



República Federativa do Brasil
Ministério da Economia
Instituto Nacional da Propriedade Industrial

(21) PI 1014465-0 A2



(22) Data do Depósito: 23/04/2010

(43) Data da Publicação Nacional: 15/09/2020

(54) **Título:** MÉTODO E SISTEMA PARA PROPORCIONAR OTIMIZAÇÃO DA PLATAFORMA DE VISUALIZAÇÃO E MEIO LEGÍVEL POR COMPUTADOR

(51) **Int. Cl.:** G06F 3/14; G06F 9/06.

(30) **Prioridade Unionista:** 30/04/2009 US 12/433.288.

(71) **Depositante(es):** MICROSOFT CORPORATION.

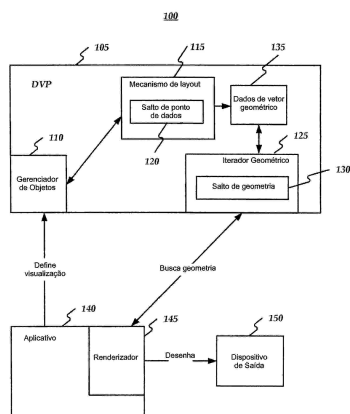
(72) **Inventor(es):** BARRY CHRISTOPHER ALLYN; B. SCOTT RUBLE.

(86) **Pedido PCT:** PCT US2010032307 de 23/04/2010

(87) **Publicação PCT:** WO 2010/126802 de 04/11/2010

(85) **Data da Fase Nacional:** 13/10/2011

(57) **Resumo:** MÉTODO E SISTEMA PARA PROPORCIONAR OTIMIZAÇÃO DA PLATAFORMA DE VISUALIZAÇÃO E MEIO LEGÍVEL POR COMPUTADOR. A presente invenção proporciona otimização da plataforma de visualização de dados. Os aplicativos podem fornecer valores de dados e solicitar a criação de uma visualização a partir de uma plataforma de visualização de dados (DVP). O DVP pode compor uma pluralidade de registros geométricos associados a um subconjunto dos valores de dados de visualização. O aplicativo pode processar a visualização iterando através dos vetores geométricos e traduzindo um subconjunto dos vetores em instruções de desenho para saída em um dispositivo de exibição.



Relatório Descritivo da Patente de Invenção para "**MÉTODO E SISTEMA PARA PROPORCIONAR OTIMIZAÇÃO DA PLATAFORMA DE VISUALIZAÇÃO E MEIO LEGÍVEL POR COMPUTADOR**".

ANTECEDENTES DA INVENÇÃO

5 As otimizações de desempenho da plataforma de visualização de dados oferecem pior desempenho na geração de objetos visuais. Em algumas situações, a geração de objetos visuais pode fazer uso intenso de recursos, exigindo alto desempenho do computador. Por exemplo, a geração de gráficos e mapas pode exigir uma grande quantidade de potência compu-
10 tacional e/ou memória, especialmente quando o objeto compreende um grande número de pontos de dados. A estratégia convencional consiste em renderizar cada ponto de dados, independente do tamanho do objeto resultante. Isso pode gerar problemas, pois certas operações no objeto podem fazer com que o computador demore a responder. Por exemplo, a exibição
15 de um mapa grande, a seleção de uma parte de um gráfico detalhado, a rolagem, impressão e/ou a modificação do objeto podem exigir uma grande quantidade de potência computacional e podem fazer com que o computador demore a responder.

SUMÁRIO

20 Otimizações de desempenho da plataforma de visualização de dados podem ser proporcionadas. A intenção deste sumário é a de apresentar, de maneira simplificada, uma seleção de conceitos descritos em detalhes a seguir na Descrição Detalhada. O presente Sumário não pretende identificar aspectos críticos ou essenciais da matéria reivindicada. Tampouco
25 deve este Sumário ser usado para limitar o âmbito da matéria reivindicada. Pode-se obter uma otimização da plataforma de visualização de dados. Os aplicativos podem fornecer valores de dados e solicitar a criação de uma visualização a partir de uma plataforma de visualização de dados (DVP). O DVP pode compor uma pluralidade de registros geométricos associados a
30 um subconjunto dos valores de dados de visualização. O aplicativo pode renderizar a visualização iterando através dos vetores geométricos e traduzindo um subconjunto dos vetores em instruções de desenho para saída em

um dispositivo de exibição.

Tanto a descrição geral anterior quanto a descrição detalhada seguinte apresentam exemplos e são meramente explanatórias. Logo, a descrição geral anterior e a descrição detalhada seguinte não devem ser consideradas como restritivas. Além disso, podem ser proporcionados outros aspectos ou variações além dos apresentados na presente invenção. Por exemplo, as concretizações podem ser direcionadas a várias combinações e subcombinações de aspectos descritas na descrição detalhada.

DESCRIÇÃO RESUMIDA DOS DESENHOS

Os desenhos em anexo, que são incorporados e constituem uma parte desta revelação, ilustram várias concretizações da presente invenção. Nos desenhos:

A FIG. 1 é um diagrama de blocos de um ambiente de operação;

A FIG. 2 é um diagrama de fluxo de dados de um método para otimizar o desempenho da plataforma de visualização; e

A FIG. 3 é um diagrama de blocos de um sistema que inclui um dispositivo de computação.

DESCRIÇÃO DETALHADA DA INVENÇÃO

A descrição detalhada a seguir refere-se aos desenhos em anexo. Sempre que possível, usam-se os mesmos números de referência nos desenhos e na descrição seguinte para designar os mesmos elementos, ou elementos similares. Embora sejam descritas concretizações da invenção, são possíveis modificações, adaptações e outras implementações. Por exemplo, substituições, adições ou modificações podem ser feitas nos elementos ilustrados nos desenhos, e os métodos aqui descritos podem ser modificados mediante a substituição, reordenação ou adição de estágios aos métodos revelados. Logo, a descrição detalhada a seguir não limita a invenção. Em vez disso, o âmbito correto da invenção é definido pelas reivindicações em anexo.

Otimizações de desempenho da plataforma de visualização de dados (DVP) podem ser proporcionadas. De acordo com as concretizações da presente invenção, uma DVP pode ignorar pontos de dados durante a

composição de um objeto de acordo com uma resolução de exibição e pode construir um conjunto mínimo de vetores de geometria na memória. A DVP também pode ignorar vetores compostos durante a renderização do objeto. A DVP pode ainda permitir o escalonamento dos algoritmos para ignorar
5 pontos de dados e/ou vetores a fim de expor diferentes configurações de qualidade, tal como pelo uso de uma interface de programação de aplicativos (API) que permite o controle sobre configurações tal como o consumo máximo de memória, tempo e/ou contagens de pontos de dados.

A FIG. 1 é um diagrama de blocos de um ambiente operacional
10 100 para proporcionar uma plataforma de visualização 105. A plataforma de visualização 105 pode compreender um módulo de objeto 110, um mecanismo de layout 115 e um iterador geométrico 125. O mecanismo de layout 115 pode compreender um algoritmo de salto de ponto de dados 120. O iterador geométrico 125 pode compreender um algoritmo de salto geométrico
15 130. O iterador geométrico 125 e o mecanismo de layout 115 podem criar, atualizar, ler, recuperar e/ou fornecer uma pluralidade de dados de vetor geométrico 135. O ambiente operacional 100 pode adicionalmente compreender um aplicativo 140 compreendendo um renderizador 145 operativo para renderizar instruções de desenho a um dispositivo de saída 150, tal como
20 um monitor, tela, impressora e/ou outro dispositivo de exibição. A plataforma de visualização 105 pode compreender uma arquitetura que permite a criação de uma visualização de dados, tal como um gráfico, e pode expor um aspecto interativo da visualização. A arquitetura pode oferecer integração com múltiplas plataformas de renderização. Quando um usuário seleciona o aspecto exposto, a arquitetura pode traduzir a seleção em um formato comum e modificar a visualização de dados de acordo com regras de layout, independente da plataforma de renderização.

O aplicativo 140 pode definir uma visualização através do módulo de objeto 110. O módulo de objeto 110 pode chamar o mecanismo de layout 115, que pode construir um conjunto de registros geométricos na memória na forma de dados de vetor geométrico 135. O aplicativo 140 pode posteriormente precisar renderizar a visualização e pode chamar o iterador
30

geométrico 125, que pode acessar os dados de vetor geométrico 135 e retorná-los ao aplicativo 140. O aplicativo 140 pode traduzir cada primitiva geométrica nos dados de vetor geométrico 135 em instruções de desenho para renderização ao dispositivo de saída 150.

5 A plataforma de visualização 105 pode compreender um núcleo compartilhado compreendendo bibliotecas de software e/ou utilitários para proporcionar visualizações interativas. O núcleo compartilhado pode ser implementado, por exemplo, em C++ ou C#, e pode ser independente da plataforma. O núcleo compartilhado pode compreender utilitários de visualização
10 para proporcionar layouts, formas e/ou geometria, serviços de linha, renderização tridimensional, geração de quadros de animação e/ou pontos de acesso interativos. A plataforma de visualização 110 pode adicionalmente compreender interfaces de programação de aplicativos (APIs) para interagir com o aplicativo 140.

15 Duas áreas para otimização de desempenho podem ser durante a composição dos dados de vetor geométrico 135 e a tradução dos dados de vetor geométrico 135 em instruções de desenho para renderização. Por exemplo, durante a composição dos dados de vetor geométrico 135, o uso de memória pode aumentar consideravelmente à medida que aumenta a quantidade de dados. A quantidade de memória necessária pode ser reduzida
20 mediante o uso do algoritmo de salto de ponto de dados 120 de modo a compor vetores geométricos para um subconjunto dos dados. O algoritmo de salto geométrico 130 pode operar de forma a reduzir o número de instruções de desenho que precisam ser renderizadas mediante a tradução de um sub-
25 conjunto de dados de vetor geométrico 135.

A FIG. 2 é um diagrama de fluxo de dados que apresenta os estágios gerais envolvidos em um método de acordo com uma concretização da invenção para proporcionar otimizações da plataforma de visualização de dados. O método 200 pode ser implementado usando um dispositivo de
30 computação 300 conforme descrito acima em mais detalhes abaixo em relação à FIG. 3. Formas de implementação dos estágios do método 200 serão descritas em mais detalhes adiante. O método 200 pode começar no bloco

inicial 205 e prosseguir ao estágio 210, em que o dispositivo de computação 300 pode definir uma visualização. Por exemplo, o aplicativo 140 pode enviar uma solicitação à plataforma de visualização 105 para criar um objeto visual. A solicitação pode compreender um valor de dados, uma série de dados e/ou um tipo de objeto, tal como um gráfico de linhas, um gráfico de barras, um gráfico de torta ou outro tipo de gráfico. A plataforma de visualização 105 pode usar o gerenciador de objetos para definir uma localização de memória para o objeto visual.

A partir do estágio 210, em que o dispositivo de computação 300 definiu a visualização, o método 200 pode avançar para o estágio 220, em que o dispositivo de computação 300 pode compor pelo menos um vetor geométrico associado à visualização. Por exemplo, o aplicativo 140 pode definir uma visualização de um gráfico compreendendo 1.000 pontos de dados, em que cada ponto de dado pode ser representado por uma forma octagonal desenhada. Nos sistemas convencionais, a plataforma de visualização 105 pode compor vetores de geometria para cada um dos 1.000 pontos de dados. De acordo com as concretizações da invenção, o mecanismo de layout 115 pode usar o algoritmo de salto de ponto de dado 120 para compor vetores geométricos para um subconjunto dos 1.000 pontos de dados. Isso pode permitir que a exibição de um objeto visual com qualidade de esboço e/ou pré-visualização. Ainda de acordo com as concretizações da invenção, o mecanismo de layout 115 pode ser operativo para determinar se os pontos de dados dentro dos 1.000 pontos de dados estão suficientemente próximos uns dos outros para serem indistinguíveis, com base nas características do aplicativo 140 e/ou do dispositivo de saída 150. Por exemplo, se todo um subconjunto de 30 pontos de dados dentro dos 1.000 pontos de dados sobrepõem ou entram dentro de um limiar predefinido um do outro, tal como de 5 pixels, o algoritmo de salto de ponto de dados 120 pode ser operativo para saltar a composição de todos os 30 pontos de dados e pode compor apenas um dos 30 pontos de dados.

De acordo com as concretizações da invenção, o aplicativo 140 pode definir um gráfico de coluna compreendendo 10.000 pontos de dados.

O mecanismo de layout 115 pode chamar uma classe de série de coluna que pode percorrer os dados e vetores compostos em busca de caixas para os pontos de dados e colocá-los nos dados de vetor geométrico 135. O mecanismo de layout 115 pode receber, do aplicativo 140, por exemplo, um tamanho de gráfico em pixels e determinar quantos pontos de dados podem ser exibidos neste tamanho do gráfico. Por exemplo, um gráfico de 300 pixels de largura só poderá exibir 300 pontos de dados sem sobreposição. O mecanismo de layout 135, assim, pode compor dados vetoriais para um subconjunto de 300 pontos dos 10.000 pontos de dados. O tamanho do subconjunto pode ser baseado, por exemplo, na resolução do dispositivo de saída 150.

O tamanho do subconjunto também pode ser baseado, por exemplo, em uma determinação, do mecanismo de layout 115, de que certos pontos de dados serão cobertos por outros pontos de dados, e o mecanismo de layout 115, dessa forma, pode saltar a composição dos dados vetoriais para os pontos de dados cobertos. De acordo com as concretizações da invenção, o mecanismo de layout 115 pode determinar que um ponto de dados posterior ocupa exatamente o mesmo pixel que o ponto de dado anterior, e, assim, o último ponto de dado pode ser saltado. De acordo com concretizações adicionais da invenção, o algoritmo de salto de ponto de dados 120 pode usar um critério configurável para determinar se o último ponto está suficientemente próximo do ponto de dado anterior para ser saltado. Por exemplo, o mecanismo de layout 115 pode definir o critério de salto de pontos de dados dentro de cinco pixels de um ponto de dado anterior. Ainda de acordo com as concretizações da invenção, o critério pode ser aumentado à medida que o estágio de composição prossegue a fim de limitar a quantidade de memória consumida pela composição dos dados de vetor geométrico 135. O critério também pode ser recebido do aplicativo 140, tal como por meio da seleção de um controle de interface pelo usuário.

Após o dispositivo de computação 300 compor os vetores geométricos no estágio 220, o método 200 pode continuar no estágio 230, em que o dispositivo de computação 300 pode receber uma solicitação para

renderizar a visualização. Por exemplo, o aplicativo 140 pode solicitar a renderização da visualização de acordo com um comando do usuário.

Após o dispositivo de computação 300 receber a solicitação de renderização no estágio 230, o método 200 pode prosseguir para o estágio 240, em que o dispositivo de computação 300 pode iterar através dos vetores geométricos compostos e traduzi-los em instruções de desenho. Por exemplo, a plataforma de visualização 105 pode traduzir um vetor geométrico compreendendo um eixo x de um objeto de gráfico em uma instrução de desenho para uma linha, em que a instrução de desenho compreende dados associados, tal como um ponto inicial para a linha, um ponto final para a linha, uma cor para a linha e/ou uma largura para a linha.

De acordo com as concretizações da invenção, a plataforma de visualização 105 pode receber pelo menos uma instrução de desenho suportada do aplicativo. A plataforma de visualização 105 pode ser operativa para traduzir os vetores geométricos compostos nas instruções de desenho suportadas. Por exemplo, a plataforma de visualização 105 pode ser operativa para linearizar um vetor geométrico compreendendo um octágono em instruções de desenho compreendendo um ponto inicial e final para cada um de um conjunto de oito linhas usadas para renderizar uma forma octagonal no dispositivo de saída 150. O aplicativo 140 pode, em vez disso ou adicionalmente, informar a plataforma de visualização de uma instrução de desenho suportada compreendendo um ponto central e uma largura para uma forma octagonal de modo que a plataforma de visualização 105 possa traduzir o vetor geométrico compreendendo um octágono na instrução de desenho suportada em vez de linearizar a forma octagonal. O aplicativo 140 pode, dessa forma, reduzir o número de instruções de desenho traduzidas que a plataforma de visualização 105 pode precisar fornecer.

O dispositivo de computação 300 pode utilizar o algoritmo de salto de geometria 130 para reduzir o número de vetores geométricos que precisam ser traduzidos e renderizados. Por exemplo, se a visualização compreender um gráfico de 100 pixels por 200 pixels usando formas octagonais para pontos de dados, cada octágono pode compreender uma forma de

10 pixels por 10 pixels, ou menor, e alguns segmentos de linha podem compreender comprimentos zero. O algoritmo de salto de geometria 130 pode, por exemplo, determinar que um vetor geométrico em particular está associado a um critério a ser saltado, por exemplo, o vetor pode ser traduzido em
5 uma instrução de desenho para uma linha com o mesmo ponto inicial e final. Em tal exemplo, o algoritmo de salto de geometria 130 pode saltar o vetor geométrico em particular em vez de traduzi-lo em uma instrução de desenho e enviá-lo ao aplicativo 140 para renderização. O algoritmo de salto de geometria 130 pode saltar um dos vetores geométricos de acordo com muitos
10 outros critérios, tal como um tamanho de exibição, um tipo de dado, um tipo de visualização, um número total de pontos de dados e um número total de vetores geométricos.

De acordo com as concretizações da invenção, o aplicativo 140 pode buscar as instruções de desenho em lotes. Por exemplo, o aplicativo
15 140 pode alocar memória suficiente para receber 50 das instruções de desenho. A plataforma de visualização 105 pode então fornecer 50 instruções de desenho e informar o aplicativo 140 de se há mais instruções de desenho aguardando. A plataforma de visualização 105 pode esperar o aplicativo 140 solicitar um lote subsequente e repetir o processo.

20 A partir do estágio 240, o método 200 pode avançar para o estágio 250, em que o dispositivo de computação 300 pode renderizar a visualização para um dispositivo de saída. Por exemplo, o aplicativo 140 pode receber as instruções de desenho no renderizador 145 e desenhar a visualização em uma tela compreendendo o dispositivo de saída 150.

25 Após o dispositivo de computação 300 renderizar a visualização no estágio 250, o método 200 pode avançar para o estágio 260, em que o dispositivo de computação 300 pode determinar se ocorreu uma alteração na exibição. Por exemplo, a visualização exibida pode ser movida, ampliada, redimensionada e/ou obscurecida total ou parcialmente. Se o dispositivo de
30 computação determinar que ocorreu uma alteração na exibição da visualização, o método 300 pode retornar ao estágio 250, em que o dispositivo de computação 300 pode renderizar novamente a visualização. O salto do pon-

to de dados pode acontecer durante a composição no estágio 220 quando os dados de vetor geométrico 135 são armazenados em cache, ao passo que o salto geométrico pode acontecer toda vez que a visualização é renderizada no estágio 250. Então, por exemplo, cada vez que o aplicativo 140 precisa renderizar a visualização, tal como quando uma janela é movida, minimizada, maximizada ou ampliada, o aplicativo 140 pode receber um novo conjunto de instruções de desenho do iterador geométrico 125. Quando o aplicativo 140 reduz o zoom, por exemplo, menos pixels podem ser exibidos na visualização e o salto geométrico pode ser mais agressivo, saltando mais vetores geométricos.

De acordo com as concretizações da invenção, o salto de ponto de dados e o salto de geometria podem ser usados separadamente e/ou juntos e podem ser controlados por meio de uma preferência de configuração exposta do usuário. Por exemplo, um usuário que deseja ampliar em mais detalhes um gráfico pode desativar o salto de ponto de dados a fim de fornecer a maior quantidade de dados, ao mesmo tempo em que permite o salto de geometria para impedir a tradução de vetores não visíveis no estado ampliado.

Uma vez que o dispositivo de computação 300 tenha determinado se ocorreu uma alteração na exibição no estágio 260 e renderizado novamente a visualização, se necessário, o método 200 pode terminar no estágio 270.

Uma concretização de acordo com a invenção pode compreender um sistema para proporcionar otimização da plataforma de visualização. O sistema pode compreender um armazenamento em memória e uma unidade de processamento acoplada ao armazenamento em memória. A unidade de processamento pode ser operativa para definir uma visualização, criar uma pluralidade de registros de geometria, receber uma solicitação para exibir a visualização, iterar através da pluralidade de vetores geométricos, traduzir pelo menos um da pluralidade de registros geométricos em pelo menos uma instrução de desenho, e exibir a visualização em um dispositivo de exibição.

Outra concretização de acordo com a invenção pode compreender um sistema para proporcionar otimização da plataforma de visualização. O sistema pode compreender um armazenamento em memória e uma unidade de processamento acoplada ao armazenamento em memória. A unidade de processamento pode ser operativa para criar um objeto de visualização associado a uma pluralidade de valores de dados, compor pelo menos um primeiro valor da pluralidade de valores de dados em um vetor geométrico associado ao objeto de visualização, e saltar a composição de pelo menos um segundo valor da pluralidade de valores de dados.

Ainda outra concretização de acordo com a invenção pode compreender um sistema para proporcionar otimização de uma plataforma de visualização. O sistema pode compreender um armazenamento em memória e uma unidade de processamento acoplada ao armazenamento em memória. A unidade de processamento pode ser operativa para receber uma solicitação, de um aplicativo do usuário, para criar um objeto de visualização, compor uma pluralidade de vetores geométricos associados ao objeto de visualização, receber uma solicitação de renderização do aplicativo do usuário, traduzir cada um de um subconjunto da pluralidade de vetores geométricos em pelo menos uma instrução de desenho, enviar a pelo menos uma instrução de desenho associada a cada um do subconjunto da pluralidade de vetores geométricos ao aplicativo do usuário e renderizar o objeto de visualização. A unidade de processamento pode ser adicionalmente operativa para receber uma segunda solicitação de renderização do aplicativo do usuário em resposta a uma alteração detectada que afeta o objeto de visualização renderizado, traduzir cada um de um segundo subconjunto da pluralidade de vetores geométricos em pelo menos uma instrução de desenho, enviar a pelo menos uma instrução de desenho associada a cada um do segundo subconjunto da pluralidade de vetores geométricos ao aplicativo do usuário, e renderizar novamente o objeto de visualização.

A FIG. 3 é um diagrama de blocos de um sistema que inclui o dispositivo de computação 300. De acordo com a concretização da invenção, o armazenamento em memória supramencionado e a unidade de pro-

cessamento podem ser implementados em um dispositivo de computação, tal como o dispositivo de computação 300 da FIG. 3. Qualquer combinação adequada de hardware, software ou firmware pode ser usada para implementar o armazenamento em memória e a unidade de processamento. Por exemplo, o armazenamento em memória e a unidade de processamento podem ser implementados com o dispositivo de computação 300 ou qualquer um dos outros dispositivos de computação 318, em combinação com o dispositivo de computação 300. O sistema, dispositivo e processadores supramencionados são exemplos e outros sistemas, dispositivos e processadores podem compreender o armazenamento em memória e unidade de processamento supramencionados, de acordo com as concretizações da invenção. Além disso, o dispositivo de computação 300 pode compreender um ambiente operacional para o sistema 100, como descrito acima. O sistema 100 pode operar em outros ambientes e não se limita ao dispositivo de computação 300.

Com referência à FIG. 3, um sistema de acordo com uma concretização da invenção pode incluir um dispositivo de computação, tal como o dispositivo de computação 300. Em uma configuração básica, o dispositivo de computação 300 pode incluir pelo menos uma unidade de processamento 302 e uma memória do sistema 304. Dependendo da configuração e do tipo do dispositivo de computação, a memória do sistema 304 pode compreender, mas não se limita à memória volátil (por exemplo, memória de acesso aleatório (RAM)), não-volátil (por exemplo, memória somente para leitura (ROM)), memória flash ou qualquer combinação destas. A memória do sistema 304 pode incluir o sistema operacional 305, um ou mais módulos de programação 306 e pode incluir a plataforma de visualização 110. O sistema operacional 305, por exemplo, pode ser adequado para controlar a operação do dispositivo de computação 300. Em uma concretização, os módulos de programação 306 podem incluir o aplicativo do usuário 140. Além do mais, as concretizações da invenção podem ser praticadas em conjunto com uma biblioteca gráfica, outros sistemas operacionais ou qualquer outro programa aplicativo, e não se limitam a nenhum aplicativo ou sistema em particular.

Esta configuração básica é ilustrada na FIG. 3 por esses componentes dentro de uma linha tracejada 308.

O dispositivo de computação 300 pode ter recursos ou funcionalidades adicionais. Por exemplo, o dispositivo de computação 300 também
5 pode incluir dispositivos de armazenamento de dados adicionais (removíveis e/ou não-removíveis), tais como, por exemplo, discos magnéticos, discos ópticos ou fitas. Tal armazenamento adicional é ilustrado na FIG. 3 por um armazenamento removível 309 e por um armazenamento não-removível 310. Os meios de armazenamento em computador podem incluir meios volá-
10 teis e não-voláteis, removíveis e não-removíveis implementados em qualquer método ou tecnologia para o armazenamento de informações, tais como instruções legíveis por computador, estruturas de dados, módulos de programa ou outros dados. Todos dentro a memória do sistema 304, o armazenamento removível 309 e o armazenamento não-removível 310 são
15 exemplos de meios de armazenamento de computador (isto é, armazenamento em memória). Os meios de armazenamento para computador podem incluir, sem a isto se restringirem, RAM, ROM, memória somente para leitura eletricamente apagável (EEPROM), memória flash ou outra tecnologia de memória, CD-ROM, discos versáteis digitais (DVD) ou outro armazenamento
20 em disco óptico, cassetes magnéticos, fita magnética, armazenamento em disco magnético ou outros dispositivos de armazenamento magnético, ou qualquer outro meio que possa ser usado para ser usado para armazenar informações e que possa ser acessado pelo dispositivo de computação 300. Qualquer meio de armazenamento de computador desse tipo pode ser parte
25 do dispositivo 300. O dispositivo de computação 300 também pode ter dispositivo(s) de entrada 312, tal como um teclado, mouse, caneta, dispositivo de entrada de som, dispositivo de entrada por toque, etc. Também é possível incluir dispositivo(s) de saída 314, tal como uma tela, alto-falantes, uma impressora, etc. Os dispositivos supramencionados são apenas exemplos; é
30 possível empregar outros.

O dispositivo de computação 300 também pode conter uma conexão de comunicação 316 que possibilita que o dispositivo 300 se comuni-

que com outros dispositivos de computação 318, tal como por meio de uma rede em um ambiente de computação distribuída, por exemplo, uma intranet ou a Internet. A conexão de comunicação 316 é um exemplo de meio de comunicação. Normalmente, os meios de comunicação podem ser concreti-
5 zados por instruções legíveis por computador, estruturas de dados, módulos de programa ou outros dados em um sinal de dados modulado, tal como uma onda portadora ou outro mecanismo de transporte, além de incluírem quaisquer meios de distribuição de informações. O termo "sinal de dados modulados" pode descrever um sinal que uma ou mais características ajustadas ou alteradas de tal maneira a codificar as informações no sinal. A título
10 exemplificativo, e sem limitação, os meios de comunicação podem incluir meios com fio, tal como uma rede com fio ou conexão direta com fio, e meios sem fio, tal como acústico, radiofrequência, infravermelho e outros meios sem fio. O termo "meios legíveis por computador", conforme usado no presente documento, pode incluir tanto meios de armazenamento quanto meios
15 de comunicação.

Como mencionado acima, uma série de módulos de programa e arquivos de dados podem ser armazenados na memória do sistema 304, inclusive o sistema operacional 305. Enquanto são executados na unidade
20 de processamento 302, os módulos de programação 306 (por exemplo, o aplicativo do usuário 140) podem executar processos incluindo, por exemplo, um ou mais estágios do método 200, como descrito acima. O processo supramencionado é um exemplo, e a unidade de processamento 302 pode executar outros processos. Outros módulos de programação que podem ser
25 usados de acordo com as concretizações da presente invenção podem incluir aplicativos de contatos e correio eletrônico, aplicativos de processamento de texto, aplicativos de planilha, aplicativos de base de dados, aplicativos de apresentação de slides, programas aplicativos de desenho ou auxiliados por computador, etc.

30 Em geral, de acordo com as concretizações da invenção, os módulos de programa podem incluir rotinas, programas, componentes, estruturas de dados, e outros tipos de estruturas que efetuam tarefas específicas

ou implementar tipos de dados abstratos específicos. Além disso, as concretizações da invenção podem ser praticadas com outras configurações de sistema de computador, incluindo dispositivos portáteis, sistemas multiprocessadores, componentes eletrônicos programados pelo consumidor ou baseados em microprocessador, computadores de grande porte, entre outros. 5 As concretizações da invenção também podem ser praticadas em ambientes de computação distribuída, em que as tarefas são efetuadas por dispositivos de processamento remotos que estão ligados por meio de uma rede de comunicações. Em um ambiente de computação distribuída, os módulos de programa podem estar localizados tanto em dispositivos de armazenamento em memória local quanto remota. Além do mais, as concretizações da invenção podem ser praticadas em um circuito elétrico compreendendo elementos eletrônicos distintos, chips eletrônicos integrados contendo portas lógicas, um circuito utilizando um microprocessador, ou um único chip contendo elementos eletrônicos ou microprocessadores. As concretizações da invenção também podem ser praticadas usando outras tecnologias capazes de realizar operações lógicas, tal como, por exemplo, AND, OR e NOT, incluindo, mas sem se limitar a tecnologias mecânicas, ópticas, fluidicas e quânticas. Além disso, as concretizações da invenção podem ser praticadas em um computador de uso geral ou em quaisquer outros circuitos ou sistemas. 10 15 20 25

É possível implementar concretizações da invenção, por exemplo, como um processo de computador (método), um sistema de computação ou como um artigo de manufatura, tal como um produto de programa de computador ou meio legível por computador. O produto de programa de computador pode ser um meio de armazenamento de computador legível por um sistema de computador e codificando um programa de computador de instruções para executar um processo de computador. O produto de programa de computador também pode ser um sinal transmitido em um portador legível por um sistema de computador e codificando um programa de computador de instruções para executar um processo de computador. Sendo assim, a presente invenção pode ser incorporada em hardware e/ou em 30

software (incluindo firmware, software residente, microcódigo, etc.). Em outras palavras, as concretizações da presente invenção podem assumir a forma de um produto de programa de computador em um meio de armazenamento legível por computador ou utilizável por computador contendo um

5 código de programa legível por computador ou utilizável por computador incorporado no meio para uso por ou em conexão com um sistema de execução de instruções. Um meio "utilizável por computador" ou "legível por computador" pode ser qualquer meio capaz de armazenar, comunicar, propagar ou transportar o programa para uso por ou em relação ao sistema, aparelho

10 ou dispositivo de execução de instruções.

O meio legível por computador ou utilizável por computador pode ser, por exemplo, mas sem a isto se restringir, um sistema, aparelho, dispositivo ou meio de propagação eletrônico, magnético, óptico, eletromagnético, infravermelho ou semiconductor. Mais exemplos de meios legíveis por

15 computador específicos (uma lista não exaustiva) e meios legíveis por computador específicos podem incluir os seguintes: uma conexão elétrica com um ou mais fios, um disquete portátil de computador, uma memória de acesso aleatório (RAM), uma memória somente para leitura (ROM), uma memória somente para leitura programável apagável (EPROM ou memória Flash),

20 uma fibra óptica e um disco compacto de memória somente para leitura (CD-ROM). Observe que o meio legível por computador ou utilizável por computador poderia ser até mesmo papel ou outro meio adequado no qual o programa fosse impresso, uma vez que o programa poderia ser capturado eletronicamente, por exemplo, por um meio de digitalização óptica do papel ou

25 outro meio, para ser então convertido, interpretado ou processado de alguma outra forma de uma maneira adequada, se necessário, e, então, armazenado na memória do computador.

As concretizações da presente invenção, por exemplo, são descritas acima com referência aos diagramas de blocos e/ou ilustrações operacionais dos métodos, sistemas e produtos de programa de computador de

30 acordo com as concretizações da invenção. As funções/atos observados nos blocos podem ocorrer fora da ordem apresentada em qualquer fluxograma.

Por exemplo, dois blocos apresentados em sucessão podem, de fato, ser executados de forma substancialmente simultânea, ou, algumas vezes, os blocos podem ser executados na ordem inversa, dependendo da funcionalidade / dos atos envolvidos.

5 Embora certas concretizações da invenção tenham sido descritas, podem existir outras concretizações. Além disso, embora as concretizações da presente invenção tenham sido descritas como estando associadas aos dados armazenados em memória e em outros meios de armazenamento, os dados também podem ser armazenados ou lidos de outros meios legíveis por computador, tais como dispositivos de armazenamento secundários, como discos rígidos, discos flexíveis ou um CD-ROM, uma onda portadora da Internet ou outras formas de RAM ou ROM. Além disso, os estágios dos métodos revelados podem ser modificados de qualquer maneira, inclusive mediante a reordenação dos estágios e/ou pela inserção ou exclusão de estágios, sem divergir da invenção.

10

15

 Todos os direitos, inclusive direitos autorais no código, aqui incluídos, pertencem ao Titular e são propriedades deste. O Titular retém e preserva todos os direitos sobre o código aqui incluso, e concede permissão de reprodução do material somente em conjunto com a reprodução da patente concedida e não para outros fins.

20

 Embora o relatório descritivo inclua exemplos, o âmbito da invenção é indicado pelas reivindicações a seguir. Além do mais, embora o relatório descritivo tenha sido descrito numa linguagem específica aos aspectos estruturais e/ou etapas metodológicas, as reivindicações não se limitam aos aspectos ou etapas descritos acima. Em vez disso, os aspectos e procedimentos específicos descritos acima são revelados como exemplos das concretizações da invenção.

25

REIVINDICAÇÕES

1. Método (200) para proporcionar otimização da plataforma de visualização (150), **caracterizado pelo fato de que** compreende as etapas de:

- 5 definir (210) uma visualização;
criar (220) uma pluralidade de vetores geométricos;
receber (230) uma solicitação para exibir a visualização;
iterar (240) através da pluralidade de vetores geométricos (135);
saltar pelo menos um primeiro vetor geométrico (135) da plurali-
10 dade de registros geométricos de acordo com um algoritmo de salto geomé-
trico compreendendo pelo menos um dos seguintes critérios: detecção de
linha de comprimento zero, um número total da pluralidade de vetores geo-
métricos, um número total de pontos de dados, um tamanho de exibição, um
tipo de dado e um tipo de visualização;
15 traduzir pelo menos um segundo vetor da pluralidade de regis-
tros geométricos em pelo menos uma instrução de desenho; e
exibir (250) a visualização em um dispositivo de exibição (150).

2. Método, de acordo com a reivindicação 1, **caracterizado pelo fato de que** ainda compreende as etapas de:

- 20 determinar se pelo menos um segundo vetor da pluralidade de
vetores geométricos compreende uma localização de pixel associada a pelo
menos um terceiro vetor da pluralidade de vetores geométricos; e
em resposta à determinação de que pelo menos um segundo ve-
tor da pluralidade de vetores geométricos compreende a localização de pixel
25 associada com pelo menos um segundo vetor da pluralidade de vetores ge-
ométricos, saltar pelo menos um segundo vetor da pluralidade de vetores
geométricos (135).

3. Método, de acordo com a reivindicação 1, **caracterizado pelo fato de que** ainda compreende as etapas de:

- 30 enviar uma pluralidade dos vetores geométricos traduzidos a um
aplicativo (140); e
renderizar a pluralidade dos vetores geométricos traduzidos para

exibição pelo aplicativo (140).

4. Método, de acordo com a reivindicação 3, **caracterizado pelo fato de que** ainda compreende as etapas de:

5 receber uma solicitação a partir do aplicativo (140) por um lote de vetores geométricos traduzidos;

enviar um subconjunto da pluralidade de vetores geométricos traduzidos ao aplicativo (140), em que o subconjunto da pluralidade de vetores geométricos traduzidos compreende uma série de registros geométricos traduzidos associados à solicitação de lote recebida; e

10 aguardar para enviar um subconjunto restante da pluralidade de vetores geométricos traduzidos.

5. Método, de acordo com a reivindicação 1, **caracterizado pelo fato de que** a etapa de traduzir pelo menos um da pluralidade de vetores geométricos em pelo menos uma instrução de desenho compreende linearizar pelo menos um da pluralidade de vetores geométricos (135).

6. Método, de acordo com a reivindicação 1, **caracterizado pelo fato de que** ainda compreende a etapa de:

20 receber pelo menos uma instrução de desenho suportada, em que traduzir pelo menos um da pluralidade de vetores geométricos em pelo menos uma instrução de desenho compreende traduzir pelo menos um da pluralidade de vetores geométricos em pelo menos uma instrução de desenho suportada.

7. Sistema (300) para proporcionar otimização da plataforma de visualização (150) **caracterizado pelo fato de que** compreende:

25 um armazenamento em memória (309, 310); e

uma unidade de processamento (320) acoplada ao armazenamento em memória (309, 310), em que a unidade de processamento (302) é operativa para:

30 criar (210) um objeto de visualização associado a uma pluralidade de valores de dados,

compor (220) pelo menos um primeiro valor da pluralidade de valores de dados em um vetor geométrico associado ao

objeto de visualização,

saltar a composição de pelo menos um segundo valor da pluralidade de valores de dados de acordo com um critério predefinido, e

5 traduzir pelo menos um subconjunto da pluralidade de vetores geométricos (135) em uma pluralidade de instruções de desenho, em que o subconjunto da pluralidade de vetores geométricos (135) é selecionado de acordo com um algoritmo de salto geométrico compreendendo pelo menos um dos seguintes
10 critérios: detecção de linha de comprimento zero, um número total da pluralidade de vetores geométricos, um número total de pontos de dados, um tamanho de exibição, um tipo de dado e um tipo de visualização.

8. Sistema, de acordo com a reivindicação 7, **caracterizado pelo fato de que** o critério predefinido é configurável pelo usuário.

9. Sistema, de acordo com a reivindicação 7, **caracterizado pelo fato de que** o algoritmo de salto geométrico é ainda é operativo para:

traduzir pelo menos um primeiro vetor da pluralidade de registros geométricos em pelo menos uma instrução de desenho;

20 iterar pelo menos um segundo vetor da pluralidade de vetores geométricos;

determinar se pelo menos um segundo vetor da pluralidade de vetores geométricos compreende um ponto no objeto de visualização próximo para o ponto no objeto de visualização associado com pelo menos um
25 primeiro vetor; e

em resposta à determinação de que pelo menos um segundo vetor da pluralidade de vetores geométricos compreende o ponto no objeto de visualização próximo para o ponto no objeto de visualização associado com pelo menos um primeiro vetor, saltar a tradução de pelo menos um segundo
30 vetor em pelo menos uma instrução de desenho.

10. Sistema, de acordo com a reivindicação 7, **caracterizado pelo fato de que** o algoritmo de salto geométrico é ainda é operativo para:

receber pelo menos uma mudança no objeto de visualização; e em resposta a recepção de pelo menos uma mudança no objeto de visualização, renderizar novamente o objeto de visualização.

5 11. Sistema, de acordo com a reivindicação 10, **caracterizado pelo fato de que** a pelo menos uma mudança no objeto de visualização compreende pelo menos um dos seguintes: redimensionar o objeto de visualização, aumentar o objeto de visualização, diminuir o objeto de visualização, um movimento do objeto de visualização, uma cobertura parcial do objeto de visualização, e um movimento de um aplicativo.

10 12. Sistema, de acordo com a reivindicação 7, **caracterizado pelo fato de que** ainda compreende um aplicativo de usuário operativo para renderizar o objeto de visualização para um dispositivo de exibição.

15 13. Meio legível por computador **caracterizado pelo fato de que** armazena um conjunto de instruções que, quando executadas, realizam um método (200) para otimizar uma plataforma de visualização (150), o método (200) sendo executado pelo conjunto de instruções compreendendo:

20 receber (210) uma solicitação a partir de um aplicativo do usuário (140) para criar um objeto de visualização, em que a solicitação compreende uma pluralidade de valores de dados e um tipo de visualização e em que o objeto de visualização compreende uma representação bidimensional de uma série de pontos de dados como um compreendendo um valor associado;

25 compor (220) uma pluralidade de vetores geométricos (135) associados com o objeto de visualização, em que compor a pluralidade de vetores geométricos (135) compreende:

30 compor pelo menos um vetor geométrico associado com pelo menos um da pluralidade de valores de dados, e saltar pelo menos um da pluralidade de valores de dados de acordo com um critério compreendendo pelo menos um dos seguintes: um tamanho de exibição, um tipo de dado, um número total de pontos de dados a ser exibido, um número total da pluralidade de vetores geométricos (135), e um tamanho de

uma quantidade de sobreposição entre os objetos de exibição;
receber (230) uma solicitação de renderização a partir do aplicativo do usuário (140);

traduzir cada um de um subconjunto da pluralidade de vetores geométricos (135) em pelo menos uma instrução de desenho, em que o subconjunto da pluralidade de vetores geométricos (135) é selecionado de acordo com um algoritmo de salto de geometria (130) compreendendo o critério de pelo menos um dos seguintes: a detecção de linha de comprimento zero, o número total da pluralidade de vetores geométricos, o número total de pontos de dados, o tamanho de exibição, o tipo de dado e o tipo de visualização;

enviar pelo menos uma instrução de desenho associada a cada um do subconjunto da pluralidade de vetores geométricos (135) ao aplicativo do usuário (140);

renderizar (250) o objeto de visualização, em que a renderização do objeto de visualização compreende renderizar pelo menos uma instrução de desenho associada a cada um do subconjunto da pluralidade de vetores geométricos (135) para um dispositivo de exibição (150);

receber uma segunda solicitação de renderização a partir do aplicativo do usuário (140) em resposta a uma alteração detectada que afeta o objeto de visualização renderizado;

traduzir cada um de um segundo subconjunto da pluralidade dos vetores geométricos (135) em pelo menos uma instrução de desenho;

enviar pelo menos uma instrução de desenho associada a cada um do segundo subconjunto da pluralidade de vetores geométricos (135) ao aplicativo do usuário (140); e

renderizar novamente o objeto de visualização.

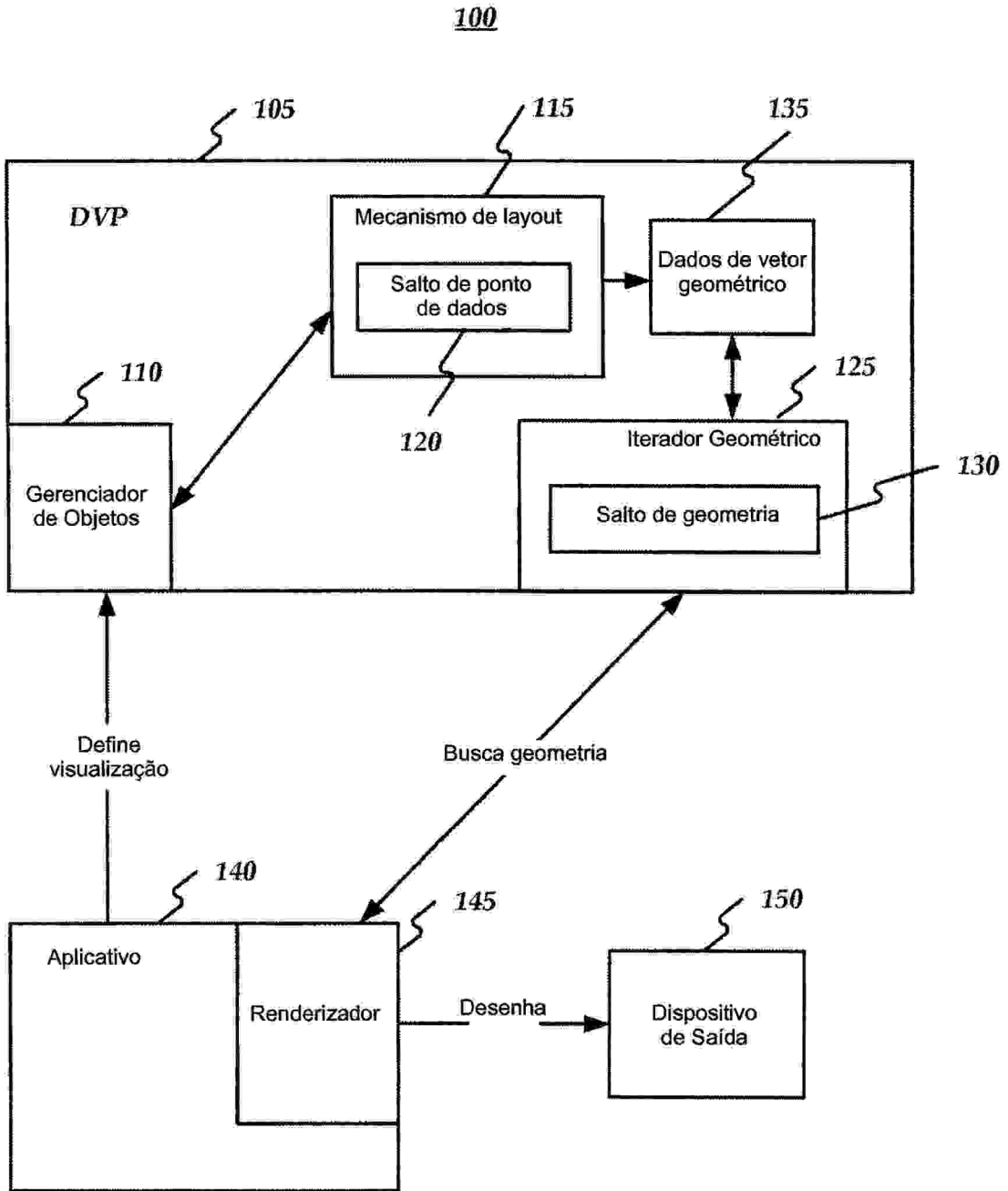
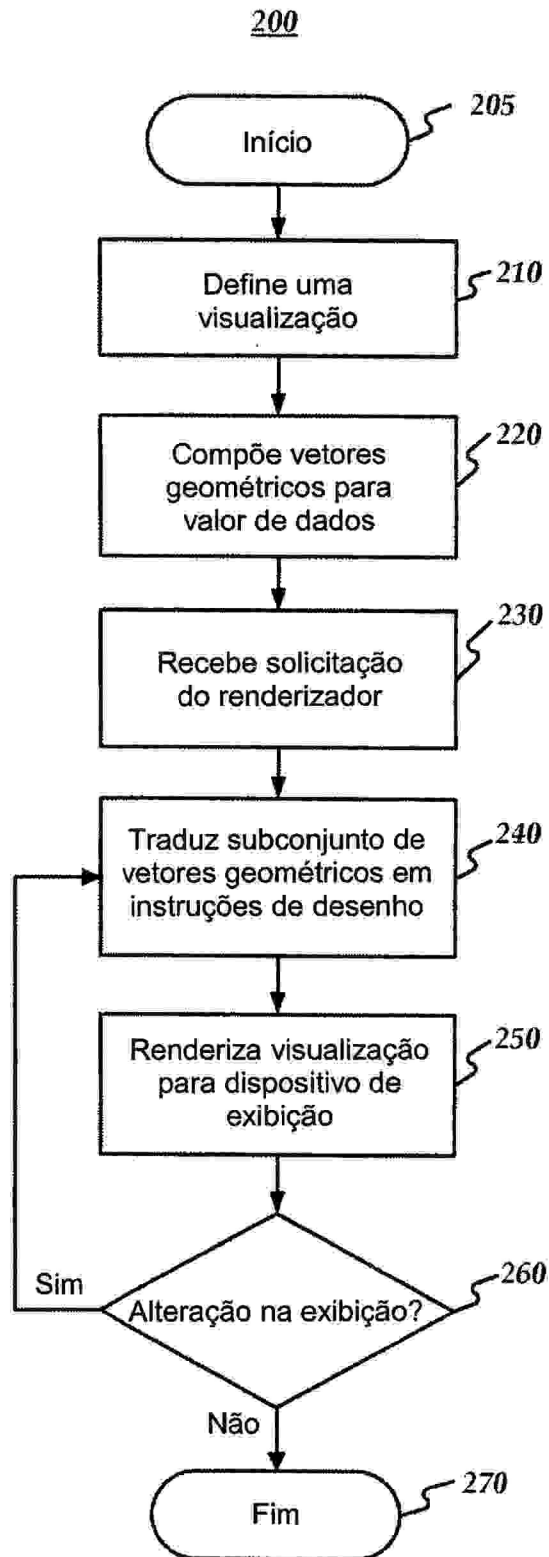


FIG. 1

**FIG. 2**

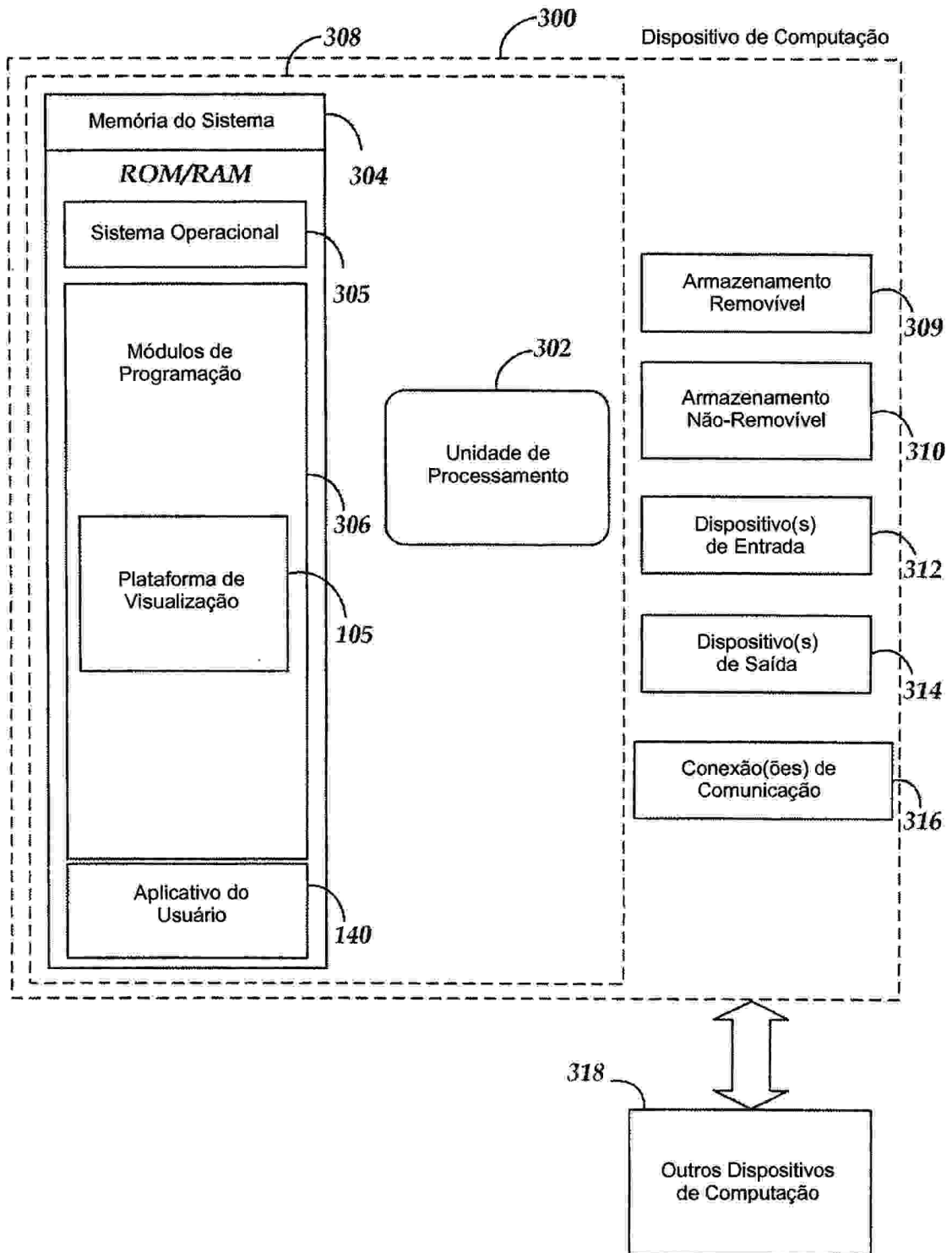


FIG. 3

RESUMO

Patente de Invenção: **"MÉTODO E SISTEMA PARA PROPORCIONAR OTIMIZAÇÃO DA PLATAFORMA DE VISUALIZAÇÃO E MEIO LEGÍVEL POR COMPUTADOR"**.

5 A presente invenção proporciona otimização da plataforma de
visualização de dados. Os aplicativos podem fornecer valores de dados e
solicitar a criação de uma visualização a partir de uma plataforma de visuali-
zação de dados (DVP). O DVP pode compor uma pluralidade de registros
geométricos associados a um subconjunto dos valores de dados de visuali-
10 zação. O aplicativo pode processar a visualização iterando através dos veto-
res geométricos e traduzindo um subconjunto dos vetores em instruções de
desenho para saída em um dispositivo de exibição.