Universidade de Aveiro | 17 a 19 de Janeiro 2007

www.ieeta.pt/capsi2006

CAPSI'2006

7ª Conferência da Associação Portuguesa de Sistemas de Informação

Livro de Actas

Editores
Joaquim Arnaldo Martins
Joaquim Sousa Pinto
Carlos Ferreira

CAPSI'2006 – 7ª Conferência da Associação Portuguesa de Sistemas de Informação

Universidade de Aveiro, 18 e 19 de Janeiro 2007

www.ieeta.pt/capsi2006

Livro de Actas

Editores:

Joaquim Arnaldo Martins

Joaquim Sousa Pinto

Carlos Ferreira

FICHA TÉCNICA:

ISBN: 972-789-219-1
ISBN-13: 978-972-789-219-8
Dep. Legal: 252275/06
Edição: Janeiro 2007

Editores: Joaquim Arnaldo Martins,
Joaquim Sousa Pinto,
Carlos Ferreira

URL: http://www.ieeta.pt/capsi2006

Design Web e Capa: Sandra Cruz
Organização Local: Anabela Viegas, IEETA

Compressão de Estruturas Multidimensionais de Dados com vista ao seu Armazenamento em Plataformas Móveis de Pequena Dimensão

Jorge Ribeiro [1], Orlando Belo [2]

1) Departamento de Ciências Básicas e Computacionais, Escola Superior de Gestão e Tecnologia
Viana do Castelo, Portugal
jribeiro@estg.ipvc.pt

2) Departamento de Informática, Escola de Engenharia, Universidade do Minho
Campus de Gualtar, Braga, Portugal
obelo@di.uminho.pt

Resumo

Nos dias de hoje, os gestores necessitam cada vez mais de informação sobre as várias actividades de negócio das suas organizações, para tomarem decisões estratégicas com base em toda a informação coerente das várias vertentes de negócio. O histórico da informação destas organizações é armazenado em repositórios de informação, sobre os quais os servidores de processamento analítico trabalham todas as combinações possíveis de análise, com base em "todas" as possíveis *queries* que os agentes de decisão costumam lançar. Com a evolução dos computadores de bolso, e a sua utilização em grande escala em termos de comunicações e acesso aos dados, os agentes de decisão têm a possibilidade de aceder à informação em qualquer altura e em qualquer lugar. Seria extremamente útil para estes agentes a utilização da sua informação analítica em dispositivos móveis. Este trabalho debruça-se, essencialmente, sobre o problema de processar, comprimir e migrar estruturas multidimensionais de dados para plataformas móveis de pequena dimensão, analisando os algoritmos Dwarf, Cubo Supresso, Cubo Prefixo e Quotient Cube.

Palavras chave: Sistemas de Processamento Analítico, Estruturas Multidimensionais de Dados, Processamento de Cubos e Compressão de Estruturas Multidimensionais.

1. Introdução

Os sistemas móveis "reduziram" drasticamente o espaço e o tempo entre dois lugares, entre duas pessoas. Hoje, disponibilizam-nos um leque de serviços tão vasto que muitas das coisas que fazemos diariamente, nos nossos postos de trabalho, podem ser feitas à distância, sem qualquer problema. Seguindo de muito perto este advento da mobilidade, os fabricantes de plataformas computacionais de pequena dimensão integraram vários dispositivos, que lhes garantem não só a sua autonomia em termos de serviço como também lhes asseguram novos meios de comunicação com outros dispositivos. Como consequência, é frequente encontrarmos já integrados em pequenas plataformas computacionais móveis não só os serviços usualmente disponíveis em plataformas fixas pessoais como também os serviços assegurados pelos telefones móveis actuais mais sofisticados. Os sistemas de processamento analítico, vulgarmente definidos como sistemas *OnLine Analytical Processing* (OLAP) [Codd et al. 1993], operam normalmente sobre a informação contida nos repositórios dos *Sistemas de Data Warehousing* (SDW) e alargam significativamente os horizontes de análise dos agentes de decisão das organizações, ao disponibilizarem meios muito expeditos de análise sobre as várias áreas de actividade das organizações. Estes sistemas assentam sobre a informação armazenada nos SDW, pre-calculando e processando todas as combinações dos agrupamentos de cada entidade dos SDW, e, por fim, materializando-as em *Estruturas Multidimensionais de Dados* (EMD) ou, como vulgarmente são designadas, em Cubo de Dados [Gray et al. 1997]. Os

agentes de decisão mais activos costumam ter o hábito de analisar todos os tipos de informação que têm à sua disposição, desenvolvendo processos de procura constante, em busca deste ou daquele indício, que lhes tragam algumas vantagens no desenvolvimento das suas actividades e no posicionamento das suas organizações nos mercados em que estão inseridas. Hoje são poucos os agentes de decisão que estão a trabalhar num único lugar, gerindo os negócios a partir daí. Todavia, apesar dos avanços enormes das tecnologias da informação e das comunicações, os agentes têm, por vezes, algumas dificuldades em aceder aos seus sistemas de informação e, em particular, às estruturas de dados de suporte à decisão que habitualmente utilizam nas suas actividades diárias. Isto porque, embora as limitações destes dispositivos móveis estejam a ser eliminadas dia após dia, estes continuam a ter algumas dificuldades quando nos reportamos às suas características de mobilidade, portabilidade e comunicação sem fios. É o caso da capacidade de armazenamento, a limitação da largura de banda, as reduzidas dimensões dos ecrãs para análise da informação, as quebras de comunicação frequentes, entre outros. Será possível, assim, que um dispositivo móvel tenha a capacidade de processar informação analítica volumosa? Haverá a possibilidade de armazenar informação analítica num dispositivo móvel em que à partida não seria possível o seu armazenamento dada a limitação física de espaço? Na realidade tudo isso é possível actualmente. Recentemente várias técnicas [Sismanis et al. 2002] [Wang et al. 2002] vocacionadas para o processamento e compressão de EMD foram apresentadas, conseguindo comprimir de forma eficiente este tipo de estruturas sem ter a necessidade de proceder à descompressão da informação quando as *queries* são solicitadas. Em vários cenários reais, consegue-se já aplicar técnicas de processamento analítico e de compressão de EMD com vista à sua exploração em dispositivos móveis, tendo em conta as várias limitações associadas a estes dispositivos, relativamente ao armazenamento e consequentemente visualização da informação que a priori não seria possível ser nele armazenada. Desta maneira, os gestores de negócio conseguem aceder a informação analítica para a tomada de decisões estratégicas, tornando-os, deste modo, independentes das estações de trabalho a que estavam habituados para analisarem a informação e abrindo novas formas de visualização da informação de que necessitam, de forma simples e rápida. Neste trabalho, apresentamos o POCKET CUBES, um sistema protótipo especificamente orientado para o processamento analítico e compressão de EMD, utilizando algoritmos como Cubo Supresso e Cubo Prefixo. Aqui, demonstramos que é possível reduzir o espaço de armazenamento em cerca de 70% de maneira a conseguir armazenar uma EMD em plataformas móveis (ou não) que à partida não teriam capacidade de a armazenar.

2. Processamento e Compressão de Estruturas Analíticas

O processamento de EMD é, em regra geral, uma tarefa computacional bastante difícil, uma vez que, a partir da informação armazenada nos repositórios analíticos, são processadas todas as combinações possíveis de agregação entre as várias dimensões de análise das suas estruturas de dados [Kimball et al. 1998]. Vários estudos sobre o processamento de EMD [Harinarayan et al. 1996] [Han et al. 2001] [Xin et al 2003] foram apresentados na área do processamento de EMD, especialmente sobre a adopção de metodologias para a selecção de partições de EMD a serem processadas seguindo uma determinada orientação: Top-Down [Zhao et al. 1997] e Bottom-up Computation [Beyer & Ramakrishan 1999]. Para analisar essa dificuldade do processamento, consideremos genericamente uma relação R com n atributos de dimensão (D1, D2, ..., Dn) e um atributo de medida M. A EMD processada a partir desta relação R em todas as dimensões, aplicando o operador de agrupamento (GROUP-BY) irá gerar 2n agrupamentos distintos constituindo desta forma a EMD final [Gray et al. 1997]. Consideremos, também, a relação R (A,B,C,M) da tabela 1 como exemplo de uma simples relação (base) típica em repositórios analíticos. Esta relação R tem três dimensões {A}, {B}, {C} e uma medida M, sendo adoptada a função SOMA como função de agregação. Para facilitar a análise de resultados com designações reais, podemos considerar que as dimensões {A}, {B} e {C} correspondem respectivamente, ao Dia da semana, aos Produtos e às Lojas, enquanto que o atributo de medida

corresponde ao total de Vendas efectuado ao longo do tempo pelas várias filiais de uma hipotética organização.

	A	B	C	M
t1	1	8	7	6
t2	1	3	7	12
t3	2	8	3	9

(a)

ou

DIAS	Produtos	Lojas	Vendas
Sabado	Computador	Lisboa	6
Sabado	Teclado	Lisboa	12
Domingo	Computador	Braga	9

(b)

Tabela 1 – Tabela base de um *Data Warehouse*

Esta relação R contém três tuplos ou células (t1, t2 e t3) e irá processar 23 agrupamentos (figura 1 (a)), de maneira a garantir todas as possibilidades de análise da informação. Cada aplicação do operador de agrupamento corresponde a um conjunto de tuplos (ou células), podendo ser também descritos como tuplos sobre as dimensões. As células do processamento da EMD são apresentadas na figura 1 (a), sendo a interligação das células de cada agrupamento apresentadas na figura 1 (d). O símbolo "*", como valor de uma dimensão, denota a generalização do valor da dimensão, ou seja, um valor especial para denotar o valor "ALL" em que qualquer valor pode ser assumido. Para exemplificar basicamente estas operações, consideremos a figura 1 (d) por exemplo que, a partir da célula (1,8,7:6) se efectua a operação de refinamento (aplicando o operador de agrupamento na dimensão {C}) para a célula (1,8,*:6), e a partir deste para (1,*,*:18) aplicando o operador de agrupamento na dimensão {B}. Posteriormente a agregação pode ser efectuada para a célula (1,3,*:12), seleccionando a dimensão {C} no operador de agrupamento. Esta navegação sobre os dados e sobre os agrupamentos, pode ser facilmente visualizada nas figuras 1 (b) e na figura 1 (d) . A implementação realizada do processamento da EMD está apresentada na figura 1 (a), em que cada uma das células apresentadas pertence a um sub-agrupamento, sendo o valor da coluna "ID-Agrupamento" a identificação do sub-agrupamento correspondente à célula. O valor desta coluna tem uma correspondente relação com a dimensão da tabela 1, sendo descrita esta equivalência na figura 1 (c). Esta numeração deve-se ao facto do processamento das dimensões (independentemente das designações a atribuir a cada uma delas) ter de ser numerado consoante a ordem do processamento do algoritmo na orientação da selecção das dimensões a serem particionadas. Uma das particularidades do processamento analítico é o facto de, através da interligação das várias células da EMD, se poder formar o denominado reticulado de células da EMD (figura 1 (d)), vulgarmente designado por "cube lattice". A interligação das células dos vários agrupamentos permitem semanticamente executar as operações de agrupamento ou refinamento entre as células através de algum atributo de dimensão necessárias para as ferramentas de processamento analítico. Ao analisar o reticulado das células da EMD e tendo em atenção o aumento do número de células interligadas entre si, surgem entre estas ligações algumas características peculiares que merecem ser analisadas. Uma delas corresponde à existência de várias células com o mesmo valor agregado e a outra refere-se à redundância semântica que ocorre nas interligações entre as várias células. Esta última característica é aquela pela qual os algoritmos mais recentes [Sismanis et al. 2002] [Wang et al. 2002] de processamento e compressão de EMD dão especial atenção, tentando conseguir aglomerar células com partilha de atributos de dimensão, por forma a armazenar apenas alguns tuplos e assim, reduzir o espaço de armazenamento da EMD. Esta característica da redundância entre as células da EMD pode ser dividida em dois tipos: redundância Prefixa e redundância Sufixa [Sismanis et al. 2002]. A generalidade dos algoritmos Dwarf, Cubo Supresso e Quotient Cube exploram o fenómeno da redundância Prefixa e Sufixa de modo eficiente, tentando criar estruturas com referências entre as células da EMD de maneira, a armazenar fisicamente menos células da EMD, mantendo no entanto, todos os requisitos funcionais das ferramentas OLAP. Na maioria dos casos, estes repositórios analíticos (DW) armazenam grandes volumes de dados e como consequência, o processamento de todos os agrupamentos é na generalidade dos casos muito custoso, quer em termos computacionais, quer em termos de espaço de armazenamento. Neste caso, em particular, a geração dos vários agrupamentos ocupam muito espaço extra e uma das formas de

minimizar estas dificuldades é a utilização de técnicas de optimização, que podem ser caracterizadas resumidamente em: vistas materializadas [Gupta & Mumick 2005], utilização de técnicas de indexação, como é o caso das Cube Forest [Johnson & Shasha 1997], Cubetrees [Roussopoulos et al. 1997], QC-Trees [Lackshmanan et al. 2003] e DC-Trees [Ester et al. 2000], processamento dos dados de forma paralela e distribuída e as técnicas de compressão de dados [Sismanis et al. 2002] [Wang et al. 2002] [Lackshaman et al. 2002].

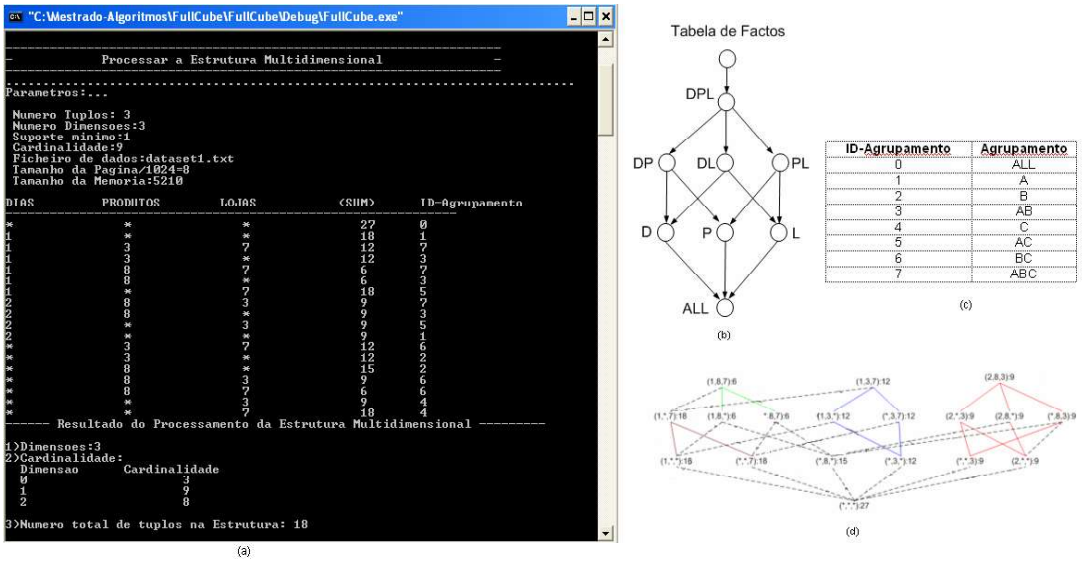

Figura 1 – Resultado do processamento da EMD a partir do exemplo da Tabela 1

As técnicas de redução de dados tentam adoptar estratégias de redução de dados, como por exemplo o somatório dos dados ou modelos de aproximação de respostas às *queries*, seleccionando estimativas de valores para disponibilizarem rapidamente a informação. Estes estudos centram-se no desenvolvimento de técnicas estatísticas em que produzem somatórios genéricos através da representação de esquemas matemáticos (matrizes multidimensionais), de maneira a armazenar os valores e aplicar técnicas matemáticas sobre eles, de forma a produzir representações reduzidas dos espaços multidimensionais. Contudo este tipo de técnicas de redução tem pouca aplicação em ambientes de redução de EMD, uma vez que os agentes de decisão necessitam de respostas exactas às *queries* invocadas e não de aproximações. Por esta razão, focaremos o nosso trabalho nas técnicas de processamento e compressão de EMD. Neste sentido, o estudo de Sismanis et al. (2002) apresenta uma estrutura denominada por "Dwarf" (Figura 2 (a)), para armazenar a EMD eliminando os dois tipos de redundância encontrados entre as células da EMD. Conseguem criar uma estrutura altamente compressa de durante o processamento da EMD. O trabalho desenvolvido por Long-gang et al. (2004) (Figura 2 (b)) verifica que embora a estrutura Dwarf elimine tanto a redundância Prefixa como a Sufixa, na realidade em alguns casos não se consegue eliminar totalmente a redundância Sufixa, criando adicionalmente nodos redundantes. Wang et al. (2002) apresentam o Cubo Supresso, baseando-se no conceito Tuplo Base Simples "Base Single Tuple (BST)", suprimindo tuplos da EMD num único tuplo representativo. Conseguem com esta técnica evitar o armazenamento de todos os tuplos da EMD. Neste estudo são apresentados também além do Cubo Supresso (BST - Cubo Supresso), o Cubo Mínimo, o BU - Cubo Supresso e o BU - Cubo Supresso Reordenado. Feng et al. (2004) melhoraram o Cubo Supresso detectando que o mesmo continuava a não eliminar totalmente a redundância Prefixa, propondo uma nova estrutura de armazenamento, partilhando os prefixos entre os tuplos, denominando este novo algoritmo de Cubo Prefixo ("PrefixCube") (figura 2 (c)).

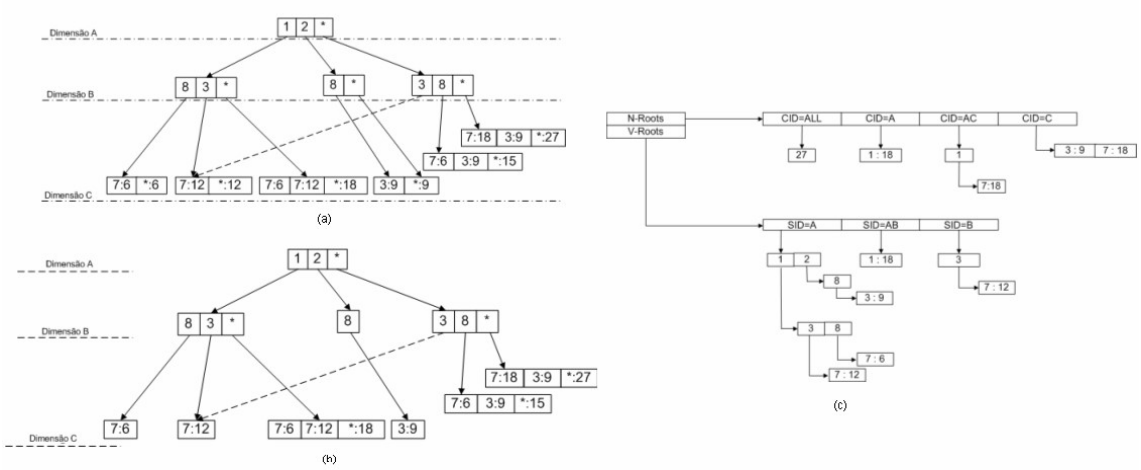

Figura 2 – Resultado dos algoritmos: Dwarf, Dwarf Supresso e Cubo Prefixo

Lakshmanan et al. (2002), baseando-se na ideia de agrupar células com o mesmo valor agregado em classes de células, apresentam o "Quotient Cube". Neste algoritmo de processamento e compressão de EMD, em vez de, armazenar todas as células da EMD apenas armazena os tuplos limite superior e inferior de cada uma das classes. O mesmo autor num outro estudo [Lakshmanan et al. 2003] apresenta uma estrutura denominada de QC-Trees, para implementar o algoritmo QC.

3. Desempenho dos Algoritmos de Processamento e de Compressão

Neste trabalho implementámos os algoritmos BU - Cubo Supresso e Cubo Prefixo para processamento de EMD. Para este efeito utilizamos vários conjuntos de dados caracterizados, essencialmente, em dois tipos: dados sintéticos e dados reais, tendo sido utilizada a SOMA como função de agregação. Todas as experiências foram realizadas em máquinas com processador Pentium IV 3Ghz com 512 MB DDR RAM, disco de 40 GB com 72 RPM, sistema operativo Windows XP Profissional e o conjunto de dados lido a partir de um ficheiro. Os algoritmos foram implementados em Visual C++ 6.0. Os resultados obtidos para a análise do rácio de compressão de EMD estão apresentados na figura 3 (a), (b) e (c). Como podemos analisar, a eficiência do algoritmo Cubo Prefixo face ao BU - Cubo Supresso é, em média, cerca de 25.5% para conjuntos de dados reais. Por outro lado, com o aumento da densidade do conjunto de dados, a taxa de compressão do algoritmo BU – Cubo Supresso aumenta, ao contrário do obtido pelo algoritmo Cubo Prefixo, que efectivamente diminuiu. A distribuição Zipf [Zipf 1949] foi utilizada para criar conjuntos de dados de maneira a analisar a dispersão dos dados e como podemos constatar, assim que o factor Zipf aumenta, o Cubo Prefixo tem melhor desempenho no rácio de compressão sobre o BU - Cubo Supresso. Em termos gerais, o algoritmo Cubo Prefixo é em média cerca de 30% melhor que o BU – Cubo Supresso. Outra observação que se pode analisar é o facto da curva do Cubo Prefixo ser regular (sem grandes variações), indicando que o Cubo Prefixo não ser sensível a variações na dispersão dos dados. Para a análise do tempo de processamento e compressão de cada um dos algoritmos (figura 3 (d) (e) e (f)), e embora consideremos os resultados obtidos como satisfatórios, temos consciência de que, se as características do equipamento fossem melhores em termos de processador e em termos de memória e rapidez de armazenamento, conseguiríamos obter melhores resultados. Como podemos analisar nessas figuras, para conjuntos com poucas dimensões, o tempo de processamento é bastante inferior do que o obtido para conjuntos com maior número de dimensões. Por outro lado, embora a maior parte dos conjuntos o algoritmo Cubo Prefixo seja mais eficiente em termos de tempo de processamento comparativamente com o BU – Cubo Supresso, para conjuntos de dados com menor número de dimensões, o algoritmo BU – Cubo Supresso é mais eficiente, ocorrendo o inverso com o aumento do número de

dimensões. A razão prende-se com o facto do algoritmo Cubo Prefixo despender mais tempo na análise de prefixos comuns entre os tuplos por ter menos dimensões, a eficiência da análise da partilha de prefixos comuns no tempo de armazenamento não se nota, notando-se mais à medida que o número de dimensões aumenta no conjunto de dados.

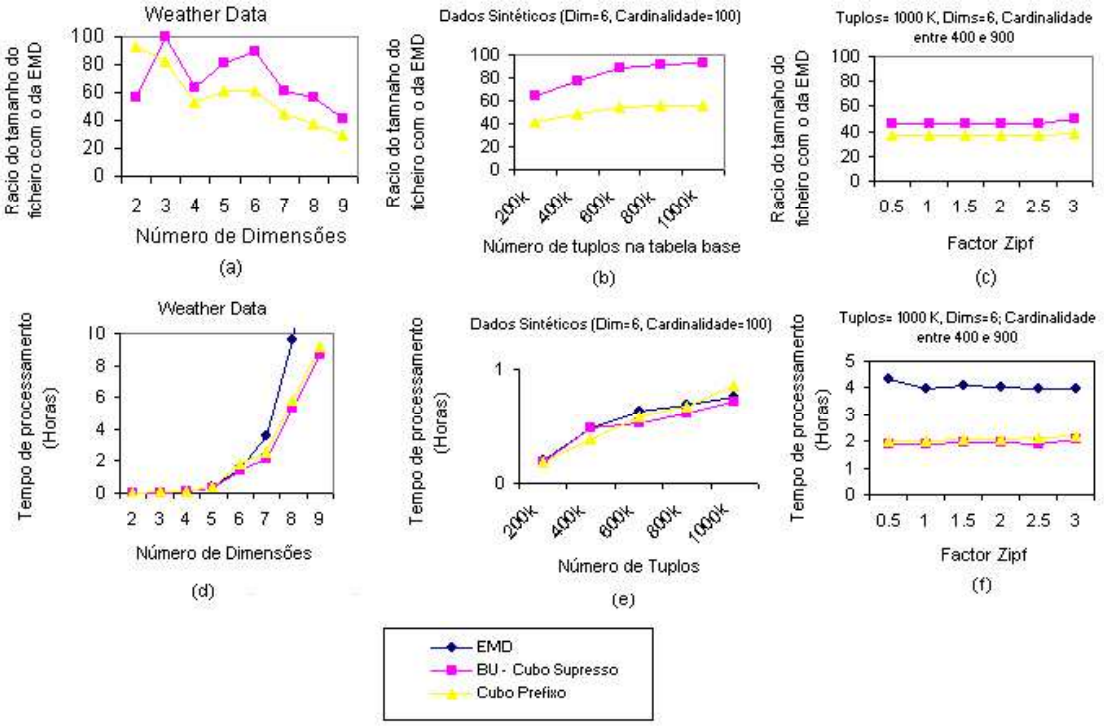

Figura 3 – Avaliação do rácio de compressão e tempo de processamento de EMD

Analisando a globalidade dos algoritmos de processamento e compressão de estruturas multidimensionais abordados neste nosso estudo e embora não tivéssemos implementado a estrutura QC-Tree (para implementação do algoritmo Quotient Cube), consideramos que em termos de rácio de redução esta estrutura deve superar os resultados obtidos por qualquer um dos outros algoritmos de compressão abordados, mesmo quando comparada com os resultados obtidos pelo algoritmo Cubo Prefixo. Isto porque a estrutura QC-Tree segue a semântica do algoritmo Quotient Cube, aglomerando células da estrutura multidimensional em classes necessitando apenas de armazenar as células limite inferior e superior de cada uma das classes, enquanto que os outros algoritmos necessitam de armazenar mais células. Por exemplo, o algoritmo Dwarf necessita de armazenar todas as células ao passo que os algoritmos BU – Cubo Supresso e Cubo Prefixo embora suprimam alguns tuplos (não necessitando de serem armazenados) e optando ou não pela técnica de partilha de prefixos entre as células, necessitam de armazenar sempre mais células do que a estrutura QC-Tree. Manter nos dias de hoje uma EMD actualizada é um requisito em ambientes de processamento analítico, uma vez que é necessário disponibilizar a informação mais recente aos agentes de decisão. Na generalidade dos casos, o volume da informação nas EMD é elevado e as actualizações são normalmente frequentes e volumosas. Vários estudos [Mumick et al. 1997] definem e debruçam-se sobre esta problemática da manutenção incremental da informação em repositórios analíticos. Todos eles têm o objectivo de evitar o reprocessamento de toda a informação analítica, apresentando metodologias e esquemas de manutenção de maneira eficiente e transparente para os agentes de decisão. O processo de manutenção de vistas também é um problema bastante estudado, havendo alguns estudos realizados [Hurtado et al. 1999]. Cada um dos algoritmos apresentados em epígrafe, como o Dwarf, o Cubo Supresso e o Quotient Cube, para actualizarem a informação contida nas suas estruturas apresentam alguns métodos eficientes para o efeito.

4. Um Caso de Aplicação Prática

4.1 Hardware Utilizado e Conjunto de Dados

O sistema POCKET CUBES permite processar e comprimir estruturas multidimensionais e, posteriormente, migrá-las para dispositivos de armazenamento com restrições de espaço, passíveis de serem utilizados em plataformas móveis de pequena dimensão. Para a análise do desempenho deste protótipo e tendo em consideração a restrição do limite de espaço de armazenamento da informação em dispositivos móveis, utilizamos um cartão *Secure Card* de 1GB Ultra II com capacidade de transferência de 9 Mbps. Para efectuar o processamento da EMD recorremos a uma estação de trabalho tal como a apresentada na secção 2.1 e com uma base de dados em Microsoft SQL Server 2000. As restrições que se colocam no caso em estudo referem-se ao espaço máximo de armazenamento, ao tempo disponível para se efectuar o processamento da EMD e ao tempo de transferência da informação para dispositivos de armazenamento. Assim, a restrição de espaço de armazenamento do dispositivo foi definido como 1 GB, a velocidade (média) de 9 Mbps referente à transferência da informação para o dispositivo de armazenamento e o tempo disponível para o processamento e compressão da EMD foi de 11 Horas. O conjunto de dados deste caso de estudo contém um milhão de registos referentes às Vendas dos últimos cinco anos registadas por uma organização que se dedica ao comércio de artigos e serviços de Optometria, dispondo de cinco lojas e trinta vendedores. Esta organização possui 5869 artigos caracterizados por dez categorias e 128 marcas, estando registados na base de dados neste momento 9202 clientes. Os requisitos de análise da informação por parte dos gestores desta organização centram-se no somatório das Vendas por períodos (ano e mês), por marca de artigo, pelo artigo, pela categoria dos artigos, por vendedor, por loja e finalmente por cliente.

4.2 Processamento e Compressão

O conjunto de dados estava armazenado, tal como referimos, numa base de dados *Microsoft SQL Server 2000* e o fluxo da componente de processamento do POCKET CUBES foi implementado com o recurso a Serviços de Transformação de Dados (*Data Transformation Services* (DTS)), também do *SQL Server 2000*. Este fluxo de tarefas foi dividido em vários de menor dimensão, cada um associado a um processamento específico, nomeadamente: análise das restrições do tempo disponível e do espaço máximo de armazenamento e para a selecção das dimensões e o número de registos para a selecção do conjunto de dados. Depois desta primeira tarefa, de maneira a seleccionar o mais adequado para garantir o processamento com base nas restrições e com base nas restrições definidas, o sistema com base no número de dimensões e no número de registos do conjunto de dados, analisa um conjunto de estimativas em termos de espaço de armazenamento, de tempo de processamento para os algoritmos BU Cubo Supresso e Cubo Prefixo, e tempo de transferência da informação para o dispositivo de armazenamento. Para este exemplo, os valores estimados para o conjunto de dados foram de 5.3 horas para o tempo de processamento do Cubo BU-Supresso e 5.81 Horas para o Cubo Prefixo. Em termos de espaço de armazenamento estimado foi de 795.2 MB para o BU - Cubo Supresso e 535.54 MB para o Cubo Prefixo. Com base nos valores estimados para o armazenamento e considerando a restrição da velocidade média de transferência da EMD para o dispositivo, os valores foram de 20 minutos para o BU-Cubo Supresso e de 14 minutos para o Cubo Prefixo. Com base nestes valores da velocidade de transferência, a estimativa para o tempo de processamento (e transferência) dos algoritmos BU - Cubo Supresso e Cubo Prefixo é de 5.5 Horas e 5.95 Horas, respectivamente. Desta maneira, ambos os algoritmos conseguem processar e transferir a EMD para o dispositivo de armazenamento dentro da restrição de tempo disponível. Em termos de limite máximo para o espaço de armazenamento, foi estipulado em 1 GB, donde ambos os algoritmos (com base nas estimativas) conseguem garantir esta restrição. Caso a restrição de espaço máximo de armazenamento fosse por exemplo de 700 MB, então o fluxo de decisão iria seleccionar o algoritmo Cubo Prefixo, uma vez que o algoritmo BU - Cubo Supresso (com base nas estimativas) não conseguiria garantir a restrição de espaço disponível.

Tal como referimos no estudo dos algoritmos de processamento e compressão Cubo Supresso e Cubo Prefixo, na selecção de um determinado agrupamento, o protótipo implementa o método de "Expansão", extraindo os tuplos supressos a partir dos tuplos BST. Na eventualidade de nenhum dos algoritmos conseguir garantir ambas as restrições, o protótipo contempla um "plano de contingência" para definir um conjunto de possíveis novas dimensões e um novo número de tuplos para construir um novo conjunto de dados a ser processado. Este "plano" é definido pelos administradores do sistema tendo como base as necessidades dos agentes de decisão. Por exemplo, se não for possível processar a EMD na sua totalidade então, os agentes poderão delinear algumas alternativas, como por exemplo, as dimensões vendas por ano, mês e por vendedor. Desta maneira, o conjunto de dados será mais pequeno o mesmo acontecendo com a EMD. Assim, o fluxo de decisão e com base nas restrições definidas, o algoritmo seleccionado foi o Cubo Supresso, demorando cerca de quatro horas e trinta minutos em que a EMD ocupou cerca de 580 MB. Opcionalmente, com o objectivo de analisar o desempenho do protótipo, procedemos ao processamento da EMD com o algoritmo Cubo Prefixo e um outro teste efectuando, apenas, o processamento da EMD sem proceder à compressão. Para o processamento do Cubo Prefixo, obtivemos um espaço total de 714 MB (748.714.920 bytes) demorando cerca de 5 Horas o seu processamento. Quanto ao processamento da EMD, sem efectuar a tarefa de compressão obtivemos um espaço total de 2.230 MB demorando cerca de 4 horas e 94 minutos. Com efeito, o rácio de compressão de 0.35 obtido pelo algoritmo Cubo Supresso e um rácio de 0.24 obtido pelo algoritmo Cubo Prefixo. O algoritmo que conseguiu processar e comprimir a EMD, ocupando um menor espaço de armazenamento, foi o Cubo Prefixo. Com base nestes resultados, o protótipo foi capaz de processar e comprimir a EMD garantindo as restrições de tempo disponível para o processamento e o espaço máximo de armazenamento. Com efeito, podemos observar os seguintes pontos: o protótipo foi capaz de processar e comprimir a EMD, garantindo as restrições pré-definidas. Os valores estimados estiveram próximos dos realmente obtidos, após o processamento, considerando que o valor de cerca de 37% como margem de erro poder ser aceitável com base nas condições do equipamento de suporte. Conseguiu-se reduzir a EMD em cerca de 75%, armazenando-a em dispositivos que à partida, não teriam capacidade de o fazer.

4.3 Visualização da Informação

Desejavelmente, a visualização e interrogação da informação analítica são uma necessidade cada vez maior por parte dos gestores. Todavia, a implementação deste protótipo apenas contempla a visualização dos registos de um agrupamento previamente seleccionado, não criando métodos para explorar a informação, tanto a nível dos vários tipos de *query*, como, disponibilizar operações de navegação sobre a informação. Como referimos ao longo deste estudo, por um lado, este será um dos trabalhos futuros e, por outro, as operações de navegação podem ser disponibilizadas por ferramentas de processamento analítico que poderiam importar os vários ficheiros processados pelo protótipo e fornecer aos gestores meios de análise mais expedita. Na generalidade dos casos, os agentes de decisão transportam na sua vida quotidiana computadores de bolso que lhes permitem aceder à informação da sua organização. Porém, dado o seu tamanho tão pequeno, limitam o potencial de visualizar toda a informação, obrigando em vários casos quando o volume de informação é enorme, a uma constante selecção de páginas de maneira a se conseguir visualizar toda a informação. Perante este facto, a visualização de grandes volumes de dados neste tipo de dispositivos deverá ser ponderada pela sua importância e necessidade dos agentes de decisão, sendo por vezes preferível a visualização de somatórios genéricos de informação em vez de analisar a informação com mais detalhe. Isto porque, à medida que se vai seleccionando outros agrupamentos com vários níveis de agregação, o número de registos aumenta. Para este caso de estudo e como referimos, o protótipo processou 256 agrupamentos, sendo disponibilizada a informação analítica (Figura 4) em binário ou em XML.

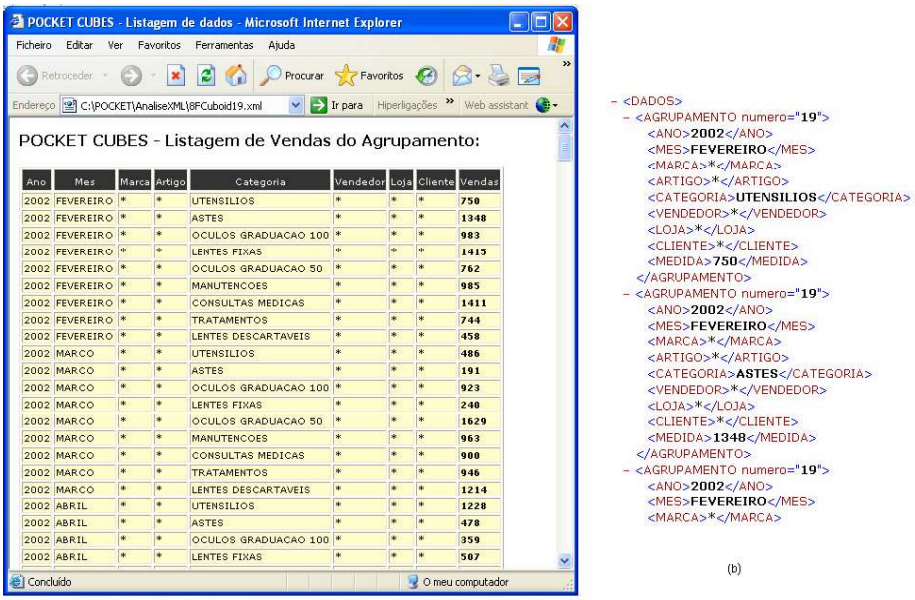

Figura 4 – Visualização da Informação da EMD para o caso de estudo

5. Conclusões

Os estudos que foram desenvolvidos durante a realização deste trabalho, sobre os algoritmos de processamento e compressão de estruturas multidimensionais de dados, permite-nos, agora, ter uma ideia concreta acerca da forma como migrar repositórios analíticos para plataformas móveis de pequena dimensão. Este problema de migração já não é algo que nos levanta grandes dificuldades se tivermos em conta os algoritmos estudados e as suas características de aplicação. Apesar disso, sabemos que não podemos migrar toda e qualquer estrutura multidimensional devido a problemas de espaço de armazenamento, tempo de processamento e tempo de transferência da informação para dispositivos de armazenamento com algumas limitações em termos de velocidade de transferência da informação. Com o conhecimento adquirido neste trabalho de dissertação, podemos, com certeza, preparar e migrar com sucesso estruturas multidimensionais de pequena e médio porte para plataformas móveis de pequena dimensão. Desta forma, conseguimos garantir o tão desejado objectivo de colocar num POCKET PC (ou noutro dispositivo semelhante) uma estrutura de suporte à decisão a ser manipulada, eventualmente, por uma folha de cálculo, disponibilizando desta maneira aos agentes de decisão a Em suma, neste nosso estudo foram abordados áreas e temas de grande importância para quem implementa, ou pensa implementar, uma aplicação de processamento analítico numa plataforma computacional de pequena dimensão. Entendemos que os assuntos abordados e o trabalho desenvolvido abrem caminho a novos projectos, podendo desta maneira, disponibilizar aos agentes de decisão, ferramentas de visualização de informação analítica directamente nos seus computadores de bolso.

6. Referências

Beyer K. e R. Ramakrishanan Bottom-up computation of sparse and iceberg cubes, in SIGMOD, 1999.

Codd E.F, S.B. Codd e C.T. Salley. Providing OLAP (OnLine Analytical Processing) to User-Analysts: An IT Mandate. Technical Report, E.F. Codd and Associates, 1993.

Ester M., J. Kohlhammer, e H.-P. Kriegel. The DC-Tree: A Fully Dynamic Index Structure for Data Warehouses. In Proc. of ICDE, páginas: 379–388, San Diego, California, 2000.

Gray J. e S. Chaudhuri, et al. Data Cube: A Relational Aggregation Operator Generalizing Group-by, Cross-tables and Sub-Totals, in Data Mining and Knowledge Discovery, Vol.1, No.1, páginas: 29-53, 1997.

Gupta H. e I. Mumick. Selection of views to materialize in a data warehouse. In IEEE Transcations on Knowledge and Data Engineering, vol. 17, páginas 24-43, Janeiro 2005.

Han J., J. Pei, G. Dong, e K. Wang. Eficient computation of iceberg cubes with complex measures. In SIGMOD'01. Harinarayan, V., Rajaraman, A., e Ullman, J. Implementing Data Cube Efficiently. In Proc. Of the 1996 ACM-SIGMOD Conference, 1996.

Hurtado C., A. O. Mendelzon e A. Vaisman. Updating OLAP dimensions. In DOLAP'99, 1999.

Johnson T. e D. Shasha. Some Approaches to Index Design for Cube Forests. Data Engineering Bulletin, 20(1), March 1997.

Kimball, R. L. Reeves et al. The Data Warehouse Lifecycle ToolKit – Expert Methods for Designing, Developing, and Deploying Data Warehouses. Jonh Wiley & Sons, 1998.

Mueller J. P.. Visual C++ 6.0 - From the ground up – The accelerated Track for Professional Programmers, 2nd ed., McGraw-Hill, 1998.

Mumick I. D. Quass e B. Mumick. Maintenance of Data Cubes and Summary Tables in a Warehouse, In Proc. ACM SIGMOD, Tucson, páginas: 100-111, 1997.

Roussopoulos N., Y. Kotidis, e M. Roussopoulos. Cubetree: Organization of and Bulk Incremental Updates on the Data Cube. In SIGMOD 1997, Tucson.

Sismanis Y., A. Deligiannakis, N. Roussopoulos e Y. Kotidis. Dwarf: Shrinking the PetaCube, In Proc. Of ACM SIGMOD, páginas: 464-475, Madison, Wisconsin, USA, 2002.

Wang W. J. Feng, H. Lu and J. Yu. "Condensed Cube: An Effective Approach to Reducing Data Cube Size". In Proc. Of ICDE, páginas. 155-165, San Jose, California, USA, 2002.

Zhao Y., P.M.Deshpande and J.F. Naughton. An array-based algorithm for simultaneous multidimensional aggregates. In SIGMOD'97, 1997.

Zipf G. K.. Human behaviour and the principle of least effort. Addison-Wesley, MA, 1949. Xin D. J. Han, X. Li e B. Wah.Star-cubing: computing iceberg cubes by top-down and bottomup integrations. In VLDB, 2003.