

# Reconstrucción virtual del mito Emberá de la creación del agua utilizando agentes autónomos



## Andrés Quiroz

Ingeniero de Sistemas, Universidad Eafit. Actualmente estudiante de posgrado en Ingeniería de la Computación en Rutgers (EEUU). En el momento de escritura de éste artículo era Joven Investigador de Colciencias adjunto al Laboratorio de Realidad Virtual, Universidad Eafit. [aquirozh@eafit.edu.co](mailto:aquirozh@eafit.edu.co)

## Santiago Vélez

Estudiante de Ingeniería de Sistemas, auxiliar de investigación en Laboratorio de Realidad Virtual, Universidad Eafit.

[svelezgu@eafit.edu.co](mailto:svelezgu@eafit.edu.co)

## Juliana Restrepo

Magíster en Educación de la Universidad Javeriana. Licenciada en Educación Preescolar de Universidad Lasallista. Co-investigadora y asesora pedagógica del laboratorio de Realidad Virtual, Universidad Eafit. [jrestre@eafit.edu.co](mailto:jrestre@eafit.edu.co)

## Helmuth Trefftz

Doctor en Ingeniería de la Computación, Universidad de Rutgers (EEUU). Maestría en Ciencias de la Computación, Maharishi University of Management (EEUU). Ingeniero de Sistemas, Universidad Eafit. Director del Laboratorio de Realidad Virtual, Universidad Eafit, categoría C Colciencias. Miembro IEEE y ACM.

[htrefftz@eafit.edu.co](mailto:htrefftz@eafit.edu.co)

## Juan David Velásquez

Doctor en Ciencias de la Computación, M.I.T. (EEUU). Ingeniero de Sistemas, Universidad Eafit. Gerente y fundador de la empresa Factoring Market. Miembro IEEE y ACM. Evaluador proyectos de Colciencias.

[juan.velasquez@factoringmarket.com](mailto:juan.velasquez@factoringmarket.com)

Recepción: 29 de julio de 2005 | Aceptación 5 de diciembre de 2005

## Resumen

Este artículo describe un Entorno Virtual Proyectado, en el cual se reconstruye el mito Emberá de la creación del agua. Desde el punto de vista técnico, un objetivo del proyecto es lograr que la interacción con los usuarios (niños en edad preescolar), se realice por medio de voz y *joystick*. Explícitamente se evita el uso de texto escrito, *mouse* y teclado. Desde el punto de vista pedagógico, el objetivo del proyecto es lograr que los niños aprendan algunos elementos puntuales de la cultura Emberá, mientras buscan ganar el juego implementado. Se presentan también los resultados del experimento realizado con 25 niños de edad preescolar, en el cual se compara el aprendizaje logrado por medio del uso del programa, con el de una clase magistral tradicional.

## Palabras Clave

Realidad Virtual  
Agentes Autónomos  
Educación  
Evaluación de Sistemas de  
Enseñanza

## Virtual reconstruction of the Emberá myth of the creation of water using autonomous agents

### Abstract

This article describes a Projected Virtual Environment, in which the Emberá myth of the creation of water is reconstructed. From the technical point of view, an objective of the project is to achieve that the interaction with the users (preschool children), is done through voice and a joystick. It is explicitly forbidden to use written texts, mouse, and the keyboard. From the teaching point of view, the objective of the project is to achieve that the kids learn some main elements of the Emberá culture while they desire winning the implemented game. The article also shows the results of the experiment carried out with 25 preschool kids, which compares the learning reached with the use of the program and the one in a traditional class.

### Key words

Virtual reality  
Autonomous agents  
Education  
Teaching System Evaluation

### Introducción



Los siguientes párrafos describen los avances técnicos que posibilitan la construcción de un sistema como el descrito en este artículo.

Los grandes desarrollos en dispositivos novedosos de interacción hombre-máquina permiten la creación de sistemas con interfaces “multimodales”, esto es, programas con los cuales el usuario interactúa utilizando diferentes canales de comunicación (texto, voz, gestos, etc.). Un objetivo que se persigue con los sistemas multimodales es lograr una interacción más natural con el computador. Por otro lado, se busca también posibilitar una interacción efectiva por parte de usuarios que no pueden utilizar interfaces tradicionales basados en *mouse* y teclado, bien sea por limitaciones físicas (usuarios ciegos, cuadripléjicos, etc.) o limitaciones cognitivas (niños en edad preescolar).

Los programas de Realidad Virtual permiten que el usuario se mueva dentro del entorno virtual e interactúe con los objetos del mismo, utilizando dispositivos como guantes, “joystick” o similares. Los ambientes virtuales proyectados han ganado popularidad frente a los cascos de RV, ya que permiten experiencias grupales y reducen problemas ergonómicos. Sin embargo, no son muchos los sistemas de Realidad Virtual que utilizan reconocimiento o síntesis de voz.

Los avances en Inteligencia Artificial han permitido crear caracteres virtuales autónomos que interactúan con los usuarios exhibiendo algún grado de inteligencia. Esta interacción se puede dar con diversos fines, tales como entretenimiento, entrenamiento o educación.

Integrando estos elementos tecnológicos, hemos creado un entorno de realidad virtual proyectado. Los usuarios son niños en edad preescolar que se mueven dentro del entorno virtual utilizando un “joystick”. En ciertos puntos del mundo virtual se encuentran con agentes virtuales autónomos. Los agentes autónomos detectan la cercanía del usuario virtual y responden con gestos, preguntas y sugerencias. El niño responde a las preguntas utilizando su voz y el agente responde utilizando respuestas de voz pregrabadas, bien sea dando refuerzo positivo o negativo a la respuesta del niño.

Desde el punto de vista pedagógico, el objetivo del trabajo descrito es validar la utilidad pedagógica del entorno virtual creado, comparándolo con una clase tradicional. Se trata de medir tanto aprendizaje como cambios actitudinales (así sean leves) hacia las culturas indígenas, por parte de los niños en edad preescolar.

El resto de este artículo está estructurado de la siguiente manera: en la sección 1 se describen trabajos con alguna similaridad al propuesto. En la

sección 2 se presenta el entorno virtual construido. En la sección 3 se describe el experimento llevado a cabo para validar la eficiencia pedagógica del entorno virtual. Los resultados son descritos en la sección 4. Finalmente, se presentan las conclusiones y el trabajo futuro.

## 1. Trabajo Relacionado

Esta sección describe los trabajos que constituyen la base para el proyecto. Inicialmente se describen los trabajos relacionados con el módulo de Inteligencia Artificial, luego los relacionados con la interacción multimodal y, finalmente, los relacionados con el motor gráfico.

### 1.1 Agentes Inteligentes

En un campo tan amplio como el de la inteligencia artificial, es importante aclarar el alcance dentro del cual se ha considerado su aplicación en el proyecto, para poder enfocar la presentación del trabajo relacionado. La idea del proyecto es contar una historia con la ayuda de personajes virtuales, logrando una inmersión del participante a través de su interacción con estos personajes y con el mundo del que hacen parte. Dichos personajes deben responder entonces a estímulos originados en las acciones del participante, las cuales no siempre se dan de la misma manera ni deben, en diferentes puntos de la historia o ante diferentes secuencias de acciones anteriores, producir el mismo resultado. Es a partir de esta concepción que consideramos que los personajes de la aldea son agentes inteligentes y, particularmente, consideramos que estos agentes juegan un papel de guías en la interacción del participante con el mundo virtual y la comprensión de la historia que se quiere contar.

Esta utilización de agentes inteligentes se ha realizado en varios tipos de ambientes virtuales. En (Isbister y Doyle,

1999) se describen dos proyectos de este tipo. Uno es el proyecto Digital City, el cual es un conjunto de recursos web de Kyoto, Japón, incluyendo recorridos 3D de diferentes sitios de la ciudad. El proyecto incluye la opción, para grupos de navegantes, de realizar recorridos del sitio, guiados por un agente virtual que ofrece datos y referencias de los lugares visitados. En este caso, el agente adapta la cantidad de información proporcionada y la forma de contarla según el interés y naturaleza del grupo (a partir del grupo de edades, preguntas, discusión, etc.). Similarmente, un agente inteligente sirve de guía en un prototipo de un museo de arte basado en Internet. Este agente proporciona información de las obras, espontáneamente o a petición, a medida que el usuario recorre el museo, obteniendo la información directamente de anotaciones contenidas en el sitio.

El enfoque del proyecto de la aldea puede compararse con los videojuegos de tipo aventura, en los que un personaje acompaña al usuario y le proporciona información sobre los lugares visitados y pistas útiles sobre las acciones que puede realizar; además, incluye otros personajes con los que el usuario puede hablar, y que a su vez pueden proporcionar pistas o servir de intermediarios para resolver el objetivo del juego. Zelda y King's Quest son ejemplos clásicos de este tipo de juego. Por supuesto, el nivel de inteligencia de los personajes en estos juegos es muy bajo. En (Doyle, 2002) se describe un prototipo de juego de

este estilo construido sobre una arquitectura de agentes que permite mucha más sofisticación en su comportamiento.

Precisamente, una preocupación importante en la realización de este proyecto fue el nivel de inteligencia que se le podría dar a los agentes desarrollados. Existen muchos criterios diferentes para medir la inteligencia de

un agente, o bien, para establecer si el comportamiento observado de un agente



es inteligente o no<sup>1</sup>. Basados en Doyle (2002), consideramos que la credibilidad (*believability*) debe ser el criterio más importante para evaluar la inteligencia de nuestros agentes y, por ende, el objetivo del diseño realizado para su desarrollo. El enfoque de este criterio está en la forma y cualidad de los comportamientos realizados por los agentes, en tanto el participante puede llegar a sentir que está realmente interactuando con un personaje en vez de con un programa de computador. Esto implica reducir la predecibilidad y repetitividad de estos comportamientos, permitir la expresión de emoción y, en general, dar la impresión de una personalidad (no necesariamente humana o perfectamente racional). Esto es importante porque contribuye en gran medida a la inmersión del usuario en el ambiente virtual. Además de Doyle (2002), que se basa en la interpretación de anotaciones (indicaciones o pistas) en el ambiente por parte del agente, según su propia programación; existen otras técnicas para lograr uno o varios de estos aspectos: Velásquez (1999), Kline and Blumberg (1999), Bates (1994) y Elliot (1998) exploran el manejo de emociones (o, más bien, respuestas emocionales) de los agentes, mientras que Blumberg (1994) ejemplifica una rama de desarrollo, llamada etología, que busca emular los principios que rigen el comportamiento animal.

El criterio de credibilidad contrasta con el criterio de la efectividad del agente, en cuanto a la consecución de unos objetivos establecidos, cuyo enfoque está en el manejo de la variabilidad e impredecibilidad del entorno. La variedad e impredecibilidad de las acciones del agente no es tan importante en este caso, con tal de que estas acciones sean las correctas para el caso dado o produzcan el resultado requerido (Doyle, 2002). Sin embargo, no se puede descartar la importancia de la efectividad en el desarrollo de los agentes de la aldea, y de los agentes guía en general, ya que el objetivo final es contar la historia y transmitir todos los elementos conceptuales que se han establecido.

Otros criterios de inteligencia, de menor importancia relativa para este proyecto, incluyen: la autonomía,

<sup>1</sup> La inteligencia es un concepto tan difícil de definir y tan poco comprendido que en este tipo de sistemas requiere una sustancial simplificación.

que implica la capacidad del agente de funcionar sin dirección o intromisión del usuario, y la adaptabilidad del agente, que incluye la capacidad de funcionar bien en diferentes entornos o en un entorno cambiante, y la capacidad de aprender y mejorar su comportamiento con el tiempo (Maes, 1994).

## 1.2 Interfaces Multimodales

Buscando brindar a los usuarios de este sistema una experiencia inmersiva y una interacción natural con sus objetos y personajes, uno de los objetivos del proyecto es explorar la implementación de interfaces multimodales, es decir, la integración de varios modos de interacción hombre-máquina, como el reconocimiento de voz, rastreo corporal (para el reconocimiento de movimientos o gestos, como guantes y rastreadores de mirada) y la utilización de otros dispositivos de entrada más tradicionales como *joystick* o *mouse*. Este tema también comprende los modos de mostrar información al usuario, como la síntesis de voz, información textual o táctil. En la interacción entre seres humanos, la comunicación oral se ve complementada por información visual y por la interpretación del contexto, de tal forma que expresiones ambiguas (llamadas anáforas y expresiones deícticas) presentes en el lenguaje natural (p. ej. “mueve eso ahí”), pueden ser interpretadas correctamente. El trabajo en el campo de las interfaces multimodales está dirigido a lograr que los diferentes modos disponibles para la interacción con la máquina (o, en este caso, con un personaje emulado por la máquina), se complementen entre sí para posibilitar una interacción comparable con la interacción entre humanos.

Este es un campo de investigación bastante activo. Actualmente existen plataformas de *software* que se pueden utilizar para la gestión de interfaces multimodales, como Galaxy Communicator de MIT (Seneff et al., 1998) y Open Agent Architecture de SRI Internacional (Martin et al., 1999), aunque ninguna de ellas es un sistema multimodal completo, sino que proporcionan comunicación y coordinación entre los módulos del sistema (Flippo, 2003). Los sistemas que se han desarrollado para utilizar interfaces multimodales son, en general, dependientes de su dominio, aunque muchos de ellos comparten

técnicas de diseño. Algunos ejemplos se encuentran en Bolt (1980), Cohen *et al.*, 1997 y Toledano *et al.*, 2003). Las tareas básicas de un sistema multimodal son la *fusión* (combinación de información de los diferentes dispositivos de interacción y resolución de ambigüedades) y la *filtración* (selección de los modos adecuados para presentar la información al usuario en tiempo de ejecución) (Flippo, 2003).

Antes de lograr este objetivo de integración es necesario considerar por separado la implementación de cada uno de los modos de interacción y su aplicación adecuada, según las tareas que el usuario debe realizar en el ambiente. El complemento mutuo de los dispositivos de interacción en un sistema multimodal no sólo tiene que ser en función de la resolución de ambigüedades, sino que también se puede realizar al hacer disponible al usuario la mejor manera de realizar una acción para cada una de las tareas que se requieren. En un ambiente virtual como la aldea, por ejemplo, la mejor manera de interactuar con los personajes es a través del habla, el movimiento dentro del mundo se realiza perfectamente por medio de un *joystick* y la interacción con objetos por medio de gestos manuales capturados por un rastreador de posición.

### 1.3 Motor gráfico

La utilización de motores gráficos permite llegar a un nivel de abstracción en el que los programadores pueden dejar de preocuparse por los detalles técnicos de la implementación y enfocarse en la creación del entorno virtual. Estos motores pueden incorporar diferentes funcionalidades, además del manejo gráfico de la aplicación, como el manejo de audio y de dispositivos de entrada, convirtiéndose en una herramienta que facilita la creación de mundos virtuales interactivos.

Generalmente, la creación de un motor requiere de mucho tiempo, por lo cual la mayoría de las empresas que programan juegos y aplicaciones gráficas suelen utilizar algún motor disponible en el mercado en vez de crear el propio.

Se puede encontrar gran cantidad de motores con diferentes características, existen motores libres creados por comunidades de programadores en

Internet y motores comerciales que pueden costar miles de dólares. Hay motores que sólo se encargan de manejar la parte gráfica de las aplicaciones y hay otros que incorporan mucha más funcionalidad, como el manejo de audio o el de los dispositivos de entrada.

La idea detrás de la creación de un motor es generar un conjunto de herramientas que permitan a los programadores centrarse en la interacción que tendrá el usuario con la aplicación, en vez de tener que lidiar con todos los detalles técnicos que esto acarrea. Por esto, es común que dichos motores se centren en algunas funcionalidades predefinidas y, por tanto, no todos los motores sirven para crear cualquier tipo de aplicación o de juego; algunos motores se centran en juegos en primera persona, mientras que otros están diseñados para crear aplicaciones CAD y otros para la programación de juegos de estrategia en tercera persona. Por esto se hace necesaria la evaluación de varios motores antes de decidir cuáles pueden ser útiles para la creación de una aplicación en específico.

En la página de Internet de *Devmaster 3d Engines* se encuentran descripciones de gran cantidad de motores tanto comerciales como libres, allí se pueden ver las diferentes capacidades de los motores, los sistemas operativos en que trabajan y los lenguajes en los cuales pueden ser utilizados.

Tres motores fueron evaluados para ser usados en el proyecto: OGRE (Ogre, 2004), Irrlicht (Irrlicht, 2004) y The Nebula Device 2 (Nebula Device 2, 2004). Estos fueron preseleccionados entre muchos otros por ser libres, ser escritos en C++ y por presentar las más altas calificaciones en varios listados de motores en Internet.

Irrlicht es un motor de alto rendimiento, multi-plataforma escrito en C++. Es un API de alto nivel para crear aplicaciones como juegos o visualizaciones científicas en 2D o 3D. Viene con documentación e integra tecnologías como sombras dinámicas, sistema de partículas, animación de personajes, tecnología de espacios interiores y exteriores y detección de colisiones. Todo esto es accesible a través de una bien diseñada interfaz en C++, que es fácil de utilizar.

OGRE (Object-Oriented Graphics Rendering Engine) es un motor 3D orientado a escenas muy flexible. Está escrito en C++ y diseñado para hacer más fácil e intuitiva la producción de juegos y demos utilizando *hardware* 3D. La librería de clases abstrae todos los detalles para utilizar librerías como Direct3D y OpenGL, y provee una interfaz basada en objetos que existen en un mundo y otras clases intuitivas.

The Nebula Device 2 es un motor libre para la creación de juegos o visualizaciones 3D en tiempo real, escrito en C++. La versión 2 es un moderno motor de renderización que utiliza a fondo la tecnología de *shaders*. Permite la utilización de *scripts* a través de TCL/TK y Lua, con soporte para Python, Java y toda la gama de lenguajes .NET (en desarrollo). Actualmente soporta Directx 9, con soporte para OpenGL en desarrollo. Corre en Windows, y están siendo creadas versiones para Linux y MacOSX.

La decisión de utilizar The Nebula Device 2 se basó principalmente en el soporte que ofrece este motor para *scripting* y para la utilización de *shaders* ya que ambas tecnologías permiten acelerar el proceso de desarrollo del proyecto. Además este motor, aunque es libre, ha sido utilizado para la creación de varios juegos comerciales en Alemania. A diferencia de los otros, Nebula2 no es un proyecto independiente, pues fue creado por la empresa RadonLabs GmbH (RadonLabs, 2004) en Alemania, la cual distribuye juegos a nivel comercial, entre ellos “Project Nomads”, el cual utiliza Nebula2 como motor en su programación.

## 2. Nuestra propuesta

El sistema que se creó para esta investigación consiste en un sistema de realidad virtual proyectada con interfaces multimodales. El despliegue visual se realiza sobre una pantalla de proyección trasera cuyas dimensiones son 2 metros de ancho por 1.5 metros e altura. En la proyección trasera, los participantes del mundo virtual están situados en la parte frontal de la pantalla y el proyector se sitúa en la parte trasera de la misma. Esto permite que los usuarios no proyecten sombras sobre la pantalla. Si bien la pantalla permite proyección estereoscópica, esta característica no se utilizó. Como entrada al sistema

se utiliza el *joystick*, para moverse dentro del mundo virtual; y ciertos comandos de voz. Las salidas de la aplicación consisten en (i) el despliegue visual del mundo y (ii) sonido, utilizando frases pregrabadas que la aplicación escoge dependiendo del contexto y de la interacción con el participante. En la Figura 4 se muestran dos niños interactuando con la aplicación. Uno de ellos dirige el movimiento del avatar utilizando el *joystick* y el otro utiliza el micrófono para interactuar con los agentes sintéticos.

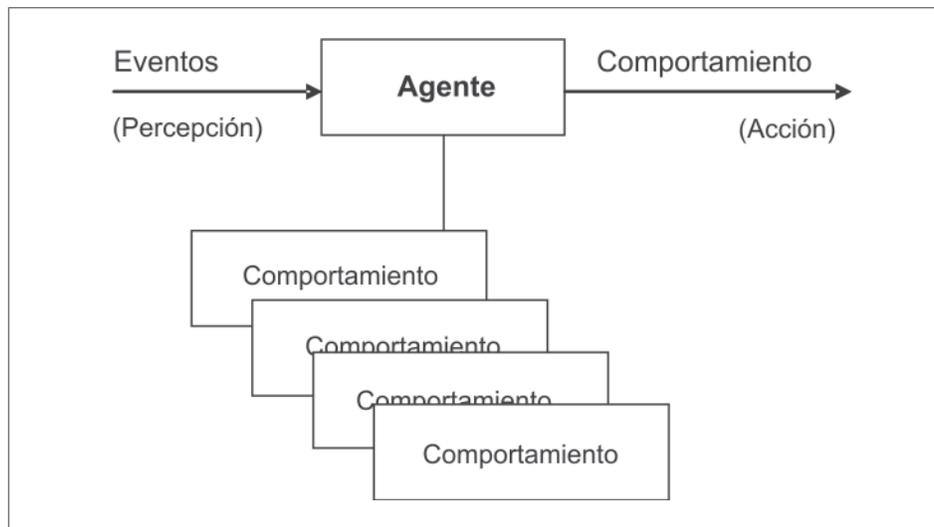
En esta sección se describe cómo se implementaron los agentes autónomos de la aplicación, la integración de las múltiples modalidades y cómo se integraron los diferentes subsistemas utilizando Nebula. Se presenta también un resumen del juego creado.

### 2.1 Agentes Autónomos

La arquitectura desarrollada para el presente proyecto está basada en la respuesta asincrónica a eventos en forma de comportamientos independientes, controlados por el criterio de los agentes a los que pertenecen. El modelo utilizado aplica varias de las técnicas expuestas en Doyle (2002), aunque se le quiso dar un carácter más general. Como puede verse en la Figura 1, el eje del modelo es el programa agente, capaz de recibir y procesar una cantidad de tipos de eventos específicos en el momento en el que son generados y luego escoger entre los comportamientos a su disposición uno o varios que sean una respuesta apropiada (o no inapropiada) a los eventos recibidos.

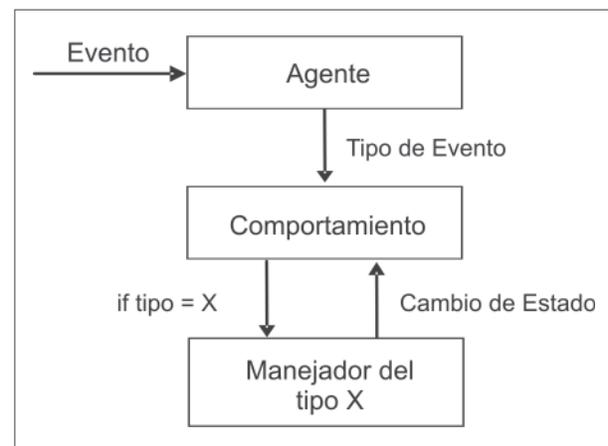
Aunque el control de acción es realizado por el programa agente, desde el punto de vista de programación los módulos de comportamiento contienen un componente de decisión fundamental para el funcionamiento de la arquitectura. Con el fin de favorecer la sencillez y extensibilidad del modelo, cada módulo de comportamiento implementa un mecanismo de procesamiento de eventos, de tal forma que el mismo comportamiento tome la decisión de cuándo puede realizarse. Esto simplifica el modelo, porque cada módulo es responsable de su propio funcionamiento, sin depender de otros, y lo hace extensible porque evita tener que añadir y/o modificar extensamente el código del agente cada vez que se añade un nuevo comportamiento.

**Figura 1.** El agente tiene a su disposición una colección de comportamientos, de los cuales escoge un subconjunto ante la recepción de eventos específicos



El mecanismo de decisión desde el punto de vista del comportamiento, que se detalla en la Figura 2, es el siguiente. Primero, el comportamiento, que se ha matriculado a un agente, recibe de éste un evento que puede o no estar relacionado con él. Esta relación se determina mediante una lista de manejadores de eventos que se debe mantener para cada comportamiento. Cada manejador de eventos corresponde a un tipo específico de evento, de tal forma que un comportamiento sólo procesa un evento si existe en su lista un manejador de eventos para ese tipo particular. El procesamiento realizado por el manejador de eventos es totalmente dependiente del comportamiento particular (su implementación es una clase abstracta que el programador implementa según sus necesidades), pero el fin principal de los manejadores es determinar la activación o desactivación del comportamiento (aunque, como se verá enseguida, este no siempre es el caso). Un comportamiento activo se convierte entonces en candidato para ser ejecutado por el programa de selección del agente, que se debe preocupar entonces por la sincronización entre los comportamientos y la regulación de cada uno de ellos, según reglas más generales de concurrencia (cuáles comportamientos pueden o no ejecutarse al mismo tiempo) o de duración, frecuencia, aleatoriedad, etc. Un comportamiento inactivo no puede (o debe) ser ejecutado por el agente.

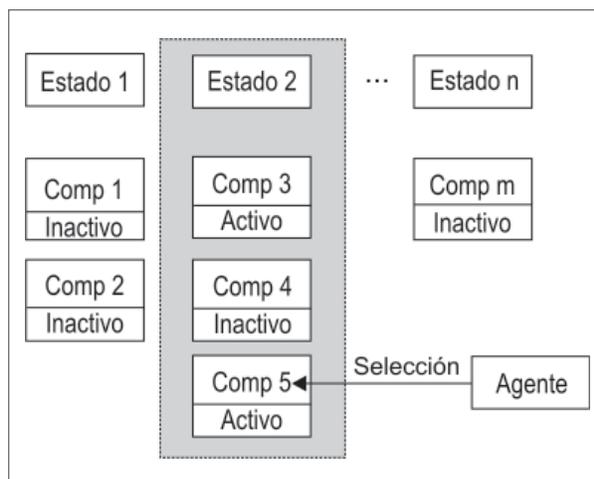
**Figura 2.** Caso específico para uno de los manejadores de uno de los comportamientos de un agente. El cambio de estado del comportamiento puede ser de activación/desactivación, de tal manera que luego el agente escoge entre los comportamientos activos



Como se mencionó anteriormente, aunque el propósito principal de los manejadores de eventos es controlar la activación y desactivación de los comportamientos, también pueden utilizarse para gestionar el contexto. La información de contexto es de vital importancia para el adecuado comportamiento de cualquier agente, ya que sirve de parámetro para modificar las características de ejecución de los comportamientos, aún ante los mismos eventos. Por ejemplo, en el contexto amigable de una fiesta, el comportamiento

de saludo de un agente puede ser realizado con entusiasmo, mientras que en un ambiente de negocios o de oficina, el mismo comportamiento sería realizado con seriedad y respeto. En el caso de la aldea virtual, el único contexto que se manejó fue el de la historia, representada por un objeto mediante una concatenación de estados. La transición entre estos estados se daba a medida que se cumplían ciertos eventos clave de la historia. Estas transiciones generaban a su vez un tipo de evento especial (evento de cambio de estado de la historia), que podía ser procesado por manejadores correspondientes para habilitar o deshabilitar los comportamientos de los agentes (en contraste con la activación y la desactivación). El estado de la historia servía entonces como una condición adicional de activación, de tal manera que un comportamiento sólo puede ser activado si está habilitado, como se demuestra en la Figura 3 (y sólo ejecutado si está activo, como ya se explicó anteriormente).

**Figura 3.** La organización de la figura muestra, debajo de los estados de una historia, el conjunto de comportamientos habilitados en ese estado. El estado actual es el que corresponde al recuadro sombreado y sólo ahí se encuentran comportamientos activos, entre los cuales el agente hace su elección. Un comportamiento puede estar habilitado en varios estados de la historia, caso que no es representado en la figura.



La arquitectura desarrollada para este proyecto, entonces, permite dotar de inteligencia a los agentes que representan a los personajes, mediante el manejo

de tres aspectos. Primero, se tiene la cantidad de comportamientos disponibles a un agente. Cuantos más comportamientos se puedan realizar simultánea o alternativamente, más variada y completa será la actuación del agente. Se tiene también como aspecto relacionado la granularidad de los comportamientos, que se refiere a qué tantas o qué tan complejas son las acciones realizadas por un solo comportamiento. Cuanto mayor número o complejidad (menor granularidad) de acciones, la credibilidad del agente será menor, en tanto que su comportamiento parecerá empaquetado, especialmente si el mismo comportamiento es ejecutado más de una vez. Por ejemplo, en vez de programar un comportamiento de saludo que incluya las palabras, la sonrisa y un gesto con las manos, se pueden programar por separado los comportamientos de las palabras de saludo, la sonrisa (o gesto facial) y el gesto con las manos. De esta manera, el agente no siempre saludará de la misma manera, sino que tiene capacidad de escoger entre varios comportamientos según el contexto u algún otro criterio, todos los cuales serán válidos (estarán activos) para una situación de saludo. El tercer aspecto es, por supuesto, el manejo del contexto, que puede inclusive compensar los problemas de una baja granularidad, ya que una granularidad alta también añade problemas de complejidad en el diseño del agente. Cuanto más rica sea la información de contexto y más variadas las respuestas de los diferentes comportamientos de respuesta, más creíble será la actuación del agente.

Este último punto fue el que con menor rigurosidad se implementó en el desarrollo de la arquitectura de IA, en cuanto a reusabilidad y extensibilidad. Si bien la cantidad y granularidad de los comportamientos es algo que los programadores pueden manejar a su antojo en la arquitectura desarrollada, no se desarrolló un mecanismo estándar para comunicar ni interpretar el contexto. Esto es, en parte, por la complejidad y variedad de tipos de contexto que habría posibilidad de codificar. Isbister y Doyle (1999) y Doyle (2002) sugieren codificar el contexto en anotaciones dependientes del ambiente en el que se desenvuelve el agente, de tal manera que éste pueda leer e interpretar dichas anotaciones a medida que va recorriendo el mundo virtual. Este modelo se presta muy bien para el tipo de aplicación al que

pertenece este proyecto, ya que cada historia tiene y hace parte de un entorno físico, que a su vez puede tener una cantidad de características propias que pueden dar forma al comportamiento de agentes guías genéricos. Este es un aspecto que puede hacer parte de nuestro trabajo futuro.

Con el desarrollo anterior, el nivel de inteligencia incorporada a los personajes del mundo mítico de la aldea es bajo (comparable con aquel encontrado en los juegos de aventura mencionados anteriormente). Con un guión casi completamente lineal y un rango limitado de acciones por parte del usuario, la programación del agente pudo haber sido realizada con patrones de programación más convencionales. Sin embargo, lo importante de este desarrollo es que es bastante genérico y permite el desarrollo de agentes mucho más sofisticados que cumplan las condiciones de credibilidad, por medio del diseño de los comportamientos y la aplicación de diferentes técnicas de decisión.

## 2.2 Múltiples Modalidades

El trabajo realizado para este proyecto es un primer paso en el campo de las interfaces multimodales, para explorar la implementación y aplicación de modos diferentes para cada una de las interacciones del usuario dentro de la aldea virtual. La idea fue crear los módulos necesarios para manejar la entrada de cada uno de los dispositivos, observar la interacción de los usuarios y sacar conclusiones acerca de sus ventajas y desventajas, en cuanto a facilidad de uso y naturalidad. No se realizó ningún tipo de fusión ni fisión multimodal.

A continuación se describirán cada uno de los modos de interacción utilizados y las observaciones hechas para cada uno de ellos. La figura 4 muestra una foto de dos usuarios interactuando con el sistema mediante los diferentes dispositivos.

**Figura 4.** Niños utilizando el programa frente a la pantalla de proyección



El usuario de la aldea virtual tiene la capacidad de comunicarse por medio del habla con los personajes que la habitan. La implementación de un sistema de reconocimiento de voz para lograr este objetivo presenta numerosas dificultades, aunque existen varios sistemas y marcos de desarrollo comerciales y no comerciales que pueden utilizarse para este propósito. Entre los primeros están ViaVoice de IBM, Dragon Naturally Speaking y OpenSpeech de Speech Works, mientras que entre los segundos están Sphinx de CMU y HTK de la Universidad de Cambridge (Flippo, 2003)

La primera consideración al buscar un reconocedor de voz, es el hecho de que sea dependiente o independiente del usuario. Los sistemas dependientes del usuario requieren que éste realice una sesión de entrenamiento antes de poder utilizar el sistema, y la efectividad del reconocimiento depende en gran parte de este entrenamiento, de tal manera que el reconocimiento es pobre para un usuario que no ha entrenado el sistema o que lo utiliza

con una configuración ajena a la propia. Todos los sistemas comerciales mencionados son dependientes del usuario. Los sistemas independientes del usuario, por otra parte, no requieren dicho entrenamiento y, aunque su efectividad no llega a ser tan alta como la de los reconocedores dependientes del usuario, alcanzan un nivel aceptable para un rango amplio de usuarios. Tanto Sphinx como HTK pueden ser independientes del usuario. Dada la naturaleza del proyecto y la forma como se realizaría la experiencia (ver sección 2.4), además del carácter libre de las herramientas, se optó por un sistema independiente del usuario. Se escogió HTK por su implementación, la cual facilitaba su integración a nuestro desarrollo. HTK también permite crear reconocedores dependientes del usuario y mecanismos para extender el vocabulario reconocido.

El reconocedor del HTK depende de un diccionario fonético, el cual contiene pronunciaciones de las diferentes unidades sonoras que se pueden reconocer en un idioma, así como las transiciones en la pronunciación continua de estos fonemas. Véase (Young *et. al.*, 2002) para una descripción completa de esta herramienta. Una desventaja de la selección del reconocedor es el hecho de que no se consigue un diccionario fonético diferente al del inglés. Aunque un diccionario propio del español se puede construir utilizando las herramientas que el sistema proporciona, ésta es una tarea dispendiosa, especialmente si se quiere que el reconocedor resultante sea independiente del usuario. Por esta razón se utilizó el diccionario inglés, utilizando para las palabras del vocabulario reconocido fonemas que fueran lo más cercanos posibles a los del español. El vocabulario utilizado, entonces, tuvo que ser reducido para lograr resultados aceptables de reconocimiento y la entrada de voz se limitó a la respuesta de preguntas realizadas por los personajes del sistema (el usuario no podía comenzar la conversación). Para cada pregunta se utilizó un reconocedor específico para las posibles respuestas de esa pregunta, obteniendo un margen de error mucho más pequeño del que se habría tenido utilizando vocabularios más grandes. Éste es un punto que se puede mejorar.

Además de comunicarse con los personajes, el usuario debe poder moverse dentro del mundo virtual. Para ello, se decidió utilizar un *joystick*,

común en los videojuegos a los que nuestro público objetivo debía estar acostumbrado. En los mundos recreados, el usuario estaba limitado a caminar sobre la superficie del terreno simulado, lo que simplificaba la interacción al reducirla a dos grados de libertad. Como era de esperarse, no hubo inconvenientes con la utilización del *joystick* como dispositivo para el control del movimiento, ni desde el punto de vista de aplicación ni del de implementación, ya que el motor Nebula por defecto soporta la utilización de este tipo de dispositivo.

Aunque no pudo probarse en la experiencia realizada, se implementó también un módulo de interacción por medio de un rastreador electromagnético de seis grados de libertad (posición y orientación en el espacio). El propósito de este módulo es la interacción y posible manipulación de objetos que se encuentran en el mundo virtual. En el caso de los mundos recreados para la experiencia, el usuario debía romper una vasija y abrir la puerta de una cueva. Para ello, el rastreador electromagnético controla la posición de un apuntador (una varita mágica) en tres dimensiones, que aparece frente a la posición del usuario cuando ello es necesario. Moviendo el apuntador con el rastreador, como lo haría con un objeto real, el usuario puede realizar las interacciones deseadas con los objetos frente a él, en este caso, por medio de una colisión. El rastreador también está integrado con una interfaz de *mouse*, por lo que es posible programar interacciones más complejas que combinen el movimiento con eventos de los botones del *mouse* (p. ej. seleccionar o agarrar y arrastrar). Sin embargo, en la experiencia fue necesario utilizar un botón del *joystick* para realizar las acciones requeridas.

En general, la interacción de los usuarios con el sistema fue bastante satisfactoria. A pesar de algunas dificultades con el reconocimiento de voz, debidos más que todo a factores circunstanciales, los usuarios pudieron navegar sin problemas en el mundo virtual y resolver el problema planteado.

### 2.3 Integración en Nebula

El motor Nebula2 cuenta con una arquitectura de servidores en la cual cada servidor realiza una

función específica. Hay un servidor principal o Kernel, al cual se inscriben servidores como el de video (GFX), audio, *input* y *scripting* dependiendo de las necesidades de la aplicación. Esta arquitectura permite utilizar sólo las funcionalidades necesarias, y al mismo tiempo, facilita la modificación individual de los componentes del motor, para ser adaptados a los requerimientos específicos del proyecto.

Se hizo uso de la capacidad de utilizar el lenguaje de *scripting* TCL/TK, incorporada en el motor Nebula2 para implementar archivos de configuración que permitieran definir el comportamiento de los dispositivos de entrada y la organización de los mundos virtuales, los objetos y los personajes. Esto facilitó la creación de la aplicación, ya que se eliminó la necesidad de recompilar todo el proyecto cada vez que se necesitaba hacer una prueba con una configuración diferente.

Los modelos 3D de los personajes, objetos y mundo, fueron creados en la aplicación Maya 6.0, de la compañía Alias. El siguiente paso fue exportar los modelos a formato .obj, para luego utilizar la herramienta *nmeshtool.exe* que convierte estos archivos a formato .n3d2, versión mejorada del .obj, creada exclusivamente para ser usada en Nebula2. La herramienta *nmeshtool.exe* viene incluida en el paquete de distribución de Nebula2, junto con su código fuente, al igual que otras herramientas, lo cual permite adaptarlas a las necesidades específicas de los programadores.

Para crear las escenas se utilizó un conjunto de archivos con extensión .n2 que contenían código en lenguaje (*script*) TCL. Estos archivos contenían la información detallada de cada uno de los objetos pertenecientes a las escenas (mundos virtuales), especificando para cada objeto atributos como la posición, rotación y el modelo (archivo .n3d2) a utilizar. La utilización de estos *scripts* para cargar la información del mundo, facilitó el proceso de pruebas al permitir cambiar el contenido del mundo virtual, sin tener la necesidad de compilar de nuevo la aplicación.

Toda la programación fue realizada en C++, a excepción de los archivos de configuración, escritos en TCL. La integración con los módulos de IA y de Interacción se realizó con facilidad, ya que estos también se adaptaron a la arquitectura de Nebula2.

Se utilizó un modelo de programación iterativo en el que cada cierto tiempo se producía una versión funcional de la aplicación que iba incorporando nuevas funcionalidades.

## 2.4 El juego

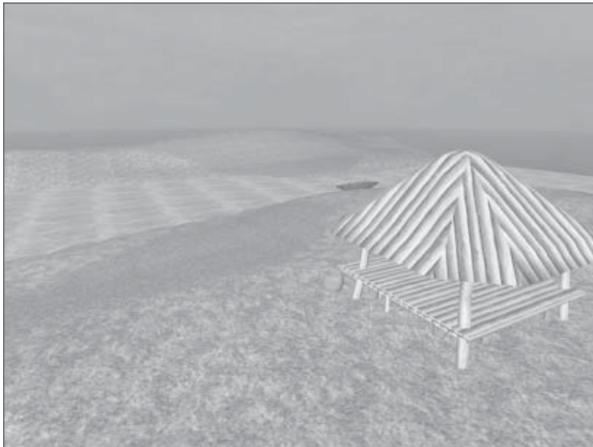
El juego, implementado con las tecnologías descritas anteriormente, se desarrolla en dos mundos: En el primer mundo el niño se encuentra con un indígena virtual, que le pregunta si quiere visitar primero el tambo o el río. El indígena virtual es un agente autónomo, sus acciones y su diálogo están determinados por la arquitectura de agentes descrita en la sección 3.1. El niño responde utilizando su voz.

Dependiendo de la respuesta, el guía virtual invita al niño a conocer uno u otro. En cada sitio el indígena menciona algunos conceptos relativos a la cultura Emberá. Al dejar cada sitio, el guía le hace una pregunta al niño para tratar de medir su retención de los términos recién aprendidos.

Al culminar el primer mundo, el niño aparece en un segundo mundo, que corresponde con el mito de la creación del agua de la cultura Emberá. En este mundo el niño es recibido por *Caragabí*, el protagonista del mito. *Caragabí*, convertido en un colibrí, acompaña y guía al niño hasta conseguir liberar el agua, que estaba siendo acaparada por *Gentzerá*. Durante esta etapa el niño debe buscar algunos elementos dentro del espacio virtual y utilizar un rastreador 3D, que se despliega en el mundo virtual como una varita mágica, para quebrar una vasija y para mover una piedra que cubre una cueva.

Las Figuras 5, 6 y 7 muestran varios momentos del juego.

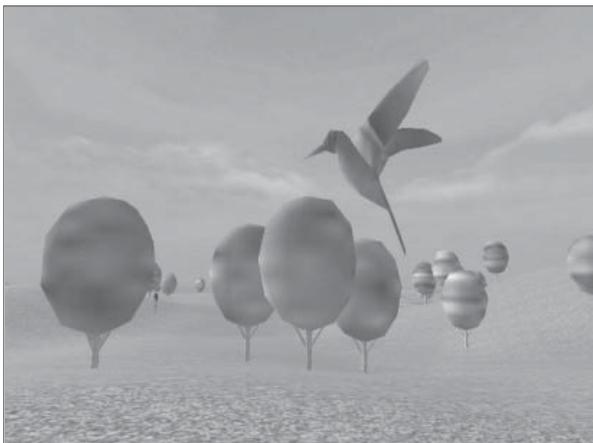
**Figura 5.** El tambo y el río



**Figura 6.** *Gentzerá*, el carácter virtual que tiene escondido el agua



**Figura 7.** *Caragabí*, un carácter virtual autónomo, convertido en Colibrí



### 3. Experimento

El experimento se desarrolló con 25 alumnos del colegio San Ignacio de Loyola, del grado Transición, cuyas edades oscilaban entre 6 a 7 años. El experimento tuvo como objetivo medir la sensibilización del grupo, a las diferentes manifestaciones y modos de vivir de una cultura indígena, en este caso la cultura Emberá.

Con la herramienta de Realidad Virtual se creó un prototipo que representaba una aldea indígena, con sus características y diferentes personajes representativos. A través de este programa, los niños pudieron ver y ponerse en contacto con las formas de vida, costumbres y mitos de la cultura, y generar un cambio en las estructuras cognitivas y en su valoración hacia las diversas formas culturales que habitan nuestro país.

Con los alumnos del grado Transición se realizó una sesión previa, como medio de preparación para la experiencia con el proyecto de la Aldea Virtual. Durante la clase los alumnos hablaron sobre lo que sabían acerca de los indígenas, por qué tenían interés en conocerlos y algunos aspectos que les eran desconocidos (pre-saberes). “Hay evidencia de que el aprendizaje se incrementa cuando los maestros prestan atención a los saberes y creencias con que los aprendices llegan a una tarea de aprendizaje; usan estos conocimientos como punto de partida para la instrucción nueva; y hacen un seguimiento constante de las concepciones cambiantes de los estudiantes a medida que avanza la instrucción” (Bransford *et. al.*, 2000).

Los temas que surgieron durante la sesión de trabajo con los niños fueron:

- Su forma de vida (vivienda, ubicación geográfica, medios de transporte)
- La alimentación (la cacería, forma de cocinar los alimentos)
- Dónde y cómo viven
- Sus costumbres
- La estructura familiar y social
- Su forma de vestirse y la pintura corporal

Al finalizar la sesión, cuando todos los niños habían aportado sobre el tema y aclarado algunas dudas, se realizó un trabajo conjunto, en el que ellos elaboraron una cartelera con láminas y fotos de indígenas, y se pusieron mensajes alusivos a los valores que se deben fomentar para la convivencia con estas culturas, tan diferentes a la nuestra. La idea con esta primera sesión era la preparación a la experiencia central y la posibilidad de aplicar una prueba de entrada (pre-test) a los alumnos, que ayudó a conocer la información que tenían sobre el tema y la valoración de las diversas formas de manifestación cultural. De esta forma se respetaba la metodología empleada por la institución educativa y realizábamos la experiencia articulada a la manera de trabajar de los niños. “Una extensión lógica de la concepción de que el conocimiento nuevo debe construirse a partir del conocimiento preexistente, es que los maestros necesitan prestar atención a las comprensiones incompletas, a las falsas creencias y a las versiones ingenuas de conceptos con los que los aprendices llegan a abordar una materia de estudio. Los maestros deben, entonces, partir de aquellas ideas, de tal manera que ayuden a cada estudiante a lograr una comprensión más madura. Si se ignoran las ideas y creencias iniciales de los estudiantes, las comprensiones que ellos realizan pueden ser muy diferentes de las que el profesor se propone alcanzar” (Bransford *et. al.*, 2000). De igual manera, se les contó sobre el trabajo que iban a realizar en la universidad y se dividió el grupo en 13 alumnos para el grupo experimental y 12 alumnos para el grupo control.

En la segunda sesión, llevada a cabo en la Universidad Eafit, los alumnos fueron divididos, de manera aleatoria, en los grupos control y experimental, y se trabajó con ellos en un periodo de 15 minutos, donde se les amplió la información sobre las tribus indígenas, de manera especial la cultura Emberá. Se narró el mito sobre la creación del agua y al terminar la sesión los grupos fueron separados para el trabajo individual.

Los alumnos del grupo experimental trabajaron por parejas, interactuando con el programa a través del micrófono, con el cual respondían las preguntas de los personajes de la aldea. De igual manera, manejaban el *joystick* para desplazarse y realizar varias tareas asig-

nadas por los personajes del mundo virtual (abrir la roca, romper la vasija, seguir al colibrí).

En todo momento los niños mostraron gran motivación y deseo de participar, ayudando en las tareas que los personajes les sugerían, resolviendo problemas conjuntamente y mostrando una curiosidad especial todo el tiempo. Se movían con destreza por el mundo y exploraban cada espacio, superando los diferentes obstáculos: “los niños son aprendices activos que traen un punto de vista al escenario de aprendizaje. El mundo al que ingresan no es una creciente y bulliciosa confusión, donde cada estímulo tiene la misma importancia. Más bien, el cerebro de un infante les da prelación a ciertas clases de información: el lenguaje, los conceptos básicos de número, las propiedades físicas y el movimiento de objetos animados e inanimados” (Bransford *et. al.*, 2000).

Los alumnos del grupo control asistieron a la narración del mito y después estuvieron de manera separada (del grupo experimental) en una sesión de preguntas y respuestas relacionadas con lo trabajado sobre los Emberás.

A los niños del grupo control se les notó su capacidad para entender el tema trabajado y responder de manera efectiva a las preguntas planteadas sobre los diferentes aspectos tratados; su motivación era alta, pero al no tener la herramienta era más difícil la interacción con el grupo y se perdía la concentración.

Al concluir la experiencia, se aplicó el pos-test a cada uno de los alumnos participantes, con la finalidad de medir el avance en la adquisición de conocimientos y el cambio en algunos de los valores necesarios para el entendimiento y aceptación de la diversidad cultural.

#### 4. Resultados

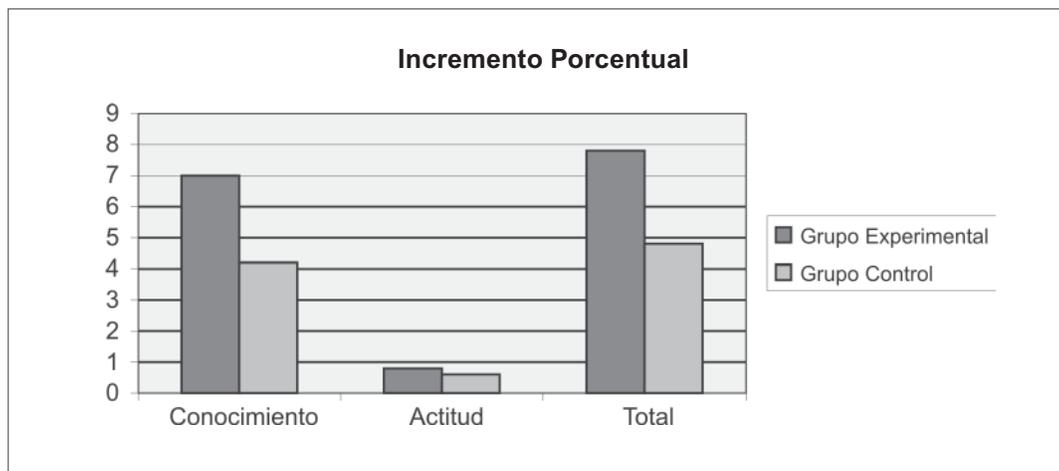
Luego de recibir la clase normal (grupo de control) y culminar exitosamente el juego (grupo experimental), los niños respondían verbalmente un conjunto de preguntas (pos-test). Las respuestas de los niños fueron transcritas y luego calificadas por un experto en pedagogía. En el pos-test, 9

preguntas apuntaban a medir conocimiento y 1 pregunta a medir cambios actitudinales. Algunas de las preguntas para medir conocimiento aludían la capacidad del niño de recordar conceptos concretos; por ejemplo: el nombre de la cultura que se estaba conociendo, el nombre de los diferentes personajes del mito, la ubicación geográfica de la cultura, etc. Otras preguntas implicaban una capacidad básica de análisis de las relaciones con otros seres humanos; por ejemplo: diferencias entre las costumbres de los Emberás y nuestra cultura, diferencias en las relaciones con el entorno, etc. La pregunta para

medir el cambio actitudinal apuntaba a descubrir en el niño la actitud que debemos tener hacia personas de culturas diferentes. En la calificación de esta pregunta se buscaba identificar, en la respuesta, la mención de valores tales como respeto, aceptación, interés por la diferencia, etc.

La Figura 8 muestra las diferencias en las valoraciones cuantitativas de los grupos experimental y de control. Las valoraciones numéricas son los resultados del pos-test.

**Figura 8.** Notas obtenidas por los niños en el pos-test: grupo experimental y grupo de control



Los mismos datos están descritos en la tabla 1.

**Tabla 1.** Notas obtenidas por los niños en el pos-test

	Grupo Experimental	Grupo Control
Media	7.70	4.83
Desv. Est.	0.99	2.78
N	13	12

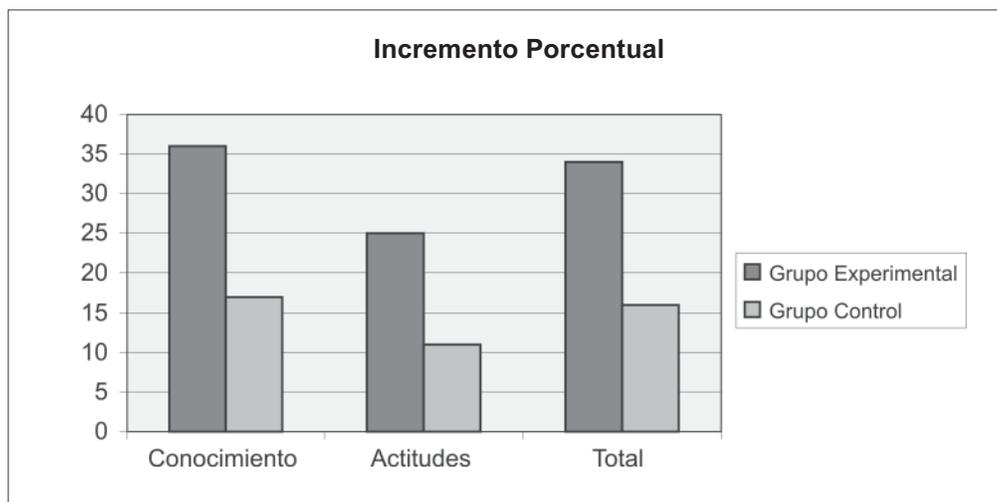
Al aplicar un test *t de student* sobre los datos ( $p = 0.0025$ ), se puede concluir que existe una diferencia significativa entre los resultados finales de los niños del grupo experimental y grupo de control. Los resultados de los niños que realizaron la experiencia en el entorno virtual, son mejores que los de los niños que recibieron la clase normal.

También es interesante observar el incremento porcentual en las valoraciones cuantitativas entre el pre-test y el pos-test. Esta diferencia indicaría qué tanto aprendieron los participantes de cada grupo durante la experiencia de realidad virtual (grupo experimental) o durante la clase normal (grupo de control).

La figura 9 describe los incrementos porcentuales entre el pre-test y el pos-test, tanto para el grupo experimental como para el grupo de control.



**Figura 9.** Incremento porcentual en las valoraciones cuantitativas entre el pre-test y el pos-test. Grupo experimental y grupo de control



Los mismos datos están descritos en la tabla 2.

**Tabla 2.** Diferencia porcentual entre las valoraciones cuantitativas del pre-test y el pos-test

	Grupo Experimental	Grupo Control
Media	0.34	0.16
Desv. Est.	0.20	0.26
N	13	12

## Agradecimientos

Agradecemos al Colegio San Ignacio, a su rector, directivas, profesoras y alumnos que nos acompañaron en la experiencia. Agradecemos también a Jesús Alejandro Vega, quien facilitó algunos de los modelos de los personajes utilizados en el juego. Lázaro Mesa aportó información muy útil respecto a los Emberá; siguiendo su recomendación decidimos representar el mito de la creación del agua. La Universidad Eafit apoyó completamente la ejecución del proyecto.

## Conclusiones y trabajo futuro

Los resultados sugieren que un entorno virtual con interfaces multimodales, diseñado y construido de manera adecuada, permite que niños de edad preescolar adquieran conocimientos de manera significativa mientras “juegan”. Así mismo, parece que se logran algunos ligeros cambios actitudinales hacia la cultura descrita en el entorno virtual. Estos resultados sugieren que la implementación de un entorno de este tipo en sitios públicos, tales como museos interactivos, tendría un impacto positivo en cuanto al conocimiento y respeto de las culturas ancestrales.

Desde el punto de vista técnico, es conveniente experimentar con visión estéreo y sonido 5.1, con miras a lograr una experiencia más inmersiva. Adicionalmente, en un proyecto futuro con más tiempo disponible, se puede mejorar el realismo de los movimientos y las características físicas de los personajes y del entorno. Sin embargo, sería interesante determinar si un mayor realismo conduce, o no, a un mejor aprendizaje por parte de los usuarios.

Sería muy importante también probar el uso de múltiples pantallas de proyección (por lo menos 3), conformando un verdadero ambiente tipo “CAVE” (Cruz-Neira *et. al.*, 1992), (Browning *et. al.*, 1993). En un ambiente así la experiencia debe ser mucho más inmersiva y motivadora.

Desde el punto de vista pedagógico, se recomienda tener más tiempo con los niños, antes y después de la experiencia. Esto permitiría, por un lado, articular de manera más profunda y transparente la experiencia con los procesos del colegio y los proyectos realizados dentro del aula de clase. Por otro lado, permitiría culminar con un trabajo en el cual los niños expresen el valor de los contenidos abordados realizando una actividad (un cuento, un mural grupal, una muestra artística en torno a la cultura indígena, etc.)

## Bibliografía

Bates, J. (1994). “The role of emotion in believable agents.” En: *Communications of the ACM*, 37(7). pp.122–125.

Blumberg, B. (1994). “Action-selection in hamsterdam: Lessons from ethology”. In *Proceedings of SAB94*. pp. 108–117.

Bolt, R.A. (1980). ”Put-that-there: Voice and gesture at the graphics interface”. In: *Computer Graphics*. 14(3). July. pp. 262–270.

Bransford, J.; Brown, A. L. and Cocking, R. R. (2000). *How People Learn: Brain, Mind, Experience, and School*. National Academies Press.

Browning, D.; Cruz-Neira, C.; Sandin, D. and DeFanti, T. (1993). “Virtual reality: The design and implementation of the cave”. In: *Proceedings of SIGGRAPH'93*, ACM, August. pp. 135 -142.

Cohen, P.R.; Johnston, M.; McGee, D.; Oviatt, S.; Pittman, J.; Smith, I.; Chen, L.; and Clow, J. (1997). “Quickset: Multimodal interaction for distributed applications”. In: *ACM Internations Multimedia Conference*, New York. pp. 31 – 40.

Cruz-Neira, C.; Sandin, D.; DeFanti, T.; Kenyon, R. and Hart, J. (1992). “The cave: Audio visual experience automatic virtual environment”. In: *Communications of the ACM*, June. 35(6). pp. 64 - 72.

Devmaster 3d engines. (2004). <from: <http://www.devmaster.net/engines/> > (Consulta: Diciembre 2004.

Doyle, Patrick (2002). “Believability through context: Using knowledge in the world to create intelligent characters”. In: *Proceedings of the First international joint conference on Autonomous agents and multiagent systems*. pp. 342 - 349.

Elliott, C. (1998). "Story-morphing in the affective reasoning paradigm: Generating stories semi-automatically for use with emotionally intelligent agents". In: *Proceedings of Autonomous Agents*.

Flippo, Frans (2003). "A natural human-computer interface for controlling wheeled robotic vehicles". Master's thesis, Delft University of Technology, August.

Irrlicht 3d engine. (2004). <from: <http://irrlicht.sourceforge.net/> >. (Consulta: Diciembre, 2004)

Isbister, Katherine and Doyle, Patrick (1999). "Touring machines: Guide agents for sharing stories about digital places". In *Narrative Intelligence: AAAI Fall Symposium*. pp. 93 -100.

Kline, C. and Blumberg, B. (1999). "Synthetic character design". In *Proceedings of Symposium on AI and Creativity in Entertainment and Visual Art*.

Maes, P. (1994). "Modeling adaptive autonomous agents". In: *Artificial Life, I*, (1&2). p. 9.

Martin, D.L.; Cheyer, A.J. and Moran, D.B. (1999). "The open agent architecture: A framework for building distributed software systems". In: *Applied Artificial Intelligence*, Jan-March.13(1-2). pp. 91-128.

Object-oriented graphics rendering engine (2004). <from: <http://www.ogre3d.org/>>. (Consulta: Dic, 2004)

Radon labs. (2004). <from: <http://www.project-nomads.de/english/index1.htm>>. (Consulta: Diciembre, 2004)

Seneff, Stephanie; Hurley, Ed; Lau, Raymond; Pao, Christine; Schmid, Philipp and Zue, Victor (1998). "A reference architecture for conversational system development". In: *Proceedings of ICSLP*, November. pp. 931 - 934.

The nebula device 2. (2004). <from: <http://nebuladevice.cubik.org/> > (Consulta: Diciembre, 2004)

Toledano, D.; Wang, S.B.; Cyphers, S. and Glass, J. (2002). "Extending the galaxy communicator architecture for multimodal interaction research". Submitted to *ACM Trans. on Human-Computer Interaction*, August.

Velasquez, J. D. (1999). "From affect programs to higher cognitive emotions: An emotion-based control approach". In: *Proceedings of Autonomous Agents Conference*.

Young, S.; Evermann, G.; Hain, T.; Kershaw, D.; Moore, G.; Odell, J.; Ollason, D.; Povey, D.; Valtchev, V. and Woodland, P. (2002). "*The HTK Book*" (for version 3.2.1). Cambridge Universit