

**MARCO DE REFERENCIA PARA LA INTEGRACIÓN
ARQUITECTÓNICA DE SERVICIOS DE STREAMING A
SISTEMAS DE GESTIÓN DE APRENDIZAJE DE CÓDIGO
ABIERTO**



Monografía

**Rubén Darío Benavides Cabrera
José Leonardo Díaz Ordoñez**

Director
Mag. Mario Fernando Solarte Sarasty

Universidad del Cauca

**Facultad de Ingeniería Electrónica y Telecomunicaciones
DEPARTAMENTOS DE TELEMÁTICA Y SISTEMAS**
Línea de Investigación en Aplicaciones y Servicios sobre Internet

Popayán, Mayo de 2010

Tabla de Contenido

Introducción	1
Capítulo 1. Base Conceptual.....	5
1.1 Sistemas de Gestión de Aprendizaje.....	5
1.1.1 Sistemas de gestión de aprendizaje de código abierto	8
1.1.1.1 .LRN	8
1.1.1.2 Moodle.....	10
1.2 Estándar SCORM.....	12
1.2.1 Versiones	13
1.2.2 SCORM y la WEB	14
1.2.3 SCORM y la trazabilidad del estudiante.....	14
1.2.4 Paquetes de aprendizaje	15
1.2.5 Núcleo de SCORM	16
1.3 Streaming.....	17
1.3.1 Definición.....	17
1.3.2 Características de la tecnología <i>streaming</i>	18
1.3.2.1 Arquitectura	18
1.3.2.2 Modos de transmisión.....	18
1.3.2.3 ¿Cómo trabaja el <i>streaming</i> ?.....	18
1.3.3 Aplicaciones y servicios de <i>streaming</i>	19
1.3.3.1 Video bajo demanda.....	19
1.3.3.2 Video con señal en vivo	20
1.3.3.3 Audio bajo demanda y en vivo	21
1.3.3.4 Aplicaciones colaborativas.....	21
1.3.3.5 RPC <i>Streaming</i>	21
1.3.3.6 <i>Callbacks</i> o Tecnología <i>Push</i>	22
1.3.3.7 <i>Shared Object</i> , objetos compartidos	23
1.3.4 Tecnologías para hacer <i>streaming</i>	23
1.3.4.1 Tecnología basada en QuickTime.....	23
1.3.4.2 Tecnología basada en RealMedia.....	24
1.3.4.3 Tecnología basada en Windows Media	26
1.3.4.4 Tecnología basada en Flash.....	27

Capítulo 2. Lineamientos para la integración de servicios de <i>streaming</i> en sistemas de aprendizaje de código abierto	30
2.1 <i>Contextualización</i>	30
2.2 <i>M1: Modelo de dinámica del material</i>	31
2.2.1 Entrega del material.....	31
2.2.2 Interacción con el material.....	32
2.2.3 Interacción con el profesor.....	33
2.2.4 Interacción entre estudiantes.....	33
2.2.5 Realimentación del material.....	34
2.2.6 Representación gráfica del modelo M1	34
2.3 <i>M2: Modelo de servicios de streaming</i>	36
2.4 <i>M3: Modelo para selección de tecnología</i>	40
2.4.1 Atributos de la temática	41
2.4.2 Participación de los actores de la actividad formativa.....	41
2.4.3 Costos	42
2.4.4 Aplicación del modelo M3.....	42
2.5 <i>Lineamientos técnicos</i>	43
2.5.1 Sistema Operativo	44
2.5.2 Servidor y lenguaje de programación	44
2.5.3 Protocolo	45
2.5.4 Ancho de banda	46
2.5.5 Costos	46
2.5.6 Lineamientos de integración de las tecnologías de <i>streaming</i> y del LMS.....	47
2.5.6.1 Nivel de comunicación	49
2.5.6.2 Nivel de datos.....	50
2.5.6.3 Nivel de lógica de negocio	53
2.5.6.4 Nivel de presentación	53
2.5.6.5 Seguridad	54
Capítulo 3. Arquitectura de Referencia	57
3.1 <i>Descripción general del sistema</i>	57
3.2 <i>Paquete RALO</i>	58
3.2.1 <i>Estructura</i>	59
3.3 <i>Ciclo de vida del paquete RALO</i>	59
3.4 <i>Casos de uso esenciales del sistema</i>	60

3.4.1	<i>Actores y entidades</i>	60
3.4.2	<i>Realización casos de uso esenciales</i>	61
3.5	<i>Referencia para el diseño de la arquitectura</i>	65
3.6	<i>Secuencia en la arquitectura</i>	68
3.7	<i>Descripción detallada de la arquitectura</i>	69
3.7.1	<i>StreamelClient</i>	74
3.7.2	<i>StreamelWeb</i>	75
3.7.3	<i>StreamelService</i>	76
Capítulo 4. Validación de los lineamientos para la integración de servicios de streaming		79
4.1	<i>Introducción</i>	79
4.2	<i>Descripción de los requerimientos del servicio</i>	79
4.3	<i>Aplicación del modelo M3</i>	80
4.3.1	<i>Aplicación Modelo M1</i>	80
4.3.2	<i>Aplicación Modelo M2</i>	81
4.3.3	<i>Refinamiento de tecnologías</i>	82
4.3.3.1	<i>Atributos de la temática</i>	83
4.3.3.2	<i>Participación de los actores de la actividad formativa</i>	83
4.3.3.3	<i>Refinamiento y selección de tecnologías</i>	84
4.4	<i>Implementación de los servicios de streaming</i>	86
4.4.1	<i>Módulo reproducción de audio/video</i>	89
4.4.2	<i>Módulo presentación de diapositivas</i>	90
4.4.3	<i>Módulo de línea de tiempo</i>	90
4.4.4	<i>Módulo de comentarios</i>	91
Capítulo 5. Conclusiones, recomendaciones y trabajos futuros		94
5.1	<i>Conclusiones</i>	94
5.2	<i>Recomendaciones y trabajos futuros</i>	97
Bibliografía		99

Lista de Figuras

Figura 1. Arquitecturas OpenACS y .LRN	9
Figura 2. Arquitectura .LNR multi-nivel.....	10
Figura 3. RPC streaming, respuestas asíncronas	22
Figura 4. Diagrama de transición de estados para servicios con callbacks	22
Figura 5. Representación grafica del modelo M1	35
Figura 6. Mecanismos de comunicación	49
Figura 7. Modelo de datos de trazabilidad	51
Figura 8. Modelo relacional básico para plataforma de streaming.....	52
Figura 9. Lógica de negocio cliente y lógica negocio servidor	53
Figura 10. Presentación SCO en Moodle y .LRN	54
Figura 11. Vista general para la integración de servicios de <i>streaming</i>	57
Figura 12. Casos de uso mínimos asociados a la arquitectura de referencia	60
Figura 13. Organización para los servicios multimedia. [91].....	67
Figura 14. Adición de middleware para extensión de servicios en el LMS	68
Figura 15. Secuencia seguida por la arquitectura de referencia.....	69
Figura 16. Arquitectura de referencia detallada.....	71
Figura 17. Diagrama general de actividad del sistema	72
Figura 18. Diagrama general de secuencia para la arquitectura de referencia.....	73
Figura 19. Componentes básicos del la aplicación Flex StreamelClient.....	74
Figura 20. Plataforma para el despliegue de servicios de <i>streaming</i>	76
Figura 21. Diagrama de secuencia para servicio de video <i>streaming</i> básico.....	78
Figura 22. Componentes de la implementación plataforma de <i>streaming</i>	87
Figura 23. Aplicación del lado del cliente de consumo del servicio de video <i>streaming</i>	88
Figura 24. Vista general patrón presentation model.....	89

Lista de Tablas

Tabla 1 Factores comparativos útiles para la selección de la tecnología.	29
Tabla 2. Ejemplos relación material-interacción	33
Tabla 3. Análisis clase tradicional-video <i>streaming</i> , usando el modelo M1	36
Tabla 4. Caracterización de los servicios y aplicaciones de <i>streaming</i> , a partir de la naturaleza del material.....	38
Tabla 5. Identificación de las interacciones M1 en las aplicaciones de la tecnología de <i>streaming</i>	40
Tabla 6. Resumen fases aplicación modelo M3.	43
Tabla 7. Caso de uso esencial: Seleccionar servicio de <i>streaming</i>	62
Tabla 8. Caso de uso esencial: Configurar servicio de <i>streaming</i>	62
Tabla 9. Caso de uso esencial: Cargar recurso multimedia	63
Tabla 10. Caso de uso esencial: Exportar paquete RALO	64
Tabla 11. Caso de uso esencial: Exportar paquete RALO	65
Tabla 12: Caso de uso esencial: Consultar objeto de aprendizaje	65
Tabla 13. Modelo M1 para caso de estudio	81
Tabla 14. Relación dinámica del material con servicios de <i>streaming</i> para caso de estudio.	81
Tabla 15. Relación dinámica del material con servicios de <i>streaming</i> para caso de estudio refinada.....	82
Tabla 16. Refinamiento tecnologías de <i>streaming</i> para audio/video.	85
Tabla 17. Refinamiento tecnologías de <i>streaming</i> para servicio de presentación de diapositivas.....	86
Tabla 18. Refinamiento tecnologías de <i>streaming</i> para servicio de comentarios	86

Introducción

Con la marcada tendencia mundial a favor del avance de la sociedad del conocimiento, Colombia, como país en vía de desarrollo, ha comprendido la importancia de la información y el conocimiento como bienes intangibles que potencian el desarrollo social, político y económico [1, 2]. El uso de nuevas Tecnologías de Información y Comunicación, TIC, tales como el uso de computadoras, Internet y otros dispositivos, se ha vuelto una herramienta para muchos indispensable en estos contextos. En esta nueva era conocida como Sociedad de la Información y el Conocimiento, la educación tiene un importante papel que cumplir: fomentar el desarrollo del capital humano, para un mejor aprovechamiento de la información y la adquisición de conocimiento [3].

La educación electrónica o *e-learning* hace uso de los medios de comunicación ofrecidos por las TIC para la interacción entre los actores del proceso educativo. En algunos casos esta estrategia educativa se desarrolla bajo un enfoque tecnológico, con la idea de que la sofisticación de dicho entorno proporcionará la calidad del proceso de enseñanza, dejando de lado factores importantes como el contenido o incluso al propio estudiante [4]. La mejor estrategia de éxito en este tipo de ambientes relaciona al estudiante, el contenido y la tecnología adaptados al entorno, a las actividades de aprendizaje y los tipos de conocimiento que se quieren abordar [5]. Una de las principales herramientas del e-learning es el sistema de gestión de aprendizaje, LMS, Learning Management System, el cual básicamente provee y gestiona servicios de aprendizaje en línea, sin embargo, como se analiza en [6] la mayoría de LMS son usados parcialmente frente a todas sus posibilidades, en su gran mayoría, son usadas para publicar material docente o bien para comunicarse con los alumnos a través de correo electrónico o foros.

Es importante resaltar el papel del profesor como fuente de conocimiento y experiencia, siendo él, la persona que determina cuándo y cómo los estudiantes recibirán los materiales formativos, de ahí la importancia de brindar mejores y más avanzados medios de comunicación entre el profesor y el estudiante en pro de un modelo *e-learning* que mantenga ciertas características del modelo educativo presencial. Estrategias como el uso de videoconferencia [7], distribución de video por demanda, la integración de contenido audiovisual y el desarrollo de aplicaciones colaborativas en tiempo real [8], facilitan el proceso de aprendizaje, rescatan algunas prácticas pedagógicas no adoptadas

hasta el momento en los ambientes virtuales y ofrecen al estudiante una percepción de individualidad y personalización en su proceso formativo.

En los últimos años se ha popularizado el uso del *streaming* en Internet para la transmisión de video y audio en tiempo real o bajo demanda, debido a esto, muchas veces el proceso de *streaming* es asociado esencialmente a la transmisión de contenido multimedia (imágenes, audio y video) [9], sin embargo su aplicación en otras áreas ha permitido la creación de nuevos sistemas que ofrecen beneficios adicionales y únicos [10-13]. El *streaming* en ambientes virtuales de educación puede mejorar la calidad de los servicios existentes, agregando características de interactividad, o facilitando la creación de nuevos servicios que mejoren la formación educativa en diferentes áreas [14].

Este proyecto de investigación busca responder la pregunta, ¿Cómo se pueden integrar servicios de streaming a un entorno virtual de aprendizaje, que hace uso de un LMS basado en software libre, cuidando la integridad del sistema?, teniendo en cuenta la necesidad de contar con un sistema integral de aprendizaje que permita la creación de nuevos servicios, idealmente definiendo nuevas relaciones entre los usuarios (estudiantes, profesores, etc.) y el LMS, es decir que los servicios ofrezcan la posibilidad de aplicar nuevas estrategias de aprendizaje.

La integración que se quiere alcanzar con el LMS se enfoca a las aplicaciones y servicios basados en tecnología *streaming*, teniendo en cuenta factores importantes como:

- Seguridad a partir de los métodos de autenticación del usuario con la plataforma LMS.
- Herramientas de trazabilidad que se pueden agregar a los servicios basados en *streaming* y la forma en que se pueden administrar.
- Cómo se integran las aplicaciones o servicios dentro de la arquitectura de comunicación implementada por el LMS.
- Integridad de la plataforma LMS, considerando que a las herramientas basadas en software libre se les hace mantenimiento de forma constante.

Dentro de los objetivos específicos que se quieren alcanzar en este trabajo de investigación se incluyen la definición de una base conceptual sobre la tecnología de *streaming*, sus aplicaciones o servicios y las tecnologías que se pueden usar para implementarlos. Además se busca definir los lineamientos necesarios para la integración de los servicios de streaming en el ambiente virtual de aprendizaje desde el punto de vista metodológico y técnico, lo que implica a su vez, la definición de una arquitectura que sirva de referencia para dicha integración. Finalmente se busca validar las propuestas de la investigación ejemplificando la integración de un servicio de video streaming, aclarando que aunque no es un mecanismo formal ni completo de certificación, nos permite verificar

de forma preliminar el funcionamiento de los lineamientos y las características de la arquitectura de referencia.

En el primer capítulo se presenta la base conceptual de los temas relacionados con las aplicaciones y servicios de *streaming*, incluyendo las tecnologías más usadas y otras tecnologías que permitieron integrar estos servicios con la plataforma LMS.

En el segundo capítulo se presenta los lineamientos de índole técnica y metodológica propuestos para la integración de servicios de *streaming* a un sistema de gestión de aprendizaje de código abierto.

En el tercer capítulo se propone una arquitectura general para la creación de aplicaciones y servicios de *streaming*, describiendo cada uno de los módulos que componen esta arquitectura y la forma en que se comunican.

En el capítulo 4 se utilizan los lineamientos y recomendaciones propuestas, para el diseño e integración de un servicio de video *streaming* en el LMS EVA (Entorno Virtual de Aprendizaje) de la Universidad del Cauca [15], con el objetivo de realizar una verificación preliminar de dichos lineamiento y arquitectura de referencia de los capítulos anteriores.

Para finalizar, en el quinto y último capítulo se presentan las conclusiones a las cuales se llegó con la culminación del proyecto y los trabajos futuros que se pueden realizar teniendo como base este proyecto de grado.

Adicional al documento se presentan algunos anexos que complementan la información descrita, así:

- El Anexo 1 agrupa información relacionada sobre los protocolos y estándares que complementan la base conceptual del capítulo 1 y que sirve de referencia para la elaboración de los lineamientos y propuesta de arquitectura de integración. También se presenta un listado de los codec [16] de video más utilizados que tienen relación con la tecnología de *streaming*, dicha información es útil para lo relacionado con la implementación del caso de estudio del capítulo 4.
- El Anexo 2 presenta detalles sobre la implementación real del servicio de video *streaming* y algunos resultados de pruebas realizadas que complementan la información descrita en el capítulo 4 sobre el caso de estudio.
- El Anexo 3 sirve como referencia para la actividad de evaluación de los costos asociados a la producción de los recursos multimedia, descrita en uno de los modelos conceptuales del capítulo 2.

- El Anexo 4 describe en forma detallada los cambios necesarios en la implementación del estándar SCORM de .LRN para el manejo de nuevos modelos de datos, basados en el estándar CMI.
- El Anexo 5 presenta el modelo de encuesta y análisis realizados para evaluar algunas características del servicio de video streaming integrado al LMS EVA durante las fases de prueba del caso de estudio.
- Como anexo digital se incluyen los artefactos (código fuente, archivos de configuración, utilidades software, etc.) utilizados en la implementación y puesta en marcha de la plataforma de servicios de *streaming* integrada con EVA.

Capítulo 1. Base Conceptual

1.1 Sistemas de Gestión de Aprendizaje

Sistema de Gestión de Aprendizaje, o LMS, es un término que es usado para un amplio rango de sistemas que organizan y proveen acceso a servicios de aprendizaje en línea para estudiantes y profesores en general. Estos servicios usualmente incluyen control de acceso, provisión de contenido de aprendizaje, herramientas de comunicación, administración de grupos de usuarios, entre otros. A menudo un LMS es referido como una plataforma de aprendizaje.

Como se analiza en [6], generalmente los LMS son usados parcialmente frente a todas sus posibilidades, en su gran mayoría, son usados para publicar material docente o bien para comunicarse con los alumnos a través de correo electrónico o foros. También destaca las barreras existentes en la implantación de este tipo de plataformas, resaltando el valor tradicional de las clases magistrales dentro de un sistema que tiende a favorecer la investigación.

Si bien los LMS brindan herramientas que facilitan la transferencia de conocimiento entre los estudiantes y el profesor, es importante avanzar hacia un ambiente virtual donde el estudiante es fuente de conocimiento y donde la interacción entre ellos mejora la calidad de su formación [3, 17]. Bajo esta perspectiva, los LMS actuales limitan el desarrollo de servicios [18] donde el estudiante es el motor o donde la interacción entre ellos genera el conocimiento; evidencia de lo anterior radica en las actividades tradicionales que se realizan en la adaptación y personalización de este tipo de sistemas para satisfacer las necesidades y objetivos de cada universidad o ambiente educativo [6].

Según Miguel Zapata [19, 20], una plataforma de teleformación, o un sistema de gestión de aprendizaje en red, es una herramienta informática y telemática organizada en función de unos objetivos formativos de forma integral, es decir, se pueden conseguir exclusivamente dentro de ella, y de unos principios de intervención psicopedagógica y organizativos, de manera que se cumplen los siguientes criterios básicos:

- Posibilita el acceso remoto tanto a profesores como a alumnos en cualquier momento desde cualquier lugar con conexión a Internet o a redes con protocolo TCP/IP.
- Utiliza un navegador web. Permite a los usuarios acceder a la información a través de navegadores estándares (como Mozilla Firefox, Internet Explorer, Opera, etc.), utilizando el protocolo de comunicación HTTP.
- El acceso es independiente de la plataforma o computador de cada usuario, es decir, se hace uso de estándares de manera que la información puede ser visualizada y tratada en las mismas condiciones, con las mismas funciones y con el mismo aspecto en cualquier computador.
- Tiene una arquitectura cliente/servidor. Permite retirar y depositar información de forma organizada y facilita la centralización de esta información.
- El acceso al sistema es restringido y selectivo.
- Incluye como elemento básico una interfaz gráfica común, con un único punto de acceso, de manera que en ella se integran los diferentes elementos multimedia que constituyen los cursos: texto, gráficos, vídeo, sonidos, animaciones, etc.
- Utiliza páginas elaboradas con un estándar aceptado por el protocolo HTTP: HTML o XML.
- Facilita la presentación de la información en formato multimedia. Los formatos HTML o XML permiten la inclusión de información multimedia. Pueden utilizarse gráficos, animaciones, audio y vídeo (ya sea mediante la transferencia de ficheros tradicional o en tiempo real).
- Permite la actualización y la edición de la información con los medios propios que han de ser sencillos o con los medios estándares de que disponga el usuario. Tanto de las páginas web como de los documentos depositados.
- Permite estructurar la información y los espacios en formato hipertextual. De esta manera la información se puede organizar, estructurada a través de enlaces y asociaciones de tipo conceptual y funcional, de forma que queden diferenciados distintos espacios y que esto sea perceptible por los usuarios.
- Permite establecer diferentes niveles de usuarios con distintos privilegios de acceso. Debe contemplar al menos: el administrador, que se encarga del mantenimiento del servidor, y de administrar espacios, claves y privilegios; el

coordinador o responsable de curso, es el perfil del profesor que diseña, y se responsabiliza del desarrollo del curso, de la coordinación docente y organizativa del curso en la plataforma; los profesores tutores, encargados de Sistemas de gestión del aprendizaje.

Durante los últimos años han aparecido varios de estos sistemas, con ellos se quiere proporcionar aplicaciones que permitan mantener entornos de aprendizaje virtuales útiles y eficaces para todos los actores involucrados.

Desde el punto de vista de la arquitectura, un LMS debería tener las siguientes características [21]:

- **Abierta:** El objetivo es crear aplicaciones *e-learning* interoperables y conectables entre sí de forma sencilla, es decir, que herramientas comerciales de distintos fabricantes puedan ensamblarse en un único sistema global.
- **Escalable:** Independiente del tamaño inicial con que se conciba el sistema, la arquitectura debe estar definida de tal forma que permita su crecimiento. Por ejemplo, al ir aumentando el repositorio de objetos educativos las aplicaciones encargadas de gestionarlos deben tener capacidad suficiente para no sobrecargarse.
- **Global:** Permitir la diversidad lingüística y cultural. Este es uno de los objetivos con mayor dificultad, puesto que la gran mayoría de las aplicaciones están destinadas para una audiencia específica.
- **Integrada:** No sólo entre los componentes del propio sistema sino entre otras aplicaciones no directamente relacionadas con el aprendizaje. El objetivo es conseguir interoperabilidad entre ellas.
- **Flexible:** Vista desde tres perspectivas complementarias; adaptabilidad a la estructura de la organización donde se implanta, la adaptabilidad a los planes de estudio de la institución y la capacidad de adaptación a los contenidos y estilos pedagógicos de la organización.

El objetivo primario de un LMS es proporcionar y administrar los diferentes recursos de un sistema de enseñanza a los estudiantes, siguiendo su progreso y su rendimiento [6]. En particular, las características principales de un LMS, que lo diferencian de las herramientas tradicionales de enseñanza son: personalización, búsqueda y navegación en catálogos educativos, registro, seguimiento de la evolución del estudiante, mecanismos de valoración y evaluación.

1.1.1 Sistemas de gestión de aprendizaje de código abierto

Hablamos de LMS de código abierto, cuando los sistemas se distribuyen con una licencia que permite ver y modificar el código fuente base de la aplicación. Esto significa que los usuarios, programadores, profesores y administradores pueden adaptar las plataformas a sus necesidades, tanto tecnológicas como pedagógicas. Además, implica que una parte básica y central del sistema se encuentra en manos de comunidades y de usuarios que aportan tiempo de investigación y desarrollo al proyecto.

Las diversas plataformas de tele-educación, libres o propietarias, proveen principalmente soporte para tareas administrativas genéricas, también ofrecen algunas herramientas básicas para soportar las actividades de enseñanza y aprendizaje. De la misma forma en que se desarrollan nuevos sistemas con mejores características y se actualizan los existentes, toman fuerza las iniciativas para definir diferentes estándares que permitan el intercambio de material de aprendizaje entre las diferentes plataformas [21-23].

1.1.1.1 .LRN

.LRN [24] es una herramienta muy completa para el desarrollo ágil de comunidades de aprendizaje web, específicamente en el contexto de la administración de contenido y cursos. El sistema se basa en el *framework* de desarrollo OpenACS [25] e incluye la definición de modelos de datos, lógica de negocio y plantillas que facilitan la puesta en marcha de este tipo de sistemas.

.LRN no es considerado un sistema de *e-learning* tradicional, dada su arquitectura basada en un *framework* para comunidades altamente escalable, que es ideal para soportar los procesos de enseñanza, investigación y administración [26].

Por defecto el sistema consiste de un sistema de administración de portales robusto, un conjunto completo de herramientas de colaboración y soportado sobre una capa de infraestructura de tipo empresarial, basada a su vez en estándares abiertos.

Tecnológicamente .LRN es uno de los LMS más avanzados, implementando tecnologías de punta y ofreciendo servicios de alta calidad. La plataforma tiene una arquitectura cliente servidor [26, 27]. En su forma más general los componentes bajo los cuales la plataforma está basada se muestran en la figura 1; entre ellos:

- Sistema de base de datos (PostgreSQL, Oracle)
- Servidor Web (AOLserver)
- Framework base (OpenACS)

- Medio de acceso (Navegador Web en el cliente)

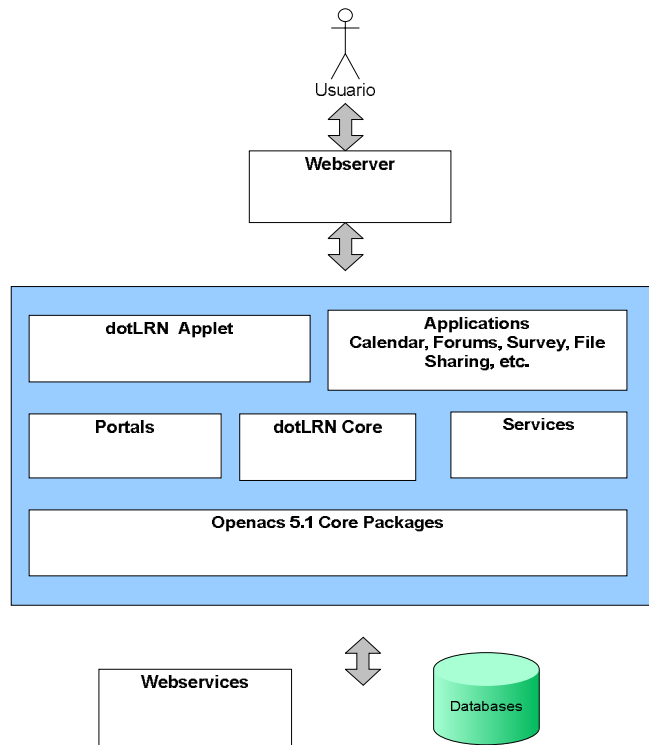


Figura 1. Arquitecturas OpenACS y .LRN

La arquitectura fue diseñada por ingenieros del MIT, Massachusetts Institute of Technology, para facilitar la implementación de comunidades virtuales con un mínimo de esfuerzo y al mismo tiempo con la suficiente capacidad de extenderse a nuevas áreas de desarrollo, incluso a aquellas que no fueron contempladas en el diseño original [21].

Los usuarios, o incluso administradores del sistema tienen una interfaz web que les permite crear departamentos y escuelas dentro de los cuales se distribuyen los cursos. Cada uno de estos cursos tiene una página creada automáticamente (portal), dentro de la cual aparecen todas las funcionalidades que el profesor considere importante para un curso particular. La apariencia de este portal puede ser adaptada por el profesor sin necesidad de programar. Cada usuario del sistema (Profesor, Estudiante, Ayudante, Administrador, etc) tiene una página personal donde la información de todos los cursos y comunidades a las que pertenece es agrupada y presentada de manera uniforme.

La figura 2 hace una completa presentación de la arquitectura de .LRN en cada uno de sus niveles.



Figura 2. Arquitectura .LNR multi-nivel.

Uno de los puntos más fuertes dentro de .LRN es el soporte a diferentes estándares relacionados con los recursos de aprendizaje y su estructura, entre otros.

Sin lugar a duda el estándar más importante de la última década es SCORM, un estándar que reúne lo mejor de diferentes modelos y crea un lenguaje común dentro del variado y siempre cambiante mundo del *e-learning* [28].

1.1.1.2 Moodle

Moodle, proviene del acrónimo *Modular Object Oriented Dynamic Learning*, es un LMS desarrollado originalmente por Martin Dougiamas, un profesor con título en ciencias de la computación y la educación; con principios constructivistas, plasmados directamente en la estrategia pedagogía sobre la cual se basa Moodle [29].

El constructivismo afirma que el aprendizaje es particularmente efectivo cuando se construye algo para que otros experimenten. Puede ser cualquier cosa desde una frase hablada o un artículo en Internet, a artefactos más complejos como una pintura, una casa o un paquete de software.

El constructivismo social extiende el constructivismo en el ámbito social, en donde son los grupos los que aportan el conocimiento en beneficio de cada uno de los miembros, creando una pequeña cultura colaborativa de “artefectos” con significados compartidos.

Desde la perspectiva aplicada en Moodle, el constructivismo, el perfil del estudiante tiene las siguientes características:

- Los estudiantes acceden a las clases con un punto de vista establecido, formado por tiempo de experiencia y conocimiento relacionado.
- Todas las experiencias del estudiante, así como la interpretación de sus observaciones y punto de vista son modelados a medida que se evoluciona y complementa.
- El cambio en su punto de vista implica un duro trabajo.
- Los estudiantes aprenden unos de otros, así como los profesores.
- Los estudiantes aprenden más cuando se practica o se “hace”.
- La creación de mecanismos y las oportunidades que tienen todos de expresar su punto de vista favorece la creación de nuevas y mejores ideas.

Al igual otros LMS, Moodle ofrece herramientas que facilitan los procesos de enseñanza, permite crear espacios virtuales de trabajo formados por recursos de información (en formato textual, multimedia, entre muchos otros) así como recursos de formación tipo tareas enviadas mediante un entorno web, exámenes, encuestas, foros entre otros [30]. A nivel tecnológico, Moodle implementa diferentes estándares que garantizan calidad y robustez de la plataforma:

- Nivel de autenticación integrado con componentes que extienden su funcionalidad y permiten utilizar servicios externos como LDAP, Radius, Shibboleth, OpenID, IMS Enterprise, entre otros.
- Especificaciones abiertas para importación y exportación cuestionarios: formato GIFT, IMS QTI, XML y XHTML.
- Objetos de aprendizaje con soporte sobre IMS Content Packaging, SCORM, AICC (CBT), Learning Activity Management System, LAMS.
- Medios de seguimiento y herramientas sociales: Especificaciones RSS o ATOM.
- Interoperabilidad mediante la definición e implementación de modelos propietarios de Blackboard y WebCT.

1.2 Estándar SCORM

Los sistemas de gestión de contenidos web originales usaban formatos propietarios para los contenidos que distribuían. Como resultado, no era posible el intercambio de tales contenidos. El estándar SCORM es una especificación que permite crear objetos pedagógicos estructurados, lo que hace posible el crear contenidos que puedan importarse dentro de sistemas de gestión de aprendizaje diferentes, siempre que éstos soporten la norma SCORM.

La sigla SCORM se extrae de la frase inglesa *Sharable Content Object Reference Model*, en español Modelo de Referencia para Objetos de Contenido Reusables [31]. SCORM es fruto del trabajo liderado por ADL, *Advance Distributed Learning Initiative*, establecida por el departamento de defensa de los Estados Unidos de América en 1997, la cual busca aplicar los desarrollos, investigaciones y avances de diferentes grupos en el área del *e-learning*; esto incluye extraer y aplicar los componentes más importantes de diferentes modelos usados por los diferentes grupos, entre ellos IMS Global Learning Consortium, el AICC, Aviation Industry CBT (Computer-Based Training) Committee, ARIADNE, Alliance of Remote Instructional Authoring and Distribution Networks for Europe, IEEE LTSC, IEEE Learning Technology Standards Committee, entre otros. Adicionalmente, esta iniciativa tiene como objetivo facilitar la interoperabilidad de las herramientas de aprendizaje y contenidos educativos a escala global.

Las principales ventajas que ofrece la implementación del estándar SCORM sobre una plataforma de *e-learning* son las siguientes [32]:

- Posibilidad de importar a la plataforma contenidos de *e-learning* ya desarrollados bajo el estándar SCORM.
- Interoperabilidad: Independencia con respecto a proveedores, ya que si se desea cambiar de plataforma, los contenidos ya creados o importados serán compatibles con cualquier otra plataforma SCORM del mercado.
- Accesibilidad: El estándar propone y ofrece un medio común de acceso a los objetos de aprendizaje, independiente de la ubicación remota. Sin embargo no impone criterios sobre la forma en que se deben diseñar e implementar los mismos, ni su naturaleza.
- Durabilidad: Independiente a los cambios tecnológicos de la plataforma los objetos de aprendizaje existentes mantienen sus propiedades originales. De otro lado el

estándar SCORM, en cada una de sus versiones, está diseñado para ser compatible con versiones anteriores del mismo.

- Reusabilidad: los objetos de aprendizaje se pueden reutilizar en diferentes aplicaciones o formar parte de otros contenidos.
- Garantía de calidad: la obtención del certificado SCORM pone de manifiesto que la plataforma certificada ha superado una serie de pruebas y requisitos, que a su vez garantizan elevadas prestaciones y un alto nivel de eficacia [33].

1.2.1 Versiones

Desde la conformación de ADL en 1997, se han publicado cuatro versiones principales del estándar SCORM.

Versión 1.0: En enero del año 2000 la versión 1.0 entra en fase de pruebas, donde muchos participantes intentan implementar la especificación, como es de esperarse ADL recibe numerosos reportes de preguntas y problemas. En esta versión la sigla SCORM proviene del nombre *Sharable Courseware Object Reference Model*.

Versión 1.1: Incluye correcciones y mejoras basadas en las lecciones aprendidas durante las pruebas de la versión 1.0.

Esta versión mantiene el mismo contexto manejado en la versión anterior. Aunque no se incrementaron requerimientos ni ampliaron las funcionalidades, se hizo un proceso de depuración de modelo el cual permitió facilitar los procesos de implementación, mantenimiento y mejoró la organización del modelo de referencia. Los principales cambios fueron:

- Cambio de la palabra *Courseware* en el título de la especificación a *Content*, para reflejar el hecho que la especificación aplicaba en diferentes niveles de los componentes de un “curso” y no sólo al “curso” en su totalidad.
- Se redujo un número significativo de elementos definidos en el formato de la estructura de un curso y en la especificación del modelo de datos CMI, *Computer Managed Instruction* [31]. Esto basado en la no utilización de algunos elementos opcionales en la especificación original, y que generó un entorno de características básicas, anticipándose a modelos de datos más robustos que se encontraban bajo desarrollo de diferentes grupos de estandarización.
- Se estructura la especificación en grupos funcionales, cada uno manteniendo sus propias sub-secciones.
- Cambios en el API del entorno de ejecución que mejoraban el rendimiento, esto implicó cambios en las implementaciones de la especificación de la versión anterior.

Versión 1.2: Octubre de 2001, la versión 1.2 mantiene los objetivos y contexto de las versiones anteriores. Se hacen ajustes y correcciones sobre la versión anterior. El principal cambio es la adopción e integración de las especificaciones para el empaquetamiento del contenido y la descripción, usando meta-datos, para los recursos de aprendizaje definidos por la IMS.

Versión 1.3: Conocida también como SCORM 2004. A diferencia de versiones anteriores el contexto de trabajo se hace más amplio, así como las capacidades y características. Dentro de las nuevas características [34], se destacan:

- Permite a los desarrolladores agregar reglas al documento que describe el paquete de aprendizaje para determinar la forma exacta en que se debe presentar el contenido al estudiante. Por ejemplo, el autor del paquete puede determinar que para acceder a algún contenido es necesario haber completado otro y obtener un puntaje mínimo en alguna prueba dentro del mismo.
- Amplía y modifica los elementos del modelo de datos del entorno de ejecución para estandarizar y mejorar la interoperabilidad entre sistemas.

1.2.2 SCORM y la WEB

El estándar SCORM asume una infraestructura web como base para su implementación técnica, argumentada bajo las siguientes:

- La infraestructura y las tecnologías enfocadas a la web se expanden rápidamente y proveen de un entorno base que favorece el desarrollo de nuevas tecnologías de aprendizaje y conocimiento.
- Los estándares para tecnologías de aprendizaje enfocados en la web no se encuentran ampliamente difundidos.
- El contenido web puede ser distribuido usando casi cualquier medio, por ejemplo en CD-ROM, dentro de sistemas personales e independientes (*stand-alone*) o en entornos basados en red.

Esta aproximación adopta la transición de la industria hacia la distribución de contenido común o estandarizado y de formatos abiertos. Los sistemas operativos actuales interpretan y dar soporte de forma nativa al contenido web en la mayoría de sus formatos. El camino y objetivo común es el de extender las tecnologías de aprendizaje para que puedan usarse de forma nativa, sea localmente, mediante redes locales o sobre el Internet [31].

1.2.3 SCORM y la trazabilidad del estudiante

Las características de trazabilidad tradicionales en los sistemas de entrenamiento basado en computador CBT, *Computer Based Training*, ofrecen las bases pedagógicas para creación de entornos de aprendizaje adaptativos. Históricamente, CMI, ha ofrecido a los sistemas CBT las capacidades de realizar procesos de trazabilidad eliminando las necesidades de herramientas y estrategias propietarias. En el caso de los LMS la trazabilidad se debe realizar sobre el progreso del estudiante dentro del proceso de formación y sobre el dominio del conocimiento adquirido. Esto implica recolectar información del perfil del estudiante, entregar el contenido de aprendizaje, monitorear las interacciones claves entre el contenido y el estudiante y el rendimiento que muestra el estudiante una vez ha asimilado el contenido y finalmente determinar cuál debe ser la siguiente experiencia de aprendizaje [31].

La creación de objetos de aprendizaje con características de trazabilidad implica la definición de un modelo estándar de la información que va a ser trazable. Este modelo estándar es conocido dentro de SCORM como el modelo de datos del entorno de ejecución, en la actualidad se sigue el modelo de datos CMI propuesto por la AICC.

Los elementos del modelo CMI están divididos en dos categorías: obligatorios y opcionales. Los elementos obligatorios deben ser implementados en su totalidad, y conforme a la especificación, por los LMS que soporten el estándar SCORM. Sin embargo, los objetos de aprendizaje son libres de usar cualquier tipo de elemento dentro del modelo de datos CMI, el único requerimiento es el de informar al LMS cuando es inicializado o terminado.

1.2.4 Paquetes de aprendizaje

SCORM es un estándar de paquetes de objetos de aprendizaje reutilizables. Los objetos de aprendizaje, o SCO, *Sharable Content Object*, son pequeñas unidades de aprendizaje en un soporte digital como por ejemplo páginas web, animaciones de Flash, multimedia, applets de Java, etc. Y un paquete no es otra cosa que una serie de objetos de aprendizaje juntos [31]. El SCO se define como un objeto que permite a un aprendiz lograr unos objetivos de aprendizaje, pero no impone requisitos sobre la forma en que se deben lograr los objetivos.

La dinámica en la creación de un paquete de aprendizaje SCORM básico es la siguiente [33, 35]:

- El proceso se inicia con la creación de contenidos de aprendizaje.
- El autor de los contenidos define su estructura para facilitar el proceso de aprendizaje. SCORM no especifica la naturaleza del contenido, es decisión de

cada autor, el definir la forma y proceso mediante el cual se hace llegar el conocimiento deseado al estudiante. El proceso puede ser mediante lecturas, juegos, videos, etc.

- El autor agrupa en un único fichero los recursos digitales (páginas web, hojas de estilos, scripts, imágenes, *applets*, animaciones flash, etc.) que al ser ejecutados en el navegador crean el entorno de presentación del contenido. El paquete creado es un archivo comprimido en formato .zip.
- El paquete debe incluir un documento que describe los diferentes recursos que forman el objeto de aprendizaje, la secuencia en la cual se deben presentar los contenidos y los meta-datos que describen de una forma semántica el objeto de aprendizaje. En SCORM este documento se identifica con el nombre *imsmanifest.xml*.
- El paquete se deja en un repositorio (es importante la idea de compartirlos) o bien se distribuye por la red para su uso.

1.2.5 Núcleo de SCORM

SCORM tiene sus bases en dos partes fundamentales que permiten definir, de forma general, la forma en que se construye el objeto de aprendizaje y el proceso de comunicación e interacción con el LMS [31].

El Modelo de Agregación de Contenidos (*Content Aggregation Model*), describe los métodos y procedimientos necesarios para almacenar, identificar, intercambiar y de recuperar los contenidos de aprendizaje. Este modelo representa un medio pedagógicamente neutral para que los diseñadores y las herramientas de enseñanza, desarrollen recursos de aprendizaje con el objetivo de entregar una experiencia de aprendizaje deseada. Una **experiencia de aprendizaje** consiste en actividades que están soportadas por recursos de aprendizaje electrónico o no-electrónicos.

Una de las actividades en el proceso de creación y entrega de experiencias de aprendizaje involucra la creación, descubrimiento y unión, o agregación, de simples elementos electrónicos (texto, imágenes, sonidos, páginas web HTML) en recursos de aprendizaje más complejos, y después organizar estos recursos en una secuencia predefinida y distribuible [31]. Este modelo se basa en los siguientes:

- **Modelo de Contenido:** Nomenclatura que define los componentes de contenido para una experiencia de aprendizaje.
- **Metadatos:** Un mecanismo para describir instancias específicas de los componentes del modelo de contenido.

- Empaquetamiento del contenido: Define cómo representar el comportamiento propuesto para una experiencia de aprendizaje (estructura del contenido) y como empaquetar los recursos de aprendizaje para su uso en diferentes ambientes (empaquetamiento del contenido)

El Entorno de Ejecución (*Run-Time Enviroment*), describe las exigencias que debe implementar el LMS para que pueda gestionar el entorno de ejecución con el contenido SCORM. Para lograr la retroalimentación y control de una experiencia de aprendizaje, es necesaria una comunicación normalizada entre los contenidos (sobre los que interactúan los participantes) y la plataforma LMS. El entorno de ejecución formaliza y establece el envío de los datos e información en ambos sentidos de manera estandarizada y compatible con las tecnologías de Internet. Para ello, se ha definido un API, *Application Program Interface*, en Javascript, que suministra un mecanismo estándar de comunicación con un LMS, independientemente de la herramienta utilizada para desarrollar el contenido [31].

1.3 Streaming

Muchas de las definiciones que se encuentran en distintas fuentes de información hacen una estrecha relación del *streaming* con los contenidos multimedia, así, la mejor definición que se puede encontrar traduce “*Streaming*: proceso de proporcionar un flujo constante de audio o de datos de vídeo de modo que un usuario de Internet pueda tener acceso a éste mientras se descarga” [9]. Sin embargo, si se tienen en cuenta diferentes desarrollos e investigaciones alrededor de este tema, el concepto tradicional queda limitado.

Para contextualizar el significado de *streaming* en el presente documento, se propone el siguiente concepto.

1.3.1 Definición

El *streaming* es una tecnología que permite transmitir un flujo constante de información de un recurso digital (audio, video, mensaje, software, archivo, etc.), de tal forma que la información recibida puede interpretarse en el otro extremo, es decir, tiene información suficiente para su procesamiento parcial, sin la necesidad de esperar la transmisión completa del recurso digital.

Algunas aplicaciones comunes se observan en los servicios de *streaming* de video, audio, *streaming* de imágenes [10, 36], *streaming* de software [11], *streaming* de mensajes, este último usado generalmente en los juegos multi-jugador en red [37].

1.3.2 Características de la tecnología *streaming*

1.3.2.1 Arquitectura

El procedimiento para intercambiar información relacionada con la tecnología *streaming*, en general, sigue un modelo cliente servidor.

Los recursos son elementos digitales como videos, sonidos, mensajes, etc. Los cuales son distribuidos de forma transparente por aplicaciones software que se ejecutan del lado del servidor (Servicios Web, EJB, Servlets, etc.) [38].

1.3.2.2 Modos de transmisión

Tradicionalmente la tecnología de *streaming*, en redes IP, hace uso del modo de transmisión ***unicast***, donde un recurso es enviado a un único destinatario por cada petición, esto implica que el ancho de banda necesario para cubrir un grupo de clientes es proporcional al número de clientes que hacen la petición del recurso al servidor [39].

El modo de transmisión ***multicast*** se usa cuando un recurso se puede enviar a un grupo de clientes al mismo tiempo. Su ventaja más importante es el ahorro del ancho de banda necesario para que los clientes tengan acceso a la transmisión, sin embargo, esta ventaja implica una limitante en la interactividad que cada cliente percibe, debido a que la transmisión es asociada al grupo y no a sus participantes.

El modo *multicast* ofrece beneficios sólo a un grupo de servicios de *streaming* bajo necesidades puntuales, además, es necesario el uso de equipos de infraestructura especializados (*routers multicast*) [40].

1.3.2.3 ¿Cómo trabaja el *streaming*?

La estrategia usada por la tecnología *streaming* consiste en la creación de un *buffer* del lado del cliente para almacenar la información parcial a medida que llega, esto implica una serie de sincronizaciones entre la información del *buffer*, la información que se procesa parcialmente desde el *buffer* y la información que se descarga progresivamente para llenar nuevamente el *buffer*. Esta estrategia es transparente para el cliente, variando de acuerdo al tipo de servicio de *streaming* y tecnología que se use.

El *buffer* ofrece diferentes ventajas dependiendo del servicio de *streaming*, por ejemplo, en el caso de un servicio de *streaming* de video el buffer habilita un mecanismo de protección en caso de pérdida de la conexión entre el cliente y el servidor, o si reinicia la reproducción del video después de una pausa, garantizando en cierta medida la reproducción continua del video mientras se restablece la comunicación [38].

Hoy en día es común ver que diferentes empresas promuevan nuevos servicios en donde se permite ver archivos multimedia a través de Internet. Por esto es importante aclarar la diferencia existente entre el envío de archivos a través de *streaming* y la descarga de archivos multimedia por Internet.

La descarga de archivos multimedia usando servidores Web es llamado envío progresivo de archivos multimedia o de datos por HTTP, este tipo de envíos no es *streaming* del todo, pues simplemente se está transmitiendo la totalidad del recurso en un solo envío hacia el dispositivo destino o partes del mismo, pero manteniendo una conexión y descarga constante independiente de si se reproduce o no [41].

En el anexo 1 se presenta un listado de los protocolos utilizados para la tecnología de *streaming*. Se destaca la atención sobre el protocolo RTMP, asociado a la tecnología Flash, debido a que en los capítulos siguientes se hace constante mención a este.

1.3.3 Aplicaciones y servicios de *streaming*

La tecnología de *streaming* ofrece la posibilidad de desarrollar diferentes aplicaciones o servicios con características particulares de acuerdo a las necesidades del entorno de aplicación. En algunos casos es necesaria la definición de una infraestructura de soporte especial que ofrezca las facilidades necesarias para el correcto funcionamiento del servicio.

1.3.3.1 Video bajo demanda

Desde sus inicios, la tecnología de *streaming* fue concebida y desarrollada como solución a este servicio, su mejora y evolución todavía se desarrolla.

El video bajo demanda o VOD, “*video on demand*”, tiene el objetivo de permitir a los usuarios del servicio la reproducción en tiempo real de un contenido audio visual, es decir, una vez se hace la petición de un video en particular, se inicia la reproducción sin necesidad de esperar la transferencia total del mismo. El video puede ser un recurso almacenado o puede provenir de una transmisión en vivo y en directo.

El servicio VOD puede ser implementado sin la necesidad de usar *streaming*, en este caso, la forma en que se distribuye el recurso audio-visual se conoce como descarga

progresiva. La descarga progresiva se enfoca en la transmisión de recursos multimedia, permitiendo simular algunas características del *streaming* como la habilidad de avanzar o retroceder a medida que este se reproduce, un servicio de video bajo demanda reconocido, que implementa esta estrategia de distribución se encuentra en YouTube. A diferencia del *streaming*, la descarga progresiva almacena el recurso multimedia de forma temporal, esto implica por ejemplo, que al pausar la reproducción del recurso, internamente se continúa la descarga hasta completar la totalidad del archivo multimedia. Citando el mismo ejemplo mediante el uso de *streaming*, al pausar la reproducción solo se continúa la descarga del recurso hasta llenar el buffer del lado del cliente, usualmente de unos pocos segundos, después de esto no se consumen más recursos de red.

1.3.3.2 Video con señal en vivo

Es un caso particular del servicio de VOD, el servicio con señal en vivo permite a los usuarios la reproducción en tiempo de real de un recurso que se genera al mismo instante. El concepto de calidad del servicio en este caso se puede ver, de forma general [42], desde dos puntos de vista:

- a. La calidad del recurso audio-visual, basado en parámetros como resolución, tasa de bits, fluidez de la reproducción, entre otros.
- b. La calidad del recurso audio-visual, basada en la realidad temporal del instante observado con relación al origen de la transmisión, en otras palabras, cuando tiempo ha transcurrido entre lo que se está observando y el momento real en que ocurrió.

En el primer caso la calidad del servicio depende directamente de las características físicas y tecnológicas bajo las cuales se implementa, entre estas tipo de códec de video, ancho de banda para la distribución del contenido, velocidad de las conexiones, entre otras; en el segundo caso existen parámetros adicionales como la duración de las fases de pre-procesamiento del origen de datos, pre-codificación o edición, así como la velocidad y ancho de banda de los canales de publicación del contenido hacia el servidor de distribución [43].

Lo anterior demuestra la existencia de condiciones particulares asociadas a los servicios de *streaming* [43].

La aplicación más conocida de este tipo de servicio es la **videoconferencia**, la cual puede ser de dos tipos; punto a punto o multipunto. En la videoconferencia punto a punto, solo dos nodos están conectados y pueden interactuar de forma asíncrona, análogo a una

llamada telefónica convencional. En el caso de la video conferencia multi-punto, se conectan e interactúan tres o más nodos; debido a las implicaciones en la sincronización durante la participación de los nodos se establecen estrategias de moderación.

En los servicios de video *streaming* los *codecs* juegan un papel sumamente importante, en la sección 1.3.5 se presentan los más relevantes asociados a la distribución sobre Internet.

1.3.3.3 Audio bajo demanda y en vivo

Tiene las mismas características que los servicios de video bajo demanda y en vivo, sin la transmisión del canal de video. Estos servicios son muy comunes en las transmisiones que realizan emisoras que tienen presencia en internet.

Debe aclararse la diferencia entre audio bajo demanda, basado en *streaming* y el audio bajo demanda, conocido regularmente como *PodCast*, en el segundo, el audio es descargado de forma completa para luego reproducirse en equipos portátiles como MP3, iPod, etc. También es posible implementar el *PodCast* usando estrategias de descarga progresiva y permitir la reproducción similar a una transmisión de *streaming*.

1.3.3.4 Aplicaciones colaborativas

Las aplicaciones colaborativas permiten que varios usuarios puedan trabajar en una tarea común. Aunque es posible implementar estas aplicaciones con otras tecnologías, mediante el *streaming* se logran servicios más livianos, con respuestas en el tiempo más eficiente, debido a la no necesidad de invocaciones del tipo “Petición/Respuesta” y a la definición de conexiones persistentes que mantienen un canal para la transmisión de la información en todo momento.

Dentro de las aplicaciones colaborativas que se implementan mediante *streaming* se encuentran los chats, tableros compartidos y juegos multi-jugador [37].

1.3.3.5 RPC Streaming

Este tipo de aplicación se encuentra implementada actualmente sobre la plataforma Flash. Ofrece capacidades del lado del servidor de aceptar llamadas a diferentes funciones de forma paralela y asíncrona, esto implica que el cliente que realiza los llamados RPC también implementa una lógica de negocio asíncrona. La figura 3 muestra un ejemplo de la comunicación presente en este tipo de aplicación.

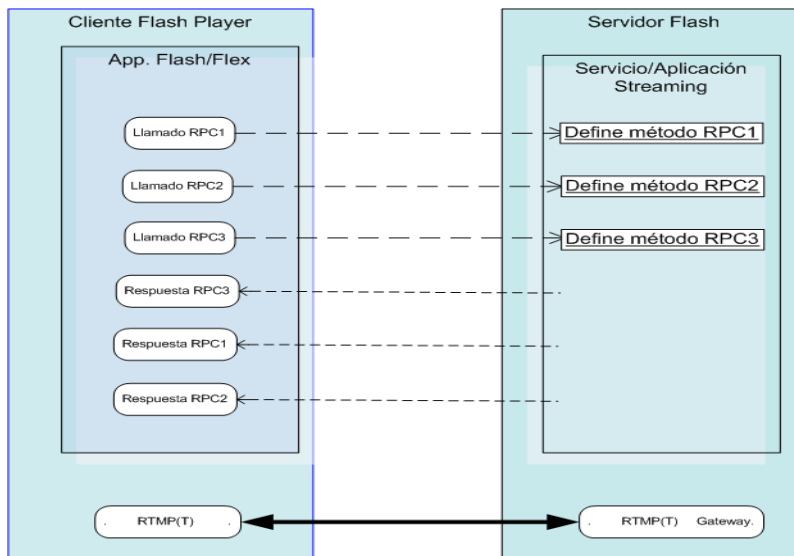


Figura 3. RPC streaming, respuestas asíncronas

1.3.3.6 Callbacks o Tecnología Push

Implementada en la actualidad sobre la plataforma Flash, las aplicaciones que hacen uso de esta tecnología pueden realizar llamados a métodos del lado del cliente de forma asíncrona. Mientras que del lado del servidor alguna aplicación de *streaming* se encuentre en ejecución, esta puede realizar llamadas sobre cualquier cliente que tenga una conexión activa con el servidor. Este tipo de aplicación se usa comúnmente para informar continuamente al cliente sobre el estado de algún proceso, en general se entiende que a una petición pueden corresponder muchas respuestas. La figura 4 muestra un patrón común de intercambio de mensajes asociado a este tipo de tecnología.

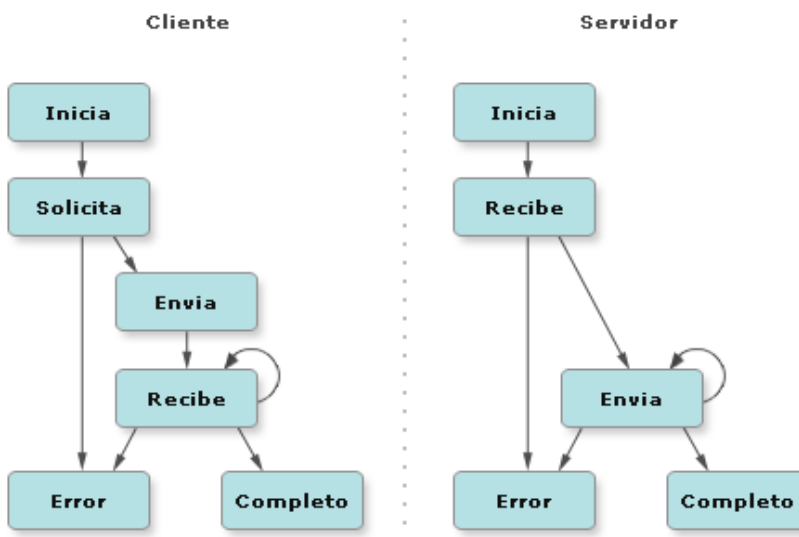


Figura 4. Diagrama de transición de estados para servicios con callbacks

1.3.3.7 *Shared Object*, objetos compartidos

Exclusivo de la plataforma Flash, ofrece capacidades de intercambio en tiempo real entre diferentes clientes conectados al servidor. Los objetos compartidos permiten que todos los clientes compartan información de forma transparente y en tiempo real, usualmente uno de ellos es el publicador de la información y los restantes son los suscriptores del objeto. Esta tecnología hace uso extensivo de invocaciones RPC y *callbacks* para mantener actualizada la información en cada cliente.

1.3.4 Tecnologías para hacer *streaming*

Existen cuatro tecnologías principales y populares que ofrecen servicios de *streaming* dirigidos a un ambiente cuya fuente es el Internet. Las tecnologías basadas en QuickTime, Windows Media y RealMedia, se enfocan únicamente a servicios de video y audio *streaming*; La tecnología basada en Adobe Flash, además de ofrecer servicios de *streaming* de video y audio, ofrece herramientas avanzadas que permiten ofrecer nuevos servicios, pudiendo controlar que tipo de información es transportada por los canales de *streaming* y ofreciendo una capa de aplicación con amplias opciones de desarrollo.

A continuación se analizan las tecnologías, teniendo en cuenta tres componentes importantes: El cliente, el servidor de *streaming* y los mecanismos de comunicación entre los dos. Finalmente se presentan las ventajas y desventajas asociadas a la tecnología en particular.

1.3.4.1 Tecnología basada en QuickTime

Cliente: Del lado del cliente se necesita el reproductor QuickTime, incluido de forma nativa en sistemas operativos Mac-OS (versión QuickTime X), existe una versión oficial para el sistema operativo Windows (versión QuickTime 7.6) [44] y versiones no oficiales de código abierto para sistema operativo Unix/ Linux (versión QuickTime 7.6).

Servidor: Existen 2 tipos de servidores dirigidos a esta tecnología:

- a. **QuickTime Streaming Server, QTSS [45].** Servidor propietario, desarrollado por Apple, se comercializa principalmente sobre plataformas Mac-OS Server. Soporta los formatos de QuickTime basados en el estándar H.263 y versiones MPEG-1 y MPEG-2.
- b. **Darwin Streaming Server, DSS [46].** Servidor de código abierto bajo licencia APSL [47], desarrollado por Apple, funcionalmente similar a QTSS, sin embargo se trata de un servidor multiplataforma (Mac-OS, Windows, Linux).

Medio de Comunicación: La tecnología QuickTime se basa en el uso de protocolos HTTP, RTSP y RTP. El recurso de *streaming* se accede de forma natural en los ambientes de internet mediante el protocolo HTTP, una vez, se descargan las cabeceras e inicia el servicio de video *streaming* se hace uso de los protocolos RTP y RTSP para transporte y control.

Ventajas:

- QuickTime hace gestión de credenciales para el control de acceso al contenido de forma nativa mediante los dos servidores. La información de sesión es codificada mediante estándares criptográficos fuertes.
- Mediante las funcionalidades multicapa, se ofrece la posibilidad de incluir canales con información textual en los flujos de video, dirigidos al uso de subtítulos en diferentes lenguajes.
- Soluciones de bajo costo, gracias a las tecnologías de código abierto.

Desventajas:

- La tecnología no realiza procesos de gestión de ancho de banda, de tal forma que se deben crear y ofrecer manualmente contenidos para diferentes velocidades de conexión de los clientes.
- Requiere que los recursos a transmitir cuenten con cabeceras especiales "*hint track*", las cuales se generan mediante codificadores especiales y comerciales, lo que implica sobre costos relacionados de forma indirecta.
- No se puede asegurar un mercado general, debido a la necesidad de instalación del reproductor QuickTime del lado del cliente.
- La calidad de los recursos basados en QuickTime es inferior a los recursos ofrecidos por las tecnologías de la competencia.
- No se han implementado cambios relevantes que mejoren la calidad de la tecnología desde el año 2001.
- Se limita al *streaming* de audio y de video, sin la posibilidad de crear nuevos servicios sobre la propia plataforma de *streaming*.
- El reproductor no se integra, ni interactúa con otras tecnologías del lado del cliente de forma bidireccional (HTML puede definir elementos para cargar el reproductor, sin embargo, el reproductor no puede comunicarse con el interprete de HTML).

1.3.4.2 Tecnología basada en RealMedia

Cliente: Del lado del cliente se necesita el reproductor RealPlayer [48], existe una versión oficial para el sistema operativo Windows y Mac-OS versiones no oficiales de código abierto para sistema operativo Unix/ Linux (versión QuickTime 7.6). Adicionalmente,

existen implementaciones de código abierto oficiales de RealNetwork del reproductor Helix DNA [49], compatible con las tecnologías existentes asociadas a RealMedia. Dadas las ventajas del cliente Helix DNA, en su mayoría por el código abierto, ha sido implementado en diferentes ambientes y contextos, por ejemplo en equipos de escritorio (todas las arquitecturas y sistemas operativos), ambientes móviles y en equipos enfocados al entretenimiento en el hogar, como reproductores de DVD, teatros en casa, televisores, etc.

Servidor: *RealNetworks* ofrece dos tipos de soluciones, la primera conocida como Helix DNA Server [50], se trata de un desarrollo de código abierto bajo licencia RPSL [51], multiplataforma (Linux y Windows), que busca convertirse en una plataforma de distribución universal para diferentes formatos y *codecs* de audio/video. Actualmente tiene la capacidad de distribuir contenido multimedia MP3 y formatos RealMedia (.rm, .ra, .rv), con la capacidad de extender su funcionalidad mediante *plugins* (más servicios, más formatos). La segunda opción se basa en un producto comercial conocido como Helix Server, al igual que Helix DNA es un sistema multiplataforma, con soporte para formatos convencionales como RealMedia y adicionalmente compatible con un número elevado de formatos de video y audio como 3GPP (H.263 y H.264), Windows Media, QuickTime, MP3, ACC. El sistema de licenciamiento se basa y limita en el número de clientes concurrentes que puede manejar el servidor. Su principal ventaja radica en las tecnologías relacionadas que permiten mejorar el desempeño del sistema basado en la escalabilidad y disponibilidad de los servicios.

Medio de Comunicación: La tecnología RealMedia hace uso de protocolos como HTTP, RTSP, RTP, RSTP enmascarado sobre HTTP, usando los protocolos de transporte TCP y UDP (unicast y multicast). El recurso de *streaming* se accede de forma natural en los ambientes de internet mediante el protocolo HTTP, una vez, se descargan las cabeceras e inicia el servicio de video/audio *streaming* se hace uso de los protocolos RTP y RTSP para transporte y control.

Ventajas:

- Los servidores soportan un gran número de formatos de video y audio [52], sin limitarse a los formatos de su propia tecnología.
- Los formatos de audio y video de esta tecnología ofrece calidades similares, en algunos casos mejores, que otros formatos en el mercado.
- El mercado asociado a esta tecnología va más allá de los equipos de cómputo tradicionales, pudiendo usar equipos móviles o aparatos electrodomésticos como clientes potenciales.

- Soporta diferentes mecanismos de autenticación para la protección de los recursos audio visuales.
- Ofrece servicios adicionales a los de audio y video, asociados principalmente al control en la reproducción de los mismos, su administración y percepción (Metadatos, vistas previas, listas de reproducción, entre otros).
- Herramientas de desarrollo avanzadas (APIs, SDKs) para la creación de nuevas funcionalidades y servicios.
- Existen herramientas de codificación libre que facilita la creación de recursos para el trabajo óptimo de la tecnología.

Desventajas:

- Aunque parte de la tecnología tiene acceso al código fuente, se debe respetar con las licencias y derechos de autor que en algunos casos implica costos extras.
- El cliente de reproducción se limita al acceso a servicios de audio y video *streaming*.
- El cliente de reproducción debe ser instalado en los equipos de cómputo tradicionales.
- El cliente no permite la comunicación bidireccional con tecnologías como HTML.

1.3.4.3 Tecnología basada en Windows Media

Cliente: Del lado del cliente se necesita el reproductor Windows Media Player, existen versiones oficiales que sólo se incluyen con las diferentes versiones del sistema operativo Windows. Existen versiones no oficiales de código abierto para sistemas operativos Linux, Mac-OS. El reproductor y los formatos de audio/video están protegidos por derechos de autor y sistemas de patente que prohíben su uso en sistemas de carácter comercial sin el pago de la licencia correspondiente. Microsoft ha desarrollado la tecnología *SilverLight* [53] que da soporte y sirve como cliente, multiplataforma (Windows, Linux, Mac-OS), para la tecnología basada en Windows Media. Desarrollado a manera de *plugin*, interactúa con la mayoría de navegadores web.

Servidor: En la actualidad Microsoft ofrece una plataforma para realizar *streaming* de audio y video denominada Microsoft Windows Media Services [54]. Se trata de una plataforma de altas prestaciones, enfocada a un entorno empresarial con puntos fuertes en aspectos como la escalabilidad, mecanismos de proxy/cache, administración flexible, distribución del contenido mediante mecanismos seguros y extensibilidad basada en *plugins*. Se enfoca principalmente en el *streaming* de formatos Windows Media y MP3, aunque se pueden comprar *plugins* para realizar *streaming* de otro tipo de formatos.

Medio de Comunicación: La tecnología Windows Media hace uso de protocolos como HTTP, RTSP, RTP, MMS, MMS enmascarado sobre HTTP, usando los protocolos de transporte TCP y UDP (*unicast* y *multicast*). El recurso de *streaming* se accede de forma natural en los ambientes de internet mediante el protocolo HTTP o MMS, una vez, se descargan las cabeceras e inicia el servicio de video/audio *streaming* se realiza un proceso similar al explicado en el literal 1.3.3.12 del anexo 1.

Ventajas:

- El cliente se encuentra pre instalado en el sistema operativo Windows, por lo que existe una alta penetración en el mercado actual.
- Los formatos de Windows Media ofrecen calidad similar o superior a otros formatos de audio/video del mercado, sin embargo su mayor ventaja radica en el rendimiento que se obtiene durante la codificación y decodificación de los mismos, pudiéndose realizar codificación en tiempo real en computadores comunes de calidad estándar.
- Microsoft distribuye oficialmente software de codificación de manera gratuita.
- Mucha experiencia y desarrollo en el área del *streaming* de audio y video.

Desventajas:

- Servicios de *streaming* solamente de audio y video.
- Formatos de audio/video que necesitan de licencias para servicios comerciales.
- Los servicios ofrecidos se enfocan a plataformas Windows.
- El cliente no permite comunicación bidireccional con entornos de ejecución que tienen relación con la especificación HTML, por ejemplo Javascript.

1.3.4.4 Tecnología basada en Flash

Cliente: Del lado del cliente se necesita instalar la versión del reproductor Flash Player. Este reproductor es desarrollado por Adobe Inc., es distribuido de forma oficial y gratuita para todos los sistemas operativos de la actualidad [55]. El reproductor sirve como entorno de ejecución para las aplicaciones desarrolladas mediante dos tecnologías. La primera conocida como Flash está orientada como un producto comercial que necesita de una licencia para su uso. Por otra parte, existe la tecnología Flex [56, 57], la cual se basa en software libre y ofrece las mismas funcionalidades que Flash. A diferencia de las tecnologías tradicionales el reproductor Flash Player ofrece un gran número de herramientas para desarrollar diferentes aplicaciones simples o avanzadas que usen desde simples animaciones hasta aplicaciones en tiempo real con llamados RPC,

servicios web o aplicaciones interactivas que hacen uso de periféricos como cámaras web y/o micrófono. Así mismo, integra en un solo paquete un conjunto de *codecs* de audio y video para las aplicaciones que incluyan *streaming*.

Servidor: La tecnología Flash [58] actualmente cuenta con diversas plataformas de desarrollo entre las más destacadas se encuentran *Flash Media Server* [59], FSM, Wowza Media Server Pro [60] y Red5 [61]. De las tres plataformas, Red5 es la única basada en software libre, desarrollada íntegramente en Java, mediante tecnología de *servlets* y soportada principalmente por el *framework* de desarrollo Spring. Red5 se encuentra en su versión de desarrollo 0.9, no calificada como Beta aún; implementa la especificación oficial AMF [62] para la representación de los datos y una especificación propia del protocolo RTMP, RTMPT y RTMPS para el transporte de los flujos de información.

Medio de Comunicación: La tecnología Flash hace uso de protocolos como HTTP, RTMP, RTMPT, RTMPS, usando únicamente el protocolo de transporte TCP. El recurso de *streaming* se accede de forma natural en los ambientes de internet mediante el protocolo HTTP o mediante una conexión RTMP, en su forma más general, y un procedimiento especial que indica al servidor el recurso solicitado.

Ventajas:

- Existen implementaciones oficiales del cliente para la mayoría de sistemas operativos y según [63] la penetración del mismo en el mercado de equipos en internet es casi total.
- Ofrece herramientas que permiten aprovechar el potencial de la tecnología *streaming* y desarrollar nuevos servicios.
- Tecnología libre que no impone requisitos a la hora de desarrollar, compilar y desplegar las aplicaciones. De igual manera del lado del cliente es posible utilizar los *codecs* propietarios, por ejemplo VP6 [64] para el video y NellyMoser [65] para el audio, sin preocuparse por incumplimiento de licencias¹.
- Las aplicaciones para audio y video *streaming* están preparadas para contenidos con alta definición.

Desventajas:

¹ En el caso de video, solo aplica para reproducción, en el caso de audio solo para captura desde el micrófono y transmisión hasta el servidor. El almacenamiento y distribución incumplen algunos derechos de autor y propiedad intelectual, requiriendo de licencias individuales especiales.

- Los formatos de video requieren de mayor procesamiento, si se comparan con otros de tecnologías como Windows Media o RealMedia.
- La creación de recursos audio/visuales está sujeta de licencias comerciales si se hacen uso de algunos *codecs*.
- En el caso de las plataformas comerciales el costo es elevado, en las plataformas libres, debido a su poca madurez, el soporte es limitado y la documentación escasa y en algunos casos desactualizada.
- En Red5, específicamente, la curva de aprendizaje es lenta dado que en la actualidad no existe una versión “*Release Candidate*” que formalice un API de desarrollo, ocasionando que en cada versión o publicación las aplicaciones desarrolladas pierdan su compatibilidad.

La tabla 1 lista algunos factores importantes que se deben tener en cuenta a la hora de evaluar las tecnologías.

Tecnologías	Quick Time	Real Media	Windows Media	Flash
Ofrece servicios adicionales al audio y video?	NO	NO	NO	SI
¿Servicios de streaming bloqueados por firewall?	SI	SI	SI	NO
¿Disponibilidad de herramientas open source (Entornos de desarrollo SDK, cliente de reproducción)?	SI	SI	SI	SI
¿Cliente de consumo de servicios desde el navegador?	SI	SI	SI	SI
¿Permite interactuar mediante Javascript con el contenido del navegador?	NO	NO	NO	SI
¿Requiere de codecs adicionales para la reproducción de audio/video o integra sus propios codecs de forma nativa?	SI	SI	SI	NO
¿Preparado para alta definición?	NO	SI	SI	SI
¿Permite el uso de cámaras web y/o micrófono para la publicación de flujos de video/audio de forma nativa?	NO	NO	NO	SI

Tabla 1 Factores comparativos útiles para la selección de la tecnología.

Capítulo 2. Lineamientos para la integración de servicios de *streaming* en sistemas de aprendizaje de código abierto

2.1 Contextualización

Cuando hablamos de integración de servicios de *streaming* a un LMS, nos referimos a la introducción de servicios existentes o nuevos servicios, los dos basados en tecnología *streaming*, a una plataforma que ya se encuentra implementada bajo criterios tecnológicos particulares, que hace uso de un modelo pedagógico educativo y que tiene sus propias herramientas para extender su funcionalidad.

Dadas las tendencias actuales, donde la información y los datos son integrados y presentados para mejorar la experiencia del usuario, se destaca la importancia de integrar los diferentes mecanismos de comunicación dentro de los ambientes virtuales de educación, esto es, permitir a los estudiantes tener a su disponibilidad los canales de distribución del material de formación y las herramientas de interacción, que le permitan disfrutar de una actividad formativa integral, sin tener que interactuar con servicios individuales (foro, chat, descargas), cuya relación está dada por factores explícitos al diseño del sistema LMS.

Para definir los lineamientos de integración es necesario contextualizar los servicios de *streaming* en el marco del *e-learning*, con ello se pretende que dichos servicios tengan un impacto positivo dentro de la estrategia educativa establecida con el LMS. Así pues, en el presente capítulo se presentan un conjunto de lineamientos agrupados en dos categorías, los primeros de carácter metodológico que permiten analizar los servicios en el entorno del LMS, las interacciones, las tecnologías asociadas a los servicios y la manera en que se debería hacer la selección de la tecnología. La segunda categoría de lineamientos agrupa un conjunto de recomendaciones tecnológicas, que describen los factores importantes a tener en cuenta para la integración de los servicios en un LMS.

El término material fue seleccionado para describir lo que se está proporcionando por medio del servicio dentro de la plataforma LMS, es decir, hace referencia a la información

y datos que ofrece, no se debe generalizar el material a un recurso particular como lectura, video, audio, juego, etc. Además, caracterizando el material como un elemento pasivo, es decir, que no es conocimiento a menos que el estudiante haga algo con él.

2.2 M1: Modelo de dinámica del material

El diseño de servicios y material de aprendizaje, ha sido estudiado desde hace mucho tiempo, alrededor de este tema se han presentado diferentes teorías y modelos que sirven de guía para el desarrollo de actividades de aprendizaje que efectivamente cumplan los objetivos de enseñanza, dichos modelos y teorías se enmarcan en lo que se conoce como el diseño instruccional [66, 67]. Dentro de los diferentes modelos y teorías presentados en los estudios del diseño instruccional, Bates y Taylor [68, 69] sugieren que el proceso de aprendizaje puede ser descrito como interacciones entre los diferentes actores el proceso formativo, inclusive el contenido como eje central. Así pues, para determinar las características que los servicios de *streaming*, y en general servicios de diferente naturaleza, deben tener en cuenta para lograr una integración efectiva dentro del LMS, se propone un modelo, que agrupa y propone diferentes características [68-72], con el objetivo de describir y analizar las interacciones del material, ofrecido por cualquier servicio dentro del LMS y así plantear, en etapas posteriores, los lineamientos que tengan en cuenta estos atributos.

Las interacciones más relevantes que se van a tener en cuenta para describir la dinámica en el proceso formativo, están definidas en las siguientes:

- Entrega del material
- Interacción con el material
- Interacción con el profesor
- Interacción entre estudiantes
- Interacción de calidad

2.2.1 Entrega del material

Partimos de la identificación del concepto de entrega del material, en adelante EM, en el ambiente formativo presencial tradicional, como la información relevante que adquieren los estudiantes durante la sesión formativa con el profesor. En este sentido, el material es entregado mediante las palabras del profesor o las palabras escritas en un tablero o diapositivas, pero es asimilado y almacenado principalmente en los recuerdos y en lo que

pueda tomar nota. Por otro lado, en el ambiente virtual se observa una clara diferencia ya que los estudiantes interactúan con “grabaciones” de lo que intenta transmitir el profesor, es decir que en lo virtual la interacción se da con materiales (lecturas, resúmenes, diapositivas, video grabaciones, etc.) y se limita a lo que el material permita realizar. Hay que aclarar que existen diferentes canales que permiten complementar las interacciones, por ejemplo foros, chats, etc. Pero siguen manteniéndose como canales independientes.

El material en un ambiente virtual puede ser entregado de diferentes formas, por ejemplo:

- La descarga de una o varias imágenes.
- La presentación de un texto digital.
- La descarga de un documento de texto.
- La reproducción de un recurso audio/visual.
- La presentación de un concepto o apreciación en un ítem en un foro, etc.

Los servicios pueden o no tener en cuenta este atributo, pueden existir servicios que ofrezcan servicios a terceros, como por ejemplo un servicio de traducción interno, un servicio de resaltado o de categorización.

2.2.2 Interacción con el material

En la educación presencial tradicional la identificación de la interacción con el material, en adelante IM, puede ser un tanto confusa, ya que se puede no distinguir el límite entre la entrega del material y la interacción con el mismo. Retomando el ejemplo del punto anterior, en una clase presencial la entrega del material proviene de las palabras del profesor y de cualquier ayuda visual en la cual se esté apoyando, podemos definir que este tipo de interacción (IM) solo ocurre cuando el estudiante hace algo con el material. Por ejemplo, si el estudiante solicita que se repita la diapositiva anterior, o en un video poder pausar, retroceder, avanzar, o en un juego tener la posibilidad de explorar diferentes opciones.

En los sistemas de educación virtual, debido a que usualmente el material es una “grabación”, la IM depende de las posibilidades del servicio que distribuye el contenido y en consecuencia está limitada a las características de la tecnología con la cual se implementa dicho servicio. Algunos ejemplos que representan la relación material-interacción:

Material	Interacción
Libros	Buscar una definición en el libro de referencia.
Video conferencia	Iniciar, pausar, detener, descargar.

Audio conferencia	Iniciar, pausar, detener, descargar.
Juegos Multimedia	Iniciar, jugar, otras opciones.
Software	Descargar, instalar, ejecutar
Textos, tutoriales electrónicos	Descargar
Radio y televisión	Iniciar.
Chats	Iniciar, Responder, terminar

Tabla 2. Ejemplos relación material-interacción

2.2.3 Interacción con el profesor

En un ambiente de aprendizaje virtual los profesores piensan y toman decisiones en torno al diseño y empleo del espacio, el tiempo y la disposición de los materiales:

- ¿Cómo debe ser la organización espacial?
- ¿Cómo va a ser la selección, elaboración y disposición (accesibilidad y visibilidad) de los materiales para el aprendizaje (según un modelo pedagógico, teniendo en cuenta las diferencias individuales y grupales, y adecuados a los intereses y necesidades)?
- ¿Cómo van a estar almacenados y clasificados los materiales?
- ¿Cómo se van a manejar los mensajes y la información dentro del espacio de tal manera que estén claras las responsabilidades y se propicie la autonomía?
- ¿Cómo propiciar las interacciones con el conocimiento, con los otros compañeros y consigo mismo? [73]

En un ambiente tradicional la interacción con el profesor generalmente ocurre durante la actividad de aprendizaje, mediante intervenciones y preguntas. También puede ocurrir en espacios diferentes como la oficina en espacios para consultas o mediante medios de comunicación alternativos como correos, blogs, mensajería instantánea, etc. En el ambiente presencial (cara a cara) se dice que la interacción con el profesor es netamente síncrona; por otro lado en un ambiente virtual tradicionalmente han existido canales alternos a los canales de distribución de contenido para realizar este tipo de interacción, los más comunes, los foros, chats y el *e-mail*.

2.2.4 Interacción entre estudiantes

La interacción entre los estudiantes, en adelante IE, en los ambientes presenciales tradicionales se pueden organizar en dos grupos [72], formales e informales, los primeros se caracterizan por presentarse en eventos que están dentro de las actividades de formación establecidas, por ejemplo en exposiciones, sesiones de discusión, trabajos en grupo, entre otras. Las interacciones informales suceden en lugares y situaciones externas a las actividades de formación donde los estudiantes principalmente hablan sobre temas relacionados con su formación.

Los dos tipos de interacciones se caracterizan, independientes de que ocurran en el ambiente presencial o virtual, por ser de naturaleza dialéctica. En el ambiente virtual las interacciones se observan en los servicios ofrecidos por foros, e-mail, mensajería instantánea, etc.

La interacción entre estudiantes se convierte en el pilar de los modelos pedagógicos que buscan un escenario donde el estudiante es el generador del conocimiento, así mismo, es el que propone y refina el proceso formativo que se adapta a su entorno.

2.2.5 Realimentación del material

En la actualidad la educación presencial tradicional está dirigida por resultados, los cuales pueden ser determinados de varias formas:

- De acuerdo al número de estudiantes que efectivamente cumplieron los objetivos.
- Por los resultados de los exámenes en cada ciclo del proceso formativo.
- Realimentación mediante encuestas.

Sin embargo, para la realimentación del material, en adelante RM, el análisis que se puede realizar de esta manera es demasiado general y simple. En el ambiente virtual existe una ventaja que radica en la individualidad de cada actividad formativa, que puede basarse alrededor de un recurso de aprendizaje, de tal forma que la evaluación y mejoramiento del recurso genera un impacto directo en cada actividad formativa, la trazabilidad es más fácil, aun más, si consideramos que el recurso es reutilizable y que cada estudiante puede aportar criterios de evaluación.

Los LMS tradicionalmente mantienen estadísticas sobre el material, información como el número de veces que un estudiante accede, el tiempo que dedica a la actividad educativa, entre otras, esta información es determinada y almacenada por la plataforma, partiendo de la dinámica plataforma-estudiante.

2.2.6 Representación gráfica del modelo M1

La figura 5 resume gráficamente las interacciones relacionadas con el material en una actividad formativa.

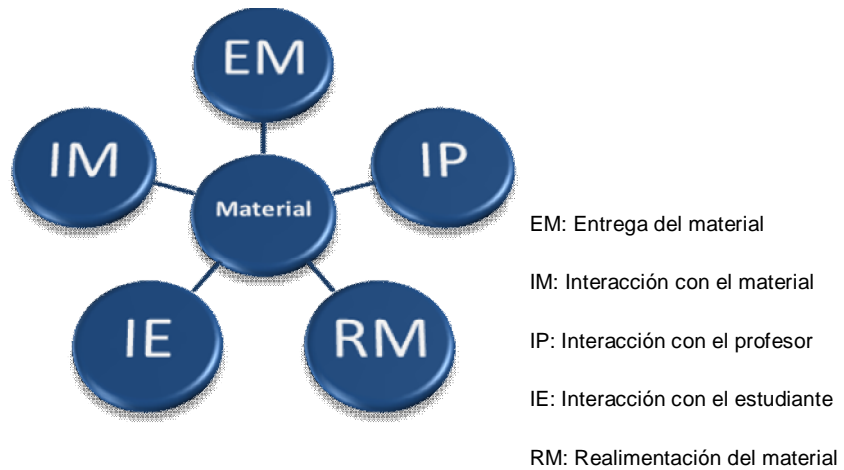


Figura 5. Representación grafica del modelo M1

A continuación se presenta un breve ejemplo de la aplicación del modelo M1, para la identificación de los atributos en la dinámica del material y para el análisis del servicio.

Se compara una clase tradicional y una grabación en video de la misma, distribuida mediante un servicio de video *streaming* convencional.

Interacción	Clase Tradicional	Servicio video <i>streaming</i>
Entrega de material	Voz y explicación del profesor, diapositivas, libros guía, referencias bibliográficas.	En el audio y video.
Interacción con el material	Tomar notas, buscar libros de referencia y actividades de los libros.	Iniciar, pausar, avanzar, retroceder.
Interacción con el profesor	Consultas, preguntas y respuestas en clase.	Ninguna propia del servicio.
Interacción entre estudiantes	Posible si existe trabajo en grupo en la clase, exposiciones.	Ninguna planeada.
Realimentación del material	Posible mediante evaluaciones o encuestas.	Ninguna.
Modelo		

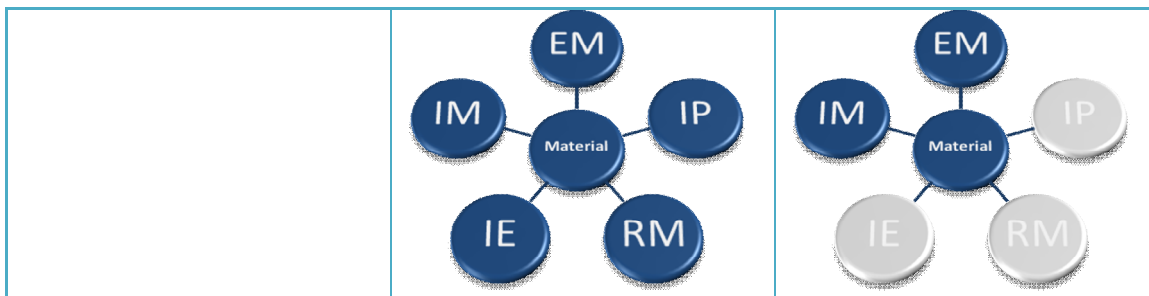


Tabla 3. Análisis clase tradicional-video *streaming*, usando el modelo M1.

El análisis del modelo permite al diseñador del servicio o del contenido, evaluar si para un área o tema en particular es permisible eliminar el trabajo colaborativo en el grupo de estudiantes o si la guía del profesor no se requiere para cumplir los objetivos de la actividad, etc. En el capítulo 4, durante el desarrollo del caso de estudio, se presenta un ejemplo detallado de la aplicación del modelo M1, indicando la forma en que se tienen en cuenta las anteriores interacciones.

2.3 M2: Modelo de servicios de *streaming*

Es importante destacar el adecuado uso de las tecnologías para el desarrollo de las actividades y/o servicios de aprendizaje en el LMS. Esto se puede lograr de manera eficiente conociendo la naturaleza y diseño de las tecnologías, identificando el tipo de servicio o actividad para las cuales fueron creadas y cómo se pueden adaptar a los eventos requeridos dentro del servicio que se quiere ofrecer. En muchas universidades se está observando el alcance del uso de las nuevas tecnologías dentro del modelo de enseñanza, pero los profesores o diseñadores en muchos casos no son expertos en materia de las tecnologías, por ende es necesario desarrollar herramientas de guía para tomar decisiones en el momento de escoger la tecnología más adecuada a la hora de implementar la actividad [74].

El segundo modelo busca determinar las interacciones existentes, descritas en el modelo M1, que pueden lograrse con las diferentes aplicaciones de la tecnología *streaming*, descritas en la sección 1.3.4.

A partir de este punto nos centramos principalmente en la descripción de las aplicaciones de las tecnologías de *streaming*, sin embargo el modelo puede ser adaptado para describir diferentes servicios, incluso los servicios tradicionales existentes en los LMS.

Para la primera parte del modelo hacemos uso de los tipos de material más relevantes dentro de los ambientes virtuales de educación [71], para caracterizar los servicios y aplicaciones de la tecnología de *streaming*.

Texto. El texto es efectivo en la mayoría de actividades de formación, sin embargo no se debe usar en ambientes donde la audiencia objetivo tiene pocas habilidades de lectura o una motivación baja. El texto es mejor que otros tipos de materiales en temas complejos como tendencias económicas o temas abstractos como ecuaciones químicas o cuando el tema tiene una estructura avanzada. “El texto por sí solo no representa adecuadamente la riqueza de lo que el mundo ofrece a través de nuestros sentidos e imaginación” [71], por lo que para maximizar su efectividad se combina con otros tipos de materiales.

Audio. El audio es más efectivo cuando el tema es simple, concreto y tiene poca estructura, como es el caso general de las sesiones de formación en lenguajes extranjeros [71]. También se puede usar para enseñar otro tipo de habilidades cuando se usa el audio con otros tipos de materiales. Se recomienda el uso de audio cuando se quiere ganar la atención de la audiencia y además cuando esta tiene bajo nivel de lectura.

Imágenes. Útil para identificar y clasificar objetos destacando relaciones espaciales, además facilita la conversión de temas abstractos en concretos mediante gráficos. Cuando se combina con otros tipos de materiales, aprovechando las técnicas de asociación, se reduce el tiempo de asimilación de conceptos.

Video. Se puede usar el video para mostrar realismo, permitir que los estudiantes copien, reconozcan movimientos cuando se quiere instruir habilidades psicomotoras, procedimientos o procesos. Principalmente usado para obtener un alto grado de atención, sin embargo no es muy eficiente cuando se necesita instruir información con un alto grado de detalle. Es recomendable acompañar este tipo de material con algún resumen o material que simule funcionalidad de tabla de contenido.

Cuando se combina con audio provee un aprendizaje con mayor detalle y mejor facilidad de asimilación.

Animaciones. Las animaciones son útiles para una gran variedad de propósitos como:

- Mostrar la relación entre objetos e ideas.
- Simular resultados de acciones.
- Mostrar secuencia de pasos dentro de un procediendo o tarea.
- Convertir conceptos abstractos a concretos mostrando algún grado de dinámica.

Partimos de la idea de que el servicio a evaluar se presta de forma independiente a cualquier otro que pueda aplicar en alguna de las categorías y que su modo de comunicación puede ser síncrona o asíncrona.

	Texto	Audio/Video	Imágenes	Otros Datos	Colaborativo	Bidireccional
Video/Audio Bajo Demanda		✓				✓
Video/Audio Vivo		✓				✓
Videoconferencia Punto-Punto		✓				✓
Videoconferencia Multi-punto		✓			✓	✓
Shared Objects	✓		✓	✓	✓	✓
<i>Callbacks</i>	✓		✓	✓		
<i>RPC Streaming</i>	✓		✓	✓	✓	✓
Descarga progresiva ²		✓	✓	✓		✓

Tabla 4. Caracterización de los servicios y aplicaciones de *streaming*, a partir de la naturaleza del material

Una vez caracterizado el servicio o aplicación de la tecnología de *streaming*, evaluamos las interacciones existentes con relación a lo expuesto en el modelo M1.

Entrega del material: Todas las aplicaciones de *streaming*, listadas en la tabla 3, permiten enviar al estudiante información, es decir que pueden ser utilizadas como medio para enviar contenido formativo.

Interacción con el material: A excepción de la aplicación de *streaming* “*Callbacks*”, las aplicaciones de *streaming* permiten que el cliente envíe mensajes de control al servidor, esto significa que los servicios que se puedan desarrollar pueden agregar mecanismos para que el estudiante interactúe con el servicio.

² Descarga progresiva, aunque no se base en streaming se tiene en cuenta como estrategia válida y útil para el desarrollo de nuevos servicios.

Interacción con el profesor: Inicialmente solo las aplicaciones interactivas (videoconferencia punto a punto y multipunto) las cuales se caracterizan para este propósito cumplirían con este tipo de interacción. Las aplicaciones de video/audio bajo demanda y en vivo, no permiten que los actores del servicio interactúen por lo que son descartados. La aplicación de *Callbacks* dada su naturaleza unidireccional claramente no permite implementar este tipo de interacción. En la aplicación de objetos compartidos, o *shared objects*, se propone no tenerla en cuenta para la implementación de este tipo de interacción, ya que se enfoca al desarrollo de aplicaciones colaborativas en tiempo real, por lo que requeriría que el profesor y el estudiante se encuentren al mismo tiempo y que iría en contra de uno de los objetivos de los LMS. Finalmente la aplicación de RPC Streaming permite desarrollar nuevos mecanismos de comunicación entre estudiante y el profesor, independientemente de si se requiere sincronización entre los actores o si es un medio asíncrono.

Interacción entre estudiantes: Se aplica el mismo análisis que la interacción anterior, con la modificación asociada a la videoconferencia punto a punto y la aplicación de *shared objects*. La primera no se incluye debido a que el servicio de videoconferencia punto a punto se enfoca a la comunicación entre profesor-estudiante y aunque el servicio puede ser utilizado para este propósito se debería permitir un servicio no excluyente, es decir que permita no solamente la comunicación de dos estudiantes, sino que se pueda evaluar el uso de la videoconferencia multi-punto. Para el segundo caso, *shared objects*, se tiene en cuenta debido a que permite el desarrollo de servicios colaborativos que pueden incluir mecanismos de interacción para los estudiantes.

Realimentación del material: Para cumplir con la interacción “realimentación del material” los servicios o aplicaciones requieren de una comunicación bidireccional durante el funcionamiento del servicio de tal forma que se pueda obtener información relevante acerca del contenido que se entrega al estudiante, por ejemplo estadísticas de uso.

Para resumir lo expuesto anteriormente se presenta la tabla 5.

		Interacciones M1				
		EM	IM	IP	IE	RM
Aplicaciones de streaming	Video/Audio Bajo Demanda	✓	✓			
	Video/Audio Vivo	✓	✓			
	Videoconferencia Punto-Punto	✓	✓	✓		✓
	Videoconferencia Multi-punto	✓	✓	✓	✓	✓
	Shared Objects	✓	✓		✓	✓

Callbacks ó Tecnología Push	✓				
RPC <i>Streaming</i>	✓	✓	✓	✓	✓
Descarga progresiva	✓	✓			

Tabla 5. Identificación de las interacciones M1 en las aplicaciones de la tecnología de *streaming*.

Existe una característica más que no se ha mencionado de forma explícita en este modelo, determinada por el tipo de comunicación, síncrona o asíncrona, que use el servicio para lograr la interacción entre los actores de la actividad de aprendizaje (estudiante-estudiante, estudiante-profesor o estudiante-material). Lo anterior se debe a que el modelo M2 sólo trata de analizar las diferentes opciones tecnológicas que tiene el diseñador a la hora de implementar el servicio, y es éste quien determina el tipo de comunicación que dicho servicio va a ofrecer, sin embargo cabe aclarar que no todas las tecnologías descritas facilitan la selección de la una o la otra, por lo que se establecieron las categorías “Colaborativo” y “Bidireccional” (Ver tabla 2) para ayudar en esta decisión.

Con la información descrita en las tablas 4 y 5, el diseñador de servicios de aprendizaje puede determinar qué tipo de tecnología es útil para lograr que el estudiante perciba una experiencia de aprendizaje deseada, de igual forma, lograr las interacciones con el material que maximicen la apropiación del conocimiento.

2.4 M3: Modelo para selección de tecnología

Mucho se ha tratado en el área de la educación virtual sobre la integración de tecnología debido a que su inclusión en el proceso formativo afecta directamente la calidad de la educación. Aunque existe bastante información al respecto, mucha de ella se centra en la transformación de un ambiente presencial a uno virtual. El desarrollo de este modelo parte del hecho que se cuenta con un ambiente virtual establecido, más específicamente, haciendo uso de un LMS sobre el cual se quiere integrar nuevos servicios.

Hemos seleccionado un conjunto de autores, cuyas ideas y propuestas nos sirven de punto de partida para seleccionar los criterios más relevantes que se incluyen en este modelo.

Fenrich [71] menciona que el costo es el principal factor para la implementación de soluciones multimedia, y señala que los productos desarrollados tienen mayor éxito

cuando solucionan las necesidades más importantes, no generan la necesidad de desarrollar habilidades especiales para su uso y que además que el producto no sea percibido como una competencia frente a actores y procesos dentro del proceso formativo existente. Acuña y Montaña [75] aportan una visión sobre la innovación de la docencia, vista desde la pedagogía, tecnología, la cultura y el aprendizaje organizacional, identifican algunos problemas en los modelos pedagógicos que implican el uso de las Tecnologías de la Información y la Comunicación, y finalmente ofrecen su punto de vista sobre lo que los servicios en general deben permitir a los estudiantes, un aprendizaje autónomo basado en la creación de nuevos canales de comunicación. Bates [68] evalúa con un alto grado de detalle los tipos de tecnologías y un conjunto de características externas que deberían tenerse en cuenta para determinar si el uso de dichos servicios son aptos para los estudiantes o para el material, este último evaluado por Debevc, Venuti y Peljhan en [76].

2.4.1 Atributos de la temática

El modelo M3 inicia identificando los atributos de la temática que se quiere presentar mediante las tecnologías, en general de cualquier naturaleza. Esto se refiere a los atributos necesarios para la eficiencia y efectividad de los contenidos.

A pesar de que los atributos de una actividad generalmente suponen que el profesor o diseñador posee suficiente experiencia en la materia, siempre es conveniente preguntarse si la forma y fondo del contenido son realmente necesarios o deseables en cada tipo de actividad, por ejemplo si el texto es necesario o deseable, los colores, un juego, los videos, etc. También se debe determinar los parámetros que permiten evaluar la calidad del material.

2.4.2 Participación de los actores de la actividad formativa

La participación de estudiantes y profesores es requerida debido a la necesidad de prever si los contenidos de la actividad son accesibles para todos en general y evaluar si las destrezas necesarias en el uso de la tecnología están al alcance de todos. Por ende, es pertinente antes de todo realizar una serie de preguntas con el fin de fortalecer estos conceptos.

- ¿Qué destrezas se necesitan desarrollar en el uso de la tecnología?
- ¿Qué implicaciones económicas tiene la tecnología para los usuarios del sistema?
- ¿La infraestructura tecnológica de estudiantes, profesores y el LMS en general está lista para la integración de la tecnología?
- ¿Se necesita capacitación para el uso de la nueva tecnología? ¿Cuánto tiempo?

2.4.3 Costos

No solamente se evalúa el costo de la tecnología, sino los costos que se generan debido a la necesidad de capacitación, el costo de producción de los materiales que se ofrezcan con la tecnología y el costo que se genera para el profesor en cuanto al tiempo que debe dedicar en la preparación de las actividades de formación que usen la tecnología.

Estos tipos de costos afectan directamente la calidad del material y en consecuencia la efectividad de la actividad formativa. Dentro de la evaluación de costos es importante tener en cuenta la naturaleza de los tipos de materiales que se van a usar, de tal manera que de acuerdo al tema, su estructura y el contexto de aplicación, mejoren la calidad y efectividad de la sesión de aprendizaje. Dicha evaluación debe determinar si los materiales óptimos para la actividad existen, se pueden adquirir gratuitamente o mediante pago, o si por el contrario deben ser producidos desde cero.

Inicialmente se pueden determinar los costos de los materiales teniendo en cuenta dos factores:

- El costo del recurso digital (imagen, video, audio, modelo 3d, etc.), en caso de que se pueda adquirir mediante terceros.
- El costo de producción y/o edición de los recursos, ya sea que se produzcan desde cero o sean adquiridos. Teniendo en cuenta que la producción/edición requiere de la inversión de tiempo y en algunos casos de personal capacitado y especializado de acuerdo a la complejidad en el tratamiento de los recursos y sus procesos asociados.

Los costos asociados al desarrollo del material por parte del profesor, dependen entre otros, de las habilidades y conocimientos propios alrededor de la materia, debido a esto es difícil encontrar un conjunto de métricas eficientes, que sirva como referente para determinar esta información. En el Anexo 3, se presenta en resumen información relevante acerca de este tema, difundida por Matt Lobel, soportada por 16 años de experiencia y más de 500 clientes en el área del *e-learning*.

2.4.4 Aplicación del modelo M3

El modelo M3 se basa en el seguimiento de un proceso iterativo e incremental:

En la primera fase se analiza la actividad formativa y servicio que se quiera desarrollar, desde el punto de vista de las interacciones con el material y de acuerdo a las categorías

descritas en el modelo M1. En la segunda fase se hace una lista de las tecnologías disponibles según lo descrito en el modelo M2, preseleccionadas de acuerdo al grupo de criterios que apliquen. En una tercera fase se hace un refinamiento de las tecnologías seleccionadas, teniendo en cuenta los criterios de participación y costos presentados en los literales anteriores. Una cuarta fase puede ser desarrollada, de acuerdo a lo propuesto en el modelo del diseño instruccional ADDIE [67], donde el servicio se compara con servicios similares u otros que tengan relación con las actividades de formación para evaluar su impacto o factores de mejoramiento. La tabla 5 presenta el resumen de las actividades relacionadas con la aplicación del modelo M3.

Fase 1	Utilizar las categorías descritas en el modelo de integración de servicios
Fase 2	Listar todas las tecnologías apropiadas descritas en la mecánica del tema dentro de cada categoría del modelo de dinámica del material.
Fase 3	Refinar la lista de tecnologías basadas en las participaciones de estudiantes y profesores y costos.
Fase 4	Comparar ventajas y desventajas cuando sea posible.

Tabla 6. Resumen fases aplicación modelo M3.

En el capítulo 4 se presenta un caso de estudio de la integración de un servicio basado en tecnología de *streaming*, donde se aplican los 3 modelos presentados anteriormente.

Una vez seleccionada la tecnología y caracterizado el servicio se pasa a la integración con el LMS, desde un punto de vista técnico.

2.5 Lineamientos técnicos

En el capítulo 1, literal 1.3.4, se presentaron las tecnologías más relevantes asociadas a la tecnología de *streaming*. Esta sección parte de la recomendación de seleccionar la tecnología Flash como eje central para el desarrollo de nuevos servicios o la integración de servicios existentes, destacando algunos criterios de selección en relación a la información de la tabla 1:

- La tecnología Flash ofrece una plataforma con *kits* de desarrollo (*frameworks*, API, Entorno de Desarrollo) de libre acceso.
- El cliente Flashplayer se integra a la mayoría de navegadores existentes, en los principales sistemas operativos.

- El desarrollo de la tecnología es liderado por una importante compañía, Adobe Inc., por lo que el desarrollo está garantizado al mediano y largo plazo.
- Ofrece soporte para todas las aplicaciones de *streaming* descritas de forma nativa.
- Aunque hasta la versión 9.X del cliente FlashPlayer hace uso de protocolos basados en TCP, a partir de su versión 10 da soporte para aplicaciones basadas en protocolo UDP.
- Existe interacción entre el cliente FlashPlayer y el lenguaje de scripting Javascript, lo cual permite crear aplicaciones que interactúen a nivel del navegador del cliente.
- Integra codecs de audio y video de forma nativa, facilitando un entorno común de desarrollo donde se asegura que las aplicaciones basadas en audio y video *streaming* se ejecutan sin mayores inconvenientes.
- La tecnología Flash está lista para la integración de nuevas características relacionadas con el *streaming*, como por ejemplo los estándares de video de alta definición, el soporte para ambientes móviles, entre otras.
- Existen diferentes tipos de servidores para la creación de soluciones mediante tecnología Flash, ya sea con licencia comercial con su correspondiente soporte técnico, como servidores basados en software libre, con el soporte que ofrece la comunidad de usuarios.

2.5.1 Sistema Operativo

La mayoría de servidores evaluados alrededor de la tecnología Flash funcionan bajo los sistemas operativos Windows y Linux. Debido a los costos de licencias y a la gran trayectoria de Linux en el ambiente de servidores, se recomienda el uso de distribuciones basadas en Linux para la instalación de los servidores.

2.5.2 Servidor y lenguaje de programación

Para la selección del servidor se presentan dos escenarios:

Para un ambiente donde se minimizan costos:

La elección del servidor para este escenario es Red5, debido a que está implementado en tecnologías de gran trayectoria en el área de los servidores como Java, así mismo, permite su instalación como servidor independiente o como una aplicación basada en servlets sobre un servidor de aplicaciones como Tomcat, heredando algunas características como rendimiento, aprovechamiento de recursos hardware (multi procesadores), gestión remota (JMX) y sobre todo las facilidades de escalabilidad.

El servidor está desarrollado sobre la tecnología Servlet 2.5, encontrada en la especificación 5 de JEE, pero lo más importante es que hace uso del framework Spring,

especificación 2.X, el cual dirige los desarrollos bajo el modelo de Inyección de dependencias, la cual se basa a su vez en los principios básicos de programación “*Alta Cohesion*” y “*Bajo Acoplamiento*”.

Aunque el soporte de primera mano está dirigido por la comunidad de usuarios del proyecto, existen empresas dedicadas a dar soporte comercial de alto nivel a la misma plataforma.

Para un ambiente sin consideración de costos debido al servidor:

Este segundo ambiente, se caracteriza la elección del servidor independiente del costo que genere.

Bajo estas condiciones se evaluaron las soluciones Flash Media Server (FMS) y Wowza Media Server en sus versiones de prueba, limitadas únicamente a 10 conexiones concurrentes. El resultado es la recomendación en la elección de Wowza debido, entre otras, a los siguientes:

- Wowza tiene todas las funcionalidades de FMS y además ofrece servicios adicionales como *streaming* HTTP para iPhone, *streaming* de contenido 3GPP, y para la tecnología Silverlight de Microsoft.
- Soporte nativo para IPTV basado en el estándar MPEG-TS.
- Soporte de protocolos RTSP/RTP unicast y multicast, sobre TCP y UDP.
- Uso eficiente de los recursos de equipos de alto rendimiento (Hasta 16 CPU Cores)
- Programación de aplicaciones en Java
- Sistema de balanceo de cargas integrado
- Precio de licencia 4.5 veces menos que FMS

El lenguaje de programación recomendado para los dos escenarios es Java, debido a su portabilidad, facilidad y a que las plataformas seleccionadas dan soporte para este lenguaje.

2.5.3 Protocolo

Sobre la selección del protocolo más recomendado dentro de la tecnología Flash, se sugiere el uso del protocolo RTMP, debido a que es el protocolo por defecto para manejar esta tecnología, sin embargo la adopción de este protocolo genera un número de consideraciones que se deben evaluar por parte del diseñador del servicio de *streaming*, pero más importante por el administrador de la red.

- El uso del protocolo RTMP implica el uso del puerto 1935, dado que no es un puerto común, los sistemas de seguridad que se hayan implementado dentro de la red pueden bloquear cualquier tipo de transmisión asociada. Si se establece que el servicio se va a prestar sólo dentro de la red local, el protocolo puede ser usado

sin problemas, por el contrario si se piensa desarrollar un servicio ofrecido hacia Internet, sólo se puede usar si el administrador de red, de acuerdo a las políticas de seguridad, lo permite.

- Teóricamente RTMP es más eficiente con respecto al ancho de banda, en comparación a RTMPT, debido a que el enmascaramiento de RTMP sobre HTTP genera tráfico extra asociado a las cabeceras de HTTP requeridas en cada petición tipo POST. sin embargo RTMPT puede solucionar algunos problemas asociados a los firewall generales, en el anexo 2 se presentan resultados sobre el protocolo RTMPT que muestran que en la práctica el ancho de banda no es el principal factor de análisis.

Adicional a los dos protocolos presentados, existe el protocolo RTMPS que a la fecha de desarrollo de este proyecto no es soportado oficialmente por Red5, ni existe una especificación oficial ni formal, por lo que su evaluación debería realizarse una vez esté soportado de forma oficial por la herramienta en trabajos futuros.

2.5.4 Ancho de banda

Basados en el informe más reciente de la CRT para el año 2009 [77] y enfocándonos al servicio tradicional del *streaming* de video, se sugiere que los servicios de *streaming* se diseñen basados en un ancho de banda de 512Kbps, ya que con ese parámetro se cubrirían, de mantenerse la tendencia, un 52.9% de los nuevos suscriptores de servicios de Internet dedicado y un 80,5% de los actuales clientes de Internet en el país [78].

Es importante precisar que para el distribuidor de los servicios de *streaming* el ancho de banda es uno de los factores importantes a la hora de seleccionar los servicios que se pueden ofrecer y el entorno donde se despliegan. Por ejemplo, para un ambiente intranet donde la red local se mantiene estable con un alto número de clientes de un servicio de *streaming* los costos de inversión o mantenimiento pueden ser relativamente bajos, pero en un ambiente donde los servicios se acceden desde Internet el costo asociado al ancho de banda de subida puede ser demasiado elevado.

El análisis de este aspecto argumenta nuevamente las recomendaciones establecidas en el literal 2.4.2, donde la participación de los administradores de red es esencial para la integración de estos servicios. También es importante realizar pruebas piloto de los servicios para tener una visión más real que permita evaluar de forma integral su impacto.

2.5.5 Costos

En el literal 2.4.3 se presentaron conceptos sobre los costos relacionados con el diseño y creación de los materiales que se ofrecen mediante los servicios de *streaming*, en este

apartado se presentan al lector los aspectos básicos desde la perspectiva técnica que determinan los costos generados en la integración de servicios de *streaming*.

Para cada servicio de *streaming* que se quiera integrar se debe realizar un análisis sobre el ancho de banda necesario para su funcionamiento y consumo. Luego se responde a la pregunta ¿Cuántos usuarios concurrentes van a estar consumiendo el servicio en cualquier instante?, teniendo en cuenta que diferentes servicios pueden consumirse al mismo tiempo.

De igual forma es importante establecer la cantidad de equipos de infraestructura necesarios para dar inicio a la prestación de servicios, esto requiere seleccionar cuidadosamente el hardware, con las características de memoria, nivel de procesamiento e interfaz de red, necesario para cubrir la demanda en el consumo de los servicios previstos.

Con base en lo anterior se define el contexto de aplicación de los servicios, bien sea en un ambiente local o un ambiente externo.

Si se establece la prestación de un conjunto de servicios basados en *streaming* integrados con un LMS hacia un público o clientes en Internet, se recomienda adoptar una estrategia de *outsourcing*, que permita reducir costos en lo que se refiere a equipos, pero sobre todo al ancho de banda. Particularmente se recomienda hacer uso de los beneficios que ofrecen los servicios de computación en la nube, *cloud computing*, donde se puede controlar dinámicamente los recursos necesarios y pagar sólo por lo que se necesita. Por ejemplo se puede iniciar con un servidor de características normales la prestación del servicio, en caso de que se determine que en cierto horario las necesidades aumentan, se puede “arrancar” otro servidor y así entre los dos dar un mejor soporte. Finalmente cuando las condiciones requeridas regresen a la normalidad se puede detener el segundo servidor y sólo pagar por las horas que estuvo en funcionamiento.

2.5.6 Lineamientos de integración de las tecnologías de *streaming* y del LMS

Esta sección presenta los lineamientos propios para la integración de las tecnologías de *streaming*, mencionadas con anterioridad, y las tecnologías asociadas al LMS.

Antes que nada, se establecen un conjunto de condiciones que, a nuestro criterio, los lineamientos deben respetar para mantener la integridad, seguridad y otros factores asociados a la plataforma LMS:

- El software libre se caracteriza en general por que el soporte es ofrecido de primera mano por la comunidad alrededor del proyecto, pero también es

característico los procesos de actualización que realizan de forma continua, las cuales mantienen la seguridad y estabilidad de la plataforma. Por lo anterior, es importante que los LMS basados en software libre extiendan su funcionalidad basados en las herramientas que para esa tarea se definan, es decir, que en lo posible el núcleo del sistema debe mantenerse intacto y así poder garantizar los beneficios de soporte y actualización que se puedan ofrecer.

- Para la integración de nuevos servicios o funcionalidades en los LMS se debe tener presente que el LMS ya cuenta con una información definida (usuarios, materiales, mensajes, etc.) y un recorrido hacia el cumplimiento de los objetivos educativos, por eso es importante que los nuevos servicios no afecten negativamente los beneficios de los servicios que ya se encuentren en funcionamiento. De la misma forma, si los nuevos servicios trabajan con base en datos relevantes para el funcionamiento del LMS, se debe tener cuidado con su tratamiento y en lo posible manejar copias de los mismos, mientras se evalúa y determina la madurez del sistema que gestiona los nuevos servicios.
- Independiente del lenguaje de programación la gran mayoría de LMS coinciden en el uso de las tecnologías de HTML para la presentación de los materiales de formación, también en el uso de las tecnologías de CSS para los estilos y Javascript para realizar lógica de negocio básica del lado del cliente. En consecuencia se debería aprovechar la experiencia de los estudiantes con estas tecnologías y en lo posible hacer uso de las mismas cuando se diseñen e implementen nuevos servicios.
- Uno de los objetivos principales de los LMS es la administración de los materiales de formación por lo que la creación e integración de nuevos servicios debería mantener esta característica.
- Al igual que la condición anterior, los servicios basados en *streaming* deberían desarrollarse para que se puedan consumir desde cualquier lugar y en cualquier momento.
- Otro aspecto importante que puede condicionar el desarrollo de los servicios, pero más de la plataforma, radica en la práctica común de manejar más de un LMS dentro de los centros educativos y además que pueden o no ser implementadas usando la misma herramienta software.

Iniciamos definiendo los mecanismos existentes de extensibilidad para la plataforma del LMS que permita la integración de los servicios de *streaming*:

1. *Fork*³ del LMS

³ *Fork*: En ingeniería de software, hace referencia a la modificación de un proyecto a partir de su código fuente para crear una nueva herramienta con características diferentes al original.

2. Desarrollo de “módulos” o “*plugins*” mediante sus propias API’s
3. Aprovechamiento de estándares existentes [31, 79-82]

Con las condiciones definidas anteriormente seleccionamos la tercera opción de extensibilidad, aprovechando los estándares existentes para la interoperabilidad, ya que nos permite mantener la integridad del LMS y además ampliamos la cobertura de sistemas LMS.

Seleccionamos SCORM, como estándar para la integración de objetos de aprendizaje (SCO), aprovechando la definición del SCO, como “...*Un SCO representa el nivel más bajo de granularidad de un recurso de aprendizaje que puede ser monitoreado por el LMS usando el entorno de ejecución SCORM...*”, es decir que el SCO no precisa un tipo de contenido específico, por el contrario, sólo existe para entregar un conocimiento deseado y cumplir unos objetivos de aprendizaje. De igual forma, SCORM no define la tecnología necesaria para la implementación del mismo.

En el capítulo 3 se describe en detalle la manera en que se integran los servicios de *streaming* con la plataforma LMS mediante la definición de un SCO de SCORM.

Establecemos cuatro niveles de integración para los servicios de *streaming*.

2.5.6.1 Nivel de comunicación

Partiendo de la estandarización que ofrece SCORM acerca de los objetos de aprendizaje se propone el uso de los mecanismos de comunicación establecidos por el entorno de ejecución (RTE) de SCORM en cada LMS. Esto es, hacer uso del API en Javascript del lado del cliente para mantener la comunicación con el LMS. Así mismo, se debe mantener un canal directo con el servidor de *streaming* donde se consuman los diferentes servicios.

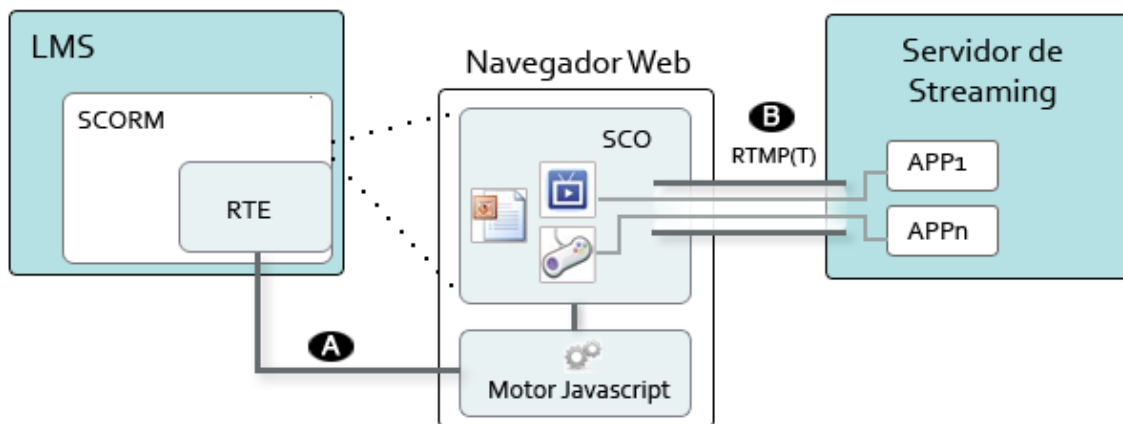


Figura 6. Mecanismos de comunicación

En la figura 6 se identifican los mecanismos de comunicación (A y B) que se deberían establecer para la interacción entre el LMS y el servidor de *streaming*. El mecanismo de comunicación A permite enviar información hacia el LMS desde el SCO mediante el API definida por SCORM a través de Javascript (avance dentro del material, reproducción completa, pausa o terminación entre otros), por otro lado los servicios que se ejecutan dentro del SCO hacen uso de una sola conexión RTMP(T) con las aplicaciones que corren en el servidor de *streaming*, en el capítulo 3 se detalla mas esta relación.

2.5.6.2 Nivel de datos

En este nivel se tiene en cuenta cuatro tipos de información:

- para la relación SCO-LMS
- para la relación SCO-Servidor de *streaming*
- administración de los recursos multimedia en el servidor de *streaming*
- modelo de datos para la plataforma de *streaming* (modelo relacional Base de Datos)

Relación SCO-LMS, hace referencia a los datos que se envían y reciben desde el SCO y el entorno de ejecución de SCORM del lado del LMS. Estos datos se basan en el modelo de referencia de datos CMI para la interoperabilidad entre plataformas de aprendizaje, sobre todo para los procesos de trazabilidad, por ejemplo para determinar cuando un estudiante ha accedido a un objeto de aprendizaje, si ha completado la actividad o para mantener información sobre los resultados de un examen.

RTE permite que los servicios de *streaming* consulten alguna información, definida en el modelo CMI, para su funcionamiento. Sin embargo, alguna información es reservada y opcional, dependiendo de la implementación propia del estándar SCORM de dicha plataforma.

Relación SCO-Servidor de *streaming*, debido al canal de comunicación RTMP(T) entre el SCO y el servidor de *streaming*. En esta parte los datos no pueden generalizarse, debido a que cada tipo de aplicación de la tecnología de *streaming* define su propia estructura. Para los servicios de video *streaming* los datos están en función de los códecs de audio y video, para los servicios basados en procedimientos remotos los datos dependen del tipo de respuesta generalmente mediante AMF.

Anteriormente mencionamos la Realimentación del material como una interacción que permite caracterizar las actividades de aprendizaje, como primera medida se propone la creación de un servicio de trazabilidad que permita entre otras monitorear eventos específicos generados mediante la interacción del estudiante con el servicio de *streaming*.

Basado en la versatilidad del modelo de datos CMI propuesto por SCORM se propone la definición de un modelo similar para la gestión de los datos de trazabilidad, descrito en la figura 9.

ID	Elemento	Valor	Campo temporal	Estudiante	Servicio <i>streaming</i>	Fecha
----	----------	-------	----------------	------------	---------------------------	-------

Figura 7. Modelo de datos de trazabilidad

La figura 7 muestra los diferentes componentes que describen el modelo de datos sugerido para la trazabilidad, basado en la estructura y organización del modelo CMI de SCORM.

- ID, este campo debe mantener una única referencia sobre un registro de trazabilidad. Se recomienda manejar tipos de datos con límites de valores altos por ejemplo, tipo BIGINT del lenguaje SQL.
- Elemento, el nombre del evento o dato de trazabilidad que se quiere registrar. El tipo de dato sugerido es una cadena de caracteres no mayor a 255 caracteres. En el capítulo 4 se presenta el caso de estudio donde se define un modelo de datos de trazabilidad para el servicio de *streaming*. La estructura de este elemento se sugiere de la forma “XX.YY.ZZ”, separada mediante el carácter “.” (punto), donde XX representa el nombre del modelo de datos, YY el nombre de la categoría del elemento de trazabilidad y ZZ es el nombre del elemento. La cantidad de valores en la estructura es sugerida pero no se limita a 3.
- Valor, un dato de cualquier tipo que da un significado al elemento dentro de un rango definido. Por ejemplo para un elemento que registra si un estudiante está reproduciendo un recurso audio-visual el campo Valor puede definir uno de tres posibles estados “pausado”, “reproduciendo” ó “detenido”.
- Campo temporal, está dedicado a los servicios de *streaming* que tienen un componente temporal propio, entre estos, los servicios de *streaming* de video y audio. Este valor registra en qué momento se genera un evento desde la escala temporal del servicio, por ejemplo, en un evento que registra cuando un estudiante avanza desde el minuto X de un video hasta el minuto Y del mismo, el valor de este campo tendría el valor de X. En el capítulo 4 se muestra más claramente el uso de este campo.
- Estudiante, un identificador del estudiante al cual pertenece este registro de trazabilidad.
- Tipo de servicios, el servicio de *streaming* que genera el registro de trazabilidad.
- Fecha, registra el momento en que se crea el registro de trazabilidad.

Administración de los recursos multimedia en el servidor de *streaming*, dado que los servicios de *streaming* se apoyan en el uso de recursos multimedia estos deberían de almacenarse en la propia plataforma de *streaming*, debido a esto existe la necesidad de gestionarlos de tal forma que los objetivos de organización de los LMS se mantengan.

Para esta tarea se propone usar el estándar MPEG7, como mecanismo de descripción de los recursos multimedia, que permita desarrollar una primera fase de gestión y que a futuro no limite el desarrollo de sistemas de gestión más avanzados.

Modelo de datos para la plataforma de *streaming*. De acuerdo a lo establecido en la condiciones para la definición de los lineamientos, la plataforma de *streaming* debe definir un modelo de datos relacional que tenga en cuenta la definición de la información básica del estudiante y del profesor, que mantenga de alguna forma la estructura de datos básica de un LMS. Se presenta una estructura de referencia básica entre las relaciones que consideramos se deben mantener del lado del servidor de *streaming*.

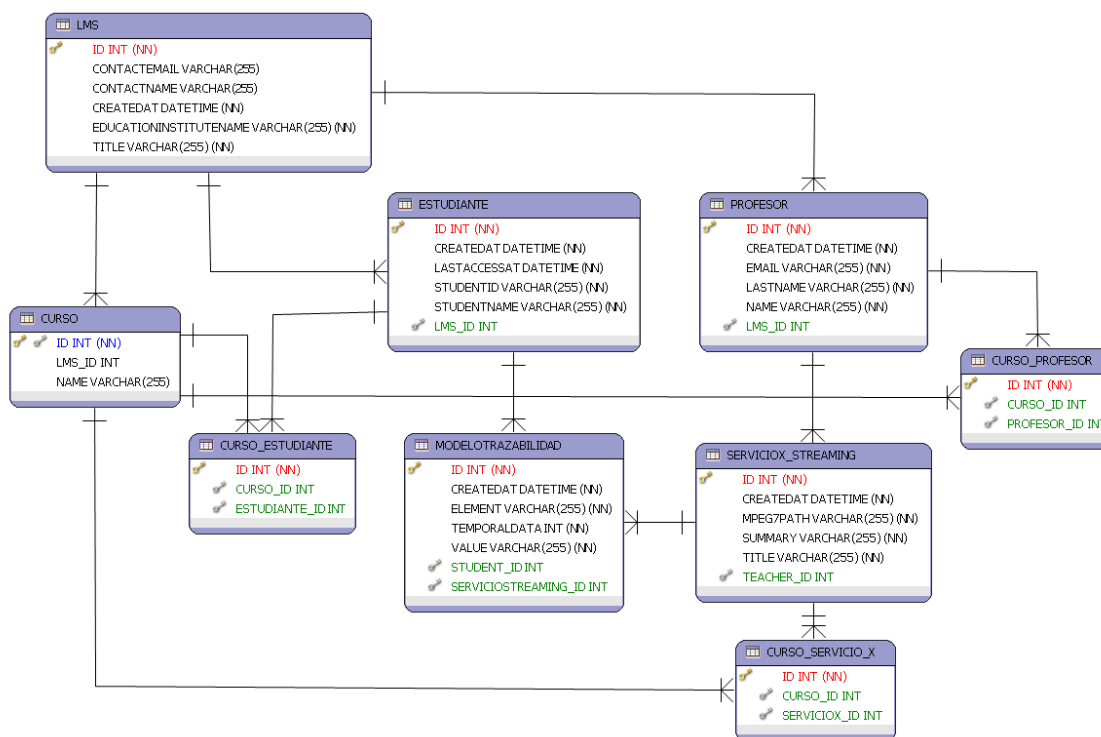


Figura 8. Modelo relacional básico para plataforma de streaming.

Se destaca la definición de tablas de estudiantes, profesores y cursos, que permiten trabajar con copias de dichos datos en el LMS, para mantener la integridad de los sistemas. También se muestra la relación que permite gestionar la trazabilidad del estudiante por cada servicio de *streaming*.

2.5.6.3 Nivel de lógica de negocio

Los procesos de negocio que se requieran para los servicios de *streaming* pueden distribuirse entre el servidor y el cliente

Desde la fase de diseño de los servicios de *streaming* se debe tener presente que gracias al planteamiento de una arquitectura distribuida se cuenta con dos entornos para realizar procesos de lógica de negocio.

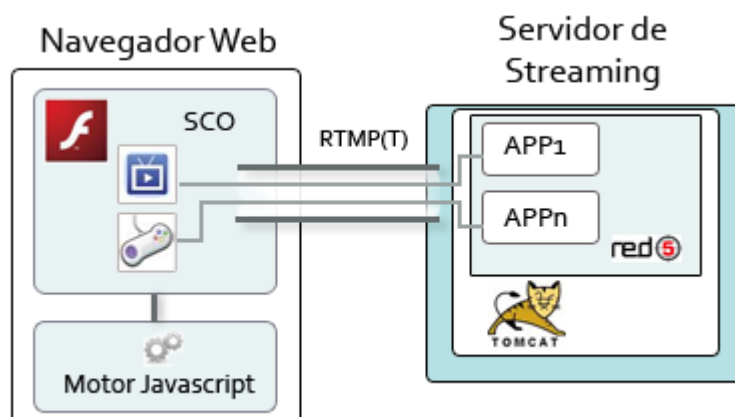


Figura 9. Lógica de negocio cliente y lógica negocio servidor

El SCO permite el uso del entorno de ejecución de Flash Player, esto permite que los clientes desarrollados para consumir los servicios del lado del servidor de *streaming* puedan implementar lógica de negocio directamente en el cliente. Se recomienda aprovechar estas ventajas para liberar de carga de procesamiento al servidor de *streaming*, teniendo en cuenta las limitantes que pueda definir la tecnología.

Debido a la selección de la herramienta para la implementación de los servicios de *streaming*, se cuenta con un entorno apto para el desarrollo de lógica de negocio avanzada basada en las posibilidades del servidor de aplicaciones Tomcat.

2.5.6.4 Nivel de presentación

Existen varios aspectos importantes a tener en cuenta a la hora de diseñar los servicios de *streaming* para los LMS, el primero asociado a la usabilidad desde el punto de vista del ancho de banda, debido a que los LMS se caracterizan por brindar servicios que requieren bajas velocidades de navegación, de igual forma los servicios de *streaming* deberían permitir ofrecer el servicio a diferentes velocidades, idealmente de forma automática.

El segundo aspecto importante se origina desde las características de las aplicaciones web, donde la resolución de la pantalla condiciona las dimensiones de la interfaz de usuario, además se tiene que tener en cuenta que las diferentes implementaciones del estándar SCORM pueden ubicar el despliegue del SCO de diferentes formas, por ejemplo:

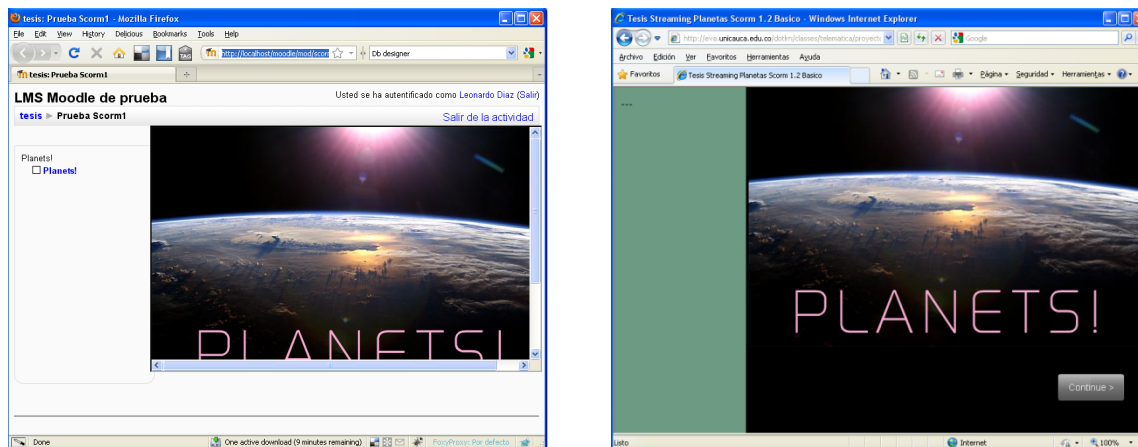


Figura 10. Presentación SCO en Moodle y .LRN

Para mediar este problema, se plantea que los clientes que consumen los servicios de *streaming* dispongan los elementos gráficos de dos formas, la primera basada en una interfaz de usuario simple, que esté pensada para bajas dimensiones y la segunda para un estilo de pantalla completa, aprovechando este tipo de funcionalidad en el reproductor FlashPlayer.

2.5.6.5 Seguridad

En los ambientes virtuales de formación el concepto de seguridad difiere en comparación a otro tipo de sistemas debido a los objetivos sobre los cuales se fundamenta la implementación de este tipo de sistemas, enseñanza y evaluación. De tal forma que la seguridad vista desde esos objetivos se enfoca en tres aspectos fundamentales [83]:

- Identificación del estudiante, las más general de las características de seguridad y trata sobre los mecanismos que permiten determinar que sólo los estudiantes tienen acceso a la plataforma de formación y que son en realidad quien dicen ser.
- Garantizar que el estudiante en realidad está viviendo la actividad de aprendizaje como se desea, y por ejemplo, no sólo está dando clics aleatorios en el material de aprendizaje mientras observa videos en *Youtube*.
- Garantizar que el estudiante retiene el conocimiento, teniendo en cuenta que puede buscar y preguntar lo que quiera usando cualquier tipo de herramienta.

Dicho esto, es evidente que sin importar los mecanismos técnicos que garanticen la seguridad en la plataforma de formación, de la misma forma que se hace en otros tipos de sistemas, la seguridad que en realidad necesitan los LMS debe dirigirse desde otro enfoque.

Los lineamientos que se proponen en esta sección se enfocan a la protección de los materiales ofrecidos por los servicios de *streaming*.

Teniendo en cuenta los mecanismos de comunicación, limitaciones y características de la plataforma LMS y el sistema de *streaming*, la primera recomendación se basa en la definición de mecanismos para verificar que solo los estudiantes de un curso en particular tengan acceso ciertos servicios de *streaming*. Se presentan las posibilidades para este propósito, teniendo en cuenta que gracias a la abstracción de los servicios de *streaming* como objetos de aprendizaje, el propio LMS ofrece un mecanismo de control de acceso sobre el despliegue del cliente que permite consumir el servicio.

- Verificación de parámetros de acceso desde el cliente del servicio de forma manual. Esto es solicitar mediante una interfaz de usuario adecuada para este propósito información (nombre de usuario y contraseña o clave de acceso al curso, etc.). Se destaca el beneficio de tener un sistema de seguridad altamente comprobado y eficiente, pero que necesitaría de un sistema de gestión adicional al de la plataforma LMS.
- Verificación de parámetros de acceso desde el cliente del servicio de forma automática. Basado en las posibilidades que permite la implementación SCORM dentro del LMS, se cuenta con un mecanismo que permite consultar algún tipo de información relevante para realizar procesos de validación. Sin embargo la información es limitada, dependiente de la forma en que se implementa el estándar e inseguro desde el punto de vista técnico, ya que la comunicación se realiza mediante Javascript, siendo relativamente fácil interceptar y modificar la información para engañar a los dos sistemas.
- Verificación de acceso desde el cliente del servicio mediante parámetros determinados en el sistema de *streaming*. Esto se refiere a la identificación individual de cada objeto de aprendizaje mediante un grupo de parámetros (*token* único de identificación del servicio, de la plataforma LMS y del profesor que requiere el servicio). Este tipo de validación permite un mecanismo de control más robusto para los propósitos de seguimiento sobre dónde y quien ofrece los servicios de *streaming*, útil cuando el sistema de *streaming* se use con un propósito económico.

Adicionalmente para proteger la información que se transmite desde el cliente del servicio de *streaming* y la plataforma de *streaming* se puede hacer uso de protocolos de encriptación como HTTPS y RTMPS. Como ya se mencionó anteriormente RTMPS, al momento del desarrollo del proyecto, no es manejado oficialmente por la plataforma sugerida, y por tal razón no se puede evaluar su desempeño y generar las recomendaciones asociadas, sin embargo se menciona esperando que cuando se establezcan las APIs y se lance la versión estable oficial de Red5 se tenga soporte completo de estas facilidades.

Capítulo 3. Arquitectura de Referencia

3.1 Descripción general del sistema

La arquitectura de referencia que se propone se centra en la distribución de servicios de *streaming* como objetos de aprendizaje, aprovechando que la especificación de SCORM define el SCO como un medio para lograr unos objetivos de aprendizaje y no impone requisitos sobre la manera en que los objetivos son cumplidos.

Gracias a que los LMS han hecho un gran esfuerzo para integrar diferentes estándares se ha conseguido interoperabilidad entre las plataformas en varios niveles. Gracias a SCORM los LMS permiten importar y exportar objetos de aprendizaje estructurados y denominados como paquetes SCORM. Los paquetes SCORM típicamente son almacenados y gestionados por la propia plataforma LMS. Sin embargo debido a la naturaleza de los servicios de *streaming* los objetos de aprendizaje basados en esta tecnología no deberían almacenarse por completo en la plataforma LMS, bien sea que su tamaño es demasiado grande, porque el medio de acceso y ejecución no es tradicional, o porque la funcionalidad extra que ofrece el contenido necesita de una infraestructura extra que potencie los resultados y mejore los beneficios [4].

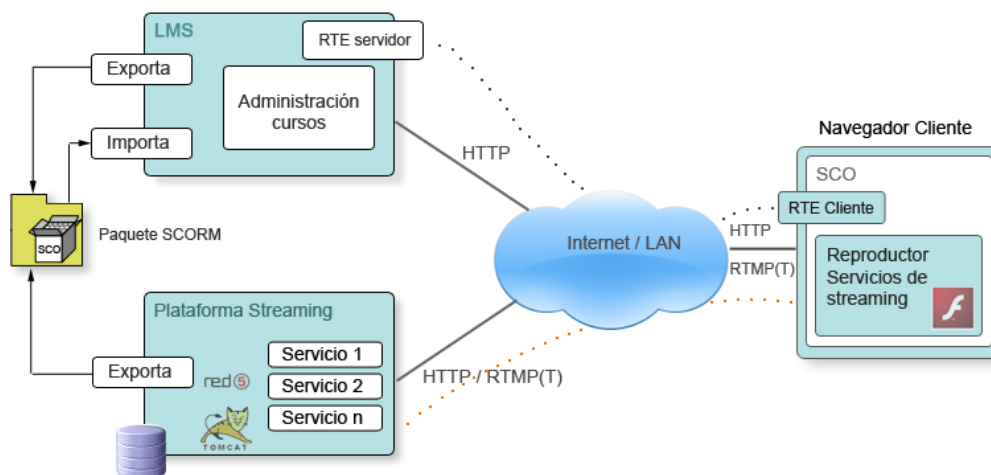


Figura 11. Vista general para la integración de servicios de *streaming*

En nuestra arquitectura se tratan los puntos anteriores bajo la definición de un objeto de aprendizaje de acceso remoto (RALO), *Remote Access Learning Object*, el cual se basa

en un objeto de aprendizaje SCORM tradicional, usado como mecanismo de control y consumo sobre un servicio de *streaming* remoto. Este servicio no necesariamente requiere estar dentro del mismo sistema que el LMS, sino que puede encontrarse en un sistema proveedor de contenido centralizado en la misma red o en Internet.

Como ya se mencionó en el Capítulo 1, existen diferentes versiones del estándar SCORM, la más reciente es SCORM 2004. En las pruebas realizadas para evaluar el comportamiento de los LMS con cada versión SCORM, mediante la *suite* de pruebas que ofrece ADL para validar las implementaciones, se encontró que varios de ellos sólo implementan partes de la especificación, esto impone una limitante en cuanto al soporte y garantías a la hora de integrar el paquete. Por el contrario el estándar SCORM 1.2 define diferentes elementos opcionales, por lo que su implementación es más flexible y evidencia de ello es que la mayoría de LMS ya dan soporte certificado para este estándar [84], por lo que se supone su uso en este trabajo cuando se hable de SCORM.

3.2 Paquete RALO

Consideramos el paquete RALO como un paquete SCORM con características adicionales que permite la comunicación entre el cliente web, que ejecuta el objeto de aprendizaje, y una plataforma remota para consumir algún tipo de servicio de *streaming*.

Algunas de las características más relevantes del paquete RALO son:

- Comunicación bidireccional con el LMS, limitada a los mensajes definidos en el modelo CMI usando el RTE del cliente, basado en un API en Javascript, con el RTE del LMS, implementado bajo su propio lenguaje.
- Comunicación bidireccional con la plataforma de *streaming* mediante un canal basado en el protocolo RTMP(T). El modelo de datos es propio y depende de la información que cada servicio quiera comunicar.
- Establece un canal común sobre el cual se consumen todos los servicios de *streaming*.
- Un entorno de ejecución con capacidades para implementar lógica de negocio compleja, en otros términos la posibilidad de crear clientes pesados.
- Acceso a recursos extra como micrófono, cámara web y archivos del sistema de forma estandarizada, controlada y abstraída independiente del sistema de ejecución (sistema operativo, equipo hardware, etc.)

3.2.1 Estructura

Al igual que un paquete SCORM, el paquete RALO describe la estructura del objeto de aprendizaje tradicional, adicionando la descripción del servicio de *streaming* que se va a consumir.

En su forma más general el paquete incluye:

lmsmanifest.xml, nombre de archivo obligatorio, definido por el estándar SCORM para la descripción del SCO.

ClienteServicio.swf, nombre de archivo opcional, un tipo de archivo de la tecnología Flash que se ejecuta en el navegador del cliente para desplegar la interfaz de usuario que le permite consumir los servicios desde la plataforma de *streaming*.

Index.html, permite cargar los recursos necesarios (swf, archivo de estilos, archivo de javascript) para el correcto despliegue del cliente en el navegador web.

SCORM_API.js, archivo de script para manejar el API del RTE de SCORM y establecer la comunicación necesaria con el LMS.

Descripcion_servicio_streaming.xml, archivo de descripción del servicio de *streaming* que contiene la información necesaria para establecer la comunicación con la plataforma de *streaming*. Su contenido depende directamente del servicio que se quiera consumir, en el anexo digital se presenta un ejemplo del contenido de este documento. En general la información necesaria que se debe tener en cuenta incluye:

- Identificador único del recurso que consume el servicio.
- Identificar los servicios que ofrece el recurso (Audio/video, videoconferencia, chat, juego, etc.)
- Parámetros de validación necesarios para establecer la comunicación con el servidor (nombre de usuario, contraseña, token, etc.).
- Información relevante para el consumo de los servicios de *streaming*.

Adicional a los archivos mencionados, el paquete puede incluir archivos de estilos CSS, imágenes y cualquier otro tipo de recurso digital que mejora la presentación, usabilidad y demás factores relacionados con el uso del mismo.

3.3 Ciclo de vida del paquete RALO

El paquete RALO es generado en la plataforma de servicios de *streaming* mediante una aplicación web que se encarga de crear la estructura requerida por SCORM y adicionar

los recursos digitales (html, javascript, swf, xml) necesarios para su ejecución del lado del cliente.

Una vez exportado el paquete RALO consiste en un archivo comprimido .ZIP, el cual es entregado generalmente al profesor encargado de un curso dentro del LMS.

Teniendo los permisos suficientes el profesor carga en su curso dentro del LMS el paquete RALO, usando las interfaces propias de la implementación de SCORM. Después de esto el paquete RALO existe dentro del repositorio de contenido del LMS, en espera de que cualquier estudiante con los permisos suficientes acceda a este.

Cuando el estudiante accede al curso que contiene el paquete RALO, el LMS descarga los archivos necesarios para que el estudiante pueda hacer uso del objeto de aprendizaje y consumir los servicios.

3.4 Casos de uso esenciales del sistema

La figura 14 presenta los casos de uso mínimos, asociados a los servicios que la arquitectura propuesta debe ofrecer, para facilitar la integración de servicios de *streaming*.

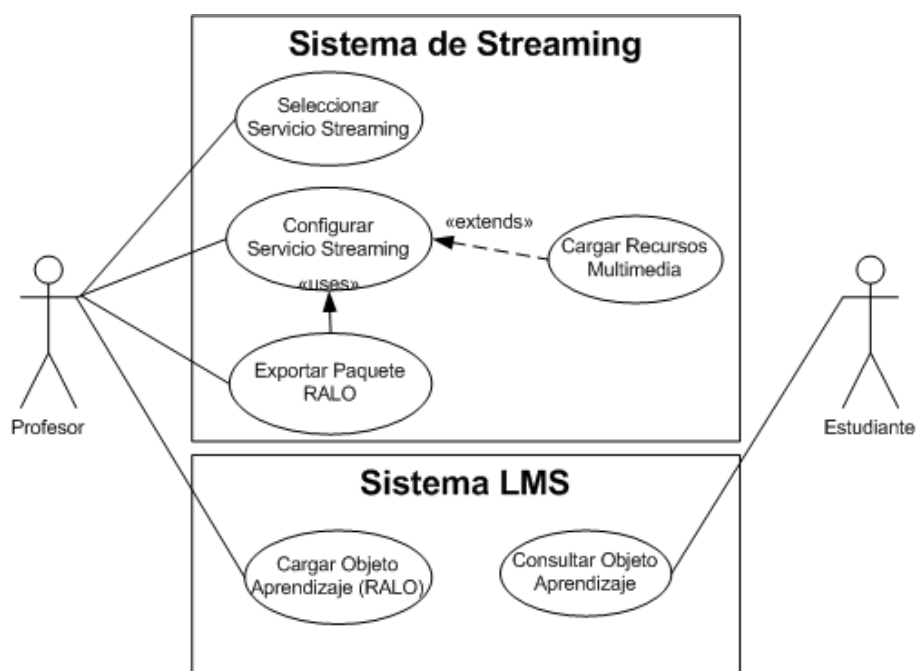


Figura 12. Casos de uso mínimos asociados a la arquitectura de referencia

3.4.1 Actores y entidades

Profesor: Corresponde al usuario que crea los contenidos de aprendizaje, los cuales hacen uso de los servicios de *streaming* para su distribución hacia los estudiantes. Los contenidos pueden tener asociados recursos multimedia como videos, audio, imágenes, etc.

Estudiante: Corresponde al usuario objetivo que accede a los contenidos de aprendizaje, distribuidos mediante los servicios de *streaming*.

Aplicación Web: Corresponde a la aplicación encargada de administrar vía Web información relacionada con los servicios y aplicaciones de *streaming* que se hayan desplegado en la plataforma y realizar algunas acciones sobre los mismos.

Base de datos: Un subsistema encargado de almacenar y administrar información de varios tipos:

- Sobre los usuarios, sus roles y permisos.
- Sobre los servicios y aplicaciones de *streaming* e información relevante para los mismos.
- Sobre los recursos multimedia que puedan existir dentro de la plataforma.

3.4.2 Realización casos de uso esenciales

Caso de uso	Seleccionar Servicio de <i>Streaming</i>
Actores	Profesor (iniciador).
Tipo	Primario.
Propósito	Seleccionar el tipo de servicio de <i>streaming</i> sobre el cual se desea realizar alguna acción.
Resumen	La aplicación web dentro de la plataforma de <i>streaming</i> muestra al profesor un listado de los servicios o aplicaciones de streaming que se encuentran desplegados para que éste seleccione el requerido.
Precondiciones	<ul style="list-style-type: none"> • El Profesor debe encontrarse validado dentro de la aplicación web mediante sus credenciales (Nombre de usuario y contraseña)
Flujo Principal	<ul style="list-style-type: none"> • Este caso de uso inicia cuando el usuario accede a la aplicación Web. • La aplicación Web realiza consultas sobre la de bases de datos para determinar los servicios y aplicaciones de <i>streaming</i> que estén habilitados para el usuario.

- La aplicación Web presenta el listado de los servicios y aplicaciones de *streaming* mediante una interfaz web apropiada.

Pos condiciones

- El usuario puede escoger entre los diferentes resultados para ver en detalle las opciones del servicio o aplicación de *streaming*.

Tabla 7. Caso de uso esencial: Seleccionar servicio de *streaming*

Caso de uso	Configurar Servicio de <i>Streaming</i>
Actores	Profesor (iniciador).
Tipo	Primario.
Propósito	Modificar el servicio o aplicación de streaming de acuerdo a necesidades particulares para satisfacer los objetivos de la actividad formativa que se quiera presentar a los estudiantes.
Resumen	Cada servicio o aplicación de <i>streaming</i> tiene asociado un conjunto de parámetros que permiten adaptar el servicio a las necesidades requeridas, por ejemplo en el servicio de video-streaming, algunos parámetros de configuración serian el recurso multimedia que se va a distribuir y las dimensiones por defecto del mismo.
Precondiciones	<ul style="list-style-type: none"> • El Profesor debe encontrarse validado dentro de la aplicación web mediante sus credenciales (Nombre de usuario y contraseña)
Flujo Principal	<ul style="list-style-type: none"> • Inicia después de que el profesor obtiene el listado de los servicios o aplicaciones que tiene a su disposición y selecciona una de ellas. • La aplicación Web muestra las opciones de configuración que ofrece el servicio, siendo esta información propia de cada servicio. • El usuario puede modificar los parámetros de acuerdo a sus necesidades y almacenarlas para luego crear el objeto de aprendizaje.
Pos condiciones	<ul style="list-style-type: none"> - El usuario puede exportar servicio o aplicación de streaming con las configuraciones realizadas como un objeto de aprendizaje para ser usado por el LMS

Tabla 8. Caso de uso esencial: Configurar servicio de *streaming*

Caso de uso		Cargar Recurso Multimedia
Actores	Profesor (iniciador).	
Tipo	Primario.	
Propósito	Permite al profesor subir a la plataforma de streaming los recursos multimedia (audio, video, imágenes, etc.) que pueden ser utilizados por los servicios de streaming para su funcionamiento.	
Resumen	El profesor puede crear diferentes recursos multimedia que mejoran la experiencia de aprendizaje, estos recursos son distribuidos por los servicios de streaming. La plataforma debe permitir cargar los recursos para realizar sobre estos algunos procesos como adaptación de formatos, recodificación, indexación y su posterior distribución	
Precondiciones	<ul style="list-style-type: none"> El Profesor debe encontrarse validado dentro de la aplicación web mediante sus credenciales (Nombre de usuario y contraseña) 	
Flujo Principal	<ul style="list-style-type: none"> Inicia cuando el usuario se encuentra configurando un servicio que requiere de un recurso multimedia La aplicación Web ofrece los mecanismos requeridos para que el usuario realice la carga del recurso hacia el servidor. La aplicación web utiliza servicios internos para almacenar el recurso y programar, opcionalmente, actividades relacionadas con la adaptación o recodificación del recurso en caso de necesitarlo. Cuando se almacena queda disponible para que el usuario lo utilice como contenido distribuible mediante algún servicio de streaming. 	
Pos condiciones	<ul style="list-style-type: none"> El usuario puede hacer uso del recurso en las diferentes aplicaciones o servicios disponibles en la plataforma. 	

Tabla 9. Caso de uso esencial: Cargar recurso multimedia

Caso de uso		Exportar Paquete RALO
Actores	Profesor (iniciador).	
Tipo	Primario.	
Propósito	Permite al usuario exportar el objeto de aprendizaje que consume los	

	servicios de streaming, previamente configurado.
Resumen	La aplicación Web ofrece los mecanismos necesarios para exportar el paquete RALO, bajo la especificación SCORM, que contiene el cliente que consume los servicios de streaming.
Precondiciones	<ul style="list-style-type: none"> • El Profesor debe encontrarse validado dentro de la aplicación web mediante sus credenciales (Nombre de usuario y contraseña) • El Profesor debe haber configurado previamente el servicio para que cumpla con los objetivos establecidos.
Flujo Principal	<ul style="list-style-type: none"> • Inicia cuando el profesor termina de configurar un servicio o aplicación de streaming. • La aplicación Web realiza diferentes procesos para crear el objeto de aprendizaje que permite consumir los servicios de streaming por parte del estudiante, de igual forma crea los descriptores asociados a la especificación SCORM y finalmente empaqueta mediante el formato ZIP los archivos necesarios para este propósito. • El paquete es entregado al profesor para que lo almacene en su equipo de trabajo.
Pos condiciones	<ul style="list-style-type: none"> - El usuario puede utilizar el paquete RALO, basado en el estándar SCORM en el LMS que soporte el mismo estándar.

Tabla 10. Caso de uso esencial: Exportar paquete RALO

Caso de uso	Cargar Objeto de Aprendizaje (RALO)
Actores	Profesor (iniciador).
Tipo	Primario.
Propósito	Permite al usuario cargar el paquete bajo el estándar SCORM en la plataforma LMS
Resumen	Dentro del sistema LMS que implementa el estándar SCORM se permite al profesor cargar los objetos de aprendizaje que se pueden ser reproducidos por los estudiantes dentro de un curso virtual en particular
Precondiciones	<ul style="list-style-type: none"> • El LMS debe soportar el estándar SCORM • El Profesor debe estar validado dentro del LMS con sus credenciales (Nombre de usuario y contraseña).

Flujo Principal

- Inicia cuando el profesor selecciona la opción de subir el objeto de aprendizaje en la plataforma LMS.
- El LMS ofrece la interfaz web para cargar el objeto de aprendizaje.
- El profesor selecciona el paquete, previamente exportado por la plataforma de streaming, y lo sube en el LMS.
- El LMS lo almacena y publica para su consumo por parte de los estudiantes.

Tabla 11. Caso de uso esencial: Exportar paquete RALO

Caso de uso	Consultar Objeto de Aprendizaje
Actores	Estudiante (iniciador).
Tipo	Primario.
Propósito	Permite al estudiante acceder al objeto de aprendizaje desde el LMS
Resumen	Dado que el LMS soporta el estándar SCORM
Precondiciones	<ul style="list-style-type: none">• El LMS debe soportar el estándar SCORM• El Estudiante debe estar validado dentro del LMS con sus credenciales (Nombre de usuario y contraseña).
Flujo Principal	<ul style="list-style-type: none">• Inicia cuando el estudiante solicita el objeto de aprendizaje en el LMS.• El LMS retorna la interfaz propia para la ejecución del objeto de aprendizaje, que incluye el RTE basado en Javascript para la comunicación entre el LMS y el objeto de aprendizaje.
Pos condiciones	<ul style="list-style-type: none">- El usuario interactúa con el objeto de aprendizaje

Tabla 12: Caso de uso esencial: Consultar objeto de aprendizaje

3.5 Referencia para el diseño de la arquitectura

La arquitectura de referencia para la integración de servicios de streaming se basa en dos propuestas para su diseño, la primera asociada a la organización de los componentes que

ofrecen el servicio de streaming y la segunda, la forma en que se consumen dichos servicios desde la plataforma LMS.

Lo primero que se debe tener en cuenta es la necesidad de contar con un sistema distribuido que permita a los diferentes sistemas (LMS y Plataforma de *streaming*) interactuar para cumplir un objetivo común, en nuestro caso el de proveer un servicio de educación virtual basado en tecnología *streaming*. La tendencia actual en el área de sistemas distribuidos se enfoca en el desarrollo de sistemas basado en servicios, ya sea siguiendo el enfoque SOA, *Service Oriented Architecture* [85], MDA, *Model Driven Architecture* [86] u OGSA, *Open Grid Services Architecture* [87, 88], esta última enfocada al desarrollo de sistemas de tipo empresarial o sistemas que requieren una infraestructura con alto nivel de procesamiento, normalizando los servicios habituales (Gestión de tareas, de recursos, seguridad, etc.), para una aplicación GRID [89], definiendo además las interfaces para estos servicios [90]. OGSA cumple sus objetivos mediante foros abiertos a la comunidad interesada para el diseño de arquitecturas, exploración y pruebas de nuevas tendencias en temas relacionados o sobre la propia arquitectura, compartiendo buenas estrategias y consolidando todo el trabajo como nuevos estándares.

Como base para la arquitectura tenemos en cuenta la propuesta de Amoretti [91], basada en OGSA, en relación al diseño de los servicios de distribución multimedia, mediante *streaming*, en un ambiente *e-learning*. La figura 13 resume la propuesta de Amoretti[91].

Los servicios propuestos en 89, se presentan como herramientas para tareas convencionales en este tipo de sistemas:

Servicio AAA, *Authentication, Autorization y Accounting*, descritos como servicios básicos que deben existir dentro de la plataforma, para los procesos de validación de credenciales, asignación de permisos y para el control de los créditos necesarios para la reproducción de los recursos o acceso a los servicios adicionales.

El servicio de adaptación de contenido, está enfocado a la recodificación de recursos audio/visuales, visto como un proceso de encolamiento, es decir, procesa uno a uno los recursos multimedia a medida que se cargan dentro de la plataforma, de tal manera que para un recurso original pueden existir n versiones diferentes recodificadas con diferentes parámetros, los más importantes el alto, ancho y la tasa de bits para los videos y el la frecuencia de muestre y tasa de bits para los audios.

Servicios de almacenamiento, estructurados como subsistemas independientes (*Multimedia Content, Metadata, Local Search Service*) que permiten gestionar de forma efectiva y mediante colaboración automática los recursos multimedia y demás información descriptiva. Los servicios descritos presentan un primer acercamiento a la necesidad de

contar con un sistema de almacenamiento y distribución avanzado que puede ser implementado mediante soluciones hardware y software especializados para estas tareas, por ejemplo, redes de distribución de contenido, CDN [97].

Los servicios de *Networking Monitoring Service* y *QoS Management Service*, como servicios para garantizar calidad del servicio, QoS, están enfocados como servicios para la capa de transporte, específicamente para el protocolo UDP sobre el cual se soporta el protocolo RTP, usado como protocolo para el servicio de *streaming*.

El servicio *Multimedia Service* se propone como el servicio de *frontend*, punto de entrada de la plataforma e interfaz de comunicación hacia el exterior, el cual orquesta los demás servicios para ofrecer la funcionalidad requerida.

Al usar como referencia este tipo de trabajo, se mejora la calidad de la propuesta al tener una solución probada efectivamente en un entorno e-learning y soportada por la investigación de un grupo mayor de profesionales. Aunque dicha arquitectura se limita a los servicios de audio y video, la arquitectura propuesta en este documento retoma algunos de los conceptos para la definición de los servicios requeridos, agrupándolos en la plataforma de servicios de *streaming*, y así dar soporte a las nuevas aplicaciones, teniendo en cuenta que se define y propone una capa de conectividad o *middleware* [92] basado en tecnología Flash.

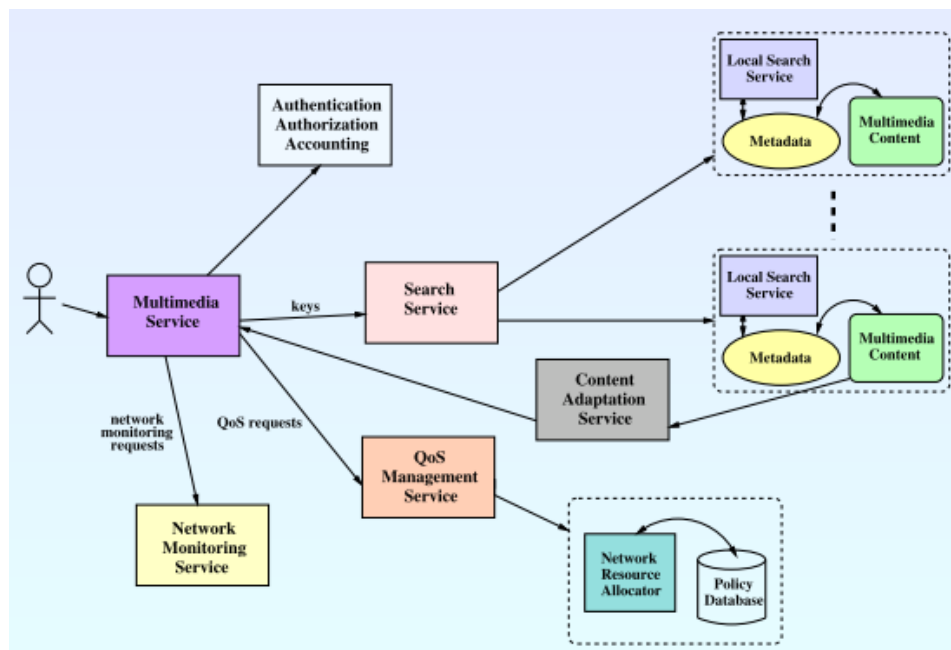


Figura 13. Organización para los servicios multimedia. [91]

La segunda parte de la definición de la arquitectura está asociada al consumo de los servicios, para este objetivo se utiliza SCORM como medio de encapsulación para el cliente de consumo de servicios de streaming. Esta elección ya ha sido probada en otros proyectos [93-96] que buscan la extensibilidad de la plataforma LMS para varios propósitos, la arquitectura de referencia va un paso más adelante, proponiendo un mecanismo de extensibilidad de los elementos definidos en un nuevo modelo de datos basado en el estándar CMI, descrito en el capítulo 2 y presentando un ejemplo de implementación en el anexo 4, con el objetivo de mejorar los procesos de seguimiento del estudiante y del contenido. La figura 14 muestra el cambio básico necesario a lo definido en SCORM para la extensibilidad de los servicios de la plataforma LMS.

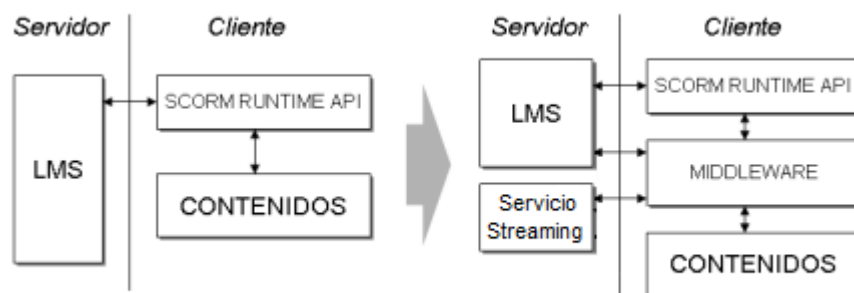


Figura 14. Adición de middleware para extensión de servicios en el LMS

La adición de la capa de *middleware* permite mantener la integridad del estándar SCORM, es decir, no se agregan modificaciones sobre los mecanismos de comunicación entre el LMS y el RTE en el cliente, pero el *middleware* permite la comunicación con interfaces, que representen lógica de negocio específica, dentro del LMS o hacia otro sistema como la plataforma de *streaming*.

3.6 Secuencia en la arquitectura

Enfocada sobre los objetos de aprendizaje SCO, como mecanismo esencial para el consumo de los servicios de streaming y en consecuencia su integración con la plataforma LMS. La figura 15 muestra la secuencia que se sigue para la arquitectura de referencia propuesto.

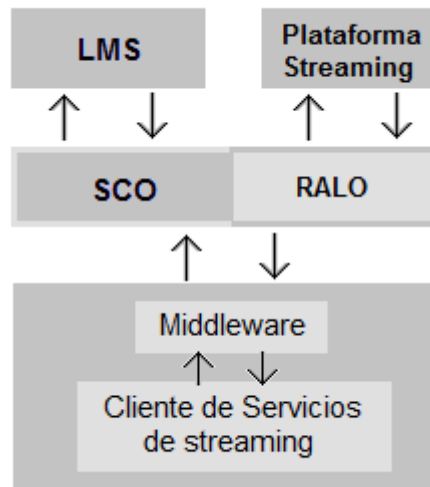


Figura 15. Secuencia seguida por la arquitectura de referencia

Basado en la figura 15:

1. El LMS determina el SCO a lanzar.
2. El LMS ejecuta el SCO conforme al estándar SCORM
3. El SCO ejecutado, comportándose como un RALO, inicializa y carga el servicio desde la plataforma de streaming.
4. El estudiante interactúa con el servicio, mediante el cliente del servicio de streaming.
5. El servicio se comunica con el LMS y la plataforma de streaming a través del middleware (Envía mensajes basado en CMI y otros).

3.7 Descripción detallada de la arquitectura

En la figura 16 se muestra en detalle los componentes que conforman la arquitectura propuesta para la integración de servicios de *streaming*, también se destacan con numerales componentes e interacciones más relevantes, las cuales se describen a continuación.

(1) Se muestra la actividad de una aplicación web, denominada StreamelWeb, cuyo objetivo entre otros es el de exportar los paquetes bajo el estándar SCORM. Las demás funcionalidades se relacionan con la gestión de información de estudiantes, profesores,

tipos de servicios, asociados a las recomendaciones establecidas para el modelo de datos de la plataforma de *streaming* en la sección 2.5.6.2.

(2) Se muestra al LMS, el cual no requiere ningún tipo de modificación. Se aprovecha las funcionalidades que este provee para la instalación del paquete SCORM.

(3), (4) El paquete SCORM, en su forma más general, define un SCO, almacenado y gestionado dentro del repositorio de contenidos del LMS, agrupa diferentes recursos digitales como páginas web, archivos de estilos, de scripts, entre otros, los cuales al interactuar buscan ofrecer una experiencia de aprendizaje para el estudiante.

(5) El SCO define un recurso HTML inicial, el cual es la primera interfaz de interacción que se presenta al estudiante cuando accede al objeto de aprendizaje.

(6) El navegador web es la herramienta con la cual es estudiante accede a los servicios de la plataforma LMS y en consecuencia al objeto de aprendizaje. El acceso a los objetos de aprendizaje se realiza mediante el protocolo de comunicación web tradicional HTTP.

(7), (8) El recurso HTML (5) es interpretado por el navegador web. El HTML permite asignar y calcular todas las propiedades de un objeto antes de mostrarlo correctamente en pantalla y en el navegador para su correcta visualización, éstos incluyen etiquetas de carga de estilos, scripts, y la etiqueta necesaria para cargar el entorno de ejecución de FlashPlayer.

(9) Cuando se carga el entorno FlashPlayer se hacen invocaciones HTTP hacia la plataforma LMS para descargar la aplicación Flex. Ésta a su vez puede invocar recursos adicionales como imágenes, archivos de configuración, documentos XML, entre otros, para su correcta ejecución.

(10) Presenta una estructura general de la aplicación Flex que permite consumir los servicios de *streaming*. La aplicación puede estar estructurada mediante módulos (definidos en la tecnología Flex) que consuman servicios específicos y que permita su re- uso entre diferentes aplicaciones.

(11), (12) La aplicación Flex establece un mecanismo de comunicación con la plataforma de *streaming* mediante el protocolo RTMP(T), a través de la cual se consumen los diferentes servicios de *streaming*.

(13) Para los servicios de *streaming* que hagan uso de recursos multimedia, se descarga de la plataforma el documento descriptor basado en el estándar MPEG7 asociado a dicho recurso o al mismo servicio.

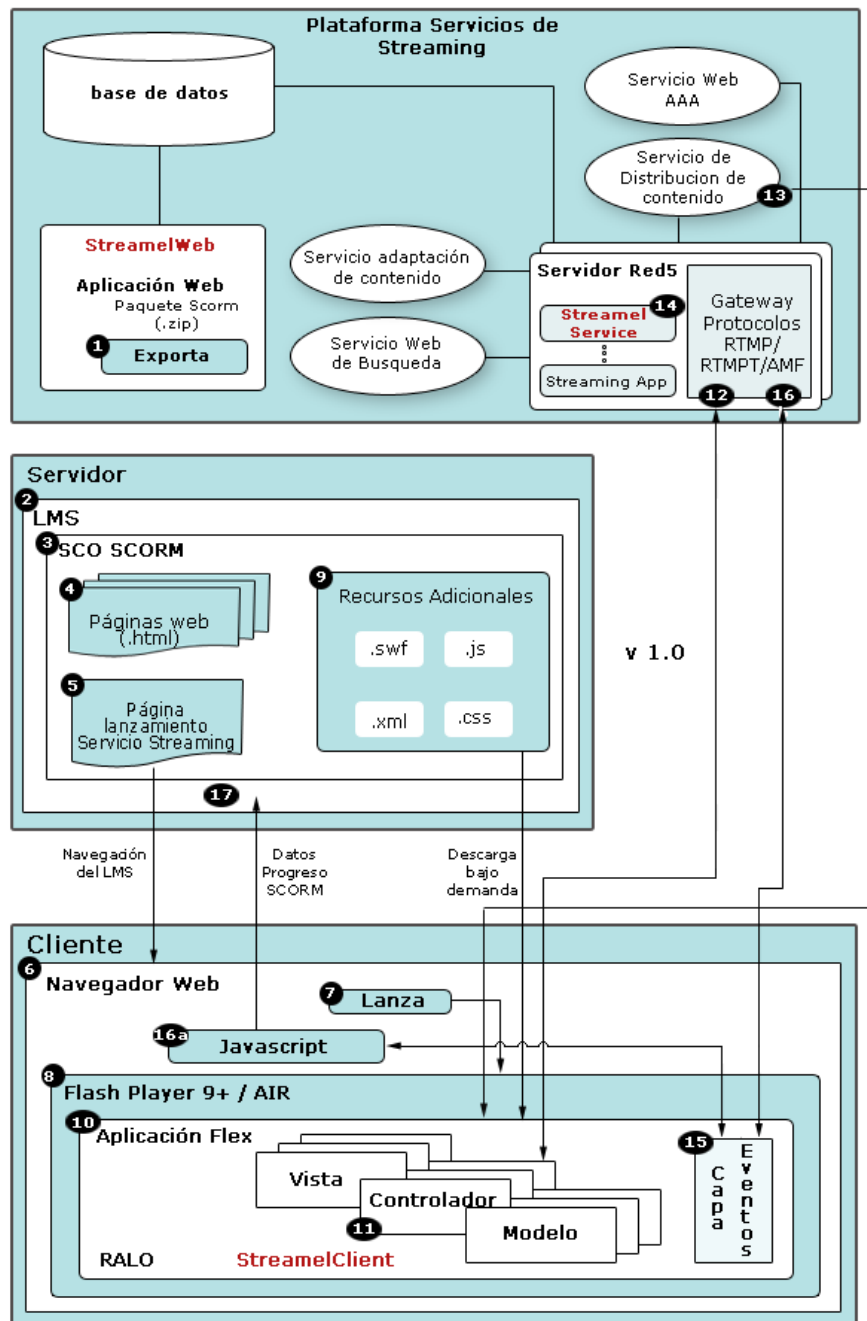


Figura 16. Arquitectura de referencia detallada

(14) Cuando se tiene establecida la comunicación con la plataforma de *streaming*, la aplicación StreamelClient consume los servicios, denominados en la figura 16 como StreamelServices. Este módulo ofrece la funcionalidad específica de la distribución de contenidos como video, audio y diapositivas, para ello se soporta en módulos internos de la plataforma como servicios de adaptación de contenido, servicios de consulta sobre una base de datos, etc.

(15) Los SCO de SCORM definen mecanismos de comunicación para interactuar con el LMS, dado que dentro de la arquitectura este SCO está concebido como un RALO, soportado en su mayoría por la tecnología Flash/Flex, se define una capa que permite enviar información hacia el LMS (16a), (17) y hacia la propia plataforma de *streaming* (16b), con el objetivo que establecer mecanismos de trazabilidad del servicio y en consecuencia identificar la calidad del material.

Para complementar la descripción de la plataforma, la figura 17 presenta un diagrama de actividad entre los módulos más importantes (StreamelClient, StreamelService y el propio LMS), la figura 18 muestra de manera más clara, a partir de lo descrito anteriormente, la dinámica de la arquitectura mediante un diagrama de secuencia y para finalizar, se describen los mecanismos de comunicación mediante un ejemplo de la aplicación específica de video streaming en la figura 21.

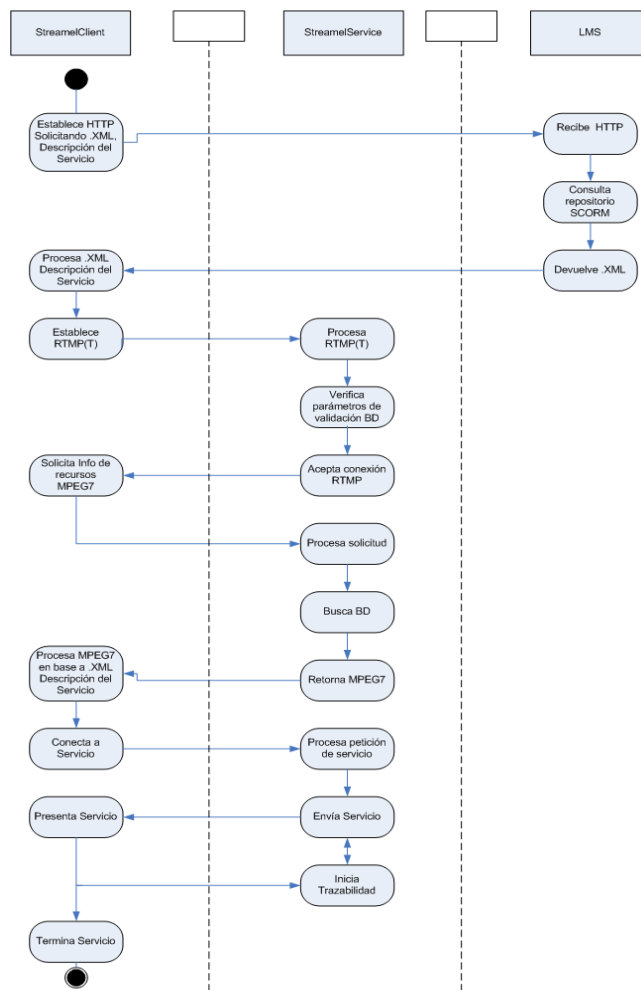


Figura 17. Diagrama general de actividad del sistema

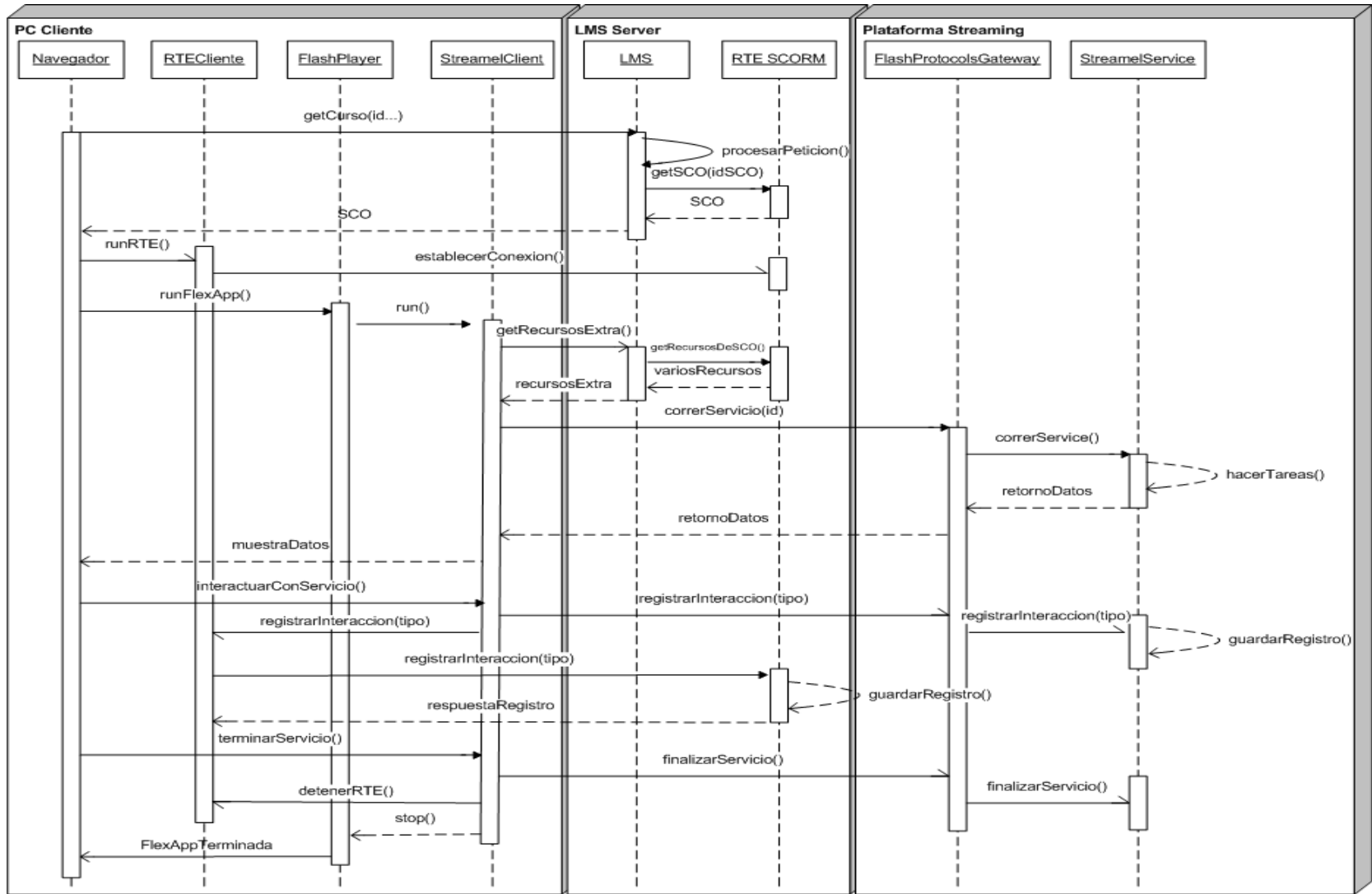


Figura 18. Diagrama general de secuencia para la arquitectura de referencia.

3.7.1 StreamelClient

La aplicación basada en tecnología Flex se ejecuta sobre el cliente FlashPlayer en el navegador web. Como requerimiento inicial se establece el uso como mínimo de la versión 9.0.115.0 de FlashPlayer, que incluye, entre otras, soporte para ActionScript 3, códecs de audio y video, basados en estándares abiertos, HE-AAC y H.264 respectivamente [97].

También es necesario definir un mecanismo que permita cargar el entorno de ejecución de FlashPlayer garantizando que los diferentes navegadores funcionen de acuerdo a lo esperado, para esa tarea se hace uso de las utilidades del script AC_OETags.js, desarrollado por Macromedia, ahora parte de Adobe Inc.

Para hacer uso del API de SCORM desde el cliente Flex, se hace uso de la herramienta SCORM API Wrapper para ActionScript 3, desarrollada por Philip Hutchison [98].

Los componentes más relevantes para la aplicación Flex se describen en la figura 19.

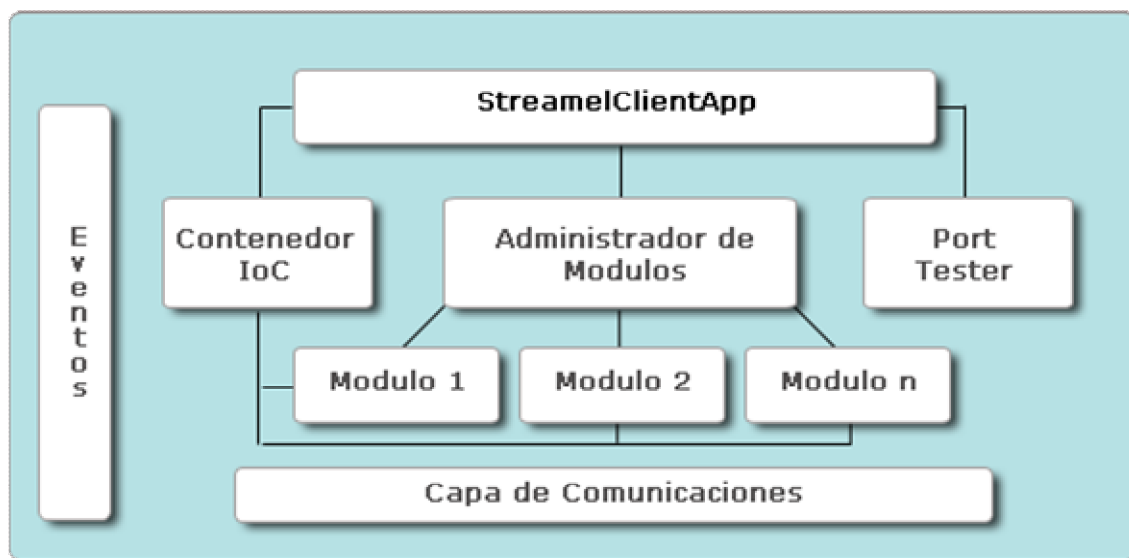


Figura 19. Componentes básicos de la aplicación Flex StreamelClient.

StreamelClientApp, es el punto de acceso e inicio de la aplicación, se encarga de inicializar el contenedor para la inyección de dependencias, cargar la información necesaria para configurar los servicios, establecer los canales de comunicación con el LMS y la plataforma de *streaming*, usando el componente "Port Tester".

El contenedor de inyección de dependencias o IoC, *inversion of control*, facilita la estructuración de los componentes para mantener una baja cohesión entre las dependencias del mismo y así facilitar la extensibilidad de la aplicación.

El administrador de módulos, como su nombre lo indica, se encarga de la gestión de los módulos que ofrecen una funcionalidad específica asociada a uno o varios servicios de *streaming*, esto es inicializar, reproducir y detener cada módulo. La inicialización de los módulos implica, por ejemplo, la inyección del canal de comunicación que se va a utilizar para consumir algún servicio de *streaming*.

El componente port tester, se trata de una utilidad que permite establecer el canal de comunicaciones, identificando de forma automática limitaciones de la red (reglas de firewall, limitaciones NAT, entre otras) y solventándolas usando estrategias como prueba conexión en diferentes puertos o enmascarando la conexión sobre HTTP usando el protocolo RTMPT.

La capa de comunicaciones sirve como mecanismo de abstracción para cualquier otro componente para la comunicación con el LMS y la plataforma de *streaming*. En cualquier caso la plataforma gestiona si debe enviar la información de control a los dos sistemas o sólo a uno de ellos, basado en los tipos de mensajes que soporta cada sistema.

La capa de eventos permite que cualquier componente registre y escuche eventos de cualquier tipo, ya sean eventos asociados a la interfaz de usuario o eventos de comunicación de red o de control.

Los módulos son los componentes que permiten al usuario acceder, usar e interactuar con el servicio de *streaming*. Incluyen su propia interfaz de usuario, lógica de negocio y definición de datos para funcionar correctamente.

3.7.2 StreamelWeb

StreamelWeb es el servicio que permite exportar el objeto de aprendizaje bajo el estándar SCORM, de acuerdo a lo especificado en el literal 3.3.1, sobre la estructura del paquete RALO. Este servicio puede ser considerado como una aplicación web tradicional por lo que su diseño puede hacerse usando tecnologías convencionales, Jsp, Spring Web MVC, Java Server Faces, Struts, etc. Se sugiera el uso de este tipo de tecnologías aprovechando las capacidades del servidor Tomcat, del lado de la plataforma de *streaming*, y así reducir la necesidad de incluir más servidores.

3.7.3 StreamelService

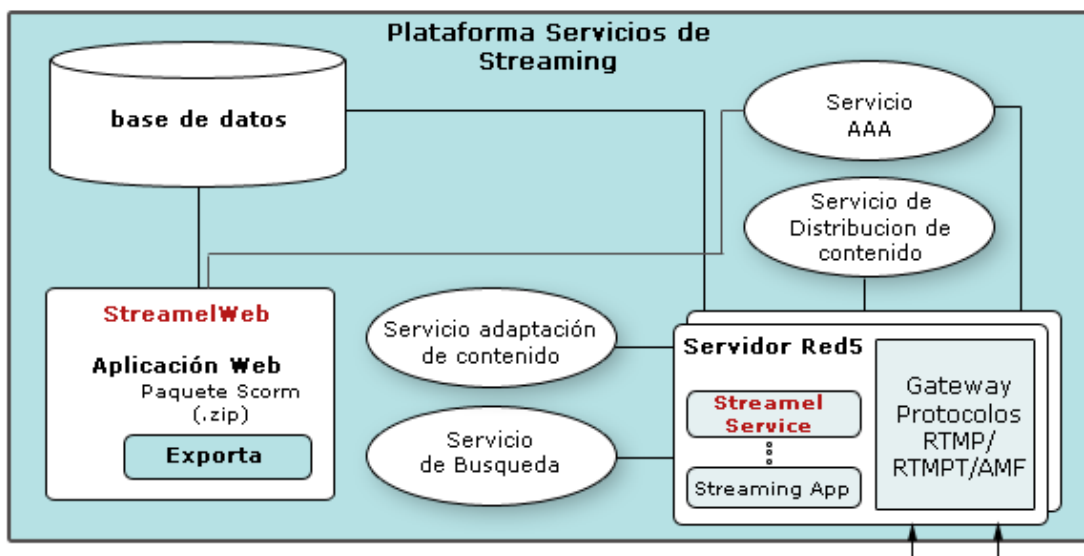


Figura 20. Plataforma para el despliegue de servicios de *streaming*

El StreamelService representa la aplicación que implementa el algoritmo necesario para ofrecer el servicio de *streaming*, por ejemplo para un servicio de video *streaming*, la aplicación tiene en cuenta la ubicación del recurso multimedia que va a entregar, así mismo hace uso de las funcionalidades del servicio de adaptación de contenido para enviar el recurso que mejor se desempeñe sobre las limitantes de red del cliente.

Para facilitar el desarrollo de nuevos servicios de *streaming* es necesario ofrecer un conjunto de herramientas para realizar tareas comunes. Dichas tareas van desde la validación de los permisos que un cliente tiene sobre dicho servicio, hasta la carga de un recurso multimedia para su distribución. Algunas tareas pueden parecer obvias, pero debe tenerse en cuenta que los servicios soportan llamados concurrentes, es decir, invocaciones desde diferentes hilos de forma asíncrona.

Para las herramientas se destacan las características requeridas, más no la forma en que se deben implementar:

- Disponibilidad concurrente, la herramienta debe soportar su ejecución de forma concurrente, es decir, puede ser llamada desde varios hilos al mismo tiempo.
- Facilidad de crecimiento, la herramienta debe crecer a medida que los servicios lo requieran.
- Reusabilidad, las herramientas pueden ser usadas por diferentes componentes.

Algunas de las herramientas comunes se presentan en la figura 20 y se explican de forma conceptual a continuación:

Servicio AAA, Authentication, Authorization y Accounting

Se trata de un servicio necesario para las tareas de autenticación, autorización y contabilización, este último se refiere al seguimiento del consumo de los recursos, en este caso los servicios de *streaming*, pudiéndose asociar también para los procesos de trazabilidad.

Servicio de adaptación de contenido

Este servicio es vital para las tareas que requieren el envío de recursos multimedia como audio, video e imágenes bajo condiciones especiales, por ejemplo:

- Video a resoluciones bajas para mantener una presentación continua para clientes con bajo ancho de banda.
- Imágenes redimensionadas para dispositivos móviles.
- Transcodificación para mantener la compatibilidad con tecnologías alternativas.

Se aclara que los servicios que requieran adaptación de contenido en vivo, sobre todo para la re-codificación de video, deben analizarse con mucho cuidado, debido a las necesidades de alto procesamiento requerido.

Servicio de distribución de contenido

Visto para la distribución de recursos digitales para los servicios de *streaming*. Este tipo de servicios pueden estar formados por soluciones puramente software, que abstraen el acceso y almacenamiento a los recursos digitales o soluciones software/hardware como CDN [99], "*content delivery network services*".

En la figura 21 se presenta la secuencia básica de comunicación entre los diferentes componentes del lado del servidor para un servicio básico de video *streaming*.

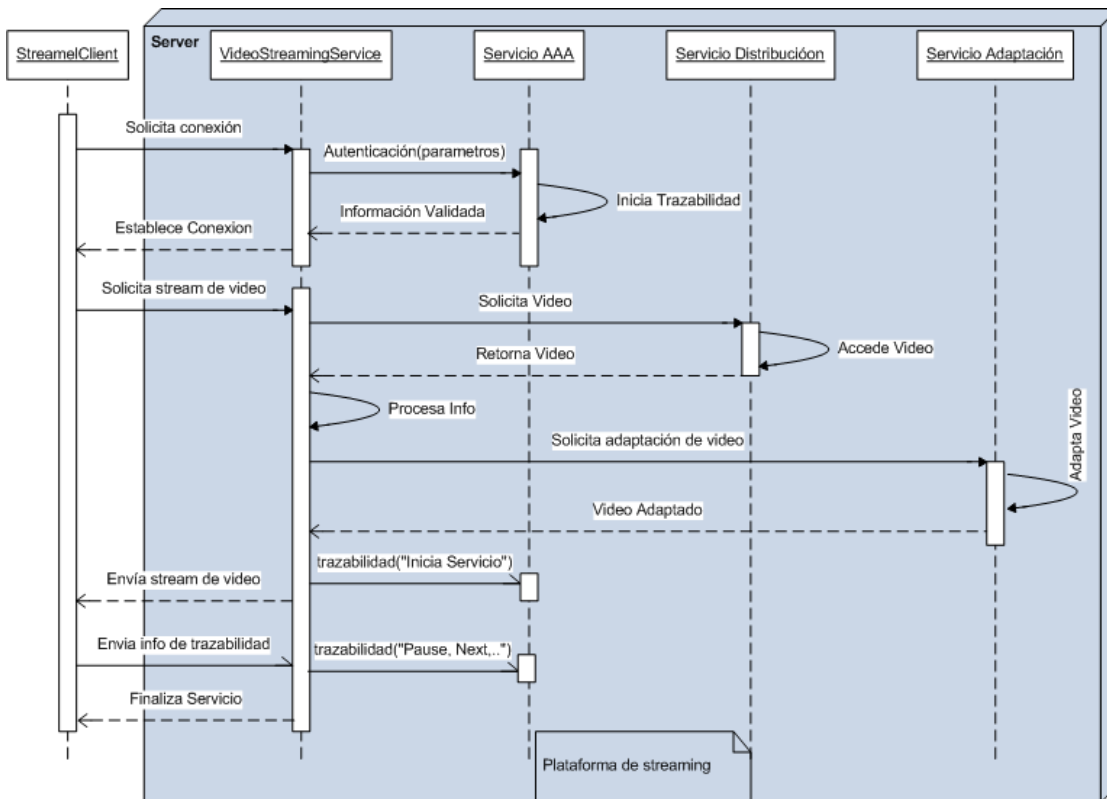


Figura 21. Diagrama de secuencia para servicio de video *streaming* básico

Capítulo 4. Validación de los lineamientos para la integración de servicios de *streaming*

4.1 Introducción

Para la validación de los lineamientos presentados en el Capítulo 2 sobre la integración de servicios de *streaming* y de la arquitectura de referencia presentada en el Capítulo 3, se presenta el caso de estudio incluyendo los siguientes aspectos:

- Definición de los requerimientos mínimos de los servicios de *streaming* que se van integrar a la plataforma EVA para la educación virtual dentro de la universidad del Cauca.
- Aplicación de los lineamientos metodológicos para la caracterización de los servicios de *streaming* y cómo se logra su implementación.
- Aplicación de los lineamientos de índole técnica para la implementación de la plataforma de servicios de *streaming* basados en la arquitectura de referencia del capítulo 3.
- Resultados obtenidos durante las fases de pruebas del prototipo.

4.2 Descripción de los requerimientos del servicio

Con el desarrollo del caso de estudio se pretende integrar un servicio de video streaming, al entorno virtual, que permita a los estudiantes acceder a la actividad de aprendizaje que normalmente sigue una metodología de exposición apoyada con diapositivas (clase magistral tradicional) en el modelo de educación presencial tradicional. Adicionalmente es necesario mantener las ventajas mínimas que ofrecen los LMS, esto es, la disponibilidad del material en cualquier momento y lugar.

Lo anterior describe de forma general el servicio de aprendizaje que se necesita analizar. Según lo mencionado en el capítulo 2, el material está representado en las palabras del expositor, en el texto e imágenes que pueden encontrarse en las diapositivas y en las notas que pueda tomar el estudiante, con esta información podemos iniciar el proceso de

aplicación del modelo M3 para la selección de la tecnología de *streaming* con la cual se va a implementar el servicio dentro del ambiente virtual.

Las características del modelo permiten evaluar los servicios, aplicaciones y tecnologías con las cuales se pueden presentar actividades de formación. En esta sección evaluamos las tecnologías en su forma más general, pero se pueden evaluar aplicaciones específicas de la tecnología una vez hayan sido definidas, por ejemplo evaluar servicios de comentarios con componente temporal, reproductor de diapositivas interactivas, video *streaming* con características de accesibilidad, etc.

4.3 Aplicación del modelo M3

Siguiendo las recomendaciones descritas para el modelo M3, el proceso inicia describiendo el servicio tradicional mediante el modelo M1, luego se relacionan los resultados del modelo M1 con las características de los servicios descritos en el modelo M2, finalmente se hace un refinamiento de las tecnologías, de acuerdo a las características y requerimientos particulares establecidos con anterioridad.

4.3.1 Aplicación Modelo M1.

Se caracteriza el servicio que se quiere llevar al ambiente virtual

Categorías del Modelo M1	Descripción
Entrega del material	Texto e imágenes en diapositivas Texto en las propias notas del estudiante Palabras del expositor Expresiones corporales del expositor
Interacción con el material	Con las notas escritas por el estudiante Con las diapositivas, si las entrega el expositor
Interacción con el profesor	Preguntas durante la exposición o en horarios de consulta
Interacción entre estudiantes	No aplica, aunque puede existir de manera informal fuera de la sesión de exposición
Realimentación del material	No aplica de forma directa, puede determinarse mediante exámenes o encuestas.
Modelo M1	

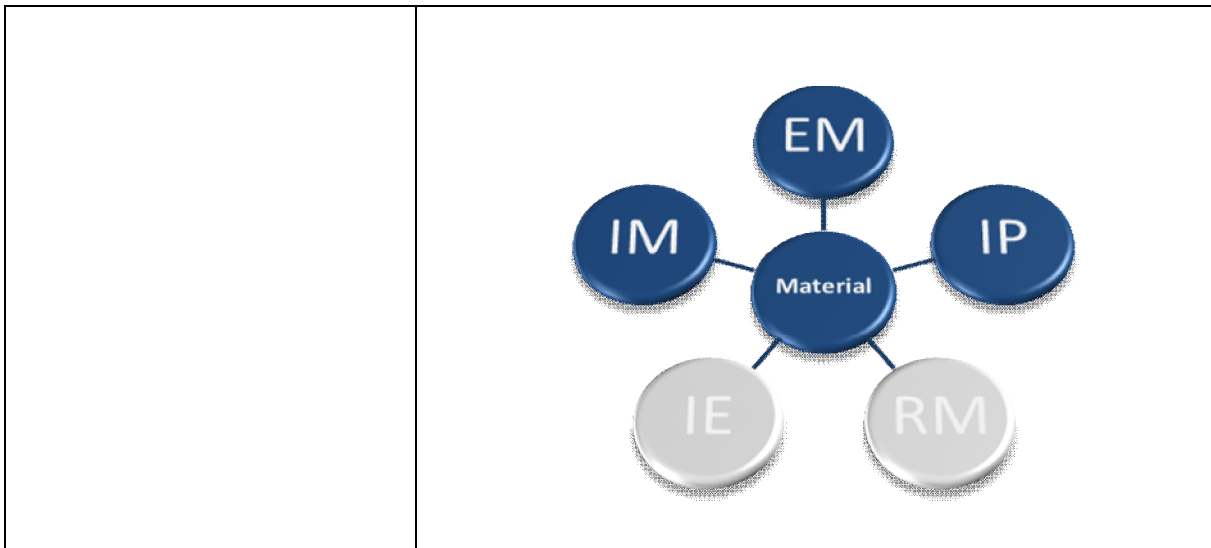


Tabla 13. Modelo M1 para caso de estudio

De este primer modelo podemos concluir que debido a la condición sobre la disponibilidad en cualquier momento del material dentro del LMS, la interacción con el profesor queda condicionada debido a que los estudiantes y el profesor pueden usar el contenido en intervalos de tiempo diferentes.

4.3.2 Aplicación Modelo M2

Con base en la tabla 13, y la información determinada del modelo M1 en capítulo 2, podemos identificar las tecnologías basadas en *streaming* que pueden ser utilizadas para la implementación de los servicios, teniendo en cuenta solo las interacciones que se deberían considerar para la actividad estudiada.

		Interacciones M1				
		EM	IM	IP	IE	EF
Aplicaciones de <i>streaming</i>	Video/Audio Bajo Demanda	✓	✓			
	Video/Audio Vivo	✓	✓			
	Videoconferencia Punto-Punto	✓	✓	✓		✓
	Videoconferencia Multi-punto	✓	✓	✓	✓	✓
	Shared Objects	✓	✓		✓	✓
	Callbacks ó Tecnología Push	✓				
	<i>Streaming</i> RPC	✓	✓	✓	✓	✓
	Descarga progresiva	✓	✓			

Tabla 14. Relación dinámica del material con servicios de *streaming* para caso de estudio.

Al igual que en el punto anterior, la interacción con el profesor para las aplicaciones de video conferencia se ven afectadas por la condición inicial sobre el momento en el cual el estudiante y el profesor pueden hacer uso del servicio.

En este punto es evidente que la interacción con el profesor, debido a la condición de acceso en cualquier momento, puede lograrse usando otros mecanismos tradicionales que provee el LMS como foros, e-mail, etc.

Para favorecer la dinámica del caso de estudio y probar las metodologías de educación que tratan de transformar al estudiante como actor activo dentro del proceso de formación, se pretende agregar a las dinámicas definidas la interacción entre estudiantes y como valor agregado la necesidad de obtener información relevante para determinar la Realimentación del material. Teniendo en cuenta lo anterior, la tabla 14 se actualiza:

		Interacciones M1				
		EM	IM	IP	IE	EF
Aplicaciones de <i>streaming</i>	Video/Audio Bajo Demanda	✓	✓			
	Video/Audio Vivo	✓	✓			
	Videoconferencia Punto-Punto	✓	✓	✓		✓
	Videoconferencia Multi-punto	✓	✓	✓	✓	✓
	Shared Objects	✓	✓		✓	✓
	Callbacks ó Tecnología Push	✓				
	Streaming RPC	✓	✓	✓	✓	✓
	Descarga progresiva	✓	✓			

Tabla 15. Relación dinámica del material con servicios de *streaming* para caso de estudio refinada.

4.3.3 Refinamiento de tecnologías.

Para realizar el refinamiento y viabilidad de los servicios de *streaming* se hacen algunos supuestos, el primero asociado a la existencia de la plataforma para el despliegue de servicios de *streaming*, el segundo, la no existencia de parámetros de validación reales del modelo, de los servicios y de datos de referencia asociados a experiencias anteriores.

4.3.3.1 Atributos de la temática

Para el caso de estudio se pretende socializar temáticas relacionadas con el área de conocimiento del departamento de Telemática, por lo que las actividades de formación tradicional ya hacen uso de este tipo de metodologías y existe algo de material de referencia que puede ser utilizado para los nuevos servicios y/o esta optimizado para su propósito.

Para mantener la información de las presentaciones o diapositivas se puede realizar algún tipo de conversión que permita mostrar al estudiante el mismo contenido que se trabajaba en la actividad presencial, por ejemplo convirtiendo las diapositivas y textos, a imágenes individuales dentro del servicio que se muestra en el LMS.

Si la información contenida en las palabras del profesor ofrece más conocimiento relevante sobre el tema que las diapositivas, es necesario mantener como mínimo una grabación de la sesión de formación presencial y establecer mecanismos que favorezcan la accesibilidad hacia el recurso.

Si las expresiones corporales y gestuales del profesor, durante la exposición, mejoran la apropiación del conocimiento o ilustran más claramente el objetivo educativo es conveniente mantener una grabación en video de la sesión.

4.3.3.2 Participación de los actores de la actividad formativa

Para analizar de forma general estos aspectos se da respuesta a las preguntas planteadas en los lineamientos metodológicos del Capítulo 2.

¿Qué destrezas se necesitan desarrollar en el uso de la tecnología?

Ninguna/poca suponiendo que los estudiantes ya tienen experiencia en el uso de las herramientas del LMS. La forma en que se implementan y ofrecen los servicios de *streaming* no alteran el flujo tradicional que se ofrece en los sistemas LMS tradicionales, además de aprovecha de tecnologías de acceso común, usadas en sistemas de entretenimiento muy usadas en internet.

¿Qué implicaciones económicas tiene la tecnología para los usuarios del sistema?

Ninguna/poca, dado que los estudiantes acceden al LMS de forma remota, es decir mediante Internet y las herramientas de consumo de los servicios son de carácter libre, por lo que no se incurre en inversiones adicionales.

¿La infraestructura tecnológica de estudiantes, profesores y el LMS en general está lista para la integración de la tecnología?

Sí para el estudiante, dado que con las mismas herramientas de acceso al LMS se puede consumir los servicios basados en *streaming*.

Sí para el LMS, si este soporta el estándar SCORM, con el cual se integran los servicios.

Para el profesor la infraestructura tecnológica depende de la naturaleza del servicio de *streaming* a la hora de crear los contenidos.

¿Se necesita capacitación para el uso de la nueva tecnología? ¿Cuánto tiempo?

Depende de la naturaleza del servicio y de las capacidades del mismo. Además teniendo en cuenta algunas situaciones específicas de quien va a utilizar la tecnología:

- Es alguien que no tienen ningún conocimiento sobre las tecnologías de la información?
- Sabe algo o ha utilizado algún tipo de tecnología relacionada con las TICs?.
- Sabe o ha trabajado con el tema de *e-learning*?

4.3.3.3 Refinamiento y selección de tecnologías

Con la información obtenida hasta este punto podemos establecer los requisitos mínimos sobre la forma en que se estructura el servicio.


Para realizar el refinamiento de las tecnologías es necesario especificar más los servicios, identificando claramente los componentes del servicio y el tipo de material que favorece la actividad formativa.

En nuestro caso de estudio el servicio requiere:

- Grabación como mínimo del audio de la exposición, idealmente se debe ofrecer el video de la sesión. Esto implica la necesidad de utilizar un reproductor de audio/video, que permita entre otras las funcionalidades de inicio/pausa y de avanzar/retroceder sobre la línea de tiempo del video.
- Para mantener la integridad de las diapositivas se requiere una aplicación que permita mostrar imágenes/texto asociados a cada diapositiva de tal forma que exista sincronización entre la reproducción del audio/video y la diapositiva, de la misma forma en que se haría durante la sesión de exposición tradicional.
- Para lograr la interacción entre estudiantes se necesita una aplicación que reúna comentarios/ideas sobre partes importantes de la grabación de la exposición. Los

comentarios/ideas no son sobre el recurso en general, sino sobre partes específicas del contenido.

Con lo anterior podemos realizar la refinación de las tecnologías que se van a utilizar para cada tipo de contenido que se va a presentar en el caso de estudio, teniendo en cuenta algunas de sus características y limitantes para desarrollar cada tipo de contenido:

En las tablas 16,17 y 18, se utiliza la información de la tabla 15 para analizar la tecnología correspondiente y se usa el símbolo  para indicar cuál fue la tecnología seleccionada.


Tecnología	Audio/Video
Video/Audio Vivo	Los estudiantes y el profesor requieren estar disponibles en el mismo momento. Interacción con el material limitada debido a que es un recurso en vivo, no pudiendo adelantarse a puntos específicos del contenido.
Descarga progresiva	Uso constante del ancho de banda Los recursos se descargan, almacenan en el equipo y luego se reproducen. Permite avanzar y retroceder sobre la línea de tiempo del contenido
Video/Audio Bajo Demanda	Solo usa el ancho de banda necesario durante la reproducción. El recurso no se descarga al equipo, se reproduce desde un buffer en memoria.  Permite avanzar y retroceder sobre la línea de tiempo del contenido

Tabla 16. Refinamiento tecnologías de *streaming* para audio/video.

Tecnología	Presentación Diapositivas
Shared Objects	Eficiente para sincronización de datos de tamaño pequeño Enfocado a aplicaciones colaborativas, que requieren que los usuarios se encuentren en línea al mismo tiempo. Permite interacción
Callbacks Tecnología Push	o Útil cuando el servidor requiere enviar mensajes al cliente de forma asíncrona.
Descarga normal	Uso eficiente del ancho de banda Los recursos se descargan, almacenan en el equipo y luego se despliegan al usuario.

Streaming RPC	Uso eficiente del ancho de banda No es necesario almacenar el recurso descargado en el equipo del cliente. ✓
----------------------	---

Tabla 17. Refinamiento tecnologías de *streaming* para servicio de presentación de diapositivas

Tecnología	Comentarios entre estudiantes
Streaming RPC	Permite invocar métodos remotos. Soporte a invocaciones paralelas y concurrentes de los métodos que exponen servicios. ✓
Shared Objects	Eficiente para sincronización de datos de tamaño pequeño Enfocado a aplicaciones colaborativas, que requieren que los usuarios se encuentren en línea al mismo tiempo. Permite interacción

Tabla 18. Refinamiento tecnologías de *streaming* para servicio de comentarios

4.4 Implementación de los servicios de *streaming*

Los servicios de *streaming* están formados por dos componentes principales, por un lado el componente que se despliega dentro del servidor Red5, el cual implementa y ejecuta parte de la lógica asociada al servicio propiamente dicho, y por otro lado, el componente del lado del cliente, ejecutado sobre el navegador web, que permite consumir los servicios y que puede o no tener responsabilidades asociadas a la lógica del servicio. La figura 22 describe en forma general los componentes implementados del lado del servidor Red5 para satisfacer las necesidades del caso de estudio y las tecnologías asociadas a la implementación real. De los componentes mostrados en la figura 19, solo los componentes Hibernate_JDBC, StreamelWeb y ServicioSeguridad no fueron desarrollados en su totalidad:

- Hibernate_JDBC: Se utilizó la implementación proporcionada mediante el proyecto atomikos [100].
- ServicioSeguridad: La implementación desarrollada no define un servicio de este tipo de forma, común, sin embargo, se implementaron los mecanismos de seguridad otras clases y parte de los algoritmos de validación están dentro de la aplicación StreamelService, por lo que el trabajo necesario para su creación se limita a la agrupación de funcionalidad ya implementada.
- StreamelWeb: Esta implementado parcialmente. La funcionalidad desarrollada se enfoca a la gestión de la información del servicio de video *streaming*.

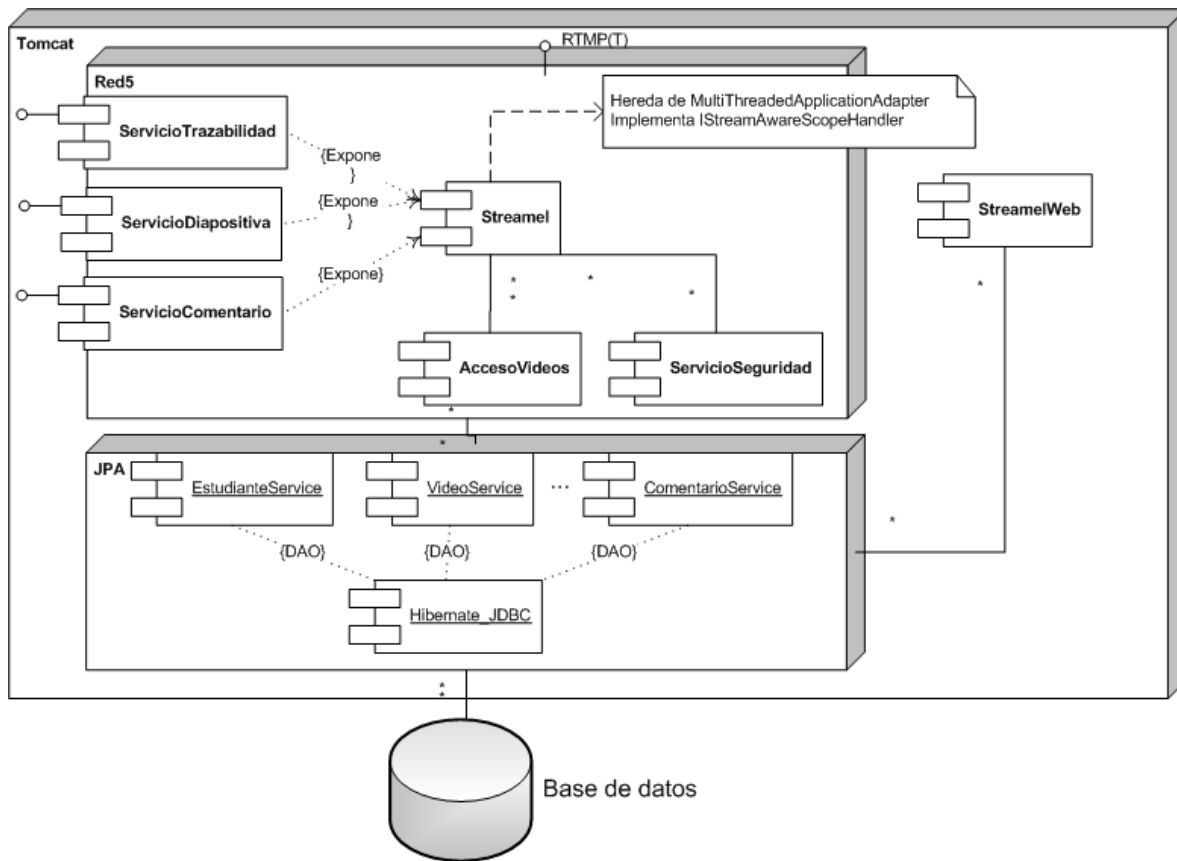


Figura 22. Componentes de la implementación plataforma de *streaming*

El componente **Streamel**, define el punto de entrada para todos los servicios de *streaming* que se puedan ofrecer desde el cliente Flash, dentro de sus funcionalidades están:

- Es la aplicación que establece un contexto dentro de servidor Red5, para administrar y controlar las conexiones desde el cliente.
- Hace uso del API de Red5 para gestionar el canal de comunicaciones mediante el protocolo RTMP(T).
- Inicia las tareas de validación para determinar si el estudiante está autorizado para acceder los servicios solicitados.
- Expone hacia el cliente Flash los métodos ofrecidos por otros servicios, mediante la conexión RTMP(T).
- Controla las transmisiones desde y hacia el cliente de los recursos de audio/video *streaming*, apoyándose de las utilidades de Red5.

El componente **ServicioTrazabilidad** utiliza los servicios ofrecidos por la capa de persistencia para almacenar la información que se envía desde el cliente con propósitos de trazabilidad.

El componente **ServicioDiapositivas** implementa funcionalidades que permiten realizar algunas acciones sobre las diapositivas, entre otras, transformar a SWF y cargar diapositivas para su transmisión al cliente.

El componente **ServicioComentario**, implementa funcionalidades relacionadas con el almacenamiento y listados de comentarios de los videos.

Del lado del cliente se implementa la aplicación que consume los servicios desde la plataforma de *streaming*.

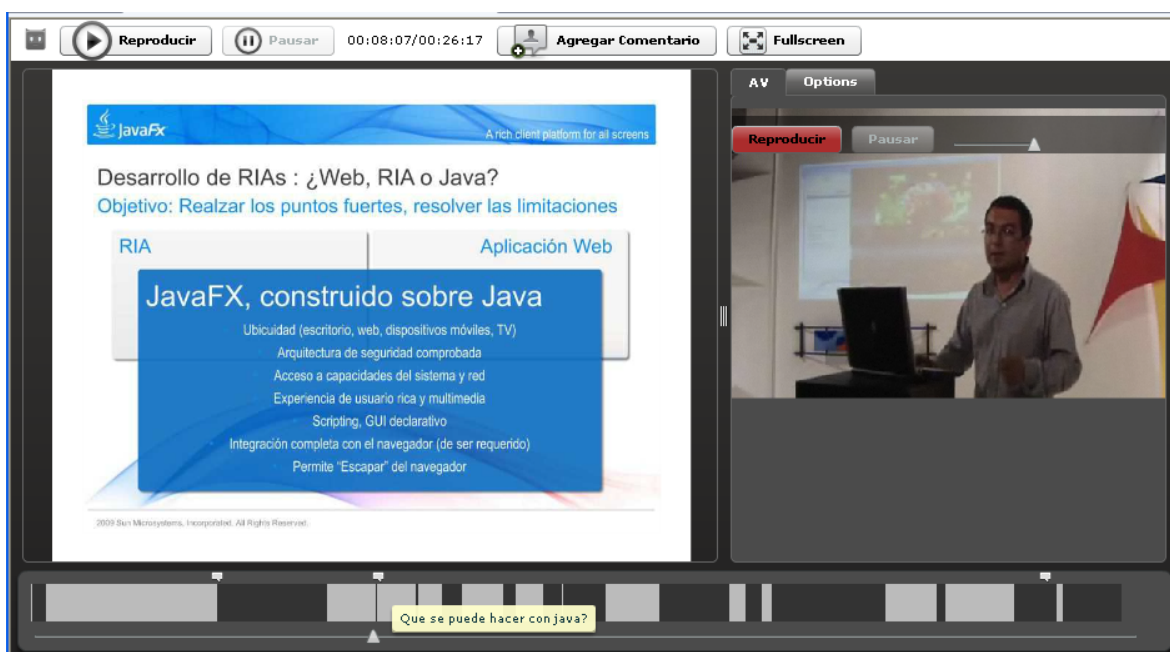


Figura 23. Aplicación del lado del cliente de consumo del servicio de video *streaming*

La implementación de la aplicación se realizó de acuerdo a las recomendaciones establecidas en el capítulo 3, de tal manera, que su estructura consiste en un grupo de componentes, entre los cuales destacan los módulos generales que permiten consumir los servicios.

La aplicación en general hace uso del *framework SpringActionscript* [101] para realizar los procedimientos de inyección de dependencias en tiempo de ejecución, esto facilita la reutilización, modificación de las funcionalidades desarrolladas o la inclusión de nuevas.

Para la fase de inicialización de la aplicación se usa el *framework Cairngorn* [102], ampliamente utilizado para el desarrollo de aplicaciones basadas en tecnología Flash. Esta fase incluye la carga de los descriptores XML de los servicios de *streaming* y MPEG7 asociado a la descripción del recurso multimedia y presentación de diapositivas.

Para la capa de comunicaciones, la cual mantiene comunicación con el servidor Red5, se utilizan algunas de las herramientas incorporadas en el proyecto Jedai [103]. Específicamente se aprovechan clases relacionadas con el objeto Red5Connection para enmascarar la conexión con el servidor, mediante el protocolo RTMP(T) y manteniendo la compatibilidad con la clase nativa flash.net.NetConnection de la tecnología Flash y en consecuencia con cualquier otro tipo de servidor de *streaming*.

La aplicación está dividida, básicamente por 4 módulos que permiten la integración con los servicios de *streaming*. El módulo de reproducción de audio/video (A), el módulo de presentación de diapositivas (B), el módulo de línea de tiempo (C) y el módulo de presentación de comentarios (D).

Cada módulo hace uso del patrón “*Presentation Model*” [104, 105], el cual permite delegar responsabilidades a clases específicas asociadas al control del modelo de datos general y a la presentación en la interfaz de usuario. Este patrón ofrece la posibilidad de crear interfaces gráficas de usuario adicionales enfocadas para otros ambientes como el móvil.

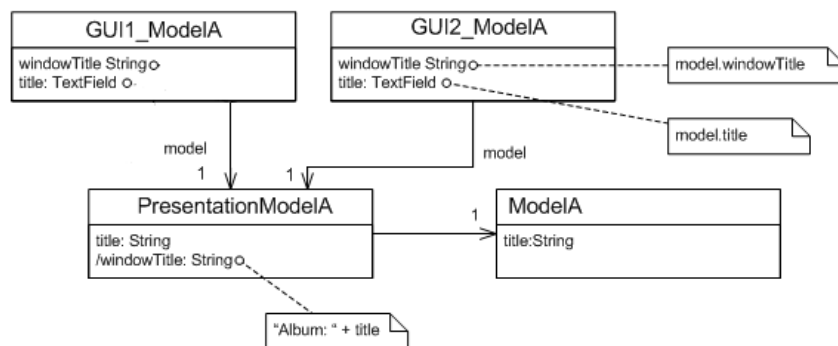


Figura 24. Vista general patrón presentation model.

4.4.1 Módulo reproducción de audio/video

Módulo encargado de la reproducción de audio y video bajo *streaming*, mediante una conexión Red5Connection previamente establecida.

Hace uso de la información del elemento <MediaInformation> y <MediaProfile> definida en MPEG7 para determinar otros recursos multimedia relacionados con el recurso multimedia maestro, esto enfocado a la adaptación de contenido.

Publica y despacha eventos asociados al estado de reproducción del recurso multimedia, conteniendo información relevante como la duración total del recurso y el tiempo actual de reproducción.

Ofrece funcionalidades que permiten a otros componentes o módulos controlar el comportamiento del recurso, es decir, permite iniciar/pausar e ir a puntos específicos dentro de la línea de tiempo del recurso.

4.4.2 Módulo presentación de diapositivas

Este módulo se registra a los eventos del módulo anterior para determinar la diapositiva que debe presentarse en función del tiempo que lleva reproduciéndose el recurso audio-visual. Para lograr este proceso de sincronización se hace uso de la información definida en los elementos <Description>, de tipo “SummaryDescriptionType”, y <VisualSummaryComponent> definidos por el estándar MPEG7. También se usa el elemento <MediaUri> para determinar la forma en que se debe obtener la diapositiva, esto facilita la definición de diferentes mecanismos y protocolos de acceso hacia este tipo de recursos.

Cuando las diapositivas deben ser descargadas por el módulo basada en una URI, *Uniform Resource Identifier*, que usa el protocolo RTMP, se hace uso de la tecnología de RPC *streaming* para la descarga de la información en forma de bytes, esto permite que desde el lado del cliente y del servidor se establezcan mecanismos de encriptación y/o protocolos de seguridad que protejan los derechos de autor sobre este tipo de recursos.

Junto con las funcionalidades del servicio de *streaming* del lado del servidor, permite transformar las diapositivas en elementos SWF, con soporte de objetos vectorizados, es decir que si se aumenta o disminuye el tamaño de la diapositiva, esta mantiene la calidad, incluso si contiene texto. Lo anterior también posibilita la carga animaciones tradicionales o interactivas, sin embargo esto requiere de un mejor análisis dado las implicaciones de seguridad, ya que se abre la puerta para la ejecución de código malicioso.

4.4.3 Módulo de línea de tiempo

Este módulo hace uso de la información contenida dentro del descriptor MPEG7, al igual que los dos módulos anteriores, para crear un interfaz de usuario usable, la cual relaciona la duración del recurso audio-visual, con el número de diapositivas y el momento en que se despliegan, esto permite visualizar de forma general la estructura de la sesión de formación.

Aprovecha las funcionalidades del módulo de reproducción audio-visual para ofrecer la posibilidad de “desplazarse” dentro del contenido del recurso multimedia de forma rápida.

4.4.4 Módulo de comentarios

Apoyado de la ayuda visual del módulo de línea de tiempo, permite desplegar el listado de comentarios asociados al recurso audio-visual, de forma descriptiva, esto es, de acuerdo al componente temporal de los comentarios.

Ofrece funcionalidades especiales que permiten a los estudiantes publicar nuevos comentarios, teniendo en cuenta el estado de reproducción actual.

Para obtener el lista de comentarios el módulo hace llamados basado en la tecnología RPC, invocando métodos en la plataforma de *streaming*, que a su vez realizan las consultas necesarias en la base de datos.

Los módulos descritos anteriormente hacen uso de la capa de comunicaciones para realizar las tareas de trazabilidad. De acuerdo a lo propuesto en los lineamientos del capítulo 2, sobre la relación SCO-Servidor de *streaming* en el nivel de datos, se presenta el modelo usado en el caso de estudio para analizar las interacciones de los estudiantes en el uso de los servicios descritos:

streamel.video.state		
Valor Temporal: Obligatorio	Definición: El(los) estado(s) del recurso audio-visual durante el tiempo que el estudiante accede e interactúa con el SCO (RALO)	
Tipo de dato: Cadena caracteres	El campo temporal, en segundos, representa el instante, basado en la duración del recurso audio-visual y su línea temporal, en el cual se genera el registro.	
Valores: Started Paused Resumed Completed	Formato: Un grupo de caracteres. cuatro posibles valores: <ul style="list-style-type: none"> • Started: Indica que el video inicia su proceso de reproducción normal. El valor temporal diferente de 0, cero, indica que el video inicia su reproducción desde un instante diferente. • Paused: Indica que el video ha sido pausado en algún instante. • Resumed: Cuando el estudiante inicia nuevamente la reproducción del video después de que lo ha pausado. • Completed: Cuando el video ha completado su reproducción total. 	
Ejemplo:		
Elemento	Valor	Campo Temporal
streamel.video.state	Started	0

	<table border="1"> <tr> <td>streamel.video.state</td> <td>Completed</td> <td>350</td> </tr> </table> <p>Describe una reproducción normal sin pausas de un recurso multimedia de 5 minutos y 50 segundos de duración.</p>	streamel.video.state	Completed	350															
streamel.video.state	Completed	350																	
streamel.video.rewind_to y streamel.video.forward_to																			
<p>Valor Temporal: Obligatorio</p> <p>Tipo de dato: numérico</p> <p>Valores: Un valor mayor e igual que 0 y menor o igual que la duración total en segundos del recurso audio visual</p>	<p>Definición: Describe la acción de avanzar o retroceder sobre la línea de tiempo del recurso audio-visual. El componente temporal indica el momento, en segundos, en que se realiza la acción, desde el punto de vista de la reproducción actual del recurso.</p> <p>Formato: Un valor numérico que puede incluir décimas o milésimas de segundo:</p> <p>Ejemplo:</p> <table border="1"> <thead> <tr> <th>Elemento</th> <th>Valor</th> <th>Campo Temporal</th> </tr> </thead> <tbody> <tr> <td>streamel.video.state</td> <td>Started</td> <td>0</td> </tr> <tr> <td>streamel.video.forward_to</td> <td>120.3</td> <td>90</td> </tr> <tr> <td>streamel.video.rewind_to</td> <td>88</td> <td>150</td> </tr> <tr> <td>streamel.video.rewind_to</td> <td>95.5</td> <td>300</td> </tr> <tr> <td>streamel.video.state</td> <td>Completed</td> <td>350</td> </tr> </tbody> </table> <p>Describe una reproducción que inicia normalmente, a los cero segundos, cuando el estudiante ha alcanzado 1 minuto y 30 segundos de reproducción normal, adelanta el video al instante aproximado de los 2 minutos. El video se reproduce por 30 segundos más y ahora el estudiante retrocede al instante ubicado en 1 minuto y 28 segundos. La reproducción continua normalmente hasta el minuto 5 y nuevamente el estudiante retrocede a un instante anterior, ubicado en el minuto 1 con 35 segundos aproximadamente, desde este punto en adelante la reproducción continua normalmente hasta que se completa.</p> <p>El conjunto de registros permite al profesor o analista de datos de trazabilidad identificar periodos de tiempo importantes, que pueden representar dificultades o interés particular sobre aspectos contenidos en el video.</p>	Elemento	Valor	Campo Temporal	streamel.video.state	Started	0	streamel.video.forward_to	120.3	90	streamel.video.rewind_to	88	150	streamel.video.rewind_to	95.5	300	streamel.video.state	Completed	350
Elemento	Valor	Campo Temporal																	
streamel.video.state	Started	0																	
streamel.video.forward_to	120.3	90																	
streamel.video.rewind_to	88	150																	
streamel.video.rewind_to	95.5	300																	
streamel.video.state	Completed	350																	
streamel.video.switchTo																			
<p>Valor Temporal: Opcional</p>	<p>Definición: Describe la acción del estudiante cuando accede a un recurso audio-visual relacionado con el recurso maestro, es decir,</p>																		

<p>Tipo de dato: Cadena caracteres Valores: El identificador del recurso audio-visual relacionado con el recurso maestro.</p>	<p>accede a otro recurso multimedia que puede estar recodificado a otro <i>bitrate</i> para efectos de adaptación de contenido. El valor temporal es opcional, ya que este valor no agrega mayor información.</p> <p>Ejemplo:</p> <table border="1"> <thead> <tr> <th>Elemento</th> <th>Valor</th> <th>Campo Temporal</th> </tr> </thead> <tbody> <tr> <td>streamel.video.state</td> <td>Started</td> <td>0</td> </tr> <tr> <td>streamel.video.switchTo</td> <td>videoXY_audio_only.mp3</td> <td>10</td> </tr> <tr> <td>streamel.video.state</td> <td>Completed</td> <td>350</td> </tr> </tbody> </table> <p>Muestra el inicio de la reproducción normal de un recurso audio visual que puede incluir video a cualquier <i>Bitrate</i>, en el instante ubicado a los 10 segundos, el estudiante prefiere reproducir un recurso que representa la misma sesión de aprendizaje, pero que solo está formada por audio, desde ese momento la reproducción continua hasta completarse.</p>	Elemento	Valor	Campo Temporal	streamel.video.state	Started	0	streamel.video.switchTo	videoXY_audio_only.mp3	10	streamel.video.state	Completed	350
Elemento	Valor	Campo Temporal											
streamel.video.state	Started	0											
streamel.video.switchTo	videoXY_audio_only.mp3	10											
streamel.video.state	Completed	350											

En el anexo 2 se presentan algunas pruebas y resultados obtenidos relacionadas con el rendimiento de la aplicación y la puesta en marcha del caso de estudio anteriormente mencionado, en el anexo 5 se presenta el modelo de encuesta utilizado para evaluar el servicio de video *streaming* implementado y su correspondiente análisis.

Capítulo 5. Conclusiones, recomendaciones y trabajos futuros

5.1 Conclusiones

Se creó un marco de referencia alrededor de la tecnología de *streaming* que permitió definir los conceptos necesarios para proponer los lineamientos de integración de este tipo de servicios a los sistemas de gestión de aprendizaje, sin embargo durante las fases de implementación del caso de estudio y su verificación, la realimentación lograda con esta información permitió complementar, corregir y estabilizar el marco de referencia. Esto nos confirma que un proceso de desarrollo iterativo e incremental, como el basado en el MCS, adoptado por el departamento de Telemática de la universidad del Cauca es efectivo.

Aunque el *streaming* inició y se define como tecnología enfocada para la transmisión de audio y video, con el estudio realizado se logró constatar que existen otros tipos de datos que pueden ser transmitidos mediante el streaming y además otro tipo de aplicaciones para esta tecnología. Este proyecto presentó una definición que mantiene concordancia con su historia pero deja abierta la posibilidad en el desarrollo y definición de nuevas aplicaciones.

La base conceptual resume las diferentes opciones tecnológicas, existentes hasta el momento, relacionadas con la tecnología de streaming. Aunque para el desarrollo de las recomendaciones y el caso de estudio se seleccionó la tecnología Flash, los lineamientos y recomendaciones pueden ser aplicados y adaptados para el uso de otras tecnologías, todo depende del contexto donde se requiera la integración de este tipo de servicios y de las condiciones o requerimientos particulares establecidos.

La adopción de este marco de referencia para la creación de servicios de *streaming* permite mejorar la implementación de los mismos, al tener en cuenta no solamente el aspecto tecnológico, sino también otros aspectos, como el de los diseños de los servicios, las interacciones con el material y su tipo. Además permite tener como punto de referencia conceptos y criterios importantes tanto para el diseñador de los servicios como para la institución u organización que los va a implantar, entre estos, el presupuesto,

impacto de las nuevas herramientas para los participantes y adaptabilidad a los estilos metodológicos de la institución.

Los modelos propuestos en los lineamientos metodológicos fueron formulados bajo una fuerte base bibliográfica que reúnen la experiencia de diferentes autores en esta área. Los modelos pueden ser utilizados para caracterizar, diseñar e implementar nuevos servicios, incluso si no son basados en *streaming*. Dichos modelos no solamente describen los servicios a partir de sus necesidades, sino que permiten a los diseñadores tener en cuenta diferentes opciones tecnológicas o aplicaciones de la tecnología para su implementación, como el video bajo demanda, la video conferencia, el streaming RPC, etc. De esta forma se fomenta el potencial para extender el uso de la tecnología para las actividades de aprendizaje.

El uso de los estándares dentro de la arquitectura propuesta permite que otros desarrolladores aprovechen sus beneficios sin la necesidad de conocer la lógica de fondo, por ejemplo, para el uso de las aplicaciones y servicios desarrollados en el caso de estudio, no es requerido conocer la programación interna, simplemente se requiere entender de forma general algunos elementos dentro del estándar MPEG7, descritos en el capítulo 4, literales 4.4.1 a 4.4.3, para la descripción de los recursos multimedia.

Gracias a la forma en que se define el estándar SCORM se cuenta con una ventana que permite integrar nuevas tecnologías y en consecuencia nuevos servicios para los ambientes virtuales de educación, manteniendo la integridad de la plataforma software.

La forma en que se planteó la integración de los servicios de *streaming* y la forma en que propuso la arquitectura permitieron que el sistema sea reutilizable, es decir, que los servicios puedan consumirse desde diferentes sistemas.

El manejo de *streaming* haciendo uso de los protocolos RTMP y RTMPT permite ofrecer servicios avanzados accesibles, desde el punto de vista de los requerimientos (hardware, al no requerir equipos especializados, y software, el navegador y reproductor Flash como medio de acceso común), teniendo en cuenta las diferentes limitantes que existen en las redes de datos y las medidas de seguridad que se implementan en las instituciones educativas, por ejemplo, navegación mediante proxy y reglas de firewall que limitan la tráfico desde y hacia internet.

La implementación del servicio de *streaming* es posible, pero se deben tener en cuenta las características de las redes y la tecnología a usar a la hora de implementar el servicio, Entre estos factores se encuentran el ancho de banda, la velocidad de transmisión, los estándares soportados, las características de calidad de servicio entre otros. Estos

aspectos son la clave para determinar el conjunto de especificaciones a emplear durante la implementación del mismo.

Algunos servicios de *streaming*, soportados por la tecnología Flash, en especial *streaming* RPC, facilitan la creación de una capa de seguridad adicional, dado que se cuenta con un entorno de ejecución en cada lado del sistema, cliente y servidor, que permite crear aplicaciones con lógica compleja, útil para la protección de la transmisión y de los recursos multimedia como tal.

La plataforma propuesta facilita la trazabilidad de cualquier evento generado desde el servicio de streaming, cuando se ejecuta en el cliente. Es importante para el tutor hacer uso dichas fortalezas para validar la efectividad de sus estrategias educativas basadas en los servicios de *streaming*. Así mismo, es importante mejorar y desarrollar nuevas herramientas que permitan analizar los datos que almacena la plataforma de streaming, inicialmente, para la generación de reportes ó de extenderse el LMS como se describe en el anexo 4, en la propia plataforma de e-learning.,

Gracias a las tecnologías propuestas y a la manera en que se sugiere la estructuración de los sistemas, se logra tener una plataforma que da soporte a nuevos requerimientos y nuevos ambientes, como el ambiente móvil.

Así como el Internet ha evolucionado hacia la integración de servicios para mejorar la experiencia de usuario, los servicios de *streaming* pueden y deberían ser cooperativos, brindando experiencias más satisfactorias y sobre todo más integrales.

El software libre resultado de este trabajo, permitirá a quienes estén interesados en continuarlo, una serie de facilidades y garantías para contribuir con nuevos servicios.

La integración de servicios de *streaming*, sean existentes o nuevos, no garantiza el mejoramiento en la calidad de educación, por consiguiente es de vital importancia contribuir con propuestas y desarrollos sobre el tema, logrando con ello el fortalecimiento de la base conceptual, la validación de los lineamientos y arquitectura propuesta de una manera formal, pero sobre todo determinar la efectividad de los servicios con una ventana de tiempo más extensa.

Es importante preguntarse desde lo tecnológico cómo se pueden facilitar las dinámicas de un curso en lugar de afectarlas. Se debe tener en cuenta que las tecnologías pueden llegar a ser un factor trascendental y hasta una barrera, cuando se tienen procesos de comunicación soportados por nuevas tecnologías.

Desde el punto de vista comercial se abre una nueva puerta para brindar servicios de enseñanza en línea más atractivos bajo un modelo no presencial. En este caso es importante que se tengan en cuenta los servicios de *outsourcing* que existen en el mercado para el despliegue de la plataforma de *streaming*, ya que puede reducir considerablemente los costos de funcionamiento y maximizar las ganancias, a diferencia de si se despliega la plataforma en la infraestructura interna de la institución.

5.2 Recomendaciones y trabajos futuros

Para mejorar el impacto de los servicios, es recomendable diseñarlos acompañados de especialistas en pedagogía, diseñadores gráficos y comunicadores, quienes pueden ayudar en la determinación de los puntos de evaluación y de la estrategia de funcionamiento del mismo

Aunque SCORM ofrece herramientas importantes para la integración de nuevos servicios bajo la definición de un SCO, la forma en que se especifica e implementan los mecanismos de comunicación con el LMS, facilitan a cualquier estudiante con conocimientos en Javascript y la propia especificación de SCORM, engañar a la plataforma LMS explotando fallas de seguridad conceptuales. Por esta razón, se recomienda que para la implementación de servicios de evaluación, como exámenes, se tomen medidas de seguridad adicionales y no se limite la confianza a los datos reportados por la implementación del estándar SCORM.

Desarrollar e implementar mecanismos de seguridad avanzados para los procesos de acceso y comunicación entre el LMS, el cliente de consumo de servicios de *streaming* (RALO) y la plataforma de *streaming*.

Uno de los servicios más importantes que se deberían desarrollar e integrar con los servicios de *streaming* mencionados en los capítulos 2 y 4, debe tener en cuenta la personalización de la experiencia del usuario, dentro de esto se incluyen herramientas que permitan mantener registro de los contenidos o partes de los contenidos que son favoritos para los estudiantes, estas herramientas pueden implementarse teniendo en cuenta los principios establecidos en los servicios de *bookmarks* o los servicios de etiquetas (*tags*), comunes dentro de la definición de la Web 2.0 y los redes sociales.

Los servicios de *streaming* deberían acompañarse de aplicaciones de administración enfocadas hacia los profesores con el objetivo de facilitar las tareas de creación de contenidos, configuración de los servicios y especialmente reduciendo o automatizando el trabajo de bajo nivel requerido por los servicios, por ejemplo creando procesos que adapten o re codifiquen automáticamente el contenido.

Desarrollar una estrategia pedagógica para realizar trazabilidad y para interpretar la información existente, posteriormente actualizar los lineamientos y servicios propuestos para este fin.

Para mejorar la comunicación entre el SCO (RALO) y el LMS se recomienda modificar la implementación del estándar SCORM dentro del LMS, manteniendo su compatibilidad, pero agregando funcionalidad para el procesamiento de nuevos mensajes. Esto garantiza en cierta medida la integridad de la plataforma y ofrece nuevas posibilidades de servicios. Lo anterior también implica la definición de un nuevo modelo de datos, que en lo posible mantenga la estructura del modelo CMI.

Bibliografía

- [1] Agenda de Conectividad, "Colombia 2019: Avanzar hacia una sociedad informada," 2006.
- [2] Asamblea Nacional por la Educación, "Plan Decenal de Educación 2006 -2015," 2008.
- [3] O. Calderon, C. Collazos, V. Quintero, and A. Toledo, "Tendencias de la Educación en la Sociedad de las Tecnologías de la Información," *Colombia Aprende*, 2005.
- [4] J. Salinas, "La gestión de los entornos virtuales de formación," in *La calidad de la formación en red en el espacio europeo de educación superior.*, Tarragona, 2005.
- [5] J. Cabero, "Bases pedagógicas del e-learning," *Revista de Universidad y Sociedad del Conocimiento*, vol. 3, 2006.
- [6] F. Prados, I. Boada, J. Poch, J. Soler, and J. Soler, "El e-Learning como complemento a las clases presenciales un caso práctico: el proyecto ACME," in *Congreso Virtual Educa*, Girona, España, 2004.
- [7] Unidad de Formación a Distancia y Recursos Didácticos, "La videoconferencia como recurso didáctico en la enseñanza superior," in *Instituto de Ciencias de la Educación de la Universidad de Murcia*, 2004.
- [8] RENATA, "Educación virtual basada en televisión interactiva para apoyar procesos educativos a distancia.," Último Acceso: Enero 2009.
[En línea] Disponible en:
<http://www.renata.edu.co/index.php/component/content/article/18-proyectos/220-proyectos-en-desarrollo-convocatorias-ministerio-de-educacion-colciencias-renata-2006.html#tv>
- [9] J. Daintith and E. Wright, "Streaming," in *Dictionary of Computing*, 2004 ed Oxford: Oxford University Press, 2004.
- [10] D. Shai, G. Nitzan, and S. Alexander, "Three-dimensional image streaming system and method for medical images," G06K 9/36 ed, 2008.
- [11] P. Kuacharoen, "Embedded software streaming via block streaming," ACM, 2004.
- [12] W. Cheng, W. T. Ooi, S. Mondet, R. Grigoras, and G. Morin, "GAMES@LARGE Graphics Streaming Architecture," in *International Symposium on Consumer Electronics 2008*, Vilamoura, Portugal, 2008.
- [13] C. Wei, O. Wei Tsang, M. Sebastien, G. Romulus, and M. Geraldine, *An analytical model for progressive mesh streaming*. Augsburg, Germany: ACM, 2007.
- [14] R. Parfenovics and M. Fletcher, "Streaming media for higher education: Signs of settling," in *Beyond the comfort zone: Proceedings of the 21st ASCILITE Conference*, 2004.
- [15] Universidad del Cauca, "EVA - Entorno Virtual de Aprendizaje," Último Acceso: Octubre 2008.
[En línea] Disponible en:
<http://eva.unicauca.edu.co>
- [16] J. DAINTITH, "codec," in *A Dictionary of Computing*, 2004 ed, 2004.
- [17] C. Dalsgaard, "Social software: E-learning beyond learning management systems," *European Journal of Open, Distance and E-Learning*, 2006.

- [18] N. Wilson and J. Uhomoibhi, "Digital media technology management and online learning in higher education," in *European Union Nature Information System 2006*, Tartu, Estonia, 2006.
- [19] M. Zapata, "Sistemas de gestión del aprendizaje - Plataformas de teleformación," in *RED: Revista de Educación a Distancia*. vol. 9: Universidad de Murcia: Departamento de Psicología Evolutiva y de la Educación, 2003.
- [20] M. Zapata, "Evaluación de un Sistema de Gestión del Aprendizaje," in *RED: Revista de Educación a Distancia*. vol. 9: Universidad de Murcia: Departamento de Psicología Evolutiva y de la Educación, 2003.
- [21] P. G. Pozo, "Implantación de dotLRN en la universidad de Valencia. Proyecto: Aula Virtual," in *Ingeniería Informatica* Valencia: Universidad de Valencia, 2004.
- [22] Advance Distributed Learning, "ADL Research Work With Us," Último Acceso: Junio 2009.
[En línea] Disponible en:
<http://www.adlnet.gov/WorkWithUs/default.aspx>
- [23] Academic ADL Co-Lab, "Academic ADL Co-Lab Partners," Último Acceso: Junio 2009.
[En línea] Disponible en:
<http://www.academiccolab.org/partners>
- [24] .LRN Consortium, ".LRN: Learn, Research, Network," Último Acceso: Octubre 2009.
[En línea] Disponible en:
<http://www.dotlrn.com/>
- [25] OpenACS Community, "OpenACS: The Toolkit for Online Communities," Último Acceso: Octubre 2009.
[En línea] Disponible en:
<http://openacs.org/>
- [26] R. P. Vargas, A. Rodríguez, V. Aguilar, and R. Hernández, "Blocks Organizer For .LRN," in *7TH OPENACS/.LRN Conference*, Valencia-España, 2008.
- [27] P. Moreno, A. Oliver, M. Arevalillo, and V. Cerverón, "Análisis del uso universitario de plataformas de gestión del aprendizaje y su relación con la innovación educativa y la calidad docente en el marco del EEES. Resultados en la Universitat de València y procedimientos para la generalización," in *Conferencia sobre software libre en la educación superior*, 2008.
- [28] Universidad EAFIT, "Gestión de la tecnología E-learning," in *REVISTA Universidad EAFIT*. vol. 41 Medellín, Colombia, 2005.
- [29] Moodle, "Acerca de Moodle," Último Acceso: Octubre 2009.
[En línea] Disponible en:
http://docs.moodle.org/es/Acerca_de_Moodle
- [30] A. de la Torre, "Introducción a la plataforma Moodle," Último Acceso: Octubre 2009.
[En línea] Disponible en:
http://www.adelat.org/media/docum/moodle/docum/23_cap01.pdf
- [31] Advance Distributed Learning, "SCORM 1.2 Specification," 2009.
- [32] O. Mekpiroona, P. Tammarattananonta, N. Buasrounga, N. Apitiwongmanita, B. Pravalpruka, and T. Supnithia, "SCORM in Open Source LMS : A case study of LEARNSQUARE," Thailand: National Electronics and Computer Technology Center, 2008.
- [33] O. C. Santos, C. Barrera, E. G. y. Restrepo, J. Granado, L. Martin, E. Raffenne, A. Rodriguez, and J. G. Boticario, "Delivery of accessible and SCORM-based courses

- in dotLRN," in *International Conference and Workshops on Community Based Environments: OpenACS and .LRN Spring Conference*, Vienna, Austria, 2007.
- [34] Advance Distributed Learning, "SCORM 2004 Specification. Rev. 4," 2009.
- [35] A. Berggren, D. Burgos, J. M. Fontana, D. Hinkelman, A. H. Vu Hung, and G. Tielemans, "Practical and Pedagogical Issues for Teacher Adoption of IMS Learning Design Standards in Moodle LMS," *Journal of Interactive Media in Education*, 2005.
- [36] Z. Zhang, Q. Sun, and D. He, "A Standardized JPEG2000 Image Authentication Solution based on Digital Signature and Watermarking," 2006.
- [37] I. Nave, H. David, A. Shani, A. Laikari, P. Eisert, and P. Fechteler, "GAMES@LARGE Graphics Streaming Architecture," in *International Symposium on Consumer Electronics 2008*, Vilamoura, Portugal, 2008.
- [38] G. C. Demetriades, "Streaming Media Building and Implementing a Complete Streaming System," Wiley Publishing, Inc., 2003.
- [39] Microsoft Corporation, "About unicast streaming," Último Acceso: Marzo 2009.
[En línea] Disponible en:
[http://technet.microsoft.com/en-us/library/cc730839\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc730839(WS.10).aspx)
- [40] Microsoft Corporation, "About multicast streaming," Último Acceso: Marzo 2009.
[En línea] Disponible en:
[http://technet.microsoft.com/en-us/library/cc753242\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc753242(WS.10).aspx)
- [41] G. E. Chanchi and G. A. Gallego, "Marco de referencia para el despliegue del servicio de IPTV móvil en Colombia," in *Departamento de Telemática Popayán: Universidad del Cauca*, 2008.
- [42] F. Li, "Video-on-demand: scalability and QoS control," in *Distributed multimedia databases: techniques & applications*: IGI Publishing, 2002, pp. 111-119.
- [43] P. Bellavista, A. Corradi, and C. Stefanelli, "Application-Level QoS Control for Video-on-Demand." vol. 7: IEEE Internet Computing, 2003, pp. 16-24.
- [44] Apple Inc., "QuickTime Player," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://www.apple.com/es/quicktime/player/>
- [45] Apple Inc., "QuickTime Streaming Server," Último Acceso: Abril 2010.
[En línea] Disponible en:
<http://www.apple.com/es/quicktime/streamingserver/>
- [46] Apple Inc, "Darwin Streaming Server," Último Acceso: Enero 2010.
[En línea] Disponible en:
<http://developer.apple.com/opensource/server/streaming/index.html>
- [47] Apple Inc., "APPLE PUBLIC SOURCE LICENSE," 2003.
- [48] RealNetworks Inc., "Real Player," Último Acceso: Enero 2010.
[En línea] Disponible en:
<http://www.real.com/realplayer/features>
- [49] RealNetworks Inc., "Helix DNA Client " Último Acceso: Octubre 2009.
[En línea] Disponible en:
<https://helix-client.helixcommunity.org/>
- [50] RealNetworks Inc., "The Helix DNA™ Server " Último Acceso: Octubre 2009.
[En línea] Disponible en:
<https://helix-server.helixcommunity.org/>
- [51] RealNetworks Inc., "RealNetworks Public Source License Version 1.0 " Último Acceso: Octubre 2009.
[En línea] Disponible en:
<https://community.helixcommunity.org/Content/rpsl>

- [52] RealNetworks Inc., "Codecs & Formats," Último Acceso: Octubre 2009.
[En línea] Disponible en:
<https://datatype.helixcommunity.org/>
- [53] Microsoft Corporation, "¿Qué es Silverlight?," Último Acceso: Octubre 2009.
[En línea] Disponible en:
<http://www.microsoft.com/silverlight/overview/default.aspx>
- [54] Microsoft Corporation, "Windows Media Services," Último Acceso: Octubre 2009.
[En línea] Disponible en:
<http://www.microsoft.com/windows/windowsmedia/forpros/server/version.aspx>
- [55] Adobe Inc., "Otrase versiones para plataformas:," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://get.adobe.com/es/flashplayer/otherversions/>
- [56] Adobe Inc., "Flex Overview," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://www.adobe.com/products/flex/overview/>
- [57] Adobe Inc., "Flex SDK," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://opensource.adobe.com/wiki/display/flexsdk/Flex+SDK>
- [58] Adobe Inc., "Adobe Flash Plataform," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://www.adobe.com/flashplatform/>
- [59] adobe Inc., "Flash Media Server," Último Acceso: Agosto 2010.
[En línea] Disponible en:
<https://www.adobe.com/es/products/flashmediaserver/>
- [60] Wowza Media Systems, "Wowza Media Server 2 Specifications," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://www.wowzamedia.com/specs.html>
- [61] Red 5 Project, "Red5, Open source flash server," Último Acceso: Junio 2009.
[En línea] Disponible en:
<http://red5.org/>
- [62] Adobe Systems Inc, "AMF 3 Specification," Rev. 2008.
- [63] Millward Brown and Adobe Inc., "Adobe Flash Player Version Penetration," Último Acceso: Septiembre 2009.
[En línea] Disponible en:
http://www.adobe.com/products/player_census/flashplayer/version_penetration.html
- [64] On2 Technologies Inc., "On2 VP6 for Flash 8 Video," 2005.
- [65] N. H. Page, "NellyMoser," Último Acceso: Octubre 2009.
[En línea] Disponible en:
<http://www.nellymoser.com/>
- [66] C. M. Reigeluth, *Instructional-design Theories and Models: A new paradigm of instructional theory*, illustrated ed. Londres: Routledge, 1999.
- [67] A. M. Armstrong, *Instructional design in the real world: a view from the trenches*: Idea Group Inc, 2004.
- [68] A. W. Bates and T. Bates, *Technology, e-learning and distance education*. Londres: Routledge, 2005.
- [69] J. Taylor, "Fifth generation distance education," Department of Education, Training and Youth Affairs, Australia Report No. 40, 2001.

- [70] H. Santos, I. Mendes, C. P. Leão, F. O. Soares, and E. Silva, "Streaming Media in Different Contexts: Engineering and Education Graduation," in *IADIS International Conference e-Learning*, 2007.
- [71] P. Fenrich, *Creating instructional multimedia solutions: practical guidelines for the real world*. Informing Science, 2005.
- [72] G. Areitio and A. Areitio, "Nuevas formas de trabajo para el docente frente a los nuevos modelos de enseñanza universitaria," in *REVISTA ELECTRÓNICA DE GEOGRAFÍA Y CIENCIAS SOCIALES* vol. VI Universidad de Barcelona: Scripta Nova, 2002.
- [73] D. P. O. Pineda, "¿Qué es un ambiente virtual de aprendizaje?," Último Acceso: Octubre 2009.
[En línea] Disponible en:
<http://aprendeonline.udea.edu.co/lms/moodle/mod/resource/view.php?id=48359>
- [74] C. J. Bonk, "R2D2: A model for using technology in education," eSchool Media Inc., 2009.
- [75] C. B. Acuña and V. Montaña, "Modelo de incorporación de TIC en el proceso de innovación docente para la implementación de un B-Learning," in *Revista Didáctica, Innovación y Multimedia* Barcelona: Departamento de Pedagogía Aplicada, Facultad de Educación - Universidad Autónoma de Barcelona, 2004.
- [76] M. Debevč, M. Z. Venuti, and Ž. Peljhan, *E-learning material planning and preparation*. Maribor, Slovenia: University of Maribor - Faculty of Electrical Engineering and Computer Science 2000.
- [77] Comisión de Regulación de Comunicaciones, "Informe trimestral de conectividad - 2° Periodo de 2009," 2009.
- [78] Comisión de Regulación de Comunicaciones, "Informe trimestral de conectividad - 3° Periodo de 2009," 2009.
- [79] I. IMS Global Learning Consortium, "IMS-LD," Último Acceso: Agosto 2009.
[En línea] Disponible en:
<http://www.imsglobal.org/learningdesign/>
- [80] I. IMS Global Learning Consortium, "IMS-CP," Último Acceso: Agosto 2009.
[En línea] Disponible en:
<http://www.imsglobal.org/content/packaging>
- [81] I. IMS Global Learning Consortium, "IMS-MD," Último Acceso: Agosto 2009.
[En línea] Disponible en:
<http://www.imsglobal.org/metadata>
- [82] I. IMS Global Learning Consortium, "IMS-QTI," Último Acceso: Agosto 2009.
[En línea] Disponible en:
<http://www.imsglobal.org/question>
- [83] M. Rustici, "SCORM Security – Some Perspective," Último Acceso: Noviembre 2009.
[En línea] Disponible en:
<http://www.scorm.com/blog/2009/04/scorm-security-some-perspective/>
- [84] A. Advance Distributed Learning, "Certified Products," Último Acceso: Noviembre 2009.
[En línea] Disponible en:
<http://www.adlnet.gov/Technologies/scorm/Custom%20Pages/Certified%20Products.aspx>
- [85] IBM, "Service Oriented Architecture - SOA," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://www-01.ibm.com/software/solutions/soa/>

- [86] Object Management Group Inc., "Model Driven Architecture - MDA," Último Acceso: Noviembre 2009.
[En línea] Disponible en:
<http://www.omg.org/mda/>
- [87] Open Grid Forum, "Open Grid Services Architecture - OGSA," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://www.globus.org/ogsa/>
- [88] D. Gannon, K. Chiu, M. Govindaraju, and A. Slominski, "An Analysis of The Open Grid Services Architecture," in *Indiana University* Bloomington, IN: Department of Computer Science, 2004.
- [89] J. Joseph and C. Fellenstein, "Grid Computing," IBM Press., 2004.
- [90] CYTED-Grid, "Tecnología GRID como motor del desarrollo regional," 2010.
- [91] M. Amoretti, R. Bertolazzi, M. Reggiani, F. Zanichelli, and G. Conte, "Designing Grid services for multimedia streaming in an e-learning environment: Research Articles," *Concurr. Comput. : Pract. Exper.*, vol. 18, pp. 911-923, 2006.
- [92] R. Pressman, *Ingeniería del Software: Un enfoque practico*, 6 ed.: McGraw Hill Higher Education, 2004.
- [93] R. Drira, M. Laroussi, A. Derycke, and H. B. Ghezala, "A multi-layer, metadata driven approach enhancing SCORM to support adaptive and mobile learning objects," in *3rd WSEAS/IASME International Conference on ENGINEERING EDUCATION*, Vouliagmeni, Greece, 2006.
- [94] A. Jiménez, J. R. Rodríguez, and Z. M. Alfaro, "Extensión al secuenciamiento de SCORM para incluir a las simulaciones en los sistemas de E-learning," in *Current Developments in Technology-Assisted Education: Facultad de Informática, Universidad Politécnica de Madrid*, 2006.
- [95] J. V. M. Tur, A. P. Puig, and J. M. Roig, "Middleware para la extensión de las funcionalidades de contenidos y plataformas de e-learning," *Centro de Formación Tecnológica de Renfe en Valencia - Universidad Cardenal Herrera - Univ. Politécnica de Valencia*, 2005.
- [96] A. d. A. Jiménez, J. R. Rodríguez, and Z. M. Alfaro, "Architecture for Including Simulations in E-learning Systems: an Extension to SCORM," in *Facultad de Informática: Universidad Politécnica de Madrid* 2007.
- [97] Adobe Inc., "Adobe Delivers Flash Player 9 with H.264 Video Support," Último Acceso: Marzo de 2009.
[En línea] Disponible en:
<http://www.adobe.com/aboutadobe/pressroom/pressreleases/200712/120407adobemovies.tar.html>
- [98] P. Hutchison, "SCORM API wrapper for ActionScript 3.0," Último Acceso: Marzo 2009.
[En línea] Disponible en:
<http://pipwerks.com/2008/01/27/new-scorm-api-wrapper-for-actionscript-30/>
- [99] Tata Communications, "Content Delivery Network (CDN) ServiceFuture proof advanced content delivery ", 2009.
- [100] Atomikos BVBA, "Hibernate Integration," Último Acceso: Mayo 2010.
[En línea] Disponible en:
<http://www.atomikos.com/Documentation/HibernateIntegration>
- [101] Spring Source, "Spring Actionscript," Último Acceso: Junio 2009.
[En línea] Disponible en:
<http://www.springactionscript.org/>

- [102] Adobe Inc., "Cairngorm," Último Acceso: Julio 2009.
[En línea] Disponible en:
<http://opensource.adobe.com/wiki/display/cairngorm/Cairngorm>
- [103] Infrared5, "Jedai," Último Acceso: Agosto 2009.
[En línea] Disponible en:
<http://code.google.com/p/jedai/>
- [104] M. Fowler, "Presentation Model," Último Acceso: Junio 2009.
[En línea] Disponible en:
<http://www.martinfowler.com/eaDev/PresentationModel.html>
- [105] T. Sugden, "Applying the Presentation Model in Flex," Último Acceso: [En línea]
Disponible en:
http://blogs.adobe.com/tomsugden/2009/08/applying_the_presentation_mode.html