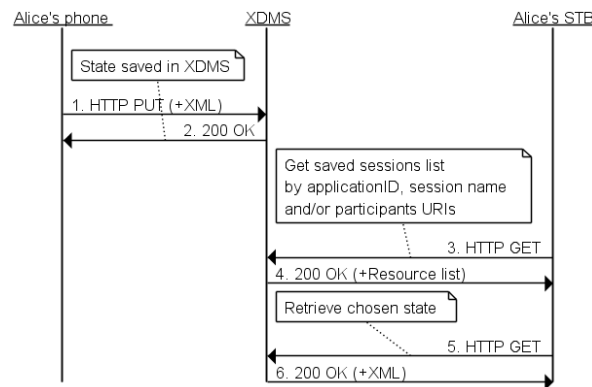


Fig. 35. Almacenamiento y recuperación de estado a través de XDMS

### 4.5.3 Soporte multidispositivo/multimodo

Las actividades gestionadas por el LCE soportan diversos modos de interacción en función de la implementación ofrecida por el desarrollador:

- Software específico en aplicación cliente (una aplicación Java Micro Edition, por ejemplo)
- Cliente SIP/IMS estándar con audio y vídeo. Interacción por sesión de audio o vídeo (requiere servidor de soporte por el desarrollador)
- Cliente SIP/IMS estándar con audio, vídeo, mensajería y navegador web. Interacción por navegador web (requiere servidor de soporte por el desarrollador, bien en el servidor de aplicaciones o en el software específico de alguno de los usuarios participantes)

El acceso a una actividad siempre comienza con la petición de inicio de sesión a la URI de servicio de la actividad, descubierta mediante el proceso de búsqueda u otros medios (invitación por otro usuario a través de REFER, p.ej.). La Fig. 36 muestra el diagrama de decisión seguido por el LCE al procesar dicho INVITE para un ejemplo de juego en red.

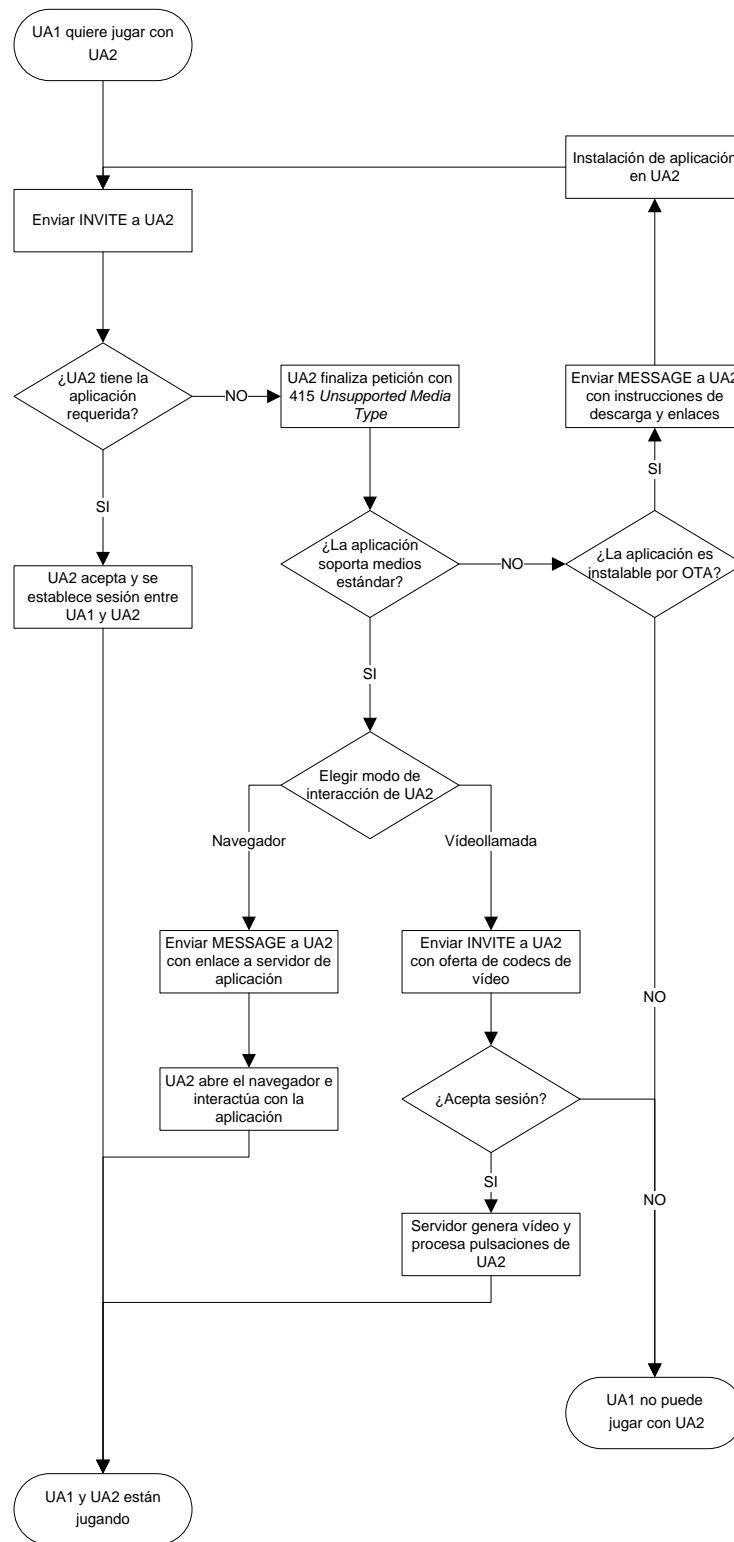


Fig. 36. Diagrama de decisión de adaptación de aplicaciones

## 4.6 Implementación y pruebas

La implementación actual de LCE está realizada en Java, concretamente haciendo uso de la tecnología *SipServlet* presentada en la sección 2.5.2, y ha sido probada en una maqueta de IMS compuesta por un núcleo IMS emulado (*Ericsson SDS*), un gestor de grupos y presencia (XDMS 1.0) también emulado por el entorno SDS y el servidor de aplicaciones de código abierto *Sun Sailfin 1.0* que es el que ejecuta el código del habilitador LCE.

La implementación mantiene el concepto de orientación a actividades multiusuario y el uso de la funcionalidad de gestión de listas de usuarios y suscripciones del XDMS, si bien la no disponibilidad práctica de servidores XDM 2.0 con soporte para *Shared Group* (la especificación está en estado *Release Candidate* desde Agosto de 2009) implica algunas diferencias en cuanto a comportamiento a la hora de desplegarlo. En efecto, la implementación actual se basa en *Shared Lists*, es decir, grupos de usuarios - entendidos como lista de URIs con atributos - como vimos en 2.4.2.

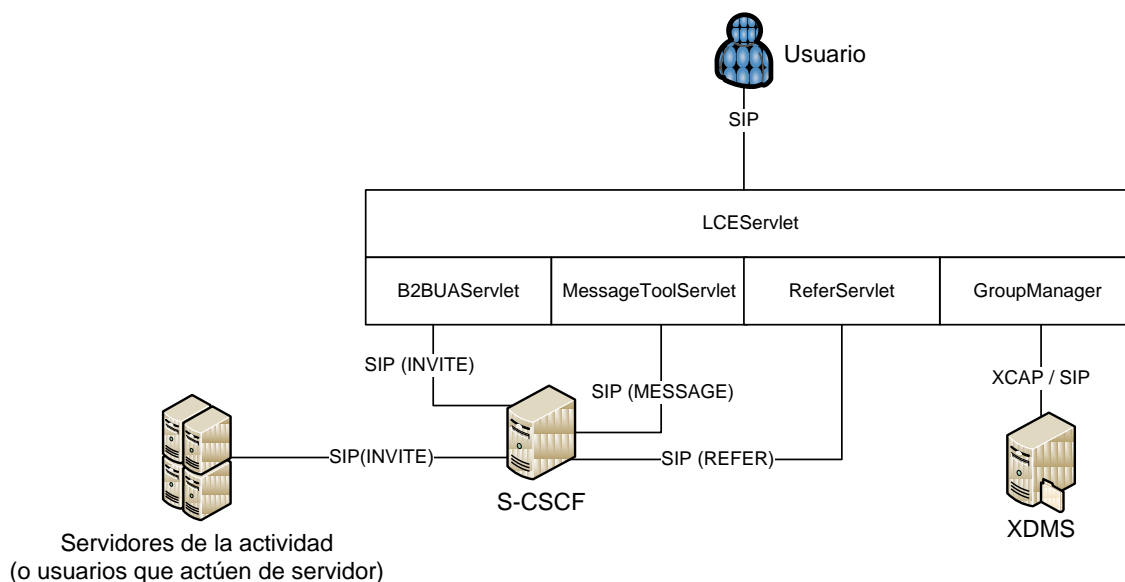
Los *Shared Group* pueden considerarse una extensión de las *Shared List* a las que se ha añadido más elementos como la definición del perfil de usuario permitido para formar parte del grupo, como vimos en 2.4.5. Funcionalmente, un *Shared Group* tiene implicaciones adicionales a la hora de establecer sesiones para aplicaciones multiusuario pues se automatizan algunas tareas como la creación de una URI única para la actividad. Las *Shared List* sin embargo son listas de usuarios y atributos, no orientados en principio a actividades multiusuario sino como agenda de contactos, si bien soportan suscripción a cambios y notificación de eventos como cualquier otro documento XML definido en el XDMS. En la implementación actual del LCE, la lista de usuarios constituye el estado de la actividad de forma biunívoca, es decir, si se añade/elimina un participante de la actividad (mediante el procedimiento de inicio de sesión a la URI de la actividad) se añade/elimina de la lista, y si se añade/elimina de la lista se añade/elimina de la actividad.

Los elementos principales de la implementación del LCE se muestran en la Fig. 37, el servlet SIP principal denominado *LCEServlet* constituye el punto de entrada de la señalización SIP correspondiente a las peticiones de inicio de sesión por los usuarios. Este servlet implementa la lógica del habilitador LCE y se apoya en módulos funcionalmente independientes desarrollados en el ámbito de la Tesis y que se detallan en [58] y [59]. Éstos son:

- *GroupManager*, es un módulo de comunicaciones con el XDMS que proporciona una interfaz Java a la lógica del habilitador y permite la gestión de grupos (listas) de usuarios y la creación y recuperación de atributos de

grupo. Las actividades asociadas a un grupo de usuarios se almacenan como atributo del grupo, así como los participantes que están formando parte en ese momento de la actividad. Se comunica con el nodo XDMS a través de las interfaces XCAP y SIP del servidor XDM.

- *B2BUAServlet*, es un servlet que facilita la implementación de la funcionalidad de *Back To Back User Agent*, necesaria para que el habilitador LCE actúe como intermediario de señalización SIP.
- *MessageToolServlet*, es un servlet que gestiona el envío de mensajes SIP MESSAGE, su aplicación es enviar los mensajes de redirección necesarios para los casos de interacción web explicados en 4.5.3.
- *ReferServlet*, necesario en los casos en que el habilitador LCE tiene que invitar a los usuarios a unirse a la actividad.



**Fig. 37. Diagrama de bloques de implementación de LCE**

La funcionalidad básica del habilitador LCE ha sido probada mediante un conjunto de pruebas de caja blanca (considerando la división de funcionalidad de la Fig. 37) y se ha validado en la práctica mediante la implementación de tres casos de uso que se describen en la sección 6.2, concretamente un servicio de videoconferencia orientado a grupos, una pizarra virtual y una sala de espera virtual (emulando el comportamiento de una sala de espera médica).



## 4.7 Conclusiones

En este capítulo se ha descrito el habilitador ligero de gestión de actividades (*Lightweight Conferencing Enabler* o LCE), un habilitador novedoso que facilita la creación de aplicaciones multiusuario en el Subsistema IP Multimedia (IMS). El habilitador es independiente del tipo de medio requerido por la aplicación, al contrario que ocurre con los servicios de conferencias estándar de IMS como pulsar para hablar (PoC) o los basados en MRF que están restringidos a audio, vídeo o texto. Además, es independiente de la topología de red requerida por la aplicación, siendo apto para aplicaciones multimedia multiusuario de alta interactividad que requieren frecuentemente arquitecturas descentralizadas como las aplicaciones de trabajo colaborativo o los juegos en red. Igualmente, el habilitador LCE permite la definición de actividades multiusuario como composición de servicios, cubriendo las necesidades de aplicaciones que requieran de distintos servidores especializados para ofrecer la funcionalidad completa (p.ej. un servicio de pizarra virtual donde los usuarios puedan hablar entre ellos, realizando la mezcla del audio el MRF y la gestión del estado de la pizarra un servidor al efecto).

La novedad principal de este habilitador es el uso intensivo de las características del servidor de gestión de documentos XML (XDMS) definido por OMA, en su versión 2.0, tales como la gestión de grupos compartidos (*Shared Group*), gestión de suscripciones a cambios en los documentos y capacidades de búsqueda. En concreto, el habilitador ligero de gestión de actividades toma el rol de nodo de control de sesiones de grupo (*Group Session Controlling Function*), conforme a la especificación de OMA. Además, se diferencia de las soluciones habituales de conferencia en IMS porque extiende el modelo de grupo compartido para incluir información del estado de la conferencia o actividad multiusuario. Así, el habilitador actualiza el estado de la actividad en el propio XDMS como documento XML, aprovechando las capacidades de este nodo para gestionar suscripciones a documentos de forma eficiente.

Las aplicaciones multiusuario que hagan uso de este habilitador se benefician de las funciones de comunidad proporcionadas por XDMS, p.ej. búsqueda y publicación de información de las actividades como grupos compartidos, al tiempo que simplifica la lógica de las aplicaciones al gestionar la aplicación de las políticas de acceso al grupo y actualizar el estado de la actividad automáticamente, incluso en aplicaciones descentralizadas sin servidor o que dependan de varios servicios centralizados independientes.



# 5. Middleware de soporte a aplicaciones HIMMA

## 5.1 Introducción

La propuesta de definición de un middleware de provisión de aplicaciones multimedia multiusuario de alta interactividad (HIMMA) en el IMS surge de la necesidad de facilitar el desarrollo de este tipo de aplicaciones aprovechando las ventajas que proporciona un entorno IMS, fundamentalmente la derivada de poder prescindir de infraestructura específica para la provisión de la aplicación.

Así, se va a presentar un *middleware*, o software intermediario, junto a su interfaz de programación de aplicaciones HIMMA que ofrezca la posibilidad de desplegarse formando topologías de red descentralizadas y soporte también arquitecturas clásicas del tipo cliente-servidor en caso de que la aplicación lo precise.

En la revisión del estado de la técnica, concretamente en la sección 2.5.1 vimos que existen interfaces estandarizadas en el ámbito de Java para el acceso a servicios IMS, que abstraen el uso del protocolo SIP para el establecimiento de sesiones, si bien sólo facilitan el establecimiento de sesiones extremo a extremo, siguiendo la misma concepción de SIP con un marcado carácter cliente-servidor.

Para facilitar el desarrollo de aplicaciones HIMMA en IMS de forma distribuida es necesario abstraer la gestión de las múltiples sesiones que componen una actividad multiusuario. Por ejemplo, un juego de tres jugadores en IMS con arquitectura cliente/servidor implica que cada cliente tiene que establecer una sesión con el servidor, sin embargo, si la arquitectura es P2P pura, cada cliente tiene que establecer dos sesiones, una con cada uno de los otros dos jugadores.

El middleware presentado se plantea como una extensión a la funcionalidad ofrecida por los interfaces de la plataforma para clientes IMS (ICP) estandarizada en las JSR 281 y JSR 325, abstrayendo el concepto de “sesión” y presentando el concepto de “actividad”, como se recoge en la Fig. 38.

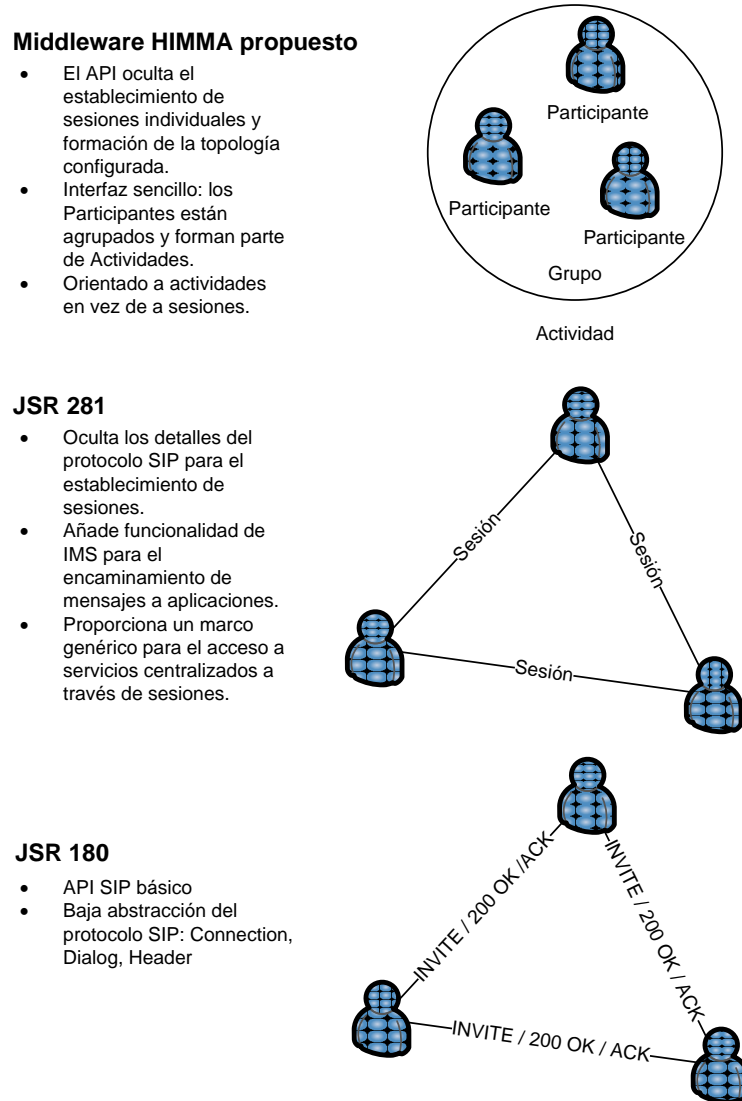


Fig. 38. Comparativa de niveles de abstracción de HIMMA API, JSR 281 y JSR 180

La implementación del middleware se encarga de gestionar las sesiones individuales entre los participantes de acuerdo a la topología de red establecida y las características de los dispositivos involucrados, de modo que el desarrollador únicamente tiene que preocuparse de añadir usuarios a la actividad y gestionar la lógica específica de la aplicación.

## 5.2 Arquitectura

Dentro de la arquitectura lógica en un escenario IMS, el middleware está situado en los dispositivos de usuario, ofreciendo sus funciones a las aplicaciones finales y basándose en las funciones IMS provistas por el API de los dispositivos. En la Fig. 39 se presenta esta arquitectura lógica en un escenario de comunicación entre dos participantes usando una (o más) aplicaciones IMS multiusuario. Se puede apreciar la abstracción que hace la interfaz ICP, haciendo la señalización SIP transparente al middleware. De igual modo, el middleware hace transparente tanto la interfaz ICP como la red IMS a las aplicaciones desplegadas por encima.

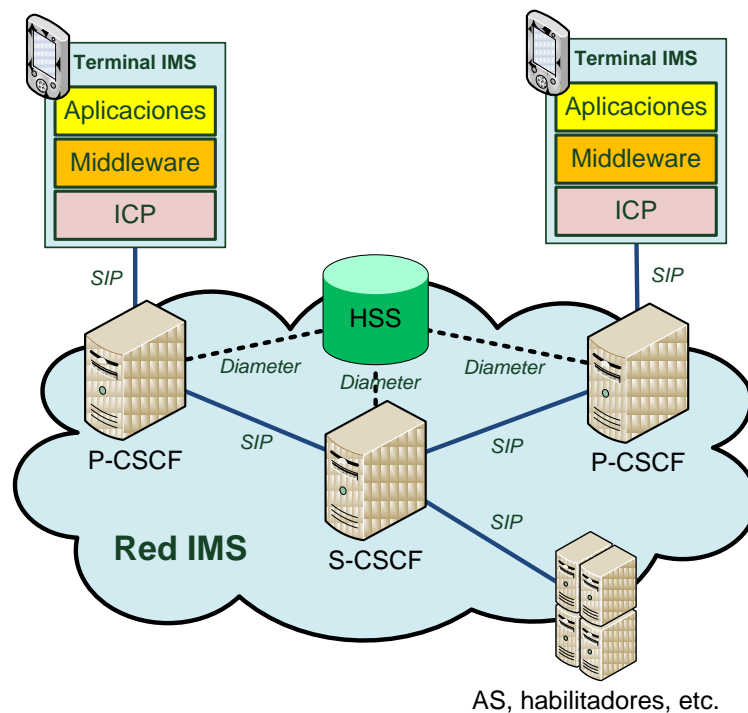


Fig. 39. Middleware de soporte a aplicaciones HIMMA dentro de la arquitectura lógica de IMS

Las aplicaciones que hacen uso del middleware se ejecutan en el terminal del usuario, en los extremos finales de la red. Hay que destacar que el diseño planteado no distingue entre extremos clientes y extremos servidores, sino que trata a todos como iguales de forma que un agente usuario cliente (*User Agent Client, UAC*) o un agente usuario servidor (*User Agent Server, UAS*) son vistos por el middleware como una misma entidad, un nodo.

Desde un punto de vista de arquitectura software, el middleware de soporte a aplicaciones HIMMA está compuesto por dos capas, como se muestra en la Fig. 40, aunque sólo la capa superior es visible para el desarrollador de la aplicación final:

- **Capa superior:** proporciona el mayor grado de abstracción al modelar las comunicaciones multiusuario como actividades, independientemente del mecanismo de comunicación subyacente, número de sesiones, etc. A ella pertenecen los conceptos de participante y grupo de usuarios, que son los elementos que forman parte de las denominadas “sesiones de actividad”.
- **Capa inferior:** proporciona conocimiento sobre la topología de red utilizada para la prestación de la aplicación, así como de las sesiones (en terminología SIP) individuales que componen la comunicación de un participante con los demás durante una actividad en curso. Esta capa permite extender la lógica del middleware con nuevos modelos de comportamiento distribuidos.

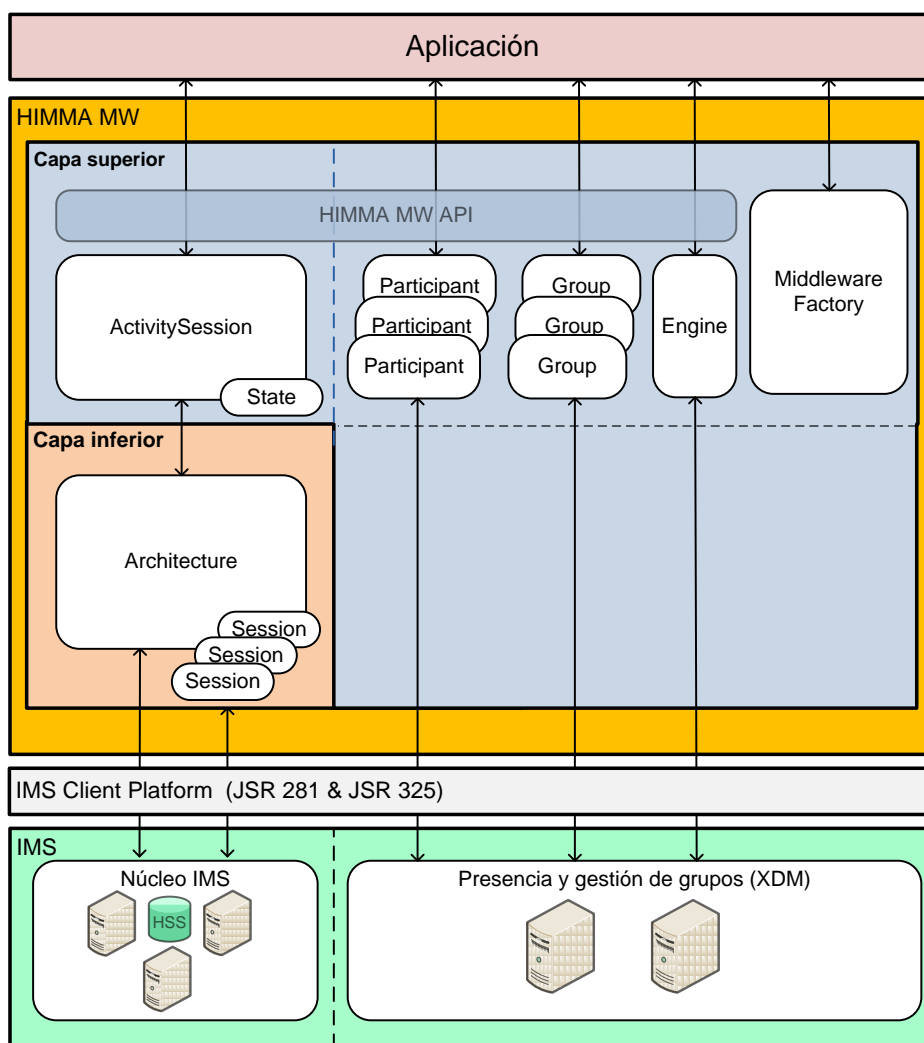


Fig. 40. Modelo de dos capas del middleware de soporte a aplicaciones HIMMA

Las características fundamentales de la **capa de mayor nivel de abstracción** son:

- Las aplicaciones manejan instancias de clase *Participant* como elemento identificador de un usuario, no SIP URIs (aunque pueden conocerlas).
- El modelo de participante incluye información sobre la URI del usuario que representa, pero le es transparente al desarrollador.
- No presenta las sesiones individuales que forman una actividad multiusuario (según sea la topología de red configurada), sino una “sesión de actividad” (*ActivitySession*) con múltiples participantes. No hay concepto de sesión SIP extremo a extremo aunque el intercambio de información se produzca estableciendo sesiones individuales entre distintos usuarios de la actividad.
- La red IMS es completamente transparente a esta capa en cuanto a la señalización SIP y transporte de datos.
- Las aplicaciones pueden interactuar con los cuerpos de las peticiones y respuestas a los servidores de presencia y grupos (XDMS) a través de las clases que modelan a los participantes (*Participant*), los grupos (*Group*) y el motor de búsqueda (*Engine*).
- Abstracción de la topología de red. Esta capa no entiende, por tanto, de nodos, roles de nodo cliente y nodo supernodo, topologías cliente-servidor y P2P, mecanismos de descubrimiento y asignación, etc. No obstante, el desarrollador de la aplicación habrá indicado previamente qué tipo de aplicación es para que el middleware pueda establecer una topología adecuada.
- La capa también se abstrae de los parámetros de la conexión y capacidades que se usan en el middleware para decidir la viabilidad de conexiones: ancho de banda mínimo y máximo, latencia, tipo de medios, tasa de bit, etc.

Por otra parte, la **capa inferior del middleware** modela el comportamiento interno asociado al establecimiento de las distintas topologías:

- Cada actividad multiusuario tiene en este nivel de abstracción un modelo llamado *Architecture*, que gestiona todas las sesiones SIP extremo-extremo (*Session*) que forman en conjunto la actividad, tanto en las funciones de señalización, como en el envío de datos en la red IMS.

- Junto con el modelo de arquitectura, existe un modelo de la sesión extremo-extremo individual. Este modelo lo implementa la clase *Session*, y modela una sesión SIP con atributos adicionales necesarios en el middleware.
- Esta capa no maneja el concepto ni el modelo de participante, sólo maneja URIs como identificador de usuario.

Los elementos identificados en la arquitectura mostrada en la Fig. 40 se han modelado directamente como clases e interfaces que componen el middleware. El middleware tiene interfaces con la aplicación HIMMA que lo use y con la red IMS a través de ICP. Se muestran organizadas en dos paquetes, la interfaz de programación de aplicaciones (paquete *es.upm.dit.ims.himma.middleware.api*) y el núcleo, que corresponde a la implementación que gestiona la topología de red y las comunicaciones con la red IMS (paquete *es.upm.dit.ims.himma.middleware.core*).

Junto con las clases relacionadas con las interfaces, en el núcleo encontramos el resto de clases que componen la implementación del middleware (*ArchitectureIMS*, *SessionIMS*, y *MiddlewareFactory*). El paquete *core* es transparente a la aplicación, que se sirve de sus servicios a través de las interfaces. Sólo la clase factoría (*MiddlewareFactory* en la figura) es usada directamente por la aplicación para instanciar nuevos objetos de las interfaces.

Además, existe una interfaz no mostrada en la figura que es *ActivitySessionListener*, conforme a la práctica habitual de aplicar el patrón observador, que debe ser implementada por la aplicación IMS desarrollada, y cuyos métodos son notificaciones desde el middleware hacia dicha aplicación. La clase que modela la actividad multiusuario (*ActivitySession*) es la que llama a estos métodos, notificando de este modo a la aplicación IMS de invitaciones, respuestas, cancelaciones y finalizaciones de actividades, mensajes y envío de datos de la red.

La relación entre las clases y las interfaces que éstas implementan se han modelado mediante una relación de generalización, la relación entre la aplicación IMS con el *Middleware*, y entre el *Middleware* y la red IMS son de tipo dependencia, ya que en ambos casos se hace uso del componente apuntado para poder realizar tareas.

## 5.3 Modelo de datos

### 5.3.1 Modelo de participante como documento de presencia

El modelo de participante incluye información sobre la participación de ese usuario en las distintas actividades en que esté involucrado, así como información



relevante para los procesos de búsqueda (datos de contacto, datos personales, etc.) e información de carácter temporal (disponibilidad, mensajes personalizados de estado, etc.).

Un participante registrado en la red IMS puede encontrarse en tres estados respecto de una actividad multiusuario:

- **No relacionado con la actividad:** el participante no tiene interés en participar ni está participando en una actividad. En tal caso la actividad no aparece en la información de presencia del participante.
- **Interesado en la actividad:** el participante tiene interés en participar en la actividad, pero no lo está haciendo aún. En este caso, la actividad aparece en la información de presencia del participante, con el estado *cerrado* (*closed*).
- **Participando en la actividad:** el participante ya está unido a la actividad. Ésta aparece en la información de presencia del participante con el estado *abierto* (*open*).

La información relativa a cada participante se publica a través del habilitador de presencia, por lo que la información permite los mecanismos propios de este habilitador: suscripción, publicación y notificación.

La interfaz *Participant* ofrece funciones para suscribirse o cancelar suscripciones a la presencia de otros participantes, y también para permitir o negar el acceso a la propia información de presencia.

La estructura de datos de presencia que envía y recibe el middleware es compatible con el estándar de presencia enriquecida definido para IMS. Recordemos que el formato estándar de presencia básico se basa en estructuras XML y se denomina PIDs (*Presence Information Data-Format*), que ha sido extendido bajo la denominación RPID (*Rich Presence Information Data-format*).

Los datos del perfil del participante del middleware se han estructurado, por tanto, en un documento de presencia con formato RPID, cuya estructura se describe a continuación.

El primer nodo del documento es el elemento *presence* de la estructura RPID y contiene toda la información del participante. Según el estándar, existen tres clases de nodos hijo posibles:

- Nodo *person*: representa a la persona, al usuario. Por tanto se ha escogido este elemento para contener la información personal del usuario, incluyendo datos personales y de perfil (nombre, edad, sexo, intereses, etc.) que se ha introducido en el subelemento *note* del elemento *person* con una estructura XML que hemos denominado *data-contact*. En segundo lugar, recoge la información de presencia actual (identificador de usuario, estado actual, humor, comentario, icono grande e icono pequeño) en el resto de subelementos de *person*. Finalmente, recoge a modo de resumen un listado con el identificador -URI- de las actividades en las que el usuario está participando.
- Nodo *tuple*: describe servicios y su relación con el usuario. Por tanto, se ha escogido este elemento para ofrecer la información de las actividades multiusuario a las que el participante está suscrito (interesado en participar) o en las que está ya participando. El elemento *tuple* contiene el identificador de la actividad, el estado del participante en ella (suscrito -*closed*- o participando -*open*-). Adicionalmente, contiene también el alias o apodo del participante en dicha actividad, y si resultase necesario, una URI de contacto específica del participante para esta actividad. Se dejan otros campos del elemento disponibles para añadir nuevos atributos.
- Nodo *device*: describe dispositivos o terminales. El middleware no hace uso directo de este elemento porque la negociación de capacidades del dispositivo se realiza por la plataforma cliente del mismo (ICP, por ejemplo).

### 5.3.2 Modelo de datos de contacto del middleware

La estructura de datos *data-contact* citada con anterioridad contiene la información de perfil y datos personales del usuario y está incluida en el subelemento *note*<sup>5</sup> del elemento *person*. Esta estructura no está estandarizada, por lo que el modelo de información es propio del middleware, si bien los campos identificados responden a la

---

<sup>5</sup> Atendiendo al esquema XML de la información RPID es posible extender con más elementos la estructura del documento de información. Sin embargo, a la hora de implementarlo, la biblioteca ICP subyacente sólo permite la obtención de los elementos estándar de información de presencia, de ahí que se haya “encapsulado” el elemento *data-contact* como una nota (*note*) en vez de como elemento dependiente de *person* directamente.

información habitual que podemos encontrar en redes sociales y aplicaciones web multiusuario actualmente:

- Nombre y apellidos: subelementos *first-name* y *last-name*, respectivamente.
- Edad y sexo: subelementos *age* y *gender*.
- Idiomas hablados: recoge los idiomas que el participante puede emplear, estableciendo una prioridad para cada uno entre 0.0 (menos prioritario) y 1.0 (más prioritario). Se organizan con la siguiente estructura XML:

```
<languages>
  <lang="language1" priority="x.x"/>
  <lang="language2" priority="x.x"/>
  ...
</languages/>
```

- Intereses y ocupación: subelementos *interests* y *occupation*.
- Música, libros, juegos y películas favoritas: subelementos *music*, *books*, *games* y *movies*.

La información de contacto publicada por el middleware está proporcionada por el propio usuario y su función principal es soportar las funciones de búsqueda.

Nótese la diferencia respecto del habilitador LCE (capítulo 4), donde el perfil de usuario está almacenado en el XDMS en forma de documento *Shared Profile*, donde hay datos como la edad que son privados y están insertados por el operador, pudiendo ser utilizados para autorizar a usuarios en función de las condiciones impuestas por el servicio. Los clientes no tienen acceso a estos datos privados en el XDMS (sólo los servidores de aplicación de confianza), por lo que el middleware –que sólo depende de software en los clientes – debe incluir su propio mecanismo para publicar información de contacto que sirva para la búsqueda de usuarios y para mostrar información del mismo configurable por el usuario como foto de perfil o aficiones.

### 5.3.3 Ejemplo de documento de presencia generado por el middleware

A continuación, en el Listado 2, se presenta un ejemplo de la información de presencia de un usuario del middleware que se enviaría al servidor de presencia (o se recibiría por otros usuarios autorizados) en formato RPID.

**Listado 2. Ejemplo de documento RPID generado por el middleware**

```

<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
  xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
  xmlns:lt="urn:ietf:params:xml:ns:location-type"
  xmlns:rpid="urn:ietf:params:xml:ns:pidf:rpid"
  entity="pres:john_doe@lab">

  <dm:person id="John">

    <rpid:activities from="2005-05-30T12:00:00+05:00"
      until="2005-05-30T17:00:00+05:00">
      <rpid:busy />
      <rpid:other>activity1@lab</rpid:other>
      <rpid:other>activity2@lab</rpid:other>
    </rpid:activities>

    <rpid:mood>
      <rpid:happy />
      <rpid:other>I'm just very happy!</rpid:other>
    </rpid:mood>

    <rpid:status-icon>
      http://example.com/pic.gif;http://example.com/bigPic.gif
    </rpid:status-icon>

    <dm:note>
      <data-contact>
        <first-name>John</first-name>
        <last-name>Doe</last-name />
        <age>22</age>
        <gender>male</gender>
        <languages>
          <lang = " es="es" " priority="0.8" />
          <lang = " en="en" " priority="0.6" />
        </languages>
        <interests>online games</interests>
        <occupation>student </occupation>
        <music>pop, classical</music>
        <books>sci-fi, Stephen King</books>
        <movies>Star Wars</movies>
      </data-contact>
    </dm:note>

    <dm:timestamp>2009-01-30T16:09:44+05:00</dm:timestamp>
  </dm:person>

  <tuple id="bs35r9">
    <status>
      <basic>open</basic>
    </status>

```

```

    <dm:deviceID>urn:device:0003ba4811e3</dm:deviceID>
    <rpид:relationship>
      <rpид:self />
    </rpид:relationship>
    <rpид:service-class>
      <rpид:electronic />
    </rpид:service-class>
    <contact priority="0.8">sip:john_doe.activity1@lab</contact>
    <note>nickname:Johnny</note>
  </tuple>

  <tuple id="ts65a8">
    <status>
      <basic>closed</basic>
    </status>
    <dm:deviceID>urn:device:0003ba4811e3</dm:deviceID>
    <rpид:relationship>
      <rpид:self />
    </rpид:relationship>
    <rpид:service-class>
      <rpид:electronic />
    </rpид:service-class>
    <contact priority="0.8">sip:john_doe.activity2@lab</contact>
  </tuple>
</presence>

```

El ejemplo anterior presenta al usuario *John Doe*, identificado mediante la URI *john\_doe@lab*. Se trata de un estudiante de 22 años que habla castellano (prioridad de 0,8) e inglés (prioridad de 0,6), interesado en los juegos en red, música pop y clásica, la saga de películas *Star Wars*, y los libros de terror y ciencia-ficción. John está participando en la actividad *activity1@lab*, e interesado en participar en la actividad *activity2@lab*. En la primera, John usa el alias “*Johnny*” y prefiere que contacten con él a la URI *john\_doe.activity1@lab* (quizá porque es la URI asociada a un dispositivo específico para esa actividad, como una consola de juegos), mientras que en la segunda prefiere que se contacte con él a través de la URI *john\_doe.activity2@lab*.

## 5.4 Gestión de actividades

### 5.4.1 Descripción de la actividad

Las actividades ofrecidas a través del middleware se corresponden biunívocamente con las aplicaciones que implementan la lógica y presentación de cada actividad. Por

ejemplo, un juego de cuatro en raya precisa que la aplicación de cuatro en raya esté instalada. Esta aplicación hará uso del middleware y tendrá que identificarse como aplicación asociada a la actividad concreta que implementa para que la señalización SIP relativa a cada actividad sea procesada por la aplicación correcta.

El identificador de servicios ICSI/IARI (*IMS Communication Service Identifier / IMS Application Reference ID*) definido en [60] proporciona un marco para la identificación de los servicios de comunicación y aplicaciones que utilizan los habilitadores de IMS. En los terminales, el uso de identificadores de servicio es un concepto similar al de los puertos de TCP/IP, en el sentido en que permite entregar los mensajes SIP recibidos a la aplicación correspondiente (p.ej. si un usuario envía una petición de inicio de sesión de un juego de las cuatro en raya a otro, el mensaje de INVITE generado debe ser procesado por el cliente de ese juego, si está instalado, y no por el cliente que atiende las llamadas de voz entrantes). De igual modo, el identificador de servicios se puede utilizar en la red IMS para seleccionar el servidor de aplicaciones concreto que deba involucrarse en el mensaje recibido, como por ejemplo el servidor de PoC (*push to talk over cellular*) o de mensajería, pero no aplicaría para aplicaciones P2P donde los terminales intercambian señalización directamente.

Los terminales de usuario pueden incluir a la hora de registrarse en la red una lista de qué aplicaciones tienen instaladas o para qué tipo de servicios están preparados, indicando para ello los identificadores ICSI/IARI en el mensaje de REGISTER, si bien la especificación no obliga a que los terminales expongan todas sus capacidades.

En el Listado 3 se muestra un ejemplo de mensaje de registro con indicación de capacidades, donde puede verse que siguen el formato de etiquetas de capacidades (*feature tags*) especificado en la RFC 3840. El ejemplo muestra soporte para un juego ficticio de cuatro en raya, cuyo tipo de aplicación es “*games.four-in-a-row*”, y para una aplicación estandarizada por el 3GPP (en concreto, la de compartición de imágenes - *image share*).

```
REGISTER sip:ims.lab SIP/2.0
Via: SIP/2.0/UDP 192.168.1.7:5070;branch=z9hG4bK2202.1fc7.89a
To: <sip:user1@ims.lab>
From: <sip:user1@ims.lab>;tag=20899fc-89a.1fc7
Call-ID: 4501-3e-1d41f7@192.168.1.7
CSeq: 2203 REGISTER
Max-Forwards: 70
Contact: <sip:user1@192.168.1.7:5070>;expires=3600;pgm.service.icp=oma;+g.games.four-in-a-row ;+g.3gpp.app_ref="urn%3Aurnxxx%3A3gpp-application.ims.iari.gsmals";
Content-Length: 0
Expires: 3600
Supported: path
```

### Listado 3. Ejemplo de REGISTER con indicación de aplicaciones soportadas

El mecanismo de exposición de capacidades es análogo para los demás tipos de mensaje SIP, como INVITE u OPTIONS, por lo que los terminales no sólo pueden registrar sus capacidades en la red, sino que pueden intercambiarlas entre ellos gracias a las etiquetas de las cabeceras *Contact* y *Accept-Contact* de los mensajes. Como ya indicamos anteriormente, el identificador IARI se utiliza para distinguir las aplicaciones y poder entregar los mensajes SIP a la que corresponda.

La Fig. 41 ilustra el uso del identificador en un terminal que tiene las funciones de voz y mensajería estándar y, además, tiene instaladas dos aplicaciones de valor añadido que hacen uso de SIP como protocolo de señalización, como podrían ser un juego y una aplicación de pizarra virtual. Los mensajes SIP entrantes que no pertenezcan a una sesión ya establecida se encaminan en función de los identificadores y el tipo de mensaje, de modo que si es un MESSAGE, por ejemplo, puede ir dirigido a una aplicación que se comunique mediante mensajería o a la propia bandeja de entrada de mensajes recibidos del usuario.

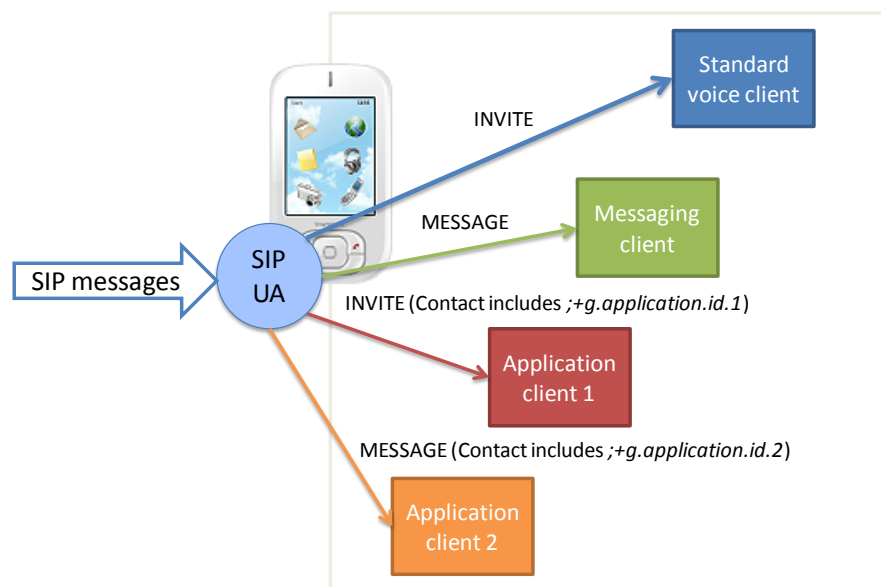


Fig. 41. Encaminamiento de mensajes SIP en el terminal basado en IARI

Nótese que un agente de usuario podría discriminar los mensajes por el tipo MIME de contenido, de forma similar a cómo los navegadores web cargan los distintos elementos de la página que requieren aplicaciones externas, como los reproductores de medios. Sin embargo, este comportamiento traería consigo una limitación importante al no permitir que haya más de una aplicación atendiendo al mismo tipo de contenido (como, por ejemplo, *application/xml*).

El middleware HIMMA propuesto en la Tesis utiliza el identificador ICSI/IARI para distinguir aplicaciones y encaminar la señalización a la instancia apropiada. El habilitador LCE, por su parte, se basa en URIs únicas para la identificación de actividades y emplea el tipo de medios de la sesión SIP como forma de discernir si el usuario tiene instalado el software específico para su tratamiento.

#### 5.4.2 Unión a una actividad

La unión a una actividad siempre comienza con un INVITE a la URI de contacto de cualquier participante de la misma. Sin embargo, la configuración de sesiones finalmente establecidas depende de la topología requerida por la aplicación.

El mecanismo propuesto en la Tesis de gestión descentralizada de actividades multiusuario en el IMS contempla tres modelos de interacción entre los participantes de una actividad:

- **Cliente-servidor.** Los participantes no intercambian señalización o tráfico de aplicación entre ellos durante el transcurso de la actividad, manteniendo sesiones únicamente con un servidor de soporte específico de la actividad.



Los participantes sí pueden intercambiar señalización entre ellos antes de formar parte de la actividad para descubrir la dirección del servidor y conectarse a él –de hecho será lo habitual porque la primera URI conocida suele ser de un contacto–.

- **P2P puro (sin servidor específico, todos los usuarios tienen el mismo rol).** No existe infraestructura específica de la actividad y todos los participantes intercambian señalización y tráfico con todos, manteniendo una relación entre iguales.
- **P2P mixto o en cascada (sin servidor específico, los usuarios desempeñan distintos roles).** No existe infraestructura específica de la actividad, pero hay participantes que ejercen la función lógica de servidor (supernodo) para otros participantes (nodos), permitiendo topologías en cascada.

El mecanismo de descubrimiento es la forma mediante la cual un cliente consigue establecer sesión con un supernodo válido involucrado en la actividad en la que el cliente quiere participar, supuesto que únicamente conoce algún usuario involucrado en la misma, obtenido a partir de una búsqueda o directamente mediante el estado de presencia de sus contactos. Esto es aplicable para topologías P2P en cascada, si bien en el caso de topologías cliente-servidor puede particularizarse y aplicarse el mismo mecanismo para descubrir el servidor al que conectarse. En topologías P2P puras, todos con todos, no tiene sentido este mecanismo pues todos los participantes cumplen el mismo rol.

El mecanismo de descubrimiento se basa en que si un cliente invita a establecer una sesión con un nodo que no vaya a ser su supernodo, el nodo que recibe la invitación responderá con un mensaje SIP REFER hacia su propio supernodo. De este modo, el cliente inicial irá intentando establecer sesión con supernodos subiendo en la jerarquía de la topología de red, hasta encontrar un supernodo válido (estableciéndose finalmente la sesión) o bien llegando al supernodo más alto en la jerarquía, que redirigirá la invitación hacia sus clientes (pues no tiene supernodos superiores), exceptuando aquellos que generaron el REFER hacia él. Así, el cliente empezará a intentar establecer conexiones bajando ahora en la jerarquía hasta encontrar un supernodo válido (ya que no se realizan respuestas REFER hacia nodos de los que ya ha venido una respuesta REFER previa).

Este procedimiento se observa gráficamente en la Fig. 42, en un escenario con topología P2P en cascada donde los supernodos, por limitaciones de capacidad de proceso (por ejemplo, aunque podrían seguirse otros criterios), sólo pueden admitir dos nodos hijo cada uno, formando así árboles binarios. En la figura, el cliente “A” quiere participar en una actividad en la que ya está participando el cliente “B” que, sin

embargo, no es supernodo. Existe un paso previo, no mostrado en la figura, que es el conocimiento por parte de A del identificador de algún participante (B) que esté en la actividad a la que se quiere unir, así como el identificador propio de la actividad. Esta información puede obtenerla del estado de presencia de su lista de contactos o mediante los mecanismos de búsqueda provistos por XDMS.

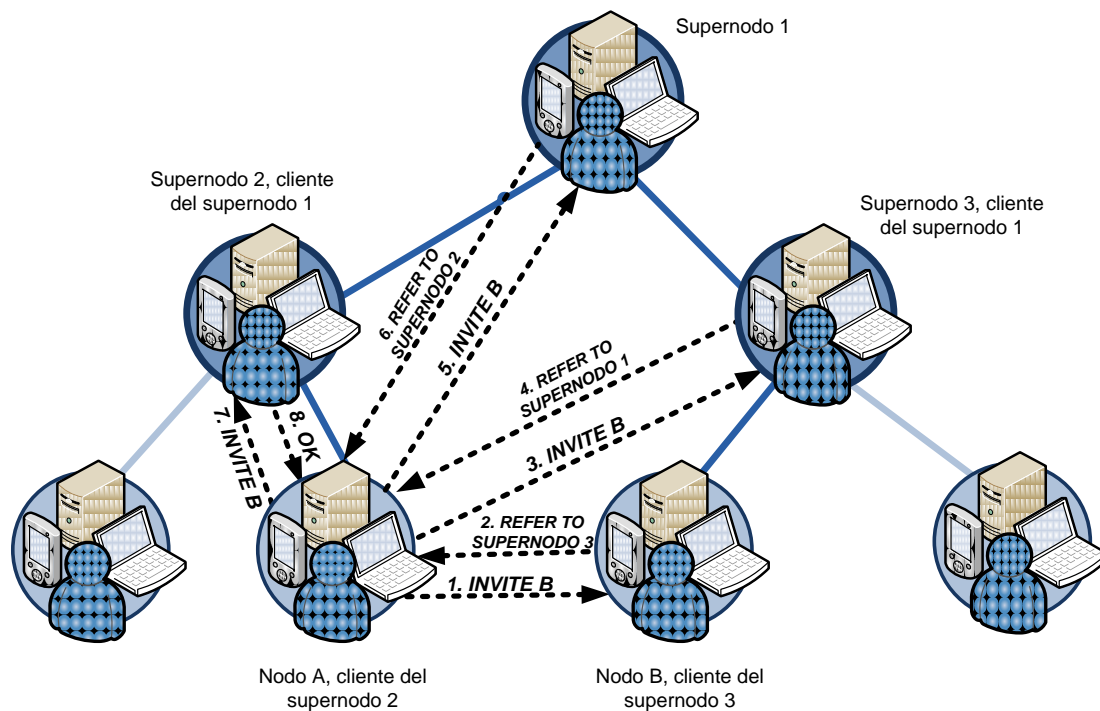


Fig. 42. Ejemplo mecanismo de descubrimiento de supernodos

Todo nodo que tenga el rol de supernodo debe contar con algún criterio para saber si debe o no aceptar la sesión con un nodo cliente. El criterio seguido depende de la aplicación, las capacidades de la red y los terminales, si bien en la Tabla 2 indicamos posibles políticas de asignación nodo-supernodo.

Tabla 2. Criterios de asignación de supernodos

Política	Descripción
Prefijado	Se trata del caso simplificado: los supernodos conocen de antemano los clientes que deben aceptar. Es adecuado en escenarios donde los participantes en la actividad se conocen entre ellos.
Por número máximo de sesiones	El supernodo acepta nodos cliente mientras no se supere un umbral de número máximo de sesiones. En función del tipo de aplicación, pueden considerarse, del mismo modo, el total de sesiones considerando las sesiones soportadas por los nodos hijos que dependen de él.
Por máximo ancho de banda	El supernodo acepta nodos cliente si el ancho de banda requerido para la sesión con éste no supera un umbral máximo establecido para el ancho de banda total que debe cursar el supernodo.
Por localización geográfica	El supernodo admite sesiones de clientes cercanos, hasta un radio límite preestablecido. La localización geográfica puede resolverse mediante coordenadas, que se enviarían como parte de la información de presencia del nodo cliente.
Por tipo de acceso u otros parámetros del cliente	El supernodo admite sesiones de clientes con un determinado tipo de conexión de acceso (xDSL, red celular, etc.) o bien atendiendo a otro tipo de características del cliente (capacidades, ancho de banda disponible, etc.)
Por número de saltos en la red	El supernodo admite sesiones de clientes que se encuentren accesibles mediante un número de saltos por los nodos de la red menor o igual a un límite establecido.
Por dominio	El supernodo admite sesiones de clientes que se encuentren en su mismo dominio IMS, sin perjuicio de ser cliente de un supernodo perteneciente otro dominio.

De este modo, el proceso de toma automática de decisiones ante una nueva petición de inicio de sesión se presenta modelado en la Fig. 43.

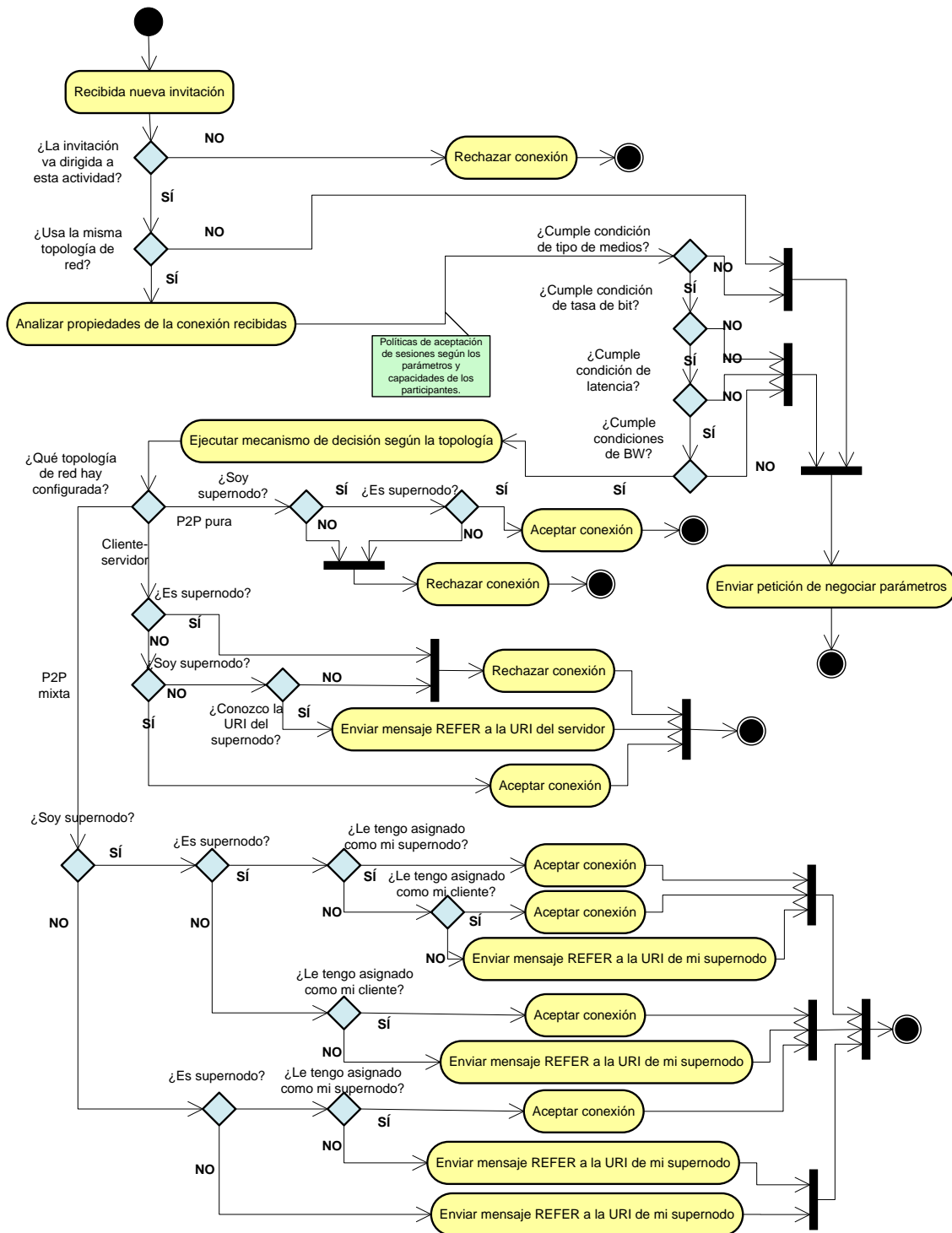


Fig. 43. Toma de decisiones ante una petición de incorporación a una actividad

Como se desprende de la Fig. 43, una invitación de nueva conexión a una actividad debe cumplir en primer lugar las políticas de aceptación según las propiedades de la conexión que se apliquen. A continuación, se aplica un mecanismo de toma de decisiones dependiendo de la topología de red asignada a la aplicación.

Las reglas de este mecanismo de decisión están basadas en los conceptos de la topología más general, considerando que los casos de P2P puro y cliente-servidor son particularizaciones del caso P2P mixto:

- **Topología P2P mixta:**
  - Un nodo sólo acepta sesiones de sus clientes y supernodos asignados.
  - Si no es un cliente o supernodo asignado, la respuesta es una redirección (siguiendo el mecanismo de descubrimiento de supernodos).
  - Cualquier otro caso no es posible según se ha definido la arquitectura de red y por defecto no se acepta.
- **Topología P2P pura:**
  - Todo nodo ha de ser supernodo del resto y, por tanto, cliente del resto. Cualquier otro caso no es posible según se ha definido la arquitectura de red y por defecto no se aceptará.
- **Topología cliente-servidor:**
  - Sólo existe un supernodo, el servidor, el resto de nodos son sus clientes.
  - El servidor nunca envía peticiones de inicio de sesión, sólo las responde.
  - Dado que el único nodo que acepta peticiones de sesión es el servidor, los clientes redireccionarán las invitaciones de nuevos participantes hacia el servidor, siempre que conozcan su dirección, rechazándola en caso contrario.

Nótese que para implementar el mecanismo de descubrimiento de supernodos, usado únicamente en la topología de red P2P mixta, toda petición de invitación debe incluir en la descripción de la sesión un campo que indique si la petición es fruto de un mensaje REFER de otro nodo, y en tal caso, la URI de dicho nodo. Con esto, el nodo que recibe la invitación, si debe responder con un nuevo mensaje REFER, no indicará una redirección hacia ese nodo, evitando bucles en el proceso de establecimiento de la sesión.

Recordemos también que este mensaje REFER siempre direccionara hacia los supernodos asignados, y en caso de no tener (y ser supernodo), hacia los clientes asignados. En un caso en el cual no se pueda hacer referencia hacia nadie, se propone

como posible política adicional de descubrimiento que si el nodo que recibe la invitación puede ejercer papel de supernodo del nodo que le invita, aquel tomará el rol de su supernodo y aceptará la conexión.

### 5.4.3 Búsqueda de actividades

El middleware proporciona métodos en su interfaz de programación (a través de la clase *Engine*) para la búsqueda de participantes potenciales para una actividad. Los criterios de búsqueda especificables son los pertenecientes al modelo de datos de contacto definido en la sección 5.3.2.

La búsqueda de participantes puede implementarse de dos formas:

- Localmente, ya que el middleware se suscribe a la información de presencia de los contactos del usuario y permite conocer en qué actividades están participando y si se puede unir a ellas o no. Está restringido a la lista de contactos del usuario.
- A través de consultas XCAP/Xquery al servidor XDM que almacena los datos de presencia. No está restringido a la lista de contactos del usuario, pero presupone que el servidor XDM de IMS permita la búsqueda dentro de documentos de presencia.

### 5.4.4 Suscripción a eventos

Los eventos relacionados con una actividad se reciben por la aplicación a través de la interfaz de escucha *ActivityListener* y pueden ser de cuatro tipos:

- **Peticiones recibidas**, por ejemplo una invitación a unirse a una actividad
- **Respuestas recibidas**, por ejemplo si un participante ha aceptado o rechazado una invitación
- **Mensajes de estado** recibidos, encapsulan cualquier tipo de información notificable por la aplicación. La semántica de los datos depende de la aplicación.
- **Datos de aplicación** recibidos, por ejemplo los datos un movimiento en un juego interactivo o un trazo de una pizarra virtual. La semántica de los datos depende de la aplicación.

Independientemente de esta interfaz pública, el middleware se suscribe automáticamente a la información de presencia de los participantes para actualizar las

instancias de *Participant* y devolver siempre información actualizada en los métodos de tipo *get*, por ejemplo en *getPicture* para obtener la imagen o avatar del participante.

#### 5.4.5 Finalización de la actividad

Para dejar de participar en una actividad basta con llamar al método *disconnect()* de la instancia de *Activity* correspondiente. El middleware automáticamente actualiza la información de presencia del usuario (publicando el nuevo estado en el XDMS) para reflejar que ha dejado de participar en la actividad. Adicionalmente, cierra las sesiones SIP involucradas.

La actividad como tal se da por concluida cuando no hay ningún usuario participando en la misma, es decir, no hay ninguna sesión establecida de esa actividad.

Nótese la diferencia con el habilitador LCE donde las actividades se reflejan como documentos XML en el XDMS y tienen un estado independientemente de que haya participantes en ese momento o no. El habilitador LCE asume que los clientes pueden no tener software específico de aplicación instalado y por ello es necesario almacenar el estado en el servidor XDM para permitir funciones como la pausa y la reanudación. En el middleware HIMMA el planteamiento es inverso, se asume que en el lado de servidor hay un soporte mínimo, concretamente el habilitador de presencia que cualquier red IMS incluye entre sus servicios básicos.

Las funciones como pausa y reanudación, si bien no están soportadas directamente por el middleware, se facilitan en gran medida. Basta con que las instancias de la aplicación de los participantes almacenen la lista de participantes y su información de estado. Cualquier instancia de la aplicación podría, consultando el estado de presencia de los participantes, reconstruir la actividad y volver a generar el árbol de sesiones en función de las condiciones de ese momento.

### 5.5 Implementación y pruebas

El API del middleware HIMMA está implementado en Java haciendo uso de la plataforma ICP (*IMS Client Platform*) disponible en el entorno Ericsson SDS así como del resto de nodos de IMS emulados por SDS (CSCF, HSS, servidor de presencia y gestor de grupos). La plataforma ICP implementa un borrador de la especificación JSR 281 de servicios IMS. Este entorno incluye un emulador de terminales móviles compatibles (actualmente Sony Ericsson P990i y M600) así como el software necesario para dotar de estas APIs a un sistema basado en Windows.

La siguiente figura describe las relaciones entre la implementación actual del middleware y la biblioteca ICP, así como las relaciones de ésta con los distintos nodos de IMS. Por claridad, no se han incluido las interfaces de escucha de ICP, pues prácticamente todas las clases tienen un *listener* asociado y el middleware implementa todos para recibir los eventos de la red IMS (eventos de presencia, invitaciones, etc.) y de la propia plataforma ICP.

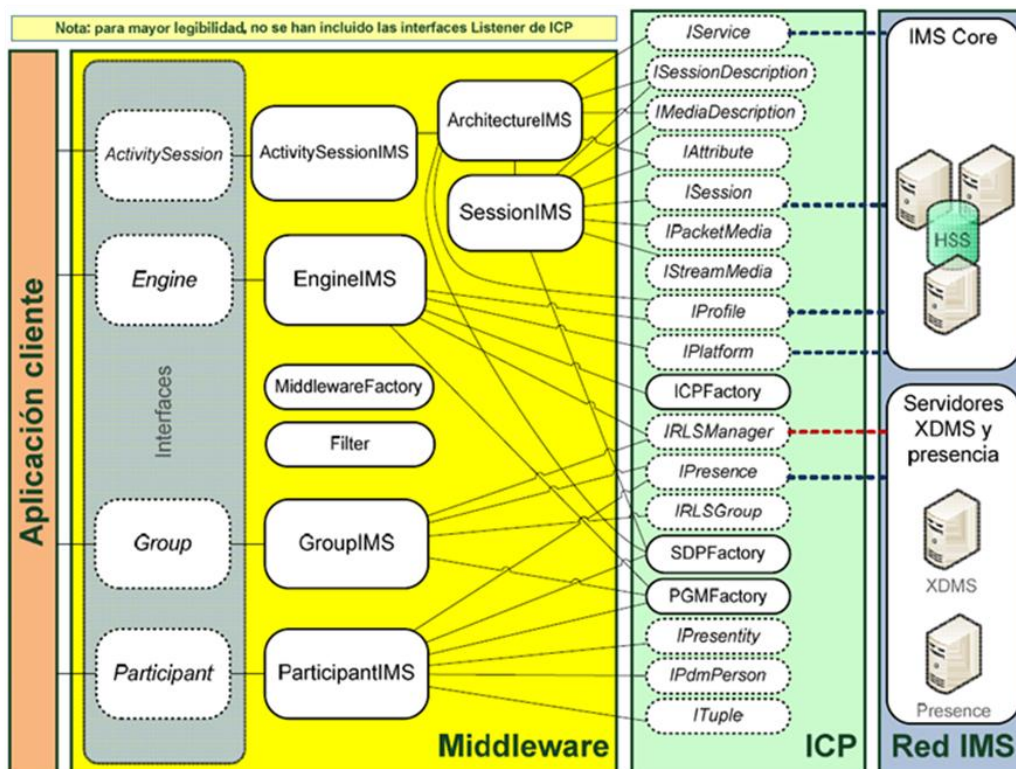


Fig. 44. Relación de las clases del middleware con la biblioteca ICP

La validación ha consistido en el desarrollo de un conjunto de pequeños casos de prueba orientados a las funcionalidades principales del middleware: gestión de presencia enriquecida y establecimiento de actividades distribuidas. El escenario completo consta de cuatro máquinas con Windows y entorno SDS instalado: una de ellas actúa como núcleo IMS con gestor de presencia y grupos, las otras tres actúan como usuarios y tienen instalado ICP, ejecutando las aplicaciones de prueba (que hacen uso del middleware, naturalmente). Los usuarios son sip:alice@lab, sip:bob@lab y sip:carol@lab, todos ellos registrados en el dominio lab definido en la máquina con SDS.

El esqueleto de los casos de prueba que hacen uso del middleware es similar al mostrado en el Listado 4 y sirve de ejemplo para mostrar el alto nivel de abstracción del middleware.



```

import es.upm.dit.ims.himma.middleware.api.*;
public MyApp implements ActivitySessionListener{
    private ActivitySession myActivity;
    public static void main(){
        Engine myEngine = MiddlewareFactory.createEngine();

        Participant myself = MiddlewareFactory.createMyself();
        myself.setAge("26");
        myself.setCountry("Spain");
        myself.setFirstName("Alice");
        myself.setGender("female");
        myself.setMood("Happy");
        myself.addLanguage("es", 0.6);
        ...
        Group myContacts = engine.getContacts();
        Participant[] myFriends = myContacts.getParticipants();
        ActivitySession myActivity =
            MiddlewareFactory.createActivitySession("myActivity@lab");
        myActivity.addActivitySessionListener(this);
        myActivity.invite(myFriends[0]);
        ...
        myActivity.sendData(data);
        ...
        myActivity.disconnect();
    }
    //Métodos implementados de ActivitySessionListener
    public void receivedRequest(Participant sender, String request){
        ...
        myActivity.acceptInvitation(sender);
    }
    public void receivedResponse (Participant sender, String response){...}
    public void receivedMessage(Participant sender, String message){...}
    public void receivedData(Participant sender, byte[] data){...}
    public void receivedStatusMessage(Participant sender, String statusMessage){...}
}

```

**Listado 4. Estructura de ejemplo de una aplicación que utilice el middleware HIMMA**

### 5.5.1 Prueba de registro en IMS

En un entorno de ejecución de aplicaciones IMS el usuario se autentica mediante el proceso de registro en la red, no siendo necesaria autenticación adicional a nivel de aplicación (se prescinde del típico proceso de registro con usuario y contraseña típico de las aplicaciones de Internet). En el caso del middleware, basado en la plataforma ICP, es la propia plataforma la que se registra automáticamente al encender el terminal emulado o, en la versión de Windows, al iniciar el sistema operativo.

### 5.5.2 Prueba de presencia

La prueba de presencia valida la formación correcta del documento de presencia enriquecida en XML para un participante, la publicación de esta información en el XDMS, así como la suscripción al estado de otros participantes de la actividad y la recepción de sus notificaciones asociadas.

La aplicación implementada, ejecutada en las máquinas de Alice y Bob, inicializa el middleware y crea una actividad ficticia denominada “actividad1@lab”, en la que participan Alice y Bob, instanciados como objetos *Participant* del middleware. Cada instancia del middleware se suscribe automáticamente a la presencia de los demás participantes (Alice en el caso de Bob, y Bob en el caso de Alice), de modo que la información consultada a través de la interfaz *Participant* esté actualizada.

Cada instancia de la aplicación modifica periódicamente diversos parámetros del perfil propio (edad, status, humor, idiomas, etc.), que son publicados en el servidor de presencia (XDMS) observándose los mensajes PUBLISH enviados.

Al mismo tiempo, cada instancia recibe a través de las interfaces de recepción de eventos (*listener*) las actualizaciones de la información de presencia de los otros participantes, mostrándolo en pantalla instantáneamente.

### 5.5.3 Prueba de establecimiento de sesiones de actividad distribuida

El caso práctico para probar el establecimiento de sesiones de actividad de forma distribuida se basa en una aplicación de mensajería instantánea a tres con la particularidad de que el tráfico de mensajes sigue una jerarquía: uno de los usuarios actúa como supernodo y los mensajes de todos los usuarios se intercambian pasando por él. Conceptualmente es equivalente a un juego en red de tres jugadores donde uno de ellos actúa de servidor, las sesiones establecidas representan las relaciones entre las aplicaciones de los jugadores (nodo, supernodo) y los mensajes los eventos del juego.

Las sesiones en este caso de prueba son sesiones MSRP de mensajes de texto. IMS es agnóstico en cuanto a tipos de medios, si bien la implementación de ICP utilizada para el middleware sólo admite RTP (para audio, aunque carece de mezclador) y MSRP, por lo que la implementación actual del middleware es válida para aplicaciones que puedan modelar los eventos de aplicación textualmente (en XML, en texto plano o en cualquier tipo MIME que se desee).

La Fig. 45 muestra los pasos de esta prueba, desde la situación de partida en que Alice y Bob comienzan a participar en una actividad, hasta que un tercero (Carol) se

une y se utiliza el mecanismo de transferencia de sesión para establecer correctamente las sesiones entre los participantes teniendo en cuenta que Bob actúa como supernodo<sup>6</sup>.

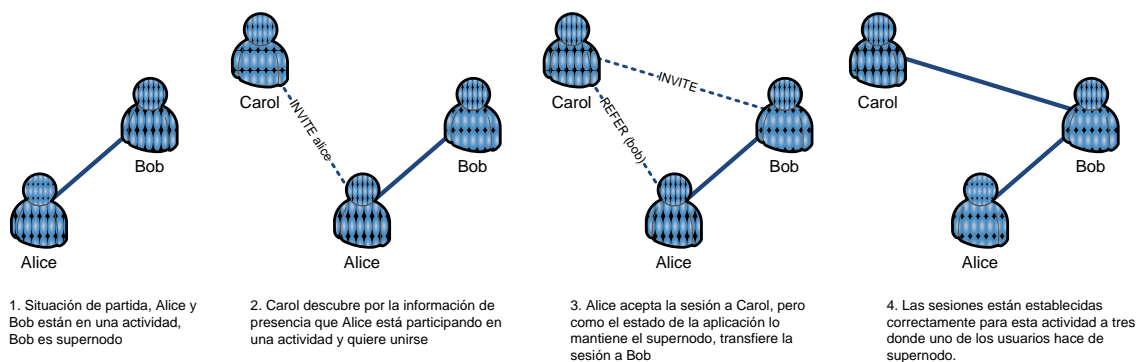


Fig. 45. Establecimiento de actividad de tres usuarios con supernodo con middleware HIMMA

En [61] pueden encontrarse detalles adicionales sobre la implementación, validación, limitaciones del entorno de pruebas y otros aspectos del middleware HIMMA.

## 5.6 Conclusiones

En este capítulo se ha descrito el middleware HIMMA (*HIMMA-MW*), un middleware de soporte a aplicaciones multimedia multiusuario de alto grado de interactividad que facilita el desarrollo de aplicaciones descentralizadas en arquitecturas IMS, reutilizando los elementos funcionales de IMS de manera novedosa en comparación a los middlewares IMS existentes. Concretamente, el middleware hace uso intensivo del mecanismo de establecimiento de sesiones y las funciones de gestión de presencia y grupos, alejándose del modelo de conferencias habitual que delega en servidores de forma centralizada, como el propio modelo de conferencias de IMS.

El middleware proporciona una interfaz Java de alto nivel donde los conceptos principales son Actividades, Grupos y Participantes. Los participantes pueden asociarse en grupos y forman parte de las actividades. Cuando un participante se incorpora a una actividad, el middleware establece las sesiones SIP subyacentes oportunas en función de la topología de la actividad (centralizada, P2P en malla o P2P

<sup>6</sup> En una aplicación de mensajería instantánea o chat puede no tener mucho sentido la incorporación de un supernodo, pudiendo enviarse los mensajes de texto directamente a los destinatarios. Sin embargo, en aplicaciones que requieran mantener un estado y actualizarlo conforme a los eventos recibidos es habitual mantener una estructura jerárquica, de uno o varios niveles en función del número de usuarios, mediante un servidor en soluciones centralizadas o mediante supernodos en soluciones P2P como la propuesta. Estos supernodos actualizan el estado de la aplicación conforme a los eventos recibidos de los demás y envían actualizaciones de estado a los participantes involucrados.

jerárquico), liberando al desarrollador de la gestión de sesiones SIP y detalles del protocolo. De igual forma, el middleware gestiona la información de presencia automáticamente para publicar el estado de participación en las distintas actividades y permitir la búsqueda de actividades a través de los contactos del usuario o de las funciones de búsqueda que proporcione la red IMS a través del nodo XDMS.

Se ha desarrollado una implementación de referencia del middleware basada en ICP (IMS Client Platform), un entorno de ejecución para aplicaciones IMS que se basa en la JSR 281, si bien la interfaz del middleware no tiene ninguna dependencia con ICP, por lo que puede portarse a otras plataformas.

La validación práctica se ha llevado a cabo a través de casos de prueba relevantes orientados a una actividad de intercambio de mensajes de texto (extrapolable a cualquier tipo de información requerida por la aplicación), comprobando que se establecen correctamente las topologías y se actualizan los documentos de presencia en la red.

## 6. Aplicación a distintos entornos

En este capítulo se presenta la aplicación del marco genérico de aplicaciones multimedia multiusuario de alta interactividad (HIMMA) propuesto en la Tesis a dos escenarios relevantes: servicios para juegos multijugador en red y servicios para aplicaciones profesionales. En cada escenario se muestran un conjunto de casos de uso que sirven de validación crítica de las contribuciones presentadas hasta el momento en los capítulos 3 (arquitectura global de soporte a aplicaciones HIMMA), 4 (habilitador LCE de gestión de actividades multiusuario) y 5 (middleware HIMMA).

Adicionalmente, se presenta un apartado de consideraciones sobre escalabilidad donde se discute el impacto que tiene la aplicación de este marco de soporte a aplicaciones HIMMA en un entorno IMS.

### 6.1 Servicios de juego en red multijugador

La vocación de IMS como red de operador Todo IP convergente para la prestación de servicios multimedia, en redes fijas y móviles, la hace a priori una alternativa apropiada para plantear un escenario de provisión de juegos multijugador en IMS que pueda ofrecer, al menos, las características de las plataformas de entretenimiento existentes actualmente y, además, añadir características innovadoras facilitadas por su arquitectura.

Como señalamos en la sección 2.1.1, los juegos multijugador constituyen un caso de estudio reseñable de aplicaciones multimedia multiusuario de alta interactividad (HIMMA), lo que unido al volumen de negocio y número de usuarios que utilizan sistemas de entretenimiento o mini-juegos web podrían convertirse en la *killer-application* de IMS, siempre y cuando se aprovechen las ventajas de esta arquitectura para estimular la interacción de los usuarios.

En las plataformas de juego actuales encontramos tanto plataformas accesibles vía Internet como plataformas desplegadas en el entorno de un operador móvil, pero en todos los casos presentan una infraestructura específica para ofrecer todos sus servicios, aun cuando gran parte pueden considerarse duplicados, especialmente los relativos a gestión de usuarios, comunicación entre usuarios y facturación, entre otros.

La situación actual del mercado con distintas plataformas de juegos con servicios básicos duplicados que se representaba en la Fig. 1 (y reproducida por comodidad en la Fig. 46), se debe principalmente a que las redes donde se despliegan no ofrecen más que un servicio de transporte, que en el caso de Internet es del tipo *best-effort*, y cada plataforma implementa sus propios mecanismos para la provisión de cada servicio, produciendo una desagregación de la base de usuarios, dificultando la interoperabilidad y replicando esfuerzo por parte de los desarrolladores si quieren alcanzar un mayor público objetivo.

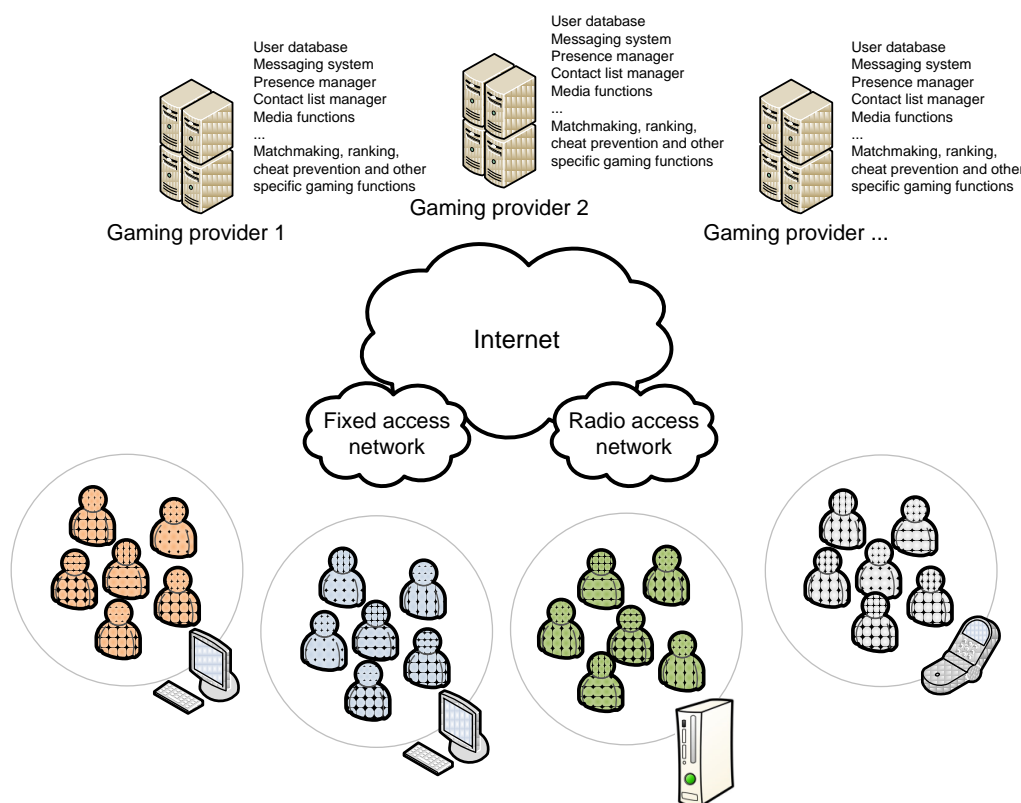
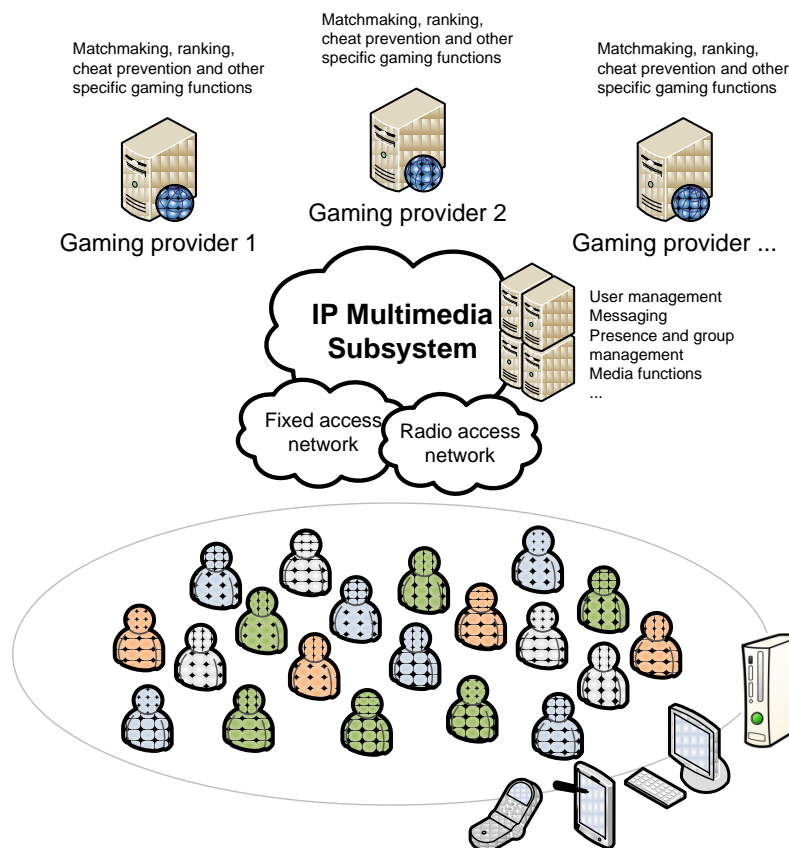


Fig. 46. Situación actual de desagregación y duplicidad funcional en servicios de juegos en red

A priori, una arquitectura multimedia como IMS puede suplir estas limitaciones y permitir que los desarrolladores y proveedores de juegos en red centren sus esfuerzos en añadir funciones de valor añadido al tiempo que consiguen un mayor número de usuarios potenciales, que dejan de estar desagregados, y un menor coste de mantenimiento de infraestructura del lado de servidor, al configurarse un escenario como el propuesto en la Fig. 47, que es similar al escenario objetivo mostrado

previamente en la Fig. 46, pero particularizado para los servicios de juegos en red que estamos considerando en este apartado.



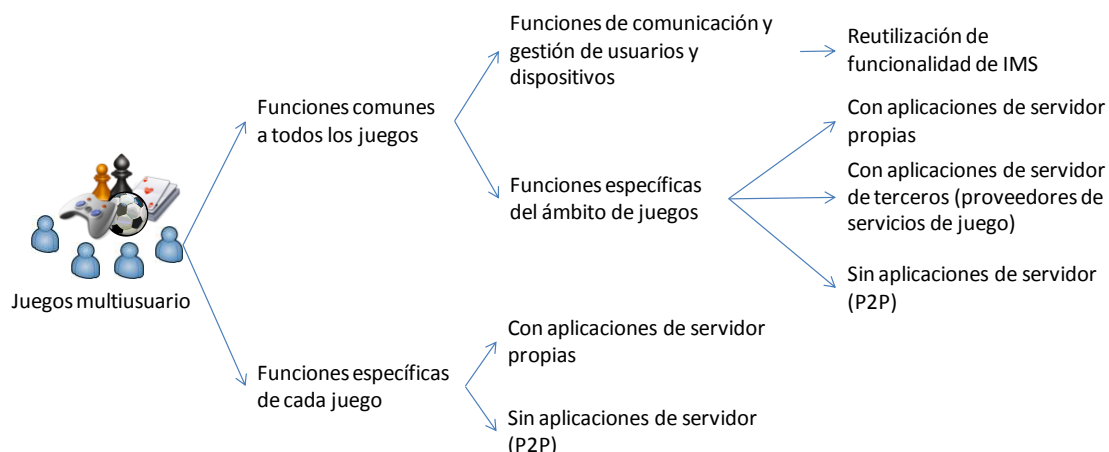
**Fig. 47. Solución convergente para servicios de juegos en red basada en IMS**

En el escenario propuesto de provisión de juegos multijugador en redes IMS se trasladan las funciones básicas de comunicación multimedia entre usuarios y gestión de dispositivos a los nodos de la arquitectura IMS, liberando de estas tareas a los servidores del proveedor de juegos, que pasan a dedicarse exclusivamente al ofrecimiento de funciones específicas de juegos como la generación de rankings, prevención de trampas, aplicación de algoritmos de emparejamiento, etc.

Una de las ventajas de este escenario, desde el punto de vista del desarrollador de juegos, es que no es obligatorio disponer de medios para ofrecer funciones multijugador en sus juegos, ya que IMS proporciona las funciones imprescindibles como son la gestión de listas de amigos, información de presencia, mensajería y establecimiento de sesiones multimedia, por lo que el juego, entendido como una aplicación instalable en el dispositivo del usuario, puede ser autosuficiente. En un entorno de red IP no IMS, la adición de funciones multiusuario requiere soporte de servidores de aplicación obligatoriamente, bien proporcionado por el propio desarrollador o por terceros, pues los usuarios únicamente están identificados por una

dirección IP que, habitualmente, es dinámica o ni siquiera es suya (caso de servicios de traducción de direcciones - NAT - intermedios).

La Fig. 48 pone de manifiesto las distintas opciones disponibles para el desarrollador a la hora de crear un juego multiusuario, de acuerdo a la funcionalidad requerida y la disponibilidad de infraestructura específica.



**Fig. 48. Caracterización de las funciones de juego multiusuario y alternativas de despliegue en IMS**

Como hemos indicado anteriormente, el contar con una arquitectura de red que proporciona servicios de comunicación IP de forma estándar constituye un gran aliciente para la incorporación de funciones multiusuario en las aplicaciones. Por un lado, permite la entrada en el mercado de pequeñas compañías de desarrollo e incluso desarrolladores *freelance*, que no requieren una inversión inicial en infraestructura de servidor para gestionar el registro y comunicaciones entre los usuarios y evitan el efecto de “muerte por éxito” si la infraestructura no puede escalar convenientemente. Por otra parte, los desarrolladores con mayores recursos pueden enriquecer la experiencia de usuario centrándose en los aspectos más específicos de los juegos ofrecidos mediante la incorporación de servicios propios o proporcionados por terceros (proveedores de servicio de juegos).

Nótese que el ofrecimiento de aplicaciones sin soporte de servidores específicos es lo que se denomina P2P (*peer to peer*), pero en la práctica es necesario que haya un nodo de arranque o *bootstrap* conocido al que los usuarios o *peers* se conectan por primera vez para obtener listas de otros *peers* con los que comunicarse. En IMS veremos que puede prescindirse de este nodo de arranque gracias a los mecanismos de gestión de



usuarios, pudiendo ofrecer aplicaciones completamente descentralizadas desde el punto de vista del desarrollador<sup>7</sup>.

En las siguientes secciones analizaremos la funcionalidad encontrada en plataformas de juego en red comerciales e identificaremos cuáles son las funciones que pueden ofrecerse reutilizando la funcionalidad ofrecida por IMS y las extensiones presentadas en la Tesis, definiendo el abanico de funciones mínimo disponible para el desarrollador que no cuente con infraestructura específica de soporte para juegos.

### 6.1.1 Funcionalidad de las plataformas de videojuegos comerciales

Para poder evaluar de forma crítica la aplicación de las contribuciones presentadas a un escenario de juego multijugador en red es necesario un paso previo de análisis de requisitos de este tipo de aplicaciones. Fruto de este análisis original se obtendrá un conjunto de funciones característico y actualizado de los sistemas de juego en red multijugador actuales y topologías de red empleadas habitualmente, que permitirán la evaluación de IMS como arquitectura base para soportar aplicaciones de juegos en red y las ventajas de las funciones complementarias proporcionadas por LCE y el middleware HIMMA.

Las plataformas de juego pueden estudiarse desde distintos ámbitos, incluyendo características audiovisuales (2D, 3D, alta definición, etc.) o interfaces de control (teclados, pads, giroscopios y acelerómetros, etc.). En este trabajo de Tesis prestaremos atención a las características relacionadas con el uso de la red de comunicaciones (Internet generalmente). Con el fin de identificar y clasificar estas características, hemos analizado nueve casos: el juego WoW y las plataformas Xbox Live, Sony PSN, Nintendo WFC, Nokia SNAP/N-Gage, Helio 3G, Andago Games, Exit Games Neutron 5.0 y Steam. Nokia SNAP/N-Gage y Helio 3G son plataformas de juego multijugador para operadores de red móvil. Andago Games y Exit Games Neutron 5.0 son plataformas genéricas que proporcionan soporte a juegos para PC, móviles y juegos web. Finalmente, Steam es una plataforma especializada en distribución y descarga de juegos.

El resultado del análisis es un total de 35 características relacionadas con la red, encontradas a lo largo de la mayor parte de los sistemas de juegos relevantes en PC, móviles y consolas de videojuegos. Las características están clasificadas en tres grupos, descritos a continuación:

---

<sup>7</sup> Evidentemente, desde el punto de vista de conectividad no son descentralizadas porque requieren los servidores de IMS para funcionar, pero para el desarrollador son transparentes.

- *Características generales.* Incluye las características básicas de la plataforma, como el tipo de dispositivos y redes soportadas, y funciones generales relacionadas con pagos, seguridad, actualizaciones de software, etc.
- *Características de comunicación directa.* Incluye funciones donde los usuarios se comunican directamente. Por directamente se entiende a comunicaciones limitadas en el tiempo, donde los participantes han sido seleccionados explícitamente para formar parte de la comunicación. Por ejemplo, establecer una videoconferencia a tres se considera comunicación directa. Recibir el estado de presencia de un amigo, si bien es una comunicación, se incluye dentro de la categoría de servicios de comunidad.
- *Características de comunidad o sociales.* Incluye funciones destinadas a potenciar la participación de los usuarios en distintas actividades, bien de forma activa (p.ej. organizando torneos entre usuarios) o de forma pasiva (p.ej. mostrando una lista de amigos disponibles para jugar).

Cabe destacar que no existe ninguna plataforma de juegos que implemente todas las características identificadas, pero para cada característica identificada existe al menos una plataforma o juego que lo implementa.

Tabla 3. Síntesis de características generales de plataformas de juego en red

Características	Descripción
<b>Dispositivos</b>	Ofrecimiento de servicios a uno o más tipos de dispositivos, p.ej., consolas de videojuegos, teléfonos móviles o PCs.
<b>Red</b>	Red utilizada, típicamente Internet o red de operador.
<b>Soporte interoperador</b>	Capacidad para que usuarios de distintos operadores jueguen entre ellos (sólo aplica a redes de operador).
<b>Actualizaciones automáticas</b>	Instalación automática de actualizaciones software que mejoren la funcionalidad o solucionen problemas.
<b>Contenido descargable</b>	Soporte para descargar nuevo contenido que mejore los juegos, p.ej. nuevos circuitos para un juego de carreras de coches.
<b>Información de juego</b>	Proporciona información sobre los juegos disponibles (descripción, tipo, capturas de pantalla, opciones de compra, etc.)
<b>Control paternal</b>	Proporciona mecanismos para forzar determinadas condiciones en el uso de los juegos (p.ej. prohibir hablar con desconocidos o establecer un tiempo máximo de juego diario). Está orientado a proteger a los niños.
<b>Detección de trampas</b>	Conjunto de mecanismos destinados a garantizar que los jugadores respetan las reglas de cada juego y no usan software modificado o alteran el tráfico de la red para conseguir una ventaja competitiva.
<b>Opciones de pago</b>	Tipos de pago permitidos para acceder a los servicios ofrecidos (p.ej. suscripción, tarjetas de prepago o pago por uso)
<b>Compra/alquiler de juegos</b>	Posibilidad de adquirir licencias para jugar durante un tiempo limitado o ilimitado.
<b>Regalo de juegos</b>	Permite a un usuario disfrutar de un juego porque otro se lo ha regalado.
<b>Disponibilidad</b>	Capacidad de mantener el servicio sin una reducción significativa de la Calidad de Experiencia (QoE) en situaciones desfavorables como el aumento puntual del número de usuarios o fallos en algún componente.
<b>Soporte de juegos multijugador masivos</b>	Soporte de juegos que simulan mundos virtuales con cientos de jugadores.

La Tabla 3 describe las características generales identificadas durante el análisis de las plataformas de juego actuales más destacables. Llama la atención que las plataformas de juego en red están ligados habitualmente a un dispositivo específico, y tanto los servicios en red como los dispositivos están desarrollados por el mismo fabricante como un sistema cerrado. Este es el caso de las principales consolas de videojuegos.

Aunque existen algunas plataformas que proporcionan soporte para más de un dispositivo, por ejemplo Nintendo WFC soporta tanto la consola Wii como las portátiles DS, los jugadores de ambas máquinas no pueden jugar entre ellos. Hay algunas excepciones, como el juego *Halo 2* para PC que permite jugar contra los usuarios de *Halo 2* para Xbox 360 a través de la plataforma *Xbox Live*.

**Tabla 4. Síntesis de características de comunicación directa**

Características	Descripción
<b>Audioconferencia</b>	Establecimiento de llamadas de voz con múltiples usuarios, donde dos o más usuarios pueden hablar y escucharse simultáneamente.
<b>Videoconferencia</b>	Establecimiento de conferencias de vídeo y audio entre múltiples usuarios, donde dos o más usuarios pueden verse (y oírse) unos a los otros.
<b>Mensajería instantánea</b>	Charla basada en texto (chat)
<b>Mensajería multimedia</b>	Intercambio de mensajes con contenido multimedia como texto, audio, vídeo o imágenes.
<b>Compartición de fotos</b>	Compartición de imágenes entre varios usuarios.

Conforme han evolucionado las capacidades multimedia y de conectividad de los dispositivos de juego y ordenadores personales se han incorporado nuevas funciones de comunicación en los juegos, conforme muestra la Tabla 4.

Los servicios de comunicación pueden utilizarse de forma paralela a los juegos para mejorar la interactividad. Por ejemplo, algunas plataformas de juego con soporte de audioconferencia como *Xbox Live* permiten que los jugadores hablen dentro y fuera de un juego concreto. Esta característica permite crear la sensación de continuidad a lo largo de diversas actividades de la plataforma. En otras plataformas, como Nintendo

WFC, cuando los usuarios salen de un juego dejan de hablar y escucharse. En juegos para móviles o juegos web no suele incorporarse la funcionalidad de audio o vídeo conferencia por limitaciones del entorno de ejecución o de los propios terminales, por lo que la comunicación entre usuario suele limitarse a la mensajería instantánea (chat textual).

La videoconferencia en los juegos está presente únicamente en algunos juegos para las consolas PS3 y Xbox 360, casi todos juegos de mesa (póker, ajedrez, etc.) que requieren pocos recursos de red como indicamos en [62].

Los servicios de compartición de fotos se incluyen en distintas plataformas, con distintos nombres comerciales como *Photo Party* en *Xbox Live* o *Photo Channel* en Nintendo WFC. En todo caso constituyen aplicaciones independientes que no pueden usarse en paralelo a los juegos.

**Tabla 5. Síntesis de características de comunidad o sociales**

Características	Descripción
<b>Comunidades de juego</b>	Puntos de encuentro virtuales donde los usuarios de un mismo juego pueden hablar y exponer ideas.
<b>Torneos</b>	Gestión de campeonatos donde diferentes usuarios juegan entre ellos de forma ordenada.
<b>Clasificaciones y logros/premios</b>	Mantenimiento de listas de usuarios y sus correspondientes puntuaciones/hitos en cada juego.
<b>Reputación o karma</b>	Permite a los usuarios valorar el comportamiento de otros con el fin de minimizar el impacto de jugadores poco deportivos o respetuosos.
<b>Clanes (grupos)</b>	Permite crear grupos de usuarios estables que pueden jugar con/contra otros grupos de usuarios.
<b>Gestión de salas de juego</b>	Creación de puntos de encuentro virtuales para usuarios que quieren jugar a un juego.
<b>Presencia</b>	Información acerca del estado de otros usuarios (conectado, ausente, no disponible, jugando a cierto juego, etc.)
<b>Búsqueda de usuarios</b>	Permite encontrar usuarios con los que jugar basado en distintos criterios (juego, proximidad, idioma, edad, etc.)

<b>Partidas igualadas (matchmaking)</b>	Algoritmo que calcula el nivel de habilidad de un jugador en función de sus partidas pasadas a cierto juego y resultados de las mismas. Permite establecer partidas entre usuarios con un nivel de habilidad similar.
<b>Expulsión de usuarios</b>	Permite a los usuarios en una partida expulsar a un determinado usuario cuyo comportamiento no sea apropiado. Puede estar restringido a jugadores con alta reputación o con permisos de moderación.
<b>Prohibiciones temporales o permanentes</b>	Mecanismo global o por juego que evita que cierto usuario interactúe con los demás, p.ej. por haber hecho trampas en una partida.
<b>Perfil de usuario</b>	Información acerca de las preferencias de un usuario, puntuaciones, etc. Puede ser global o por juego.
<b>Listas de amigos</b>	Lista de amigos con los que comunicarse rápidamente para iniciar una partida u otras actividades.
<b>Modo fantasma</b>	Modo de juego donde un usuario juega contra las acciones pregrabadas de otro jugador en el mismo escenario. Es común en juegos de coches donde la mejor carrera de un usuario en un circuito dado aparece como un coche translúcido (el fantasma) al que hay que ganar.
<b>Modo espectador</b>	Permite a un usuario ver (no jugar) una partida en curso de otros usuarios.
<b>Persistencia</b>	En mundos virtuales, permite que el avatar, o representación en el mundo virtual del jugador, siga vivo incluso si el usuario deja de jugar. El usuario puede volver a controlar el personaje más adelante.

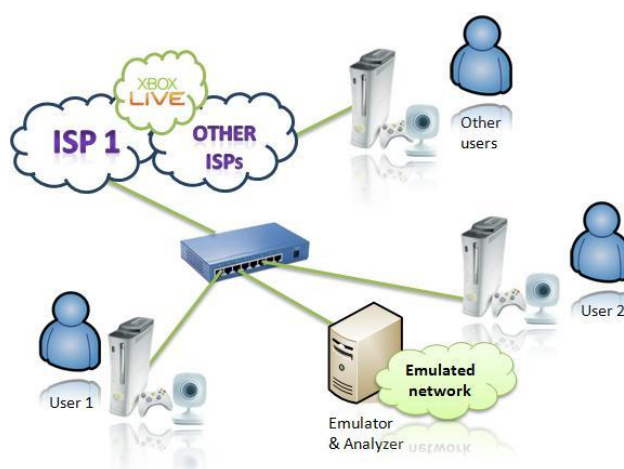
Jugar con o contra otros usuarios y hablar con ellos es sólo una de las vías de interacción que proporcionan las plataformas actuales. Alrededor de estos servicios de comunicación directa encontramos un conjunto de servicios destinados a potenciar la creación de comunidades de usuarios y fomentar la participación y, en cierto modo, la fidelización, de los mismos. La Tabla 5 muestra las características de comunidad identificadas durante el análisis de las plataformas de juego actuales.

El análisis de plataformas de soporte a juegos en red presentado anteriormente se basa fundamentalmente en las características y funciones proporcionadas al usuario

final, bien indicadas por los fabricantes o extraídas de la experiencia con las mismas. Sin embargo, los detalles internos de funcionamiento, como la gestión de las partidas multijugador en red, no están disponibles en la información que proporcionan los fabricantes y tampoco existen artículos recientes u otras fuentes de información que la ofrezcan, exceptuando el trabajo de Zander [63], donde analiza el juego Halo 2 para la consola Xbox (predecesora de la actual Xbox 360) o de Armitage [64], que señala las arquitecturas cliente-servidor de Quake y otros juegos similares.

Con el fin de conocer mejor los requisitos de red de los juegos actuales y el modelo que siguen a la hora de ofrecer partidas multijugador, hemos analizado el tráfico de un conjunto relevante de juegos para la plataforma Microsoft Xbox 360, por contar con un mayor número de usuarios online y un catálogo de juegos más extenso que las competidoras. Las conclusiones obtenidas se han presentado parcialmente en [62].

El escenario de análisis, mostrado en la Fig. 49, consta de un switch con función de replicación de paquetes (*port-mirroring*) al que están conectadas dos consolas Xbox 360 con cámara y auriculares y una máquina con el software *netem* de emulación de red y *Wireshark* como analizador de protocolos. Además, el escenario está conectado a Internet con un servicio de banda ancha que permite a los usuarios locales establecer partidas multijugador con usuarios externos a través del servicio Xbox Live.



**Fig. 49. Escenario de análisis de un sistema de entretenimiento multijugador actual**

El conjunto de juegos seleccionado, que abarca los principales géneros multijugador a excepción de los denominados MMORPG (juegos de rol multijugador masivos), incluye:

- *Gears of War* (GoW), juego de acción en tercera persona
- *Halo 3*, juego de acción en primera persona
- *Call of Duty 4* (CoD), juego de acción en primera persona

- *Project Gotham Racing 3 y 4* (PGR3, PGR4), juegos de carreras automovilísticas
- *Pinball FX* (PFX), juego de pinball
- *Virtua Tennis*, juego de tenis

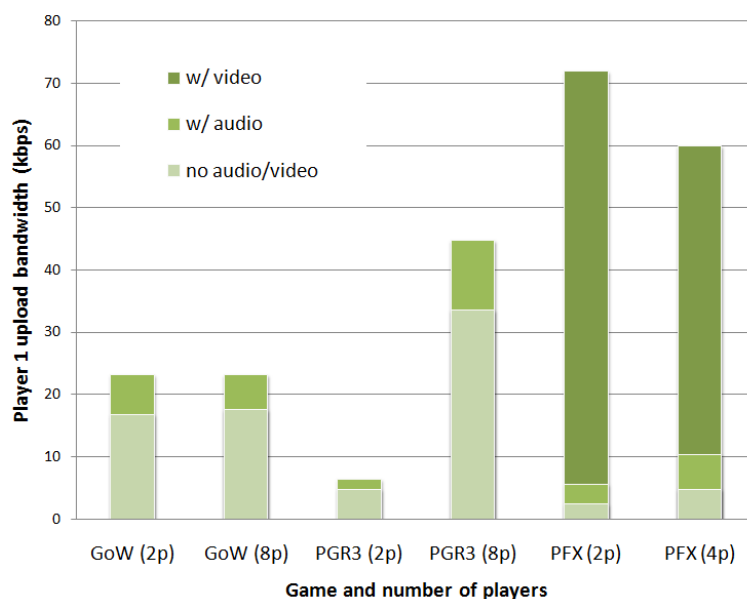
Todos los juegos (de la lista y de la plataforma en general) permiten a los usuarios hablar entre ellos durante la partida, sin embargo no todos los juegos permiten establecer videoconferencia de forma simultánea. En concreto, del catálogo de juegos de Xbox 360 sólo lo permiten algunos juegos de mesa como *UNO*, *Backgammon* y el citado *Pinball FX*, precisamente todos ellos de interactividad reducida por ser por turnos o sin interacción directa entre los participantes.

El tráfico de paquetes intercambiado por las consolas está cifrado, por lo que la única información disponible a través de la captura de tráfico es con qué direcciones intercambia paquetes cada consola, el tamaño de los paquetes y la tasa de envío de los mismos, información suficiente de todas maneras para identificar la topología empleada en los distintos juegos multijugador.

La activación de audio o vídeo entre los jugadores, en los juegos que lo soportan, se produce incrementando el tamaño de los paquetes, por lo que no se puede atender a criterios de diferenciación de flujos pues la información del juego, el audio y el vídeo parecen compartir la carga de cada paquete. Esto implica igualmente que los servicios de audio y vídeo siguen la misma topología de red que el tráfico específico del juego.

En la Fig. 50 presentamos el tráfico medio de subida, medido en una de las consolas, generado por tres de los juegos estudiados (GoW, PGR3 y PFX) y que resultan especialmente representativos porque los demás siguen sus mismas pautas. La figura indica en el eje de abscisas los distintos juegos analizados y el número de jugadores de las partidas estudiadas, y en el eje de ordenadas el ancho de banda de subida en kilobits si estaba activada la videoconferencia (en caso de soportarse), sólo la audioconferencia (todos lo soportan) o ninguna de las dos (es decir, sólo el tráfico específico del juego). De esta forma, pese a que todos los medios comparten flujo de datos, es posible estimar el peso de cada componente dentro del flujo.





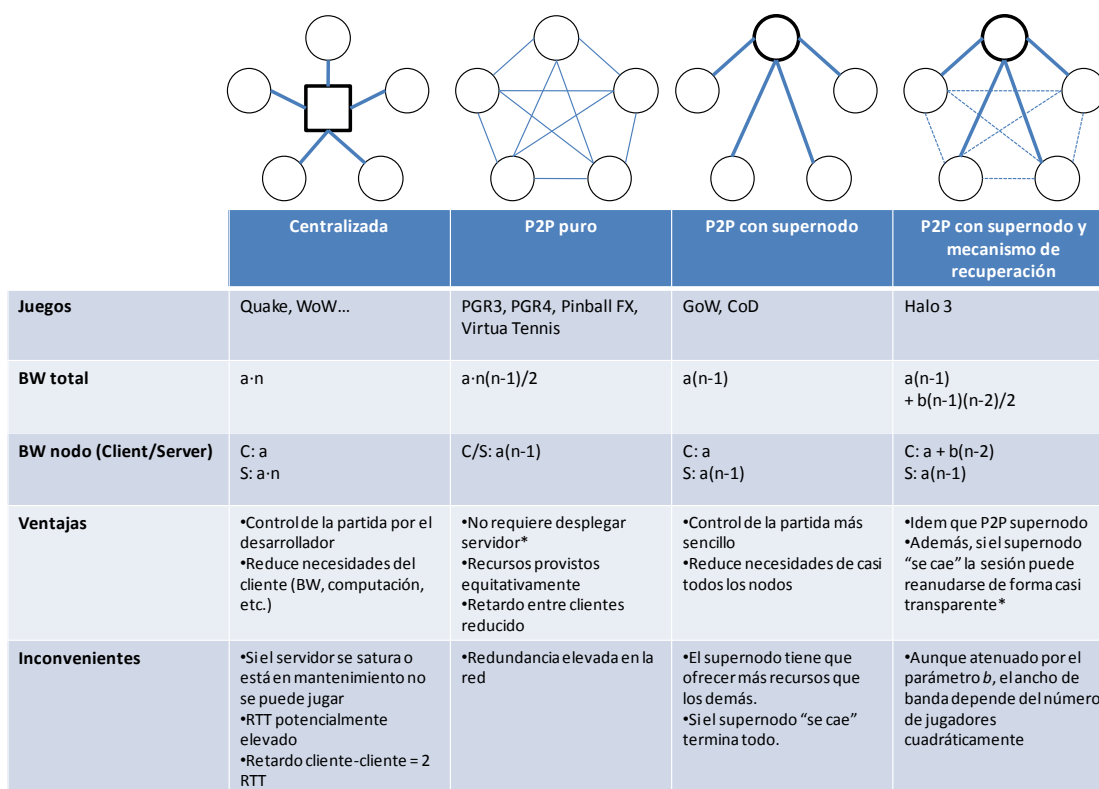
**Fig. 50. Consumo de ancho de banda de subida por jugador en distintos juegos de Xbox 360**

En primer lugar, cabe destacar el bajo ancho de banda requerido por PFX sin audio o vídeo, alrededor de 2kbps para dos jugadores y unos 5 kbps para cuatro jugadores. La razón es que cada jugador juega en una mesa de pinball distinta y los jugadores sólo conocen las puntuaciones de los adversarios en cada momento, no hay interacción directa entre ellos y el intercambio de las puntuaciones no requiere mayor ancho de banda. De hecho, este es el juego donde la incorporación de audio y vídeo supone un mayor impacto en requisitos de red, doblando el ancho de banda de subida requerido al introducir audio y multiplicándolo por diez si introducimos videoconferencia. Puede observarse que el ancho de banda de subida requerido en el caso de cuatro jugadores para vídeo es menor que para dos jugadores, la explicación es que en las partidas de cuatro jugadores la ventana de vídeo de cada jugador es más pequeña que en el caso de dos y, por tanto, ocupa menos.

En el caso de GoW y PGR3 el tráfico de juego es similar al identificado por Zander para Halo 2 [63], sobre los 15-30 kbps. Un aspecto clave en PGR3 es que el tráfico de subida de una consola es mucho mayor en el caso de ocho jugadores que en el de cuatro, en comparación con GoW. El análisis muestra que en PGR3 (y en PFX, de hecho) la información de juego se envía a todos los jugadores a la vez desde cada consola, siendo el tráfico directamente proporcional al número de jugadores. En GoW, sin embargo, uno de los jugadores actúa de servidor - denominado anfitrión - (la figura muestra un jugador que actúa de cliente), y el resto de jugadores le envían el tráfico, por lo que el ancho de banda de subida es prácticamente independiente del número de jugadores.

En ninguno de los casos estudiados existe un servidor que dé soporte a las partidas, todos los juegos analizados intercambian tráfico constante únicamente con las IPs de las consolas involucradas en la partida, por lo que todos los casos son P2P, bien completamente distribuidos (como PGR3 y PFX) o con supernodo (GoW). Sí que hemos observado que siempre existe un intercambio de tráfico puntual cada 10 segundos con un servidor situado en los Estados Unidos que, previsiblemente, es el servidor de presencia de Xbox Live y no está involucrado en la prestación del juego. Igualmente, durante los momentos anteriores y posteriores a la partida se contacta nuevamente con servidores de Estados Unidos, contactos previsiblemente correspondientes a los servicios de búsqueda de usuarios y registro de puntuaciones de Xbox Live.

Esta información nos sirve para elaborar una comparativa de las arquitecturas de red habituales en juegos multijugador populares y que hemos plasmado en la Fig. 51.



\* En Xbox 360, aunque los juegos son P2P, las funciones de establecimiento de partida, puntuaciones, etc. dependen de servidores centrales y si se pierde la conexión con ellos se impide jugar pese a ser técnicamente posible al haber conexión entre los peers. El modo ad-hoc sólo se contempla en red local.

**Fig. 51. Comparativa de arquitecturas de red habituales en juegos multijugador populares**

En la figura se presentan los distintos tipos de arquitecturas identificadas, los casos de P2P se corresponden a los estudiados en el sistema Xbox 360. Habíamos indicado que todos los juegos analizados de esta plataforma seguían topologías similares a PGR3 o GoW, sin embargo el juego Halo 3 introduce una mejora respecto del modelo

de GoW destinada a posibilitar la continuación de la partida en caso de que el jugador o la consola que ejerce de supernodo abandone la partida o, simplemente, pierda la conexión. Esta mejora se traduce en conservar un flujo de tráfico de ancho de banda reducido entre todos los jugadores para poder reconstruir el estado en caso de caída del anfitrión de la partida.

En general, delegar en una topología dependiente únicamente de los usuarios involucrados tiene la ventaja de prescindir del mantenimiento de servidores de soporte y, en este caso particular, la consola Xbox 360 se beneficia de que sólo ejecuta código firmado y el tráfico está cifrado, por lo que un usuario malicioso tiene poco margen de acción para hacer trampa (p.ej. indicar que ha conseguido una puntuación más alta que la real en un juego).

El habilitador LCE propuesto en la Tesis y el middleware HIMMA soportan diferentes topologías. Esta flexibilidad es necesaria porque incluso en plataformas comerciales no todos los juegos multijugador resuelven la comunicación entre los participantes con la misma topología, como hemos estudiado en el caso de la consola *Xbox 360*.

Tras este detallado análisis de las plataformas de juego actuales y la síntesis de características llevada a cabo, en la siguiente sección se discuten las ventajas de utilizar la arquitectura IMS y las contribuciones propuestas en la Tesis a la hora de ofrecer un servicio de juego en red multijugador basado en tecnologías estándar.

### **6.1.2 Presentación y análisis del caso de uso**

Para estudiar la adecuación para proveer servicios de juego en red del marco genérico de aplicaciones HIMMA propuesto en la Tesis, que extiende la arquitectura IMS a través del habilitador LCE y el middleware HIMMA, nos centraremos en un caso de uso enfocado en las distintas fases de interacción de un usuario con la plataforma de juegos. Este caso de uso nos proporcionará un hilo conductor para presentar las características deseables, estudiar cómo proporcionarlas en un entorno IMS con las ventajas asociadas que conlleva tanto la propia arquitectura IMS como los elementos complementarios propuestos en la Tesis. De igual forma, servirá para identificar las características innovadoras que permitan mejorar la experiencia de usuario en cada fase del caso de uso.

El caso de uso presentado a continuación pretende plasmar de forma secuencial la interacción habitual de un usuario, que llamaremos Alice para seguir la convención habitual, que quiere jugar con otras personas en un sistema de entretenimiento, como

podría ser la consola que tiene en el salón de su casa con su correspondiente servicio online o el teléfono móvil 3G que lleva en su bolsillo en todo momento.

*Imaginemos que Alice es una estudiante universitaria residente en Madrid que se encuentra en casa un domingo por la tarde y le apetece pasar el rato delante del televisor echando unas partidas con su recién estrenada consola de videojuegos con servicio online, pues la programación de esa tarde le resulta muy aburrida. Entonces, Alice coge el mando de la consola y enciende la misma para empezar a jugar.*

*Como el hermano pequeño de Alice vive con ella y también juega habitualmente a esa consola, al encender ésta lo primero que aparece es una pantalla para elegir el perfil de usuario que va a activarse en ese momento para tener acceso a las partidas guardadas, historial de juego, listas de amigos, etc. Dado que Alice tiene protegido por contraseña el acceso a su perfil de usuario para evitar que su hermano pueda curiosear su perfil o gastar bromas a sus amigos haciéndose pasar por ella, Alice introduce con el mando los caracteres que forman su contraseña, moviéndose por un teclado virtual que aparece en pantalla. Una vez listo, Alice ve su pantalla de bienvenida y puede empezar a interactuar con la plataforma de juegos y los demás usuarios.*

*Una vez autenticado en la plataforma, Alice elige uno de sus juegos favoritos de su lista de juegos disponibles y empieza a ejecutarse.*

*Elegido el juego, la pantalla muestra las opciones propias del mismo, entre otras la comenzar una partida multijugador, opción que Alice elige porque quiere jugar con otras personas por resultarle más entretenido que jugar contra la máquina. Selecciona una opción denominada partida rápida, que busca jugadores al azar, y entonces se queda en espera unos segundos mientras que el sistema encuentra otros jugadores que quieran jugar a ese mismo juego en ese momento.*

*Al poco tiempo, Alice ya está en una partida con un conjunto de jugadores al azar y, gracias a las capacidades multimedia de su consola y la conexión de banda ancha, puede hablar con los demás jugadores y escucharles mientras juegan.*

Este caso de uso básico nos servirá de punto de partida para identificar las distintas fases que intervienen en la interacción habitual con una plataforma de juegos, si bien se irá ampliando al tiempo que se presenten variaciones que ayuden a mejorar la definición del escenario y a identificar funciones innovadoras en el mismo.

Del análisis del caso de uso se desprende la Fig. 52, que muestra de forma secuencial las cinco fases identificadas que definen el ciclo de vida habitual de interacción con la

plataforma y que incluyen tres momentos de decisión para el usuario: *¿con qué juego?*, *¿a qué juego?* y *¿con quién juego?*.

El orden de fases indicado puede variar, por ejemplo, podría elegirse antes con quién se quiere jugar y luego decidir a qué juego, pero las fases identificadas van a estar siempre presentes y es en cada una de ellas donde la plataforma que dé servicio al usuario tiene que demostrar flexibilidad y adaptabilidad para mejorar su experiencia de uso.

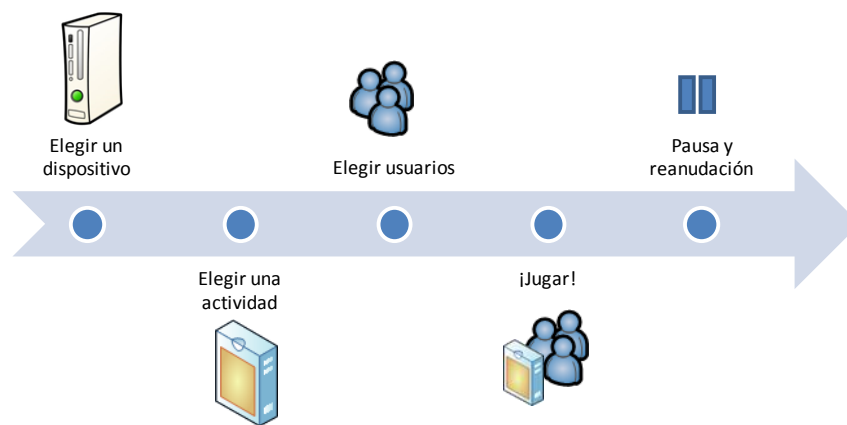


Fig. 52. Fases habituales de interacción con una plataforma de juegos

Las siguientes secciones profundizan en cada una de estas fases, indicando qué características o funciones de las plataformas analizadas intervienen más en cada una y cómo el IMS puede contribuir a ellas, aportando en cada caso las referencias a mecanismos estándar aplicables y a las contribuciones de la Tesis (habilitador LCE y middleware HIMMA).

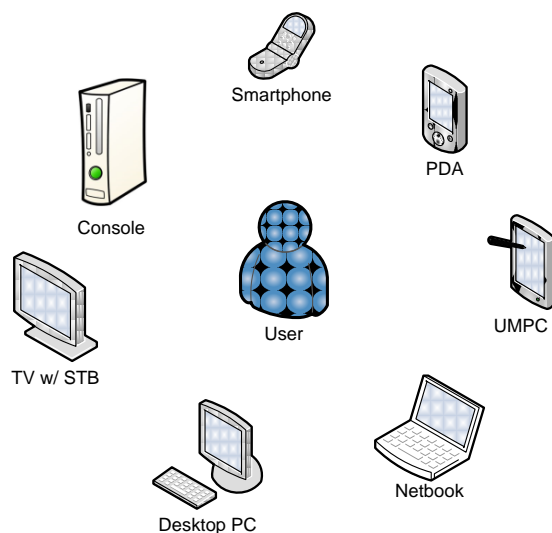
#### 6.1.2.1 Fase 1 - Eligiendo un dispositivo

La primera fase para disfrutar de las aplicaciones y servicios ofrecidos por cualquier plataforma, ya sea de entretenimiento como el caso que nos ocupa o de otros tipos, es imprescindible contar con un terminal de usuario (*user equipment*) apropiado para la aplicación, red de acceso y plataforma de servicios.

Como vimos al analizar la situación del mercado actual, las plataformas de entretenimiento existentes tienden a ser soluciones completas que abarcan el hardware, software y servidores necesarios para ofrecer el servicio a sus usuarios. Así, plataformas como *Battle.net* dan servicio exclusivamente a ciertos juegos del mismo fabricante (*Blizzard* en este caso) disponibles únicamente para ordenadores personales.

Otras plataformas, como *Nintendo WFC* o *Sony PS Network* están ligadas exclusivamente a consolas de sus respectivas compañías, mientras que *Xbox Live* sólo es accesible desde la consola de *Microsoft* u ordenadores con su sistema operativo *Windows Vista*. Otras, únicamente permiten acceder mediante aplicaciones instaladas en teléfonos móviles, ligados o no a una red de operador concreta.

En la práctica, si bien los usuarios disponen de múltiples dispositivos que ofrecen una gran flexibilidad para acceder a servicios en línea tanto dentro como fuera del hogar, como muestra la Fig. 53, a veces la decisión de qué dispositivo utilizar no depende únicamente de las preferencias o necesidades del usuario, sino que viene forzada por otros factores como la adecuación al tipo de aplicación o juego que se desea utilizar o si los amigos con los que se desea jugar tienen acceso a la misma plataforma de entretenimiento.



**Fig. 53. Fase de elección de dispositivo**

La *Open Mobile Alliance* señala en su reciente informe de evolución de juego en dispositivos móviles [15][15] las capacidades más importantes de los dispositivos para ofrecer una correcta experiencia al usuario. Aquí no nos vamos a limitar a dispositivos móviles, pero el conjunto de capacidades que señalan puede extrapolarse sin dificultad a cualquier tipo de dispositivo pues se centra en siete apartados generales: memoria, multimedia, gráficos, pantalla, teclado, ergonomía y conectividad.

En la Fig. 54 presentamos una síntesis de los aspectos cubiertos por el informe de OMA, adaptados a todo tipo de dispositivos. Las aportaciones principales son la introducción de nuevas características dentro de cada apartado que no aplican actualmente a terminales móviles pero que pueden, o podrán encontrarse en un futuro, en otro tipo de dispositivos habituales, como la alta definición y los gráficos 3D

estereoscópicos, por ejemplo. Además, hemos añadido la categoría de “entorno de ejecución”, pues en los móviles el estándar de facto para juegos es J2ME, pero en otros dispositivos hay más entornos además de Java, como Flash, AJAX, etc. Igualmente, hemos agrupado las categorías “teclado” y “ergonomía” en una nueva que hemos denominado “controlador” para ganar en generalidad, pues en el documento original la ergonomía hace referencia a métodos de control alternativos al teclado y, actualmente, las consolas no tienen teclado y hasta algunos móviles empiezan a sustituirlos por interfaces táctiles.

La plataforma de entretenimiento ideal debería permitir que el usuario elija en cada momento el dispositivo y red de acceso más apropiado en cada momento para sus necesidades, acomodando en la medida de lo posible la prestación de los servicios a las limitaciones impuestas por las capacidades del terminal y aprovechando las capacidades de aquellos terminales más potentes.

El soporte de múltiples dispositivos es algo que las plataformas de juego actuales no soportan habitualmente al formar parte de una solución completa del fabricante, pero en la arquitectura IMS la independencia del acceso es un pilar fundamental, tanto en lo relativo a tecnologías de red como a tipos de dispositivos. En efecto, el protocolo de señalización SIP, clave en la prestación de servicios en redes IMS, soporta el registro de capacidades de los terminales y la negociación de capacidades entre extremos para acordar sesiones multimedia aptas para los extremos involucrados.

El habilitador LCE propuesto en la Tesis como parte de un conjunto de extensiones a la arquitectura IMS (ver Capítulo 3) tiene en cuenta este aspecto a la hora de seleccionar el mejor modo de interacción para el dispositivo empleado en función de sus capacidades.

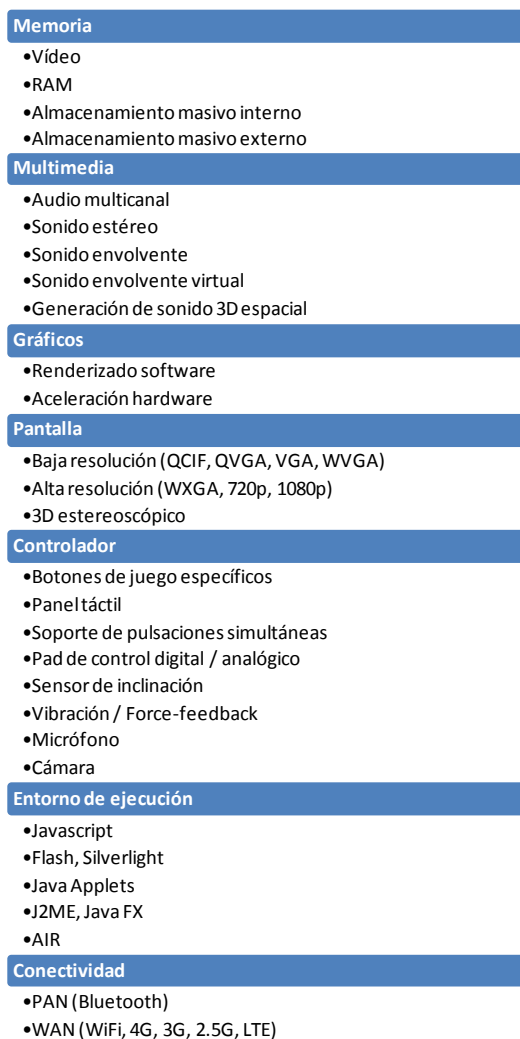


Fig. 54. Síntesis de capacidades de dispositivos de juego

### 6.1.2.2 Fase 2 – Registrándose en la plataforma

Elegido el terminal con el que se va a acceder a los servicios provistos por la plataforma, tiene lugar el proceso de registro, mediante el cual tanto el usuario como el propio terminal, en su caso, se autentican para adquirir los permisos necesarios de uso de los distintos servicios y poder personalizarlos de acuerdo a la identidad del usuario y, en caso de que la plataforma lo soporte, las características del terminal utilizado.

Nótese que nos referimos con “registro” al proceso de autenticación habitual a la hora de acceder a los servicios. En otros ámbitos, como en el de servicios ofrecidos en páginas web, esta fase recibe el nombre de “inicio de sesión” (curiosa traducción al español del inglés *login*, donde no hay referencia a sesiones), que no utilizaremos aquí para evitar confusiones con el inicio de sesiones del protocolo SIP. Por otro lado, mientras en otros ámbitos denominan “registro” al proceso de dar los datos necesarios (personales, bancarios, etc.) para disponer de un nombre de usuario y contraseña en el sistema, aquí lo denominaremos “darse de alta”.



Tomando como ejemplo la consola *Xbox 360* de *Microsoft*, el usuario que quiera jugar en red con ella tendrá que darse de alta la primera vez en *Xbox Live*, aportando sus datos personales y una cuenta de correo electrónico válida, así como elegir un nombre de usuario, una contraseña y aceptar el contrato con las condiciones de servicio aplicables. Con este proceso de darse de alta, el usuario ya posee un perfil de usuario en la plataforma donde almacenar su historial de juegos, amigos, etc. Posteriormente, cada vez que la consola se enciende, si está conectada a Internet, el usuario selecciona su perfil y puede registrarse (iniciar sesión según la traducción) en la misma para acceder a los servicios en línea que proporciona. El proceso de autenticación puede ser transparente en sucesivas ocasiones si el usuario marcó la opción de recordar contraseña, en cuyo caso es el propio dispositivo el que se autentica con la contraseña almacenada sin intervención del usuario.

Si ese mismo usuario tuviera además una consola *Sony PS3*, por ejemplo, tendría que seguir el mismo procedimiento de darse de alta la primera vez, en este caso en la *Sony PS Network*, aportando todos los datos y obteniendo otro nombre de usuario y contraseña, así como registrarse cada vez que vaya a jugar con la misma.

#### **6.1.2.2.1 Registro único**

En un entorno IMS, al contrario que sucede en servicios prestados por Internet, los usuarios ya cuentan con un identificador de usuario privado en forma de URI (IMPI - *IP Multimedia Private Identity*) con el que se registran y autentican en la red al encender el terminal (ver Fig. 55) y uno o varios identificadores públicos (IMPU - *IP Multimedia Public Identity*) con los que pueden acceder a los distintos servicios y que están ligados a la misma identidad, es decir, comparten la información del perfil de usuario.

De esta manera, un proveedor de servicios de juegos que ofrezca servicios de juego en línea en el IMS no tiene por qué obligar al usuario a darse de alta explícitamente en el servicio y aportar todos sus datos, incluidos los de facturación, al tiempo que simplifica su infraestructura de gestión de usuarios. Por otro lado, los usuarios de IMS podrían mantener la misma identidad en todos los juegos o aplicaciones, independientemente del proveedor y cambiar entre proveedores de juego de forma transparente.

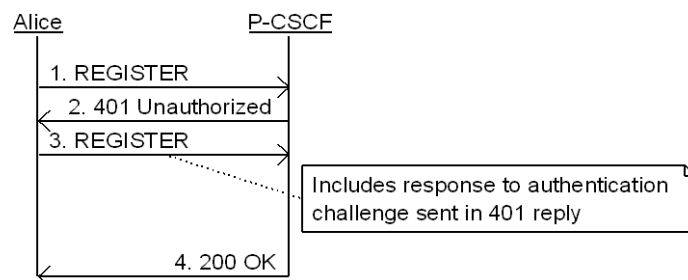


Fig. 55. Registro con autenticación en IMS

### 6.1.2.2.2 Perfil de usuario

Todos los usuarios de la red IMS cuentan con un perfil de usuario almacenado en el HSS que incluye todos los datos relativos a la prestación de servicios, incluyendo credenciales de autenticación, datos de facturación y condiciones de acceso a los distintos servicios que proporcione la red. El HSS permite además almacenar pequeños datos específicos de la aplicación (*transparent data*, [65] y [66]). Así, el proveedor del servicio de juegos no precisa desplegar y mantener una base de datos de usuarios, mientras que para los usuarios el acceso a los servicios de juegos es transparente porque no tienen que volver a dar sus datos para darse de alta en el mismo.

Los usuarios de IMS están identificados por una URI en todo momento, manteniendo una misma identidad en todos los servicios, si bien pueden contar con varias identidades públicas (por ejemplo, ser visible con cierta identidad al servicio de juegos y con otra distinta para la empresa donde trabaja).

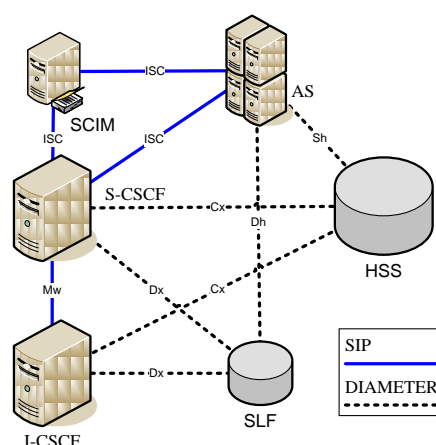


Fig. 56. Elementos de IMS relacionados con el perfil de usuario

En un entorno con varios HSS, como podría ser el de un operador nacional, el SLF (*Subscriber Location Function*) permite localizar el HSS correspondiente al perfil de usuario que se desea consultar.

El HSS no obstante sólo es indicado para almacenar pequeños datos muy ligados al funcionamiento del servicio (condiciones de encaminamiento de mensajes SIP a servicios concretos, por ejemplo). Para otro tipo de información como grupos, presencia, y en general, cualquier información XML, se emplea el habilitador de gestión de documentos XML (XDM). El perfil de usuario por ejemplo también puede almacenarse en el XDM de forma estándar de acuerdo a la especificación de *Shared Profile* propuesta por OMA. Este perfil es más genérico que el existente en el HSS, así, en el HSS se almacenan datos privados como aspectos de facturación de servicios, mientras que en el XDM se almacenan datos de perfil con mayor visibilidad como aficiones, actividades favoritas, idiomas hablados, etc.

El habilitador LCE propuesto en la Tesis hace uso automáticamente del gestor XDM y el perfil de usuario público para mantener el estado de las actividades, como podría ser un juego, y ofrecer funciones de valor añadido como la búsqueda o la pausa y reanudación. De igual forma, el middleware HIMMA se aprovecha del registro único en IMS, no siendo necesario aportar datos adicionales para que el usuario participe en una actividad creada por el middleware.

### 6.1.2.3 Fase 3 – Seleccionando una aplicación

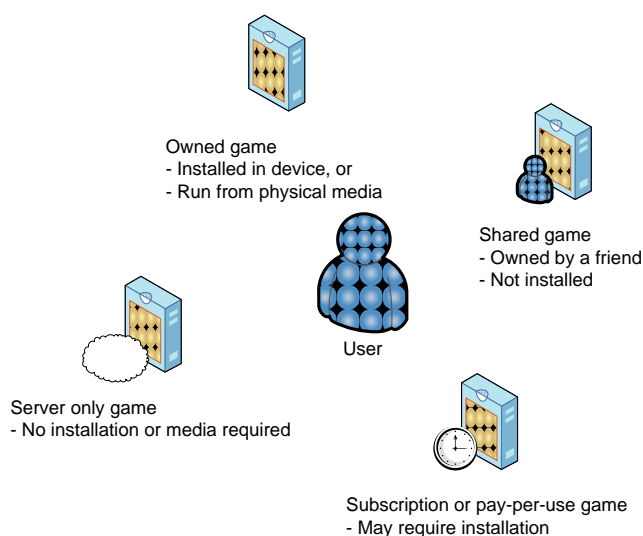
La tercera fase en el marco de la interacción del usuario con los servicios ofrecidos por la plataforma de entretenimiento es habitualmente elegir la aplicación o servicio a disfrutar.

En el caso de consolas de videojuegos, cuyos juegos se distribuyen principalmente en formato físico, ya sea en DVD, *Blu-ray* o formatos propietarios como cartuchos o mini-discos UMD, la selección de la aplicación conlleva escoger el soporte físico del juego e introducirlo en el dispositivo, aunque empiezan a introducirse mecanismos de descarga y tiendas virtuales que permitan prescindir del formato físico aprovechando el aumento de la capacidad de almacenamiento y el ancho de banda disponible.

No obstante, existen más alternativas para disfrutar de una aplicación además de ser el propietario del soporte físico o tener la aplicación instalada en el terminal, y todas ellas se presentan en la Fig. 57:

- El caso tradicional, clasificado como “juego en propiedad”, responde al escenario de compra de un juego o su descarga e instalación donde el usuario puede utilizarlo en cualquier momento en el terminal donde lo instaló.
- El caso de “juegos por suscripción o pago por uso”, son juegos que pueden incluir soporte físico y requerir instalación, pero que además necesitan realizar una suscripción o un pago para acceder a todas sus funciones. Este es el caso de los juegos de rol más conocidos como *World of Warcraft*.

- El caso de “juego compartido”. No es un caso habitual en las plataformas analizadas, pero sí es posible, por ejemplo, en las consolas portátiles *Nintendo DS*, que utilizan cartuchos propietarios como soporte físico y permiten, en juegos que lo soporten, las partidas multijugador entre amigos donde sólo uno de los participantes sea propietario del juego. Por supuesto, una vez finalizada la partida, sólo el propietario puede seguir disfrutando del juego e invitar a los jugadores que crea conveniente.
- El caso de “juegos de servidor”. Con la proliferación de diversas tecnologías que permiten acceder a aplicaciones sin necesidad de instalarlas o descargarlas y que han sido aplicadas fundamentalmente en entornos web, como los *applets* de Java, aplicaciones Flash o AJAX, algunos juegos multijugador sólo requieren soporte específico en el lado de servidor porque en los clientes es suficiente con un navegador compatible con la tecnología utilizada. Estos juegos suelen ser del tipo denominado “*casual*”, mini juegos sencillos, técnicamente poco exigentes para los terminales, pero suficientemente adictivos y que han demostrado un interés generalizado e incluso un cambio en el modelo de negocio: el juego en sí suele ser gratuito, pero con publicidad contextual asociada. Su facilidad de acceso ha sido clave para ganar popularidad, pero tienen el inconveniente de no ser convergentes: en un juego de *Xbox 360*, por ejemplo, tres jugadores pueden estar jugando y hablando, dejar el juego mientras siguen hablando e iniciar una partida a otro juego distinto, con una aplicación web actualmente no es posible.



**Fig. 57. Fase de elección de la actividad**

En un mercado donde la mayor parte de jugadores son del tipo denominado “*casual*”, en contraposición a los jugadores tradicionales denominados “*hardcore gamers*”, el IMS resulta idóneo para la provisión de este tipo de juegos y presenta una ventaja clara frente a los sistemas específicos de videojuegos: no es necesario contar con un terminal específico para juegos, los usuarios no precisan registrarse para jugar y

mantener un seguimiento de sus resultados, pueden jugar e invitar a sus amigos al mismo tiempo que hablan con ellos e incluso pueden acceder a funciones de valor añadido que sean de pago sin necesidad de aportar sus datos de facturación a terceros.

Las facilidades multidispositivo de LCE permiten, si el proveedor de la aplicación lo implementa, seleccionar el tipo de interacción más apropiado para un usuario en función de las capacidades de un dispositivo. Por ejemplo, si la aplicación precisa de software específico instalado en el terminal y no está instalada, el usuario recibe automáticamente los datos para su descarga, o si es posible la interacción a través de navegador web (juegos sólo de servidor) sin instalar software adicional se abrirá el navegador con la URL apropiada que le identifica, sin tener que darse de alta o registrarse en el servidor de juegos.

En el caso del middleware HIMMA va implícita la instalación de software adicional en el terminal (la propia aplicación que hace uso del middleware), por lo que no aplica a los casos de sólo servidor.

#### **6.1.2.3.1 Contenido descargable e información de juegos**

La información sobre juegos disponibles y el contenido descargable de los mismos es una función específica de cada proveedor de juegos, si bien la plataforma puede ofrecer un entorno unificado para publicar esta información y permitir la descarga del contenido de valor añadido mediante una red de distribución de contenido que tenga copias locales para mejorar la entrega al usuario final.

#### **6.1.2.3.2 Alquiler y compra de juegos**

El IMS incluye una arquitectura de cobro (*charging architecture*), definida en [67], [68], [69] y [70], que permite controlar el acceso a los servicios de pago por parte del usuario y facilita la oferta de servicios de valor añadido a los proveedores al no tener que gestionar los datos de facturación de los usuarios.

La arquitectura de cobro, ilustrada en la Fig. 58, contempla dos tipos de cobros posibles:

- *Offline*: Se usa en servicios cuyo pago sea periódico, típico en servicios de suscripción que se pagan a final de mes.
- *Online*: Se usa en servicios cuyo pago se produce en el momento de usarlos.

En una misma sesión se puede incurrir en los dos gastos, por ejemplo, se puede tener contratado de forma mensual el uso de los servicios específicos de juegos (mantenimiento de rankings, torneos con premio, etc.) y, adicionalmente, descargar un nuevo juego pagándolo en el momento.

En el caso de cargo *offline*, los nodos de la red involucrados enviarán información de los gastos al nodo *Charging Data Functions* (CDF) mediante la interfaz Rf (basada en el protocolo Diameter), que se encargará de componer toda esa información para crear el *Charging Data Record* (CDR), con información de todos los gastos, que lo enviará al *Charging Gateway Function* o CGF. Éste, a su vez, enviará el CDR al sistema de facturación *Billing System* (BS), externo al IMS y que se encarga de cobrar los gastos.

En el caso *online*, todos los gastos imputables serán enviados al *Online Charging System* (OCS) a través de la interfaz Ro (también basada en Diameter) que a su vez lo remitirá al BS para su descuento inmediato.

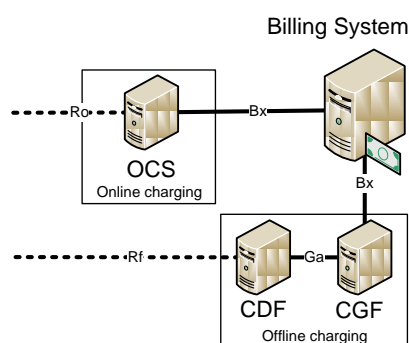


Fig. 58. Elementos relacionados con la tarificación de servicios en IMS

En las propuestas presentadas en la Tesis no hemos entrado en el campo de la facturación de servicios, si bien la orientación a sesiones de LCE y el middleware HIMMA permiten la aplicación de criterios de facturación. Por ejemplo, una actividad definida en el LCE que quiera marcarse como de “suscripción” podría tener *triggers* (disparadores) asociados en el HSS de modo que las peticiones de inicio de sesión hacia esa actividad sólo se cursen si el usuario es suscriptor. Este mecanismo es aplicable incluso para las interacciones vía web soportadas por LCE, pues el punto de partida inicial es el establecimiento de una sesión y, si el usuario no fuera suscriptor, el usuario no recibiría la URL única del servidor que debe atender la actividad.

### 6.1.2.3.3 Regalo de juegos

La función de regalo de juegos – sin formato físico, se entiende – no es muy habitual en las plataformas de entretenimiento actuales, de hecho ninguna de las grandes plataformas actuales lo soporta, lo cual contrasta sobremanera con el hecho de que una parte importante de la recaudación del sector de videojuegos corresponde a regalos –

especialmente en Navidad, como explica Carlos Iglesias, secretario general de aDeSe<sup>8</sup> en [71].

En otros ámbitos sí que encontramos estas funciones de regalo de contenidos descargables, como en la tienda de música iTunes de Apple [72] que, a propósito, también distribuye juegos para la familia de dispositivos iPod e iPhone - aunque no la consideramos plataforma de juegos por no aportar más funciones que la propia de distribuir el juego y gestionar el pago si no fuera gratuito.

El caso de la tienda iTunes, generalizable a otros sistemas, nos va a permitir identificar uno de los principales inconvenientes que existe para desplegar de forma exitosa la característica de regalo de juegos, y no es otra que la dificultad para hacer el regalo. En esta conocida tienda de música, para hacer un regalo hay que ser usuario de iTunes, estar dado de alta y tener una tarjeta de crédito asociada, elegir un producto e indicar la dirección de correo electrónico del destinatario del regalo. Tras efectuar el pago del regalo, el destinatario recibe un correo con las instrucciones para crearse una cuenta de iTunes, si no tiene, y descargarse la canción regalada a través de la tienda.

Si se implementara un mecanismo análogo en cualquiera de las plataformas de juegos actuales:

- La persona que quiere regalar un juego a alguien tendría que ser usuario de la plataforma que da servicio a ese juego, cuando posiblemente ni siquiera esa persona sea jugador habitual (caso típico de los padres que regalan un juego a su hijo).
- La persona que quiere regalar un juego tiene que introducir los números de su tarjeta de crédito en el servicio de venta de juegos, con la desconfianza que puede suponer.
- La persona que quiere regalar un juego tiene que proporcionar un dato que identifique al receptor del regalo, como la dirección de correo electrónico asociada al servicio de juegos (que suele diferir de la usada normalmente por el usuario para sus comunicaciones personales), o los identificadores internos que utilice la plataforma. En la práctica es una complicación añadida si el que regala no conoce esa información y tiene que averiguarla, corriendo el riesgo de descubrir la sorpresa.

En un entorno basado en IMS, el servicio de regalo de juegos, extrapolable a otro tipo de contenidos digitales e incluso físicos, puede simplificarse de cara a los usuarios implicados, fomentando el regalo de este tipo de bienes:

---

<sup>8</sup> Asociación Española de Empresas de Software de Entretenimiento

- La persona que quiere regalar un juego a alguien, sólo por tener un teléfono móvil, ya está registrado en la red IMS y puede acceder a servicios de valor añadido de pago, sin ser usuario de ningún servicio de juegos.
- La persona que quiere regalar un juego no precisa dar sus datos bancarios a una tercera empresa porque el cobro se verá reflejado en la factura telefónica.
- La persona que quiere regalar un juego, únicamente tiene que proporcionar el identificador público del destinatario en la red IMS, independientemente de que dentro de cada juego el destinatario muestre otras identidades.

#### 6.1.2.4 Fase 4 - Eligiendo los jugadores

La cuarta fase identificada es la previa al establecimiento de la partida multijugador o, en general, de la aplicación multiusuario que vaya a consumirse. En este punto el usuario ya tiene el terminal apropiado, está registrado en la plataforma que le ofrece el servicio, sabe a qué va a jugar y sólo le hace falta tener alguien con quién jugar.

Las opciones en este caso son múltiples, dependiendo de las facilidades que ofrezca la plataforma, y se resumen en la Fig. 59. Habitualmente, el jugador dispone de una lista de contactos con sus amigos que, combinada con la información de presencia, es un punto de partida donde encontrar gente disponible para jugar. Una extensión de este concepto es el de listas de amigos específicas de la aplicación, poco frecuente en las plataformas analizadas y que puede resultar muy interesante pues es posible que los usuarios tengan amigos favoritos para jugar a cierto juego pero no a otros. Esta lista de favoritos puede incluso generarse dinámicamente en función del número de partidas establecidas con cada usuario y aplicación, de forma transparente para los usuarios.

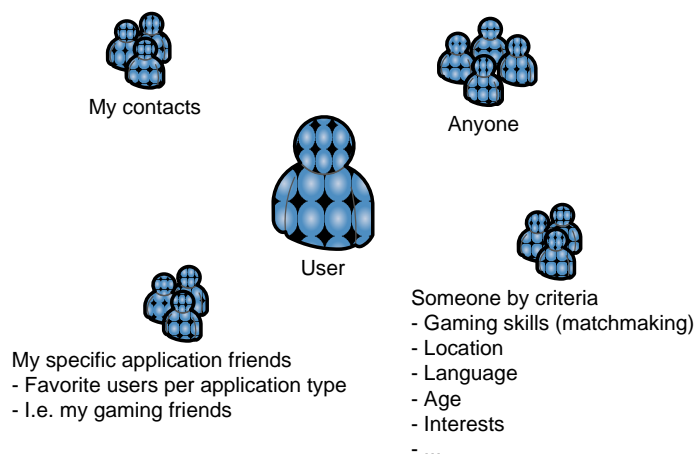


Fig. 59. Fase de elección de compañeros de juego



Otras alternativas para encontrar alguien con quien jugar residen en las funciones de búsqueda de usuarios desconocidos, que permiten localizar usuarios disponibles para jugar siguiendo una serie de criterios, como el nivel de habilidad en ese juego, el idioma, el país o cualquier otro dato disponible de los perfiles de usuario. El establecimiento de torneos o competiciones organizadas de juego también constituye una forma implícita de encontrar usuarios, así como el descubrimiento de usuarios mediante mecanismos externos como páginas web o foros, en los que no nos centraremos.

#### 6.1.2.4.1 Listas de amigos

La gestión de grupos de contactos o *buddy lists* se implementa en IMS mediante el servidor XDMS, presentado en la Fig. 60, que sustituye al antiguo PGM (*Presence and Group Management*). El XDMS es un repositorio en el cual se almacenan documentos en formato XML usados por una o varias aplicaciones.

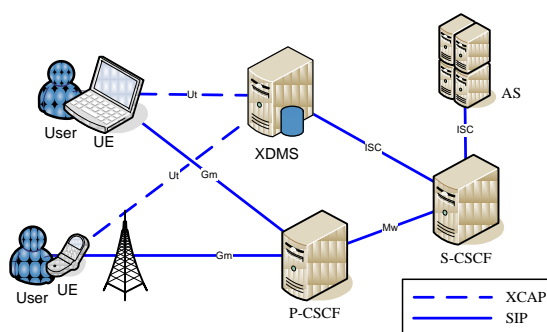


Fig. 60. Elementos relacionados con la gestión de grupos en IMS

La especificación de XDMS define dos características principales:

- Uso del protocolo XCAP (*XML Configuration Access Protocol*) con el que las aplicaciones pueden almacenar y manipular sus datos de servicio, almacenados como documentos XML. Puede verse un ejemplo de la sintaxis de este protocolo en el Listado 5, así como un flujo de mensajes habitual en la Fig. 61 y Fig. 62.
- Mecanismo SIP de suscripción/notificación en el que las aplicaciones pueden ser notificados de cambios de los documentos XML.

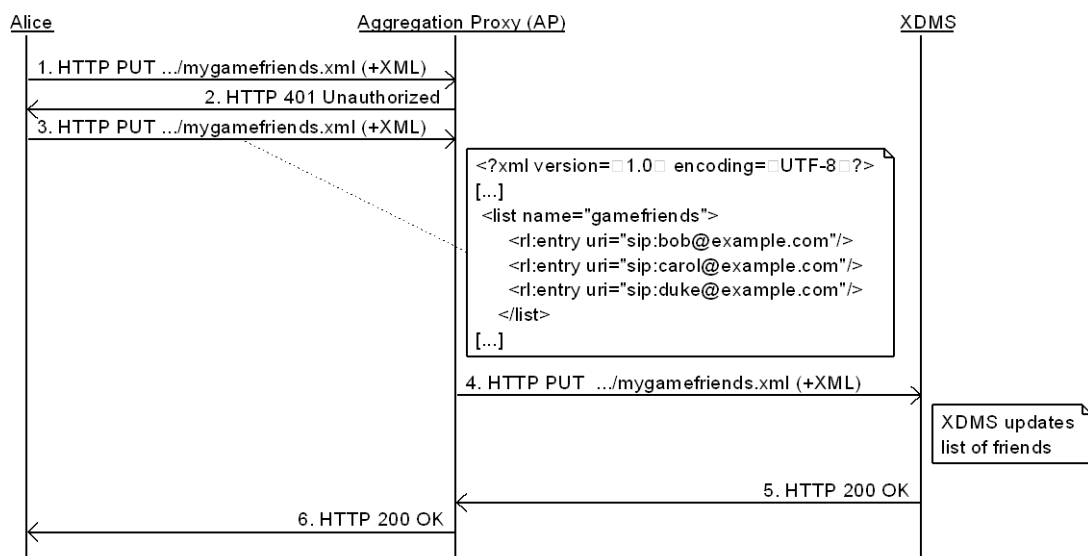
En el caso concreto del servicio de listas de amigos, en el XDMS pueden definirse varias listas, cada una con un nombre distinto, que en realidad son documentos XML que incluyen el conjunto de URIs o identificadores de los amigos.

```

PUT /services/resource-lists/users/sip:alice@ejemplo/amigos.xml
Content-Type: application/resource-lists+xml
Content-Length: (...)
<resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<list name="amigos">
<display-name>amigos</display-name>
<entry uri="sip:bob@ejemplo">
<display-name/>
</entry>
<entry uri="sip:carol@ejemplo">
<display-name/>
</entry>
</list>
</resource-lists>
    
```

**Listado 5.** Ejemplo de mensaje PUT para crear una lista de amigos en el gestor de grupos

Mediante una aplicación específica en el servidor de aplicaciones se podrían crear listas de amigos dinámicas compuestas, por ejemplo, por usuarios que tengan las mismas aficiones.



**Fig. 61.** Ejemplo de almacenamiento de contactos en el XDMS

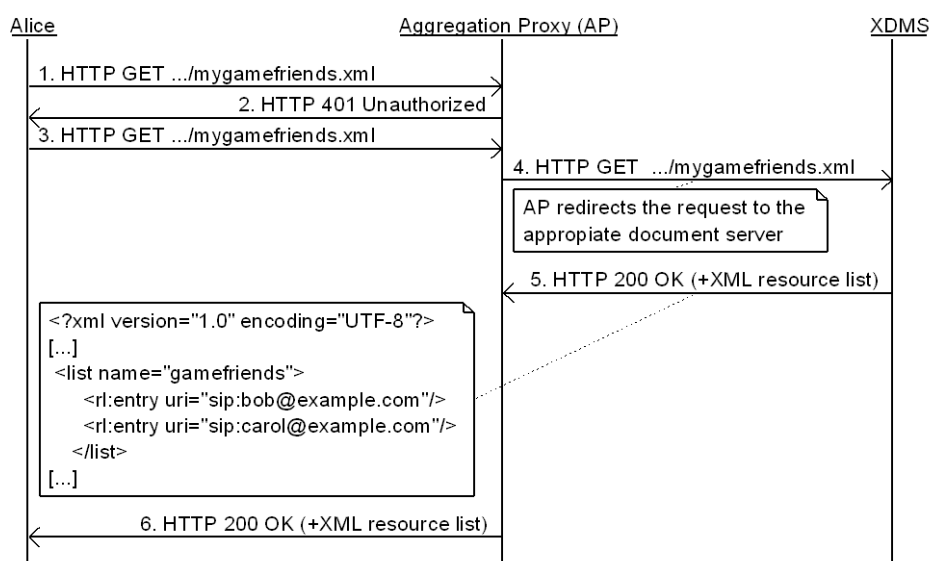


Fig. 62. Ejemplo de recuperación de contactos del XDMS

#### 6.1.2.4.2 Clanes

Los clanes son conjuntos de jugadores, habitualmente estables, que participan con otros en los juegos que lo soporten, por ejemplo carreras de coches donde cada clan forma un equipo, o juegos de batallas donde cada clan representa uno de los bandos.

En IMS, gracias al habilitador XDMS para gestión de grupos, combinado con el servicio de presencia de sus miembros, se puede conocer si un clan o grupo está conectado en cierto momento y preparado para formar parte de una actividad.

#### 6.1.2.4.3 Presencia

El servicio de presencia viene dado, al igual que la gestión de grupos, por el servidor XDMS. En este caso, la información de presencia se modela como un XML, cuyo formato está especificado en la RFC 4480, *Rich Presence Extensions to the Presence Information Data Format* [73].

La información de presencia va más allá del estado del usuario (conectado, ausente, no disponible, o frases escritas por el propio usuario), sino que contempla estados de ánimos, predisposición (*willingness*) a utilizar cierta aplicación (como por ejemplo, “me apetece jugar al cuatro en raya”) e información sobre los servicios en los que está participando en ese momento. Igualmente, conforme se muestra en la Tabla 6, la información de presencia especificada por [73] puede tener caducidad (desde, hasta) y notas personales, así como referirse tanto a la persona como al propio dispositivo.

Tabla 6. Elementos de información de presencia enriquecida

Element	From/until	Note	<person>	<tuple>	<device>
<activities>	x	x	x		
<class>			x	x	x
<deviceID>				x	
<mood>	x	x	x		
<place-is>	x	x	x		
<place-type>	x	x	x		
<privacy>	x	x	x	x	
<relationship>		x		x	
<service-class>		x		x	
<sphere>	x		x		
<status-icon>	x		x	x	
<time-offset>	x		x		
<user-input>			x	x	x

Los contactos de un usuario, así como aplicaciones externas, pueden conocer los cambios en la información de presencia de un usuario a través del mecanismo de notificación de eventos de SIP, implementado en el XDMS.

El middleware HIMMA propuesto en la Tesis hace un uso intensivo del servicio de presencia y la información enriquecida para modelar las actividades en las que participan los usuarios, aislando al desarrollador de la interacción con el servicio de presencia.

#### 6.1.2.4.4 Creación de salas de juego y partidas igualadas

En las plataformas de juego disponibles actualmente, los usuarios que desean encontrar jugadores para establecer una partida entran en las denominadas salas de juego, quedando a la espera de que la plataforma los localice.

Gracias al habilitador de presencia de IMS no es necesario que el usuario espere activamente a que el sistema encuentre un jugador apropiado para iniciar una sesión de juego, con lo que el concepto de sala de juego desaparece, sino que puede especificar su predisposición (*willingness*) a usar cierta aplicación o juego y seguir con

sus actividades. Mientras tanto, el servicio de búsqueda de usuarios, que cuenta con acceso a la información de presencia de los jugadores y a su perfil de jugador (almacenado en el HSS, XDMS o un servidor externo), puede organizar automáticamente las partidas cuando encuentre usuarios que cumplan los criterios establecidos (nivel de habilidad si es una partida igualada, idioma, etc.).

El habilitador LCE propuesto puede resultar muy ventajoso para la organización de juegos. Las actividades en LCE se describen como ficheros XML en el servidor XDM y permiten la inclusión de condiciones de participación. Evidentemente, el perfil de usuario *Shared Profile* propuesto por OMA y utilizado por LCE no incluye información específica de juegos, por lo que habría que extenderlo –conforme a las reglas de OMA– con datos específicos como niveles de habilidad u otros parámetros comparativos que se utilicen a la hora de describir las condiciones de participación.

#### 6.1.2.4.5 Torneos

Se propone la creación de un habilitador de gestión de eventos multiusuario programados que permita el registro de participantes, la gestión de fases del torneo, y la notificación a los participantes de la proximidad del evento.

El habilitador permite la creación del torneo por parte de los organizadores del mismo, indicando fechas, requisitos de inscripción, coste, tipo de evento (mejor tiempo, uno contra uno, etc.) y otra información relevante. Una vez abierta la fase de inscripción, los usuarios se registran, quedando incluidos dentro de un grupo establecido en el servidor XDMS para la gestión del torneo. Cuando una fase del torneo está próxima, el gestor envía automáticamente un mensaje de recordatorio a los participantes. Finalmente, al llegar la hora de la competición, el sistema comprueba la información de presencia de los participantes e inicia las sesiones de juego automáticamente mediante un mecanismo de 3PCC (*Third Party Call Control*), de manera similar a como muestra la Fig. 63 para un hipotético torneo de póker virtual.

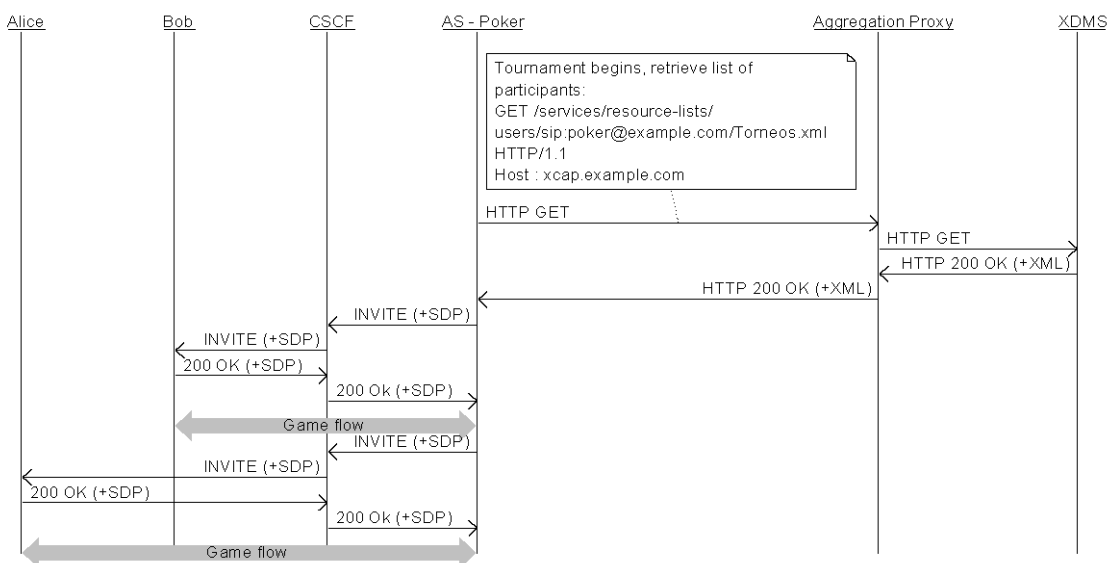


Fig. 63. Ejemplo del uso de XDMS y 3PCC para creación de torneos

#### 6.1.2.4.6 Sistema de reputación

El sistema de reputación se basa en las opiniones que los usuarios ofrecen de otros usuarios con los que han jugado, estando orientado principalmente a filtrar aquellos jugadores que, de alguna manera, deterioran la experiencia del juego por su comportamiento. Imaginemos por ejemplo un usuario de un juego de conducción que se dedica a chocar con otros usuarios con su vehículo para echarles de la carretera y llegar el primero. O imaginemos un entorno multimedia, donde las interacciones durante el juego habitualmente incluyen voz o vídeo entre los participantes, y uno de los jugadores emplea vocabulario obsceno u ofensivo para los demás.

La necesidad del sistema de reputación viene derivada de la posibilidad de jugar y comunicarse con usuarios desconocidos a través de los procedimientos de búsqueda de usuarios o creación de partidas igualadas.

La concepción de IMS y, en general, de los sistemas de establecimiento de conferencias en SIP, asume que las partes implicadas tienen interés en comunicarse y conocen sus identificadores de usuario o de los servidores de conferencia a los que conectarse, relegando en mecanismos externos el descubrimiento necesario de usuarios o servidores, que deben ser los que implanten los sistemas de reputación (recepción de opiniones, cálculo de los parámetros de reputación de cada usuario y consulta por terceros). Por supuesto, los usuarios tienen la opción de prohibir a ciertos usuarios ponerse en contacto con ellos mediante la denominada lista de bloqueo de contactos, una lista estándar de la agenda de direcciones del usuario tal que los usuarios que aparezcan en ella no pueden conocer el estado del usuario que los bloquea ni ponerse en comunicación con él.

Cabe destacar que la introducción del nodo XDMS para la gestión de grupos y la definición de un perfil de usuario compartido [74] sí constituye un primer paso hacia el establecimiento de comunicaciones con usuarios desconocidos (de hecho el estándar de OMA menciona explícitamente como posible aplicación encontrar usuarios para un chat). El inconveniente del mecanismo propuesto por OMA es que la información de perfil compartida con otros usuarios está especificada por el propio usuario, por lo que no tiene sentido indicar ahí los parámetros de reputación, fácilmente manipulables por el usuario valorado.

En este caso de búsqueda de usuarios por XDMS, las características de gestión de reputación de usuarios vendrían ofrecidas por un servidor al efecto, conforme muestra la Fig. 64.

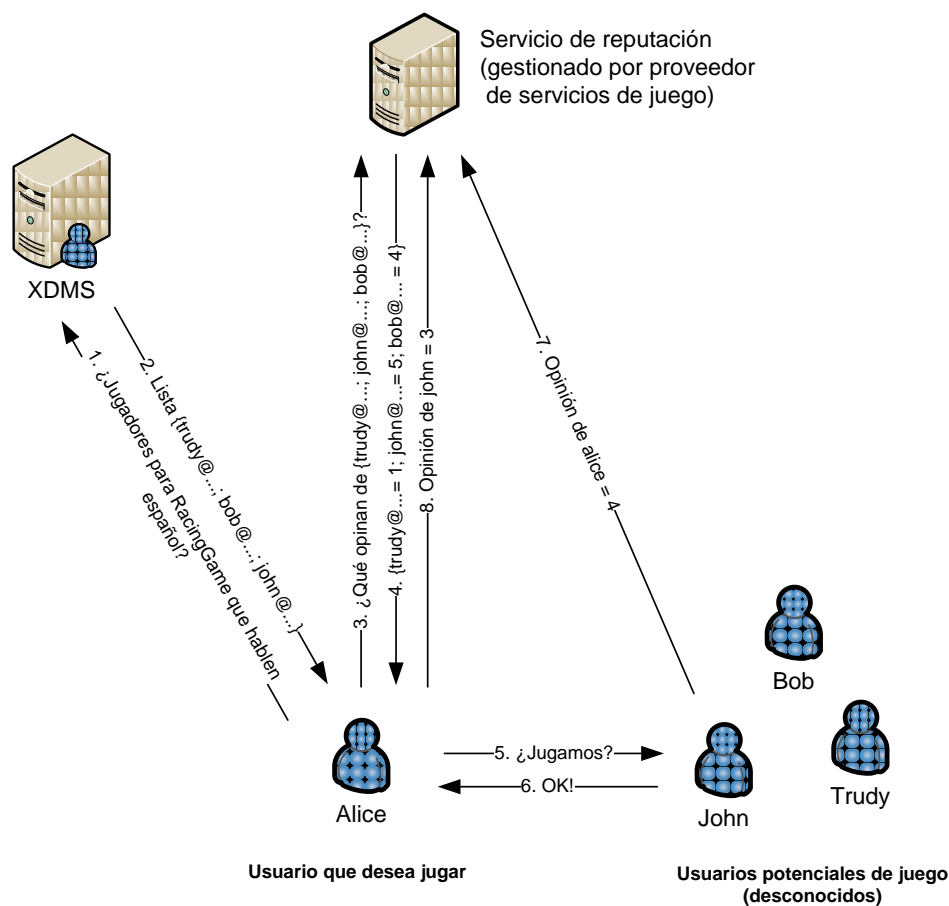


Fig. 64. Mejora del proceso de búsqueda de usuarios en IMS con mecanismo de reputación

El proceso de búsqueda mejorada con mecanismo de reputación se compone por tanto de tres pasos básicos:

- Obtención, mediante el XDMS, de los usuarios que cumplan los requisitos especificados (tipo de juego, idioma, proximidad, etc.) y filtrado posterior por valoración mediante consulta al servicio de reputación.
- Establecimiento de la actividad con el usuario desconocido mejor valorado disponible.
- Reporte de la opinión del usuario desconocido (por ambas partes).

El escenario mostrado en la Fig. 64 puede optimizarse desde el punto de vista de mensajes intercambiados a costa de una mayor complejidad en la arquitectura de servicios, que pasaría por integrar en el XDMS los mecanismos de cálculo de reputación para incluir automáticamente en la información compartida del perfil de cada usuario los parámetros de reputación calculados, evitando las dependencias con servidores externos en el proceso de búsqueda.

#### 6.1.2.4.7 *Rankings*

La función de ranking o clasificación es muy específica de juegos, dado su carácter competitivo, y la arquitectura IMS no proporciona ningún mecanismo que permita mantener este tipo de información, por lo que es necesario un servicio externo encargado de recibir las puntuaciones conseguidas por cada usuario en cada juego y elaborar las listas pertinentes. Este servicio es responsabilidad, lógicamente, del proveedor de servicios de juego.

#### 6.1.2.5 **Fase 5 - Jugando**

Con las fases anteriores ya está todo preparado: terminales, juegos y jugadores. Comienza la partida.

El establecimiento de una partida multijugador conlleva el establecimiento de los canales de comunicación adecuados entre las entidades involucradas, principalmente usuarios y servidores de juego en caso necesario: protocolos, puertos, formatos de la información, etc.

La ventaja de IMS es su orientación a sesiones multimedia, que se negocian entre las partes para acordar los parámetros de comunicación necesarios, gracias al mecanismo de establecimiento de sesiones de SIP, mostrado en la Fig. 65, que incluye entre otras funciones la negociación de capacidades.

En aplicaciones tan sensibles a los parámetros de red como son los juegos en red, IMS brinda la posibilidad de introducir mecanismos de calidad de servicio (QoS), que no son de aplicación en redes con encaminamiento *best-effort* como Internet.



Por otro lado, la capacidad multimedia de IMS hace que sea posible mantener, en paralelo al juego en sí, servicios de comunicación como voz o video conferencia, que pueden permanecer aun acabada la partida, contribuyendo a una mayor interacción entre los usuarios y proporcionando sensación de continuidad en el servicio de juegos. Incluso puede combinarse con otros servicios como el de presencia para indicar que en ese momento se está jugando y no aceptar llamadas entrantes, por ejemplo.

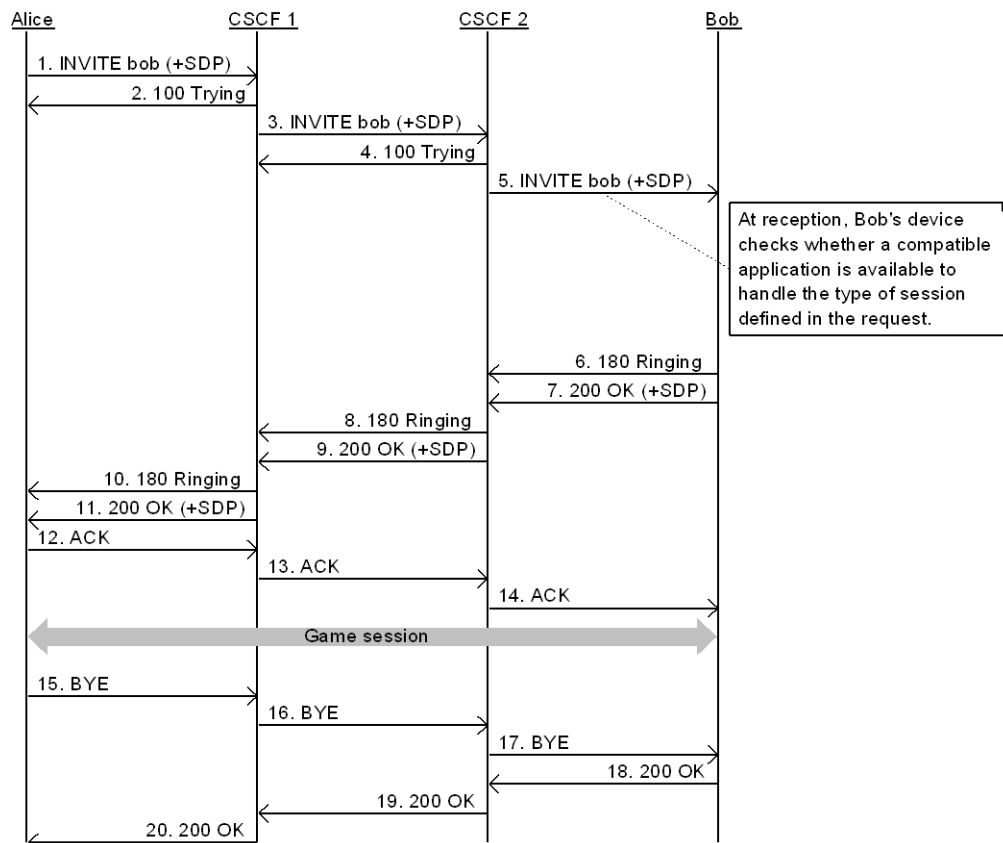


Fig. 65. Establecimiento de sesión de juego de dos jugadores

#### 6.1.2.5.1 Audioconferencia y videoconferencia

Las conferencias de audio y vídeo entre múltiples usuarios es una funcionalidad que en IMS se ofrece mediante un servidor de conferencias, que se apoya en el MRF (*Media Resource Function*) para la mezcla y distribución del flujo multimedia de cada usuario, aunque existe la posibilidad de que sean los propios terminales de los usuarios los que realicen dicha mezcla.

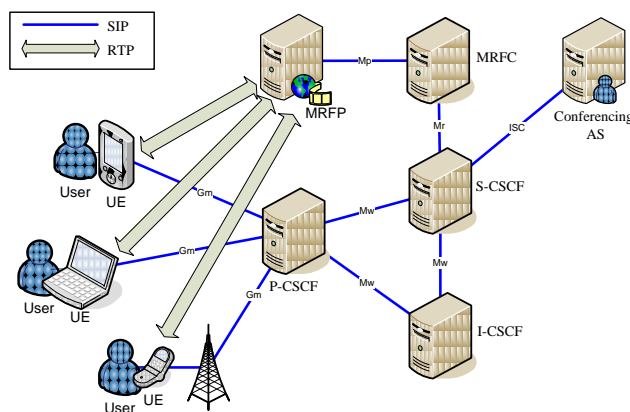


Fig. 66. Elementos de IMS relacionados con conferencias audiovisuales

La ventaja de emplear una solución de conferencia basada en MRF es que permite adaptar los flujos multimedia a cada usuario en función de sus capacidades (ancho de banda disponible, tamaño de pantalla, etc.), aprovechando las funciones de transcodificación que proporciona dicho nodo. Nuevamente, los escenarios de juego presentes en la actualidad requieren habitualmente el mismo tipo de terminal, exceptuando los mini-juegos web o juegos en Flash que siguen un criterio de funcionalidad mínima y ejecutan en cualquier navegador con soporte para esta tecnología, a costa de no aprovechar las capacidades de terminales más potentes que puedan tener los jugadores.

#### 6.1.2.5.2 Mensajería instantánea y mensajería multimedia

El soporte de mensajería y mensajería instantánea es habitual en los clientes de IMS. En el caso de mensajería instantánea se utiliza el protocolo MSRP (*Message Session Relay Protocol*) para establecer una sesión de mensajes directamente entre los participantes e intercambiar el texto de la conversación. En el caso de mensajería multimedia no instantánea se recurre a los mensajes MESSAGE de SIP, que no requieren el establecimiento previo de sesión y permiten incluir en el cuerpo del mensaje cualquier tipo de contenido MIME, de modo similar a los correos electrónicos.

Cabe destacar que la especificación de SIP no incluye un mecanismo para almacenar los mensajes en caso de que el usuario destino no pueda recibirlo, como ocurre en los mensajes SMS y MMS comunes actualmente en redes móviles GSM y UMTS, si bien se espera que sea una funcionalidad ofrecida por el operador de la red.

En caso de no disponer de soporte de almacenamiento y reenvío, el terminal del usuario emisor puede suscribirse a los cambios de presencia del usuario destino para reenviar el mensaje tan pronto esté disponible.

#### 6.1.2.5.3 Modo observador

El modo observador es un modo especial de participación donde el usuario no participa activamente en el juego y se limita a ver las acciones de los demás jugadores, por lo que no tiene ningún requisito adicional al ofrecimiento del juego más que poder indicar de alguna manera que el usuario desea añadirse únicamente como observador.

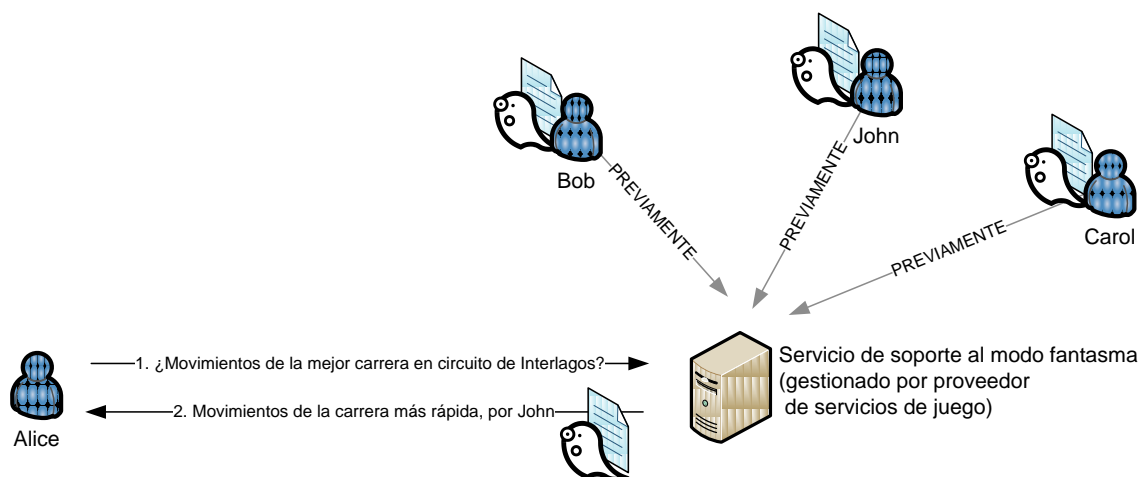
En IMS, heredado del protocolo de señalización SIP, es posible indicar el sentido de la comunicación a la hora de establecer una sesión, indicando en la descripción de la sesión (cuerpo con contenido de tipo SDP) si es bidireccional (*sendrecv*), de sólo recepción (*recvonly*) o de sólo envío (*sendonly*), que corresponderían respectivamente a una partida normal, a un usuario que desea observar una partida y a una invitación para que un tercero observe la partida.

#### 6.1.2.5.4 *Modo fantasma*

El denominado modo fantasma se basa en que uno o varios de los participantes de una actividad se comportan conforme a los movimientos y acciones de un usuario que fueron grabados con anterioridad, no pudiendo interactuar con el mismo (de ahí que esos participantes se representen de forma translúcida y se les denomine fantasmas). La aplicación de este mecanismo depende del tipo de juego, en concreto exige que las acciones de cada participante no condicionen las de los otros, como el caso habitual de carreras de coches contrarreloj, donde el fantasma representa la carrera con mejor tiempo de otro usuario o del mismo y es el objetivo a batir.

Desde el punto de vista de comunicaciones no presenta interactividad entre los usuarios porque los fantasmas corresponden a movimientos almacenados previamente y su tamaño suele ser razonable como para permitir la descarga completa de los movimientos del fantasma en los dispositivos locales y reproducirlo. No obstante, sí es necesaria la existencia de un repositorio que almacene estos movimientos, los clasifique siguiendo algún criterio (p.ej. tiempo en completar una vuelta al circuito de carreras), los ponga a disposición de otros jugadores y permita, del mismo modo, incorporar nuevos movimientos, conforme indica la Fig. 67.

Este servicio se presta por tanto a través del proveedor de servicios de juegos, que dispondrá de la infraestructura y aplicaciones de servidor necesarios para su ofrecimiento.



**Fig. 67. Servicio de soporte al modo fantasma**

Existe la opción de ofrecer la función de forma descentralizada, permitiendo que sean los propios usuarios los que sirvan, desde sus dispositivos, los movimientos almacenados de sus actuaciones en el juego. Sin embargo, esta solución conlleva bastantes limitaciones. La más evidente es que precisa que el usuario del que se quiera descargar el fantasma esté disponible en ese momento y pueda establecer una sesión de datos para transferirla. De igual modo, en una solución descentralizada, los usuarios tienen que conocer los identificadores de los usuarios a los que solicitar el fantasma - no es un problema si un usuario quiere competir contra el fantasma de un amigo que está en su agenda de contactos, pero no permitiría localizar el fantasma de "el usuario que ha hecho una mejor vuelta a este circuito", precisando nuevamente un servicio centralizado.

### 6.1.2.6 Fase 6 - Pausa y reanudación

La configuración actual de la partida (jugadores involucrados, tiempo real y tiempo del juego, estados, etc.) se puede almacenar en el XDMS para su posterior recuperación. En el caso de aplicaciones complejas donde la información de estado sea invariable, en el XDMS se almacenarán únicamente URLs que apunten a la localización real de la información de estado.

## 6.1.3 Conclusiones

### 6.1.3.1 Ventajas del LCE a un escenario de juego en red en IMS

La utilización del habilitador ligero de conferencias LCE especificado en el capítulo 4 supone una ventaja en cada una de las fases identificadas.

Fase	Beneficios de introducir LCE en IMS
<b>Eligiendo dispositivo</b>	Gestión automática del tipo de interacción del dispositivo por parte de LCE en función de las capacidades del mismo.
<b>Registrándose en la plataforma</b>	No es necesario un registro adicional al de IMS para disfrutar de su funcionalidad
<b>Seleccionando una aplicación</b> <b>Eligiendo a los jugadores</b>	LCE permite buscar tanto aplicaciones como usuarios que estén jugando en ese momento o hayan especificado su deseo de empezar a jugar.  Sin necesidad de infraestructura adicional de servidor.
<b>Jugando</b>	Permite establecer comunicaciones multiusuario tanto centralizadas como descentralizadas, liberando a los usuarios y servidores del mantenimiento del estado de la actividad.  Se facilita la recuperación de la actividad si existen problemas de conectividad con el servidor central o usuario que haga de servidor (foco).
<b>Pausa y reanudación</b>	Almacenamiento del estado de la actividad y posterior reanudación.

La Fig. 68 muestra un escenario de juego en red para tres jugadores donde Alice configura en el LCE a qué aplicación y el tipo de usuarios con los que quiere jugar. Posteriormente, Bob y Carol (el procedimiento de Carol no se muestra por ser análogo al de Bob) buscan usuarios que quieran jugar y encuentran la actividad definida por Alice, a la cual se unen. El diagrama incluye la interacción del habilitador LCE con los nodos de IMS, con la única simplificación de no mostrar la señalización entre CSCFs.

### 6.1.3.2 Aplicación del middleware HIMMA a un escenario de juego en red en IMS

La utilización del middleware HIMMA especificado en el capítulo 5 supone una ventaja en cada una de las fases identificadas.

Fase	Beneficios de desarrollar el juego con el middleware HIMMA en IMS
<b>Eligiendo dispositivo</b>	Facilita el desarrollo de aplicaciones
<b>Registrándose en la plataforma</b>	No es necesario un registro adicional al de IMS para disfrutar de su funcionalidad
<b>Seleccionando una aplicación</b>	El middleware actualiza la información de presencia y es posible conocer las actividades en las que están los amigos.
<b>Eligiendo a los jugadores</b>	Sin necesidad de infraestructura adicional de servidor.
<b>Jugando</b>	Permite establecer comunicaciones multiusuario tanto centralizadas como descentralizadas, liberando a los usuarios y servidores del mantenimiento del estado de la actividad.
<b>Pausa y reanudación</b>	Facilita el proceso de pausa y reanudación al contar con un API sencilla, sin embargo es necesario que la lógica de la aplicación gestione el estado interno.

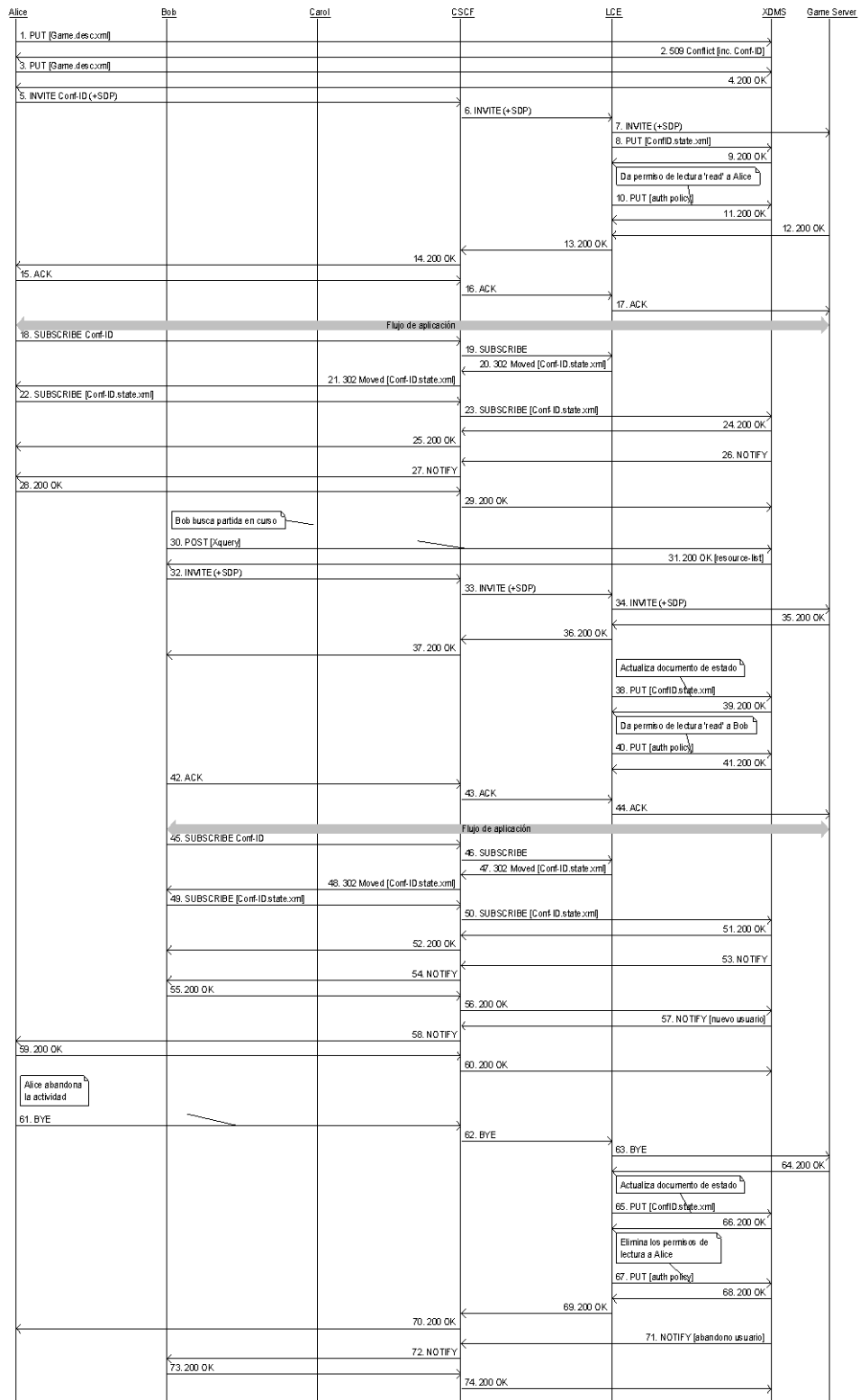


Fig. 68. Creación y búsqueda de partidas con LCE

## 6.2 Servicios profesionales

En esta sección se muestra la aplicación de las contribuciones de la Tesis para implementar una serie de servicios básicos (sala de espera, videoconferencia orientada a grupos y pizarra virtual) de utilidad para aplicaciones de tipo profesional. En particular, los servicios expuestos se han desarrollado en el marco de una aplicación de tele-asistencia en el proyecto AmIVital, si bien son aplicables a otros ámbitos profesionales dada su generalidad.

El objetivo general del Proyecto CENIT AmIVital (2007-2010), es el desarrollo de una **nueva generación de tecnologías y herramientas TIC** para el modelado, diseño, operación, e implementación de dispositivos y sistemas de inteligencia ambiental (AmI), cuyo fin es la provisión de servicios y soportes personales de la vida independiente, el bienestar y la salud. Para el control de las comunicaciones en AmIVital se ha propuesto la arquitectura de red IMS por la flexibilidad que ofrece a la hora de diseñar servicios de valor añadido. Desde un punto de vista lógico, el control de las comunicaciones comprende un conjunto de capas bien diferenciadas y con distintos niveles de abstracción, como muestra la Fig. 69. Ya vimos en 2.2 que, lejos de ser una simple capa de transporte IP, IMS presenta el concepto de habilitadores de servicios como componentes reutilizables por las aplicaciones que hacen uso de las capacidades subyacentes de la red y evitan duplicar funcionalidad entre aplicaciones. Ejemplos de habilitadores son gestión de grupos de usuarios, presencia, mensajería instantánea, videoconferencia, o el propio habilitador LCE propuesto en la Tesis.

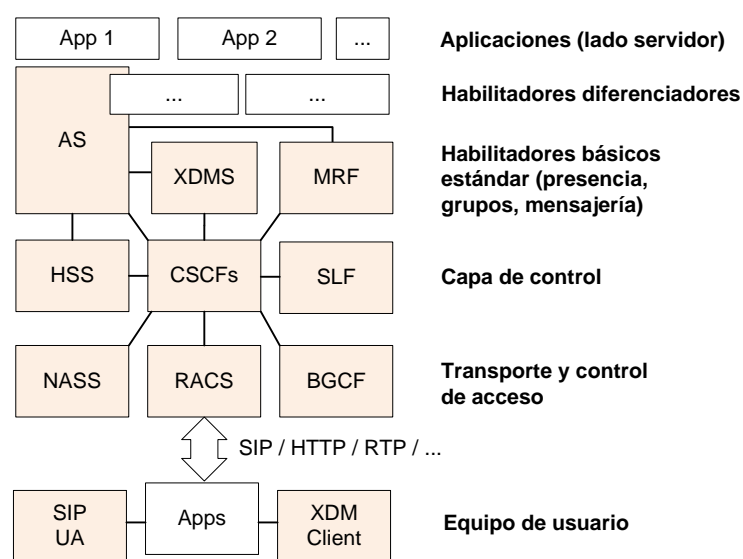


Fig. 69. Agrupación funcional de los nodos de control de las comunicaciones



En el marco del Proyecto AmIVital se definen habilitadores diferenciadores innovadores que soportan las distintas aplicaciones orientadas al bienestar y la salud, como aquellas propias del escenario de consulta médica remota. El conjunto de habilitadores o servicios de valor añadido especificados (y desarrollados como prueba de concepto) incluyen:

- **Cola de espera virtual**, para establecer el orden en que un grupo de usuarios puede acceder a los recursos limitados de una aplicación y mantener a los usuarios en espera hasta que puedan ser atendidos.
- **Sala de espera virtual personalizada**, para ofrecer a cada usuario acceso a distintos recursos multimedia relevantes durante el periodo de espera asociado a la prestación de otros servicios (p.ej. una consulta médica remota basada en videoconferencia).
- **Videoconferencia orientada a grupos**, para permitir una conferencia de vídeo entre los miembros de un grupo de usuarios.
- **Presencia enriquecida**, para conocer la disponibilidad de los usuarios más allá del clásico disponible/ausente/desconectado, proporcionando una mayor información acerca de las capacidades de los dispositivos del usuario y el uso de los servicios disponibles.
- **Compartición de imágenes con anotaciones (pizarra virtual)**, para permitir a un grupo de usuarios compartir imágenes en tiempo real y realizar anotaciones en las mismas.
- **Repositorio de imágenes**, para permitir almacenar contenido gráfico en un repositorio (p.ej. radiografías) y utilizarlo posteriormente en combinación con otros servicios (p.ej. pizarra virtual)
- **Servicio de calendario**, para permitir la gestión y automatización de recordatorios de citas y otros eventos.

Todos los habilitadores definidos proporcionan una doble interfaz:

- Una interfaz IMS que permite la comunicación con el núcleo IMS usando sus métodos y protocolos estándar (SIP, XCAP y Diameter).
- Una interfaz de Servicios Web, la cual permite acceder a las diferentes funciones de los habilitadores abstrayendo los protocolos específicos y la arquitectura de IMS.

Gran parte de los habilitadores o servicios desarrollados en el transcurso del proyecto AmIVital tienen en común la gestión de múltiples usuarios y comunicaciones multimedia de diversos grados de interacción con terminales potencialmente heterogéneos (terminales móviles, equipos dedicados, ordenadores de sobremesa, etc.),

por lo que constituyen una excelente fuente para validar en la práctica el habilitador ligero de conferencias, LCE; presentado en esta Tesis.

A continuación se presentan con más detalle los habilitadores diferenciadores desarrollados más significativos dentro del ámbito de la Tesis y que han servido de aplicación de los conceptos aquí expuestos. En concreto, se presentan las ventajas que aporta la introducción del habilitador ligero de conferencias LCE como base para la creación de dichos habilitadores diferenciadores o servicios de valor añadido.

### **6.2.1 Sala de espera virtual personalizada**

El servicio de valor añadido de salas de espera virtuales permite a las aplicaciones disponer de un mecanismo de gestión de la provisión del servicio cuando dependan de recursos limitados, proporcionando los medios para ordenar y priorizar el conjunto de usuarios que quiere disfrutar de la aplicación en cierto momento. Los usuarios en espera pueden recibir notificaciones para conocer su turno, tiempo estimado de espera y otra información relevante, así como acceder a distintos recursos multimedia proporcionados por un tercero (noticias, revistas, mini-juegos, etc.)

Un ejemplo de aplicación de este habilitador enmarcado en el ámbito del proyecto AmIVital lo encontramos en una aplicación de teleconsulta médica, donde un médico, que constituye el recurso limitado, atiende de forma remota (a través de vídeo llamada, por ejemplo) las consultas con cita previa de los ciudadanos de cierto distrito. Mediante la utilización del servicio de espera virtual, la aplicación puede gestionar las llamadas recibidas para asegurar que se mantiene el orden de atención establecido e indicar al llamante que todavía hay, por ejemplo, cuatro personas esperando ser atendidas y se estima que en 10 minutos podrán atenderle. El usuario podría entonces colgar el terminal y recibir un aviso a falta de un minuto para volver a llamar sin perder su turno.

La aplicación puede configurar la sala de espera virtual tanto para servicios sin cita previa, donde normalmente el orden de atención será de tipo FIFO (el primero en solicitar una sesión será el primero en ser atendido cuando el recurso quede libre) o servicios con cita previa, donde los usuarios tienen una hora de atención programada con anterioridad y se atenderá a los usuarios dando prioridad a la hora programada, de forma independiente de su orden de llegada a la cola de espera, emulando el mecanismo habitual de las consultas médicas presenciales con cita previa.

La información relativa a la sala de espera virtual, incluyendo los recursos a los que los miembros del grupo tengan acceso, se ofrece al usuario en forma de mensaje multimedia SIP, pudiendo contener texto enriquecido, audio, imágenes, enlaces a

páginas web y, en general, cualquier tipo de contenido multimedia que admita el terminal.

La Fig. 70 muestra la secuencia de mensajes SIP involucrada para la prestación del servicio de sala de espera virtual. El elemento “IMS Core” de la figura comprende los nodos CSCF y HSS, mientras que el nodo “AS” corresponde al servidor de aplicaciones (que ejecuta tanto la lógica de la sala como el código del habilitador LCE). El diagrama únicamente muestra la señalización intercambiada con usuarios o aplicaciones externas, por lo que no se muestran las interacciones del servidor de aplicaciones con otros elementos, entre ellos el XDMS. Las llamadas a función que aparecen pertenecen a la interfaz de servicios web incorporada a los habilitadores para facilitar su gestión, así por ejemplo, la función *createGroup* provoca la creación mediante XCAP de un grupo en el XDMS y la función *activateVirtualWaitingRoom* es el equivalente a configurar la actividad con los parámetros necesarios de la aplicación. Si recordamos, el habilitador LCE propuesto en la Tesis interactúa con otras aplicaciones tanto a través de señalización directa SIP como mediante la modificación de documentos XML en el XDMS a través del protocolo XCAP. La capa de webservices añadida para este caso práctico evita la necesidad de que las aplicaciones implementen el protocolo XCAP.

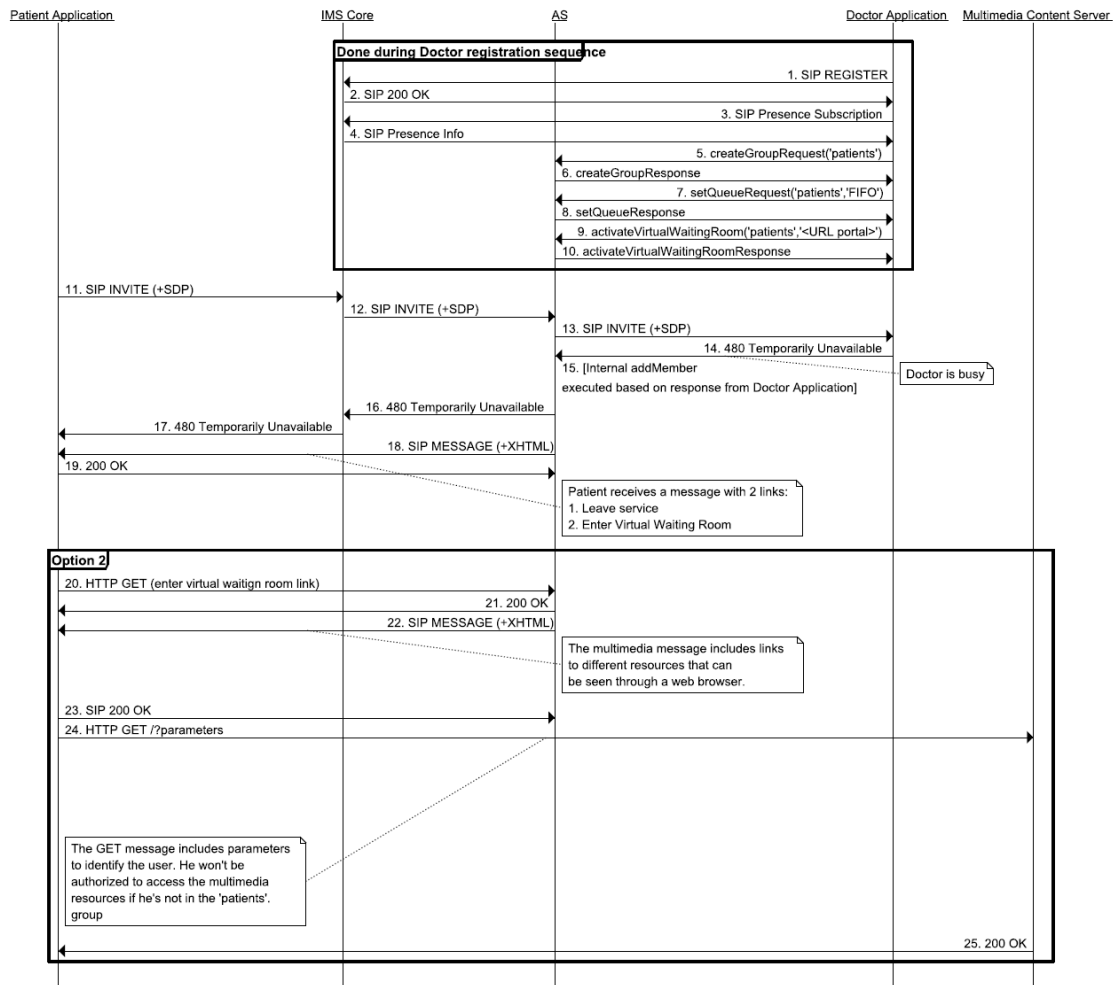


Fig. 70. Sala de espera virtual

El servicio que se ha descrito se basa en añadir a los usuarios que llamen al médico a un grupo (actividad) que tiene activado el servicio de sala de espera y enviarles un enlace al servidor de contenidos, por lo que no hay sesiones SIP involucradas. Las ventajas de usar el habilitador LCE en este escenario es la posibilidad de ofrecer servicios adicionales de forma sencilla a los usuarios que están esperando. Al estar en un grupo, LCE puede gestionar servicios adicionales como videoconferencia, compartición de imágenes, etc. para los usuarios que están en la sala de espera, sin necesidad de que la propia aplicación de sala de espera sepa gestionar esos servicios – basta con activar los servicios deseados. Así, supuesto que la sala de espera permita a los usuarios hablar entre ellos, el habilitador LCE se encargaría de establecer las sesiones SIP correspondientes de los usuarios con el nodo de mezcla de medios de IMS (MRF), de modo que si un usuario abandona la sala de espera (se le extrae del grupo porque se le va a atender), LCE cierra la sesión de audio de ese usuario con los demás de la sala. Del mismo modo, si entran nuevos usuarios, se les añadiría a la conversación.

## 6.2.2 Videoconferencia orientada a grupos

El servicio de valor añadido de videoconferencia orientada a grupos permite ofrecer a las aplicaciones la posibilidad de proporcionar a un grupo de usuarios, de forma general, y a los usuarios que estén en una sala de espera virtual, de forma particular, un servicio de conferencia de audio y vídeo entre los mismos. La adición o sustracción de un usuario en la sala de espera (es decir, en el grupo) se traduce en la invitación o despedida de dicho usuario en la videoconferencia.

La creación de conferencias programadas no pertenece al ámbito de este habilitador, para ello será necesaria una aplicación externa que dispare la llamada al habilitador en el momento oportuno para iniciar la conferencia.

El moderador de la videoconferencia crea la misma a través de la interfaz de web services (ya vimos en 6.2.1 que la misión de estas llamadas a funciones era ocultar el protocolo XCAP y la gestión de documentos XML), proporcionando el nombre del grupo donde están recogidos los usuarios que van a formar parte de la videoconferencia. A través de esa misma interfaz puede agregar o eliminar usuarios del grupo, que el LCE traduce automáticamente en añadirlos o expulsarlos de la videoconferencia. Nótese que la adición o sustracción de usuarios puede hacerse directamente mediante la interfaz XCAP del gestor de grupos.

Dado que este servicio de valor añadido está pensado para que los participantes lo consuman a través de terminales estándar con soporte para videoconferencia, el moderador puede configurar la videoconferencia para permitir o no invitaciones externas por parte de los participantes mediante mecanismos estandarizados (SIP REFER).

El habilitador también incluye la funcionalidad de notificación de eventos de conferencia, la cual permite que cada cliente se suscriba a eventos relacionados con la misma. El cliente será notificado de los cambios que se produce en el estado de la conferencia, como por ejemplo cuando un usuario entra o sale de la conferencia, con el mecanismo estándar de SIP (SUBSCRIBE/NOTIFY) de acuerdo a la RFC4575 (ver capítulos 5.6 en particular).

La Fig. 71 muestra el diagrama de secuencia para un caso en que dos usuarios (un paciente y un médico concretamente), se encuentran en una vídeollamada y se añade un tercer participante (un familiar del paciente o un médico especializado, por ejemplo). El habilitador LCE en este caso (situado en el servidor de aplicaciones AS) es

el encargado de hacer las transferencias de sesión del paciente y médico al MRF (situado en la línea IMS Core del diagrama por ser parte de la arquitectura estándar de IMS) y de invitar al tercer participante a unirse al MRF igualmente con el fin de que la mezcla de audio y vídeo se produzca y todos estén hablando entre ellos. En este escenario la capa de web services aísla a la aplicación del médico de comunicarse directamente con el XDMS, si bien esas llamadas tienen relación directa con la modificación del documento de descripción de la actividad en XDMS –que no se muestran por claridad-.

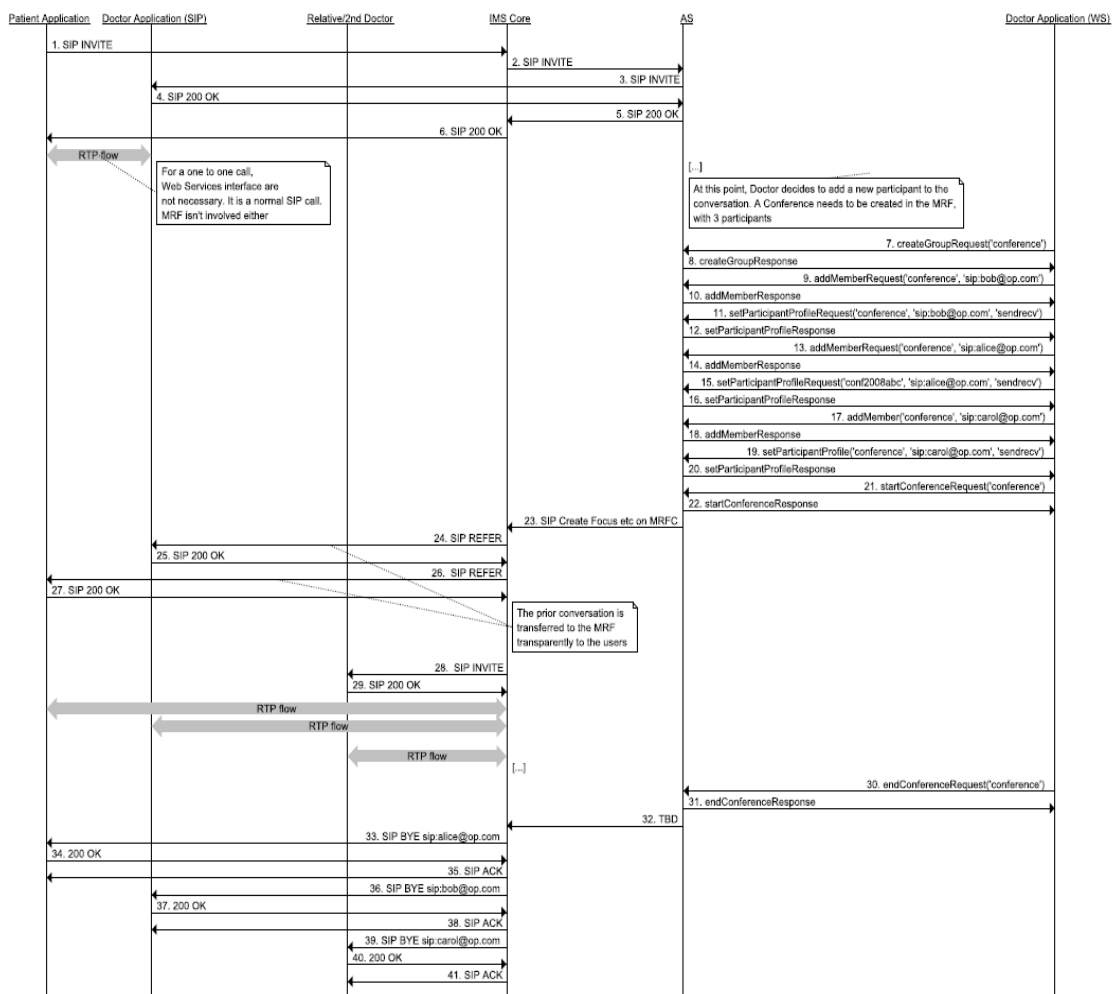


Fig. 71. Videoconferencia AmIVital

### 6.2.3 Compartición de imágenes con anotaciones (pizarra virtual)

El servicio de valor añadido de compartición de imágenes con anotaciones (o pizarra virtual) permite ofrecer a las aplicaciones la posibilidad de proporcionar a un grupo de usuarios, de forma general, y a los usuarios que estén disfrutando del servicio de sala de espera o videoconferencia, de forma particular, un servicio de pizarra interactiva y de compartición de imágenes, que permita a los miembros de un grupo compartir un conjunto de imágenes de manera sincronizada y realizar anotaciones en la misma que serán vistas por todos.

Si bien este servicio trabaja únicamente con colecciones de imágenes, un servidor externo podría convertir otros tipos de fichero, por ejemplo documentos de varias páginas o presentaciones, a un conjunto de imágenes (una imagen por página, típicamente) que, una vez proporcionados al servicio, sería equivalente a visualizar de forma compartida el documento o presentación original.

El servicio permite que el moderador de la sesión de pizarra virtual actúe sobre el grupo de participantes (añadiendo o eliminando miembros del grupo, administrando los permisos individuales de cada uno), cambie la colección de imágenes compartidas, tenga control sobre qué imagen están viendo en ese momento los usuarios y permita centrar la imagen en visualización en un punto determinado para centrar la atención de los usuarios en el mismo. Todas estas operaciones se harán mediante web services.

Los participantes, por su parte, interactúan con el servicio mediante una aplicación cliente diseñada específicamente para ese fin o mediante una interfaz web. Las formas de interacción soportadas son las de avanzar y retroceder por el conjunto de imágenes compartidas, señalar, pintar o borrar sobre la pizarra (en el caso de que el administrador les haya proporcionado permisos), y añadir imágenes a la sesión compartida subiéndolas con el navegador/aplicación a la sesión de pizarra o al repositorio de imágenes.

El LCE en este caso aporta las ventajas ya repasadas en los servicios anteriores de sala de espera virtual y videoconferencia, y adicionalmente el soporte para distintos tipos de interacción. Concretamente, al añadir a un usuario al grupo para que forme parte de la actividad, entra en acción el proceso de decisión explicado en 4.5.3, creando una sesión SIP del usuario con el servidor de pizarra virtual o proporcionando un enlace único web que permita al servidor de pizarra adaptar el contenido a la interacción web de ese usuario. La Fig. 72 muestra cómo dos usuarios participan en una actividad de pizarra virtual a través de una aplicación, mientras que en la Fig. 73 se muestra cómo se añadirían en caso de no disponer de la aplicación de pizarra instalada en sus dispositivos.

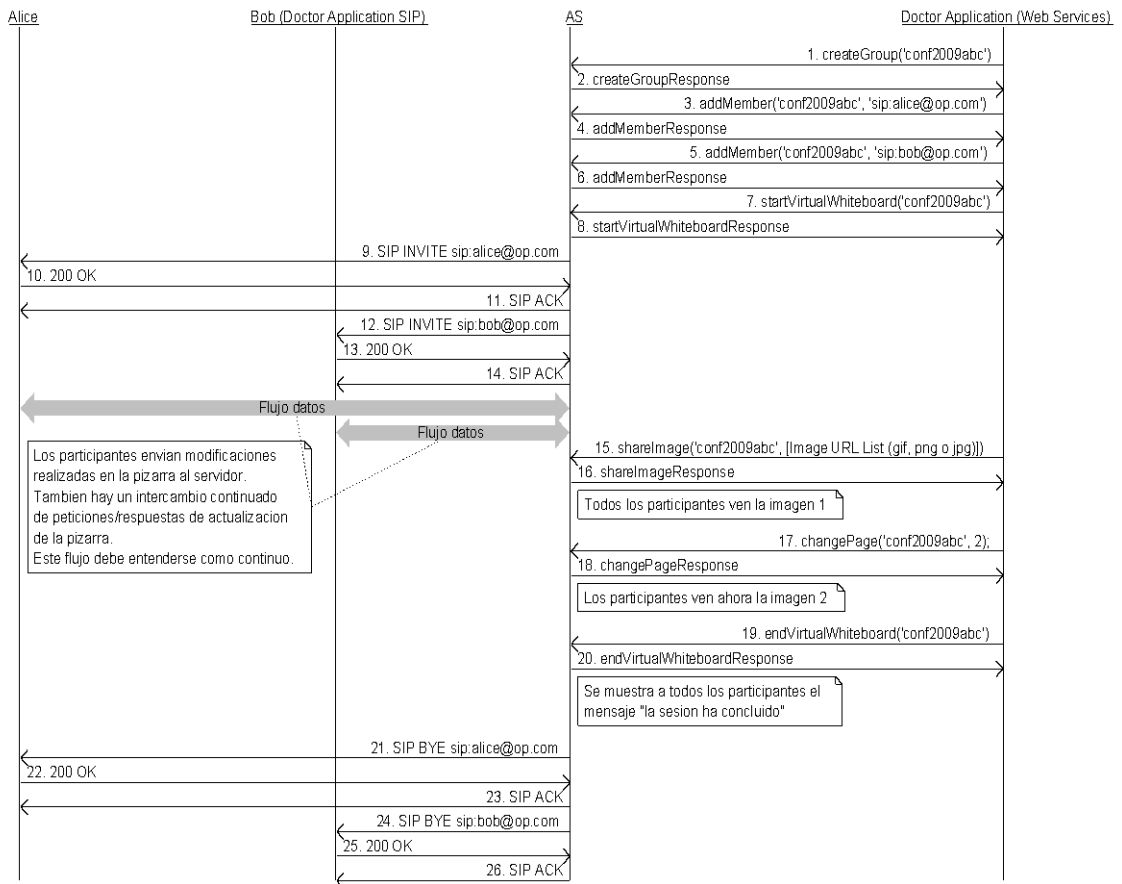


Fig. 72. Aplicación de LCE a pizarra virtual



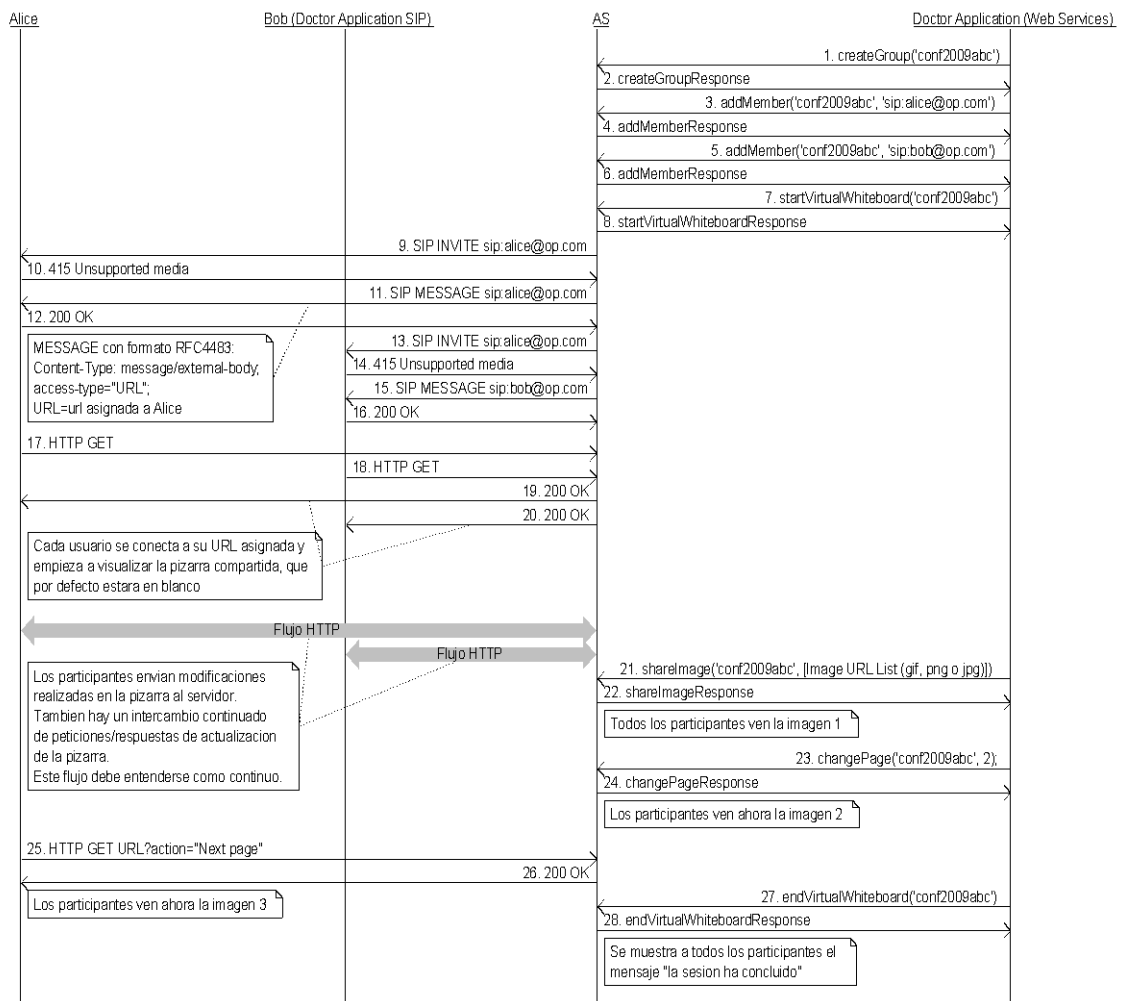


Fig. 73. Establecimiento de una sesión de pizarra virtual con RFC 4483

### 6.2.4 Conclusiones

La utilización de una arquitectura IMS y del habilitador ligero de conferencias LCE especificado en el Capítulo 4 facilita el desarrollo de los servicios orientados a aplicaciones profesionales, como los expuestos previamente, que requieren la participación de múltiples usuarios.

Además de las ventajas inherentes de ofrecer estos servicios en una arquitectura IMS (autenticación de usuarios, mezcla de audio y vídeo por el MRF, gestión de grupos), se añaden las derivadas de emplear el habilitador LCE como soporte para el establecimiento de las comunicaciones multiusuario:

Servicio	Beneficios de introducir LCE en IMS
<b>Sala de espera virtual</b>	Sincronización con el servicio de videoconferencia. La sala de espera puede configurarse para que los usuarios puedan hablar y/o verse entre ellos (activando adicionalmente el servicio de videoconferencia al grupo de usuarios). Si un usuario sale de la sala de espera, el LCE le quita de la audio/vídeo conferencia.
<b>Videoconferencia orientada a grupos</b>	<p>Establecimiento de las sesiones necesarias con el MRF de todos los usuarios añadidos al grupo.</p> <p>Sincronización con los demás servicios, especialmente con la pizarra virtual. Los usuarios en una videoconferencia pueden empezar a compartir imágenes y pintar, encargándose el LCE de establecer las sesiones necesarias. Si se añade un nuevo usuario a la videoconferencia, se añade también a la compartición de imágenes.</p>
<b>Pizarra virtual</b>	<p>Gestión automática del tipo de interacción del dispositivo por parte de LCE en función de las capacidades del mismo (establecimiento de sesión SIP, interacción web, descarga e instalación de aplicación).</p> <p>Sincronización con los demás servicios y con la configuración del grupo.</p>

Como se desprende de la tabla anterior, una de las ventajas principales de la introducción de LCE en este escenario es la orientación a grupos. LCE define las actividades como un grupo de usuarios y servicios asociados, permitiendo la composición sencilla de distintos servicios y liberando a las aplicaciones de gestionar las sesiones SIP involucradas. Adicionalmente, y de forma más notable en el caso de la pizarra virtual, el habilitador LCE selecciona el tipo de interacción más adecuado para el usuario, permitiendo que usuarios sin aplicación de pizarra instalada puedan acceder mediante una interfaz web.

En el escenario de tele-asistencia, que ha servido de validación práctica de los servicios profesionales descritos en secciones anteriores, es más apropiada la aplicación del habilitador LCE en lugar del middleware HIMMA. La razón es evitar la necesidad de instalar software específico en los dispositivos del usuario y permitir la interacción

con los servicios desde diversos tipos de dispositivos (siempre y cuando incluyan al menos un cliente SIP estándar de telefonía y navegador web). Las ventajas del middleware HIMMA se hacen más patentes en escenarios con mayor carácter descentralizado como el de los juegos en red, como vimos en 6.1.3.2, donde la aplicación puede no contar con infraestructura específica de servidor pero sí es habitual la necesidad de ejecutar software en el dispositivo.

### 6.3 Consideraciones sobre escalabilidad

Tras analizar la aplicación funcional a entornos de juegos en red y de servicios profesionales del marco de soporte a aplicaciones HIMMA propuesto en la Tesis, cabe presentar una serie de consideraciones sobre la escalabilidad de las contribuciones planteadas, tanto generales como de aplicación a los escenarios analizados. No en vano la escalabilidad es uno de los criterios de diseño de partida, expuestos en el apartado 3.2.1.

Ya hemos visto que el marco de soporte define dos nuevos elementos funcionales dentro de la arquitectura IMS: el habilitador ligero de gestión de actividades (LCE) y el middleware HIMMA, así como el comportamiento de los mismos mediante el uso de protocolos estándar. Por ello, para analizar la escalabilidad hay que considerar diversos parámetros que pueden dividirse en dos grandes grupos:

- La capacidad de almacenamiento y procesamiento requerida por los equipos que intervienen (servidores de aplicación y XDMS en el caso del LCE, dispositivos de usuario en el caso del middleware)
- El impacto sobre la red de comunicaciones (número de mensajes SIP que han de procesar los CSCFs, peticiones XCAP al XDMS, ancho de banda necesario, número de sesiones establecidas por actividad, etc.)

Respecto de las limitaciones físicas de los equipos involucrados, existen dos técnicas habituales para incrementar el número de usuarios al que poder prestar un servicio. Una alternativa es incrementar la capacidad de las máquinas de las que depende, renovando el hardware o, si son máquinas virtuales, asignando más recursos. La alternativa preferida no obstante es aumentar el número de instancias o copias del servidor de aplicación que lo sustenta, de modo que las peticiones de los usuarios se repartan entre dichas instancias y no haya que modificar las que ya estaban en servicio.

En una aplicación multiusuario, sin embargo, es posible que las peticiones SIP de usuarios que participan en la misma actividad sean tratadas por instancias distintas del servidor de aplicaciones, de ahí la importancia de que la información que haya que mantener por parte de cada instancia pueda ser compartida por el resto. En el caso de

LCE esta información se almacena en el XDMS, permitiendo al propio LCE escalar convenientemente desde el punto de vista de la implementación. En este sentido la escalabilidad de las actividades gestionadas por LCE dependen de la capacidad del XDMS. El XDMS forma parte de la infraestructura del operador IMS y soporta de forma estándar un abanico de servicios de IMS como las agendas de contactos de los usuarios o la información de presencia. A pesar de que la solución propuesta en la Tesis incrementa el tamaño de datos a gestionar en el XDMS, no debe suponer un impacto apreciable teniendo en cuenta que el XDMS ya debe estar correctamente dimensionado para soportar, por ejemplo, el servicio de presencia de todos los usuarios conectados.

El tamaño de los datos añadidos al XDMS por actividad multiusuario es comparable al de un único documento de presencia de un usuario, y además hay que tener en cuenta que no todos los usuarios conectados estarán participando en actividades multiusuario, pero sí actualizando periódicamente su información de presencia.

En efecto, en el caso de contar con el gestor LCE definido en la Tesis, las actividades dispuestas por las aplicaciones se modelan como documentos XML almacenados en el XDMS, así como el estado de las actividades (participantes actuales, participantes permitidos, políticas de acceso, etc.). Esto permite una gran flexibilidad a la hora de gestionar las actividades a cambio de apoyarse en la infraestructura de gestión de documentos XML del dominio IMS. Concretamente, supone la creación y mantenimiento de dos documentos por conjunto de usuarios participando en la actividad, el documento de descripción y el de estado. Durante la fase de selección de aplicación o de usuarios únicamente aplica el documento de descripción, creado por el organizador de la actividad. La creación del documento de actividad es independiente del número de usuarios de la misma: una petición al XDMS y su respuesta. Sí puede variar el tamaño del documento creado en función del número de usuarios y reglas de acceso que se definan, lo que implica más o menos campos y atributos dentro del XML (del orden de decenas de bytes por cada usuario, incluyendo su URI y permisos de participación). Esto hace que el documento de descripción de actividad apenas tenga 1 KB ó 2 KB típicamente.

Con estas consideraciones, el tamaño de los datos añadidos al XDMS para soportar la gestión de actividades por parte de LCE puede modelarse como sigue:

$$D = \sum_{i=1}^N (k_1 n_i + k_2)$$

siendo  $N$  el número de actividades registradas en el XDMS,  $n_i$  el número de usuarios de la actividad  $i$ ,  $k_1$  el tamaño medio de datos añadido al documento por la

incorporación de un nuevo usuario (alrededor de 200 bytes), y  $k_2$  el tamaño medio de la información de una actividad exceptuando la relacionada a cada usuario (alrededor de 2 KB).

Si definimos  $\bar{n} = \sum_{i=1}^N \frac{1}{N} n_i$  como el número medio de usuarios por actividad, tenemos

que el espacio medio necesario para gestionar todas las actividades en un dominio IMS es:

$$\bar{D} = N(k_1 \bar{n} + k_2)$$

Dado que el XDMS de un operador ya está preparado para manejar la agenda de contactos y la información de presencia actualizada periódicamente de todos los usuarios, no supone a priori un impacto apreciable en la infraestructura de soporte, máxime teniendo en cuenta que no todos los usuarios conectados van a estar haciendo uso de actividades multiusuario.

En el caso de optar por una solución basada en el middleware HIMMA no existen documentos que describan explícitamente las actividades o usuarios y es el propio middleware el encargado de detectar los usuarios y aplicaciones disponibles para participar en función de la información de presencia de los contactos. En este aspecto, el impacto sobre el XDMS es mucho menor que al usar LCE porque únicamente se añaden ciertos campos al documento de presencia, que ya existe y se actualiza periódicamente para notificar el estado de disponibilidad del usuario (independientemente de usar el middleware, se entiende). Conviene aclarar que el middleware no se apoya en ninguna otra función de la red IMS más que el servicio de presencia, buscando actividades en la información de presencia de los contactos del usuario (bien amigos de la agenda de direcciones o contactos previos con los que ha participado en alguna actividad). Si bien el impacto sobre el XDMS es menor que en el caso de LCE, una actividad basada en LCE ofrece mayor flexibilidad al posibilitar búsquedas globales de actividades e incluir criterios de selección de participantes, ya que cada actividad está descrita como un documento en el XDMS y se aprovechan las facilidades de búsqueda ofrecidas por este nodo.

Una ventaja heredada de basarse en la arquitectura IMS y con impacto positivo en la escalabilidad, tanto para el LCE como el middleware, es que el registro es único y se realiza en la infraestructura del operador, que está preparada para gestionar un gran

volumen de datos y evita que el proveedor del servicio despliegue infraestructura específica a tal fin<sup>9</sup>.

Además del impacto relacionado con la capacidad de almacenamiento y procesamiento requerida por los equipos que intervienen, hay que considerar el uso de la red impuesto por la solución, concretamente el relacionado con el incremento de consultas y actualizaciones en el nodo XDMS de la red debido al uso del servicio de presencia por parte del middleware y la gestión de documentos de LCE. Nótese que durante el transcurso de una actividad multiusuario intervienen también los procesos de establecimiento de sesiones SIP necesarias para que los clientes IMS de cada participante se comuniquen entre sí conforme a la topología definida. Sin embargo, la solución propuesta no implica un incremento del tráfico de control SIP necesario para establecer las sesiones ya que éste depende exclusivamente de la topología seleccionada por la aplicación.

El número de consultas al XDMS y suscripciones a cambios asociadas depende de las distintas fases de la aplicación. Así, la creación de la actividad únicamente implica una consulta de tipo PUT para crear su documento asociado en el XDMS. La unión de un usuario a la actividad implica, además de los pertinentes establecimientos de sesión, la suscripción por parte de ese usuario a la actividad, la actualización del documento y el envío por parte del XDMS de las notificaciones al resto de participantes.

La Tabla 7 resume el impacto que tiene la solución propuesta en el XDMS distinguiendo cada una de las fases y si las actividades se gestionan a través del LCE o del middleware. La tabla contempla el impacto en función del número de usuarios por actividad y el número total de actividades gestionadas.

---

<sup>9</sup> El mantenimiento de una infraestructura de registro de usuarios que funcione permanentemente y soporte picos de usuarios (*flash-crowds*) supone un gran coste que los pequeños y medianos proveedores de servicio no pueden asumir. Un ejemplo de la importancia de este punto lo encontramos en la plataforma de juegos Xbox Live, que en las navidades de 2007 impidió jugar en red durante un día a todos los usuarios de la consola Xbox 360 por saturación de sus servidores de registro [75], pese a que muchos de los juegos realmente implementan arquitecturas descentralizadas y los servidores específicos, en su caso, sí se encontraban operativos.

Tabla 7. Síntesis del impacto de consultas al XDMS impuestas por el marco de soporte a aplicaciones HIMMA

Acción	Consultas XCAP	Peticiónes SIP	Impacto del número de usuarios de una actividad ( $n_i$ )	Impacto del número de actividades en el dominio (N)
<b>Creación de una actividad (con LCE)</b>	Creación documento descripción (1 PUT) Creación documento de estado (1 PUT)	1 INVITE al LCE, sólo en caso de usar URI de factoría (LCE crea documento de actividad por defecto)	Incremento del tamaño del documento y, por tanto, del cuerpo de la petición	No, a excepción del espacio empleado por los documentos o mecanismos internos de indexación de XDMS
<b>Creación de una actividad (con middleware)</b>	Actualización documento presencia	1 PUBLISH	No, la crea el organizador una única vez	No
<b>Búsqueda de la actividad (con LCE)</b>	1 POST	No	No, los usuarios no son un criterio de búsqueda (se buscan actividades y condiciones de participación)	Rendimiento del motor de búsqueda de XDMS. Atenuado por uso de plantillas de búsqueda
<b>Búsqueda de la actividad (con middleware)</b>	No, búsqueda local en base a información de presencia de contactos	No	No	No
<b>Unión a una actividad o salida (por cada usuario, LCE)</b>	Modificación documento de estado (1 PUT) Modificación permisos (1 PUT)	2 SUBSCRIBE 1 REFER (LCE recibe el SUBSCRIBE y lo redirige al XDMS) 1 NOTIFY a cada participante	Lineal, número de NOTIFY proporcional a usuarios y entradas/salidas	No
<b>Unión a una actividad o salida (por cada usuario, middleware)</b>	No	1 PUBLISH 1 NOTIFY a cada contacto	No, información de presencia enviada a los contactos independientemente de participación	No, no hay documentos que definan explícitamente las actividades
<b>Pausa de la actividad (por organizador, LCE)</b>	Modificación documento de estado (1 PUT) Modificación permisos (1 PUT)		No, se pausa una vez por el organizador	No

Una vez detallados los aspectos de escalabilidad específicos del LCE y el middleware HIMMA, conviene destacar que las aplicaciones finales que hagan uso de éstas u otras soluciones también tienen límites de escalabilidad intrínsecos que dependen en gran medida de factores tales como la topología de red elegida, disponibilidad de servidores dedicados, capacidad de procesamiento exigida a los equipos por la lógica de la aplicación, o incluso criterios de usabilidad, entre otros. En efecto, el soporte de topologías descentralizadas, bien en malla o con un dispositivo con el rol de supernodo, impone limitaciones de escalabilidad ligadas a estas topologías. Por un lado, el crecimiento exponencial del número de sesiones en función del número de usuarios de una actividad si es una malla. Por otro, la capacidad de los dispositivos que desempeñan la función de supernodo, en su caso.

Un ejemplo del impacto en la escalabilidad de aplicaciones HIMMA debida a las propias aplicaciones lo encontramos en el caso de las consolas de videojuegos multijugador en red, donde es habitual que los juegos con topologías descentralizadas no admitan más de 8 jugadores por limitación de potencia de los dispositivos (consolas) de los usuarios, que tienen que recibir las acciones de los distintos jugadores, calcular el estado del juego en cada momento y enviar las actualizaciones de estado a los demás. Por ello, juegos de acción como *Frontlines* de Xbox 360 han conseguido el record, en esa consola, de 50 jugadores en red simultáneos en cada partida [75] gracias al despliegue de casi un centenar de servidores a nivel mundial.

Otro ejemplo relevante y con repercusión mediática son los juegos de rol multijugador masivos (MMORPG). De gran penetración en el mercado, se anuncian como juegos donde millones de usuarios participan en un mundo virtual. En la práctica, juegos como *World of Warcraft*, con la mayor cuota de mercado en juegos de este tipo, obligan al jugador (que ejecuta un software específico en su ordenador) a escoger un servidor al que conectarse siempre, de tal forma que sólo puede interactuar<sup>10</sup> con el mundo virtual que modela ese servidor y con los jugadores que estén conectados al mismo. Cada servidor o mundo alberga miles de jugadores registrados, sus perfiles y el estado del mundo en todo momento, precisando gran cantidad de memoria y capacidad de proceso para gestionar las acciones de cada jugador y mantener el estado del mundo virtual. La escalabilidad en las propuestas

---

<sup>10</sup> La interacción entre el conjunto de usuarios en el caso de los juegos multijugador masivos MMORPG no es tan directa como en un juego de acción (de disparos o de carreras, por ejemplo). Esto se debe a la separación entre la fase de “exploración” del mundo virtual (donde efectivamente hay cientos de jugadores y pueden ver sus posiciones, caminar o enviarse mensajes), y las fases de “batalla” (fases de acción con grupos de jugadores reducidos).



comerciales de este tipo de juegos se basa en ampliar el número de mundos y/o en ampliar la capacidad individual de cada uno, aumentando la capacidad de proceso de su servidor asociado. Tanto es así que el citado juego *World of Warcraft* hace uso de supercomputadores para el despliegue y gestión de sus mundos virtuales [76], con una capacidad de proceso comparable a la de los grandes motores de búsqueda y superior a la que pueda disponer un operador de IMS en el mercado.

Tras este análisis sobre la escalabilidad se desprenden un conjunto de características que, cualitativamente, indican que el marco genérico de soporte a aplicaciones HIMMA propuesto ejerce un impacto bajo sobre la infraestructura IMS y aprovecha convenientemente el potencial de los habilitadores de IMS como la gestión de presencia o la gestión de documentos para aportar una capa de control de actividades HIMMA.



## 7. Conclusiones y trabajos futuros

La progresiva generalización del acceso a redes de datos y el aumento de su velocidad, tanto a través de accesos fijos como móviles, está potenciando la oferta de aplicaciones multimedia interactivas que incentivan la participación de sus usuarios. Adicionalmente, la comunicación e interacción entre los propios usuarios se considera un valor añadido para las aplicaciones pues genera *efecto red*<sup>11</sup> e incentiva a otros usuarios a unirse, concepto explotado en diversas aplicaciones como las redes sociales o los juegos multijugador en red donde gran parte de su valor reside en la alta interacción entre sus usuarios. A este tipo de aplicaciones que reúnen las características de multimedia, multiusuario y de alta interactividad, las hemos denominado por su acrónimo HIMMA (*Highly Interactive Multimedia Multiuser Applications*).

Actualmente, las aplicaciones multimedia multiusuario de alta interactividad (HIMMA) están ofrecidas principalmente a través de Internet (para el público en general, como pudieran ser los juegos en red) o mediante redes IP privadas bien físicas o virtuales (en ámbitos corporativos, típicamente con aplicaciones de trabajo colaborativo). En cualquiera de estos casos, la red únicamente ofrece servicio de transporte de datos IP, por lo que cada aplicación debe implementar el resto de funciones habituales de este tipo de aplicaciones: gestión de perfiles de usuario, autenticación, tarificación, comunicación multimedia, mensajería, gestión de la movilidad, etc. La incorporación de estas funciones precisa la creación, el despliegue y mantenimiento de infraestructuras dedicadas por parte del proveedor de cada aplicación, lo que también lleva a que las soluciones no sean estándar y se dificulte la interoperabilidad entre los usuarios.

---

<sup>11</sup> El efecto red se produce cuando cada nuevo usuario añade valor al producto por el hecho de unirse al conjunto de usuarios.

El Subsistema IP Multimedia (IMS) [4], estandarizado por el 3GPP como la arquitectura Todo-IP para provisión de servicios multimedia en UMTS a partir de la *Release 5*<sup>12</sup>, ya comparte la visión de la red como algo más que un medio de transporte, dotándola de una serie de servicios básicos, denominados habilitadores o *enablers*, que las aplicaciones pueden utilizar, como la gestión de información de presencia, mensajería, gestión de grupos o gestión de sesiones multimedia, redundando en una menor infraestructura específica necesaria por parte de los desarrolladores y proveedores de aplicaciones. Además, otra ventaja de IMS derivada de ser una arquitectura desplegada en un entorno de operador es que permite optimizar los recursos de red, así como incluir aspectos de calidad de servicio (QoS) no presentes en Internet. Por ello, IMS tiene el potencial de reducir el coste de las aplicaciones HIMMA al tiempo que permite ofrecer a los usuarios una mejor experiencia tanto en funcionalidad como en prestaciones y, a los desarrolladores, una mayor facilidad para crear aplicaciones avanzadas.

El objetivo global de la Tesis es **proponer mecanismos y reglas de diseño que contribuyan a mejorar el soporte estándar que las redes IP ofrecen a las aplicaciones HIMMA, en un entorno convergente basado en la arquitectura IMS con habilitadores originales para el soporte de aplicaciones HIMMA definidos en la Tesis**. Para conseguir este objetivo, en primer lugar se ha profundizado en el análisis del estado de la técnica en lo relativo a las aplicaciones HIMMA y su soporte en redes IP. Posteriormente, se ha definido un marco general para la provisión de aplicaciones HIMMA en entornos IMS, incluyendo la propuesta de módulos habilitadores específicos para dichas aplicaciones, y se ha mostrado la aplicación de las propuestas de la Tesis en casos de uso de aplicaciones HIMMA profesionales y de entretenimiento.

## 7.1 Contribuciones

La primera contribución de esta Tesis es la **definición de un marco general de provisión de aplicaciones HIMMA basado en la arquitectura IMS**. El marco definido responde a la motivación de la Tesis de extender la arquitectura IMS, manteniendo el uso de tecnologías estándar, para ofrecer un mejor soporte a aplicaciones HIMMA reutilizando la funcionalidad proporcionada por IMS.

Concretamente, en el Capítulo 3 se han presentado los principios de diseño que han guiado la definición del marco de soporte a aplicaciones HIMMA, sus características

---

<sup>12</sup> Actualmente se comercializa el servicio de telefonía UMTS (3G) basado en la Release 99, anterior a la Release 5 y, por tanto, a la introducción de IMS. Tanto las llamadas de voz como las de vídeo se cursan a través de circuitos, sin un entorno Todo-IP.

generales y los enfoques seguidos a la hora de extender la arquitectura IMS. Por un lado, extendiendo la funcionalidad del lado servidor mediante el concepto de *habilitador* y, por otro, extendiendo la funcionalidad del lado de cliente mediante un *middleware*.

El **habilitador de gestión de conferencias ligero (LCE)** constituye una segunda contribución relevante. El LCE se encarga de gestionar actividades multiusuario y mantener su estado aprovechando las capacidades del habilitador de gestión de documentos XML (XDM) definido por OMA y adoptado por 3GPP para el IMS. Entre sus ventajas cabe destacar la independencia del tipo de medio requerido por la aplicación (los servicios de conferencias estándar de IMS como PoC o los basados en MRF están restringidos a audio, vídeo o texto), el soporte de múltiples topologías (es apto para aplicaciones HIMMA que requieran arquitecturas descentralizadas por no disponer de servidor central) y la posibilidad de definir actividades multiusuario como composición de servicios, cubriendo las necesidades de aplicaciones que requieran de distintos servidores especializados para ofrecer la funcionalidad completa (p.ej. un servicio de pizarra virtual donde los usuarios puedan hablar entre ellos, realizando la mezcla del audio el MRF y la gestión del estado de la pizarra un servidor al efecto). Adicionalmente, el LCE aporta funciones avanzadas como la pausa y la reanudación de actividades o la compatibilidad con distintos tipos de interacción por parte de la aplicación (cliente SIP de audio y vídeo, cliente SIP de audio y vídeo con navegador web y aplicación cliente específica), por lo que supone una ventaja para entornos con múltiples dispositivos de acceso a las actividades.

Como tercera contribución, el **middleware HIMMA (HIMMA-MW)** se encarga de proporcionar un conjunto de funciones que facilitan el desarrollo de aplicaciones HIMMA en entornos IMS sin necesidad de infraestructura adicional a la provista por IMS, haciendo un uso innovador del servicio de presencia para el descubrimiento de actividades multiusuario. Sus principales aspectos innovadores residen en el uso intensivo del mecanismo de establecimiento de sesiones y las funciones de gestión de presencia y grupos, alejándose del modelo de conferencias habitual que delega en servidores de forma centralizada, como el propio modelo de conferencias de IMS. El middleware proporciona una interfaz Java de alto nivel donde los conceptos principales son Actividades, Grupos y Participantes. Los participantes pueden asociarse en grupos y forman parte de las actividades. Cuando un participante se incorpora a una actividad, el middleware establece las sesiones SIP subyacentes oportunas en función de la topología de la actividad (centralizada, P2P en malla o P2P jerárquico), liberando al desarrollador de la gestión de sesiones SIP y detalles del protocolo. De igual forma, el middleware gestiona la información de presencia automáticamente para publicar el estado de participación en las distintas actividades y

permitir la búsqueda de actividades a través de los contactos del usuario o de las funciones de búsqueda que proporcione la red IMS a través del nodo XDMS

En los Capítulos 4 y 5 se han definido ambos elementos, LCE e HIMMA-MW, detallando su estructura interna, interfaces ofrecidos, funcionamiento en el seno de la arquitectura IMS y ejemplos de uso, así como las implementaciones y las pruebas prácticas realizadas.

Aunque las propuestas de la Tesis se han planteado con suficiente generalidad para que puedan dar soporte a todo tipo de aplicaciones HIMMA, en el Capítulo 6 se ha hecho un **análisis detallado de la aplicación tanto del habilitador LCE como del middleware HIMMA-MW a dos casos relevantes, incluyendo ejemplos de implementación**: por un lado, un servicio de juegos en red como ejemplo representativo de aplicación HIMMA de entretenimiento y, por otro, servicios de asistencia sanitaria, como ejemplo de aplicación HIMMA de tipo profesional. El análisis realizado y los servicios implementados muestran las ventajas que aportan las contribuciones de la Tesis en ambos escenarios.

En el escenario de juegos en red se han seleccionado una serie de casos de uso relevantes a través del análisis de las fases sucesivas de la experiencia de un jugador durante la interacción con una plataforma de juego en red. A efectos de evaluar críticamente las ventajas del acercamiento propuesto en la Tesis a este tipo de escenario, se ha realizado una síntesis original de 35 funciones o servicios prestados por el conjunto de plataformas de juego en red más relevantes del mercado. Las fases identificadas, en las que se enmarcan las funciones identificadas, incluyen la fase de elección del dispositivo de usuario, el registro en la plataforma, la selección de la aplicación/juego y de los jugadores, la participación en el propio juego y la pausa/reanudación. En cada fase se discuten las ventajas del marco genérico de soporte a aplicaciones HIMMA presentado en la Tesis, así como del habilitador LCE y el middleware HIMMA. Algunas ventajas destacables son la utilización de tecnologías estándar, el mantenimiento de una misma identidad entre todos los juegos independientemente del proveedor, la posibilidad de crear juegos en red con servicios de valor añadido sin infraestructura de servidor específica (gracias a las funciones de comunidad de LCE y al middleware HIMMA), y el establecimiento automático de las sesiones SIP requeridas por el juego en función de la topología de red requerida.

En el escenario de asistencia sanitaria se estudia la aplicación práctica de los conceptos presentados al proyecto CENIT AmIVital. Este análisis crítico ha resultado de especial interés para validar la aplicabilidad de los conceptos e ideas presentados en la Tesis a través de tres servicios de carácter profesional (sala de espera virtual, videoconferencia orientada a grupos y pizarra virtual) enmarcados en un caso de uso

de asistencia sanitaria a distancia. En estos servicios se ponen de manifiesto tanto las ventajas inherentes de la arquitectura IMS (autenticación de usuarios o mezcla de audio y vídeo por el MRF) como de la orientación a grupos del habilitador LCE y el soporte para distintos tipos de interacción. Al emplear el servidor de gestión de documentos XML como almacén de documentos de descripción de actividades, el habilitador LCE facilita la combinación de servicios y su uso de manera sincronizada, permitiendo, por ejemplo, que los usuarios que están en la sala de espera virtual puedan hablar entre ellos y, si se añaden o eliminan usuarios de la sala (modelada como un grupo), el LCE automáticamente lo traduce en incorporar al nuevo usuario o eliminarlo del servicio de audio/vídeo conferencia. En el caso de la pizarra, el LCE sigue mostrando las ventajas de la combinación de servicios (el desarrollador de la pizarra únicamente se encarga de la lógica de la pizarra independientemente de que haya servicios adicionales activados para sus participantes) y además habilita la interacción a través de una aplicación específica de pizarra o mediante web para los dispositivos que no tengan aplicación específica instalada, gracias a su soporte multimodo.

Dado que la solución de soporte a aplicaciones HIMMA propuesta se basa en aprovechar los habilitadores de una red IMS como el gestor de presencia o la gestión de documentos XML, tras el análisis de los dos casos de uso anteriores, el Capítulo 6 se cierra con un **apartado dedicado a las consideraciones sobre escalabilidad**, analizando el impacto sobre los equipos y la red IMS que impone el uso tanto del habilitador LCE como del middleware. De este análisis cualitativo se desprende que el empleo de la solución propuesta para dar soporte a aplicaciones HIMMA, si bien incrementa el tamaño de los datos a gestionar en el XDMS así como el número de consultas a procesar por el mismo, no debería causar un impacto apreciable en una red que ya esté dimensionada adecuadamente para ofrecer a sus usuarios servicios básicos como presencia, agenda de contactos remota, etc.

En definitiva, el diseño de aplicaciones innovadoras y la demostración de las ventajas que proporciona IMS a los actores involucrados, son probablemente los principales escollos que tiene que superar IMS para asentarse en el mercado. Las ideas propuestas en la Tesis para el soporte de aplicaciones HIMMA en una arquitectura IMS ayudan a promover el desarrollo de nuevas aplicaciones multimedia multiusuario altamente interactivas, componentes que se consideran clave para el éxito en un futuro cercano, como están empezando a demostrar las redes sociales, los juegos en red, o aplicaciones colaborativas en los ámbitos de la enseñanza, la asistencia sanitaria o el trabajo.

Cabe destacar que las contribuciones de la Tesis son igualmente válidas en escenarios donde no se disponga de una arquitectura IMS completa. En efecto, el marco genérico de soporte a aplicaciones HIMMA propuesto en la Tesis puede desplegarse en cualquier red IP instalando los componentes necesarios de la arquitectura como el servidor XDMS e intermediarios SIP. Estos componentes, independientemente de que estén desplegados en el operador o no, proporcionan interfaces estándar y constituyen una ventaja frente a soluciones propietarias de gestión de actividades HIMMA.

Tanto los resultados obtenidos en estudios preliminares al comienzo de la elaboración de la Tesis como sus contribuciones principales resumidas en este apartado, incluyendo su aplicación a los casos prácticos de estudio, se han publicado en los siguientes trabajos: [62], [77], [78], [79] y [80].

Adicionalmente, los detalles de implementación del LCE relacionados con la gestión de grupos, así como el middleware HIMMA y su código fuente, se presentan en los trabajos [58], [59] y [61], dirigidos por el autor.

## 7.2 Trabajos futuros

A continuación se presentan algunos aspectos relacionados con la continuidad del trabajo de la Tesis y su aplicación práctica, así como posibles líneas de investigación futura.

### 7.2.1 Contribución a estándares

El habilitador LCE de gestión de actividades multiusuario hace uso de dos tipos de documento XML almacenados en el XDMS con el fin de implementar su funcionalidad e interactuar con otras aplicaciones sin necesidad de definir nuevos protocolos. Concretamente, el modelo de descripción de actividades (una extensión del modelo OMA *Shared Group*) y el modelo de estado de aplicaciones (una extensión del modelo de estado de conferencias de SIP) utilizados en el habilitador LCE son susceptibles de estandarización.

El registro de estos modelos de documento XML (denominados *application usages* en la terminología del gestor de documentos XDMS) se realiza a través del procedimiento estándar de publicación de RFCs, siguiendo las indicaciones de la RFC 4825.



### 7.2.2 Adaptación a nuevos estándares de IMS

La validación práctica de las ideas presentadas en el habilitador LCE se ha realizado en una maqueta de IMS con XDMS 1.0 y soporte para *Shared List* (grupos de usuarios), como se presentó en el apartado 4.6. Parte de la funcionalidad del LCE requiere sin embargo un servidor XDM 2.0 con soporte para *Shared Group*, no disponible actualmente por tratarse de una especificación muy reciente.

La implementación del middleware HIMMA está realizada utilizando la plataforma Ericsson ICP (*IMS Client Platform*), que cumple la especificación pre-JSR 281, conforme se indica en el apartado 5.5. La versión final de la JSR 281 incluyó cambios considerables respecto de los borradores, incluyendo la eliminación del soporte para el servicio de presencia y grupos, imprescindible para el middleware HIMMA. Estas funciones se movieron a la JSR 325, aprobada en Enero de 2010.

El trabajo futuro en esta línea incluye por tanto identificar implementaciones de XDM 2.0 con soporte para *Shared Group*, de cara a validar y adaptar la implementación actual del LCE, así como entornos con las API Java JSR 281 y JSR 325 para adaptar el código del middleware a estas interfaces.

### 7.2.3 Aplicación a otros entornos

Además de los proyectos indicados en el apartado 1.5, que están relacionados con la Tesis y han influido en su desarrollo directa o indirectamente, dentro del grupo de investigación donde se ha desarrollado la Tesis se está trabajando en otros proyectos relacionados con SIP/IMS donde se plantea la aplicación de las contribuciones aquí expuestas.

Uno de estos proyectos es el proyecto europeo LiLa (*Library of Labs*), que persigue la creación de una red de laboratorios virtuales para la realización de experimentos de forma remota. Se plantea la participación en la gestión de múltiples laboratorios utilizando la arquitectura IMS y el LCE para el establecimiento de sesiones con los laboratorios virtuales y control del acceso.

El escenario de partida, mostrado en la Fig. 74 y obtenido de [81], supone la conexión con el servidor remoto (que ejecuta los experimentos en un laboratorio o los simula) a través de applets Java que muestran la interfaz de usuario del experimento. La integración de la arquitectura IMS en este escenario conlleva una serie de ventajas:

- Autenticación única de los usuarios. Las sesiones SIP de los usuarios con el servidor de experimentos estarían autenticadas por los intermediarios

CSCFs de cada dominio, que disponen de mecanismos para la creación de círculos de confianza.

- Control de sesiones. La orientación a sesiones de las comunicaciones con el servidor de experimentos permite forzar políticas de uso de forma transparente a los servidores. Por ejemplo, para la reserva de turnos de prácticas remotas, el LCE puede extenderse con una condición temporal tal que un conjunto de usuarios tenga acceso al laboratorio de experimentos sólo durante un periodo de tiempo de una hora y media a partir de las 17:00. A las 18:30, LCE finalizaría las sesiones SIP automáticamente y el servidor quedaría libre, sin necesidad de implementar lógica de gestión de turnos en ningún servidor de experimentos.
- Comunicaciones multimedia. Las facilidades de comunicación provistas por SIP, unidas a habilitadores como la gestión de agendas de contacto o presencia, permiten aportar a los experimentos un marco de comunicaciones integral que favorezca la comunicación entre alumnos y docentes en caso necesario (p.ej. una duda en una práctica remota podría ser atendida por un profesor cuyo estado de presencia en ese momento fuera disponible).

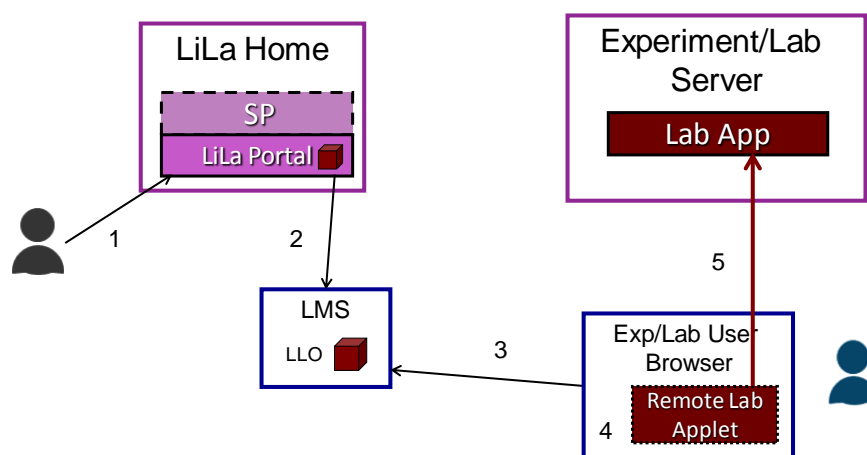


Fig. 74. Esquema actual de acceso a experimentos remotos en LiLa

Todas estas áreas servirán para evaluar la aplicación de las contribuciones de la Tesis en nuevos escenarios de servicio y se espera que sirva para evaluar aspectos de escalabilidad en entornos reales con gran volumen de usuarios.

### 7.2.4 Composición de servicios IMS / Web

Los servicios ofrecidos en la Web pueden beneficiarse de las capacidades de IMS tales como la autenticación de usuarios, el uso de habilitadores como el servicio de presencia, agenda de contactos o suscripción a eventos, y los servicios básicos de comunicación multimedia como las llamadas de voz o vídeo sobre IP. Por otro lado, las aplicaciones de IMS pueden enriquecerse gracias a la variedad de servicios y contenidos disponibles en la Web.

La consecución de este objetivo conlleva la definición de interfaces (APIs) estándares y abiertas para el acceso a los servicios ofrecidos por la red IMS desde la Web, e igualmente para el acceso a los servicios Web por los usuarios y aplicaciones de IMS. Actualmente, diversos operadores empiezan a ofrecer APIs sencillas basadas en HTTP REST para acceder a sus servicios desde aplicaciones web. Sin embargo, cada operador ofrece su API y los desarrolladores de aplicaciones se encuentran con una situación de fragmentación donde según el operador del usuario hay que usar un API u otra para acceder a la misma funcionalidad, teniendo que multiplicar los esfuerzos de implementación. La GSM Association<sup>13</sup> ha propuesto un conjunto de APIs en vías de estandarización por Open Mobile Alliance (OMA), denominado *OneAPI* [82], para el acceso a servicios como el envío de mensajes SMS y MMS, localización o facturación independientemente del operador.

La variedad de servicios que ofrece IMS precisa la definición de mecanismos e interfaces que permitan de forma sencilla la composición de aplicaciones basadas en funciones ofrecidas a través de la Web y funciones ofrecidas en las redes IMS. El marco de soporte a aplicaciones multiusuario definido en la Tesis puede ser un punto de partida, modelando las aplicaciones compuestas como documentos XML dentro del XDMS y extendiendo la funcionalidad de la red IMS mediante habilitadores que gestionen la interacción con servicios Web (p.ej. autenticándose de forma automática en los mismos o programando suscripciones SIP para notificar de eventos en esos servicios si necesidad de hacer consultas periódicas).

---

<sup>13</sup> Asociación formada por más de 800 operadores móviles y 200 fabricantes de dispositivos del mundo.  
<http://www.gsmworld.com/about-us/index.htm>

### 7.2.5 Gestión de redes vehiculares en redes IMS-LTE

Un área actualmente en expansión en el campo de la investigación es el vehículo conectado. Los vehículos actualmente incorporan numerosos adelantos tecnológicos para mejorar la seguridad y el confort del conductor y pasajeros, pero carecen de conectividad con otros vehículos o la red. La introducción de comunicaciones entre vehículos (V2V) o entre vehículos e infraestructura de red (V2I) pueden mejorar los transportes, facilitando mecanismos de gestión de la circulación, aviso de accidentes cercanos o establecimiento de comunicaciones con centros de emergencia de forma autónoma entre otros [83]. También pueden hacer más agradable el viaje a los pasajeros con servicios de entretenimiento, juegos, guías turísticas, etc.

Las comunicaciones en estos casos imponen requisitos de movilidad, ancho de banda y disponibilidad, por lo que se contemplan tanto comunicaciones ad-hoc entre vehículos como comunicaciones a través de una infraestructura estable con acceso inalámbrico. El proyecto LTE (*Long Term Evolution*<sup>14</sup>) dirigido por 3GPP y recogido en la *Release 8* se centra en la mejora de la red de acceso UTRAN de UMTS y el núcleo de red para optimizar el acceso radio de las redes móviles y posibilitar servicios de banda ancha en movilidad. A las mejoras en la parte de acceso se une la incorporación del núcleo de red IMS para la gestión de servicios multimedia Todo IP.

La tecnología de acceso radio definida en LTE permite conexiones de banda ancha estables a alta velocidad (superando los 120 km/h), habilitando nuevos servicios en movilidad, mientras que la arquitectura IMS subyacente proporciona un conjunto de servicios de valor añadido como los vistos a lo largo de esta memoria. La combinación IMS-LTE resulta muy interesante para su aplicación a entornos de redes vehiculares gracias a la disponibilidad de servicios como la autenticación de usuarios, presencia con geolocalización, gestión de grupos o establecimiento de sesiones multimedia, que permitirían crear servicios y nuevos habilitadores que extraigan información de los vehículos conectados y la procesen para aportar un valor añadido a los mismos.

Los puntos abiertos de esta línea de trabajo incluyen la identificación de nuevos habilitadores de servicio de utilidad para la gestión de redes vehiculares que aprovechen al máximo la infraestructura disponible en IMS, así como el estudio del impacto y alternativas de optimización de la señalización y mecanismos de calidad de servicio derivados de la alta movilidad y potencial concentración de vehículos en un mismo entorno geográfico.

---

<sup>14</sup> <http://www.3gpp.org/LTE>

# Referencias

- [1] Michael Rees, "Supporting Highly-Interactive Web Applications with Server-Side Code," in *The Eleventh Australasian World Wide Web Conference*, Royal Pines Resort, Gold Coast, 2005.
- [2] Tzu-Chien Liu, "Applying wireless technologies to build a highly interactive learning environment," in *Wireless and Mobile Technologies in Education, 2002. Proceedings. IEEE International Workshop*, 29-30 Aug. 2002, pp. 63 - 70.
- [3] Darren Waters. (2007, octubre) EA wants 'open gaming platform'. [Online]. <http://news.bbc.co.uk/1/hi/technology/7052420.stm>
- [4] Gonzalo Camarillo and Miguel-Angel Garcia-Martin, *The 3G IP Multimedia Subsystem (IMS) : Merging the Internet and the Cellular Worlds.*: John Wiley & Sons, August 2004.
- [5] ETSI. (2003) TISPAN - Defining the Next Generation Network. [Online]. <http://www.etsi.org/tispan/>
- [6] Graham Finnie, "IMS Deployments Updated: Promise & Challenges," *Heavy Reading*, vol. 5, no. 12, Julio 2007.
- [7] One Voice. (2009, November) Global Telecom Companies Announce a Standards Based Solution for Voice and SMS Services over LTE.
- [8] James Brightman. (2008, abril) Video Games Explode: Global Revenues Now on Par with Box Office. [Online]. <http://www.gamedaily.com/articles/news/video-games-explode-global-revenues-now-on-par-with-box-office/?biz=1>

- 
- [9] Jonathan Fildes. (2006, septiembre) Videogames shine in high-definition era. [Online]. <http://news.bbc.co.uk/2/hi/technology/5377052.stm>
- [10] Blizzard Entertainment. Battle.net. [Online]. <http://www.battle.net>
- [11] Microsoft. Xbox LIVE Memberships. [Online]. <http://www.xbox.com/en-US/live/memberships/>
- [12] Tor Thorsen. (2009, January 6th) 28 million Xbox 360s sold, 17 million on Xbox Live. [Online]. <http://www.gamespot.com/news/6202733.html>
- [13] ComScore Inc. (2007, julio) Worldwide Online Gaming Community Reaches 217 Million People. [Online]. <http://www.comscore.com/press/release.asp?press=1521>
- [14] comScore. (2009, enero) Game On! Online Gaming Surges as Gamers Seek Out Free Alternatives in Tight Economy. [Online]. <http://www.comscore.com/press/release.asp?press=2706>
- [15] Open Mobile Alliance, "White Paper on Mobile Gaming Evolution - OMA-WP-MGPC-20080610-A," junio 10, 2008. [Online]. [http://www.openmobilealliance.org/Technical/release\\_program/docs/CopyrightClick.aspx?pck=MGPC&file=V1\\_0-20080610-A/OMA-WP-MGPC-20080610-A.pdf](http://www.openmobilealliance.org/Technical/release_program/docs/CopyrightClick.aspx?pck=MGPC&file=V1_0-20080610-A/OMA-WP-MGPC-20080610-A.pdf)
- [16] Mike Snider, "Video game sales hit record despite economic downturn," *USA Today*, January 2009.
- [17] comScore. (2009, enero) Smartphones Provide Extra Mana for Mobile Games Industry as Audience for Downloaded Games Grows 17 Percent. [Online]. [http://www.comscore.com/Press\\_Events/Press\\_Releases/2009/1/Mobile\\_Gaming\\_Grows](http://www.comscore.com/Press_Events/Press_Releases/2009/1/Mobile_Gaming_Grows)
- [18] MultiMedia Intelligence, "300 Million Multimedia Phones in '08 Says MultiMedia Intelligence," 2007.
- [19] Mehdi Cherifia (AFP). (2010, marzo) Los videojuegos gratis en Internet crecen. y dan cada vez más dinero. [Online]. <http://www.elmundo.es/elmundo/2010/03/30/navegante/1269935453.html>

- [20] 3GPP. 3rd Generation Partnership Project. [Online]. <http://www.3gpp.org/>
- [21] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, "SIP: Session Initiation Protocol," Internet Engineering Task Force, RFC 2543, March 1999.
- [22] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," Internet Engineering Task Force, RFC 1889, Jan. 1996.
- [23] 3GPP, "Conferencing using the IP Multimedia (IM) Core Network (CN) subsystem; Stage 3 (Release 7) - TS 24.147 V7.10.0 (2008-12),".
- [24] J. Rosenberg, "A Framework for Conferencing with the Session Initiation Protocol (SIP)," February 2006.
- [25] M. Barnes, C. Boulton, and O. Levin, "A Framework for Centralized Conferencing (draft-ietf-xcon-framework-11)," 11 de abril de 2008.
- [26] S P. Romano et al., "Requirements for Distributed Conferencing (draft-romano-dcon-requirements-02.txt)," 23 de enero de 2008.
- [27] A. Buono, S. Loreto, L. Miniero, and S.P. Romano, "A distributed IMS enabled conferencing architecture on top of a standard centralized conferencing framework [IP Multimedia Systems (IMS) Infrastructure and Services]," *IEEE Communications Magazine*, vol. 45, no. 3, pp. 152-159, marzo 2007.
- [28] J. Rosenberg, "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)," May 2007.
- [29] A.B. Roach, "Session Initiation Protocol (SIP)-Specific Event Notification," June 2002.
- [30] Scott Boag et al., "XQuery 1.0: An XML Query Language," January 2007.
- [31] OMA. (2009, August) Shared Group XDM Specification. [Online]. [http://www.openmobilealliance.org/Technical/release\\_program/docs/CopyrightClick.aspx?pck=XDM&file=V2\\_0-20090810-C/OMA-TS-XDM\\_Shared\\_Group-V1\\_0-20090810-C.pdf](http://www.openmobilealliance.org/Technical/release_program/docs/CopyrightClick.aspx?pck=XDM&file=V2_0-20090810-C/OMA-TS-XDM_Shared_Group-V1_0-20090810-C.pdf)

- 
- [32] Piotr Kessler and Stefan Svenberg. (2009, April) JSR 281: IMS Services API. [Online]. <http://www.jcp.org/en/jsr/detail?id=281>
- [33] Martin Gunnang and Niclas Palm. (2010, January) JSR 325: IMS Communication Enablers (ICE). [Online]. <http://www.jcp.org/en/jsr/detail?id=325>
- [34] Yannis Cosmadopoulos and Mihir Kulkarni. (2008, August) JSR 289: SIP Servlet v1.1. [Online]. <http://www.jcp.org/en/jsr/detail?id=289>
- [35] OMA Game Services WG, "Gaming Platform Version 1.0," June 2003.
- [36] OMA Game Services WG, "Game Services Client/Server Interface Specification," March 2006.
- [37] OMA Game Services WG, "Game Services Architecture Candidate Version 1.0," March 2006.
- [38] OMA Game Services WG, "In-Game Communications Requirements," March 2004.
- [39] Open Mobile Alliance, "IMS in OMA," September 2005.
- [40] A. Akkawi, S. Schaller, O. Wellnitz, and L. Wolf, "A Mobile Gaming Platform for the IMS," in *Proceedings of 3rd International Workshop on Network and System Support for Games (Netgames 2004)*, Portland, USA, August 2004.
- [41] M. Balakrishnan and M. Sadasivan, "Mobile interactive game interworking in IMS," in *International Conference on IP Multimedia Subsystem Architecture and Applications*, 2007, pp. 1-5.
- [42] Aameek Singh and Arup Acharya. (2004, Noviembre) Multiplayer Network Gaming with the Session Initiation Protocol. [Online]. <http://researchweb.watson.ibm.com/people/a/arup/rc23477.pdf>
- [43] Acharya and Arup, "SIP based VoIP multiplayer network games," 20050181872, agosto 18, 2005.
- [44] Yasuhiro Araki, Akio Yamamoto, and Michael Sweeney, "Dynamic Community Entertainment Services Composition on Next Generation Mobile Network IP Multimedia Subsystem," in *SAINT-W '07: Proceedings of the 2007 International Symposium on Applications and the Internet Workshops*,



Washington, DC, USA, 2007.

- [45] O. Rashid, P. Coulton, and R. Edwards, "Implications of IMS and SIP on the Evolution of Mobile Applications," in *IEEE Tenth International Symposium on Consumer Electronics. ISCE '06*, 2006, pp. 1-6.
- [46] Christoph Neumann, Nicolas Prigent, Matteo Varvello, and Kyoungwon Suh, "Challenges in peer-to-peer gaming," *SIGCOMM Computer Communication*, vol. 37, no. 1, pp. 79-82, 2007.
- [47] Nathaniel E. Baughnman, Marc Liberatore, and Brian Neil Levine, "Cheat-Proof Payout for Centralized and Peer-to-Peer Gaming," *IEEE/ACM Transactions on Networking (TON)*, vol. 15, no. 1, pp. 1-13, febrero 2007.
- [48] Fu Chunyan, Belqasmi Fatna, Alrubaye Mohammed, and Karunamurthy Rajesh, "A Case Study on Multiparty Multimedia Game Development in IMS," in *13th International Conference on Intelligence in Next Generation Networks (ICIN): "Beyond the Bit Pipes"*, 2009.
- [49] F. Belqasmi, C. Fu, M. Alrubaye, and R. Glitho, "Design and Implementation of Advanced Multimedia Conferencing Applications in the 3GPP IP Multimedia Subsystem," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 156-163, November 2009.
- [50] Ericsson, "World's first IMS Mobile Multiplayer gaming solution for mass-market devices," *Nota de prensa*, febrero 2006.
- [51] Lucent Technologies, "Lucent Expands IMS Portfolio with New IP Services for Next-Gen Mobile and Wireline Networks," *Nota de prensa*, junio 2005.
- [52] Inova IT. (2008) Inova IT Promotional Gaming Capability. [Online]. [http://www.imaginecupaccelerator.com/files/inova\\_it\\_gaming\\_capability\\_product\\_info.doc](http://www.imaginecupaccelerator.com/files/inova_it_gaming_capability_product_info.doc)
- [53] Nokia. Forum Nokia - SNAP Mobile Introduction. [Online]. <http://forum.nokia.com/snapmobile>
- [54] Exit Games. (2008, febrero) Exit Games Unveils Neutron® 5.0 at GDC 2008. [Online]. <http://www.exitgames.com/exitgamesandneutron>

- 
- [55] J. Rosenberg, H. Schulzrinne, and Ed. O. Levin, "A Session Initiation Protocol (SIP) Event Package for Conference State," Agosto de 2006.
- [56] O. Novo, G. Camarillo, D. Morgan, and R. Even, "Conference Information Data Model for Centralized Conferencing (XCON) (draft-ietf-xcon-common-data-model-10.txt)," 28 de marzo de 2008.
- [57] R. Sparks, "The Session Initiation Protocol (SIP) Refer Method," 2003.
- [58] Juan Carlos Hernández Sacramento and Alberto Hernández Ortiz, "Desarrollo de la arquitectura de un habilitador de servicios para aplicaciones de trabajo colaborativo en el Subsistema Multimedia IP," ETSI Telecomunicación - Universidad Politécnica de Madrid, Proyecto Fin de Carrera 2008.
- [59] Joaquín Navarro Salmerón and Alberto Hernández Ortiz, "Especificación y desarrollo de habilitadores de servicio orientados a grupos de usuarios en el Subsistema Multimedia IP (IMS)," ETSI Telecomunicación - Universidad Politécnica de Madrid, Madrid, Proyecto Fin de Carrera 2009.
- [60] 3GPP, "IP Multimedia Subsystem (IMS); Stage 2 - TS 23.228 Rel-7," 2008.
- [61] Álvaro Marí Diego and Alberto Hernández Ortiz, "Desarrollo de una arquitectura de red descentralizada para la provisión de aplicaciones multiusuario interactivas en el Subsistema Multimedia IP (IMS)," ETSI Telecomunicación - Universidad Politécnica de Madrid, Proyecto Fin de Carrera 2009.
- [62] Alberto Hernandez, Enrique Vazquez, and Carolina Garcia, "Towards next generation network requirements for next generation gaming," in *CoNEXT '07: Proceedings of the 2007 ACM CoNEXT conference*, New York, USA, 2007.
- [63] Sebastian Zander and Grenville Armitage, "A traffic model for the Xbox game Halo 2," in *Proceedings of the international workshop on Network and operating systems support for digital audio and video*, Stevenson, Washington, USA, 2005, pp. 13-18.
- [64] G. Armitage, M. Claypool, and P. Branch, *Networking and Online Games: Understanding and Engineering Multiplayer Internet Games.*: John Wiley & Sons, June 2006.

- [65] 3GPP, "IP Multimedia Subsystem (IMS) Sh interface; Signalling flows and message contents - TS 29.328,".
- [66] 3GPP, "Sh Interface based on the Diameter protocol; Protocol details - TS29.329,".
- [67] 3GPP, "Telecommunication management; Charging management; Charging principles - TS 32.200,".
- [68] 3GPP, "Telecommunication management; Charging management; Charging architecture and principles - TS 32.240,".
- [69] 3GPP, "Telecommunication management; Charging management; IP Multimedia Subsystem (IMS) charging - TS 32.260,".
- [70] 3GPP, "Telecommunication management; Charging management; Charging data description for the IP Multimedia Subsystem (IMS) - TS 32.225,".
- [71] Daniel Roldán. (2009, enero) Los videojuegos vencen a la recesión. [Online].  
<http://www.lasprovincias.es/valencia/20090125/euros/videojuegos-vencen-recesion-20090125.html>
- [72] Apple Inc. (2008) Enviar un regalo iTunes con iTunes Store.
- [73] H. Schulzrinne, V. Gurbani, P. Kyzivat, and J. Rosenberg, "RPID: Rich Presence Extensions to the Presence Information Data Format (PIDF)," 2006.
- [74] OMA. (2008, septiembre) Shared Profile XDM Specification. [Online].  
[http://www.openmobilealliance.org/Technical/release\\_program/docs/XDM/V2\\_0-20080916-C/OMA-TS-XDM\\_Shared\\_Profile-V1\\_0-20080916-C.pdf](http://www.openmobilealliance.org/Technical/release_program/docs/XDM/V2_0-20080916-C/OMA-TS-XDM_Shared_Profile-V1_0-20080916-C.pdf)
- [75] THQ. (2008, Feb) Frontlines: Fuel of War will support 50 players online over Xbox Live! [Online]. <http://www.thq-games.com/uk/news/show/3323>
- [76] Ashlee Vance. (2008, November) Computing From Weather to Warcraft - NYTimes.com. [Online].  
<http://www.nytimes.com/2008/11/18/technology/business-computing/18super.html>

- [77] A. Hernández, E. Vázquez, P. Capelastegui, and F. González, "Lightweight conferencing enabler for multiparty applications in the IMS," in *Proceedings of the International Conference on Next Generation Mobile Applications, Services and Technologies, 2009. NGMAST '09*, Cardiff, Wales, UK, 2009.
- [78] A. Hernández, E. Vázquez, P. Capelastegui, and M. Álvarez-Campana, "Middleware for decentralized multimedia multiparty applications in the IP Multimedia Subsystem," in *Proceedings of Third ERCIM Workshop on eMobility, held in conjunction with 7th International Conference on Wired/Wireless Internet Communications (WWIC 2009)*, Enschede, The Netherlands, 2009.
- [79] P. Capelastegui et al., "Gestión de grupos en servicios de valor añadido sobre redes IMS," in *V Congreso Iberoamericano de Telemática (CITA 2009)*, Gijón, 2009, pp. 120-125.
- [80] P.A. Moreno et al., "Teleconsulting: a medical application based on IP Multimedia Subsystem Technology for Ambient Assisted Living," in *Proceedings of the 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies. ISABEL 2009*, Bratislava, Slovak Republic, 2009.
- [81] Víctor Villagrà, Verónica Mateos Luis Bellido, "Federated authentication and authorization for reusable learning objects," in *Proceedings of Global Engineering Education Conference (EDUCON 2010)*, Madrid, Spain, 2010.
- [82] GSM Association. (2010, February) One API - GSM World. [Online]. [http://www.gsmworld.com/our-work/mobile\\_lifestyle/oneapi.htm](http://www.gsmworld.com/our-work/mobile_lifestyle/oneapi.htm)
- [83] Manabu Tsukada, José Santa, JinHyeock Choi, Thierry Ernst Yacine Khaled, "A Usage Oriented Analysis of Vehicular Networks: from Technologies to Applications," *Journal of Communications; Vol 4, No 5 (2009): Special Issue: Wireless Communications for Emergency Communications and Rural Wideband Services*, pp. 357-368, June 2009.
- [84] Dennis Luemkemann and Manfred Jakesch. (2008, mayo) End-to-end IMS Application Development with Nokia Siemens Networks IDP Development Tools. [Online]. <http://www.oracle.com/technology/pub/articles/dev2arch/2008/03/ims-application-development.html>

- [85] Technical Specification Group 3rd Generation Partnership Project, "Services and System Aspects, IP Multimedia Subsystem (IMS); Stage 2 (Release 5)," TS 23.228 V6.0.0, January 2003.
- [86] M. Poikselka, A. Niemi, H. Khartabil, and G. Mayer, *The IMS: IP Multimedia Concepts and Services.*: John Wiley & Sons, March 2006.
- [87] R Suomela, E Rasanen, A Koivisto, and J Mattila, "Open-Source Game Development with the Multi-user Publishing Environment (MUPE) Application Platform," in *Lecture Notes in Computer Science*, 2004.
- [88] Nokia. (2007) MUPE: Multi-User Publishing Environment. [Online]. <http://www.mupe.net/>
- [89] Hirohiko Niizumi and Tor Thorsen. (2006, March 15th) PlayStation Network Platform detailed. [Online]. <http://www.gamespot.com/news/6145981.html>
- [90] Open Mobile Alliance, "OMA Service Environment (OSE)," July 2006.
- [91] Kuan-Ta Chen, Chun-Ying Huang, Polly Huang, and Chin-Laung Lei, "An empirical evaluation of TCP performance in online games," in *ACM International Conference Proceeding Series; Vol. 266 Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology*, 2006.
- [92] K.-T Chen, P. Huang, G.-S. Wang, C.-Y. Huang, and C.-L. Lei, "On the Sensitivity of Online Game Playing Time to Network QoS," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, 2006.
- [93] J. Yan and H. Choi, "Security issues in online games," *The Electronic Library*, vol. 20, no. 2, pp. 125-133, 2002.
- [94] Brendan Sinclair. (2008, enero) Xbox Live suffers holiday hangover. [Online]. <http://www.gamespot.com/news/6184221.html>
- [95] Kuan-Ta Chen, Polly Huang, and Chin-Laung Lei, "How sensitive are online gamers to network quality?," *Communications of the ACM*, vol. 49, no. 11, pp. 34-38, noviembre 2006.

- 
- [96] Matthias Dick, Oliver Wellnitz, and Lars Wolf, "Analysis of factors affecting players' performance and perception in multiplayer games," in *Proceedings of 4th ACM SIGCOMM workshop on Network and system support for games*, New York, NY, USA, 2005, pp. 1-7.
- [97] Johannes Färber, "Network game traffic modelling," in *Proceedings of the 1st workshop on Network and system support for games*, 2002, pp. 53-57.
- [98] L. Gautier and C. Diot, "Design and evaluation of MiMaze a multi-player game on the Internet," in *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*, 1998, p. 233.
- [99] Carsten Griwodz, "State replication for multiplayer games," in *Proceedings of the 1st workshop on Network and system support for games*, 2002, pp. 29-35.
- [100] YoungTae Han and HongShik Park, "Distinctive Traffic Characteristics of Pure and Game P2P Applications," in *Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference*, 2008, pp. 405-408.
- [101] Tristan Henderson and Saleem Bhatti, "Networked games: a QoS-sensitive application for QoS-insensitive users?," in *Proceedings of the ACM SIGCOMM workshop on Revisiting IP QoS: What have we learned, why do we care?*, 2003, pp. 141-147.
- [102] Alberto Hernandez, Manuel Álvarez-Campana, Enrique Vazquez, and Vicente Olmedo, "The IP Multimedia Subsystem (IMS). Quality of service and performance simulation," in *Proceedings of the 14th Annual Workshop HPSUA*, Munich, 2007, pp. 142-151.
- [103] Aaron St. John and Brian Neil Levine, "Supporting P2P gaming when players have heterogeneous resources," in *NOSSDAV '05: Proceedings of the international workshop on Network and operating systems support for digital audio and video*, 2005, pp. 1-6.
- [104] Lei Liang, "Entertainment everywhere - Relative QoS optimization for multiparty online gaming in diffserv networks," *IEEE Communications Magazine*, vol. 43, no. 5, p. 75, 2005.
- [105] M. Mauve, V. Hilt, C. Kuhmunch, and W. Effelsberg, "RTP/I-toward a common application level protocol for distributed interactive media," *IEEE Transactions on Multimedia*, vol. 3, no. 1, pp. 152-161, 2001.

- [106] Robert Mitchell. (2007, mayo) Everything but the Game. [Online]. <http://www.ogdc2007.com/session.php?id=16>
- [107] Sangheon Pack et al., "Game transport protocol: lightweight reliable transport protocol for massive interactive on-line game," in *Proceedings of SPIE*, vol. 4861, 2002.
- [108] Lothar Pantel and Lars C. Wolf, "On the impact of delay on real-time multiplayer games," in *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video* , 2002, pp. 23-29.
- [109] Joseph D. Pellegrino and Constantinos Dovrolis, "Bandwidth requirement and state consistency in three multiplayer game architectures," in *Proceedings of the 2nd workshop on Network and system support for games*, 2003, pp. 52-59.
- [110] Peter Quax, Patrick Monsieurs, Wim Lamotte, Danny De Vleeschauwer, and Natalie Degrande, "Objective and subjective evaluation of the influence of small amounts of delay and jitter on a recent first person shooter game," in *Proceedings of 3rd ACM SIGCOMM workshop on Network and system support for games*, 2004, pp. 152-156.
- [111] T. Renier, Kim Lynggarg Larsen, G. Castro, and H.P. Schwefel, "Mid-Session Macro-Mobility in IMS-Based Networks," *Vehicular Technology Magazine*, vol. 2, no. 1, pp. 20-27, marzo 2007.
- [112] Christian Schaefer, Thomas Enderes, Hartmut Ritter, and Marina Zitterbart, "Subjective quality assessment for multiplayer real-time games," in *Proceedings of the 1st workshop on Network and system support for games* , Braunschweig, Germany, 2002, pp. 74-78.
- [113] Jouni Smed and Harri Hakonen, *Algorithms and networking for computer games.*: John Wiley, 2006.
- [114] Yow-Jian Lin, K. Guo, and S. Paul, "Sync-MS: synchronized messaging service for real-time multi-player distributed game," in *Proceedings of 10th IEEE International Conference on Network Protocols*, 2002, pp. 155-164.
- [115] N. Imai, M. Isomura, and H. Horiuchi, "Multi-Device Seamless Service by User-Oriented Session in All-IP Network," in *IEEE International Conference on Communications, 2007. ICC '07.* , Glasgow, 2007, pp. 1929-1934.

- 
- [116] D. Linner, F. Kirsch, I. Radusch, and S. Steglich, "Context-aware multimedia provisioning for pervasive games," in *Seventh IEEE International Symposium on Multimedia*, 2005, p. 9.
- [117] Min-Xiou Chen, Chen-Jui Peng, Ren-Hung, and Hwang, "SSIP: Split a SIP session over multiple devices," *Computer Standards & Interfaces*, vol. 29, no. 5, pp. 531-545, julio 2007.
- [118] L. Skorin-Kapov, M. Mosmondor, O. Dobrijevic, and M. Matijasevic, "Application-Level QoS Negotiation and Signaling for Advanced Multimedia Services in the IMS," *IEEE Communications Magazine*, vol. 45, no. 7, pp. 108-116, julio 2007.
- [119] Jasper Aartse Tuijn and Dennis Bijwaard, "Spanning a multimedia session across multiple devices," *Bell Labs Technical Journal*, vol. 12, no. 4, pp. 179-193, febrero 2008.
- [120] N. Imai et al., "Dynamic Session Modification between Multiple Devices in the IMS/MMD Architecture," in *IEEE Wireless Communications and Networking Conference, 2008. WCNC 2008*, Las Vegas, NV, USA, 2008, pp. 3255-3260.
- [121] Jani Hautakorpi and Gonzalo Camarillo, "Evaluation of DHTs from the viewpoint of interpersonal communications," in *Proceedings of the 6th international conference on Mobile and ubiquitous multimedia*, Oulu, Finland, 2007, pp. 74-83.
- [122] Noëmie Simoni, Chunyang Yin, Rhéa Berberi, and Ghislain Du Chene, "An NGN middleware based on an enhanced IMS," in *Proceedings of the 2007 Workshop on Middleware for next-generation converged networks and applications*, Port Beach, California, 2007.
- [123] A. Al-Hezmi, C. Riede, O. Friedrich, S. Arbanowski, and T. Magedanz, "Cross-fertilization of IMS and IPTV services over NGN," in *Innovations in NGN: Future Network and Services, 2008. K-INGN 2008. First ITU-T Kaleidoscope Academic Conference*, Geneva, 2008, pp. 153-160.
- [124] Exit Games. (2008) Neutron - Multiplayer Engine for Cross-Platform Gaming - Overview. [Online]. <http://www.exitgames.com/product/overview>



- [125] W. Munkongpitakkun, S. Kamolphiwong, and S. Sae-Wong, "Enhanced web session mobility based on SIP," in *Proceedings of the 4th international conference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology*, Singapore, 2007, pp. 346-350.
- [126] J. Rosenberg, H. Schulzrinne, and Ed. R. Mahy, "An INVITE-Initiated Dialog Event Package for the Session Initiation Protocol (SIP)," Noviembre de 2005.