

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 559 680**

51 Int. Cl.:

H04L 29/06 (2006.01)

G06F 17/20 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **06.12.2007 E 07845839 (5)**

97 Fecha y número de publicación de la concesión europea: **21.10.2015 EP 2124419**

54 Título: **Dispositivo de gestión orientado a objetos para mensaje ASN.1**

30 Prioridad:

11.12.2006 CN 200610161181

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

15.02.2016

73 Titular/es:

**ZTE CORPORATION (100.0%)
ZTE PLAZA, KEJI ROAD SOUTH, HI-TECH
INDUSTRIAL PARK, NANSHAN DISTRICT
SHENZHEN, GUANGDONG 518057, CN**

72 Inventor/es:

ZHANG, XINPING

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 559 680 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

Dispositivo de gestión orientado a objetos para mensaje ASN.1

5 CAMPO DE LA INVENCION

La presente invención se refiere a un dispositivo de gestión para mensaje ASN.1 y más en particular, a un dispositivo de gestión orientado a objetos para mensaje ASN.1.

10 ANTECEDENTES DE LA INVENCION

ASN.1 (Abstract Syntax Notation 1) es la norma definida por la ITU para el formato de los mensajes transmitidos en red y proporciona un formato estándar para conversión de datos entre los nodos. Cada nodo solamente se encarga de los traducidos desde/a ASN.1 mientras que no conoce necesariamente el formato de los datos memorizados en cualquier lugar en la red. ASN.1 está constituido por dos partes, una parte es ISO 8824/ITU X.208 que describe los datos, el tipo de datos y el formato de secuencia en el mensaje, es decir, la sintaxis de datos; la otra parte es ISO 8825/ITU X.209 que describe cómo combinar cada parte de datos en un mensaje, es decir, la regla de codificación de datos básica, que fue inicialmente desarrollada como una parte de X.409 y más adelante, de forma independiente convirtiéndose en una norma. La norma PKI (Public Key Infrastructure) de la 1ª generación se basa principalmente en ASN.1 y se utiliza en SNMP para definir el formato de la unidad de datos de protocolo SNMP y su objeto y se aplica ampliamente en otras áreas de la comunicación y ordenador.

Por lo tanto, la forma de utilizar el formato del mensaje ASN.1 para la comunicación es la forma común en un sistema de gestión de redes de telecomunicaciones.

25 Un mensaje ASN.1 tiene las características siguientes:

1. El mensaje tiene una estructura arborescente que puede definir la disposición jerárquica de mensajes en la forma deseada.
2. El tipo de un nodo en el mensaje puede ser diverso y puede ser un tipo de datos simples tal como INTEGER, GRAPHICSTRING, ENUM, etc. o un tipo de datos compuestos que incluyen nodos hijos, tal como SET, SEQUENCE, SET OF o SEQUENCE OF.
3. En las estructuras de SET y SEQUENCE, puede no existir un determinado nodo, puesto que el nodo del mensaje correspondiente está permitido que se establezca como OPTIONAL lo que significa que "puede o no puede existir".
4. Cuando se utiliza un mensaje ASN.1, en primer lugar una estructura de ASN.1 se define en el mensaje ASN.1 y luego, es objeto de compilación en una estructura de lenguaje de programación intermedia tal como Java, C++ por un compilador y luego se realiza la comunicación utilizando la estructura del lenguaje de programación correspondiente.
5. ASN.1 soportas BER (Basic Encoding Rules), DER (Distinguished Encoding Rules) y VAL (notación de valor de ASN.1) para codificar el ejemplo de estructura de lenguaje intermedia en flujo de bits, pero existe un problema cuando el flujo de bits se decodifica en el ejemplo de estructura de lenguaje de soporte. En primer lugar, necesita conocer la estructura correspondiente al flujo de bits con el fin de utilizar el método de decodificación de la estructura a decodificar y si la estructura incluye ANY (el tipo incierto), los datos de este tipo sólo pueden decodificarse después de que se determine su tipo específico por intermedio de otro campo miembro.

El documento US 2005/0114355 A1 da a conocer un método para realizar conversiones de estructuras de datos, en donde una estructura de datos comprende al menos dos elementos situados en un orden predeterminado en la estructura de datos. La estructura de datos se define utilizando un lenguaje de definición y se representa como una cadena de bits origen. Durante la conversión, la cadena de bits origen se convierte en una cadena de bits objetivo. El método para realizar la conversión de estructura de datos gestiona los elementos de estructuras de datos en el mismo orden en que están situados en la estructura de datos sin realizar un análisis sintáctico del flujo de bits origen.

El método típico de utilización del mensaje ASN.1 en software C/S es:

1. Establecer la conexión entre un cliente y un servidor;
 2. Interaccionar los protocolos entre el cliente y el servidor:
- a) Un remitente de mensaje que rellena la cabecera y cuerpo del mensaje ASN.1 en donde la cabecera incluye un código de orden y el cuerpo aplica el tipo de ANY y su tipo de datos real varía dependiendo del código de orden;

- b) El remitente del mensaje codifica el mensaje en un flujo de bits y lo envía a un receptor de mensaje;
 - c) El receptor de mensaje que recibe el flujo de bits, decodifica la cabecera del mensaje y luego, decodifica el cuerpo del mensaje con el tipo de datos especial en conformidad con el código de orden;
 - d) El receptor del mensaje responde en conformidad con el contenido del mensaje;
3. Se completa la interacción y se desconecta la conexión.

Un mensaje estimulado se necesita siempre en la herramienta de prueba de interfaz del mensaje y el mensaje estimulado tiene dos orígenes, uno es generar el lenguaje de script explicatorio y el otro es establecer ejemplos de la estructura del mensaje y modificar, además, los ejemplos por intermedio de datos persistentes. Sin embargo, en programas de aplicación, a veces el mensaje debe conservarse persistentemente como un fichero o memorizarse en la base de datos y reconstruir el mensaje a partir del mensaje persistente. Además, es deseable que los datos en el fichero sean legibles y que los datos en ese fichero se puedan modificar.

En términos generales, BER se utiliza para la transmisión en el subsistema de gestión de red. Sin embargo, puesto que no es susceptible de lectura y no puede recuperar directamente los datos ANY, el flujo de bits de BER no es adecuado para memorizar el mensaje de forma persistente, como en el caso de DER, que es un BER con codificación única. Mientras que VAL (ASN.1 notación de valor) es un formato de caracteres de valor de nombre legible que utiliza "{" y "}" para realizar la estructura jerárquica de marcador de arborescencia y tiene las características siguientes:

1. El nodo no más exterior aplica el formato de "valor de nombre" y el nodo raíz más exterior aplica el formato de "valor de nombre de tipo de estructura de datos".
2. Para el valor de SET, SEQUENCE, los datos se introducen entre "{" y "}" y se utiliza "," para separar los miembros.
3. Para el valor de SET OF, SEQUENCE OF, el conjunto de datos se encierra en "{" y "}" y se utiliza "," para separar los elementos en el conjunto.
4. Para el valor de ANY, si los datos que contienen es una cadena de bytes, se visualiza utilizando el sistema hexadecimal; y si es un tipo de datos ASN.1 especial, se visualiza utilizando el formato de "valor de nombre de tipo de estructura de datos especial".

Las características del formato VAL hacen posible su recuperación directa como ejemplo de mensaje ASN.1 desde el flujo de bits. Sin embargo, las plataformas de compilación de ASN.1 actuales, incluyendo DSET y ESNACC no soportan la función de recuperación completa del ejemplo de mensaje ASN.1.

SUMARIO DE LA INVENCION

La presente invención ofrece un dispositivo de gestión orientado a objetos para un mensaje ASN.1 para resolver el problema técnico de que el ejemplo de estructura de mensaje debe restablecerse después de memorizarse de forma persistente, para superar el inconveniente de que la decodificación podría realizarse solamente si el tipo de datos del flujo de bits se conociera por anticipado, para disminuir la dificultad de la construcción del mensaje y para el conocimiento significativo de la carga de trabajo del desarrollo y mantenimiento de mensajes.

Con el fin de resolver el problema técnico anterior, la presente invención ofrece un dispositivo según la reivindicación 1. Mejoras adicionales y formas de realización se dan a conocer en las reivindicaciones subordinadas. Además, se da a conocer un dispositivo de gestión orientado a objetos para el mensaje ASN.1, que incluye:

un módulo de soporte inferior de ASN.1 para la compilación del mensaje ASN.1 en el código de ejemplo de lenguaje de programación;

un módulo de manipulación común para el procesamiento del mensaje para soportar las aplicaciones de mensajes;

un mecanismo de inserción de mensajes para generar una inserción del cuerpo del mensaje a partir del código generado por el módulo de manipulación común, con la tabla de descripción de clase siendo la interfaz de inserción;

en donde el módulo de manipulación común o el mecanismo de inserción de mensaje se utiliza para modificar el código de ejemplo de lenguaje de programación generado por el módulo de soporte inferior de ASN.1, para utilizar funciones virtuales para añadir la información de descripción de clase incluyendo la información de descripción de estructura de clase y las funciones para crear el ejemplo de clase; para añadir el soporte de manipulación de ejemplo de clase para el código generado; para establecer la tabla de descripción de clase TopClass que incluye la relación correspondiente entre el nombre de TopClass y la descripción en el código generado y la TopClass es la

clase de capa superior y no las clases hijos.

Además, dicho código de ejemplo de lenguaje de programación es un código de programación de C++ o JAVA.

5 Además, para el tipo de datos simples y para el tipo de datos compuestos, dicho módulo de manipulación común o mecanismo de inserción de mensaje complementa el nombre de tipo, el valor entero y la función de creación de ejemplo de clase del tipo de datos; para datos SET y SEQUENCE, complementa el nombre de cada miembro, establece si el tipo de datos es opcional y complementa la tabla correspondiente de la información de descripción del miembro; para ENUM, complementa la tabla correspondiente entre el símbolo del miembro y el valor de número.

10 Además, dicho módulo de manipulación común o mecanismo de inserción de mensaje añade la tabla de descripción de clase en el código de ejemplo de lenguaje de programación, indica la información de descripción de clase en la forma de tabla de descripción de clase, toma el nombre de clase como la palabra clave y clasifica los elementos en la tabla en conformidad con el nombre de clase o la convierte en una denominada tabla *harsh* para ser indexada de forma conveniente.

15 Además, para el tipo de datos simples, la adición del soporte de la manipulación de ejemplo de clase en el ejemplo de lenguaje de programación por el módulo de manipulación común o el mecanismo de inserción de mensaje incluye:

20 añadir el soporte de funciones virtuales para obtener y establecer el valor de datos;

25 para el tipo de datos compuestos, la adición del soporte de la manipulación de ejemplo de clase en el código de ejemplo de lenguaje de programación por el módulo de manipulación común o el mecanismo de inserción de mensaje incluye:

- a) añadir el soporte de ruta de marcador de punto;
- 30 b) añadir el soporte de establecimiento de un símbolo para indicar si existen, o no, los miembros en SEQUENCE y SET;
- c) añadir el soporte de obtención de la longitud de la red matricial de datos, estableciendo la longitud de la red de datos y los elementos transversales en la red de datos para la red de datos de SEQUENCE OF y SET OF;
- 35 d) añadir el soporte de la operación de selección para los datos CHOICE;
- e) añadir el soporte de establecimiento de los datos ANY que pueden establecer u obtener la variable del tipo real;
- 40 f) añadir el soporte de obtención del objeto del nodo hijo correspondiente por intermedio de la ruta de marcador de punto.

Además, el módulo de manipulación común utiliza la lista de enlaces de arborescencia *AsnTreeObject* para indicar la estructura del mensaje ASN.1.

45 Además, el módulo de manipulación común o el mecanismo de inserción de mensajes se utiliza para generar datos VAL a partir del código generado.

Además, el módulo de manipulación común o el mecanismo de inserción de mensajes se utiliza para recuperar el ejemplo de lenguaje de programación a partir de los datos VAL.

50 Asimismo, el módulo de manipulación común o el mecanismo de inserción de mensaje se utiliza para convertir los datos VAL en datos XML.

Además, dicho dispositivo de gestión orientado a objetos para el mensaje ASN.1 comprende un paquete de interfaz gráfica y un módulo de edición de mensaje gráfico.

Dicho paquete de interfaz gráfica se utiliza para desarrollar la interfaz gráfica.

60 Dicho módulo de edición de mensaje gráfico utiliza el paquete gráfico y el módulo de manipulación común para proporcionar a los usuarios una función de edición de mensaje de interfaz gráfica para modificar, crear y editar el mensaje ASN.1 por intermedio de la modificación de la tabla de lista de enlaces de arborescencia *AsnTreeObject*.

Además, dicho dispositivo de gestión orientado a objetos para el mensaje ASN.1 comprende, además, el paquete de soporte de script y el módulo de manipulación de script de mensajes.

65 Dicho paquete de soporte de script se utiliza para la ejecución del script y para soportar la extensión del script.

Dicho módulo de manipulación de script de mensaje utiliza el paquete de soporte de script y el módulo de manipulación común para añadir la función de manipulación de mensaje ASN.1 en el script para modificar, crear y editar el mensaje ASN.1 por intermedio de la modificación de la lista de enlaces de arborescencia AsnTreeObject.

5 La presente invención tiene las ventajas siguientes:

1. El mensaje ASN.1 puede memorizarse de forma persistente y los datos persistentemente memorizados pueden recuperarse como ejemplo de estructura de datos y los datos ANY en los datos han sido también decodificados.
- 10 2. El mensaje puede aplicar una manipulación de símbolo de cadena de valor de nombre para modificar fácilmente el mensaje y soportar el lenguaje de script para establecer el mensaje ASN.1.
- 15 3. El método de generación de mensajes es independiente de la definición del mensaje y el cuerpo del mensaje ASN.1 puede aplicar la forma de inserción que se genera automáticamente por MAKEFILE que compila el fichero ASN.1 para facilitar la modificación y actualización del mensaje y para disminuir la carga de trabajo de mantenimiento.
4. El dispositivo es universal y no se basa en un tipo de estructura ASN.1 especial o en un compilador ASN.1.

20 BREVE DESCRIPCIÓN DE LOS DIBUJOS

La Figura 1 ilustra la composición de la forma de realización en conformidad con la presente invención;

25 La Figura 2 ilustra el proceso de conversión de mensajes;

La Figura 3 es el sistema de herencia operativa de la estructura de clase básica ESNACC C++;

La Figura 4 es el sistema de herencia de la estructura de clase básica de DSET C++.

30 DESCRIPCIÓN DETALLADA DE LAS FORMAS DE REALIZACIÓN DE LA INVENCION

La presente invención se describirá, con más detalle, haciendo referencia a los dibujos adjuntos y a las formas de realización.

35 Según se ilustra en la Figura 1, el módulo de soporte inferior de ASN.1 utiliza la estructura de ASN.1 para realizar una inserción de mensaje y algunas manipulaciones comunes se realizan sobre la base del mecanismo de inserción de mensaje y luego, se basan en lo que antecede para realizar una edición de mensaje gráfico y una manipulación de mensaje de script. La manipulación común del mensaje comprende cualquier encapsulación de aplicación básica que incluye la creación y modificación de mensajes de tipos especiales y la creación, codificación y decodificación de ficheros VAL.

40 El módulo de soporte inferior de ASN.1: compila los ficheros ASN.1 en un código de lenguaje de programación especial tal como C++ o Java y compila la estructura de ASN.1 en una clase de lenguaje de programación. El módulo de soporte inferior de ASN.1 ofrece soporte de la manipulación de códigos generados. A modo de ejemplo, C++ ofrece, en general, varias bibliotecas que incluyen las clases básicas del lenguaje especial de ASN.1 y algunas funciones de soporte de manipulación para el código generado objeto de llamada.

45 El mecanismo de inserción de mensaje: con el fin de procesar uniformemente el cuerpo del mensaje ASN.1, el código de programación se genera a partir del mensaje ASN.1 por el compilador de ASN.1 y después de mejorarse en conformidad con algunos requisitos de funciones de interfaz, se compila en una biblioteca informática por intermedio del compilador de lenguajes. A continuación, puede ser objeto de carga dinámica por un programa de aplicación. El mecanismo de inserción puede utilizar un programa de escaneado para procesar el código de programa generado y generar una biblioteca por intermedio del compilador de lenguajes. Por lo tanto, otros módulos de aplicación solamente se basan en la interfaz y no existe necesidad de cambiarlos cuando cambia el cuerpo del mensaje. A modo de ejemplo, C++ podría generar la biblioteca dinámica con una interfaz especial; JAVA puede empaquetar una clase que soporta una interfaz especial, cargar el paquete cuando se ejecuta el código y crear una clase de JAVA para soportar la interfaz especial por intermedio de la denominada reflexión de clase.

50 El módulo de manipulación común: procesa uniformemente el mensaje sobre la base del módulo de soporte inferior de ASN.1 y la interfaz del mecanismo de inserción de mensajes para soportar adecuadamente las manipulaciones del módulo de aplicación de mensajes, tal como la creación de una instancia de clase, la modificación de los miembros de datos de clases, la generación de un mensaje a partir de los datos VAL y la comparación de dos mensajes.

65 El paquete de interfaz gráfica se ofrece por una tercera parte y puede utilizarse fácilmente por usuarios para desarrollar interfaces gráficas.

- 5 El módulo de edición de mensaje gráfico que utiliza el paquete gráfico y el módulo de manipulación común de ASN.1 para ofrecer a los usuarios la función de edición de mensaje de interfaz gráfica de fácil uso, con lo que los usuarios pueden crear y editar los mensajes de ASN.1 en la forma de lista de arborescencia que incluye el nombre de campo, nombre de tipo, valor entero, la longitud del miembro de datos y el valor del miembro de datos. Puede abrir y memorizar el formato VAL.
- 10 El paquete de soporte de script que es un paquete para la ejecución de los scripts y para soportar la extensión de script, tal como PYTHON o TCL que explica la manipulación con una eficiencia relativamente baja pero que puede utilizarse fácilmente en multi-plataformas; y el paquete puede extenderse mediante lenguajes tales como C++.
- 15 El módulo de manipulación de script de mensajes que utiliza el módulo de soporte de extensión de script y el módulo de manipulación común de ASN.1 para añadir la función de manipulación de mensajes ASN.1 en los scripts para hacer que los usuarios puedan acceder fácilmente a los mensajes mediante scripts.
- 20 La Figura 2 describe la interconversión entre los mensajes relacionados con ASN.1, la definición de la estructura de ASN.1, la estructura de clase correspondiente de C++ y el formato de flujo de bits. Además, muestra el proceso de conversión entre los datos manipulados por el módulo de manipulación de script de ASN.1 y el módulo de edición de gráficos de ASN.1 y los datos de mensajes. Más concretamente, con el método C.2 de la presente invención, el mensaje de VAL memorizado de forma persistente puede recuperarse completamente como un ejemplo de clase de C++ y codificarse de forma adicional. El método de la presente invención soporta también las aplicaciones de G y H.
- Etapa A: La estructura de clase C++ puede generarse a partir del fichero de ASN.1 por intermedio del compilador de C++ del ASN.1 y se puede crear, además, un ejemplo de estructura de clase.
- 25 Etapa B.1: Un ejemplo de estructura de clase de C++ puede codificarse en un flujo de bits BER o DER.
- Etapa B.2: El flujo de bits de BER o DER puede decodificarse en un ejemplo de estructura de clase de C++ específica.
- 30 Etapa C.1: Pueden generarse datos VAL a partir del ejemplo de estructura de clase de C++.
- Etapa C.2: Los datos VAL pueden recuperarse completamente como un ejemplo de estructura de clase de C++ mediante el método de la presente invención.
- 35 Etapa D: El ejemplo de clase de C++ puede analizarse para generar una estructura arborescente de tipo AsnTreeObject.
- Etapa E: Datos VAL puede generarse a partir de la estructura arborescente AsnTreeObject.
- 40 Etapa F: Otros formatos, tales como XML y VAL pueden interconvertirse.
- Etapa G: Los scripts de ASN.1 pueden crear, además, un ejemplo de estructura de clase de C++ modificando AsnTreeObject.
- 45 Etapa H.1: AsnTreeObject puede visualizarse en el editor de gráficos.
- Etapa H.2: El editor de gráficos de ASN.1 puede generar AsnTreeObject.
- 50 El método de construcción de mensajes se describirá en detalle haciendo referencia a un ejemplo de plataforma de prueba de gestión de red y para la plataforma de compilación de DESET ASN.1 y para la plataforma de compilación de ESNACC ASN.1, la presente invención pone en práctica el editor de mensajes y el método de utilización de scripts para crear los mensajes. Aunque la presente invención se describe principalmente utilizando C++, puede realizarse utilizando otro lenguaje de programación orientado a objetos tal como JAVA.
- 55 El contenedor:
- La presente invención utiliza un contenedor para la memorización intermedia de la información de los nodos de mensajes que necesitan compararse. En la práctica, el contenedor puede ser en forma de cola denominada deque, una lista o una pila. A continuación, se describirá el método de comparación de la presente invención tomando a modo de ejemplo el deque. Deque significa ampliamente una idea de proceso que puede ponerse en práctica mediante la autodefinition o utilización de un tipo deque STL: lista ofrecida en la STL (Standard Template Library). El tipo de cadena utilizada en la presente invención puede ser std::string, CString en MFC o ser autodefinida.
- 60 El origen del mensaje de prueba:
- 65 El mensaje utilizado en la presente invención se origina a partir del mensaje en la comunicación del subsistema de

varios proyectos de gestión de redes contrastados. Puesto que las versiones de la gestión de red son diferentes, el mensaje de nueva versión tiene alguna adición o supresión, lo que requiere que la plataforma de prueba pueda simular el mensaje, editar el mensaje mediante el editor de mensajes, citar o crear el mensaje mediante script.

5 La puesta en práctica de la plataforma ESNACC:

(1) La plataforma de compilación

ESNACC es un compilador de ASN.1 con código fuente abierto que soporta BER y PER y puede proporcionar, a la salida, solamente datos VAL, pero no puede decodificar los datos VAL como el ejemplo de estructura de C++. SEQUENCE OF y SET OF aplican la clase de plantilla informática y no tienen ninguna función de miembro de clase rápida uniforme para soportar la manipulación de la longitud de la red de datos y los elementos transversales de los nodos de la red de datos. Haciendo referencia a la Figura 3 para el sistema de herencia de clase básica de la clase básica superior del tipo AsnType y el sistema de herencia del tipo de mensaje ESNACC, la Figura 3 representa el gráfico jerárquico de la relación de herencia de clase básica de ESNACC y la clase básica de ESNACC es AsnType que es una clase básica abstracta. Sobre esta base, tiene lugar la herencia operativa de algunos tipos simples para soporte de los tipos de datos ASN.1 básicos tales como, AsnInt, AsnReal, AsnRelativeOid, AsnAny, AsnOids, AsnBool, AsnNull y AsnString, en donde AsnInt se deriva también de PERGeneral, AsnEnum se deriva de AsnInt y AsnOid se deriva de AsnRelativeOid; AsnList es una clase intermedia que realiza otras clases de ASN.1, AsnSetOf y AsnSeqOf se derivan de AsnList; std::string y std::list en STL se utilizan para soportar además SEQUENCE OF y SET OF. Algunos tipos de cadenas aplicados tales como VisibleString, GraphicString, Ia5String, PrintableString y NumericString se derivan del tipo de cadena básica AsnString, en donde GeneralizedTime y UTCTime se derivan de VisibleString.

25 (2) La manipulación del flujo de mensajes

Puesto que la herramienta de prueba de interfaz necesita procesar el mensaje basado en ASN.1, necesita ofrecer un editor de mensajes ASN.1 para crear un mensaje por intermedio de la interfaz gráfica; la herramienta de prueba soporta el script para la manipulación del mensaje, incluyendo la modificación del valor y de la longitud de datos. La carga de trabajo será de gran magnitud si se comienza desde la codificación y decodificación de nivel inferior del mensaje ASN.1 y por ello, necesita utilizarse el compilador de ASN.1 actual, sobre la base de que el compilador sea modificado por el código de C++ generado del mensaje ASN.1 sea objeto de un segundo escaneado para generar el código de programación de C++ deseado. Lo que antecede es el método de procesamiento que se utiliza principalmente para crear el mecanismo de inserción de mensajes.

- 35 1. La clase básica de ESNACC comprende información de descripción de clase que incluye la función de creación del tipo correspondiente y puesto que la información de descripción de algunas estructuras de C++ generadas, tales como INTEGER (1...200) no están completas, la información de descripción necesita complementarse mediante la modificación del compilador.
- 40 2. Las manipulaciones básicas del ejemplo de clase de C++ puede realizarse, pero no se soporta la ruta de marcador de punto y no tiene ninguna función virtual para obtener y establecer el valor de datos; para SEQUENCE OF y SET OF, no existe ninguna manipulación de unificación de la longitud de la red de datos. Una función de soporte de ruta de marcador de punto se añade para soportar la obtención del objeto del nodo hijo correspondiente por intermedio de la ruta de marcador de punto y la creación de un miembro no existente mediante selección de parámetros; una función de miembro se añade en AsnType para definir la interfaz de función virtual en AsnType para obtener la magnitud, establecer la magnitud y obtener las funciones de todos los nodos miembros hijos. Funciones virtuales se realizan en la AsnList de clase de plantilla informática para obtener la magnitud, establecer la magnitud y obtener las funciones de todos los nodos miembros hijos.
- 50 3. La tabla de descripción de clase de TopCLASS se establece y ESNACC soporta la fin de META a compilarse, generando la tabla de descripción de clase const AsnModuleDesc *asnModuleDescs[]. Para poder buscar elementos con rapidez en la tabla de descripción de clase, la tabla puede convertirse, además, en un mapa con el nombre como la palabra clave o red de datos secuencial.
- 55 4. El código en las etapas 1 a 3 se genera en un DLL y asnMoudleDescs variable pueden derivarse de la función AsnModuleDesc **GetMoudleDescs(), y todas las clases de TOPCLASS pueden derivarse si así se desea. De este modo, se convierte en un mecanismo de inserción cuya interfaz es GetMoudleDescs y toda la descripción de TOPCLASS puede obtenerse por intermedio de esta estructura, creando y modificando sus ejemplos. El puntero de asnMoudleDescs puede registrarse cuando se carga el mecanismo de inserción o después de que se cargue el mecanismo de inserción. La etapa A en la Figura 2 se pone en práctica y las etapas anteriores 1 a 4 se realizan en el mecanismo de inserción de mensaje según se ilustra en la Figura 1.

- 60 5. Una arborescencia AsnTreeObject puede definirse como:

```
65 class AsnTreeObject {
```

```

public:
ansistring nodename;
ansistring nodeVALue;
int ans1type;
5   ansistring typename;
    AsnType *pParent;
    AsnTreeObjectList childrenlist;
};

```

10 AsnTreeObjectList es la lista del punto de AsnTreeObject. Por supuesto, esta estructura puede añadir otros datos extras cuando sea deseable. Esta estructura es adecuada para visualizarse y modificarse en forma transversal. De forma interrelacionada, las etapas D y E en la Figura 2 se realizan. Este contenido se pone en práctica en el módulo de manipulación común representado en la Figura 1.

15 6. El ejemplo de estructura de mensaje de ASN.1 C++ y los datos VAL son mutuamente generados y la PrintVAL para generar datos VAL puede ser una función miembro en AsnType o una función independiente con el punto de AsnType como el parámetro. La función que genera el ejemplo de ASN.1 C++ a partir de los datos VAL puede ser una función independiente que utiliza los datos VAL como su parámetro y el punto de AsnType como su parámetro y el código de error como el valor del retorno y también puede ser una función miembro estática del AsnType con el mismo formato de función. Si es una función de miembro estática, la función de esta parte debe ponerse en práctica en el mecanismo de inserción y si es una función independiente, debe ponerse en práctica en otros módulos fuera del mecanismo de inserción. Las etapas C1 y C2 en la Figura 2 se realizan en este momento operativo. La etapa C.1 puede realizarse utilizando las operaciones en la etapa 1 y etapa 2 anteriores y la realización de la etapa C.2 necesita utilizar las operaciones en las etapas 1, 2 y 3 anteriores. Esta parte se pone en práctica en el mecanismo de inserción de mensajes que se representa en la Figura 1.

20 7. Para poder editar un mensaje en el editor de gráficos, los nombres de tipos de todas las estructuras de ASN.1 deben ser objeto de listado y en primer lugar, la función de creación, a modo de ejemplo, se obtiene a partir de asnMoudleDescs en conformidad con el tipo de mensaje y luego, el mensaje se visualiza utilizando el AsnTreeObject en la etapa 5 anterior y el AsnTreeObject se edita en conformidad con la información restrictiva de la descripción de clase de mensaje para generar, además, datos VAL y generar un ejemplo de ASN.1 C++ a partir de los datos VAL. La razón principal de generar un ejemplo de clase de C++ a partir de los datos VAL aunque no directamente desde AsnTreeObject es que resulta fácil generar datos VAL y se basa menos en el sistema de estructura de clase básica de plataforma de compilación y por ello, la plataforma de compilación tiene una buena transportabilidad. Con el método anterior, el mensaje puede crearse y modificarse también mediante script y memorizarse persistentemente en datos VAL.

30 8. La manipulación del mensaje de ASN.1 se soporta utilizando un lenguaje de script PYTHON mediante extensión de script y de este modo, el mensaje se crea y utiliza en la herramienta de prueba por intermedio del script.

40 La puesta en práctica de la plataforma DSET:

Con el fin de describir todavía más el sistema técnico de la presente invención, la presente invención se describirá, con mayor detalle, tomando como ejemplo la plataforma de compilación DSET con referencia a las formas de realización y los dibujos adjuntos.

45 (1) La plataforma de compilación

Para ser plataforma de compilación, DSET es un compilador de ASN.1 comercial sin código fuente abierto que soporta BER, PER y DER, puede proporcionar a la salida datos VAL y modificar los miembros de datos en el ejemplo de clase de C++ mediante los datos VAL, pero no puede identificar el tipo real de los datos ANY. La información de descripción ENUM es demasiado débil operativamente para obtener, de forma directa, la tabla correspondiente de la cadena de nombre de símbolos y el valor de datos y no tiene ninguna tabla de descripción de clase TOPCLASS. Haciendo referencia a la Figura 4 para el sistema de herencia de clase básica, la Figura 4 es la representación gráfica jerárquica de la relación de herencia de clase básica DSET y su clase básica es AbstractData, que es una clase básica abstracta, a partir de la que se obtiene una herencia operativa directa de algunos tipos básicos tales como CHOICE, AbstractReal, ENUMERATED, Integer, BOOLEAN y Any; StrucData, D_array, D_seqOfVal y D_seqwOfObj son tipos intermedios que se utilizan para poner en práctica otras clases de ASN.1, D_SeqOfVal y D_SeqOfObj se derivan de D_array, AbstractString se deriva de D_SeqOfVal y UTCTime, GeneralizedTime, PrintableString, IA5String, GraphicString y VisibleString se derivan de AbstractString.

60 (2) La manipulación del flujo de mensajes

Cuando se realiza una herramienta de prueba de interfaz basada en el mensaje ASN.1, necesita disponerse de un editor de mensajes ASN.1 para crear un mensaje mediante la interfaz gráfica; la herramienta de prueba soporta el script para manipular el mensaje, incluyendo la modificación del valor y la longitud de datos. La carga de trabajo será

de gran magnitud si se comienza desde la codificación y decodificación de nivel bajo del mensaje ASN.1 y por ello, necesita utilizar el compilador de ASN.1 actual, sobre la base de que el código de C++ generado mediante la compilación del mensaje ASN.1 es objeto de un segundo escaneado para generar el código de programación C++ deseado. Este es el método de procesamiento para crear principalmente el mecanismo de inserción de mensajes.

1. La clase básica DSET ha incluido información de descripción de clase que comprende la función de creación del tipo correspondiente y el tipo es D_AdType y cada tipo corresponde a una variable de descripción de clase global D_AdType. Puesto que la información de descripción de algunas estructuras de C++ generadas, tales como ENUM, no está completa, necesita otros métodos para obtener la tabla correspondiente de la cadena de nombre de símbolos y el valor de número.
2. Las manipulaciones básicas para el ejemplo de clase C++ han sido ya realizadas y la información de descripción de clase puede obtenerse en el ejemplo de clase por intermedio de `const D_AdType * get_adType()` pero no soporta completamente la ruta de marcador de punto y no tiene ninguna función uniforme para obtener y establecer el valor de datos; una función de soporte de ruta de marcador de punto se añade y estas dos partes pueden ponerse en práctica en otros módulos fuera del mecanismo de inserción de mensajes.
3. Se establece la tabla de descripción de clase TOPCLASS, la plataforma de compilación DSET no soporta la tabla de descripción de clase y todas las clases TOPCLASS podrían obtenerse mediante un escaneado secundario de los ficheros ASN.1 o el escaneado del código de C++ generado para generar la tabla de descripción de clase. En MAKE FILE, la tabla de descripción de clase puede generarse mediante scripts con el escaneado de AWK y SED u otro escaneado programado.
4. Un DLL se genera a partir de los códigos en las etapas 1 a 3 y derivando la función de tabla de descripción de clase y de longitud de tabla de `AdTypeEntry*getAD_TypeTbl()`, con lo que todas las clases de TOPCLASS pueden derivarse cuando sea deseable y el fichero de cabecera de C++ puede regenerarse mediante un nuevo escaneado. De este modo, se convierte en un mecanismo de inserción cuya interfaz es `getAD_TypeTbl()` y la totalidad de la descripción de TOPCLASS puede obtenerse mediante esta estructura para crear y modificar sus ejemplos. La tabla de descripción de clase puede visualizarse y registrarse después de la carga del mecanismo de inserción. La etapa A en la Figura 2 se pone en práctica y las etapas 1 a 4 anteriores se ponen en el mecanismo de inserción de mensaje según se ilustra en la Figura 1. La parte de ENUM en la etapa 1 puede ponerse en práctica en el módulo de manipulación común.
5. `AsnTreeObject` puede definirse como:

```
class AsnTreeObject {
public:
    ansistring nodename;
    ansistring nodeVALue;
    int ansitype;
    ansistring typename;
    AbstractData *pParent;
    AbstractDataList childrenlist;
};
```

Por supuesto esta estructura puede añadir otros datos extras cuando sea deseable. Esta estructura es adecuada para una visualización transversal y su modificación correspondiente. Con el mismo método, se realizan las etapas D y E en la Figura 2.

6. El ejemplo de estructura de mensaje de ASN.1 C++ y los datos VAL son mutuamente generados y la función de datos VAL es una función miembro de `AbstgractData`; la función para generar un ejemplo de ASN.1 C++ a partir de los datos VAL puede ser una función independiente que utiliza los datos VAL como su parámetro, el punto de `AbstractData` como su parámetro y el código de error como el valor de retorno. Puesto que esta función no puede modificar el programa de la clase básica DSET, la función independiente sólo puede ponerse en práctica en otro módulo fuera del mecanismo de inserción. Las etapas C1 y C2 en la Figura 2 se realizan en este momento operativo. La etapa C.1 puede ponerse en práctica utilizando las operaciones en la etapa 1 y etapa 2 anteriores y la realización de la etapa C.2 necesita utilizar las operaciones en las etapas 1, 2 y 3 anteriores. Esta parte se pone en práctica en el mecanismo de inserción de mensajes que se representa en la Figura 1.
7. Para poder editar un mensaje en el editor de gráficos, los nombres de tipos de todas las estructuras de ASN.1 deben ser incluidas en la lista correspondiente. En primer lugar, la función de creación, a modo de ejemplo, se obtiene a partir de la tabla de descripción de clase en conformidad con el tipo de mensaje y luego, el `AsnTreeObject` en la etapa 5 se utiliza para visualizar el mensaje y el `AsnTreeObject` se edita en conformidad con la información de restricción de la descripción de clase de mensaje para generar, además, datos VAL y generar un ejemplo de ASN.1 C++ a partir de los datos VAL. La razón principal de generar un ejemplo de clase de C++ a partir de los datos VAL aunque no directamente desde `AsnTreeObject` es que resulta fácil generar

datos VAL y se basa menos en el sistema de estructura de clase básica de la plataforma de compilación y por ello, la plataforma de compilación tiene una buena transportabilidad. Con el método anterior, el mensaje puede crearse y modificarse también mediante script y memorizarse de forma persistente en datos VAL.

- 5 8. Con el desarrollo del formato XML, el XML del mensaje ASN.1 puede comprender toda la información del formato VAL, incluyendo el par de valor de nombre y del símbolo, el tipo de estructura de capas exterior, el tipo real de ANY. Por lo tanto, solamente los formatos de datos que incluyen la información de VAL pueden ser mutuamente convertidos con formato VAL sobre la base de la información de descripción del mecanismo de inserción. Además, un fichero XML puede sustituir a datos VAL.

10
15 Conviene señalar que los expertos en esta técnica pueden realizar varias modificaciones o variaciones que deben caer dentro del alcance de las reivindicaciones de la presente invención en conformidad con la ilustración de solución técnica y las formas de realización especificadas de la presente invención. A modo de ejemplo, con la aparición del formato XML, el XML del mensaje ASN.1 puede comprender toda la información del formato VAL, incluyendo el par de valor de nombre y del símbolo, el tipo de estructura de capa exterior y el tipo real de ANY. Por lo tanto, solamente los formatos de datos que incluyen la información de VAL pueden ser mutuamente convertidos con formato VAL sobre la base de la información de descripción del mecanismo de inserción. Además, un fichero XML puede sustituir a datos VAL en la patente.

20 La presente invención resuelve el problema de que el ejemplo de estructura de mensaje debe ser restablecido después de que el mensaje sea memorizado de forma persistente, supera el inconveniente de que el tipo de datos del flujo de bits debe ser conocido por anticipado para la decodificación y los datos ANY pueden decodificarse en un tipo de datos específico, los datos de miembros pueden modificarse también después de que se establezca el mensaje y la modificación puede realizarse con la forma de cadena, con lo que el ejemplo de estructura de script
25 puede crearse y modificarse mediante explicación, además, el mensaje ASN.1 que puede realizarse en un mecanismo de inserción y el método de generación de mensajes es independiente de la definición del mensaje, con el compilador de ASN.1 actual, soporta el editor de mensajes de ASN.1 de la interfaz gráfica y el script para acceso al mensaje ASN.1. Si se cambia la definición del mensaje, si solamente el mecanismo de inserción generado es reeditado mientras que el método de generación es el mismo, se reduce la dificultad de construcción del mensaje y
30 se disminuye, en gran medida, la carga de trabajo de desarrollo de mensajes y su mantenimiento.

Aplicabilidad industrial

La presente invención tiene las ventajas siguientes:

- 35 1. El mensaje ASN.1 puede memorizarse de forma persistente y los datos persistentemente memorizados pueden recuperarse como un ejemplo de estructura de datos y también se decodifican los datos de tipo ANY.
- 40 2. El mensaje puede aplicar la manipulación de símbolos de cadena de valor de nombre para modificar fácilmente el mensaje y soportar el lenguaje de script para establecer el mensaje ASN.1.
- 45 3. El método de generación de mensajes es independiente de la definición del mensaje y el cuerpo del mensaje ASN.1 puede aplicar la forma de inserción que se genera automáticamente mediante MAKEFILE con la compilación del fichero ASN.1, con lo que se modifica y actualiza fácilmente el mensaje y se reduce la carga de trabajo de mantenimiento.
- 50 4. El dispositivo es universal y no se basa en un tipo de estructura de ASN.1 especial o en un compilador de ASN.1.

50

55

REIVINDICACIONES

1. Un dispositivo de procesamiento orientado a objetos para un mensaje ASN.1, que comprende:

5 un módulo de soporte inferior ASN.1 para compilar el mensaje ASN.1 en un código de lenguaje de programación a modo de ejemplo;

10 un módulo de manipulación común para el procesamiento del mensaje para soporte de aplicaciones de mensaje; caracterizado por cuanto que el dispositivo comprende, además, un mecanismo de inserción de mensaje para generar una inserción de cuerpo de mensaje a partir del código generado al compilar el mensaje ASN.1, con una tabla de descripción de clase que desempeña la función de interfaz de inserción;

15 el módulo de manipulación común o el mecanismo de inserción de mensaje se utiliza para modificar el código de lenguaje de programación, a modo de ejemplo, generado por el módulo de soporte inferior de ASN.1, para utilizar las funciones virtuales para complementar la información de descripción de clase, incluyendo la información de descripción de estructura de clase y funciones para crear un ejemplo de clase; para añadir un soporte de manipulación de ejemplo de clase para el código generado;

20 para establecer una tabla de descripción de clase TopClass que incluye la relación correspondiente entre un nombre de TopClass y una descripción de clase en el código generado y la TopClass es una clase de capa superior pero no una clase hijo;

25 en donde el módulo de manipulación común o el mecanismo de inserción de mensaje se utiliza para recuperar un ejemplo de mensaje de lenguaje de programación a partir de datos en formato VAL.

2. Un dispositivo según la reivindicación 1, en donde dicho código de ejemplo de lenguaje de programación es un código de programación C++ o JAVA.

30 3. Un dispositivo según la reivindicación 1, en donde, para un tipo de datos simples y un tipo de datos compuestos, dicho módulo de manipulación común o dicho mecanismo de inserción de mensaje añade el nombre del tipo, el valor entero y la función de creación de ejemplo de clase del tipo de datos; para los tipos de datos SET y SEQUENCE, se añade un nombre de cada miembro, se establece si el tipo de datos es opcional y añade la tabla correspondiente de la información de descripción del miembro; para un tipo de datos ENUM, añade la tabla correspondiente entre una representación simbólica del miembro y un valor numérico.

35 4. Un dispositivo según la reivindicación 1, en donde dicho módulo de manipulación común o mecanismo de inserción de mensaje añade la tabla de descripción de clase en el código de ejemplo de lenguaje de programación, indica la información de descripción de clase bajo la forma de una tabla de descripción de clase, toma un nombre de clase como una palabra clave y clasifica los elementos en la tabla en conformidad con el nombre de clase o los transforma en una tabla tipo *hash* con el fin de poder buscarlos de forma conveniente.

5. Un dispositivo según la reivindicación 1, en donde:

45 para el tipo de datos simples, la adición del soporte de la manipulación de ejemplo de clase en el código de ejemplo de lenguaje de programación mediante el módulo de manipulación común o el mecanismo de inserción de mensaje, comprende:

50 la adición de un soporte de las funciones virtuales, de modo que pueda obtenerse un valor de datos y pueda establecerse el valor de datos;

para el tipo de datos compuestos, la adición del soporte de manipulación de ejemplo de clase en el código de ejemplo de lenguaje de programación mediante el código de manipulación común o el mecanismo de inserción de mensaje comprende:

55 a) la adición de un soporte de una ruta de marcador de punto;

b) la adición de un soporte del establecimiento de un símbolo que sirve para indicar si existen, o no, miembros en SEQUENCE y SET;

60 c) la adición de un soporte de la obtención de una longitud de una red matricial de datos, estableciendo la longitud de datos y elementos transversales en la red de datos para una clase de red de datos SEQUENCE OF y una clase de red de datos SET OF;

65 d) la adición de un estado de una operación de selección para un tipo de datos CHOICE;

e) la adición de un estado de establecimiento de un tipo de datos ANY, que puede establecer u obtener una variable

de un tipo real;

f) la adición de un soporte de obtención de un objeto del nodo hijo correspondiente a través de la ruta de marcador de punto.

5 **6.** Un dispositivo según una cualquiera de las reivindicaciones 1 a 5, en donde el módulo de manipulación común utiliza una lista de enlaces de arborescencia *AnsTreeObject* para indicar una estructura de mensaje ASN.1.

10 **7.** Un dispositivo según la reivindicación 1, en donde el módulo de manipulación común o el mecanismo de inserción de mensaje se utiliza para generar datos en el formato VAL a partir del código generado.

8. Un dispositivo según la reivindicación 7, en donde el módulo de manipulación común o el mecanismo de inserción de mensaje se utiliza para convertir los datos en el formato VAL en datos en el formato XML.

15 **9.** Un dispositivo según la reivindicación 6, en donde dicho dispositivo de procesamiento orientado al objeto para un mensaje ASN.1 comprende también un paquete de interfaz gráfica y un módulo de edición de mensaje gráfico,

en donde dicho paquete de interfaz gráfica se utiliza para desarrollar una interfaz gráfica;

20 utilizando dicho módulo de edición de mensaje gráfico el paquete gráfico y el módulo de manipulación común para proporcionar a un usuario una función de edición de mensaje de interfaz gráfica con el fin de modificar, crear y editar el mensaje ASN.1 por intermedio de la modificación de la lista de enlaces de arborescencia *AsnTreeObject*.

25 **10.** Un dispositivo según la reivindicación 6, en donde dicho dispositivo de procesamiento orientado a objetos para un mensaje ASN.1 comprende también un paquete de soporte del denominado *script* y un módulo de manipulación de script de mensaje,

en donde dicho paquete de soporte de script se utiliza para la ejecución de un script y para soportar una extensión de script;

30 dicho módulo de manipulación de script de mensaje utiliza el paquete de soporte de script y el módulo de manipulación común para añadir una función de manipulación de mensaje ASN.1 en el script para modificar, crear y editar el mensaje ASN.1 por intermedio de la modificación de la lista de enlaces de arborescencia *AsnTreeObject*.

35

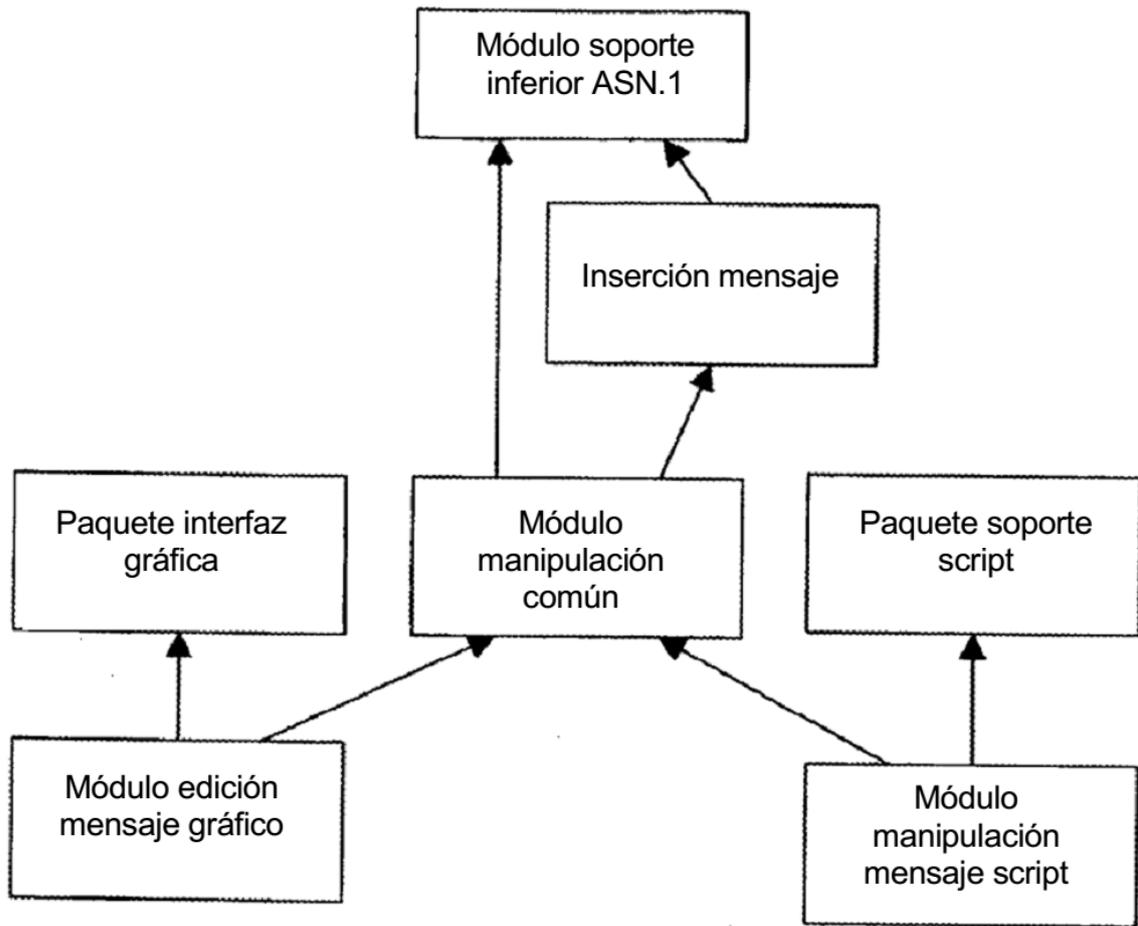


FIG. 1

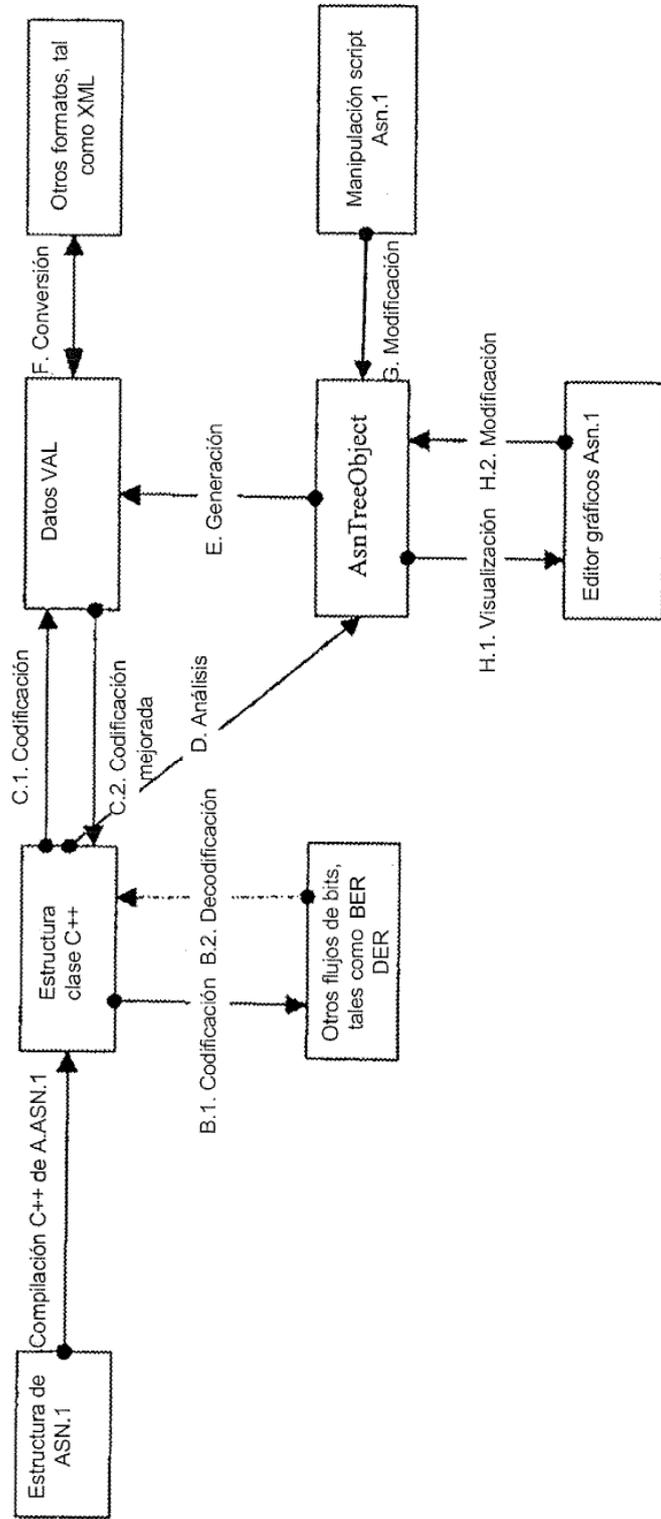


FIG. 2

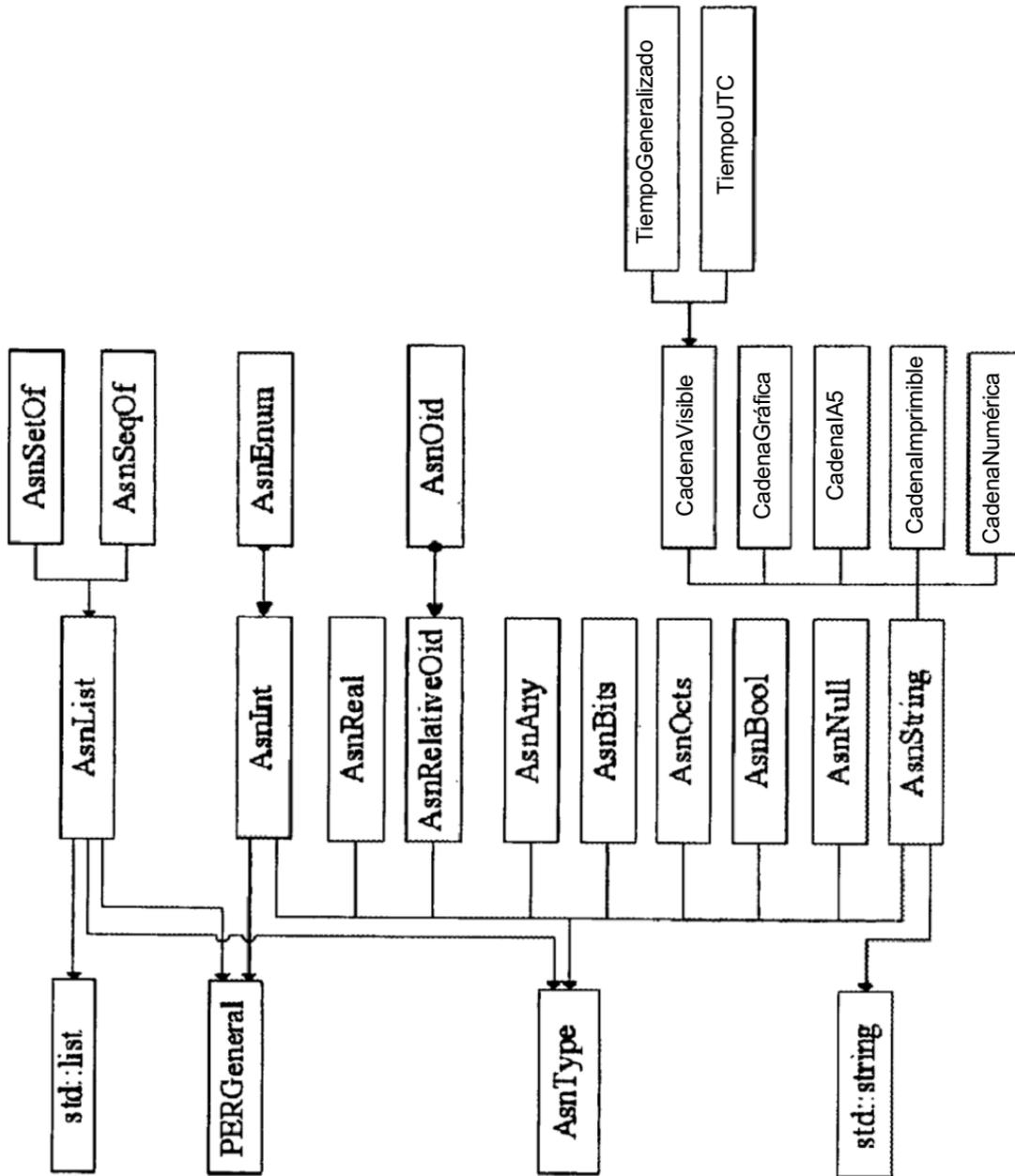


FIG. 3

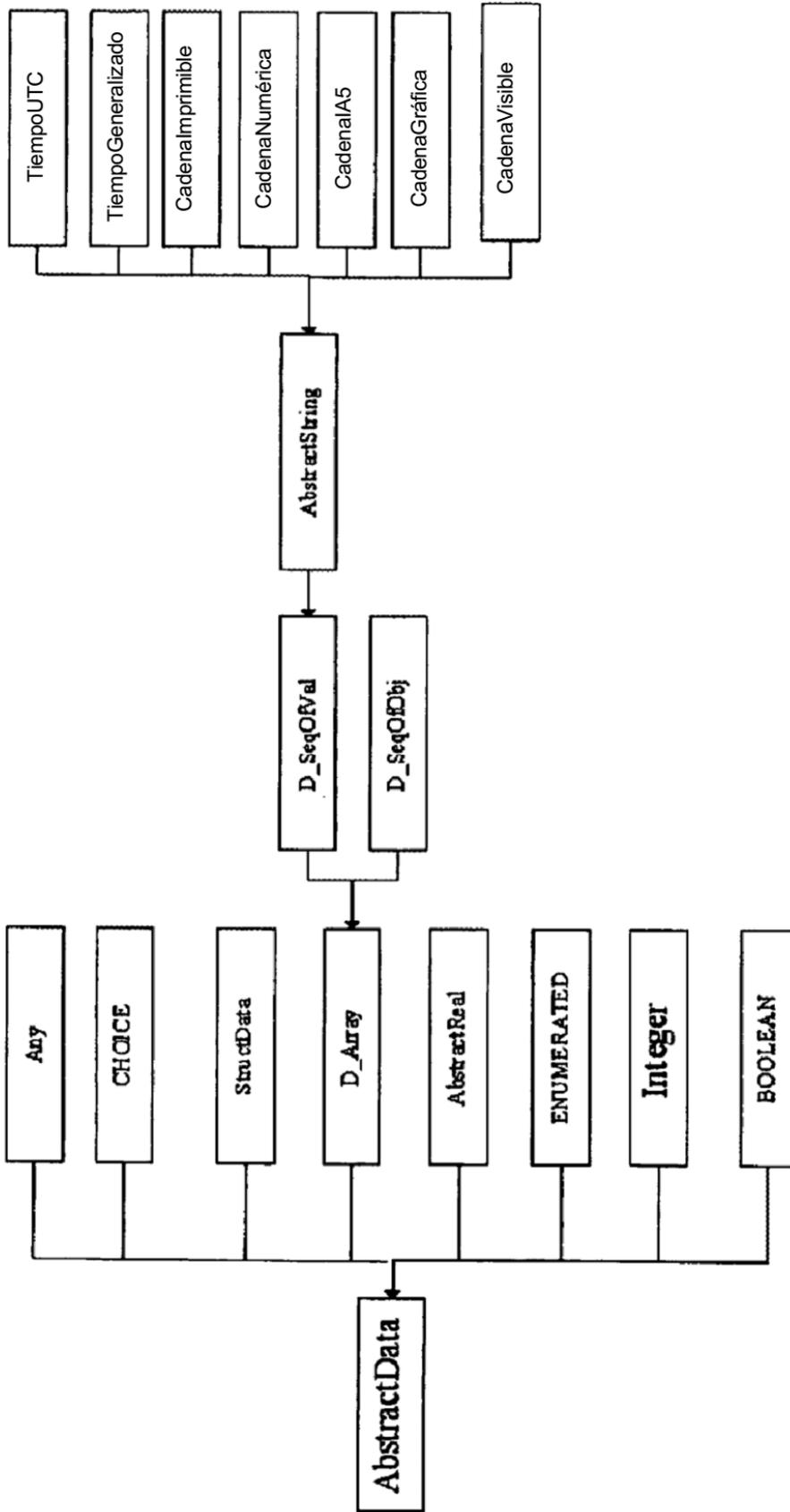


FIG. 4