



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA

⑪ Número de publicación: **2 226 772**

⑤① Int. Cl.<sup>7</sup>: **G06T 17/10**

⑫

TRADUCCIÓN DE PATENTE EUROPEA

T3

⑧⑥ Número de solicitud europea: **00901050 .5**

⑧⑥ Fecha de presentación: **21.01.2000**

⑧⑦ Número de publicación de la solicitud: **1323136**

⑧⑦ Fecha de publicación de la solicitud: **02.07.2003**

⑤④ Título: **Modelado de realidad virtual.**

③⑩ Prioridad: **22.01.1999 DK 9599**

④⑤ Fecha de publicación de la mención BOPI:  
**01.04.2005**

④⑤ Fecha de la publicación del folleto de la patente:  
**01.04.2005**

⑦③ Titular/es: **LEGO AS.**  
**Aastvej, 1**  
**7190 Billund, DK**

⑦② Inventor/es: **Thomsen, Michael;**  
**Krogh, Poul y**  
**Timcenko, Olga**

⑦④ Agente: **Curell Suñol, Marcelino**

ES 2 226 772 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

**DESCRIPCIÓN**

Modelado de realidad virtual.

**5 Campo de la invención**

La presente invención se refiere al campo del modelado de realidad virtual asistido por ordenador mediante elementos geométricos predefinidos.

**10 Antecedentes de la invención**

El modelado de realidad virtual asistido por ordenador consiste en la tarea de crear un modelo de un objeto geométrico, interpretar el modelo, manipular el modelo y tratar el modelo de algún otro modo en un sistema informático.

15 Desde un primer punto de vista, el modelado de realidad virtual es una cuestión interesante en la medida en que permite visualizar las ideas antes de ejecutarlas en el mundo real. Si el modelo de realidad virtual puede modificarse con suficiente facilidad, el tiempo que puede ahorrarse en el procedimiento de creación y perfilado de un objeto geométrico será muy superior al que se puede ahorrar si el mismo procedimiento se lleva a cabo en el mundo real. La simple tarea de pintar un objeto en el mundo real puede llevar fácilmente varias horas, mientras que el ordenador puede tardar varios milisegundos o segundos en aplicar un nuevo color para visualizar un modelo.

20 Desde un segundo punto de vista, el modelado de realidad virtual es interesante, porque permite crear un modelo de un objeto que existe en el mundo real y visualizar y manipular de alguna forma el modelo por medio de un ordenador. De este modo el modelo del objeto del mundo real puede almacenarse con diversos fines (por ejemplo, para documentación avanzada).

Aunque existe una cantidad enorme de posibles aplicaciones para el modelado de realidad virtual asistido por ordenador, una aplicación particular consiste en utilizar el modelado de realidad virtual para el ocio o la educación. Mucho antes de la creación del ordenador electrónico, ya se conocían diversos tipos de modelado de conceptos del mundo real. En particular, los conceptos que utilizan conceptos modulares o semimodulares fueron, y siguen siendo, muy comunes. Habitualmente, estos conceptos proporcionan un grupo de elementos prefabricados que pueden interconectarse entre sí de alguna forma predeterminada según los módulos de los elementos prefabricados. Los elementos prefabricados representan objetos muy conocidos adaptados a una tarea de modelado concreta. Por ejemplo, cuando se crea un modelo de una casa, los elementos pueden mostrarse como ladrillos, tejas, puertas y ventanas. Esta forma de seleccionar los elementos tiene como objetivo reducir significativamente la cantidad de trabajo que conlleva la creación del modelo de una casa, en comparación con una situación en la que es necesario definir todos los detalles de la casa cada vez que debe crearse un nuevo modelo. No obstante, esta simplicidad en la creación del modelo tiene como contrapartida la pérdida de la libertad completa para crear una casa u otro objeto.

40 Esta propuesta de disponer de elementos predefinidos es muy conocida en el ámbito de la técnica de modelado de realidad virtual asistida por ordenador. También es muy conocido el concepto de tener elementos modulares que se pueden interconectar entre sí. Pero en lo referente a la representación de dichos modelos en el ordenador, la técnica anterior no saca el máximo partido de la utilidad del concepto de tener elementos predefinidos y sistemas modulares.

45 En la utilización de los ordenadores para el diseño y el modelado asistido por ordenador, la tarea que más dedicación requiere desde el punto de vista de la informática ha sido la visualización del modelo, incluido el cálculo de cómo debe aparecer el modelo. Esto se debe, entre otros motivos, a que la complejidad y el perfeccionamiento de los modelos de realidad virtual han ido aumentando para seguir la permanente evolución de la tecnología informática y la potencia informática disponibles en el mercado.

50 En lo referente a la transferencia y el intercambio de los datos que representan un modelo, los inconvenientes de la complejidad y el perfeccionamiento de los modelos de realidad virtual son muy evidentes. Actualmente, se conecta al ordenador algún tipo de unidad externa en forma de unidad de almacenamiento o una red de comunicación informática, ampliando de ese modo la trayectoria de la señal para el almacenamiento, la copia o la transmisión y la recepción de un modelo. Habitualmente, dichas trayectorias de señal tienen un ancho de banda relativamente bajo en comparación con las trayectorias de señal de un solo ordenador. Por lo tanto, se plantea la necesidad de disponer de un sistema eficaz para representar los modelos.

**Técnica anterior**

60 Un sistema para representar un modelo de un objeto geométrico es el conocido lenguaje de modelado de realidad virtual conocido también como "VRML". Este sistema se utiliza a menudo en Internet. El lenguaje "VRML" se basa en ciertos tipos de polígonos definidos mediante un conjunto de parámetros. Todos los modelos se basan en estos polígonos especificando los parámetros (por ejemplo, delimitando la posición de las esquinas de los polígonos en un espacio tridimensional). Es posible definir la posición, orientación y escala de un grupo de polígonos mediante una transformación común.

Por medio de este sistema de modelado, todas las tareas de modelado posibles pueden modelarse de la forma de-

seada, ya que el sistema no impone ninguna limitación sobre el usuario. En realidad, éste es uno de los inconvenientes del sistema, puesto que cada polígono se basa en una extensa representación. En un ordenador, esta representación comprende números de tipo real para cada parámetro del modelo. Además, debido a la naturaleza de la mayor parte de paradigmas de modelado, se necesita una cantidad abrumadora de polígonos (y en consecuencia, un número abrumador de parámetros de tipo real). Considérese, por ejemplo, una superficie curva. En un modelo muy grosero, dicha superficie puede modelarse mediante un solo polígono. No obstante, si se necesita un modelo perfeccionado, esta superficie curva debe descomponerse en una serie de polígonos, que comprende por lo menos la cantidad de polígonos necesaria para representar la superficie con suficiente uniformidad.

En lo que respecta a la manipulación de un modelo VRML, la descripción geométrica debe ampliarse con una metadescripción que define cada una de las posibles manipulaciones. Por lo tanto, si el modelo VRML no se proporciona en un principio con dicha metadescripción, no podrá efectuarse ninguna manipulación del modelo. Ésta es una de las consecuencias del sistema de modelado completamente libre.

Por lo tanto, la técnica anterior presenta el problema de que los modelos obtenidos mediante modelado asistido por ordenador requieren una capacidad de almacenamiento inmensa.

En la técnica anterior, existen diversos tipos de sistemas de compresión; no obstante, en dichos sistemas de compresión, es necesario actuar sobre un gran bloque de datos para comprimir los datos con eficacia. Por otra parte, dichos sistemas de compresión requieren un intervalo de tiempo relativamente grande para llevar a cabo la compresión y la descompresión. Esta cuestión suele ser crucial cuando se trata de tareas de modelado interactivo que comprenden la transferencia de modelos (por ejemplo, por medio de una red informática). Por consiguiente, se plantea la necesidad de disponer de una representación de modelos geométricos que se genere e interprete con rapidez y que no requiera sistemas de compresión adicionales.

## Sumario de la invención

Un primer objetivo de la presente invención es proporcionar una representación compacta de un modelo de un objeto geométrico.

Un segundo objetivo de la presente invención es proporcionar una representación compacta de un modelo de un objeto geométrico que permita el modelado de objetos complejos que comprenden partes móviles o giratorias.

Un tercer objetivo de la presente invención es proporcionar una estructura de modelo y una representación de la estructura de modelo que pueda ser generada e interpretada de forma muy rápida.

Un cuarto objetivo de la presente invención es proporcionar una estructura de modelo y una representación de la estructura de modelo que sea útil para representar objetos compuestos de reproducciones de bloques o elementos de construcción de juguete modulares.

Un quinto objetivo de la presente invención es proporcionar una representación de una estructura de modelo que sea adecuada para ser distribuida por medio de una red informática, en el sentido de tener un requisito de ancho de banda relativamente bajo y un tiempo de procesamiento corto cuando el modelo se somete a visualización, se genera o se manipula.

Un sexto objetivo de la presente invención es proporcionar una estructura de modelo y una representación de la estructura de modelo que no requiera ninguna metadescripción adicional del objeto geométrico cuando el modelo se somete a manipulación.

Según la presente invención, los objetivos mencionados en los párrafos anteriores se llevan a cabo dentro del alcance de la presente invención según las reivindicaciones adjuntas. En particular, estos objetivos se cumplen cuando el modelo de un objeto geométrico comprende bits en una primera estructura de datos codificados para identificar un primer conjunto de elementos a partir de un grupo de representaciones de formas geométricas para un grupo de elementos, y para representar las posiciones de los elementos mediante coordenadas enteras en un primer sistema de coordenadas, y bits en una segunda estructura de datos codificados para representar una transformación espacial de un segundo sistema de coordenadas en relación con el primer sistema de coordenadas.

De esta forma, se obtiene una representación muy compacta de un objeto geométrico, a la vez que el modelo permite el modelado de estructuras geométricas complejas que comprenden partes móviles. De hecho, el modelo diferencia entre las partes rígidas y confina la posición de los elementos dentro de dichas partes rígidas para obtener una representación compacta. No obstante, cuando es necesario representar una transformación espacial de dichas partes rígidas o segmentos, puede aplicarse una representación no restringida o menos restringida de la interrelación. En consecuencia, también se obtiene una representación del modelo relativamente flexible.

Debido a la representación entera de las posiciones de los elementos, es posible procesar el modelo una y otra vez en un ordenador sin introducir errores de redondeo en los respectivos sistemas de coordenadas. Por consiguiente, los modelos que comprenden cantidades enormes de elementos en un segmento no se distorsionan debido a dichos errores de redondeo.

Debido al hecho de que el modelo sólo comprende referencias a las representaciones geométricas de los elementos, el modelo no necesita ser modificado para ser interpretado con un grupo seleccionado de elementos. En consecuencia, puede seleccionarse un formato gráfico específico para ser interpretado correctamente en un ordenador específico (por ejemplo, en el sentido de proporcionar un tiempo de presentación suficientemente rápido cuando el modelo se muestra en pantalla).

Debido a la representación entera restringida de las posiciones de los elementos dentro de los respectivos sistemas de coordenadas, las posibles posiciones de los elementos se dan implícitamente, es decir, las rotaciones, los desplazamientos libres, etc. no son posibles dentro de un sistema de coordenadas.

Puesto que la representación compacta del modelo se obtiene mediante restricciones sobre los elementos, no se necesita ninguna compresión adicional y, por consiguiente, la estructura del modelo puede ser generada e interpretada en secuencia escribiendo o leyendo una serie de sentencias utilizadas para representar el modelo.

### 15 Breve descripción de los dibujos

A continuación, se describirá la presente invención con mayor profundidad, en relación con un ejemplo de realización preferido y con los dibujos, en los que:

20 la Fig. 1 muestra la arquitectura jerárquica de un modelo de un objeto geométrico según un lenguaje de modelado de realidad virtual de técnica anterior;

la Fig. 2a muestra un primer objeto sujeto a modelado según la presente invención;

25 la Fig. 2b muestra un segundo objeto sujeto a modelado según la presente invención;

la Fig. 3 muestra la estructura jerárquica de un modelo de un objeto geométrico según la presente invención;

30 la Fig. 4 muestra dos familias de elementos;

la Fig. 5 muestra la primera etapa de un procedimiento de construcción de un modelo de un tercer objeto geométrico;

35 la Fig. 6 muestra la segunda etapa de un procedimiento de construcción de un modelo del tercer objeto geométrico;

la Fig. 7 muestra la tercera etapa de un procedimiento de construcción de un modelo del tercer objeto geométrico;

la Fig. 8 muestra un cuarto objeto para modelar según la presente invención;

40 la Fig. 9 muestra un sistema informático de cliente para diseñar, almacenar, manipular y compartir modelos geométricos según la presente invención;

45 la Fig. 10 muestra un sistema informático de servidor para procesar construcciones geométricas según la presente invención;

la Fig. 11 muestra un diagrama de flujo de un procedimiento para construir de forma simultánea segmentos y modelos, mientras se someten los segmentos y modelos a manipulación;

50 la Fig. 12 muestra un diagrama de flujo de un procedimiento para acceder a un mundo común accesible por múltiples clientes y compartirlo y

la Fig. 13 muestra un diagrama de flujo de un procedimiento para transferir información de modelo.

### 55 Descripción detallada de una realización preferida

La Fig. 1 muestra la estructura jerárquica de un modelo de un objeto geométrico, según un lenguaje de modelado de realidad virtual de técnica anterior. Los expertos en la materia conocen dicho lenguaje de modelado de realidad virtual por las siglas "VRML". Los elementos básicos del lenguaje VRML son polígonos no cóncavos bidimensionales delimitados por sus coordenadas en un sistema rectangular tridimensional de coordenadas. Si los ejes de un sistema rectangular tridimensional de coordenadas se denotan como eje de las X, eje de las Y y eje de las Z, respectivamente, un polígono convexo bidimensional, que puede ser un rectángulo, es delimitado por las coordenadas de las cuatro esquinas del rectángulo (es decir, por cuatro grupos de coordenadas X, Y y Z.). Debido a la estructura basada en polígonos del lenguaje VRML, el diseñador de un modelo de realidad virtual de un objeto ficticio o existente del mundo real debe dividir el objeto en polígonos. Por lo tanto, para componer un modelo VRML preciso y bien definido de un objeto que tiene una superficie curva y uniforme deberá utilizarse una cantidad enorme de polígonos para que el modelo aparezca con una superficie bien definida y uniforme. No obstante, puede componerse un modelo VRML más grosero del objeto a partir de un número inferior de polígonos, reduciendo de ese modo la complejidad del modelo. Para que el diseñador pueda modelar cualquier objeto, deberá poder situar libremente los polígonos bidimensionales

## ES 2 226 772 T3

en el sistema tridimensional de coordenadas (de ahora en adelante, denominado también “espacio tridimensional” o simplemente “espacio”). En un ordenador, las coordenadas de los polígonos se representan mediante números de tipo “real” (por ejemplo, mediante números de coma flotante).

5 Para manejar incluso modelos bastante simples, los polígonos se distribuyen en una jerarquía de estructura en árbol invertida 101. Cada rama 102 del árbol se conecta a un nodo para delimitar la jerarquía. Cada nodo de un nivel dado de la jerarquía es una referencia común para otros nodos o polígonos de un nivel inferior de la jerarquía. La jerarquía está orientada de arriba hacia abajo y finaliza en los polígonos que constituyen el nivel más bajo de la jerarquía.

10 Los polígonos 103 y 104 son definidos por las coordenadas X, Y y Z y se relacionan con el mismo nodo 107 por medio de las interfaces 105 y 106. También otros nodos 108 que tienen polígonos (no mostrados) pueden relacionarse con el nodo 107. Por medio de la parte de interfaz 109 del nodo 107, es posible trasladar, girar y escalar los polígonos y nodos de un nivel inferior de la jerarquía conectados de forma directa o indirecta con el nodo 107, mediante una transformación común. Por lo tanto, es posible agrupar varios polígonos y tratar el grupo mediante una sola transformación  
15 que comprende traslación, rotación y escalado.

Todos los parámetros utilizados en la definición de los polígonos y los nodos, es decir, coordenadas, traslaciones, rotaciones y escalas, son del tipo real y, por lo tanto, se representan como números de coma flotante en un ordenador. Se selecciona esta representación para tener la capacidad de manejar cualquier tarea de modelado. No obstante, esta  
20 representación requiere una enorme capacidad de almacenamiento y en realidad representa una traba para el almacenamiento y la transferencia de modelos -incluida la manipulación en tiempo real de los modelos.

La Fig. 2a muestra un objeto simple que comprende un primer y un segundo segmento, sometido a modelado según la presente invención. El modelo del objeto tiene dos segmentos 201 y 202 que tienen los elementos 203, 204 y 205 y  
25 206 y 207, respectivamente. Los ejes  $x_1$ ,  $y_1$  y  $z_1$  abarcan el sistema de coordenadas 209 y pertenecen al segmento 201 y los ejes  $x_2$ ,  $y_2$  y  $z_2$  abarcan el sistema de coordenadas 211 y pertenecen al segmento 202.

Debe observarse que los elementos 203, 204 y 205 del primer segmento se sitúan sólo según los módulos 208  
30 del sistema de coordenadas 209. Asimismo, los elementos 206 y 207 del segundo segmento se sitúan sólo según los módulos 210 del sistema de coordenadas 211.

El objeto se divide en dos segmentos con el propósito indicado a continuación.

La Fig. 2b muestra el objeto simple mostrado en la Fig. 2a, en el que el segundo segmento ha sido sometido a  
35 rotación en relación con el primer segmento. En esta ilustración, se pone de manifiesto que existen dos sistemas de coordenadas diferentes y no coincidentes definidos por los ejes  $x_1$ ,  $y_1$ ,  $z_1$  y  $x_2$ ,  $y_2$  y  $z_2$ , respectivamente. Los elementos del segmento 201 pueden situarse sólo según los módulos 208 de la retícula 209. De la misma manera, los elementos del segmento 202 pueden situarse sólo según los módulos 210 de la retícula 211. Los elementos de un segmento están  
40 pues limitados en cuanto a su posición. Las retículas se muestran como retículas tridimensionales, pero habitualmente serán tridimensionales, permitiendo la distribución de los elementos en tres dimensiones. Además, las orientaciones de los elementos dentro de un segmento se limitan a rotaciones en etapas de 90 grados con respecto a los tres ejes ( $x_1$ ,  $y_1$ ,  $z_1$  y  $x_2$ ,  $y_2$  y  $z_2$ , respectivamente) de los segmentos.

El segmento 202 se sitúa según la retícula 209 del segmento 201, pero el segmento 202 puede girar con respecto  
45 al eje  $z_1$ . El modelo simple se ha ampliado y, entonces, será posible modelar objetos avanzados que comprendan una secuencia de segmentos que pueden girar. Debe observarse que, a pesar de que se introduzca esta característica de rotación en el modelo simple, la representación sólo se amplía con un único número que representa el ángulo de rotación, ya que la posición del segmento 202 está limitada a la retícula 209 y los elementos 204 y 205 también están limitados a una retícula (la retícula 211).

50 El modelo se puede ampliar añadiendo de forma gradual más elementos a los segmentos e interconectando nuevos segmentos que tienen elementos con los segmentos existentes 201 y 202. En realidad, pueden modelarse objetos geométricos bastante complejos de esta forma, a pesar de las fuertes restricciones relativas a la posición y a la orientación. Estas fuertes restricciones son la clave de una representación muy compacta del modelo de un objeto geométrico  
55 complejo.

En particular, un modelo según la presente invención será superior a los modelos de realidad virtual de técnica anterior cuando el objeto sujeto a modelado comprenda una cantidad relativamente baja de partes rígidas que pueden desplazarse o girar unas respecto de otras y cuando las partes rígidas deban modelarse con muchos elementos. Un  
60 ejemplo de dicho objeto puede ser un puente que tiene dos tramos voladizos. Un modelo de este objeto puede comprender sólo dos partes rígidas móviles, es decir, los tramos voladizos, pero los respectivos tramos voladizos y las plataformas que los sostienen pueden comprender estructuras rígidas, tales como cercas, calzadas y torres que deben modelarse por medio de muchos elementos.

65 A continuación, se describe cómo se genera la representación según la presente invención y cómo pueden añadirse otras características al modelo según la presente invención.

La Fig. 3 muestra la estructura jerárquica de un modelo de un objeto geométrico según la presente invención.

## ES 2 226 772 T3

5 El objetivo de la jerarquía no es simplemente proporcionar agrupamientos lógicos de primitivas geométricas que pueden someterse a una transformación común, sino también añadir diferentes tipos de restricciones sobre los parámetros utilizados en los respectivos niveles de la jerarquía, permitiendo de ese modo seleccionar una representación adecuada de los parámetros a tenor de las restricciones y en el sentido de establecer una representación compacta del modelo.

10 Debe observarse que la jerarquía concierne a la relación espacial entre los elementos, no a la representación concreta de las formas geométricas de los elementos. Es posible seleccionar un formato adecuado (por ejemplo, VRML o 3DMF) para la representación concreta de la forma de los elementos.

15 Un modelo según un ejemplo de realización de la presente invención puede organizarse por medio de una jerarquía 316 que comprende cuatro niveles básicos; “nivel Mundo”, “nivel Modelo”, “nivel Segmento” y “nivel Elemento”.

20 En el nivel Mundo se ha definido el bloque Mundo 315 que representa un mundo virtual. El objetivo primario del bloque Mundo consiste en definir un espacio tridimensional que tiene un origen que es una referencia común para los niveles inferiores de la jerarquía.

25 En el nivel Modelo, pueden existir varios bloques Modelo 311 y 314 que representan modelos del mundo virtual. Los bloques Modelo comprenden campos 312 y 313 y bloques de tipo Segmento 309 y 307. En un ejemplo de realización preferido, los campos del bloque Modelo son los campos “Traslación”, “Rotación” y “Escalado”. Los campos definen la posición, la orientación y el tamaño del modelo en el mundo virtual. La posición, la orientación y el tamaño del modelo pueden definirse libremente (es decir, el modelo no está sujeto a ninguna limitación sobre los campos). En un ordenador, los campos Traslación, Rotación y Escalado pueden representarse mediante números de tipo real (o sea, mediante números de coma flotante).

30 En el nivel Segmento, pueden existir varios bloques Segmento 307 y 309 que representan los segmentos de un modelo. Los bloques Segmento comprenden los campos 308 y 310 y los bloques 301 y 302 de tipo elemento o bloques Segmento subordinados 305. Los campos 310 del bloque Segmento son los campos Desplazamiento, Rotación y Eje. Los campos definen la posición de un eje dentro del segmento, la rotación en términos de ángulo alrededor del eje y la orientación del eje.

35 La posición del eje define, por consiguiente, una conexión utilizada para conectar el bloque Segmento 305 con un bloque Segmento subordinado 305.

40 La posición del eje dentro del segmento está limitada a las posiciones permitidas por las restricciones del segmento. Dichas restricciones pueden imponerse por medio de un sistema tridimensional de coordenadas, en el que los elementos pueden situarse en etapas modulares. La rotación angular alrededor del eje no está limitada en un ejemplo de realización básico de la presente invención. La orientación del eje está limitada a etapas de 90 grados alrededor de tres ejes ortogonales del sistema de coordenadas (véase la Fig. 2b).

45 El campo Desplazamiento puede tener tres parámetros representados mediante números de tipo entero en un ordenador. Como alternativa, puede utilizarse el campo Traslación que tiene tres parámetros de tipo entero.

Además, el campo Rotación puede tener tres parámetros que comprenden una matriz de tres números de tipo real, y el campo Eje puede representarse por medio de una matriz de tres números de tipo entero corto o puede representarse simplemente indicando uno de los tres ejes del sistema de coordenadas, mediante bits individuales.

50 En el nivel Elemento, pueden existir varios bloques Elemento 301 y 302 que representan los elementos de un segmento. Los bloques Elemento no comprenden ningún bloque subordinado, sino sólo los campos 303 y 304. Por lo tanto, los bloques Elemento terminan la jerarquía descendente. Los campos del bloque Elemento son los campos Desplazamiento, Orientación y Referencia de forma. Los elementos de posición y orientación son las partes más limitadas de un modelo. El campo Desplazamiento es un desplazamiento tridimensional limitado a valores discretos. Preferentemente, si las restricciones se imponen por medio de un sistema tridimensional de coordenadas en el que los elementos pueden situarse en etapas modulares, el campo Desplazamiento puede representarse por medio de una matriz de números de tipo entero. Además, el campo Orientación está limitado de la misma manera que el campo Eje del segmento Bloque, es decir, está limitado a etapas de 90 grados alrededor de los tres ejes ortogonales del sistema de coordenadas. Por lo tanto, el campo Eje puede representarse mediante una matriz de tres números de tipo entero corto o puede representarse simplemente indicando uno de los tres ejes del sistema de coordenadas.

60 Por medio del campo Referencia de forma, los elementos incluyen una referencia a una descripción geométrica que puede interpretarse y utilizarse con fines de visualización. La referencia de forma es preferentemente un número de referencia que puede utilizarse en una consulta a una base de datos que comprende una descripción geométrica de los elementos individuales. Habitualmente, los modelos se basan en un grupo de elementos de tipos diferentes. La base de datos comprende una descripción geométrica de los elementos individuales según cualquier descripción geométrica preferida. Las descripciones geométricas preferidas pueden ser VRML, 3DMF, etc. Además, en la base de datos pueden almacenarse varios tipos diferentes de descripciones geométricas para cada elemento. Por consiguiente, es posible seleccionar una descripción geométrica que puede ser adecuada en algún sentido, por ejemplo, en el sentido de proporcionar una buena solución de compromiso entre la obtención de una visualización razonablemente rápida

## ES 2 226 772 T3

del modelo y la obtención de una resolución suficientemente alta de la visualización. Debe remarcar que lo que está limitado a las coordenadas enteras es sólo la posición de los elementos (no las coordenadas u otros tipos de parámetros utilizados para representar las formas de los elementos).

5 Los bloques Elemento 301 y 302 también pueden comprender otros campos que proporcionan una referencia al material del que está hecho el elemento, una referencia a la decoración o la textura. Dichos campos pueden denotarse como “Material de referencia” y “Versión de decoración”. En lo que respecta a la descripción geométrica, puede almacenarse una descripción de los materiales y la decoración o la textura en una base de datos, con un formato adecuado.

10 Por lo tanto, el efecto de limitar la estructura de un modelo según la presente invención (en comparación, por ejemplo, con la estructura del modelo VRML) se basa en el hecho de que, en la mayor parte de tareas de modelado, las estructuras del nivel inferior de la jerarquía se utilizan con más frecuencia. En consecuencia, limitando en concreto estas estructuras de nivel más bajo y adaptando una representación de estas estructuras sólo de la forma estrictamente necesaria, será posible obtener una representación del modelo muy compacta. Considerando la jerarquía de manera global, se puede decir que el bloque Modelo permite una transformación libre, el bloque Segmento permite una transformación parcialmente limitada y el bloque Elemento permite sólo una transformación muy limitada.

15 Una vez definida una posible interpretación geométrica de los elementos y los segmentos y la jerarquía en la que se organizan los elementos, segmentos, modelos y mundos, se procederá a definir un ejemplo de realización preferida de una correspondiente estructura de código que se utiliza para representar un modelo real.

20 Debe observarse que un ejemplo de realización básico de la presente invención puede comprender un solo segmento, con elementos que tienen posiciones limitadas definidas por las coordenadas de un sistema de coordenadas limitado, en un mundo libre (es decir, en un sistema arbitrario de coordenadas).

### *Estructura del código*

30 La estructura del código comprende dos tipos de primitivas: los bloques y los campos. Un bloque se define mediante un nombre de bloque y llaves { } que encierran el contenido. Un bloque puede contener un grupo de campos y otros bloques subordinados. Un campo se define mediante un nombre de campo y un grupo de parámetros situados a continuación del nombre de campo.

### *Bloques y campos*

35 A continuación, se proporcionan los bloques con su nombre de bloque seguido de su contexto entre paréntesis. El contexto es el bloque en el que pueden existir, siendo denominado Archivo el de nivel superior. El término “Archivo” se utiliza como referencia, por ejemplo, a un archivo electrónico almacenado en medios legibles por ordenador, tales como la memoria de un ordenador, un CD-ROM, un videodisco digital (DVD), un disquete, un disco duro, una memoria volátil (por ejemplo, una memoria de acceso aleatorio), etc.

Cada bloque tiene una descripción corta. Para reducir la complejidad de esta definición, los bloques subordinados se escriben con su nombre de bloque sólo.

### *Mundo (Archivo)*

45 Las primitivas importantes desde el punto de vista estructural son los bloques de la jerarquía, es decir, el bloque Mundo, el bloque Modelo, el bloque Segmento y el bloque Elemento. El bloque Mundo se refiere al archivo “Archivo” que contiene una descripción del mundo.

50 El mundo se describe mediante la siguiente sintaxis:

```
Mundo
{
55 Opciones { . . . }
Modelo { . . . }
}
```

60 En un mundo virtual según la presente invención, puede existir una serie de modelos contenidos en la sentencia “Modelo { . . . }”. No obstante, es posible colocar en el mundo otras características que pueden describirse o no según la presente invención. Dichas características pueden ser características de fondo, sonido, eventos predefinidos, etc.

65 Por otra parte, el bloque Mundo se utiliza para establecer una referencia común para los modelos (y otras características) del mundo. Preferentemente, esta referencia adopta la forma de un sistema de coordenadas.

## ES 2 226 772 T3

### *Modelo (Mundo)*

El bloque Modelo puede contener varios modelos o varios bloques Segmento. Los campos del bloque Modelo son: Rotación, Traslación y Escalado.

```
5
  Modelo
  {
  10  Rotación {real; real; real}
  10  Traslación {real; real; real}
  10  Escalado {real; real; real}
  10  Segmento{. . .}
  }
```

15 Debe mencionarse que todos los parámetros de rotación y de orientación se basan en el conocido sistema de coordenadas dextrógiro, en el que el pulgar sigue los valores positivos de uno de los ejes y los valores angulares positivos se cuentan en la dirección de los otros dedos de la mano.

20 El campo Rotación comprende tres parámetros que determinan cómo se orientará el modelo en el mundo. Preferentemente, los parámetros adoptan la forma de los conocidos ángulos de Euler. Es decir, en primer lugar, el modelo se gira según el primer parámetro de los tres que indica un ángulo alrededor de un primer eje de un sistema de coordenadas rectangular y, en segundo lugar, el modelo se gira según el segundo parámetro que indica un ángulo alrededor de un segundo eje, cuya orientación ha cambiado debido a la primera rotación del modelo alrededor del primer eje.

25 Este procedimiento continúa hasta que el modelo ha sido girado de la forma deseada mediante los tres parámetros del campo de rotación. Sin embargo, es posible aplicar otros procedimientos para indicar la orientación o la rotación de un objeto.

30 Aunque en la sintaxis del campo Rotación de la definición anterior de un modelo los parámetros son de tipo “real”, pueden utilizarse otros tipos de parámetros, tales como enteros, un tipo de parámetro dado enumerado que comprende los valores de grados posibles {0, 10, 20,..., 350} u otros.

35 El campo Traslación también comprende tres parámetros de tipo real. Cada uno de estos parámetros indica una traslación del modelo a lo largo de uno de los tres ejes ortogonales del sistema de coordenadas. Por lo tanto, el campo Traslación permite traslaciones que no están limitadas a los módulos del sistema de coordenadas modular.

### *Segmento (Modelo, Segmento)*

40 El bloque Segmento contiene una parte estructural formada por bloques Elemento y, opcionalmente, bloques Segmento subordinados. Los campos del bloque Segmento son los campos Desplazamiento, Orientación, Eje y Rotación.

```
  Segmento
  {
  45  Desplazamiento [entero; entero; entero];
  45  Orientación [2 bits; 2 bits; 2 bits];
  45  Eje [3 bits];
  45  Rotación [real];
  45  Elemento { . . . }
  50  Segmento { . . . }
  }
```

55 El campo Desplazamiento describe la posición del origen del segmento actual con respecto al origen de otro segmento que, en la jerarquía, es superior al segmento actual. Si no existe ningún segmento superior, el campo Desplazamiento se omite. El campo Desplazamiento comprende tres parámetros de tipo entero para describir la posición. Limitando esta posición a valores enteros, se obtiene una reducción significativa de la cantidad de datos necesaria para representar un modelo completo.

60 El campo Orientación (omitido en la Fig. 3) describe la orientación del segmento, es decir, indica si el segmento está orientado al revés, está girado 90 grados hacia la izquierda, etc. El campo Orientación comprende tres parámetros de dos bits para describir esta orientación en etapas de 90 grados, alrededor de uno o más de los tres ejes del sistema de coordenadas.

65 El campo Eje describe la orientación de un eje en torno al cual puede girar el segmento. Sólo se utiliza un parámetro de tres bits para indicar uno de los tres ejes del sistema de coordenadas, ya que la orientación se describe en relación con el origen del segmento actual.

## ES 2 226 772 T3

El campo Rotación describe una rotación del segmento actual alrededor del eje definido anteriormente en el campo Eje. El campo Rotación comprende un solo parámetro de tipo real. Este parámetro puede utilizarse para indicar un ángulo estático del segmento actual o puede modificarse para simular una manipulación (por ejemplo, una rotación del segmento actual con una velocidad angular especificada).

5 De este modo, mediante las limitaciones definidas anteriormente, se consigue reducir de forma significativa la cantidad de datos necesaria para representar un modelo completo. Este beneficio se obtiene sin provocar el descontento de los usuarios que crean un modelo según la presente invención, ya que en un sistema de construcción modular o una realidad virtual modular se esperan limitaciones. Sin embargo, si las restricciones son un factor limitante en una parte  
10 de una tarea de modelado, puede aplicarse, por ejemplo, un campo Traslación como el definido en el bloque Modelo para superar o eliminar las limitaciones sobre un desplazamiento, utilizando números no enteros (por ejemplo, de tipo real).

15 Las sentencias “Elemento{. . .}” y “Segmento{. . .}” indican que el segmento en cuestión puede comprender elementos y segmentos subordinados según la jerarquía de un modelo según la presente invención.

### *Elemento (Segmento)*

20 El bloque Elemento es la primitiva geométrica según la presente invención. Remite al cuerpo geométrico por medio del campo “Referencia de forma”. No obstante, pueden definirse otros campos de referencia, tales como “Referencia de material” y “Versión de decoración”. Otros campos disponibles son el campo “Orientación” y “Desplazamiento”.

```
Elemento
25 {
  Desplazamiento[entero; entero; entero];
  Orientación[2 bits; 2 bits; 2 bits];
  Referenciaforma[entero];
  Referenciamaterial[entero];
30 Referenciadecoración[entero];
}
```

35 El campo Desplazamiento se utiliza para delimitar la posición del elemento dentro de un segmento según los módulos de la retícula modular. Por lo tanto, pueden utilizarse parámetros de tipo entero. Como alternativa, puede utilizarse un tipo de parámetros que comprende una serie de caracteres alfanuméricos, en los campos Referencia de forma, Referencia de material y Referencia de decoración.

40 La orientación del elemento se define mediante el campo Orientación y tres parámetros de 2 bits. Por lo tanto, la orientación del elemento se limita a la retícula modular del segmento.

45 Para procesar un modelo concreto de forma eficaz, el bloque Elemento comprende una referencia a una descripción geométrica del elemento, en vez de toda la descripción geométrica que puede comprender una gran cantidad de datos geométricos. Por lo tanto, el aspecto de un modelo puede cambiarse intercambiando un parámetro de Referencia de forma del bloque Elemento por otro parámetro de Referencia de forma.

A continuación, se hará referencia a un ejemplo de un modelo concreto según la presente invención.

50 La Fig. 4 muestra dos familias de elementos. La primera familia comprende los elementos 401, 402 y 403. La segunda familia comprende los elementos 404, 405 y 406. Cada uno de los elementos tiene una descripción geométrica que está asociada a una referencia, de tal manera que, cuando se solicite un elemento que tiene una referencia concreta, sea posible recuperar la descripción correspondiente. La descripción geométrica puede proporcionarse designando los elementos por medio de software de diseño asistido por ordenador y, por consiguiente, generando datos con un formato especificado que pueden almacenarse en medios adecuados y extraerse de éstos. Dicho formato especificado puede ser,  
55 por ejemplo, VRML, 3DMF, etc. Por lo tanto, los datos pueden recuperarse para interpretar visualmente un modelo virtual según la presente invención.

60 Cada uno de los elementos que se designa tiene una posición y una orientación de referencia, de manera que, cuando los elementos se utilizan en un modelo, su posición y orientación en el modelo puede describirse en relación con la posición y la orientación de referencia. Para definir esta posición y orientación de referencia, los elementos de esta exposición se muestran por motivos ilustrativos sólo en un sistema de coordenadas ortogonales que tiene un eje de las X, un eje de las Y y un eje de las Z.

65 Pese a los diferentes aspectos de los elementos individuales, cada elemento tiene un respectivo módulo de referencia correspondiente a un módulo de la retícula modular de un segmento. Este módulo de referencia viene indicado con líneas punteadas o un recuadro.

A continuación, se hará referencia a los elementos concretos. El elemento 401 es un elemento en forma de L que

## ES 2 226 772 T3

ocupa tres módulos de retícula. A este elemento se le asigna la identificación de tipo "T401". El elemento 402 es un elemento cúbico que ocupa una sola unidad de retícula. A este elemento se le asigna la identificación de tipo "T402". El elemento 403 es un elemento en forma de T que ocupa cuatro unidades de retícula. A este elemento se le asigna la identificación de tipo "T403".

De la misma manera, el elemento 404 es un elemento tubular en forma de L que ocupa tres módulos de retícula. A este elemento se le asigna la identificación de tipo "T404". El elemento 405 es un elemento tubular recto que ocupa una sola unidad de retícula. A este elemento se le asigna la identificación de tipo "T405". El elemento 406 es un elemento tubular en forma de T que ocupa cuatro módulos de retícula. A este elemento se le asigna la identificación de tipo "T406".

Por lo tanto, es posible almacenar una descripción geométrica para cada uno de los elementos 401 a 406 y recuperar la descripción geométrica de un elemento especificado a partir del lenguaje de modelado virtual según la presente invención, por medio de la identificación de tipo (es decir, una referencia).

Preferentemente, los elementos de las respectivas familias se designan de tal forma que pueden estar interconectados con otros elementos de sus respectivas familias. Esto puede indicarse visualmente, designando los elementos para que comprendan una interfaz reconocible que adopta la forma de una superficie que coincide con el plano entre dos módulos de retícula vecinos. En el caso de los elementos tubulares 404, 405 y 406, los extremos de los tubos pueden constituir la interfaz reconocible. Se muestran otros elementos, por ejemplo, en las patentes U.S. n° 4.214.403; n° 4.176.493; n° 5.795.210; n° 4.124.949 y n° 5.645.463.

La Fig. 5 muestra la primera etapa de un procedimiento de construcción de un modelo de un objeto geométrico. El elemento 501 del tipo "T401" se dispone en un segmento que tiene un sistema de coordenadas ortogonales delimitado por los ejes X1, Y1 y Z1. Los ejes sólo son ilustrativos y pueden mostrarse o no cuando el modelo está sujeto a visualización.

En un ejemplo de realización preferido del modelo de realidad virtual, se establece que el modelo más pequeño posible debe comprender por lo menos un segmento, y por lo tanto el elemento 501 se dispone en un segmento. Debe observarse que el elemento 501 tiene una orientación respecto de los ejes X1, Y1 y Z1 que es distinta a la orientación de referencia de la definición del tipo de elemento (véase la Fig. 4). Esta orientación se describe preferentemente como una multiplicidad de etapas de 90 grados del elemento. El elemento 501 es orientado mediante una rotación de 270 grados alrededor del eje paralelo al eje X1, midiéndose los ángulos según un sistema dextrógiro de coordenadas (en el que el dedo pulgar sigue el eje de las X y adopta los valores positivos de la dirección circular indicada por los dedos). En lo sucesivo, se utilizará esta definición.

El código utilizado preferido para representar el segmento que tiene un único elemento 501 es el siguiente:

INICIO CÓDIGO:

```
Segmento
{
  Elemento
  {
    Referenciaforma(T401);
    Orientación(270, 0, 0);
  }
}
```

FIN CÓDIGO.

Según la sintaxis del código, se define un segmento que tiene un elemento. Además, en el campo Referencia de forma se indica que el tipo del elemento es "T401". La orientación del elemento se especifica por medio del campo Orientación y los parámetros (270, 0, 0) que indican que se ha aplicado al elemento una rotación de 270 grados en relación con el eje X1. Debe observarse que los parámetros de este ejemplo se proporcionan en grados, sin embargo, debido a la limitación de 90 grados, esta representación puede codificarse sólo en dos bits por parámetro.

Por lo tanto, este código podrá ser interpretado por una aplicación de intérprete visual que copia la descripción geométrica del elemento, calcula la posición y la orientación del elemento según la sentencia determinada en el código anterior y calcula la proyección deseada sobre un plano o una esfera para su presentación por los medios de presentación en pantalla.

Se selecciona un tamaño de retícula relativamente grosero en comparación con la resolución utilizada para presentar el elemento. Habitualmente, el tamaño de la retícula se adapta para que tenga capacidad sólo para manejar la interconexión o simplemente permitir la alineación en algún sentido de los elementos de una familia de elementos.

## ES 2 226 772 T3

A pesar de que los elementos se sitúan en un sistema de coordenadas por medio de coordenadas enteras, la definición o la representación de las formas geométricas de los elementos puede comprender coordenadas no enteras. El aspecto importante es que la posición de los elementos se representa por medio de enteros.

5 La Fig. 6 muestra la segunda etapa de un procedimiento de construcción de un modelo de objeto geométrico. En esta etapa, se añaden más elementos y el efecto del modelo virtual según la presente invención resultará más evidente. En este caso, el objeto geométrico comprende tres elementos 601, 602 y 603 de los respectivos tipos “T401”, “T403” y “T402”. La posición y la orientación de los respectivos elementos se proporcionan mediante el código siguiente:

10 INICIO CÓDIGO:

```
Segmento
{
15 Elemento
{
Referenciaforma (T401);
Orientación (270, 0, 0);
20 }
Elemento
{
Referenciaforma (T403);
25 Orientación (0, 0, 270);
Desplazamiento (3, 0, 0);
}
Elemento
30 {
Referenciaforma (T402);
Desplazamiento (3, 2, 0);
}
}
35
```

FIN CÓDIGO.

40 La primera parte del código que define el elemento 601 ya se ha introducido. Como se verá, los otros elementos (602 y 603) del segmento se definen en relación con este elemento (601).

45 La siguiente parte de este código define el elemento 602 del tipo “T403”. Como se observa, este elemento no tiene la misma orientación que la de la definición del tipo de elemento; por consiguiente, es necesario añadir un campo Orientación que indique que el elemento ha girado 270 grados alrededor del eje Z1. Además, al elemento 602 se le aplica un desplazamiento que tiene un valor entero de tres módulos a lo largo del eje X1.

La última parte del código define la posición del elemento 603 del tipo “T402”. A este elemento se le ha aplicado un desplazamiento de un valor entero de tres módulos a lo largo del eje X1 y un desplazamiento de un valor entero de dos módulos a lo largo del eje Y1.

50 Debido a que los elementos de un segmento pueden situarse sólo según los módulos de la retícula, los parámetros utilizados para definir la posición y la orientación de los elementos pueden representarse con mucha eficacia (por ejemplo, mediante valores enteros). De esta forma, es posible añadir más elementos al segmento, si así se desea.

55 En cuanto a la manipulación, es fácil de determinar que los elementos del segmento no pueden manipularse unos en relación con otros, debido a las restricciones impuestas por la retícula modular.

60 La Fig. 7 muestra la segunda etapa de un procedimiento de creación del modelo de un objeto geométrico, según la presente invención. En esta etapa, se adjuntan al modelo todavía más elementos y el efecto del modelo virtual según la presente invención es todavía más evidente. Además, en esta etapa se añaden elementos que no están necesariamente sujetos a las restricciones de posición y orientación impuestas sobre un segmento. Por este motivo, es necesario ampliar la descripción del modelo. Sin embargo, ya que esta ampliación de la descripción sólo se utiliza cuando es necesario (por ejemplo, para describir rotaciones), es posible mantener un modelo globalmente compacto y adecuado.

65 En este caso, el objeto geométrico comprende cinco elementos 701, 702, 703, 704 y 705 de los respectivos tipos “T401”, “T403”, “T402”, “T403” y “T401”. La posición y la orientación de los respectivos elementos se proporcionan mediante el código siguiente:

## ES 2 226 772 T3

INICIO CÓDIGO:

```
Segmento
5 {
  Elemento
  {
10 Referenciaforma (T401);
    Orientación (270, 0, 0);
  }
  Elemento
  {
15 Referenciaforma (T403);
    Orientación (0, 0, 270);
    Desplazamiento (3, 0, 0);
  }
  Elemento
20 {
    Referenciaforma (T402)
    Desplazamiento (3, 2, 0);
  }
25 Segmento
  {
    Desplazamiento (0, 0, 2);
    Orientación (90, 0, 0);
30 Rotación (225);
    Eje (0, 0, 1);
  Elemento
  {
35 Referenciaforma (T403);
    Desplazamiento (0, -1, 0);
  }
  Elemento
  {
40 Referenciaforma (T401);
    Desplazamiento (-3, -1, 0);
    Orientación (270, 0, 0);
  }
45 }
}
```

FIN CÓDIGO.

50 La primera parte del código define los elementos 701, 702 y 703 introducidos en la Fig. 6. La siguiente parte del código define otro segmento que se gira 45 grados respecto de un eje paralelo al eje de las Z (es decir, el eje vertical) y que tiene los elementos 704 y 705.

55 La parte del código que define los elementos 704 y 705 puede generarse mediante procedimientos diferentes, no obstante, el procedimiento paso a paso que comprende las etapas 1 a 4 proporcionado a continuación es el que se considera más fácil de comprender.

Etapas 1

60 En esta etapa de generación del código, se decide (por ejemplo, en respuesta a una instrucción del usuario) que se va a añadir un nuevo segmento al modelo. Teniendo esto en cuenta, el elemento 704 se añade como primer elemento del nuevo segmento que tiene una retícula modular (no mostrada) coincidente con la retícula o el sistema de coordenadas indicado en la definición del elemento del tipo "T403" (véase la Fig. 4). Por lo tanto, no es necesario especificar ni el campo Desplazamiento ni el campo Orientación. A continuación, se añade el elemento 705 al nuevo segmento. Este elemento presenta, sin embargo, un desplazamiento de un valor entero de -3 a lo largo del eje X1 y una orientación con un giro de 270 grados alrededor del eje de las X, respecto de la orientación del elemento de la definición de elemento del tipo "T401". Esto viene definido en los campos Desplazamiento y Orientación del código siguiente:

## ES 2 226 772 T3

INICIO CÓDIGO:

```
5   Segmento
    {
    Elemento
    {
    Referenciaforma(T403);
    }
10  Elemento
    {
    Referenciaforma (T401);
    Desplazamiento (-3, 0, 0);
15  Orientación (270, 0, 0);
    }
    }
```

20 FIN CÓDIGO.

Etapa 2

25 En esta etapa de generación del código, se determina que el origen del nuevo segmento está situado en el lugar en que el nuevo segmento se conecta al antiguo segmento, es decir, en la parte inferior del elemento en forma de T 704. Por consiguiente, las posiciones de los elementos 704 y 705 se vuelven a calcular para que concuerden con el nuevo origen del segmento.

INICIO CÓDIGO:

```
30   Segmento
    {
    Elemento
35   {
    Referenciaforma (T403);
    Desplazamiento (0, -1, 0)           // añadido
    }
40   Elemento
    {
    Referenciaforma(T401);
    Desplazamiento(-3, -1, 0);        // modificado
45   Orientación(270, 0, 0);
    }
    }
```

50 FIN CÓDIGO.

Debe observarse que esto incluye la adición de un campo Desplazamiento del elemento 704 y la modificación del campo Desplazamiento del elemento 705 para reflejar este nuevo origen.

55 Este nuevo cálculo también debe llevarse a cabo si el nuevo segmento está conectado al antiguo en una posición diferente del nuevo segmento (por ejemplo, como se muestra en la Fig. 2b). Pero, puesto que los parámetros del campo Desplazamiento son de tipo entero, este nuevo cálculo puede llevarse a cabo mucho más rápido que cuando se utilizan parámetros de tipo real. A pesar de que se efectúan muchas operaciones de suma y resta en el modelo práctico, no se introducen errores de redondeo debido a la representación entera.

60 Etapa 3

En esta etapa de generación del código, se indica dónde se interconectará el nuevo segmento con el antiguo segmento. Esto se lleva a cabo indicando qué traslación ha sufrido el origen del nuevo segmento en relación con el origen del antiguo segmento, es decir, insertando el siguiente campo Desplazamiento en el segundo bloque Segmento del código proporcionado en la etapa 2 anterior:

Desplazamiento (0, 0, 2);

## ES 2 226 772 T3

Por lo tanto, el origen del nuevo segmento sufre una traslación de valor entero 2 a lo largo del eje Z1.

### Etapa 4

5 Una vez delimitada la posición del nuevo segmento y de los elementos del nuevo segmento, puede especificarse la orientación del nuevo segmento en relación con el antiguo segmento. Esto se efectúa insertando el siguiente campo "Rotación" en el código dado en la etapa 2 anterior:

Rotación (90, 225, 0);

10

Por lo tanto, en primer lugar, el nuevo segmento se gira 90 grados alrededor del eje de las X y, en segundo lugar, se gira 225 grados alrededor del eje de las Y que ya ha sido girado 90 grados de la forma indicada.

15

En cuanto a la manipulación, como se ha indicado anteriormente, es fácil de determinar que los elementos de los segmentos no pueden manipularse unos en relación con otros, debido a las restricciones impuestas por la retícula modular. Sin embargo, también se determina con facilidad que los segmentos pueden girar unos respecto de otros, debido a la definición implícita de los segmentos. Por lo tanto, no es necesario definir ninguna metadescripción adicional, es decir, ninguna definición de las posibles manipulaciones (movimientos o rotaciones). Si se desea trasladar los segmentos unos respecto de los otros sin restricciones de las respectivas retículas modulares, puede añadirse un campo Traslación simple que defina la traslación.

20

La Fig. 8 muestra un tercer objeto simple para modelar, según la presente invención. Este objeto puede describirse con el mismo modelo que el descrito para la Fig. 7 anterior, sólo con el tipo de los respectivos elementos definidos por medio de los parámetros del campo "Referencia de forma" intercambiados con los respectivos tipos de elementos de la segunda familia de elementos definidos en la Fig. 4. El modelo comprende los elementos 801, 802, 803, 804 y 805 correspondientes a los elementos 701, 702, 703, 704 y 705 de la Fig. 7, respectivamente.

25

De esta forma se ilustra que es posible utilizar otra familia de elementos del modelo sin perder la eficacia de la descripción. En un ejemplo de realización preferido, la familia de elementos se adapta a una clase de paradigmas de modelado, de tal forma que los elementos representan objetos del mundo real muy conocidos que pueden resultar factibles para una tarea de modelado dada.

30

Además, según la exposición anterior de un ejemplo de realización preferido y según un objetivo de la presente invención, resulta evidente para los expertos en la materia que la metadescripción (descripción de las posibles manipulaciones de un objeto geométrico) se proporciona de forma implícita por medio de la descripción geométrica, modificando los parámetros de las respectivas sentencias. Si se considera, por ejemplo, la interconexión de dos segmentos, dicha interconexión es determinada por medio del campo Eje y el campo Rotación. Cuando se modifica el ángulo de rotación, la descripción geométrica cambia mientras tiene lugar la manipulación (es decir, la rotación). En tal caso, el eje de la rotación se define en la descripción geométrica y, por lo tanto, la metadescripción queda insertada en la descripción geométrica. En lo que se refiere a los elementos de un segmento, resulta evidente que estos elementos no pueden estar sujetos a ninguna rotación, puesto que no existe ningún parámetro en la descripción geométrica que permita una metadescripción de rotación. No obstante, en un ejemplo de realización práctico, el usuario dispone de controles para aplicar nuevas metadescripciones si es necesario. De hecho, dichas nuevas metadescripciones son fáciles de aplicar, porque sólo existe un grupo limitado de descripciones en la estructura del modelo y porque los parámetros están limitados a un grupo restringido de valores.

35

40

45

Una vez definida la estructura de los modelos según la presente invención y las modificaciones y manipulaciones de los modelos, se dará a conocer un sistema informático para ejecutar estas tareas.

50

55

La Fig. 9 muestra un sistema informático de cliente para designar, almacenar, manipular y compartir construcciones geométricas según la presente invención. El sistema informático del cliente designado de forma general por 901 puede utilizarse como sistema un autónomo o como un cliente en un sistema cliente/servidor. El cliente comprende memoria 902 parcialmente ejecutada como medios de memoria volátil y no volátil (por ejemplo, un disco duro y una memoria de acceso aleatorio, RAM). La memoria comprende procedimientos o programas, un intérprete de código 907, un generador de código 908, un manejador de eventos de interfaz de usuario (UI) 909 y una aplicación de modelado 910 ejecutable por la unidad central de procesamiento 903. Además, la memoria comprende datos de modelos 911.

El intérprete de código 907 está adaptado para leer e interpretar el código que define un modelo según la presente invención. En un ejemplo de realización preferido, el intérprete de código está adaptado para leer un modelo según la presente invención y convertir dicho modelo en un modelo de formato gráfico conocido para su presentación en la pantalla de un ordenador. Cuando la persona experta en la materia conoce la estructura de código definida anteriormente para representar el modelo de un objeto, podrá llevar a cabo esta conversión aplicando principios gráficos muy conocidos dentro del campo de la informática gráfica. A continuación, se proporciona una lista incompleta de las funciones ejecutadas por el intérprete de código 907:

65

Función 1: cuando se lee el bloque Mundo, el intérprete de código (CI) establece un sistema de coordenadas al cual se referirán las sentencias posteriores.

## ES 2 226 772 T3

Función 2: cuando se lee el bloque Modelo, el CI se prepara para colocar un modelo que comprende segmentos y elementos dentro del sistema de coordenadas definido por la función 1. Esto se lleva a cabo según los campos Rotación y Traslación.

5      Función 3: cuando se lee el bloque Segmento, el CI se prepara para colocar un segmento que comprende elementos, según la posición y orientación indicadas en la función 2 y según el campo Orientación que indica la orientación del segmento. Además, se leen los campos Desplazamiento, Eje y Rotación para determinar la posición de los elementos subordinados.

10     Función 4: cuando se lee el bloque Elemento, el CI lee la descripción geométrica del elemento indicado por la referencia de tipo de elemento en el campo Referencia de forma, accediendo a una base de datos almacenada, por ejemplo, en forma de datos de modelos 911.

15     El resultado de las funciones 1, 2, 3 y 4 invocado de forma secuencial o iterativa se proporciona en forma de datos proyectados sobre un plano o una esfera o en forma de datos que pueden someterse a una proyección.

20     El manejador de eventos de UI 909 está adaptado para convertir una interacción de usuario con una interfaz de usuario en instrucciones de usuario adecuadas que puedan ser reconocidas por el generador de código 908. Un grupo de posibles instrucciones reconocibles puede comprender: obtener un elemento de una biblioteca de elementos, conectar un elemento con otro elemento, desconectar un elemento, rechazar un elemento, manipular un segmento (por ejemplo, iniciar una rotación), etc. Cada instrucción tiene un respectivo grupo de parámetros asociados como el definido anteriormente para cada uno de los campos.

25     El generador de código 908 está adaptado para modificar el código que describe un modelo concreto según la presente invención, de la forma descrita anteriormente y en respuesta a las instrucciones de un usuario. A continuación, se proporciona una lista incompleta de las funciones ejecutadas por el generador de código 908 (se presupone que se ha determinado un segmento previamente).

30     Función 5: cuando se selecciona un elemento de una biblioteca de elementos, su tipo se indica en el campo Referencia de forma.

35     Función 6: cuando se conecta el elemento a un elemento de un segmento, el segmento se identifica y la posición y la orientación del elemento conectado dentro del segmento se determinan y representan por medio de los campos Desplazamiento y Orientación que se insertan en la representación del modelo.

Función 7: cuando se desconecta un elemento, se suprime el código que representa la posición y la orientación del elemento.

40     Como tarea concurrente o posterior, el intérprete de código puede ejecutarse para presentar el resultado del generador de código.

La aplicación de modelado 910 está adaptada para controlar la memoria, los archivos, la interfaz de usuario, etc.

45     El usuario 905 es capaz de interactuar con el sistema informático del cliente 901 por medio de la interfaz de usuario 906.

50     Para copiar modelos, descripciones geométricas u otros datos, el sistema informático del cliente comprende una unidad de entrada/salida (E/S) 904. La unidad de entrada/salida puede ser utilizada como interfaz para diferentes tipos de medios de almacenamiento y diferentes tipos de redes informáticas (por ejemplo, Internet). Además, la unidad de entrada/salida (I/O) 904 puede utilizarse para intercambiar modelos con otros usuarios (por ejemplo, de forma interactiva).

55     El intercambio de datos entre la memoria 902, la unidad central de procesamiento (CPU) 903, la interfaz de usuario (UI) 906 y la unidad de entrada/salida 904 se lleva a cabo mediante el bus de datos 912.

60     La Fig. 10 muestra un sistema informático de servidor para manejar las construcciones geométricas según la presente invención. El sistema informático de servidor, designado de forma general por 1001, se utiliza para servir a un grupo de sistemas informáticos de clientes. El servidor comprende una memoria 1002 ejecutada en parte como medios de memoria volátil y no volátil (por ejemplo, un disco duro y una memoria de acceso aleatorio, RAM). La memoria comprende procedimientos o programas, el “intérprete de código” 1005, el “generador de código” 1006 y el “manejador de sesiones” 1007 ejecutables por la unidad central de procesamiento 1003. Además, la memoria comprende “datos de modelos” 1008, por ejemplo, en forma de descripciones geométricas, descripciones de modelos, etc.

65     Para copiar modelos, descripciones geométricas u otros datos o para distribuir datos a los sistemas informáticos de clientes, el sistema informático de servidor comprende una unidad de entrada/salida (E/S) 1004. La unidad de entrada/salida puede utilizarse como interfaz para diferentes tipos de medios de almacenamiento y diferentes tipos de redes informáticas (por ejemplo, Internet).

## ES 2 226 772 T3

El intercambio de datos entre la memoria 1002, la unidad central de procesamiento (CPU) 1003 y la unidad de entrada/salida 1004 se lleva a cabo mediante el bus de datos 1009.

5 A continuación, se procederá a describir en detalle las funciones/programas informáticos a los que se ha aludido en la descripción de las Fig. 9 y 10.

10 La Fig. 11 muestra un diagrama de flujo de un procedimiento para construir de forma simultánea segmentos y modelos mientras se someten los segmentos y modelos a manipulación. Este diagrama de flujo también describe el funcionamiento de un procedimiento ejecutado en un ordenador de cliente o servidor para procesar un modelo según la presente invención.

15 En la etapa 1101, se reciben las instrucciones del usuario introducidas por medio de los controles de usuario que forman parte de la interfaz de usuario 1102. Los controles de usuario pueden ser de muchos tipos diferentes; por ejemplo, pueden comprender un teclado de ordenador, una palanca de mando o un ratón que interactúa con una pantalla de usuario 1121, mediante los cuales se proporcionan señales a un programa para representar la interacción del usuario con la interfaz de usuario. Por ejemplo, un usuario puede seleccionar un elemento mostrado en la pantalla de usuario 1121 y llevarlo a la posición deseada mediante el ratón del ordenador, realizando la conocida operación de arrastrar y soltar. En consecuencia, se genera una instrucción de usuario que representa este movimiento del elemento. Dicha operación de la interfaz de usuario también se conoce como “programación basada en eventos”.

20 La instrucción del usuario se recibe en el bloque 1113 del diagrama de flujo y se utiliza para crear, actualizar o manipular datos de modelos 1112 según la presente invención. Los datos de modelos 1112 obtenidos como resultado del procesamiento del bloque 1113 pueden transferirse a una red de comunicación por medio de un dispositivo de entrada/salida 1117. Por lo tanto, es posible intercambiar modelos con otros usuarios, compartir modelos, copiar modelos de un servidor, guardar modelos en un servidor, etc. Habitualmente, los datos de modelos 1112 se combinan con datos que representan el aspecto de los elementos almacenados por medio del convertidor 1118. El convertidor utiliza el campo “Referencia de forma” definido anteriormente para obtener una representación geométrica de este aspecto desde la base de datos 1119. Además, en el convertidor 1118, la representación geométrica y los bloques y campos descritos anteriormente, que describen la forma de un modelo concreto, se combinan y pasan a un formato que puede ser interpretado por una máquina de realidad virtual 3D 1120. La máquina de realidad virtual 3D 1120 proporciona información que representa el modelo y permite visualizar el modelo en una pantalla de usuario 1121.

A continuación, se procederá a describir en detalle el bloque 1113 del diagrama de flujo.

35 Cuando se recibe una instrucción de usuario, se determina si la instrucción de usuario se refiere a una operación de elemento, una operación de segmento o una operación de modelo. Se detecta una operación de elemento en la etapa 1116 si se añade, extrae o cambia de una primera a una segunda posición un elemento del modelo. Se detecta una operación de segmento en la etapa 1115 si se desplaza, extrae o manipula un segmento del modelo, o si se añade un nuevo segmento al modelo. Se detecta una operación de modelo en la etapa 1114, si se manipula un modelo (es decir, si se desplaza libremente sin limitarlo a la retícula o si un segmento se desplaza libremente sin limitarlo tampoco a la retícula).

Dependiendo de la operación detectada, se crearán, actualizarán o manipularán los datos de los modelos.

45 Si el usuario selecciona un elemento de una biblioteca de elementos (que comprende, por ejemplo, los elementos mostrados en la Fig. 4), se inicia la etapa 1110 de obtención de un nuevo elemento y se determina que el elemento seleccionado es, por ejemplo, del tipo “T401”. A continuación, en la etapa 1111, el elemento seleccionado se conecta a otro elemento que ya se hallaba en el mundo o se coloca solo y por consiguiente se trata como un segmento. Cuando se conecta el elemento seleccionado, el código que representa a este elemento, su posición y su orientación se adjunta a un código existente, o se genera un nuevo código.

50 Si existe otro elemento en el mundo, el usuario puede decidir seleccionar este elemento y cambiarlo de su posición actual o desconectarlo. Si se detecta dicho evento, la parte del código existente que representa el elemento se retira y se almacena temporalmente.

55 El elemento seleccionado puede ser rechazado o puede ser conectado o situado en otra posición. Si se detecta el evento de rechazo del elemento, el elemento desaparecerá de la interfaz del usuario y el código temporal se suprimirá en la etapa 1109. En cambio, si se detecta el evento de conexión del elemento en otra posición, se generará el código que representa al elemento, su nueva posición y orientación, y se insertará en el código existente.

60 Para acelerar el procedimiento de generación o revisión de un modelo existente, puede tratarse todo un segmento que comprende otros segmentos y elementos como un solo objeto.

65 Si existe algún segmento en el mundo, el usuario puede decidir seleccionar este segmento y cambiarlo de su posición actual o desconectarlo, mediante una acción que es común a todos los segmentos y elementos del segmento sujeto a movimiento o desconexión. Si se detecta dicho evento, la parte del código existente que representa todo el segmento se retira y almacena temporalmente.

Si se detecta el evento de rechazo de todo el segmento, el segmento completo desaparecerá de la interfaz de usuario y el código temporal se suprimirá en la etapa 1106. Por otra parte, si se detecta el evento de conexión del elemento en otra posición, se generará el código que representa el segmento, su nueva posición y orientación y se insertará en el código existente.

5 Además, es posible manipular los segmentos y desplazar los modelos. Cuando en el mundo existen segmentos, éstos pueden manipularse (es decir, los elementos u otros segmentos del segmento sujeto a manipulación no se cambian, pero puede aplicarse, por ejemplo, una rotación con un ángulo especificado o una velocidad angular de rotación especificada al segmento. Esto puede llevarse a cabo detectando un evento indicado desde la interfaz de usuario y, en la etapa 1104, modificando el código del segmento sujeto a manipulación (por ejemplo, modificando los parámetros del campo Rotación de un bloque Segmento).

10 Cuando existen modelos en el mundo, éstos pueden desplazarse de forma parecida a la descrita para la manipulación de un segmento, pero los parámetros que se modifican son los parámetros de los campos Rotación y Traslación de un bloque Modelo en la etapa 1103.

En un evento predefinido, el procedimiento puede terminarse o detenerse en la etapa 1112 que comprende, por ejemplo, el almacenamiento de los segmentos y modelos creados.

20 La Fig. 12 muestra un diagrama de flujo de un procedimiento para acceder y compartir un mundo común accesible por múltiples clientes. En un ejemplo de realización preferido de la presente invención, el servidor posee un modelo de un mundo. El modelo puede distribuirse a un grupo de clientes que están autorizados para manipular o cambiar de algún modo los segmentos y modelos del mundo o para añadir nuevos modelos y segmentos al mundo. Los cambios efectuados por los respectivos clientes pueden transferirse a otros clientes que comparten el mismo mundo. De esta forma, un respectivo cliente verá cambiar su copia local del mundo cuando otros clientes efectúen cambios en el modelo del mundo. Dicha actualización de la copia local del mundo del cliente debe ejecutarse con mucha rapidez y eficacia si va a operarse con un gran número de usuarios y desean evitarse tiempos de actualización desmesurados. La presente invención cumple este objetivo y constituye la clave para proporcionar un mundo común accesible para múltiples clientes, que presenta cambios constantes debido a la interacción de los diferentes clientes con el mundo.

30 En la primera etapa 1201 del diagrama de flujo para copiar el mundo, se ejecuta un inicio de sesión entre un cliente y el servidor, que comprende la identificación del nombre del cliente, un protocolo de transferencia de datos adecuado, etc. En la siguiente etapa 1202, la descripción del mundo que posee el servidor se transfiere al cliente. Antes, durante o después de esta transferencia de datos, el cliente y el servidor intercambian información para establecer si el cliente conoce todos los componentes del modelo (modelos, segmentos, elementos, etc.). Esta verificación se lleva a cabo en la etapa 1203. Si existe algún modelo desconocido en el mundo, se transfiere la descripción adecuada de los modelos pertinentes en la etapa 1204. Si no existe ningún modelo desconocido, se considera que el cliente tiene una copia del mundo y, en consecuencia, se le permite manipular modelos y segmentos o efectuar otros cambios en el mundo.

40 A continuación, se presenta un procedimiento para proporcionar al cliente más información de acuerdo con la estructura jerárquica de un modelo según la presente invención. Se supone que los modelos, segmentos, elementos y otros componentes del modelo tienen un número de identificación exclusivo. De ese modo, por ejemplo, el servidor puede preguntar al cliente si existe un modelo concreto. Si el modelo concreto está presente, no es necesario transferir más datos; en caso contrario, el servidor puede preguntar al cliente si existe algún segmento concreto del modelo concreto. De esta forma, sólo será necesario transferir al cliente los componentes de los modelos desconocidos.

50 La Fig. 13 muestra el diagrama de flujo de un procedimiento para obtener modelos, segmentos y elementos desconocidos. No obstante, sólo se muestra cómo se obtienen los modelos desconocidos, porque el diagrama de flujo puede aplicarse de forma general a la obtención de modelos, segmentos y elementos investigando en secuencia los respectivos niveles de la jerarquía de los modelos.

60 En la primera etapa 1301 del diagrama de flujo "Obtención de modelo", se inicia el procedimiento de obtención de un modelo. En la etapa 1302, se verifica si el cliente dispone del modelo en cualquier forma. De ser así, el modelo se transfiere localmente a una base de datos en tiempo de ejecución en la etapa 1305. La base de datos en tiempo de ejecución es responsable de mantener los componentes que forman parte del mundo con el que el cliente desea interactuar. Como alternativa, si el modelo no está disponible en el entorno local sino en el remoto (por ejemplo, en el servidor), el modelo se transfiere desde el servidor a la base de datos en tiempo de ejecución situada en el cliente. Esta decisión se toma en la etapa 1303 y los datos se transfieren en la etapa 1305. Si en la etapa 1303 se decide que el modelo tampoco se halla en el entorno remoto, en la etapa 1304 se gestiona un error.

65 Si los modelos que están sujetos a transferencia desde el servidor hasta el cliente comprenden segmentos desconocidos por el cliente, estos segmentos se intercambian igualmente para los modelos desconocidos. En la etapa 1306, se verifica si existe algún segmento desconocido en los modelos intercambiados. Si existe algún segmento desconocido, éste se intercambia en la etapa 1308; de lo contrario, se permite que el cliente continúe modificando el modelo del mundo en la etapa 1307.

El ejemplo de realización preferido, mostrado en las Figuras 9, 10, 11, 12 y 13, puede ejecutarse en hardware, software o una combinación de ambos.

El software puede distribuirse mediante cualquier medio de almacenamiento de datos o de transmisión de datos. Los medios comprenden disquetes, CD-ROM, minidiscos, discos compactos o una red (por ejemplo, Internet). Por medio de la red, puede distribuirse un ejemplo de realización de software de la presente invención, descargando un programa informático.

Además, pueden distribuirse o intercambiarse descripciones geométricas de los elementos, partes de modelos o modelos completos por medio de cualquiera de los medios de almacenamiento de datos o de transmisión de datos mencionados anteriormente.

Debe observarse que, aunque la representación de los parámetros definidos para los respectivos campos se lleva a cabo por medio de letras y números fáciles de leer, la representación de un modelo según la presente invención no está en absoluto limitada a dicha representación. En realidad, habitualmente se utilizará una representación comprimida o codificada de los parámetros, efectuada mediante un procedimiento conocido dentro del campo de la compresión de datos.

Por otra parte, debe observarse que las sentencias de la estructura de código pueden convertirse en bits en una representación simbólica. De este modo, la representación simbólica puede transferirse entre los ordenadores como una estructura de datos compacta o un paquete de datos.

En las referencias citadas a continuación, se dan a conocer elementos preferidos que se van a describir mediante una descripción geométrica (por ejemplo, VRML, 3DMF, etc.) y a utilizar como elementos de un modelo según la presente invención.

La patente U.S. nº 5.645.463 cedida a Interlego AG da a conocer una reproducción de un elemento de construcción de juguete que comprende una placa de asiento y un disco montado sobre un eje en una abertura circular de la placa de asiento. Además de la placa de asiento, el disco comprende medios de conexión distribuidos en una retícula modular. En un ejemplo de realización preferido, el elemento de construcción se modela según la presente invención, tomando la placa de asiento como primer segmento o parte del primer segmento y tomando el disco como segundo segmento o parte de un segundo segmento. A partir de la descripción anterior de la presente invención, se comprenderá que es posible modelar una rotación del disco en la placa de asiento mediante el campo Eje y el campo Rotación. Cuando se modela el eje de rotación, éste se sitúa en el centro de la placa de asiento en una dirección perpendicular al plano de la placa de asiento.

La patente U.S. nº 4.214.403 cedida a Interlego AG da a conocer una reproducción de elemento de bisagra que comprende un par de reproducciones de bloques de construcción de juguete interconectados a través de una bisagra, por medio de un casquillo tubular situado en una esquina de un primer bloque de los dos y un perno giratorio complementario situado en el segundo bloque de los dos. En un ejemplo de realización preferido, el elemento de construcción se modela según la presente invención tomando el primer bloque como primer segmento y el segundo bloque como segundo segmento. Es posible modelar una rotación del primer bloque en relación con el segundo bloque mediante el campo Eje y el campo Rotación. Cuando se modela el eje de rotación, éste se sitúa en el centro del perno giratorio y sigue la dirección del perno giratorio. En la descripción de la patente US nº 4.214.403, se indica que ambos bloques pueden conectarse a otros bloques mediante una interconexión rígida. Estos otros bloques pueden modelarse como elementos del primer o el segundo segmento, respectivamente.

La patente U.S. nº 5.645.463 cedida a Interlego AG da a conocer otra reproducción de elemento de construcción de juguete que comprende varios tipos de medios de conexión, tales como orificios para sostener ejes, varillas y pestañas de conexión y medios de sujeción para interconectar los elementos a presión. También es posible modelar esta reproducción de elemento de construcción de juguete según la presente invención, incluyendo modelos de los elementos con los que está adaptado para interconectarse. Pueden conectarse otros elementos con el elemento dado a conocer mediante los medios de conexión mencionados anteriormente. Cuando dicho elemento se modela según la presente invención, es posible especificar el elemento como un segmento que admite la interconexión con otros segmentos en posiciones que coinciden con la posición de los medios de conexión. Esto se lleva a cabo adjuntando los campos Eje y Rotación descritos anteriormente.

La patente U.S. nº 5.061.219 cedida a Magic Mold Corporation da a conocer una reproducción de elemento de construcción de juguete que comprende varios tipos de medios de conexión, tales como orificios para sostener ejes, conectores con múltiples bocas de conexión que pueden tener uno o más zócalos de sujeción, y elementos estructurales (con configuración de tipo barra, por ejemplo) cuyas partes terminales están configuradas para introducirse en los manguitos de sujeción. La reproducción de elemento de construcción de juguete puede modelarse como un elemento que tiene cuatro orientaciones posibles con respecto a un primer eje, cuatro orientaciones posibles con respecto a un segundo eje que es ortogonal al primer eje y ocho orientaciones posibles con respecto a un tercer eje que es ortogonal al primer y al segundo eje. El orificio central del conector con múltiples bocas de conexión puede utilizarse como referencia para describir la posición de la reproducción del elemento constructivo.

Se dan a conocer otras reproducciones de elementos de construcción de juguete en las patentes U.S. nº 5.423.707; nº 5.137.486; nº 5.199.919; nº 5.421.762 y nº 5.238.438.

## ES 2 226 772 T3

Como resulta evidente, la presente invención puede utilizarse para modelar otros sistemas de construcción modular aparte de los mostrados aquí.

5 Aunque la presente invención se ha descrito con referencia a un sistema rectangular de coordenadas, pueden utilizarse otros sistemas de coordenadas para las definiciones utilizadas según la presente invención.

10 El código mostrado se ha seleccionado como ejemplo de realización preferido y, en consecuencia, la presente invención no está limitada al mismo. Por lo tanto, la presente invención no está limitada a las sentencias de bloque y campo concretas (por ejemplo, bloque Segmento, campo Traslación, etc.). La representación de los parámetros (por ejemplo, un valor de desplazamiento de 4, un ángulo de rotación de 90 grados, etc.) en esta exposición adopta la forma de una presentación fácil de leer y comprender. Sin embargo, los expertos en la materia conocen muy bien el mecanismo para convertir esta representación en una representación suficientemente adecuada para el almacenamiento en una estructura de datos de unos medios legibles por ordenador.

15 La estructura de datos puede considerarse una representación directa de las sentencias de bloque y campo que incluyen parámetros asociados y llaves para indicar el alcance de un bloque. De esta forma, la estructura de datos puede almacenarse, por ejemplo, como una representación ASCII directa. Por otra parte, la estructura de datos puede considerarse como una estructura de lista de elementos codificados en la que cada elemento representa un bloque o un campo identificado mediante un código binario, y en la que los parámetros asociados a los campos se almacenan, por ejemplo, con un formato de codificación binaria. No obstante, pueden elegirse otras estructuras de datos adecuadas para almacenar la información necesaria según la presente invención. Si se considera que la estructura de datos es una representación directa del código utilizado en las definiciones de los bloques y campos y utilizado en los ejemplos, la jerarquía de los bloques y campos se inserta de forma automática en la estructura de datos debido al alcance de los bloques establecido por las llaves. No obstante, esta jerarquía puede indicarse de otras formas, por ejemplo, indicando de qué segmento superior, modelo y mundo forma parte cada elemento e indicando asimismo de qué modelo y mundo forma parte cada segmento. Otra posibilidad es especificar la jerarquía identificando un bloque superior.

30 En la descripción, el término “objeto geométrico” debe interpretarse como un objeto del mundo real o del mundo virtual que está sujeto a modelado según la presente invención.

En la descripción, el término “manipulación” debe interpretarse en sentido muy amplio para comprender la acción de desplazar o girar partes de la interpretación geométrica de un modelo.

35 En la descripción, el término “metadescripción” es la descripción concreta de una manipulación.

El término general “modelar” o “modelo según la presente invención” no debe confundirse con el término “bloque Modelo”.

40 La idea básica de la presente invención es que el modelo esté muy restringido, lo que a primera vista parece limitar las capacidades del concepto de modelado. No obstante, en realidad esto permite seleccionar una representación muy compacta del modelo que conduce a una estructura de modelo muy flexible, en la medida en que su representación es muy compacta y de fácil acceso, permitiendo de ese modo un procesamiento rápido y eficaz en la modificación, interacción o manipulación de un modelo según la presente invención. Además, es posible transmitir con mucha rapidez un modelo de un objeto bastante complejo por medio de una red de transmisión.

45

50

55

60

65

REIVINDICACIONES

- 5 1. Procedimiento para generar un modelo legible por ordenador de un objeto geométrico, siendo generado el modelo en un ordenador que tiene acceso a una base de datos que incluye representaciones de formas geométricas para un grupo de elementos, comprendiendo el procedimiento las etapas siguientes:
- 10 - codificación de bits en una primera estructura de datos para determinar un primer conjunto de elementos a partir del grupo y representar las posiciones restringidas de los elementos por medio de coordenadas enteras en un primer sistema de coordenadas;
  - 15 - codificación de bits en una segunda estructura de datos para representar una transformación espacial del segundo sistema de coordenadas en relación con el primer sistema de coordenadas; y
  - 15 - obtención de la primera y la segunda estructura de datos como modelo legible por ordenador de un objeto geométrico.
2. Procedimiento según la reivindicación 1, que comprende además las etapas siguientes:
- 20 - codificación de bits en una tercera estructura de datos para determinar un segundo conjunto de elementos a partir del grupo y representar las posiciones de los elementos por medio de coordenadas enteras en un tercer sistema de coordenadas, e incluir estos bits en el modelo legible por ordenador; y
  - 25 - codificación de bits en la segunda estructura de datos para representar una transformación espacial restringida del tercer sistema de coordenadas en relación con el segundo sistema de coordenadas.
3. Procedimiento según cualquiera de las reivindicaciones 1 y 2, que comprende además la etapa de:
- 30 - codificación de los bits en la segunda estructura de datos para representar la posición de un eje y para representar la rotación del primer sistema de coordenadas alrededor del eje y en relación con el segundo sistema de coordenadas.
4. Procedimiento según cualquiera de las reivindicaciones 1 a 3, en el que la etapa de codificación de bits en la tercera estructura de datos comprende además la etapa de:
- 35 - codificación de bits en la segunda estructura de datos para representar la posición del eje por medio de coordenadas enteras en el segundo sistema de coordenadas.
5. Procedimiento según cualquiera de las reivindicaciones 1 a 4, en el que la etapa de codificación de bits en la tercera estructura de datos comprende además la etapa de:
- 40 - codificación de bits en la tercera estructura de datos para representar la orientación del eje, determinando la dirección a lo largo de un eje de un conjunto de ejes mutuamente ortogonales predefinidos.
- 45 6. Procedimiento según cualquiera de las reivindicaciones 2 a 5, que comprende además la etapa de:
- codificación de bits en la primera y la tercera estructura de datos para representar las orientaciones de los elementos en el primer y el tercer sistema de coordenadas, respectivamente.
- 50 7. Procedimiento según cualquiera de las reivindicaciones 1 a 6, que comprende además la etapa de:
- codificación de bits para representar las orientaciones de los elementos determinando la dirección a lo largo de un eje de un conjunto de ejes mutuamente ortogonales predefinidos.
- 55 8. Procedimiento según cualquiera de las reivindicaciones 1 a 7, que comprende además las etapas siguientes:
- codificación de bits en una cuarta estructura de datos para representar una transformación espacial del primer y el segundo sistema de coordenadas en relación con un cuarto sistema de coordenadas, e
  - 60 - inclusión de la cuarta estructura de datos en el modelo legible por ordenador del objeto geométrico.
9. Procedimiento según cualquiera de las reivindicaciones 1 a 8, en el que cada una de las coordenadas enteras es capaz de determinar un volumen tridimensional y viceversa, y en el que el tamaño del volumen tridimensional está adaptado para permitir una representación de la posición de dos elementos por medio de las coordenadas enteras, de tal forma que los elementos son contiguos para crear un objeto geométrico coherente.
- 65 10. Procedimiento según cualquiera de las reivindicaciones 1 a 9, en el que los elementos comprenden representaciones de formas geométricas que representan elementos de construcción de juguete muy conocidos.

## ES 2 226 772 T3

11. Procedimiento según cualquiera de las reivindicaciones 1 a 10, en el que los elementos comprenden representaciones de formas geométricas, de las cuales por lo menos una parte está adaptada para representar medios de conexión o visualizaciones de posibles interconexiones de elementos.

5 12. Procedimiento según cualquiera de las reivindicaciones 1 a 11, en el que las representaciones de las formas geométricas de un grupo de elementos comprende las coordenadas determinadas por números de tipo real de un sistema de coordenadas.

10 13. Procedimiento según cualquiera de las reivindicaciones 1 a 12, que comprende además las etapas siguientes:

- codificación de bits en la tercera estructura de datos para representar las coordenadas enteras en relación con el origen del segundo sistema de coordenadas; definiendo el origen unas coordenadas de interconexión con los elementos del primer sistema de coordenadas;

15 - definición de un nuevo origen en el tercer sistema de coordenadas en respuesta a unas nuevas coordenadas de interconexión del segundo sistema de coordenadas;

- cálculo de las nuevas posiciones de los elementos dentro del tercer sistema de coordenadas como resultado del nuevo origen; y

20 - codificación de los bits en la tercera estructura de datos para representar el resultado del cálculo.

14. Procedimiento según cualquiera de las reivindicaciones 3 a 13, que comprende además las etapas siguientes:

25 - codificación de bits en la primera estructura de datos para representar la posición del eje, definiendo de ese modo las coordenadas de interconexión con elementos del segundo sistema de coordenadas en el primer sistema de coordenadas;

30 - definición de nuevas coordenadas de interconexión en el primer sistema de coordenadas para representar una nueva posición de los elementos dentro del primer sistema de coordenadas en relación con el segundo sistema de coordenadas; y

- codificación de los bits en la primera estructura de datos para representar la nueva posición del eje, siendo la nueva posición las nuevas coordenadas de interconexión.

35 15. Procedimiento según cualquiera de las reivindicaciones 1 a 14, que comprende además las etapas siguientes:

40 - ejecución del procedimiento en un ordenador que tiene una interfaz de usuario que incluye controles de usuario para proporcionar señales que representan una interacción entre el usuario y la interfaz de usuario, comprendiendo la interacción la modificación o la manipulación de una representación geométrica del modelo legible por ordenador; y

- conversión de las señales en un modelo legible por ordenador o un modelo legible por ordenador modificado.

45 16. Procedimiento según cualquiera de las reivindicaciones 1 a 15, que comprende además la etapa de:

- transmisión de los bits codificados o el modelo legible por ordenador por medio de una red de comunicación conectada al ordenador.

50 17. Procedimiento según la reivindicación 16, que comprende además la etapa de:

- recepción de los bits codificados o el modelo legible por ordenador en un ordenador, por medio de una red de comunicación conectada al ordenador.

55 18. Procedimiento según cualquiera de las reivindicaciones 1 a 17, que comprende además las etapas siguientes:

- conexión a una base de datos local y una base de datos remota para la transferencia de datos;

60 - en primer lugar, para cada elemento que consulta la base de datos local, comprobación de la disponibilidad local de la representación de la forma geométrica; y

65 - en segundo lugar, si la representación de la forma geométrica está disponible a nivel local, recuperación de la forma geométrica de la base de datos local, o por el contrario, si la forma geométrica no está disponible a nivel local, recuperación de la forma geométrica de la base de datos remota si la forma geométrica se halla en la base de datos remota.

19. Procedimiento según cualquiera de las reivindicaciones 1 a 18, que comprende además las etapas siguientes:

## ES 2 226 772 T3

- generación de datos presentables en pantalla que representan las formas geométricas, cuyas interrelaciones espaciales se describen según la primera, la segunda y la tercera estructura de datos.

20. Medios legibles por ordenador en los que se ha grabado un programa que, cuando se ejecuta, determina que el ordenador ejecute el procedimiento según cualquiera de las reivindicaciones 1 a 19.

21. Sistema informático que dispone de medios para ejecutar el procedimiento según cualquiera de las reivindicaciones 1 a 19.

22. Modelo legible por ordenador de un objeto geométrico que dispone de estructuras de datos para utilizar el procedimiento según cualquiera de las reivindicaciones 1 a 19.

23. Procedimiento para interpretar un modelo legible por ordenador de un objeto geométrico, en el que el modelo es interpretado en un ordenador que tiene una interfaz de usuario y el ordenador tiene acceso a una base de datos que incluye representaciones de formas geométricas para un grupo de elementos, comprendiendo el procedimiento las etapas siguientes:

- decodificación de bits en una primera estructura de datos para determinar un primer conjunto de elementos a partir del grupo y determinar las posiciones de los elementos por medio de las coordenadas enteras de un primer sistema de coordenadas;
- decodificación de bits en una segunda estructura de datos para determinar una transformación espacial del segundo sistema de coordenadas en relación con el primer sistema de coordenadas; y
- recuperación de las representaciones de formas geométricas determinadas y generación de datos presentables en pantalla que representan las formas geométricas, cuya interrelación espacial se describe según la primera y la segunda estructura de datos.

24. Procedimiento según la reivindicación 23, que comprende además las etapas siguientes:

- decodificación de bits en una tercera estructura de datos para determinar un segundo conjunto de elementos a partir del grupo y determinar las posiciones de los elementos por medio de coordenadas enteras en un tercer sistema de coordenadas, e inclusión de dichos bits en el modelo legible por ordenador; y
- decodificación de bits en la segunda estructura de datos para determinar una transformación espacial limitada del tercer sistema de coordenadas en relación con el segundo sistema de coordenadas.

25. Procedimiento según cualquiera de las reivindicaciones 23 ó 24, que comprende además la etapa de:

- decodificación de bits en la segunda estructura de datos para determinar la posición de un eje y determinar una rotación del primer sistema de coordenadas alrededor del eje y en relación con el segundo sistema de coordenadas.

26. Procedimiento según cualquiera de las reivindicaciones 23 y 24, que comprende además la etapa de:

- decodificación de bits en la segunda estructura de datos para determinar la posición del eje por medio de coordenadas enteras en el segundo sistema de coordenadas.

27. Procedimiento según cualquiera de las reivindicaciones 23 a 26, que comprende además la etapa de:

- decodificación de bits en la tercera estructura de datos para determinar la orientación del eje, determinando la dirección a lo largo de un eje de un conjunto de ejes mutuamente ortogonales predefinidos.

28. Procedimiento según cualquiera de las reivindicaciones 24 a 27, que comprende además la etapa de:

- decodificación de bits en la primera y tercera estructura de datos para determinar las orientaciones de los elementos en el primer y el tercer sistema de coordenadas, respectivamente.

29. Procedimiento según la reivindicación 28, que comprende además la etapa de:

- decodificación de bits para determinar las orientaciones de los elementos leyendo la identificación de una dirección a lo largo de un eje de un conjunto de ejes mutuamente ortogonales predefinidos.

30. Procedimiento según cualquiera de las reivindicaciones 23 a 29, que comprende además las etapas siguientes:

- decodificación de bits en una cuarta estructura de datos para determinar una transformación espacial del primer y el segundo sistema de coordenadas en relación con un cuarto sistema de coordenadas; y

## ES 2 226 772 T3

- modificación de los datos presentables en pantalla para que aparezcan según la transformación espacial.

31. Procedimiento según cualquiera de las reivindicaciones 23 a 30, en el que cada una de las coordenadas enteras es capaz de determinar un volumen tridimensional y viceversa, y en el que el tamaño del volumen tridimensional está adaptado para permitir una posición presentable en pantalla de dos elementos por medio de las coordenadas enteras, de tal forma que los dos elementos están contiguos para crear un objeto geométrico coherente.

32. Procedimiento según cualquiera de las reivindicaciones 23 a 31, en el que los elementos comprenden representaciones de formas geométricas que representan reproducciones de elementos de construcción de juguete muy conocidos.

33. Procedimiento según cualquiera de las reivindicaciones 23 a 32, en el que los elementos comprenden representaciones de formas geométricas, una parte de las cuales está adaptada para representar medios de conexión o visualizaciones de posibles interconexiones de elementos.

34. Procedimiento según cualquiera de las reivindicaciones 23 a 33, en el que las representaciones de las formas geométricas de un grupo de elementos comprenden las coordenadas determinadas por números de tipo real de un sistema de coordenadas.

35. Procedimiento según cualquiera de las reivindicaciones 23 a 32, que comprende además las etapas siguientes:

- decodificación de bits en una tercera estructura de datos para representar las coordenadas enteras en relación con el origen del segundo sistema de coordenadas; definiendo el origen las coordenadas de interconexión con los elementos del primer sistema de coordenadas;
- definición de un nuevo origen en el tercer sistema de coordenadas en respuesta a unas nuevas coordenadas de interconexión del segundo sistema de coordenadas;
- cálculo de las nuevas posiciones de los elementos dentro del segundo sistema de coordenadas como resultado del nuevo origen; y
- codificación de los bits en la tercera estructura de datos para representar el resultado del cálculo.

36. Procedimiento según cualquiera de las reivindicaciones 24 a 33, que comprende además las etapas siguientes:

- decodificación de los bits en la primera estructura de datos para representar la posición del eje, definiendo de ese modo las coordenadas de interconexión en el primer sistema de coordenadas; y
- definición de nuevas coordenadas de interconexión en el primer sistema de coordenadas para representar una nueva posición de los elementos dentro del primer sistema de coordenadas en relación con el segundo sistema de coordenadas;
- codificación de los bits en la primera estructura de datos para representar la nueva posición del eje, constituyendo dicha nueva posición las nuevas coordenadas de interconexión.

37. Procedimiento según cualquiera de las reivindicaciones 27 a 34, que comprende además las etapas siguientes:

- ejecución del procedimiento en un ordenador que tiene una interfaz de usuario que incluye controles de usuario para proporcionar señales que representan una interacción entre un usuario y la interfaz de usuario, comprendiendo la interacción la modificación o la manipulación de una representación geométrica del modelo legible por ordenador; y
- conversión de las señales en un modelo legible por ordenador o un modelo legible por ordenador modificado.

38. Procedimiento según cualquiera de las reivindicaciones 23 a 37, que comprende además la etapa de:

- conversión del modelo legible por ordenador en datos presentables en pantalla, calculando una proyección de las formas geométricas, tal como aparecen según las estructuras de datos, en una superficie plana o curva.

39. Procedimiento según cualquiera de las reivindicaciones 23 a 38, que comprende además la etapa de:

- lectura del modelo legible por ordenador desde un archivo de tipo binario que representa el modelo legible por ordenador.

40. Procedimiento según cualquiera de las reivindicaciones 23 a 37, que comprende además la etapa de:

## ES 2 226 772 T3

- transmisión de los bits o el modelo legible por ordenador por medio de una red de comunicación conectada al ordenador.

5 41. Procedimiento según la reivindicación 40, que comprende además la etapa de:

- recepción de los bits o el modelo legible por ordenador en un ordenador, por medio de una red de comunicación conectada al ordenador.

10 42. Procedimiento según cualquiera de las reivindicaciones 23 a 41, que comprende además las etapas siguientes:

- separación de la base de datos en una base de datos local y una base de datos remota;
- en primer lugar, para cada elemento que consulta la base de datos, comprobación de la disponibilidad local de la representación de la forma geométrica; y
- en segundo lugar, si la forma geométrica está disponible a nivel local, recuperación de la forma geométrica de la base de datos local, o por el contrario, si la forma geométrica no está disponible a nivel local, recuperación de la forma geométrica de la base de datos remota, si la forma se halla en la base de datos remota.

20 43. Medios legibles por ordenador en los que se ha grabado un programa que, cuando se ejecuta, determina que el ordenador ejecute el procedimiento según cualquiera de las reivindicaciones 23 a 42.

25 44. Sistema informático que dispone de medios para ejecutar el procedimiento según cualquiera de las reivindicaciones 23 a 42.

30 45. Modelo legible por ordenador de un objeto geométrico que dispone de estructuras de datos para la utilización del procedimiento según cualquiera de las reivindicaciones 23 a 42.

35

40

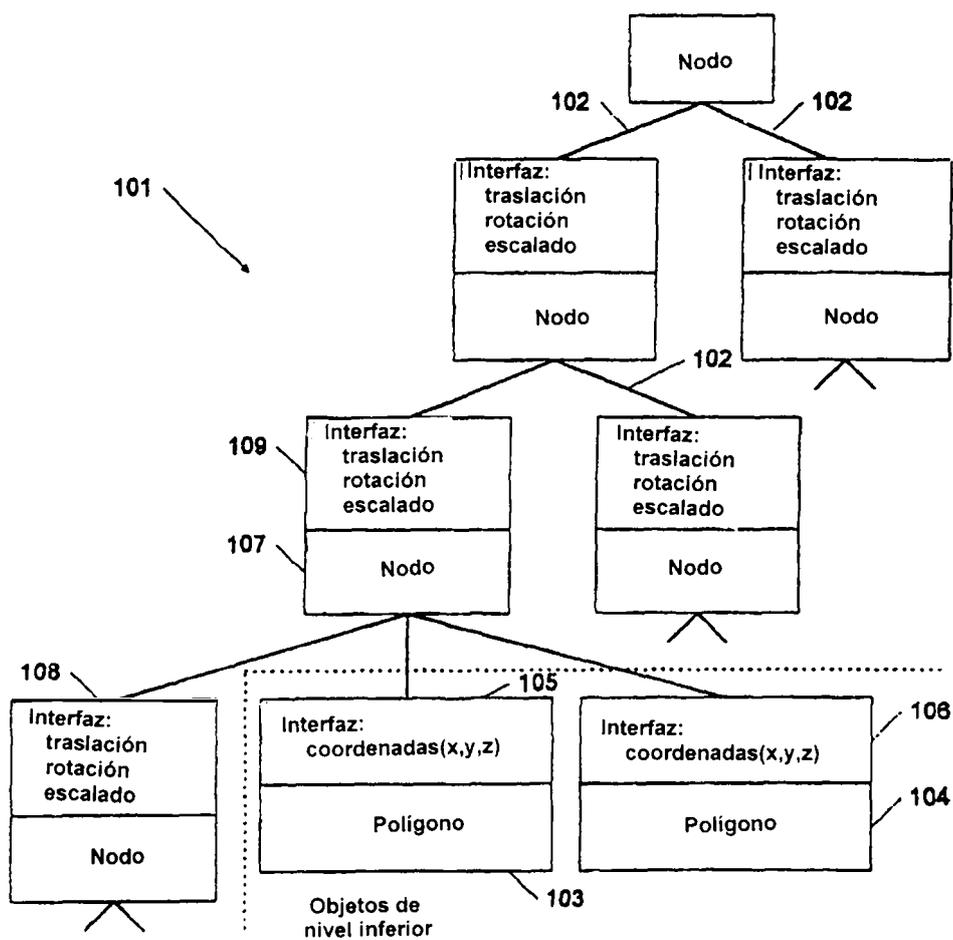
45

50

55

60

65



-Técnica anterior-

Fig. 1

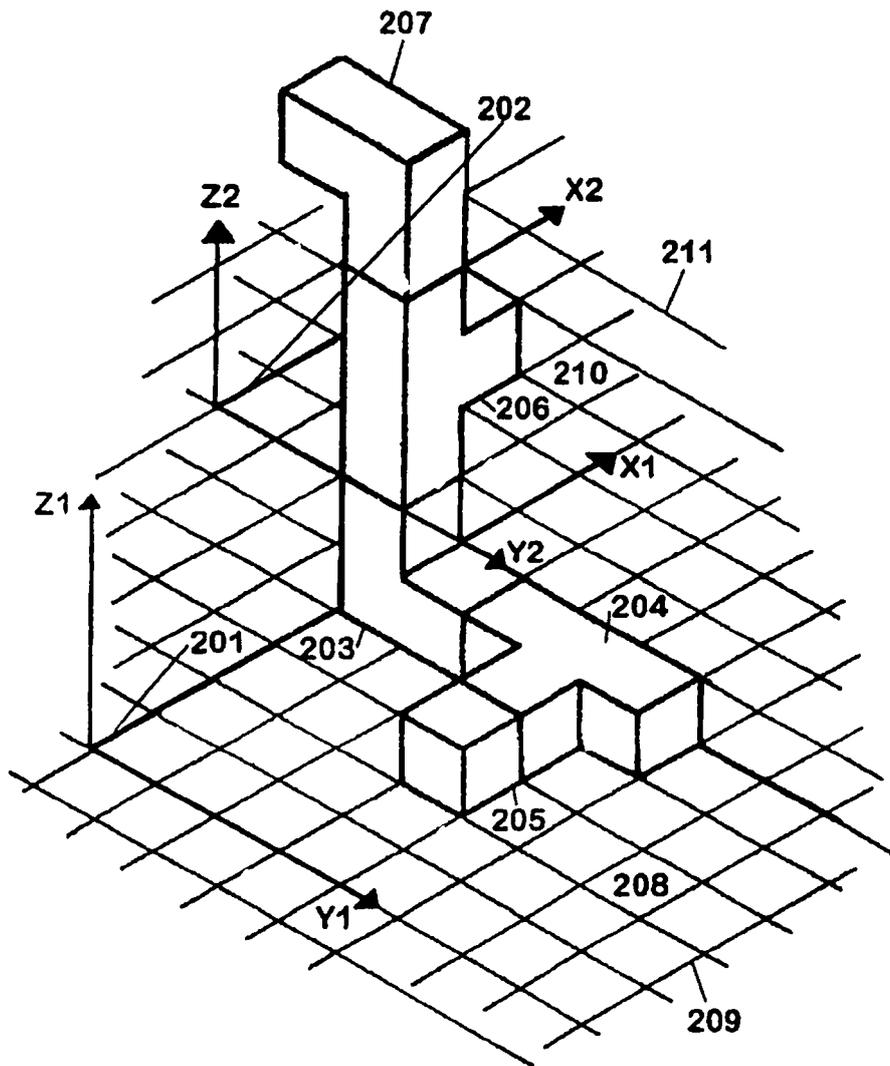


Fig. 2a

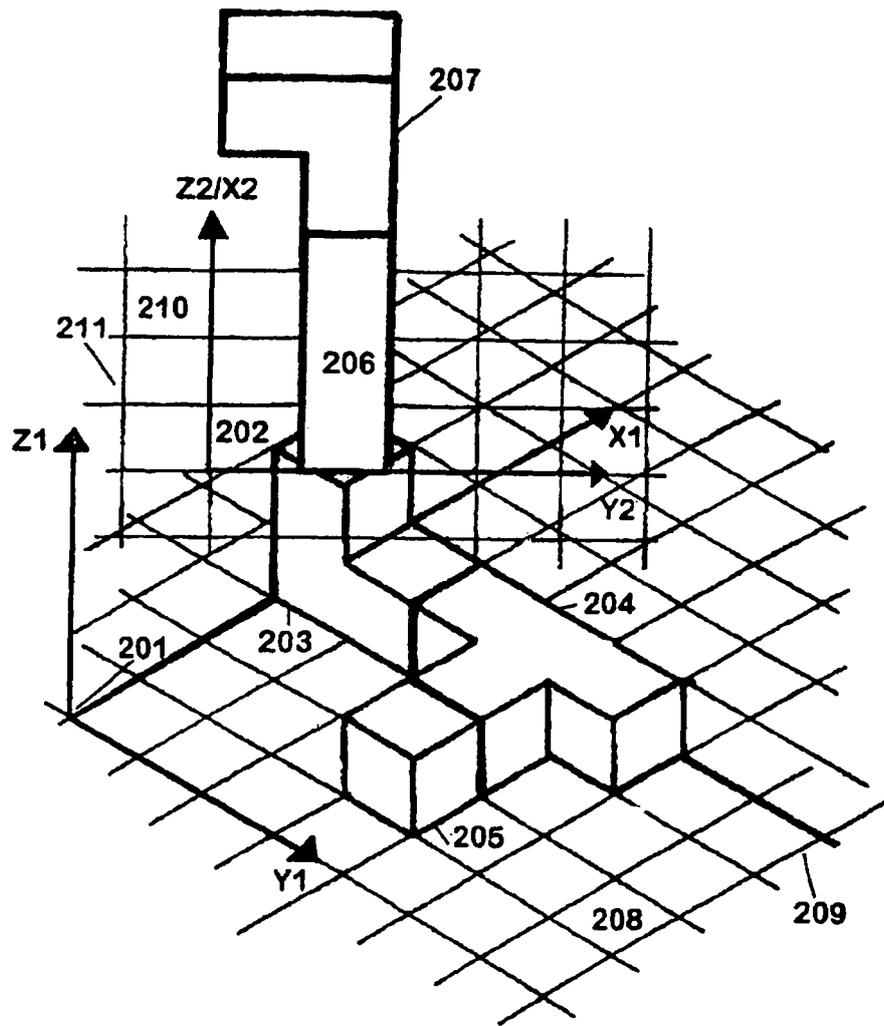


Fig. 2b

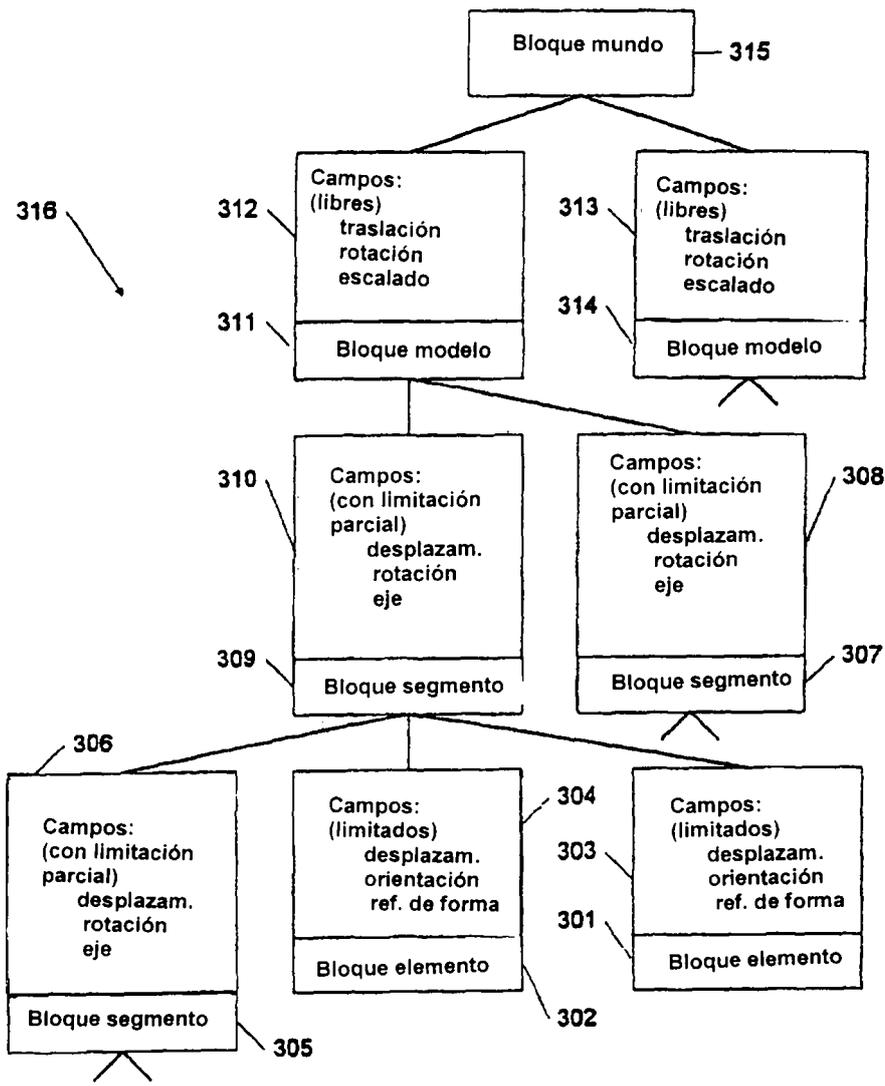


Fig. 3

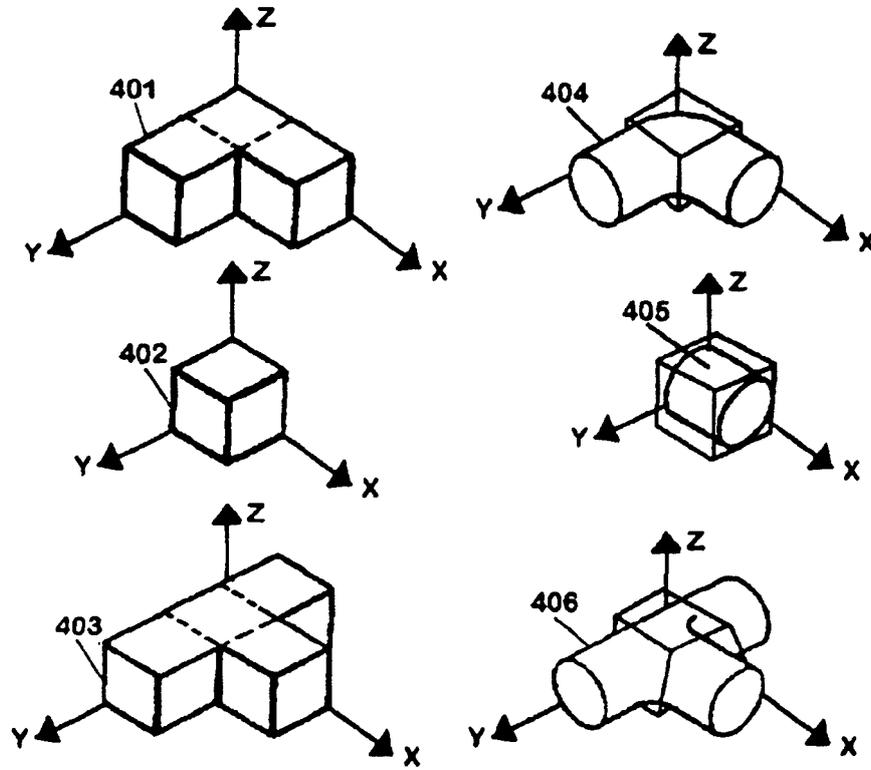


Fig. 4

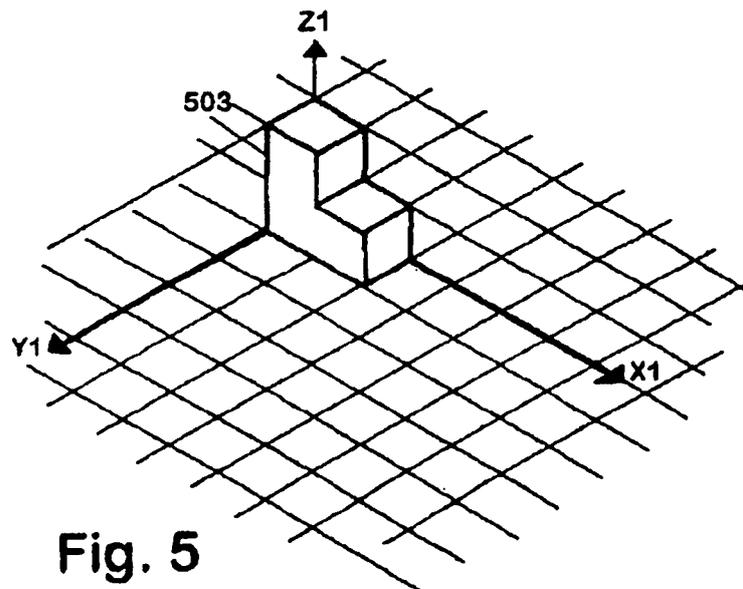


Fig. 5

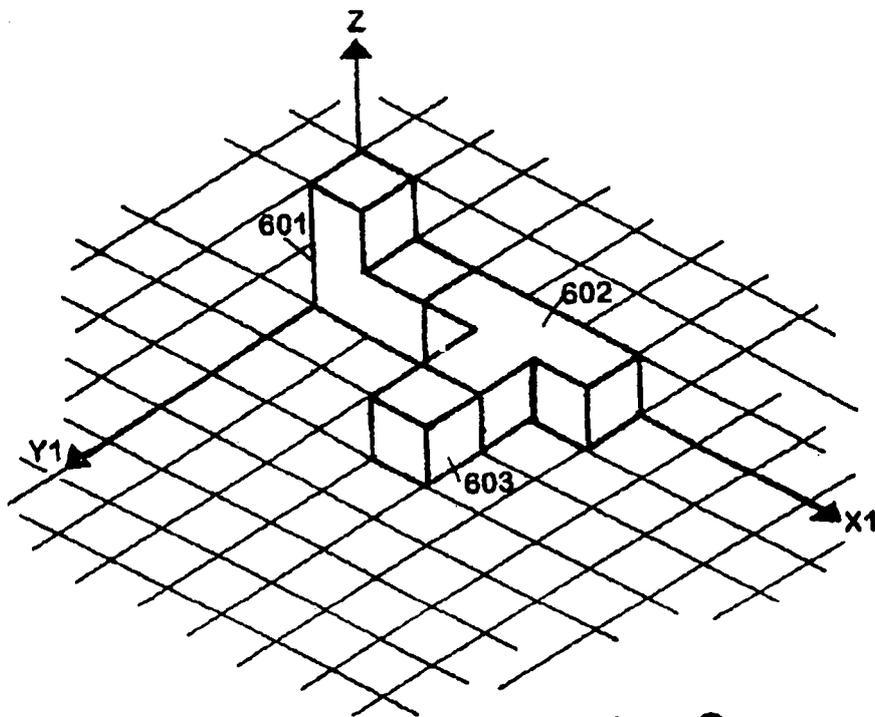


Fig. 6

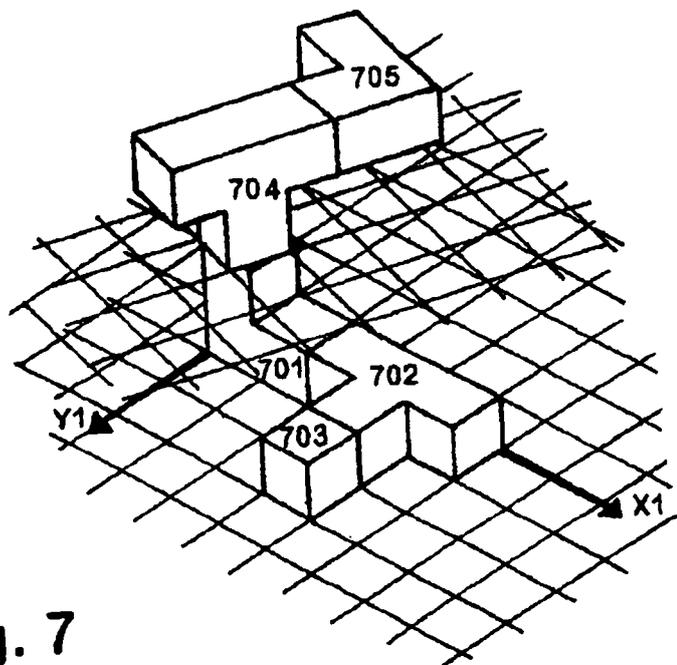
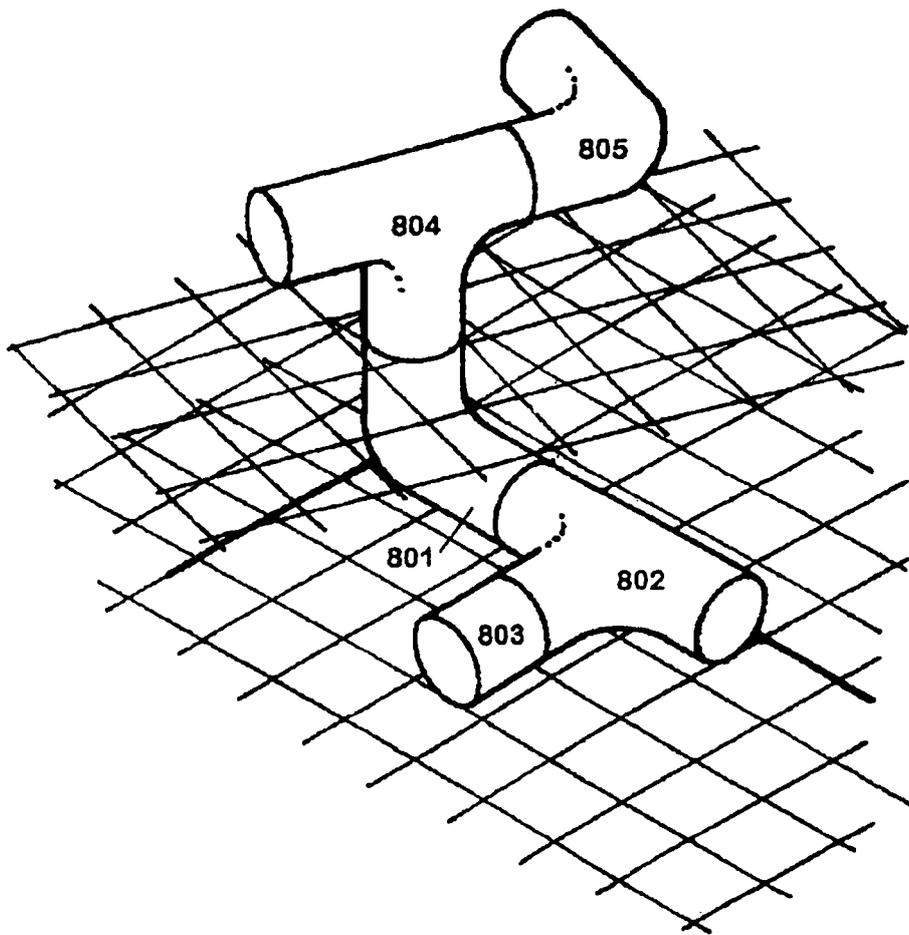


Fig. 7



**Fig. 8**

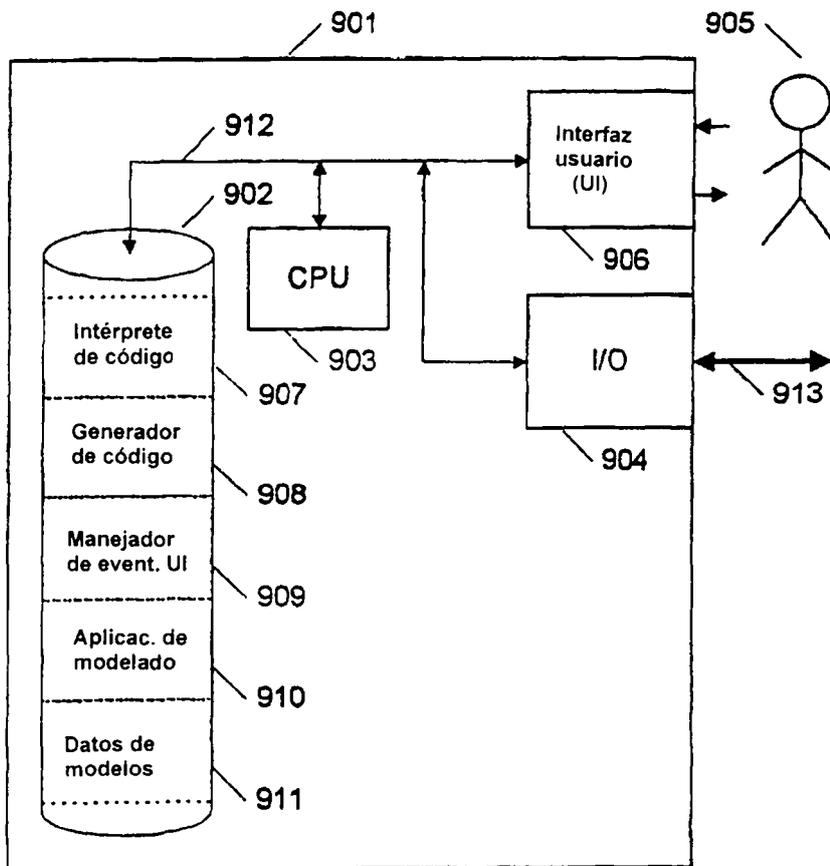


Fig. 9

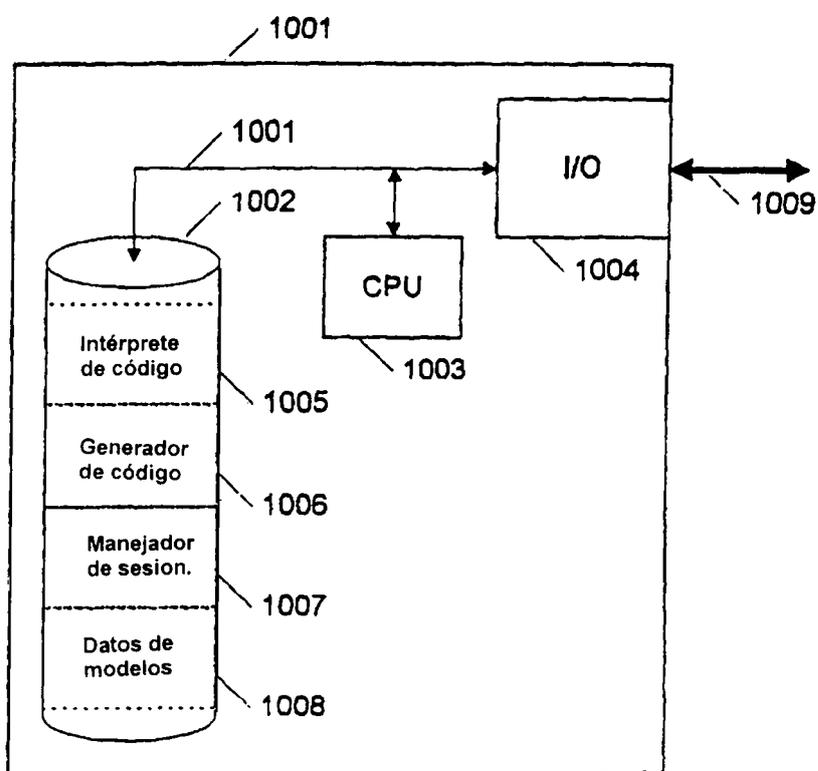


Fig. 10

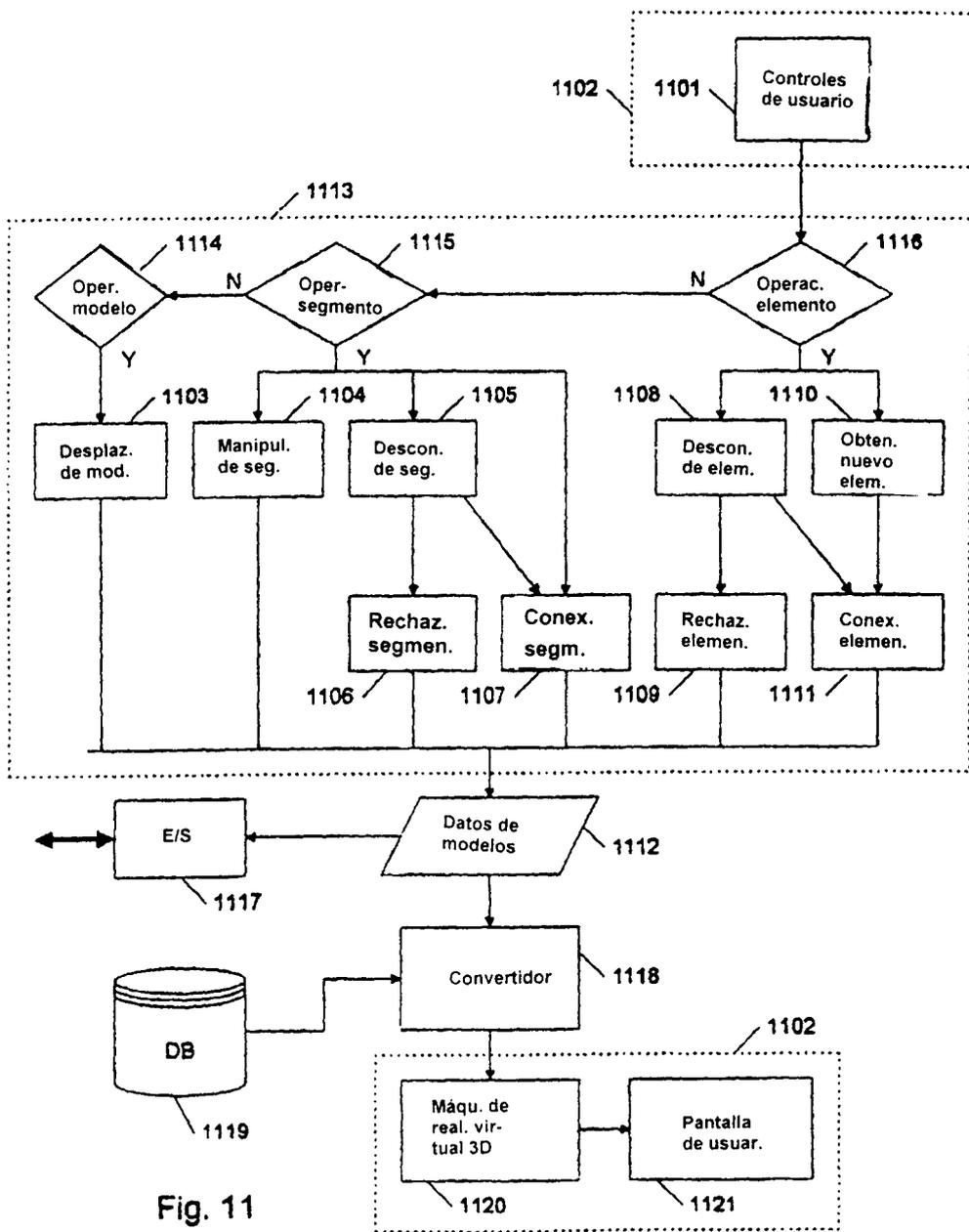


Fig. 11

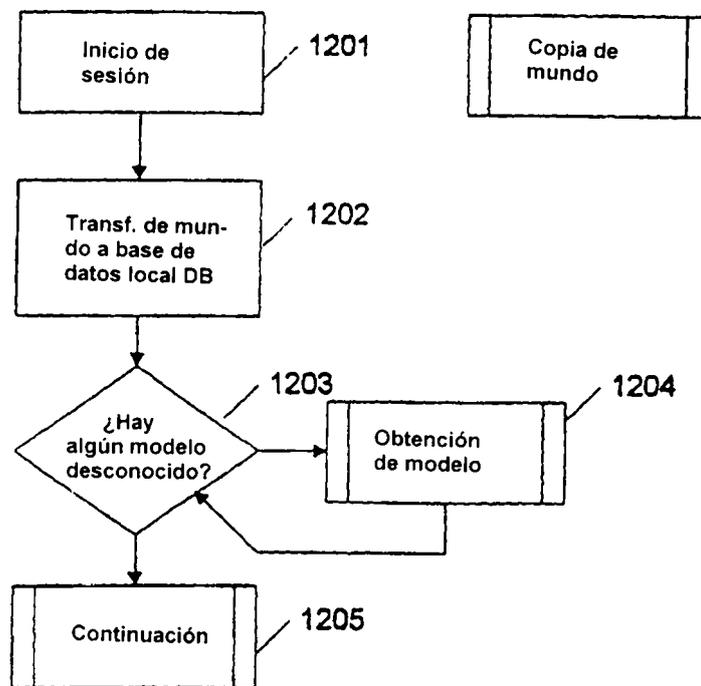


Fig. 12

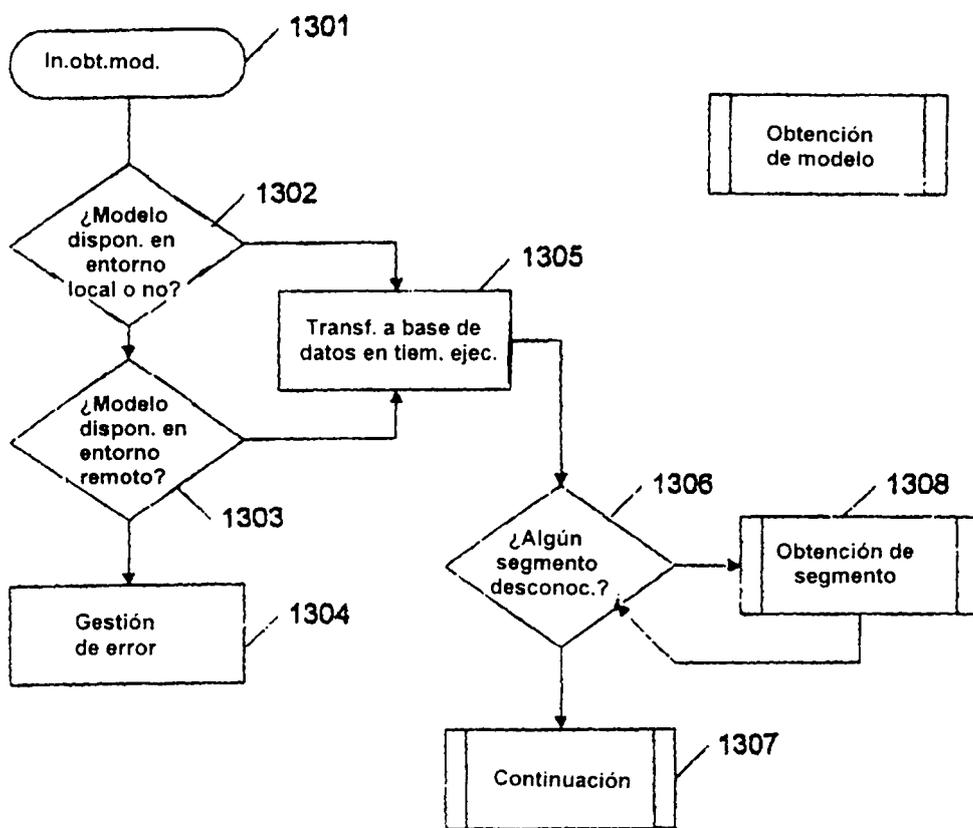


Fig. 13