

Mensajería Instantánea Educativa en un Entorno 3D

Anna Puig-Centelles¹, Pablo Prades¹, Oscar Ripolles², Miguel Chover¹

apuig@uji.es, pgrades@uji.es, oripolles@ai2.upv.es, chover@uji.es

¹ Universitat Jaume I, 12071, Castellón, España

² Universidad Politécnica de Valencia, 46022, Valencia, España

Resumen: Existe un creciente interés en la investigación de métodos docentes que utilicen las nuevas tecnologías. Las comunidades educativas están haciendo un esfuerzo por promover la generación y el uso de entornos virtuales para tutorías, seminarios, etc. El objetivo de este artículo es mejorar la comunicación entre profesores y estudiantes ofreciendo nuevas herramientas para la educación y la formación. Nuestra investigación se centra en cómo podríamos disponer de un interfaz gráfico mejorado para la mensajería, ofreciendo contenido 3D en entornos virtuales para mejorar la interacción educativa. Así, nuestra propuesta es mejorar los medios tecnológicos combinando la tutorización virtual con herramientas y métodos avanzados de gráficos por ordenador y procesamiento de imágenes.

Palabras-clave: educación; entorno 3D; mensajería; mundos virtuales; e-learning.

Abstract: There is a growing interest in researching on teaching methods using the latest technologies. Educational communities are making an effort to promote the generation and use of virtual environments for tutorial sessions, seminars, etc. The aim of this paper is to improve communication between teachers and students by providing new tools for education and training. Our research focuses on how we could have an improved graphical interface for messaging, featuring 3D content into virtual environments to improve the educational interaction. Thus, our proposal is to improve the technological means combining virtual tutoring tools with advanced methods of computer graphics and image processing.

Keywords: education, 3D environment, messaging, virtual worlds, e-learning.

1. Introducción

Este nuevo siglo está transformando la virtualidad en una realidad. La vida en mundos virtuales y chats 3D se está haciendo cada vez más importante para los usuarios de Internet. Los mundos virtuales multiusuario accesibles en la red (también conocidos como metaversos) están suponiendo un cambio sociológico sin precedentes en la

historia de la humanidad. Las fronteras, los horarios, la moneda y otras limitaciones humanas desaparecen cuando nuestro representante informatizado virtual (o *avatar*) se mueve por nuestro mundo alternativo. Estos mundos ofrecen entornos que son bastantes similares al mundo real, ya que incluyen gravedad, topografía real, medios de transporte, interacciones, etc. El tiempo y los usuarios decidirán si la vida en el ciber mundo es virtualmente una realidad (Gehorsam, 2003).

Es posible encontrar entornos virtuales que tienen múltiples aplicaciones, como representaciones 3D de reuniones en tiempo real (Nijholt, 2005) o la terapia de exposición mediante realidad virtual (VRET, Virtual Reality (graded) Exposure Therapy) con la que se pretende tratar desórdenes psicológicos (Powers & Emmelkamp, 2008). De esta forma, están surgiendo nuevas posibilidades en las representaciones 3D, aunque el potencial educativo de estos nuevos entornos no ha sido considerado en toda su extensión.

Entre los objetivos del Espacio Europeo de Educación Superior (EEES), la movilidad y la cooperación se presentan como elementos muy importantes. Las nuevas tecnologías ofrecen a la comunidad estudiantil nuevos medios de comunicación con sus tutores, quienes pueden no estar físicamente disponibles en el centro educativo. Es más, los estudiantes llegan a sentir que la interacción en la relación virtual les hacía estar más relajados que durante las tutorías realizadas en los despachos. La mayoría de las universidades han incorporado el concepto de “universidad virtual”, ofreciendo una infraestructura electrónica en la que profesores y estudiantes pueden comunicarse, compartir recursos, tener sesiones de tutoría o evaluar exámenes. Además, estos sistemas permiten a la totalidad de la comunidad educativa participar en cursos en línea, evitando las restricciones de tiempo y lugar de los programas universitarios tradicionales. Actualmente muchos sistemas de “universidad virtual” ofrecen métodos para la comunicación mediante la mensajería instantánea (MI), lo cual permite a los profesores y estudiantes interactuar y crear grupos de aprendizaje virtual en los que es posible que las conversaciones se produzcan en tiempo real.

Nuestra investigación se encamina hacia el desarrollo de una nueva herramienta educativa para proporcionar comunicación en red de forma eficaz. Nos hemos centrado en cómo podríamos disponer de un interfaz gráfico mejorado para la mensajería, ofreciendo contenido 3D en entornos virtuales para enriquecer la interacción educativa. La flexibilidad y facilidad de uso de las aplicaciones de MI es uno de los puntos clave. En este momento, el intercambio de mensajes de texto de los clientes se muestra en la ventana pero no hay representación gráfica en ella.

En este artículo la Sección 2 presenta los mundos virtuales, revisa los principales protocolos para mensajería instantánea y también algunos conceptos sobre educación a través de Internet. La Sección 3 presenta las ideas que hay detrás de la integración de una representación 3D dentro de una aplicación de MI, comentando además el protocolo de comunicación seleccionado y el motor de visualización que utilizaremos. La Sección 4 muestra cómo hemos realizado la integración de un motor de juegos dentro de una aplicación de MI y cómo hemos desarrollado un entorno 3D donde la comunicación entre usuarios tiene lugar en tiempo real. Por último, la Sección 5 ofrece algunos resultados y la Sección 6 esboza investigaciones futuras.

2. Trabajo Relacionado

El principal objetivo de nuestra investigación es combinar el atractivo de los mundos virtuales con el sencillo uso de una aplicación de MI. Es posible encontrar aplicaciones que ofrecen chats en 3D, como *IMVU*, *ICYou*, *There* o *Moove*. Alguna de ellas también permite conectarse a las redes de *MSN Messenger*, *Facebook* o *Yahoo!*, aunque se trata de aplicaciones de pago que no es posible utilizar para fines educativos.

2.1. Mundos Virtuales

Un mundo virtual es un entorno interactivo simulado al que acceden múltiples usuarios mediante un interfaz en línea, permitiéndoles habitarlo e interactuar en él mediante un avatar (Chover & Vivó, 2004). Los principales rasgos que caracterizan un mundo virtual son que permite a muchos usuarios participar en sus tareas a la vez y en tiempo real. El entorno puede representarse tanto en 2D como en 3D, y permanece incluso si los usuarios individuales no están conectados. Además, el mundo virtual permite a los usuarios desarrollar su propio contenido personalizado y formar grupos sociales o comunidades.

En (Virtual Worlds Page) podemos encontrar un estudio sobre mundos virtuales que ofrece una guía básica para mundos virtuales persistentes en Internet tanto en 2D como en 3D, haciendo especial énfasis en los aspectos de interacción social. Los mundos virtuales han sido creados para muchos fines diferentes, como juegos comerciales, vida social, creación de comunidades en Internet, educación o incluso entrenamiento militar. Podemos hacer una clasificación de los mundos virtuales multiusuario en dos grandes grupos:

- Juegos de rol multijugador masivo online (MMORPG). Tienen reglas que seguir, metas que alcanzar y objetivos que cumplir. Un gran número de jugadores interactúan entre sí, de forma que cada jugador asume el rol de un personaje ficticio y controla muchas de las acciones de ese personaje. Cabe destacar el *World of Warcraft* de *Blizzard Entertainment*, que es el MMORPG más grande del mundo con 11,5 millones de jugadores mensuales.
- Mundos persistentes multijugador masivos (MMPW). Se convierten en un espacio virtual 3D plenamente inmersivo. Son entornos en los que los usuarios forman una sociedad paralela para comunicarse e interactuar entre sí en un ciberespacio, utilizando la metáfora del mundo real pero sin sus limitaciones físicas. Entre los muchos metaversos existentes, los más conocidos son:
 - 1995: *Active worlds* de *Renderware*. Ofrece texto y sistemas de voz sobre IP, conocidos con las siglas VoIP. Existen más de 700 mundos y múltiples vehículos con los que recorrerlos.
 - 2000: *Habbo Hotel* de *Sulake Corporation*. Ofrece salas de chat a modo de habitaciones de hotel, visualizándose en proyección isométrica. En 2008 superaron las 100 millones de cuentas existentes.
 - 2003: *There* de *Makena Technologies*. Permiten comunicarse con texto y VoIP. También ofrecen mecanismos para realizar compras o jugar en red.

La comunidad virtual *There* tuvo que cerrar en 2010 debido a la caída de sus beneficios.

- 2003: *Second Life* de *Linden Lab*. Es una sociedad paralela con más de 13 millones de habitantes y una gran cantidad de dinero circulando. Alcanzo fama internacional a partir del año 2006.
- 2004: *IMVU* de *IMVU Incorporation*. Se trata de un chat 3D con 20 millones de usuarios comunicándose por MI.
- 2007: *Home* de *Sony*. Con posibilidad de crear tus propias casas donde compartir música, fotos, videos y juegos.
- 2008: *Lively* de *Google*. Se lanzó como un experimento que permitía a los usuarios crear sus propios escenarios, compartirlos con otros miembros y comunicarse en cualquiera de ellos. El experimento se cerró oficialmente el 1 de enero de 2009.
- 2008: *Twinity* de *Metaversum GmbH*. Propone la creación de metaversos basados en ciudades reales. Actualmente es posible interactuar en Berlín, Singapur y Londres.
- 2009: *OpenSimulator*. Ofrece una alternativa de código abierto para la generación de mundos virtuales compatibles con los clientes de *Second Life*.

2.2. Protocolos de Mensajería Instantánea

Los primeros desarrollos que se ofrecieron para realizar conversaciones mediante Internet fueron los basados en texto, donde el usuario se conectaba a una sala donde podía comunicarse con los distintos usuarios que estaban conectados en ese preciso momento. Las comunicaciones basadas en texto más recientes son las aplicaciones de mensajería instantánea, que no son realmente chats, puesto que se caracterizan por ser conversaciones uno a uno con las personas incluidas en la lista de contactos de un usuario. Estos sistemas también ofrecen la capacidad de comunicarse con múltiples personas simultáneamente.

La mensajería instantánea requiere el uso de un programa que actúe como cliente para que pueda conectarse. La mayoría de los servicios ofrecen información sobre presencia, indicando si las personas de la lista de contactos están en ese momento conectadas y disponibles para comunicarse. Entre los servicios más populares de MI se encuentran *MSN Messenger*, *AOL Instant Messenger*, *Google Talk*, *Jabber*, *Skype* y *Yahoo! Messenger*. Se han realizado varios intentos de crear un estándar unificado, pero la mayor parte de ellos han fracasado y cada aplicación sigue utilizando su propio protocolo de intercambio de mensajes.

El conocidísimo *MSN Messenger* es una red de MI de Microsoft que se introdujo en el mundo de Internet en Julio de 1999. El protocolo de *MSN* está basado en ASCII y utiliza sockets TCP para intercambiar una serie de comandos entre el cliente y el servidor.

El protocolo Jabber (Jabber, 2010) trata datos mediante protocolos XML y tecnologías que permiten a cualquier par de entidades de Internet intercambiar mensajes entre ellos, determinar si están presentes o no en las respectivas listas de contactos y enviar cualquier otro tipo de información estructurada en tiempo real. Las tecnologías basadas en Jabber ofrecen ventajas, ya que el protocolo es gratuito, libre, abierto, público y fácil de comprender. Los protocolos con núcleo de XML han sido formalizados y aprobados como MI y presentan una tecnología bajo la estructura llamada XMPP (Extensible Messaging and Presence Protocol, Protocolo ampliable de mensajería y presencia). La arquitectura de la red de Jabber es similar a la del correo electrónico porque cualquiera puede lanzar su propio servidor Jabber, permitiendo a individuos u organizaciones tomar el control sobre su propia MI.

2.3. Educación en línea

Hoy en día aún no está claro hacia dónde se dirige el aprendizaje informatizado basado en Internet. El concepto de educación electrónica, o *e-learning*, ha atraído la atención de los educadores, ya que las tecnologías en desarrollo son beneficiosas para mejorar la calidad de la educación. Se ha investigado mucho sobre este tema y se han presentado soluciones muy satisfactorias mediante el uso de videotutoriales (Puig-Centelles, Ripolles, Belmonte, Arregui, Coltell & Chover, 2010) o telelaboratorios (Wirz, Marín, & Sanz, 2009). Muchos artículos tratan los crecientes retos del aprendizaje virtual para los profesores, lo que puede también afectar a sus propias evaluaciones como docentes por parte del alumnado (Carr & Hagel, 2008).

Existe un interés creciente en el desarrollo de aplicaciones educativas dentro de mundos virtuales (Bronack, Sanders, Cheney, Riedl, Tashner & Matzen, 2008), existiendo ya soluciones comerciales como *Thinking Worlds* (Caspian Learning, 2010). Fruto de este interés ha sido la aparición de numerosos portales web, proyectos de investigación e incluso cursos de especialización, como el de “Mundos virtuales aplicados a la educación” ofrecido por la Universitat Oberta de Catalunya desde Febrero de 2010.

Otros investigadores están también trabajando en temas similares, que parecen también muy prometedores. En Unype (Unype, 2010), se intenta combinar el uso de una cuenta *Skype* existente con las aplicaciones de *Google Earth* y *Google Maps*. De esta manera se crea una experiencia virtual multiusuario en la que es posible ver a los avatares de los usuarios moviéndose dentro de *Google Earth*. Como alternativa, también cabe mencionar el muy interesante desarrollo de *Sloodle* (Kemp, Livingstone & Bloomfield, 2009), que combina los mundos virtuales de *Second Life* con la plataforma de educación *Moodle*.

3. Una Infraestructura 3D para Mensajería Instantánea

En la sección anterior hemos presentado diferentes características importantes de los entornos virtuales y las aplicaciones de MI. Tras esto, queremos presentar las distintas ideas que nos han animado a realizar la integración de una representación 3D dentro de una aplicación de mensajería.

Comunicarse en un entorno 3D ha sido posible durante muchos años y hoy en día existen muchas soluciones disponibles. Algunas de ellas presentan una calidad muy buena y suponen comunidades de millones de usuarios. No obstante, la generalización de las aplicaciones de MI en las que las comunicaciones tienen lugar con personas que ya conoces ofrece una estructura más cómoda para la comunicación por Internet y ha relegado los chats 3D a un segundo plano. Como consecuencia, en este momento los usuarios de MI superan con mucho la cantidad de clientes que se conectan a estos chats 3D.

Nuestra propuesta es el desarrollo de un entorno 3D en el que los usuarios puedan comunicarse y que, al mismo tiempo, reúna las capacidades y facilidad de uso de una aplicación de MI. Entendemos que las comunicaciones con miembros de una lista de contactos ofrece un entorno mucho más adecuado para el ámbito educativo, ya que cada usuario sólo verá e interactuará con los usuarios que pertenezcan a su lista de contactos, a los componentes de una clase o a toda la comunidad educativa.

En primer lugar, es importante comentar que no desarrollaremos un nuevo protocolo. Escogeremos uno de los disponibles, de modo que las redes educativas que utilicen ese protocolo podrían adoptar nuestra solución fácilmente. Queremos mantener su lista de contactos ya existente y es necesario que el uso del entorno 3D no afecte a la comunicación. Así, no debe haber problema en establecer una conversación entre un cliente que use el entorno 3D que proponemos y un cliente que use el interfaz clásico. Crearíamos un avatar para cada contacto y representaríamos su estado mediante los efectos que el motor proporcione. Así, el uso combinado de animaciones e inteligencia artificial podría aplicarse para simular el comportamiento de los contactos dentro del entorno.

Es importante comentar que el objetivo principal es sustituir el clásico interfaz de MI por una escena 3D que contenga la lista de contactos. Por lo tanto, el usuario podrá comunicarse e interactuar con los contactos dentro del escenario, pero los contactos no accederán a este entorno 3D. Sólo estamos proponiendo un interfaz 3D para la MI, no estamos creando un mundo virtual. Una posible ampliación de nuestro concepto inicial podría ser permitir al resto de usuarios de la lista de contactos el acceso al mundo personalizado para explorarlo, dar una vuelta, tener tutorías comunes y participar en juegos de investigación (Gálik, 2006).

Nuestra propuesta de utilizar la tecnología de un motor de juegos para generar el entorno 3D tiene muchas ventajas para visualizar las escenas 3D, porque evita todos los costes de gestión y la complejidad que un metaverso supone. También evitamos problemas con las redes, como el ancho de banda a la hora de efectuar descargas, de moverse interactivamente, etc. Además, no hay problema con el almacenamiento, las conexiones en red y las copias de seguridad, porque todo sucede localmente sin tráfico extra para la red.

3.1. Metodología

Un primer paso en el desarrollo de nuestra aplicación de MI con un interfaz 3D es la selección de las plataformas que serán utilizadas para enviar y recibir mensajes y para la visualización. Así, deberemos elegir un protocolo de comunicación, un cliente de MI y un motor de juegos que nos proporcione el entorno 3D.

Tras haber revisado los diferentes protocolos de MI, nos centraremos en el protocolo Jabber (Jabber, 2010). Es el más adecuado para nuestros objetivos debido al hecho de que es abierto, gratuito y portátil entre diferentes aplicaciones como *Gtalk*, *Gaim* o *Pandion*.

Es también necesario elegir una aplicación de MI para realizar las comunicaciones por medio del protocolo Jabber. Pandion (Pandion, 2010) es un cliente de MI para redes XMPP y Jabber que se ha diseñado para un manejo sencillo, e incluye características interesantes como emoticonos personalizados, avatares y el hecho que se puede ampliar fácilmente instalando plug-ins.

El protocolo Jabber y la aplicación Pandion han sido las elegidas por nuestra Universidad para desarrollar su propia plataforma de MI. Esta institución ejecuta actualmente un servidor que usa Jabber y que permite a toda la comunidad universitaria utilizar su propia cuenta de usuario para comunicarse con cualquier servidor del mundo que utilice Jabber como protocolo. Como representamos en la Figura 1, los servidores Jabber pueden comunicarse entre sí porque tienen el mismo protocolo. Además, es posible comunicarse con otro tipo de servidores con los llamados *transportes*. Una de nuestras metas es mantener un contacto cercano entre nuestro grupo de investigación y el área técnica informática de nuestra universidad, para mejorar su aplicación para comunicarse con una aplicación cliente de MI en 3D.

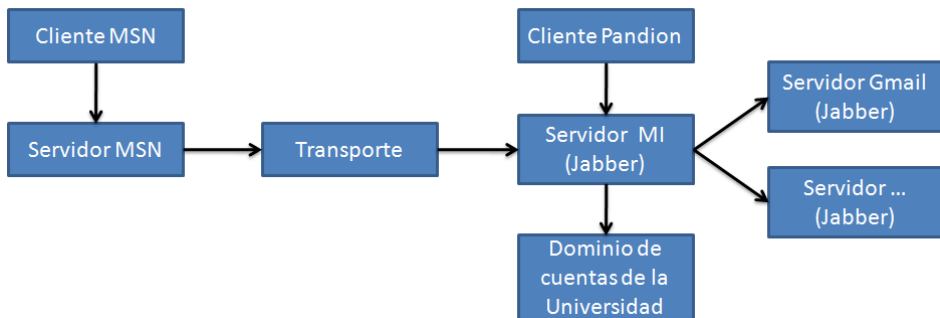


Figura 1 – Comunicación en red entre clientes de MI y servidores

Crear nuestra aplicación de MI sobre un motor de juegos puede permitirnos generar gráficos 3D realistas y fáciles de personalizar. Nosotros hemos elegido el motor Torque (Torque, 2010). Torque es un motor de juegos 3D que permite a los desarrolladores crear juegos comerciales, aunque nosotros usaremos sus capacidades de visualización para crear entornos donde los usuarios puedan establecer sus propias comunicaciones. Una importante ventaja de Torque es su habilidad para comunicarse con otros programas por Internet.

4. Detalles de Implementación

Una vez hemos seleccionado los elementos que compondrán el entorno virtual 3D para la MI, debemos encontrar el modo más adecuado de añadir una instancia en ejecución del motor de juegos seleccionado en Pandion. El primer paso hacia nuestro objetivo requiere el estudio de Pandion para entender las diferentes posibilidades que ofrece para ampliar sus capacidades y elegir la que sea más adecuada para nuestros propósitos.

4.1. Análisis de las Posibilidades de Expansión de Pandion

Pandion es una aplicación HTML (HTA). Este lenguaje de etiquetas para la definición de interfaces de usuario fue creado por Microsoft y posibilita el desarrollo de aplicaciones finales utilizando JavaScript HTML. Esta aplicación tiene dos rasgos principales que permiten a los usuarios personalizarlo: el uso de plug-ins (XML) y el uso de un fondo (HTML) para visualizar las conversaciones

Plug-ins de Pandion

Un plug-in de Pandion se encapsula en un único documento XML (Deckers 2004). Los plug-ins son extraídos, instalados y cargados cuando Pandion se inicia. Es posible tener más de un plug-in cargado al mismo tiempo en la misma estructura principal, de manera que se pueden usar distintas pestañas que permitan elegir entre los plug-ins disponibles (ver Figura 2).

En Pandion hay distintos tipos de plug-ins. Algunos de están ideados para situarse en la lista de contactos de Pandion, mientras que otros se basan en código únicamente (Deckers 2004). Los primeros consisten en código XML incrustado en el DOM (Modelo de Objetos del Documento) de *src/main.html*, que es la principal estructura de Pandion y donde se muestra la lista de contactos. Entre los plug-ins de esta categoría, la diferencia entre ellos reside en el tipo de documento HTML que implementan. Un tipo de plug-in define una pestaña de tipo `<iframe>`, que es un elemento HTML que permite incrustar otro documento HTML en su interior. El otro tipo utiliza un elemento de tipo `<div>`. Este elemento `<div>` se inicializa con los datos del primer elemento de tipo `<html>` en el archivo XML del plug-in.

Respecto a los plug-ins que se basan en código, se recomiendan para elementos que no necesitan crear una pestaña. Así, esta es la clase que se recomienda para plug-ins que no requieran visualización o abrir ventanas personalizadas. Este tipo de plug-in se utiliza para controlar eventos, como los resultantes de un mensaje entrante, de una solicitud de comunicación de presencia o incluso de pinchar con el botón derecho del ratón en la imagen de un contacto.

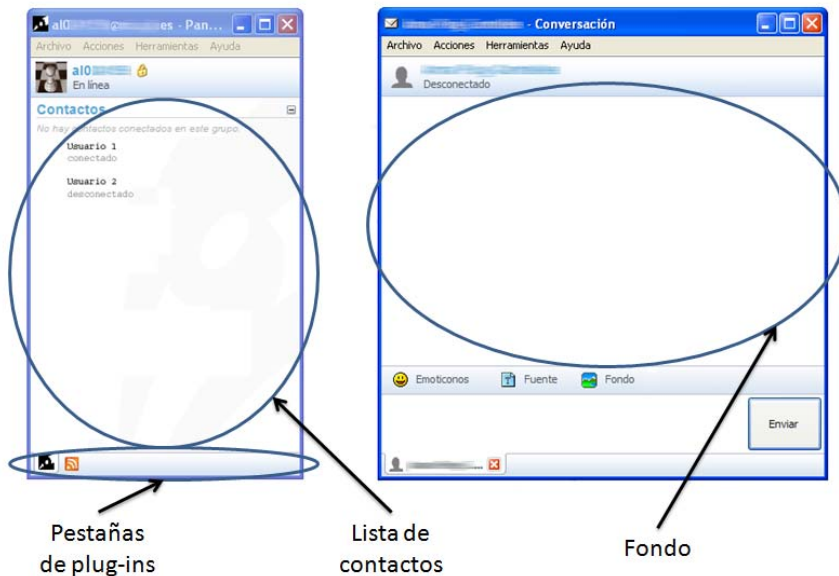


Figura 2 – Principales componentes de Pandion

Fondos de Pandion

Un fondo Pandion es el otro rasgo importante para ampliarlo, ya que nos proporciona la capacidad de crear fondos personalizados para nuestros chats. Un fondo Pandion es un código HTML que implementa un contenedor en el que los mensajes de la comunicación son visualizados. Los fondos contienen las funciones de JavaScript necesarias para manejar este comportamiento.

4.2 Integrar un motor de juegos en Pandion

En nuestro caso, será necesario extender las funcionalidades de Pandion utilizando tanto un plug-in como un fondo. La estructura básica para integrar un motor de juegos en Pandion implica elaborar un fondo donde ejecutar el motor y usar un plug-in Pandion para establecer la comunicación entre ellos. Como los plug-ins Pandion manejan todos los eventos necesarios, podríamos obtener, por ejemplo, el estado de un contacto y simularlo en el motor. Sabríamos quién está conectado, quien está desconectado, quién está ocupado, quién no está, etc.

Es importante subrayar que la instancia de motor se mostrará como fondo, mientras que toda la comunicación entre usuarios se llevará a cabo por Pandion. Así, Pandion enviará al motor toda la información y mensajes recibidos por otros usuarios, así como la información y mensajes que los usuarios envían. Esto es debido a que el motor sólo funciona como herramienta de visualización, actuando simplemente como cualquier otro modo de personalizar la apariencia de los mensajes visualizados en la aplicación de MI.

Los plug-ins nos proporcionan muchos modos de facilitar el acceso a una instancia del motor dentro de Pandion. Podríamos crear un menú desplegable para permitir al usuario escoger entre un entorno clásico 2D para iniciar una conversación o nuestra nueva implementación 3D. Esta última aparecería como una ventana emergente en la que el motor se ejecutaría inmediatamente. Consideramos que ésta sería una manera elegante de acceder al motor.

En la Figura 3 presentamos un diagrama visual resumiendo la integración del motor de juegos en Pandion. Como ya hemos explicado, la idea principal es integrar una instancia del motor dentro de un documento HTML, y posteriormente introducir este HTML como fondo para Pandion. Con este fin, será necesario integrar inicialmente el motor en un documento HTML que pueda visualizarse en un navegador web mediante un plug-in para Netscape y un control ActiveX para Internet Explorer. Todos los códigos relacionados con este proyecto se combinan con el código del motor y se recompilan para obtener los recursos deseados. Por una parte, el plug-in para Netscape usa tecnología XPConnect. Esta tecnología nos proporciona un modo de acceder al código interno del motor de juegos desde el código JavaScript. Por otra parte, ActiveX para Internet Explorer emplea su propia tecnología para darnos la misma funcionalidad del plug-in para Netscape.

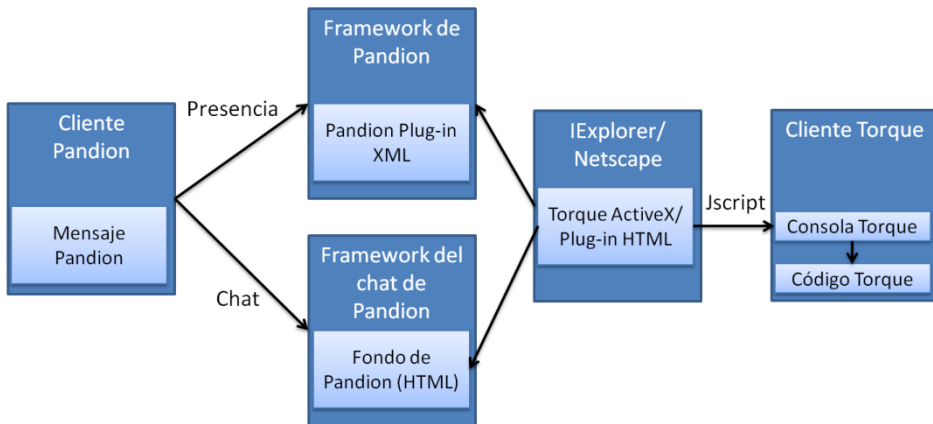


Figura 3. Diagrama para la integración del motor de juegos en Pandion.

Una vez hemos alcanzado este punto, ya somos capaces de manipular el motor de juegos desde un código HTML utilizando JavaScript. El paso siguiente será integrar este código en Pandion como fondo, de forma que el motor se convierta en el contenedor de la conversación. De este modo obtenemos un entorno interactivo en una aplicación de comunicación de MI común.

Con todo el proceso que hemos expuesto, podemos visualizar el motor en una ventana Pandion pero aún no somos capaces de comunicarnos con él. La comunicación entre Pandion y el motor de juegos será unidireccional, lo que implica que será Pandion el que se comunique con el motor. Dado que el motor de juegos actúa como un simple contenedor de la conversación, no es necesario que cada cliente que tome parte en la comunicación tenga una instancia en ejecución del motor. Así, en una conversación es

posible que algunos de los participantes estén ejecutando el nuevo entorno 3D mientras otros utilizan el interfaz clásico Pandion o cualquier otra personalización. En la siguiente sección analizaremos con detalle cómo establecer la comunicación entre Pandion y el motor de juegos.

4.3 Intercambio de mensajes de texto

La información requerida por el motor de juegos para proporcionar las características anteriormente mencionadas se extrae directamente de los mensajes del motor y paquetes de protocolos de presencia. El protocolo de mensajes del motor de juegos nos permitirá transmitir mensajes Pandion al cliente del motor de juegos que actúe como fondo Pandion personalizado.

Jabber usa un paquete de mensajes que consiste en un elemento *<message>* con los típicos atributos: *de*, *a*, y *paquete id* (Shigeoka 2002). Entre los mensajes que soporta Jabber, el tipo de mensaje en el que estamos interesados es el tipo chat. Esta clase de mensaje es la que los usuarios de Jabber envían y reciben de otros usuarios que estén en línea al mismo tiempo. Estos mensajes tienden a ser cortos y coloquiales, como el tipo de comunicación que se efectúa por teléfono.

Extraeremos la información contenida en un mensaje de chat para localizar el nombre del emisor y del receptor así como el mensaje que se envía. Pandion es el agente a cargo de esta tarea, y una vez obtenida esta información la pasará al fondo actual. El fondo normalmente imprime el mensaje en una imagen HTML. Sin embargo, como hemos convertido el fondo en una instancia del motor de juegos, la información arriba mencionada se pasará al motor mediante una función JavaScript capaz de acceder al núcleo del motor gracias a la tecnología XPCConnect o ActiveX.

Esta función JavaScript podría interactuar con cualquier función esencial del motor de juegos. Como las funciones de consola forman parte del núcleo de los motores, utilizaremos el lenguaje de scripting del motor. Por lo tanto, hacer uso de la función Jscript es como utilizar la consola del motor. En nuestro caso, usaremos esta función para llamar a la función de consola del motor de juegos que imprime la información sobre el mensaje en la ventana de diálogo principal del motor de juegos. De este modo hemos construido un fondo Pandion simple e interactivo. Tomaremos del protocolo de presencia la información sobre nuestros contactos, y usaremos este protocolo para actualizar el valor de presencia.

Como es de imaginar, estamos interesados en mostrar una etiqueta de estado. Queremos que el mensaje de estado se imprima al lado del nombre que aparece sobre la cabeza del avatar. Como se ha comentado previamente, queremos aprovechar las posibilidades de visualización del motor para simular el estado en nuestro avatar. Una función de JavaScript lo hará por nosotros: la información sobre presencia se pasará al motor de juegos mediante esa función e imprimirá el estado del avatar o creará el efecto en la forma del avatar.

5. Resultados

Tras haber completado el proceso de integración, hemos alcanzado una de nuestras metas: integrar un motor de juegos en la aplicación de MI Pandion. Incluso hemos conseguido el objetivo de permitir a los usuarios, representados como avatares, comunicarse entre sí en este entorno 3D. En este momento somos capaces de presentar la primera versión de nuestra MI con un contexto 3D. En la Figura 4 podemos ver una instantánea del motor de juegos funcionando dentro de Pandion. En la parte izquierda de la imagen podemos ver la lista de estudiantes del curso conectados al mismo tiempo, mientras que en la parte derecha el usuario puede explorar el mundo virtual de su curso y comunicarse directamente con el profesor.

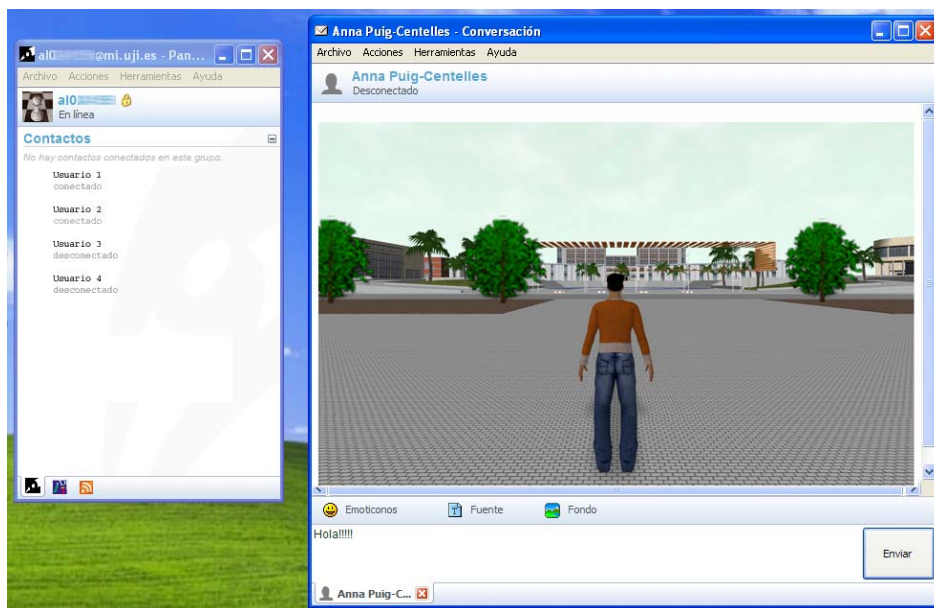


Figura 4. El motor de juegos trabajando dentro de la aplicación de MI Pandion.

6. Conclusiones

Hemos presentado una infraestructura para integrar un entorno 3D en una aplicación de MI para ofrecer un sistema de aprendizaje colaborativo dirigido a las instituciones educativas. Esta infraestructura se deriva de consideraciones de la educación científica y de resultados de la investigación en gráficos 3D.

Hemos considerado los diferentes aspectos de una aplicación de mensajería y proporcionamos diversas funciones y técnicas para la visualización gráfica de datos. De esta forma, hemos desarrollado nuestro propio enfoque sobre la aplicación actual donde hemos integrado el motor de juegos en la aplicación de MI Pandion. Debido a la sencilla funcionalidad de ampliación de Pandion mediante plug-ins, hemos desarrollado un plug-in para visualizar las escenas ofrecidas por el motor dentro de la

estructura de Pandion. Así obtenemos algunos resultados iniciales. Es importante subrayar que nuestra solución asegura la plena compatibilidad con los usuarios que no estén ejecutando la versión del entorno 3D.

Como dirección futura de nuestro trabajo, nos gustaría continuar mejorando la aplicación presentada. Por ejemplo, los plug-ins de Pandion también nos permitirían ejecutar el motor en la ventana de la lista de contactos, pudiendo crear una lista de contactos con un nuevo estilo 3D utilizando el motor.

Por otra parte, debemos considerar llevar a cabo estudios cualitativos y cuantitativos para evaluar el impacto de estas tutorías en tiempo real mediante una herramienta de MI en 3D. Por ejemplo, podemos proponer actividades de tutoría con enfoques pedagógicos entre los agentes implicados, como los estudiantes y los profesores. Además, queremos explorar el papel que este software puede jugar a la hora de facilitar y desarrollar el proceso educativo, teniendo un diálogo en un entorno de aprendizaje 3D.

Agradecimientos

Este trabajo ha sido financiado por la Universitat Jaume I (beca PREDOC/2005/54), el Ministerio de Ciencia y Tecnología (proyectos TIN2007-68066-Co4-02, TIN2007-68066-Co4-01 y TSI-020400-2009-0133), por Bancaja (proyecto P1 1B2007-56) y por la Unión Europea (proyecto ITEA2 IPO8009).

Referencias

- Bronack, S., Sanders, R., Cheney, A., Riedl, R., Tashner, J. & Matzen, N. (2008). Presence Pedagogy: Teaching and Learning in a 3D Virtual Immersive World. *International Journal of Teaching and Learning in Higher Education*, 20(1), 59-69.
- Deckers, S. (2004). Pandion plug-in specification.
- Gálik, Z. (2006). Investigative games and internet on-line activities. *Current Developments in Technology-Assisted Education*, 1619-1623.
- Carr, R. & Hagel, P. (2008). Students' evaluations of teaching quality and their unit online activity: an empirical investigation. *25th Conf. of Australasian Society for Computers in Learning in Tertiary Education (ASCILITE)*, 152-159.
- Caspian Learning (2010). <http://www.caspianlearning.co.uk>. Último acceso 1 de Abril de 2010.
- Chover, M. & Vivó, R. (2004). ¿Hay vida en el ciberespacio?, *Novática: Revista de la Asociación de Técnicos de Informática*, 168, 46-49.
- Gehorsam, R. (2003) The Coming Revolution in Massively Multiuser Persistent Worlds. *Computer*, 36(4), pp 93-95.

- Jabber (2010). <http://www.jabber.org>. Último acceso 1 de Abril de 2010.
- Kemp, J., Livingstone, D. & Bloomfield, P. (2009). SLOODLE: Connecting VLE tools with Emergent Teaching Practice in Second Life. *British Journal of Educational Technology*, 40(3), 551-555.
- Nijholt, A. (2005). Meetings in the virtuality continuum: Send your avatar. *Proceedings International Conference on CYBERWORLDS*, 75–82.
- Pandion (2010). <http://www.pandion.be>. Último acceso 1 de Abril de 2010.
- Puig-Centelles, A., Ripolles, O., Belmonte, O., Arregui, M., Coltell, O. & Chover, M. (2010). Multimedia Autonomous Learning Based on Video Tutorials. *International Technology, Education and Development Conference (INTED)*.
- Powers, M. B. & Emmelkamp, P. M. (2008) Virtual reality exposure therapy for anxiety disorders: A meta-analysis. *Journal of Anxiety Disorders*, 39(3), 250-261.
- Shigeoka, I. (2002). *Instant Messaging in Java: The Jabber Protocols*. Manning Publications.
- Torque (2010). <http://www.garagegames.com>. Último acceso 1 de Abril de 2010.
- Unype Virtual World (2010). <http://unype.com/>. Último acceso 1 de Abril de 2010.
- Virtual Worlds (2010). <http://www.virtualworldsreview.com/>. Último acceso 1 de Abril de 2010.
- Wirz, R., Marín, R. & Sanz, P. J. (2009) Design and Educational Issues within the UJI Robotics Telelaboratory: A User Interface Approach. S.G. Tzafestas (ed.), *Web-Based control and Robotics Education, Intelligent systems, control and Automation: Science and Engineering*, 38.