



19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA

11 Número de publicación: **2 266 396**

51 Int. Cl.:

**H04N 7/24** (2006.01)

**H04N 7/26** (2006.01)

**H04N 7/50** (2006.01)

12

## TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Número de solicitud europea: **02075022 .0**

86 Fecha de presentación : **13.02.1998**

87 Número de publicación de la solicitud: **1209915**

87 Fecha de publicación de la solicitud: **29.05.2002**

54 Título: **Terminal audio - visual basado en objetos y estructura de flujo de bits correspondiente.**

30 Prioridad: **14.02.1997 US 37779**

45 Fecha de publicación de la mención BOPI:  
**01.03.2007**

45 Fecha de la publicación del folleto de la patente:  
**01.03.2007**

73 Titular/es: **The Trustees of Columbia University in  
the City of New York  
Broadway and West 116th Street  
New York, New York 10027-6699, US**

72 Inventor/es: **Eleftheriadis, A. y  
Kalva, Hari**

74 Agente: **Arias Sanz, Juan**

**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**

Terminal audio-visual basado en objetos y estructura de flujo de bits correspondiente.

5 Esta invención se refiere a la representación, transmisión, procesamiento y visualización de información de vídeo y audiovisual, más concretamente, de información basada en objetos.

**Antecedentes de la invención**

10 Se han desarrollado técnicas de compresión de imagen y vídeo que, a diferencia de la codificación de forma de onda tradicional, intentan capturar estructuras de contenido visual de alto nivel. Este tipo de estructura se describe en términos de “objetos” constitutivos que tienen relevancia visual inmediata, representando objetos físicos familiares, por ejemplo, una pelota, una mesa, una persona, una melodía o una frase hablada. Los objetos se codifican de forma independiente empleando una técnica de compresión que proporciona la mejor calidad para cada objeto. Los objetos  
15 comprimidos se envían a un terminal junto con la información de composición que le dice al terminal dónde colocar los objetos en una escena. El terminal decodifica los objetos y los coloca en la escena tal como se especifica en la información de composición. Además de producir ganancias de codificación, las representaciones basadas en objetos son beneficiosas en relación con la modularidad, la reutilización del contenido, la facilidad de manipulación, la facilidad de interacción con componentes individuales de imagen y la integración del contenido natural capturado con una  
20 cámara con contenido sintético generado por ordenador.

En el documento “A Proposed Architecture for an Object-Based Audio-Visual Bitstream and Terminal” (MPEG97/M1619) presentado en la reunión de Sevilla del grupo de trabajo ISO/IEC JTC1/SC29/WG11 y en la publicación de patente EP-A-0 858 228 que es estado de la técnica en el sentido del artículo 54(3) del CPE, puede hallarse información  
25 técnica relacionada con la presente invención.

**Sumario de la invención**

La presente invención se refiere a un método y a un aparato para visualizar datos audiovisuales basados en objetos  
30 de acuerdo con las reivindicaciones adjuntas.

En una arquitectura, estructura o formato preferidos para información que va a procesarse en un terminal de vídeo o audiovisual (AV) basado en objetos, un flujo de bits orientado objeto incluye objetos, información de composición e información de demarcación de la escena. La estructura del flujo de bits permite la edición en línea, por ejemplo,  
35 copiar y pegar, la inserción/eliminación, el agrupamiento y efectos especiales.

En la arquitectura preferida, en aras de una mayor simplicidad de la edición, los objetos AV y su información de composición se transmiten o se accede a ellos en canales lógicos separados (LC, Logic Channel). La arquitectura también hace uso de la “persistencia de objetos”, sacando ventaja de algunos objetos que tienen una vida útil en el  
40 decodificador que va más allá de su tiempo de presentación inicial, hasta un tiempo de expiración seleccionado.

**Breve descripción del dibujo**

La figura 1 es un dibujo esquemático funcional de un terminal audiovisual basado en objetos, a título de ejemplo.  
45

La figura 2a es un dibujo esquemático de un paquete de composición audiovisual basado en objetos, a título de ejemplo.

La figura 2b es un dibujo esquemático de un paquete de datos audiovisual basado en objetos, a título de ejemplo.  
50

La figura 2C es un dibujo esquemático de un paquete de composición compuesto, a título de ejemplo.

La figura 3 es un dibujo esquemático de información de descripción de escena y nodo, a título de ejemplo, que utiliza la composición.  
55

La figura 4 es un dibujo esquemático, a título de ejemplo, de información de asociación de nodo-flujo.

La figura 5 es un dibujo esquemático de información a modo de ejemplo de actualización de nodo/gráfico que emplea una escena.  
60

La figura 6 es un dibujo esquemático de un diseño de terminal audiovisual, a título de ejemplo.

La figura 7 es un dibujo esquemático de un controlador del sistema audiovisual, a título de ejemplo, en el terminal según la figura 6.  
65

La figura 8 es un dibujo esquemático de flujo de información a modo de ejemplo en el controlador según la figura 7.

## Descripción detallada

Un terminal audiovisual (AV) es un componente de sistemas que desempeña un papel decisivo en la formación, presentación o visualización de contenido audiovisual. Esto incluye (aunque no de forma restrictiva) terminales de usuario final con una pantalla de monitor y altavoces, así como prestaciones de procesador central y ordenador servidor en las que se procesa la información audiovisual. En un terminal AV, la funcionalidad deseada puede implementarse en hardware, firmware o software. La información que va a procesarse puede proporcionarse al terminal desde una fuente de información remota a través de un canal de telecomunicaciones, o puede recuperarse desde un archivo local, por ejemplo. Más concretamente, un terminal audiovisual orientado objeto recibe información en forma de objetos individuales que van a combinarse en escenas según información de composición proporcionada al terminal.

La figura 1 ilustra un terminal de este tipo que incluye un demultiplexor (DMUX) 1 conectado a través de un canal lógico LC0 a un controlador del sistema o "ejecutivo" 2 a través de canales lógicos LC1 a LCn a una memoria intermedia 3. El ejecutivo 2 y la memoria intermedia 3 están conectados a decodificadores 4 que están conectados a su vez a una unidad de composición 5. Asimismo, el ejecutivo 2 está conectado a la unidad de composición 5 directamente y tiene una entrada externa para la interacción del usuario, por ejemplo.

En la arquitectura AV preferida, los objetos AV y su información de composición se transmiten o se accede a éstos en canales lógicos separados. El demultiplexor DMUX recibe la capa Mux2 de las capas inferiores y la demultiplexa en canales lógicos. LC0 porta información de composición que se transmite al ejecutivo. Los objetos AV recibidos en otros canales lógicos se almacenan en la memoria intermedia para que los decodificadores actúen sobre ellos. El ejecutivo recibe la información de composición, que incluye los sellos temporales de decodificación y presentación, e instruye de forma correspondiente al decodificador y al elemento de composición.

El sistema trata paquetes de composición de objetos (OCP, Object Composition Packet) y paquetes de datos de objetos (ODP, Object Data Packet). Un paquete de composición contiene un identificador de objeto ("ID"), sellos temporales y los "parámetros de composición" para representar el objeto. Un paquete de datos de objeto contiene un identificador de objeto, un sello temporal de caducidad en caso de objetos de persistencia y datos de objeto.

Preferiblemente, una entrada externa, tal como una interacción del usuario, se convierte en paquetes OCP y/u ODP antes de presentarse al ejecutivo. No se requieren cabeceras en un flujo de datos proporcionado a través de una red. Sin embargo, se requieren cabeceras al almacenar una presentación MPEG4 en un archivo.

Las figuras 2a y 2b ilustran la estructura de paquetes de composición y datos de forma más detallada. Las siguientes son características relevantes:

Identificador de objeto (Objeto ID) está compuesto de tipo de objeto y número de objeto. La longitud por defecto del identificador del objeto es de 2 bytes, que incluye diez bits para el número del objeto y 6, para el tipo de objeto (por ejemplo, texto, gráficos, MPEG2 VOP, objeto compuesto). Se utiliza un código ampliable para alojar más de 1023 objetos o más de 31 tipos de objeto. La siguiente convención se añadirá a: un valor de 0b111111 en los primeros seis bits del identificador del objeto corresponde a 31 más el valor del byte inmediatamente siguiente al identificador del objeto; un valor de 0b11.1111.1111 en los 10 bits menos significativos del identificador del objeto corresponde a 1023 más el valor de los dos bytes inmediatamente siguientes al identificador del objeto (sin contar los bytes de extensión del tipo de objeto, si están presentes). Se definen los siguientes tipos de objeto:

### Objetos de composición (identificadores de objeto de 16 bits)

|    |        |                                     |
|----|--------|-------------------------------------|
|    | 0X0000 | objeto de configuración de escena   |
| 50 | 0X0001 | especificación de jerarquía de nodo |
|    | 0X0002 | asociación nodo-flujo               |
|    | 0X0003 | actualización nodo/escena           |
| 55 | 0X0004 | objeto compuesto                    |

### Datos de objeto (tipo de objeto, 6 bits más significativos)

|    |           |                             |
|----|-----------|-----------------------------|
| 60 | 0b00.0010 | texto                       |
|    | 0b00.0011 | MPEG2 VOP (VOP rectangular) |

Objetos de persistencia (PO, Persistencia Object) son objetos que deberían guardarse en el decodificador para su uso posterior. Un sello temporal de caducidad (ETS, Expiration Time Stamp) indica la vida de un objeto PO en milisegundos. Un objeto PO no está disponible para el decodificador una vez transcurrido el sello ETS. Cuando va a utilizarse un objeto PO en un momento posterior en una escena, sólo necesita enviarse al terminal AV la correspondiente información de composición.

## ES 2 266 396 T3

Sello temporal de decodificación (DTS, Decoding Time Stamp) indica el tiempo en el que el decodificador debería decodificar un objeto (unidad de acceso).

5 Sello temporal de presentación (PTS, Presentation Time Stamp) indica el tiempo en el que el decodificador debería presentar un objeto (unidad de acceso).

10 Sello temporal de vida útil (LTS, Lifetime Time Stamp) indica la duración (en milisegundos) que un objeto debería visualizarse en una escena. El sello LTS está implícito en algunos casos, por ejemplo, en una secuencia de vídeo en la que un marco se visualiza para una velocidad de 1/marco o hasta que el siguiente marco esté disponible, lo que sea más largo. Un sello LTS explícito se utiliza al visualizar gráficos y texto. Un objeto AV debería decodificarse sólo una vez para utilizarlo durante su vida útil.

15 Sello temporal de caducidad (ETS, Expiration Time Stamp) es específico para soportar la noción de persistencia de objeto. Un objeto, tras ser presentado, se guarda en el decodificador (caché) hasta un momento dado por el sello ETS. Un objeto de este tipo puede utilizarse múltiples veces antes de que expire el sello ETS. Un objeto PO con un sello ETS caducado ya no está disponible para el decodificador.

20 Base temporal de objeto (OTB, Object Time Base) define la noción de tiempo de un codificador de objeto AV dado. Diferentes objetos pueden pertenecer a diferentes bases temporales. El terminal AV adapta estas bases temporales a la base local, tal como se especifica en el modelo de verificación MSDL.

25 Referencia de reloj de objeto (OCR, Object Clock Reference) puede utilizarse si es necesario transmitir la velocidad de la base OTB al decodificador. Mediante este mecanismo, las bases OTB pueden recuperarse/adaptarse en el terminal AV.

Parámetros de composición se utilizan para componer una escena (colocar un objeto en una escena). Éstos incluyen el desplazamiento de la esquina superior izquierda al marco de presentación, ángulos de rotación, factores de zoom, etc.

30 Prioridad indica la prioridad de un objeto para la transmisión, decodificación y visualización. MPEG4 admite 32 niveles de prioridad. Los números inferiores indican prioridades más altas.

Indicador de persistencia (PI, Persistence Indicator) indica si un objeto es persistente.

35 Indicador de continuación (CI, Continuation Indicator) indica el final de un objeto en el paquete actual (o continuación).

40 Agrupamiento de objetos facilita operaciones que van a aplicarse a un conjunto de objetos con una única operación. Una característica de este tipo puede utilizarse para minimizar la cantidad de información de composición enviada, así como para soportar la composición jerárquica de escenas basándose en escenas secundarias independientes. El compositor maneja los objetos componentes como un grupo. En la figura 2c se muestra la estructura de un paquete de composición compuesto (CCP, Compound Composition Packet).

45 Estructura del flujo de bits incluye paquetes de composición de objetos para describir la composición y comprobar la presentación de esos paquetes, y paquetes de datos de objeto que contienen los datos para los objetos. Una escena está compuesta por un conjunto de paquetes de composición. El flujo de bits admite la representación de escenas como una jerarquía mediante el uso de objetos de composición compuestos (CCP), también conocida como jerarquía de nodo. Un CCP permite combinar objetos de composición para crear escenas audiovisuales complejas.

50 Paquete de datos de objeto:

Identificador de objeto - mínimo 10 bits (por defecto)

55 CI y PI podrían combinarse:

00 - comienzo no persistente

01 - comienzo persistente

60 10 - continuación

11 - fin de objeto

65 Prioridad: 5 bits, sólo presentes si CI/PI es 0b00 ó 0b01.

ETS: 30 bits, presentes si CI/PI es 0b01.

## ES 2 266 396 T3

Para la codificación de vídeo basada en la predicción, se indica tipo\_VOP mediante dos bits (00 (I), 01 (P), 10 (B), 11 (PB)), facilitando la edición.

```
5  Paquete_datos_objeto{
    ID-objeto      16 bits + cualquier extensión;
    CIPI           2 bits
10  If (CIPI <=1){
    Prioridad      5 bits
    If (tipo objeto es MPEG VOP)
15      (Cualquier compresión basada en predicción)
    Tipo_VOP       2 bits
    }
20  If (CIPI == 1)
    ETS            28 bits
25  DatosObjeto
    }
```

### 30 Paquete de composición de objeto

```
Paquete_composición_objeto{
35  IDObjeto      16 bits + cualquier extensión;
    Indicador_OCR      1 bit
    Indicador_temporizadores_visualización  1 bit
40  DTS           30 bits
    if (Indicador_OCR)
        OCR         30 bits
45  If (Indicador_temporizadores_visualización){
        PTS         30 bits
50        LTS        28 bits
    }
    Parámetros_composición;
55 }
```

60

65

## ES 2 266 396 T3

Parámetros de composición se definen en la sección 2 del modelo de verificación MSDL, MPEG N1483, Systems Working, borrador V2.0.

```

5      Parámetros_composición(
      visibilidad                      1 bit
      orden_composición              5 bits
10     número_conjuntos-movimiento  2 bits
      x_delta_0                     12 bits
      y_delta_0                     12 bits
15     para (i = 1; i <= número_conjuntos-movimiento;
      i++) {
20         x_delta_i                 12 bits
         y_delta_i                 12 bits
      }
25 }

```

### Paquete de composición compuesto

```

30 Paquete_composición_compuesto{
      IDObjeto                      16 bits
      PTS                          30 bits
35     LTS                          28 bits
      Parámetros_composición
      CuentaObjeto                  8 bits
40     For (i = 0; i < CuentaObjeto; i++){
          Paquete_composición_objeto;
45     }
}

```

50 Paquete de configuración de escena (SCP, Scene Configuration Packet) se utiliza para cambiar la anchura y altura de referencia de la escena, para vaciar la memoria intermedia y para otras funciones de configuración. El tipo de objeto para paquetes SCP es 0b00.0000. Esto permite 1024 paquetes de configuración diferentes. El número de objeto 0b00.0000.0000 (identificador de objeto 0X0000) se define para el uso al vaciar las memorias intermedias del terminal.

### *Control de composición para gestión de memoria intermedia (identificador de objeto 0x0000)*

55 Las memorias intermedias del terminal AV se descargan empleando indicadores Descargar\_Caché y Actualizar\_escena. Al utilizar la arquitectura jerárquica de escena, el gráfico de escena actual se descarga y el terminal carga la nueva escena desde el flujo de bits. El uso de indicadores permite guardar la estructura de escena actual, en lugar de alinearla. Estos indicadores se utilizan para actualizar la anchura y altura de la escena de referencia cada vez que comienza una nueva escena. Si está ajustado el Indicador\_descargar\_caché, se descarga la memoria caché eliminando los objetos (si hay alguno). Si está ajustado el Indicador\_actualizar\_escena, existen dos posibilidades: (i) está ajustado el indicador\_descargar\_caché que implica que ya no se utilizarán más los objetos de la memoria caché; (ii) no está ajustado el indicador\_descargar\_caché, la nueva escena que está introduciéndose (una acción de edición en el flujo de bits) divide la escena actual y los objetos de la escena se utilizarán después del final de la nueva escena. El sello ETS de los objetos, si está presente, se detendrá durante la duración de la nueva escena introducida. El comienzo de la siguiente escena se indica mediante otro paquete de configuración de escena.

## ES 2 266 396 T3

```

Paquete_configuración_escena{
    IDObjeto                                16 bits (0x0000)
5      Indicador_desacargar_caché          1 bit
      Indicador_actualizar_caché          1 bit
      If (indicador_actualizar_caché){
10          anchura_escena_ref              12 bits
          altura_escena_ref                12 bits
      }
15 }

```

### *Control de composición para descripción de escena (identificador de objeto 0x0001)*

Se define una jerarquía de nodos que describe una escena. Los paquetes de configuración de escena también pueden utilizarse para definir una jerarquía de escena que permite una descripción de escenas como una jerarquía de objetos AV. Cada uno de los nodos en un gráfico de este tipo es un agrupamiento de nodos que agrupa las ramas y/u otros nodos del gráfico en un objeto AV compuesto. Cada nodo (rama) tiene un identificador único seguido de sus parámetros, tal como se muestra en la figura 3.

### *Control de composición para la correlación (mapping) flujo-nodo (identificador de objeto 0x0002)*

Tal como se ilustra en la figura 4, las entradas de la tabla asocian los flujos elementales de objetos en los canales lógicos con los nodos en una escena jerárquica. Los identificadores de los flujos son únicos, pero no así los identificadores de los nodos. Esto implica que más de un flujo puede asociarse con el mismo nodo.

### *Control de composición para actualizaciones de escenas (identificador de objeto 0x0003)*

La figura 5 ilustra la actualización de nodos en la jerarquía de la escena modificando los parámetros específicos del nodo. El propio gráfico puede actualizarse añadiendo/eliminando nodos en el gráfico. El tipo de actualización del paquete indica el tipo de actualización que va a llevarse a cabo en el gráfico.

### *Realización arquitectónica*

La realización descrita a continuación incluye un flujo de bits AV basado en objeto y una arquitectura de terminal. El diseño del flujo de bits especifica, en un formato binario, la forma en que se representan los objetos AV y cómo van a componerse. La estructura del terminal AV especifica la forma de decodificar y visualizar los objetos en el flujo de bits binario.

### *Arquitectura de terminal AV*

Adicionalmente a la figura 1 y haciendo referencia específicamente a la figura 6, la entrada al demultiplexor 1 es un flujo de bits basado en objeto tal como un flujo de bits MPEG-4 que consiste en objetos AV y su información de composición multiplexada en canales lógicos (LC). La composición de objetos en una escena puede especificarse como una colección de objetos con especificación de composición independiente, o como un gráfico de escenas jerárquico. La información de composición y control se incluye en la capa LC0. La información de control incluye instrucciones de control para actualizar gráficos de escena, memorias intermedias de decodificador de reinicio, etc. Los canales lógicos 1 y anteriores contienen la fecha de los objetos. El sistema incluye un controlador (o "ejecutivo") 2 que controla el funcionamiento del terminal AV.

El terminal incluye adicionalmente memorias intermedias 3, decodificadores de objetos AV 4, memorias intermedias 4' para datos decodificados, un elemento de composición 5, una pantalla de visualización 6 y una memoria caché de objetos 7. El flujo de bits de entrada puede leerse desde una conexión de red o desde un dispositivo de almacenamiento local, tal como un DVD, CD-ROM o disco duro informático. La capa LC0 que contiene la información de composición se alimenta al controlador. El demultiplexor DMUX almacena los objetos en la capa LC1 y por encima, en la ubicación de la memoria intermedia especificada por el controlador. En el caso del suministro de la red, el codificador y el servidor del flujo cooperan para garantizar que no se aporta un flujo en exceso o por defecto a las memorias de objetos de entrada. Los objetos de datos codificados se almacenan en las memorias intermedias de datos de entrada hasta que son leídas por los decodificadores en su periodo de decodificación, dado normalmente por un sello temporal de decodificación asociado. Antes de escribir un objeto de datos a la memoria intermedia, el demultiplexor DMUZ elimina los sellos temporales y otros encabezamientos del paquete de datos de objeto y los transmite al controlador para la señalización de los decodificadores y memorias intermedias de entrada adecuados. Los decodificadores, al

ser señalizados por el controlador, decodifican los datos en las memorias intermedias de entrada y los almacenan en las memorias intermedias de salida del decodificador. El terminal AV también trata la entrada externa, tal como, la interacción de un usuario.

En la memoria caché de objetos 7 los objetos se almacenan para el uso posterior a su tiempo de presentación inicial. Estos objetos permanecen en la memoria caché incluso si se elimina el nodo asociado del gráfico de la escena, pero sólo se eliminan al expirar un intervalo de tiempo asociado denominado “sello temporal de expiración”. Esta característica puede utilizarse en presentaciones en las que un objeto se utiliza repetidamente en una sesión. La composición asociada con estos objetos puede actualizarse con mensajes de actualización adecuados. Por ejemplo, puede descargarse el logotipo de la estación de radiodifusión al comienzo de la presentación y puede utilizarse la misma copia para la visualización repetida a lo largo de una sesión. Las actualizaciones de composición subsiguientes pueden cambiar la posición del logotipo en la pantalla de visualización. Los objetos que se usan nuevamente después de su primer tiempo de presentación pueden denominarse objetos persistentes.

#### Controlador del sistema (SC, System Controller)

El controlador del sistema controla la decodificación y reproducción de flujos de bits en el terminal AV. En el inicio, a partir de una interacción de usuario o al buscar una sesión en la dirección de red por defecto, el controlador SC inicializa primero el demultiplexor DMUX para la lectura desde un dispositivo de almacenamiento local o un puerto de red. El sistema lógico de control se carga en la memoria RAM del programa en el momento de la inicialización. El decodificador de instrucciones lee las instrucciones del programa y las ejecuta. La ejecución puede implicar la lectura de los datos desde las memorias intermedias de entrada (datos de composición o datos externos), inicialización de los temporizadores de objetos, carga o actualización de las tablas de objeto en la memoria RAM, carga de temporizadores de objetos o señalización de control.

La figura 7 muestra el controlador del sistema con mayor detalle. El demultiplexor DMUX lee el flujo de datos de entrada y proporciona al controlador los datos de composición en la capa LC0. Los datos de composición comienzan con la descripción de la primera escena de la presentación AV. Esta escena puede describirse como una recopilación jerárquica de objetos que utilizan paquetes de composición compuestos, o como una recopilación independiente de paquetes de composición de objetos. Una tabla que asocia los flujos elementales con los nodos en la descripción de la escena sigue directamente la descripción de la escena. El controlador carga los identificadores de objeto (identificadores de flujo) en la lista de objetos y la lista de representación que se mantienen en la memoria RAM de datos. La lista de representación contiene la lista de objetos que han de representarse en el dispositivo de visualización. Un objeto que queda deshabilitado por la interacción de un usuario se elimina de la lista de representación. Una instrucción de eliminación de nodo que se envía a través de un paquete de control de composición provoca la eliminación de los identificadores correspondientes del objeto de la lista de objetos. La jerarquía de nodos también se mantiene en la memoria RAM y se actualiza cada vez que se recibe una actualización de la composición.

El decodificador de composición lee datos de la memoria intermedia externa de datos y composición y los convierte en un formato que el decodificador de instrucciones comprende. La entrada externa incluye la interacción del usuario para seleccionar objetos, deshabilitar y habilitar objetos y ciertas operaciones predeterminadas en los objetos. Durante la ejecución del programa se forman dos listas en la memoria RAM de datos. La lista de objetos, que contiene una lista de objetos (identificadores de objeto) tratados actualmente por los decodificadores y una lista de reproducción que contiene la lista de objetos activos en la escena. Estas listas se actualizan de forma dinámica al recibirse la información de composición. Por ejemplo, si un usuario elige ocultar un objeto transmitiendo una instrucción a través de la entrada externa, el objeto se elimina de la lista de reproducción hasta que el usuario lo especifique. También ésta es la forma en que el sistema trata la entrada externa. Cada vez que se produce alguna interacción externa, el decodificador de composición lee la memoria intermedia de datos externa y lleva a cabo la operación solicitada.

El controlador SC también mantiene la temporización de cada objeto AV para indicarle a los decodificadores y memorias intermedias de los decodificadores el tiempo de decodificación y presentación. La información de temporización para los objetos AV se especifica en términos de su base temporal. El terminal utiliza el reloj del sistema para convertir la base temporal de un objeto en el tiempo del sistema. Para objetos que no necesitan decodificación, sólo son necesarios temporizadores de presentación. Estos temporizadores se cargan con los sellos temporales de decodificación y presentación para ese objeto AV. El controlador obtiene los sellos temporales del demultiplexor DMUX para cada objeto. Cuando expira un temporizador de decodificación para un objeto, se le indica al decodificador apropiado que lea datos de las memorias intermedias de entrada y que inicie el proceso de decodificación. Cuando expira un temporizador de presentación, los datos decodificados para ese objeto se transfieren a la memoria intermedia de marcos para la visualización. Podría utilizarse un enfoque dual de las memorias intermedia para permitir la escritura en una memoria intermedia de marcos mientras se visualiza en el monitor el contenido de la segunda memoria intermedia. El decodificador de instrucciones también puede reiniciar el demultiplexor DMUX o las memorias intermedias de entrada mediante la indicación de un reinicio, lo cual los inicializa al estado por defecto.

#### Flujo de información en el controlador

La figura 8 muestra el flujo de información en el controlador. Para mantener la figura sencilla las operaciones realizadas por el decodificador de instrucciones se muestran en grupos. Los tres grupos se refieren, respectivamente, a modificaciones de propiedades de objetos, la temporización de objetos y la señalización.



## *Modificaciones de propiedades de objetos*

Estas operaciones tratan los identificadores de objetos, también denominados “identificadores de flujos elementales”. Cuando se carga inicialmente una escena, se forma un gráfico de la escena con los identificadores de los objetos de la escena. El controlador también forma y mantiene una lista de los objetos de la escena (lista de objetos) y objetos activos en el objeto procedente de la lista de representación. Otras operaciones ajustan y configuran las propiedades de los objetos, tales como los parámetros de composición cuando el terminal recibe un paquete de composición.

## *Temporización de objetos*

Este grupo de operaciones trata la gestión de temporizadores de objetos para la sincronización, presentación y decodificación. Un sello temporal de objeto, especificado en términos de su base temporal de objeto, se convierte en tiempo del sistema y se ajusta el tiempo de presentación y decodificación de ese objeto. Estas operaciones también ajustan y reajustan los sellos temporales de expiración para objetos persistentes.

## *Señalización*

Las operaciones de señalización controlan el funcionamiento global del terminal. Mediante la señalización del controlador se ajustan, reajustan y operan varios componentes del terminal. El controlador comprueba los tiempos de decodificación y presentación de los objetos en la lista de representación y se los indica de forma correspondiente a las memorias intermedias de marcos de presentación y los decodificadores. También inicializa el demultiplexor DMUX para la lectura desde una red o un dispositivo de almacenamiento local. A petición del controlador, los decodificadores leen los datos procedentes de las memorias intermedias de entrada y transmiten los datos decodificados a las memorias intermedias de salida del decodificador. Los datos decodificados se pasan al dispositivo de presentación cuando lo señala el controlador.

## REIVINDICACIONES

1. Método para visualizar datos de vídeo/audiovisuales basados en objetos, que comprende lo siguiente:

- a) recibir, a lo largo del tiempo, una pluralidad de objetos de vídeo/audiovisuales e información de composición para los objetos, accediéndose a dichos objetos de vídeo/audiovisuales y a dicha información de composición para dichos objetos de vídeo/audiovisuales en canales lógicos separados;
- b) almacenar en una memoria caché (3) al menos uno de dichos objetos;
- c) emplear dicha información de composición para componer (5) escenas a partir de dichos objetos incluyendo uno de los objetos almacenados en la memoria caché; y
- d) visualizar (5) las escenas compuestas;

**caracterizado** porque

al menos uno de dichos objetos comprende un objeto persistente, presentando dicho objeto persistente un periodo de vida que se extiende más allá de su tiempo de presentación inicial, hasta un tiempo de expiración, y en el que un dicho objeto persistente utilizado repetidamente en una sesión se actualiza con mensajes de actualización adecuados.

2. Método según la reivindicación 1, en el que un sello temporal de expiración proporciona la vida de dicho objeto persistente, en el que el objeto persistente no está disponible después de transcurrido dicho sello temporal de expiración.

3. Método según la reivindicación 1, en el que un dicho objeto persistente se utiliza múltiples veces en dichas escenas.

4. Método según la reivindicación 1, en el que dichos objetos de vídeo/audiovisuales e información de composición para los objetos se disponen en una estructura de flujo de bits que comprende lo siguiente:

paquetes de composición de objetos para describir la composición y controlar la presentación de dichos paquetes; y

paquetes de datos de objeto que contienen los datos para dichos objetos.

5. Método según la reivindicación 1, en el que al menos uno de los objetos se recibe a partir de una conexión de red.

6. Método según la reivindicación 1, en el que al menos uno de los objetos se recibe desde una memoria local.

7. Método según la reivindicación 1, en el que al menos uno de los objetos se recibe a partir de una memoria local y al menos uno de los objetos se recibe a partir de una conexión de red, y en el que las escenas compuestas comprenden uno y otro objeto.

8. Método según la reivindicación 1, que comprende adicionalmente el responder a la entrada interactiva de usuario.

9. Método según la reivindicación 2, en el que dicho objeto persistente se almacena en una memoria caché de objeto (7) para usar más allá de un tiempo de presentación inicial durante el cual dichos objetos permanecen en la memoria caché (7) hasta la expiración de dicho sello temporal de expiración.

10. Método según la reivindicación 1, en el que un indicador de persistencia indica si un dicho objeto es un dicho objeto persistente o no.

11. Método según la reivindicación 4, en el que un dicho paquete de datos de objeto comprende un indicador de persistencia que indica si un dicho objeto comprende un dicho objeto persistente.

12. Método según la reivindicación 4, en el que dicho flujo de bits admite la representación de dichas escenas como una jerarquía mediante el empleo de objetos de composición de instrucciones.

13. Método según la reivindicación 4, en el que entradas de tabla asocian dichos flujos de objeto elementales en dichos canales lógicos con nodos en una escena jerárquica.

14. Aparato para visualizar datos de vídeo/audiovisuales basados en objetos que comprende:

- (a) un medio controlador (2) para controlar la adquisición a lo largo del tiempo de una pluralidad de objetos de vídeo/audiovisuales e información de composición para los objetos, accediéndose a dichos objetos de

vídeo/audiovisuales y a dicha información de composición para dichos objetos de vídeo/audiovisuales en canales lógicos separados.

(b) un medio de memoria caché (3) para almacenar al menos un dicho objeto;

(c) medios de composición (5), acoplados con la memoria (3) caché, para componer escenas a partir de dichos objetos de vídeo que incluyen uno de los objetos almacenados en la memoria caché; y

(d) un medio de visualización (6) para visualizar una dicha escena;

**caracterizado** porque

al menos uno de dichos objetos comprende un objeto persistente que presenta un periodo de vida que se extiende más allá de su tiempo de presentación inicial, hasta un tiempo de expiración, y en el que un dicho objeto persistente que se utiliza repetidamente en una sesión se actualiza con los mensajes de actualización adecuados.

15. Aparato de la reivindicación 14 que comprende un procesador que está instruido para lo siguiente:

(a) controlar la adquisición a lo largo del tiempo de una pluralidad de objetos de vídeo/audiovisuales e información de composición para los objetos, accediéndose a dichos objetos de vídeo/audiovisuales y dicha información de composición en canales lógicos separados;

(b) almacenar al menos uno de los objetos;

(c) componer (5) escenas a partir de dichos objetos de vídeo que incluyen el objeto almacenado en la memoria caché; y

(d) visualizar (6) las escenas compuestas;

**caracterizado** porque

al menos uno de dichos objetos comprende un objeto persistente y dicho objeto persistente se utiliza múltiples veces en dichas escenas.

16. Aparato de la reivindicación 14 que comprende lo siguiente:

(a) medios (1) conectados a través de un canal lógico a dicho medio (2) para controlar la adquisición a lo largo del tiempo de una pluralidad de objetos de vídeo/audiovisuales e información de composición para los objetos;

(b) medios (4) de decodificación para leer y decodificar dichos datos de dicha memoria caché (3);

(c) medios de memoria intermedia (4) para almacenar datos decodificados; y

(d) medios de memoria (7) para almacenar objetos de vídeo/audiovisuales más allá de su tiempo de presentación inicial.

17. Aparato de la reivindicación 14, en el que a dichos objetos de vídeo/audiovisuales basados en objetos se accede como a una pluralidad de flujos de objeto elementales en canales lógicos.

18. Aparato de la reivindicación 16, en el que dicho objeto persistente se almacena en una memoria caché de objetos (7) para usar más allá de dicho tiempo de presentación inicial y en el que dicho objeto permanece en dicha memoria caché (7) hasta que expire un sello temporal de expiración.

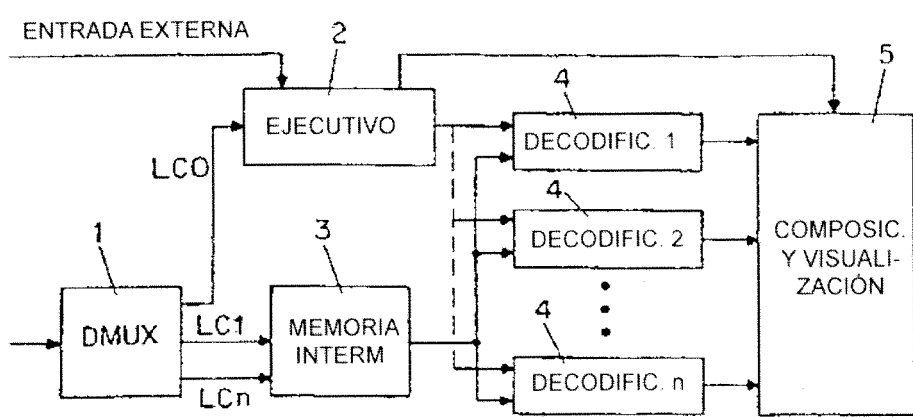


FIG. 1

|                           |
|---------------------------|
| ID OBJETO                 |
| DTS                       |
| PTS                       |
| LTS                       |
| OCR                       |
| PARÁMETROS DE COMPOSICIÓN |

FIG. 2a

|                           |
|---------------------------|
| ID OBJETO COMP            |
| PTS                       |
| LTS                       |
| CUENTA OBJETO             |
| PARAMETROS DE COMPOSICIÓN |
| ID OBJETO                 |
| DTS                       |
| OCR                       |
| PARAMETROS DE COMPOSICIÓN |

⋮

|                    |
|--------------------|
| ID OBJETO          |
| CI-PI              |
| PRIORIDAD          |
| ETS                |
| TIPO VOP (I, P, B) |
| DATOS DE OBJETO    |

FIG. 2b

|                           |
|---------------------------|
| ID OBJETO                 |
| DTS                       |
| OCR                       |
| PARAMETROS DE COMPOSICIÓN |

FIG. 2c

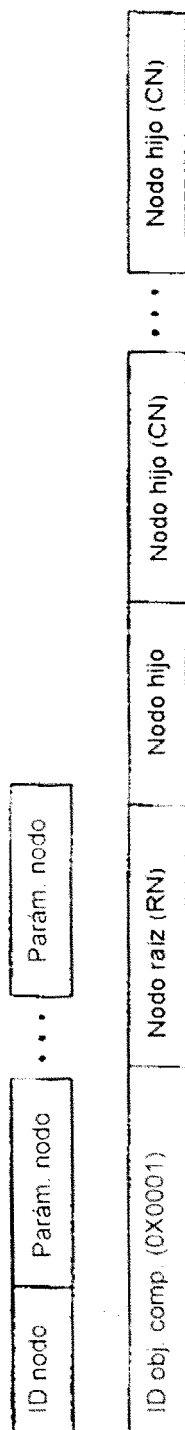


FIG. 3

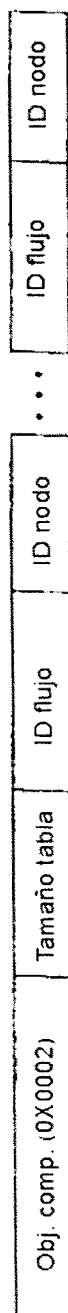


FIG. 4

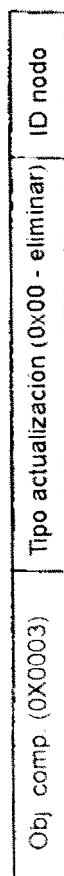
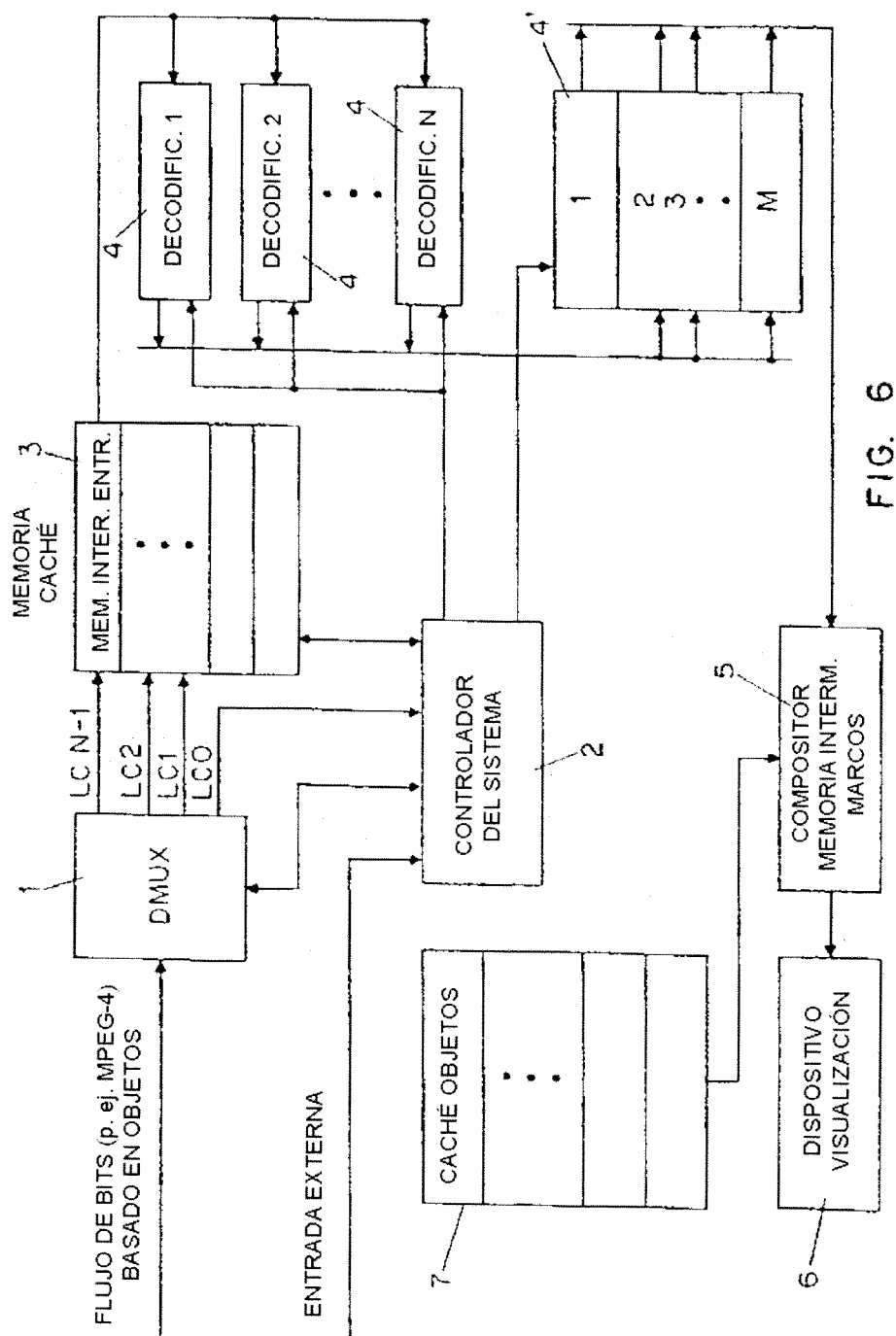


FIG. 5



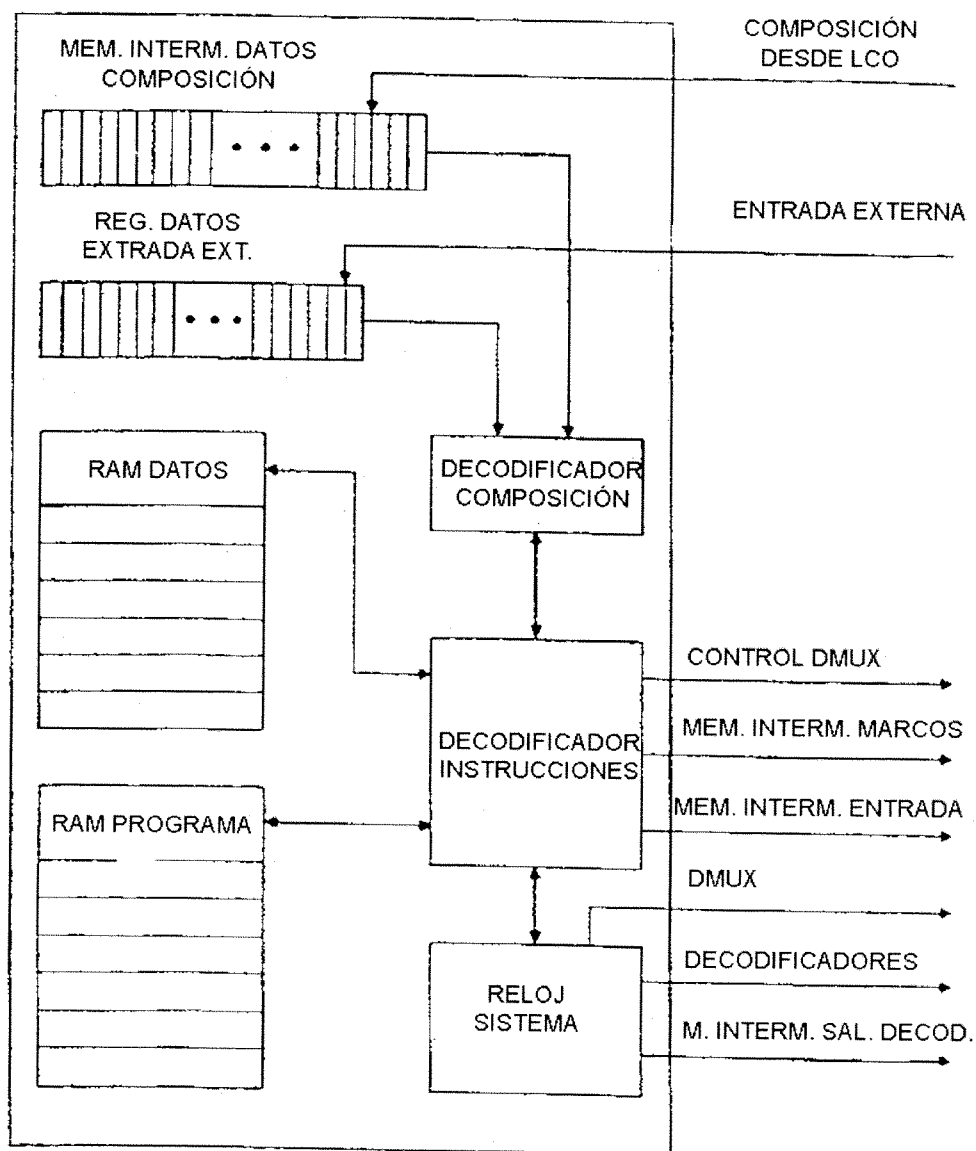


FIG. 7

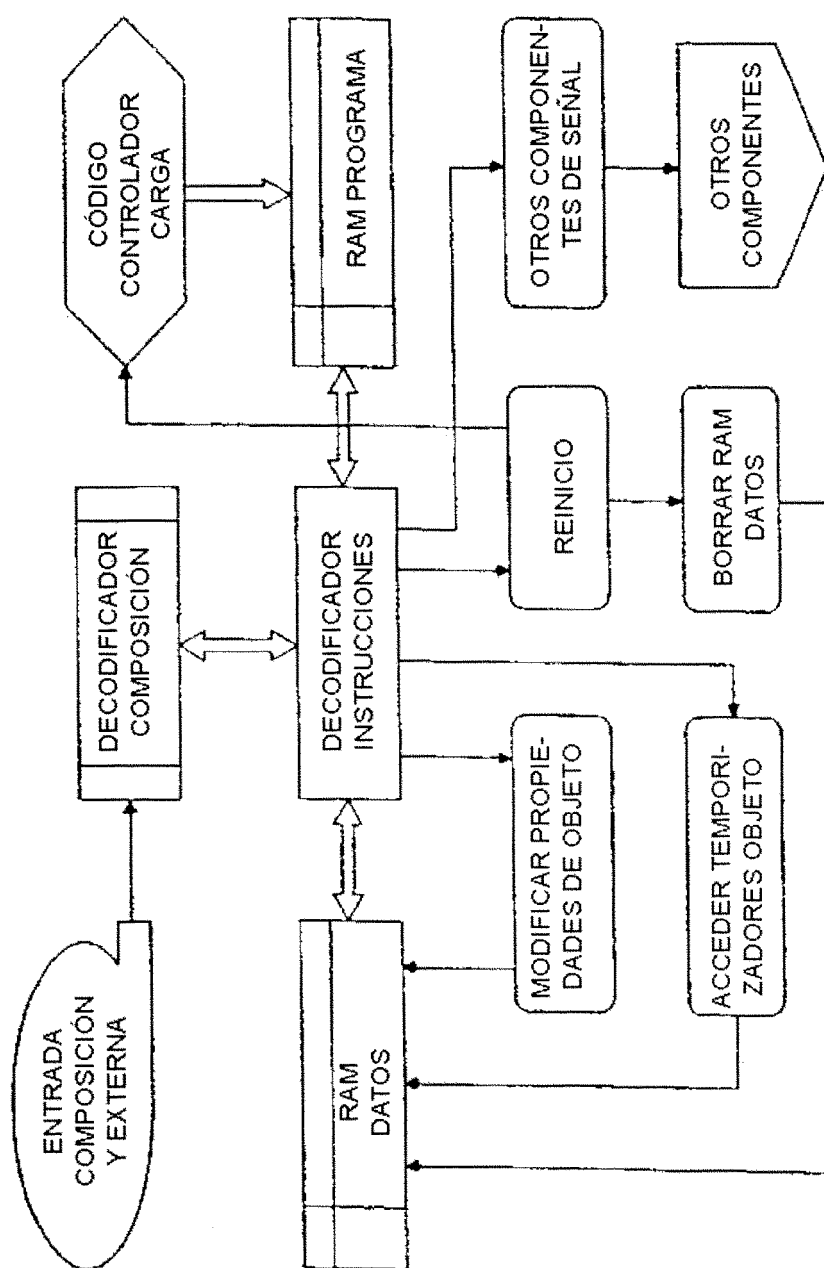


FIG. 8