

Integración de Simuladores de Vuelo y Bases de Datos Geográficos

Alexandra Diehl^{1,3}, Horacio Abatte¹, Claudio Delrieux², Juliana Gambini³

¹ CITEFA (Centro de Investigaciones Científicas y Técnicas para la Defensa),
Buenos Aires, Argentina

² Departamento de Ing. Eléctrica y Computadoras, UNS, Bahía Blanca, Argentina

³ Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Ciudad
Universitaria Pab. I, Universidad Nacional de Buenos Aires, Buenos Aires, Argentina
{adiehl, jgambini}@dc.uba.ar
habbate@citefa.gov.ar
cad@uns.edu.ar

Resumen El objetivo de este trabajo consiste en la integración de bases de datos geográficos y modelos de entornos topográficos dentro de la tecnología actual de simuladores de vuelo. Pese a los beneficios evidentes de esta integración, son muy pocos los desarrollos actualmente publicados en esta dirección. El proceso de integración implica la definición de interfaces estándares para integrar diferentes fuentes de información geográfica dentro del modelo de entorno topográfico. Luego, la visualización y edición del entorno topográfico, y finalmente la preparación de la escena para el simulador de vuelo. La utilización de sistemas de bases de datos geográficos junto a la utilización de placas gráficas modernas ofrece una oportunidad para el desarrollo de aplicaciones interactivas que manejen datos geoespaciales en cantidades masivas, la exploración de base de datos geoespaciales, y su visualización en tiempo real, con la ventaja adicional de poder realizarse con computadoras personales de bajo costo.

Key words: Simuladores de vuelo, base de datos geográficos, datos geoespaciales, visualización geográfica.

1. Introducción

El objetivo de este trabajo consiste en la integración de bases de datos geográficos y modelos de entornos topográficos dentro de la tecnología actual de simuladores de vuelo. De esa manera, es posible construir simuladores de entrenamiento con la capacidad de emular misiones en espacios geográficos reales, fusionar la información topográfica con otras bases de datos geográficas, etc.

Esto ofrece un significativo potencial en cuanto a las capacidades de visualización de los datos. La visualización geográfica consiste no solo en mostrar datos geoespaciales, sino en la posibilidad de que el usuario pueda indicar adonde ir y que mirar. Ésta habilidad puede ser alcanzada a través del uso de simuladores de vuelo dado que permiten la visualización dinámica de datos geográficos en

escenarios 3D realísticos. Pese a los beneficios evidentes de esta integración, son muy pocos los desarrollos actualmente publicados en esta dirección.

El proceso de integración consta de tres partes principales: la definición de interfaces estándares para integrar las diferentes fuentes de información geográfica dentro del modelo de entorno topográfico. Luego, la visualización y edición del entorno topográfico, y finalmente la preparación de la escena para el simulador de vuelo, es decir, la exportación de la información de elevación de terreno y objetos de la escena a un formato interno, propio del simulador.

Este proceso representa hoy un desafío a nivel tecnológico, porque implica el manejo de información topográfica en grandes volúmenes, y se requiere poder modelar y visualizar extensas áreas geográficas que permitan la simulación de actividades aéreas durante horas.

Desde este punto de vista, la utilización de placas gráficas modernas ofrece una oportunidad para el desarrollo de aplicaciones interactivas que manejen datos geospaciales en cantidades masivas, la exploración de base de datos geospaciales, y la visualización en tiempo real, con la ventaja adicional de poder realizarse a costos económicos debido al bajo valor que tienen las computadoras personales, hoy en día, en el mercado.

Para la integración de la información geográfica, la Open Geospatial Consortium (OGC) propone un conjunto de documentos técnicos englobados en estándares y especificaciones OpenGIS cuyo objetivo principal es generar interfaces espaciales y especificaciones de codificación que sean públicas y abiertas para uso global de todo el mundo. OpenGIS, implementó Geography Markup Language (GML), como lenguaje estándar para transporte e intercambio de información. GML es una gramática basada en el estándar eXtensible Markup Language (XML), que utiliza los esquemas XML para describir los modelos de transporte y almacenamiento de información geográfica.

Por otro lado, Google ha desarrollado para su producto Google Earth su propio lenguaje para intercambio y transporte de información geográfica: Keyhole Markup Language (KML). KML es un lenguaje focalizado en la visualización de datos geográficos, incluyendo anotaciones en los mapas e imágenes. Actualmente KML 2.2 utiliza ciertos elementos geométricos derivados de GML 2.1.2. La OGC y Google han acordado la armonización de KML y GML para utilizar la misma representación geométrica. Como fruto de este acuerdo, la OGC ha adoptado como estándar a KML en su versión 2.2.

La comunidad KML es grande y variada. Usuarios casuales pueden agregar referencias geográficas de sus propias viviendas, describir recorridos, itinerarios de trabajo, o aventura. Científicos utilizan KML para representar en forma precisa recursos, modelos y analizar erupciones volcánicas, patrones climáticos, actividad sísmica, etc. Organizaciones como Nacional Geographic, y la UNESCO, y muchas otras organizaciones utilizan KML para compartir información a través de Internet, como servicios, o a través de otros medios electrónicos.

La integración de diferentes fuentes de datos geográficos en una única base de datos representa un importante desafío, amplificado por la creciente comunidad científica y aficionada que publica información valiosa en Internet, dado que

implica el diseño de un modelo lo suficientemente genérico y extensible para poder incorporar información heterogénea, y a su vez la definición de interfaces simples y estándares.

Este es el marco en el que se circunscribe el presente trabajo. El mismo se encuentra dentro del proyecto Simulador de Vuelo (PAE2004-22614 ANPCyT) financiado por el subsidio PAE2004-22614 ANPCyT (Agencia Nacional de Promoción Científica y Tecnológica), y parcialmente financiado por SECyT-UNS (Secretaría de Ciencia y Técnica de la Universidad Nacional del Sur).

2. Trabajos Previos

Existen numerosas aplicaciones para visualizar y editar modelos de entornos topográficos y datos geoespaciales, pero generalmente los datos están presentados en forma estática, a través de mapas, u otras representaciones en 2D. Son pocos los trabajos en que se logran visualizaciones realísticas 3D y que permitan la exploración de los datos, y menos aún utilizando simuladores de vuelo. La tecnología de simuladores de vuelo, primero desarrollada con fines militares y en la actualidad comúnmente utilizada en la industria del entretenimiento, ofrece una mejor visualización y exploración de los entornos topográficos.

Se pueden citar los trabajos de: McDonald, S. y Stevenson, Robert. D. [10] que han desarrollado un sistema para visualización geográfica de patrones del paisaje, ecosistemas, y biodiversidad incorporando un simulador de vuelo para explorar y volar regiones geográficas; el trabajo de Pazner, M., Ross, M. y Ripley, N. [11] sobre exploración y reconocimiento aéreo de patrones geográficos (virtual reconnaissance) mediante la integración de sistemas de información geográfica y simuladores de vuelo; e incluso a Google[12] que integró un simulador de vuelo como una nueva característica de su aplicación Google Earth para volar sobre el globo terrestre.

Hoy en día, existe una amplia variedad de simuladores de vuelo civiles y militares, comerciales y de código abierto, y cada uno de ellos tiene su modelo de entorno topográfico propio. Dentro de los simuladores de vuelos existentes, se eligió y realizó un estudio de tres aplicaciones: FlightGear 1.0[3][4], X-Plane 7.3[6], y Microsoft Flight Simulator 2004[5] y el estándar OpenFlight[1][2], en su versión 16.3, para armado de escenas para simuladores de vuelo.

FlightGear. FlightGear Flight Simulator es un proyecto de desarrollo open-source, multiplataforma y cooperativo. El modelo topográfico de FlightGear divide la topografía en mosaicos con los límites paralelos a las líneas de latitud y longitud. Un mosaico es en si mismo un escenario. La estructura de la escena de FlightGear se puede dividir en tres partes: información de elevación del terreno, información geométrica de aeropuertos; y otro tipo de objetos que podrían aparecer en la escena.

Microsoft Flight Simulator (MSFS). MSFS es un simulador de vuelos civiles comercializado por Microsoft. El modelo topográfico de MSFS divide al

plano terrestre en mosaicos paralelos a las líneas de latitud y longitud. Cada mosaico es construido a partir de un conjunto de texturas: texturas de agua, texturas de terreno, y objetos que están sobre el terreno, modelados a partir de estructuras geométricas (Texture ground polygons).

X-Plane. X-Plane es un simulador de vuelos comercializado por Laminar Research. X-Plane divide el plano terrestre en una grilla formada por líneas paralelas a las líneas de latitud y longitud de la Tierra, y organiza la información de su modelo topográfico según tres tipos de información: información de entorno (datos geográficos del terreno; y ubicación y propiedades de los objetos que se encuentran sobre el terreno), información de aeropuertos (tipos de aeropuertos, elevación, número de pistas, etc.), e información de radiofaros, balizas no direccionales, y otros señaladores aéreos.

OpenFlight. El formato OpenFlight es un estándar de facto industrial para modelado de datos geoespaciales para escenas 3D, usado ampliamente en simuladores de vuelo. También OpenFlight, adopta para su estructura la división de la Tierra en mosaicos cuyos límites se corresponden a líneas de latitud y longitud. Cada mosaico tiene asociada una o más grillas reticuladas correspondiente a los diferentes niveles de detalle que tenga especificado. Un escenario consiste de uno o más mosaicos de terreno.

El formato OpenFlight consiste en una estructura jerárquica en forma de árbol que describe los componentes de la escena. La estructura de árbol consiste de diferentes tipos de nodos, como se puede observar en la Fig. 1.

El nodo *container* contiene a otros nodos hijos y le otorga cierta lógica sobre el conjunto. El nodo *geometry* representa una figura geométrica renderizable, y generalmente forma parte de un nodo *container*. El nodo *vertex* representa los bloques de construcción de los nodos *geometry*. Cada nodo *container* tiene asociado un volumen mínimo *bounding volume* que permite determinar si un sub-árbol de la escena está dentro de una vista definida. Las vistas pueden ser cajas, esferas o cilindros. El volumen debe encerrar al nodo al que está aplicado y todos sus hijos.

Como conclusión de los modelos analizados, se observa que tienen en común la división de la Tierra en una grilla uniforme, y la representación de entidades orientadas a modelar: información del terreno, objetos que pueden adoptar una ubicación dentro del entorno topográfico, por ejemplo edificios, balizas, radiofaros, etc.; y entidades orientadas a modelar objetos que por su naturaleza no pueden asociarse con una única posición, pero sí son objetos que se ubican en el entorno topográfico, como ser carreteras, límites, costas, etc.

Estos modelos se diferencian en el enfoque utilizado para representar las entidades que conforman el modelo de entorno topográfico. En el caso de Microsoft Flight Simulator, FlighGear y X-Plane definen y clasifican a las entidades con una semántica propia que permite diferenciarlos dentro del modelo. En el caso de OpenFlight el enfoque para modelar la información es genérico y permite

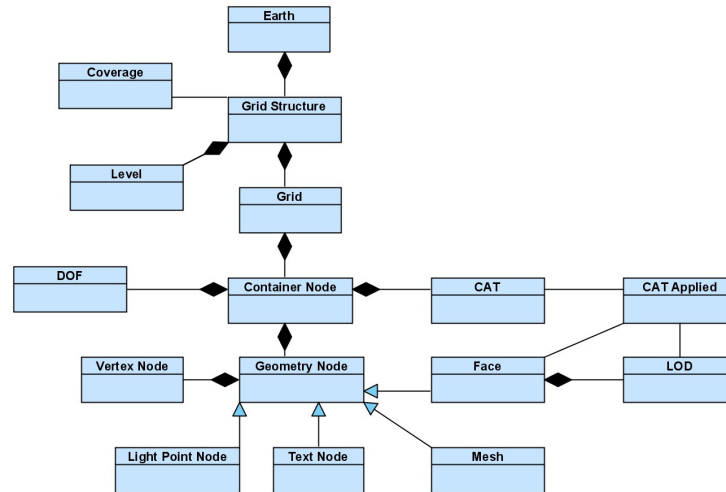


Figura 1. Diagrama de clases simplificado de la especificación *OpenFlight*.

representar a las entidades como una jerarquía de nodos y relaciones entre nodos. En el trabajo que aquí se aborda, se consideró importante lograr un diseño que permita su aplicación, no solo a simuladores de vuelo, sino a una amplia gama de simulaciones: terrestres, marítimas, fluviales, urbanas, etc. Asimismo, se consideró como requerimiento esencial que las soluciones desarrolladas sean coherentes con los marcos regulatorios y estándares internacionales que rigen la industria y el uso de simuladores de vuelo, así como las metodologías de diseño, desarrollo y validación de software.

3. Integración de Simuladores y Bases de Datos Geográficos

Se planteó la necesidad de construir un modelo de dominio para representar entornos topográficos con independencia de su extensión, que sea apropiado para el modelado a escala global de toda la superficie terrestre y con la capacidad de establecer un sistema de referencia propio del entorno topográfico, y que permita el modelado de regiones de existencia real o regiones artificiales y sea optimizado para ser accedido en tiempo real.

3.1. Diseño

Se diseñó un modelo propio inspirado en el modelo que propone la especificación OpenFlight, con contribuciones de los otros modelos de entornos topográficos analizados.

Información topográfica. En la Fig. 2 se puede observar el modelo elegido para representar la información topográfica. Nuestro modelo divide la Tierra (*Earth*) en una retícula con límites paralelos a las líneas de latitud y longitud, cada división se corresponde con un área (*Area*), cada área tiene asociada una extensión de superficie cubierta (*Coverage*), un sistema de referencia espacial (*spatial_ref_sys*), y la información de terreno (*Terrain*) (latitud, longitud y elevación de terreno). El modelo soporta diferentes niveles de detalle (*Level*), dividiendo cada área en una o más grillas reticuladas (*Grid*) con diferente número de divisiones dependiendo del nivel. El mayor nivel de detalle se corresponderá exactamente con la información de elevación de terreno (*Terrain*) más detallada que se dispone para esa área. Cada celda (*Tile*) de la retícula guarda la información de posición inicial en coordenadas geográficas originales (latitud mínima, longitud mínima) y en coordenadas proyectadas (x mínimo, y mínimo), según la proyección elegida, por ejemplo: *Universal Transverse Mercator (UTM)*.

Modelado de Objetos de la Escena. Cada área tiene asociada una lista de modelos (*Model List*) que representan los objetos que están sobre el terreno, y componen la escena (para este trabajo sólo hemos modelado objetos que pueden ser posicionados puntualmente en el espacio). Como se puede observar en la Fig. 3, un modelo (*Model*) es un objeto de alto nivel que puede estar compuesto por una o más partes (*Component*). Cada *Component* tiene asociado un sistema de coordenadas locales (*Local CoordSys*), una caja o volumen mínimo que lo encierra (*Bounding Box*), y una referencia a su componente padre en caso de que sea un objeto compuesto por varias partes.

La entidad *Local CoordSys* especifica el sistema de coordenadas local del componente, que puede ser compartido por más de un componente. La entidad *Bounding Box* permite definir el volumen mínimo que contiene a todo el componente y a sus hijos, en el caso que sea un objeto compuesto. Los componentes se forman con estructuras geométricas (*Geometry*) que pueden ser caras (polígonos cerrados), mallas, o puntos de iluminación. Cada *Geometry* está formada por uno o más vértices (*Vertex*) que contienen la posición del punto en el espacio en coordenadas locales al objeto al que pertenecen. La entidad *Vertex* guarda los vértices de cada componente que son compartidos por cada una de las caras (*Face*) a través de la entidad *Geometry Vertex*. El modelo incorpora al diseño el concepto de nivel de detalle de los objetos a través de las entidades *Skin*, *Skin Applied* y *Level Of Detail (LOD)*. Cada componente tiene asociado una o más apariencias (*Skin*) que guardan las texturas asociadas a cada nivel de detalle (*LOD*) que luego serán aplicadas a cada cara (*Face*) según la distancia al punto de vista del observador.

4. Implementación

Los DBMS relacionales con soporte para datos geoespaciales, Geospatial Information System (GIS), intentan proveer un modelo que soporte tanto datos geoespaciales: puntos, líneas, polígonos, etc.; como datos no geoespaciales: campos

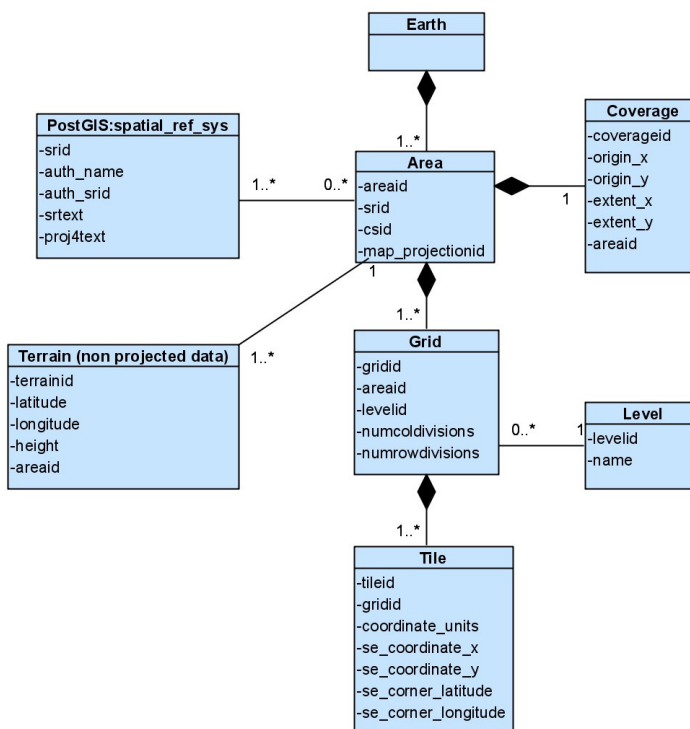


Figura 2. Segmento del modelo de entorno topográfico adoptado que representa la información topográfica.

de texto, enteros, etc., ampliando las posibilidades de explotación de la información.

Se optó por la utilización de un DBMS con soporte GIS para la construcción del modelo de datos. Dentro de los DBMS existentes en el área que soportan información geográfica, se resolvió la utilización de PostGIS[9] por ser un software libre y abierto, y estar alineado con los estándares OpenGIS Simple Feature Specification. PostGIS permite almacenar objetos geográficos de acuerdo a los estándares de OpenGIS[7] y validar que estén correctamente definidos.

Se analizaron los diferentes esquemas de aplicación de GML para mantener alineado el modelo a los estándares de OpenGIS. Para implementar estos esquemas conceptuales, GML provee dos opciones: GML schemas, que son esquemas que siguen una plantilla y tienen una sintaxis cerrada; y GML profiles que tienen una semántica y sintaxis dependiente de la aplicación que los utilice. En el presente trabajo se ha adoptado el profile “GML Point profile“ que utiliza únicamente el objeto geométrico <gml:Point> para describir las interfaces de datos entre las fuentes de información y nuestro modelo.

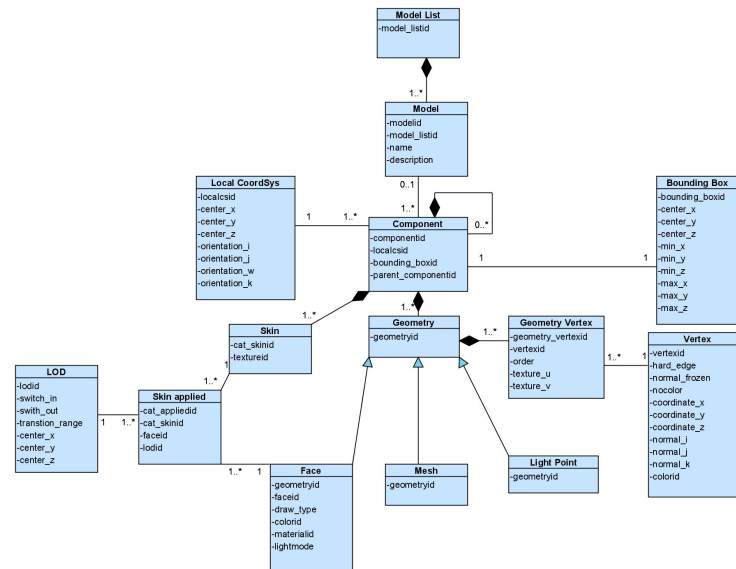


Figura 3. Segmento del modelo de entorno topográfico adoptado que representa el modelado de objetos de la escena.

5. Resultados

Desde el punto de vista del diseño se ha alcanzado un modelo:

1. Genérico: dado que permite modelar objetos simples y objetos compuestos.
2. Flexible: dado que permite especificar el sistema de referencia espacial global que se utilizará para la base de datos (sistema de coordenadas global, Datum, y proyección); y un sistema de coordenadas local para cada objeto a modelar. También permite especificar los distintos niveles de detalle que se utilizarán para cada área representada.
3. Extensible: dado que permite incorporar nuevas geometrías que hereden de la entidad *geometry* sin modificar las entidades existentes ni sus relaciones.

En la Fig. 4 se puede observar la visualización de un área de terreno representada utilizando el modelo de entorno topográfico implementado. El área corresponde a una retícula de terreno en coordenadas proyectadas Universal Transverse Mercator UTM, Datum=WSG84, zone=12 de 169 filas y 262 columnas de 30 metros de lado, con unidad de coordenadas en metros. La misma tiene coordenadas de latitud y longitud inicial de 247.038,10603026 metros, y 4.205.582,3212307 metros respectivamente.

En la Fig. 5 se puede observar un detalle de la grilla y los objetos representados. Se han posicionado tres objetos sobredimensionados sobre el terreno para que puedan ser observados a simple vista en conjunto con toda el área cubierta.

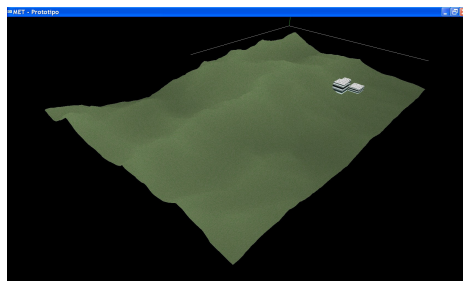


Figura 4. Visualización de la superficie de área cubierta representada.

Cuadro 1. Posición de los objetos sobre la escena. La unidad de medida se encuentra en metros.

Model Id	Center X	Center Y	Center Z	Lado	Altura
1	250.608,10603	3.522	4.207.682,321231	200	300
2	250.918,10603	3.522	4.207.682,321231	300	200
3	250.608,10603	3.522	4.207.992,321231	300	400

La tabla 1 muestra el centro de coordenadas proyectadas (x,z) y la altura (y) de cada uno de los objetos. A cada coordenada proyectada (x,z) se le ha asociado un tipo de dato geométrico POINT(x,y) nativo de PostGIS con su correspondiente sistema de referencia espacial, para poder realizar las consultas espaciales. Lo mismo se ha realizado para el terreno. La utilización de PostGIS como soporte para los datos geoespaciales permitió especificar datos geométricos, y validar que los mismos se encuentren siempre dentro de los estándares de OpenGIS haciendo más robusta la implementación y consulta de la base de datos geográfica; y a su vez aprovechar las funcionalidades específicas para manejo de datos geoespaciales, conversión entre sistemas de coordenadas e indexado espacial con el fin de mejorar el rendimiento de las consultas y recuperación de datos.

6. Conclusiones y trabajos futuros

Se construyó un modelo de entorno topográfico genérico, flexible y extensible, con contribuciones de los otros modelos de entorno topográfico analizados, y que permite su integración con simuladores de vuelo. Se analizaron los estándares y tecnologías existentes para manipulación de los datos geográficos, y se resolvió la utilización de un DBMS objeto relacional alineado con los estándares OpenGIS.

En trabajos futuros nuestros esfuerzos estarán puestos en agregar KML a nuestro modelo como interfaz de intercambio de datos, dado que representa una valiosa herramienta para integrar información de diferentes fuentes, aprovechando el trabajo realizado por las organizaciones existentes que ya colaboran entre

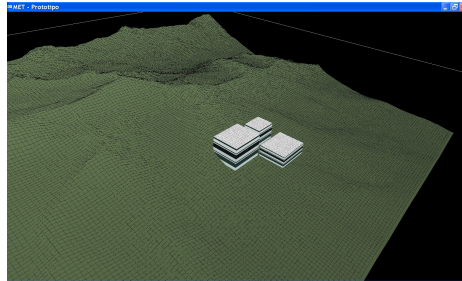


Figura 5. Visualización de una porción de la superficie de área cubierta. Vista ampliada y reticulada.

si, a través del mismo. También extender el modelo para objetos que estén posicionados sobre el terreno, pero que no tengan una ubicación única puntual: ríos, carreteras, etc.; y desarrollar herramientas que permitan la visualización y edición del modelo de entorno topográfico, así como exportar las escenas al formato propio para el simulador de vuelo; y explorar nuevas fuentes de información geográfica que permitan la integración de ambientes de operaciones a nivel conjunto (aire, mar y tierra) y combinado (países de la región).

Referencias

1. MultiGen-Paradigm, Inc.: OpenFlight Scene Description Database Specification. Version.16.3. Document revision A. (2007)
2. MultiGen-Paradigm, Inc.: Creator Terrain Studio. Terrain generation innovation from MultiGen-Paradigm, Inc. Version 1.2. (2005)
3. Olson, C.: FlightGear Scenery Generation Tools, <http://www.flightgear.org/docs/scenery/scenerygeneration/scenerygeneration.html>
4. Basler, M., Spott, M., Olson, C.: The FlightGear Manual. Version 1.0. (2007)
5. Microsoft Flight Simulator 2004 Terrain SDK: Using Textured Ground Polygons and Land/Water Mask Polygons. (2004)
6. Laminar Research.:X-Plane 8 Scenery Information Library, <http://scenery.x-plane.com/library.php>. (2007)
7. Percivall, G., Open Geospatial Consortium Inc.: OGC Reference Model Version 0.1.3. Reference number: OGC 03-040. (2003)
8. Open Geospatial Consortium Inc.: OpenGIS® Geography Markup Language (GML) Encoding Standard. Version 3.2.1. Reference number: OGC 07-036. (2007)
9. PostgreSQL version 8.2 Documentation. PostGIS Manual. Version 1.1.6
10. McDonald, S., Stevenson, Robert D.: Visualizing Geo-referenced Data with the Eco Flight Simulator