



República Federativa do Brasil  
Ministério do Desenvolvimento, Indústria  
e do Comércio Exterior  
Instituto Nacional da Propriedade Industrial.

(21) **PI1002208-2 A2**

(22) Data de Depósito: 11/02/2010  
(43) Data da Publicação: 26/07/2011  
(RPI 2116)



(51) *Int.Cl.:*  
G06F 7/06 2006.01

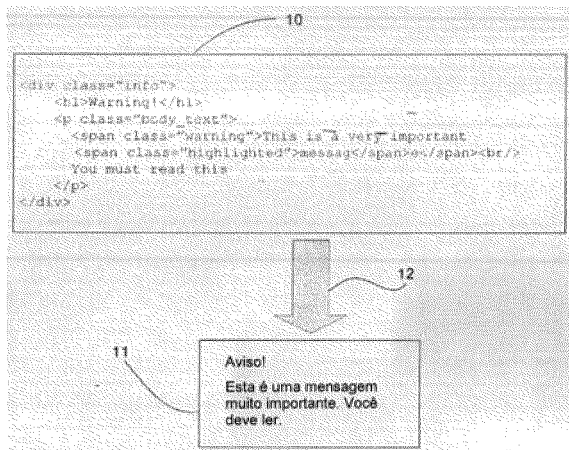
(54) Título: **MÉTODO PARA EXIBIÇÃO, PROGRAMA DE COMPUTADOR CORRESPONDENTE E DISPOSITIVO**

(30) Prioridade Unionista: 11/02/2009 FR 09/50863

(73) Titular(es): Compagnie Industrielle Et Financiere D'Ingenierie Ingenico

(72) Inventor(es): David Naccache, Pavel Polechtchouk

(57) Resumo: MÉTODO PARA EXIBIÇÃO, PROGRAMA DE COMPUTADOR CORRESPONDENTE E DISPOSITIVO. A invenção refere-se a um método para exibir as propriedades de um elemento de código fonte de um programa de computador sendo editado. De acordo com a invenção tal um método compreende: - uma etapa para selecionar, dentro do dito código fonte, o dito elemento de código fonte; - uma etapa para buscar, dentro do dito código fonte, um conjunto de elementos de código relacionados do dito elemento de código fonte selecionado, os elementos do conjunto que compreendem propriedades tipificadas; - uma etapa para criar uma estrutura de dados que compreende as propriedades dos elementos de código do conjunto de elementos de código; - uma etapa para agrupar, dentro da estrutura de dados, as propriedades em função de seus tipos; - uma etapa para exibir as propriedades agrupadas dentro de uma parte de meios de exibição do código fonte.





PI1002208-2

## "MÉTODO PARA EXIBIÇÃO, PROGRAMA DE COMPUTADOR CORRESPONDENTE E DISPOSITIVO"

### Campo da Invenção

A presente invenção refere-se a criação de aplicativos de computador.

5 A presente invenção refere-se, mais particularmente, a correção de tais aplicativos de computador durante operações funcionais ou teste de unidade ou mesmo durante operações de manutenção corretiva.

O processo de criação para aplicativos de computador, tais como aplicativos de "web" envolvem muitas etapas: o esboço de especificações, esboço de documentos de teste, criação do aplicativo-propriadamente dito bem como fases de correção de aplicativo e teste. Uma vez que todas essas etapas foram realizadas, o aplicativo está pronto para ser colocado em produção. Seguinte a este lançamento de produção, operações de manutenção corretiva podem ser realizadas.

10 Etapas de manutenção e teste requerem atualizações frequentes do código de aplicativo. Para um aplicativo do tipo "web", essas atualizações são feitas mesmo mais complexas devido ao fato de que frequentemente, um aplicativo do tipo "web" envolve muitas camadas de software; uma camada para acessar a um banco de dados, uma camada de "trabalho" que gerencia as funções específicas para o trabalho em relação ao aplicativo e uma camada de apresentação que gerencia a exibição dos dados na tela. Outras camadas também podem estar envolvidas e tornam o aplicativo mais complexo.

### Soluções da Técnica Anterior

15 As soluções existem para facilitar a correção de erros presentes em aplicativos de computador. Essas soluções são mais frequentemente implementadas no IDE (Ambiente de desenvolvimento integrado) Tais IDEs estão mais frequentemente na forma de um tela em que é possível selecionar componentes que são integrados no aplicativo, ou diretamente do código fonte. A seleção de um desses elementos do aplicativo leva a aparência de um conjunto de propriedades relacionadas ao elemento selecionado.

20 Por exemplo, quando o elemento selecionado é um componente para acessar a um banco de dados que foi colocado em uma interface de usuário de um aplicativo a ser criado, as propriedades podem ser: localização da fonte de dados, tipo de fonte de dados, protocolo para acessar a fonte de dados, etc.

Tal uma abordagem por "componente" é especialmente usada em IDEs de "comando de código", tais como, por exemplo, "Delphi". A seleção de um componente causa a aparência de suas propriedades em uma parte específica da tela.

35 No desenvolvimento de aplicativos de Web, por exemplo, Java, PHP ou diretamente em HTML, o IDE "eclipse" muito frequentemente usado. Mais uma vez, a seleção de uma parte de código fonte (por exemplo, um objeto Java), ou mesmo a abertura de um arquivo

de código fonte faz com que uma janela se abra ou apareça a qual contém um conjunto de propriedades. Esta janela é chamada de um “esboço”.

Outras implementações deste princípio podem também ser vistas no aplicativo “Adobe”, “Dreamweaver”. Quando um usuário seleciona uma parte ou um elemento de uma pagina de HTML sendo editado, o editor (o IDE) exibe em uma janela elementos que representam propriedades associadas ao indicador (tag) que é selecionado: por exemplo, se o texto selecionado pelo usuário for enquadrado em um indicador `<TD></TD>`, que representa uma célula de uma tabela, as propriedades associadas a este indicador irão ser exibidas na janela de propriedades do elemento selecionado: o usuário pode então modificar os valores das propriedades associadas a este indicador: a cor de fundo da célula da tabela, o alinhamento do texto na célula, a cor e a fonte do texto, quaisquer ações que devem ser realizadas.

Bastante frequente, o usuário é confrontado com um problema, no exemplo anterior, como o texto é rodeado pelo indicador `<TD></TD>`, apenas as propriedades deste indicador irão estar presentes na janela de modificação. Entretanto, este indicador `<TD></TD>`, é por si só rodeado por um indicador `<TR></TR>`, que é por si só rodeado por um indicador `<TABLE></TABLE>`, enquanto que no caso da linguagem HTML, como é o caso de muitas outras linguagens. É possível que muitas propriedades de outros elementos interfiram no elemento selecionado. No caso anterior, se a propriedade “cor de fundo” for considerada, esta propriedade pode ser modificada em um dos indicadores `<TD></TD>`, `<TR></TR>`, e `<TABLE></TABLE>`.

Conseqüentemente, o usuário que selecionou um texto localizado entre os indicadores `<TD></TD>`, e que deseja modificar o valor da propriedade “cor de fundo” do vermelho para o azul, deve, se o valor desta propriedade não for definido no indicador `<TD></TD>`, sucessivamente selecionar os indicadores `<TR></TR>`, para identificar o indicador para o qual o valor da propriedade “cor de fundo” foi definida para modificar este valor.

Este exemplo, que não é restritivo, mas é muito demonstrativo e bastante simples, ilustra as dificuldades encontradas pelos desenvolvedores de aplicativos de software: as propriedades dos elementos de código podem ser muito frequentemente definidas em vários lugares do código fonte, que não facilita a manutenção ou correção do aplicativo em questão como obriga o desenvolvedor a navegar em muitas partes do código fonte procurando o elemento no qual ele deseja modificar certas propriedades.

#### Sumário da Invenção

A invenção não possui essas desvantagens.

Na verdade, a invenção refere-se a um método para exibir as propriedades de um elemento de código fonte de um programa de computador sendo editado.

De acordo com a invenção, tal um método compreende:

- uma etapa para selecionar, dentro do dito código fonte, o dito elemento de código fonte;

- uma etapa para buscar, dentro do dito código fonte, um conjunto de elementos de código relacionados do dito elemento de código fonte selecionado, em que os elementos do conjunto compreendem propriedades que têm um tipo;

- uma etapa para criar uma estrutura de dados que compreende as propriedades dos elementos de código do conjunto de elementos de código;

- uma etapa para agrupar, dentro da estrutura de dados, as propriedades em função de seus tipos;

- uma etapa para exibir as propriedades agrupadas dentro de uma parte de meios de exibição.

Conseqüentemente, agrupando dentro de uma única e mesma parte (por exemplo, uma janela específica ou um pop up específico) todas as propriedades de outros elementos que podem ser usados para influenciar o elemento de código fonte anteriormente selecionado pelo usuário, a invenção facilita a correção e a manutenção do código fonte em questão evitando a necessidade de várias manutenções pelo usuário.

De acordo com a modalidade específica da invenção, o método de exibição adicionalmente compreende, antes da dita etapa de agrupamento, uma etapa para deletar, dentro da estrutura de dados, propriedades que não podem ser usadas para influenciar o elemento de código fonte anteriormente selecionado pelo usuário.

Conseqüentemente, a invenção evita levar o usuário a erros pela apresentação, dentro da parte de exibição, propriedades que não podem ser usadas para influenciar o elemento selecionado pelo usuário.

De acordo com uma característica específica da invenção, o método de exibição adicionalmente compreende, antes da etapa de agrupamento, uma etapa para deletar, dentro da estrutura de dados, propriedades que não têm valores.

Conseqüentemente, apenas as propriedades que já têm valores (isto é as propriedades que já receberam um valor) são exibidas. Isto, conseqüentemente evita exibição de muito mais informações para o usuário e a manutenção do programa de computador é assim facilitada.

De acordo com uma modalidade específica da invenção, o método de display, antes da dita etapa de seleção e durante a edição do código fonte:

- uma etapa para criar uma árvore de sintaxe hierárquica para o código fonte, incluindo para cada elemento do código fonte, pelo menos um elemento principal;

- uma etapa para memorizar a árvore de sintaxe.

Conseqüentemente, a invenção permite uma estrutura de dados hierárquica que é representativa do código fonte para ser mantida permanentemente até a data.

De acordo com uma característica específica da invenção, a etapa de busca compreende uma etapa de identificação, dentro da árvore de sintaxe memorizada, do pelo menos um elemento principal do elemento selecionado.

Consequentemente, a invenção permite uma rápida busca para ser feita para os elementos relacionados a um elemento selecionado.

A invenção também refere-se a um sistema para exibir as propriedades de um elemento de código de um programa de computador sendo editado.

De acordo com a invenção, tal um sistema compreende:

- meios para selecionar, dentro do código fonte, o elemento de código fonte;
- meios para buscar, dentro do código fonte, um conjunto de elementos de código relacionados do dito elemento de código fonte selecionado;
- meios para criar uma estrutura de dados compreendendo propriedades dos elementos de código do conjunto de elementos de código;
- meios para agrupar, dentro da estrutura de dados, as propriedades em função de seus tipos;
- meios para exibir as propriedades agrupadas dentro de uma parte de meios de exibição do código fonte.

De acordo com outro aspecto, a invenção também refere-se a um programa de computador que pode ter download realizado a partir de uma rede de comunicação e/ou armazenado em um suporte que pode ser lido por um computador e/ou executado por um microprocessador, e que compreende instruções de código de programa para a execução do método de exibição como anteriormente descrito.

#### Lista de figuras

Outras características e vantagens da invenção serão claramente aparentes a partir da leitura da seguinte descrição de uma modalidade preferida, provida simplesmente para fins de exemplo e de nenhuma maneira de forma restritiva, e os desenhos anexos, entre os quais:

- figura 1 mostra o display produzido por um navegador de web a partir de um código HTML;
- figura 2 ilustra uma interface de um aplicativo de desenvolvimento de software em que, de acordo com uma modalidade da invenção, as propriedades dos elementos são exibidas;
- figura 3 ilustra uma modalidade do método da invenção.

#### Descrição Detalhada da Invenção

##### 1. Lembrança do Princípio da Invenção

É declarado que parte da presente invenção, um elemento pode ser qualquer parte do código, tal como uma verificação visual ou um bloco de código fonte, que pode ser sele-

cionado em uma interface de máquina humana de um aplicativo de desenvolvimento de software.

Em pelo menos uma modalidade, a invenção compreende meios para agrupar propriedades que são associadas com uma porção de um código selecionado pelo usuário (tal como um elemento específico de código). Uma propriedade é um Atributo nomeado de um objeto, uma verificação ou um elemento (doravante chamado de elemento). As propriedades definem as características do objeto tal como o tamanho, a cor, a posição da tela ou o estado de um objeto (ativado ou desativado, por exemplo). As propriedades são de tipos diferentes: o tamanho é um tipo de propriedade da mesma forma que a cor, a posição, o banco de dados ao qual o objeto é conectado, etc. As propriedades podem ser em branco (elas não contêm um valor) ou com valor (elas têm um valor).

Por exemplo, no caso da edição de um arquivo contendo código em formato HTML, a invenção permite que as propriedades de um mesmo tipo envolvam um elemento de código fonte selecionado para ser agrupado em uma única localização que é facilmente acessível para o desenvolvedor.

Conseqüentemente, o desenvolvedor vê imediatamente em uma vista a localização do valor do código fonte que ele deseja modificar. Por exemplo, a propriedade "cor de fundo" que pode ser modificada por muitos elementos de um código fonte HTML irá ser exibida em uma única e mesma localização (em uma parte de uma janela específica), tantas quantas vezes quanto existirem indicadores que iriam modificar esta propriedade para o elemento código fonte selecionado. O valor desta propriedade pode, por exemplo, ser "preto". Mais precisamente, no contexto do exemplo anterior, se o elemento código fonte selecionado pelo usuário é um texto dentro de um indicador `<TD></TD>`, pelo menos três propriedades "fundo de cor" serão exibidas na janela na parte de janela considerada: uma propriedade "cor de fundo" ligada a cada indicador `<TD></TD>`, `<TR></TR>` e `<TABLE></TABLE>`.

Conseqüentemente, o usuário que acessa rapidamente as propriedades não necessita de navegar de um indicador ou conjunto de indicadores para outro e pode assim, facilmente identificar o indicador em que o valor da propriedade que ele deseja modificar seja definido.

A correção ou a manutenção do aplicativo de software é conseqüentemente facilitada. A invenção opera contrariamente aos preconceitos daqueles versados na técnica. Na verdade, em vez de racionalmente agrupar as propriedades por elemento, a invenção permite que as propriedades sejam agrupadas por tipo, independentemente dos elementos aos quais essas propriedades pertencem. Este tipo de agrupamento é muito útil para correção ou operações de manutenção de aplicativo e pode ser usada em um aplicativo e desenvolvimento de software quando o programa desenvolvido pelo usuário estiver sendo mantido, por exemplo.

Agora, em relação à figura 1, uma interface gráfica padrão (10) de um aplicativo de desenvolvimento de software é apresentado. Tal uma interface compreende uma estrutura de árvore de arquivo (101), uma janela de edição de código (102), uma janela de apresentação de resultado de prévia ou compilação ou busca (103) e uma janela para o código sendo editado (104). A última janela (104) pode compreender propriedades relacionadas ao objeto ou elemento sendo editado.

Os exemplos anteriores são apenas providos para fins de ilustração. A invenção não é de nenhuma forma restrita a este exemplo específico. Na verdade, o método da invenção pode também ser implementado em muitos tipos de aplicativos de desenvolvimento de software tais como os aplicativos “eclipse”, “Dreamwaver” e “Delphi” e para muitas outras linguagens de desenvolvimento de aplicativo de software.

Na seguinte descrição, o caso de uma modalidade da invenção para o código HTML é usado.

## 2. Descrição de uma Modalidade

Nesta modalidade, a implementação da invenção das linguagens de indicadores HTML é provida. Linguagem HTML é uma linguagem que permite páginas a serem criadas que são exibidas por um navegador web. A linguagem HTML é uma linguagem de indicador que compreende instruções que influenciam renderizações de página HTML do navegador de web e essas instruções estão na forma de indicadores em volta dos elementos definidos na página HTML.

Agora, em relação à figura 2, uma parte do código HTML (10) é apresentada e o resultado (11) desta interpretação (12) deste código HTML por um navegador web isto é a exibição correspondente.

O código HTML contém muitos indicadores tais como, por exemplo, os indicadores <H1>, <P>, e <SPAM>.

Esses indicadores são usados para indicar um navegador web na maneira de exibir o conteúdo da página. O Uso da linguagem HTML é quase sistematicamente acompanhado pelo uso de “CSS” (Cascade Style sheets (folha tipo cascata)).

Essas folhas tipo cascata (e a linguagem de computador que é usada para construir essas folhas tipo cascata) são usadas para descrever a apresentação de documentos HTML e XML. A vantagem de folhas tipo cascata é que teoricamente elas permitem todas dentre as escolhas de apresentação e decoração dos elementos HTML que formam a página (as folhas tipo cascata) para serem agrupados em um único arquivo. Elas, portanto, também contêm propriedades relacionadas aos elementos de código HTML. As folhas tipo cascata também são usadas na conversão de arquivos para o formato HTML para gerar o código HTML.

Em uso, entretanto, inventores descobriram que as folhas tipo cascata são rara-

mente usadas do início até o fim no desenvolvimento de um aplicativo de web que provê o navegador com o código HTML.

Na verdade, o uso de folhas tipo cascata é usado frequentemente na criação do aplicativo: iniciando com boas intenções, os desenvolvedores são bastante inclinados a separar de forma eficiente a apresentação (as folhas tipo cascata) e os dados. Durante a manutenção ou a correção do código, do outro lado, as folhas tipo cascata são usadas com menos frequência: frequentemente, a pessoa que criou o aplicativo não é a pessoa que realiza a manutenção corretiva e um certo grau de conhecimento do aplicativo é perdido com esta mudança.

Nesta modalidade da invenção, é, portanto, proposta para agrupar juntas as propriedades do mesmo tipo, incluindo as propriedades de uma folha tipo cascata, em uma mesma parte da tela ou mesma janela na tela. Este agrupamento é realizado quando um elemento ou uma parte de elemento é selecionado pelo usuário. Esta seleção pode ser realizada diretamente no código fonte ou mesmo em um modo de apresentação "WYSIWYG" (what You See Is What You Get).

Em relação à figura 3, as etapas gerais do método da invenção são apresentadas. Elas compreendem:

- uma etapa para selecionar 31, dentro do código fonte, o elemento de código fonte 311. Esta seleção é feita pelo usuário.
- uma etapa para busca 32, dentro do código fonte 321, um conjunto de elementos de código relacionados do elemento de código fonte selecionado;
- uma etapa para criar 33 uma estrutura de dados 331 compreendendo propriedades dos elementos de código do conjunto de elementos de código;
- uma etapa para agrupar 34, dentro da estrutura de dados, as propriedades na função de seus tipos;
- uma etapa para exibir 35 as propriedades agrupadas dentro de uma parte de meios de exibição do código fonte 351.

Em uma primeira variação desta modalidade da invenção, a seleção de um elemento de uma parte ou uma parte de elemento leva inicialmente a exibição, na forma de uma lista, de elementos que influenciam as propriedades d elemento selecionado. Seguinte a esta exibição, o usuário é capaz de selecionar, a partir da lista, o elemento que ele pretende. Subsequentemente, as propriedades do elemento selecionado são, por exemplo, exibidas, com a possibilidade do usuário de retroceder, isto é, retornar para a lista de elementos.

Em uma segunda variação desta modalidade da invenção, a seleção de um elemento ou uma parte do elemento leva a definição de uma lista de propriedades, que são exibidas. Esta lista agrupa juntas as propriedades por tipo: é conseqüentemente possível para a mesma propriedade aparecer muitas vezes nesta lista. A providência da propriedade

(o elemento relacionado ao qual ele pertence) é mostrada por meios adequados, tais como um ícone, um balão de informação ou quaisquer outros meios, de modo que o usuário não desconcerte ou perca por este display.

Conseqüentemente o usuário tem a possibilidade de imediatamente ver qual propriedade precisa ser modificada e não tem que selecionar um por um dos elementos relacionado para identificar o elemento cuja propriedade precisa ser modificada.

De acordo com outra variação desta modalidade da invenção, apenas as propriedades que são efetivamente usadas são exibidas. Conseqüentemente após a eliminação das propriedades que não influenciam o elemento selecionado, o método da invenção faz uma nova seleção de modo que apenas as propriedades que possuem valores são exibidas: conseqüentemente, quando uma única propriedade de um único elemento Possui valor, ela é a única a ser exibida. Quando muitas propriedades do mesmo tipo possuem valor, as muitas propriedades são as únicas a serem exibidas.

Desta maneira, o usuário está seguro para respeitar as escolhas de implementações e apresentação que foram feitas anteriormente não modificando uma propriedade no lugar de outra.

Claro que é possível comutar o modo onde todas as possíveis propriedades são exibidas para o modo onde apenas as propriedades efetivamente usadas são exibidas, o que é importante é que isto se refere às propriedades dos elementos relacionados.

### 20 3. Gerenciamento do Código Fonte

No nível do código fonte propriamente dito, a invenção falha em uma árvore de sintaxe. A árvore é armazenada na memória quando o código fonte é editado. Para identificar os elementos que influenciam um elemento selecionado pelo usuário (os elementos relacionados), a invenção propõe um método simples e rápido: os elementos que influenciam os elementos selecionados são os principais, na árvore, do elemento selecionado.

Conseqüentemente, considerando o exemplo dos indicadores `<TD></TD>`, `<TR></TR>` e `<TABLE></TABLE>`, o pai do elemento `<TD></TD>` é o elemento `<TR></TR>` e o pai do elemento `<TR></TR>` é o elemento `<TABLE></TABLE>`. Quaisquer estilos que forem associados a folhas tipo cascata nos elementos sucessivamente levados em consideração são também exibidos.

De acordo com outra variação desta modalidade, apenas as propriedades que podem realmente influenciar o elemento ou a parte do elemento selecionado são exibidas. Conseqüentemente, as propriedades que não têm impacto no elemento selecionado não são exibidas, de modo a evitar extensão da lista com propriedades inutilizadas.

Para fazer essa seleção de propriedades, muitos métodos são possíveis incluindo:

O uso de um critério de similaridade: apenas as propriedades de elementos selecionado que têm o mesmo nome ou nome similar aqueles do elemento selecionado são exi-

bidas (por exemplo, “cor” ou “fundo de cor”);

O uso de m tabela de referência cruzada, que permite um diagrama de influência a ser definido, contendo as relações entre as várias propriedades dos vários elementos.

5 Este diagrama de influência é verificado para determinar as propriedades a serem exibidas em função do elemento selecionado. Este diagrama de influência é determinado antes em função da linguagem. Pode ser representado na forma de estrutura de árvore de dados e/ou ser implementado em um banco de dados.

## REIVINDICAÇÕES

1. Método para exibição das propriedades de um elemento de código fonte de um programa de computador sendo editado, **CARACTERIZADO** pelo fato de que compreende:

5       - uma etapa para selecionar, dentro do dito código fonte, o dito elemento de código fonte;

          - uma etapa para buscar, dentro do dito código fonte, um conjunto de elementos de código relacionados do dito elemento de código fonte selecionado, os elementos do conjunto compreendem propriedades, que têm um tipo;

10       - uma etapa para criar uma estrutura de dados que compreende as propriedades dos elementos de código do conjunto de elementos de código;

          - uma etapa para agrupar, dentro da estrutura de dados, as propriedades em função de seus tipos;

          - uma etapa para exibir as propriedades agrupadas dentro de uma parte de meios de exibição do código fonte.

15       2. Método para exibição, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que adicionalmente compreende, antes da etapa de agrupamento, uma etapa para deletar, dentro da estrutura de dados propriedades que não podem ser usadas para influenciar o elemento de código fonte antes do manuseio pelo usuário.

20       3. Método para exibição, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que adicionalmente compreende, antes da etapa de agrupamento, uma etapa para deletar, dentro da estrutura de dados propriedades que não contêm valores.

          4. Método para exibição, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que adicionalmente compreende, antes da etapa de selecionar e durante a edição do código fonte:

25       - uma etapa para criar uma árvore de sintaxe hierárquica para o código fonte, incluindo para cada elemento do código fonte, pelo menos um elemento principal;

          - uma etapa para memorizar a árvore de sintaxe.

30       5. Método para exibição, de acordo com a reivindicação 3, **CARACTERIZADO** pelo fato de que adicionalmente compreende, antes da etapa de busca compreende uma etapa para identificar, dentro da árvore de sintaxe memorizada, o pelo menos um elemento parente do elemento selecionado.

6. Sistema para exibir um elemento de código fonte de um programa de computador sendo editado, em que o programa é gravado em uma linguagem tipo objeto ou tipo componente, **CARACTERIZADO** pelo fato de que compreende:

35       - meios para selecionar, dentro do código fonte, o elemento de código fonte;

          - meios para buscar, dentro do código fonte, um conjunto de elementos de código relacionados do dito elemento de código fonte selecionado, os elementos do conjunto com-

preendendo propriedades tipificadas;

- meios para criar uma estrutura de dados compreendendo as propriedades dos elementos de código do conjunto de elementos de código;

5 - meios para agrupar, dentro da estrutura de dados, as propriedades em função de seus tipos;

- meios para exibir as propriedades agrupadas juntas dentro de uma parte de meios de exibição do código fonte.

10 7. Programa de computador que pode ter download realizado a partir de uma rede de comunicação e/ou armazenado em um suporte que pode ser lido por um computador e/ou executado por um microprocessador, **CHARACTERIZADO** pelo fato de que compreende instruções de código de programa para a execução do método de exibição como definido em pelo menos uma das reivindicações de 1 a 5, quando ele é executado em um computador..

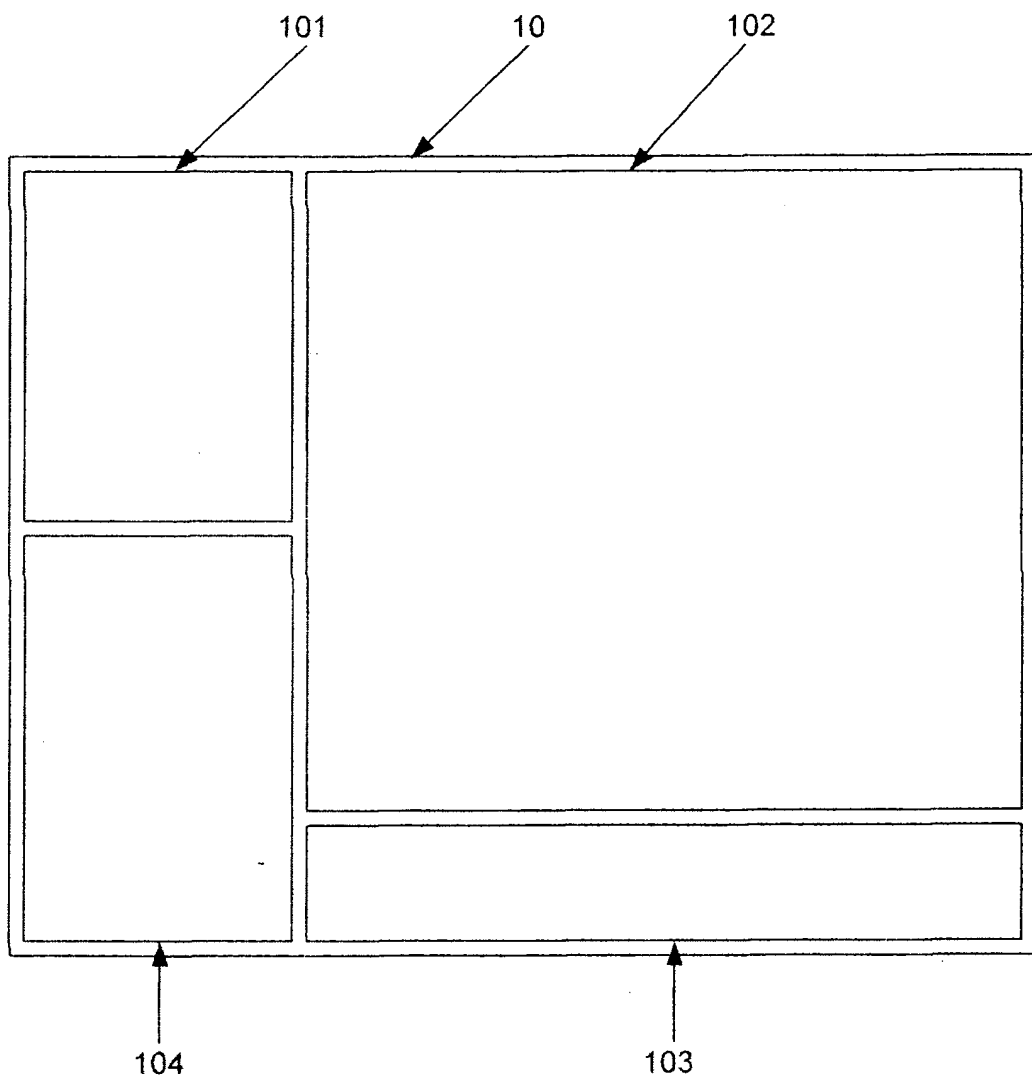


Figura 1

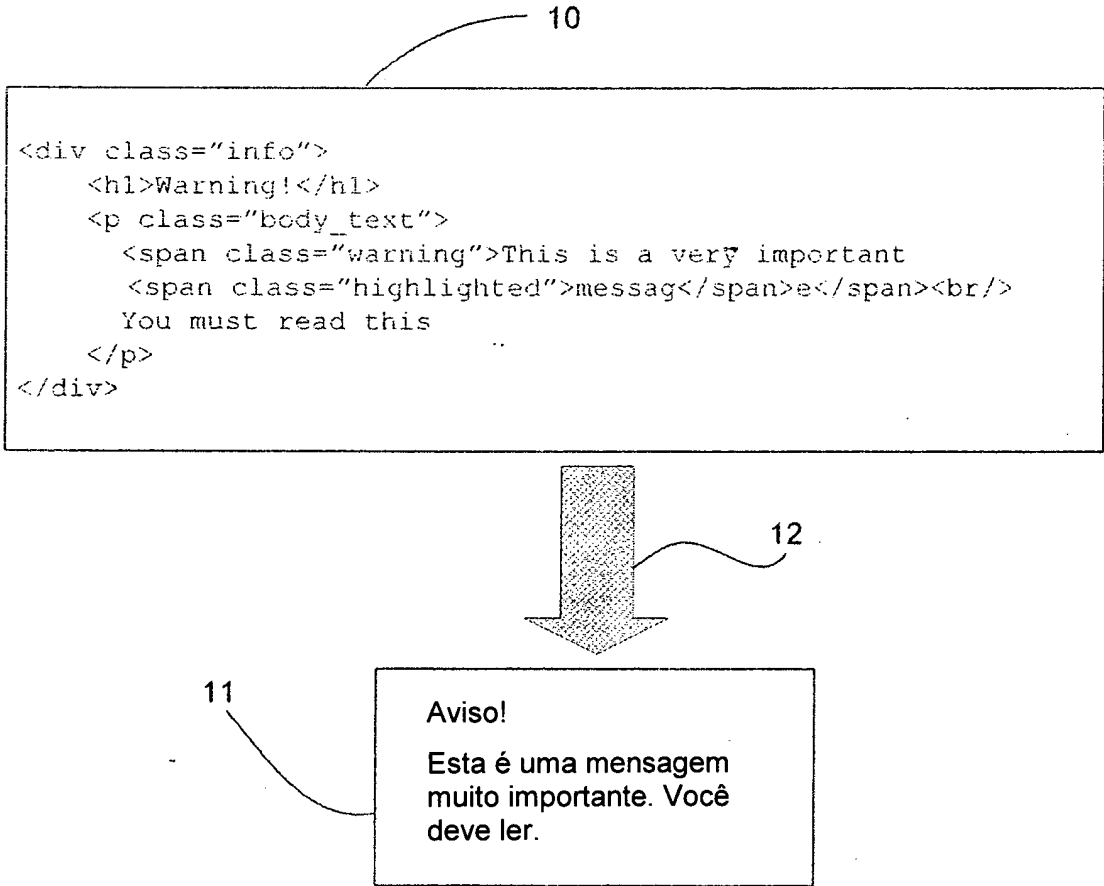


Figura 2

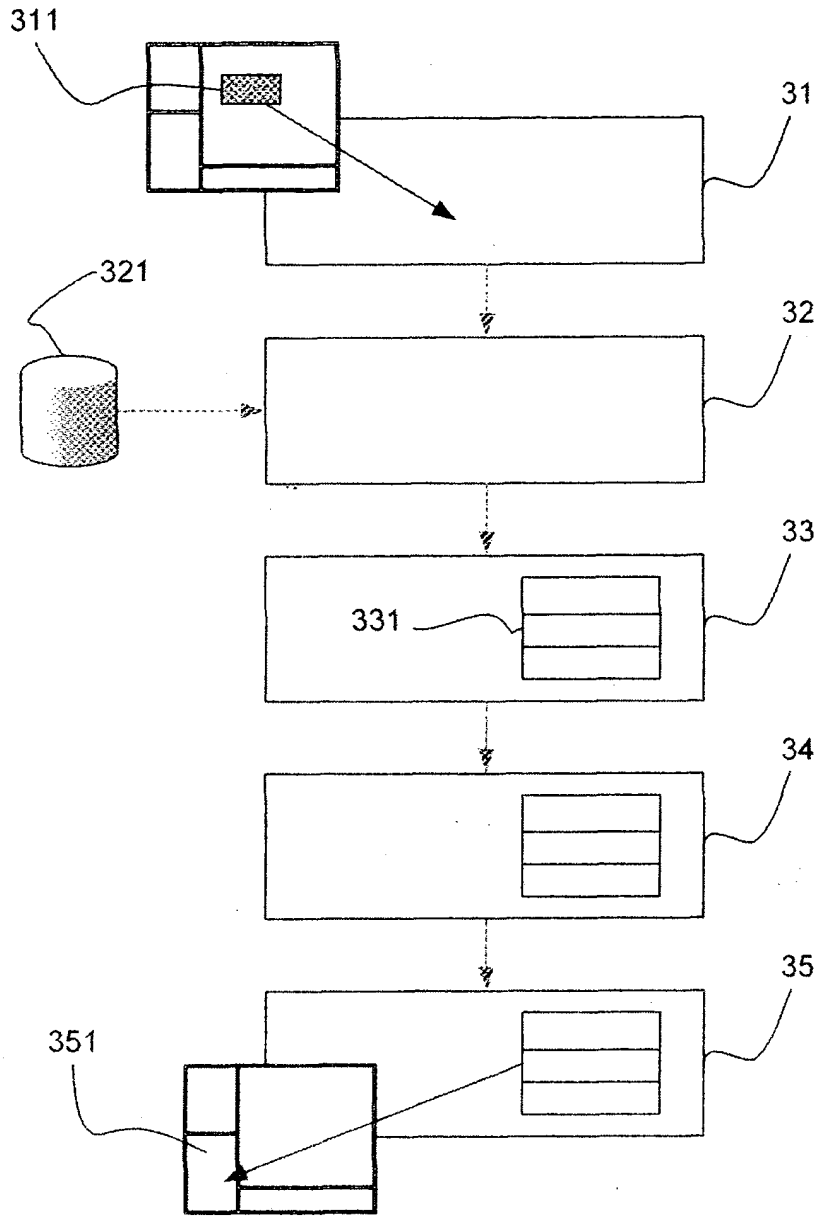


Figura 3

RESUMO**"MÉTODO PARA EXIBIÇÃO, PROGRAMA DE COMPUTADOR CORRESPONDENTE E DISPOSITIVO"**

5 A invenção refere-se a um método para exibir as propriedades de um elemento de código fonte de um programa de computador sendo editado.

De acordo com a invenção tal um método compreende:

- uma etapa para selecionar, dentro do dito código fonte, o dito elemento de código fonte;
- uma etapa para buscar, dentro do dito código fonte, um conjunto de elementos de código relacionados do dito elemento de código fonte selecionado, os elementos do conjunto que compreendem propriedades tipificadas;
- uma etapa para criar uma estrutura de dados que compreende as propriedades dos elementos de código do conjunto de elementos de código;
- uma etapa para agrupar, dentro da estrutura de dados, as propriedades em função de seus tipos;
- uma etapa para exibir as propriedades agrupadas dentro de uma parte de meios de exibição do código fonte.