

Nuevas Tecnologías Aplicadas a la Enseñanza de la Ingeniería. Análisis Comparativo de Ambientes Virtuales 3D

Francisco Lizarralde, Constanza Huapaya

Grupo de Investigación en Inteligencia Artificial Aplicada a Ingeniería,
Departamento de Matemática, Facultad de Ingeniería, U.N.M.d.P.,
Av. Juan B. Justo 4302, Mar del Plata, Argentina
{flizarra, huapaya}@fi.mdp.edu.ar

Resumen El objetivo del presente trabajo es presentar un análisis comparativo de las capacidades que ofrecen los entornos virtuales tridimensionales para su utilización dentro del ámbito educativo, como una herramienta que permita reducir la brecha cognitiva presente entre conceptos teóricos y su aplicación práctica, en particular, en la enseñanza de ciertos contenidos que conforman la currícula básica de diversas especialidades de Ingeniería. Para ello se estableció un marco de evaluación basado en ciertos aspectos considerados relevantes para la elaboración de contenidos y para su utilización como recurso didáctico, en especial, en lo que se refiere a la implementación de técnicas de simulación numérica. Posteriormente se aplicó este marco de evaluación a un conjunto de ambientes virtuales 3D, elegidos de acuerdo a cierta cualidad representativa, con la finalidad de evaluar comparativamente su desempeño.

Keywords: Ambientes Virtuales 3D, Enseñanza Ingeniería, Simulaciones Educativas

1. Introducción

En la enseñanza de la Ingeniería se suele analizar el comportamiento dinámico de diversos sistemas, así como también, se pone énfasis en el estudio de los modelos matemáticos que los representan. Sin embargo, a menudo los estudiantes suelen “perdersse” en tales representaciones, lo que dificulta el aprendizaje de los conceptos fundamentales. En este sentido, los medios tradicionales como el pizarrón o los libros de texto, resultan insuficientes para expresar la naturaleza dinámica de estos sistemas.

En este contexto, los entornos virtuales 3D emergen como un medio adecuado para la experimentación y la representación, tanto dinámica como espacial, de diversos fenómenos físicos, pues brindan la posibilidad de simular, experimentar y analizar el comportamiento de dichos sistemas dinámicos, bajo diferentes condiciones.

Además, existen fenómenos que por su escala, ya sea microscópica o astronómica, resultan imposibles de reproducir en el ámbito de un laboratorio. Sin

embargo, es posible estudiar el comportamiento de tales fenómenos, simulando numéricamente las leyes naturales que los gobiernan, al dotar de propiedades físicas a los objetos del ambiente virtual 3D y estudiar su comportamiento bajo la influencia de campos gravitatorios, eléctricos, magnéticos, etc.

De esta forma, las simulaciones actúan como mediadores entre el modelo matemático (*conocimiento abstracto*) y la forma en que el estudiante percibe su funcionamiento (*conocimiento concreto*). Lo cual permite abordar problemas tan complejos como el estudio de la mecánica de fluidos o las trayectorias de los cuerpos celestes y al mismo tiempo aumentar el interés y la motivación en el estudiante, mejorando claramente su aprendizaje.

En este sentido, los ambientes virtuales 3D operan como una extraordinaria herramienta para la adquisición de competencias y habilidades en un contexto simulado, permitiendo a los estudiantes aplicar los conocimientos teóricos adquiridos en el aula, en ambientes “*cuasi-reales*” representados en la computadora. Esto permite la generación de un modelo informático al que es posible someter a pruebas exhaustivas con la finalidad de comprobar su exactitud, sin el costo o el riesgo que implicaría realizar en el mundo real aquello que se afirma desde el punto de vista teórico. [1].

2. Metodología

Si bien los términos *Mundo Virtual 3D* y *Ambiente Virtual 3D* se suelen utilizar indistintamente, es conveniente aclarar, que dichos términos no son sinónimos. En este sentido, un *Mundo Virtual 3D* puede ofrecer facilidades de navegación, comunicación y juegos, pero si carece de facilidades para la creación de contenidos, personalización de espacios y programación del comportamiento de su entorno, no puede ser considerado como un *Ambiente Virtual 3D*.

Teniendo en cuenta estos aspectos y en concordancia con el alcance y los objetivos propuestos en el presente trabajo, se optó por tomar como objeto de estudio a los Ambientes Virtuales 3D, *Second Life*, *OpenSimulator* y *OpenCobalt*, debido a que cada uno presenta alguna característica relevante que lo distingue de los demás.

En este sentido, *Second Life* fue elegido por ser uno de los ambientes virtuales 3D más reconocidos, de mayor vigencia y cantidad de usuarios, contando en la actualidad con más de un millón de usuarios regulares. Este entorno posee, sin lugar a dudas, una clara finalidad comercial y de entretenimiento, aunque también es utilizado por diversas instituciones académicas para la creación de contenidos educativos, y principalmente para tener una presencia institucional en el mundo virtual.

El segundo ambiente virtual 3D considerado fue *OpenSimulator*, un proyecto surgido a partir de la liberación del código fuente del programa cliente de *Second Life*, a comienzos de 2007, lo que lo convirtió en una especie de alternativa de código abierto de *Second Life*. Esto representa un aspecto clave, ya que posibilita la instalación de servidores (*Grids*), totalmente independientes de la empresa

Linden Labs, e incluso abre la posibilidad de vincularlo con otros proyectos educativos de Software Libre, como por ejemplo *Moodle*.

Y por último, se seleccionó a *Open Cobalt*, un entorno muy poco conocido, íntegramente desarrollado en una versión moderna de *Smalltalk* llamada *Squeak* [2]. Este ambiente difiere totalmente del modelo tradicional que utilizan la mayoría de los ambientes virtuales 3D, los cuales administran todas las comunicaciones e interacciones a través de un servidor central. En este sentido, *Open Cobalt* utiliza un esquema descentralizado de comunicaciones del tipo P2P (*Peer to Peer*).

3. Marco de Análisis Comparativo

Establecer un marco para realizar un análisis comparativo entre diferentes ambientes virtuales 3D requiere de la consideración de un conjunto de características comunes y representativas, para su evaluación.

La FAS (*Federation of American Scientist*) ha establecido un conjunto de características para evaluar, desde un punto de vista general, a los entornos virtuales 3D [3] y según se desprende del análisis de diversos trabajos de investigación sobre esta temática [4], [5], estas propuestas varían según el criterio establecido por cada autor.

Sin embargo, la elección de este conjunto de características está fuertemente relacionada con el objetivo que se desea alcanzar a través de la utilización de dichos entornos. En este sentido, el ámbito de utilización propuesto es un ambiente universitario, de carreras de Ingeniería y sus destinatarios son estudiantes que se encuentran promediando su ciclo de estudios. Y desde el punto de vista curricular, el objetivo es simular sistemas dinámicos, con la intención que los estudiantes adquieran un conocimiento más profundo tanto del fenómeno en estudio como de los métodos numéricos involucrados en su resolución.

En este contexto se ha establecido un marco comparativo, a partir de la evaluación del siguiente conjunto de características:

- **Fidelidad de la Representación**

La fidelidad de la representación es indispensable para lograr una experiencia “*espacial*” creíble, es decir, para que el usuario tenga la sensación de estar dentro de ese espacio, interactuando con todos los elementos que allí se encuentran. Las tarjetas de video actuales poseen una gran potencia de procesamiento, lo que les permite manejar representaciones gráficas de gran calidad, texturas y efectos de iluminación con gran velocidad y en forma prácticamente autónoma, lo cual resulta imprescindible para lograr una representación dinámica convincente.

A su vez, los objetos del entorno, así como las representaciones de los usuarios (avatars) necesitan comportarse consistentemente. Es decir, el usuario espera que dentro del entorno se cumplan ciertas “*leyes físicas*”, como por ejemplo, que la intensidad de las fuentes sonoras presentes, varíe de acuerdo a la posición y la distancia que lo separa de las mismas (*sonido espacial*).

- **Navegabilidad e Interacción con el Entorno**

Por navegabilidad e interacción con el entorno, nos referimos a los mecanismos utilizados para recorrer los espacios y para modificar ciertos atributos, que afectan tanto al aspecto visual como a su comportamiento, así como también, la creación y manipulación de los objetos contenidos en el mismo.

Además, la sensación de presencia en primera persona, exclusiva de los ambientes virtuales permite generar nuevos aprendizajes, pues amplía las formas de adquirir conocimiento, en la interacción y transformación de ideas abstractas en representaciones perceptibles a través de objetos virtuales [6].

- **Comunicación entre Usuarios**

La comunicación oral puede lograrse por medio de sistemas de voz a través de Internet, también conocidos como VoIP (*Voice over Internet Protocol*); y la comunicación escrita por medio de herramientas de mensajería instantánea o IRC (*Internet Relay Chat*) integradas en la misma plataforma. Sin embargo, una característica propia de los mundos virtuales es la incorporación de la comunicación gestual, la cual se logra dotando a las representaciones visuales de los usuarios (*avatars*) de un cierto conjunto básico de movimientos o gestos.

- **Modelos 3D y contenidos Multimedia**

Los modelos 3D y los contenidos multimedia son recursos que producen una singular sinergia cuando se incorporan a los mundos virtuales. En el primer caso nos referimos a figuras, vehículos, edificios, etc.; y en el segundo caso, a elementos multimediales tales como fotografías, archivos de audio y video. En todos los casos para poder integrarlos como recursos pedagógicos, se requiere que el entorno tenga la capacidad de cargarlos y reproducirlos adecuadamente.

- **Integración de Aplicaciones 2D**

Para que un entorno virtual 3D no sea una aplicación aislada es conveniente que permita integrar aplicaciones externas dentro del mismo o al menos contar con la posibilidad de ejecutarlas aunque sea de forma remota. Esto permitiría a los estudiantes la utilización de navegadores de Internet, planillas de cálculo, editores de texto y otras aplicaciones 2D, sin necesidad de salir del “*espacio virtual*”.

- **Entorno de Programación**

Los procesos de aprendizaje y experimentación, como las simulaciones, son más efectivos y motivadores si los mismos pueden ser creados colaborativamente y compartidos con otros estudiantes. Para lograr este cometido, es necesario contar con un lenguaje dinámico y un entorno de programación integrado que permita realizar modificaciones en forma interactiva.

4. Análisis de las características

Para disminuir la granularidad del análisis se han considerado tres rasgos distintivos para cada una de las características expuestas en la sección anterior. Las figuras 1 y 2 representan un ejemplo. En cuánto a la evaluación, se ha establecido un rango de valores entre 0 y 5, lo que permite una clara representación de los atributos correspondientes a cada ambiente virtual 3D.

4.1. Fidelidad de la Representación

La Fidelidad de la Representación involucra aspectos tan diversos como la calidad de la representación gráfica de los elementos que componen el ambiente, así como también del comportamiento de los objetos móviles generado por el motor de física. Por último un aspecto no menos importante, son las herramientas que permiten editar y personalizar a la representación del usuario, es decir, de su Avatar.

En *Second Life* los simuladores son los encargados de ejecutar el motor de física (*Physics Engine*), detectar colisiones, y mantener la información de estado de todos los objetos de la región con la finalidad de enviar dicha información al visualizador, quien se encarga de mostrar la escena desde el punto de vista del usuario.

Las simulaciones físicas de los objetos móviles son manejadas a través de la interfaz de programación de aplicaciones (API) del motor de física Havok (*Havok Physics Engine*). Estas simulaciones se encuentran acotadas al ámbito de la mecánica clásica [7]. Es decir, el comportamiento de los objetos es descrito principalmente en función de la influencia de las fuerzas externas aplicadas y de la interacción con otros objetos de la región.

En el caso de *OpenSimulator* no se aprecia a primera vista una gran diferencia con el aspecto que presenta *Second Life*, en lo que se refiere al nivel de detalle y estética gráfica en general, así como en lo referente a la personalización de los *Avatars* y las herramientas que permiten construir objetos a partir de figuras primitivas. Sin embargo, la incorporación de un nuevo motor de física como *Bullet™* mejoró notablemente su nivel de realismo.

A diferencia de los ambientes 3D anteriormente analizados, en *OpenCobalt* los “mundos virtuales” no ocupan regiones rectangulares contiguas, sino que cada uno representa un espacio completamente independiente, incluso aquellos que se encuentran dentro de la misma computadora. *OpenCobalt* cuenta con los elementos básicos para crear espacios virtuales completamente diferentes, de forma muy rápida y sencilla, pero es evidente que en este aspecto, no posee el grado de sofisticación de otros ambientes como *OpenSimulator* o *Second Life*, los cuáles exhiben un nivel estético y de detalle claramente superior.

Por otra parte, si bien en el año 2008 se desarrolló un plugin para ODE (*Open Dynamics Engine*) en *Squeak*, con la intención de incorporar un motor de física en *OpenCobalt*, el proyecto fue abandonado, fundamentalmente por la imposibilidad de integrarlo junto con el esquema descentralizado de comunicaciones de *OpenCobalt*.

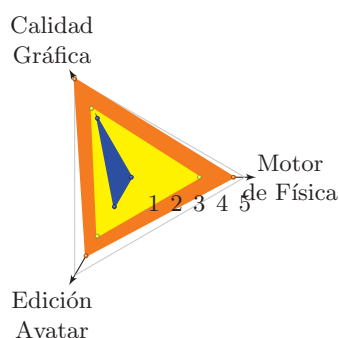


Fig. 1. Fidelidad de la Representación

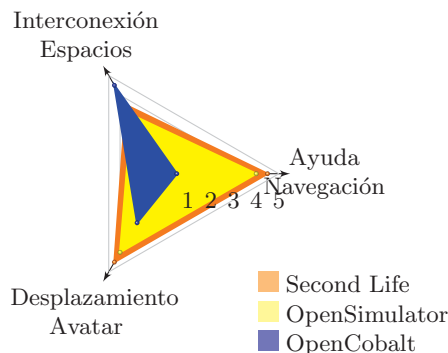


Fig. 2. Interacción con el Entorno

4.2. Navegabilidad e Interacción con el Entorno

En el esquema Cliente-Servidor que posee *Second Life*, el programa Cliente provisto por *Linden Labs*, denominado *SL Viewer*, es el que le permite al usuario conectarse con el Servidor y navegar por las diferentes regiones del entorno. En 2007, *Linden Labs* liberó gran parte del *código fuente* de este programa cliente, lo que dió lugar a que otros desarrolladores crearan sus propias versiones, como *Firestorm*, *Hippo*, *Imprudence*, *Singularity*, etc.

La tarea principal de los visualizadores es realizar las operaciones gráficas (*rendering*) de la escena. Los visualizadores se comunican con el servidor para obtener información sobre la ubicación y velocidad de los objetos que se encuentran en la región, aunque no son capaces de detectar colisiones entre ellos, ya que esta tarea es responsabilidad del motor de física y del simulador.

En *Second Life* el Avatar del usuario puede caminar, correr e incluso volar, aunque una de las formas más elementales de desplazarse por una región, es utilizando las teclas de desplazamiento *arriba*, *abajo*, *izquierda* y *derecha*. Sin embargo, para realizar desplazamientos con cierta precisión, lo usual es pasar al modo *Mouselook*, por medio de la rueda del mouse, en este modo la cámara muestra una vista en primera persona y el *Avatar* se desplazará hacia donde apunte el cursor del mouse.

La navegación dentro de una región de *OpenSimulator* es en apariencia, muy similar a la navegación en *Second Life*. Sin embargo, debido a que *OpenSimulator* puede ejecutarse en diferentes modos: *Standalone*, *Grid* y *Hypergrid*, se presentan algunas diferencias.

Una vez instalado el servidor de *OpenSimulator*, el mismo se ejecutará por defecto en el modo *Standalone*, es decir, ejecutando un mundo virtual completamente aislado. Sin embargo, si se ejecuta en modo *Hypergrid*, se puede vincular esa región con otros espacios virtuales, y una vez establecidos los enlaces, será posible *teleportarse*, simplemente utilizando el hipervínculo establecido, aunque con la ventaja de conservar los contenidos y objetos 3D propios, dentro de la misma computadora.

En el caso de *OpenCobalt* la navegación dentro de un espacio virtual 3D, se realiza en forma similar a los *juegos en primera persona*, en los que el usuario, representado por su *Avatar*, se desplaza utilizando ciertas teclas para avanzar, retroceder o girar o haciendo uso de los *botones de navegación* que se encuentran en la parte inferior derecha de la pantalla.

Por otra parte, como los espacios virtuales en *OpenCobalt* no ocupan regiones contiguas, para pasar de un espacio a otro, es necesario utilizar un *portal*. Esto incorpora un concepto interesante, ya que si bien cada espacio posee un tamaño determinado y por lo tanto es posible recorrer una cierta distancia dentro del mismo, cuando se atraviesa un portal, la distancia entre ambos espacios virtuales es prácticamente nula. En este sentido, un *portal*, no es simplemente una ventana que muestra, en tiempo real, lo que está sucediendo en el otro espacio, sino que al atravesarlo, se ingresa instantáneamente en el otro espacio. Por este motivo, como es posible tener varios portales abiertos simultáneamente, y cualquier usuario puede pasar de un espacio virtual a otro, simplemente atravesando el portal correspondiente; es probable que luego de atravesar varios portales se experimente una cierta sensación de desorientación, ya que en *OpenCobalt* no existen mapas de navegación que le indiquen al usuario donde se encuentra el portal que pueda conducirlo a su espacio de origen.

4.3. Comunicación entre Usuarios

Second Life posee diversos mecanismos de comunicación. La comunicación escrita se efectúa por medio de *Text Chat* e *Instant Messaging*, la oral por *VoIP* y por último la gestual. Debido a que *Second Life* fue concebido fundamentalmente como un lugar de encuentro y participación social, los mecanismos de comunicación juegan un papel preponderante. La comunicación a través de *VoIP* (Voice over Internet Protocol) no sólo permite la comunicación entre usuarios sino que es una herramienta clave para la realización de diversos eventos, desde foros y congresos hasta representaciones teatrales.

Por otra parte, la comunicación gestual es un aspecto muy interesante de los ambientes virtuales, ya que en el mundo real, este tipo de comunicación es tan o más importante que la comunicación oral o escrita. En este sentido *Second Life* cuenta con un menú mediante el cual se puede elegir el gesto más apropiado para dar a entender a los demás usuarios cuáles son nuestras intenciones. El repertorio de gestos es variado, y si se desea, se pueden crear nuevos gestos y agregarlos al menú.

Los mecanismos de comunicación oral y textual de *OpenSimulator* son prácticamente idénticos a los de *Second Life*. Y si bien, los usuarios utilizan principalmente el Chat y los mensajes instantáneos, la comunicación oral, a través de *VoIP*, junto con los gestos personalizables y sonidos asociados, configuran un medio de comunicación adicional, que reafirma la sensación de presencia.

OpenCobalt proporciona dos opciones de comunicación textual. Una local, para establecer comunicaciones dentro del mismo espacio, y otra externa, por medio del protocolo XMPP (*eXtensible Messaging and Presence Protocol*). Por otra parte, para la comunicación oral entre usuarios, utiliza *VoIP* (Voice over

Internet Protocol), aprovechando además, las ventajas del sonido espacial que provee *OpenAL*, una biblioteca de funciones de audio multiplataforma, desarrolladas por *Creative Labs*TM para la ejecución de audio posicional y multicanal en tres dimensiones.

4.4. Modelos 3D y contenidos Multimedia

En *Second Life* es posible generar todo tipo de objetos 3D a partir de figuras geométricas sencillas denominadas *prims*, las cuales pueden ser agrupadas para generar figuras más complejas. Además de los *prims*, existen otras estructuras denominadas *sculpted prims* o *sculpties*, y los modelos 3D de malla (*Mesh*).

Los mecanismos de edición de objetos en *OpenSimulator* a partir de figuras primitivas (*prims*), también son muy similares a los existentes en *Second Life*, ya que esta funcionalidad es aportada mayormente por el visualizador, que es precisamente el elemento común en ambos entornos. En cuanto a la importación de modelos, *OpenSimulator* permite importar modelos de malla (*Mesh*) en formato *.dae* (*Digital Asset Exchange*) y en referencia a otros contenidos multimedia *OpenSimulator* reproduce los formatos de audio o video más comunes como *.wav* y *.mpeg*.

OpenCobalt soporta la importación de modelos 3D en diversos formatos estándar, tales como *.ase*, *.kmz*, *.obj* y *.vrmf*, los cuales pueden crearse previamente con programas externos específicos como *Blender* o *Trimble SketchUp*. También pueden incorporarse modelos completos obtenidos de sitios como *Google 3D Warehouse*, sin embargo, la herramienta *3D EditBox* integrada en *OpenCobalt*, sólo es capaz de trasladar, escalar y rotar los modelos.

4.5. Integración de Aplicaciones 2D

A partir de la nueva versión del programa cliente de *Second Life* (Viewer 2.0), está disponible una nueva característica denominada *Shared Media*, mediante la cual es posible visualizar la información de un navegador web o un video de *Youtube*, simplemente arrastrando y soltando la correspondiente URL (Uniform Resource Locator) sobre la superficie de una figura primitiva (*prim*), accediendo de esta forma a contenidos de streaming de audio y video.

Una forma sencilla de integrar aplicaciones 2D en *OpenSimulator* es por medio de una pequeña aplicación, gratuita y de código abierto, denominada *Gua-camole*, que se instala junto con el servidor VNC (*Virtual Network Computing*). Esta aplicación se comunica con el servidor VNC y realiza el rendering de su salida en formato HTML5, lo que permite que los usuarios utilicen aplicaciones de escritorio dentro del ambiente, a través del visualizador de *OpenSimulator*.

En *OpenCobalt* es posible ejecutar colaborativamente aplicaciones 2D nativas de *Squeak* junto con aplicaciones 2D mono-usuario standard como navegadores web, procesadores de texto, planillas de cálculo, etc., dentro de un entorno multi-usuario colaborativo, por medio del cliente VNC que tiene incorporado.

4.6. Entorno de Programación

En cuanto a las facilidades de programación, *Second Life* posee un lenguaje de programación basado en eventos y estados, denominado **LSL** (*Linden Scripting Language*). Este lenguaje, que posee una sintaxis similar al lenguaje C, permite responder a una serie de eventos. Los eventos pueden ser generados por el mismo sistema, como los timers, o por otros usuarios, como los eventos de presencia, de chat o email, y también pueden ser producidos otros objetos 3D, por ejemplo, en el caso que se produzca una colisión.

La interfaz de este lenguaje de *scripting* proporciona más de 30 tipos de eventos y más de 400 funciones, a partir de las cuales se escriben pequeños fragmentos de código denominados *scripts*, los cuáles son interpretados y ejecutados por el servidor. En este sentido, el modelo de programación consiste en que cada objeto contiene sus propios scripts, que son los encargados de establecer el comportamiento del objeto.

Una de las principales críticas que se ha hecho a este modelo es que al no existir el concepto de clase, ni de objeto, como se entiende dentro del paradigma de programación orientada objetos, cada elemento es un prototipo que responde reactivamente a ciertos *eventos*. Por lo tanto, sus limitaciones son evidentes, ya que no existen bibliotecas de scripts que permitan su reuso, ni control de versiones, y resulta muy difícil desacoplar a los scripts de los objetos 3D. [8]

Los scripts en *OpenSimulator* están basados en **LSL** (*Linden Scripting Language*) con la intención de mantener cierto grado de compatibilidad. Sin embargo, no todas las funciones presentes en *Second Life* han sido implementadas, y por otro lado, se han incorporado algunas funciones nuevas.

El hecho que *OpenCobalt* esté íntegramente programado en *Squeak* le permite al usuario acceder todas las herramientas de programación y depuración propias de dicho lenguaje desde el mismo ambiente. *Squeak*, es un lenguaje de programación orientado a objetos, basado en clases, con capacidades de programación reflexiva y metaprogramación, que se ejecuta sobre una *Máquina Virtual* específica para cada plataforma, lo que hace que el código desarrollado sea totalmente portable inclusive entre diferentes sistemas operativos.

4.7. Resultados obtenidos

Luego de la aplicación del marco comparativo a los ambientes virtuales 3D seleccionados, referidas a cada rasgo en particular, se integraron los resultados de forma tal de obtener una representación gráfica comparativa (Spider Chart) de las seis características evaluadas, como se aprecia en la Figura 3.

5. Conclusiones

A partir de los resultados obtenidos puede afirmarse que *Second Life* se destaca por los aspectos estéticos y de comunicación entre usuarios. *OpenSimulator* aventaja al anterior por ser una aplicación de código abierto y brindar a los usuarios la posibilidad de instalar sus propios servidores. Finalmente *OpenCobalt* a

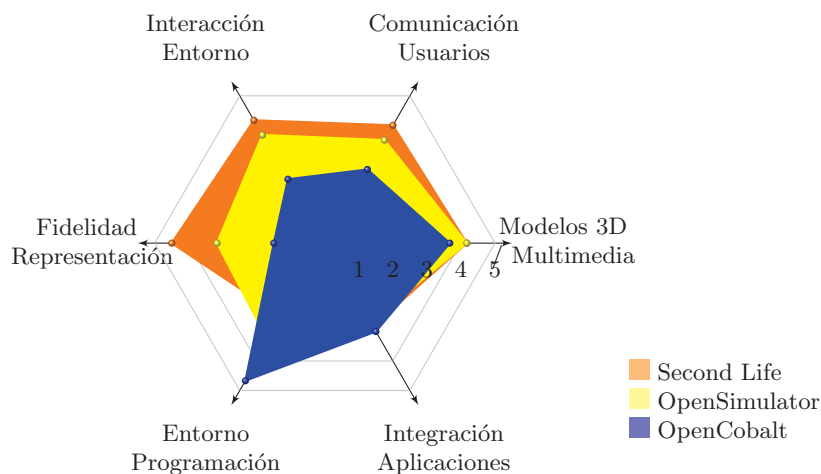


Fig. 3. Integración de las características analizadas

pesar de ser más limitado en el aspecto estético, supera a los anteriores fundamentalmente en lo que se refiere al lenguaje y las herramientas de programación, un aspecto clave para la implementación de simulaciones.

Referencias

1. Quinche, Juan, González, Franci. Entornos Virtuales 3D, Alternativa Pedagógica para el Fomento del Aprendizaje Colaborativo y Gestión del Conocimiento en Uniminuto. *Revista Formación Universitaria*. 4, pp.45-54. (2011)
2. Black, Andrew, Ducasse, Stéphane, Nierstrasz, Oscar, Pollet, Damien, Cassou, Damien, Denker, Markus : Squeak by Example. Square Bracket Associates, Switzerland (2009)
3. De Freitas, Sara: Serious Virtual Worlds. A scoping study. *JISC Virtual Worlds & Serious Gaming Resources*, pp.4-49 (2008)
4. Blas, Nicoletta, Garzotto, Franca, Poggi, Caterina. Web Engineering at the Frontier of the Web 2.0: Design Patterns for Online 3D Shared Spaces. *World Wide Web*. 4. 12. pp.345-379. (2009)
5. Tsiatsos, Thrasyvoulos, Konstantinidis, Andreas, Pomportsis, Andreas S. Evaluation Framework for Collaborative Educational Virtual Environments. *Educational Technology & Society*. 2. 13. pp.65-77. (2013)
6. Fredes, Claudio, Hernández, Juan, Díaz, Daniel. Potencial y Problemas de la Simulación en Ambientes Virtuales para el Aprendizaje. *Revista Formación Universitaria*. 1. 5. pp.45-56. (2012)
7. Dos Santos, Renato: Second Life as a Platform for Physics Simulations and Microworlds: An Evaluation. *Proceedings of the CBLIS 2012*, pp.173-180, Centre for Research in Science and Mathematics, Barcelona (2012)
8. Lopes, Cristina: The Worst Language Ever Designed : The Case for Better Programming Languages for 3D Environments. In: *POPL 2014: 41st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, California (2014)