

INTERFACE GID-SAP2000-VRML

(Aplicación práctica a la modelización virtual de edificios del patrimonio histórico)

Muñoz Salinas, Francisco¹

Maristany i Carreras, Jordi²

López-Rey Laurens, Javier³

Palabra Clave: Métodos numéricos, Preproceso, Postproceso, Entornos virtuales, VRML

Abstrac:

The numerical modeling of historic buildings is an essential tool for understanding the structural behavior, even more whit the aim is to assume an analysis for conservation and / or rehabilitation. At present time, most building's numerical simulation software is divided into three parts: the preprocessor, which prepares all the necessary information (geometry, boundary conditions, loads, etc.), the Analysis, where all calculations take place (FEM) and the postprocessing where decisions are taking based on the analysis of the results obtained in the previous steps (Muñoz, Maristany 1999). At that last point is, where the user of the computer programs usually find limitations in the graphical interpretation of the results. Some numerical results with endless list of numbers are not the best way to address the problem and its graphical output is not yet fully developed. That is, graphic interface is still very basic, let alone a real-time dynamic visualization in 3D. It's important to have both powerful hardware and software tools to visualize the obtained results. Currently, there are emerging many structural analysis programs that have highly advanced postprocessing, but they still display limitations in space, they operate only on 2 or 3 dimensions with space constraints. For this reason, this paper seeks to integrate different current computer programs with 3D graphical representation. Thus, the visualization of numerical results (tension, stress and strain values) can be much more interesting. This is to enable users to virtually navigate in real time, through any 3D structure and better understand their behavior. In short, this research aims to develop an interface between the computer programs of structural calculations and representation software with virtual 3D graphics (vrml- virtual reality modeling language), with practical application to historic buildings. As a closure for the paper we develop a practical example based on Mexico City's Cathedral tabernacle

Resumen:

La modelización numérica de edificios históricos representa una herramienta imprescindible para comprender su comportamiento estructural; más aún si de lo que se trata es de asumir un análisis para su conservación y/o rehabilitación. Actualmente, la mayoría de los programas informáticos de simulación numérica de edificios arquitectónicos se dividen en tres partes: El Preproceso, el cual prepara toda la información necesaria (*geometría, condiciones de borde, cargas, etc.*); el Análisis, donde se desarrollan todos los cálculos (*M.E.F.*) y el Postproceso

¹ Muñoz Salinas, Francisco. Departamento de Construcciones Arquitectónicas I, ETSAB / ETSAV, UPC, Av. Diagonal 649, 08028, Barcelona. francisco.munoz@upc.edu

² Maristany i Carreras Jordi. Departamento de Estructuras en la Arquitectura, ETAB, UPC, Av. Diagonal 649, 08028, Barcelona, jordi.maristany@upc.edu

³ López-Rey Laurens, Javier. Departamento de Estructuras en la Arquitectura, ETAB, UPC, Av. Diagonal 649, 08028, Barcelona, Javier.lopez-rey@upc.edu

donde se analizan y toman decisiones en base a los resultados obtenidos en los pasos anteriores (Muñoz, Maristany 1999). Es en este último punto, donde el usuario de los programas de cálculo se encuentra con limitaciones en la interpretación gráfica de los resultados. Unos resultados numéricos con la interminable lista de números y valores no es la mejor manera de abordar el problema y su salida gráfica de los mismos no acaba de estar completamente desarrollada. Es decir, que la interface de representación gráfica sigue siendo muy básica y ni hablar de una visualización dinámica en tiempo real en 3 dimensiones. Por ello, es importante contar con herramientas tanto de hardware como de software que permitan visualizar los resultados obtenidos. Actualmente, empiezan a aparecer gran cantidad de programas de análisis estructural que tienen sus postprocesos muy avanzados, pero siguen teniendo limitaciones de visualización en el espacio, operan solo en dos o en 3 dimensiones con limitaciones espaciales. Por éste motivo, este documento pretende integrar los diferentes programas de cálculo que existen con los de representación gráfica en el espacio. De esta manera, la visualización de los resultados numéricos (valores tensionales, de esfuerzos y de deformación) puede ser mucho más interesante. Se trata de permitir a los usuarios navegar virtualmente, y en tiempo real, a través de cualquier estructura en 3D y poder comprender mejor su comportamiento. Resumiendo, esta investigación tiene como finalidad el desarrollar una interface entre los programas comerciales de cálculo de estructuras y los de representación gráfica y virtual en 3D (*vrml-lenguaje de modelado en realidad virtual*), con aplicación práctica a edificios históricos. Al final se desarrolla un ejemplo práctica particularizado en el Sagrario de la Catedral de la ciudad de México.

1. Introduction.

En la actualidad, el consultor de estructuras arquitectónicas se encuentra muy desconcertado a la hora de analizar la gran cantidad de información, numérica y gráfica, que proporcionan los programas de cálculo estructural. Uno de los problemas que comporta el análisis de resultados obtenidos en el cálculo estructural de edificios con geometría compleja es, a través de su lectura en formato de texto (*cientos y cientos de folios*), poder abordar las dimensiones del problema, además de tener que interpretar cada uno de los valores que se obtienen.

Visto este problema, cada vez más los investigadores apuestan por un sistema basado en los resultados gráfico-numéricos. Su representación, en la mayoría de los programas comerciales aun es muy poco interactiva y con limitadas herramientas de visualización.

De todo ello, se deduce la importancia que tiene el contar con programas informáticos que permitan visualizar, de manera ágil y sencilla, los resultados gráficos de cualquier análisis de estructuras arquitectónicas, y así, poder hacer una interpretación rápida y correcta de ellos.

2. Metodología para el análisis estructural de un edificio arquitectónico, a través de entornos gráficos virtuales.

En consecuencia, dentro del diseño de un modelo numérico existen tres procesos a desarrollar:

- a. El primero, llamado *Preproceso*, en el cual se define todo lo relacionado a la geometría del edificio, y entrada de datos en general.

- b. El *Análisis*, donde, dependiendo del tipo de problema, se aplican las distintas teorías de análisis estructural (*sistema de barras, de placa, elasticidad bi y tridimensional, etc.*).
- c. Y por último el *Posproceso*, que es la manera de visualizar los resultados numéricos a través de gráficos.

A continuación, analicemos cada uno de ellos:

2.1 El Preproceso:

Su problema fundamental radica en la generación de un modelo numérico de la estructura a estudiar. Por esta razón, la modelización, para su cálculo posterior, puede llegar a ser compleja y esto crea la necesidad de utilizar programas de dibujo (*CAD o GiD*) para su realización. En resumen, el **preproceso** es: La generación de entidades de dibujo de una geometría primitiva de la estructura. Esto significa, dibujar la estructura que se quiere estudiar y a su vez realizar una **discretización** suficiente de ella (generación de la malla).

Una vez que se tiene esta discretización, se graban los datos en un archivo, en nuestro caso con la extensión ***.dxf**, el cual, proporcionará todos los datos que sean necesarios para su posterior análisis: Nodos, coordenadas de los mismos, elementos de la malla, así como la conectividad entre ellos. Estos datos serán posteriormente interpretados por un programa de Elementos Finitos, junto con sus respectivos datos estructurales tales como: materiales, condiciones de contorno, etc.

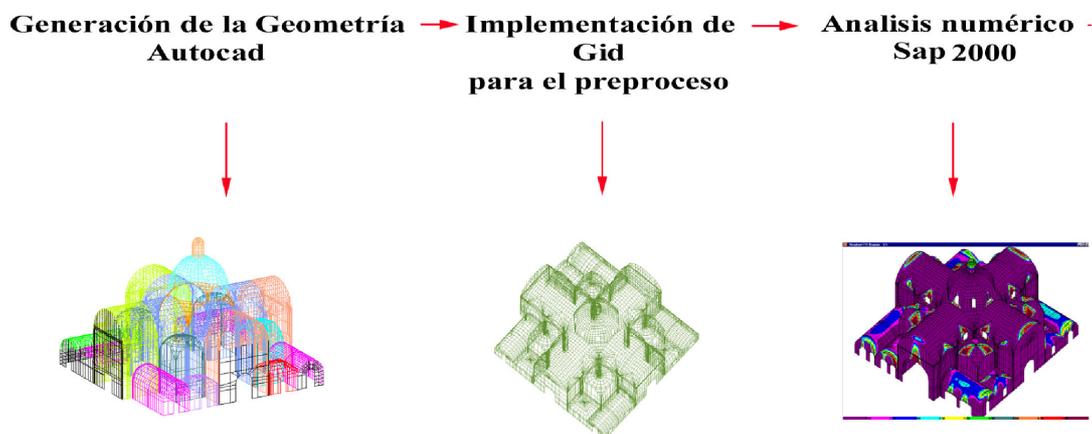
Figura 1. Diagrama del preproceso de una estructura



Esta sería la forma básica de generar un modelo numérico pero, existen otras metodologías como la Interface GiD-Sap2000 que permite generar un modelo geométrico básico en cualquier programa de CAD e importarlo a GiD (*el cual tiene un preproceso muy potente y de fácil manejo y configuración*) y, una vez dentro, terminar de definir la geometría, generar la malla,

estructurarla y configurar todos los datos del problema (*tipo de problema, condiciones de borde y datos del material*). GiD tiene una configuración abierta que te permite realizar una interface con cualquier Programa que utilice el M,E.F. y en este caso se diseño una interface con Sap2000 (Muñoz, Maristany 1999) (*programa genérico muy utilizado en todo el mundo*).

Figura 2. Metodología del Preproceso y Análisis de la Interface GiD-Sap2000



2.2 El Análisis:

En la actualidad, toda esta metodología se realiza mediante programas informáticos cada vez más avanzados y sofisticados. Al mismo tiempo se implementan nuevos métodos de cálculo que nos permiten conocer de forma ágil y fácil el comportamiento estructural de cualquier género de edificio.

Uno de ellos es Método de los Elementos Finitos que actualmente es, gracias a la informática, una de las herramientas más potentes para el cálculo de estructuras entre estudiantes, profesionales e investigadores.

Dentro del cálculo numérico, las estructuras de barras se pueden expresar a través de un número limitado de ecuaciones como sería el caso del conocido cálculo matricial. Desgraciadamente la mayoría de estructuras tienen un comportamiento continuo por lo que su representación partiendo de este número mínimo de estas ecuaciones obliga a simplificar notablemente el modelo con la consiguiente pérdida de exactitud de los resultados obtenidos.

Para poder realizar un análisis más aproximado de estas estructuras se precisa la integración de las ecuaciones diferenciales de equilibrio de cada una de las partes que forman la estructura. Dentro de las estructuras continuas más comunes en arquitectura, podemos encontrar muros de contención, depósitos de agua, cúpulas, forjados planos, etc. A pesar de que todas las estructuras continuas son tridimensionales existen algunos casos en los cuales se pueden describir sus comportamientos por medio de modelos unidimensionales (tal es el caso de las estructuras de barras), bidimensionales (elasticidad bidimensional, teoría de placas) o sólidos tridimensionales.

Actualmente, podemos afirmar que el M.E.F, es una de las herramientas más potentes para el cálculo de estructuras uni, bi, o tridimensional. Cualquier usuario puede entenderlo perfectamente, siempre y cuando tenga los conocimientos básicos tanto de cálculo matricial de barras, informática y programación, así como de resistencia de materiales y elasticidad.

2.3 El postproceso

Una vez realizado el análisis numérico de la estructura a estudiar, el siguiente paso es la interpretación de los resultados. Para ello, los programas comerciales más utilizados (Sap2000, GiD) permiten una visualización gráfica de los valores tensionales, esfuerzos y deformaciones.

En esta investigación se ha desarrollado la interface *GiD-Sap2000-VRML*, que permite al usuario realizar un análisis numérico de cualquier estructura de forma sencilla y con una interrelación, con los programas, fácil y directa.

3. Programas comerciales de pre, análisis y postproceso dentro del cálculo estructural.

3.1 El programa Sap2000

Sap2000 es un programa de análisis, elástico lineal y de segundo orden, de estructuras, por medio del método de los elementos finitos, que incluye un postprocesador gráfico para la presentación de resultados.

La preparación de datos para el desarrollo de un problema comprende básicamente:

- 1. La descripción de la geometría estructural y de los materiales, así como sus condiciones de borde y datos generales.*
- 2. La definición de los estados de carga para los cuales la estructura precisa ser analizada.*

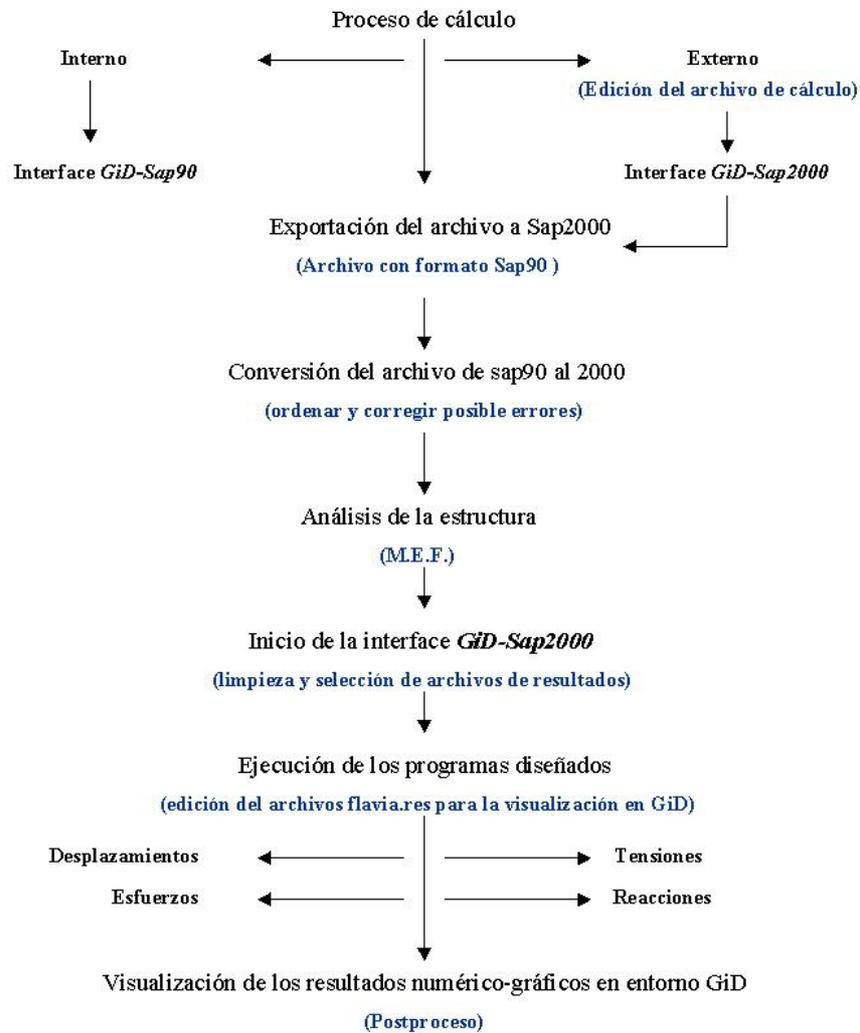
El programa cuenta con un *pre y postproceso* mucho más completo que su anterior versión Sap90, además de implementar el cálculo sísmico. Sap2000 es, actualmente, uno de los programas más utilizados entre los consultores de estructuras e investigadores.

3.2 El programa GiD

GiD es un programa de interface gráfico, que incide en el pre y postproceso y que se utiliza para definir, preparar e incluso visualizar todos los datos relacionados con la simulación numérica. Estos datos, permiten definir la geometría, los materiales, las condiciones de contorno, y toda la información que se precise para la etapa posterior del cálculo. De la misma manera, el programa puede generar mallas de Elementos Finitos y definir toda la información

de simulación numérica en un formato. También es posible, dentro del programa, ejecutar la simulación numérica y visualizar la información de los resultados. Finalmente cabe recordar que implementa programas de dibujo bastante similares al Autocad, lo que aumenta significativamente su versatilidad (CIMNE).

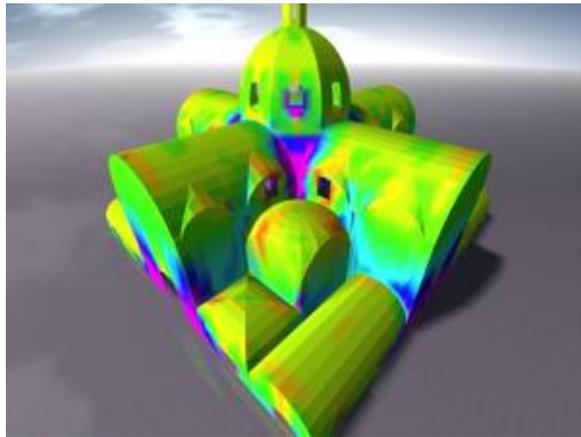
Figura 3. Diagrama de flujo de la interfaces GiD-Sap2000.



4. Interface GiD-Sap2000-VRML.

Actualmente, los programas comerciales para el cálculo de estructuras son más potentes y permiten analizar estructuras cada vez más sofisticadas. Su interface con programas estándar de dibujo (*tanto Sap2000 como GiD pueden operar con formatos *.dxf*) permite un ahorro considerable de tiempo, pero aun no está suficientemente desarrollada la compatibilidad entre los distintos programas de cálculo.

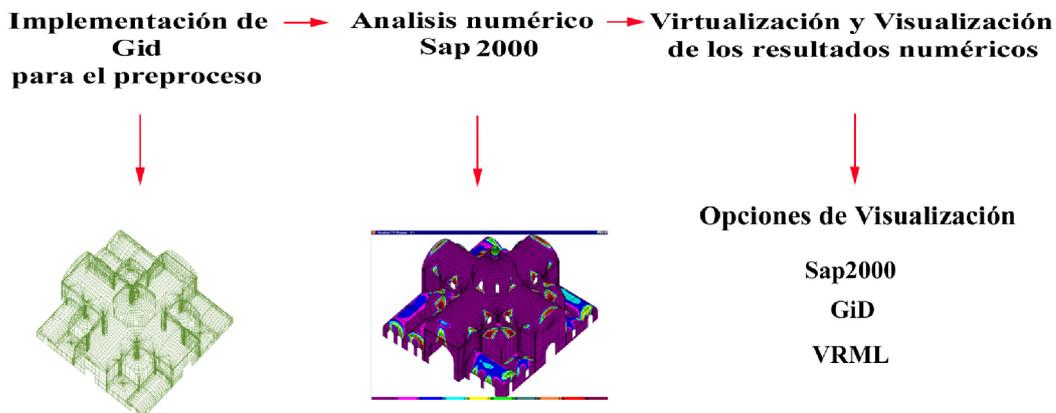
Figura 4. **Modelo numérico de El Sagrario de la Catedral de México.**



Por otra parte, por lo que respecta al postproceso, Sap2000 genera una representación gráfica muy completa, pero, con limitación de visualizaciones. GiD, por el contrario incluye herramientas de rotación dinámica, cortes de malla, modelos 3d, así como la generación de archivos de imágenes en formatos distintos como los *.tif. Pero ninguno de ellos, contempla la posibilidad de generar modelos virtuales, a través de internet de los resultados gráficos.

De ello se deduce que es muy importante que los usuarios de estos programas puedan visualizar estos modelos.

Figura 5. **Postproceso de la Interface GiD-Sap2000-VRML.**



Es en este sentido, que aquí se presenta una interface el VRML, que permite generar modelos virtuales válida para cualquier cálculo numérico y que funciona integrando los programas comerciales más utilizados por las universidades e intentando asumir los últimos avances tecnológicos.

4.1 *Generador de mundos virtuales para internet a partir de los datos de SAP 2000.*

Este programa tiene como objetivo generar un mundo virtual en formato VRML basándose en los datos proporcionados por el programa SAP 2000 y por GiD. Con esta transformación es posible visualizar y hacer un recorrido virtual por el modelo construido.

El formato VRML (*Virtual Reality Modelling Language*) es uno de los estándares actuales para la distribución de modelos tridimensionales que no sólo refleja la geometría del modelo, sino que también permite asignar propiedades y comportamientos tanto en el modelo como en la forma de visualizarlo y navegar dentro de él. En este caso en particular, las propiedades que se deseaban reflejar en el modelo virtual corresponden a alguna de las fuerzas que han sido calculadas por el programa SAP 2000. Además, esta aplicación permite visualizar las isobaras en un entorno virtual. La interfaz del programa fue construida en el entorno de desarrollo de Visual Basic, por la sencillez y rapidez que caracterizan a este entorno.

4.2 *Metodología*

El proceso de conversión pasa por las siguientes etapas:

- Lectura de los datos
- Reestructuración de los datos
- Escritura del modelo **VRML**
- Escritura de la página HTML que contiene al mundo virtual

4.2.1 *Lectura de los datos*

La entrada de datos se toma a partir de **los archivos mallas.txt y forces.txt**. del primer archivo, se extrae la información relativa a la posición de los vértices que componen el modelo. Del segundo archivo, se obtiene la definición de los polígonos que forman el modelo del objeto y la propiedad de *esfuerzos y tensiones* asociada a cada uno de los vértices que lo componen. A pesar de que el archivo *forces.txt* tiene multitud de fuerzas definidas, en el proceso de lectura sólo se lee y almacena aquella por la cual se está interesado.

4.2.2 *Reestructuración de los datos*

Para hacer la conversión es necesario hacer algunos cambios en la organización de la información. En primer lugar se establecen todos los valores de acotamiento del modelo, tanto en la geometría como en las propiedades. Esto quiere decir que se establecen los valores máximo y mínimo en cada uno de los ejes y en la fuerza seleccionada. Con estos valores se podrá ajustar la escala, la ubicación del objeto y el rango de valores representados en la propiedad de fuerza.

Otro ajuste que se debe de llevar a cabo, es referente al valor de la propiedad en cada vértice. El archivo *forces.txt*, define estas propiedades de forma independiente para cada uno de los polígonos que forman el modelo, lo que lleva a que un mismo vértice pueda tener distintos valores de propiedad dependiendo del polígono al cual esté asociado.

Para disminuir las discontinuidades producidas por este hecho, se ha decidido asociar a cada vértice el promedio de las propiedades que tiene asociadas en cada uno de los polígonos a los cuales pertenece. De tal forma que los vértices que sean interiores en la malla, tendrán 4 polígonos asociados y por lo tanto su valor final será el promedio de la propiedad en cada uno de ellos. En cambio un vértice que se encuentre en una esquina de la malla, sólo tendrá un polígono asociado y por lo tanto no hará falta hacer el promedio.

4.2.3 Escritura del modelo VRML

VRML, es un lenguaje para describir mundos virtuales y por lo tanto se debe ajustar a la sintaxis que define el lenguaje. En este caso se está empleando la especificación 2.0 del lenguaje que es la especificación en vigor.

El mundo que se va a construir consta de varios objetos: la malla del modelo aportado por SAP 2000 o GiD, un plano de referencia sobre el cual asentar el modelo, las luces existentes en el mundo virtual y las especificaciones de puntos de vista del observador.

Un archivo VRML tiene una estructura jerárquica de tipo árbol. Por lo que, el primer nodo contiene a los demás y todo lo que afecte a éste repercutirá en los nodos hijo.

En este caso a partir de un nodo de grupo se añaden los demás nodos que representarán a los objetos mencionados anteriormente.

Por ejemplo para definir un nodo para un punto de observación.

```
DEF side_view Viewpoint {
    position 0 0 20.00
    orientation 1.0 0.0 0.0 0
    fieldOfView 0.387763
    description "Lateral"
}
```

Después seguirían los nodos que definen las luces, en este caso se utiliza luz direccional.

```
DEF front_light DirectionalLight {
    ambientIntensity .1
    color 1 1 1
    direction 0 0 -1
    intensity .5
    on TRUE
}
```

A continuación viene un nodo para establecer propiedades iniciales del navegador y también un nodo para definir un color de fondo en el mundo virtual.

Y los últimos nodos son los que definen los objetos a visualizar, el plano de referencia que se construye con una de las primitivas geométricas del lenguaje (un box) el cual está definido como un nodo hijo de un nodo transformación. Esto se hace para poder colocar el plano de referencia en el sitio adecuado dependiendo del modelo a visualizar.

```

Transform {
translation 0.0 -2.21300549335834 0
children [
Shape {
appearance Appearance {
material Material { diffuseColor 0.2 0.2 0.2 } }
geometry Box { size 8 0.01 8 }
}
]
}

```

El nodo que representa la malla del modelo es bastante largo y por lo tanto sólo se mostrarán las partes que lo componen. Al igual que con el objeto del plano de referencia este objeto tendrá un nodo transformación que controla su posición y orientación y dentro de ese nodo se crea un nodo formado por polígonos (IndexedFaceSet).

```

Transform {
scale 0.125997 0.125997 0.125997
children [
Transform {
rotation 1 0 0 1.57
translation -58.3524017333984 -10.2425394058228 41.4154014587402
children
Group {
children
Group {
children
Shape { # Superficie
appearance Appearance {
material DEF estrucMat Material { shininess 0.2 }
}
geometry IndexedFaceSet {
coord Coordinate {point [ 0 0 0 ,
34.5423 -18.7154 7.32142 ,
.....
]}
color Color { color [ 0.96 0.0 .96 ,

```

```

...
                                0.35 0.06 ]}
colorPerVertex TRUE
coordIndex [
5400                                5290 5319 5499 -1
....
                                ]
                                colorIndex [
2                                    0 0 0 -1
....
                                ]
ccw FALSE #tenia valor true
solid FALSE
convex FALSE
creaseAngle 0
}
}
}
}
}
]
}
}

```

4.2.4 Escritura de la página HTML que contiene al mundo virtual

Además del archivo VRML, se adjunta un archivo HTML, es decir, una página web desde donde se llama el archivo VRML. Esto es un paso que no es necesario para la visualización, ya que los navegadores de internet pueden leer el archivo VRML sin necesidad de esta página. Sin embargo, En la página web se anexa la escala de valores que se está empleando y se podría incluir más información referente al modelo. Un ejemplo de esta página es el siguiente.

```

<html>
<head>
<title>Modelo generado para la fuerza F22</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#000000" text="#CCCCCCFF">
<table width="100%" border="0" cellspacing="10">
<tr>
<td width="80" valign="top">
<p align="center"><font face="Arial, Helvetica, sans-serif"><b><font size="2"> 1.6884
</font></b></font></p>
<table width="20" border="0" cellspacing="0" align="center">
<tr bgcolor="#FF3300"><td>&nbsp;</td></tr>
<tr bgcolor="#FF9900"><td>&nbsp;</td></tr>
<tr bgcolor="#FFFF00"><td>&nbsp;</td></tr>
<tr bgcolor="#66FF00"><td>&nbsp;</td></tr>

```

```

<tr bgcolor="#33FF66"><td>&nbsp;</td></tr>
<tr bgcolor="#33FFCC"><td>&nbsp;</td></tr>
<tr bgcolor="#33CCFF"><td>&nbsp;</td></tr>
<tr bgcolor="#0066FF"><td>&nbsp;</td></tr>
<tr bgcolor="#6600FF"><td>&nbsp;</td></tr>
<tr bgcolor="#CC00FF"><td>&nbsp;</td></tr>
<tr bgcolor="#FF00CC"><td>&nbsp;</td></tr>
</table>
<p align="center"><font face="Arial, Helvetica, sans-serif" size="2"> <b>-1.4577</b></font></p>
</td>
<td><embed src="fvF22nuevo.wrl" width="800" height="600">
</embed></td>
</tr>
</table>
</body>
</html>

```

Figura. 6. Virtualización numérica del Sagrario de la Catedral México.

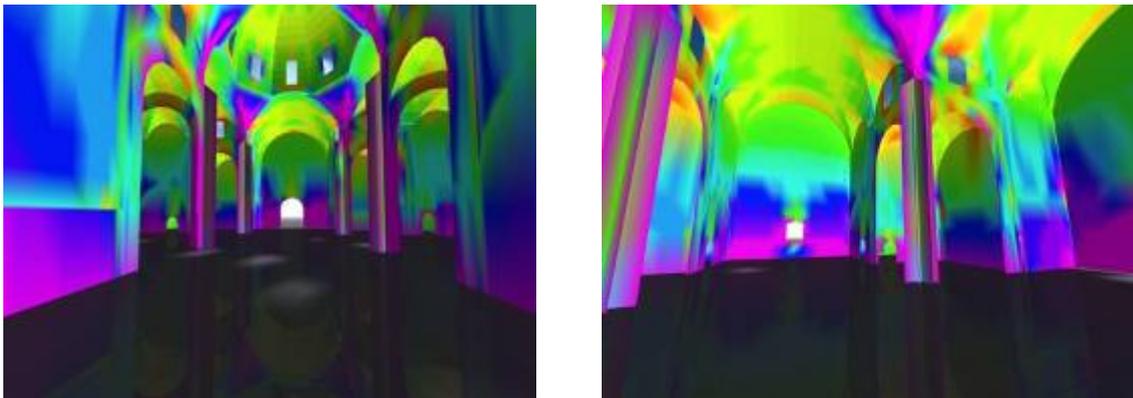


Figura. 7. Modelo virtual de los resultados numéricos del Sagrario de la Catedral México.

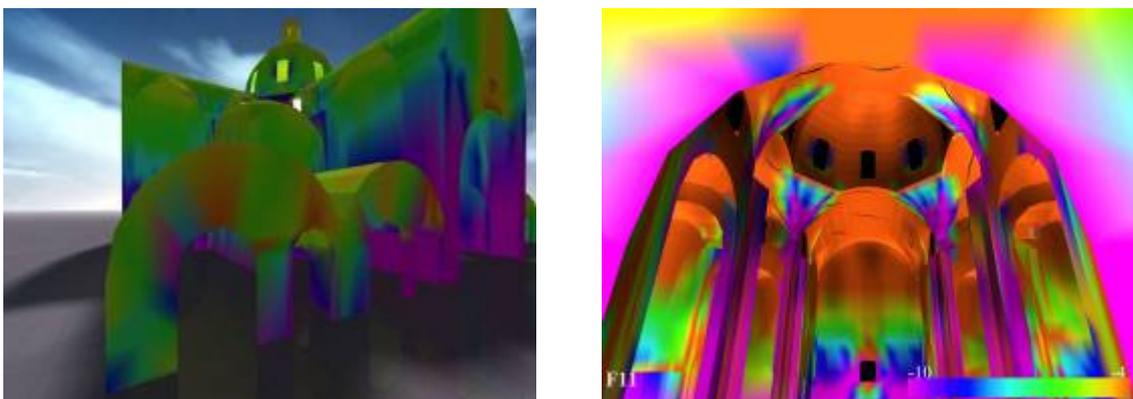
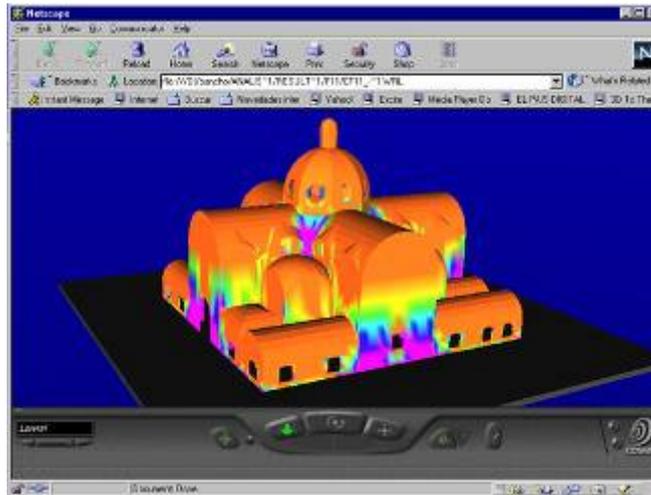


Figura 8. Virtualización numérica del Sagrario de la Catedral México por Internet.



Conclusiones:

El espacio virtual muchas veces tiene más peso que el real ya que genera percepciones más complejas. Una interpretación adecuada de los resultados numéricos permite, al usuario, un conocimiento más profundo del comportamiento estructural de edificios arquitectónicos en general, así como el de los históricos en particular. Es por ello, que esta interface integra algunos programas comerciales, cubriendo la ausencia de herramientas más potentes en la visualización gráfica de los resultados numéricos, fundamentalmente en el campo de 3D.

Bibliografía:

GID

International Center for Numerical Methods in Engineering

HISTORICAL CONSTRUCTIONS 2001

Possibilities of numerical and experimental techniques : proceedings of the 3rd international seminar : Gimaraes, Portugal, University of Minho, 7-8-9 novembre 2001
International Seminar Historical Construction (3r : 2001 : Guimaraes, Portugal)

MUÑOZ SALINAS, FRANCISCO Y MARISTANY I CARRERAS, JORDI

Pre y Postprocesadores para el Cálculo de Estructuras.

Interface GID-Sap90.

CIMNE, 1999

Primera Edición.

MUÑOZ SALINAS, FRANCISCO

Una contribución a la modelización y virtualización numérica de estructuras Arquitectónicas

Directores de tesis: López-Rey Laurens, Javier. Maristany y Carreras, Jordi.

2000

SAP 2000.

Computer Software for

Structural & Earthquake Engineering.

Developed and written in U. S. A.

SEMINARIO INTERNACIONAL SOBRE ANÁLISIS ESTRUCTURAL DE
CONSTRUCCIONES HISTÓRICAS.

Posibilidades de las técnicas numéricas y experimentales

E.T.S.E.C.C.P., U.P.C.

Barcelona, 8, 9, 10 de Noviembre 1995.