

3DTour - Mundos virtuales 3D aplicados al sector turístico

Miguel Coronado, Carlos A. Iglesias
Departamento de Ingeniería de Sistemas Telemáticos
Universidad Politécnica de Madrid
Avda. Complutense, 30. CP:28040 Madrid
miguelcb@gsi.dit.upm.es, cif@gsi.dit.upm.es

Guillermo Hernández
Idea Informática
Grupo Gesfor
C/ San Sotero, 11 CP:28037 Madrid
ghernandezc@grupogesfor.com

Resumen—El artículo presenta el mundo virtual 3DTour. El objetivo del proyecto es recrear un hotel dentro de un mundo virtual y poblarlo con empleados y clientes controlados por bots a través del desarrollo de un módulo para la integración de bots en mundos virtuales. El objetivo es estudiar la viabilidad de utilizar dicho escenario como un expositor para clientes de los servicios ofrecidos por hoteles, y evaluar el comportamiento de los usuarios del mundo virtual ante la presencia de avatares controlados por bots para poder determinar si la actitud de estos frente a un bot difiere al desconocer la naturaleza del mismo.

Palabras Clave—Mundo virtual, Wonderland, chatbot, AIML, turismo

I. INTRODUCCIÓN

Prácticamente desde el comienzo de la andadura de los ordenadores, los programadores han tenido la ambición de crear programas que fueran capaces de mantener una conversación con un interlocutor humano. Posiblemente comenzaron como un juego o entretenimiento para el programador pero hoy en día están muy extendidos y cada vez más perfeccionados. Los primeros bot conversacionales o chatterbots, como son conocidos, datan de la década de los 60, era necesario un ordenador dedicado a su funcionamiento y sólo una persona podía utilizarlos a la vez. En la actualidad, con el gran desarrollo de la Web es imposible imaginarlos desligados de Internet y son capaces de atender multitud de conversaciones simultáneamente. Ya no sólo aparecen como aplicaciones de demostración en la página del autor sino que se encuentran integrados, por ejemplo, en canales de IRC, como agentes en sistemas de mensajería instantánea, agentes expertos en FAQ, servicios de atención al cliente, sistemas domóticos... Por supuesto forman parte de numerosas aplicaciones comerciales.

La evolución en la interfaz de usuario de los ordenadores ha sufrido una evolución más vertiginosa si cabe. Desde las primeras pantallas de caracteres en blanco y negro hasta las modernas pantallas en 3D o desde el primer teclado hasta los dispositivos apuntadores inalámbricos como el Wii Remote. Cada avance en este campo ha abierto las puertas a un gran número de nuevas aplicaciones: la aparición del ratón y la mejora en la tecnología de monitores permitió la aparición de sistemas operativos con ventanas, con los joystick y pads apareció una nueva generación de videojuegos, la mejora en la potencia de las tarjetas gráficas la reproducción de vídeo de alta definición...

Los mundos virtuales, se encuentran entre las últimas innovaciones en interfaces de usuario. Al igual que las descritas anteriormente, los mundos virtuales despliegan un abanico

de posibles aplicaciones y negocios en torno a ellos. Son el escenario de juegos multijugador, albergan tele-reuniones de empresas o sirven para promoción turística mediante la recreación de museos y monumentos destacados.

Los mundos virtuales ven supeditado su valor global al uso que los usuarios hacen de ellos. El mundo virtual con mayor nivel de detalle en la construcción del escenario no tendrá valor alguno si no hay una comunidad de usuarios que lo pueble diariamente, ya que ningún usuario querrá utilizar dicho mundo si no hay nadie con quien interactuar.

La tecnología de bots puede ayudar a suplir esta carencia, puesto que se se pueden incluir avatares en el mundo virtual asociados a bot conversacionales de manera que otros usuarios no solo perciban su presencia, sino que puedan mantener una conversación en lenguaje natural o incluso llevar a cabo transacciones dentro del mundo virtual.

En este artículo se describe un sistema en el que se ha completado con éxito la integración de bots en un mundo virtual, un escenario actualmente en auge, y orientado hacia una aplicación corporativa del sector turístico. La plataforma integra un mundo virtual creado utilizando Wonderland y diversos bots con distintas personalidades creados utilizando la tecnología de bots AIML.

Este trabajo se ha desarrollado dentro del proyecto 3DTour Mundos virtuales 3D aplicados al sector turístico para la creación de un mundo virtual que contenga varios agentes expertos en información turística de Madrid.

El resto del artículo se estructura como sigue. La sección II presentan los retos tecnológicos que se presentan al acometer este proyecto. A continuación, la sección III presenta la arquitectura del sistema centrándose en los dos componentes principales, el mundo virtual en Wonderland y el servidor de bots con la base de conocimiento desarrollada para este proyecto. Por último, la sección IV recoge las principales conclusiones y los trabajos futuros.

II. RETOS TECNOLÓGICOS

Desde el punto de vista tecnológico, el proyecto 3DTour plantea diversos retos. Por un lado, se pretende innovar utilizando la tecnología de mundos virtuales 3D en auge actualmente, sobre todo en el sector de ocio, enfocándolo hacia una aplicación corporativa del sector turístico. Se pretende así explotar la interactividad e inmersión presente en todo mundo virtual para ofrecer una visión lúdica que permita captar de manera eficaz la atención de los usuarios. Por otro lado, para dotar de más realismo a los servicios que se ofertarán en el

mundo virtual se integrará la tecnología de bots, que además facilitará el acceso a la información a los clientes. Puesto que el entorno es propicio para camuflar los bots como otros usuarios humanos se incluirán también otros bots con el papel de usuarios en el mundo virtual y no solo empleados de manera que el usuario que se conecta perciba actividad en el escenario y se encuentre acompañado mientras explora el entorno virtual diseñado para 3DTour.

Los mundos virtuales tratan de simular entornos artificiales semejantes a la vida real en los que el hombre interactúa con máquinas. Por ello, facilitan una comunicación con mayor interactividad e inmersión, que crea una nueva forma de socializar a las personas, entretener, innovar y realizar transacciones comerciales.

Actualmente los mundos virtuales tienen una gran acogida, sobre todo dentro del sector ocio. Debido a este auge y como hemos visto, han surgido múltiples mundos virtuales, entre los que podemos destacar Second Life, OpenSim y Wonderland [5]. Para el desarrollo de 3DTour se ha decidido utilizar Wonderland, la tecnología de Sun Microsystems para desarrollar nuestro mundo virtual que se enfocará en los servicios turísticos.

El Proyecto Wonderland es una herramienta 3D para crear mundos virtuales colaborativos. Tiene como objetivo proveer un mundo seguro en el cual las organizaciones puedan hacer negocios o permitir la colaboración online entre empleados.

Wonderland está diseñada con una arquitectura modular y abierta que es compatible con los estándares. y el foco en prestaciones profesionales que incluyen vídeo de alta definición, audio hifi, grabación de sesiones, gestión de acceso a recursos, federación de mundos virtuales, realidad mixta, etc.

Está orientada a entornos profesionales y está ya siendo utilizada en escenarios de colaboración y formación inmersiva en grandes organizaciones (y las administraciones públicas lo son, incluyendo especialmente la educación y la sanidad).

La elección de Wonderland como herramienta 3D para 3DTour supone una serie de ventajas con respecto a otros mundos virtuales como Second Life. Wonderland [5] proporciona mundos virtuales centrados en el trabajo colaborativo que permiten conversaciones de voz entre avatares y la posibilidad de compartir aplicaciones dentro del mundo virtual, todos estos aspectos no soportados por Second Life. Al estar basado en la tecnología Java, es multiplataforma, por lo que puede ejecutarse en los principales sistemas operativos del mercado. Su código fuente está disponible para el cliente y para el servidor bajo licencia GPL, lo que hace posible que -al contrario de lo que pasa en entornos centralizados como Second Life-, cada usuario pueda tener su propio servidor de Wonderland y dar acceso a él a quien quiera.

Como se ha dicho anteriormente, la integración de la tecnología de bots en el proyecto, facilitará el acceso de los clientes a la información. Los bots [11] son programas diseñados para interactuar con otros programas, servicios de Internet u operadores humanos de manera semejante a como lo haría una persona. Además, existen bots de charla o bots conversacionales que simulan mantener una conversación con una persona, ya sea de tipo texto a través del teclado o por medio de sonido dotando de mayor realismo a la interacción con el usuario. Esto nos permite exponer la información de

los servicios turísticos de forma real e incluso permite al usuario interactuar con la organización fácilmente, aportando gran dosis de realismo a los procesos que se van a llevar a cabo.

A. Aplicaciones y escenarios de mundos virtuales

La tecnología de mundos virtuales abre un amplio abanico de posibles aplicaciones y escenarios. Su principal baza es ofrecer al usuario la posibilidad de hacer algo que no puede hacer, en ese momento y en ese lugar, en la vida real. El avatar del usuario, puede transportarse inmediatamente a puntos del mundo virtual pudiendo visitar cualquier monumento y lugar de interés recreado en el entorno virtual, permite realizar varias acciones a la vez pudiendo mantener varias conversaciones al mismo tiempo o mientras se asiste a una exposición o una reunión.

Muchos mundos virtuales están orientados a entornos profesionales -los que no están orientados al ocio- y son utilizados en escenarios de colaboración y formación inmersiva en grandes organizaciones.

Como ocurre con otras tecnologías innovadoras, la tecnología de mundos virtuales se está aplicando a la educación [15] y formación de personal. Por ejemplo, en el entorno sanitario se puede educar a los jóvenes mediante un juego en buenas prácticas de salud con niveles de dificultad crecientes, competiciones individuales y por equipos; en el ámbito docente se puede extender el mundo virtual para permitir a los alumnos realizar simulaciones de experimentos, en el campo de la seguridad en empresas o edificios públicos recreando simulaciones de incendios en los que se pueda incluir un fuego en el escenario para hacerlos más realistas...

Los mundos virtuales también son una herramienta con múltiples posibilidades en la propagación y preservación de la cultura [9]. Actualmente existen varios mundos virtuales para fomentar el turismo y el acceso a la cultura. Todos ellos se centran en la recreación de monumentos, museos, ciudades u oficinas de turismo, los cuales únicamente proporcionan información. Este es el caso del casco histórico de Gijón o la plaza mayor de Valladolid.

Sin embargo hay que saber valorar las posibles aplicaciones con sentido común: si en un mundo virtual un usuario puede transportarse de un punto a otro a su antojo no tiene sentido que se diseñe un coche o un carruaje como medio de transporte por muy realista que sea, pues ningún usuario querrá usarlo ya que es una pérdida de tiempo y no tiene valor para el usuario [13]. Igual ocurre con las propuestas de recreación de aulas de universidad, de manera que se imparta clase "presencial" a través del mundo virtual en lugar de dar clase en las aulas de la universidad. Si se plantea la clase como una reunión en la que los alumnos se citan con el profesor en el mundo virtual y éste expone la lección con una presentación de diapositivas, es más que probable que las transparencias se vean mejor impresas en el pupitre del alumno o en su navegador, de manera que hacer uso de las posibilidades del mundo virtual para mostrar presentaciones en un caso como este no tiene sentido. Sin embargo, el valor añadido que los mundos virtuales pueden aportar a un caso como éste es mostrar una presentación en 3D para comprender mejor la lección, facilitar a los alumnos el asistir a clase en caso de ser discapacitados o tener que desplazarse un largo trayecto



Fig. 1. Captura del mundo virtual de Wonderland creado para 3DTour: Información turística

para asistir a clase todos los días, o realizar simulaciones y experimentos a través de la potencia del servidor del mundo virtual [8].

Con 3DTour y gracias a la tecnología de los mundos virtuales y su integración con bots, se pretende dar valor añadido e ir más allá de la meramente informativa para que los usuarios además puedan interactuar entre ellos y con la empresa, así como proporcionar distintos servicios reales que puedan ser consumidos en el mundo virtual.

Evidentemente toda tecnología innovadora en la línea del internet del futuro ha de ir demostrando su valor en casos de uso concretos y nadie como los que conocen las necesidades de una organización para identificar las oportunidades en donde probar su utilidad. Es por eso por lo que en 3DTour se ha realizado este proyecto en un marco demostrativo orientado a el sector hotelero hablando, a través de la colaboración profesionales del sector para considerar la posibilidad de implantar el proyecto como expositor de los servicios de una cadena hotelera.

B. Tecnología de bots en mundos virtuales

Con el desarrollo tecnológico de la Web la tecnología de bots está cada vez más presente en Internet, aumenta gradualmente el número de sitios Web que dan servicios de hosting de bots [1] o que tienen un negocio relacionado con servicios de bots. Sin embargo, aunque podemos encontrar numerosos bots en páginas dedicadas a la demostración de los mismos o en sistemas de mensajería instantánea, no es tan frecuente encontrarlos en la página Web de una organización o de una empresa. Los mundos virtuales ofrecen un escenario idóneo para incorporar masivamente estos bots en un entorno en el que tienen muchas más capacidades de las que tienen en una página Web. De hecho, un usuario y un bot que controla un avatar en un mundo virtual tienen las mismas capacidades, de manera que, salvando las distancias están en igualdad de condiciones.

A lo largo de los años los investigadores han diseñado diferentes fórmulas de medir la inteligencia de un bot (al igual que para medir la inteligencia de una persona) [10]. El test más antiguo, y también uno de los más reconocidos, es el denominado test de Turing [16]. Dicho test es una prueba descrita por A. M. Turing en 1950 según la cual si un bot es capaz de hacerse pasar por un humano en una conversación, de manera que su interlocutor no lo descubra, dicho bot habrá conseguido tener inteligencia artificial.

La principal pega que tiene esta prueba está en que si se advierte al juez de que el interlocutor puede ser una máquina, este estará atento a detalles insignificantes para juzgar si existe inteligencia artificial. Por lo general, en una conversación por mensajería instantánea un usuario tiende a acortar la escritura de las palabras para escribir más deprisa, por otro lado es posible que por desconocimiento o error se cometan faltas de ortografía. Un bot que de verdad tuviese inteligencia artificial no cometería tales errores, pero al no cometerlos un evaluador podría sospechar que no es un usuario humano. Por otro lado, si no se advierte al interlocutor de la posibilidad de que esté hablando con una máquina el método no será un examen con rigor científico.

El mero hecho de que los bots sean fácilmente reconocidos como máquinas hace que los interlocutores humanos descuiden la conversación al comunicarse con ellos. Esto hace complicar la evaluación de la base de conocimiento de un bot, pues los usuarios tienden a dirigir la conversación con palabras clave en lugar de construir frases correctas. Los mundos virtuales constituyen un escenario idóneo para la evaluación de bots conversacionales pues cualquier usuario que converse con ellos tiende a considerarlos, a priori, como otro usuario del sistema [3]. De esta manera se las conversaciones registradas en el logger pueden ser utilizadas para la mejora de la base de conocimiento del bot.

Por otro lado, la integración de tecnología de bots en el mundo virtual supone un aporte de valor añadido para el mundo virtual. Añade un mayor grado de interacción, pues permite realizar acciones tales como reservar una sala de juntas, consultar las reservas disponibles o solicitar la reproducción de un video a través de una conversación en lenguaje natural, dando así una mayor dosis de realismo al escenario. En general los usuarios valoran muy positivamente los sistemas con los que se puede interactuar a través del lenguaje natural para desempeñar una tarea [12].

La inclusión de bots en el mundo virtual también convierte a este en un mundo más acogedor, pues el usuario percibe mayor grado de actividad al encontrar un mayor número de avatares en el escenario [2]. Este concepto, ha llevado a la práctica en [7] con la inclusión de bots en foros con el propósito de responder rápidamente a las preguntas cuya respuesta conoce el sistema y dar así una sensación de eficiencia en el servicio.



Fig. 2. Avatar de un bot recepcionista en 3DTour

C. Bots en el sector turístico

Al igual que los mundos virtuales, los bots conversacionales están siendo utilizados en diversos campos de aplicación como agentes que sirven información o ofrecen promociones a través de mensajería instantánea, agentes expertos en FAQ incrustados en páginas Web, servicios de atención al cliente o sistemas domóticos.

En general, para que un bot conversacional sea apto para realizar una tarea es necesario que el volumen de conocimiento que necesita conocer sea acotado. Esto no significa que el conocimiento del bot no pueda cambiar y ampliarse a lo largo del tiempo, pero sí es necesario que dicho conocimiento se ciña a un tema concreto (o a un conjunto de temas) para así poder ofrecer mejores capacidades de conversación.

Este es precisamente el caso del sector turístico. Bien sea como promotor cultural o como empleado de un complejo hotelero, el conocimiento que necesita el bot se restringe a conocimientos generales acerca de la región en que se encuentra y conocimientos más a fondo del monumento u hotel para el que trabaja. Los bots en el sector turístico constituyen una opción de valor pues prácticamente pueden desempeñar el trabajo que realizan los empleados de una oficina de turismo o un recepcionista de un hotel.

En el caso de un empleado de un hotel, el carácter inteligente de los sistemas de bots les permite actualizar su base de conocimiento de manera que siempre disponen de la última información acerca de las habitaciones disponibles, los horarios en que están ocupadas las salas de juntas, tienen acceso a la base de datos sobre precios de los servicios ofrecidos por el hotel y también pueden recoger información de festejos y eventos programados en la ciudad.

Los sistemas de bots también son capaces de recolectar información a partir de las respuestas del usuario pudiendo así intuir sus gustos y preferencias. Por lo tanto, se puede plantear la conexión de el sistema de bots con un sistema recomendador de tal manera que este aconseje a su interlocutor sobre espectáculos o actividades a realizar.

III. ARQUITECTURA DEL SISTEMA

Para 3DTour se ha desarrollado un mundo virtual que representa un complejo hotelero con diversas dependencias: hall, salones, habitaciones, sala de reuniones, recepción... Dicho escenario está integrado en un mundo virtual de Wonderland por lo que los usuarios pueden moverse e interactuar con el mundo. Para dotar al mundo de mayor realismo se han desarrollado varios bots que han sido integrados en el mundo virtual, de manera que el usuario pueda interactuar con ellos, tanto dialogando como solicitando la reserva de una habitación, de la sala de juntas o información acerca del hotel.

A. Wonderland

El proyecto Wonderland es una herramienta de código abierto para desarrollar mundos virtuales en 3D. Una de las claves del modelo de Wonderland es proporcionar un entorno robusto en términos de seguridad, escalabilidad, fiabilidad y funcionalidad de tal manera que se puedan realizar teleconferencias a través de esta herramienta.

La arquitectura cliente-servidor del proyecto Wonderland soporta la creación de numerosos mundos virtuales. Esto es



Fig. 3. Exterior del hotel construido en 3DTour

posible gracias al sistema de módulos flexibles. Este esquema permite a los desarrolladores extender la aplicación de cliente y del servidor, pero también empaquetar un conjunto de texturas o un mundo virtual completo como módulos fáciles de instalar.

Wonderland está construido sobre la plataforma Darkstar y el motor gráfico 3D jmonkeyEngine. El proyecto Darkstar es una plataforma de comunicaciones y persistencia en videojuegos encargada de gestionar el estado del cliente y del mundo mientras interactúa con Wonderland. jMonkeyEngine es un moderno motor gráfico en 3D, enteramente escrito en Java, que proporciona una APIs gráficas, funciones primitivas y soporte para shaders. Además dispone de un panel de administración Web para la gestión del mundo virtual que está basado en el servidor Glassfish.

Open Wonderland está subdividido en varios proyectos de código abierto:

- Wonderland es el núcleo de la aplicación, incluye las aplicaciones de cliente y servidor así como el panel de administración Web. El proyecto principal de Wonderland también contiene un cierto número de módulos principales responsables de funcionalidades como los avatares, aplicaciones compartidas, edición de mundos virtuales, seguridad...
- Wonderland Modules es un repositorio de módulos que constituyen un conjunto de extensiones para Wonderland. En el repositorio se incluyen tanto módulos certificados desarrollados junto al núcleo de Wonderland, tales como la grabadora de sonidos o las sticky-notes, como módulos experimentales desarrollados por programadores de Wonderland o por la comunidad.
- MTGame es un motor de gráficos de alto rendimiento que extiende a jMonkeyEngine. MTGame añade soporte multithread para presentación de gráficos mejorados haciendo uso de las capacidades del hardware actual.
- Avatares utiliza MTGame para generar personajes 3D con un aspecto realista a partir de unos archivos de plantilla genéricos.
- jVoiceBridge es una plataforma para tratamiento de audio enteramente escrita en Java. Incluye un servidor SIP y un teléfono personal. Soporta audio en estéreo de alta calidad y generación de efectos espaciales para la adaptar el audio a la ubicación en que está siendo escuchado.

En la figura siguiente se muestra un esquema con la arquitectura detallada de Wonderland.

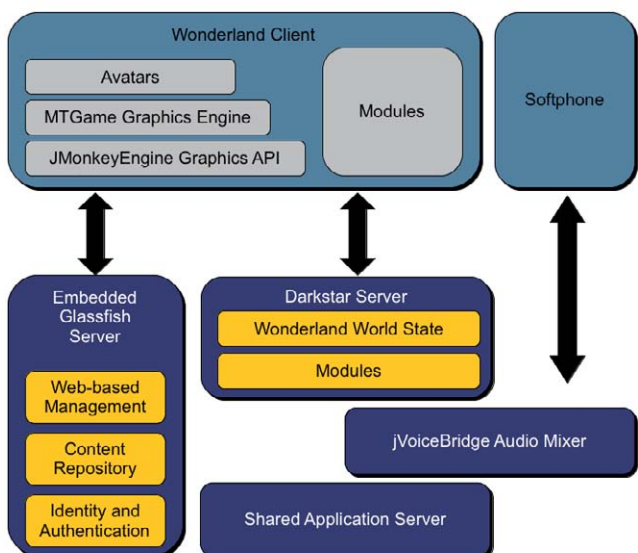


Fig. 4. Arquitectura de Wonderland

Para la integración de los bots en el mundo virtual ha sido necesario desarrollar un módulo de servidor, de tal manera que Wonderland fuera capaz de acceder al servicio de bots. La aplicación de cliente de Wonderland no ha tenido que ser modificada ya que para dicha aplicación la conversación con un bot no discierne en nada de una conversación con otro usuario pues es el servidor en el que maneja las conversaciones.

B. GSIBot

El servidor de bots GSIBot es una plataforma para el despliegue de servicio de bots conversacionales, que permite desplegar y gestionar, a través de un panel de administración Web, un número ilimitado de bots conversacionales. GSIBot ofrece facilidades para el acceso a los servicios de bots que proporciona, pues presenta diversas interfaces de acceso que permiten a múltiples aplicaciones y servicios de terceros acceder los bots por medio del protocolo HTTP a través de una url, a través de protocolos de mensajería instantánea o por medio de el estándar de comunicación FIPA-ACL utilizado en el sistema multiagente JADE [6].

AIML

Los bots son desarrollados utilizando la tecnología AIML [4]. El lenguaje AIML, acrónimo de Artificial Intelligence Markup Language, es un dialecto de XML específicamente diseñado para crear bots conversacionales. La especificación de AIML no solo describe los elementos del lenguaje sino también el comportamiento de los programas que los procesan. AIML es un lenguaje fácil de aprender que permite ampliar el conocimiento de un bot existente o crear uno desde cero en muy poco tiempo.

Un conjunto de ficheros AIML forman, por tanto, la base de conocimiento de un bot conversacional. GSIBot permite desplegar un bot conversacional a partir de uno o más archivos AIML.

Los elementos AIML de mayor nivel son:

- `<aiml>`: la etiqueta que engloba a todo un documento AIML.

- `<category>`: la etiqueta que define una unidad de conocimiento en la base de conocimiento del bot. De ahora en adelante nos referiremos a ella como una regla.
- `<pattern>`: la etiqueta que contiene un patrón que debe casar con la entrada de usuario para que se ofrezca la respuesta asociada a dicho patrón. Cada unidad de conocimiento tiene asociado un patrón.
- `<template>`: la etiqueta que contiene la respuesta a una entrada de usuario. Dicha respuesta habitualmente consta de una frase textual aunque puede contener acciones asociadas.

La especificación completa de AIML engloba un amplio conjunto de etiquetas [14].

La siguiente figura muestra un ejemplo de una regla AIML sencilla:

```

<category>
  <pattern>QUE TAL</pattern>
  <template>
    <think>
      <set name="topic">SALUDO</set>
    </think>
    Bien, ¿Y usted?
  </template>
</category>

```

Fig. 5. Ejemplo de código AIML

AIML ofrece multitud de posibilidades a la hora de confeccionar los patrones. Los patrones AIML definidos por las etiquetas `<pattern>` son ordenados, insensibles a mayúsculas y consideran los espacios en blanco. Los patrones AIML permiten la utilización de dos tipos de comodines `_` y `*`, el primero más prioritario que el segundo. A la hora de confeccionar un patrón debemos imaginar la frase que el usuario introduciría acerca de un tema, se mantienen en el patrón las palabras que son importantes para poder expresar dicha idea y el resto se sustituye por comodines. Es importante destacar que un patrón no se verifica si en el lugar de los comodines hay espacios en blanco.

Cuando la frase introducida por el usuario puede casar con varios patrones sólo se verificará aquel que se ajuste con mayor precisión, es decir, que contenga mayor número de palabras de las introducidas por el usuario. Esto es otra característica importante de AIML: el orden de escritura de las reglas no es importante. Esto permite una fácil escalabilidad del sistema. El hecho de que los comodines no se verifiquen si coinciden con espacios en blanco hace que los patrones sean más selectivos pero a su vez hace que sea necesario repetir varias veces la misma regla (con igual respuesta) con patrones con distinta disposición de los comodines.

El servidor de bots

El servidor de bots es la aplicación principal, que permite desplegar bots con conocimiento expresado en AIML. El sistema se ha desarrollado con tecnología Java Enterprise Edition y se despliega en el contenedor de servlets Apache Tomcat.

El servidor está basado en la plataforma programD como intérprete de AIML. ProgramD es la plataforma de código abierto para bots AIML más utilizada en el mundo. Implementa todas las capacidades de AIML y soporta un número

ilimitado de bots una simple instancia en el servidor. ProgramD es altamente configurable por medio de un conjunto de ficheros de configuración, que permiten activar o desactivar muchas de sus características. Una de las más interesantes es la posibilidad de definir múltiples bots con distintas personalidades, es decir, que contengan el conocimiento de distintos archivos AIML. De esta manera, si se estructura una base de conocimiento en varios archivos AIML se pueden definir bots que contengan archivos comunes y archivos propios de su personalidad.

Tal y como se muestra en la figura 6, el servidor de bots añade otros módulos y adaptadores a programD con el fin de proporcionar nuevas funcionalidades. Los principales elementos de esta arquitectura son:

- Adaptadores para la gestión multimodal de los usuarios. En el alcance de este proyecto se gestiona la entrada remota y se proporciona una interfaz HTTP abierta para su integración con otros sistemas.
- Cada uno de los bots definidos en programD se encargan de procesar las peticiones de usuarios, integrando un parser de AIML para procesar la base de conocimiento.
- Un sistema de logger integrado para gestionar el servicio y facilitar su mejora continua. Se encarga de recoger las conversaciones con otros usuarios etiquetándolas para realizar seguimiento de usuarios y control de errores.
- El panel de administración ofrece una interfaz sencilla e intuitiva para la administración del servidor. Permite visualizar el estado de los bots cargados, ofrece medidas del tamaño de la base conocimiento y del uso de los mismos. El panel de administración incorpora un script de conversación rápida con cualquiera de los bots cargados para probar el correcto funcionamiento de los mismos.
- El módulo de plugins se encarga de añadir funcionalidad adicional al lenguaje AIML. Sin extender las especificaciones se definen un sistema de meta-etiquetas incrustadas en las respuestas que son leídas por el gestor de plugins y provocan la ejecución de un proceso remoto. Tienen un papel fundamental en la integración con los mundos virtuales pues permiten transmitir instrucciones al mundo virtual.

Interfaz Web de acceso

El servidor ofrece una interfaz sencilla para facilitar su integración con los clientes (Web, IM, JADE), según la URI:

```
http://<url-servicio>?q=<consulta>&bot=<id-bot>
[&type=<response-type>]
```

donde:

- <url-servicio>es la URL donde se está ofreciendo el servicio.
- <consulta>es la cadena con la consulta del usuario.
- <id-bot>es el id del bot al que se le envía la consulta.
- <response-type>es el tipo de respuesta solicitado. Puede ser HTML, XML o JSON. Es un parámetro opcional, por defecto HTML.

La respuesta ofrecida por el bot no sólo contiene la cadena de respuesta que el bot da a la consulta sino también el valor de todos los parámetros definidos en el AIML para esa

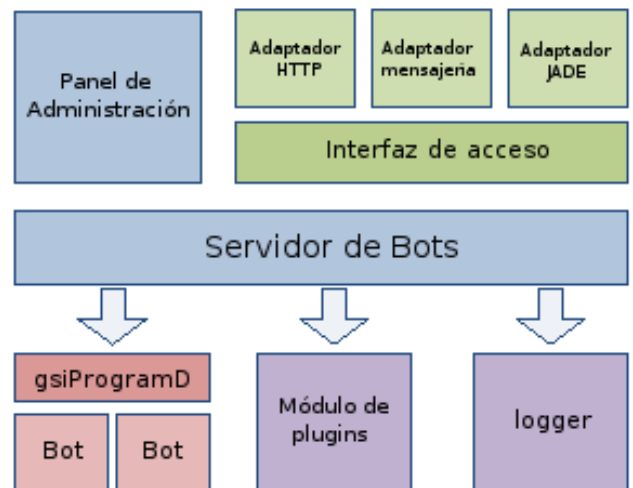


Fig. 6. Arquitectura de Gsibot

respuesta, el nombre del bot, el usuario y la última consulta del usuario. Un ejemplo de respuesta en formato JSON es:

```
{ "dialog": { "sessionid": "4A5800A89B1778",
  "bot": "3DTour",
  "q": "hola ",
  "response": "Hola, ¿qué tal?",
  "test": "General Saludo",
  "topic": "SALUDO",
  "state": "saludo"}}
```

Fig. 7. Ejemplo de código AIML

El diseño de adaptadores para acceso remoto es un elemento clave para la integración de los bots en el mundo virtual, ya que permite mantener la independencia de ambos sistemas a pesar de su funcionamiento en conjunto. El mundo virtual y el servidor de bots corren en servidores distintos, pudiendo incluso estar en máquinas distintas.

La base de conocimiento proporcionada con la aplicación ha sido diseñada con forme a los criterios de estilo mencionados en [17]. Incluye variables AIML de depuración, para mostrar los estados de ánimo del bot y para mostrar URL relacionadas con la respuesta o a las que se haga referencia en la misma. La base de conocimiento esta estructurada en distintos ficheros en función del tema del conocimiento tratado para su fácil gestión y mantenimiento. De esta manera, por medio de los archivos de configuración de programD están definidos varios bots que comparten el conocimiento de algunos de estos archivos y tienen el conocimiento de archivos propios de cada bot.

C. Herramientas para gestión y prueba del corpus

En 3DTour se han definido bots para distintos empleados del hotel gracias a esta base de conocimiento segmentada en archivos. De manera que existen archivos de conversación general que son comunes a todos los empleados y archivos de conversación propia de la recepcionista, los camareros o los botones. Los bots de clientes del hotel también comparten archivos con los empleados del hotel aunque para ellos se

Gestión de Conocimiento

Paso 1
Paso 2
Paso 3: Presentación de las reducciones
Paso 4
Paso 5

Aquí se presentan las deducciones llevadas a cabo.

Usuario:	¿	que	lugares	de	interes	puedo	visitar	?
Reducción:		que	lugares	de	interes	poder	visitar	
Usuario:	¿	que	lugares	de	interes	hay en los	alrededores	?
Reducción:		que	lugares	de	interes	hay en los	alrededores	
Usuario:	¿	que	lugares	me	recomienda		visitar	?
Reducción:		que	lugares	me	recomendar		visitar	
Usuario:	¿	que	monumentos	puedo			visitar	en los
Reducción:		que	monumento	poder			visitar	en los

Verifique las deducciones llevadas a cabo.

visitar es	<input checked="" type="radio"/>	una palabra clave	<input type="radio"/>	el topic	<input type="radio"/>	un matiz
ciudad es	<input type="radio"/>	una palabra clave	<input type="radio"/>	el topic	<input checked="" type="radio"/>	un matiz
monumento es	<input type="radio"/>	una palabra clave	<input type="radio"/>	el topic	<input checked="" type="radio"/>	un matiz
alrededores es	<input type="radio"/>	una palabra clave	<input type="radio"/>	el topic	<input checked="" type="radio"/>	un matiz
lugares es	<input type="radio"/>	una palabra clave	<input type="radio"/>	el topic	<input checked="" type="radio"/>	un matiz
interes es	<input type="radio"/>	una palabra clave	<input type="radio"/>	el topic	<input checked="" type="radio"/>	un matiz

Anterior
Siguiente

Ayuda

Configuración

Selección dinámica

Ha seleccionado la palabra visitar.

¿Desea eliminar la relevancia de dicha palabra?

Desmarcar

Cancelar

Fig. 8. Ejemplo de interfaz de usuario de gestión de preguntas

han eliminado los archivos de conocimiento general acerca del hotel y del entorno.

Para programar la base de conocimiento de un bot con capacidad para hablar razonablemente acerca de un determinado tema hace falta una gran cantidad de patrones AIML. En el caso de un bot con el conocimiento equivalente al de un empleado de un hotel, como es el caso de los bots programados para 3DTour hacen falta en torno a 4500 patrones. Realizar esta programación manualmente es un trabajo repetitivo, lento y complicado; pues es necesario escribir patrones con la misma estructura repetidas veces y por ello existe el riesgo de que al añadir un nuevo patrón se esté interfiriendo con uno incluido anteriormente o viceversa.

Para facilitar esta tarea se han diseñado una herramienta de asistencia en el desarrollo de bots escritos en AIML que cubren estas dos dificultades, el corpus tester garantiza que una base de conocimiento está libre de errores y el gestor de conocimiento permite al usuario trabajar en la base de conocimiento de un bot sin tener conocimientos en AIML.

Corpus Tester

El módulo de pruebas automatizadas, CorpusTester, es una herramienta de mantenimiento y gestión de bases de conocimiento que permite garantizar la calidad e integridad de una base de conocimiento o la independencia de dos bases de conocimiento pudiendo así mezclarse archivos de estas sin que interfieran entre sí. Con esto se soluciona el problema existente en AIML con interferencia entre patrones y la agregación de bases de conocimiento.

Está desarrollado utilizando el framework HTTPUnit para

acceso a sitios Web. Es un componente independiente que por medio de la interfaz ofrecida por bot accede a la base de conocimiento para realizar las pruebas. El funcionamiento de las pruebas está basado en un archivo XML, (Knowledge Test File) en el que se recoge el comportamiento deseado de un bot. Este archivo está formado por un conjunto de pruebas (testcase). Cada una de las pruebas está formada por una o más entradas de usuario: las que se envían al bot sucesivamente, y una o más comprobaciones: que se llevan a cabo sobre la respuesta obtenida. Tras ejecutar todas las comprobaciones el programa proporciona un informe detallado con las pruebas que se han realizado y aquellas que no se han superado.

Los archivos KnowledgeTestFile también pueden utilizarse como descriptores de requisitos del cliente. De manera que se escriben antes que la base de conocimiento y se programa esta para que cumpla con lo especificado en el KTF.

Aplicación de gestión de conocimiento.

La aplicación gestión de conocimiento permite ampliar una base de conocimiento en AIML, hace que el bot aprenda. Éste es un proceso delicado, requiere el uso de algoritmos que garanticen el funcionamiento de los patrones incluidos y a la vez que no interfieran en el conocimiento ya existente. Para ello se basa en el diseño de prioridades entre patrones de AIML para incluir patrones más ajustados al concepto para los que se garantiza que no hay interferencias y otros patrones más ambiguos que pretenden adivinar lo que el usuario quiere decir. Gestión de conocimiento está integrado con corpus tester, de manera que cada modificación en la base de conocimiento es verificada automáticamente a través de la



Fig. 9. Ejemplo de mapa de patrones generados

herramienta de pruebas.

Se pueden plantear muchas formas de automatizar la fase de enseñanza al bot. El sistema puede ser más o menos autónomo en función de la cantidad de decisiones que se toman sin consultar al usuario, decisiones de las que –por tanto– el sistema debe estar seguro –o bastante seguro. Un sistema totalmente autónomo sería capaz de, por ejemplo, leer un cierto número conversaciones guardadas hechas a través de clientes de mensajería instantánea e inducir a partir de ellas los patrones para generar AIML. Un sistema nada autónomo sería aquel que pidiera al usuario cada uno de los valores a introducir en cada uno de las etiquetas AIML. En tal caso el sistema se asemeja a un editor de XML.

A medida que la autonomía del sistema aumenta también aumenta la probabilidad de cometer un error y por tanto, crear una entrada en la base de conocimiento que sea incorrecta: al aumentar la autonomía disminuye la fiabilidad. Si el sistema no es nada autónomo, y por tanto nada inteligente, todas las deducciones hechas se corroboran con el usuario y por lo tanto se le exige mayor nivel de cualificación en el uso de la herramienta.

Se pretende liberar de trabajo al usuario y desarrollar una interfaz intuitiva cuyo uso necesite la menor explicación posible. Para ello se ha alcanzado un compromiso entre autonomía y la fiabilidad desarrollando una interfaz semiautomática o asistida. El sistema toma aquellas decisiones sobre las que posee certeza absoluta. En el resto de casos presenta al usuario las decisiones tomadas y pide su corroboración. En estos casos se traducen las consideraciones propias de AIML a términos fácilmente entendibles por el usuario.

A modo de ejemplo, consideremos que el usuario de la aplicación pretende ampliar la base de conocimiento para que el bot sea capaz de recomendar al usuario qué lugares visitar para hacer turismo. Primeramente el usuario debe introducir un conjunto de frases que pregunten por ese concepto. En la figura 8 se presenta la interfaz de usuario en un paso intermedio, en el que a partir de las frases introducidas por éste la aplicación muestra las deducciones realizadas sobre las palabras clave, las más relevantes. A partir de estas deducciones se genera un conjunto de patrones AIML para introducir el conocimiento en el bot. La lógica de patrones AIML responde al mapa mental mostrado en la figura 9. Gracias al diseño de prioridades que sigue AIML el bot siempre escogerá la respuesta más adecuada.

IV. CONCLUSIONES

La integración de bots en mundos virtuales se plantea como una asociación muy positiva para el producto final; los usuarios valoran muy positivamente los sistemas con los que se puede interactuar a través del lenguaje natural para desempeñar una tarea [12]. El sector turístico ha resultado ser un marco de trabajo idóneo para los sistemas de bots pues

se requiere tener información detallada acerca de una tema concreto y no hay necesidad de tener conocimiento acerca de un amplio conjunto de temas.

El proyecto ha concluido satisfactoriamente de manera que se estudia seguir otras vías de trabajo en la integración de tecnología de bots con mundos virtuales; tales como integrar el control del avatar a través del AIML pudiendo hacer que este se desplazase y cambie de estancias, o el desarrollo de una base de conocimiento con la capacidad de recolectar información del usuario de tal manera que la implementación de un sistema recomendador conectado al servidor que permita a los bots aconsejar a su interlocutor sobre espectáculos y actividades.

En cuanto al prototipo de gestión de la base de conocimiento, las pruebas realizadas con usuarios no técnicos han mostrado la facilidad de las herramientas de gestión de diálogos para generar conversaciones sin requerir conocimientos técnicos. Fruto de estas pruebas se trabajará en mejorar su usabilidad para que no requiera formación.

AGRADECIMIENTOS

El proyecto ha sido llevado a cabo conjuntamente por el Grupo de Sistemas Inteligentes de la Universidad Politécnica de Madrid, Idea Informática empresa del Grupo Gesfor y Skyworks.

REFERENCIAS

- [1] D. Aimless. Pandorabots - a multilingual chatbot hosting service.
- [2] R. Bartle. *Designing Virtual Worlds*. New Riders Games, 2003.
- [3] D. J. H. Burden. Deploying embodied ai into virtual worlds. *Know.-Based Syst.*, 22(7):540–544, 2009.
- [4] N. Bush. Aimpl - the artificial intelligence markup language. <http://www.alicebot.org/aiml.html>, 2005.
- [5] P. Escobar. Wonderland el mundo virtual de sun microsystems. <http://www.tecnolives.com/wonderland-el-mundo-virtual-de-sun-microsystems>, 2007.
- [6] D. G. Fabio Bellifemine, Giovanni Caire. *Programming with JADE - Basic Features*. Michael Wooldridge, Telecom Italia, Italy; Whitestein Technologies AG, Switzerland, 2006.
- [7] D. Feng, E. Shaw, J. Kim, and E. Hovy. An intelligent discussion-bot for answering student queries in threaded discussions. In *IUI '06: Proceedings of the 11th international conference on Intelligent user interfaces*, pages 171–177, New York, NY, USA, 2006. ACM.
- [8] R. Gu, M. Zhu, and N. Zhang. Adaptive web3d virtual learning environment using interest management. In *WBE '08: Proceedings of the Seventh IASTED International Conference on Web-based Education*, pages 400–405, Anaheim, CA, USA, 2008. ACTA Press.
- [9] Y.-S. Kim, T. Kesavadas, and S. M. Paley. The virtual site museum: a multi-purpose, authoritative, and functional virtual heritage resource. *Presence: Teleoper. Virtual Environ.*, 15(3):245–261, 2006.
- [10] S. Legg and M. Hutter. Universal intelligence: A definition of machine intelligence. *Minds Mach.*, 17(4):391–444, 2007.
- [11] M. Mauldin. Chatterbots, tinymuds, and the turing test: Entering the loebner prize competition (1994). In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, 2008.
- [12] S. Negi, S. Joshi, A. K. Chalamalla, and L. V. Subramaniam. Automatically extracting dialog models from conversation transcripts. In *ICDM '09: Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, pages 890–895, Washington, DC, USA, 2009. IEEE Computer Society.
- [13] Nicole.Yankelovich. Abrir wonderland. <http://mfeldstein.com/es/open-wonderland/>, 2010.
- [14] R. W. Noel Bush. Artificial intelligence markup language (aiml) version 1.0.1, 2007.
- [15] E. Prasolova-Förland. Analyzing place metaphors in 3d educational collaborative virtual environments. *Comput. Hum. Behav.*, 24(2):185–204, 2008.
- [16] A. M. Turing. Computing machinery and intelligence (1950). pages 85–107, 1989.
- [17] R. S. Wallace. *The Elements of AIML Style*. 2004.