

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 769 273**

51 Int. Cl.:

G06F 11/36 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **11.02.2010 E 10153337 (0)**

97 Fecha y número de publicación de la concesión europea: **27.11.2019 EP 2219113**

54 Título: **Procedimiento de visualización, dispositivo y producto programa de ordenador correspondiente**

30 Prioridad:

11.02.2009 FR 0950863

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

25.06.2020

73 Titular/es:

**INGENICO GROUP (100.0%)
28-32 Boulevard de Grenelle
75015 Paris, FR**

72 Inventor/es:

**NACCACHE, DAVID y
POLECHTCHOUK, PAVEL**

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 769 273 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

Procedimiento de visualización, dispositivo y producto programa de ordenador correspondiente

1. Ámbito de la invención

La presente invención se refiere al ámbito de la creación de aplicaciones informáticas.

- 5 La presente invención se refiere más concretamente a la corrección de tales aplicaciones informáticas durante operaciones de pruebas funcionales o unitarias o incluso durante operaciones de mantenimiento correctivo.

El proceso de creación de aplicaciones informáticas, tales como aplicaciones de tipo aplicaciones «web», pasa por numerosas etapas: la redacción de un pliego de condiciones, la redacción de especificaciones, de documentos de prueba, la creación de la aplicación propiamente dicha así como de las fases de prueba y de corrección aplicativa.
10 Una vez realizadas todas estas etapas, la aplicación está lista para ser puesta en producción. Posteriormente a esta producción, pueden tener lugar operaciones de mantenimiento correctivo.

Las etapas de prueba y de mantenimiento necesitan frecuentes actualizaciones del código de la aplicación. En el caso de una aplicación de tipo «web» estas actualizaciones se hacen cada vez más complejas debido a que generalmente, una aplicación de tipo «web» hace intervenir numerosas capas de software; una capa de acceso a la base de datos, una capa denominada de «oficio» que está encargada de funcionalidades propias del oficio en relación con la aplicación y una capa de presentación que está encargada de visualizar los datos en la pantalla. Otras capas pueden intervenir todavía y complicar la aplicación.
15

2. Soluciones de la técnica anterior

Existen soluciones para facilitar la corrección de los errores presentes en las aplicaciones informáticas. Estas soluciones se ponen en práctica casi siempre a nivel de las IDE («Interfaz de desarrollo»). Tales IDE se presentan casi siempre en forma de una pantalla en la cual es posible seleccionar componentes que están integrados en la aplicación, o directamente del código fuente. La selección de uno u otro de estos elementos de la aplicación, conduce a la aparición de un conjunto de propiedades relacionadas con el elemento seleccionado.
20

Por ejemplo, cuando el elemento seleccionado es un componente de acceso a una base de datos que ha sido depositada en una interfaz de usuario de una aplicación que haya que crear, las propiedades pueden ser: emplazamiento de la fuente de datos, tipo de fuente de datos, protocolo de acceso a la fuente de datos, etc.
25

Tal enfoque por «componente» es puesto en práctica en las IDE de «CodeGear™», como por ejemplo «Delphi™». La selección de un componente implica la aparición de las propiedades de este último en un lugar específico de la pantalla.

En el marco del desarrollo de aplicaciones Web por ejemplo en Java, en PHP o directamente en HTML, se utiliza muy frecuentemente la IDE «Eclipse™». Aquí también, la selección de un trozo de código fuente (por ejemplo un objeto en java) o incluso la apertura de un archivo de código fuente provoca la apertura o la aparición de una ventana que contiene un conjunto de propiedades. Esta ventana es denominada «outline».
30

Otras implementaciones de este principio son visibles igualmente en la aplicación «Dreamweaver™» de «Adobe™». Cuando un usuario selecciona una parte o un elemento de una página HTML en curso de edición, el editor (la IDE) visualiza en una ventana elementos que representan propiedades asociadas a la etiqueta que es seleccionada: por ejemplo si el texto seleccionado por el usuario está enmarcado en una etiqueta <TD></TD> que representa un caso de una tabla, las propiedades asociadas a esta etiqueta serán visualizadas en la ventana de las propiedades del elemento seleccionado: el usuario puede entonces modificar los valores de las propiedades asociadas a esta etiqueta: el color del fondo de la casilla de la tabla, la alineación del texto en la casilla, el color y el tipo de letra del texto, las eventuales acciones que haya que realizar.
35
40

Naturalmente, el usuario se enfrenta a un problema: en el ejemplo precedente, estando el texto rodeado por la etiqueta <TD></TD>, solo las propiedades de esta etiqueta están presentes en la ventana de modificación. Sin embargo, esta etiqueta <TD></TD> está a su vez rodeada por una etiqueta <TR></TR>, que a su vez está rodeada por una etiqueta <TABLA></TABLA>. Ahora bien, en el caso del lenguaje HTML, como en el caso de otros muchos lenguajes, es posible que varias propiedades de otros elementos interfieran sobre el elemento seleccionado. En el caso precedente, si se considera la propiedad «Backgroundcolor», esta propiedad es modificable en cada una de las etiquetas <TD></TD>, <TR></TR> y <TABLA></TABLA>.
45

Así, el usuario que seleccione un texto situado entre las etiquetas <TD></TD> y que desee modificar el valor de la propiedad «Backgroundcolor» de rojo a azul, debe, si el valor de la propiedad no está definido a nivel de la etiqueta <TD></TD>, seleccionar sucesivamente las etiquetas <TR></TR> y <TABLA></TABLA> para identificar la etiqueta en la que se haya definido el valor de la propiedad «Backgroundcolor» y para modificar este valor.
50

Este ejemplo, no limitativo pero muy demostrativo y por cierto bastante simple, ilustra las dificultades encontradas por el desarrollador de aplicaciones de software: las propiedades de los elementos de código pueden estar definidas

generalmente en múltiples lugares del código fuente de la aplicación, lo que no facilita el mantenimiento o la corrección de la aplicación en cuestión puesto que esto obliga al desarrollador a navegar en múltiples porciones de código fuente en la búsqueda del elemento en el que se desean modificar ciertas propiedades.

5 El documento XP002538353 concierne a los servicios de alojamiento Dreamweaver, que permiten a los usuarios crear y descargar fácilmente páginas Web.

El documento US2005204344 A1 concierne a una programación orientada a aspectos (AOP) y en particular, a un sistema y a un procedimiento de análisis de un programa y de salida de un resultado de análisis del mismo.

3. Resumen de la invención

La invención no presenta estos inconvenientes.

10 En efecto, la invención concierne a un procedimiento de visualización de propiedades de un elemento de un código fuente de un programa informático en curso de edición.

Según la invención, dicho procedimiento comprende:

- una etapa de creación de un árbol sintáctico de jerarquización del citado código fuente, que comprende para cada elemento del citado código fuente, al menos un elemento padre;

15 - una etapa de memorización del citado árbol sintáctico;

- una etapa de selección, en el seno del citado código fuente, del citado elemento de código fuente;

20 - una etapa de búsqueda, en el seno del citado código fuente, de un conjunto de elementos de código relacionados con el citado elemento de código fuente seleccionado, comprendiendo los citados elementos del citado conjunto propiedades con nombre, comprendiendo la citada etapa de búsqueda una etapa de identificación, en el seno del árbol sintáctico memorizado, del citado al menos un elemento padre del citado elemento seleccionado;

- una etapa de creación de una estructura de datos que comprenden las citadas propiedades de los citados elementos de código del citado conjunto de elementos de código;

- una etapa de reagrupamiento, en el seno de la citada estructura de datos, de las citadas propiedades en función de sus nombres;

25 - una etapa de visualización de las citadas propiedades reagrupadas en el seno de una porción de un medio de visualización del citado código fuente.

De esta manera, al reagrupar en el seno de una sola y misma porción de visualización (por ejemplo una ventana específica o un pop up particular) todas las propiedades de otros elementos que pueden ser utilizadas para influir sobre el elemento de código fuente previamente seleccionado por el usuario, la invención facilita la corrección y el

30 mantenimiento del código fuente en cuestión evitando numerosas manipulaciones al usuario.

Según un modo de realización particular de la invención, el citado procedimiento de visualización comprende además, previamente a la citada etapa de reagrupamiento, una etapa de supresión, en el seno de la citada estructura de datos, de las propiedades que no pueden ser utilizadas para influir sobre el citado elemento de código fuente previamente seleccionado por el citado usuario.

35 De esta manera, la invención evita inducir al usuario a error al presentar, en el seno de la porción de visualización, propiedades que no pueden ser utilizadas para influir sobre el elemento seleccionado por el usuario.

Según una característica particular de la invención, el citado procedimiento de visualización comprende además, previamente a la citada etapa de reagrupamiento, una etapa de supresión, en el seno de la citada estructura de datos, de las propiedades que no comprenden valores.

40 De esta manera, se visualizan solo las propiedades que han sido ya valoradas (es decir las propiedades que ya han recibido un valor). Se evita así presentar demasiadas informaciones al usuario y se facilita el mantenimiento del programa informático.

La invención concierne igualmente a un sistema de visualización de propiedades de un elemento de código fuente de un programa informático en curso de edición.

45 Según la invención, tal sistema comprende:

- medios de creación de un árbol sintáctico de jerarquización del citado código fuente, que comprenden para cada elemento de citado código fuente, al menos un elemento padre;

- medios de memorización del citado árbol sintáctico;

- medios de selección, en el seno del citado código fuente, del citado elemento de código fuente;

- medios de búsqueda, en el seno del citado código fuente, de un conjunto de elementos de código relacionados con el citado elemento de código fuente seleccionado, que comprenden medios de identificación, en el seno del citado árbol sintáctico memorizado, del citado al menos un elemento padre del citado elemento seleccionado;

5 - medios de creación de una estructura de datos que comprenden propiedades de los citados elementos de código del citado conjunto de elementos de código;

- medios de reagrupamiento, en el seno de la citada estructura de datos, de las citadas propiedades en función de sus nombres, comprendiendo los citados elementos del citado conjunto propiedades con nombre;

10 - medios de visualización de las citadas propiedades reagrupadas en el seno de una porción de un medio de visualización del citado código fuente.

Según otro aspecto, la invención concierne igualmente a un producto programa de ordenador descargable desde una red de comunicación y/o almacenado en un soporte legible por ordenador y/o ejecutable por un microprocesador, y que comprende instrucciones de código de programa para la ejecución del procedimiento de visualización tal como el descrito anteriormente.

15 **4. Lista de las figuras**

Otras características y ventajas de la invención se podrán de manifiesto de modo más claro en la lectura de la descripción que sigue de un modo de realización preferente, dado a modo de simple ejemplo ilustrativo y no limitativo, y de los dibujos anejos, en los cuales:

- la figura 1 presenta la visualización producida por un navegador web a partir de un código HTML;

20 - la figura 2 ilustra una interfaz de una aplicación de desarrollo de software en cuyo seno, según un modo de realización de la invención, se visualizan las propiedades de los elementos;

- la figura 3 ilustra un modo de realización del procedimiento de la invención.

5. Descripción detallada de la invención

5.1 Recordatorio del principio de la invención

25 Se aclara que en el marco de la presente invención, un elemento puede ser cualquier parte de código, tal como un control visual o un bloque de código fuente, que es seleccionable en una interfaz hombre máquina de una aplicación de desarrollo de software.

30 En al menos un modo de realización, la invención comprende medios para reagrupar propiedades que están asociadas a una porción de código seleccionada por el usuario (tal como un elemento de código particular). Una propiedad es un Atributo con nombre de un objeto, de un control o de un elemento (denominado elemento en lo que sigue). Las propiedades definen las características del objeto tales como el tamaño, el color, el emplazamiento en la pantalla o el estado de un objeto (por ejemplo activado o desactivado). Las propiedades están tipificadas: el tamaño es un tipo de propiedad, de la misma manera que el color, el emplazamiento, la base de datos a la cual está conectado el objeto, etc. Las propiedades pueden ser vírgenes (las mismas no comprenden valor) o valoradas (las mismas tienen un valor).

35 Por ejemplo, en el caso de la edición de un archivo que comprende un código con formato HTML, la invención permite agrupar en un solo lugar, fácilmente accesible al desarrollador, las propiedades de un mismo tipo que se refieren a un elemento de código fuente, seleccionado.

40 Así, el desarrollador, ve a simple vista dónde está definido el valor del código fuente que el mismo desea modificar. Por ejemplo, la propiedad «Backgroundcolor» que es modificable en numerosos elementos de un código fuente HTML será visualizada en un solo y mismo lugar (en una porción de ventana específica), tantas veces como etiquetas haya susceptibles de poder modificar esta propiedad para el elemento de código fuente seleccionado. El valor de esta propiedad puede ser por ejemplo «black».

45 Más concretamente, en el marco del ejemplo precedente, si el elemento de código fuente seleccionado por el usuario es un texto en el interior de una etiqueta <TD></TD>, al menos se visualizarán tres propiedades «Backgroundcolor» en la ventana o la porción de ventana considerada: una propiedad «Backgroundcolor» en relación con cada etiqueta <TD></TD>, <TR></TR> y <TABLA></TABLA>.

Así, el usuario tiene rápidamente acceso a las propiedades sin tener que navegar de una etiqueta o de un conjunto de etiquetas a otro y puede fácilmente señalar la etiqueta en la cual se ha definido el valor de la propiedad que el mismo desea modificar.

De esta manera se facilita mucho la corrección o el mantenimiento de la aplicación de software. La invención va a contracorriente de las ideas del experto en la materia. En efecto, el lugar de reagrupar de manera racional las propiedades por elemento, la invención permite reagrupar las propiedades por tipo, independientemente de los elementos a los cuales pertenecen estas propiedades. Este tipo de reagrupamiento es muy útil para las operaciones de corrección o de mantenimiento aplicativo y solo puede ser puesto en práctica en el seno de una aplicación de software de desarrollo cuando el programa desarrollado por el usuario está en curso de mantenimiento por ejemplo.

En relación con la figura 1, se presenta una interfaz gráfica clásica (10) de una aplicación de desarrollo de aplicaciones de software. Dicha interfaz comprende una arborescencia de archivos (101), una ventana de edición de código (102), una ventana de pre-visualización o de presentación de resultados de compilación o de búsqueda (103) y una ventana relativa al código en curso de edición (104). Esta última ventana (104) puede comprender propiedades relacionadas con el objeto o el elemento en curso de edición.

Los ejemplos precedentes se han presentado solamente a modo puramente ilustrativo. La invención no se limita en absoluto a este ejemplo particular. En efecto, el procedimiento de la invención puede ser puesto en práctica igualmente en el marco de numerosos tipos de aplicaciones de software de desarrollo tales como las aplicaciones «Eclipse™», «Dreamweaver™», «Delphi™» y para otros numerosos lenguajes de desarrollo de aplicaciones de software.

En lo que sigue, se presenta en particular el caso de un modo de realización de la invención para el código HTML.

5.2 Descripción de un modo de realización

En este modo de realización se presenta la puesta en práctica de la invención con el lenguaje de etiquetas HTML. El lenguaje HTML es un lenguaje que permite crear páginas que son visualizadas por un navegador Web. El lenguaje HTML es un lenguaje con etiquetas porque el mismo comprende instrucciones que influyen en el motor de procesado de páginas HTML del navegador Web y porque estas instrucciones se presentan en forma de etiquetas que rodean a los elementos definidos en la página HTML.

En relación con la figura 2 se presenta una porción de código HTML (10) y el resultado (11) de la interpretación (12) de este código HTML por un navegador Web, es decir la visualización correspondiente.

El código HTML comprende numerosas etiquetas como por ejemplo las etiquetas <H1>, <P> y .

Estas etiquetas sirven para indicar al navegador web la manera en que debe visualizar el contenido de la página. La utilización del lenguaje HTML va acompañada casi sistemáticamente de la utilización de hojas de estilo (denominadas en inglés «CSS» de «Cascading Style Sheets»).

Estas hojas de estilo (y el lenguaje informático que se utiliza para construir estas hojas de estilo) sirven para describir la presentación de los documentos HTML y XML. La ventaja de las hojas de estilo es poder teóricamente reagrupar en el seno de un solo archivo (la hoja de estilo) todas las elecciones de presentación y de decoración de los elementos HTML que componen la página. Las mismas por tanto contienen también propiedades relacionadas con los elementos de código HTML. Las hojas de estilo se utilizan generalmente durante la conversión de archivos al formato XML para generar código HTML.

Sin embargo, en la utilización, los inventores son conscientes de que las hojas de estilo rara vez se utilizan de principio a fin en el desarrollo de una aplicación Web que facilita código HTML al navegador.

En efecto, la utilización de una hoja de estilo se utiliza con relativa frecuencia durante la creación de la aplicación: comenzando con buenas intenciones, los desarrolladores están bastante dispuestos a separar de manera eficaz la presentación (las hojas de estilo) y los datos. Durante el mantenimiento o la corrección del código, en cambio, la utilización de las hojas de estilo es menos frecuente: sucede frecuentemente que la persona que ha creado la aplicación no sea la misma que la que realiza su mantenimiento correctivo y que con este cambio haya desaparecido un cierto conocimiento de la aplicación.

En este modo de realización de la invención, se propone por tanto reagrupar las propiedades de un mismo tipo, incluso las propiedades que provienen de una hoja de estilo, a nivel de una misma porción de pantalla o de una misma ventana en la pantalla. Este reagrupamiento se realiza cuando el usuario selecciona un elemento, o una porción de elemento. Esta selección puede ser realizada directamente en el código fuente o incluso en un modo de presentación denominado «WYSIWYG» (del inglés de «What You See Is What You Get»).

En relación con la figura 3 se presentan las etapas generales del procedimiento de la invención. El mismo comprende:

- una etapa de creación de un árbol sintáctico de jerarquización del citado código fuente, que comprende para cada elemento del citado código fuente, al menos un elemento padre;

- una etapa de memorización del citado árbol sintáctico;

- una etapa de selección (31), en el seno del código fuente, del elemento de código fuente 311. Esta selección es realizada por el usuario.

- una etapa de búsqueda (32), en el seno del código fuente 321, de un conjunto de elementos de código relacionados con el citado elemento de código fuente seleccionado, comprendiendo los citados elementos del citado conjunto propiedades con nombre, comprendiendo la citada etapa de búsqueda una etapa de identificación, en el seno del citado árbol sintáctico memorizado, del citado al menos un elemento padre del citado elemento seleccionado;
 - 5 - una etapa de creación (33) de una estructura de datos 331 que comprenden propiedades de los elementos de código del conjunto de elementos de código;
 - una etapa de reagrupamiento (34), en el seno de la citada estructura de datos, de las citadas propiedades en función de sus nombres;
 - 10 - una etapa de visualización (35) de las citadas propiedades reagrupadas en el seno de una porción de un medio de visualización del código fuente 351.
- En una primera variante de este modo de realización de la invención, la selección de un elemento o de una porción de elemento conduce en un primer tiempo a la visualización, en forma de una lista, de los elementos que influyen sobre las propiedades del elemento seleccionado. Tras esta visualización el usuario es capaz de seleccionar, en la lista, el elemento que le interese. En un segundo tiempo, se visualizan por ejemplo las propiedades del elemento seleccionado, con la posibilidad del usuario de volver hacia atrás, es decir a la lista de los elementos.
- 15 En una segunda variante de este modo de realización de la invención, la selección de un elemento o de una porción de elemento conduce a la elaboración de una lista de propiedades, que son visualizadas. Esta lista reagrupa las propiedades por tipo: es así posible tener varias veces la misma propiedad sucesivamente en esta lista. El origen de la propiedad (el elemento conexo al cual pertenece la misma) se indica por un medio idóneo, tal como un icono, una burbuja de información o cualquier otro medio, para que el usuario no se vea desconcertado o perdido por esta visualización.
- 20 El usuario tiene así la posibilidad de ver enseguida, a simple vista, cuál es la propiedad que debe ser modificada y no tiene necesidad de seleccionar sucesivamente todos los elementos conexos para identificar aquél cuya propiedad debe ser modificada.
- 25 Según otra variante de este modo de realización de la invención, solo se visualizan las propiedades efectivamente utilizadas. Así, posteriormente a la eliminación de las propiedades que no influyen sobre el elemento seleccionado, el procedimiento de la invención realiza una nueva selección para visualizar solamente las propiedades que están valoradas: así, cuando una sola propiedad de un solo elemento está valorada, la misma es la única en ser visualizada. Cuando varias propiedades del mismo tipo están valoradas, estas son las únicas visualizadas.
- 30 De esta manera, el usuario está seguro de respetar las elecciones de implementación y de presentación que se han realizado anteriormente al no modificar una propiedad en lugar de otra.
- Naturalmente, es posible pasar del modo en que se visualizan todas las propiedades posibles al modo en que solo se visualizan las propiedades efectivamente utilizadas, siendo lo importante que se trate de las propiedades de los elementos conexos.
- 35 **5.3 Gestión del código fuente.**
- A nivel del código fuente propiamente dicho, la invención recorta el mismo en forma de un árbol sintáctico. El árbol queda almacenado en memoria durante la edición del código fuente. Para identificar los elementos que influyen sobre un elemento seleccionado por el usuario (los elementos conexos), la invención propone un método simple y rápido: los elementos seleccionados que influyen sobre el elemento son los padres, en el árbol, del elemento seleccionado.
- 40 Así, tomando el ejemplo de las etiquetas <TD></TD>, <TR></TR> y <TABLA></TABLA>, el padre del elemento <TD></TD> es el elemento <TR></TR> y el padre del elemento <TR></TR> es el elemento <TABLA></TABLA>. Se visualizan igualmente los eventuales estilos asociados a hojas de estilo en los elementos sucesivamente tenidos en cuenta.
- 45 Según otra variante de este modo de realización, solo se visualizan las propiedades que realmente pueden influir en el elemento o la porción de elemento seleccionado. De esta manera, no se visualizan las propiedades que no tienen influencia sobre el elemento seleccionado, para no sobrecargar la lista con propiedades inútiles.
- Para realizar esta selección de propiedades, son posible varios métodos, entre los cuales:
- la utilización de un criterio de similitud: se visualizan solo las propiedades de los elementos conexos que lleven el mismo nombre, o un nombre similar a las del elemento seleccionado (por ejemplo «Color» o «BackgroundColor»);
 - 50 - la utilización de una tabla de correspondencia que permita definir un esquema de influencia que comprenda relaciones entre diferentes propiedades de diferentes elementos.

Se recorre este esquema de influencia para determinar las propiedades que haya que visualizar en función del elemento seleccionado. Este esquema de influencia es determinado de antemano en función del lenguaje. El mismo puede ser representado en forma de una estructura de datos arborescente y/o ser implementado en una base de datos.

REIVINDICACIONES

- 5 1. Procedimiento de visualización de propiedades de un elemento de un código fuente de un programa informático en curso de edición, estando el citado programa escrito en un lenguaje de tipo objeto, tomando un elemento de código fuente la forma de un objeto, procedimiento puesto en práctica en el seno de una aplicación de desarrollo de aplicaciones de software, caracterizado por que el mismo comprende:
- una etapa de creación de un árbol sintáctico de jerarquización del citado código fuente, que comprende para cada elemento del citado código fuente, al menos un elemento padre;
 - una etapa de memorización del citado árbol sintáctico;
 - una etapa de selección (31), en el seno del citado código fuente, del citado elemento de código de fuente;
- 10 - una etapa de búsqueda (32), en el seno del citado código fuente, de un conjunto de elementos de código relacionados con el citado elemento de código fuente seleccionado, comprendiendo los citados elementos del citado conjunto propiedades con nombre, comprendiendo la citada etapa de búsqueda una etapa de identificación, en el seno del citado árbol sintáctico memorizado, del citado al menos un elemento padre del citado elemento seleccionado;
- 15 - una etapa de creación (33) de una estructura de datos que comprenden las citadas propiedades de los citados elementos de código del citado conjunto de elementos de código;
- una etapa de reagrupamiento (34), en el seno de la citada estructura de datos, de las citadas propiedades en función de sus nombres;
 - una etapa de visualización (35) de las citadas propiedades reagrupadas en el seno de una porción de un medio de visualización del citado código fuente.
- 20 2. Procedimiento de visualización según la reivindicación 1, caracterizado por que el mismo comprende además, previamente a la citada etapa de reagrupamiento, una etapa de supresión, en el seno de la citada estructura de datos, de las propiedades que no pueden ser utilizadas para influir sobre el citado elemento de código fuente, previamente seleccionado.
- 25 3. Procedimiento de visualización según la reivindicación 1, caracterizado por que el mismo comprende además, previamente a la citada etapa de reagrupamiento, una etapa de supresión, en el seno de la citada estructura de datos, de las propiedades que no comprenden valores.
- 30 4. Sistema de visualización de propiedades de un elemento de un código fuente de un programa informático en curso de edición, estando escrito el citado programa en un lenguaje de tipo objeto, tomando un elemento de código fuente la forma de un objeto, puesto en práctica en el seno de una aplicación de desarrollo de aplicaciones de software, caracterizado por que el mismo comprende:
- medios de creación de un árbol sintáctico de jerarquización del citado código fuente, que comprenden para cada elemento de código fuente, al menos un elemento padre;
 - medios de memorización del citado árbol sintáctico;
 - medios de selección, en el seno del citado código fuente, del citado elemento de código fuente;
- 35 - medios de búsqueda, en el seno del citado código fuente, de un conjunto de elementos de código relacionados con el citado elemento de código fuente seleccionado, que comprenden medios de identificación, en el seno del citado árbol sintáctico memorizado, del citado al menos un elemento padre del citado elemento seleccionado;
- medios de creación de una estructura de datos que comprenden propiedades de los citados elementos de código del citado conjunto de elementos de código;
- 40 - medios de reagrupamiento, en el seno de la citada estructura de datos, de las citadas propiedades en función de sus nombres, comprendiendo los citados elementos del citado conjunto propiedades con nombre;
- medios de visualización de las citadas propiedades reagrupadas en el seno de una porción de un medio de visualización del citado código fuente.
- 45 5. Procedimiento programa de ordenador descargable desde una red de comunicación y/o almacenado en un soporte legible por ordenador y/o ejecutable por un microprocesador, caracterizado por que el mismo comprende instrucciones de código de programa para la ejecución del procedimiento de visualización según al menos una de las reivindicaciones 1 a 3, cuando el mismo es ejecutado en un ordenador.

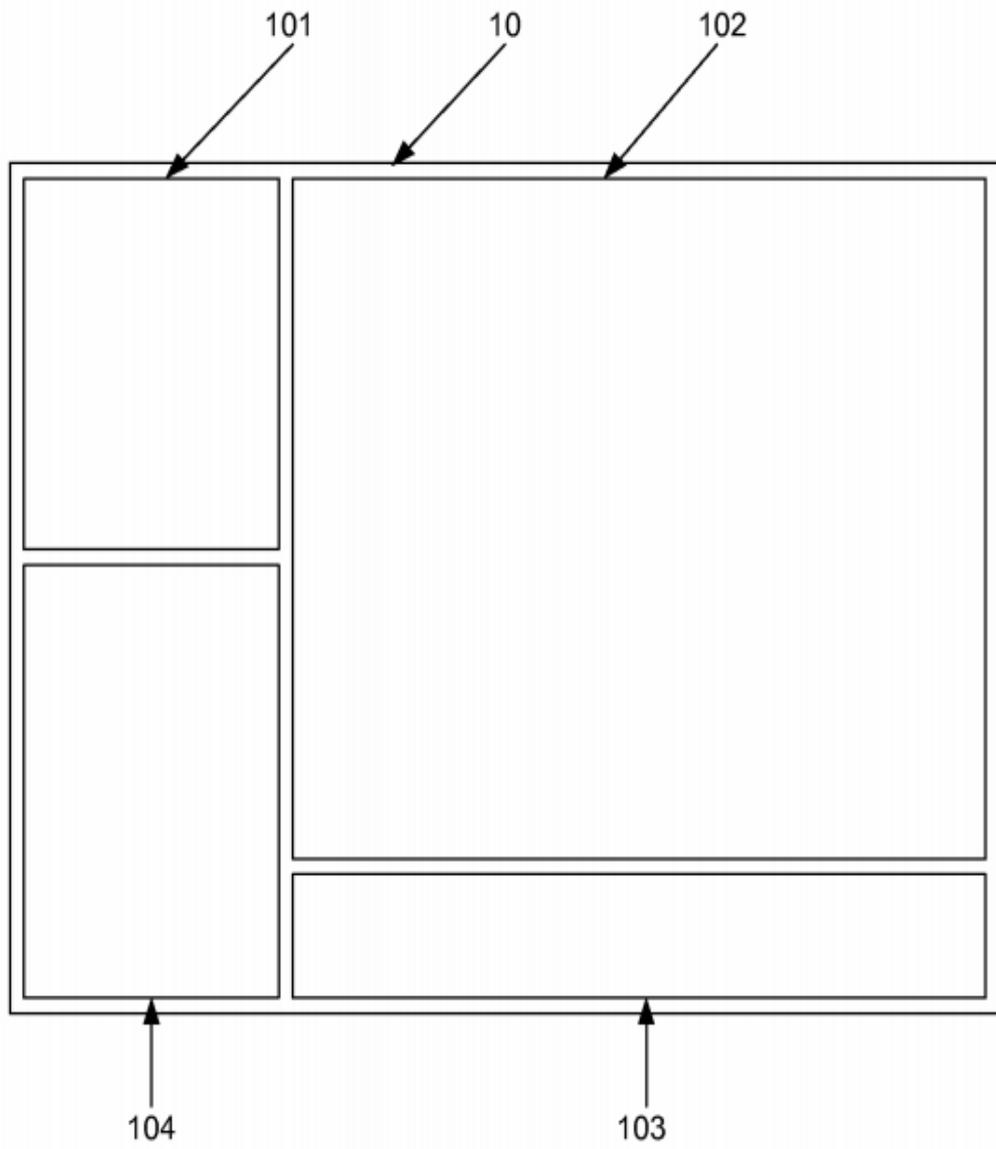


Figura 1

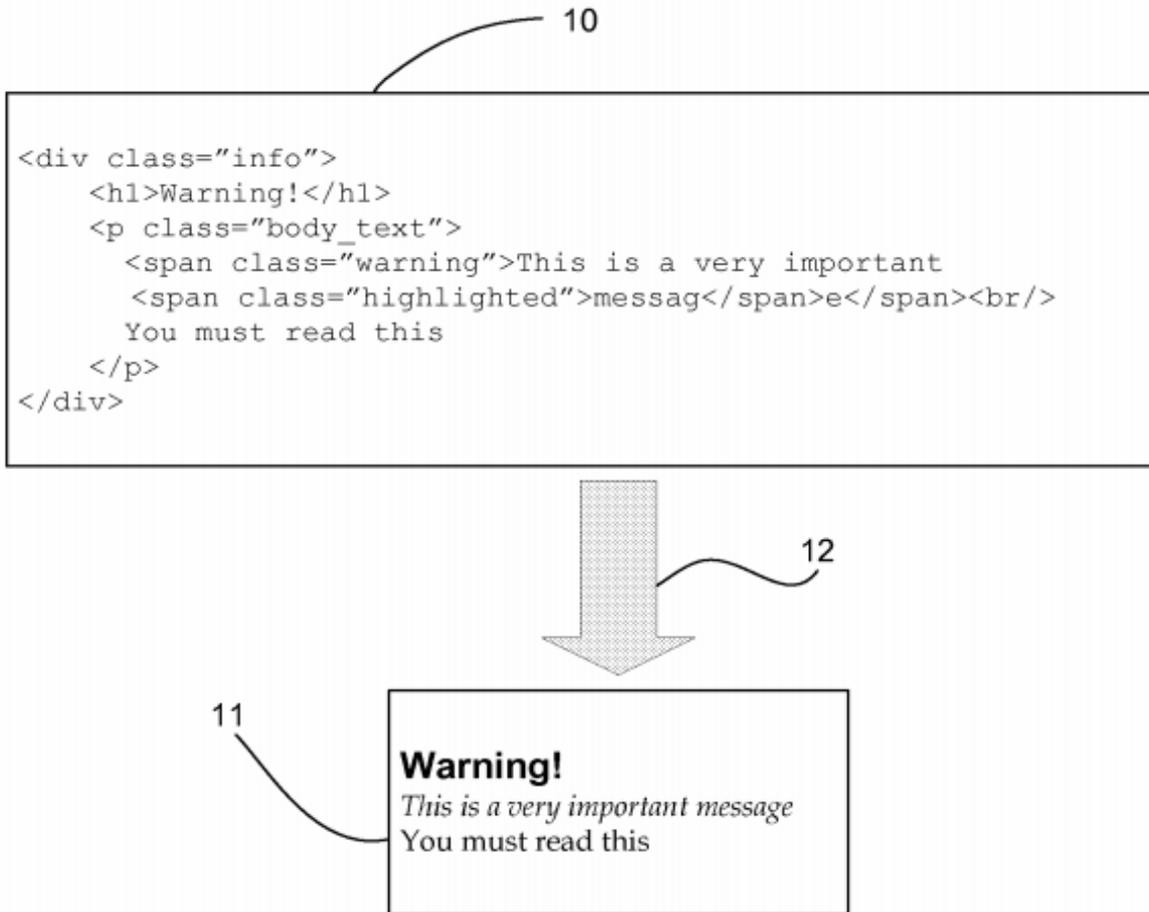


Figura 2

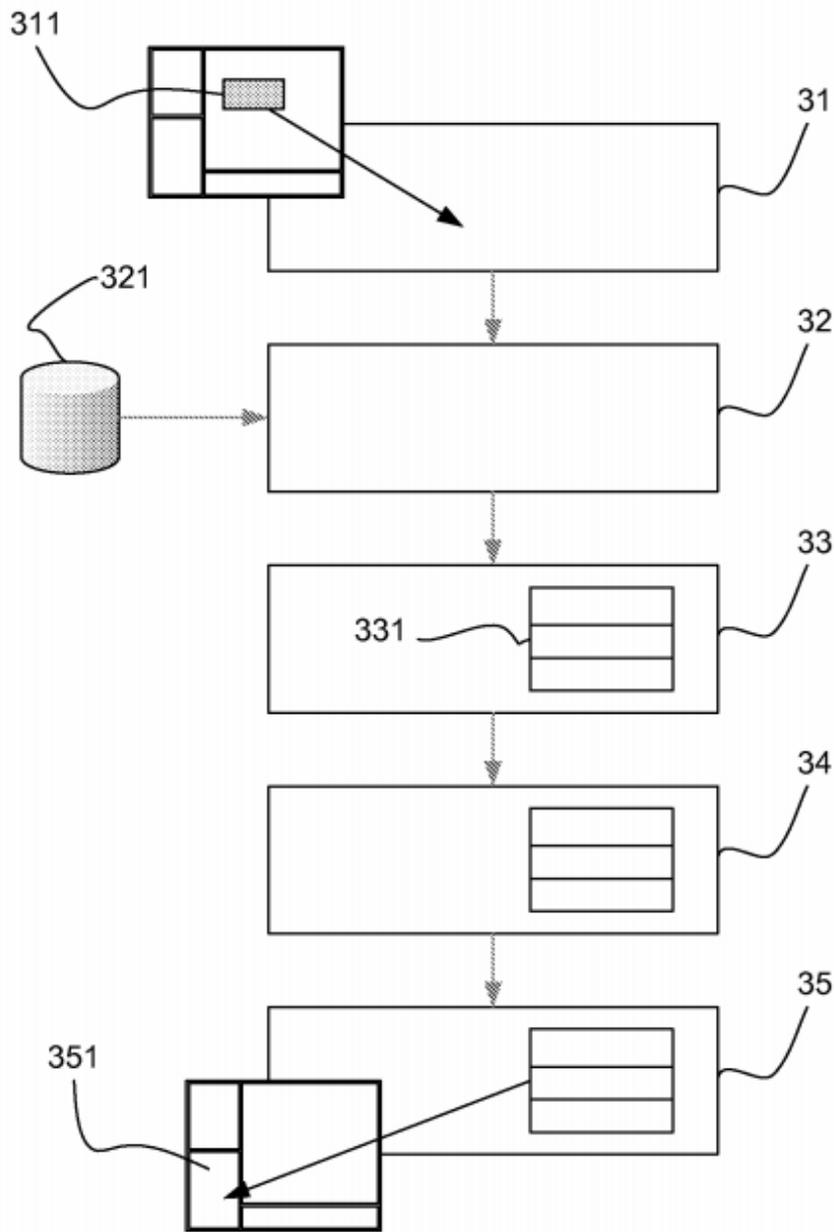


Figura 3