



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

(11) PI 0906256-4 B1



(22) Data do Depósito: 07/01/2009

(45) Data de Concessão: 26/11/2019

(54) Título: MEIO DE ARMAZENAMENTO EM COMPUTADOR TENDO INSTRUÇÕES EXECUTÁVEIS POR COMPUTADOR INCORPORADAS NO MESMO, SISTEMA DE COMPUTADOR PARA EXECUTAR UM MÉTODO DE INICIAÇÃO DE UM APLICATIVO DE SERVIÇO E MÉTODO COMPUTADORIZADO PARA CONFIGURAR PROGRAMAS DE COMPONENTE DE UM APLICATIVO DE SERVIÇO

(51) Int.Cl.: G06F 9/44; G06F 15/16; G06F 13/14; G06F 9/06.

(30) Prioridade Unionista: 30/01/2008 US 12/022.756.

(73) Titular(es): MICROSOFT TECHNOLOGY LICENSING, LLC.

(72) Inventor(es): JOSE BERNABEU-AUBAN; YOUSEF A. KHALIDI.

(86) Pedido PCT: PCT US2009030283 de 07/01/2009

(87) Publicação PCT: WO 2009/099694 de 13/08/2009

(85) Data do Início da Fase Nacional: 29/06/2010

(57) Resumo: MEIO DE ARMAZENAMENTO EM COMPUTADOR TENDO INSTRUÇÕES EXECUTÁVEIS POR COMPUTADOR INCORPORADAS NO MESMO, SISTEMA DE COMPUTADOR PARA EXECUTAR UM MÉTODO DE INICIAÇÃO DE UM APLICATIVO DE SERVIÇO E MÉTODO COMPUTADORIZADO PARA CONFIGURAR PROGRAMAS DE COMPONENTE DE UM APLICATIVO DE SERVIÇO Métodos, sistemas e meios de armazenamento em computador possuindo instruções executáveis por computador consubstanciadas nos mesmos que, quando executadas, realizam os métodos de acordo com as modalidades, para o gerenciamento de programas de componente (201, 202, 203, 204) dentro de um aplicativo de serviço (210) de acordo com um modelo de serviço. Inicialmente, as configurações, que podem ser administradas para os programas de componente (201, 202, 203, 204), são extraídas a partir do modelo de serviço. Um processo de iniciação é iniciado para gerenciar o aplicativo de serviço (210) de acordo com as mudanças para um nível de carga no aplicativo de serviço (210) e/ou para especificações dentro do modelo de serviço. O processo de iniciação inclui o desenvolvimento, ou identificação como disponível, dos programas de componente (201, 202, 203, 204) dentro do centro de dados (225), e a propagação automática dos valores formalizados derivados das configurações por todos os programas de componente (201, 202, 203, 204) desenvolvidos fornecendo, assim, a funcionalidade do aplicativo de serviço (210) consistente com o modelo de serviço.

"MEIO DE ARMAZENAMENTO EM COMPUTADOR TENDO INSTRUÇÕES EXECUTÁVEIS POR COMPUTADOR INCORPORADAS NO MESMO, SISTEMA DE COMPUTADOR PARA EXECUTAR UM MÉTODO DE INICIAÇÃO DE UM APLICATIVO DE SERVIÇO E MÉTODO COMPUTADORIZADO PARA CONFIGURAR PROGRAMAS DE COMPONENTE DE UM APLICATIVO DE SERVIÇO"

FUNDAMENTOS

[001] Tipicamente, os aplicativos de software são escritos para permitir muitos graus de liberdade em sua configuração. Quando destacando essa característica, vários usuários são capazes de configurar um aplicativo de software em particular para realizar de forma que seja específica a cada um dos usuários. Dessa forma, essa liberdade incorporada a um único aplicativo de software permite que o aplicativo de software produza resultados diferentes.

[002] Esse tipo de aplicativo de software é utilizado por provedores de serviço de aplicativo (ASPs) que permitem que os usuários manipulem remotamente o aplicativo através da Internet. Visto que o aplicativo de software inclui graus de liberdade, o usuário pode fornecer exigências de desempenho para os ASPs para programar manualmente o aplicativo. Adicionalmente, a maior parte dos aplicativos de software inclui elementos subjacentes separados que devem ser identificados individualmente e manualmente programados de modo que o aplicativo de software possa expressar as exigências de desempenho. O processo de programação manual dos elementos subjacentes de acordo com as exigências de desempenho recebidas é demorado e tende a erro. De acordo, esse método ad hoc para personalização de um aplicativo de software introduz a fragilidade no fornecimento do aplicativo de software do ASPs. Essas desvantagens de programação manual são exageradas quando vários usuários estão continuamente

oferecendo, e mudando, suas exigências de desempenho respectivas.

SUMÁRIO

[003] Esse sumário é fornecido para introduzir conceitos de uma forma simplificada que são descritos abaixo na Descrição Detalhada. Esse sumário não pretende identificar características chave e características essenciais da presente matéria reivindicada, nem deve ser utilizado como um auxílio na determinação do escopo da presente matéria reivindicada.

[004] As modalidades da presente invenção fornecem uma abordagem de modelagem de serviço que reduz as desvantagens da programação manual pelo fornecimento de uma estrutura (por exemplo, esquema de iniciação) para o gerenciamento automático dos programas de componente dentro de um aplicativo de serviço de acordo com um modelo de serviço. Inicialmente, um gatilho para iniciar os programas de componente dentro do aplicativo de serviço é detectado. Nas modalidades, o gatilho é baseado em uma indicação de uma mudança para um nível de carga no aplicativo de serviço (por exemplo, pelo monitoramento de tráfego em um sítio da rede hospedando o aplicativo de serviço) ou uma indicação de uma mudança nas especificações dentro de um modelo de serviço (por exemplo, emendando um contrato de descrição de serviço estabelecido entre um provedor de serviço de aplicativo e um cliente).

[005] Incidente à detecção de gatilho, as configurações são extraídas das especificações do modelo de serviço, onde as configurações (por exemplo, endereços de programas de componentes, identificadores de algoritmo, localizações de serviço, endereços de recurso de memória, e similares) são utilizadas para configurar os parâmetros dentro dos programas de componente. Em um caso, os programas de componente são desenvolvidos em vários dispositivos de computação dentro de um centro de dados que executa o aplicativo de serviço. Em

outro caso, os programas de componente estabelecidos anteriormente residentes em vários dispositivos de computação são identificados como estando disponíveis para iniciação. Esses programas de componente desenvolvidos e/ou identificados são configurados de acordo com um processo de iniciação que inclui as etapas lógicas a seguir: acesso a mapas (por exemplo, mapa de transformação, mapa de estrutura), derivação de valores formalizados a partir das configurações de acordo com o mapa de transformação, propagação automática de valores formalizados através dos programas de componente, localização de parâmetros correspondentes aos valores formalizados de acordo com o mapa de estrutura, e instalação dos valores formalizados nos parâmetros adequados. Esse processo de iniciação robusto configura os programas de componente automaticamente, dessa forma, reduzindo substancialmente as desvantagens (discutidas acima) inerentes à programação manual.

BREVE DESCRIÇÃO DOS DESENHOS

[006] As modalidades da presente invenção são descritas em detalhes abaixo com referência às figuras dos desenhos em anexo, onde:

[007] A figura 1 é um diagrama em bloco de um ambiente de computação ilustrativo adequado para uso na implementação de modalidades da presente invenção;

[008] A figura 2 é um diagrama em bloco de um sistema de computação ilustrativo adequado para uso na implementação de modalidades da presente invenção;

[009] A figura 3 é um fluxograma ilustrando um método de gerenciamento de componentes dentro de um aplicativo de serviço de acordo com um modelo de serviço, de acordo com uma modalidade da presente invenção;

[0010] A figura 4 e a figura 5 são fluxogramas ilustrando métodos

para a instalação dos valores formalizados em parâmetros adequados dos programas de componente; e

[0011] A figura 6 é um fluxograma de um método para responder a um evento de acionamento, de acordo com uma modalidade da presente invenção.

DESCRÍÇÃO DETALHADA

[0012] A presente matéria das modalidades da presente invenção é descrita com especificidade para corresponder às exigências. No entanto, a descrição propriamente dita não pretende limitar o escopo dessa patente. Ao invés disso, os inventores contemplaram que a presente matéria reivindicada também pode ser consubstanciada de outras formas, para incluir diferentes etapas ou combinações de etapas similares às descritas nesse documento, em conjunto com outras tecnologias atuais ou futuras. Ademais, apesar de os termos "etapa" e/ou "bloco" poderem ser utilizados aqui para conotar diferentes elementos de métodos empregados, os termos não devem ser interpretados como implicando em qualquer ordem particular dentre ou entre várias etapas aqui descritas a menos e exceto quando a ordem das etapas individuais é explicitamente descrita.

[0013] As modalidades da presente invenção se referem aos métodos, sistemas e mídia de armazenamento em computador possuindo instruções executáveis por computador consubstanciadas nos mesmos que quando executadas, realizam os métodos de acordo com as modalidades, para gerenciar automaticamente os programas de componente dentro de um aplicativo de serviço de acordo com um modelo de serviço. Inicialmente, um gatilho para iniciar os programas de componente dentro do aplicativo de serviço é detectado. Nas modalidades, o gatilho é baseado em uma indicação de uma mudança para um nível de carga no aplicativo de serviço (por exemplo, pelo monitoramento do tráfego em um sítio da rede hospedando o aplicativo de serviço), ou

uma indicação de uma mudança para as especificações dentro de um modelo de serviço (por exemplo, por meio de emendas ao contrato de descrição de serviço estabelecido entre um provedor de serviço de aplicativo e um cliente).

[0014] Incidente à detecção de gatilho, as configurações são extraídas a partir das especificações do modelo de serviço, onde as configurações (por exemplo, endereços dos programas de componente, identificadores de algoritmo, localizações de serviço, endereços de recurso de memória, e similares) são utilizadas para configurar os parâmetros dentro dos programas de componente. Em um caso, os programas de componente são desenvolvidos em vários dispositivos de computação dentro de um centro de dados que executa o aplicativo de serviço. Em outro caso, os programas de componente previamente estabelecidos residentes em vários dispositivos de computação são identificados como estando disponíveis para iniciação. Esses programas de componente desenvolvidos e/ou identificados são configurados de acordo com um processo de iniciação que inclui as seguintes etapas lógicas: acesso a mapas (por exemplo, mapa de transformação, mapa de estrutura) derivando os valores formalizados a partir das configurações de acordo com o mapa de transformação, propagação automática dos valores formalizados por todos os programas de componente, localização de parâmetros correspondentes aos valores formalizados de acordo com o mapa de estrutura, e instalação dos valores formalizados nos parâmetros adequados.

[0015] De acordo, em um aspecto, as modalidades da presente invenção se referem a um ou mais meios de armazenamento por computador possuindo instruções executáveis por computador consubstanciadas no mesmo que, quando executadas, realizam um método de gerenciamento de um ou mais programas de componente dentro de um aplicativo de serviço de acordo com um modelo de serviço.

O método inclui a determinação de um nível de carga no aplicativo de serviço, onde o aplicativo de serviço inclui os programas de componente que são executáveis em dispositivos de computação separados dentro de um centro de dados, extraíndo da configuração de modelo de serviço configurações que podem ser administradas para os programas de componente, e iniciando um processo de iniciação para o gerenciamento do aplicativo de serviço de acordo com o nível de carga. O processo de iniciação inclui o desenvolvimento de programas de componente dentro do centro de dados, a propagação automática de valores formalizados derivados das configurações por todos os programas de componente desenvolvidos, e pelo menos a instalação temporária dos valores formalizados em programas de componente, fornecendo, assim, a funcionalidade do aplicativo de serviço consistente com o modelo de serviço. Nas modalidades, o método inclui adicionalmente a execução do aplicativo de serviço de acordo com os valores formalizados instalados.

[0016] Em outro aspecto, as modalidades da presente invenção se referem a um sistema de computador consubstanciado em um ou mais meios de armazenamento em computador possuindo instruções executáveis por computador consubstanciadas para realizar um método de iniciação de um aplicativo de serviço pela propagação das configurações para os programas de componente, promovendo, assim, a funcionalidade do aplicativo de serviço. O sistema de computador inclui um primeiro dispositivo de computação e um segundo dispositivo de computação. O primeiro dispositivo de computação executa os programas de componente que suporta a operação do aplicativo de serviço. O segundo dispositivo de computação inicia um processo de iniciação para implementação do aplicativo de serviço de acordo com um modelo de serviço. O processo de iniciação inclui a extração das configurações a partir das especificações dentro do modelo de serviço,

identificando os programas de componente residentes no primeiro dispositivo de computação como estando disponíveis para iniciação, interrogando os parâmetros dentro dos programas de componente para estabelecer se alguns dos programas de componente são compatíveis com as configurações, propagando automaticamente os valores formalizados derivados das configurações por todos os programas de componente compatíveis, e configurando os parâmetros dos programas de componente compatíveis de acordo com os valores formalizados.

[0017] Em outro aspecto, as modalidades da presente invenção se referem a um método computadorizado para configurar os programas de componente de um aplicativo de serviço, operando em um centro de dados, de acordo com um esquema de iniciação. O método computadorizado inclui, pelo menos, as seguintes etapas: determinação das configurações pela interrogação de um modelo de serviço; acesso ao esquema de iniciação, onde o esquema de iniciação indica quais programas de componente desenvolver a fim de satisfazer o modelo de serviço, e onde o esquema de iniciação fornece um mapa de transformação; desenvolvendo os programas de componente indicados em uma pluralidade de dispositivos de computação dentro do centro de dados; convertendo as configurações em valores formalizados com base, em parte, no mapa de transformação; distribuindo automaticamente os valores formalizados para os programas de componente desenvolvidos; e pelo menos armazenando temporariamente os valores formalizados distribuídos em associação com os programas de componente desenvolvidos.

[0018] Tendo descrito de forma breve uma visão geral das modalidades da presente invenção, um ambiente operacional ilustrativo adequado para a implementação de modalidades da presente invenção é descrita abaixo.

[0019] Com referência aos desenhos em geral, e inicialmente à

figura 1 em particular, um ambiente operacional ilustrativo para implementação das modalidades da presente invenção é ilustrado e designado geralmente como o dispositivo de computação 100. O dispositivo de computação 100 é apenas um exemplo de um ambiente de computação adequado e não deve sugerir qualquer limitação ao escopo do uso ou funcionalidade das modalidades da presente invenção. Nem deve o ambiente de computação 100 ser interpretado como tendo qualquer dependência ou exigência com relação a qualquer um ou a uma combinação dos componentes ilustrados.

[0020] As modalidades da presente invenção podem ser descritas no contexto geral do código de computador ou instruções utilizáveis por máquina, incluindo instruções executáveis por computador tal como componentes de programa, sendo executados por um computador ou outra máquina, tal como um assistente de dados pessoal ou outro dispositivo portátil. Geralmente, os componentes de programa incluem rotinas, programas, objetos, componentes, estruturas de dados e similares se referem ao código que realiza tarefas particulares, ou implementam tipos de dados abstratos particulares. As modalidades da presente invenção podem ser praticadas em uma variedade de configurações de sistema, incluindo dispositivos portáteis, partes eletrônicas de consumidor computadores de finalidade geral, dispositivos de computação especializados, etc. As modalidades da invenção também podem ser praticadas em ambientes de computação distribuídos onde tarefas são realizadas por dispositivos de processamento remoto que são conectados através de uma rede de comunicações.

[0021] Com referência continuada à figura 1, o dispositivo de computação 100 inclui um barramento 110 que acopla direta ou indiretamente os seguintes dispositivos: memória 112, um ou mais processadores 114, um ou mais componentes de apresentação 116, portas de entrada/saída (I/O) 118, componentes I/O 120, e um suprimento de

potência ilustrativo 122. O barramento 110 representa o que pode ser um ou mais barramentos (tal como um barramento de endereço, barramento de dados, ou combinação dos mesmos). Apesar de os vários blocos da figura 1 serem ilustrados com linhas para fins de brevidade, em realidade, a delinearção de vários componentes não está tão clara, e metaforicamente, as linhas devem ser mais precisamente cinza e confusas. Por exemplo, pode-se considerar um componente de apresentação tal como um dispositivo de exibição como sendo um componente I/O. Além disso, os processadores possuem memória. Reconhece-se, portanto que tal é a natureza da técnica e reitera-se que o diagrama da figura 1 é meramente ilustrativo de um dispositivo de computação ilustrativo que pode ser utilizado com relação a uma ou mais modalidades da presente invenção. A distinção não é feita entre tais categorias como "estação de trabalho", "laptop", "dispositivo portátil", etc., visto que todos são contemplados dentro do escopo da figura 1 e referência a "computador" ou "dispositivo de computação".

[0022] O dispositivo de computação 100 inclui tipicamente uma variedade de mídia legível por computador. Por meio de exemplo, e não de limitação, a mídia legível por computador pode compreender Memória de Acesso Randômico (RAM); Memória de Leitura Apenas (ROM); Memória de Leitura Apenas Eletronicamente Eliminável e Programável (EEPROM); memória flash ou outras tecnologias de memória; CDROM; discos versáteis digitais (DVDs) ou outra mídia ótica ou holográfica; fitas magnéticas, armazenamento em disco magnético ou outros dispositivos de armazenamento magnético, ou qualquer outro meio que possa ser utilizado para codificar informação desejada e seja acessado por dispositivo de computação 100.

[0023] A memória 112 inclui a mídia de armazenamento em computador na forma de memória volátil e/ou memória não volátil. A memória pode ser removível, não removível, ou uma combinação das

mesmas. Os dispositivos de hardware ilustrativos incluem memória de estado sólido, discos rígidos, acionadores de disco ótico, etc. O dispositivo de computação 100 inclui um ou mais processadores que podem ler dados a partir de várias entidades tal como a memória 112 ou componentes I/O 120. Os componentes de apresentação 116 apresentam indicações de dados para um usuário ou outro dispositivo. Os componentes de apresentação ilustrativos incluem um dispositivo de exibição, alto falante, componente de impressão, componente de vibração, etc. Portas I/O 118 permitem que o dispositivo de computação 100 seja logicamente acoplado a outros dispositivos incluindo componentes I/O 120, alguns dos quais podem ser embutidos. Componentes ilustrativos incluem um microfone, joystick, parte de jogo, satélite, digitalizador, impressora, dispositivo sem fio, etc.

[0024] Voltando-se agora à figura 2, um diagrama em bloco é ilustrado, de acordo com uma modalidade da presente invenção, ilustrando um sistema de computação 200 configurado para gerenciar os programas de componente 201, 202, 203, 204 dentro de um aplicativo de serviço 210 de acordo com um modelo de serviço. Será compreendido e apreciado pelos versados na técnica que o sistema de computação 200 ilustrado na figura 2 é meramente um exemplo de um ambiente de sistema de computação adequado e não pretende sugerir qualquer limitação ao escopo de uso ou funcionalidade das modalidades da presente invenção. Nem deve o sistema de computação 200 ser interpretado como tendo qualquer dependência ou exigência relacionada com qualquer componente singular ou combinação de componentes ilustrados aqui. Adicionalmente, apesar de vários blocos da figura 2 serem ilustrados com linhas por motivos de clareza, na realidade, a delinearção de vários componentes não está clara, e metaforicamente, as linhas seriam mais precisamente cinza e confusas.

[0025] O sistema de computação 200 inclui um dispositivo de

computação de cliente 215, provedores de serviço dependentes 220, e um centro de dados 225, todos em comunicação um com o outro através de uma rede (não ilustrada). A rede pode incluir, sem limitação, uma ou mais redes de área local (LANs) e/ou redes de área ampla (WANs). Tais ambientes de rede são lugar comum em escritórios, redes de computador empresariais, intranets e na Internet. De acordo, a rede não é descrita adicionalmente aqui.

[0026] Cada um dos dispositivos de computação de cliente 215, provedores de serviço dependentes 220, e centro de dados 225 ilustrados na figura 2 pode ser qualquer tipo de dispositivo de computação, tal como, por exemplo, dispositivo de computação 100 descrito acima com referência à figura 1. Por meio de exemplo apenas, e não de limitação, cada um dos dispositivos de computação de cliente 215, provedores de serviço dependentes 220 pode ser um computador pessoal, um computador desktop, um computador laptop, um dispositivo portátil, um aparelho móvel, um dispositivo eletrônico de consumidor, e similar. Nas modalidades, o centro de dados 225 pode ser um servidor, uma coleção de dispositivos de computação, ou outras máquinas capazes de executar o aplicativo de serviço 210. Adicionalmente, o dispositivo de computação de cliente 215 pode incluir adicionalmente um aplicativo de navegador de rede 230 residente no mesmo para acessar o aplicativo de serviço 210 através de percursos de rede com ou sem fio para o centro de dados 225. Deve-se notar, no entanto, que as modalidades da presente invenção não são limitadas à implementação de tais dispositivos de computação, mas pode ser implementada em qualquer um dentre uma variedade de diferentes tipos de dispositivos dentro do escopo das modalidades.

[0027] Como ilustrado na figura 2, o dispositivo de computação de cliente 215 é configurado para executar o aplicativo de navegador de rede 230 para acessar o aplicativo de serviço 210. O aplicativo de ser-

viço 210 pode ser acessado de forma geralmente direta, através do percurso de comunicação 235, ou por meio de provedores de serviço dependentes 220. Nas modalidades, o acesso através dos provedores de serviço dependentes 220 inclui o transporte de credenciais fornecidas por usuário 240. Tipicamente, as credenciais fornecidas por usuário 240 são dados de conexão, senhas ou qualquer outra informação fornecida por um cliente para satisfazer o protocolo de segurança implementado em provedores de serviço dependentes 220. Por sua vez, os provedores de serviço dependentes 220 permitem credenciais 240, dentro das mensagens 245, com um ou mais componentes 201, 202, 203, 204.

[0028] Nas modalidades, os provedores de serviço dependentes 220 são localizados fora do centro de dados 225 e gerenciados por uma entidade separada de um provedor de serviço de aplicativo que mantém o aplicativo de serviço 210. No entanto, os provedores de serviço dependentes 220 são capazes de coordenar as atividades com os programas de componente 201, 202, 203, 204 permutando mensagens 245 entre os mesmos. A coordenação das atividades entre os provedores de serviço dependentes 220 e os programas de componente 201, 202, 203, 204 é facilitada por localizações de recurso desses provedores de serviço dependentes alvo 220. Essas localizações de recurso são tipicamente estabelecidas de acordo com as especificações dentro de um modelo de serviço.

[0029] Geralmente, o modelo de serviço é uma blueprint de interface que fornece instruções para o gerenciamento dos programas de componente 201, 202, 203, e 204 do aplicativo de serviço 210. De acordo, o modelo de serviço inclui especificações que identificam tais aspectos de gerenciamento quanto a quais mapas empregar durante a iniciação; quais dispositivos de computação utilizar durante a propagação de programas de componente 201, 202, 203 e 204; onde os per-

cursos de comunicação 246 e 247 entre os programas de componente 201, 202, 203, e 204 estão localizados; e qualquer outra informação que descreva a forma particular na qual o aplicativo de serviço 210 será executado pelo centro de dados 225. Em um caso, o modelo de serviço é gerado, em parte, de acordo com um contrato de descrição de serviço estabelecido entre um cliente e um provedor de serviço de aplicativo. Como resultado do contrato de descrição de serviço, o cliente é fornecido com acesso ao aplicativo de serviço 210, e outro software, mantido pelo provedor de serviço de aplicativo.

[0030] Adicionalmente, o modelo de serviço fornece ganchos dentro das especificações que permitem que as configurações implícitas e explícitas sejam extraídas a partir das mesmas. Nas modalidades, as configurações implícitas fornecem a informação estrutural (por exemplo, endereços de componente, localizações de recurso, e endereços de recurso de memória) para facilitar a interação entre cada um dos programas de componente 201, 202, 203, 204 e/ou outros componentes (por exemplo, provedores de serviços dependentes 220, recursos de memória 250). Adicionalmente, as configurações implícitas fornecem interfaces de programação de aplicativo (APIs) para consumo e tradução de mensagens entre os componentes acima, permitindo, assim, que os componentes se comuniquem harmoniosamente. Nas modalidades, as configurações explícitas fornecem instruções (por exemplo, identificadores de algoritmo e outros valores de configuração) para os parâmetros dentro dos programas de componente 201, 202, 203 e 204 que influenciam a funcionalidade. Por meio de exemplo apenas, as credenciais especificadas para arquivamento de um cliente no aplicativo de serviço 210 são memorizadas no modelo de serviço. Ganchos no modelo de serviço fornecem essas credenciais para as configurações explícitas, que distribuem as credenciais para os componentes 201 como valores esperados. Esses valores esperados são com-

parados com as credenciais fornecidas por usuário 240 que são portadas a partir dos provedores de serviço dependentes 220 na mensagem 245. Depois do recebimento da mensagem 245, o componente 201 compara as credenciais fornecidas por usuário 240 com os valores esperados. Se a comparação resultar em uma combinação, o cliente recebe acesso ao aplicativo de serviço 210.

[0031] O centro de dados 225 é fornecido para execução de aplicativo de serviço 210, dentre outras coisas. O aplicativo de serviço 210 inclui um conjunto de programas de componente 201, 202, 203 e 204 que podem ser escalonados para utilização (por exemplo, número de cliques de cliente por hora) ou pode emendas ao modelo de serviço. Tipicamente, o aplicativo de serviço 210 é caracterizado pelo fato de ser constituído de muitos componentes diferentes (por exemplo, programas de componente 201, 202, 203 e 204), que são similares aos aplicativos tradicionais. Em uma modalidade ilustrativa, o aplicativo de serviço 210 gerencia os diferentes componentes de modo que esses componentes funcionem juntos como um serviço de Internet combinado. Por meio de exemplo, o aplicativo de serviço 210 é um serviço de blog que permite que uma variedade de clientes publiquem comentários em um sítio de rede particular. Visto que pode haver potencialmente milhares de clientes, todos acessando o serviço de blog, múltiplos componentes são desenvolvidos pelo serviço de blog para suportar sua operação continuada. Esses componentes coordenam as ações um com o outro para funcionar como um único aplicativo, dessa forma, o desenvolvimento e eliminação de componentes é transparente para os clientes.

[0032] Apesar de um caso de aplicativo de serviço 210 ter sido ilustrado e descrito, deve ser compreendido e apreciado pelos versados na técnica que múltiplos aplicativos de serviço podem ser executados dentro do centro de dados 225, e essas modalidades da presen-

te invenção não estão limitadas a esse aplicativo de serviço 210 ilustrado e descrito. Nas modalidades, tendo mais de um aplicativo de serviço, o aplicativo de serviço 210 e outros aplicativos de serviço (não ilustrados) podem operar independentemente e/ou em cooperação um com o outro. Em um caso, os aplicativos de serviço rodando simultaneamente no centro de dados 225 compartilham o conjunto de programas de componente 201, 202, 203 e 204. Por meio de exemplo apenas, o conjunto de programas de componente 202, 202, 203 e 204 pode ser consumido por um novo aplicativo de serviço (por exemplo, fornecido novamente de acordo com as exigências de utilização) ou compartilhado entre os aplicativos de serviço (por exemplo, uma pluralidade de aplicativos de serviço acessando um componente de programa). Dessa forma, os recursos no centro de dados 225, nas modalidades, fornecem a interoperacionalidade de muitos aplicativos de serviço interdependentes ou independentes.

[0033] Nas modalidades, o centro de dados 225 inclui várias máquinas, tal como o primeiro dispositivo de computação 255 e o segundo dispositivo de computação 265, os recursos de memória 250. Os dispositivos de computação 255 e 265 podem ser qualquer tipo de dispositivo de computação, tal como, por exemplo, o dispositivo de computação 100 (ver figura 1), um computador pessoal, servidor, uma coleção de dispositivos de computação, ou outras máquinas capazes de executar o aplicativo de serviço 210. Adicionalmente, os dispositivos de computação 255 e 265 são capazes de armazenar e executar os programas de componente 201, 202, 203 e 204. Os programas de componente 201, 202, 203 e 204 consubstanciam vários componentes de um conjunto de programas de componente que são gerenciados pelo aplicativo de serviço 210. Nas modalidades, os programas de componente 201, 202, 203 e 204 são desenvolvidos, ou escritos, de acordo com as especificações dentro do modelo de serviço. Em um

caso, um ou mais dos programas de componente 201, 202, 203 e 204 são desenvolvidos de acordo com um conjunto de regras dentro do modelo de serviço que permite que os parâmetros 270 reconfigurem automaticamente mediante a determinação de que uma atualização nas configurações ocorreu. Apesar de quatro programas de componente serem apresentados na figura 2 e descritos aqui, as modalidades da presente invenção contempla a utilização de qualquer número de programas de componente para suportar a operação do aplicativo de serviço 210. Em uma modalidade ilustrativa, o número de programas de componente se refere a um nível de carga de cliente no aplicativo de serviço 210, onde quanto maior o nível de carga, mais programas de componente são empregados.

[0034] Os parâmetros 270 residem tipicamente em programas de componente 201, 202, 203 e 204 ou em elementos 266 e 275 residentes em componentes 201 e 204, respectivamente. Geralmente, os parâmetros sintonizam as decisões que cada um dos componentes realiza internamente, refinando, assim, o desempenho coordenado dos componentes. Os parâmetros 270 são ilustrativos em número e natureza. Isso é, apesar de determinados parâmetros serem ilustrados em programas ou elementos de componente específicos, as modalidades da presente invenção contemplam parâmetros de zero a muitos residentes em qualquer um dos programas de componente ou elementos.

[0035] Os recursos de memória 250 residem tipicamente no centro de dados 225 e são configurados para armazenar dados. Nas modalidades, os recursos de memória 250 são pesquisáveis para um ou mais dos itens armazenados em associação com os mesmos. Será compreendido e apreciado pelos versados na técnica que os dados armazenados no armazenador de dados 250 podem ser configuráveis e podem incluir qualquer informação relevante à execução do aplicativo de serviço 210. O conteúdo e volume de tal informação não devem

limitar o escopo das modalidades da presente invenção de forma alguma. Adicionalmente, apesar de ilustrado como um único componente independente, os recursos de memória 250 podem, de fato, ser uma pluralidade de armazenadores de dados, por exemplo, um agrupamento de base de dados. Adicionalmente, os dados armazenados nos recursos de memória 250 são acessíveis pelos programas de componente 201, 202, 203 e 204. Em uma modalidade, as configurações são fornecidas com endereços de recurso de memória que permitem que um ou mais dos programas de componente 201, 202, 203 e 204 localizem os recursos de memória 250, ou outra alocação de memória adequada, a fim de acessar os dados a partir daí.

[0036] Em operação, o aplicativo de serviço 210 é configurado para gerenciar os programas de componente 201, 202, 203 e 204. Um aspecto do gerenciamento envolve o estabelecimento e a configuração dos programas de componente 201, 202, 203 e 204 mediante a detecção de um evento de acionamento. O evento de acionamento alerta o aplicativo de serviço 210 para iniciar os programas de componente 201, 202, 203 e 204 dentro do aplicativo de serviço 210 mediante a detecção do evento de acionamento. Nas modalidades, o gatilho é baseado em uma indicação de uma mudança para um nível de carga no aplicativo de serviço 210 (por exemplo, pelo monitoramento de tráfego a partir de um ou mais dos dispositivos de computação de cliente 215) ou uma indicação de uma mudança nas especificações dentro de um modelo de serviço (por exemplo, por meio de emendas ao contrato de descrição de serviço estabelecido entre um provedor de serviço de aplicativo e um cliente), como mais completamente discutido abaixo com referência à figura 6.

[0037] Incidente à detecção de gatilho, as configurações 280 são extraídas das especificações e/ou ganchos do modelo de serviço. Geralmente, as configurações 280 facilitam o gerenciamento adequado

do aplicativo de serviço 210. Nas modalidades, as configurações 280 podem ser configurações implícitas ou explicitas, como mais completamente discutido acima. Em casos particulares, as configurações 280 incluem endereços de programas de componente, identificadores de algoritmo, localizações de serviço, endereços de recurso de memória, ou quaisquer outras sequências de dados utilizadas para configurar parâmetros dentro dos programas de componente.

[0038] Incidente a, ou simultaneamente com a extração das configurações 280, os programas de componente 201, 202, 203 e 204 são estabelecidos para corresponder à carga, satisfazer o modelo de serviço atualizado, ou para qualquer outra razão que pertença às operações do aplicativo de serviço 210. Em um caso, o estabelecimento de programas de componente 201, 202, 203 e 204 inclui o desenvolvimento de programas de componente 201, 202, 203 e 204 em vários dispositivos de computação 255 e 265 dentro do centro de dados 225. Em outro caso, o estabelecimento de programas de componente 201, 202, 203 e 204 inclui a identificação como estando disponível para iniciação de programas de componente previamente estabelecidos residentes em vários dispositivos de computação 255 e 256 e são identificados. Apesar de duas formas diferentes de estabelecimento de programas de componente 201, 202, 203 e 204 terem sido ilustradas, deve ser compreendido e apreciado pelos versados na técnica que outros métodos para o estabelecimento de programas de componente 201, 202, 203 e 204 podem ser utilizados, e que as modalidades da presente invenção não devem ser limitadas a essas formas ilustradas e descritas aqui.

[0039] Esses programas de componente desenvolvidos e/ou identificados 201, 202, 203 e 204 são configurados de acordo com um processo de iniciação que inclui as seguintes etapas lógicas: acesso a mapas (por exemplo, mapa de transformação, mapa de estrutura), de-

rivação dos valores formalizados 290 a partir das configurações 280 de acordo com o mapa de transformação, a propagação automática de valores formalizados 290 por todos os programas de componente 201, 202, 203 e 204, localização de parâmetros 270 correspondentes aos valores formalizados 290 de acordo com o mapa de estrutura, e instalação dos valores formalizados 290 em parâmetros adequados 270. Em uma modalidade ilustrativa, os parâmetros 270 são interrogados pelo aplicativo de serviço 210 para determinar se cada um dos componentes 201, 202, 203 e 204 é compatível com as configurações 280 antes da instalação dos valores formalizados 290.

[0040] Voltando-se agora à figura 3, um fluxograma é ilustrado que mostra um método geral 300 para o gerenciamento dos componentes dentro de um aplicativo de serviço de acordo com um modelo de serviço, de acordo com uma modalidade da presente invenção. Inicialmente, como indicado no bloco 305, uma indicação (por exemplo, evento de acionamento) é recebida em um aplicativo de serviço. Os eventos de acionamento são discutidos mais completamente com referência à figura 6. Tipicamente, a indicação alerta o aplicativo de serviço de que mais ou menos programas de componente devem ser estabelecidos para manter a operação uniforme do aplicativo de serviço. Em uma modalidade ilustrativa, a adição e subtração dos programas de componente é transparente para um cliente que está atualmente acessando o aplicativo de serviço. Como indicado no bloco 310, as configurações são extraídas a partir dos ganchos mediante interrogação das especificações dentro de um modelo de serviço. Como mais completamente discutido acima, vários tipos de configurações podem ser extraídos, tal como configurações implícitas e explícitas.

[0041] Como indicado no bloco 315, um processo de iniciação para o gerenciamento do aplicativo de serviço, de acordo com os eventos de acionamento, é iniciado. Nas modalidades o processo de iniciação

incluir, pelo menos, as etapas lógicas a seguir: estabelecimento de programas de componente; derivação dos valores formalizados; propagação automática dos valores formalizados; e instalação dos valores formalizados. Em particular, como indicado no bloco 320, os programas de componente são estabelecidos pelo desenvolvimento (por exemplo, geração de novos programas de componente), ou identificação como disponível, de programas de componente existentes em máquinas dentro do centro de dados. Em uma modalidade ilustrativa de identificação de programas de componente existentes disponíveis, os parâmetros dentro dos programas de componente existentes são interrogados para determinar se são compatíveis com as configurações. De acordo, esses programas de componente "compatíveis" recebem valores formalizados durante a propagação.

[0042] Nas modalidades, esses programas de componente são localizados em um ou vários dispositivos de computação dentro de um centro de dados. Esse estilo de desenvolvimento remoto é ativado por instruções (por exemplo, endereços de componente) fornecidas para os dispositivos de componente pelas configurações. Adicionalmente, as configurações incluem instruções para a configuração de uma API em alguns dos programas de componente permitindo, dessa forma, que cada programa de componente coordene com outros tipos e casos de programas de componente, recursos de memória, e provedores de serviço dependentes remotos (por exemplo, serviço mantido pelos parceiros de comunicação).

[0043] Como indicado no bloco 325, os valores formalizados são derivados a partir das configurações. Nas modalidades, os valores formalizados são derivados de acordo com um mapa de transformação, um estado determinado de cada um dos programas de componente ou uma combinação dos mesmos. Esses valores formalizados são automaticamente propagados para os programas de componente

desenvolvidos, os programas de componente compatíveis, ou uma combinação dos mesmos, como indicado no bloco 330. Essas etapas (por exemplo, blocos 320, 325 e 330) podem ser realizadas em qualquer ordem ou simultaneamente. Como indicado no bloco 335, os valores formalizados são instalados nos parâmetros dos programas de componente. Em outras modalidades, os valores formalizados são temporariamente armazenados em associação com um parâmetro correspondente. Incidente à instalação dos parâmetros, o aplicativo de serviço é executado de acordo com os valores formalizados como indicado no bloco 340.

[0044] Voltando-se agora às figuras 4 e 5, os fluxogramas são ilustrados mostrando os métodos 400 e 500 para instalação dos valores formalizados em parâmetros adequados dos programas de componente, de acordo com uma modalidade da presente invenção. Com relação à figura 4, inicialmente, uma estrutura de mapa é acessada pelo aplicativo de serviço, como indicado no bloco 410. Em outras modalidades, o mapa de estrutura é acessado pelas configurações ou armazenado aí. Como indicado no bloco 420, os parâmetros dentro dos programas de componente são localizados de acordo com a lógica dentro do mapa de estruturas. De acordo, a lógica do mapa de estrutura auxilia na direção dos valores formalizados para os parâmetros adequados, promovendo, assim, uma instalação adequada dos valores formalizados, como indicado no bloco 430.

[0045] Com referência à figura 5, um estado de cada programa de componente estabelecido é estabelecido, como indicado no bloco 510. Nas modalidades, o estado geralmente se refere a um caso ou tipo de programa de componente, ou a configuração existente do programa de componente. Como indicado no bloco 520, um mapa de transformação é acessado. Em uma modalidade ilustrativa, o mapa de transformação inclui lógica que fornece a expressão das configurações como valores

formalizados mediante o registro do estado determinado de cada um dos programas de componente. Com base na lógica, os valores formalizados são derivados e correspondem a cada caso de um programa de componente.

[0046] A figura 6 é um fluxograma ilustrando um método 600 para responder a um evento de acionamento, de acordo com uma modalidade da presente invenção. Como indicado no bloco 610, uma indicação das emendas a um contrato de descrição de serviço é recebida. Como indicado no bloco 620, uma determinação de se as emendas afetam as especificações dentro do modelo de serviço é realizada. Se a determinação indicar que nenhum efeito substancial é gerado, a presente configuração dos programas de componente é mantida, como indicado no bloco 630. Como indicado no bloco 640, o tráfego no aplicativo de serviço é monitorado. Como indicado no bloco 650, uma determinação de se existe uma mudança em um nível de uma carga é feita. Se a determinação indicar que não há mudança, ou ajuste substancial, no nível de carga, a presente configuração dos programas de componente é mantida, como indicado no bloco 630.

[0047] No entanto, se as determinações dos blocos 620 e 650 indicarem que um evento de acionamento ocorreu, (por exemplo, uma mudança substancial no nível de carga ocorreu, ou uma emenda na especificação resultou em um efeito substancial), as configurações atualizadas são atualizadas de acordo com o modelo de serviço, como indicado no bloco 660. A utilização do modelo de serviço para criar as configurações atualizadas é discutida mais completamente acima com referência à figura 2. Como indicado no bloco 670, os valores formalizados são derivados das configurações atualizadas. Como indicado no bloco 680, os programas de componente são reconfigurados com os valores formalizados derivados. Nas modalidades, a reconfiguração inclui as etapas listadas na figura 3.

[0048] As modalidades da presente invenção foram descritas com relação às modalidades particulares, que devem, em todos os aspectos, ser ilustrativas ao invés de restritivas. As modalidades alternativas se tornarão aparentes aos versados na técnica à qual as modalidades da presente invenção pertencem sem se distanciar de seu escopo.

[0049] A partir do acima exposto, será observado que essa invenção é uma bem adaptada para alcançar todas as finalidades apresentadas acima, juntamente com outras vantagens que são óbvias e inerentes ao sistema e ao método. Será compreendido que determinadas características e subcombinações têm utilidade e podem ser empregadas sem referência a outras características e subcombinações. Isso é contemplado e está dentro do escopo das reivindicações.

REIVINDICAÇÕES

1. Meio de armazenamento em computador tendo instruções executáveis por computador incorporadas no mesmo que, quando executadas, executam um método para o gerenciamento de um ou mais programas de componente (201, 202, 203, 204) dentro de um aplicativo de serviço (210) de acordo com um modelo de serviço, o método **caracterizado pelo fato de que** comprehende as etapas de:

determinar um nível de carga no aplicativo de serviço (210), sendo que o aplicativo de serviço (210) comprehende os um ou mais programas de componente (201, 202, 203, 204) que são executáveis em dispositivos de computação (255, 265) separados dentro de um centro de dados (225);

extraír (310), a partir das configurações de modelo de serviço, configurações que podem ser administradas para os um ou mais programas de componente (201, 202, 203, 204);

iniciar (315) um processo de iniciação para o gerenciamento do aplicativo de serviço (210) de acordo com o nível da carga, o processo de iniciação comprehendendo:

(a) desenvolver (320) os um ou mais programas de componente (201, 202, 203, 204) dentro do centro de dados (225); e

(b) determinar (510) um estado de cada um dos um ou mais programas de componente (201, 202, 203, 204) desenvolvidos;

(c) acessar (520) um mapa de transformação, em que o mapa de transformação inclui lógica para expressar as configurações como valores formalizados que correspondem ao estado de cada um dos um ou mais programas de componente (201, 202, 203, 204);

(d) empregar a lógica do mapa de transformação para derivar (530) os valores formalizados a partir das configurações de acordo com a lógica do mapa de transformação; e

(e) propagar automaticamente (330) os valores formaliza-

dos derivados ao longo dos um ou mais programas de componente (201, 202, 203, 204) desenvolvidos; e

instalar (335, 430), pelo menos temporariamente, os valores formalizados nos um ou mais programas de componente (201, 202, 203, 204), fornecendo, assim, a funcionalidade do aplicativo de serviço (210) consistente com o modelo de serviço.

2. Meio de armazenamento em computador, de acordo com a reivindicação 1, **caracterizado pelo fato de que** determinar um nível de carga compreende monitorar o tráfego em um sítio da rede hospedando o aplicativo de serviço (210).

3. Meio de armazenamento em computador, de acordo com a reivindicação 1, **caracterizado pelo fato de que** o aplicativo de serviço (210) compreende software residente em um centro de dados (225) mantido por um provedor de serviço de aplicativo acessível a um cliente através de um aplicativo de navegador de rede (230).

4. Meio de armazenamento em computador, de acordo com a reivindicação 3, **caracterizado pelo fato de que** o modelo de serviço é gerado, em parte, de acordo com um contrato de descrição de serviço estabelecido entre o provedor de serviço de aplicativo e o cliente.

5. Meio de armazenamento em computador, de acordo com a reivindicação 1, **caracterizado pelo fato de que** ainda compreende executar o aplicativo de serviço (210) de acordo com os valores formalizados instalados.

6. Sistema de computador para executar um método de iniciação de um aplicativo de serviço (210) pela propagação das configurações para um ou mais programas de componente (201, 202, 203, 204), promovendo, assim, a funcionalidade do aplicativo de serviço (210), o sistema de computador sendo **caracterizado pelo fato de que** compreende:

um primeiro dispositivo de computação (255) para executar os um ou mais programas de componente (201, 202, 203, 204) que suportam a operação do aplicativo de serviço (210); e

um segundo dispositivo de computação (265) para iniciar um processo de iniciação para implementação do aplicativo de serviço (210) de acordo com um modelo de serviço, o processo de iniciação compreendendo:

(a) extrair configurações das especificações dentro do modelo de serviço;

(b) identificar os um ou mais programas de componente (201, 202, 203, 204) residentes no primeiro dispositivo de computação (255) como estando disponível para iniciação;

(c) interrogar os parâmetros dentro dos um ou mais programas de componente (201, 202, 203, 204) para estabelecer se alguns dentre os um ou mais programas de componente (201, 202, 203, 204) são compatíveis com as configurações;

(d) acessar um mapa de estrutura a partir dos um ou mais programas de componente (201, 202, 203, 204) compatíveis, onde o mapa de estrutura inclui lógica para localizar os parâmetros dentro dos um ou mais programas de componente (201, 202, 203, 204) compatíveis;

(e) propagar automaticamente os valores formalizados derivados das configurações ao longo dos um ou mais programas de componente (201, 202, 203, 204) compatíveis;

(f) direcionar os valores formalizados derivados a partir das configurações para os parâmetros de acordo com a lógica do mapa de estrutura; e

(g) configurar os parâmetros dos um ou mais programas de componente (201, 202, 203, 204) compatíveis de acordo com os valores formalizados.

7. Sistema de computador, de acordo com a reivindicação 6, **caracterizado pelo fato de que** o processo de iniciação ainda compreende:

receber (610) uma indicação de uma emenda para as especificações do modelo de serviço; e

extraír (660) as configurações atualizadas a partir das especificações emendadas.

8. Sistema de computador, de acordo com a reivindicação 7, **caracterizado pelo fato de que** os um ou mais programas de componente (201, 202, 203, 204) residentes no primeiro dispositivo de computação (255) são desenvolvidos de acordo com um conjunto de regras que permite que os parâmetros se reconfigurem automaticamente mediante o recebimento de valores formalizados das configurações atualizadas.

9. Sistema de computador, de acordo com a reivindicação 6, **caracterizado pelo fato de que** o segundo dispositivo de computação (265) é adicionalmente configurado para executar os um ou mais programas de componente (201, 202, 203, 204) que suportam a operação do aplicativo de serviço (210).

10. Sistema de computador, de acordo com a reivindicação 9, **caracterizado pelo fato de que** as configurações incluem endereços de componente, os endereços de componente permitindo que um ou mais programas de componente (201, 202, 203, 204) residentes no primeiro dispositivo de computação (255) coordenem as atividades com os um ou mais programas de componente (201, 202, 203, 204) residentes no segundo dispositivo de computação (265).

11. Sistema de computador, de acordo com a reivindicação 6, **caracterizado pelo fato de que** as configurações incluem identificadores de algoritmo, os identificadores de algoritmo instruindo os um ou mais programas de componente (201, 202, 203, 204) a lançar um

algoritmo em particular, afetando, assim, a execução do aplicativo de serviço (210).

12. Sistema de computador, de acordo com a reivindicação 6, **caracterizado pelo fato de que** ainda compreende um conjunto de serviços dependentes para suportar a operação do aplicativo de serviço (210), o aplicativo de serviço (210) sendo gerenciado por um provedor de serviço de aplicativo e o conjunto de serviços dependentes sendo gerenciado por pelo menos uma entidade separada do provedor de serviço de aplicativo.

13. Sistema de computador, de acordo com a reivindicação 12, **caracterizado pelo fato de que** as configurações incluem localizações de recurso, as localizações de recurso permitindo que um ou mais programas de componente (201, 202, 203, 204) coordenem as atividades com o conjunto de serviços dependentes localizados externamente do primeiro dispositivo de computação (255) e do segundo dispositivo de computação (265).

14. Sistema de computador, de acordo com a reivindicação 13, **caracterizado pelo fato de que** a coordenação de atividades compreende:

permutar mensagens entre o conjunto de serviços dependentes e um ou mais programas de componente (201, 202, 203, 204); e

comparar as mensagens permutadas com os valores formalizados instalados nos parâmetros de um ou mais programas de componente (201, 202, 203, 204) compatíveis.

15. Sistema de computador, de acordo com a reivindicação 6, **caracterizado pelo fato de que** o primeiro dispositivo de computação (255) e o segundo dispositivo de computação (265) compreendem um centro de dados (225), sendo que o centro de dados (225) inclui recursos de memória que armazenam dados, onde os dados nos re-

cursos de memória são acessíveis por um ou mais dos programas de componente (201, 202, 203, 204).

16. Sistema de computador, de acordo com a reivindicação 15, **caracterizado pelo fato de que** as configurações incluem endereços de recurso de memória, os endereços de recurso de memória permitindo que um ou mais programas de componente (201, 202, 203, 204) localizem os recursos de memória adequados para acesso aos dados.

17. Método computadorizado para configurar programas de componente (201, 202, 203, 204) de um aplicativo de serviço (210), operando em um centro de dados (225), de acordo com um esquema de iniciação, o método **caracterizado pelo fato de que** compreende as etapas de:

determinar a configuração pela interrogação de um modelo de serviço;

acessar o esquema de iniciação, onde o esquema de iniciação indica quais programas de componente (201, 202, 203, 204) desenvolver a fim de satisfazer o modelo de serviço, e onde o esquema de iniciação fornece um mapa de transformação;

desenvolver os programas de componente (201, 202, 203, 204) indicados em uma pluralidade de dispositivos de computação (255, 265) dentro do centro de dados (225);

converter as configurações em valores formalizados com base, em parte, no mapa de transformação, sendo que os valores formalizados incluem valores esperados que, mediante a correspondência de uma credencial fornecida por usuário (240) com um valor esperado correspondente dentre os valores esperados, são configurados para permitir acesso de um usuário ao aplicativo de serviço (210) através de um ou mais dentre os programas de componente (201, 202, 203, 204);

distribuir automaticamente os valores formalizados para os programas de componente (201, 202, 203, 204) desenvolvidos; e
armazenar, pelo menos temporariamente, os valores formalizados distribuídos em associação com os programas de componente (201, 202, 203, 204) desenvolvidos.

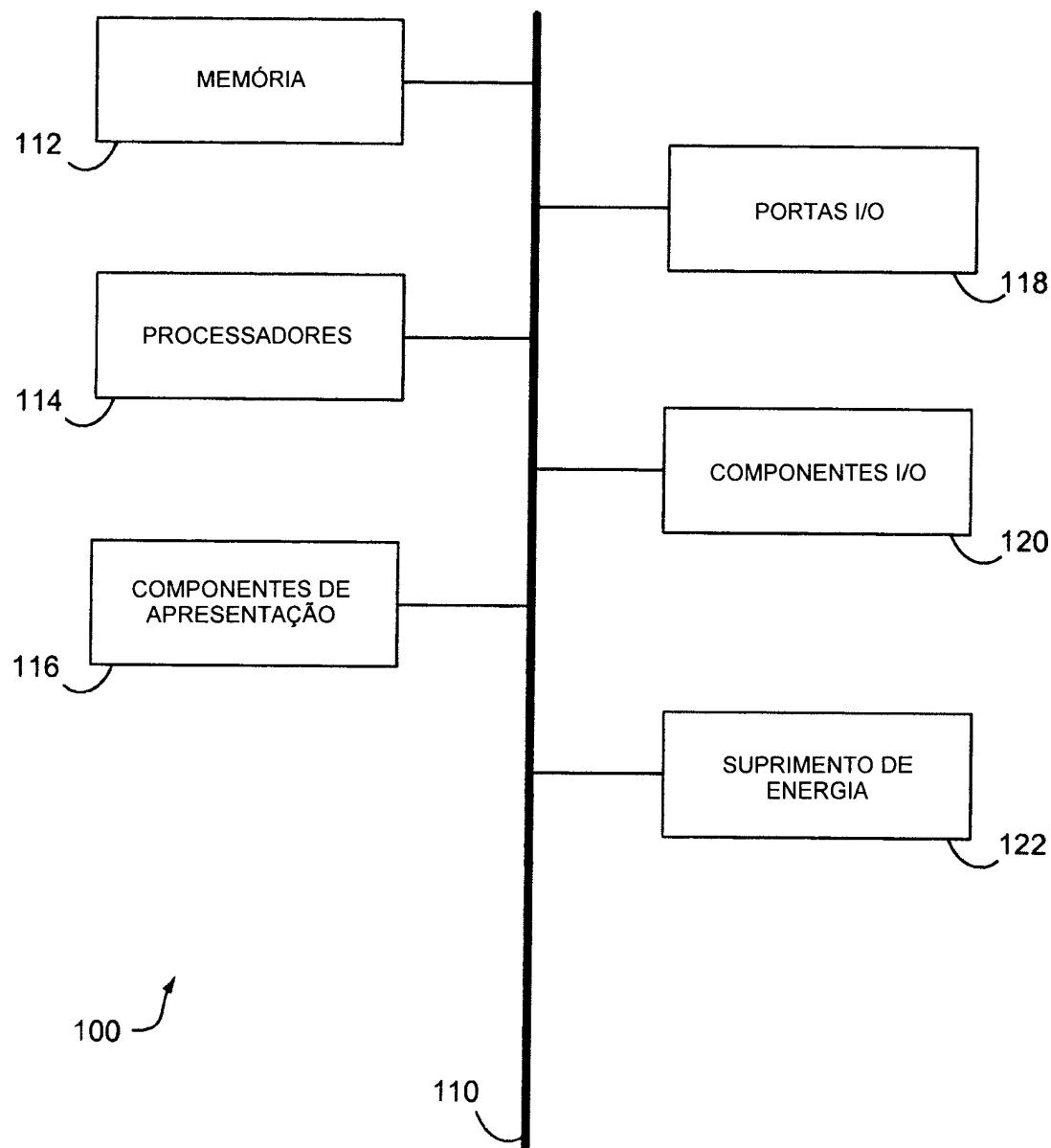


FIG. 1.

FIG. 2.

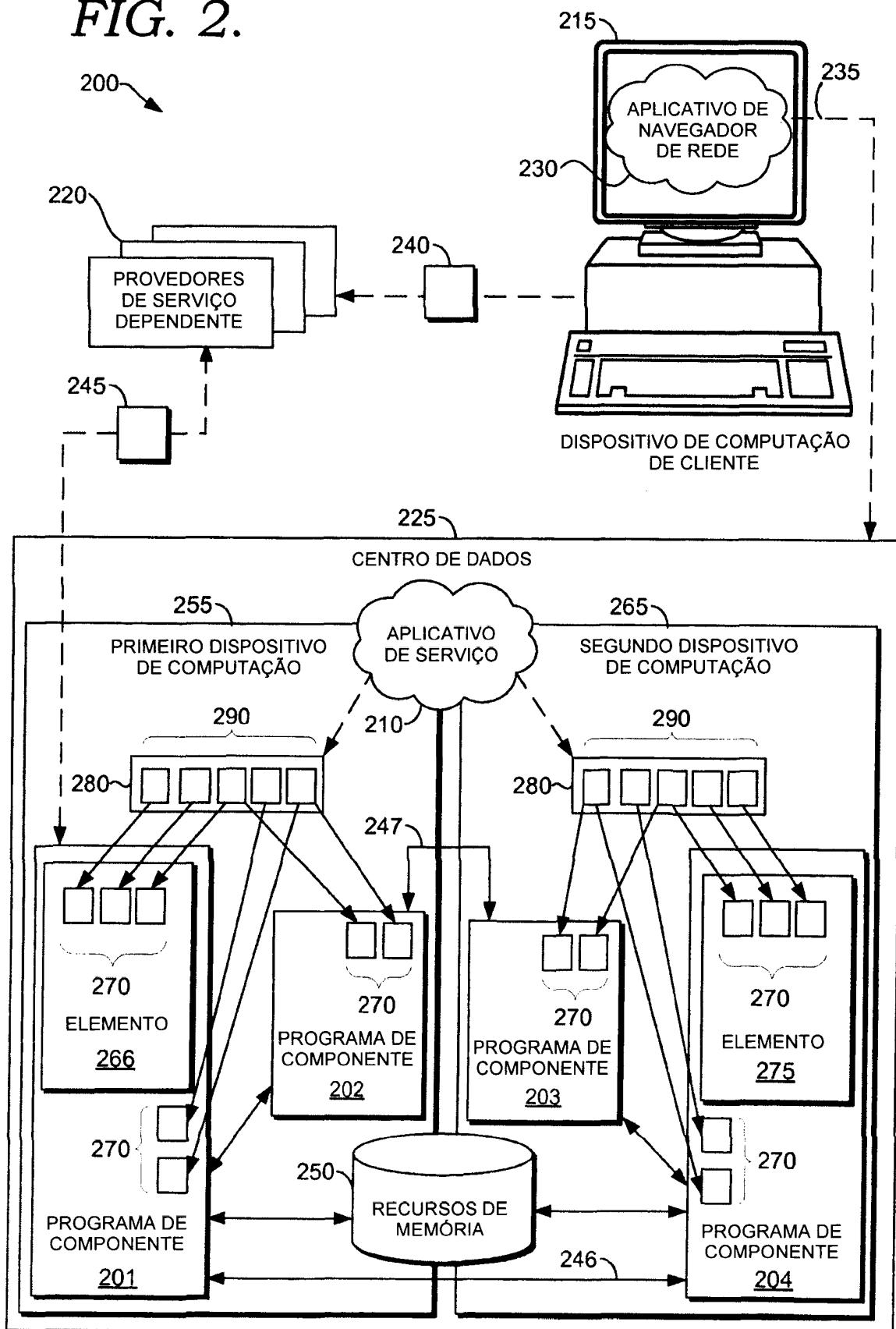


FIG. 3.

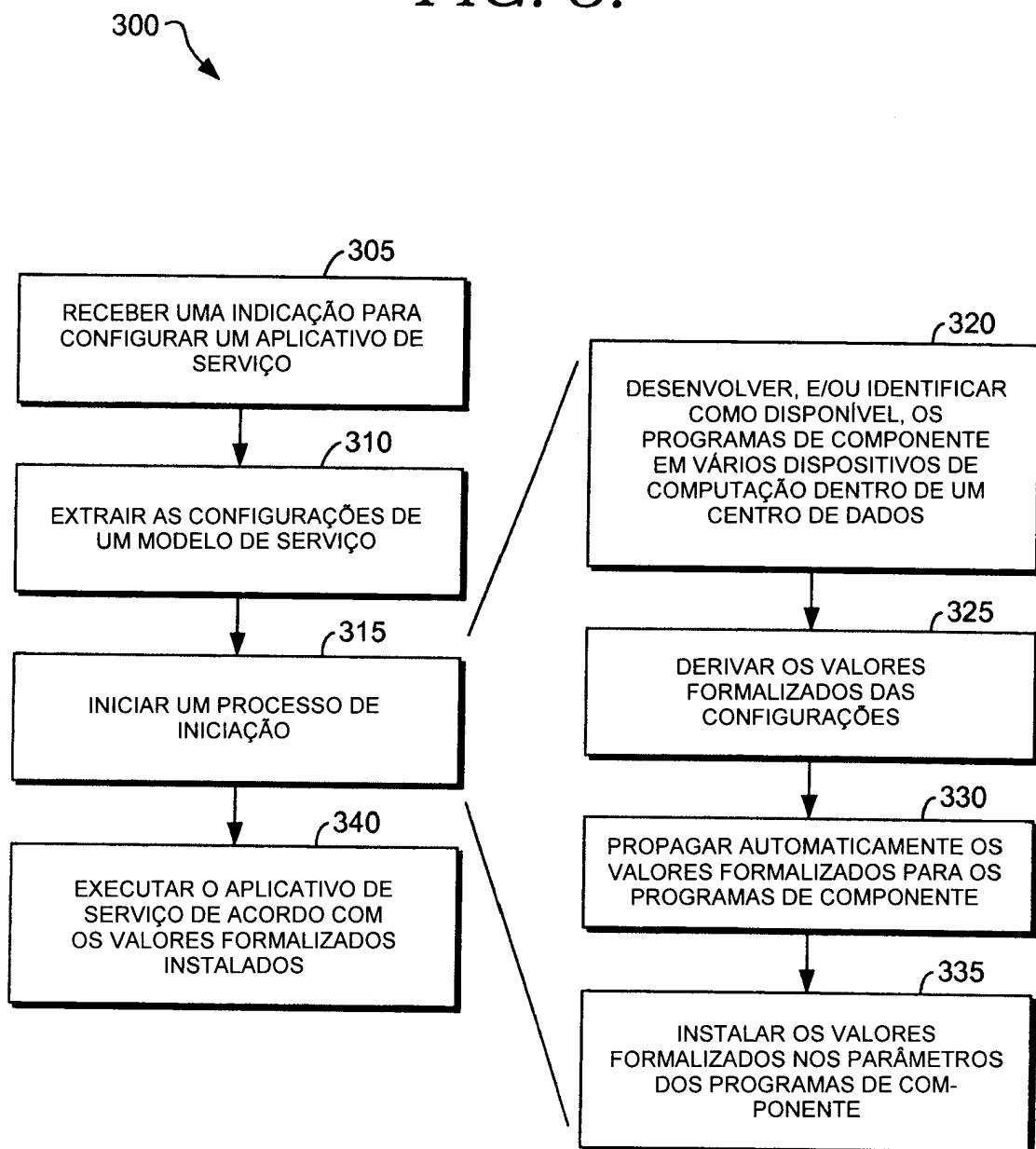


FIG. 4.

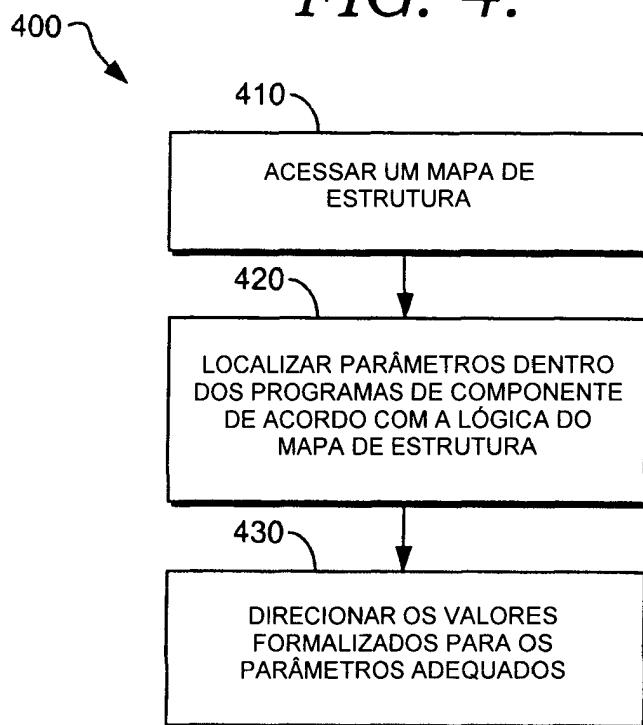
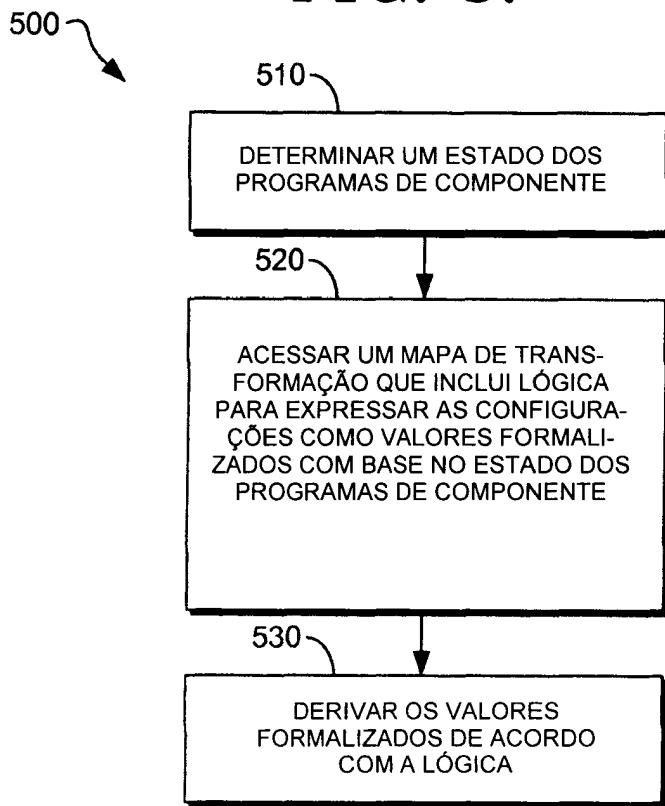


FIG. 5.



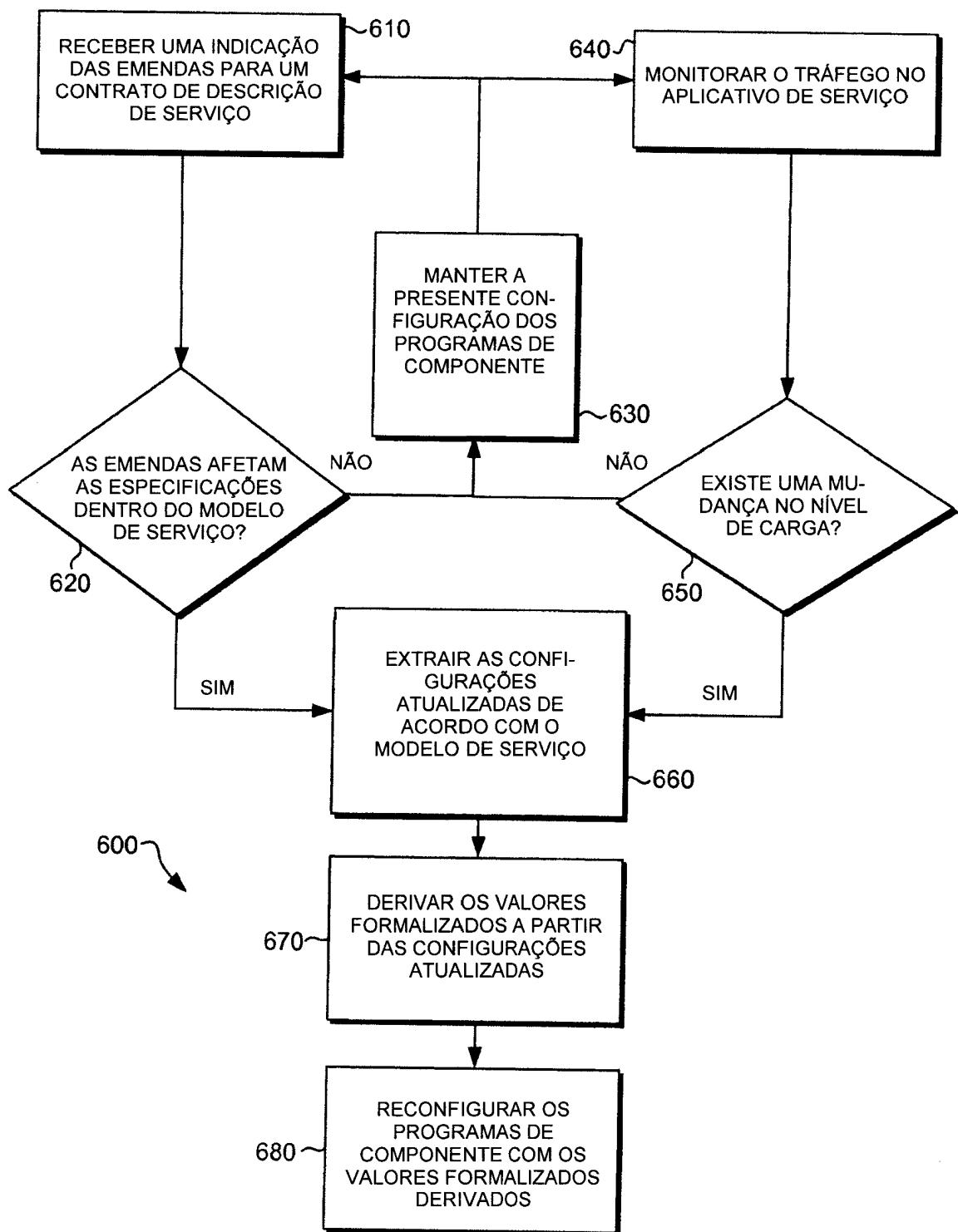


FIG. 6.