

# Identificação de Hierarquias Incompletas em Estruturas Multidimensionais de Dados

Bruno Oliveira, Centro de I&D ALGORITMI Universidade do Minho, Portugal,  
Id4103@alunos.uminho.pt

Orlando Brito, Centro de I&D ALGORITMI Universidade do Minho, Portugal,  
obelo@di.uminho.pt

## Resumo

Recorrendo a operações de roll-up e drill-down, as ferramentas OLAP (On-line Analytical Processing) permitem uma análise dinâmica de grandes volumes de dados históricos armazenados em Sistemas de Data Warehousing. De forma a realizar as várias operações de análise de dados, os sistemas OLAP necessitam de uma definição correta das hierarquias das suas dimensões de análise, muitas vezes limitadas em termos de estrutura, principalmente quando comparadas com as hierarquias do mundo real. Além disso, hoje, ainda não existe um consenso no domínio da modelação conceptual em termos de hierarquias, para que se possa desenvolver uma especificação formal adequada, bem como uma notação gráfica que facilite a sua integração nas próprias ferramentas OLAP. Neste artigo apresentamos um processo para a identificação de hierarquias incompletas de dimensões, descrevendo a forma como são identificadas e relevando a importância da sua descoberta. Para sustentar esse processo, começamos por realizar uma análise comparativa de três abordagens para a modelação conceptual de hierarquias OLAP, possibilitando assim um melhor conhecimento para o problema em questão.

**Palavras-chave:** Data Warehousing; Sistemas de Processamento Analítico; OLAP; Hierarquias OLAP; Análise e Modelação Conceptual de Dados; Identificação de Hierarquias incompletas.

## 1. INTRODUÇÃO

Com o passar dos anos observou-se que o facto de uma organização possuir grandes quantidades de dados por si só não é suficiente para suportar os seus processos de tomada de decisão. É necessário, também, que os seus dados estejam corretos e devidamente enquadrados e alinhados com os objectivos da organização, de modo a que sejam úteis para os gestores que diariamente tomam decisões no seio de uma organização. Foi com base em enquadramentos como este que surgiu o conceito de Business Intelligence (BI), um termo que engloba os processos de obtenção, armazenamento, análise e partilha de dados, com o objectivo de utilizar a informação obtida no sentido de facilitar o processo de tomada de decisão dentro da própria organização.

O principal objectivo de um sistema de BI passa por suportar vários tipos de decisão dentro de uma organização, uma vez que, ao longo dos anos, a importância da análise de dados tem aumentado significativamente, devido, essencialmente, às vantagens competitivas que possibilita no processo de tomada de decisão. De forma a suportar o armazenamento de grandes volumes de informação, organizados por assuntos, e com carácter temporal surgiu o conceito de data warehouse (DW), um

repositório que proporciona a integração de dados históricos organizados especificamente para o processamento analítico dos dados. Segundo Inmon (1993), um DW é um tipo de base de dados que gere uma grande quantidade de dados, que deixam de estar orientados aos maiores “sujeitos” ou elementos de maior impacto numa empresa, isto é, deixam de estar orientados à gestão de uma determinada área (gestão de stocks, encomendas de clientes, ações de marketing, etc.) para estarem orientados a perspectivas de análise, como clientes, produtos ou promoções, refletindo as necessidades mais vulgares de um sistema para apoio à decisão.

A estrutura de um DW é normalmente representada recorrendo a esquema com configuração em estrela (star-schema) ou esquema em floco de neve (snow-flake-schema), baseados em vistas multidimensionais dos dados, envolvendo usualmente uma tabela de factos (ou mais), tabelas de dimensão e hierarquias de agregação. As tabelas de factos representam o core da informação de análise, disponibilizando medidas que formam os elementos de análise. Por sua vez, as tabelas de dimensão contêm atributos (níveis de dimensão) que permitem explorar as medidas sobre diferentes perspectivas. Estes atributos, por si, podem formar hierarquias que indicam caminhos para a agregação de dados, possibilitando ao utilizador a visualização de dados mais ou menos detalhados conforme se vai deslocando na própria hierarquia.

As ferramentas para sistemas de processamento analítico de dados - On-Line Analytical Processing (OLAP) - permitem aos utilizadores responsáveis pela tomada de decisão manipular dados armazenados num DW. Estas ferramentas utilizam hierarquias que permitem ao utilizador obter uma visão geral dos dados, realizando operações de roll-up, e uma visão mais detalhada dos dados, através de operações de drill-down. No entanto estas ferramentas são um pouco restritivas no tipo de hierarquias disponibilizadas, principalmente se tivermos em consideração hierarquias que aparecem em aplicações utilizadas no nosso quotidiano. Por isso vários trabalhos têm sido apresentados com o objectivo de fornecer uma visão conceptual das hierarquias, nomeadamente (Aballó et al., 2002), (Golfarelli e Rizzi, 2009), (Luján-Mora et al., 2006), (Malinowski e Zimányi, 2006), (Malinowski e Zimányi, 2004) ou (Tryfona e Busborg, 1999), que fornecem inclusive uma categorização das próprias hierarquias (Golfarelli e Rizzi 2009) (Malinowski e Zimányi, 2004) (Tryfona e Busborg, 1999), com o principal objetivo de facilitar a sua implementação nas ferramentas OLAP.

No presente artigo apresentamos uma análise comparativa para a modelação conceptual de hierarquias OLAP. Com base na comparação e categorização de hierarquias apresentadas em (Malinowski e Zimányi 2004), apresentamos para as abordagens apresentadas em (Malinowski e Zimányi 2004), (Aballó et al., 2002) e (Golfarelli e Rizzi, 2009), a respetiva representação da hierarquia, com o principal intuito de analisar qual a notação mais simples e completa para a especificação de hierarquias em ambientes OLAP. Sugerimos também um algoritmo para a identificação de hierarquias incompletas, que afectam negativamente a qualidade dos dados

processados pelas ferramentas OLAP. Nesse sentido, organizámos este artigo da seguinte forma: na secção 2 apresentamos em termos gerais o conceito de hierarquia e a sua forma tradicional de representação; na secção 3 apresentamos algumas das notações mais utilizadas na representação de hierarquias; na secção 4 enunciamos e descrevemos a categorização das hierarquias e a sua correspondente representação; na secção 5 apresentamos, em traços gerais, um algoritmo para a identificação de hierarquias incompletas; e por fim, na secção 6, apresentamos as conclusões relativas ao trabalho apresentado.

## **2. HIERARQUIA OLAP**

Uma hierarquia pode ser considerada como um conjunto de relações binárias entre vários níveis de uma dimensão de análise (Malinowski e Zimányi, 2004). Geralmente as hierarquias são representadas recorrendo a árvores de nodos com vários níveis que descendem de um nodo superior designado usualmente por root. Cada nível da árvore é designado por nível dimensional ou da dimensão e representa um determinado nível de detalhe na cadeia de agregação associada com a hierarquia. A sequência de todos estes níveis é referida como caminhos de agregação e o seu tamanho é determinado pelo número de níveis de agregação que contém. Os nodos da árvore de uma hierarquia que contenham filhos são designados por nodos pai. Um nodo pode ser simultaneamente um nodo pai e um nodo filho dependendo do nível da árvore em que está posicionado. Cada instanciação de um nível é considerada um membro, possuindo uma Cardinalidade que representa o número máximo e mínimo de membros num determinado nível que pode ser relacionado com outro membro de outro nível. O nodo root (ou All) representa a vista mais generalizada dos dados, ou seja, algo similar à aplicação de uma função de agregação sem a definição de grupos. Em (Aballó et al., 2002), o nodo root de uma árvore de uma hierarquia é identificado como All, enquanto que, por exemplo, em (Malinowski e Zimányi, 2004) os autores consideram que a sua representação no modelo conceptual pode ser considerada ambígua para os decisores. Vulgarmente, as hierarquias são representadas utilizando uma única tabela que representa todos os dados relevantes (flat-table) ou através de uma estrutura normalizada designada como floco de neve – snowflake (Kimball e Ross, 2002).

## **3. REPRESENTAÇÃO CONCEPTUAL DE HIERARQUIAS**

Com base na modelação conceptual de hierarquias OLAP, apresentamos agora uma análise comparativa de três metodologias para a modelação conceptual de hierarquias: 1) uma abordagem baseada no modelo E-R (Entidade-Relacionamento) apresentada em (Malinowski e Zimányi, 2004; 2006); 2) uma notação específica apresentada por (Golfarelli e Rizzi, 2009); e 3) uma abordagem baseada na notação UML (Unified Modeling Language) apresentada em (Aballó et al., 2002).

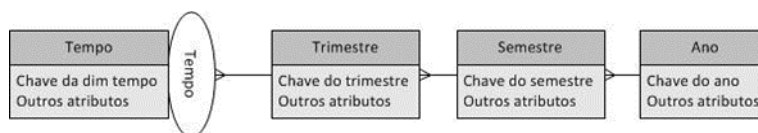


Figura 1 – Exemplo de uma hierarquia simétrica na notação de Malinowski e Zimányi (2004).

Com base na análise comparativa apresentada por (Malinowski e Zimányi, 2004), estas três abordagens provaram ser as mais completas de forma a representarem os vários tipos de hierarquias, categorizadas no mesmo trabalho através da notação MultiDimER (Figura 1). Esta categorização permite a representação dos vários níveis de dimensão associados com uma dada dimensão. Ainda de acordo com a mesma notação, as hierarquias podem expressar diferentes estruturas com base em determinados critérios de análise. Assim é apresentado um artefacto específico que, associado com a dimensão ‘Tempo’ (Figura 1), permite especificar um ou mais critérios de análise das hierarquias associadas. Os seus autores defendem a sua abordagem com base na necessidade de analisar uma dimensão tendo por base vários critérios de análise, de forma a explorar o conceito de partilha de níveis entre hierarquias.

Em (Golfarelli e Rizzi, 1998) foi apresentado o Dimensional Fact Model (DFM), um modelo conceptual essencialmente gráfico criado especificamente para a modelação de data marts. Os seus autores classificam o modelo como simples, expressivo e apropriado para a comunicação com vários tipos de utilizadores. A modelação tem por base uma tabela de factos que, como é vulgar, contém um certo conjunto de medidas e possui um número de arcos correspondente ao número de dimensões associadas à tabela de factos (Figura 2). O primeiro círculo do arco (root) representa a dimensão de análise com os atributos dimensionais que descrevem a dimensão, tendo a si associados um conjunto de atributos descritivos utilizados na sua descrição. As hierarquias são representadas por uma árvore cujos nodos são os atributos dimensionais e os arcos (ou segmentos de reta) representam associações muitos-para-um entre os atributos dimensionais.

Por sua vez, em (Aballó et al., 2002) apresenta-se uma abordagem orientada a objetos para a modelação multidimensional, recorrendo à linguagem UML (Figura 3) de forma a evitar a introdução de novos conceitos e aproveitar a familiarização dos utilizadores com a linguagem, facilitando, em consequência, a sua implementação. As hierarquias são representadas através de um grafo, no qual cada vértice corresponde a um nível, que se encontra conectado por associações que refletem a decomposição de um nível.

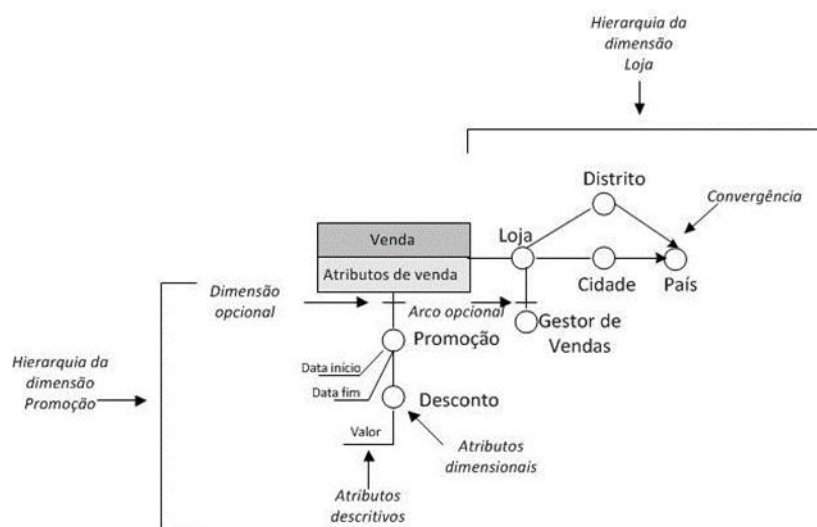


Figura 2 – Exemplo da notação inicialmente proposta em (Golfarelli e Rizzi, 1998) e aumentada em (Golfarelli e Rizzi, 2009).

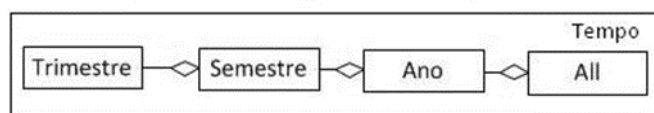


Figura 3 – Alguns elementos da notação apresentada em (Aballó et al., 2002).

#### 4. A DEFINIÇÃO DE HIERARQUIAS

Para introduzir os vários tipos de hierarquias neste artigo, vamos utilizar a categorização proposta em (Malinowski e Zimányi, 2004). Associado com cada tipo de hierarquia escolhido, apresentaremos também a sua conversão, para cada uma das notações utilizadas na comparação que queremos realizar. Em casos específicos, os autores do trabalho referido atribuíram nomes diferentes a hierarquias equivalentes, fazendo essa distinção ao longo da descrição. Se para uma determinada abordagem não for possível a representação de um tipo de hierarquia em particular, essa abordagem será omitida para a hierarquia em questão.

##### 4.1. Hierarquias simples simétricas

As hierarquias simples são hierarquias que utilizam apenas um critério de análise e na qual a relação entre os seus membros pode ser representada recorrendo a uma árvore com vários nodos. As hierarquias simples podem ser categorizadas em hierarquias simétricas e assimétricas. Nas primeiras existe apenas um caminho entre os níveis da hierarquia onde todos níveis da árvore são obrigatórios, isto é, todos os nodos pai têm de ter pelo menos um nodo filho e um nodo filho não pode pertencer a mais do que um nodo pai. Neste tipo de hierarquias cada nível possui o mesmo número de elementos. Na Figura 1 apresentou-se uma hierarquia simétrica, recorrendo à representação

apresentada em (Malinowski e Zimányi, 2004), na qual é possível observar que o relacionamento entre os níveis da hierarquia é obrigatório (1:N) e que ‘Tempo’ é o critério de análise de uma possível dimensão temporal, como um calendário porexemplo. Recorrendo à notação apresentada em (Golfarelli e Rizzi, 1998; 2009), na Figura 4 apresentamos uma tabela de factos contendo uma árvore de nodos representando uma hierarquia simétrica para o caso da dimensão ‘Tempo’.

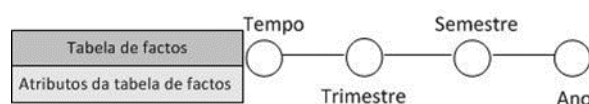


Figura 4 – Exemplo de uma hierarquia simétrica na notação proposta em (Golfarelli e Rizzi, 1998; 2009).

Com base na notação UML, na Figura 3 está apresentada uma hierarquia simétrica recorrendo à notação proposta por (Aballó et al., 2002). Os autores deste trabalho não fazem distinção entre hierarquias simétricas e hierarquias assimétricas, mesmo quando outros autores identificaram que as hierarquias assimétricas não podem ser representadas nesta notação (Malinowski e Zimányi, 2004). No nosso entendimento, embora a representação simétrica corresponda à Figura 3, achamos que a inclusão da associação de agregação para a definição de uma hierarquia simétrica poderá levar a erros de interpretação, uma vez que uma agregação na linguagem UML estabelece uma relação de não obrigatoriedade entre os elementos envolvidos na associação. Acreditamos, assim, que uma hierarquia simétrica deveria ser representada através de uma relação de composição que estabelece uma relação todo-parte, através de uma relação de obrigatoriedade entre os vários níveis da hierarquia. De salientar que para a dimensão ‘Tempo’, para além dos nodos: ‘Ano’, ‘Semestre’ e ‘Trimestre’, é também representado o nodo ‘All’ que, como já referimos, representa a agregação de todos os registos da hierarquia.

#### 4.2. Hierarquias simples assimétricas

Enquanto que as hierarquias simétricas implicam que cada nodo pai possua obrigatoriamente um nodo filho, as hierarquias assimétricas não possuem esse tipo de obrigatoriedade, provocando uma árvore não balanceada, i.e, alguns ramos da árvore da hierarquia podem possuir um número de nodos diferente, uma vez que os nodos pai podem não ter nenhum nodo filho associado. Na notação proposta por Malinowski e Zimányi (2004), as hierarquias assimétricas são representadas através do sinal de não obrigatoriedade, como acontece no modelo Entidade-Relacionamento. Por exemplo, ‘Ano’ tem obrigatoriamente pelo menos um mês, no entanto um mês pode não possuir nenhum feriado associado. Esta situação origina uma hierarquia assimétrica. Recorrendo a arcos opcionais, Golfarelli e Rizzi (2009) apresentam um conceito muito similar às hierarquias assimétricas definidas pelos autores anteriores. Através de um segmento de recta vertical (similar à notação do diagrama

Entidade-Relação) é definido o arco de ligação cuja multiplicidade mínima definida, para o atributo para o qual o arco se dirige, é zero (Figura 2).

#### 4.3. Hierarquias não-abrangidas (non-covering)

Alguns problemas relacionados com as cardinalidades entre os níveis de uma dimensão são tipicamente incluídos na categoria das hierarquias não-abrangidas (non-covering) (Pedersen e Jensen, 1999). Uma dessas hierarquias surge quando existem diferentes estruturas de hierarquia que convergem num nível específico. Por exemplo, com a hierarquia ‘Região -> Distrito -> Cidade’ é possível gerar diferentes caminhos para a cidade do “Porto”. Por exemplo, se considerarmos a designação ‘Porto’ como um distrito estaremos perante uma hierarquia assimétrica, na qual nem todos os nodos pai têm o mesmo número de filhos. No entanto se considerarmos ‘Porto’ como uma cidade estamos perante uma hierarquia não-abrangida, uma vez que ‘Porto’ pode ser encarado tanto como um distrito como uma cidade.

Em (Malinowski e Zimányi, 2004) é apresentada uma proposta de modelação deste tipo de hierarquias utilizando um operador próprio de exclusividade (Figura 5). Este operador indica que os caminhos entre os níveis são exclusivos, i.e, apenas um caminho é válido para uma determinada instância da hierarquia.

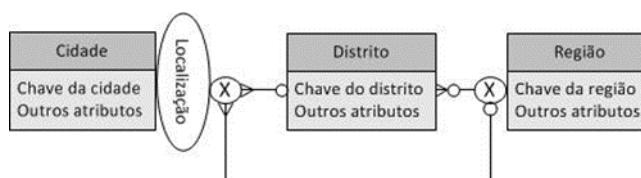


Figura 5 – Hierarquia não-abrangida representada na notação proposta em (Malinowski e Zimányi, 2004).

Quanto a Golfarelli e Rizzi (2009), estes referem-se a este tipo de hierarquias como hierarquias incompletas, nas quais um ou mais níveis de agregação não estão presentes em algumas das suas instâncias. Este tipo de hierarquia é distinguido graficamente recorrendo-se a um segmento de recta horizontal sobre o círculo que representa o nível na hierarquia (Figura 6).

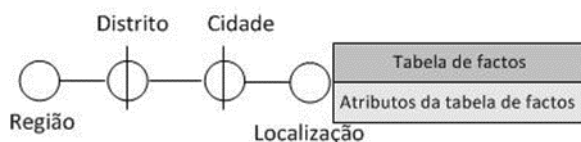


Figura 6 – Hierarquia incompleta representada na notação proposta por (Golfarelli & Rizzi 2009).

É importante mencionar a diferença entre as hierarquias incompletas e as hierarquias com arcos opcionais (Figura 2). Nas primeiras, um ou mais valores de atributos para as instâncias da hierarquia

para todas as posições da hierarquia (incluindo o início e o fim) podem não ser representados, enquanto que nas hierarquias com arcos opcionais apenas os níveis descendentes de uma dada posição da hierarquia é que não são representados. Se apenas o atributo que representa o maior detalhe da hierarquia (por exemplo ‘Cidade’ na Figura 6) não for representado, então as duas abordagens de modelação são equivalentes.

#### 4.4. Hierarquias non-strict

As hierarquias simples baseiam-se em relacionamentos 1:N, nas quais um nodo filho está associado a um único pai e um nodo pai pode possuir vários nodos filho. No entanto, os relacionamentos do tipo M:N são bastante frequentes entre os nodos de uma hierarquia. Uma hierarquia é considerada strict se todos os relacionamentos existentes são do tipo 1:N e non-strict no caso de possuírem pelo menos um relacionamento de M:N. Por isso uma hierarquia simétrica ou assimétrica pode ser considerada como uma hierarquia strict ou non-strict, respetivamente. Segundo a modelação apresentada em (Malinowski e Zimányi, 2004) as hierarquias non-strict são representadas por relacionamentos M:N - por exemplo, um mês possui uma ou mais semanas e uma semana pode pertencer a mais do que um mês.

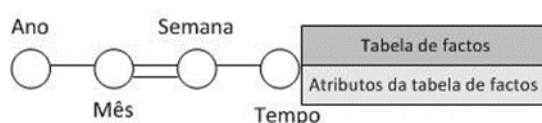


Figura 7 – Hierarquia de múltiplos arcos apresentada em (Golfarelli e Rizzi, 2009).

Golfarelli e Rizzi (2009) referem-se a este tipo de hierarquias como hierarquias de múltiplos arcos. De forma a representar este tipo de hierarquia estes autores sugerem a inclusão de um arco múltiplo, de forma a que um único valor de um nodo A corresponda a vários valores de um nodo B e vice-versa (Figura 7). Por sua vez, em (Aballó et al., 2002), defende-se que os relacionamentos M:N devem ser representados nas dimensões, uma vez que influenciam as medidas através da agregação desse tipo de relacionamento, podendo este ser representado recorrendo-se a um nível da hierarquia associativo, semelhante a uma classe associativa na linguagem UML.

#### 4.5. Hierarquias múltiplas

As hierarquias múltiplas são representadas pela partilha de níveis entre hierarquias simples. Este tipo de hierarquias é acolhido por um grafo, uma vez que os nodos filhos podem estar associados a um ou mais nodos pai, não necessariamente pertencentes ao mesmo nível da hierarquia. As hierarquias simples constituintes de uma hierarquia múltipla partilham o mesmo critério de análise. Os nodos das hierarquias podem formar pontos de partilha de junção (joining level) ou então pontos de partilha de separação (splitting level). Nas hierarquias múltiplas não é semanticamente correto



percorrer simultaneamente as diferentes hierarquias que compõem a hierarquia, devendo ser escolhida uma alternativa para o critério de análise a aplicar (Malinowski e Zimányi, 2006). Este tipo de hierarquias é categorizado por (Malinowski e Zimányi, 2004) como hierarquias múltiplas inclusivas. Nas hierarquias múltiplas alternativas deverá ser selecionado um caminho (hierarquia) para análise. Recorrendo-se à notação de (Malinowski e Zimányi, 2004) na Figura 8 está apresentada uma hierarquia múltipla alternativa em que a agregação pode ser realizada por ‘Ano-Mês’ ou por ‘Ano-Trimestre’.

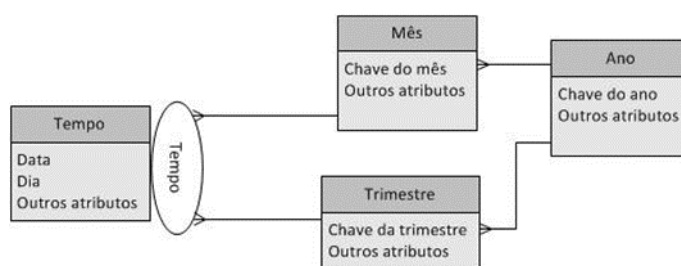


Figura 8 – Hierarquia múltipla alternativa representada na notação proposta em (Malinowski e Zimányi, 2004).

Em (Golfarelli e Rizzi, 2009), as hierarquias múltiplas são definidas como hierarquias convergentes, uma vez que cada hierarquia representa um caminho distinto que converge no mesmo atributo de dimensão. Adicionalmente é necessário identificar os arcos que convergem, representados através de uma seta direcionada para o atributo de dimensão (Figura 2). Adicionalmente, nesse trabalho os autores também apresentam o conceito de hierarquias convergentes redundantes, que ocorrem sem que existam caminhos alternativos que converjam para um atributo dimensional que não incluía atributos intermédios. Como tal, esse caminho redundante deverá ser eliminado, o que originará uma nova representação da hierarquia (Figura 9).



Figura 9 – Hierarquia Convergente Redundante (à esquerda) e a sua correta representação (à direita).

Usando a notação YAM (Aballó et al., 2002), na Figura 10 apresentamos uma hierarquia múltipla recorrendo a elementos de agregação da linguagem UML.

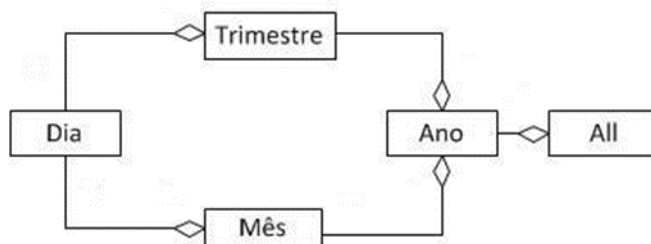
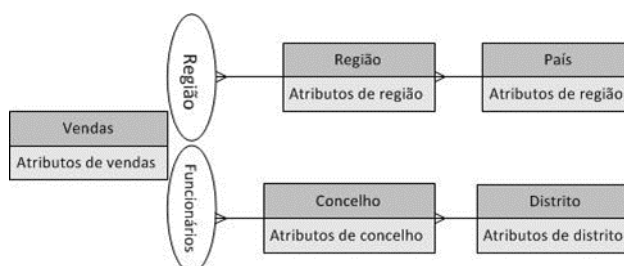


Figura 10 – Hierarquia múltipla alternativa representada na notação proposta em (Aballó et al., 2002) .

#### 4.6. Hierarquias paralelas

As hierarquias paralelas surgem quando uma dimensão está associada a várias hierarquias que, por sua vez, possuem vários critérios de análise. As (sub-)hierarquias que compõem uma hierarquia paralela podem ser compostas pelos tipos de hierarquias previamente apresentados. Por sua vez, as hierarquias paralelas podem ser dependentes ou independentes. As hierarquias do primeiro tipo não partilham qualquer nível e dimensão, isto é, não há partilha de níveis entre as hierarquias estabelecidas.

a)



b)

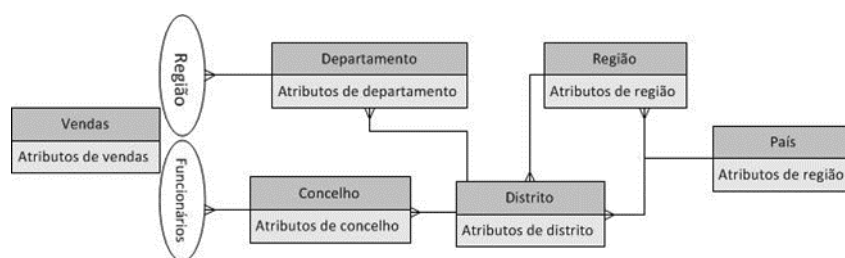


Figura 11 – a) Hierarquia paralela independente e b) Hierarquia paralela dependente, representadas na notação proposta em (Malinowski e Zimányi, 2004).

Na Figura 11a está apresentado um exemplo de uma hierarquia paralela independente, com as hierarquias ‘Região – País’ e ‘Concelho – Distrito’ a não partilharem nenhum nível entre si.

Além disso estas hierarquias possuem critérios de análise diferenciados sobre a tabela de factos, sendo essa característica essencial para que esta possa ser considerada como uma hierarquia paralela.

As hierarquias paralelas dependentes possuem (sub-)hierarquias que partilham os mesmos níveis de hierarquia. Na Figura 11b, segundo a notação de Malinowski e Zimányi (2004), ‘Região’ e ‘Distrito’ partilham o nível ‘País’ e ‘Departamento’, enquanto que ‘Concelho’ partilha o nível ‘Distrito’, estabelecendo uma relação de dependência entre as hierarquias. Golfarelli e Rizzi (2009) apresentam este tipo de hierarquias como hierarquias partilhadas. Nestas hierarquias os nodos são representados por um duplo círculo, estando implícito que todos os nodos descendentes de um atributo partilhado são também eles partilhados (Figura 12). Caso contrário, as hierarquias devem ser representadas separadamente (Figura 12). De realçar que nesta notação os critérios de análise de cada hierarquia não estão representados, tornando um pouco difícil a sua distinção em relação às hierarquias múltiplas.

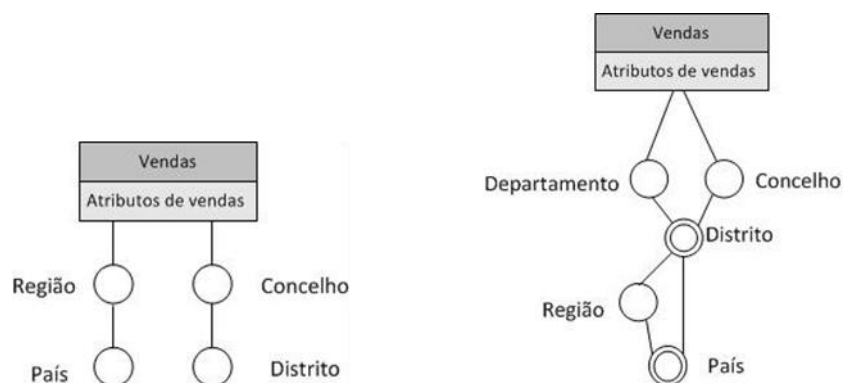


Figura 12 – Uma hierarquia paralela (à esquerda) e uma hierarquia partilhada (à direita) representadas na notação proposta em (Golfarelli e Rizzi, 2009).

Embora não esteja claramente definido em (Aballó et al., 2002), apresentamos também aqui uma possível representação de uma Hierarquia Paralela Independente e uma hierarquia paralela dependente (Figura 13) recorrendo à notação YAM para a sua representação.

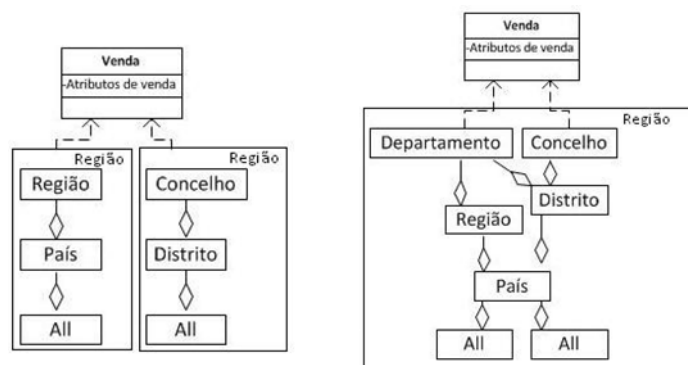


Figura 13 – Uma hierarquia paralela independente (à esquerda) e uma hierarquia dependente (à direita) representadas na notação proposta em (Aballó et al., 2002).

#### **4.7. Outras hierarquias**

As notações analisadas incluem no entanto representações de hierarquias que a) não são contempladas em todas as abordagens, ou que b) não são consideradas pelas ferramentas OLAP.

Um tipo de hierarquias não apresentadas em (Malinowski e Zimányi, 2004) são as hierarquias recursivas (ou não balanceadas) que são apresentadas em (Golfarelli e Rizzi, 2009) e representadas em (Aballó et al., 2002). Ao contrário das hierarquias incompletas apresentadas por Golfarelli e Rizzi (2009), as relações pai-filho das hierarquias recursivas são consistentes, embora possam ter tamanhos diferentes. Consideremos o seguinte exemplo: uma empresa que se dedica a organizar passeios de avião possui funcionários que são responsáveis pela mecânica dos aviões e funcionários que são pilotos de aviões. No entanto, os pilotos de aviões podem também ser mecânicos e vice-versa. Ao considerarmos este exemplo, vemos que não é possível distinguir os vários níveis de agregação.

Por vezes, hierarquias que existem em aplicações não são implementadas pelos sistemas comerciais OLAP. Um desses exemplos sucede quando uma dada dimensão inclui alguns subtipos, com a sua própria hierarquia, que podem ser representados através de uma relação de generalização ou especialização (Aballó et al., 2002; Luján-Mora et al., 2006; Malinowski e Zimányi, 2004). Em (Malinowski e Zimányi, 2004) este tipo de hierarquias é definido como uma hierarquia generalizada, recorrendo-se para isso à representação de níveis de hierarquia partilhadas que possuem a mesma granularidade de agregação. A notação proposta baseia-se na definição das hierarquias não abrangidas, já apresentadas anteriormente com a definição do operador de exclusividade na seleção do caminho de agregação a seguir, uma vez que a hierarquia do tipo não abrangida é um caso especial da hierarquia generalizada. No entanto, a nível conceptual a caracterização das hierarquias generalizadas difere da sua implementação lógica (Bauer et al., 2000; Cabibbo e Torlone, 2000), em que, por exemplo, se utiliza uma representação dos subtipos em hierarquias distintas, por forma a ser possível implementá-las em sistemas comerciais.

### **5. IDENTIFICAÇÃO DE HIERARQUIAS INCOMPLETAS**

Adicionalmente ao trabalho de comparação de abordagens para a classificação de hierarquias, nesta secção apresentamos um algoritmo que desenvolvemos para a identificação de hierarquias incoerentes para o processamento de um cubo de dados. Na realidade, pretendemos, somente, relevar a importância da identificação e definição correta de uma hierarquia, uma vez que a sua má definição pode afetar, seriamente, qualquer processo de tomada de decisão que assente nas estruturas multidimensionais nas quais tais hierarquias estejam definidas.

A implementação de um cubo de dados é uma das tarefas mais complexas em OLAP, uma vez que envolve a computação e o armazenamento do resultado da agregação de cláusulas Group By para

todas as combinações de todos os atributos de dimensão constituintes de uma hierarquia. As cláusulas de agrupamento são processadas para que o cubo possa ser construído, podendo ser representadas recorrendo a um grafo, que na terminologia OLAP se reconhece como lattice (Harinarayan et al., 1996). Nesta estrutura, cada nodo representa uma query de agregação, conectado por uma associação simples com cada nodo cuja query de agregação possa ser agrupada como parte integrante de um agrupamento mais generalizado. O algoritmo 2D (Gray et al., 1996) foi o primeiro algoritmo apresentado para a computação de um cubo, baseando-se, essencialmente, na criação de resultados parciais, isto é na criação de cláusulas Group By que depois são sujeitas a uma operação de união para a formação final do cubo de dados.

Posteriormente, em (Gray et al. 1996) sugeriu-se o sinalizador “ALL” para representar a agregação total de um determinado nível da hierarquia. Desta forma, a aplicação da operação da união das queries iria dar origem a um cubo de dados. No entanto se considerarmos, por exemplo, a hierarquia ‘País -> Região -> Cidade’, e porque se trata de uma hierarquia assimétrica o cubo de dados é povoado com vários valores NULL. Com base no sinalizador “ALL” e na existência de valores nulos num registo, é possível especificar um mecanismo simples para a identificação hierarquias incompletas. Este processo é importante uma vez que este tipo de hierarquia pode originar agregações de dados erradas, o que não é nada conveniente nem correto em ambientes de processamento analítico de dados.

```


Função IDENTIFICA_HIERARQUIAS (nodos)
  nível <- 1;
  para (i <- 0; i < tamanho(nodos)-1; i++)
    grupos <- gera_grupos (nodos[i]);
    resultado <- verdadeiro;
    para (j <- 0; j < num_grupos(grupos[j]) && resultado=verdadeiro);j++)
      resultado = validaNodo(grupos[j], nível+1,nodos);
    fimpara
  fimpara

Função VALIDANODO(nodo,nível,nodos)
  resultado <- verdadeiro;
  para (j <- nível; j <= tamanho(nodos);j++)
    grupos <- gera_grupos (nodos[j]);
    para (i <- 0; i < num_grupos(grupos) && resultado=verdadeiro;i++)
      se agrega(nodo,grupos[i]) // não existem nulos após a agregação
        resultado <- validaNodo(agrega(nodo,grupos[i]), nível+1);
    Escreve (agrega(nodo,grupos[i]))

```

Figura 14 – Algoritmo para a identificação de hierarquias incompletas.

Cubo de dados			
Região	Distrito	Cidade	Vendas
Norte	Porto	Guimarães	45
Norte	Porto	Fafe	23
Norte	Porto	ALL	468
Norte	Braga	NULL	0
Norte	ALL	ALL	1090
Centro	Aveiro	NULL	13
Centro	Leiria	NULL	190
...	...	...	...
Centro	Coimbra	ALL	250
Sul	Lisboa	Lisboa	394
Sul	Lisboa	Sintra	593
Sul	Lisboa	ALL	1980
...	...	...	...



Cubo de dados			
Região	Distrito	Cidade	Vendas
Norte	Porto	Guimarães	45
Norte	Porto	Fafe	23
Norte	Porto	ALL	468
Norte	ALL	ALL	1090
Centro	Coimbra	ALL	250
Sul	Lisboa	Lisboa	394
Sul	Lisboa	Sintra	593
Sul	Lisboa	ALL	1980
...	...	...	...

Figura 15 – Exemplo de aplicação do algoritmo apresentado na Figura 14.

O algoritmo apresentado na Figura 14 percorre cada um dos níveis da hierarquia e, para o primeiro nível da lattice, particiona cada um desses níveis em grupos, basicamente aplicando uma função Group By, originando para o atributo A, por exemplo, vários subconjuntos <A1, A2, A3,...> que representam um grupo. De seguida invoca a função validaNodo, que recebe a agregação resultante de cada grupo do primeiro nível de hierarquia. Este procedimento percorre recursivamente cada nível da hierarquia e agrega cada grupo de cada atributo de dimensão de forma a produzir todos os grupos de agregação para essa dimensão. Isto é, considerando a hierarquia A->B->C, inicialmente o atributo A é particionado em grupos, <A1, A2, A3,...> e para cada grupo é analisado se este possui agregação com os grupos da dimensão B de forma a produzir um conjunto <a1, b1>, que irá resultar no conjuntos <a1,b1,c1>. Uma vez terminada as combinações nas dimensões é produzido o conjunto <a1,b1,cn> em que n representa um grupo da dimensão C. De seguida a combinação dos níveis é obtida recorrendo às partições <a1, bn> até terminar no primeiro nível de hierarquia, para posteriormente repetir o processo para a agregação da dimensão B. A aplicação do algoritmo apresentado na Figura 14, permite a exclusão dos registos que são parte de uma hierarquia inconsistente, formando um cubo de dados sem a existência de valores NULL – Figura 15.

A existência de valores NULL gera uma hierarquia incoerente, uma vez que determinados nodos pai não possuem nodos filho. Com a identificação dos níveis que quebram a hierarquia, permite-se redesenhar a hierarquia obtendo resultados mais coerentes. A nível lógico, as hierarquias incompletas são tipicamente decompostas em hierarquias mais simples, de forma a que seja possível realizar a sua implementação. Por exemplo, no caso das hierarquias assimétricas, tipicamente este tipo de hierarquias são convertidas em hierarquias simétricas recorrendo ao preenchimento dos níveis em falta através de marcadores próprios.

## **6. CONCLUSÕES**

Com o objectivo de fornecer uma perspectiva conceptual na modelação de hierarquias OLAP, apresentámos neste artigo a comparação entre três modelos, nomeadamente o MultiDimER, o DFM e o YAM. O modelo MultiDimER apresenta uma proposta de modelação baseada no modelo E-R, com a inclusão de artefactos que enriquecem e reduzem ambiguidades na interpretação do modelo. A inclusão de um elemento de notação que representa o critério de análise de uma hierarquia facilita bastante a compreensão permitindo inclusivamente distinguir as hierarquias múltiplas das hierarquias paralelas, através da identificação de critérios de análise diferenciados. Apesar de ser uma notação simples e poderosa, quando necessitamos de representar um maior número de níveis de dimensão, dimensões e atributos das dimensões, esta notação torna-se um pouco extensa em termos gráficos. Quanto à DFM, esta é uma notação bastante mais reduzida na representação da tabela de factos, dimensões, atributos de dimensão e relacionamentos, o que permite uma representação mais elegante, principalmente na modelação de um maior número de dimensões e atributos. Naturalmente que a categorização das hierarquias realizadas nesta abordagem tem em consideração a própria implementação da notação, o que torna a notação gráfica com os termos apresentados por Golfarelli e Rizzi (2009) bastante mais intuitiva. À semelhança da notação YAM, a DFM não permite a definição de critérios de análise de dimensão. No entanto, consideramos que a definição dos níveis de dimensão por si só define o critério de análise. Adicionalmente, na notação DFM as associações são claras quanto à análise de hierarquias múltiplas (convergentes) e hierarquias paralelas dependentes (partilhadas). A notação YAM é a mais antiga das abordagens apresentadas, como tal, sofre de uma indefinição da categorização dos tipos de hierarquia, o que muitas das vezes torna difícil a sua correta representação. Um exemplo disso mesmo é a definição de hierarquias simétricas e assimétricas, cuja distinção não é possível fazer. Complementarmente a este trabalho de comparação, apresentámos também uma proposta para um algoritmo capaz de identificar hierarquias incoerentes para o processamento de um cubo de dados, para que fosse possível evitar a representação de hierarquias incompletas, uma vez que estas podem conduzir à obtenção de dados incorretos, podendo afetar seriamente o processo de tomada de decisão em causa. Este tipo de hierarquias violam o princípio da “perfeição” (Lenz e Shoshani, 1997) e implicam um tratamento especial para que possam ser manipuladas por uma ferramenta OLAP convencional.

## **REFERÊNCIAS**

- Aballó, A., Samos, J. e Saltor, F., 2002. YAM2 (Yet Another Multidimensional Model): An extension of UML. In Proc. of the Int. Database Engineering and Application Symposium, pp.172-181
- Bauer, A., Hümmer, W. e Lehner, W., 2000. An Alternative Relational OLAP Modeling Approach Data Warehousing and Knowledge Discovery. In Y. Kambayashi, M. Mohania, & A. Tjoa, eds. Springer Berlin / Heidelberg, pp. 189-198. Available at: [http://dx.doi.org/10.1007/3-540-44466-1\\_19](http://dx.doi.org/10.1007/3-540-44466-1_19).

- Cabibbo, L. e Torlone, R., 2000. The Design and Development of a Logical System for OLAP. Proceedings of the Second International Conference on Data Warehousing and Knowledge Discovery, pp.1-10.
- Golfarelli, M. e Rizzi, S., 1998. A Methodological Framework for Data Warehouse Design. Proc. of the 1st ACM Int. Workshop on Data Warehousing and OLAP, pp.3-9.
- Golfarelli, M. e Rizzi, S., 2009. Data Warehouse Design: Modern Principles and Methodologies, McGraw-Hill. Available at: <http://books.google.pt/books?id=R7qqNwAACAAJ>.
- Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D., e Venkatrao, M., 1996. Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Total. Proceedings of the Twelfth International Conference on Data Engineering, pp.152-159.
- Harinarayan, V., Rajaraman, A. e Ullman, J.D., 1996. Implementing data cubes efficiently. SIGMOD Rec., 25(2), pp.205-216.
- Inmon, W.H., 1993. Building the data warehouse, John Wiley & Sons. Available at: <http://books.google.com.ni/books?id=bHP1Wc4CdGEC>.
- Kimball, R. e Ross, M., 2002. The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, Wiley. Available at: <http://books.google.pt/books?id=2OCbq8Azdm8C>.
- Lenz, H.-J. e Shoshani, A., 1997. Summarizability in OLAP and Statistical Data Bases. Scientific and Statistical Database Management.
- Luján-Mora, S., Trujillo, J. e Song, I.-Y., 2006. A UML profile for multidimensional modeling in data warehouses. Data & Knowledge Engineering, 59(3), pp.725-769. Available at: <http://www.sciencedirect.com/science/article/pii/S0169023X0500176X>.
- Malinowski, E. e Zimányi, E., 2006. Hierarchies in a multidimensional model: from conceptual modeling to logical representation. Data Knowl. Eng., 59(2), pp.348-377.
- Malinowski, E. e Zimányi, E., 2004. OLAP Hierarchies: A Conceptual Perspective Advanced Information Systems Engineering. In A. Persson & J. Stirna, eds. Springer Berlin / Heidelberg, pp. 19-35.
- Pedersen, T.B. e Jensen, C.S., 1999. Multidimensional data modeling for complex data I. C. Society, ed. Proc. of 15th Int. Conf. on Data Engineering (ICDE).
- Tryfona, N. e Busborg, F., 1999. starER: a conceptual model for data warehouse design. Proceedings of the 2nd ACM international workshop on Data warehousing and OLAP.