

# Esquema de Representación Espacial Multinivel para el Modelamiento Geométrico de Sistemas Complejos

## Spatial Multilevel Representation for the Geometric Modeling of Complex Systems

Juan C. Ibarra L., Ing., Oscar J. Chavarro G., MSc., David A. Castro M., Ing., José Hernández P., PhD.  
Grupo IMAGINE, Universidad de los Andes, Bogotá, Colombia  
{ju-ibarr, oj.chavarro184, da-casatr, jhernand}@uniandes.edu.co

Recibido para revisión 28 de Noviembre de 2007, aceptado 14 de Febrero de 2008, versión final 28 de Febrero de 2008

**Resumen**—En este artículo se presenta una propuesta de estructura y un prototipo de visualización y navegación para el manejo de información volumétrica compleja, específicamente órganos humanos. La estructura de datos está basada en grafos jerárquicos y permite tener una representación multi-escala de información espacial y una visualización más detallada de una región de interés, al tiempo que mantiene la contextualización de la misma a su nivel global. La estructura de datos se ha implementado como una librería en C++ para ser utilizada como base en la construcción de ambientes inmersivos de investigación, aprendizaje y toma de decisiones.

**Palabras Clave**—*E-Learning* y ambientes Colaborativos de Aprendizaje, Esquema de representación geométrica, Estructura de Datos Geométricos, Visión por Computador y Realidad Virtual.

**Abstract**—In this paper a new data structure for the representation of complex volumetric data is presented, and accompanied with a visualization and navigation prototype for managing it, particularly for modeling human organs. The data structure is based upon hierarchical graphs which permit a multiscale information and a detailed visualization of a region of interest, and a contextual visualization of the global level information. The datastructure has been implemented on C++ for its use in immersive environments for research, education and decision support.

**Keywords**— *E-learning* and collaborative learning environments, graphical representations, geometric data structures, virtual reality and computer vision.

### I. INTRODUCCIÓN

En la actualidad se encuentran disponibles diversas estructuras de datos para la representación geométrica, de entre las cuales se destacan los grafos de escena [ROHL1994]. Desde los primeros desarrollos en el área [CLAR1976], se han identificado situaciones en las cuales el tener una jerarquía de objetos geométricos ayuda a manejar geometrías complejas. Por ejemplo, el concepto de niveles de detalle (LOD por sus siglas en inglés) consiste en aprovechar las diferentes posiciones y orientaciones que puede tomar un observador para cambiar entre diferentes versiones del mismo objeto geométrico, colocando una versión más detallada cuando el observador se encuentre cerca del objeto y una versión menos detallada cuando el observador y el objeto se encuentren lejos. Otros autores han extendido esta idea a la oclusión [TELL1991] [HOWA1996] [HUBB1999], para descartar del todo el objeto geométrico en el caso de que no sea visible. Estas estructuras de datos han sido menos utilizadas para representar diversos niveles de abstracción (estructuras multinivel). Recientemente, [FU2007] ha plasmado la idea de tener una estructura de datos capaz de almacenar objetos geométricos de muy diferentes tamaños, como una galaxia, una estrella y un planeta, o en el caso del proyecto propuesto, un planeta, una ciudad, un edificio, un ladrillo, una cabeza, un cerebro, una neurona y una molécula de ADN. La estructura propuesta es similar al octree presentado en [YEME2003].

En este artículo se presenta una propuesta de esquema de representación de objetos complejos multi-nivel con vistas múltiples. También se presenta un esquema de visualización interactiva que, basado sobre dicho esquema de representación, permite navegar en la información del objeto en un esquema

foco-contexto[RAO1994] de tal forma que se pueda acceder a información detallada conservando, en la visualización, información sobre su contexto (entorno). Adicionalmente, el esquema de navegación permite visualizar la existencia de las diferentes vistas y navegar entre ellas.

Las características incorporadas en la estructura de datos propuesta mantienen la funcionalidad de los niveles de detalle de las estructuras tipo grafo de escena, y la extienden mejorando el manejo de la información periférica o contextual, las múltiples "vistas" (que representan intereses temáticos o funcionales).

El manejo de la información periférica o contextual respecto a un foco, y la representación de intereses o puntos de vista se definen sobre la estructura a través de sub-grafos que permiten representar la información de interés sobre todo el conjunto utilizando técnicas de foco-contexto, se ha tratado anteriormente [ARIZ2006].

La múltiple visibilidad sucede cuando diferentes usuarios visualizan la misma información desde diferentes puntos de vista, permitiendo la creación de un ambiente de colaboración distribuido entre distintos observadores del mismo conjunto de datos. La propuesta presentada en este artículo está orientada a ambientes colaborativos, con posibilidad de nodos con visualización inmersiva.

La estructura de datos se ha implementado en C++ y sobre ella se ha desarrollado una aplicación capaz de manejar la estructura en un ambiente de realidad virtual.

El artículo se encuentra dividido en 5 partes: la definición formal de la Estructura de datos en donde se especifica el modelo de datos diseñado y la sustentación de las decisiones de dicho diseño, el estado actual del proyecto en donde se habla de la implementación de *SVGAT*(Grafos Anatómicos/Temáticos), las plataformas tecnológicas utilizadas y las técnicas de visualización e interacción hombre máquina. Por último se encuentran los resultados, el trabajo futuro y la bibliografía.

## II. DEFINICIÓN DE LA ESTRUCTURA DE DATOS

La Estructura de datos representa una entidad volumétrica compleja tridimensional que evoluciona en un espacio 3-Euclidiano y cuyo volumen es finito y acotado. Dicha entidad se encuentra representada en una primera instancia como un **Grafo Anatómico** sobre el cual se definen **Grafos Temáticos** los cuales son visibles para el usuario. Estos grafos son navegables a partir de un **Cursor** y visualizados a partir de la definición de un **Observador**.

### A Grafo Anatómico

Es un grafo jerárquico en donde su estructura principal es similar a la de un árbol pero que además contiene relaciones entre nodos del mismo nivel. Estas relaciones representan características que facilitan la navegabilidad entre nodos tomando en cuenta aspectos diferentes a la jerarquía. En el

caso específico de la estructura, los niveles jerárquicos están definidos como características de contención, es decir, un nodo padre contiene espacialmente a sus nodos hijos. Se definen también relaciones entre nodos del mismo nivel como relaciones de vecindad en el espacio. La definición que se propone implica que cada uno de los nodos que componen el grafo jerárquico deben ser parte de una partición del espacio volumétrico total.

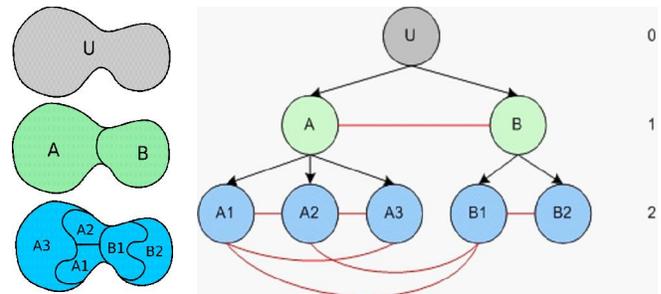


Fig. 1. Ejemplo de Grafo Anatómico.

Cada uno de los nodos que componen el **Grafo Anatómico** es llamado un **Nodo Anatómico** el cual define la representación de un Objeto de Interés que tiene varias características:

- **Posición:** los valores  $(x, y, z)$  en el espacio 3-Euclidiano en donde el objeto se encuentra.
- **Orientación:** vector que describe la dirección  $(x, y, z)$  de los ejes generadores del espacio local en donde el objeto se encuentra
- **Volumen:** Geometría que describe la forma del Objeto de Interés. Esta geometría debe estar bien formada desde el punto de vista geométrico y topológico [MANT1988], es decir, que debe ser una 2-variedad, cerrada y orientable.

Cada uno de los **Nodos Anatómicos** se construyen a partir de sus descendientes, en donde la unión de varios **Nodos Anatómicos** componen uno nuevo. Este proceso se denomina **Aglomerar**.

Debido a la relación de contención que define el **Grafo Anatómico**, se pueden crear particiones sobre el volumen total (raíz del grafo). Dichas particiones se llaman **Aglomerados** los cuales definen un nivel de detalle del espacio representado. Este **nivel de detalle** se refiere a la profundidad en donde se encuentra la partición del espacio representado. Los Aglomerados por lo tanto, pueden tener nodos que se encuentren a diferente **nivel de detalle**, ya que la estructura principal del grafo es un árbol que no necesariamente se encuentra balanceado.

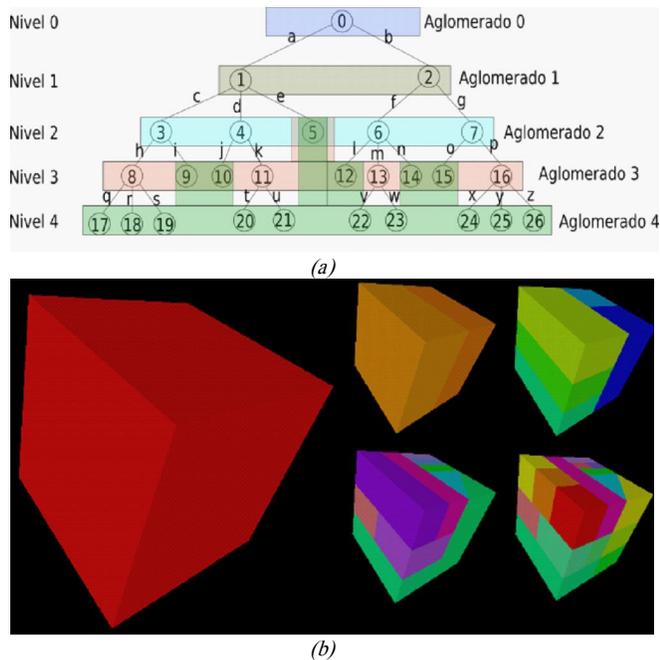


Fig. 2. Ejemplo de Grafo Anatómico con sus respectivos aglomerados. (a) estructura de datos (b) implementación en SVGAT de la estructura de datos

### B Grafo Temático

Son sub-grafos sobre el *Grafo Anatómico* que contienen información relevante para el usuario: un *Tema*. Un *Grafo Temático* tiene dos tipos de nodos que lo componen organizados en contextos: un *Contexto Temático* en el cual se encuentran los nodos que comparten una característica que los hace diferenciables y un *Contexto Anatómico* que es el conjunto de *nodos anatómicos* que complementan al contexto temático y forman una partición en el espacio. Los nodos del *Contexto Temático* llamados *nodos temáticos* son *nodos anatómicos* que contienen información adicional que los identifica como componentes del *tema*.

Al ser una partición, los nodos que componen un *Grafo Temático* deben cumplir ciertas características:

- Todos los nodos del *Grafo Temático* deben pertenecer al *Grafo Anatómico*.
- Los *nodos temáticos* no deben tener ni descendientes ni ancestros dentro del *contexto temático* ni anatómico.
- Todo *Grafo temático* debe formar una partición sobre la raíz del *Grafo Anatómico*.

Los *Grafos Temáticos* son construidos a partir del *Contexto Temático* siguiendo el siguiente algoritmo:

- Validar el contexto temático: ninguno de los nodos del contexto temático deben tener relaciones de jerarquía entre ellos.
- Armar el contexto anatómico

- Agregar los hermanos de los nodos que no tengan descendientes ni ancestros en el contexto temático
- Agregar los nodos de los aglomerados que contienen nodos temáticos que no tengan descendientes ni ancestros en el contexto temático ni anatómico ya calculado.

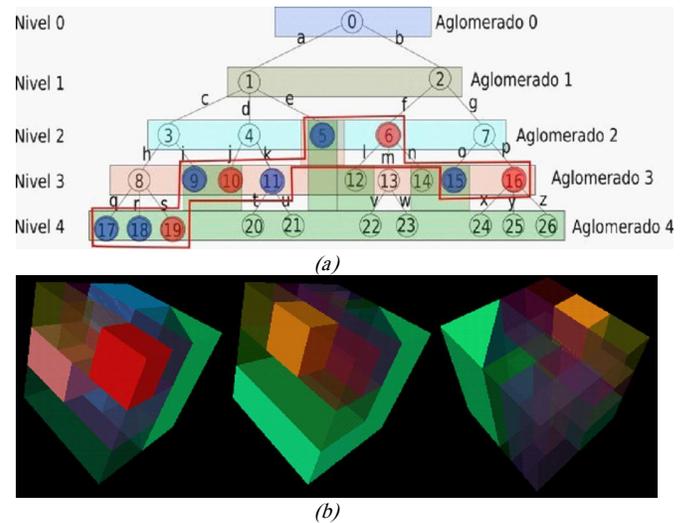


Fig. 3. Ejemplo de Grafo Temático. (a) estructura de datos de un grafo temático. (b) tres ejemplos de grafos temáticos en la implementación SVGAT

### C Cursor

Es un marcador al nodo anatómico que se encuentra actualmente seleccionado. Dicha selección permite moverse transversalmente a través del *Grafo Anatómico* y así cambiar la representación de la entidad volumétrica. Esta navegación permite concentrarse en un área de interés y bajar de nivel de detalle en una dirección determinada sin perder el contexto geométrico del área.

### D Observador

Se define como un vector y una distancia focal. El vector especifica la posición y orientación del observador y la distancia focal especifica la *escala de visualización*. Esta escala define un volumen de visualización dentro del cual se encuentra la geometría que se desea ver.

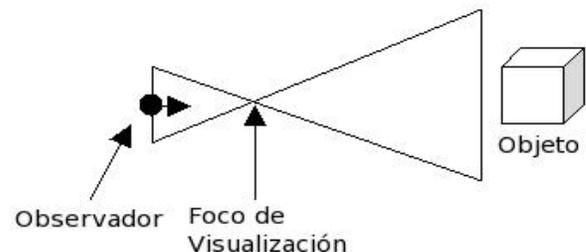


Fig. 4. Representación de un Observador.

III. IMPLEMENTACIÓN ACTUAL

La estructura de datos se ha implementado en C++, y hace parte de la plataforma de desarrollo del grupo IMAGINE [.wFIGU2008], [.wIMAG2008a], [.wIMAG2008b] y [.wOSG2008]. Como Herramientas de apoyo se está utilizando el sistema de grafos de escena OpenSG [.wOSG2008], el cual se utiliza también como motor de visualización. Para el manejo de la interacción hombre máquina se utiliza el sistema de filtros de interacción InTml [FIGU2002], Framework que permite definir modularmente la forma en que se interactúa con información 3D. InTml permite a los diversos participantes aprovechar múltiples dispositivos de interacción. Para visualización en gran formato, se utiliza Chromium[HUMP2001]. Para implementar la multi-visualización en un ambiente colaborativo distribuido, se utiliza la plataforma AccessGrid [.wAG2008] que permite el envío de la información a través de internet y provee mecanismos de tele-colaboración.

InTml [.wOSG2008], cuyo desarrollo adelanta el grupo Imagine, es un marco de desarrollo que ofrece componentes, llamados filtros, que se conectan entre si para construir una aplicación de realidad virtual con un fin específico. Existen filtros

encargados de manejar dispositivos, otros para la representación geométrica de objetos y otros para la descripción de técnicas de interacción. Por ejemplo, podrían tomarse para una aplicación que navegue por un mundo tridimensional un filtro encargado de tomar las entradas de un joystick, y que las coordenadas de movimiento pasen a otro filtro que se encargue de mover la cámara, y otro filtro que se encargue de pintar el mundo. InTml está desarrollado en C++ y utiliza las librerías OpenSG [.wOSG2008], VrJuggler [JUST1998] y VRPN [RUSS2001]. El lenguaje de descripción de aplicaciones de InTml es XML así que siempre que en varias máquinas se ejecute InTml se puede portar cualquier aplicación sin necesidad de compilaciones tediosas pues InTml sólo debe leer la descripción de la aplicación y organizar sus filtros para ejecutarla.

La implementación de la estructura de datos ha sido encapsulada en una librería de encadenamiento dinámico e integrada a InTml como una serie de filtros. Al comunicarse con otros filtros InTml, se ha permitido tomar ventaja del uso de dispositivos y técnicas de interacción ya disponibles.

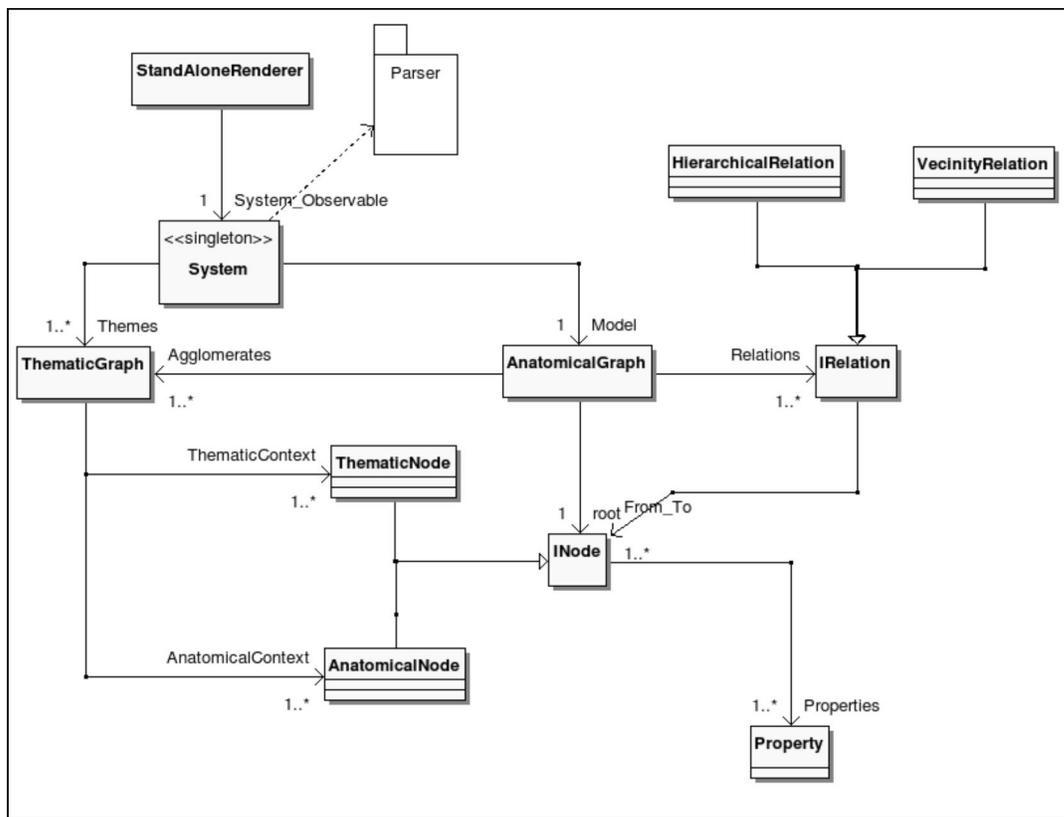


Fig. 5. Diagrama de clases del sistema SVGAT.

La aplicación, *SVGAT* está constituida por varios filtros InTml. Toma como entrada varios archivos XML que describen los grafos anatómicos y temáticos y los carga en memoria. Luego esa información es tomada por este mismo filtro y pasada al

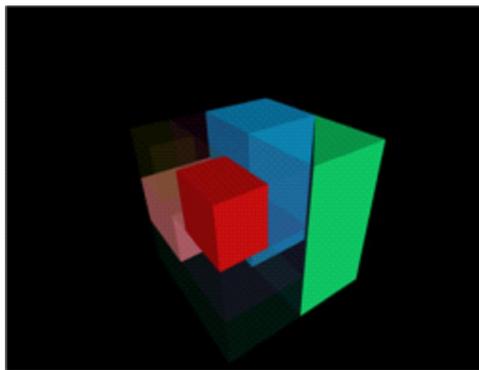
esquema de filtros de InTml dentro de un esquema de interacción 3D, en donde el usuario final manipula la estructura con la ayuda de técnicas de interacción 3D.

La aplicación, **SVGAT** implementa la definición de la estructura de datos de grafos anatómicos y temáticos anteriormente descrita. Se crearon 2 esquemas xml para la definición de los grafos anatómicos y temáticos. Usando el XSD parser de Code synthesis [.wXSD2007] se generaron las clases encargadas de leer los archivos xml.

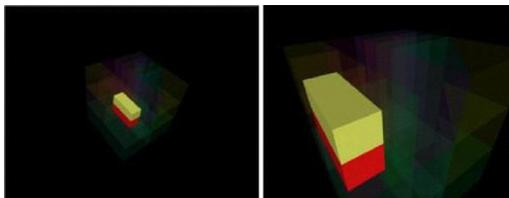
#### IV. RESULTADOS

Con base en las definiciones arriba presentadas se diseñó e implementó el sistema SVGAT (desarrollado en C++) que adicionalmente al manejador de la estructura de datos (que asegura su persistencia basada en XML), propone un esquema interactivo de navegación-visualización multi-nivel, multipuntos de vista. En la figura 5 se presenta el diagrama de clases UML del sistema.

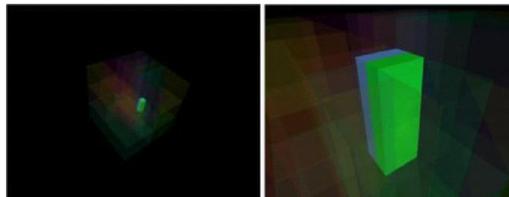
La figura 6 muestra un ejemplo sobre un objeto cúbico multinivel, se presentan tres escenas en las que el foco está en niveles diferentes de un mismo grafo anatómico. Se puede observar que la visualización muestra el contexto en cada nivel con bloques translúcidos.



(a)



(b)



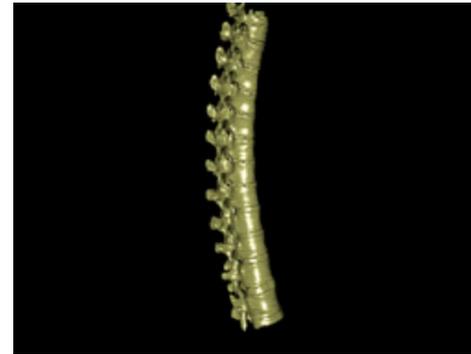
(c)

Figura 6. (a) Primer nivel de un objeto cúbico (b) Segundo nivel del objeto cúbico, normal a la izquierda y con acercamiento a la derecha.

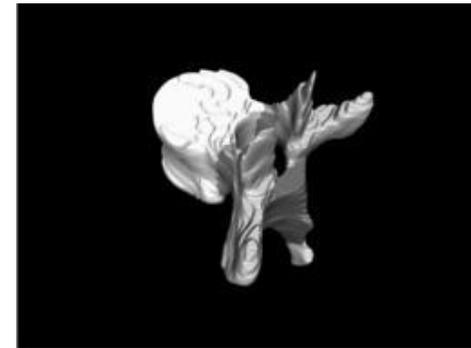
(c) Tercer nivel del objeto cúbico, normal a la izquierda y con acercamiento a la derecha.

La figura 7, utilizando un modelo de la región del tórax basado en el proyecto del humano visible [.wVIS2008], también multinivel, muestra la potencialidad de navegar, sin perder el contexto entre

diferentes niveles con tipos diferentes de información visualizada. (una imagen 2D de una célula). (la del tórax).



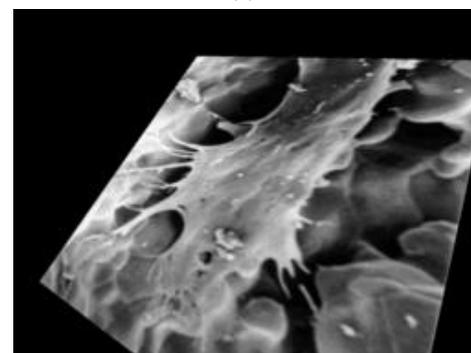
(a)



(b)



(c)



(d)

Fig. 7. Ejemplo de anatomía del tórax implementado en SVGAT. (a) Primer nivel (b) Segundo nivel sin contexto (c) Segundo nivel con contexto (d) Tercer nivel (imagen tomada [.wAZOM2008]).

La figura 8, muestra la ejecución del sistema embebida en una sesión de teleconferencia access-grid, en el marco del proyecto de teleconferencia enriquecida [HER2007], [wIMAG2008b].

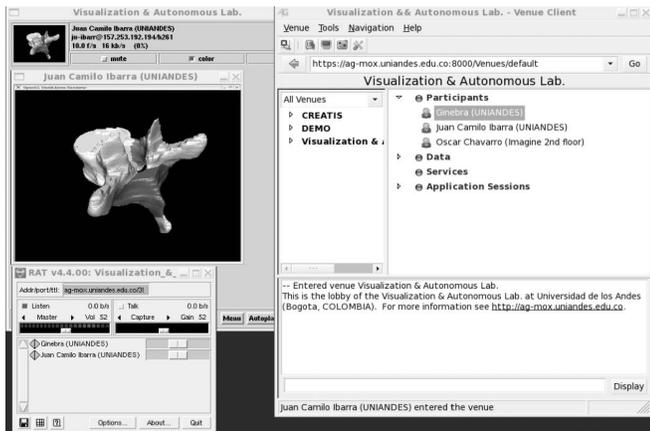


Figura 8. Ejemplo con AccessGrid

## V. CONCLUSIONES Y TRABAJO FUTURO

Se ha presentado una estructura de datos multinivel, multipuntos de vista que, junto con un sistema de navegación-visualización, constituyen un prototipo genérico para el manejo de estructuras complejas multiescala con diferentes puntos de vista. Las pruebas presentadas sobre modelos anatómicos, ilustran el valor agregado de la propuesta: el tratamiento natural de diferentes puntos de vista para diferentes usuarios, el desempeño en ambientes de telecolaboración y la capacidad de integración en plataformas de realidad virtual.

Se ha identificado un gran potencial en el uso de este tipo de sistemas para el soporte a la toma de decisiones en donde el objeto de interés es complejo. Por ejemplo, en el diseño arquitectural de edificios grandes, en donde las tuberías, las estructuras, el cableado eléctrico y de datos, el uso de espacios, la estética y muchos otros factores constituyen diversos temas sobre un solo modelo 3d complejo. En estos escenarios, el posibilitar una herramienta de trabajo cooperativo entre distintos perfiles de usuarios (arquitectos, ingenieros civiles, electricistas, decoradores de interiores, etc.) seguramente establecerá un gran aporte al quehacer profesional multidisciplinar.

Como trabajos futuros es importante mencionar tres frentes: La realización de pruebas de uso por parte de usuarios en campos específicos (ej: cerebro humano), ambientes de edición y mantenimiento de contenidos y por último, y tal vez el más retador para la estructura propuesta, la introducción de comportamientos dinámicos de los objetos representados.

## VI. RECONOCIMIENTOS

Este proyecto se ha beneficiado de la participación activa y crítica del profesor Pablo Figueroa, líder del proyecto InTml y

de Camilo Pérez y Darwin Martínez quienes participaron en las definiciones preliminares de la estructura de datos aquí propuesta.

Este sistema es resultado parcial del proyecto Colciencias: **1204-330-18871**, cofinanciado con la Facultad de Ingeniería de la Universidad de los Andes.

## VII. REFERENCIAS

[ARIZ2006] Ariza Núñez, Oscar Javier. "Escalabilidad y uso de técnicas foco más contexto en atlas médicos 3D", Tesis de Maestría, Universidad de los Andes, 2006.

[CLAR1976] Clark, James. "Hierarchical geometric models for visible-surface algorithms" *Communications of the ACM* Volume 19, Issue 10, Oct. 1976, pp. 547-554.

[FIGU2002] Figueroa, Pablo. Green, Mark. Hoover, H. James. "InTml: a description language for VR applications", *ACM Proceeding of the seventh international conference on 3D Web technology 2002*, pp. 53-58.

[FU2007] Fu, Chi-Wing. Hanson, Andrew J. "A Transparently Scalable Visualization Architecture for Exploring the Universe". *IEEE Transactions on Visualization and Computer Graphics*. Vol. 13 No. 1. January/February 2007. pp. 108-121

[HER2007] Hernández, JT, Díaz, E, Figueroa, P, De la Rosa, F, "El desarrollo de aplicaciones colaborativas de alta calidad: una realidad sobre la Red Académica de alto desempeño (Renata)". *Revista de Ingeniería*. , v.26, 2007.

[HOWA1996] Hubbard, R. Xiao D. Gibson, S. "MAVERIK - the manchester virtual environment interface kernel". In *Proceedings of 3rd Eurographics Workshop on Virtual Environments*, February 1996. Monte-Carlo.

[HUBB1999] Hubbard, R. Cook, J. Keates, M. Gibson, S. Howard, T. Murta, A. West, A. Pettifer, S. "GNU/MAVERIK: A micro-kernel for large-scale virtual environments." In *Proceedings of ACM Symposium on Virtual Reality Software and Technology*, December 1999. London. También disponible en la página de Advanced Interfaces Group - School of computer science - University of Manchester. <http://aig.cs.man.ac.uk/maverik/maverik.php> Navegada en junio 27 de 2005.

[HUMP2001] Humphreys, Greg. Eldridge, Matthew. Buck, Ian. Stoll, Gordon. Everett, Matthew. Hanrahan, Pat. "WireGL: a scalable graphics system for clusters", *ACM SIGGRAPH Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001. pp. 129-140.

[JUST1998] Just, Christopher. Bierbaum, Allen. Baker, Albert. Cruz-Neira, Carolina. "VR Juggler: A Framework for Virtual Reality Development". *Immersive Projection Technology Workshop*, 1998.

[MANT1988] Mäntillä, M. "An Introduction to Solid Modeling" *Computer Science Press*, Maryland, 1988

[RAO1994] Ramana Rao, Stuart K. Card, "The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information", *Proc. ACM Conf. Human Factors in Computing Systems*, CHI, pp. 318-322, 1994.

[ROHL1994] Rohlf, John. Helman, James. "IRIS performer: a high performance multiprocessing toolkit for real-time 3D graphics", *ACM Proceedings of the 21st annual conference on Computer graphics and interactive techniques SIGGRAPH 1994*. pp. 381 - 394.

[RUSS2001] Russell M. Taylor II, Thomas C. Hudson, Adam Seeger, Hans Weber, Jeffrey Juliano, Aron T. Helsen, "VRPN: A Device-

Independent, Network-Transparent VR Peripheral System," Proceedings of the ACM Symposium on Virtual Reality Software & Technology 2001, VRST 2001. Banff Centre, Canada, November 15-17, 2001.

[TELL1991] Teller, Seth. Séquiu, Carlo. "Visibility preprocessing for interactive walkthroughs". ACM SIGGRAPH Computer Graphics, Volume 25, Issue 4, July 1991. pp. 61-70.

[YEME2003] Yemez, Yucel. Schmitt, Francis. "Multilevel Representation and Transmission of Real Objects with Progressive Octree Particles", IEEE Transactions on Visualization and Computer Graphics, Vol. 9 No. 4, oct-dec 2003, pp. 551-569

[.wAG2008] "Access Grid", página web: <<http://www.accessgrid.org>> Último acceso, Miércoles, 16 de Enero de 2008.

[.wFIGU2008] Figueroa, Pablo. "SourceForge InTml project page", página web: <<http://sourceforge.net/projects/intml>>, Último acceso, Jueves, 28 de Enero de 2008.

[.wIMAG2008a] GRUPO IMAGINE "Proyectos del grupo de investigación imagine", página web: <<http://imagine2.uniandes.edu.co>>, Último acceso, Jueves, 28 de Enero de 2008.

[.wIMAG2008b] GRUPO IMAGINE "Plataforma colaborativa del grupo IMAGINE", página web: <<http://ag-mox.uniandes.edu.co/site>>, Último acceso, Jueves, 28 de Enero de 2008.

[.wOSG2008] "OpenSG", página web: <<http://www.opensg.org>>, Último acceso, Miércoles, 16 de Enero de 2008.

[.wVIS2008] National Library of Medicine. "The Visible Human Project", página web: <<http://www.nlm.nih.gov/research/visible>>, Último acceso, Miércoles, 16 de Enero de 2008.

[.wXSD2007] CodeSynthesis. "XSD: XML Data Binding for C++", página web: <<http://www.codesynthesis.com/products/xsd>>, Último acceso, Miércoles, 16 de Enero de 2008.

[.wAZOM2008] "Imagen de histología" Imagen tomada de [http://www.azom.com/work/LZa2TZ1lqZ22O09M4pCh\\_files/image010.jpg](http://www.azom.com/work/LZa2TZ1lqZ22O09M4pCh_files/image010.jpg), Último acceso, Febrero 25 de 2008.

**Juan C. Ibarra López.** Recibió el título de Ingeniero de Sistemas y Computación de la Universidad de los Andes (Bogotá, Colombia) en el 2007. Actualmente se encuentra estudiando Maestría en Ingeniería de Sistemas en la Universidad de los Andes, Bogotá, Colombia. Su campo de Investigación es el de Computación Visual más específicamente el de ambientes inmersivos de aprendizaje. A partir de septiembre de 2007 es asistente Graduado en el proyecto de Investigación "Diseño y construcción de un sistema de visualización inmersiva, escalable y de bajo costo aplicado al entrenamiento y al entretenimiento educativo basados en imágenes y modelos 3D de órganos humanos" financiado por Colciencias - Instituto Colombiano para el Desarrollo de la Ciencia y la Tecnología "Francisco José de Caldas" en la Universidad de los Andes, Bogotá, Colombia.

**Oscar Javier Chavarro García.** Recibió el título de Ingeniero de Sistemas y Computación de la Universidad de los Andes (Bogotá, Colombia) en 1997, y MSc. en Computación Gráfica en 1998. Trabajó en la industria de telecomunicaciones y como profesor asistente en el área de Computación Gráfica en la Pontificia Universidad Javeriana. Actualmente se encuentra en su primer año de estudio doctoral en la Universidad de los Andes. Es miembro de la IEEE Computer Society y la ACM SIGGRAPH. Es el autor de Vitral Software Development Kit y la plataforma AQUYNZA.

**David Antonio Castro Martínez.** Recibió el título de Ingeniero de Sistemas y Computación de la Universidad de los Andes (Bogotá, Colombia)

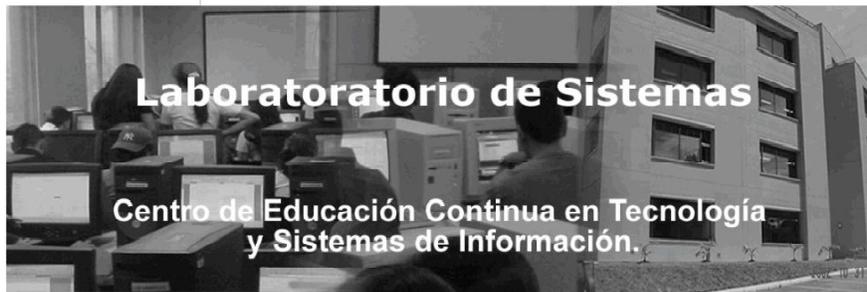
en 2007. Actualmente se encuentra estudiando Maestría en Ingeniería de Sistemas en la Universidad de los Andes, Bogotá, Colombia. Su campo de Investigación es el de Computación Visual más específicamente el de tecnología móvil. A partir de septiembre de 2007 es asistente Graduado en el proyecto de Investigación "Diseño y construcción de un sistema de visualización inmersiva, escalable y de bajo costo aplicado al entrenamiento y al entretenimiento educativo basados en imágenes y modelos 3D de órganos humanos" financiado por Colciencias - Instituto Colombiano para el Desarrollo de la Ciencia y la Tecnología "Francisco José de Caldas" en la Universidad de los Andes, Bogotá, Colombia.

**José Tiberio Hernández Peñaloza.** Ingeniero de Sistemas y Computación y MSc. de la Universidad de los Andes, Doctor Ingeniero del ENSTA-Paris en 1983. Actualmente es profesor Asociado y Director del grupo IMAGINE de la Universidad de los Andes en Bogotá. Sus áreas de interés son la Computación Visual (Modelado geométrico, ambientes colaborativos de apoyo a la toma de decisiones, Imagenología Médica), y la Innovación en Ingeniería.

# Universidad Nacional de Colombia Sede Medellín

## Facultad de Minas

**120 años**   
TRABAJO Y RECTITUD



### Misión

Ofrecer servicios de apoyo a la docencia en cuanto a la operación de computadores y del software adecuado con miras al desarrollo integral de los futuros ingenieros.

### Visión

Avanzamos en la búsqueda de convertir el Laboratorio de Sistemas e Informática en una dependencia ágil, moderna, facilitadora de procesos y cambios, atenta a las necesidades de otras dependencias de la Universidad, cuya labor apoyamos y articulamos. Serviremos con dinamismo, amabilidad y efectividad a todos los integrantes de la comunidad universitaria y a la sociedad en general. Uniremos esfuerzos para construir un ambiente de trabajo cada vez más positivo que propicie la participación, la creatividad y el desarrollo profesional de los integrantes del equipo de trabajo. Propondremos por un Laboratorio como instrumento gestor y generador de proyectos de investigación sustentado en un equipo interdisciplinario de trabajo en torno a la informática aplicada a la ingeniería.

### Cursos

- Lenguajes de Programación: Diseño de Páginas Web en ASP.NET con VB.NET, Programación Web PHP y MYSQL, MS-Visual Basic Básico y Avanzado, Java
- Generales: Latex, ARCGIS, MS -Office (Word, Excel y PowerPoint), Excel Financiero, Excel Avanzado, Mantenimiento de Hardware y Software Niveles I y II, MS -Access Básico, MS -Project Básico (Programación y Gerencia de Proyectos), AUTOCAD 2D Básico y 3D, Matlab, Moodle de Apoyo a los Procesos de Enseñanza y Aprendizaje.

Para mayor información, por favor comunicarse a los teléfonos: 4 255313, 4255312, 4255355. Calle 59A No. 63 – 020 Medellín (Colombia), Bloque M7, quinto piso.

Email: [labsis@unalmed.edu.co](mailto:labsis@unalmed.edu.co)

<http://xue.unalmed.edu.co/cursos>

