

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

(21) **PI 1002963-0 A2**

(22) Data de Depósito: 13/05/2010
(43) Data da Publicação: 20/03/2012
(RPI 2150)



(51) *Int.Cl.:*
H04L 12/54

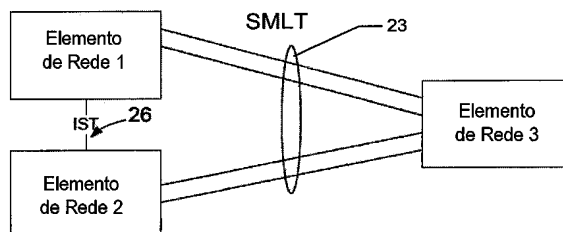
(54) Título: MÉTODO E APARELHO PARA MANUTENÇÃO DE TABELAS DE ESTADO DE PORTA EM UM PLANO DE DIRECIONAMENTO DE UM ELEMENTO DE REDE

(30) Prioridade Unionista: 03/05/2010 US 12/772,457,
13/05/2009 US 61/178,020

(73) Titular(es): Avaya Inc.

(72) Inventor(es): Hamid Assarpour

(57) Resumo: MÉTODO E APARELHO PARA MANUTENÇÃO DE TABELAS DE ESTADO DE PORTA EM UM PLANO DE DIRECIONAMENTO DE UM ELEMENTO DE REDE. Um método, um aparelho e um produto de programa de computador para a manutenção de tabelas de estado de porta em um plano de direcionamento de um elemento de rede são apresentados. O estado de um primeiro conjunto de portas associadas a uma primeira unidade de dados de encaminhamento (FDU) é periodicamente determinado, a primeira FDU sendo uma de uma pluralidade de FDU's. O estado determinado é usado para a atualização de uma primeira tabela de estado de porta do banco de dados de estado de porta associado à primeira FDU. O estado determinado é transmitido para cada uma das outras FDU's no elemento de rede, para se permitir que cada uma das outras FDU's armazene o estado do primeiro conjunto de portas em um banco de dados de estado de porta local para cada uma das outras FDU's. O banco de dados de estado de porta é usado pelo plano de direcionamento para a realização de um novo roteamento rápido de pacotes.



**MÉTODO E APARELHO PARA MANUTENÇÃO DE TABELAS DE ESTADO DE
PORTA EM UM PLANO DE DIRECIONAMENTO DE UM ELEMENTO DE REDE
REFERÊNCIA CRUZADA A PEDIDOS RELACIONADOS**

O presente pedido reivindica o benefício do Pedido de
5 Patente Provisória U.S. N° 61/178.020, depositado em 13 de
maio de 2009, o qual é incorporado aqui como referência em
sua totalidade.

ANTECEDENTES

As redes de comunicação de dados podem incluir vários
10 computadores, servidores, nós, roteadores, comutadores,
centros de conexão, proxies e outros dispositivos acoplados
a e configurados para passarem dados uns para os outros.
Estes dispositivos são referidos aqui como "elementos de
rede", e podem prover uma variedade de recursos de rede em
15 uma rede. Os dados são comunicados através de redes de
comunicação de dados pela passagem de unidades de dados de
protocolo (tais como pacotes, células, quadros ou
segmentos) entre os elementos de rede por enlaces de
comunicação na rede. Uma unidade de dados de protocolo em
20 particular pode ser manipulada por múltiplos elementos de
rede e através de múltiplos enlaces de comunicação,
conforme viajar entre sua fonte e seu destino pela rede.
Anfitriões, tais como computadores, telefones, telefones
celulares, assistentes digitais pessoais, e outros tipos de
25 eletrônicos de consumidor se conectam à e transmitem /
recebem dados pela rede de comunicação e, daí, são usuários
dos serviços de comunicação oferecidos pela rede de
comunicação.

Os elementos de rede tipicamente são implementados
30 para terem um plano de controle que controle a operação do

elemento de rede e um plano de dados que lida com o tráfego fluindo através da rede. O plano de dados tipicamente terá uma coleção de placas de linha tendo portas que se conectam a enlaces na rede. Os dados são recebidos em uma porta em particular, comutada no plano de dados, e extraídos em uma ou mais outras portas para outros enlaces na rede. Para se permitir que os dados sejam manipulados rapidamente, o plano de dados tipicamente é implementado em hardware, de modo que todas as decisões quanto a como manipular os dados que são realizadas usando-se consultas de hardware, etc.

As portas podem falhar por muitas razões, incluindo uma falha de placa de linha, uma falha do enlace conectado à porta (por exemplo, um corte de linha), uma falha de placa de linha de extremidade distante, etc. Os termos tronco de enlace múltiplo (MLT), grupo de agregação de enlace (LAG) e portas lógicas são sinônimos, e estes termos são usados de forma intercambiável. Da mesma forma, o percurso de dados de encaminhamento interno no elemento de rede pode falhar, o que pode fazer com que uma porta ou um conjunto de portas pareça ter falhado, ou pode haver algumas outras falhas ao longo da conexão lógica / virtual até o ponto final de par externo de porta. Há numerosas razões pelas quais uma porta pode falhar.

No caso de uma porta falhar, o tráfego definido para a porta deve ser desviado para fluir para fora por uma porta alternativa, para se permitir que uma conectividade seja restaurada através da rede. Para minimização do impacto sobre o tráfego sendo manipulado pelo elemento de rede, por exemplo, para minimização do tempo parado e da perda de pacote, quanto mais rápido um novo roteamento do tráfego

puder ocorrer, melhor. Preferencialmente, seria vantajoso permitir que o tráfego se recuperasse de uma falha para uma porta alternativa em até dez milissegundos (ms). Preferencialmente, no caso de LAG ou MLT, o tráfego deve se

5 espalhar através das portas remanescentes, ao invés de ser todo movido da porta com falha para uma porta alternativa designada em particular, para se evitar que a porta alternativa designada seja sobrecarregada com o tráfego.

SUMÁRIO

10 Os mecanismos convencionais, tais como aqueles explicados acima sofrem de uma variedade de deficiências. Uma dessas deficiências é a quantidade de tempo gasta por elementos de rede convencionais para se recuperarem de uma falha. Quanto maior a quantidade de tempo gasta para a

15 recuperação, maior o atraso na performance (incluindo mais pacotes perdidos), bem como o carregamento das outras portas para vencerem a porta com falha. As soluções atuais não provêem um método compreensivo para se detectarem eficientemente falhas de porta e distribuírem estados de

20 porta para se permitir um novo roteamento rápido durante um encaminhamento de pacote / célula / quadro. Estas soluções tipicamente fazem uso extensivo de gerenciamento e/ou um de um software de plano de controle para se lidar com um subconjunto de falhas. Sua abordagem resulta em um tempo

25 parado mais longo da rede e em uma perda de pacote mais alta.

As modalidades da invenção suplantam significativamente essas deficiências e provêem mecanismos e técnicas que provêem um método e um aparelho para

30 manutenção de tabelas de estado de porta em um plano de

direcionamento de um elemento de rede. É muito crítico minimizar o tempo parado de rede e a perda de pacote e prover uma recuperação de falha sem emendas em dispositivos de rede, quando essas falhas ocorrerem. Esta invenção
5 permite uma detecção muito rápida e eficiente e uma distribuição dos estados de porta em um agrupamento de nós. Este mecanismo permite que o caminho rápido efetue um novo roteamento rápido sem emendas no advento dessas falhas, sem uma assistência dinâmica de um software de gerenciamento
10 e/ou de plano de controle.

Em uma modalidade em particular de um método para manutenção de tabelas de estado de porta em um plano de direcionamento de um elemento de rede, o método inclui periodicamente determinar um estado de um primeiro conjunto
15 de portas associadas a uma primeira unidade de dados de encaminhamento (FDU), a primeira FDU sendo uma de uma pluralidade de FDUs implementando funções de encaminhamento no plano de direcionamento do elemento de rede. O método ainda inclui o uso do estado determinado para a atualização
20 de uma primeira tabela de estado de porta do banco de dados de estado de porta associado à primeira FDU. Adicionalmente, o método inclui a transmissão do estado determinado para cada outra das FDUs no elemento de rede, para se permitir que cada uma das outras FDUs armazene o
25 estado do primeiro conjunto de portas em um banco de dados de estado de porta local para cada uma das outras FDUs.

Outras modalidades incluem um meio que pode ser lido em computador que tem um código que pode ser lido em computador nele para manutenção de tabelas de estado de
30 porta em um plano de direcionamento de um elemento de rede.

O meio que pode ser lido em computador inclui instruções para periodicamente se determinar um estado de um primeiro conjunto de portas associadas a uma primeira unidade de dados de encaminhamento (FDU), a primeira FDU sendo uma de
5 uma pluralidade de FDUs implementando funções de encaminhamento no plano de direcionamento do elemento de rede. O meio que pode ser lido em computador ainda inclui instruções para o uso do estado determinado para a atualização de uma primeira tabela de estado de porta do
10 banco de dados de estado de porta associado à primeira FDU. Adicionalmente, o meio que pode ser lido em computador inclui instruções para a transmissão do estado determinado para cada outra das FDUs no elemento de rede, para se permitir que cada uma das outras FDUs armazene o estado do
15 primeiro conjunto de portas em um banco de dados de estado de porta local para cada uma das outras FDUs.

Ainda outras modalidades incluem um dispositivo computadorizado (elemento de rede) configurado para processar todas as operações de método mostradas aqui como
20 modalidades da invenção. Nessas modalidades, o dispositivo computadorizado inclui um sistema de memória, um processador, uma interface de comunicações em um mecanismo de interconexão conectando estes componentes. O sistema de memória é codificado com um processo que provê a manutenção
25 de tabelas de estado de porta em um plano de direcionamento de um elemento de rede, conforme explicado aqui, que, quando realizado (por exemplo, quando da execução) no processador, opera conforme explicado aqui no dispositivo computadorizado para a realização de todas as modalidades
30 de método e operações explicadas aqui como modalidades da

invenção. Assim, qualquer dispositivo computadorizado que realize ou seja programado para manter tabelas de estado de porta em um plano de direcionamento de um elemento de rede conforme explicado aqui é uma modalidade da invenção.

5 Outros arranjos de modalidades da invenção que são mostrados aqui incluem programas de software para a realização das etapas de modalidade de método e operações resumidas acima e mostradas em detalhes abaixo. Mais particularmente, um produto de programa de computador é uma
10 modalidade que tem um meio que pode ser lido em computador que inclui uma lógica codificada ali que, quando realizada em um dispositivo computadorizado, provê operações associadas para manutenção de tabelas de estado de porta em um plano de direcionamento de um elemento de rede, conforme
15 explicado aqui. A lógica de programa de computador, quando executada em pelo menos um processador com um sistema de computação, faz com que o processador realize as operações (por exemplo, os métodos) indicadas aqui como modalidades da invenção. Esses arranjos da invenção tipicamente são
20 providos como um software, um código e/ou outras estruturas de dados dispostas ou codificadas em um meio que pode ser lido em computador, tal como um meio ótico (por exemplo, um CD-ROM), um disco flexível ou rígido, ou um outro meio, tais como um firmware ou microcódigo em um ou mais dentre
25 chips de ROM, RAM ou PROM, ou como um circuito integrado específico de aplicação (ASIC), ou como imagens de software transferíveis (via download) em um ou mais módulos, bibliotecas compartilhadas, etc. O software ou o firmware ou outras dessas configurações podem ser instalados no
30 dispositivo computadorizado para se fazer com que o

dispositivo computadorizado realize as técnicas explicadas aqui como modalidades da invenção. Os processos de software que operam em uma coleção de dispositivos computadorizados, tal como em um grupo de dispositivos de comunicações de dados ou outras entidades, também podem prover o sistema da invenção. O sistema da invenção pode ser distribuído entre muitos processos de software em vários dispositivos de comunicações de dados, ou todos os processos poderiam rodar em um conjunto pequeno de computadores dedicados, ou em um computador sozinho.

É para ser entendido que as modalidades da invenção podem ser concretizadas estritamente como um programa de software, como software e hardware, ou como hardware e/ou um circuito sozinho, tal como em um dispositivo de comunicações de dados. Os recursos da invenção conforme explicado aqui podem ser empregados em dispositivos de comunicações de dados e/ou sistemas de software para esses dispositivos, tais como aqueles fabricados pela Avaya, Inc. de Lincroft, Nova Jersey.

Note que cada um dos diferentes recursos, técnicas, configurações, etc. discutidos nesta exposição pode ser executado independentemente ou em combinação. Assim sendo, a presente invenção pode ser concretizada e vista de muitas formas diferentes. Também, note que esta seção de sumário aqui não especifica toda modalidade e/ou aspecto de novidade em incremento da presente exposição ou da invenção reivindicada. Ao invés disso, este sumário apenas provê uma discussão preliminar de modalidades diferentes e pontos correspondentes de novidade em relação a técnicas convencionais. Para detalhes adicionais, elementos e/ou

perspectivas possíveis (permutações) da invenção, o leitor é dirigido para a seção de Descrição Detalhada e para as figuras correspondentes da presente exposição, conforme discutido adicionalmente abaixo.

5 BREVE DESCRIÇÃO DOS DESENHOS

O precedente será evidente a partir da descrição mais particular a seguir de modalidades preferidas da invenção, conforme ilustrado nos desenhos associados, nos quais caracteres de referência iguais se referem às mesmas partes por todas as várias vistas. Os desenhos não estão necessariamente em escala, a ênfase sendo imposta, ao invés disso, na ilustração dos princípios da invenção.

A Figura 1 é um diagrama de blocos funcional que mostra um primeiro exemplo de organização de um agrupamento de nós;

a Figura 2 é um diagrama de blocos funcional que mostra um outro exemplo de organização de um agrupamento de nós;

a Figura 3 é um diagrama de blocos funcional que mostra um outro exemplo de organização de um agrupamento de nós;

a Figura 4 é um diagrama de blocos funcional que mostra um outro exemplo de organização de um agrupamento de nós;

a Figura 5 é um diagrama de blocos funcional que mostra um outro exemplo de organização de um agrupamento de nós;

a Figura 6 é um diagrama de blocos funcional que mostra um outro exemplo de organização de um agrupamento de nós;

a Figura 7 é um diagrama de blocos funcional de uma rede de comunicação de exemplo;

a Figura 8 é um diagrama de blocos funcional de um elemento de rede de exemplo;

5 a Figura 9 é um diagrama de blocos funcional de uma placa de linha de exemplo que pode ser usada em um elemento de rede, tal como o elemento de rede da Figura 8;

a Figura 10 é um diagrama de blocos funcional de um banco de dados de porta de exemplo que pode ser usado em
10 uma placa de linha, tal como na placa de linha da Figura 9;

a Figura 11 é um diagrama de blocos funcional de um agrupamento de exemplo de elementos de rede que mostra o fluxo de informação de estado de porta entre os componentes do caminho de dados no agrupamento de nó;

15 a Figura 12 ilustra uma arquitetura de exemplo de sistema de computador para um elemento de rede que mantém uma tabela de estado de porta de acordo com as modalidades da invenção;

a Figura 13 compreende um fluxograma de um método para
20 manutenção de tabelas de estado de porta de acordo com as modalidades da invenção; e

a Figura 14 compreende um fluxograma de um método para a atualização de uma tabela de estado de porta em um elemento de rede de acordo com as modalidades da invenção.

25 DESCRIÇÃO DETALHADA

Ao longo do tempo, a maneira pela qual os elementos de rede manipulam dados evoluiu. Por exemplo, dois ou mais enlaces físicos podem se estender entre um grupo de elementos de rede e ser usados coletivamente como um MLT ou
30 um LAG. A Figura 1 mostra um exemplo de dois elementos de

rede (elemento de rede 1 e elemento de rede 2) conectados por múltiplos enlaces 24a a d que foram agrupados para formarem um tronco de enlace múltiplo 22. Em particular, cada um dos enlaces 24a a d no MLT 22 pode ser usado por qualquer um dos elementos de rede para encaminhamento de dados para os outros. Assim, se o elemento de rede 1 tiver dados (por exemplo, um quadro / pacote) a enviar para o elemento de rede 2, o elemento de rede 1 poderá selecionar um dos enlaces 24a a 24d a partir do MLT 22 e transmitir o pacote por aquele enlace para o elemento de rede 2.

A Figura 2 mostra uma outra forma de exemplo pela qual os elementos de rede podem ser interconectados. Especificamente, neste exemplo, o elemento de rede 1 e o elemento de rede 2 são interconectados por um tronco intercomutador (IST) 26, o qual pode ser um enlace único ou em si pode ser um tronco de enlace múltiplo. Quando os enlaces do tronco de enlace múltiplo são fisicamente conectados a dois elementos de rede diferentes, o MLT é denominado um tronco de enlace múltiplo dividido (SMLT). Os elementos de rede 1 e 2 podem ter, cada um, um ou mais enlaces que se conectam ao elemento de rede 3, os quais podem ser agrupados em conjunto para formarem um SMLT 23. Assim, se o elemento de rede 1 tiver dados (por exemplo, um quadro / pacote) a enviar para o elemento de rede 3, o elemento de rede 1 poderá selecionar um dos enlaces de SMLT conectados a ele ou poderá transmitir o pacote em um dos enlaces associados ao tronco intercomutador 26, para se permitir que o elemento de rede 2 encaminhe os dados em um de seus enlaces associados ao SMLT para o elemento de rede

A Figura 3 mostra um outro exemplo no qual o elemento de rede 1 não tem quaisquer enlaces conectados ao SMLT 23, mas é conectado pelos ISTs 25 e 26 a dois outros elementos de rede (elemento de rede 2 e elemento de rede 3) que realmente têm portas conectadas aos enlaces associados ao SMLT. Neste cenário, se o elemento de rede 1 tiver dados a enviar no SMLT, ele selecionará um dos enlaces de IST (note que cada enlace de IST pode em si ser um SMLT) e encaminhar os dados para o elemento de rede 2 ou 3. Os ISTs podem ser físicos e se estender diretamente entre dois elementos de rede, ou podem ser lógicos e se estender em túneis através de um ou mais elementos de rede intermediários.

A Figura 4 mostra um outro exemplo no qual o elemento de rede 1 também participa no SMLT 23. Neste caso, se o elemento de rede 1 tiver dados a enviar, ele poderá encaminhar os dados em um de seus enlaces associados ao SMLT 23, ou poderá encaminhar os dados em um dos enlaces associados a um dos ISTs 25 ou 26, para se permitir que os dados sejam encaminhados no SMLT 23.

As Figuras 5 e 6 mostram uma outra forma de interconexão de elementos de rede em um arranjo de SMLT quadrado. No arranjo mostrado na Figura 5, quatro elementos de rede são interconectados através dos ISTs 25 a 28, em um arranjo quadrado, e na Figura 6 os quatro elementos de rede são interconectados através dos ISTs 25 a 30 em um arranjo em malha. Os ISTs podem ser físicos e se estender diretamente entre dois elementos de rede, ou podem ser lógicos e se estender em túneis através de um ou mais elementos de rede intermediários.

Embora vários exemplos de formas pelas quais os

dispositivos de rede podem ser interconectados tenham sido mostradas, há outras formas para a interconexão de um agrupamento de elementos de rede também, e não se pretende que este conjunto de exemplo de arquiteturas de interconexão seja exaustivo. Assim, estes exemplos foram meramente pretendidos para a provisão de um exemplo representativo de umas poucas formas de interconexão de elementos de rede. Um grupo de elementos de rede será referido aqui como um agrupamento.

10 A Figura 7 mostra uma rede de comunicação de exemplo na qual os assinantes 12 se conectam a um comutador de borda 14. O comutador de borda 14 se conecta a comutadores de núcleo 16, os quais encaminham dados através da rede em enlaces 18. Cada um destes comutadores pode ser um roteador-comutador físico ou pode ser múltiplos dispositivos conectados em conjunto para operarem como um agrupamento. Cada um dos enlaces 18 pode ser um MLT ou, onde o roteador / comutador é implementado como múltiplos dispositivos físicos, pode ser um SMLT. A partir do ponto de vista de roteamento de rede, pode haver múltiplas formas para que um pacote atravessasse a rede. Por exemplo, na Figura 7, o comutador de borda A pode ser capaz de transmitir um pacote para o comutador de borda B através dos comutadores de núcleo C e D, ou, alternativamente, pode ser capaz de transmitir o pacote através dos comutadores de núcleo E e F. Um protocolo de roteamento de camada de rede pode ser usado para a determinação do percurso a ser usado para transmissão do pacote.

Conforme citado acima, dependendo da maneira pela qual os elementos de rede são interconectados, pode haver muitas

formas para que o elemento de rede encaminhe um quadro /
pacote para se permitir que o quadro / pacote atinja seu
destino. Conforme usado aqui, o termo "agrupamento" é usado
para uma referência a um ou mais nós provendo uma
5 resiliência de nível de nó no nível de rede. Assim, na
Figura 1, o elemento de rede 1 seria um agrupamento; na
Figura 2, os elementos de rede 1 e 2 seriam um agrupamento,
e nas Fig. 3 e 4, os elementos de rede 1, 2 e 3 seriam um
agrupamento, e nas Fig. 5 e 6, os elementos de rede 1 a 4
10 seriam um agrupamento. Conforme citado acima, há outras
formas de organização de nós em um agrupamento.

As conexões lógicas entre os nós de agrupamento são
referidas aqui como troncos intercomutador (ISTs). Os ISTs
podem ser enlaces físicos que se estendem a partir de um
15 elemento de rede até um elemento de rede vizinho no
agrupamento, ou podem ser enlaces lógicos que criam um
túnel através de um ou mais elementos de rede
intermediários no agrupamento. O nó que recebe um pacote a
partir de uma porta não de IST será referido como um nó
20 local. Todos os outros nós no agrupamento são referidos
como nós remotos com respeito ao pacote recebido. Quando um
IST é selecionado para uso, o IST é selecionado em uma base
rotacional. Isto é feito para se evitar a seleção
repetidamente do mesmo IST, e é usado no lugar de uma
25 função de prova (hash).

Dois ou mais enlaces podem ser agrupados para a
formação de um tronco de enlace múltiplo (MLT). A cada MLT
será atribuído um ID de grupo de MLT (MLT-ID), o qual é um
valor global no agrupamento e único através dos nós de
30 agrupamento. Um MLT com todos os seus membros de porta

apenas em um nó é referido como um grupo de MLT normal. Um grupo de MLT em que seus membros de porta estão em dois ou mais nós é referido como um grupo de MLT ou de SMLT dividido.

5 Quando uma porta lógica é implementada como um MLT ou um SMLT, há na realidade múltiplas portas físicas que são capazes de encaminhar um pacote para seu próximo salto na rede. Assim sendo, se uma das portas de um MLT / SMLT falhasse, seria vantajoso fazer com que o pacote fosse
10 encaminhado em uma das portas remanescentes, de modo que o pacote possa atravessar a rede, ao invés de ser perdido. Da mesma forma, ao invés de se designar uma porta primária e de backup para cada porta no MLT / SMLT, seria vantajoso compartilhar a carga dos pacotes através das portas
15 remanescentes do MLT / SMLT, de modo que os pacotes possam ser distribuídos através das portas remanescentes que estiverem LIGADAS. De acordo com uma modalidade, este processo é implementado em hardware, de modo que o caminho rápido (plano de dados) possa acomodar automaticamente
20 falhas de porta individuais e múltiplas e automaticamente redirecionar o tráfego de pacote através das portas remanescentes de uma maneira eqüitativa.

A Figura 8 mostra um diagrama de blocos funcional de um elemento de rede de exemplo 20, o qual pode ser usado
25 como qualquer um dos elementos de rede mostrados em qualquer uma das Fig. 1 a 6. No exemplo mostrado na Figura 8, o elemento de rede inclui um plano de controle 31 e um plano de dados 32. O plano de controle tem uma ou mais CPUs 34 e geralmente roda processos de controle, tais como
30 processos de roteamento, processos de gerenciamento, etc. O

plano de controle programa o plano de dados para instruir o plano de dados quanto a como encaminhar os dados na rede.

O plano de dados 32 pode ser construído usando-se muitas arquiteturas diferentes, e o exemplo mostrado na Figura 8 é apenas um exemplo dessa arquitetura. No exemplo mostrado na Figura 8, o plano de dados inclui uma pluralidade de placas de linha 36, cada uma das quais implementando múltiplas portas físicas as quais se conectam a enlaces na rede. As placas de linha nesta modalidade são interconectadas por uma malha de comutação 40, embora em outras modalidades as placas de linha possam ser interconectadas diretamente e realizar funções de comutação de uma maneira distribuída.

Conforme mostrado na Figura 9, cada placa de linha 36 inclui uma pluralidade de portas 38, as quais se conectam fisicamente aos enlaces na rede. A placa de linha também inclui uma ou mais unidades funcionais 42 que processam os pacotes recebidos a partir das portas anexadas. Conforme usado aqui, a unidade funcional que processa pacotes a partir das portas anexadas em ambas as direções de ingresso e de egresso e toma decisões de encaminhamento é referida como uma unidade de caminho de dados de encaminhamento ou FDU 42. A placa de linha também inclui um processador de gerenciamento / controle (MCP) 44 que interage com o plano de controle para se permitir que o plano de controle programe instruções para a FDU 42 e, opcionalmente, para outros componentes na placa de linha, de modo que a FDU 42 manipule os dados apropriadamente na rede. O MCP 44 também checka periodicamente o status da FDU 42 e outros componentes da placa de linha, para detectar quando uma

falha ocorre.

Com referência, agora, à Figura 10, cada FDU 42 mantém um banco de dados de estado de porta 46. Este banco de dados de estado de porta 46 mantém os estados de enlace físico e os estados de conexão para sua porta local, bem como para suas portas remotas. O banco de dados de estado de porta 46 inclui duas tabelas - uma tabela de estado de porta local 48 e uma tabela de estado de porta remota 50. A tabela de estado de porta local 48 mantém os estados de porta pertencentes ao nó local, e a tabela de estado de porta remota 50 mantém os estados de porta referentes aos nós remotos no agrupamento. No caso de grupos de MLT, a FDU mantém os membros de porta de FDU no nó local na tabela local, e os membros de porta em todos os outros nós remotos na tabela remota.

Quando a FDU recebe um pacote, é requerido encontrar uma porta no MLT (ou no SMLT) que esteja LIGADA para encaminhamento do pacote em direção ao seu destino na rede. Uma porta está em um estado LIGADO ou em um estado DESLIGADO. Conforme citado acima, quando todas as portas no MLT são locais, a FDU precisará determinar qual das portas locais está no estado LIGADO, de modo que não tente encaminhar o pacote por uma porta que esteja DESLIGADA. Da mesma forma, quando as portas associadas ao MLT não são todas locais (por exemplo, SMLT), a FDU precisará selecionar uma porta em um elemento de rede físico separado que esteja associado ao SMLT e tenha um estado LIGADO.

De acordo com uma modalidade, cada FDU mantém um banco de dados de estado de porta 46 que usa para manter o estado de cada porta no agrupamento. O banco de dados mantém os

estados de enlace físico e os estados de conexão para sua porta local, bem como para todas as suas portas remotas. O banco de dados inclui duas tabelas - especificamente, uma tabela local 48 e uma tabela remota 50. A tabela local mantém os estados de porta pertencentes a todas as FDUs no nó local, e a tabela remota mantém os estados de porta de todas as portas em todos os nós remotos no agrupamento. No caso de grupos de MLT, a FDU mantém o estado dos membros de porta de MLT que estejam no nó local na tabela local, e mantém o estado dos membros de porta de MLT que estejam em todos os nós remotos na tabela remota. A tabela local também mantém o estado de todas as portas de IST. Quando um pacote é recebido, a FDU usará o banco de dados de estado de porta para determinar uma porta para o pacote que esteja LIGADA e encaminhará o pacote para aquela porta para ser extraído na rede.

Uma vez que o banco de dados de estado de porta 46 é usado pela FDU 42 para a tomada de decisões de encaminhamento, é importante manter a tabela de estado de porta atualizada, de modo que contenha uma informação atual sobre o estado de cada uma das portas. Uma vez que cada placa de linha tem uma ou mais FDUs, é requerido que cada FDU (em cada placa de linha) se sincronize com as FDUs em todas as placas de linha no elemento de rede local, bem como com todas as outras FDUs nos outros elementos de rede no agrupamento.

Em uma modalidade, cada placa de linha mantém temporizadores de batimento cardíaco. Em cada momento em que um em particular dos temporizadores de batimento cardíaco expira, um motor de batimento cardíaco 54 gera um

pacote de batimento cardíaco e envia o pacote de batimento cardíaco para a FDU local 42 naquela placa de linha. O pacote de batimento cardíaco porta o status de enlace físico local de todas as portas na placa de linha para 5 informar à FDU local o estado das portas naquela placa de linha. O pacote indica o ID de porta global (GPID) e um ID de elemento de rede. A FDU usa este estado para a atualização de seu banco de dados de estado de porta local. A FDU também encaminhará o pacote para todas as outras FDUs 10 no nó local, bem como para todas as outras FDUs no nó local, bem como para todas as outras FDUs nos outros nós no agrupamento. Cada FDU usa o estado de porta portado pelo pacote para a atualização de seu banco de dados de estado de porta.

15 Além disso, para a manutenção de um temporizador associado à coleta / ao relatório de sua própria informação de estado de porta, cada placa de linha / FDU também manterá um conjunto de temporizadores associados a todas as outras FDUs no nó local e todas as outras FDUs em outros 20 nós no agrupamento. Cada FDU espera receber pacotes de batimento cardíaco periódicos de toda outra FDU local e remota. Assim, um temporizador de recepção local é mantido por FDU (para cada FDU no elemento de rede local, bem como para cada FDU em cada elemento de rede remoto no 25 agrupamento). Uma falha de FDU (local ou remota) é detectada, se o temporizador de recepção correspondente espirar. Quando um pacote de batimento cardíaco não é recebido antes de uma expiração do temporizador de recepção, cada porta associada à FDU será regulada para 30 DESLIGADA, de modo que os pacotes não sejam enviados para

portas associadas à FDU, até ela ser restaurada.

Os pacotes de batimento cardíaco permitem que as FDU's portem uma informação de estado para cada outra e permitem que cada FDU conheça o estado de todas as portas no plano de dados. Conforme descrito abaixo, isto permite que o plano de dados automaticamente se ajuste a falhas de porta, de modo que os dados possam ser redirecionados para as portas que estejam LIGADAS e para longe de portas que estejam DESLIGADAS. Tudo isso acontece sem uma intervenção do plano de controle e, daí, o plano de controle não é notificado de uma falha de uma porta / placa de linha em particular. Para se permitir que o plano de controle tome ciência de falhas de plano de dados, o processador de gerenciamento / controle 44 periodicamente injeta e extrai pacotes de batimento cardíaco para e para fora de sua FDU local 42. Cada pacote de batimento cardíaco injetado faz um laço completamente através da FDU alvo e das portas associadas e, então, é extraído de volta para o processador. O pacote de batimento cardíaco de gerenciamento atravessa todos os blocos funcionais em ambos os caminhos de dados de ingresso e de egresso. A cada vez em que o processador de controle injeta um pacote de batimento cardíaco de gerenciamento desta natureza, ele dá a partida em seu temporizador de recepção correspondente. O processador de controle detecta uma falha da placa de linha se o temporizador de recepção expirar. O processador usa esta informação para regular um alarme de sistema, o qual será levado para o plano de controle 30. O plano de controle assim pode tomar ciência quanto a uma falha de plano de dados. Contudo, uma vez que o plano de dados tem

um mecanismo de autocura, para a acomodação de falhas de porta e redirecionamento de tráfego de modo conforme, não é requerido que o plano de controle esteja envolvido no redirecionamento do tráfego e, daí, uma notificação do plano de controle da falha não é crítica para a restauração do tráfego através do elemento de rede.

Os pacotes de batimento cardíaco também são usados em cada FDU para a determinação do estado de suas portas. Em uma modalidade, cada FDU mantém um par de temporizadores por porta anexada que é configurada em uma conexão lógica / virtual. Um dos temporizadores é usado para a geração de pacotes de batimento cardíaco a serem enviados pela conexão. O outro temporizador (temporizador de recepção) é usado para a detecção de uma falha de conexão. Este temporizador expira se o pacote de batimento cardíaco do outro ponto final da conexão não for recebido a tempo. A FDU atualiza sua tabela de estado de porta com a chegada de pacotes de batimento cardíaco e expirações de temporizador de recepção.

Cada FDU em cada placa de linha mantém sua própria tabela de estado de porta 46. Esta tabela mantém os estados de enlace físico e os estados de conexão para suas portas locais, bem como todas as portas remotas de todas as FDU's no agrupamento. A FDU usa os pacotes de batimento cardíaco recebidos e as mensagens de expiração de temporizador (devido ao tempo esgotado de conexão ou a uma falha de FDU remota) para a atualização da tabela. A tabela é dividida em dois segmentos: estados de porta pertencentes ao nó local e estados de porta pertencentes aos nós remotos. A tabela de estado de porta também mantém uma informação de

grupo de MLT e SMLT. A tabela de estado de porta é usada pela lógica de encaminhamento para a realização de um novo roteamento rápido, conforme explicado em maiores detalhes abaixo.

5 A Figura 11 mostra uma disseminação do pacote de estado de porta em um agrupamento de quatro nós, em que as setas escuras representam o fluxo do pacote de estado de porta para todas as FDUs no agrupamento. Conforme mostrado nesta figura, o pacote 1 será gerado contendo o estado de
10 cada porta associada a uma FDU em particular. Este pacote é mostrado com o número de referência 1 na placa de linha esquerda de topo do elemento de rede esquerdo de topo. Este pacote será passado para a FDU, de modo que a FDU possa atualizar seu banco de dados de estado de porta para
15 refletir o status atual de suas portas. O pacote então será passado para cada uma das outras FDUs no nó local (setas 2). Em uma modalidade, isto pode ser implementado ao se fazer com que o pacote seja difundido pela malha de comutação para todas as FDUs no nó local. O pacote também
20 será encaminhado para os outros nós no agrupamento (setas 3), de modo que o estado da porta possa ser distribuído (setas 4) para cada FDU associada a cada nó do agrupamento. Sempre que uma FDU receber um pacote contendo uma informação de estado de porta, ela usará a informação para
25 a atualização de seu próprio banco de dados de estado de porta. Isto permite que o banco de dados de estado de porta de todas as FDUs no agrupamento seja sincronizado.

Pode haver vários elementos de rede em um agrupamento, múltiplas FDUs em um elemento de rede, e múltiplas portas
30 suportadas por meio de cada FDU. Para se permitir que cada

nó mantenha um acompanhamento de quais FDUs proveram pacotes de estado, e para correlação dos pacotes de estado de porta em particular com as FDUs em particular, um esquema de numeração pode ser implementado.

5 Preferencialmente, o esquema de numeração é implementado para ser único no sentido de agrupamento, de modo que cada FDU no agrupamento possa ser unicamente identificada. Em uma modalidade, o pacote de estado de porta traz uma informação sobre cada uma de suas portas. O pacote

10 especificará o ID de nó de fonte e o ID de porta global (GPID). O ID de porta global é o identificador globalmente único (globalmente único em um nó) que permite que a porta seja identificada unicamente no banco de dados de porta.

As mensagens prévias descreveram como as FDUs trocavam

15 mensagens para se permitir que o estado de porta fosse sincronizado entre as portas no caminho de dados. Em operação, esta informação de estado de porta permitirá que as FDUs selecionem uma porta disponível para um pacote de dados em particular com uma confiança que a porta

20 selecionada esteja LIGADA. Conforme as portas se DESLIGAM, as FDUs no agrupamento pararão de selecionar aquelas portas e, ao invés disso, selecionarão portas alternativas no MLT / SMLT associado à porta desligada para serem usadas para manipulação do pacote. Assim sendo, o caminho de dados é

25 capaz de acomodar automaticamente falhas de porta, falhas de placa de linha, etc., para um novo roteamento dos pacotes para as portas alternativas disponíveis, sem envolvimento do plano de controle. Daí, um novo roteamento dos pacotes pode ser feito rapidamente em um elemento de

30 rede e entre agrupamentos de elementos de rede em menos de

10 ms.

A Figura 12 é um diagrama de blocos que ilustra uma arquitetura de exemplo de uma unidade de dados de encaminhamento (FDU) 110 que executa, roda, interpreta, opera ou realiza de outra forma um aplicativo de tabela de estado de porta 140-1 e um processo de tabela de estado de porta 140-2 adequado para uso na explicação da configuração de exemplo mostrada aqui. Conforme mostrado neste exemplo, a FDU 110 inclui um mecanismo de interconexão 111, tal como um barramento de dados ou um outro circuito, que se acopla a um sistema de memória 112, um processador 113, uma interface de entrada / saída 114, e uma interface de comunicações 115. A interface de comunicações 115 permite que a FDU 110 se comunique com outros dispositivos de sistema (isto é, outros computadores) em uma rede (não mostrada).

O sistema de memória 112 é qualquer tipo de meio que pode ser lido em computador e, neste exemplo, é codificado com um aplicativo de tabela de estado de porta 140-1, conforme explicado aqui. O aplicativo de tabela de estado de porta 140-1 pode ser concretizado como um código de subsequente, tais como dados e/ou instruções lógicas (por exemplo, um código armazenado na memória ou em um outro meio que pode ser lido em computador, tal como um disco removível) que suporta uma funcionalidade de processamento de acordo com modalidades diferentes descritas aqui. Durante uma operação da FDU 110, o processador 113 acessa o sistema de memória 112 através da interconexão 111 de modo a abrir, rodar, executar, interpretar ou efetuar de outra forma as instruções lógicas de um aplicativo de tabela de

estado de porta 140-1. A execução de um aplicativo de
tabela de estado de porta 140-1 desta maneira produz uma
funcionalidade de processamento no processo de tabela de
estado de porta 140-2. Em outras palavras, o processo de
5 tabela de estado de porta 140-2 representa uma ou mais
porções ou instâncias de tempo de rodada de um aplicativo
de tabela de estado de porta 140-1 (ou o aplicativo de
tabela de estado de porta 140-1 inteiro) realizando ou se
executando dentro do ou no processador 113 na FDU 110 no
10 tempo de rodada.

É notado que as configurações de exemplo mostradas
aqui incluem o aplicativo de tabela de estado de porta 140-
1 em si (isto é, na forma de instruções lógicas não
executadas ou não realizadas e/ou dados). O aplicativo de
15 tabela de estado de porta 140-1 pode ser armazenado em um
meio que pode ser lido em computador (tal como um disco
flexível), um disco rígido, um meio que pode ser lido em
computador eletrônico, magnético, ótico ou outro. Um
aplicativo de tabela de estado de porta 140-1 também pode
20 ser armazenado em um sistema de memória 112, tal como em um
firmware, uma memória apenas de leitura (ROM) ou, como
neste exemplo, como um código executável, por exemplo, em
uma memória de acesso randômico (RAM). Além destas
modalidades, também deve ser notado que as outras
25 modalidades aqui incluem a execução de um aplicativo de
tabela de estado de porta 140-1 no processador 113 como o
processo de tabela de estado de porta 140-2. Aqueles
versados na técnica entenderão que o FDU 110 pode incluir
outros processos e/ou componentes de software e de
30 hardware, tal como um sistema operacional não mostrado

neste exemplo.

Durante uma operação, o processador 113 da FDU 110 acessa o sistema de memória 112 através da interconexão 111, de modo a se abrirem, rodarem, executarem, 5 interpretarem ou realizarem de outra forma as instruções lógicas do aplicativo de tabela de estado de porta 140-1. A execução do aplicativo de tabela de estado de porta 140-1 produz uma funcionalidade de processamento no processo de tabela de estado de porta 140-2. Em outras palavras, o 10 processo de tabela de estado de porta 140-2 representa uma ou mais porções do aplicativo de tabela de estado de porta 140-1 (ou o aplicativo inteiro) se realizando dentro do ou no processador 113 no sistema de computador 100.

Deve ser notado que, além do processo de tabela de 15 estado de porta 140-2, as modalidades aqui incluem o aplicativo de tabela de estado de porta 140-1 em si (isto é, as instruções lógicas não executadas ou não realizadas e/ou dados). O aplicativo de tabela de estado de porta 140-1 pode ser armazenado em um meio que pode ser lido em 20 computador, tal como um disco flexível, um disco rígido ou um meio ótico. O aplicativo de tabela de estado de porta 140-1 também pode ser armazenado em um sistema de tipo de memória, como em um firmware, uma memória apenas de leitura (ROM), ou, como neste exemplo, como um código executável no 25 sistema de memória 112 (por exemplo, em uma memória de acesso randômico ou RAM).

Além destas modalidades, também deve ser notado que outras modalidades aqui incluem a execução do aplicativo de tabela de estado de porta 140-1 no processador 113 como o 30 processo de tabela de estado de porta 140-2. Aqueles

versados na técnica entenderão que a FDU 110 pode incluir outros processos e/ou componentes de software e de hardware, tal como um sistema operacional que controle a alocação e o uso de recursos de hardware associados à FDU 110.

Os fluxogramas dos métodos presentemente mostrados são descritos nas Figuras 13 e 14. Os elementos retangulares são denotados aqui "blocos de processamento" e representam instruções de software de computador ou grupos de instruções. Alternativamente, os blocos de processamento representam etapas realizadas por circuitos funcionalmente equivalentes, tal como um circuito de processador de sinal digital ou um circuito integrado específico de aplicação (ASIC). Os fluxogramas não descrevem a sintaxe de qualquer linguagem de programação em particular. Ao invés disso, os fluxogramas ilustram a informação funcional que alguém de conhecimento comum na técnica requer para a fabricação de circuitos ou para a geração de um software de computador para a realização do processamento requerido de acordo com a presente invenção. Deve ser notado que muitos elementos de programa de rotina, tais como a inicialização de laços e de variáveis e o uso de variáveis temporárias, não são mostrados. Será apreciado por aqueles de conhecimento comum na técnica que a menos que indicado de outra forma aqui, a seqüência em particular de etapas descritas é ilustrativa apenas, e pode ser variada, sem que se desvie do espírito da invenção. Assim, a menos que declarado de outra forma, as etapas descritas abaixo são sem ordem, significando que, quando possível, as etapas podem ser realizadas em qualquer ordem conveniente ou desejável.

Com referência, agora, à Figura 13, uma modalidade em particular de um método 200 de manutenção de um banco de dados de estado de porta em um plano de direcionamento de um elemento de rede é mostrada. O método 200 começa com o bloco de processamento 202, o qual mostra periodicamente a determinação de um estado de um primeiro conjunto de portas associadas a uma primeira unidade de dados de encaminhamento (FDU), a primeira FDU sendo uma de uma pluralidade de FDUs implementando funções de encaminhamento no plano de direcionamento do elemento de rede.

O bloco de processamento 204 estabelece o uso do estado determinado para a atualização de uma primeira tabela de estado de porta do banco de dados de estado de porta associado à primeira FDU. Conforme mostrado no bloco de processamento 206, o primeiro banco de dados de estado de porta inclui uma tabela de estado de porta local e uma tabela de estado de porta remota, e onde os estados de enlace físico e os estados de conexão para portas locais são mantidos na tabela de estado de porta local, e onde os estados de enlace físico e os estados de conexão para portas remotas são mantidos na tabela de estado de porta remota. Conforme mostrado adicionalmente no bloco de processamento 208, o processo inclui a manutenção do estado de membro de porta de tronco de enlace múltiplo (MLT) que esteja na primeira FDU na tabela de estado de porta local e a manutenção do estado de membro de porta de tronco de enlace múltiplo (MLT) que esteja nas outras FDUs na tabela de estado de porta remota.

O processamento continua com o bloco de processamento 210, o qual recita a transmissão do estado determinado para

cada outra FDU no elemento de rede para se permitir que as outras FDUs armazenem o estado do primeiro conjunto de portas em um banco de dados de estado de porta local para cada uma das outras FDUs. O bloco de processamento 212
5 mostra a manutenção dos estados de enlace físico e dos estados de conexão para as portas locais e as portas remotas no primeiro banco de dados de estado de porta. O bloco de processamento 214 declara o uso do banco de dados de estado de porta pelo plano de direcionamento para a
10 realização de um novo roteamento rápido de pacotes.

Com referência, agora, à Figura 14, uma modalidade em particular de um método 250 para atualização de uma tabela de estado de porta é mostrado. O método 250 começa com o bloco de processamento 252, o qual mostra o recebimento de
15 um pacote de batimento cardíaco contendo um status de enlace físico local de portas em uma placa de linha contendo a primeira FDU, e onde a primeira FDU usa o status de enlace físico local de portas na placa de linha do pacote de batimento cardíaco para a atualização do banco de
20 dados de estado de porta.

O bloco de processamento 254 declara a manutenção de um temporizador para cada uma das outras FDUs para se determinar se pacotes de batimento cardíaco são recebidos a partir das outras FDUs, onde no caso de um temporizador
25 expirar antes do recebimento de um pacote de batimento cardíaco a partir da FDU associada àquele temporizador, cada porta associada àquela FDU é determinada como estando no estado DESLIGADO, de modo que os pacotes não sejam enviados para portas tendo o estado DESLIGADO, até que a
30 FDU seja restaurada.

O bloco de processamento 256 recita a manutenção de um par de temporizadores locais por porta anexada da primeira FDU, a porta anexada configurada em uma conexão lógica / virtual, onde um dos temporizadores locais é usado para a geração de pacotes de batimento cardíaco a serem enviados pela porta anexada e onde um outro dos temporizadores locais é usado para a detecção de uma falha de conexão da porta anexada, quando uma falha de conexão existir, quando o pacote de batimento cardíaco enviado não for recebido. O bloco de processamento 258 mostra que a primeira FDU atualiza seu banco de dados de estado de porta com a chegada de pacotes de batimento cardíaco e expirações de temporizador.

O(s) dispositivo(s) ou sistemas de computador que se integram ao(s) processador(es) podem incluir, por exemplo, um computador(es) pessoal(is), estação(ões) de trabalho (por exemplo, Sun, HP), assistente(s) digital(is) pessoal(is) (PDA(s)), dispositivo(s) portátil(eis) tal(is) como telefone(s) celular(es), laptop(s), computador(es) portátil(eis), ou um outro dispositivo(s) capaz de ser integrado com um processador(es) que podem operar conforme provido aqui. Assim sendo, o(s) dispositivo(s) provido(s) aqui não é (são) exaustivo(s) e é (são) provido(s) para ilustração e não limitação.

As referências a "um microprocessador" e "um processador" ou "ao microprocessador" e "ao processador" podem ser entendidas como incluindo um ou mais microprocessadores que podem se comunicar em um ambiente(s) independente(s) e/ou distribuído(s), e, assim, podem ser configurados para comunicação através de comunicações com

fio ou sem fio com outros processadores, onde um ou mais processadores como esse podem ser configurados para operarem em um ou mais dispositivos controlados por processador, que podem ser dispositivos similares ou diferentes. O uso dessa terminologia de "microprocessador" ou "processador" assim também pode ser entendida como incluindo uma unidade de processamento central, uma unidade de lógica aritmética, um circuito integrado (IC) específico de aplicação e/ou um motor de tarefa, com esses exemplos providos para ilustração e não limitação.

Mais ainda, referências à memória, a menos que especificadas de outra forma, podem incluir um ou mais elementos de memória que podem ser lidos em processador e acessíveis e/ou componentes que podem ser internos ao dispositivo controlado por processador, externos ao dispositivo controlado por processador, e/ou podem ser acessados através de uma rede com fio ou sem fio usando-se uma variedade de protocolos de comunicação, e, a menos que especificado de outra forma, podem ser dispostos para incluírem uma combinação de dispositivos de memória externos e internos, onde essa memória pode ser contígua e/ou dividida com base na aplicação. Assim sendo, referências a um banco de dados podem ser entendidas como incluindo uma ou mais associações de memória, onde essas referências podem incluir produtos de banco de dados comercialmente disponíveis (por exemplo, SQL, Informix, Oracle) e também bancos de dados proprietários, e também podem incluir outras estruturas para associação de memória, tais como enlaces, filas, gráficos, árvores, com essas estruturas providas para ilustração e não limitação.

As referências a uma rede, a menos que providas de outra forma, podem incluir uma ou mais intranets e/ou a internet, bem como uma rede virtual. As referências aqui a instruções e microprocessador ou instruções executáveis em microprocessador de acordo com o dito acima podem ser entendidas como incluindo um hardware programável.

A menos que declarado de outra forma, o uso da palavra "substancialmente" pode ser construído como incluindo uma relação precisa, uma condição, um arranjo, uma orientação e/ou uma outra característica, e desvios da mesma conforme entendido por alguém de conhecimento comum na técnica, até a extensão em que esses desvios não afetem materialmente os métodos e sistemas mostrados.

Por toda a presente exposição, o uso dos artigos "um" ou "uma" para a modificação de um nome pode ser entendido como sendo feito por conveniência e para incluir um ou mais de um do nome modificado, a menos que especificamente declarado de outra forma.

Elementos, componentes, módulos e/ou partes dos mesmos que são descritos e/ou retratados de outra forma por todas as figuras para comunicação, estarem associados a e/ou serem com base em alguma outra coisa podem ser entendidos para então se comunicarem, estarem associados a ou serem com base em de uma maneira direta e/ou indireta, a menos que estipulado de outra forma aqui.

Embora os métodos e sistemas tenham sido descritos em relação a uma modalidade específica dos mesmos, eles não são limitados dessa forma. Obviamente, muitas modificações e variações podem se tornar evidentes à luz dos ensinamentos acima. Muitas mudanças adicionais nos

detalhes, materiais e arranjo de partes, aqui descritos e ilustrados, podem ser feitas por aqueles versados na técnica.

Tendo descrito as modalidades preferidas da invenção, agora, tornar-se-á evidente para aqueles de conhecimento comum na técnica que outras modalidades incorporando estes mesmos conceitos podem ser usadas. Adicionalmente, o software incluído como parte da invenção pode ser concretizado em um produto de programa de computador que inclui um meio usável em computador. Por exemplo, um meio usável em computador como esse pode incluir um dispositivo de memória legível, tal como um dispositivo de disco rígido, um CD-ROM, um DVD-ROM, ou um disquete de computador, tendo segmentos de código de programa que podem ser lidos em computador armazenados ali. O meio que pode ser lido em computador também pode incluir um enlace de comunicações, ótico, com fio ou sem fio, tendo segmentos de código de programa portados ali como sinais digitais ou analógicos. Assim sendo, é submetido que a invenção não deve ser limitada às modalidades descritas, mas, ao invés disso, deve ser limitada apenas pelo espírito e pelo escopo das reivindicações em apenso.

REIVINDICAÇÕES

1. Método de manutenção de um banco de dados de estado de porta em um plano de direcionamento de um elemento de rede caracterizado pelo fato de compreender as etapas de:

5 determinar periodicamente um estado de um primeiro conjunto de portas associadas a uma primeira unidade de dados de encaminhamento (FDU), a primeira FDU sendo uma de uma pluralidade de FDUs implementando funções de encaminhamento no plano de direcionamento do elemento de rede;

10 uso do estado determinado para a atualização de uma primeira tabela de estado de porta do banco de dados de estado de porta associado à primeira FDU; e

15 transmissão do estado determinado para cada uma das outras FDUs no elemento de rede, para se permitir que cada uma das referidas outras FDUs armazene o estado do primeiro conjunto de portas em um banco de dados de estado de porta local para cada uma das referidas outras FDUs.

20 2. Método, de acordo com a reivindicação 1, caracterizado pelo fato de ainda compreender a manutenção de estados de enlace físico e estados de conexão para portas locais e portas remotas no referido primeiro banco de dados de estado de porta.

25 3. Método, de acordo com a reivindicação 2, caracterizado pelo fato de o referido banco de dados de estado de porta incluir uma tabela de estado de porta local e uma tabela de estado de porta remota, e pelo fato de os estados de enlace físico e os estados de conexão para as portas locais serem mantidos na referida tabela de estado de porta local e pelo fato de os estados de enlace físico e

30

os estados de conexão para as portas remotas serem mantidos na referida tabela de estado de porta remota.

4. Método, de acordo com a reivindicação 3, caracterizado pelo fato de ainda compreender a manutenção do estado de membro de porta de tronco de enlace múltiplo (MLT) que está na referida primeira FDU na referida tabela de estado de porta local e a manutenção do estado de membro de porta de tronco de enlace múltiplo (MLT) que está nas referidas outras FDUs na referida tabela de estado de porta remota.

5. Método, de acordo com a reivindicação 1, caracterizado pelo fato de ainda compreender o recebimento de um pacote de batimento cardíaco contendo um status de enlace físico local de portas em uma placa de linha contendo a referida primeira FDU, e pelo fato de a referida primeira FDU usar o status de enlace físico local de portas na placa de linha do referido pacote de batimento cardíaco para a atualização do referido banco de dados de estado de porta.

6. Método, de acordo com a reivindicação 5, caracterizado pelo fato de ainda compreender a manutenção de um temporizador para cada uma das referidas outras FDUs para se determinar se os pacotes de batimento cardíaco são recebidos a partir das outras FDUs, onde, no caso de um temporizador expirar antes do recebimento de um pacote de batimento cardíaco a partir da FDU associada àquele temporizador cada porta associada àquela FDU ser determinada como estando no estado DESLIGADO, de modo que os pacotes não sejam enviados para portas tendo o estado DESLIGADO, até aquela FDU ser restaurada.

7. Método, de acordo com a reivindicação 6, caracterizado pelo fato de ainda compreender a manutenção de um par de temporizadores locais por porta anexada da referida primeira FDU, a porta anexada configurada em uma conexão lógica / virtual, pelo fato de um dos referidos temporizadores locais ser usado para a geração de pacotes de batimento cardíaco a serem enviados pela referida porta anexada, e pelo fato de um outro dos referidos temporizadores ser usado para a detecção de uma falha de conexão da porta anexada, quando uma falha de conexão existir, quando o pacote de batimento cardíaco enviado não for recebido.

8. Método, de acordo com a reivindicação 7, caracterizado pelo fato de a referida primeira FDU atualizar seu banco de dados de estado de porta com a chegada de pacotes de batimento cardíaco e expirações de temporizador.

9. Elemento de rede caracterizado pelo fato de compreender:

uma memória;
um processador;
uma interface de comunicações;
um mecanismo de interconexão acoplando a memória, o processador e a interface de comunicações; e

onde a memória é codificada com um aplicativo mantendo um banco de dados de estado de porta, que, quando executado no processador, provê um processo para processamento de uma informação, o processo fazendo com que o elemento de rede realize as operações de:

determinar periodicamente um estado de um primeiro

conjunto de portas associadas a uma primeira unidade de dados de encaminhamento (FDU), a primeira FDU sendo uma de uma pluralidade de FDU's implementando funções de encaminhamento no plano de direcionamento do elemento de rede;

uso do estado determinado para a atualização de uma primeira tabela de estado de porta do banco de dados de estado de porta associado à primeira FDU; e

transmissão do estado determinado para cada uma das outras FDU's no elemento de rede, para se permitir que cada uma das referidas outras FDU's armazene o estado do primeiro conjunto de portas em um banco de dados de estado de porta local para cada uma das referidas outras FDU's.

10. Meio de armazenamento que pode ser lido em computador caracterizado pelo fato de ter um código que pode ser lido em computador para manutenção de um banco de dados de estado de porta, o meio incluindo instruções nas quais um sistema de computador realiza operações compreendendo:

determinar periodicamente um estado de um primeiro conjunto de portas associadas a uma primeira unidade de dados de encaminhamento (FDU), a primeira FDU sendo uma de uma pluralidade de FDU's implementando funções de encaminhamento no plano de direcionamento do elemento de rede;

uso do estado determinado para a atualização de uma primeira tabela de estado de porta do banco de dados de estado de porta associado à primeira FDU; e

transmissão do estado determinado para cada uma das outras FDU's no elemento de rede, para se permitir que cada

uma das referidas outras FDUs armazene o estado do primeiro conjunto de portas em um banco de dados de estado de porta local para cada uma das referidas outras FDUs.

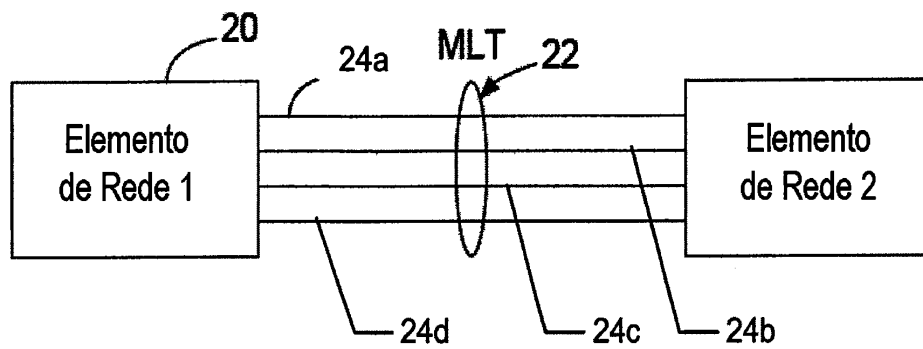
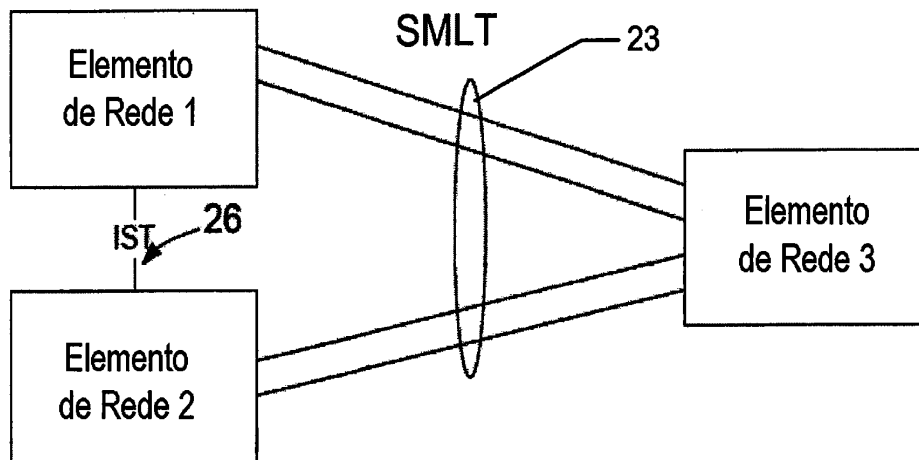
Figura 1**Figura 2**

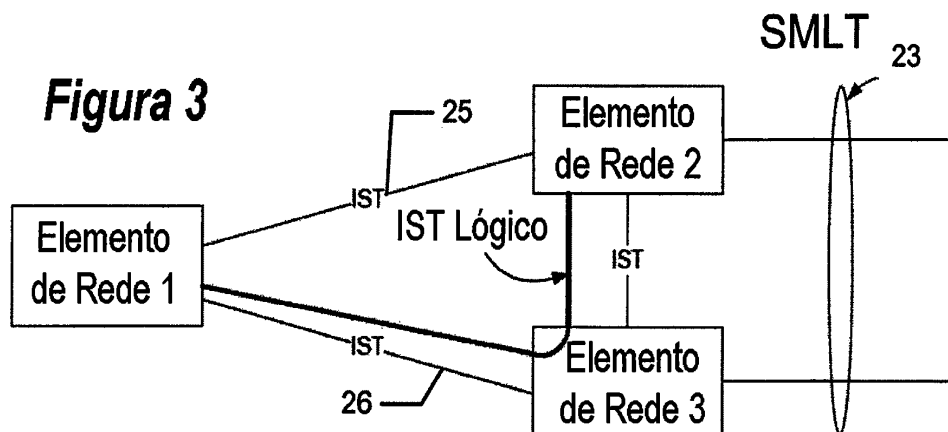
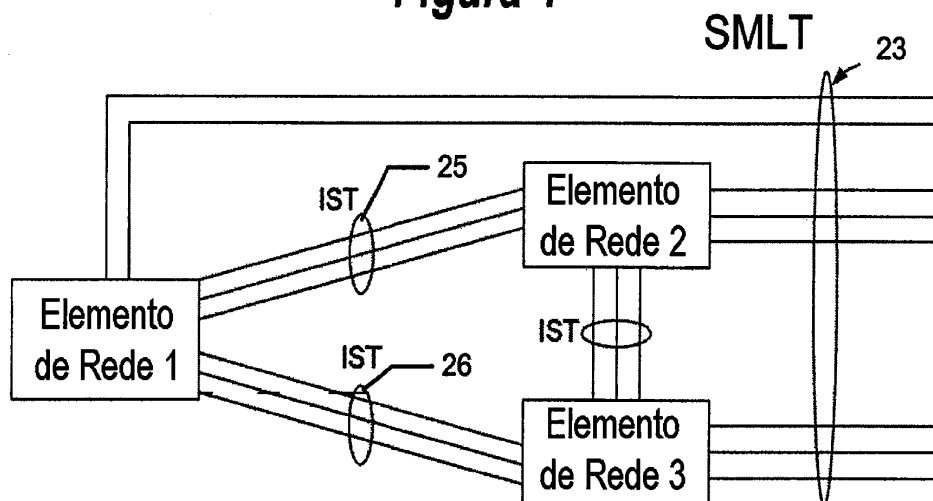
Figura 3**Figura 4**

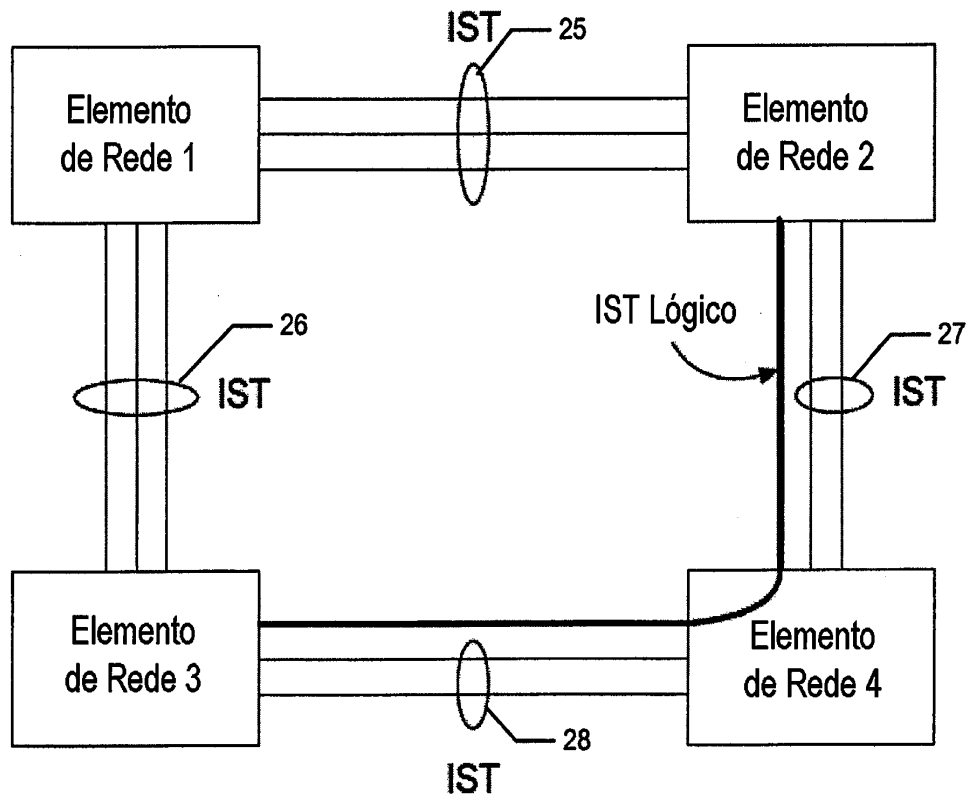
Figura 5

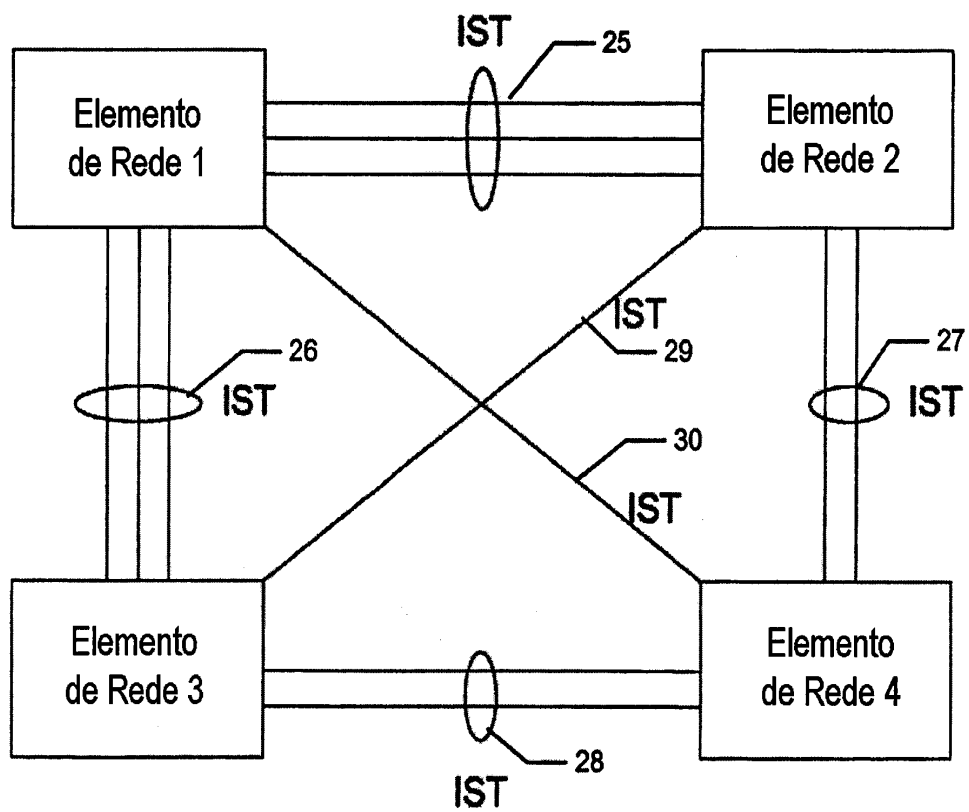
Figura 6

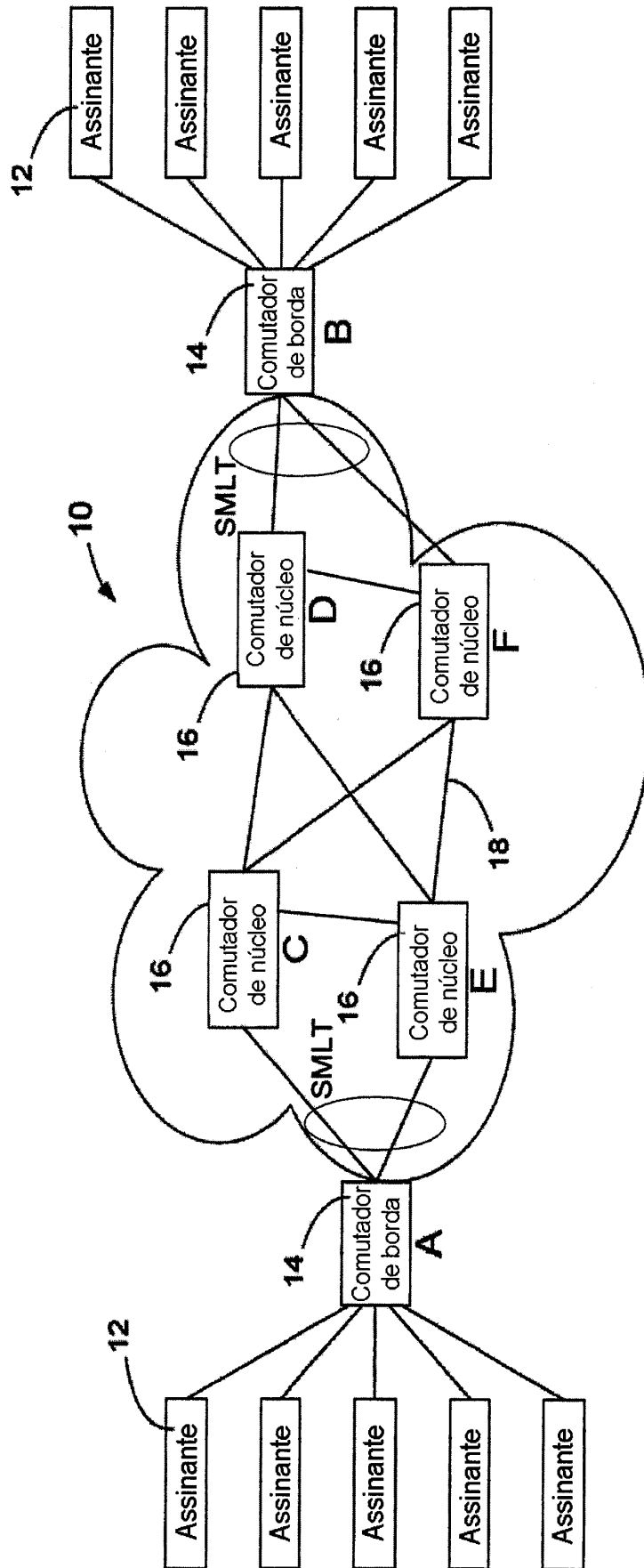
Figura 7

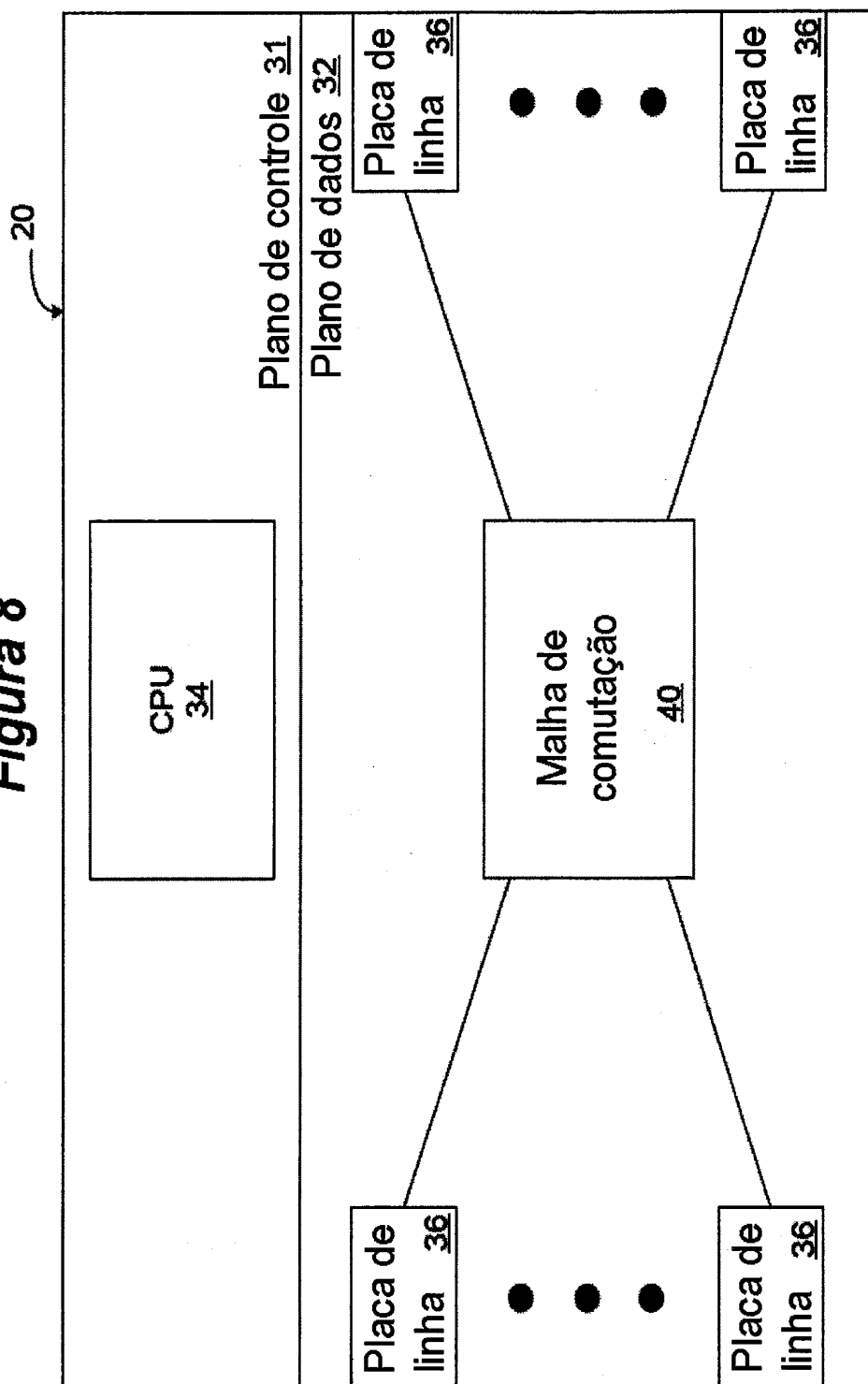
Figura 8

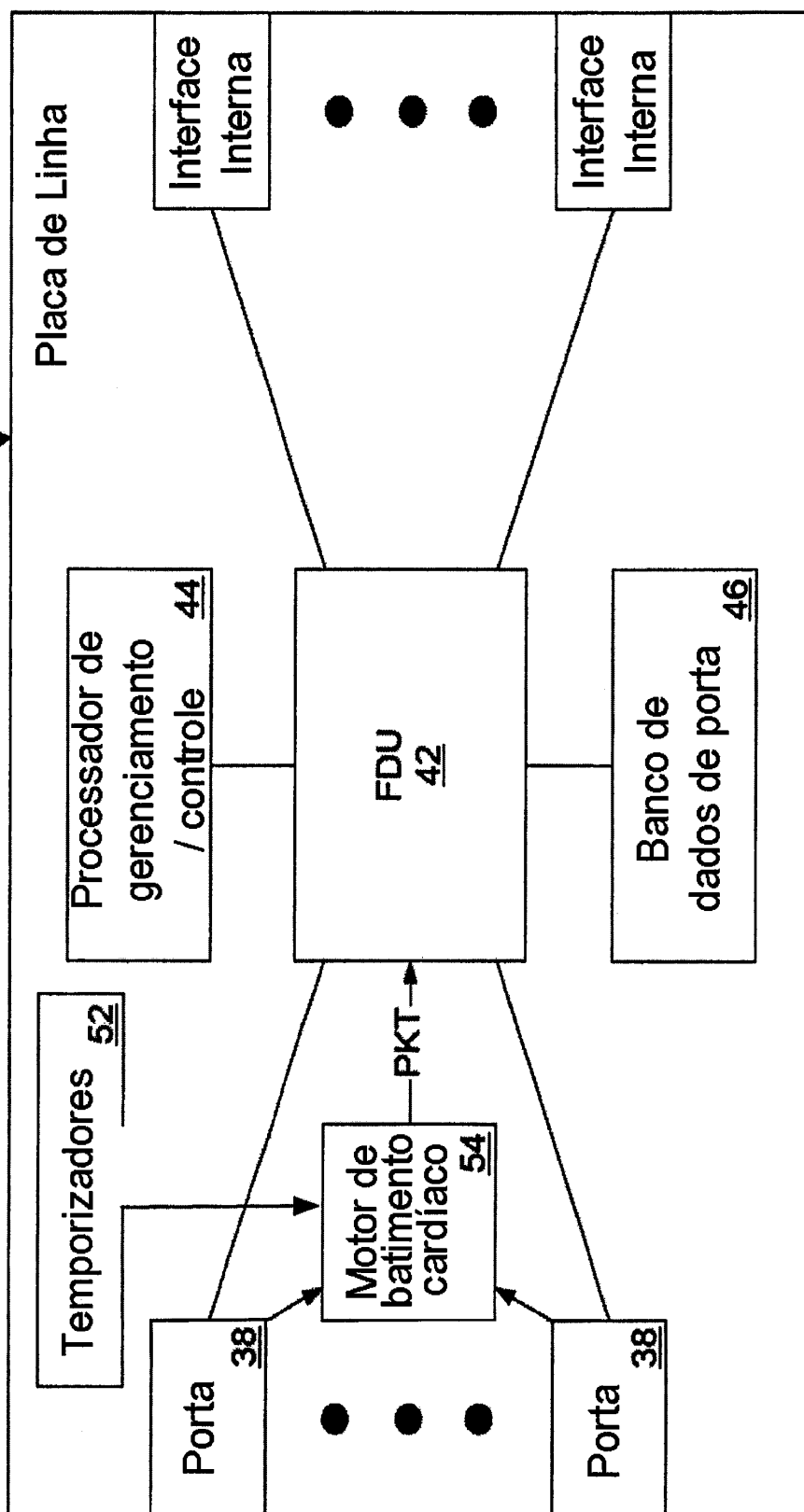
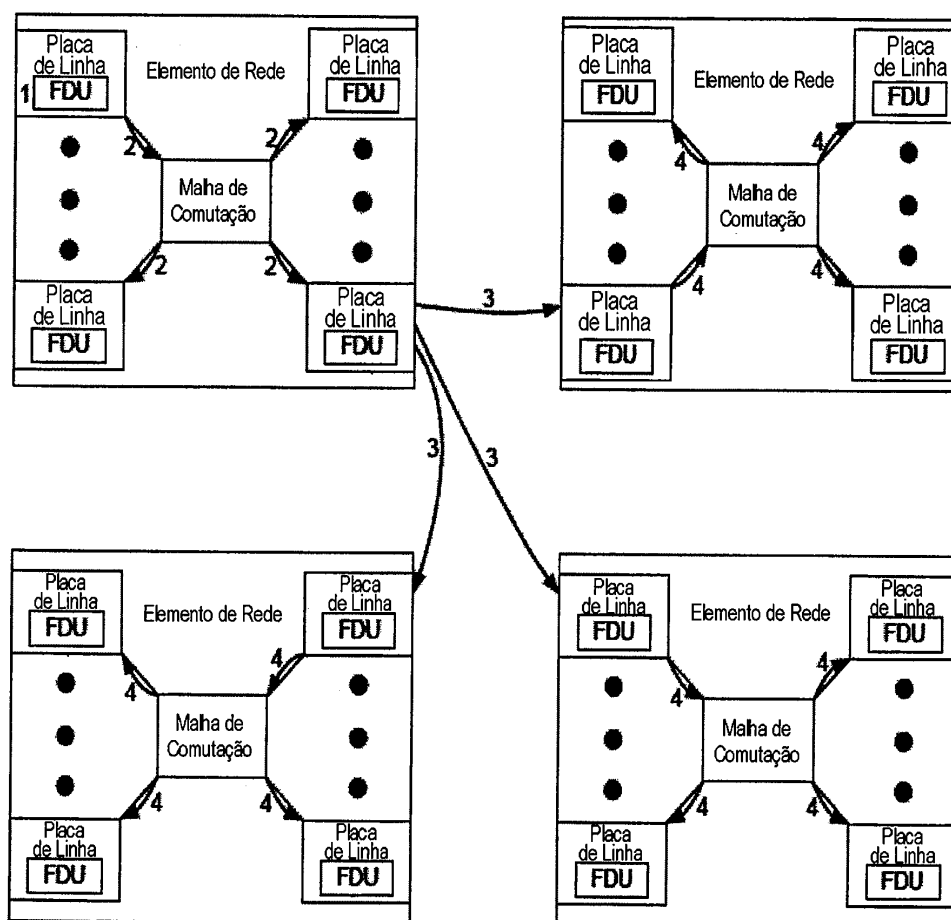
Figura 9

Figura 10

Banco de datos de porta <u>46</u>	
Lógica de controle de banco de dados	Dados de porta local
	<u>48</u>
	Dados de porta remota
	<u>50</u>
	<u>56</u>

Figura 11

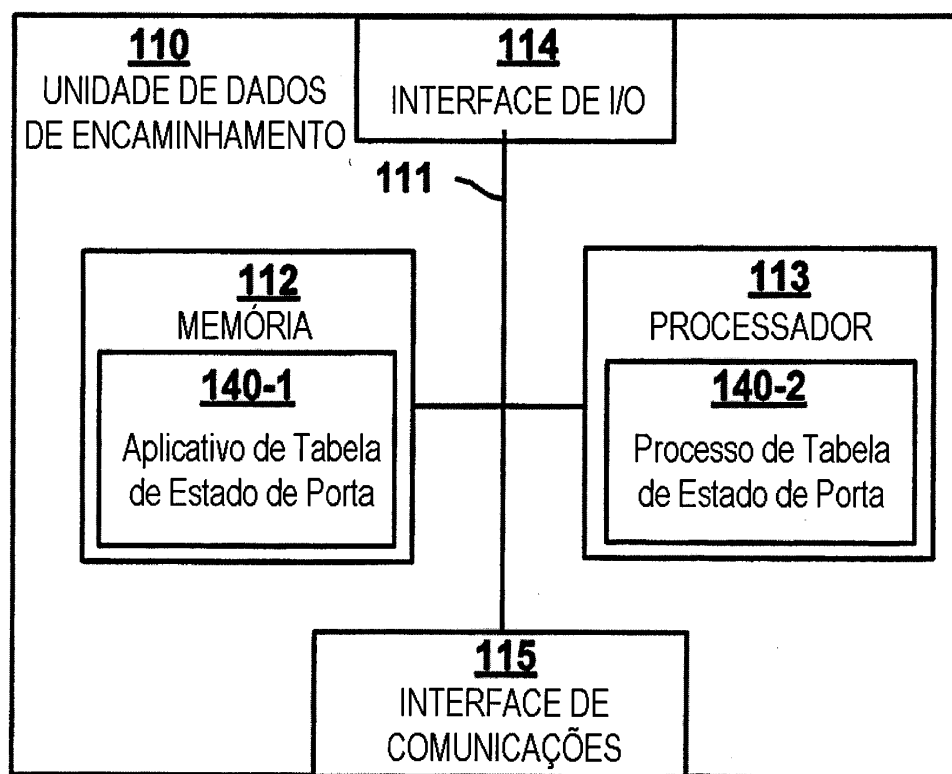
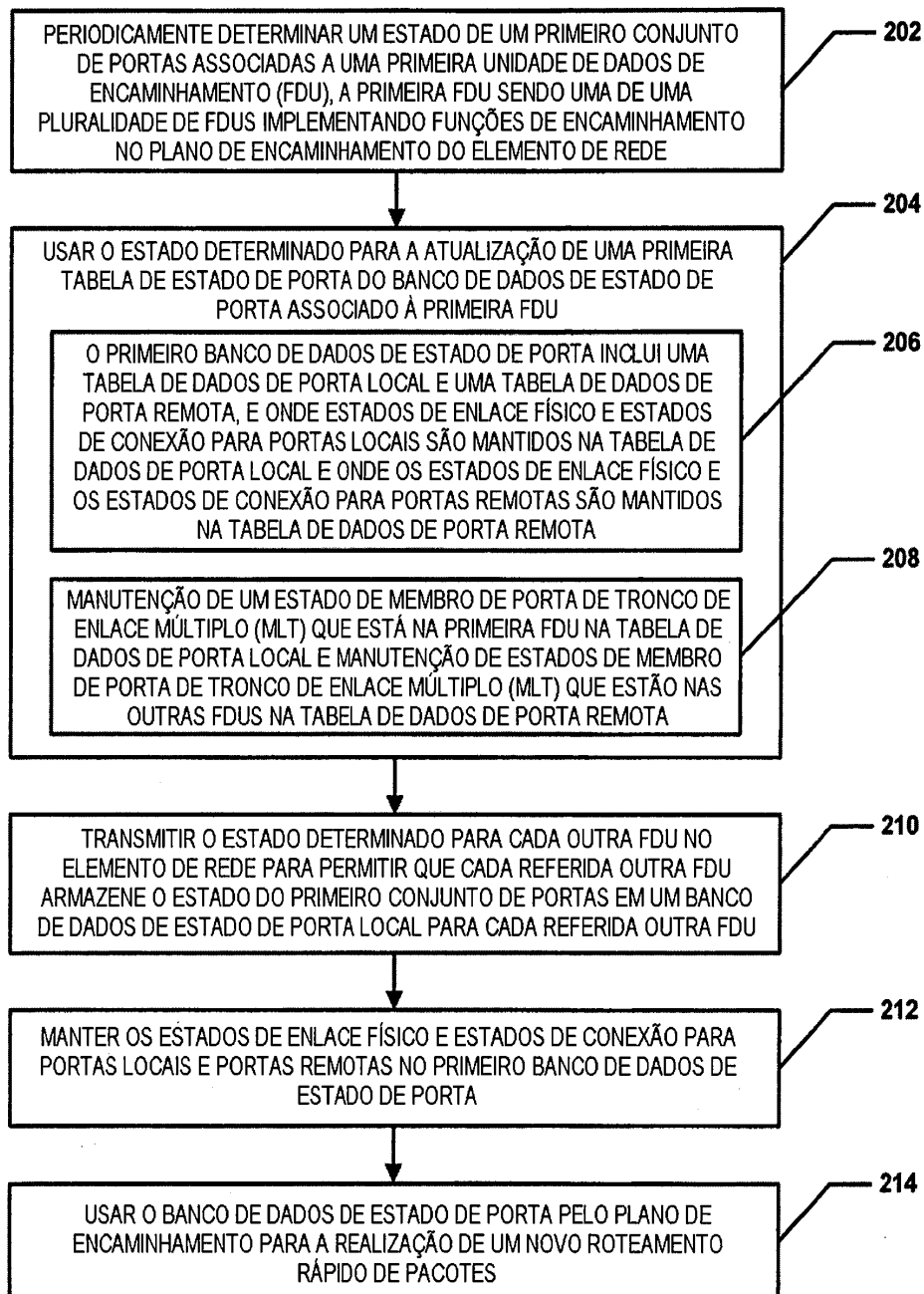
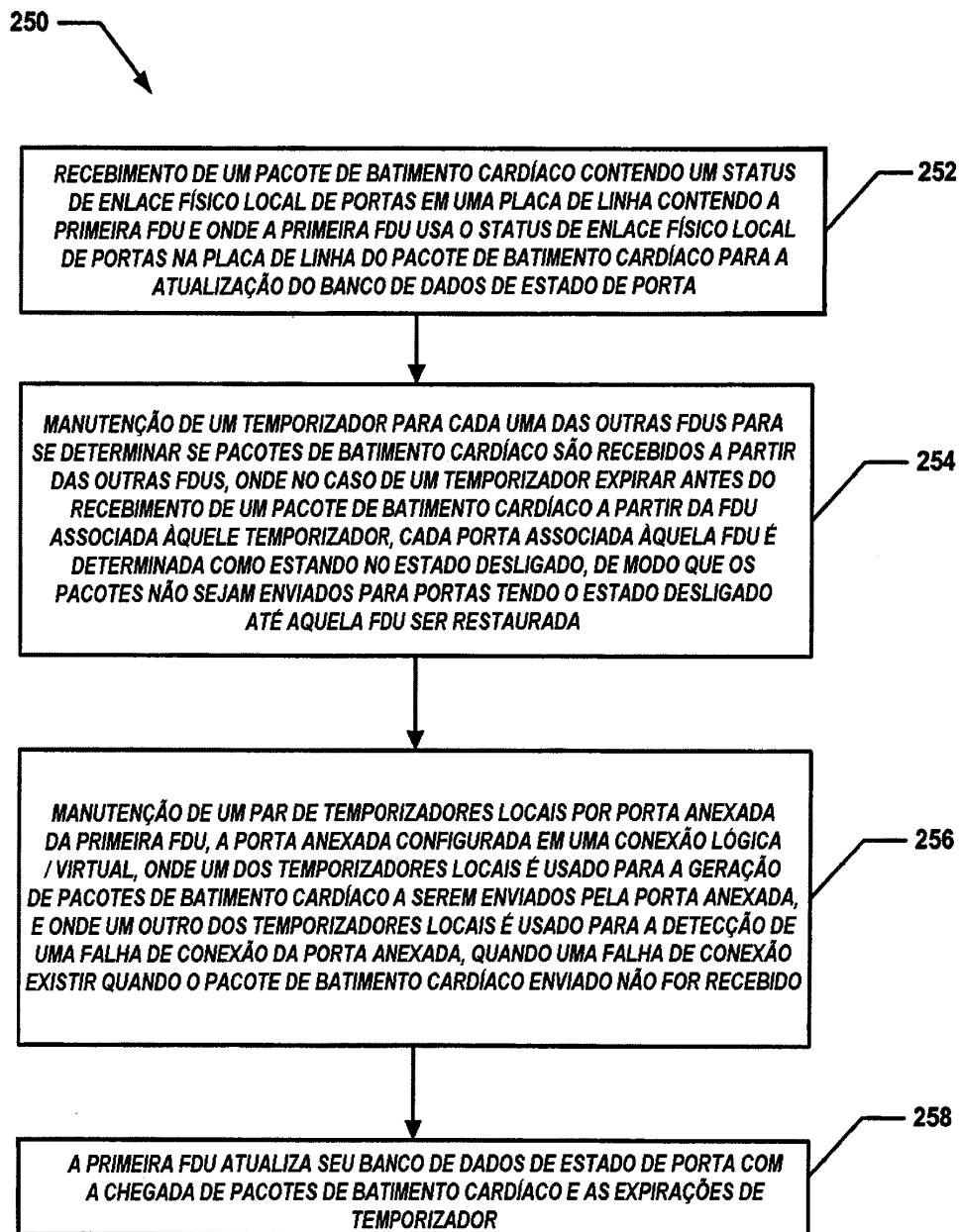


Figura 12

200

**FIGURA 13**

**FIGURA 14**

RESUMO**MÉTODO E APARELHO PARA MANUTENÇÃO DE TABELAS DE ESTADO DE PORTA EM UM PLANO DE DIRECIONAMENTO DE UM ELEMENTO DE REDE**

Um método, um aparelho e um produto de programa de
5 computador para a manutenção de tabelas de estado de porta
em um plano de direcionamento de um elemento de rede são
apresentados. O estado de um primeiro conjunto de portas
associadas a uma primeira unidade de dados de
encaminhamento (FDU) é periodicamente determinado, a
10 primeira FDU sendo uma de uma pluralidade de FDUs. O estado
determinado é usado para a atualização de uma primeira
tabela de estado de porta do banco de dados de estado de
porta associado à primeira FDU. O estado determinado é
transmitido para cada uma das outras FDUs no elemento de
15 rede, para se permitir que cada uma das outras FDUs
armazene o estado do primeiro conjunto de portas em um
banco de dados de estado de porta local para cada uma das
outras FDUs. O banco de dados de estado de porta é usado
pelo plano de direcionamento para a realização de um novo
20 roteamento rápido de pacotes.