



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

(11) PI 0400304-7 B1

(22) Data do Depósito: 23/03/2004

(45) Data de Concessão: 05/07/2016



(54) Título: MÉTODO DE BUSCA DE DADOS DE GUIA DE PROGRAMAÇÃO ELETRÔNICA (EPG) E MECANISMO DE BUSCA PARA DADOS DE GUIA DE PROGRAMAÇÃO ELETRÔNICA (EPG)

(51) Int.Cl.: G06F 7/10

(30) Prioridade Unionista: 24/03/2003 US 10/395.679

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

(72) Inventor(es): SCOTT D. SANDERS

Relatório Descritivo da Patente de Invenção para “**MÉTODO DE BUSCA DE DADOS DE GUIA DE PROGRAMAÇÃO ELETRÔNICA (EPG) E MECANISMO DE BUSCA PARA DADOS DE GUIA DE PROGRAMAÇÃO ELETRÔNICA (EPG)**”.

LISTAGEM DE PROGRAMA DE COMPUTADOR

[001] Um conjunto de listagens de arquivo de texto XML usado em conformidade com o assunto em questão é apresentado no anexo após o resumo, em três folhas de papel e está incorporado ao presente relatório descritivo à guisa de referência. A listagem de arquivo de texto XML consiste em uma listagem exemplificativa de atributos (por exemplo, o bloco 310 da Figura 3).

CAMPO DA TÉCNICA

[002] A presente invenção refere-se, em geral, a comunicações de dados em multimídia e, especificamente, ao texto livre e busca de atributos de dados de guia de programação eletrônica (EPG).

ANTECEDENTES

[003] Em um sistema de televisão digital, os usuários buscam, tipicamente, informações com visualização de programas em um guia de programação eletrônica (EPG). Um EPG consiste em um banco de dados de programa e um metadado de programação que se refere a programas disponibilizados a partir de um ponto central (“*headend*”) (uma grade de canais distribuídos (*lineup*) pelo provedor da programação), durante um intervalo de tempo, por exemplo, uma semana. Um usuário procura por programas, designando os termos da palavra-chave, a fim de realizar uma busca por texto de títulos de programas. Um resultado da busca, tipicamente o título de um programa, que se encaixa nos critérios da busca é conhecido como um “acerto”.

[004] Os programas de televisão também podem ser encontrados por busca de atributos de programas, não apenas por palavras-chave do título do programa. Embora o título de um programa possa ser vi-

sualizado como um atributo do programa, o termo “atributo” é utilizado mais tecnicamente nas artes da multimídia para descrever muitas características sem títulos de um programa. Como tal, os atributos incluem uma miríade de características sobre um programa e sua apresentação. Um programa e seus atributos se encaixam, tipicamente, dentro de uma vasta hierarquia de informações EPG. Alguns atributos, tais como “categoria”, podem descrever um amplo conjunto de programas, dos quais o programa que está sendo visado é um membro, ou seja, o atributo de “categoria” descreve um nível de hierarquia inclusive do programa que está sendo visado, de modo que a busca de um usuário pelo atributo de “categoria” resulta em acertos de cada programa na categoria. Uma busca por “westerns” (faroestes), por exemplo, pode resultar em muitos acertos que pertencem ao filme faroeste e gênero de televisão.

[005] Os atributos também podem descrever características próprias para o título de um programa, isto é, atributos podem descrever opções que fazem com que duas versões do mesmo programa variem de uma para outra. Este tipo mais estreito de atributo descreve elementos da hierarquia EPG “abaixo” de um programa particular, isto é, que faz com que o mesmo programa tenha um quê de diferente. Por exemplo, uma versão do programa “The Mountain”, estrelando Spencer Tracy, pode ter atributos em “preto & branco” e “monofônico”, enquanto outra versão possui atributos de “cor”, “Digital DOLBY®”, “legenda oculta” (*close captioning*), e “classificação para menores” (*rated PG*). Uma busca de texto limitada a elementos da palavra-chave do título irá manifestar ambas as versões como acertos, considerando-se que uma busca de texto por elementos de palavra-chave do título combinado com uma busca de atributo para “Dolby” irá manifestar apenas as versões do sistema de som DOLBY® digital e DOLBY® Estéreo.

[006] Uma busca de texto convencional para um elemento de palavra-chave do título é realizada quando se ingressa o texto de palavra-chave dentro de um campo de busca da interface de um usuário (UI), e uma busca de atributo convencional é realizada quando se ingressa o critério do atributo (por exemplo, em caixas de seleção), separadamente a partir das palavras-chave de busca de texto em uma parte separada da UI, ou em uma UI conjuntamente separado. Os acertos provenientes de uma busca de texto separada e a busca de atributo separada são combinados de forma lógica apenas no final de ambas as buscas para produzir uma lista de acertos.

[007] A Figura 1 mostra uma busca de texto convencional 100 e uma busca de atributo convencional 102, em que as palavras chave e os critérios de atributo são ingressados separadamente, tal como em um campo de busca separado das caixas de seleção de atributo que são exibidas através de uma ou mais UIs.

[008] No bloco 104, a(s) palavra(s)-chave de texto são ingressadas. No bloco 106, os critérios de atributo são ingressados separadamente ao se ingressar a(s) palavra(s)-chave. Uma busca de texto é realizada 108, tipicamente no título do programa, usando a(s) palavra(s)-chave e uma busca de atributo é realizada 110 usando os critérios de busca dos atributos ingressados. Cada busca separada produz resultados separados 112, 114. A filtragem da lógica 116 pode comparar os resultados 112, 114 e produz uma lista de acertos 118 que satisfaz tipicamente tanto a busca de texto 100 como a busca de atributo 102.

[009] Uma vez que aos usuários se solicita tipicamente a busca de atributos em uma parte separada de uma UI, pode-se restringir sua entrada de busca para indagações de estilo da palavra-chave das palavras do título do programa apenas, ao mesmo tempo em que outros ingressam incorretamente os atributos como palavras chave para uma

busca de texto de títulos de programa, sem se dar conta que o sistema de televisão digital interpreta os atributos ingressados desta forma como termos comuns de busca por título de programa.

SUMÁRIO

[0010] O objetivo da presente invenção inclui um mecanismo de busca para dados de guia de programa eletrônico (EPG) e métodos relacionados. Em um método exemplificativo, uma cadeia de busca de texto pode ser normalizada formando termos passíveis de busca e os termos interpretados, tanto como termos de busca de texto, ou como termos de busca de atributo. As perguntas que usam a lógica de busca de complexidade variada são criadas de acordo com a interpretação dos termos na cadeia de busca. Uma ou mais buscas em bancos de dados EPG e/ou recursos em Rede de EPG são realizadas para cada texto e cada termo de atributo e os resultados são dados uma classificação de relevância, de acordo com a interpretação de cada cadeia de busca. Os resultados de busca combinados podem ser classificados, filtrados e agrupados para exibição ao usuário. Os resultados também podem ser exibidos progressivamente, na medida em que o usuário ingressa cada caracter de uma cadeia de busca.

[0011] Em uma implementação, uma lista de atributos pode ser usada para separar os termos do texto dos termos do atributo. Uma lista exemplificativa pode conter adicionalmente índices ou metadados para cada atributo que dirige o curso das buscas realizadas para cada termo de atributo na cadeia da busca.

BREVE DESCRIÇÃO DOS DESENHOS

[0012] A Figura 1 é uma representação gráfica do texto da técnica anterior e buscas de atributo.

[0013] A Figura 2 é uma representação gráfica de uma busca exemplificativa, de acordo com o objeto da presente invenção.

[0014] A Figura 3 é um diagrama em bloco de um mecanismo de

busca exemplificativo.

[0015] A Figura 4 é uma representação gráfica de uma busca exemplificativa realizada pelo mecanismo de busca exemplificativo da Figura 3.

[0016] A Figura 5 é uma representação gráfica de uma busca progressiva exemplificativa realizada pelo mecanismo de busca exemplificativo da Figura 3.

[0017] A Figura 6 é um fluxograma de um método exemplificativo, de acordo com o objeto da presente invenção.

[0018] A Figura 7 é um diagrama em bloco de um ambiente do dispositivo de computação exemplificativo, no qual se pratica o objeto da presente invenção.

DESCRIÇÃO DETALHADA

VISÃO GERAL

[0019] A busca rápida, intuitiva e precisa de dados do guia de programação eletrônica (EPG), é importante para os usuários finais em sistemas de televisão digital, incluindo tanto os sistemas com "codificadores" acoplados a um aparelho de televisão, como os sistemas por computador pessoal (PC), capazes de receberem e registrarem as difusões televisivas. Uma matéria exemplificativa descreve um mecanismo de busca e métodos relacionados para ingressar e interpretar o texto de um usuário e as perguntas de atributo em um sistema de televisão digital. Este mecanismo de busca rápida e precisa interpreta, por exemplo, uma única entrada da cadeia de texto, feita por um usuário e indica a intenção provável do usuário, exibindo um conjunto de resultados de acertos que tenta se encaixar na interpretação do quê o usuário está buscando. Estas capacidades de busca avançada são importantes, na medida em que os mercados de gravador de vídeo digital em PC (DVR) e de aparelhos codificadores se expandem rapidamente.

[0020] A Figura 2 proporciona uma visão geral de uma busca

exemplificativa 200 nos bancos de dados EPG ou outros recursos. A busca exemplificativa 200 começa com uma cadeia de busca 202 ingressada por ou recebida de um usuário ou recuperada de um armazenamento. Um mecanismo de busca exemplificativo 204 recebe a cadeia de busca 202, realizando inúmeras transformações nos termos da cadeia para gerar possíveis instruções de pergunta de vários níveis de complexidade.

[0021] Uma única cadeia de busca 202 pode gerar uma ou mais buscas independentes, dependendo da complexidade e da indeterminação da cadeia de busca 202 e, desta forma, de acordo com uma implementação, um intérprete de cadeia de 214 tenta compor, o mínimo possível de perguntas (uma se possível) e o tão restritas quanto possível, com base na cadeia particular 202. Por exemplo, em uma implementação a cadeia 214 alcança esse número mínimo de perguntas concisas e/ou restritivas, determinando quais os termos na cadeia de busca 202 são atributos conhecidos a serem pesquisados como termos de busca de atributo. Os termos restantes na cadeia de busca 202 podem ser termos de atributo possíveis ou termos de texto, mas quando se determina os atributos conhecidos (por exemplo, usando um banco de dados ou lista de atributos conhecidos), muitas vezes se restringe consideravelmente uma pergunta. Sendo assim, o intérprete de cadeia 214 “considera” a série de busca de textos 206, as buscas de atributo 208, as buscas de texto condicionais 210, e as buscas de atributo condicionais 212, usando vários métodos e recursos, conforme tratado mais completamente abaixo, tomando-se como referência a Figura 3.

[0022] Em uma implementação, uma determinação de força bruta do provável significado da cadeia 202 é concluída ao se determinar uma busca preliminar em cada termo normalizado na cadeia de busca 202, tanto em um banco de dados EPG e/ou em um dicionário online e

classificação de resultados. Por exemplo, se “Ryan” for um termo de busca, então “Saving Private Ryan” (“O Resgate do Soldado Ryan”), poderia ser selecionado passando por “Ryan O’Neal”, como a interpretação da busca se fosse mais comum no banco de dados ou no dicionário. Em outra implementação, o intérprete de cadeia 214 tenta diferentes combinações dos termos da busca até que uma das combinações fosse encontrada nos recursos EPG sendo pesquisados ou fosse favorecida de forma lógica. Por conseguinte, os termos de busca “lua” e “papel” ocorrendo aleatoriamente, mas como os únicos termos de texto entre uma host de termos de atributos em uma cadeia de busca 202, resultassem na interpretação “Lua de Papel”, que seria classificado favoravelmente se realmente fosse encontrado como um acerto através de uma pergunta.

[0023] O intérprete da cadeia 214 chega, assim, em uma ou mais interpretações, que resultam em uma pergunta 216 (ou perguntas). A pergunta 216 é usada para realizar uma ou mais buscas resultando em uma lista de acertos combinados, filtrados e/ou classificados 222, que é exibida para o usuário. Esta busca exemplificativa 200 é apenas um exemplo de como uma ou mais buscas sensíveis de texto e de atributo podem se derivar a partir de apenas uma cadeia de busca 202. A cadeia de busca 202 é analisada, transformada e verificada, por exemplo, pelo intérprete de cadeia 214, para conseguir uma interpretação consciente de atributo e de texto dos termos na cadeia de busca 202.

[0024] Em uma implementação, a lista de acerto combinada, filtrada e/ou classificada 222 é atualizada como vários resultados de retorno de trajetórias de busca. Alternativamente, os resultados podem aparecer progressivamente na lista de acerto combinado, filtrado e/ou classificado 222, como a cadeia de busca 202 que está sendo ingresada, isto é, uma busca nova e/ou modificada é iniciada na medida em

que cada caracter é adicionado na cadeia de busca 202.

MECANISMO DE BUSCA EXEMPLIFICATIVO

[0025] A Figura 3 mostra o mecanismo de busca exemplificativo 204 da Figura 2 em maiores detalhes. O mecanismo de busca exemplificativo 204 é acoplado de forma comunicativa a um ou mais bancos de dados 300. Certos recursos e processos executados pelo mecanismo de busca exemplificativo 204 serão descritos agora, mas, logicamente, um mecanismo de busca exemplificativo 204 pode ter recursos e executar processos que são adicionais aos ilustrados, por exemplo, indexando o conteúdo dos bancos de dados. Várias implementações do mecanismo de busca exemplificativo 204, logicamente, irão funcionar com vários idiomas humanos diferentes. Sendo assim, alguns recursos do mecanismo de busca exemplificativo 204 podem não se aplicar ou podem não ser necessários com alguns idiomas.

[0026] De um modo geral, o mecanismo de busca exemplificativo 204 executa busca(s) no banco de dados EPG, primeiramente expondo um receptor de cadeia de busca 301, tal como uma interface de programa de aplicativo (API), que recebe expressões de busca e acessa um conjunto de restrições geradas tanto por meio de programação ou pelo usuário. Depois de criar perguntas com base parcialmente nas restrições, o mecanismo de busca exemplificativo 204 retorna ao conjunto de resultados.

[0027] O receptor de cadeia de busca 301 pode aceitar a cadeia de busca 202 através de uma ou mais UIs. As UIs para busca de dados EPG podem ser otimizadas para um sistema de “dois-rodapés” ou de “dez-rodapés”. Um sistema de dois rodapés consiste de um sistema de televisão digital, no qual o usuário possui a capacidade de ingressar com as informações, típica em um sistema de PC. Por exemplo, o sistema de dois rodapés poderia utilizar um teclado completo, um mouse e um monitor. Um sistema de dez rodapés, por outro lado, con-

siste de um sistema de televisão digital, no qual o usuário possui capacidade limitada de ingressar dados, em comparação com um sistema de dois rodapés. Um sistema de dez rodapés poderia ser gerado pela UI, por meio de um aparelho codificador e uma cadeia de busca 202 ingressada usando-se um controle remoto com relativamente poucos atributos a serem atuados.

[0028] Devido à entrada de dados e à navegação junto a uma UI, de modo geral, ser mais difícil com um sistema de dez rodapés, é útil minimizar a quantidade de dados necessários para capturar a intenção do usuário de ingressar uma busca dentro do sistema de dez rodapés, ao mesmo tempo em que maximiza a precisão e a potência da busca.

[0029] O receptor da cadeia de busca 301, logicamente, pode aceitar a entrada ingressada por um usuário a partir de qualquer número de campos de entrada, caixas de seleção, comutadores e/ou dispositivos. Se a entrada for apenas uma única cadeia de busca 202, no entanto, o mecanismo de busca exemplificativo 204 pode “decodificar” a cadeia de busca para separar do texto e da busca de atributo, informações e construir perguntas apropriadas.

[0030] Um normalizador 302 acoplado comunicativamente com o receptor de cadeia de busca 301 converte irregularidades (isto é, converte o formato dos termos “não padronizados) na cadeia de busca 202 formando termos que podem ser aplicados de forma consistente nas várias perguntas. Uma entrada não padronizada consiste em um termo ou uma parte da cadeia de busca 202 que possui texto, caracter, idioma e/ou formalidades de pontuação que não se encaixam nas formalidades do(s) banco(s) de dados EPG 300 a ser(em) pesquisado(s). Uma entrada não-padronizada corre o risco de não ser compreensível em parte ou no todo ao intérprete da cadeia 304, e poderia resultar na falta de qualificação do acerto quando suas formalidades não se encaixam no(s) banco(s) de dados EPG 300 a ser(em) pesqui-

sado(s).

[0031] Como parte da solicitação de busca, o receptor da cadeia de busca 301 muitas vezes passa por uma cadeia ingressada pelo usuário, por exemplo, “A Asa do Oeste”. O normalizador 302, ou outro módulo do mecanismo de busca exemplificativo 204, pode empregar a separação de termo para tentar dividir a frase em cada uma das palavras que a compõe, realizando uma busca de texto para cada palavra individualmente. Esta separação formando palavras individuais possui a vantagem de encontrar correspondências quando o usuário omitiu ou redispôs as palavras em um título bem conhecido, mas isso também gera mais acertos que o pretendido pelo usuário. Os termos individuais da busca por palavra muitas vezes é separado com espaços em branco e limitado por traços. O normalizador 302 também pode remover espaços extra entre as palavras de uma frase citada.

[0032] Os usuários nem sempre ingressam com uma cadeia de busca 202, usando a letra correta. Por exemplo, a frase “A Asa do Oeste” pode ser datilografado como “a asa do oeste”. Adicionalmente, as informações do banco de dados EPG, algumas vezes contém letras não convencionais em frases como “BILHETE DE DOMINGO”, ao invés de “Bilhete de Domingo”, ou “bilhete de domingo”. O normalizador 302 pode realizar a normalização da carteira de letras, convertendo cada termo de busca e também o texto do banco de dados EPG que está sendo pesquisado, em uma carteira de letras comum (tanto para cima como para baixo), por exemplo, seguindo as normas do local atual.

[0033] O normalizador 302 também pode normalizar determinados símbolos e marcas. Uma marca diacrítica consiste de um caracter de combinação, tal como um acento ou um trema que pode aparecer em muitos alfabetos de idiomas internacionais, normalmente acompanhando as vogais. Em uma implementação da matéria do presente

pedido, os termos da busca e o texto do banco de dados que está sendo pesquisado são comparados como se as marcas diacríticas não estivessem presentes. Os símbolos e os caracteres especiais do alfabeto também podem ser rescritos e/ou adicionados aos termos da busca pelo normalizador 302 para facilitar uma busca acurada. Por exemplo, os caracteres, tais como o *eset* alemão “ß”, pode ser rescrito como “ss”, de forma que uma busca contendo o *eset* alemão, irá produzir acertos, mesmo se a forma do *eset* ou a forma com “ss” estiverem presentes no texto que está sendo pesquisado.

[0034] A normalização do símbolo envolve a remoção ou a conversão de símbolos, tal como pontuação, tanto para o(s) termo(s) da busca, como o texto que está sendo pesquisado quanto a uma correspondência precisa. Por exemplo, pontos finais dentro da abreviação, tais como “F.B.I.”, e asteriscos em “M*A*S*H” são removidos. Símbolos, tais como “&” e “@” podem ser convertidos para a sua forma escrita “e” e “em”. Símbolos substituídos por letras podem ser convertidos, por exemplo, o sinal “\$” em “Vega\$”. Esta normalização dos símbolos pode resultar em buscas adicionais. Por exemplo, o mecanismo de busca exemplificativo 204 poderia buscar por “Vegas” e “Vega\$”, depois de o normalizador 302 ter processado a cadeia de busca 202, ao invés de usar apenas uma forma pela convenção e normalizar o termo da busca e o texto que está sendo pesquisado, de acordo com uma forma convencional.

[0035] Os artigos do discurso, bem como palavras bastante onipresentes, mas não descritivas, tais como “a”, “o”, “é”, “são”, “e”, “la”, “los”, “les”, etc., podem ser removidos dos termos da busca pelo normalizador 302 para acelerar o processamento. Em algumas implementações, os artigos e palavras onipresentes não são retiradas ou são buscadas condicionalmente se elas forem as únicas palavras presentes na busca, ou se um primeiro passo da busca não produzir resulta-

dos.

[0036] O mecanismo de busca exemplificativo 204 pode varrer a cadeia de busca antecipadamente por símbolos que alteram o significado da busca de alguma forma. Por exemplo, colocando-se na frente de uma palavra um sinal de menos “-”, poderia indicar que a palavra não pode aparecer em uma correspondência. O agrupamento dos termos com aspas poderia indicar que o grupo somente deveria ser considerado quando os termos no grupo aparecem em uma determinada ordem, e juntos. As palavras “E”, “OU” e “NÃO” podem ser usadas de uma maneira especial para influenciar os operadores de busca. Isso permite que um usuário controle mais a busca.

[0037] Em uma implementação, o normalizador 302 ou outros componentes do mecanismo de busca 204, pode aplicar a verificação da ortografia à cadeia de busca 202 conforme ingressados. Inúmeras estratégias são empregadas para auxiliar o usuário a evitar erros quando se ingressa a cadeia de busca 202. O usuário pode ser alertado quanto à possível presença de palavras erroneamente grafadas no texto de busca antes de a busca ser executada, seja através de uma caixa de diálogo de ortografia ou outro indicador de texto, tal como o destaque em “linhas onduladas” INTELLISENSE®. Alternativamente, a verificação da ortografia pode ser adiada para o final da execução da busca e realizada apenas se não forem encontrados resultados exatos. Por último, a verificação da ortografia pode ser realizada de forma transparente. As formas corretas das palavras interpretadas pelo normalizador 302 a serem erroneamente grafadas podem ser automaticamente anexadas na cadeia de busca 202 e cotadas como buscas condicionais. A classificação de relevância (a ser tratada abaixo), pode então ser usada para reduzir a classificação desses termos de busca condicionais se as buscas produzirem acertos que realmente contêm os termos da busca erroneamente grafados.

[0038] O normalizador 302 também pode usar a correção automática, que é uma forma simplificada de verificar a ortografia. Ao invés de comparar cada termo com um dicionário, as palavras são corrigidas conforme são datilografadas, usando uma tabela de consulta simples de palavras erroneamente grafadas de forma comum. Em casos raros, nos quais a correção automática toma a decisão errada (corrigindo uma palavra o usuário intencionalmente significa “grafar erroneamente”), o usuário pode usar a tecla de retrocesso (*backspace*) uma vez para desfazer a correção automática. Na medida em que o usuário datilografa, uma lista instantânea (*pop-up*) de possíveis conclusões de palavras também pode ser apresentada pelo normalizador 302, com base em um índice de texto total pré-existente. Isso possibilita que o usuário ingresse com os critérios de busca mais rapidamente.

[0039] Um intérprete de cadeia 304 é acoplada comunicativamente com o normalizador 302. Em uma implementação, o intérprete de cadeia 304 inclui um transformador de cadeia 306 e um identificador de atributo 308 acoplado comunicativamente com uma lista de atributos 310. O intérprete de cadeia 304 também pode ser acoplado comunicativamente com uma tabela de restrições de busca 312.

[0040] O transformador de cadeia 306 pode testar os termos da busca em uma cadeia de busca 202 seja isoladamente ou em combinação para conseguir uma interpretação da cadeia de busca 202 e para determinar se a cadeia deveria ser rearrumada. O transformador de cadeia 306 pode usar um rearrumador e /ou esquema de ajuntamento para tentar combinações dos vários termos dentro da cadeia de busca 202, e em várias ordens para identificar possíveis atributos para os quais o usuário esteve tentando buscar. Sendo assim, “*dolby the closed mile green captioning*” poderia produzir candidatos de termo de busca incluindo “*dolby*”, “*the Green Mile*”, e “*closed captioning*”. Se parecer que dois ou mais dos termos deveriam ficar mutuamente agru-

pados, o transformador da cadeia pode transpor os termos e/ou adicionar transparentemente o grupo pressuposto para os termos da busca na cadeia de busca 202. Para se alcançar isso, o transformador de cadeia 306 pode testar os termos da busca confrontando com a lista de atributos 310 e/ou a tabela de restrições 312.

[0041] O transformador de cadeia 306 pode derivar um termo de busca para ampliar o escopo de uma busca. “Derivação” é um processo de remoção de prefixos e/ou sufixos a partir de um termo de busca para permitir que a raiz do termo da busca procure o termo da busca original. Isso produz mais acertos que capturam mais variantes do termo de busca original. Por exemplo, a palavra “divino” pode ser derivada para “divin”, de forma que se corresponda com “divindade”, “divinação”, “divinador”, “divindade”, etc. O transformados da cadeia 306 pode usar um algoritmo de derivação conhecido, tal como o algoritmo “Porter”. A derivação apresenta a vantagem de gerar acertos que poderiam ter sido perdidos se os termos da busca fossem indagados corretamente, mas apresenta a desvantagem de gerar muitas vezes acertos irrelevantes. Este efeito pode ser reduzido pelo uso de um classificador de relevância, tanto no transformador de cadeia 306, como em outra parte do mecanismo de busca exemplificativo 204, tal como o compilador de resultados (descrito abaixo). Em algumas implementações, a derivação requer uma etapa de pré-processamento de dados que pode ocasionar uma redução na velocidade da busca. Por conseguinte, em algumas implementações, a derivação se limita a palavras especiais, por exemplo, palavras raras, conforme determinado por uma frequência do índice de ocorrência.

[0042] O transformador de cadeia 306 pode realizar transformações adicionais de termo e cadeia. A ramificação de palavras envolve a geração de um conjunto de termos de busca a partir de um único termo. Uma lista de palavras relacionadas pode ser usada. Por exem-

plo, o termo de busca “NFL” poderia acarretar na inclusão do termo “futebol” na cadeia de busca 202, e assim “NFL” poderia gerar uma busca adicional para a palavra “futebol”. O termo “trinta” poderia ser acrescentado à cadeia de busca 202 se o termo “30” aparecer na cadeia de busca 202, etc. Os termos relacionados podem vir a partir de uma lista de exceção relevante aos dados que estão sendo pesquisados, e/ou uma lista de exceção armazenada na tabela de restrições definida por usuário ou determinada programaticamente 312. A ramificação de palavras também pode diminuir as buscas e pode gerar resultados irrelevantes, mas também irá gerar acertos em alguns casos onde nenhum tinha sido encontrado. Um classificador de relevância pode ser usado para colocar resultados menos relevantes no final da lista de acerto 222.

[0043] Também pode ser usada a pluralização. A pluralização é uma forma simples de derivação de palavras, que gera termos de busca adicionais com base na forma singular ou plural da palavra original. Por exemplo, se o termo de busca for “neném”, um termo de busca adicional “nenéns” pode ser incluído.

[0044] Opcionalmente, o intérprete de cadeia 304 ou outro componente do mecanismo de busca 204, pode adicionar análise de idioma natural aos demais recursos e processos de interpretação de cadeia. Este tipo de análise inclui a conversão de uma frase de busca, tal como “Quando o filme Guerra nas Estrelas foi exibido em Dolby-Digital?”, dentro dos termos exequíveis de busca que atende à solicitação de busca do usuário. Muito embora a análise de idioma natural seja muito poderosa, a gramática normalmente se limita à escolha do usuário dos significados de nome/verbo. Adicionalmente, dado o número aumentado de palavras na cadeia de busca pelo idioma natural 202 (com relação a uma busca de termo Booleano), a busca pelo idioma natural pode não ser uma ótima escolha num sistema de dez rodapés, no qual

os termos de busca têm que ser ingressados usando um teclado do controle remoto. Por último, a análise do idioma natural pode requerer para sua implementação mais potência no processamento.

[0045] Os recursos e processos realizados pelo transformador de cadeia 306 e pelos demais módulos do mecanismo de busca exemplificativo 204 são descritos em determinada ordem, mas a ordem na qual esses recursos e processos são apresentados não significa indicar uma ordem sugerida de uso, através de um processamento em canalização. Em muitos casos, a ordem e/ou o desempenho de cada um desses recursos e processos é importante aos resultados gerados.

[0046] O identificador de atributo 308 separa os termos de busca de texto dos termos de busca de atributo por vários métodos, tais como a comparação do candidato ou os termos potenciais de busca com os atributos na lista de atributos 310. O identificador de atributos 308 também pode receber um retorno a partir dos resultados de busca de texto e aplicar o processo de eliminação aos termos na cadeia de busca 202 para, assim encontrar atributos. Nesse caso, o identificador de atributo 308 pode postular vários relacionamentos Booleanos entre os termos de busca. Por exemplo os termos de busca “*green mile dolby*” poderiam ser interpretados como “green’ OU ‘mile’ E dolby”, ou “green mile’ E dolby”, mas poderiam gerar uma busca que retorna apenas exemplos da exibição “The Green Mile” in Dolby® digital ou som DOLBY® Estéreo, por causa do segundo conjunto de relações Booleanas postuladas.

[0047] A listagem de atributos 310, logicamente, pode ficar armazenada na memória do computador para uma interpretação rápida da cadeia. Se uma lista de aproximadamente duzentos atributos de programação de televisão for usada na lista de atributos 310, então talvez virtualmente todos os atributos relevantes para a programação de televisão que seriam ingressados por um usuário possam ser identificados

pelo identificador de atributo 308 e separado dos termos de busca.

[0048] Uma representação de uma listagem exemplificativa de atributos 310 considerada como uma estrutura de dados exemplificativa em Linguagem de Marcação Extensível (XML), aparece no Anexo A: “Listagem de Atributos Exemplificativa”. Esta representação contém apenas uma seleção de amostra de elementos, valores e atributos de programação de televisão. Desta forma, uma listagem típica de atributos 310 pode conter mais ou menos entradas que as mostradas, e/ou elementos, atributos e/ou valores diferentes.

[0049] Conforme demonstrado pela versão XML de uma listagem exemplificativa de atributos 310 no Anexo A, a listagem de atributos 310 não precisa ser limitada a uma lista sequencial de elementos de atributo que proporciona apenas uma correspondência verdadeira/falsa ou “existe/não existe”, quando apresentada com um candidato do termo de busca de atributo. A listagem de atributos 310 pode conter metadados 311 pelo menos para alguns atributos, em que os metadados de um atributo se refere aos recursos de busca e aos valores para usar na busca. Os metadados 311 também podem relacionar o atributo a um conjunto de condições de busca lógica na tabela de restrições 312. Ademais, a listagem de atributos 310 pode conter índices(s) 313 para se referir a um atributo na lista de atributos 310 para outros atributos e valores relacionados que podem ser procurados razoavelmente para produzir acertos que se conformam à cadeia de busca 202.

[0050] Em algumas implementações, o(s) índice(s) 313 e/ou metadados 311 podem conter lógica de ramificação relativamente complexa e/ou uma estrutura de árvore de condições de busca lógicas que podem gerar acertos que se conformam à intenção do usuário na construção da cadeia de busca 202. Por exemplo, se o identificador de atributo 308 verificar a listagem de atributos 310 para o termo de busca “estéreo”, uma correspondência pode ser encontrada designando o

nome apropriado do atributo e/ou designando os campos de busca na informação EPG com probabilidade de conter os atributos estéreo de programas, por exemplo, programas “estéreo” podem ser encontrados quando se olha nos campos de “ScheduleEntryAttribute” dos registros no banco de dados. Ademais, quando se descobre que o termo de busca candidato se encaixa nos critérios de um atributo e quando se designa os campos de busca nos dados EPG, a listagem de atributos 310 também pode produzir um ou mais valores exatos, por exemplo, “MSEPGA_STEREO”, que pode representar um termo de busca, chave, ou categoria com a qual (ou na qual) se encontra programas estéreo.

[0051] A capacidade de indexação da lista de atributos 310 pode ser multi ramificada. Por exemplo, se o identificador de atributos 308 verificar a lista de atributos 310 para o termo de busca candidato “esportes”, a lista de atributos 310 pode identificar “esportes” como um atributo, pode identificá-lo também como um tipo de atributo “categoria”, e direcionar o identificador de atributo 308 para contra verificar três outras categorias quanto a programas relevantes: “novidades”, “especiais”, e “MSEPGC_SPORTS”.

[0052] Quando se dispõe os recursos e processos descritos acima para o transformador de cadeia 306 e se identifica os atributos e metadados relacionados através de uma lista de atributos 310, o intérprete de cadeia 304 consegue um conjunto e termos de busca que tenta aproximar a intensão do usuário na construção da cadeia de dados 202. Em outras palavras, o intérprete de cadeia 304 consegue um conjunto de termos de busca de texto e termos de busca de atributo que possuem uma alta probabilidade de retornar os acertos que se encaixam no que a maior parte dos usuários estaria procurando se eles tivessem construído a cadeia de busca 202. Por exemplo, se o usuário construir a cadeia “*scifi dolby letterbox*”, o mecanismo de busca exem-

plificativo 204 poderia retornar uma lista de acerto 222 de exibições de ficção científica que está sendo mostrada em som DOLBY® Estéreo em formato original (*letterbox*). Se o usuário construir uma cadeia “primetime network comedies”, o mecanismo de busca exemplificativo 204 poderia retornar uma lista de acerto 222 de comédia em horário nobre mostra em estações de trabalho. Se o usuário construir a cadeia “drama – classificação para menores”, o mecanismo de busca exemplificativo 204 poderia retornar uma lista de acerto 222 de exibições de drama com uma classificação para menores (PG). Se o usuário construir a cadeia de busca “Discovery”, o mecanismo de busca exemplificativo 204 poderia retornar para uma lista de acerto 222 ambas as exibições com “discovery” no título e de canais com “discovery” no nome do canal, por exemplo, o “DISCOVERY CHANNEL®”.

[0053] Um gerador de pergunta 314 é acoplado de forma comunicativa com o intérprete de cadeia 304. O gerador de pergunta 314 pode incluir um formulador de pergunta de texto 316 e um formulador de pergunta de atributo 318. Em uma implementação, o formulador de pergunta de texto 316 e o formulador de pergunta de atributo 318 compõe o mínimo (muitas vezes apenas um) e o mais restrito possível, uma pergunta ou perguntas para busca no(s) banco de dado(s) EPG 300 e recursos EPG com base na Rede 334. Se múltiplas perguntas forem compostas, então os resultados podem ser combinados.

[0054] Para cada termo de busca que for recebido a partir do intérprete de cadeia 304, o gerador de pergunta 314 pode aplicar um operador que afeta o modo no qual a correspondência é descoberta durante uma pergunta do(s) banco(s) de dados EPG 300 ou recursos EPG com base em Rede 334. Em uma implementação, cada operador especifica uma posição ou configuração na qual o termo de busca tem (ou não tem) que ser descoberto no texto que está sendo pesquisado. Uns poucos operadores exemplificativos serão descritos agora. Um

operador “exato” especifica que o termo de busca deveria se corresponder exatamente a uma determinada parte do texto que está sendo pesquisado, isto é, se o gerador de pergunta 314 está buscando um certo tipo de campo, o termo de busca teria que se corresponder exatamente com o campo. Um operador de “começar- com (frase)” especifica que o termo da busca deveria ser encontrado no início do campo do texto que está sendo pesquisado. Um operador de “começar-com (palavra)” especifica que o termo da busca deveria ser o início de uma palavra dentro do campo do texto que está sendo pesquisado. Um operador de “contém” especifica que o termo da busca pode ser encontrado em qualquer lugar no campo do texto que está sendo pesquisado.

[0055] Para cada um dos operadores acima descritos, um “contra operador” (a forma NÃO), pode ser usado para especificar que um acerto não deveria ser gerado pelo mecanismo de busca 204 se fosse verdade a respectiva condição. Os operadores de busca “E” e “OU” também podem ser usados para especificar se todos os termos têm que ser encontrados no campo do texto que está sendo pesquisado ou se apenas alguns dos termos têm que se corresponder.

[0056] O gerenciador de dados 319 usa os vários termos de busca produzidos pelo gerador de pergunta 314 para realizar buscas no(s) banco(s) de dados EPG 300. Em algumas implementações da presente invenção, o gerenciador de dados 319 pode ser ou pode empregar um mecanismo de banco de dados SQL (linguagem de pergunta estruturada), ou um mecanismo de banco de dados JET e se assim for, pode combinar algumas das funções do compilador de resultados 320. Por conseguinte, o gerenciador de dados 319 pode executar internamente trajetórias de busca múltiplas e combiná-las. Desta forma, o(s) banco(s) de dados EPG 300 podem assumir inúmeros formatos diferentes (por exemplo, XML, binário, JET, SQL, etc.), alguns mais ade-

quados para armazenamento e alguns mais adequados para redes cruzadas de transmissão. Dependendo das restrições do meio de transmissão, o formato ótimo para transmissão pode ser diferente do formato ótimo para armazenamento e/ou busca. Inúmeras destas escolhas de formato e suas resistências e fraquezas relativas serão discutidas agora.

[0057] Alguns serviços guia transmitem um guia de múltiplos dias a clientes que usam XML que é formatado para se corresponder com um esquema definido. O XML é um formato conveniente para se trabalhar com ele por ser de fácil compreensão, legível por humanos e possui um grande número de ferramentas existentes. O XML não é, entretanto, um formato muito compacto. Por esta razão, o guia eHome é comprimido em um arquivo CAB antes de ser transmitido e então descomprimido depois de a recepção ser concluída.

[0058] Arquivos XML grandes não são pesquisados facilmente. Por exemplo, uma quantidade de dez dias de dados EPG pode preencher um arquivo de 6,6 megabytes. Carregar um arquivo XML deste tamanho dentro de um modelo de objeto de documento (DOM) requer quase noventa megabytes de memória de acesso aleatório. A varredura de um arquivo grande destes durante cada busca é impraticável. Por essas razões, o formato XML pode ser usado principalmente para transmissão e armazenamento persistente. Quando muitos clientes carregam o arquivo de dados EPG, o dado não textual é armazenado na memória e todo dado baseado em texto é escrito em um arquivo mapeado por memória para busca. Alguns dados EPG APIs podem ter que implementar um banco de dados personalizado para acomodar mudanças em padrões de dados e/ou busca.

[0059] Outra escolha para o(s) banco(s) de dados 300 é um formato binário. Embora os formatos binários sejam com frequência muito compactos, eles geralmente dependem de ferramentas personalizadas

para gerá-los e navegá-los. Como os arquivos XML, os arquivos binários não são normalmente ótimos para busca e podem requerer um formato de armazenamento intermediário alternativo para esse propósito.

[0060] O mecanismo de busca 204 também pode usar o mecanismo de banco de dados Jet, que é bem suportado com a ferramenta de banco de dados, para perguntar ao(s) banco(s) de dados EPG 300. Um API redistribuível chamado MDAC proporciona APIs ricos para criar e perguntar banco de dados Jet usando SQL, ADO ou OLE DB. Uma vantagem de usar um banco de dados Jet para o(s) banco(s) de dados EPG 300 é a busca fácil usando comandos SQL padrão da indústria e fácil visualização usando software comum. O desempenho das buscas usando-se o Jet é em geral mais rápido que muitos outros métodos (uma exceção é o Servidor SQL), já que os índices são gerados e armazenados de forma apropriada. Os bancos de dados Jet também ficam armazenados em um único arquivo facilitando a manipulação de todo o banco de dados EPG 300.

[0061] Como o mecanismo de banco de dados Jet, o SQL consiste de uma plataforma de banco de dados bem suportada para o(s) banco(s) de dados EPG com uma biblioteca extensa de ferramentas e APIs. Muito embora uma versão atual bem conhecida do Servidor SQL rode nas máquinas de servidor, uma versão de Servidor SQL chamada MSDE (ou a Edição de Mesa SQL), pode ser instalada em muitas versões de software do sistema operacional atual. A busca por um banco de dados SQL bem indexado, provavelmente é mais rápida e mais flexível que outras técnicas de busca de dados conhecida. Os bancos de dados, entretanto, não são facilmente transmitidos através de uma rede. Uma etapa de pós processamento de carga de dados EPG dentro do banco de dados SQL a partir de outro formato iria requerer usar MSDE. Ademais, diferentemente da versão do servidor, o MSDE pode

nem sempre suportar uma busca de idioma natural ou uma indexação de texto completo (embora ambas pudessem ser emuladas usando outros métodos).

[0062] Deve-se verificar que em muitos casos, o formato nativo do(s) banco(s) de dados EPG 300 pode não estar na forma final requerida pelo mecanismo de busca exemplificativo 204. Por exemplo, os índices de dados poderiam precisar serem gerados ou os dados poderiam precisar serem lidos dentro de um banco de dados alternativo, que é otimizado para busca, mas difícil de transmitir através de uma rede. Em uma implementação, o mecanismo de busca 204 descarta dados indesejados e mescla os dados específicos de usuário para uma busca mais rápida.

[0063] Se forem usados os índices de dados, o mecanismo de busca 204 pode implementar a indexação de texto completo. Isso envolve a geração de um índice de palavra inverso em cada elemento de dados EPG baseados em texto. A indexação de texto completa aumenta em muito o desempenho de buscas às expensas do(s) banco(s) de dados EPG maior(es) e um tempo aumentado para a geração de índice.

[0064] O mecanismo de busca 204 também pode usar a indexação de assinatura. Como a indexação de texto completo, a indexação de assinatura consiste de um método de identificação rápida de possíveis correspondências de busca. No(s) banco(s) de dados EPG 300, cada fileira de dados (programa, canal, etc.), possui uma assinatura binária similar para um algoritmo de pesquisa (*hash*), que indica quais as letras que ocorrem em cada palavra. O desempenho de computações similares no termo de busca e uma palavra proveniente do texto que está sendo pesquisado, então correspondendo aos resultados usando aritmética binária, permite uma aproximação rápida de se as correspondências de palavra se igualam ao termo da busca. Uma se-

gunda busca, mais intensiva por processador, pode então ser necessária para determinar se a fileira que contém a palavra correspondente deveria gerar um acerto.

[0065] Conforme mencionado acima, de passagem, o objeto da presente invenção não se limita à capacidade que o mecanismo de busca 204 tem de derivar os termos de busca de texto e os termos de busca de atributo a partir de uma única cadeia de busca 202. Um mecanismo de busca exemplificativo 204 pode derivar os termos de busca a partir de múltiplas fontes, tais como vários campos de entrada em uma UI, além de derivar os termos de busca de texto e termos de busca de atributo a partir de uma única cadeia de busca 202. Por exemplo, em alguns produtos, atributos tais como “estéreo”, “legenda oculta”, “tempo de começo” e “ano de liberação”, muitas vezes são disponibilizados como termos de busca, mas são ingressados em algumas caixas de seleção da UI, botões de rádio, etc. Eles também podem ser acomodados pelo mecanismo de busca exemplificativo 204 ao ser passado para a API de busca como termos de busca. Esta acomodação não limita o mecanismo de busca 204 a partir da derivação de outros termos de atributo e termos de busca de texto a partir de uma única cadeia de busca 202.

[0066] O mecanismo de busca exemplificativo 204 pode induzir o usuário a escolher campos sobre o qual o usuário deseja efetuar a busca. Então, além de efetuar a busca automaticamente através de todos os domínios (por exemplo, título, descrição, canais, lançamento, etc.), e classificar os acertos por sua relevância prognosticada, o mecanismo de busca exemplificativo 204 também pode permitir que o usuário restrinja manualmente os campos e/ou domínios de busca.

[0067] O mecanismo de busca exemplificativo 204 também inclui um compilador de resultados 320 acoplado comunicativamente com o gerenciador de dados 319 e o intérprete de cadeia 304. Um receptor

de acerto 322 no compilador de resultados 320 coleta acertos a partir de várias modalidades de busca ou a partir de uma única modalidade de busca, pesquisando em múltiplos bancos de dados. Devido ao fato de múltiplas buscas poderem estar transpirando simultaneamente, o receptor de acerto pode compreender uma derivação múltipla ou uma área de memória que pode armazenar acertos temporariamente dependendo de outra compilação.

[0068] No mecanismo exemplificativo 204, um comparador/filtro de busca 324 é acoplado comunicativamente no receptor de acerto 322 e com o intérprete de cadeia 304. O comparador/filtro de busca 324 testa os acertos contra o conjunto de interpretação de cadeia pelo intérprete de cadeia 304. Conforme discutido acima, o intérprete de cadeia 304 consegue uma hipótese de trabalho, ou interpretação, da intenção do usuário na construção da cadeia de busca 202 através do transformador de cadeia 306 e o identificador de atributo 308. No compilador de resultados 320, se um acerto não concordar com a interpretação, o acerto pode ser filtrado, ou seja, eliminado da lista de acerto exibida 222 ou rebaixado por propósitos de classificação na lista de acerto exibida 222. Em uma implementação, o compilador/filtro de busca 324 varre o resultado ajustado um a um para determinar se cada resultado é realmente um acerto. Isso é mais provável se uma pergunta não pudesse ser fatorada até um nível fino o bastante para excluir todas as perdas ao mesmo tempo em que ainda inclui todos os acertos. Como um exemplo de interpretação, se a cadeia de busca original 202 for “pato dolby” e o intérprete de cadeia 304 favorecer a interpretação “Howard o Pato, por Thomas Dolby, com relação à interpretação “Pato Donald em Dolby Digital”, então o acerto de Pato Donald poderia ser eliminado completamente da lista de acertos 222 ou pelo menos dado uma classificação inferior.

[0069] Um classificador de resultados 326 pode organizar acertos

para exibir por relevância ou outros critérios, mesmo quando todas as buscas forem finalizadas ou, em algumas implementações, na medida em que os resultados forem recebidos pouco a pouco pelo receptor de acerto 322 e passados através do comparador/filtro de busca 324. Os acertos gerados pelo mecanismo de busca 204, muita vezes, podem ser classificados para que os resultados de acertos possam ficar classificados para exibição de forma tal a apresentar primeiramente os resultados mais prováveis, isto é, os acertos que são mais próximos à interpretação da intenção do usuário na construção de cada cadeia 202 (como hipóteses pelo intérprete de cadeia 304). As correspondências exatas de cada termo de busca, por exemplo, podem ter conseguido uma classificação de relevância de 100% e dado prioridade de topo. Aos acertos que se correspondem somente com o texto a ser pesquisado depois de alguma transformação, tal como derivação ou correção ortográfica, são dados normalmente uma relevância mais baixa.

[0070] Um filtro de inclusão 328 pode ficar acoplado comunicativamente com o classificador de resultados 326. Em algumas variações, o filtro de inclusão 328 pode ser ajustado programaticamente para filtrar em uma certa maneira, enquanto em outras variações o filtro de inclusão 328 pode ser ajustado pelo usuário. Por exemplo, o filtro de inclusão 328 pode ser ajustado ou adaptado para excluir inteiramente quaisquer resultados de acertos que sejam menores que 100% relevantes, quando pelo menos um acerto de 100% relevância é encontrado. Em outros casos, o filtro de inclusão 328 pode ser ajustado ou adaptado para truncar uma longa lista de resultados, exibindo apenas um determinado número de resultados de topo, a despeito de alguns dos resultados tendo 100% de relevância. Em ainda outros casos, o filtro de inclusão 328 pode permitir apenas resultados de acerto que tenham uma classificação atribuída acima de um certo percentual.

Por conseguinte, se a busca for o filme “Onde Cresce a Samambaia Vermelha” e o único acerto for “A Hora do Esqueleto Vermelho”, com uma classificação de relevância de apenas 15%, então o filtro de inclusão 328 poderia exibir “nenhum resultado encontrado”.

[0071] O filtro de inclusão 328 também pode restringir o número de resultados até um valor definido e/ou restringir os resultados dependendo do controle de parentesco e/ou o usuário que conduz a busca. Em um contexto de registro, o filtro de inclusão 328 pode eliminar os resultados que não serão exibidos no futuro.

[0072] Um atualizador de exibição 330 é incluído no compilador de resultados 320 para controlar a exibição 332 na qual a lista de acerto 222 é apresentada ao usuário. Existem muitas maneiras possíveis de apresentar a lista de acerto 222 dos resultados de busca. Os acertos podem ser exibidos de forma alfabética ou cronológica. Os resultados também podem ser agrupados e classificados por relevância. Em algumas implementações, o atualizador de exibição 330 pode ser uma carteira de compensação para resultados, ou seja, o atualizador de exibição 330 pode manter um registro do que está sendo exibido. Se os resultados de acerto forem recebidos pouco a pouco, então o atualizador de exibição 330 pode realizar uma função de integração de combinações de resultados antigos com os resultados novos e decidir quais os resultados a serem exibidos. Durante uma busca feita em etapas, o atualizador de exibição 330 pode setorizar a visualização para mostrar a ordem na qual os grupos de resultados de busca são recebidos e/ou pode também setorizar a exibição para mostrar quais os acertos são relacionados com os termos da busca. Durante uma busca progressiva (a ser tratado mais completamente abaixo), o atualizador de exibição 330 pode descartar todos ou a maior parte dos resultados exibidos a cada vez que o usuário adicionar (por exemplo, tipos e /ou entradas), um novo caracter alfanumérico na cadeia de

busca 202. Sendo assim, em uma busca progressiva, o atualizador de exibição 330 reconfigura os resultados da busca a partir do zero, à medida que desenvolve o significado da cadeia de busca 202 que está sendo ingressada.

[0073] Logicamente, em algumas implementações, os vários módulos do compilador de resultados 320 podem ser combinados mutuamente ou ordenado de forma diferente. Em variações, alguns dos módulos poderiam ser eliminados ou outros módulos de banco de dados convencional poderiam ser adicionados.

[0074] Variações do mecanismo de busca exemplificativo 204 podem ser adaptadas para se adequarem à velocidade e à potência de processamento do hardware e/ou ambiente de software disponíveis usados na implementação. Uma busca pode demorar mais a executar do que o aceitável pelo usuário. O mecanismo de busca 204 pode retornar assincronamente os acertos como eles são encontrados ao mesmo tempo em que a(s) busca(s) ainda estão executando. Alternativamente, um mecanismo de busca exemplificativo 204 pode dividir de forma inteligente a busca em pedaços, atualizando o resultado ajustado periodicamente até que a busca seja concluída. Por exemplo, uma busca de texto por uma correspondência de palavra exata pode ser executada (e os resultados retornados), antes de as funções de derivação e normalização serem aplicadas e retornadas em um passe subsequente. As buscas também podem ser anuladas no tempo, buscando programas em grupos divididos de acordo com os tempos do ar esquematizados.

[0075] Em outra opção, se estiverem disponíveis armazenamentos de dados adicionais para o mecanismo de busca exemplificativo 204, o domínio de busca pode ser ampliado. O mecanismo de busca 204 poderia executar buscas através de cada armazenamento de dados, mesclando os resultados antes de finalizar o conjunto de resultados.

Isso pode ser usado, por exemplo, para combinar a busca de banco(s) de dados locais EPG 300 com a busca dos recursos EPG baseados na Rede 334. Adicionalmente, os programas pré-gravados em um meio local também podem ser adicionados no domínio de busca. Estas buscas podem ser executadas através de múltiplos APIs e mescladas antes da exibição.

[0076] O mecanismo de busca exemplificativo 204 também pode persistir os critérios de busca para execução posterior ou de antecedentes. Os usuários podem estar interessados em criar uma busca que pode ser chamada novamente sem reentrar na cadeia de busca 202. Isso também é útil para manter o rastreamento de quais os acertos um usuário já revisou de modo que, em algumas implementações, alguns resultados podem ficar escondidos, separados, ou “escalado em Gray” para uma leitura cuidadosa mais fácil. As buscas persistentes poderiam ser rodadas todas as vezes em que a informação do(s) banco(s) de dados EPG 300 fosse atualizada, para que o usuário pudesse ser informado de eventos futuros que ele estivesse interessado.

[0077] Os critérios de busca podem ser serializados, por exemplo, em macros e atua como um mecanismo de serialização para definir registros automáticos, isto é, história-listas de buscas. Indicações também podem ser atribuídas a cada serialização de termos de busca, para que o usuário possa selecionar a serialização elegendo a indicação. Uma busca serializada pode ser trocada com outros usuários via email, links da Rede e/ou embutida em uma difusão de TV para ligações “clique para gravar”. O mecanismo de busca exemplificativo 204 também pode definir um ou mais formatos de serialização. A busca e/ou serialização pode ser armazenada em uma forma binária para armazenamento compacto ou lançamento em uma forma ASCII para fácil transmissão via email. A serialização pode ser usada sozinha ou em conjunto com o registro automático, no qual o usuário define uma

busca e indica que todas as apresentações futuras (menos exemplos duplicados) que se correspondem com os critérios de busca devem ser gravados.

BUSCA EXEMPLIFICATIVA

[0078] A Figura 4 mostra uma busca exemplificativa 400, de acordo com o objeto do presente pedido, que pode ser concluída por meio de um mecanismo de busca, tal como o mecanismo de busca exemplificativo 204. Uma cadeia de busca relativamente indefinida 202, tal como “doLBy WEST-rated ‘g’ (*Oeste de classificação “g” em doLby*) é recebida a partir de um usuário ou de armazenamento persistente através do mecanismo de busca exemplificativo 204. Essa cadeia de busca 202 pode não ser restringível a uma única pergunta concisa. O mecanismo de busca exemplificativo 204, entretanto, possui grande versatilidade para executar múltiplas buscas para encontrar prováveis acertos. O normalizador exemplificativo 302 transforma a cadeia em uma forma padronizada exemplificativa, isto é, uma cadeia de busca normalizada 402: *OESTE DE CLASSIFICAÇÃO “G” EM DOLBY*. Nesse caso, o normalizador 302 trocou o formato e algumas letras em “doLby” para conseguir um termo de busca de texto que irá se corresponder com o tipo de letra do texto a ser pesquisado, por exemplo nos) banco(s) de dados EPG 300. O normalizador 302 separa termos de busca usando os espaços em branco entre as palavras e traços entre “OESTE” e “classificado”, como limites. O normalizador 302 retirou as aspas em volta do “g” e colocou toda a cadeia de termos de busca de uma forma, isto é, letras de imprensa.

[0079] A cadeia de busca 402 é passada para o intérprete de cadeia exemplificativo 304, que pode chegar em uma interpretação quase que conclusiva da intenção do usuário na construção da cadeia de busca 202. Uma interpretação quase que conclusiva pode ser alcançada se os termos de busca de atributo que entraram pelo usuário fo-

rem claramente delineáveis a partir de termos de busca através, por exemplo, da lista de atributos 310. O intérprete de cadeia 304 pode não ser capaz de alcançar uma separação conclusiva entre o texto e os termos de atributo, entretanto, em cujo caso o intérprete de cadeia 304 pode passar os termos de busca para o gerador de pergunta 314 para buscas alternadas e/ou condicionais. Ou seja, se houver ambiguidade sobre um determinado termo, o termo pode ser usado em uma busca de texto e também em uma busca de atributo para ver, por exemplo, quantos acertos resultaram de cada tipo de busca. Consequentemente, o termo de busca “DOLBY” pode ser uma palavra em um título de programa, tal como “A Experiência em Dolby Digital”, pode ser um atributo do criador do programa, tal como “Thomas Dolby”; e pode ser um atributo qualitativo de programa, tal como “DOLBY® digital”. O gerador de pergunta exemplificativo 314 pode testar cada uma dessas possibilidades: como duas buscas de atributo 406 (DOLBY® digital e som DOLBY® Estéreo), como uma busca de texto alternativa 410 (“dolby” como um termo em um título de programa), e como uma busca de atributo alternativa (lançada e dirigida com o nome “Dolby”). O gerador de pergunta 314 pode decidir não buscar “DOLBY” como uma busca de texto primária 406, uma vez que a classificação de relevância de “DOLBY” como um atributo é muito alta. Entretanto, se “DOLBY” puder ser encontrado em uma busca de texto, então descobrir-se-á pela busca de texto alternativa 410. Por conseguinte, buscas alternativas ou condicionais podem ser usadas como cópia no caso de nenhum acerto ser retornado pelas buscas com prioridade mais elevada e/ou buscas alternativas 410, 412 puderem ser usadas como sondagens ou modificadores para ganhar mais informações sobre a cadeia de busca 202, isto é, para fortalecer uma interpretação da cadeia de busca 202 para visar novas avenidas de busca.

[0080] O termo de busca “OESTE” proveniente da cadeia de bus-

ca normalizada 402 também poderia ser usado em uma busca de atributo alternativa 412, como “OESTE” pode ser uma palavra oriunda de um título de programa, ou em alguns esquemas pode ser um atributo de pessoa associado com o programa. Se “OESTE” não for encontrado em uma varredura da lista de atributos 310, pelo intérprete de cadeia 304, entretanto, o intérprete de cadeia 304 pode formar uma conjectura forte de que “OESTE” seria um texto primário para uma busca de títulos de programa.

[0081] O intérprete de cadeia 304 detecta que o termo de busca “CLASSIFICAÇÃO” precede o termos de busca “G”, então os dois termos prováveis de serem interpretados como “CLASSIFICAÇÃO G” e pesquisado conformemente. Muito embora um único termo “CLASSIFICAÇÃO” possa ser perguntado em uma busca de texto alternativa 410 (isto é, um programa com uma palavra “classificada” no título), tal busca alternativa tem a probabilidade de receber uma classificação de relevância baixa quando “CLASSIFICAÇÃO” e “G” forem achados juntos na cadeia 202 na ordem apropriada de atributo bem conhecido.

[0082] O gerador de pergunta 314 e/ou mecanismo de busca 204 prossegue com várias buscas de texto 406, buscas de atributo 408, buscas de texto alternativos 410 e buscas de atributo alternativas 412. O número de buscas que podem ser efetuadas simultaneamente ou em ordem lógica, conforme prescrito pela extensão e significado da cadeia de busca 202, não é limitado. Em alguns casos, o mecanismo de busca exemplificativo 204 realiza todas as buscas e relaciona os resultados de acordo com a relevância depois de um “primeiro passe” através de todas as buscas. Em outros casos, o mecanismo de busca exemplificativo 204 segue uma árvore de condições lógicas ajustadas pelo intérprete de cadeia 304 ou o gerador de pergunta 314. Por exemplo, as condições lógicas poderiam ser “se as buscas de texto 406 produzirem A, então prosseguir com as buscas de atributo 408 e

uma busca de texto alternativa 410 usando o termo de busca X, mas se as buscas de texto 406 produzirem B, então prosseguir com três buscas de atributo alternativas 412". A lógica pode estipular adicionalmente que A é uma quantidade de acertos e B é um acerto particular contendo uma correspondência de texto particular.

[0083] Cada tipo de busca pode retornar uma listagem de acerto intermediária tendo os resultados daquela busca particular. As listagens de acerto de buscas de texto 414 no termo de busca "OESTE" pode conter títulos de programas, tais como "Como o Oeste foi Conquistado", "Estória do Lado Oeste", "O Oeste do Tempo dos Pioneiros", "A Asa do Oeste", etc. Com um único termo relativamente vago, tal como "OESTE", o gerador de pergunta 314, muito provavelmente, usa um operador "em cadeia" para encontrar quaisquer títulos de programa contendo a palavra "OESTE".

[0084] As listagens de acerto de buscas de atributo 415 pode conter ou tentar retornar um enorme contexto de títulos de programas tendo atributos de som "DOLBY® Estéreo", "DOLBY® digital", e "classificação para menores". Sabendo que esses atributos são um tanto onipresentes, o gerador de pergunta 314 pode preceder tais buscas de atributo 408 sobre atributos gerais e ao invés disso incluir os atributos como condições e/ou filtros durante as buscas de texto 406. Em outras palavras, um acerto de busca de texto (406) que contém o termo de busca "OESTE" pode ser desabilitado se não estiver associado também com pelo menos um atributo Dolby e uma classificação "G".

[0085] Uma vez que as buscas de texto 406 para "OESTE" resultam em um grande número de acertos diversos, o mecanismo de busca 204 tem a probabilidade de realizar as buscas de texto alternativas 410 para "DOLBY", "CLASSIFICAÇÃO", e "G" e, talvez, realizar uma contagem de acertos para cada um reunir informações adicionais. Talvez se houver um programa intitulado "Dolby Oeste", o mecanismo de

busca 204 trocaria sua interpretação inicial à luz da correspondência de título exata – termos de busca adjacentes que o mecanismo de busca 204 admitiu inicialmente foram uma mistura de texto e termos de busca de atributo.

[0086] As buscas de atributo alternativas 412 podem apresentar resultados adicionais, por exemplo, em esquemas que usam nomes e atributos sobre o lançamento e a direção. Por conseguinte, os termos de busca de atributo alternativos “DOLBY” e “OESTE” apresentavam quatro categorias: filmes de Thomas Dolby, filmes de Steve Dolby, filmes de Dominic West, e filmes de Shane West. O termo de busca de atributo alternativo “G” também apresentava uma imensa porção de títulos de programa associados com uma classificação “G”, mas pode ser filtrado por causa de seu tamanho ou porque sua busca já está coberta nas buscas de atributo principais 408.

[0087] Finalmente, todas as listas de acerto 414, 416, 418, 420 provenientes de várias buscas 406, 408, 410, 412 são processadas pelo compilador de resultados 320. O comparador/filtro de busca 324 pode comparar os acertos recebidos para interpretação da cadeia de busca 202 ajustada pelo intérprete de cadeia 304. O classificador de resultados 326 pode dar a cada acerto uma classificação de relevância. O filtro de inclusão 328 pode eliminar acertos cumulativos, de baixa classificação e/ou excedentes (muitos), bem como aqueles acertos que não se encaixam na interpretação da cadeia de busca 202. Na listagem de acerto filtrado ilustrada 422, se o intérprete de cadeia 304 determinou previamente um relacionamento Booleano entre os termos de busca, isto é, conseguiu uma interpretação da cadeia de busca 202 de que o título do programa inclui o termo “OESTE” E que o programa deveria ser em DOLBY® digital E classificado como “G”, apenas “Como o Oeste foi Conquistado” e relativamente poucos títulos de programa serão exibidos. Logicamente, se a interpretação conseguida

pelo mecanismo de busca exemplificativo 204 não está próxima da intenção do usuário (como prova a ausência de resultados úteis, conforme determinado pelo usuário), o usuário pode sempre modificar a busca, especialmente desde que em muitas implementações todo o processo pode acontecer quase que instantaneamente.

BUSCA PROGRESSIVA EXEMPLIFICATIVA

[0088] A Figura 5 mostra uma busca progressiva exemplificativa, de acordo com o objeto da presente invenção. Devido ao hardware e/ou software disponíveis para construir os módulos de um mecanismo de busca, este mecanismo de busca exemplificativo 204 ser poderoso o bastante para proporcionar grande velocidade na busca, é possível em uma implementação de um mecanismo de busca exemplificativo 204 realizar buscas progressivas.

[0089] Em uma busca progressiva, os resultados da busca são atualizados dinamicamente tão rápido quanto os resultados podem ser gerados à medida que o usuário datilografa ou digita os caracteres na UI. Uma nova batelada de buscas é iniciada depois de cada caracter ser ingressado (ou removido). Em uma busca não progressiva, ao usuário se requer que ele inicie manualmente a busca atuando uma tecla “Enter” e/ou indicando de alguma outra maneira que a cadeia da busca 202 está completa.

[0090] A Figura 5 apresenta cinco mostras da cadeia de busca 202 “A Escolha da Asa do Oeste em Dolby”, durante a sua entrada para busca em banco(s) de dados EPG 300 que têm umas poucas semanas de valia de informações EPG e uma coleta de programas gravados localmente na unidade de disco rígido local de um sistema de gravação de vídeo digital. No bloco 502, a palavra “The” foi ingressada e o mecanismo de busca exemplificativo 204 postou zero resultados 504. Isso pode ser porque o normalizador 302 reconheceu “A” como um artigo da fala e suprimiu as buscas que são baseadas somente no

artigo. Entretanto, quando o “A” foi ingressado, um sistema muito rápido poderia ter retornado os programas com um “A” isolado no título do programa, por exemplo, “Sr. A.”

[0091] No bloco 506, as palavras “O Oeste” foram ingressadas e o mecanismo de busca exemplificativo 204 retornou cinquenta resultados 508, inclusive estes títulos como “A Asa do Oeste”, “Como o Oeste foi Conquistado”, etc.

[0092] No bloco 510, as palavras “A Asa do Oeste” foi ingressada, restringindo o número de acertos de cinquenta para dez resultados 512. Nesse ponto, o título do programa provavelmente está certo e os dez resultados provavelmente consistem de episódios de TV e de cinema “A Asa do Oeste”, cada um tendo o seu próprio número de identificação de programa.

[0093] No bloco 514, as palavras “A Asa do Oeste em Dolby”, foram ingressadas e os acertos de qualificação são mais restringidos para cinco resultados 516. Aquelas episódios dos dez anteriores que não são em DOLBY® digital ou que têm algum outro atributo Dolby são eliminados.

[0094] No bloco 518, as palavras “A Escolha da Asa do Oeste em Dolby” foram ingressadas. Isso produz apenas dois resultados 520. Os dois acertos consistem daqueles episódios de “A Asa do Oeste” que possuem o atributo “Dolby” e que possuem um atributo de “Escolha”, talvez como uma palavra no campo de descrição para o episódio.

[0095] Uma vez que vários resultados de acerto são exibidos na medida em que a cadeia de busca 202 está sendo ingressada, talvez o episódio desejado “A Asa dos Oeste” tivesse aparecido na exibição 332 inicialmente no processo, por exemplo quando as palavras “O Oeste” no bloco 506 foram ingressadas, e o usuário foi capaz imediatamente de realçar e selecionar o episódio desejado.

MÉTODOS EXEMPLIFICATIVOS

[0096] A Figura 6 mostra um método exemplificativo 600 de desempenho de um texto livre e busca de atributo, de acordo com o objeto da presente invenção. Esse método 600 pode ser realizado por meio de um módulo, tal como o mecanismo de busca exemplificativo 204 mostrado nas Figuras 2, 3 e 5. No fluxograma, as operações são resumidas em blocos individualizados. As operações podem ser realizadas em hardware e/ou como instruções legíveis por máquina (software ou firmware) que podem ser executadas por meio de um processador.

[0097] No bloco 602, uma cadeia de busca é recebida. A cadeia pode ser recebida a partir de um usuário através de uma ou mais UIs, ou pode ser recebida a partir de um armazenamento como uma macro armazenada ou busca serializada. A cadeia de busca também pode ser recebida um caractere de uma vez, um novo conjunto de buscas realizadas progressivamente a cada vez que um novo caractere é recebido e adicionado na cadeia de busca crescente. Espera-se que a cadeia recebida seja uma cadeia de texto, com atributos soletrados na forma de texto. Entretanto, uma vez que a cadeia recebida seja normalizada, de forma que os elementos de não texto sejam convertidos em texto padronizado ou símbolos de busca padronizada, a cadeia recebida pode ser uma combinação de caracteres alfanuméricos, símbolos atribuídos ao conjunto ASCII, e/ou caracteres do alfabeto internacional. Em algumas variações, a cadeia de busca também pode compreender a entrada de caixa de seleção, entrada de tecla condicional, ícones clicados, etc.

[0098] No bloco 604, a cadeia é normalizada. A normalização consiste em converter a cadeia recebida em termos de busca que sejam inteligíveis para buscar fontes de informação EPG particulares, ou seja, que tenham um formato que se encaixe no formato dos dados nas informações EPG a serem pesquisadas. O objetivo consiste em elimi-

nar meras diferenças no texto, tipo de letra e formalidades de caracter evitando uma sinonímia entre os termos e as informações EPG que de alguma outra forma tivessem produzido um acerto, mas por diferenças de formalidades. Por conseguinte, a normalização é um kit completo de ferramentas de verificação ortográfica, auto correção, separação de palavras, padronização de letras em caixa alta e caixa baixa, normalização de símbolos, artigo de eliminação da fala, retirada de pontuação, etc., ambos os termos da busca e possivelmente das informações de banco de dados alvo. Por exemplo, “mash” e “M*A*S*H*” será sempre correspondido, independentemente de ambas as formas serem encontradas na busca, em uma peça alvo ou informações sendo buscadas, ou ambos.

[0099] No bloco 606, a cadeia de busca é separada em ternos de texto e termos de atributo. Essa separação efetua e acompanha uma interpretação da cadeia de busca. A separação em termos de texto e termos de atributo resulta em alguma interpretação, bem como alguma transformação e transposição dos termos passíveis de busca pela normalização. Uma vez que a cadeia de busca média é tipicamente apenas cinco ou seis termos de busca ou menos, um esquema pode ser seguido ao tentar todas as combinações possíveis dos termos de busca em vários agrupamentos de tamanho e em todas as ordens possíveis. Por conseguinte, a cadeia de busca pode ser misturada para testar vários agrupamentos e combinações dos termos de busca.

[00100] Na medida em que vários termos de busca candidatos e frases de busca (agrupamentos) são produzidos, eles podem ser testados ou interpretados contra uma listagem pré-existente de atributos para determinar quais os termos da busca e frases são atributos conhecidos. Os termos de busca candidatos também podem ser testados contra quaisquer resultados de busca que já podem ter sido realizados na cadeia de busca, uma vez que algumas vezes as várias buscas

realizadas em uma cadeia de busca não são feitas simultaneamente, mas realizadas sequencialmente ou iterativamente, de acordo com a lógica de busca condicional que possa ser ajustada na medida em que a busca restringe durante a interpretação da cadeia de busca mencionada acima.

[00101] Uma varredura rápida das palavras na frase de busca também pode revelar outros detalhes pretendidos pelo usuário, que constituem outro elemento de interpretação. Caso haja muitos artigos pequenos e conjunções, tais como “em”, “e”, “é” e “o”, então a cadeia de busca poderia ser interpretada como uma entrada de idioma natural. Entretanto, uma cadeia de pesquisa pequena com apenas uns termos um pouco concisos é provável que seja interpretada como uma busca de estilo Booleano.

[00102] No bloco 608, uma ou mais buscas são conduzidas para cada termo de texto e cada termo de atributo. Perguntas de pesquisa relativamente complexas podem ser ajustadas dependendo do texto e dos termos de atributo presentes e como eles se relacionam entre si. Como um exemplo improvável, mas ilustrativo, a cadeia de busca “faroeste após ao atores de ficção científica de 1980” poderia ser interpretada como “busca por faroestes depois de 1980, mas apenas aqueles com atores que estrelaram em filmes de ficção científica”. Esta busca vincularia uma categoria de “faroestes” e usando a lógica subtrativa para eliminar ambos os faroestes criados antes de 1980 e os faroestes que não tinham atores que também estrelaram e filmes de ficção científica. Muitas buscas de termo de texto e de termo de atributo podem ser realizadas imediatamente, bem como sequencialmente ou de forma iterativa, dependendo dos resultados recebidos. Por conseguinte, um mecanismo de busca que realiza o método exemplificativo 600, de tal forma que o mecanismo de busca exemplificativo 204 pode ser uma derivação múltipla, iniciando buscas que transpiram “em

todas as direções imediatamente”.

[00103] No bloco 610, os resultados da busca são combinados, classificados e/ou filtrados para exibição. A combinação é realizada normalmente de acordo com as relações Booleanas postuladas durante a separação de termos de atributo a partir dos termos de texto. Ou seja, como os resultados da busca são combinados é predeterminado em parte pela interpretação da cadeia de busca conseguida durante a separação dos termos de texto e de atributo e durante a formação da pergunta. Como os resultados recebem classificação de relevância também pode depender da interpretação conseguida antes no processo. A filtragem pode depender das preferências pré-estabelecidas do usuário para exibição, ou podem depender de um esquema no qual os resultados portam uma classificação de relevância abaixo do limite são eliminados da visualização.

[00104] No bloco 612, os resultados da busca são exibidos para o usuário. Os resultados podem ser exibidos depois de toda a busca ser feita, mas eles também podem ser exibidos na medida em que são recebidos e compilados. Em uma busca progressiva, os resultados são exibidos tão rapidamente quanto a cadeia de busca está sendo ingressada, sendo que cada caractere ingressado começa um novo jogo completo de buscas.

[00105] O método supra-descrito exemplificativo 600 é apenas um exemplo do objeto da presente invenção. Não significa que ele é limitante descreve um processo do exemplo que pode ter muitas variações disponibilizadas para o usuário.

Dispositivo de Computação Exemplificativo

[00106] A Figura 7 mostra um computador exemplificativo 700 adequado como um ambiente para praticar aspectos do objeto da presente invenção. Os componentes de computador exemplificativo 700 podem incluir, sem caráter limitativo, uma unidade de processamento

720, uma memória de sistema 730, e um barramento de sistema 721 que acopla vários componentes de sistema incluindo a memória de sistema 730 para a unidade de processamento 720. O barramento de sistema 721 pode ser de inúmeros tipos de estruturas de barramento incluindo um barramento de memória ou controlador de memória, um barramento periférico, e um barramento local usando quaisquer entre uma variedade de arquiteturas de barramento. À guisa de exemplo, e sem caráter limitativo, estas arquiteturas incluem o barramento da Arquitetura Padrão da Indústria (ISA), o barramento da Arquitetura de Micro Canal (MCA), o barramento ISA intensificado, o barramento local da Associação de Padrões Vídeoeletrônicos (VESA) e o barramento de Interconexão de Componente Periférico (PCI) também conhecido como barramento Mezanino.

[00107] O computador exemplificativo 700 inclui tipicamente uma variedade de meios legíveis em computador. Os meios legíveis em computador podem ser qualquer meio disponível que pode ser acessado pelo computador exemplificativo 700 e inclui meios voláteis e não-voláteis e meios não-removíveis. À guisa de exemplo, e sem caráter limitativo, o meio legível em computador pode compreender meios de armazenamento e computador e meios de comunicação. Os meios de armazenamento em computador incluem meios voláteis e não-voláteis, removível e não removível implementados em qualquer método ou tecnologia de armazenamento de informação, tal como as instruções legíveis em computador, estruturas de dados, módulos de programa, ou outros dados. Os meios de armazenamento em computador incluem, sem caráter limitativo, RAM, ROM, EEPROM, memória flash ou outra tecnologia de memória, CD-ROM, discos digitais versáteis (DVD) ou outro armazenamento de disco óptico, cassetes magnéticos, fita magnética, armazenamento de disco magnético ou outros dispositivos de armazenamento magnético, ou qualquer outro meio que possa

ser usado para armazenar as informações desejadas e que possam ser acessadas pelo computador exemplificativo 700. O meio de comunicação concretiza, tipicamente instruções legíveis em computador, estruturas de dados, módulos de programa ou outros dados em um sinal de dados modulado, tal como uma onda portadora ou outro mecanismo de transporte e inclui qualquer meio de entrega de informação. O termo “sinal de dado modulado” significa um sinal que possui uma ou mais de suas características ajustadas ou trocadas de maneira tal de modo a codificar informações no sinal. À guisa de exemplo, e sem caráter limitativo, os meios de comunicação inclui meio cabeado, tal como uma rede cabeada ou conexão cabeada direta e meio sem fio, tal como meio sem fio acústico, RF, infravermelho ou outro meio sem fio. As combinações de quaisquer dos acima também deveriam estar incluídas dentro do escopo dos meios legíveis em computador.

[00108] A memória do sistema 730 inclui o meio de armazenamento em computador na forma de memória volátil e/ou não volátil, tal memória de apenas leitura (ROM) 731 e memória de acesso aleatório (RAM) 732. Um sistema básico de entrada saída 733 (BIOS) contendo rotinas básicas que ajudam a transferir informações entre elementos dentro do computador exemplificativo 700, tal como durante a inicialização, é armazenado tipicamente no ROM 731. O RAM 732 contém tipicamente dados e módulos de programa que são acessíveis imediatamente a e/ou presentemente sendo operados pela unidade de processamento 720. À guisa de exemplo, e sem caráter limitativo, a Figura 7 ilustra o sistema operacional 734, programas de aplicativo 735, um mecanismo de busca exemplificativo 204, outros módulos de programa 736, e dados de programa 737. Muito embora o mecanismo de busca exemplificativo 204 seja representado como software na memória de acesso aleatório 732, outras implementações de um mecanismo de busca exemplificativo 204 podem ser hardware ou combinações de software

e d hardware.

[00109] O computador exemplificativo 700 também pode incluir outros meios de armazenamento em computador removíveis/não-removíveis, voláteis/não-voláteis. Apenas à guisa de exemplo, a Figura 7 ilustra uma unidade de disco rígido 741 que lê ou escreve em meios magnéticos não removível, não volátil, uma unidade de disco rígido 751 que lê ou escreve em um disco magnético removível, não-volátil 752, e uma unidade de disco óptico 755 que lê ou escreve em um disco óptico removível, não volátil 756, tal como um CD ROM ou outros meios ópticos. Outros meios de armazenamento em computador removível/não-removível, volátil-não-volátil, que podem ser usados no ambiente operacional exemplificativo incluem sem caráter limitativo, cassetes de fita magnética, cartões de memória flash, discos versáteis digitais, fitas de vídeo digital, RAM de estado sólido, ROM de estado sólido e similares. A unidade de disco rígido 741 é conectada tipicamente no barramento de sistema 721 através de uma interface de memória não removível, tal como interface 740, e unidade de disco magnético 751 e unidade de disco óptico 755, são conectados tipicamente no barramento de sistema 721 por meio de uma interface de memória, tal como uma interface 750.

[00110] As unidades e seus meios de armazenamento de computador associados, tratados acima e ilustrados na Figura 7, proporcionam o armazenamento de instruções legíveis em computador, estruturas de dados, módulos de programa e outros dados para computador exemplificativos 700. Na Figura 7, por exemplo, a unidade de disco rígido 741 é ilustrada como sistema operacional de armazenamento 744, programas aplicativos 745, outros módulos de programa 746 e dados de programa 747. Observa-se que esses componentes também podem ser os mesmos ou diferentes do sistema operacional 734, programas de aplicativos 735, outros módulos de programa 736 e dados

de programa 737. Ao sistema operacional 744, programas de aplicativos 745, outros módulos de programa 746, e dados de programa 747 são atribuídos números diferentes aqui para ilustrar que, no mínimo, eles são cópias diferentes. Um usuário pode ingressar comandos e informações no computador exemplificativo 700, através de dispositivos de entrada, tais como um teclado 762 e dispositivo indicador 761, referido comumente como um mouse, trackball ou tecla de toque. Outros dispositivos de entrada (não mostrados) podem incluir um microfone, joystick, game pad, antena parabólica, digitalizador ou similares. Estes e os demais dispositivos de entrada são muitas vezes conectados na unidade de processamento 720 através de uma interface de entrada com usuário 760 que é acoplado no barramento de sistema, mas pode ser conectado por outras estruturas de interface e de barramento, tais como uma porta paralela, porta de jogo ou barramento serial universal (USB). Um monitor 791, ou outro tipo de dispositivo para exibição também é conectado no barramento de sistema 721, através de uma interface, tal como uma interface de vídeo 790. Além do monitor, os computadores também podem incluir outros dispositivos de saída periféricos, tais como alto-falantes 797 e impressora 796, que podem ficar conectados através de uma interface periférica de saída 795.

[00111] O computador pode operar em um ambiente em rede usando conexões lógicas com um ou mais computadores remoto, tais como um computador remoto 780. O computador remoto 780 pode ser um computador pessoal, um servidor, um roteador, uma rede PC, um dispositivo não-hierárquico ou outro nó de rede comum, e inclui tipicamente muitos ou todos os elementos descritos acima, com relação ao computador exemplificativo 700, muito embora apenas um dispositivo de armazenamento de memória 781 tenha sido ilustrado na Figura 7. As conexões lógicas representadas na Figura 7 incluem uma rede de

área local (LAN) 771 e uma rede de área remota (WAN) 773, mas também incluem outras redes. Estes ambientes de rede são lugares comuns nos escritórios, redes de computador de âmbito empresarial, intranets e a Internet.

[00112] Quando usado em um ambiente de rede LAN, o computador exemplificativo 700 é conectado à LAN 771 através de uma interface de rede ou adaptador 770. Quando usado em um ambiente de rede WAN, o computador exemplificativo 700 inclui, tipicamente, um modem 772 ou outro dispositivo para estabelecer as comunicações sobre a WAN 773, tal como Internet. O modem 772, que pode ser interno ou externo, pode ser conectado ao barramento de sistema 721 através da interface de entrada de usuário 760, ou outro mecanismo apropriado. Em um ambiente de rede, os módulos de programa representados com relação ao computador exemplificativo 700, ou partes dos mesmos, podem ser armazenados no dispositivo de armazenamento de memória remoto. À guisa de exemplo, e sem caráter limitativo, a Figura 7 ilustra programas de aplicativos remotos 785 como residentes no dispositivo de memória 781. Verificar-se-á que as conexões de rede mostradas são exemplificativas e outros dispositivos de estabelecer uma ligação de comunicações entre os computadores podem ser usados.

[00113] Logicamente, o objeto da presente invenção também pode ser usada em muitos outros ambientes, tais como um aparelho codificador tendo hardware apropriado, um console de jogo e muitos outros tipos de dispositivos de processamento. Cada um desses ambientes pode conter alguns, mas não todos os componentes descritos acima para o computador exemplificativo 700.

CONCLUSÃO

[00114] Deve-se verificar que o objeto da presente invenção descrito acima pode ser implementado em hardware, em software ou em

ambos, no hardware e no software. Em certas implementações, o sistema exemplificativo e os métodos relacionados podem ser descritos no contexto geral de instruções executáveis por computador, tais como módulos de programa, sendo executados por um computador. De modo geral, os módulos de programa incluem rotinas, programas, objetos, componentes, estruturas de dados, etc., que realizam tarefas particulares ou implementam tipos de dados abstratos. O objeto da presente invenção também pode ser praticado em ambientes de comunicações distribuídas, onde as tarefas são executadas através de comunicações sem fio por meio de dispositivos de processamento remotos que ficam ligados através de uma rede de comunicações. Em uma rede sem fio, os módulos de programa podem ficar localizados em ambos os meios de armazenamento de dispositivos de comunicações local e o remoto incluindo dispositivos de armazenamento de memória.

[00115] As tratativas acima descrevem sistemas e métodos exemplificativos para texto livre e busca de atributos de dados EPG. Muito embora a invenção tenha sido descrita em linguagem específica para os recursos estruturais e/ou atos metodológicos, deve-se compreender que o objeto definido nas reivindicações em anexo não se limita necessariamente aos recursos ou atos específicos descritos. Pelo contrário, os recursos e atos específicos são descritos como formas exemplificativas de implementar a invenção reivindicada.

Anexo A: lista exemplificativa de atributos

```

<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<searchInfo>
  <categoryAndAttributeMappings separatorChar="|">
    <mapping keywords="closed captioned|closed captioning|closed
captions" field="ScheduleEntryAttribute" value="MSEPGA_CC" />
    <mapping keywords="stereo" field="ScheduleEntryAttribute"
value="MSEPGA_STEREO" />
    <mapping keywords="repeat|rerun|re run"
field="ScheduleEntryAttribute" value="MSEPGA_REPEAT" />
    <mapping keywords="live" field="ScheduleEntryAttribute"
value="MSEPGA_LIVE" />
    <mapping keywords="subtitled|subtitles"
field="ScheduleEntryAttribute" value="MSEPGA_SUBTITLED" />
    <mapping keywords="season premiere|season premier"
field="ScheduleEntryAttribute" value="MSEPGA_SEASONPREMIERE" />
    <mapping keywords="season finale" field="ScheduleEntryAttribute"
value="MSEPGA_SEASONFINALE" />
    <mapping keywords="series premiere|season premier"
field="ScheduleEntryAttribute" value="MSEPGA_SERIESPREMIERE" />
    <mapping keywords="series finale" field="ScheduleEntryAttribute"
value="MSEPGA_SERIESFINALE" />
    <mapping keywords="premiere" field="ScheduleEntryAttribute"
value="MSEPGA_PREMIERE" />
    <mapping keywords="finale" field="ScheduleEntryAttribute"
value="MSEPGA_FINAL" />
    <mapping keywords="letterbox" field="ScheduleEntryAttribute"
value="MSEPGA_LETTERBOX" />
    <mapping keywords="hdtv" field="ScheduleEntryAttribute"
value="MSEPGA_HDTV" />
    <mapping keywords="sap" field="ScheduleEntryAttribute"
value="MSEPGA_SAP" />
    <mapping keywords="dolby" field="ScheduleEntryAttribute"
value="MSEPGA_DOLBY" />
    <mapping keywords="dvs" field="ScheduleEntryAttribute"
value="MSEPGA_DVS" />
    <mapping keywords="rated pg 13|rated pg13"
field="ProgramRatingAttribute" value="MSEPGR_PG13" />
    <mapping keywords="rated pg" field="ProgramRatingAttribute"
value="MSEPGR_PG" />
    <mapping keywords="rated nc 17|rated nc17"
field="ProgramRatingAttribute" value="MSEPGR_NC17" />
    <mapping keywords="rated g" field="ProgramRatingAttribute"
value="MSEPGR_G" />
    <mapping keywords="rated r" field="ProgramRatingAttribute"
value="MSEPGR_R" />
    <mapping keywords="rated x" field="ProgramRatingAttribute"
value="MSEPGR_X" />
    <mapping keywords="rated ao" field="ProgramRatingAttribute"
value="MSEPGR_AO" />
    <mapping keywords="paid programming" field="ProgramAttribute"
value="MSEPGA_PAIDPROGRAMMING" />
    <mapping keywords="miniseries" field="ProgramAttribute"
value="MSEPGA_MINISERIES" />
  
```



```

    <mapping keywords="action adventure series" field="Category"
value="Series-701" />
    <mapping keywords="action adventure movie|action adventure movies"
field="Category" value="CMOVIE_ACTIONADV" />
    <mapping keywords="action adventure" field="Category"
value="CMOVIE_ACTIONADV|Series-701" />
    <mapping keywords="comedy|comedies" field="Category" value="Movies-
504|Series-703|Kids-303" />
    <mapping keywords="comedy series" field="Category" value="Series-
703" />
    <mapping keywords="comedy movie" field="Category" value="Movies-504"
/>
    <mapping keywords="documentary|documentaries" field="Category"
value="Educational-103" />
    <mapping keywords="drama" field="Category"
value="CMOVIE_DRAMA|Series-705" />
    <mapping keywords="educational series" field="Category"
value="Series-706" />
    <mapping keywords="educational" field="Category" value="Series-
706|Kids-304|Educational-1" />
    <mapping keywords="movies" field="Category" value="MSEPGC_MOVIES" />
    <mapping keywords="peliculas" field="Category" value="MSEPGC_MOVIES"
/>
    <mapping keywords="music" field="Category" value="Movies-508|Series-
709|Special-803" />
    <mapping keywords="news" field="Category" value="MSEPGC_NEWS" />
    <mapping keywords="science fiction movie|science fiction movies"
field="Category" value="Movies-511" />
    <mapping keywords="science fiction" field="Category" value="Movies-
511" />
    <mapping keywords="fantasy movie|fantasy movies" field="Category"
value="Movies-506" />
    <mapping keywords="fantasy" field="Category" value="Movies-506" />
    <mapping keywords="soap operas|soaps" field="Category"
value="Series-711" />
    <mapping keywords="sports" field="Category" value="News-605|Special-
805|MSEPGC_SPORTS" />
    <mapping keywords="adults only|xxx" field="Category" value="Movies-
502|Lifestyle-401" />
    <mapping keywords="kids movies" field="Category" value="Movies-503"
/>
    <mapping keywords="kids series" field="Category" value="Series-702"
/>
    <mapping keywords="kids" field="Category" value="Series-702|Movies-
503|Kids-3" />
    <mapping keywords="musicals" field="Category" value="Movies-508" />
    <mapping keywords="mystery movies|mysteries|mystery"
field="Category" value="Movies-509" />
    <mapping keywords="love stories|romance" field="Category"
value="Movies-510" />
    <mapping keywords="suspense" field="Category" value="Movies-512" />
    <mapping keywords="westerns|western" field="Category" value="Movies-
513" />
    <mapping keywords="animated|cartoon" field="Category" value="Kids-
302" />
    <mapping keywords="baseball|mlb" field="Category" value="CBASEBALL"
/>

```

```

    <mapping keywords="basketball|nba" field="Category"
value="CBASKETBALL" />
    <mapping keywords="boxing" field="Category" value="CBOXING" />
    <mapping keywords="football|nfl" field="Category" value="CFOOTBALL"
/>
    <mapping keywords="golf|pga|lpga|usga" field="Category"
value="CGOLF" />
    <mapping keywords="hockey|nhl" field="Category" value="CHOCKEY" />
    <mapping keywords="outdoor sports" field="Category" value="COUTDOOR"
/>
    <mapping keywords="racing" field="Category" value="CRACING" />
    <mapping keywords="soccer" field="Category" value="CSOCCER" />
    <mapping keywords="tennis" field="Category" value="CTENNIS" />
    <mapping keywords="sports news" field="Category" value="News-605" />
    <mapping keywords="sports" field="Category"
value="MSEPGC_SPORTS|News-605" />
    <mapping keywords="business news|financial news" field="Category"
value="News-601" />
    <mapping keywords="interviews|interview" field="Category"
value="News-602" />
    <mapping keywords="public affairs" field="Category" value="News-604"
/>
    <mapping keywords="weather" field="Category" value="News-606" />
    <mapping keywords="cooking" field="Category" value="Series-704" />
    <mapping keywords="game shows|game show" field="Category"
value="Series-707" />
    <mapping keywords="how to|howto" field="Category" value="Series-708"
/>
    <mapping keywords="talk shows|talk show" field="Category"
value="Series-712" />
    <mapping keywords="travel" field="Category" value="Series-713" />
    <mapping keywords="religious" field="Category" value="Special-
804|Lifestyle-409" />
    <mapping keywords="holidays|holiday" field="Category"
value="Special-802" />
    <mapping keywords="biography" field="Category" value="Educational-
102" />
    <mapping keywords="science" field="Category" value="Educational-105"
/>
    <mapping keywords="collectibles" field="Category" value="Lifestyle-
402" />
    <mapping keywords="exercise" field="Category" value="Lifestyle-404"
/>
    <mapping keywords="health" field="Category" value="Lifestyle-405" />
    <mapping keywords="home garden" field="Category" value="Lifestyle-
406" />
    <mapping keywords="international" field="Category" value="Lifestyle-
407" />
    <mapping keywords="outdoors" field="Category" value="Lifestyle-408" />
    <mapping keywords="primetime|prime time" field="TimeOfDay"
value="1140|1380" />
    <mapping keywords="morning" field="TimeOfDay" value="300|720" />
    <mapping keywords="afternoon" field="TimeOfDay" value="720|1080" />
    <mapping keywords="evening" field="TimeOfDay" value="1020|1200" />
    <mapping keywords="late night" field="TimeOfDay" value="1380|300" />
</categoryAndAttributeMappings>
</searchInfo>

```

REIVINDICAÇÕES

1. Método de busca de dados de guia de programação eletrônica (EPG) **caracterizado** pelo fato de que compreende:

receber (602) uma cadeia de busca de texto por meio de um receptor de cadeia de busca (301), em que a cadeia de busca de texto contém um ou mais termos de texto e um ou mais termos de atributo;

normalizar (604) a uma cadeia de busca de texto para fazer termos buscáveis por meio de:

aplicar a verificação ortográfica para a cadeia de busca de texto;

identificar uma palavra como estando escrita incorretamente; e

anexar uma forma corrigida da palavra identificada como estando escrita incorretamente aos termos buscáveis para realizar uma busca condicional com base na forma corrigida da palavra;

separar (606) os termos buscáveis em termos de texto e termos de atributo por meio de um intérprete de cadeia (214, 304), o intérprete de cadeia (214, 304) analisando a cadeia de busca de texto para separar os termos buscáveis de texto em termos de texto e termos de atributo comparando os termos buscáveis com uma lista de atributos para identificar os termos de atributos;

executar (608) uma busca de texto nos dados de EPG para cada termo de texto;

executar (608) uma busca de atributo nos dados de EPG para cada termo de atributo; e

classificar resultados das buscas de texto e de atributo para exibir os resultados de busca em conformidade com a cadeia de busca de texto, a classificação compreendendo baixar uma classificação de relevância de acertos resultantes da busca condicional na for-

ma corrigida da palavra quando os resultados da busca são obtidos com base na palavra identificada como escrita incorretamente.

2. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que a normalização (604) compreende a conversão das formalidades dos idiomas dos termos passíveis de busca para se corresponderem com as formalidades do idioma dos dados EPG.

3. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que a separação (606) inclui a comparação de termos buscáveis com resultados intermédios obtidos a partir de uma busca de texto alternativo utilizando um ou mais dos termos de atributo identificado comparando os termos buscáveis com a lista de atributos para a aquisição de informação adicional para separar os termos de texto dos termos de atributos além de comparar os termos buscáveis com a lista de atributos

4. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que a separação (606) inclui a comparação dos termos buscáveis com os resultados provenientes de uma busca de texto para separar os termos de texto dos termos de atributo.

5. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que a separação (606) compreende ainda o agrupamento dos termos buscáveis em várias combinações para se conseguir um conjunto de frases passíveis de busca.

6. Método, de acordo com a reivindicação 5, **caracterizado** pelo fato de que a separação (606) compreende ainda a transposição da ordem de pelo menos dois dos termos passíveis de busca antes do agrupamento.

7. Método, de acordo com a reivindicação 5, **caracterizado** pelo fato de que compreende ainda a comparação das frases passíveis de busca com atributos em uma lista de atributos para determinar se quaisquer das frases passíveis de busca são atributos.

8. Método, de acordo com a reivindicação 7, **caracterizado** pelo fato de que a lista de atributos contém um índice relacionado com os atributos indexados para recursos de destino e valores de busca para cada atributo indexado.

9. Método, de acordo com a reivindicação 7, **caracterizado** pelo fato de que a lista de atributos contém ramificação lógica associada com um atributo na lista de atributos para guiar uma busca usando o atributo.

10. Método, de acordo com a reivindicação 7, **caracterizado** pelo fato de que a lista de atributos contém metadados sobre um atributo na lista de atributos para guiar uma busca usando o atributo.

11. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que os termos de busca de separação compreendem a formação de uma interpretação da intenção do usuário na construção da cadeia de busca de texto.

12. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que a combinação, classificação e filtragem são realizadas de acordo com a interpretação.

13. Mecanismo de busca (204) para dados de guia de programação eletrônica (EPG) **caracterizado** pelo fato de que compreende:

um normalizador (302) para aplicar a verificação ortográfica a uma cadeia de busca para identificar uma palavra como sendo escrita incorretamente, o normalizador (302) anexando uma forma corrigida da palavra identificada como estando escrita incorretamente para buscar termos na cadeia de busca para realizar uma busca condicional com base na forma corrigida da palavra;

um intérprete de cadeia (214, 304) acoplado de forma comunicativa com uma lista de atributos, em que o intérprete de cadeia postula o significado dos termos na cadeia de busca;

o dito intérprete de cadeia (214, 304) compreende ainda um identificador de atributo para separar termos de atributos de termos de texto na cadeia de busca comparando termos de atributo candidatos na cadeia de busca com a lista de atributos para identificar os termos de atributos;

um gerador de consulta acoplado ao intérprete de cadeia (214, 304), o dito gerador de consulta (314) gerando uma ou mais consultas de atributo para buscar dados de guia de programação eletrônica (EPG) com base nos termos de atributos identificados e gerar uma ou mais consultas de texto para buscar os dados de EPG com base nos termos de texto; e

um classificador de resultados para diminuir a classificação de relevância de acertos resultantes da busca condicional na forma corrigida da palavra quando pelo menos uma das uma ou mais consultas de atributo ou uma ou mais consultas de texto retorna acertos que contenham a palavra identificada como sendo escrita incorretamente.

14. Mecanismo de busca, de acordo com a reivindicação 13, **caracterizado** pelo fato de que o intérprete de cadeia compreende ainda um transformador de cadeia e um identificador de atributo.

15. Mecanismo de busca, de acordo com a reivindicação 14, **caracterizado** pelo fato de que o identificador de atributo separa os termos de atributo dos termos de texto na cadeia de busca usando a listagem de atributos.

16. Mecanismo de busca, de acordo com a reivindicação 13, **caracterizado** pelo fato de que o mecanismo de busca realiza buscas em bancos de dados EPG e recursos EPG baseados na rede.

17. Mecanismo de busca, de acordo com a reivindicação 13, **caracterizado** pelo fato de que compreende ainda um compilador de resultados tendo um comparador/filtro de busca e um classificador de resultados.

18. Mecanismo de busca, de acordo com a reivindicação 17, **caracterizado** pelo fato de que o comparador/filtro de busca compara os resultados da busca com o significado de termos postulados pelo intérprete de cadeia.

19. Mecanismo de busca, de acordo com a reivindicação 17, **caracterizado** pelo fato de que o classificador de resultados atribui uma classificação de relevância aos resultados da busca baseados no significado de termos postulados pelo intérprete de cadeia.

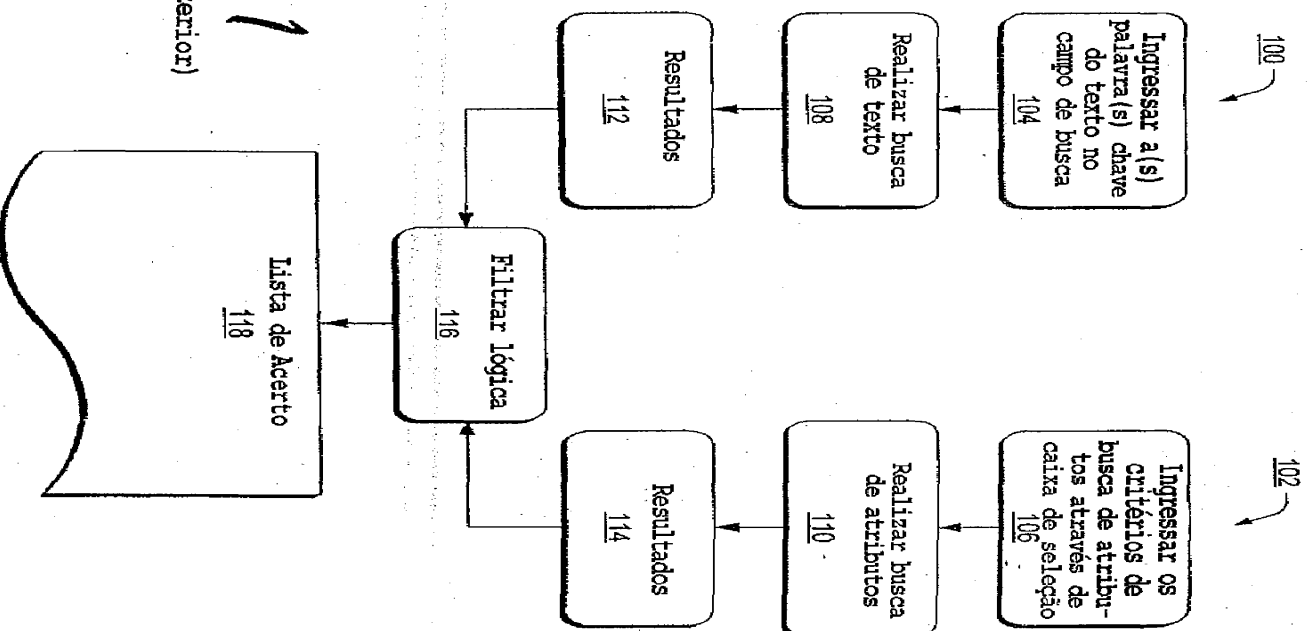


Fig. 1
(Técnica anterior)

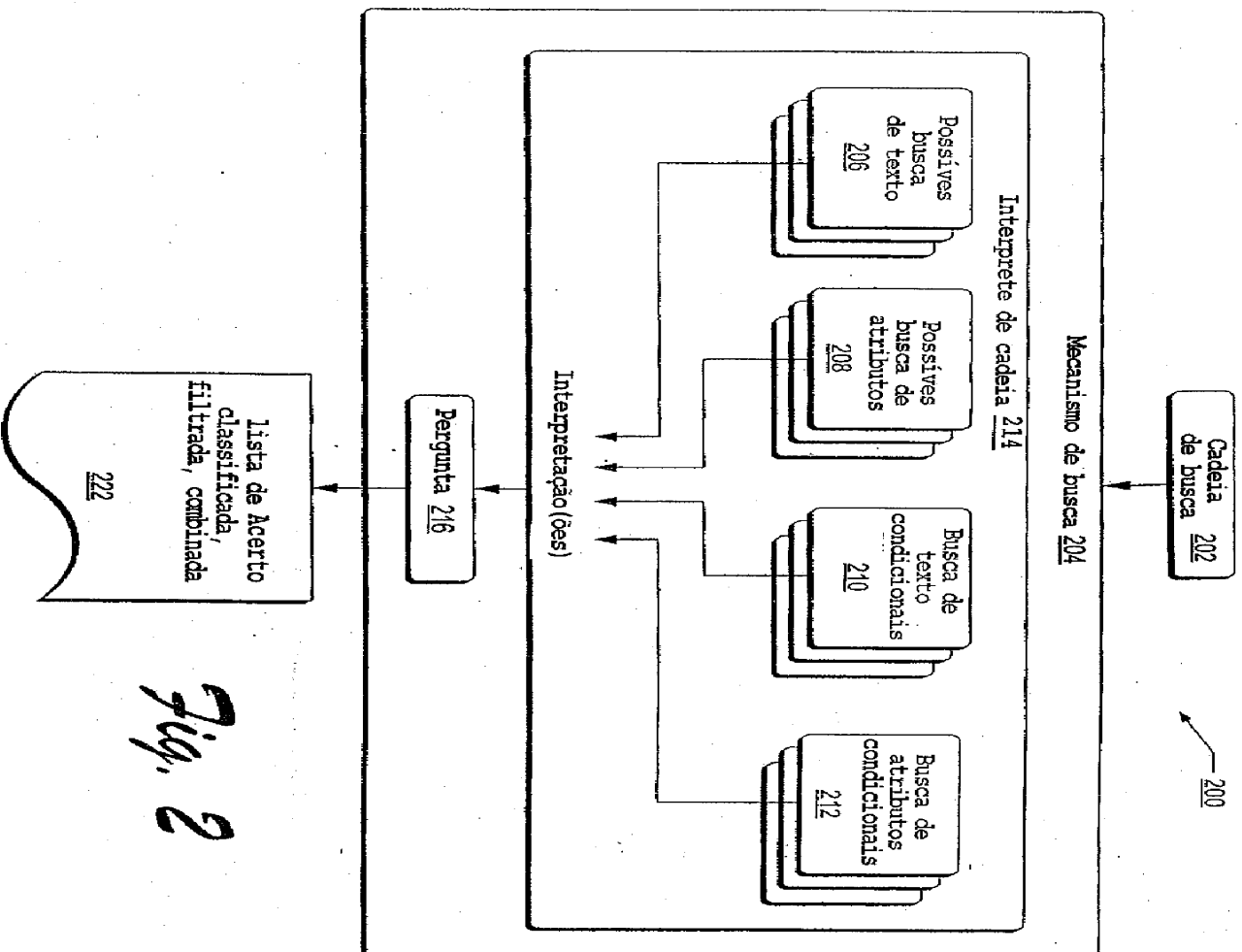


Fig. 2

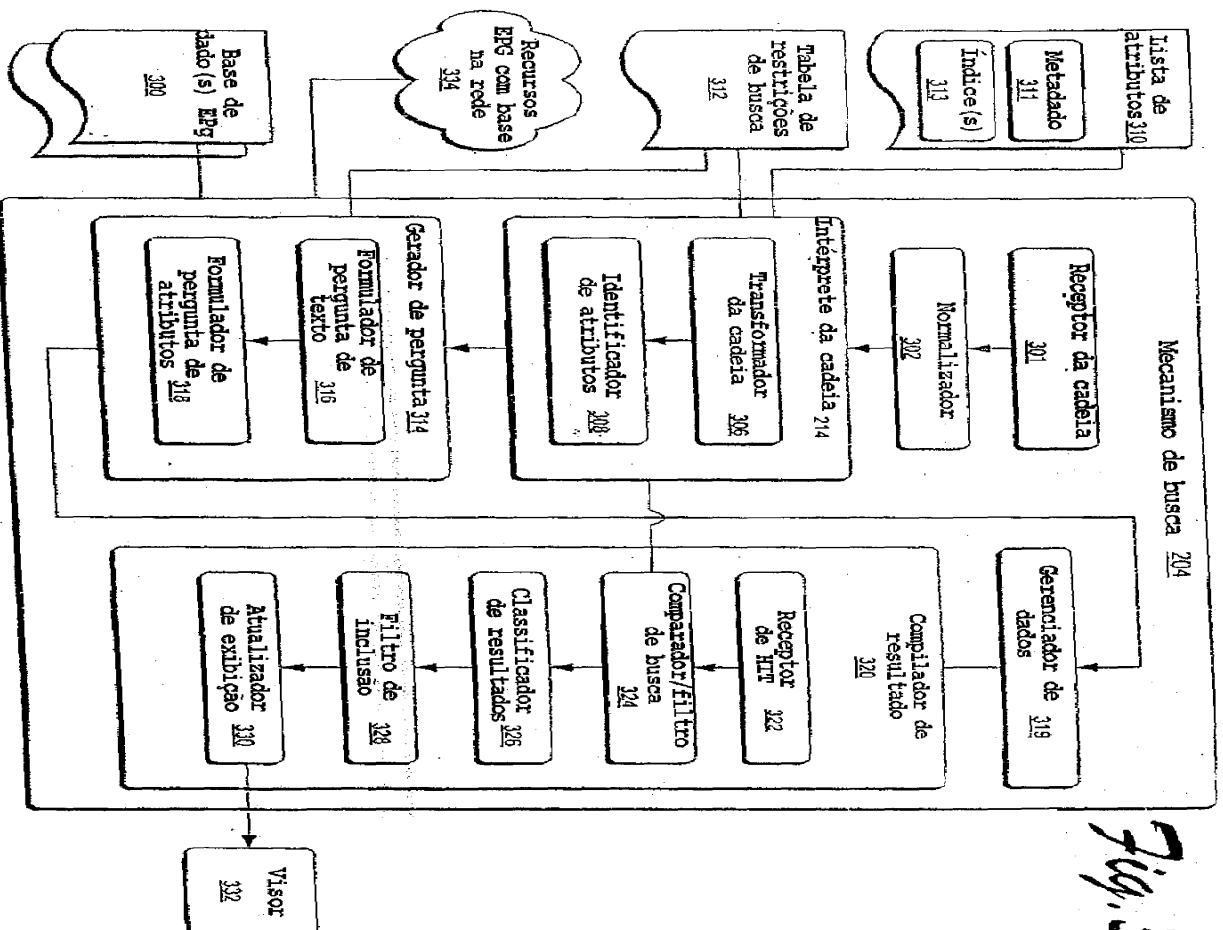


Fig. 3

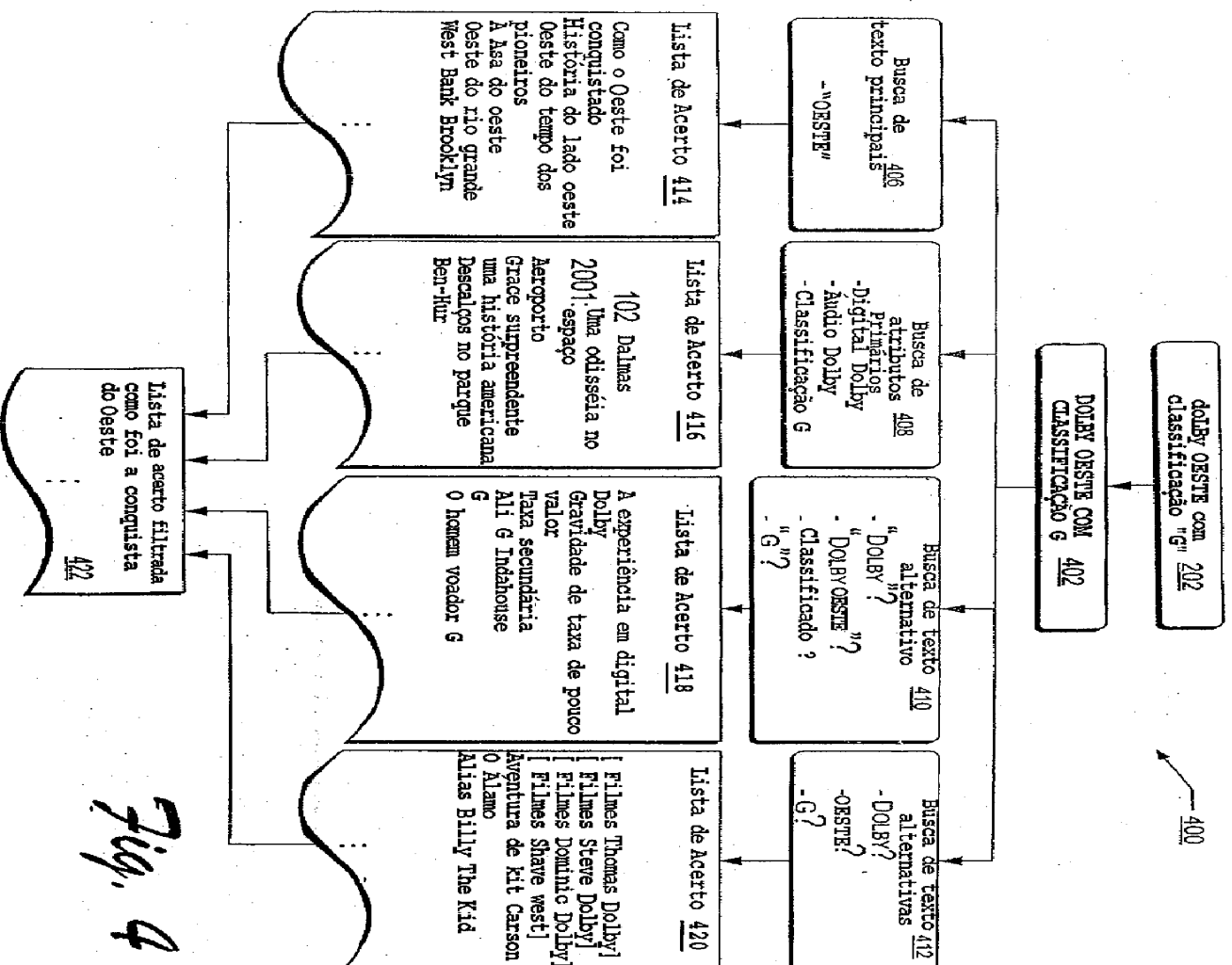


Fig. 4

Busca progressiva 500

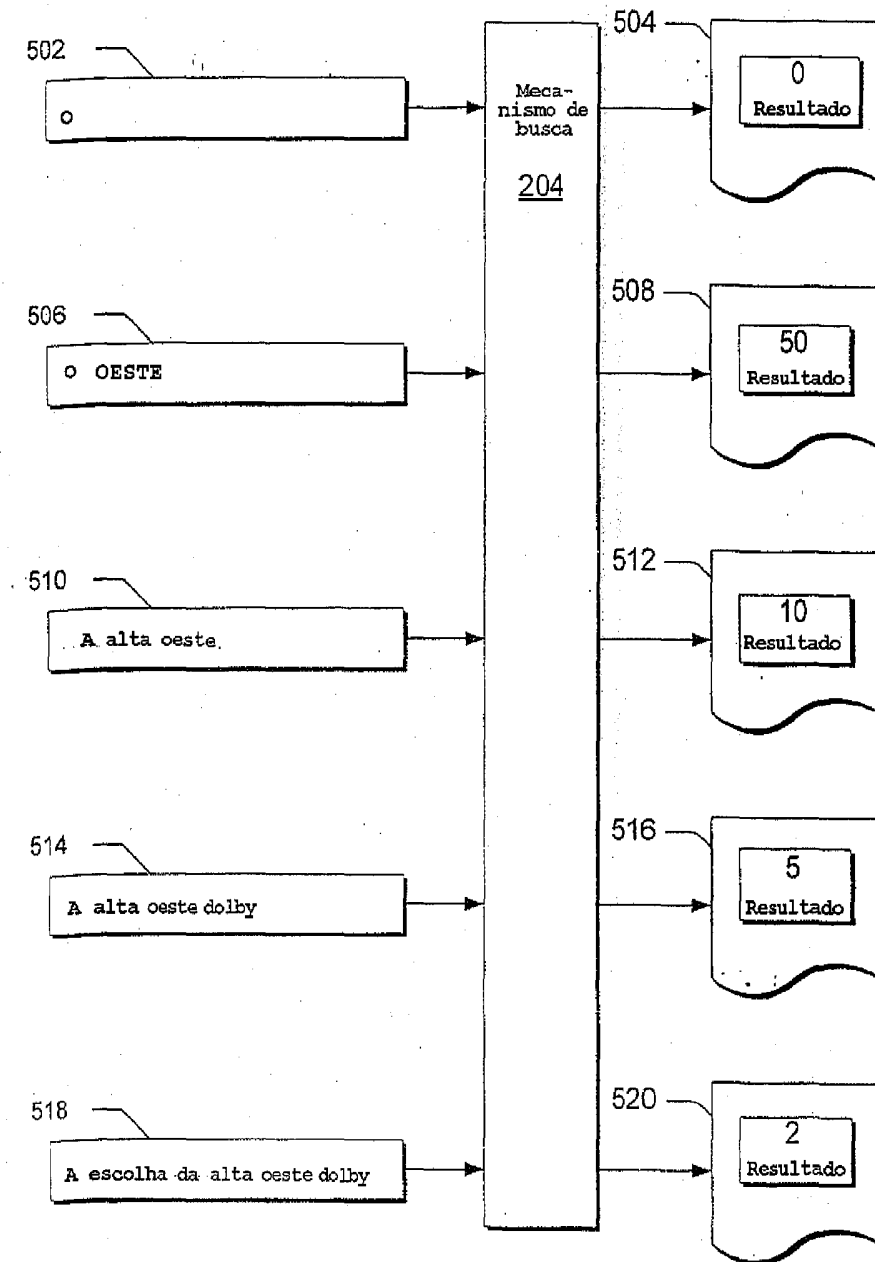
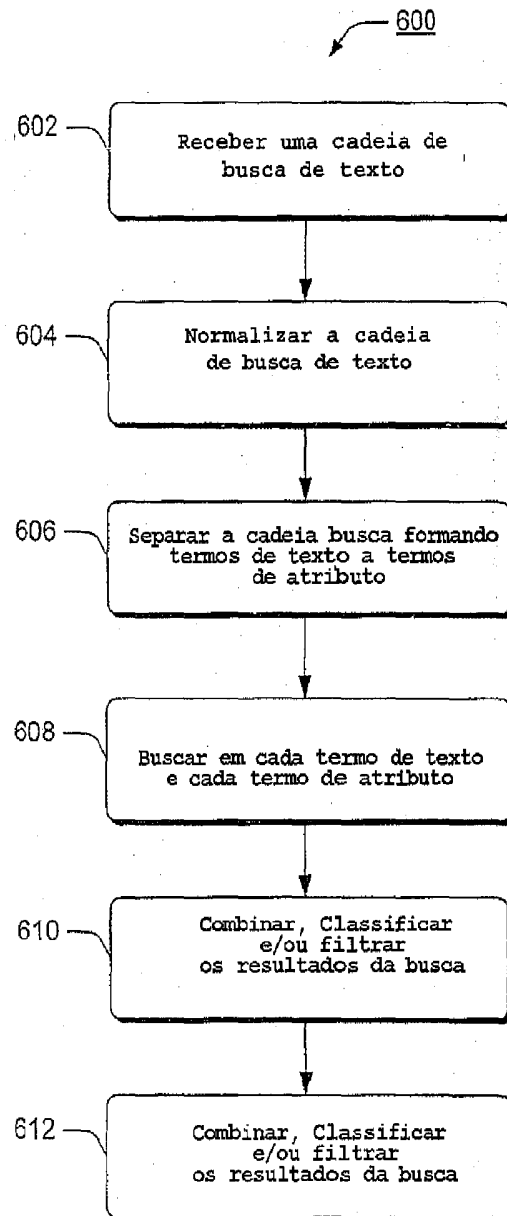
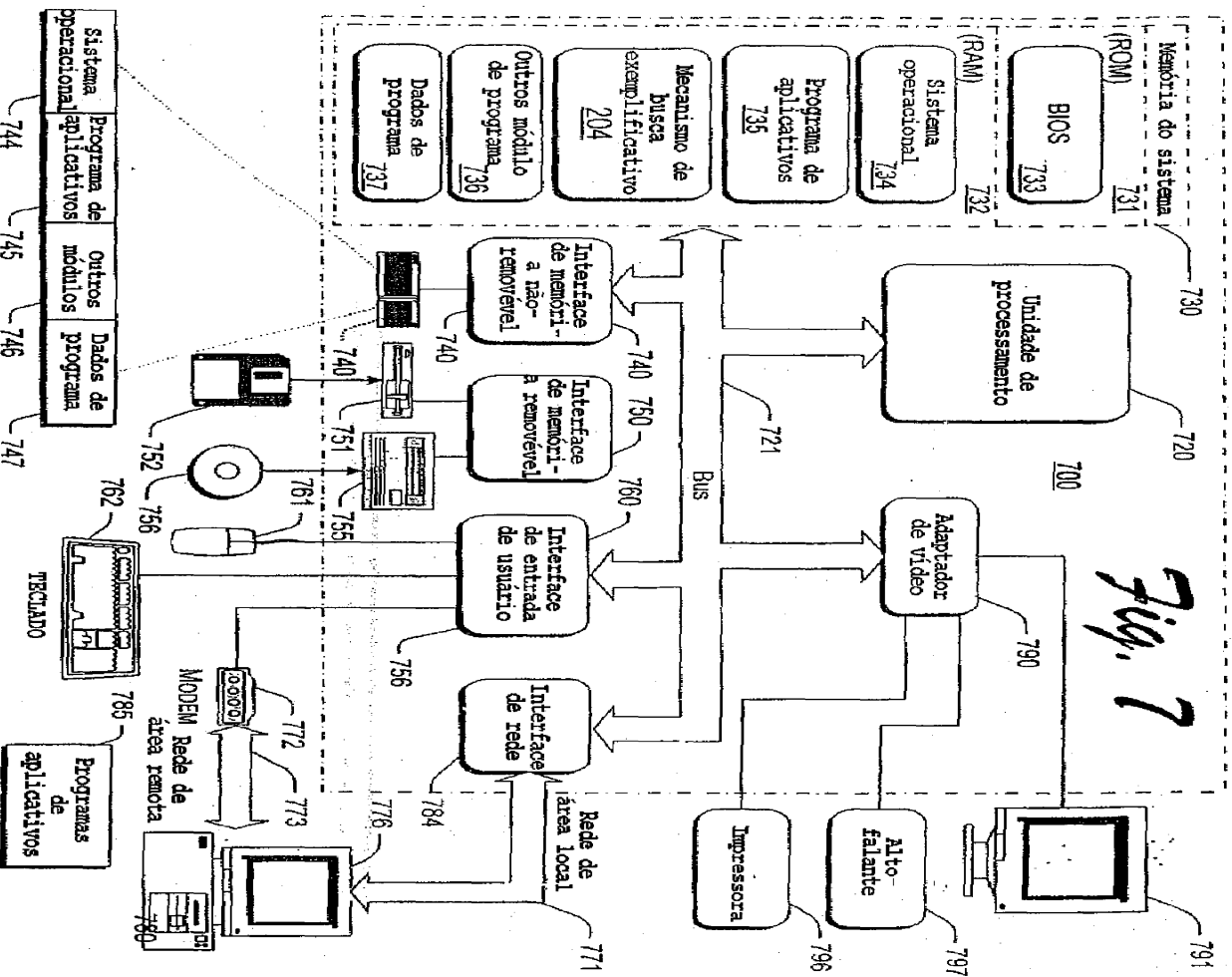


Fig. 5

*Fig. 6*



RESUMO

Patente de Invenção: **“MÉTODO DE BUSCA DE DADOS DE GUIA DE PROGRAMAÇÃO ELETRÔNICA (EPG) E MECANISMO DE BUSCA PARA DADOS DE GUIA DE PROGRAMAÇÃO ELETRÔNICA (EPG)”**.

O objeto da presente invenção inclui um mecanismo de busca para dados se guia de programação eletrônica (EPG) e os métodos relacionados. Em um método exemplificativo, uma cadeia de busca de texto pode ser normalizada formando termos passíveis de busca e termos interpretados tanto como termos de busca de texto, como termos de busca de atributo. Uma ou mais perguntas com condições de busca de graus variados de complexidade são criadas, de acordo com a interpretação dos termos da cadeia de busca. Uma ou mais buscas nos bancos de dados EPG e/ou recursos baseados na Rede, são realizadas tendo-se como base a interpretação dos termos de texto e de atributo e aos resultados é dada uma classificação de relevância, de acordo com a interpretação. Os resultados de busca combinados podem ser agrupados, classificados e filtrados para exibir para o usuário. Os resultados também podem ser exibidos progressivamente, na medida em que cada caracter, de uma cadeia de busca, é ingressado pelo usuário.