

## OCHO CUESTIONES CRUCIALES EN LA CREACIÓN DE MODELOS VIRTUALES DE GRANDES DIMENSIONES

JAVIER MONEDERO ISORNA

Profesor del Departamento de Expresión Gráfica Arquitectónica I

Universidad Politécnica de Cataluña

Barcelona - España

[www.upc.edu/ega1](http://www.upc.edu/ega1)

---

### Introducción

Entre 2000 y 2003 aproximadamente, el Laboratorio de Modelización Virtual de la Ciudad de la ETSAB ha trabajado en una serie de proyectos que han supuesto una experiencia sin precedentes en las escuelas de arquitectura de España, en una vía que ya ha sido recorrida por laboratorios más avanzados de otros países. Entre los principales proyectos de este período cabe citar, el modelado del Paseo de Gracia de Barcelona, la reconstrucción virtual de la Ciudad de Cádiz en el siglo XVIII y el modelado del recinto general del Forum 2004. A estos hay que sumar toda una serie de proyectos de menor envergadura pero que han servido también como una experiencia enormemente valiosa.

A lo largo de este recorrido, sobre todo al comienzo, se han analizado extensamente otros trabajos previos. Como sería de esperar han aparecido muchas coincidencias, la mayoría previsibles y ciertas diferencias. Algunas de estas diferencias pueden ser atribuidas, quizás, a limitaciones técnicas, aunque la rapidez de la evolución tecnológica suele jugar casi siempre a favor de los recién llegados. Otras, las más interesantes, son debidas a distintos enfoques o estrategias y se prestan a una comparación instructiva entre metodologías de trabajo y a consideraciones generales sobre las intenciones que orientan la elaboración de los modelos, sea debido a los tipos de encargo, sea debido a otras causas.

Estas semejanzas y diferencias se han resumido en ocho cuestiones que, en mi opinión, suponen en la actualidad otras tantas encrucijadas en las que hay que tomar decisiones no triviales. A lo largo de la presentación, y en función del tiempo disponible, se comentarán aspectos concretos ligados al desarrollo de los proyectos mencionados.

En dichos proyectos han colaborado, principalmente, Francisco Muñoz, responsable del funcionamiento general del Laboratorio, Andrés Lupiáñez, Marc Pujol, Carolina Ruiz, Andreína Linares y Héctor Zapata, además del autor de la ponencia como director del Area de Modelado del Laboratorio.

### 1. Adquisición de datos y organización de la información

Es obvio que no se puede crear un modelo, grande o pequeño, sin contar con información adecuada. Pero muchos de los problemas que surgen, a corto o medio plazo, en la generación de grandes modelos, derivan de que no se ha reconocido toda la importancia que tiene esta fase del proceso, tanto por lo que respecta a la obtención de los datos como a su tratamiento posterior para unificarlos en un todo coherente.

Podemos clasificar los datos de varios modos. Los que interesan para crear un modelo son datos gráficos, aunque los datos textuales o numéricos también pueden aportar información relevante. La información gráfica digital se concreta en dos tipos de formatos: vectores (líneas, polígonos) y mapas de bits (imágenes). Los datos vectoriales requieren, en principio, mediciones precisas y, en consecuencia, sólo pueden obtenerse de dos modos: si el cliente (público o privado) los suministra o si hay un encargo específico para obtener estos datos con suficiente precisión. Hay una tercera posibilidad, muy enojosa y con la que nos hemos encontrado en algún caso y es que el cliente se comprometa a suministrar algo que, cuando llega, no reúne las condiciones adecuadas.

Desde el punto de vista de su tratamiento y organización, podemos considerar que los datos gráficos pueden ser de dos tipos: a) datos no estructurados, tal como nubes de puntos obtenidas con un escaner3D, ortofotos o mapas de diversos tipos, fotografías pedrestres o aéreas, etc.; b) datos estructurados, básicamente vectores o polígonos.

Si sólo se cuenta con datos no estructurados estos necesitan ser convertidos de algún modo a datos estructurados para construir un modelo. Hay diversas posibilidades. Las nubes de puntos pueden ser procesadas para obtener caras poligonales, las fotografías pueden procesarse para obtener bordes significativos y, de ahí, vectores en un plano. Y varias fotografías vectorizadas pueden combinarse entre sí para derivar vectores 3D y configuraciones espaciales más o menos exactas. Pero también hay métodos más sencillos. Pueden hacerse levantamientos semimanuales a partir de imágenes, por medio de rectificaciones (ortofotos) simples, si se cuenta con alguna medida de referencia y se aceptan tolerancias que pueden llegar a ser del orden de unos 10 cms. Si la finalidad del modelo es visualizar un proyecto, puede ser suficiente contar con medidas generales (ejes de calles, anchuras y alturas globales de bloques) para hacer estas estimaciones. Aquí intervienen los datos textuales o numéricos pues es posible que aporten información relevante.

Dado que la información sobre el territorio es una condición previa para cualquier trabajo de modelado, la única observación que merece la pena hacer al respecto es que se debe contabilizar en el presupuesto una partida específica para adquirir este tipo de datos que, por lo general, se proporcionan, en un formato adecuado, por parte de instituciones públicas o por parte de quienes hayan encargado el modelo. Pero la información sobre la altura y el volumen de los edificios no suele estar disponible de este modo. Por esta razón, dentro de este apartado ocupan un lugar especial los métodos de levantamiento de fachadas. Hay métodos automáticos en donde se están produciendo avances espectaculares, pero aún insuficientes, en los últimos años y que merece la pena comentar.

En primer lugar, la obtención de nubes de puntos por medio de escáners 3D, que está a la espera de métodos que permitan procesarlos de un modo realmente eficiente para que puedan ser convertidos con rapidez en datos estructurados, algo que aún deja bastante que desear. En segundo lugar, los métodos de modelado automático de fachadas a partir de la detección de rasgos sobresalientes y el emparejamiento automático de varias imágenes (*feature based modeling*) que están avanzando notablemente aunque siguen dependiendo de tomas desde todos los ángulos lo que (dado que es difícil contar con fotografías áreas a corta distancia) redundan en pérdidas considerables de precisión en los pisos superiores (hasta medio metro, en las pruebas que hicimos hace unos tres o cuatro años con Photomodeller). Nuevos métodos, que incluyen precisión de subpixel (*subpixel accuracy*) podrían apuntar la vía para solucionar este problema. Por ahora hay que recalcar el hecho de que los métodos manuales suelen ser, con diferencia, más rentables en muchos casos.

Dicho todo esto también hay que añadir que es preciso distinguir con claridad las diferentes fases del trabajo. La adquisición de datos es un encargo específico que, en principio, es independiente del modelado y la simulación, y debe ser contabilizado como un encargo diferenciado. Esto es realmente crucial desde el punto de vista económico. Si no hay información suficiente desde el comienzo, el proceso quedará detenido más pronto o más tarde hasta que esa información se obtenga o se obtendrán resultados de baja calidad. Y puede ser difícil convencer a quien ha hecho el encargo, una vez que este se ha iniciado, de que esto es debido a una información insuficiente.

## **2. Tipo de modelo y estructura básica de los datos**

Un modelo es una simplificación, que preserva los rasgos más relevantes para la finalidad a la que está destinado. Si el objetivo principal es la visualización tan sólo se necesita modelar la piel, la superficie de los objetos. Toda otra información adicional supone una carga innecesaria y, a menudo, una fuente de problemas. Si el objetivo principal fuera el análisis estructural tendríamos la situación inversa, lo que se requiere es una subdivisión adecuada del volumen que pueda asociarse a diferentes acciones mecánicas (como en el método de elementos finitos). Si el objetivo principal es la producción, lo que se requiere es una subdivisión precisa en subelementos por medio de operaciones que puedan ser replicadas mediante métodos numéricos u operaciones bien definidas. No es posible, por ahora, crear un modelo que satisfaga todos estos requisitos y es dudoso que pueda llegar a serlo pues un modelo, por definición, es “una abstracción interesada”.

La finalidad principal, en nuestro caso, es la visualización, es decir, la obtención de un modelo que consista en una superficie continua de topología más o menos compleja y espesor cero. En un modelo de estas características los datos se estructuran por medio de una base interna, jerárquica, que está optimizada para ser recorrida con rapidez pero no para obtener información derivada, como ocurre en una base relacional.

La organización de los datos de un sistema de este tipo es muy limitada y, como veremos en el último punto, resulta problemática a la hora de relacionarla con otros sistemas. Dentro de esta limitación caben diversas estrategias.

El caso más sencillo y cuya estructura viene dada en buena medida externamente, es el de un modelo del territorio en que todos los elementos tienen una estructura similar: un par de coordenadas X, Y, más una coordenada Z, sin que existan plegamientos internos. En algunos de los modelos desarrollados en el Laboratorio se partía de la existencia de un DTM (Digital Terrain Model) y de ortofotos correspondientes a la misma subdivisión. Los datos principales son, así, rectángulos con dos coordenadas UTM que representan la esquina superior izquierda y la inferior derecha de tal modo que WN (x1,y1) y ES (x2,y2) representarían un segmento de territorio de  $(x2-y2)$  por  $(y1-y2)$  metros. La coordenada z viene dada por el DTM. Estos rectángulos se pueden poner en relación con mapas de bits (las ortofotos) de tal modo que la resolución resultante es de 1 pixel = 2.5 metros.

En un segundo estadio, podemos proyectar un entorno urbano sobre una estructura de este tipo (o, en casos más simples, con una resolución más tosca, dar por bueno el mismo procedimiento aunque incluya edificios) para obtener un DTM que incluya la huella de edificios y calles. Las caras poligonales resultantes pueden extraerse y pueden diferenciarse, como objetos o capas separadas, para recibir mapas adecuados. Esto sería un ejemplo, a gran escala, de un sistema de subdivisión de superficies que implica una cierta reestructuración jerárquica de datos dentro de los límites de un modelo de estas características.

En un tercer estadio, las alternativas comienzan a ser más elaboradas y dependen de los casos. Las huellas extruidas mencionadas pueden extraerse en direcciones perpendiculares al eje z, con lo que se rompería la homogeneidad de la estructura original pero manteniéndonos aún dentro de una metodología de modelado tan simple que prácticamente no requiere personal especializado. Sin embargo, en muchos casos, será preferible (entre otras cosas porque quizás ya existe) “dejar caer” (con los ajustes necesarios), sobre el DTM, un modelo completo de una determinada zona urbana. Ahora bien, esta alternativa supone, en general, una pérdida considerable de homogeneidad y una mayor exigencia del tiempo dedicado a la gestión de objetos, capas, elementos singulares, etc. Un análisis de algunos de los trabajos que hemos llevado a cabo, junto con mi propia experiencia docente, lleva inevitablemente a la conclusión de que se pierden una considerable cantidad de horas debido a las dificultades para fijar (o conseguir que se acepte) una metodología adecuada para estos casos. Dificultades que, como se decía más arriba, vienen aumentadas por la propia complejidad del software utilizado. La cuestión clave, en este caso, además de la elección del tipo de modelo, es la determinación de una metodología adecuada para los tipos básicos que hemos señalado y, sobre todo para la de este tercer estadio.

### **3. Precisión, Resolución, Nivel de Detalle**

Al igual que ocurre con cualquier tipo de modelo una de las primeras preguntas que es necesario plantear es ¿a qué nivel de detalle, geométrico o visual o ambos, queremos llegar? El grado de precisión geométrica dependerá, en principio, de la información suministrada. Pero la exigencia de precisión depende de la información de salida. Y la información de entrada puede ser muy precisa para determinados aspectos pero no para otros. Podemos contar, por ejemplo, con una información catastral que nos permita situar las calles, las aceras y los contornos de los edificios con una precisión del orden de 1 cm. Pero puede no existir información comparable para los edificios y que sea inevitable aceptar precisiones del orden de 5 o 10 cms.

Será preciso comenzar por aclarar, por consiguiente, si la precisión con que se van a modelar estos edificios es geométrica o visual. Si se requiere precisión geométrica elevada el trabajo de modelado es considerablemente mayor que si se requiere tan sólo precisión visual que puede ser simulada con geometría falsa (mapas de bits) a condición de que el edificio no tenga demasiados resaltes y de que la distancia de observación no exceda de ciertos límites.

La precisión geométrica se necesita cuando el modelo va destinado a la producción. Como este no es el caso en la gran mayoría de los modelos de los que estamos hablando, los requisitos de precisión se transforman, en general, en requisitos de resolución.

Pero también en este caso habrá que discutir el nivel de detalle requerido en función de la distancia mínima a la que se vayan a obtener imágenes del modelo. Una resolución del orden de un 1 pixel = 0.5 cms es una resolución bastante alta pero puede ser el mínimo indispensable si queremos que, por ejemplo, las juntas de un muro de ladrillo sean visibles (que se proyecten al menos sobre 1 pixel). Pero esto puede incrementar de modo considerable la cantidad de memoria requerida para el procesamiento de texturas. Una primera solución es rebajar esta resolución o reservarla para casos especiales.

En los modelos que hemos desarrollado, el grado de resolución ha variado desde 1 pixel = 20 o 50 cms, para modelos grandes y muy simples, hasta 1 pixel = 0.10 cms en modelos pequeños modelados con gran detalle. Resoluciones más características, de modelos como el del Paseo de Gracia o el Forum estarían en torno a 1 pixel = 1 cm. Sin embargo, la tendencia general, que hemos adoptado en el último caso y que se mencionará más adelante (y que se desarrolla con más extensión en otra ponencia) es trabajar con multirresolución.

En cualquier caso, la decisión sobre los niveles de precisión geométrica (vectores) y visual (mapas) debe tomarse desde el comienzo, en conjunción con lo mencionado en el primer punto, la adquisición de datos.

#### **4. Método de modelado, exigencias de editabilidad, subdivisión del trabajo**

Tal como ha puntualizado más arriba, el método de modelado viene dado por las intenciones y, si el modelo es primordialmente visual, la salida óptima debería ser un objeto que puede ser considerado como una superficie continua de topología compleja y espesor cero.

Esto quiere decir que, durante el proceso de modelado, se pueden utilizar todo tipo de recursos corrientes: manipulación de subelementos de mallas, superficies total o parcialmente extruídas, objetos y subobjetos paramétricos, operaciones booleanas, operadores y modificadores locales de diversos tipos, etc. En particular, el uso de elementos paramétricos, cada vez más corriente, permite modelar con agilidad sabiendo que cualquier ajuste posterior se podrá introducir sin más que variar numérica o intuitivamente un determinado parámetro.

Sin embargo, todos estos recursos deben ser utilizados con ciertas reservas y abandonados al final. Si deben ser utilizados con reservas es porque sigue siendo un requisito importante que las mallas poligonales subyacentes sean regulares: un objeto con varios agujeros pequeños en una esquina, obtenidos mediante operaciones booleanas es, internamente, una malla poligonal compuesta de triángulos muy grandes por un lado y muy pequeños por otro, que pueden dar lugar a errores o imperfecciones de diversos tipos. Y si deben ser abandonados al final es porque lo único que nos importa es la malla, y toda información adicional es peso muerto y complicaciones añadidas a la gestión del modelo.

El método ideal para trabajar con un modelo de estas características, desde el punto de vista del resultado final es, por consiguiente, la generación directa de mallas poligonales regulares, con polígonos de tamaños y configuraciones semejantes, que se adecúen bien a los objetos que recubren. Lo que permite, por otra parte, en manos de un operador experto, un alto grado de editabilidad. Sin embargo estos requisitos no se cumplen con facilidad cuando se trata de un proyecto vivo, en donde lo que se busca es apoyar el análisis de alternativas y se necesita introducir modificaciones sistemáticas, ligadas directamente a elementos arquitectónicos claramente diferenciados. En estos casos puede ser inevitable sacrificar unas ventajas (la obtención de una malla poligonal bien conformada) para obtener otras: contar con un modelo en el que todos los elementos, ventanas, puertas, huecos, resaltes, etc., puedan modificarse con agilidad.

Una solución adecuada, que hemos probado en más de un caso, pero que implica gastos adicionales de tiempo, es mantener el modelo paramétrico en un archivo singular y convertirlo, “al final”, en una malla poligonal, y revisar esta malla para regularizarla por medios manuales o automáticos. El mayor inconveniente de esta solución es que, como indican las comillas anteriores, ese “final” casi nunca es el “verdadero final” y esta operación de conversión y regularización tiene que volver a repetirse.

Un último aspecto que tiene también que ver con la metodología de trabajo es la subdivisión del modelo en diferentes sectores que puedan distribuirse para trabajar en red con referencias externas. Esto obliga a una segmentación en divisiones geométricas horizontales y, quizás, en algún caso, niveles verticales y a una gestión cuidadosa de los protocolos de prioridades, actualización, etc.

## **5. Software principal, formato del modelo y compatibilidad de datos**

En teoría, sólo cuando se hayan respondido las cuestiones anteriores tiene pleno sentido la elección del software más adecuado. En la práctica, esto suele depender de diversos factores coyunturales que van del coste o disponibilidad a la mayor o menor familiaridad con las opciones posibles.

Hay un número considerable de programas que se utilizan en la creación de modelos 3D en arquitectura y urbanismo. Y todos ellos permiten trabajar con diferentes métodos y diversas opciones de editabilidad y (hasta cierto punto) conversión a otros formatos. Los más utilizados, entre los programas compartidos por diferentes tipos de profesionales, son AutoCad, Microstation, 3D Studio, Maya, Form Z, Lightwave, Solid Works... Y, entre los programas específicos de arquitectura y urbanismo, ArchiCad, Allplan o Architectural Desktop. Esta lista se incrementaría enormemente si incluyéramos todos los programas que realmente se utilizan por quienes producen modelos arquitectónicos de diferentes tipos, y cada año aparece algún producto nuevo de interés. Uno de los más recientes (al menos uno de los últimos que quien esto escribe ha probado), que resulta de una agilidad sorprendente y que puede desplazar a cualquiera de los anteriores para crear modelos relativamente sencillos con notable agilidad, es SketchUp. Si lo menciono, a riesgo de que estas líneas queden anticuadas en poco tiempo, es porque apunta a una tendencia clara: cada vez hay más productos sencillos y baratos, que sacan buen partido de la creciente rapidez de los PC, para dar respuesta a usuarios que quieren introducirse con rapidez en los misterios de la modelización. Junto a este programa habría que mencionar también los numerosos plug-in que proporcionan métodos de cálculo de iluminación por radiación y representación de materiales con mayor eficacia y más rapidez, y a un coste menor que otros programas más completos. Los profesionales que llevan mucho tiempo trabajando en este sector tienden a resistirse y a menospreciar estos productos pues cuesta aceptar que alguien pueda adquirir en pocas semanas unas habilidades que ha costado años dominar. Pero hay que ponderar muy bien cuánto conocimiento se necesita para crear un modelo relativamente simple. Y todo indica que las curvas de aprendizaje se están reduciendo espectacularmente, gracias entre otras cosas a estos nuevos productos.

Pero la cuestión seguirá siendo ¿qué método de trabajo y, en consecuencia, qué programa escoger para crear un modelo virtual? La respuesta no puede ser “tanto da”. Los primeros modelos que se hicieron en el Laboratorio se modelaban en AutoCad y se remataban en 3D Studio por la sencilla, pero importante razón, de que era más fácil encontrar becarios que conocieran estos programas. Más adelante se llegó a trabajar directamente en 3D Studio, lo que supuso una importante reducción de horas y de problemas. Pero, en otro de nuestros primeros trabajos grandes, tuvimos que trabajar en colaboración con un equipo que utilizaba Maya, lo que supuso igualmente una pérdida importante de horas y una ganancia no deseada de malentendidos. En otros casos ha habido que combinar archivos provenientes de Microstation con otros generados en AutoCad para terminar el trabajo en 3D Studio lo que también ha supuesto pequeños problemas y retardos.

Gran parte de estos problemas pueden evitarse mediante una metodología de trabajo que combine el modelado paramétrico (cada vez más imprescindible) con la conversión y regularización periódica de las mallas poligonales internas. Pero esto es algo que sólo se comprende adecuadamente después de muchas horas de vuelo y mi experiencia como director de proyectos es que los becarios tienden a considerar esta insistencia una “manía” de alguien formado “a la antigua”.

Sin embargo con esto sólo se minimiza una cuestión crucial que aquí sólo puede enunciarse: un laboratorio de modelización virtual está obligado a casarse con un programa principal. Y las novias/ novios aumentan cada año y todas/todos muestran unos atractivos atributos a un coste cada vez menor. Pero las bodas universitarias son lentas y costosas, requieren demasiado tiempo de cortejo y demasiadas formalidades y, al final, el atractivo se ha desvanecido.

## 6. Multirresolución

La especificación del nivel de detalle requerido es, como hemos visto, una de las cuestiones que hay que resolver desde el comienzo. Pero aceptar como nivel de detalle de todo el modelo el nivel correspondiente a la distancia de observación más cercana, supone una carga que, en la mayoría de los casos, no se podrá asumir. Es ineficaz mantener en escena un edificio modelado con decenas de miles de polígonos si va a aparecer en el fondo de la imagen y el resultado sería el mismo si estuviera modelado con unas pocas docenas de polígonos. De ahí que, desde hace algunos años, en la misma medida en que avanzaban la demanda para crear grandes modelados, se haya investigado, cada vez con mayor intensidad, en la búsqueda de técnicas adecuadas para solucionar este problema.

Resulta así que, en un modelo de grandes dimensiones, es cada vez más recomendable utilizar diferentes niveles de detalle para un mismo objeto para optimizar el rendimiento del sistema. Si el modelo va a ser recorrido interactivamente esta recomendación se convierte prácticamente en una obligación.

La primera cuestión a la que hay que responder por consiguiente es ¿cuántos niveles necesitamos y a qué distancia será preciso cambiar de uno a otro? Sea cual sea la respuesta, supondrá una carga adicional sobre el proceso de modelado, si bien hay varios medios de simplificar el proceso. Esto nos lleva a la siguiente cuestión ¿el proceso de simplificación será previo al proceso de representación (LOD estático) o se efectuará en el momento de la representación (LOD dinámico)? En la situación actual, con pocos navegadores que incorporen el segundo sistema y, sobre todo, con las características propias de los modelos arquitectónicos que no incluyen, en general, superficies libres que son las que más se benefician del segundo sistema, la respuesta es, por el momento sencilla (LOD estático). Pero puede cambiar en breve plazo, sobre todo debido a la comodidad que supone no tener que generar manualmente una versión simplificada.

La tercera cuestión que hay que plantearse, entonces, es ¿qué técnica específica resultará más adecuada? Pues para trabajar con LOD estático hay también varias opciones, la generación manual o automática, de un número determinado de niveles y la utilización de uno u otro motor de rendering que proporcione transiciones más o menos suaves. La generación manual de una versión simplificada puede evitarse o aliviarse con sistemas automáticos de optimización de superficies si bien, en modelos arquitectónicos y en determinados casos, puede ser preferible controlar directamente el proceso. Por lo que respecta al número de niveles, la respuesta dependerá del tipo de elemento pero, en la mayoría de los casos que hemos probado directamente o que hemos analizado, 3 parece ser una cifra mágica: una versión muy elaborada, con miles de polígonos, si el objeto va a verse en primer plano, otra versión simplificada, con cientos de polígonos, si va a ocupar una cuarta parte de la pantalla o algo menos, y una versión ultrasimplificada, con pocas docenas de polígonos, si va a ocupar una pequeña parte de la pantalla.

La multirresolución implica cuestiones de interés técnico pero también conceptual, que se comentan más extensamente en otra ponencia sobre este tema, al igual que la del siguiente apartado.

## **7. Sistema de navegación**

Desde 1994 y, sobre todo, desde 1997, quedó abierta la posibilidad de exportar un modelo a formato VRML (Virtual Reality Modeling Language) un estándar abierto que se creó con la intención de proporcionar, para modelos 3D, el equivalente de un sistema como HTML. Desde la aparición de la segunda versión (en 1997), el Web Consortium, que se hace cargo de su desarrollo, ha promocionado diversos avances técnicos que es de esperar que se concreten pronto en el desarrollo de un nuevo estándar, X3D.

Sin embargo, parece que el destino de cualquier intento de normativa pública, como se ha comprobado repetidas veces a lo largo de la historia de la informática, es quedar rezagada con rapidez y terminar por ser prácticamente inútil debido al dinamismo del sector. Aunque han aparecido nuevas versiones y se anuncian, desde hace tiempo, otras mejores, el número de sistemas privados que ofrecen tecnologías mucho más rápidas y eficaces para navegar interactivamente sigue creciendo de un modo espectacular. Gran parte de este crecimiento proviene de la industria de los juegos que ha creado una demanda, cada vez más exigente, de sistemas de navegación en entornos cada vez más complejos, incorporando sistemas de iluminación y cálculo de sombras y todo tipo de efectos que van bastante más allá de la detección de colisiones, que no están ni parece que vayan a estar en mucho tiempo disponibles en las versiones estándar.

Esto obliga, si se quiere alcanzar un cierto nivel de calidad, a utilizar sistemas comerciales o de producción propia para obtener resultados comparables a los de sistemas cada vez más populares y que crean expectativas que no es fácil defraudar. Pero, aunque los algoritmos básicos sean bien conocidos, los sistemas de producción propia que pueden crearse en laboratorios universitarios difícilmente pueden competir con los comerciales. En la actualidad hay entre 500 y 1.000 motores 3D de diversos tipos que incorporan sistemas de rendering y navegación interactiva basada en sistemas de subdivisión jerárquica de la escena (BSP, BSP/PVS, Portal Engines, etc.) para optimizar los resultados. Prácticamente todos estos sistemas requieren programación directa en C++ y el desarrollo de un producto relativamente complejo que requiere dedicación exclusiva por parte de técnicos especializados. A este inconveniente, que es superable, hay que añadir las dificultades de intercambio de información y de compatibilidad del producto.

Es decir que tenemos, por un lado, un sistema con un grado de compatibilidad y distribución prácticamente universal pero con limitaciones considerables por lo que hace al rendimiento y la calidad de los resultados, el VRML y, por otro, un sistema abierto con unas características que son exactamente las contrarias: autoría propia pero mayor potencia. Como en tantas ocasiones, la respuesta, por poco que nos guste a la inmensa mayoría de los departamentos universitarios, es más que probable que venga del triunfo de un producto comercial sobre los otros. Esta es la solución que nos hemos visto obligados a adoptar en algún caso y de la que se da cuenta en la otra ponencia sobre multirresolución y navegación interactiva que mencionábamos más arriba y que también se presenta en este congreso.

## **8. Relación con otros sistemas de información**

El modelo es la base pero, en la mayoría de los casos, no es el objetivo final. Entre las aplicaciones principales de los grandes modelos que cabe citar están las siguientes: soporte a profesionales y técnicos en arquitectura, ingeniería y urbanismo, comunicación de propuestas o alternativas a clientes y al público en general, formación de arquitectos y urbanistas, información cultural y turística, reconstrucción de entornos históricos, soporte a equipos de vigilancia, bomberos, policía, ambulancias, etc., promoción de la ciudad o de barrios concretos para finalidades diversas, soporte a empresas de telecomunicaciones para planificar conexiones áreas, soporte al mantenimiento de infraestructuras, soporte a la detección y resolución de patologías, análisis inmobiliario, comercio virtual, información en línea a vehículos, entrenamiento de conductores. Una lista impresionante pero que probablemente se puede ampliar.

En la mayoría de estos casos se requiere algo más que un modelo navegable: se requiere incorporar información adicional, sea por medio de hipervínculos integrados en el modelo navegable (algo que, por cierto, sí es posible hacer con facilidad en VRML), sea con otras técnicas.

Debido a la estructura interna de los modelos destinados prioritariamente a la visualización no es fácil, como se decía más arriba, obtener información derivada directamente de los datos del modelo o ponerlos en relación con bases de datos relacionales. Una base de datos relacional necesita contar con objetos identificables. En el caso de un modelo arquitectónico o urbanístico esto quiere decir que se debe contar con polígonos (2D) o volúmenes (3D) cerrados. Es posible, en algunos casos, estructurar la información de este modo o reestructurarla por medios semiautomáticos. Pero ya hemos visto que la estructura que mejor se adapta a la navegación interactiva eficiente es la de una superficie continua con múltiples pliegues. De ahí que haya una cierta incompatibilidad inicial entre la necesidad de crear un modelo que sea eficiente desde el punto de vista de la navegación interactiva y uno que permita incorporar información asociada a objetos específicos.

Esto no quiere decir que no sea posible proyectar sobre un modelo de estas características una estructura suficientemente eficaz como para asociar información a todos los elementos que interese, sea mediante hipervínculos directos, sea mediante su asociación más directa con otra base de datos relacional. Pero, en general, esto supone un trabajo adicional que aún no ha sido, al menos en nuestro caso, suficientemente investigado. Las referencias de otros trabajos similares, en otros países, de los que tenemos noticia, no parecen ser lo suficientemente exitosas como para concluir que esta sea una cuestión en vías de ser solucionada satisfactoriamente. Parece ser que queda abierta, así, una interesante línea de investigación. Que, en mi opinión, no podrá ser plenamente resuelta mientras no se resuelvan adecuadamente la mayoría de los puntos anteriores.

## Referencias

- Batty, M. et al. (1998). "Modelling Virtual Urban Environments". *CASA Working Paper Series*, Paper 1.
- Bourdakis, V. (2001). "On Developing Standards for the Creation of VR City Models, in Architectural Information Management". *Proceedings of the ECAADE Conference*, Helsinki 29, 31.8.
- Day, A. K.; Radford, D. (1998) "An overview of city simulation". *CAADRIA '98. Proceedings of the third conference on Computer Aided Design and Research in Asia*.
- Delaney, B. (2000). "Visualization in Urban Planning: They Didn't Build LA in a Day". *IEEE Computer Graphics and Applications*, May/June 2000, 10-16.
- Eberly, D. H. (2000). *3D Game Engine Design*. Addison Wesley.
- Enis, G.; Maver, T. (2001). "Visit VR Glasgow. Welcoming multiple visitors to the Virtual City, in Architectural Information Management". *Proceedings of the ECAADE Conference*, Helsinki 29, 31.8.2001.
- Faust, N.; Ribarsky, W. (2001). "Development of Tools for Construction of Urban Databases and Their Efficient Visualization". *Modeling and Visualizing the Digital Earth*, M. Abdelguerfi (ed), Kluwer, Amsterdam.
- Klosterman, R. E. (1994). "Large-scale Urban Models: Retrospect and Prospect". *Journal of the American Planning Association*, 60 (1), pp 17-29.
- Hall, R. (1999). "Realtime 3D visual analysis of very large models at low cost". *ECAADE17 Liverpool*, pp 437-441.
- Ligget, R.; Jepson, W. "An Integrated Environment for Urban Simulation". *Environment and Planning B*, 22, 291-302.
- Wegener, M. 1994, "Operational Urban Models: State of the Art". *Journal of the American Planning Association*, 60 (1), pp 17-29, 1994.
- Web 3D Consortium: <http://www.web3d.org/> (entrada general al Consorcio)