



Pontifícia Universidade Católica do Rio Grande do Sul
Faculdade de Informática
Mestrado em Informática



Um Estudo sobre Data Warehouse através de uma Experiência Prática

Trabalho Individual I

por

Kellyne Marques Santos

Profa. Dra. Karin Becker
Orientadora

Porto Alegre, setembro de 1999.

Índice

I	INTRODUÇÃO.....	6
II	CONCEITOS FUNDAMENTAIS.....	7
II.1	SISTEMAS DE APOIO À DECISÃO (SAD).....	8
II.2	SISTEMAS FONTES	10
II.3	DATA WAREHOUSE.....	11
II.3.1	CONCEITO	11
II.3.2	CARACTERÍSTICAS.....	12
II.4	DATA MARTS.....	13
II.5	DATA STAGING.....	13
II.5.1	EXTRAÇÃO	14
II.5.2	TRANSFORMAÇÃO	15
II.5.3	CARGA E INDEXAÇÃO.....	15
II.5.4	VERIFICAÇÃO DA QUALIDADE DO PROCESSO	16
II.6	TIPOS DE DADOS	16
II.6.1	DADOS DO NEGÓCIO.....	16
II.6.2	METADADOS	17
II.7	ARQUITETURA DE DADOS PARA DW	18
II.7.1	ARQUITETURA DE DADOS COM UMA CAMADA	18
II.7.2	ARQUITETURA DE DADOS COM DUAS CAMADAS	19
II.7.3	ARQUITETURA DE DADOS COM TRÊS CAMADAS	19
III	MODELAGEM DIMENSIONAL E RECURSOS ANALÍTICOS.....	21
III.1	MODELAGEM DIMENSIONAL.....	21
III.1.1	CONCEITOS BÁSICOS.....	21
III.1.2	ESQUEMA ESTRELA	24
III.1.3	VARIANTES DO ESQUEMA ESTRELA	25
III.1.4	MODELAGEM DIMENSIONAL × MODELAGEM ENTIDADE/RELACIONAMENTO.....	25
III.2	PROJETO DE ESQUEMAS MULTIDIMENSIONAIS	27
III.2.1	TIPOS DE DIMENSÕES	27
III.2.2	GRANULARIDADE	30
III.2.3	AGREGAÇÃO	30
III.2.4	DIMENSÕES E FATOS CONFORMADOS	31
III.2.5	DATA WAREHOUSE BUS ARCHITECTURE.....	32
III.3	BANCOS DE DADOS MULTIDIMENSIONAIS	33
III.3.1	RECURSOS ANALÍTICOS	34
III.4	EXPLORAÇÃO MULTIDIMENSIONAL.....	35
III.4.1	OPERAÇÕES.....	35
III.4.2	RELATÓRIOS	37
III.4.3	MINERAÇÃO.....	37
III.4.4	CONSULTAS	39

IV	PROCESSO	40
IV.1	PLANEJAMENTO DO PROJETO.....	41
IV.2	PROJETO DOS DADOS	42
IV.3	ARQUITETURA	43
IV.4	IMPLEMENTAÇÃO.....	46
IV.5	CRESCIMENTO E VALIDAÇÃO	49
V	ESTUDO DE CASO	52
V.1	BASE FONTE	52
V.2	MODELAGEM	53
V.3	FERRAMENTAS ESCOLHIDAS.....	56
V.3.1	MS SQL/SERVER 7.0	57
V.3.2	FERRAMENTAS DA COGNOS	60
V.4	CRIAÇÃO DO DW	64
V.4.1	PROCESSO DE STAGING	64
V.4.2	POPULAÇÃO DO DW	64
V.4.3	MANIPULAÇÃO DO DW	67
V.5	DIFICULDADES ENCONTRADAS	68
VI	COMENTÁRIOS FINAIS	69
VII	BIBLIOGRAFIA.....	70
VIII	ANEXOS.....	73
VIII.1	DESCRIÇÃO DAS TABELAS DO SIM	73
VIII.2	CONSULTA GERADA PARA POPULAÇÃO DO DATA STAGING.....	79
VIII.3	DESCRIÇÃO DA TABELA UTILIZADA NO PROCESSO DE DATA STAGING.....	81
VIII.4	DESCRIÇÃO DAS TABELAS DO DW.....	82
VIII.4.1	FATO.....	82
VIII.4.2	DIMENSÕES	83

Tabelas

TABELA 1 – COMPARAÇÃO ENTRE SISTEMAS FONTES E DW	11
TABELA 2 – COMPARAÇÃO ENTRE OLAP E OLTP	34

Figuras

FIGURA 1 - ELEMENTOS BÁSICOS DE UM DW [KIM98A]	7
FIGURA 2 – INFORMAÇÕES DA ORGANIZAÇÃO E O FLUXO DE DECISÃO [SAG91]	8
FIGURA 3 – ARQUITETURA DE UMA CAMADA	18
FIGURA 4 - ARQUITETURA DE DUAS CAMADA	19
FIGURA 5 - ARQUITETURA DE TRÊS CAMADA	20
FIGURA 6 – DIMENSÃO TEMPO	22
FIGURA 7 – UM ESQUEMA ESTRELA GENÉRICO	24
FIGURA 8 – UM NAVEGADOR DE AGREGADOS [KIM98A]	31
FIGURA 9 – DW BUS ARCHITECTURE [KIM98A]	32
FIGURA 10 – ARRAY BIDIMENSIONAL [CAM98]	33
FIGURA 11 – CICLO DE VIDA DO PROJETO DE DW [WEI99]	41
FIGURA 12 – ESQUEMA ESTRELA DO SIM	56
FIGURA 13 - MICROSOFT SQL SERVER 7.0 DATA WAREHOUSING FRAMEWORK [HUR99] ..	58
FIGURA 14 – DTS PACKAGE	59
FIGURA 15 – OLAP SERVICES	60
FIGURA 16 – POWERPLAY TRANSFORMER	62
FIGURA 17 - POWERPLAY	62
FIGURA 18 – RELATÓRIO GERADO NO POWERPLAY	67

Abreviaturas Utilizadas

ANSI	American National Standards Institute
BD	Banco de Dados
BDM	Banco de Dados Multidimensional
CID	Classificação Internacional de Doenças
DW	Data Warehouse
DM	Data Mart
DTS	Data Transformation Service
ER	Entidade Relacionamento
FIPS	Federal Information Processing Standards
KDD	Knowledge Discovery in Databases (Descoberta de conhecimento em BD)
MOLAP	OLAP Multidimensional
ODBC	Open Database Connectivity
OIM	Open Information Model
OLAP	On-line analytical processing (processamento analítico online)
OLE	Object Linking and Embedding
OLTP	Online transaction processing (processamento transacional online)
ROLAP	OLAP Relacional
SAD	Sistema de Apoio à Decisão
SIM	Sistema de Informações de Mortalidade
SNMP	Simple Network Management Protocol
SGBD	Sistema Gerenciador de Banco de Dados
SQL	Structured Query Language

I Introdução

Tradicionalmente, os sistemas de apoio de decisão são usados para obter informação de uma quantidade limitada de dados para apoiar o processo de tomada de decisão. Porém tais sistemas de apoio de decisão têm dificuldade de lidar com múltiplas fontes de dados complexas, que são encontradas tipicamente em grandes organizações. Neste contexto surgiu a tecnologia de **data warehouse**, que conseguiu estruturar os dados com o intuito de simplificar as fases do processo de apoio à decisão.

Através deste trabalho individual, pretende-se investigar a tecnologia de data warehouse e tópicos relacionados que se façam necessários para a compreensão da mesma, especialmente a modelagem de bases dimensionais e seus recursos analíticos. Para solidificar esses conceitos, será realizado um estudo de caso utilizando os dados do Sistema de Informações de Mortalidade do Ministério do Saúde, utilizando uma ferramenta que será escolhida durante esse processo.

O presente trabalho está estruturado da seguinte forma:

Capítulo II – Conceitos Fundamentais: onde são descritos os conceitos de Sistemas de Apoio à Decisão, Sistemas Fontes, Data Warehouse e suas características, Data Mart, Data Staging e seus processos, Tipos de Dados de um Data Warehouse e as Arquiteturas de Dados de um Data Warehouse.

Capítulo III – Modelagem: descreve-se o que é modelagem dimensional e quais são os seus esquemas mais comuns. A seguir é feita uma comparação entre modelagem dimensional e modelagem E-R, e são descritos os fatores que influenciam o projeto de esquemas multidimensionais, o que são BD multidimensionais, que recursos/operações podem ser realizadas num esquema dimensional e que tipos de exploração podem ser feitas no mesmo.

Capítulo IV – Processo: são descritas as fases a serem seguidas no processo de criação de um Data Warehouse.

Capítulo V – Estudo de Caso: neste capítulo, há uma descrição do sistema fonte e das ferramentas utilizadas, como foi modelado o sistema, o processo de criação do DW e as dificuldades encontradas.

II Conceitos Fundamentais

Com o avanço das tecnologias, os usuários-finais passaram a se familiarizar com os computadores e programas, e a desejar uma visão melhor das informações do negócio, pois essas informações poderiam ser usadas como um diferencial diante dos seus concorrentes.

Porém, prover informações para esses usuários não significa apenas dar acesso a todos os dados disponíveis da empresa; é necessário tornar esses dados mais simples de serem vistos pelos usuários e dar o contexto certo para eles. Além disso, os analistas vêem a necessidade de gerenciar de uma maneira melhor os dados da companhia. A tecnologia de data warehouse evoluiu a partir destas necessidades [DEV97].

Neste capítulo serão apresentados alguns conceitos básicos dentro da tecnologia de data warehouse. Alguns desses elementos são mostrados na Figura 1, retirada de [KIM98a]. Os aspectos de modelagem de um data warehouse e acesso aos dados pelos usuários, serão abordados em mais detalhe no capítulo 3.

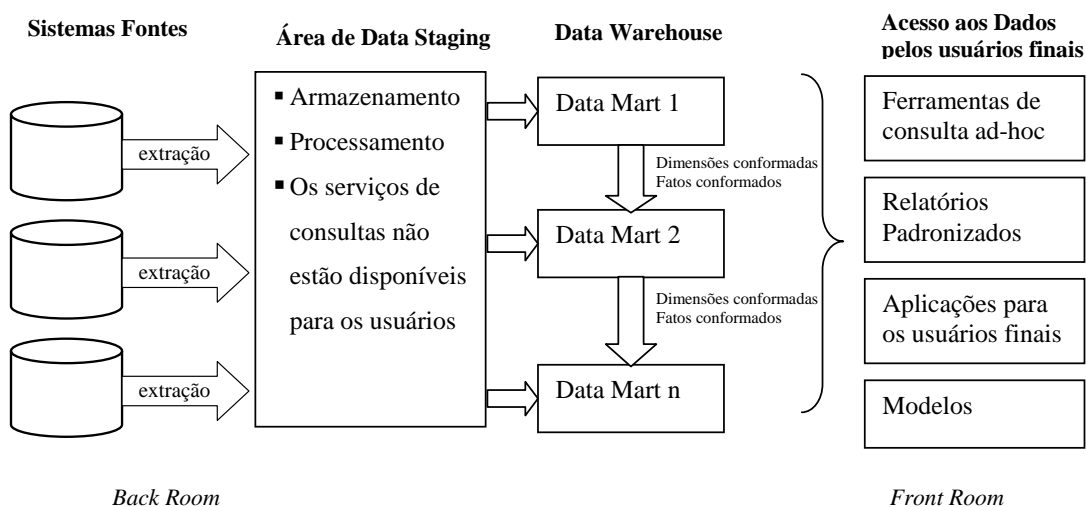


Figura 1 - Elementos básicos de um DW [KIM98a]

II.1 Sistemas de Apoio à Decisão (SAD)

Um sistema de apoio à decisão é um sistema que suporta tecnologicamente e gerencialmente a tomada de decisões [SAG91]. O seu desenvolvimento tem contribuições de várias áreas interdisciplinares, inclusive da ciência da computação.

Segundo Antony (apud [SAG91]), os tipos de decisões podem ser classificadas da seguinte forma:

1. Decisões de Planejamento Estratégico – decisões relacionadas com a escolha de objetivos e políticas, e alocação de recursos.
2. Decisões de Controle Gerencial – decisões para assegurar a eficácia na aquisição e uso dos recursos.
3. Decisões de Controle Operacional - decisões para assegurar a eficácia no desempenho das operações.
4. Decisões de Desempenho Operacional - decisões do dia-a-dia feitas durante a execução das operações.

Na Figura 2, retirada de [SAG91], podemos ver a relação entre esses tipos de decisão.

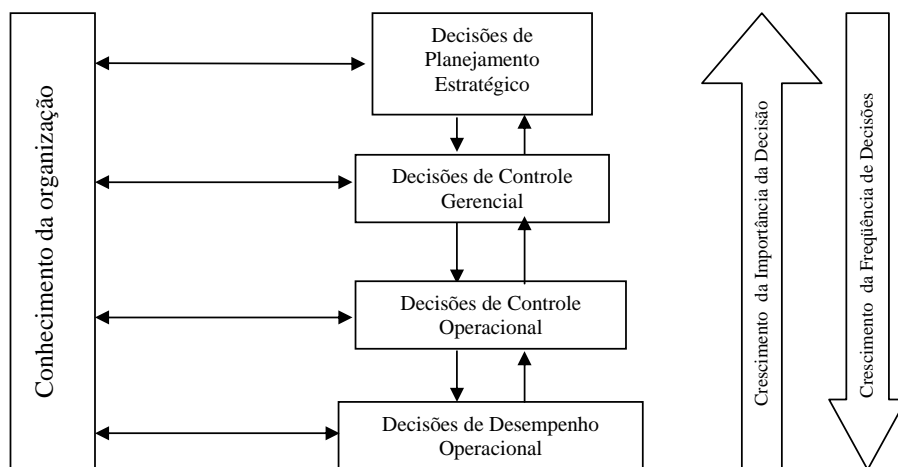


Figura 2 – Informações da Organização e o Fluxo de Decisão [SAG91]

O propósito dos SADs é apoiar os humanos no desempenho de tarefas cognitivas primárias que envolvem decisão, julgamento e escolha, e a sua maior meta é melhorar a

efetividade do conhecimento dos usuários da organização através do uso da tecnologia da informação.

As variáveis que influenciam as informações utilizadas nos SADs são (Keen e Scott-Morton apud [SAG91]):

1. Precisão inerente das informações disponíveis – as situações no controle operacional geralmente lidam com informações que são relativamente certas e precisas, já nas situações de planejamento estratégico as informações são imprecisas, incompletas e incorretas.
2. Nível de detalhe necessário – as decisões operacionais precisam de informações detalhadas, e as decisões estratégicas, agregadas.
3. Horizonte de tempo das informações necessárias - as decisões operacionais são baseadas em informações dentro de um curto período de tempo, enquanto que as decisões estratégicas estão fundamentadas num longo período.
4. Frequência de uso – as decisões estratégicas são feitas com uma frequência menor que as decisões operacionais.
5. Fonte de informações interna ou externa – as decisões operacionais são baseadas em informações que estão disponíveis internamente na organização, enquanto as decisões estratégicas são mais dependentes do contexto, que é obtido fora da organização.
6. Escopo das informações - as decisões operacionais são baseadas num escopo de informações estreito, enquanto as decisões estratégicas, em um escopo de informação mais vasto.
7. Habilidade de quantificar a informação – no planejamento estratégico, as informações são mais qualitativas, e nas decisões operacionais são mais quantitativas.
8. Atualidade da informação - no planejamento estratégico, as informações são mais antigas, e algumas vezes é difícil a obtenção das informações correntes, já no nível operacional as informações correntes são necessárias.

Em sistemas de apoio à decisão, o usuário-final precisa de dados que descrevam bem a organização, que sejam acessíveis, internamente consistentes e sejam organizados de forma a facilitar o seu acesso e carga pelas ferramentas de análise. Assim, bancos de dados que suportam SAD devem ser capazes de recuperar grandes conjuntos de dados históricos e agregados, com um tempo de resposta razoável.

II.2 Sistemas Fontes

Também conhecidos como sistemas legados ou OLTP¹, os sistemas fontes têm como principal função a captura das transações do negócio [KIM98a]. Esses tipos de sistemas mantêm poucos dados históricos, o gerenciamento de relatórios é difícil e as consultas são limitadas.

As maiores prioridades nestes tipos de sistemas são disponibilidade e rapidez. Suas atividades envolvem inserção, consulta, atualização e exclusão de dados em bancos de dados transacionais, e por isso esses tipos de bancos de dados são ditos de leitura-escrita.

Uma breve comparação entre Banco de Dados Transacionais e Data Warehouse pode ser vista na Tabela 1 [INM96] [KIM96]:

Características	Bancos de Dados Transacionais	Data Warehouse
Objetivo	Operações diárias do negócio	Analisar o negócio
Uso	Operacional	Informativo
Tipo de processamento	OLTP	OLAP
Unidade de trabalho	Inclusão, alteração, exclusão	Carga e consulta
Número de usuários	Milhares	Centenas
Tipo de usuário	Operadores	Comunidade gerencial
Interação do usuário	Somente pré-definida	Pré-definida e ad-hoc
Condições dos dados	Dados operacionais	Dados Analíticos
Volume	Megabytes – gigabytes	Gigabytes – terabytes
Histórico	60 a 90 dias	5 a 10 anos
Granularidade	Detalhados	Detalhados e resumidos
Redundância	Não ocorre	Ocorre
Características	BD transacionais	Data Warehouse
Estrutura	Estática	Variável
Manutenção desejada	Mínima	Constante

¹ *Online transaction processing* (ver III.3.1.1 e Tabela 2)

Acesso a registros	Dezenas	Milhares
Atualização	Contínua (tempo real)	Periódica (em <i>batch</i>)
Integridade	Transação	A cada atualização
Número de índices	Poucos/simples	Muitos/complexos
Intenção dos índices	Localizar um registro	Aperfeiçoar consultas

Tabela 1 – Comparação entre Sistemas Fontes e DW

II.3 Data Warehouse

II.3.1 Conceito

A noção de Data Warehouse (DW) varia muito de autor para autor.

Kimball et al. [KIM98a] definem um DW como uma fonte de dados consultável do empreendimento. Ainda segundo [KIM96a], os requisitos para um data warehouse são:

- O DW provê acesso aos dados corporativos ou organizacionais;
- Os dados em um DW são consistentes;
- Os dados em um DW podem ser separados e combinados por todas as possíveis medidas do negócio;
- Um DW não é apenas dados, mas também um conjunto de ferramentas para consulta, análise e visualização de informações.

Devlin [DEV97] afirma que um DW é simplesmente um armazém (*store*) simples, completo e consistente de dados, obtidos de uma variedade de origens. Os seus dados são tornados disponíveis aos usuários finais de uma maneira que eles possam entender e usar no contexto do negócio.

Data Warehouse na visão de [POE98], é um banco de dados analítico que é usado como fundação para os SADs. Ele é desenvolvido para grandes volumes de dados somente-leitura, provendo acesso intuitivo para informações que devem ser usadas para tomada de decisões.

Shouten em [SCH99] descreve um DW como um banco de dados preenchido por fatos derivados e agregados de um banco de dados operacional, com o único propósito de definir políticas.

Segundo [INM96a], um data warehouse (DW) é uma coleção de dados orientados por assunto, integrada, variante no tempo e não volátil, que tem por objetivo dar suporte aos processos de tomada de decisão.

II.3.2 Características

II.3.2.1 Orientado por temas

Diz respeito ao fato do DW armazenar informações sobre temas específicos importantes para o negócio da empresa em contrapartida ao fato das aplicações do ambiente operacional serem funcionais [CAM98]. A implementação de um tema normalmente corresponde a um conjunto de tabelas relacionadas.

II.3.2.2 Integrado

Diz respeito à consistência de nomes, do domínio das variáveis, etc., no sentido de que os dados, potencialmente oriundos de sistemas fontes diferentes, foram transformados para um estado uniforme. Um exemplo bastante utilizado para ilustrar a integração é o elemento de dado sexo, que em algumas aplicações pode ser codificado como F/M, em outras como H/M, ou ainda como 1/0. Os dados neste caso são convertidos para um estado uniforme no momento que são trazidos para o DW. Da mesma forma, elementos de dados com medidas diferentes (e.g.: centímetros, polegadas, metros) serão convertidos para uma mesma medida.

II.3.2.3 Variante no tempo

Diz respeito ao fato do DW referir-se a algum momento específico, isto é, em um DW, para cada mudança, é criada uma nova entrada. Essa característica é bastante importante porque as decisões normalmente são baseadas em dados históricos.

Deve-se considerar também a temporalidade dos metadados, pois sem a manutenção do histórico dos metadados, os dados históricos são invalidados com a mudança das regras do negócio [CAM98].

II.3.2.4 Não-volátil

Isto quer dizer que os dados são apenas para consulta (somente-leitura); uma vez que depois da carga inicial, eles não podem ser modificados, o que acaba os diferenciando dos bancos de dados operacionais, que permitem escrita-leitura e em geral atualizam registro a registro em múltiplas transações.

II.4 Data Marts

O termo data mart (DM) é utilizado para descrever a maneira como cada departamento individualmente implementa seu próprio sistema de gerência de informações [DEV97]. Nesse mesmo contexto, [POE98] vê um data mart como um DW orientado a assunto, que representa um subconjunto de um DW que inclui dados relevantes de uma função particular do negócio.

De uma forma simplificada, podemos considerar um Data Mart como um subconjunto lógico de um data warehouse [KIM98a].

Um DM é usualmente organizado em torno de um único processo do negócio. Ele representa um projeto que pode ser completado e executado mais rapidamente [KIM98a], e o crescimento no seu desenvolvimento em detrimento de DW, decorre algumas vezes da necessidade de obter as informações táticas e estratégicas do negócio de maneira mais imediata ou de restrições de recursos [PER98].

No próximo capítulo será visto como um DW pode ser formado a partir de DM, utilizando-se de uma arquitetura (*Bus Architecture*) e fatos e dimensões conformados.

II.5 Data Staging

A área de *Data Staging* é tudo que está entre os sistemas fontes e o DW (ver Figura 1), nela são realizados os processos que limpam, transformam, combinam, duplicam, arquivam e preparam os sistemas fontes para uso no DW [KIM98a], não

existindo necessariamente uma área física própria para isso. E esses processos são a parte mais trabalhosa na construção de um DW – pois custam em torno de 70% do esforço.

O plano para realização do processo de *Staging* é [KIM98a]:

1. Criação de um esquema de alto-nível com o fluxo da fonte até o alvo.
2. Escolha, teste e implementação de uma ferramenta de *Data Staging*.
3. Buscar as tabelas-alvo, visualizando as transformações e reestruturações nos dados.
4. Construir e testar uma tabela de dimensões² estática, para verificar problemas como segurança, transferência de arquivos, etc.
5. Construir e testar o processo lento de mudança para uma dimensão.
6. Construir e testar a carga das outras dimensões.
7. Construir e testar a carga da tabela de fatos.
8. Construir e testar o processo de carga incremental.
9. Construir e testar a carga das tabelas agregadas e/ou cargas MOLAP.
10. Projetar, construir e testar a automação do processo de *staging*.

Comentário: (Descer para)

Os principais processos da área de *Data Staging*, são:

- Extração;
- Transformação;
- Carga e Indexação;
- Checagem da qualidade do processo.

II.5.1 Extração

É o primeiro passo a ser tomado para trazer os dados para o ambiente do DW.

Pode-se dizer que extração nada mais é do que ler e entender os sistemas fontes, e copiar as partes que são necessárias para a área de *Data Staging*. Contudo, não se pode desconsiderar o esforço exigido neste momento, por volta de 60% do tempo total de construção do DW, principalmente se os diversos sistemas fontes forem bastante antigos, baseados em mainframe e contiverem uma forma de armazenamento desconhecida [KIM98a].

² Fatos e dimensões serão discutidos no capítulo 3.

Grande parte das suas tarefas estão relacionadas a determinar quais dados serão extraídos e que tipos de filtros serão aplicados, e muitas das ferramentas de extração utilizam arquivos intermediários antes de mandar os dados para o DW, exatamente para facilitar essas tarefas.

II.5.2 Transformação

Depois que os dados são extraídos para a área de *Data Staging*, os dados devem ser convertidos para se tornarem mais apresentáveis aos usuários. Existem várias formas possíveis de transformação, são elas:

- **Integração:** criação de chaves substitutas, mapeamento de chaves de um sistema para o outro, etc.
- **Verificação da integridade referencial:** entre a tabela de fatos e as dimensões.
- **Denormalização e Renormalização** da hierarquia das tabelas.
- **Limpeza** dos dados: corrigindo os erros de digitação (*Misspelling*), resolvendo conflito de domínios (e.g. conversão da representação de datas, números e caracteres de um banco de dados para outro), lidando com elementos de dados “perdidos” e definindo os formatos padrões.
- **Exclusão** dos dados selecionados dos sistemas fontes que não serão usados no DW.
- **Combinação** de bases de dados, para que os dados dos sistemas legados que utilizavam chaves estrangeiras ou uma codificação própria, passem a ter o seu valor (textual) correto.
- **Construção de agregados.**

II.5.3 Carga e Indexação

No final do processo de transformação, os dados são carregados no DW, geralmente na forma de tabelas de fatos e dimensões replicados e apresentando estas tabelas para cada data mart receptor. O data mart deve indexar os dados que chegam para facilitar a execução de consultas mais tarde.

Durante o processo de carga, serão necessários:

- Suporte para múltiplos (Data Marts) alvos.
- Otimização na carga, como e.g., criação de índices e agregados durante esse processo.
- Processo completo de suporte à carga – para exclusão e recriação de índices, particionamento de tabelas, etc.

II.5.4 Verificação da qualidade do processo

Antes de publicar o DW para a comunidade de usuários, a qualidade dos seus dados é verificada. Normalmente, relatórios de exceção são gerados com todos esses dados novos e os contadores e totais devem ser consistentes.

II.6 Tipos de Dados

Os dados de um DW são divididos, de acordo com o seu uso, em [DEV97]:

- Dados do Negócio;
- Metadados.

II.6.1 Dados do Negócio

Dados do negócio são os dados necessários para gerenciar a organização ou negócio [DEV97]. Ele representa as atividades empreendedoras do negócio e os objetos do mundo real que ele negocia, e é utilizado em sistemas fontes e sistemas de apoio à decisão.

Os tipos de dados do negócio são:

- Estruturados
 - ◆ Dados em tempo-real – são dados detalhados usados pelo negócio e acessados para escrita/leitura através de transações pré-definidas;
 - ◆ Dados derivados – são dados periódicos, em um nível detalhado ou sumarizado, derivado de algum processo sobre os dados em tempo-real e utilizado para gerenciar o negócio;

- ◆ Dados reconciliados – são um tipo especial de dados derivados, eles são utilizados para garantir a consistência dos dados através de todo o empreendimento.
- Desestruturados

Os critérios utilizados para definir esses tipos de dados são:

- Sua utilização no negócio;
- Seu escopo;
- Se eles são apenas leitura ou leitura/escrita;
- Seu posicionamento no tempo (*currency*).

II.6.2 Metadados

Metadados poderiam ser simplesmente considerados “dados sobre dados”, mas esses dados devem ser gerenciados por algum tipo de programa [DEV97]. Desta forma, os dados que descrevem o significado e a estrutura dos dados do negócio, assim como de que forma eles são criados, acessados e usados são chamados de metadados.

Para [DEV97] existem três tipos de metadados:

- Metadados de definição (*built-time*) – que são utilizados no desenvolvimento da aplicação;
- Metadados de controle – são utilizados pelo DW para controlar e gerenciar a sua infra-estrutura;
- Metadados de uso – são estruturados para uso dos usuários-finais.

Kimball et al. [KIM98a] vêem dois tipos de metadados:

- metadados *back-room* – Estão relacionados ao processo de *staging* e são mais utilizados pelos DBAs. Eles guiam a extração, limpeza e carga.
- metadados *front-room* – São mais descritivos e têm por finalidade ajudar o usuário-final. Eles ajudam no funcionamento das ferramentas de consulta e geração de relatórios.

II.7 Arquitetura de Dados para DW

Segundo [POE98], uma arquitetura é um conjunto de regras ou estruturas que provêem um ambiente para todo o projeto de um sistema ou produto, no caso, o DW.

Para Poe et al., as características que distinguem a arquitetura de um DW são:

- Os dados são extraídos de sistemas fontes, bancos de dados e arquivos;
- Os dados dos sistemas legados são integrados antes de serem carregados no DW.
- O DW é um banco de dados somente-leitura separado, projetado especificamente para apoio à decisão.
- Os usuários acessam o DW através de ferramentas e aplicações *front-end*.

Existem três tipos alternativos de arquiteturas de dados [DEV97]:

- Arquitetura de uma camada;
- Arquitetura de duas camadas;
- Arquitetura de três camadas.

II.7.1 Arquitetura de dados com uma camada

Esta arquitetura é utilizada com a finalidade de evitar a replicação dos dados. Nela, os dados são tratados da mesma forma e tanto os sistemas OLTP quanto os SADs agem sobre o mesmo conjunto de dados sem restrições de segurança.

Esse tipo de arquitetura é mais recomendada para organizações que manipulam uma grande quantidade de informações e as necessidades de análise dos dados forem limitadas, porque o suporte ao uso das informações é bastante pobre.

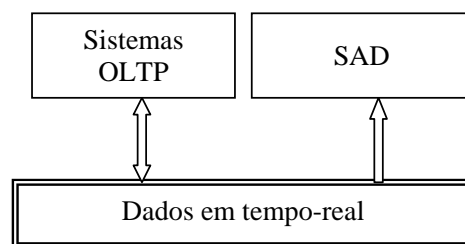


Figura 3 – Arquitetura de uma camada

II.7.2 Arquitetura de dados com duas camadas

O reconhecimento de dois níveis diferentes de utilização dos dados – operacionais e informacionais - foi um avanço, que fez com que surgisse a arquitetura de duas camadas. Um dos seus benefícios foi o fato de se dirigir melhor às necessidades do usuário-final, provendo dados separados dos operacionais, e permitindo várias derivações diferentes dos mesmos dados.

Essa abordagem fez com que o problema do uso de um só local para os dados fosse resolvido, mas da mesma forma, surgia o problema da duplicação dos dados, o que levava numa explosão nos requisitos de armazenagem de dados.

Algo bastante típico dessa arquitetura é o uso de data marts.

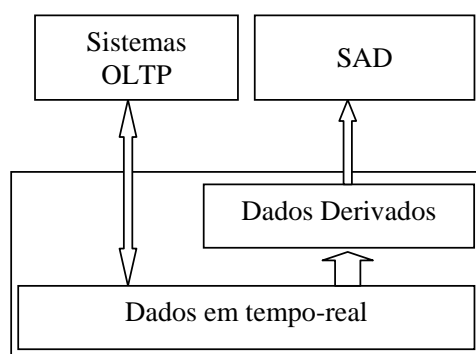


Figura 4 - Arquitetura de duas camadas

Segundo [DEV97], essa é a arquitetura mais utilizada hoje em dia.

II.7.3 Arquitetura de dados com três camadas

A transformação dos dados em tempo-real para que os dados derivados, na realidade, não é como descrito na arquitetura de duas camadas. Para que haja a derivação, é necessário o uso de uma camada intermediária – dados reconciliados.

Para isso são necessários os seguintes passos [DEV97]:

- Reconciliar os dados de diversas bases de dados na camada de dados em tempo-real.
- Derivar os dados requeridos pelos usuários através dos dados reconciliados.

Apesar da construção de uma nova área implicar em custos de armazenagem, esse custo é compensado pela redução dos custos de CPU. Além disso, a implementação física da camada de dados reconciliado deve-se a necessidade de:

- Suporte de novos usos informacionais os dados;
- Suporte à implementação de um modelo de dados;
- Suporte para reengenharia das aplicações operacionais;
- Redução do volumes de dados informacionais de gerenciamento;
- Redução das duplicações de dados informacionais do sistema;

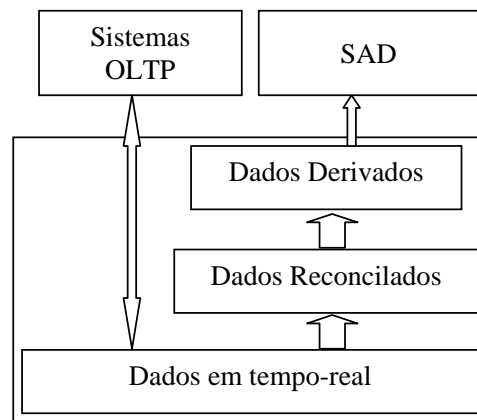


Figura 5 - Arquitetura de três camada

III Modelagem Dimensional e Recursos Analíticos

III.1 Modelagem Dimensional

Modelagem dimensional é a técnica de projeto que busca apresentar os dados de uma organização em um ambiente padrão que seja intuitivo e permita um acesso de alto desempenho.

A idéia fundamental da modelagem dimensional, é que todos os tipos de dados de um negócio podem ser representados como um tipo de cubo, onde as células do cubo contém valores mensuráveis e os limites do cubo definem as dimensões dos dados. Como esse cubo pode ter mais de três dimensões, ele é chamado de hipercubo. Todo modelo dimensional é composto por uma tabela de fatos, com múltiplas chaves, e um conjunto de tabelas menores chamadas tabelas de dimensões [KIM98a], formando o conhecido esquema estrela. A partir do esquema criado, os usuários poderão se utilizar de diversos tipos de exploração sobre os dados³ (eg. ferramentas e aplicações *front-end*, operações sobre esses dados, etc.).

III.1.1 Conceitos Básicos

Para que o processo de modelagem dimensional seja melhor compreendido, é necessário o conhecimento dos seguintes conceitos:

- Dimensões
- Fatos
- Chaves

III.1.1.1 Dimensões

Uma dimensão é uma coleção de atributos textuais, que descrevem os objetos da organização, e que estão altamente relacionados uns com os outros. As tabelas dimensionais normalmente são consideradas os pontos de entrada em um DW.

³ A exploração multidimensional será vista na seção III.4.

Os atributos dimensionais são a fonte para as restrições mais interessantes nas consultas a um DW e são virtualmente a fonte para os cabeçalhos de colunas do conjunto de respostas em SQL (e.g. relatórios).

A dimensão tempo (Figura 6) é uma dimensão especial em todo DW, pois virtualmente todo DW é uma observação em uma série de tempo de algum tipo [KIM98a]:

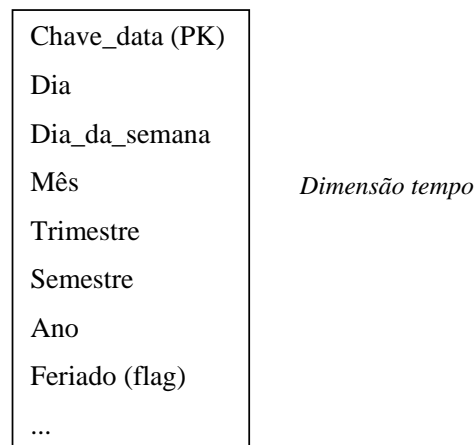


Figura 6 – Dimensão Tempo

Outros tipos de dimensões bastante encontradas em DW: clientes, produtos, vendas, localização geográfica, etc.

As operações de navegação⁴ em um sistema DW, principalmente o *drill down*, são feitos através das dimensões do seu esquema.

Atributos

Atributos são qualificadores de métricas específicas, que pertencem às dimensões e são definidos por colunas no DW. Se considerarmos a dimensão Geográfica, para conseguirmos analisá-la, dados por região, estado e cidade são necessários, cada um desses dados é um atributo da dada dimensão [MIC98b].

A hierarquia entre atributos ocorre porque normalmente os atributos de uma dimensão estão classificados em um determinada ordem. No exemplo anterior – da

⁴ As operações serão vistas na seção III.4.1.

dimensão geográfica, os atributos região, estado e cidade formam a hierarquia da dimensão.

III.1.1.2 Fatos

Um fato representa alguma coisa que não se conhece de antemão [KIM98a]. Uma tabela de fatos corresponde em um esquema relacional a uma relação muitos-para-muitos ($n-m$) entre tabelas, a sua chave primária é composta de várias chaves estrangeiras que se relacionam com as tabelas de dimensões. Os fatos mais importantes/úteis numa tabela de fatos são os fatos numéricos e aditivos.

Medidas numéricas discretas (e.g. valor das vendas na moeda corrente) são freqüentemente perfeitamente aditivas. A aditividade é crucial em aplicações DW porque essas aplicações nunca recuperam apenas um registro numa tabela de fatos; e essas centenas, milhares ou milhões de registros só podem ser adicionados para conter uma informação útil ao usuário.

Sempre que possível, os fatos de uma tabela de fatos devem ser aditivos, isto é, os fatos podem ser adicionados através de toda a dimensão. Porém existem fatos que não conseguem satisfazer essas propriedades, são:

- os fatos semi-aditivos - que são medidos pela intensidade, e sua média é feita através de todas as dimensões;
- e os não aditivos - fatos textuais, que não são desejados em um DW.

III.1.1.3 Chaves

Todas as tabelas de dimensão têm chaves de apenas uma parte (por definição elas são chaves primárias), que são definidas unicamente nesta tabela.

As chaves utilizadas em um DW devem ser chaves substitutas (*surrogates*), para que as mesmas não sejam confundidas com as chaves originais dos sistemas fontes. No momento de criação das chaves substitutas, deve-se:

- Usar um campo numérico - de preferência do tipo auto-incremento
- Evitar ao máximo que sejam usados campos do tipo data, chaves que tenham algum significado, ou a mesma chave usada no sistema OLTP, como chave das tabelas dimensionais [KIM98a].

III.1.2 Esquema Estrela

O esquema estrela é o mais comumente utilizado na modelagem dimensional. Ele é composto de uma tabela de fatos circundada por várias tabelas de dimensões e a sua natureza fere uma das premissas básicas dos sistemas OLTP, que é evitar a redundância de informações. Devido a sua ‘denormalização’, o esquema fica mais legível para o usuário – já que o número de tabelas é menor e fica mais fácil para o usuário lembrar do esquema completo, e as consultas tem um desempenho melhor - pois o número de junções se reduz.

Os benefícios do esquema estrela são [POE98]:

- Criação de um projeto de banco de dados com um tempo de resposta melhor.
- Permitir aos otimizadores do banco de dados trabalhar com um projeto de banco de dados mais simples, com o intuito de um melhor rendimento na execução dos planos.
- Comparar como os usuários finais habitualmente pensam e usam os dados.
- Simplificar a compreensão e navegação dos metadados pelos usuários e desenvolvedores.
- Alargar as chances das ferramentas de acessos aos dados *front-end*, já que algumas delas trabalham com o esquema estrela.

A Figura 7, mostra um modelo genérico de um esquema estrela [POE98]:

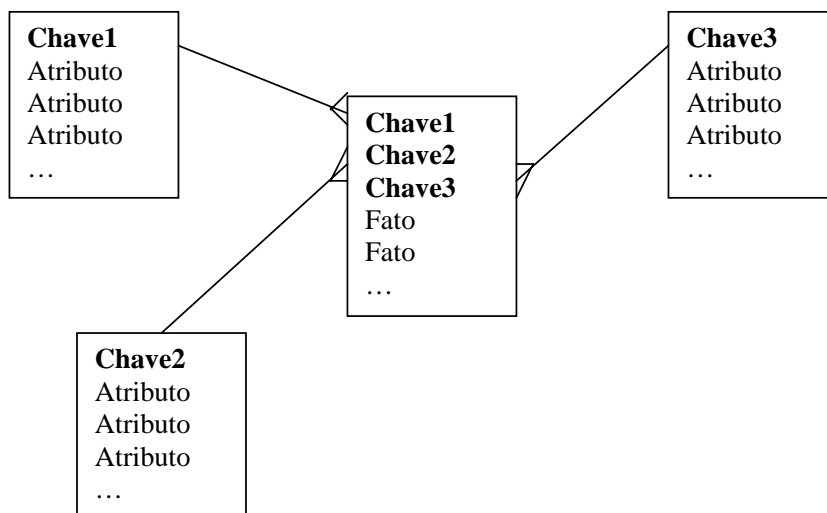


Figura 7 – Um esquema estrela genérico

III.1.3 Variantes do Esquema Estrela

As variações do esquema estrela são:

- Floco de neve

Ele armazena todas as informações dimensionais na terceira forma normal. As maiores razões para essa variação são: o aparecimento de ferramentas avançadas para apoio à decisão que podem explorar plenamente este tipo de estrutura e o maior conforto que certas organizações têm em desenvolver um projeto na terceira forma normal.

- Estrela com múltipla tabelas de fato

O uso de múltipla tabelas de fato ocorre quando os fatos não são relacionados ou quando a periodicidade dos tempos de carga diferem.

- Multi-estrela

O esquema multi-estrela ocorre quando a chave primária da tabela de fatos não é formada apenas pelas chaves estrangeiras das tabelas de dimensões.

- Estrela com tabelas associativas;
- etc.

III.1.4 Modelagem Dimensional × Modelagem Entidade/Relacionamento

Modelagem Entidade/Relacionamento

A modelagem entidade/relacionamento (ER) foi proposta por Chen em 1976, este modelo divide o BD em duas partes lógicas, as entidades (e.g. clientes, produtos) e os relacionamentos (e.g. compra, faz_parte). O modelo relacional é uma representação física do modelo ER, e ele nos fornece uma boa estratégia para armazenamento e recuperação de informações [SCH99].

O modelo de BD relacional é uma coleção de tabelas bidimensionais que são formadas por linhas e colunas – na terminologia do modelo relacional, as tabelas, linhas, e colunas são chamadas relações, atributos, e tuplas respectivamente; onde as

informações são armazenadas sem redundância, devido a normalização⁵ feita no modelo. O que para bancos de dados deste tipo, são instrumentos essenciais para a manutenção da integridade das informações e beneficiar o processamento das transações.

Diferenças entre as Modelagens

Num modelo relacional, valores indefinidos de chaves primárias são proibidos (Codd apud [SCH99]). Já num DW, um valor indefinido num atributo que está relacionado com um outro atributo tem um sentido não-ambíguo, um significado bem entendido e é inteiramente admissível.

Algumas funcionalidades, necessárias a usuários-finais, foram perdidas com a modelagem ER, devido a necessidade de tornar as transações mais eficiente [KIM97]:

- Os usuários dificilmente conseguem entender e se lembrar de um modelo ER, porque na maioria das vezes ele é composto de muitas entidades e relacionamentos;
- Não há possibilidade, pelo usuário, de navegação pelo modelo;
- Não há uma interface GUI que torne o modelo ER mais fácil de ser utilizado;
- As consultas a um modelo ER mais geral não são bem feitas pelos sistemas, tem pouca legibilidade e a recuperação de dados acaba perdendo desempenho devido as junções.
- A modelagem ER não foi projetada para o processamento analítico, como a dimensional.

A chave para entender a relação entre modelagem relacional e dimensional é que um único diagrama ER pode ser subdividido em vários diagramas dimensionais. Para transformar um diagrama ER em múltiplos diagramas dimensionais é necessário [KIM97]:

1. Separar o diagrama ER em discretos processos do negócio e modelar cada um separadamente.

⁵ Normalização é um processo da modelagem de banco de dados de relacional onde as relações ou tabelas são decompostas progressivamente em relações menores até um ponto onde todos os atributos em uma relação estão firmemente unidos com a chave primária da relação.

2. Selecionar as relações muitos-para-muitos entre as tabelas do modelo ER que contenham fatos numéricos e aditivos e designá-los como tabelas de fatos.
3. Denormalizar todas as tabelas restantes, para torná-las tabelas de dimensões. Quando as dimensões conectam mais de uma tabela de fatos, ela é chamada de dimensão conformada.

A modelagem dimensional contém as “regras do negócio” enquanto que a modelagem ER tem as “regras dos dados”. Além disso, os modelos ER são variáveis em estruturas e seus esquemas são assimétricos, enquanto que num modelo dimensional todas as dimensões servem como pontos de entradas iguais na tabela de fatos [KIM97]. A recomendação em utilizar uma modelagem dimensional deve-se, principalmente, à natureza da modelagem ER – que é mapeada em um modelo relacional normalizado, o que dificulta o processamento analítico.

III.2 Projeto de Esquemas Multidimensionais

III.2.1 Tipos de Dimensões

III.2.1.1 Dimensões de Crescimento Lento

Dimensões do tipo produto ou cliente, são exemplos clássicos de dimensões de crescimento lento. Assume-se que a chave dessas dimensões não mude nunca, mas o seu conteúdo pode mudar. Como resposta a esse tipo de mudança, o DW tem três opções:

1. Sobrescrever o registro da dimensão com os novos valores, perdendo as informações temporais;
2. Criar um novo registro usando esses valores, com uma nova chave.
3. Criar um campo “antigo” em um registro da dimensão e armazenar os valores imediatamente anteriores dos atributos.

Essas opções são conhecidas como Tipo 1, 2 e 3. A opção do tipo 1 é utilizada quando os valores antigos têm pouca importância; a do tipo 2 responde à técnica de guardar as mudanças feitas em atributos de uma dimensão, ela é usada quando uma mudança física significativa ocorre numa entidade da dimensão e é apropriado guardar a

sua história. A opção 3 responde ao uso quando a mudança é “leve” ou quando é necessário apenas armazenar os valores antigo e o novo.

III.2.1.2 Grandes Dimensões

DW que armazenam dados extremamente granularizados podem possuir algumas dimensões extremamente grandes (e.g., qualquer empresa que lida com o público em geral, tem necessidade de uma dimensão do tipo cliente que pode, facilmente, conter cinco ou dez milhões de registros de seres humanos).

Na maioria dos casos, essas dimensões muito grandes podem ser suportadas pelos SGBDs relacionais atuais. Porém, deve-se adotar um projeto para manter essas dimensões sob controle, em particular, devem ser usadas técnicas de indexação e algumas considerações, tais como:

- As dimensões não restringidas em uma consulta devem poder ser rapidamente pesquisadas, especialmente os atributos de baixa cardinalidade;
- Os atributos com restrições cruzadas nessas tabelas de dimensões devem poder ser pesquisados da forma mais eficiente possível;
- A utilização de dimensões grandes e caras não deve penalizar a consulta à tabela de fatos;
- Procurar e eliminar entradas duplicadas nessas dimensões grandes, o que pode ocorrer com frequência;
- Não criar registros adicionais para tratar o problema das dimensões que apresentam atributos com modificação lenta, pois a tabela de dimensão já é grande demais.

Os campos mais utilizados em dimensões grande, do tipo cliente, são atributos demográficos. Uma técnica muito eficiente para tratar esses tipos de atributos em tabelas grandes são as dimensões pequenas ou minidimensões.

III.2.1.3 Dimensões pequenas com crescimento rápido

Caso as mudanças nas tabelas dimensionais ocorram mais rapidamente (e.g. todo dia) e seja necessário armazenar todas as versões das informações, a técnica do Tipo 2 das dimensões de crescimento lento se aplica muito bem.

Porém é interessante definir um critério de parada para a criação de registros dimensionais, quando existem muitas mudanças em cada uma das suas medidas, pois em casos extremos como nas dimensões monstruosas de crescimento rápido, que será vista na próxima subseção, isso acontece.

III.2.1.4 Dimensões monstruosas

Esse tipo de dimensão ocorre principalmente em DW que armazenam os dados em sua forma mais granular. Isso acontece em empresas que lidam com o público em geral e têm dimensões do tipo cliente – que têm alguns milhões de registros.

Para conseguir gerenciar dimensões com essas proporções, algumas técnicas conservadoras de projeto devem ser adotadas. Em particular, as tecnologias de indexação e a abordagem de projeto de dados devem ser escolhidas para [KIM98a]:

- Suportar a navegação rápida nas dimensões
- Suportar a navegação eficiente (se não for rápida) em valores nas tabelas dimensionais
- Não penalizar as consultas à tabela de fatos devido ao uso dessas dimensões
- Descobrir e suprimir as entradas duplicadas nessa dimensão
- Não criar registros adicionais para lidar com os problemas dessas dimensões.

O pior caso ocorre quando essas dimensões monstruosas têm um **crescimento rápido**. Neste caso, algumas técnicas devem ser utilizadas para poder atualizar esses dados. Uma boa solução é dividir essa dimensão em dimensões separadas (e.g. no caso de uma dimensão de clientes, os seus dados pessoais podem ser separados dos dados demográficos).

III.2.1.5 Outros tipos de Dimensões

Dimensões degeneradas

As dimensões degeneradas geralmente ocorrem no projeto de tabelas de fatos com campos orientados a itens, que no final gera uma dimensão sem atributos. Neste caso, a dimensão não é gerada e a chave degenerada é usada para agrupar esses itens.

Dimensões lixo (junk)

No momento da extração dos dados dos sistemas fontes, alguns atributos parecem não fazer sentido na estrutura da organização – em sua maioria, esses campos são flags ou atributos texto que no mínimo têm um significado bastante obscuro.

Algumas alternativas (não muito boas) que podem ser tomadas pelo projetista são:

- Deixar esses atributos imutáveis nos registros da tabela de fatos;
- Manter cada flag ou atributo texto em sua própria dimensão;
- Descartar esses campos do projeto.

O que de melhor pode ser feito, é o estudo desses atributos e a união deles em uma ou mais dimensões lixo (*junk*).

III.2.2 Granularidade

As dimensões costumam ser, normalmente, granulares (atômicas), porque cada registro nesta tabela corresponde a uma única descrição de um dia, um produto, etc. [KIM98a]. A manutenção dos dados nessa forma, é uma boa prática dos projetistas, porque assim, as consultas podem descer (*drill-down*) até o nível mais baixo possível e é possível minerar esse dados – já que a mineração é menos efetiva em dados agregados, e conhecer melhor os hábitos dos cliente.

A tabela de fatos pode conter a sua granularidade diferente das dimensões do esquema, o nível de detalhe pode ser atômico ou agregado, dependendo da decisão feita pelo projetista do DW.

III.2.3 Agregação

Agregação pode ser considerada a sumarização de um conjunto de registros, ou melhor, é o processo de acumular fatos através de atributos pré-definidos. Os agregados têm um efeito muito grande no desempenho do DW [KIM98a]. O uso de agregados deve ser bem explorado antes do investimento em novo hardware, pois o ganho de desempenho pode chegar num aumento de velocidade na ordem de 100 ou 1000 vezes mais.

Os fatores primários para a criação de agregados pré-armazenados são [POE98]:

- Aumentar o desempenho das consultas dos usuários finais
- Reduzir o número total de ciclos da CPU utilizados.

Os pontos básicos da navegação de agregados são:

- Em um ambiente de DW bem projetado, muitos conjuntos de agregados são construídos, representando os níveis de agrupamento mais comuns através das dimensões chaves do DW.
- Um navegador de agregados (Figura 8) fica entre os pedidos do usuário e o SGBD. Ele intercepta as consultas SQL do usuário e quando possível, transforma essa consulta, para torná-la “consciente” dos agregados.

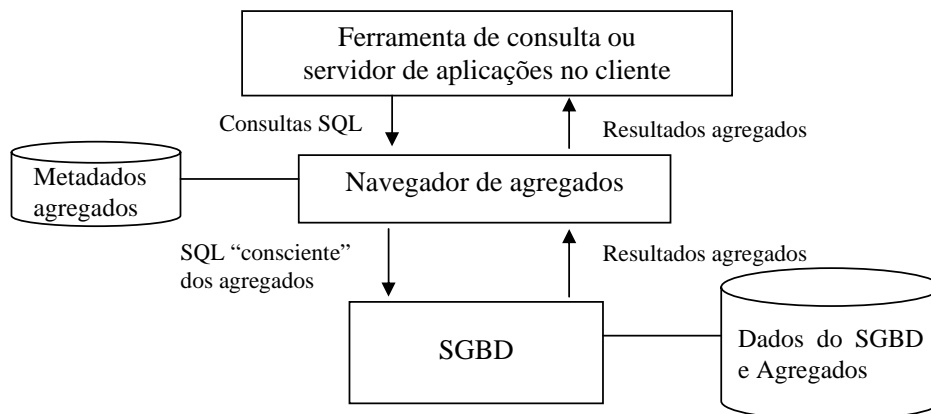


Figura 8 – Um navegador de agregados [KIM98a]

III.2.4 Dimensões e Fatos Conformados

Uma dimensão conformada (DC) é uma dimensão que significa a mesma coisa para todas as possíveis tabelas de fatos a que ela pode ser unida. Isto significa que uma dimensão conformada é identicamente a mesma dimensão para cada data mart, o que torna bastante difícil a tarefa de criá-la e mantê-la - no caso de não haver um time de desenvolvimento do DW central, pois se uma das dimensões conformadas sair do padrão em um dos data marts, o DW não poderá funcionar como um conjunto integrado.

Os fatos conformados são estabelecidos no mesmo momento que as dimensões conformadas, mas ‘conformar’ um fato é algo bastante difícil, pois muitas vezes os fatos utilizam unidades de medidas diferentes ou dão uma mesma interpretação para fatos diferentes. A definição de fatos conformados só é feita quando utilizamos a mesma

terminologia através dos data marts ou quando construímos relatórios que fazem a operação de *drill-across* nos data marts.

III.2.5 Data Warehouse Bus Architecture

As abordagens mais críticas para a construção de um DW são: construir um DW central, ou separado por áreas - construídos à medida que eles sejam necessários, mas nenhuma das duas abordagens é muito boa. O DW Bus Architecture [KIM98a] é uma abordagem passo-a-passo para construção de um DW. Nela os data marts são organizados em uma arquitetura, em volta das dimensões e fatos conformados.

O planejamento do DW é feito através de fases em uma arquitetura global, onde a cada passo um DM é implementado separadamente e a cada nova implementação de um DM o mesmo adere à essa arquitetura, se encaixando como peças de um quebra-cabeça [KIM98a]. Em certo momento, quando um número de DM for significativo, teremos um DW integrado.

Os passos seguidos pela pessoa que implementa a construção de um DW são:

1. Criar uma arquitetura que circunde e defina o escopo e implementação de um DW completo.
2. Supervisionar a construção de cada peça do DW completo.

A Figura 9 representa um exemplo conceitual de como o *DW bus* funcionaria para dois processo do negócio [KIM98a].

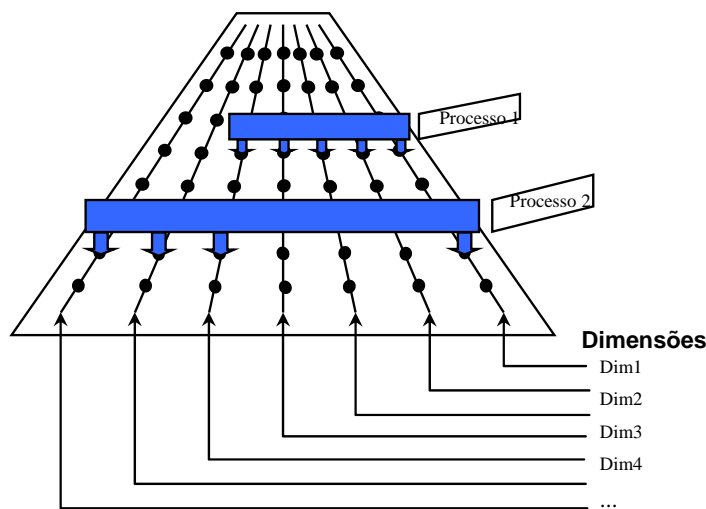


Figura 9 – DW Bus Architecture [KIM98a]

III.3 Bancos de Dados Multidimensionais

Embora seja possível utilizar um banco de dados relacional para representar uma estrutura dimensional [CAM98], seria mais interessante, utilizar um banco de dados que fosse projetado especialmente para suportá-la.

Poderíamos comparar um banco de dados multidimensional (BDM) a uma matriz, onde cada eixo corresponde a uma dimensão e cada elemento dentro de uma dimensão corresponde a uma posição. Desta forma, fica mais fácil para o usuário compreender as informações e poder manipulá-las e visualizá-las.

Um BDM armazena os dados em arrays multidimensionais e esse array tem um número fixo de dimensões, onde cada dimensão pode ser composta por múltiplos níveis, de forma que os dados possam ser agrupados.

Uma representação de um array bidimensional é apresentado na Figura 10.

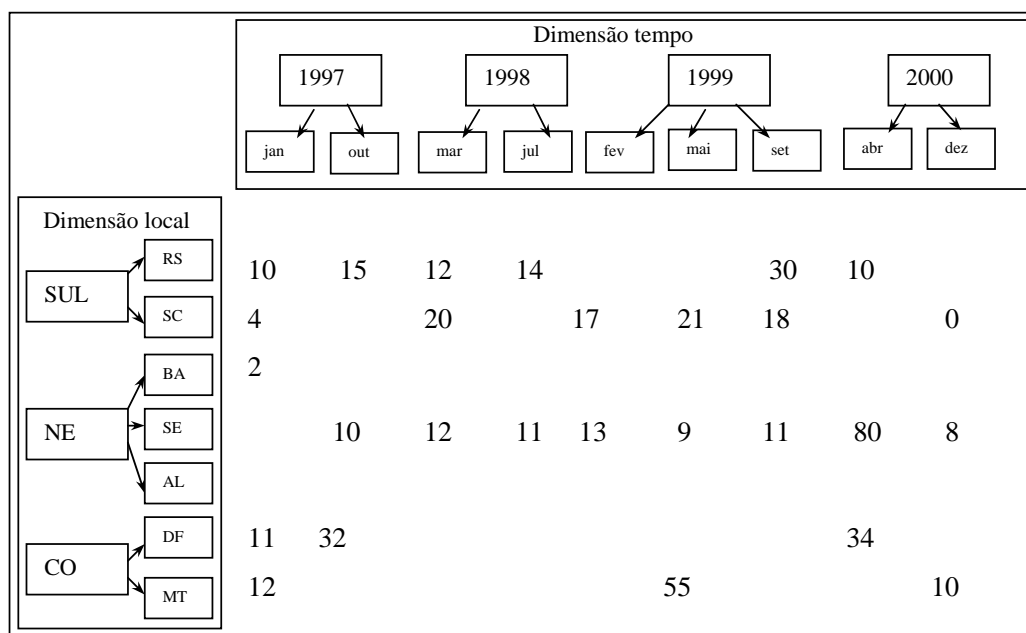


Figura 10 – Array Bidimensional [CAM98]

III.3.1 Recursos Analíticos

III.3.1.1 OLAP

OLAP (*on-line analytical processing*) ou processamento analítico online, diz respeito ao tipo de processamento e às ferramentas voltadas para análise típica de SADs. Ele sempre envolve consultas interativas aos dados - seguindo um caminho de análise através de múltiplos passos -, e possui capacidades analíticas. Os dados, neste tipo de processamento, são apresentados numa visão multidimensional e esta visão é independente de como os dados estão armazenados [CAM98].

OLAP × OLTP

Como visto na seção II.1, dentre os sistemas computacionais de uma empresa, podemos distinguir dois tipos de aplicação [SCH99]: as aplicações que apoiam os processos primários da companhia ou OLTP e as aplicações que gerenciam as informações e o controle dos processos primários (SAD) ou OLAP.

Resumidamente, as características que distinguem um sistema OLAP de um sistema OLTP são [CAM98]:

OLTP	OLAP
Relacional	Multidimensional
Individualizados	Sumarizados
Presentes	Históricos
Um registro de cada vez	Muitos registros por vez
Orientados a processo	Orientados ao negócio

Tabela 2 – Comparação entre OLAP e OLTP

III.3.1.2 ROLAP

ROLAP (OLAP relacional) é um conjunto de interfaces para o usuário e aplicações que dão aos bancos de dados relacionais uma cara de dimensional [KIM98a]. Suas ferramentas fazem uso extensivo de metadados. Ele é conhecido como uma

aplicação de três camadas (*three tier*) porque existe um servidor de aplicações entre as ferramentas de *front-end* dos usuários e o DW em si e os metadados.

III.3.1.3 MOLAP

MOLAP (OLAP multidimensional) é um conjunto de interfaces, aplicações e tecnologias de bancos de dados proprietários que tem uma cara dimensional [KIM98a]. MOLAP é uma camada de aplicação que contém três camadas ou mais (*three+ tier*); como no ROLAP, ele contém um servidor (OLAP) mais o cubo OLAP entre as ferramentas *front-end* e o DW.

Como as ferramentas MOLAP foram contruídas para o apoio à decisão, elas costumam ter funções analíticas mais poderosas.

III.4 Exploração Multidimensional

Depois de construído o DW, é chegada a hora de ele ser explorado pelos seus usuários-finais. Para isso, esses usuários podem se utilizar de operações, ferramentas e aplicações projetadas especialmente para eles.

III.4.1 Operações

As principais operações de navegação em um sistema OLAP são:

- Slice-dice;
- Drill-down;
- Drill-up;
- Alguns outros tipos de “drill”.

III.4.1.1 Slice-dice

Corresponde à técnica de mudar a ordem das dimensões, mudando desta forma a orientação segundo a qual os dados são visualizados.

III.4.1.2 Drill Down

É a forma de navegação mais antiga, seria uma maneira de pedir mais detalhes/“descer” pelas hierarquias das dimensões [KIM96b]. Todos os atributos das tabelas dimensionais podem se tornar colunas de agrupamento, e esse processo de adicionar grupos a partir de colunas pode ser composto por várias tabelas de dimensão.

Muitas ferramentas de DW, existentes hoje em dia, não conseguem implementar um *drill-down* de forma satisfatória, pois nessa operação deve haver mais de uma hierarquia bem-definida em uma dimensão e o usuário deve poder atravessar qualquer hierarquia e escolher atributos que não estejam relacionados a essa hierarquia. Na maioria das ferramentas, a cada vez que se quer adicionar um novo grupo ele deve estar hierarquicamente relacionados com os outros atributos agrupados.

III.4.1.3 Drill up

É a operação contrária ao *drill-down*, nela as colunas de agrupamento são subtraídas, mas não necessariamente na mesma ordem em que elas foram adicionadas.

III.4.1.4 Outros tipos de drill

Drill Across

É o processo de unir duas ou mais tabelas de fatos na mesma granularidade, isto é, tabelas com o mesmo conjunto de colunas agrupadas e restrições dimensionais [KIM96b]. Um relatório *drill-across* pode ser criado usando colunas de agrupamento que se aplicam a todas as tabelas de fatos utilizadas no relatório.

Drill Around

Esse processo é muito parecido com o *drill-across*, ele ocorre em um DW onde as tabelas de fatos relacionadas que compartilham dimensões em comum não estão organizados em uma ordem linear. Para gerar um relatório, as consultas são executadas separadamente para cada tabela de fatos e um *outer-join* é feito com essas consultas.

III.4.2 Relatórios

Os relatórios padrões provêm a habilidade de criar estilos para a produção de relatórios de formato fixo, que tem uma interação mínima dos usuários, uma grande audiência e execuções regulares [KIM98a].

As atividades de consulta em muitos DW são consideradas como se fossem relatórios padrões, por isso, é imprescindível que os DW suportem esses relatórios.

Os requisitos das ferramentas de criação de relatórios padrões são [KIM98a]:

- Ambiente de desenvolvimento de relatórios;
- Servidor de execução de relatórios;
- Capacidades de direção a parâmetros de variáveis;
- Agendamento da execução dos relatórios baseado em tempo ou evento;
- Execução interativa;
- Definição flexível de relatórios;
- Entrega dos relatórios de forma flexível (email, web, diretório na rede);
- Acessibilidade ao usuário para publicação;
- União de relatórios;
- Biblioteca de relatórios com capacidade de navegação;
- Distribuição em massa;
- Ferramentas de administração do ambiente de relatórios.

III.4.3 Mineração

A Mineração de Dados é tratada como sinônimo de KDD (*Knowledge Discovery in Databases*), mas KDD refere-se a todo o processo de descobrimento de conhecimentos a partir de dados, enquanto Mineração de Dados refere-se à aplicação de algoritmos para extrair padrões sobre dados sem a aplicação das outras etapas do processo de KDD.

As principais técnicas de Mineração de Dados são [FAY96]:

- Associação
- Classificação
- Predição
- Segmentação

- Análise de Séries de Tempo (*time-series*)

Tanto a mineração quanto o DW são sistemas de apoio à decisão, e apesar da mineração de dados poder ser feita sem o uso de um data warehouse, o uso deste aumenta as chances de sucesso da mineração, pois os elementos que fazem parte da natureza de um data warehouse melhoram o processo de mineração [INM96b].

A integração dessas duas tecnologias é feita devido:

- Integração de dados

Pois sem isso, o ‘minerador’ iria gastar bastante tempo nas tarefas de limpeza e integração de dados.

- Dados detalhados e sumarizados

Dados detalhados - são necessários quando o minerador precisa examinar os dados na sua maneira mais granular.

Dados sumarizados - assegura ao minerador desenvolver seu trabalho de forma melhor, já que não necessita construir tudo desde o início.

- Dados históricos

Informações bastante importantes estão escondidas nesse tipo de dados.

Informações históricas são cruciais para entender a sazonalidade do negócio e os grandes ciclos do negócio, que toda corporação está sujeita.

- Metadados

Servem como um mapa para o minerador, que os usam para descrever não o conteúdo, mas o contexto da informação.

Também no contexto de integração surge o chamado **OLAM** (*On-Line Analytical Mining*) [HAN98], pois se as ferramentas de KDD pudessem trabalhar com o processo OLAP do DW, elas poderiam oferecer recursos mais poderosos de navegação e desta forma a mineração seria mais rápida com o uso de bancos de dados multidimensionais (mudança de processamento ‘*batch*’ para on-line), além de que as operações de mineração (associação, classificação, segmentação, etc.) poderiam se unir com todas as operações OLAP (*drill*, etc.).

III.4.4 Consultas

Existem três opções de serviços de consulta que podem ser locados na arquitetura:

- No desktop;
- No servidor de aplicações;
- No banco de dados.

Hoje em dia, a maioria desses serviços são entregues como parte do conjunto de ferramentas *front-end* que residem no desktop.

IV Processo

Kimball et al. [KIM98a] descrevem em seu livro um plano de projeto do ciclo de vida de um DW – concebido através da experiência dos mesmos, com cinco fases principais. Sendo que em cada etapa dessas fases, é considerada a aceitação do usuário para que o projeto seja revisado. Tomando como base a sua classificação, as seguintes fases poderiam ser distinguidas:

- Gerenciamento do Projeto e Requisitos
- Projeto dos Dados
- Arquitetura
- Implementação
- Crescimento e Validação

Já [POE98] utiliza o seu conhecimento para descrever o ciclo de vida de apoio à decisão em dez fases distintas:

1. Planejamento
2. Agrupamento dos requerimento de dados e Modelagem
3. Projeto Físico e Desenvolvimento do Banco de Dados
4. Integração, Mapeamento e Fonte (*Sourcing*) de dados
5. População o DW
6. Automatização o processo de gerenciamento dos dados
7. Criação de um conjunto inicial de relatórios
8. Validação dos dados e Teste
9. Treinamento
10. *Rollout*

O ciclo de vida de desenvolvimento de um DW na visão de Gray e Watson (apud [WEI99]), consiste em um número de processos que devem ser executados, para assegurar que o projeto do DW foi otimizado e que os dados no mesmo estejam corretos. Esses passos podem ser agrupados segundo a Figura 11:

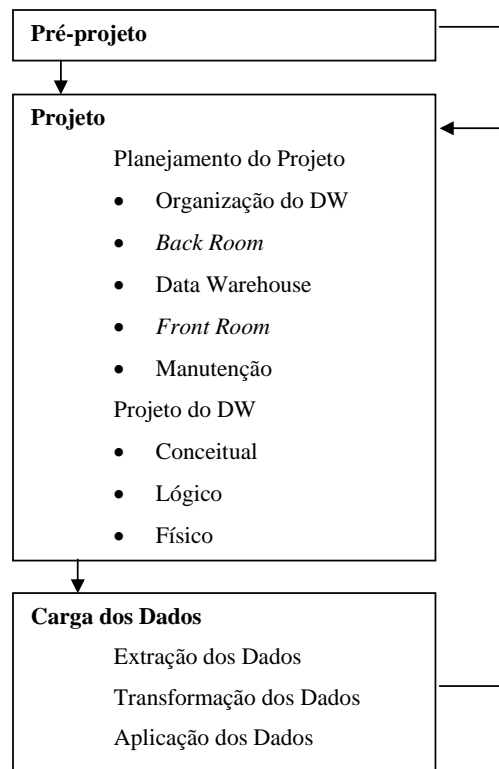


Figura 11 – Ciclo de Vida do Projeto de DW [WEI99]

Considerando o ciclo de vida apresentado por [KIM98a], e acrescentando algumas características dos demais ciclos de vida estudados, temos o esquema que segue.

IV.1 Planejamento do Projeto

Nesta fase são feitos:

- A Definição do Projeto;
- O Planejamento do Projeto e Gerenciamento;
- A Definição das Necessidades (Requerimentos) dos Usuários;

Definição do Projeto

Neste momento, é avaliada a predisposição/facilidade de criação de um data warehouse e qual o escopo do projeto, dessa forma é desenvolvido um projeto preliminar e construída a justificativa do negócio.

Planejamento do Projeto e Gerenciamento

Depois de definido o projeto, deve-se definir os recursos e linhas de tempo, os participantes do negócio e suas responsabilidades [POE98]. Além disso, estabelece-se a identidade do projeto, os seus recursos são identificados, e é preparado um rascunho do plano do projeto.

O time de projeto inicial começa a ser conduzido para iniciar o planejamento. É feita uma primeira revisão do projeto e passam a ser desenvolvidos um plano de projeto de comunicação, um programa para medir o alcance/sucesso, um processo para gerenciar o escopo, e o andamento do projeto é gerenciado [KIM98a].

Definição das Necessidades (Requerimentos) dos Usuários

O time que entrevistará os usuários, é identificado e preparado nesta fase e as entrevistas são agendadas. Depois das entrevistas serem preparadas, ela é conduzida para os usuários do negócio e para o pessoal que lida com os dados (usualmente, no Brasil, são os funcionários do CPD).

Os primeiros resultados/descobertas da entrevista passam a ser analisados e documentados, e os requisitos dos usuários são publicados. É feita uma nova revisão no escopo do projeto e os usuários dão o seu aceite no projeto.

IV.2 Projeto dos Dados

Neste instante, é definido um modelo lógico, é criado um modelo dimensional do negócio e é definida a melhor base de dados [POE98]. Ela pode ser subdividida em:

- Modelagem dimensional;
- Análise da base de dados.

Modelagem Dimensional

Nesta fase é construída uma tabela (*Matrix*).

Depois da escolha do data mart, a sua granularidade é declarada e suas dimensões são escolhidas. A seguir, é desenvolvido o diagrama da tabela de fato – que deve ser documentado em detalhes, e são desenvolvidos os detalhes das dimensões. Se existir algum fatos derivados, ele é desenvolvido.

É feita uma revisão com o usuário e se a mesma for aceita, parte-se para a revisão das recomendações do projeto do BD para ferramentas para os usuários-finais e das recomendações do projeto do BD para o SGBD. O modelo lógico, então, é completado. Outra coisa bastante importante nesta fase, é a identificação dos candidatos para agregados que serão pré-armazenados e é desenvolvida uma estratégia de projeto para a agregação de tabelas.

Deve-se fazer uma certificação do projeto do BD com o vendedor da ferramenta de SAD, para ver se existe viabilidade no projeto.

Análise das Bases de Dados

Os candidatos a base de dados são identificados, e é desenvolvida uma fonte para um mapeamento dos dados alvo. Faz-se uma navegação preliminar no conteúdo dos dados e estima-se o número de linhas (registros).

IV.3 Arquitetura

A arquitetura do DW pode ser definida através das seguintes fases:

- Projeto da arquitetura técnica
- Implementação das medidas de Segurança
- Desenvolvimento de um planejamento estratégico de Segurança
- Seleção e Instalação do Produto

Projeto da Arquitetura Técnica

Deve-se:

1. Criar uma força-tarefa da Arquitetura
2. Juntar e documentar as exigências técnicas
3. Revisar o Ambiente Técnico Atual
4. Criar um Plano de Arquitetura
5. Determinar uma abordagem com fases para a Implementação
6. Criar um Plano de Infra-estrutura
7. Desenvolver Recomendações de Configuração
8. Aceitação do usuário/Revisão do Projeto

Implementar Medidas Táticas de Segurança

Nesta fase:

1. Desenvolve-se um Plano de Segurança Tático
2. Tem-se um ambiente Físico seguro
3. São instalados os software para verificação de vírus
4. Há um acesso seguro ao Ambiente
5. Há um acesso seguro para fora de Ambiente
6. Implementa-se um esquema de senhas rigoroso
7. Implementam-se controles de instalação de Software
8. Faz-se auditoria das violações de segurança
9. Monitora-se os Privilégios de Segurança por Indivíduo
10. Aceitação do usuário/Revisão do Projeto

Desenvolver um Planejamento Estratégico de Segurança

1. Projetar a Arquitetura de Segurança
2. Implementar tokens de acesso (Eliminação de senhas)
3. Implementar chaves Públicas/Privadas para autenticação
4. Implementar refinamentos (“*Tunneling*”) seguros para acesso remoto
5. Centralizar a Autenticação e o controle de acesso
6. Implementar Certificados assinados para Downloads de Software

7. Aceitação do usuário/Revisão do Projeto

Seleção do Produto

Esses passos devem ser repetidos para cada área da seleção:

1. Desenvolver uma Matriz de Avaliação
2. Pesquisar Produtos Candidatos
3. Desenvolver uma lista pequena de produtos
4. Avaliar as Opções de Produtos
5. Protótipo opcional (deve ser repetido para produtos diferentes)
 - Selecionar os processos do negócio/Avaliação do dados
 - Definir critérios de conclusão
 - Adquirir recursos (Internos/Vendedor)
 - Determinar Configuração de Teste
 - Instalar componentes e pré-requisitos de avaliação
 - Treinar o time de avaliação
 - Desenvolver e refinar o Protótipo
 - Conduzir os testes
 - Analisar e Documentar os Resultados
6. Determinar as recomendações dos produtos
7. Apresentar os achados /resultados para a Gerencia
8. Negociar o contrato
9. Aceitação do usuário/Revisão do Projeto

Instalação do Produto

Os seguintes passos devem ser repetidos para cada produto:

1. Planejamento de instalação
2. Conhecer os pré-requisitos
3. Instalar Hardware / Software
4. Testar Hardware / Software
5. Aceitação do usuário/Revisão do Projeto

IV.4 Implementação

A fase de implementação do DW está subdividida em:

- Projeto físico do BD
- Implementação Física do BD
- Projeto e Desenvolvimento do *Data Staging*
- Popular e Validar o Banco de Dados
- Ajuste de Desempenho
- Automatização do processo de gerenciamento dos dados
- Especificação das aplicações para os usuários finais
- Desenvolvimento das aplicações para os usuários finais

Projeto Físico do Banco de Dados

1. Definir Padrões
2. Projeto físico das tabelas e colunas
3. Estimar o tamanho do BD
4. Desenvolver um plano inicial de indexação
5. Desenvolver um plano inicial de Agregação
6. Desenvolver um plano inicial de Particionamento
7. Aceitação do usuário/Revisão do Projeto

Implementação Física do Banco de Dados

1. Definição da melhor base de dados
2. Determinar os parâmetros fixos do SGBD
3. Instalar o SGBD
4. Otimizar os parâmetros mutáveis do SGBD
5. Construir uma estrutura de armazenamento física
6. “Setup RAID”
7. Completar os tamanhos de tabelas e índices
8. Criação dos objetos de dados: tabelas e índices
9. Identificação das chaves

10. Aceitação do usuário/Revisão do Projeto

Projeto e Desenvolvimento do *Data Staging*

1. Desenvolver o processo de Staging de alto nível
2. Desenvolver um plano de staging detalhado para cada tabela
3. Organizar o ambiente de desenvolvimento
4. Definir e Implementar os metadados do Staging
5. Desenvolver o primeiro processo estático para as tabelas de dimensão (Extrair, Transformar e Carregar)
6. Desenvolver o primeiro processo de manutenção das dimensões
7. Desenvolver os processos restantes sobre as tabelas de dimensão
8. Desenvolver o processo da tabela de fatos (Extrair, Transformar e Carregar)
9. Desenvolver processos incrementais para a tabela de fatos
10. Desenvolver e implementar a limpeza dos dados
11. Desenvolver e implementar o processo de agregação
12. Criação de estratégias de exceção
13. Automatizar o processo inteiro
 - Automação da extração dos dados
 - Automação da conversão dos dados
 - Criação de procedimentos de *backup* e recuperação
 - Automação da carga dos dados
 - Teste dos procedimentos automatizados
14. Desenvolver processos que garantam a qualidade dos dados
15. Implementar a Administração do BD (Arquivar, Backup e Recuperação)
16. Aceitação do usuário/Revisão do Projeto

Popular e Validar o Banco de Dados

1. Organizar o ambiente de produção
2. Carga inicial do dados de teste
3. Validação inicial dos dados /Garantia da Qualidade
4. Carga dos dados históricos
5. Executar a validação dos dados / Garantia da Qualidade

6. Aceitação do usuário/Revisão do Projeto

Ajuste de Desempenho

1. Estabelecer um ponto de referência para as consultas
2. Revisar Indexações e Agregações
3. Revisar o “tuning” específico da ferramenta
4. Conduzir continuamente a monitoração do BD
5. Aceitação do usuário/Revisão do Projeto

Especificação das aplicações para os usuários finais

1. Identificar e Priorizar os relatórios Candidatos
2. Projetar uma abordagem para navegação no modelo
3. Desenvolver uma aplicação padrão para o usuário-final
4. Documentar Detalhadamente as especificações do modelo
5. Revisar as especificações da aplicação com os próprios usuário
6. Revisar as especificações da aplicação para o usuário-final
7. Revisar o escopo do Projeto
8. Aceitação do usuário/Revisão do Projeto

Desenvolvimento das aplicações para os usuários finais

1. Selecionar uma abordagem de implementação
2. Revisar as especificações da aplicação
3. Revisar os padrões da aplicação
4. Popular as ferramentas dos usuários-finais com Metadados
5. Desenvolver as aplicações para os usuários-finais
6. Prover precisão dos dados e avaliação da limpeza
7. Desenvolver a navegação para os usuários finais
8. Revisar com os usuários
9. Documentar as aplicações dos usuários-finais
10. Desenvolver procedimentos de manutenção para as aplicações dos usuários-finais

11. Desenvolver procedimentos de lançamento das aplicações dos usuários-finais
12. Criação de um conjunto inicial de relatórios [POE98]
 - Desenvolvimento de caminhos e estruturas de navegação
 - Desenvolvimento da essência/conteúdo dos relatórios
 - Teste dos relatórios
 - Aplicação de documentos
13. Aceitação do usuário/Revisão do Projeto

IV.5 Crescimento e Validação

Neste momento, faz-se:

- A depuração do planejamento
- Os testes completos do sistema
- Depuração
- Manutenção e Crescimento do DW

Depuração do Planejamento

1. Desenvolver um checklist com a infra-estrutura do Desktop
2. Desenvolver uma estratégia inicial para educação dos usuários
3. Definir uma estratégia para apoio ao usuário
4. Definir um plano de lançamento
5. Revisar as estratégias de desenvolvimento e plano de lançamento
6. Desenvolver os materiais do curso para os usuários
7. Desenvolver procedimentos de apoio
8. Aceitação do usuário/Revisão do Projeto

Testes completos do Sistema

1. Executar o processo completo de *Data Staging*
2. Realizar procedimentos padrões de garantia de qualidade (QA)
3. Executar a essência/centro das aplicações para usuários finais

4. Revisar processo global
5. Aceitação do usuário/Revisão do Projeto

Depuração

1. Avaliar a disponibilidade de desenvolvimento
2. Configurar e Testar a Infra-estrutura do Desktop
3. Organizar (*set up*) os Privilégios de Segurança
4. Validação dos dados:
 - Utilizando o conjunto de relatórios iniciais
 - Utilizando processos padrões
 - Mudança de dados interativa
5. Educar os usuários
6. Aceitação do usuário/Revisão do Projeto

Manutenção do DW

1. Prover continuamente apoio ao usuário
2. Prover continuamente educação ao usuário
3. Manter a Infra-estrutura técnica
4. Monitorar o desempenho das consultas dos usuários-finais
5. Monitorar o desempenho do *Data Staging*
6. Monitorar continuamente o sucesso
7. Comunicar o sucesso do mercado continuamente
8. Aceitação do usuário/Revisão do Projeto

Crescimento do DW

1. Estabelecer um comitê guia para o DW
2. Estabelecer estratégia de priorização de melhoras
3. Usar interativamente o ciclo de vida dimensional do negócio

Fase pós ciclo de vida

Treinamento

1. Criação de procedimentos de apoio ao usuário
2. Projeto de programas de treinamento para a comunidade de usuários
3. Marketing interno sobre o DW

Rollout

V Estudo de Caso

V.1 Base Fonte

Como fonte de dados, foi escolhido o Sistema de Informações de Mortalidade (SIM) que contém dados de óbitos no Brasil entre 1979 e 1997. As informações foram fornecidas pelo Ministério da Saúde (MS), através do DATASUS.

Os dados entre os anos de 1979 a 1996 já são considerados como definitivos pelo MS. Basicamente, eles são separados por tipo de óbito (fetal ou não fetal). Os óbitos fetais são separados, em arquivos diferentes, apenas por ano. Os óbitos não fetais são separados por UF/ano, no intuito de facilitar a sua análise. Depois desse período, os dados ainda são vistos como provisórios/parciais e por isso eles estão separados apenas pela UF que informou o óbito.

As tabelas de óbitos⁶ são subdivididas da seguinte forma:

- Óbitos não-fetais para cada ano e estado (entre 79 e 96).
- Óbitos fetais para cada ano e estado (entre 79 e 96).
- Óbitos fetais e não-fetais para cada ano e estado (entre 95 e 97).
- Óbitos não-fetais para cada ano (entre 95 e 97), para pessoas residentes fora do país ou com residência ignorada.

Para cada ano, existem os seguintes arquivos [DAT98?]:

1. DOFETaa.DBC, contendo os dados dos óbitos fetais, de 1979 a 1996, inclusive os de residência ignorada ou no estrangeiro (onde aa corresponde ao ano);
2. DORuuua.DBC, contendo os dados dos óbitos não fetais por UF de residência, de 1979 a 1996; para os de residência ignorada ou no estrangeiro, uu é igual a IG (onde uu corresponde a UF e aa corresponde ao ano); e
3. DOIuuua.DBC, contendo os dados dos óbitos, tanto fetais como não fetais, inclusive de residência ignorada ou no estrangeiro, por UF informante, de 1995 a 1997.

⁶ A descrição de todas as tabelas do SIM se encontra nos Anexos (seção VIII)

Grande parte dos atributos dessas tabelas contém uma codificação própria. No caso do estado civil, por exemplo, a codificação é a seguinte:

- ✓ 0 ou 9: Ignorado;
- ✓ 1: Solteiro;
- ✓ 2: Casado;
- ✓ 3: Viúvo;
- ✓ 4: Separado judicialmente;
- ✓ 5: Outro.

Outros atributos fazem referência às outras tabelas do sistema. Essa tabelas contém informações sobre:

- Ocupação;
- Países;
- Estados;
- Municípios;
- Etnia (no caso de índios);
- 7 tabelas diferentes, contendo as categorias, capítulos e as doenças individualmente classificadas pelo CID - Classificação Internacional de Doenças, dependendo da Revisão dos dados (9ª Revisão até 95 e 10ª Revisão a partir de 96).

V.2 Modelagem

Apesar de não ter havido a fase de pré-projeto do DW, onde ocorrem as entrevistas com os usuários, tentou-se desenvolver um esquema que facilitasse a navegação e manipulação dos dados pelos prováveis usuários do mesmo. Dessa forma, foram construídas seis dimensões, cada uma delas representam um ponto de entrada válido e interessante no contexto do sistema. O esquema estrela do sistema acabou ficando como na Figura 12.

No momento da modelagem do sistema, foram decididos quais atributos poderiam estar na tabela de fatos e quais seriam as dimensões do esquema. Todos os atributos do

sistema fonte que tinham valores únicos⁷ foram utilizados no esquema, porque mesmo que numa primeira análise a sua utilização na DW não parecesse muito relevante, eles poderiam ser utilizados por um “minerador de dados” em busca de padrões nas mortes ocorridas no Brasil nos últimos anos.

Num primeiro momento da modelagem, pensou-se em unir os atributos que corresponderiam aos dados pessoais do morto, com as informações que só teriam validade se o mesmo fosse recém-nascido, mas percebeu-se que essas informações poderiam se situar em dimensões distintas no DW.

Também nesta fase de modelagem, colocou-se em dúvida se poderia ser criada uma dimensão tempo que pudesse ser usada tanto pela data de óbito quanto pela data de nascimento, porém notou-se que o uso da data de nascimento como dimensão era irrelevante para o esquema e que a mesma poderia ser utilizada apenas como atributo em uma dimensão. Até mesmo a possibilidade de modelar com um esquema do tipo floco de neve foi imaginado.

No final, as dimensões escolhidas foram:

- A tradicional dimensão tempo, que apareceu no esquema através do campo `data_obito`, foi mapeada para a tabela Data. Ela contém os seguintes atributos⁸: Dia do óbito, Mês do óbito e Ano do óbito.
- Os dados referentes apenas aos fetos foram unidos na tabela Feto. Seus atributos são: Ocupação do pai, Instrução do pai, Ocupação da mãe, Instrução da mãe, Idade da mãe, número de filhos vivos da mãe, número de filhos mortos da mãe, semanas de gestação, tipo gravidez, tipo de parto, Peso ao nascer.
- Os dados pessoais do morto, (e.g. sexo, idade, instrução, etc.) ficaram na tabela Pessoal. Os campos que pertencem a essa dimensão são: Sexo, Estado civil, Data de nascimento, Idade, Ocupação, Natural, Instrução, Raça e Etnia.
- Os dados geográficos foram subdivididos em duas tabelas:
 - Local de Residência do falecido - com os seguintes atributos: município, estado, região onde a pessoa morava.

⁷ valores padronizados para todo o país

⁸ o significado de cada campo pode ser obtido nos anexos (seção VIII.1), pois são equivalentes aos campos do sistema fonte.

- Local de Ocorrência do falecimento – que continha além dos atributos geográficos (cidade, UF e região); o atributo local de ocorrência, que corresponde ao local específico onde ocorreu o óbito (hospital, em casa, em via pública, etc.).
- A última dimensão contém a Causa básica da morte, e alguns atributos que poderiam ter colaborado na morte do indivíduo, são eles: Causa básica da morte, Obito_feto1, Obito_feto2, tipo violência, tipo de acidente, fonte de informação, acidente de trabalho, local do acidente, atestante, Assistência médica, Exame, Cirurgia, Necropsia.

A granularidade escolhida para o esquema foi de cada óbito individual. Como citado em [KIM96c], tabelas de fato que representam eventos são formadas apenas de chaves estrangeiras, com isso as consultas SQL construídas nesses esquemas ficam pouco legíveis. Assim, muitos projetistas, normalmente, acrescentam um campo “dummy” – que sempre tem o mesmo valor - no final da tabela de fatos (no caso da modelagem do SIM, o campo *Morto*) para facilitar a leitura dos SQLs.

O esquema desenvolvido deve ser utilizado para futuras tomadas de decisão. Com ele pode-se tentar descobrir locais onde deverão ser investidas mais verbas com saúde, já que certas questões e asserções podem ser respondidas, algumas delas são:

1. Qual o índice de mortes por determinada causa (uso da dimensão causa) em determinada região (uso da dimensão local ocorrência)? - Para a criação de programas preventivos para as doenças/causas com índices mais altos.
2. Que tipo de causa afeta mais determinadas classes sociais? – Tomando como base a ocupação do falecido (dimensão pessoal) ou dos seus pais (dimensão feto), é possível determinar a que classe social a pessoa pertence e que tipo de causa *mortis* (dimensão causa) é mais freqüente.
3. Se o índice de mortes em determinada região pós-cirurgia (uso do atributo cirurgia) ou pós-parto (uso dos atributos obito_feto1 e obito_feto2) for alto, esse pode ser um indicativo que esses procedimentos na região pesquisada devem ser melhorados.
4. Caso a quantidade de óbitos fetais for grande (dimensão feto), fazer novos investimentos com campanhas e realização de exames pré-natais.
5. Com o cruzamento de pessoas que tenham morrido por acidente de trabalho (atributo acid_trab) e a ocupação da mesma (atributo ocupação na dimensão

Comentário: Requisitos atendidos pela modelagem... Podem ser descobertas necessidades e oportunidades...

pessoal), podemos descobrir padrões (tarefa da Mineração de Dados) de profissões de maior risco e criar projetos/planos de segurança preventivos para diminuir o risco desses profissionais e evitar esse tipo de morte.

6. etc...

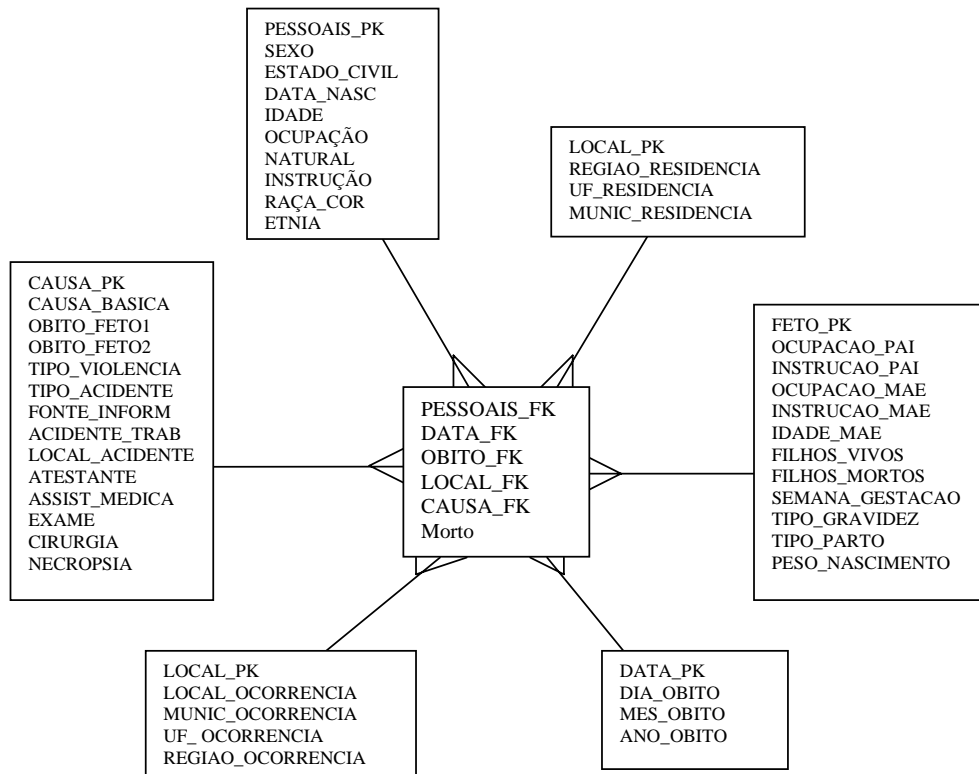


Figura 12 – Esquema Estrela do SIM

V.3 Ferramentas Escolhidas

Na fase de seleção da ferramenta, foram avaliadas as três ferramentas disponíveis na PUCRS:

- DBMiner E1.1⁹ – é a versão educacional da ferramenta desenvolvida pelo laboratório de pesquisa de sistemas de banco de dados inteligentes (*Intelligent Database Systems*) da Universidade de Simon Fraser no Canadá. Ela é mais

⁹ Maiores informações em: <http://www.dbminer.com/> e <http://db.cs.sfu.ca/DBMiner/tutorial.html>

utilizada por professores e institutos de pesquisa com a finalidade de ensinar conceitos e habilidades das tecnologias de DW e Data Mining;

- MS SQL/Server – é o BD de grande porte da Microsoft. Ele oferece algumas ferramentas para construção e manipulação de DW;
- PowerPlay e Impromptu – são ferramentas da Cognos desenvolvidas para a criação de DW.

Como banco de dados foi escolhido o MS SQL/Server, mas como os recursos analíticos do mesmo são restritos, principalmente os recursos voltados aos usuários-finais, para esta tarefa foram eleitas as ferramentas da Cognos. A não-escolha do DBMiner deveu-se ao fato da sua ferramenta estar mais voltada à tarefa de mineração e visualização, não oferecendo muitos recursos para a tarefa de *staging* e armazenamento.

V.3.1 MS SQL/Server 7.0

O MS SQL/Server é um SGBD relacional que faz parte da família BackOffice da Microsoft. Ele foi projetado para uso cliente/servidor, é acessado por aplicações que usam SQL – seguindo os padrões ANSI SQL-92 (*American National Standards Institute*) e FIPS 127-2 (*Federal Information Processing Standards* dos EUA), e é executado sobre Windows NT.

SQL/Server apoia SNMP (*Simple Network Management Protocol*), ODBC (*Open DataBase Connectivity*), e os maiores protocolos de comunicações abertos e provê integração com a Internet, replicação de dados e características de DW - através do *framework* oferecido para sua criação.

A versão 7.0 do SQL/Server foi melhorada para: suportar BD maiores – apesar de ele ainda ser um BD recomendado para uso em companhias de tamanho pequeno e médio; e fazê-lo levar a um DW – aumentando o paralelismo, melhorando seu poder de otimização em consultas mais complexas, capacidades de junções heterogêneas, utilitários para melhorar o desempenho e suporte para páginas de tamanho maior.

O *framework* apresentado pela Microsoft para DW inclui funções de extração e transformação de dados, armazenamento de dados tanto em BD relacionais como multidimensionais, acesso ao BD através de APIs (*application programming interfaces*), projeto de DW e gerenciamento do sistema [HUR99]. Ele também tem uma

arquitetura aberta e mecanismos para compartilhar os metadados. Muitos *wizards* são oferecidos com o produto para simplificar as funções mais comuns como manutenção de índices, desenvolvimento de tabelas de agregação e particionamento de dados.

A Figura 13 correlaciona os produtos e tecnologias do Microsoft Data Warehousing Framework [HUR99]:

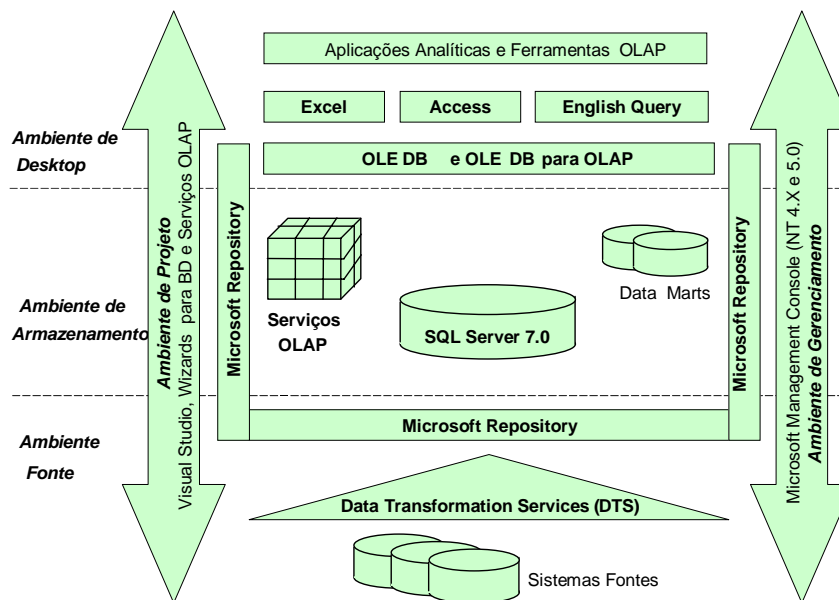


Figura 13 - Microsoft SQL Server 7.0 Data Warehousing Framework [HUR99]

O **Microsoft Repository (MR)** é uma camada que está acima do SQL/Server. Ela armazena metadados de uma variedade de sistemas, incluindo estruturas de dados OLTP e SAD. O MR consiste de um modelo de informações aberto extensível (OIM - *open information model*) e um conjunto de interfaces. Hoje, o OIM é o padrão de metadados adotado pelo Meta Data Coalition.

O **DTS package** (*Data Transformation Service package*) é uma ferramenta que vem no mesmo pacote do SQL/Server. O DTS é um serviço de importação e exportação de dados, que manipula arquivos texto, do MS Excel, do MS Access, DBase, FoxPro, Paradox, de bancos de dados via ODBC, etc. Ele também provê funções para transformação e carga dos dados para um DW ou DM.

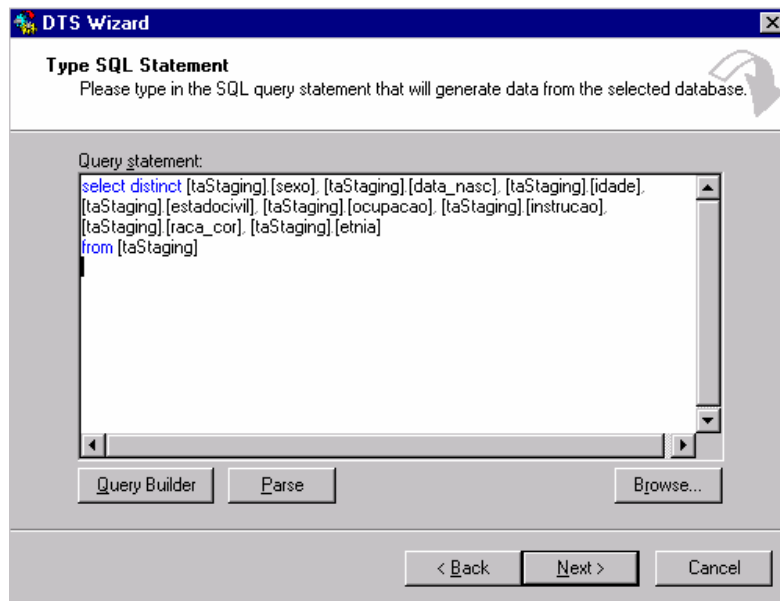


Figura 14 – DTS package

O **OLAP Services** foi uma solução construída para fazer análise multidimensional [MIC98a]. Ele inclui um servidor *middle-tier* que permite aos usuários realizar análises sofisticadas num grande volume de dados com um bom desempenho. Uma outra característica é a existência de uma *cache* no cliente e uma *engine* de cálculos chamada PivotTable Service, que ajuda aumentar o desempenho e diminuir o tráfego na rede, pois permite aos usuários administrarem análises mesmo quando estiverem desconectados da rede. O OLAP Services provê várias funcionalidades para SADs – junto a uma grande variedade de ferramentas e aplicações que apoiam serviços OLAP, através do Microsoft OLE DB para interfaces OLAP. Ele pretende facilitar o acesso a sofisticadas ferramentas analíticas e poder ajudar a reduzir os custos de DW.

Para o ambiente de desktop, são usadas as ferramentas:

- Excel 2000: que pode visualizar representações gráficas e tabulares de dados OLAP via OLE DB para OLAP.
- Access 2000: pode ser utilizado com *front-end* para o SQL/Server;
- Office 2000: inclui componentes Web que trazem algumas capacidades oferecidas pelo Access, Excel, PivotTables e gráficos para documentos HTML;

- English Query (EQ): é uma ferramenta para construção de aplicações, componente do SQL Server 7.0, onde o inglês é utilizado como linguagem de consulta. Utilizando um ambiente de desenvolvimento EQ, o desenvolvedor da aplicação mapeia termos da língua inglesa em objetos e relações do BD. Quando o usuário formula questões do tipo “How many green Paratis did we sell in the Porto Alegre dealerships in 1999?”, o EQ reconhece os termos na sentença que representam os objetos e relações do BD e geram as consultas em SQL que são enviadas para o SQL/Server.

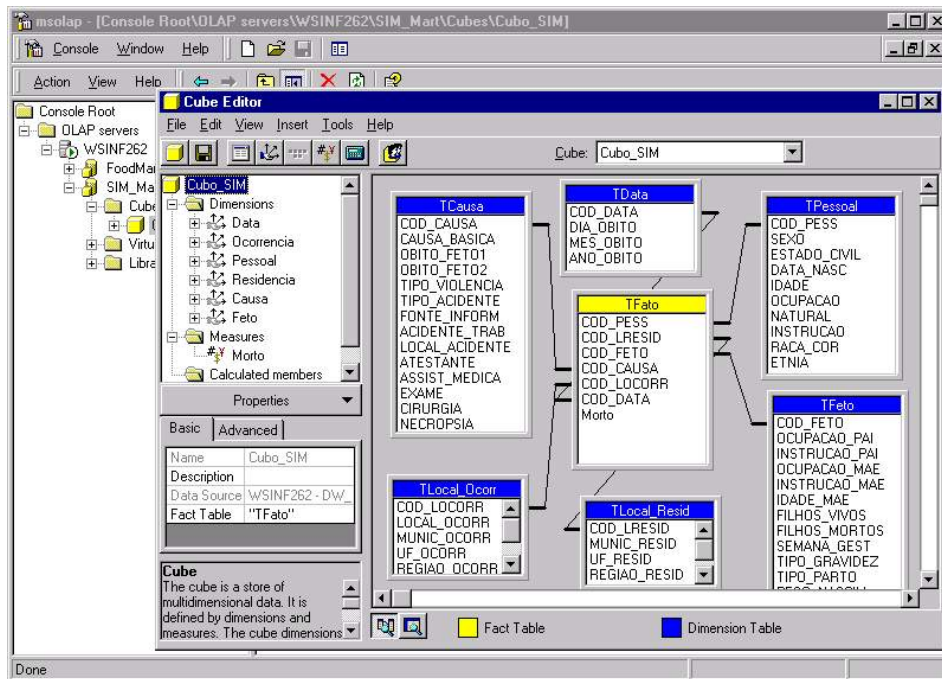


Figura 15 – OLAP Services

V.3.2 Ferramentas da Cognos

As ferramentas da Cognos para SAD e Business Intelligence provêm acesso tanto a data warehouses OLAP como relacionais. Elas suportam tanto as infraestruturas cliente/servidor quanto a Web. Elas permitem aos analistas o gerenciamento de relatórios e soluções de análise centralizadas.

As ferramentas oferecidas pela empresa para manipulação OLAP consistem em [COG99]:

PowerPlay - para análise e relatórios multidimensionais;

Impromptu – para consultas no nível de detalhes da transação;

Scenario – para análise automatizada de padrões e relacionamentos nos dados da empresa;

4Thought – para modelagem preditiva, onde todas as pessoas na corporação podem usar os dados da forma a satisfazer da melhor forma os diferentes requerimentos do negócio.

Dentre as ferramentas apresentadas, foram utilizadas nesse trabalho:

- PowerPlay, com o seu módulo Transformer;
- e alguns experimentos no Impromptu

Powerplay

PowerPlay é uma ferramenta OLAP que permite a manipulação e exploração dos dados do negócio de qualquer ângulo pelos gerentes, através de uma visão multidimensional (MOLAP). Essa ferramenta provê as seguintes funcionalidades:

- Exploração, análise, comparação;
- Filtragem e comparação dos dados online;
- Operações como *Slice-dice* e *drill down/up* (o cruzamento multidimensional pode ser feito em qualquer sentido e qualquer nível de detalhamento).

O **Transformer** é a ferramenta que mantém e controla as informações do negócio. A partir dele são criados os PowerCubes, que podem ser construídos e armazenados em plataformas NT e UNIX, e gravados em uma base de dados relacional, e que permitem aos usuários visualizar os dados da maneira que eles vêem o negócio. Como entrada de dados, ele admite consultas geradas pelo Impromptu, arquivos texto, planilhas e arquivos DBF.

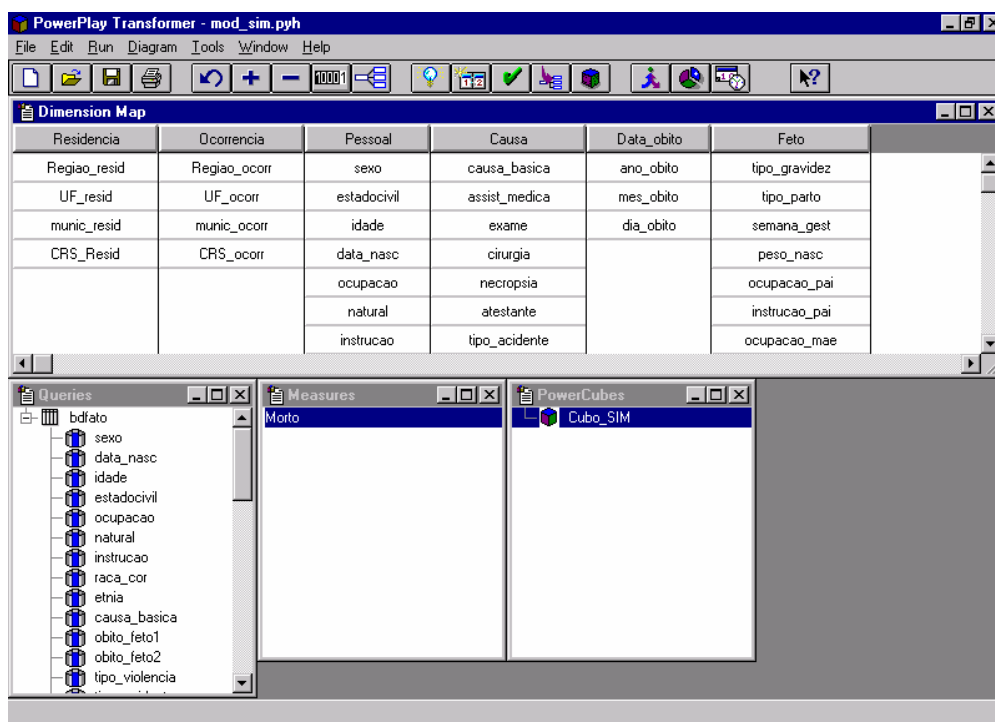


Figura 16 – PowerPlay Transformer

Os possíveis gráficos gerados pela ferramenta podem ser do tipo: linear, multilinear, barras, pizza, planilha, correlação, etc.

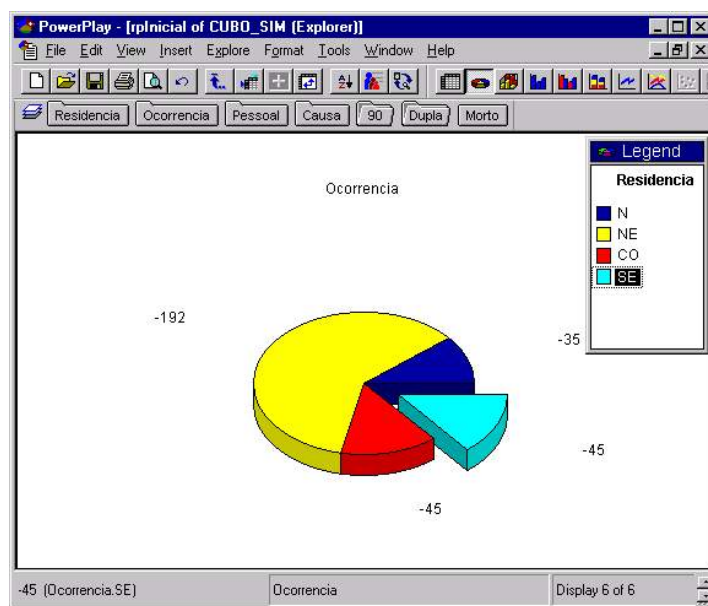


Figura 17 - PowerPlay

Dentre os bancos de dados suportados por essas ferramentas, temos:

- ♦ Oracle (7.3 e posteriores)
- ♦ Sybase SQL Server (11.2 e posteriores)
- ♦ Informix Dynamic Server (7.2 em diante) e XPS 8.21
- ♦ Microsoft SQL/Server (6.5 em diante)
- ♦ IBM DB2 (2.1) e Universal Database (5.X)
- ♦ Além disto, pode-se acessar qualquer outra base de dados através de drivers ODBC.

Impromptu

O Impromptu é uma ferramenta que permite a criação de consultas e geração de relatórios. O mecanismo chave do Impromptu são as informações do Catálogo, elas são visões das bases de dados existentes voltadas para os usuários, e refletem as regras do negócio e a estrutura da empresa.

O catálogo do Impromptu é um repositório de conhecimento empresarial e regras de acesso dos dados que organizam os dados usando a terminologia empresarial. Assim os usuários são separados das complexidades do BD, como a sintaxe SQL e a união das tabelas. Dados apresentando deste modo, facilitam a navegação através BD e a criação de relatórios pelos usuários; pois a partir do catálogo, o Impromptu é capaz de gerar automaticamente comandos SQL que buscam os dados para cada consulta ou relatório.

A arquitetura do Impromptu permite a distribuição desses catálogos; quando houver uma modificação nas regras do negócio, nas permissões de acesso, nas formas das consultas, etc., elas são automaticamente sentidas na organização.

Existe ainda o Impromptu Web Report, responsável pela publicação e distribuição de relatórios através da web. O agendamento de processamento remoto de uma consulta para horários pré-fixados, para casos de consultas pesadas, também é possível.

V.4 Criação do DW

V.4.1 Processo de Staging

Inicialmente foi criada uma única tabela no SQL/Server para receber os registros de todas as tabelas de óbito do SIM. Como alguns dados não pertenciam a uma base nacional¹⁰ - os valores variam de acordo com o estado, e outros só eram de controle interno¹¹, eles foram eliminados ainda na fase de projeto.

O início do processo de *Data Staging* foi realizado utilizando o MS Access, devido a uma maior familiarização com a ferramenta. Nele foram vinculadas as tabelas do SIM e a tabela utilizada no processo de Staging, e foram criadas consultas-inserção - para separar os campos que deveriam ir para a tabela de Staging. Nestas consultas foram feitas as conversões para a maioria dos campos de codificação própria e foram feitas uniões com as tabelas referenciadas no sentido de denormalizar o esquema ER. Ainda no MS Access foram criadas tabelas intermediárias para facilitar a descoberta da causa morte (nos óbitos após 95) e para associar a região os estados e municípios (já que o campo Região não aparece nas tabelas do SIM). O campo data_obito, contido em todas as tabelas de óbito, foi subdividido em três campos (dia, mês e ano) para facilitar na criação da dimensão tempo.

Uma hierarquia geográfica foi observada nas dimensões Local de Ocorrência e Local de Residência: Região → Estado → Cidade → CRS, e na dimensão Data_Obito, foi observada a hierarquia temporal: Ano → Mês → Dia. Nas demais dimensões, não existia nenhum tipo de hierarquia entre seus atributos.

V.4.2 População do DW

As tabelas de dimensão e fato foram criadas no SQL/Server. Para cada tabela dimensional, foram criadas chaves utilizando o recurso provido pelo BD (IDENTITY¹² sobre um atributo do tipo inteiro).

¹⁰ Informações como: área e bairro de residência, Regionais de Saúde de ocorrência e residência, etc.

¹¹ crítica e numexport.

¹² Isto significa que para cada nova linha na tabela, é associado o próximo valor de identificador (*identity*), que é igual ao último valor do identificador mais um.

A população das tabelas foi feita pelo DTS package, através das seguintes consultas:

- Tabela de Datas

```
Select distinct Dia_Obito, Mes_Obito, Ano_Obito
From TStaging;
```

- Tabela de Fetos

```
Select distinct Ocupacao_Pai, Instrucao_Pai, Ocupacao_Mae, Instrucao_Mae,
Idade_Mae, Filhos_Vivos, Filhos_Mortos, Semana_Gest, Tipo_Gravidéz,
Tipo_Partto, Peso_Nascim
From TStaging;
```

- Tabela de Local de Ocorrência

```
Select distinct Local_Ocorr, Munic_Ocorr, Uf_Ocorr, Regiao_Ocorr, CRS_Ocorr
From TStaging;
```

- Tabela de Local de Residência

```
Select distinct Munic_Resid, Uf_Resid, Regiao_Resid, CRS_Resid
From TStaging;
```

- Tabela de Dados Pessoais

```
Select distinct Sexo, Estado_Civil, Data_Nasc, Idade, Ocupacao, Natural,
Instrucao, Raca_Cor, Etnia
From TStaging;
```

- Tabelas de Causas

```
Select distinct Causa_Basica, Obito_Feto1, Obito_Feto2, Tipo_Violencia,
Tipo_Acidente, Fonte_Inform, Acidente_Trab, Local_Acidente, Atestante,
Assist_Medica, Exame, Cirurgia, Necropsia
From TStaging;
```

- Tabelas de Fatos (com algumas restrições)

```
Select C.cod_causa, D.cod_data, F.cod_feto, O.cod_locorr, R.cod_lresid,
P.cod_pess, S.morto
FROM taStaging S, TCausa C, TData D, TFeto F, TLocal_Ocorr O,
TLocal_Resid R, TPessoal P
WHERE
(D.dia_obito=S.dia_obito) AND
(D.mes_obito=S.mes_obito) AND
(D.ano_obito=S.ano_obito) AND
(O.local_ocorr=S.local_ocorr) AND
(O.uf_ocorr=S.uf_ocorr) AND
(O.munic_ocorr=S.munic_ocorr) AND
(O.regiao_ocorr=S.regiao_ocorr) AND
```

```

(R.munic_resid=S.munic_resid) AND
(R.uf_resid=S.uf_resid) AND
(R.regiao_resid=S.regiao_resid) AND
(F.ocupacao_pai=S.ocupacao_pai) AND
(F.instrucao_pai=S.instrucao_pai) AND
(F.ocupacao_mae=S.ocupacao_mae) AND
(F.instrucao_mae=S.instrucao_mae) AND
(F.idade_mae=S.idade_mae) AND
(F.filhos_vivos=S.filhos_vivos) AND
(F.filhos_mortos=S.filhos_mortos) AND
(F.semana_gest=S.semana_gest) AND
(F.tipo_gravidez=S.tipo_gravidez) AND
(F.tipo_parto=S.tipo_parto) AND
(F.peso_nascim=S.peso_nasc) AND
(C.causa_basica=S.causa_basica) AND
(C.obito_feto1=S.obito_feto1) AND
(C.obito_feto2=S.obito_feto2) AND
(C.tipo_violencia=S.tipo_violencia) AND
(C.tipo_acidente=S.tipo_acidente) AND
(C.Fonte_inform=S.fonte_inform) AND
(C.acidente_trab=S.acidente_trab) AND
(C.local_acidente=S.local_acidente) AND
(C.atestante=S.atestante) AND
(C.assist_medica=S.assist_medica) AND
(C.exame=S.exame) AND
(C.Cirurgia=S.Cirurgia) AND
(C.Necropsia=S.Necropsia) AND
(P.Sexo=S.Sexo) AND
(P.Estado_Civil=S.Estadocivil) AND
(P.Data_Nasc=S.Data_Nasc) AND
(P.Idade=S.Idade) AND
(P.Ocupacao=S.Ocupacao) AND
(P.Natural=S.Natural) AND
(P.Instrucao=S.Instrucao) AND
(P.Raca_Cor=S.Raca_Cor) AND
(P.etnia=S.etnia);

```

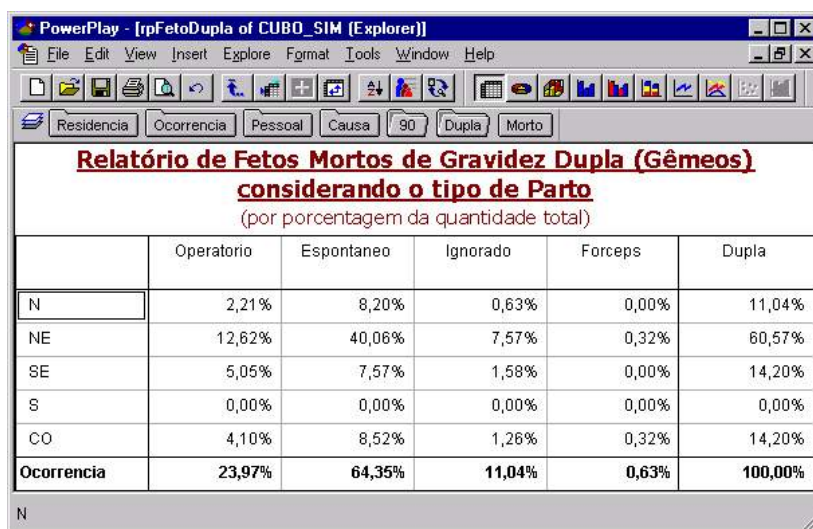
A partir da base (criada no MS SQL/Server) e a seleção das dimensões, o PowerPlay Transformer consegue gerar as categorias – que seriam cada uma das instâncias das dimensões, o catálogo de metadados e por fim cria o cubo (PowerCube). A partir do cubo criado, passa-se para a fase de manipulação do mesmo através do PowerPlay – que pode gerar relatórios/gráficos, realizar as operações analíticas, etc.

V.4.3 Manipulação do DW

As principais operações de navegação foram realizadas no cubo criado. Dentre elas:

- *Slice-dice*- Por exemplo, no caso de criar uma tabela cruzando duas dimensões e a dimensão escolhida para ocupar as colunas tinha mais atributos que a dimensão que ocupava as linhas, as mesmas era trocadas de lugar para uma melhor visualização dos resultados.
- *Drill Down/Up* – Foi utilizado o drill-down para descer em dimensões como as dimensões geográficas (locais de ocorrência e residência) e a dimensão tempo e ver os dados numa granularidade menor e utilizado o drill-up conseguir dados mais sumarizados.

Alguns relatórios e gráficos foram gerados na ferramenta, cruzando dados de algumas dimensões. Um exemplo pode ser visto na Figura 18, onde foram usadas as dimensões feto e ocorrência, e os dados foram filtrados pela dimensão data (foram considerados apenas os fetos mortos no ano de 1990 que nasceram de uma gravidez dupla):



	Operatorio	Espontaneo	Ignorado	Forceps	Dupla
N	2,21%	8,20%	0,63%	0,00%	11,04%
NE	12,62%	40,06%	7,57%	0,32%	60,57%
SE	5,05%	7,57%	1,58%	0,00%	14,20%
S	0,00%	0,00%	0,00%	0,00%	0,00%
CO	4,10%	8,52%	1,26%	0,32%	14,20%
Ocorrencia	23,97%	64,35%	11,04%	0,63%	100,00%

Figura 18 – Relatório gerado no PowerPlay

V.5 Dificuldades Encontradas

Um dos grandes problemas encontrados para a criação e manipulação dos dados no DW foram os recursos da máquina disponível para esses processos. Avistada num primeiro momento a necessidade de mais recursos, mais espaço em disco foi adquirido (habilitando uma partição não utilizada) e a memória virtual da máquina foi aumentada. Contudo, a base de dados fonte ocupava bastante espaço (mais de 1Gb), e como era necessária a criação e população da tabela utilizada no processo de *Data Staging*, o espaço disponibilizado foi quase que totalmente ocupado.

Devido a esse problema de recursos, um outro problema surgiu: a quantidade de dados manipulados. Numa primeira tentativa de carga de dados para a tabela de Staging, utilizando apenas os registros de óbitos fetais entre 90 e 97 (DOFET9X) e os óbitos não-fetais entre 90 e 92, o número de registros chegava a aproximadamente três milhões (ocupando \cong 1Gb), tornando o desempenho de consultas simples como a contagem do número de registros muito baixo e impossibilitando a ferramenta Transformer (da Cognos) de criar as dimensões do esquema estrela por falta de recursos.

Numa segunda tentativa optou-se por utilizar os dados apenas das tabelas de óbitos fetais entre 1990 e 1996. Com a utilização desses dados para popular a tabela de Staging, a mesma passou a conter um número bem menor registros (em torno de 282000), o que é um número razoável se comparado ao estudo de caso para um DW utilizando SQL/Server descrito em [SØR99] (que tem em sua maior tabela, ainda no sistema fonte, 168725 registros). Porém, da mesma maneira, o tempo e os recursos disponíveis foram de encontro a essa segunda abordagem.

Como o tempo hábil para a resolução do problema de configuração da máquina, criação e manipulação do DW era curto, e esse trabalho individual tem como finalidade maior o aprendizado de conceitos sobre DW e a sua criação e manipulação. Teve-se que restringir o universo de dados manipulados para pouco mais de 100000 registros e realizar todas as operações OLAP sobre esse número restrito de dados.

VI Comentários Finais

O uso de data warehouses em grandes (e até médias) corporações estão tornando-se cada vez mais comuns. Por isto, é de grande importância o seu estudo.

Ao longo desse trabalho, foram vistos alguns conceitos relacionados a tecnologia de DW que ajudaram a compreender o seu funcionamento. Porém, o uso de um estudo prático ajudou a uma maior familiarização e solidificação dos conceitos, que começaram a ser apreendidos no estudo teórico.

No decorrer do trabalho, foram feitas tarefas com o intuito de:

- + investigar em maior profundidade os ambientes de data warehouse e suas aplicações, através da realização de um estudo de caso, com o propósito de motivar a utilização da sua teoria, terminologia e conceitos fundamentais;
- + pesquisar e conhecer as ferramentas de projeto e implementação de DW, fazendo um estudo da(s) ferramenta(s) escolhida(s) para o desenvolvimento do projeto multidimensional;
- + Utilizar o data warehouse construído para o aprendizado de modelagem multidimensional e recursos analíticos.

Segundo o que foi visto durante esse trabalho, em especial na experiência realizada, o trabalho de criação do DW desde o início é uma tarefa bastante trabalhosa, principalmente se o(s) sistema(s) fonte(s) forem antigos e tiverem formas de armazenamento desconhecidas, e não estiverem disponíveis recursos tecnológicos que suportem esse processo.

Como extensões a este trabalho, duas frentes principais podem ser atacadas:

- ◆ Prática – Fazer uma melhor utilização dos recursos analíticos oferecidos pelo MS SQL/Server, ou fazer um estudo comparativo entre as ferramentas de DW;
- ◆ Teórica – Aprofundar mais os conhecimentos teóricos de tópicos que não foram abordados ou foram vistos de forma superficial neste trabalho.

VII Bibliografia

- [CAM98] CAMPOS, Maria Luiza; Filho, Arnaldo V. Rocha. **Data Warehouse** (Tutorial). UFRJ, 1998. Capturado em jul. 1999. Online. Disponível na Internet: <http://genesis.nce.ufrj.br/dataware/tutorial/tutorial.html>
- [COG99] COGNOS Inc. Capturado em ago. 1999. Online. Disponível na Internet: <http://www.cognos.com/>
- [DAT98?] DATASUS, 1998?. Capturado em ago. 1999. Online. Disponível na Internet: <http://www.datasus.gov.br/>
- [DEV97] DEVLIN, Barry. **Data warehouse: from architecture to implementation**. Addison Wesley Longman, 1997.
- [FAY96] FAYYAD, Usama M.; Piatetsky-Shapiro, Gregory; Smyth, Padhraic; et al. **Advances in Knowledge Discovery and Data Mining**. MIT Press. Mar, 1996.
- [FIR98] FIRESTONE, Joseph M. **Dimensional Modeling and ER Modeling in DW**. jun, 1998. Capturado em jul. 1999. Online. Disponível na Internet: <http://www.dkms.com/DMERDW.html>
- [HAN98] HAN, J.; Chee, S.; Tam, J. Y. C.; **Issues for On-Line Analytical Mining of Data Warehouses**; in: SIGMOD'96 - (DMKD'98) , 1998.
- [HUR99] Hurwitz Group, Inc. **The Microsoft Data Warehousing Framework: An End-to-End Data Warehouse Solution**. Maio, 1999. Capturado em set. 1999. Online. Disponível na Internet: <http://www.hurwitz.com> (in [Microsoft](#))
- [INM96a] INMON, William H. **Building the Data Warehouse**. John Wiley & Sons, 1996.
- [INM96b] _____. "The Data Warehouse and Data Mining", **Communications of the ACM**, v. 39 n. 11 pág. 49. Nov, 1996.
- [KIM95] KIMBALL, Ralph. **Is ER Model Hazardous to DSS?** DBMS. oct, 1995. Capturado em jul. 1999. Online. Disponível na Internet: <http://www.dbmsmag.com/9510d05.html>
- [KIM96a] _____. **The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses**, New York: J. Wiley, 1996.

- [KIM96b] _____. **Drilling Down Up and Across**. DBMS, v. 9 n. 3 Pág. 14. Mar, 1996. Capturado em jul. 1999. Online. Disponível na Internet: <http://www.dbmsmag.com/9603d05.html>
- [KIM96c] _____. **Factless Fact Table**. DBMS. Sept, 1996. Capturado em jul. 1999. Online. Disponível na Internet: <http://www.dbmsmag.com/9609d05.html>
- [KIM97] _____. **A Dimensional Modeling Manifesto**. DBMS, v. 10 n. 9 Pág. 59. Aug, 1997. Capturado em jul. 1999. Online. Disponível na Internet: <http://www.dbmsmag.com/9708d15.html>
- [KIM98a] _____, et al. **The Data Warehouse Lifecycle Toolkit: expert methods for designing, developing, and deploying data warehouses**. New York: John Wiley & Sons, 1998.
- [KIM98b] _____. **Help for Dimensional Modelings**. DBMS. Aug, 1998. Capturado em jul. 1999. Online. Disponível na Internet: <http://www.dbmsmag.com/9808d05.html>
- [LAM98?] LAMBERT, Bob. **Data Warehousing Fundamentals: What You Need To Know To Succeed**. 1998?. Capturado em ago. 1999. Online. Disponível na Internet: <http://www.datawarehouse.com/sigs/survival/article5.htm>
- [MIC98a] Microsoft Corporation. **Microsoft SQL Server 7.0 OLAP Services**. Technical Information, 1998. Capturado em ago. 1999. Online. Disponível na Internet: <http://www.microsoft.com/sql/bizsol/datawarehousing.htm>
- [MIC98b] MicroStrategy Inc. (<http://www.strategy.com/>). **Relational OLAP: An Enterprise-Wide Data Delivery Architecture**. in DM Review - White Paper, 1998. Capturado em set. 1999. Online. Disponível na Internet: <http://www.dmreview.com/>
- [MIC99] Microsoft Corporation, 1999. Capturado em ago. 1999. Online. Disponível na Internet: <http://www.microsoft.com/sql/>
- [PER98] PEREIRA, Walter A. L. **Data Warehouse**. Trabalho Individual II, Porto Alegre, 1998.
- [POE98] POE, Vidette; KLAUER, Patricia; BROBST, Stephen. **Building a Data Warehouse for Decision Support** 2nd edition. New Jersey: Prentice Hall PTR. 1998.
- [SAG91] SAGE, Andrew P. **Decision Support Systems Engineering**. John Wiley & Sons, 1991.
- [SCH99] SHOUTEN, Han. **Analysis and Design of Data Warehouses**. In: Proceedings of the International Workshop on Design and Management of Data warehouses. Jun, 1999. Capturado em ago. 1999. Online.

Disponível na Internet: <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-19/>

- [SØR99] SØRENSEN, Jens Otto; Alnor, Karl. **Creating a Data Warehouse using SQL/Server**. In: Proceedings of the International Workshop on Design and Management of Data warehouses. Jun, 1999. Capturado em ago. 1999. Online. Disponível na Internet: <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-19/>
- [WEI99] WEILBACH, JF Francois; Viktor, Herna L., **A Data Warehouse for Policy Making: A Case Study**. In: Proceedings of the 32nd. Hawaii International Conference on System Sciences, 1999.

VIII Anexos

VIII.1 Descrição das Tabelas do SIM¹³

Tabelas de Óbitos

NOME	DESCRIÇÃO
DO	Número da DO, sequencial por UF informante e por ano
CARTORIO	Código do cartório onde o óbito foi registrado
REGISTRO	Número de registro do óbito
DATAREG	Data do registro em cartório
TIPOBITO	Tipo de óbito, conforme a tabela: 1: Óbito fetal 2: Óbito não fetal
DATAOBITO	Data do óbito
ESTCIVIL	Estado civil, conforme a tabela: 0 ou 9: Ignorado 1: Solteiro 2: Casado 3: Viúvo 4: Separado judicialmente 5: Outro
SEXO	Sexo, conforme a tabela: 0 ou 9: Ignorado 1: Masculino 2: Feminino
DATANASC	Data de nascimento
IDADE	Idade, composta de dois subcampos. O primeiro, de 1 dígito, indica a unidade da idade, conforme a tabela a seguir. O segundo, de dois dígitos, indica a quantidade de unidades: 0: Idade ignorada; o segundo subcampo é também zero 1: Horas; o segundo subcampo varia de 00 a 23 2: Dias; o segundo subcampo varia de 00 a 29 3: Meses; o segundo subcampo varia de 00 a 11 4: Anos; o segundo subcampo varia de 00 a 99 5: Anos (mais de 99 anos); o segundo subcampo varia de 0 a 99;
LOCOCOR	Local de ocorrência do óbito, conforme a tabela: 0 ou 9: Ignorado 1: Hospital 2: Via pública 3: Domicílio

¹³ Informações retiradas do Help do SIM (Sistema de Informações de Mortalidade).

	4: Outro
CODIGO	Código do estabelecimento onde ocorreu o óbito, se LOCOCOR = 1. (não faz parte da base nacional. Só é válido a partir de 1995)
MUNIOCOR	Município de ocorrência do óbito, conforme codificação do IBGE.
MUNIRES	Município de residência.
BAIRES	Bairro de residência. (não faz parte da base nacional).
AREARES	Área de residência. (não faz parte da base nacional.)
OCUPACAO	Ocupação, conforme a Classificação Brasileira de Ocupações (CBO). <i>Tabela de ocupações</i>
NATURAL	Naturalidade, conforme a <i>Tabela de países</i> - Se for brasileiro, porém, o primeiro dígito contém 8 e os demais o código da UF de naturalidade.
INSTRUCAO	Instrução, conforme a tabela: 0 ou 9: Ignorado 1: Nenhuma 2: Primeiro grau 3: Segundo grau 4: Superior
OCUPPAI	Ocupação do pai, conforme a <i>Tabela de ocupações</i> .
INSTRPAI	Instrução do pai, conforme codificação de INSTRUCAO.
OCUPMAE	Ocupação da mãe, conforme a <i>Tabela de ocupações</i> .
IDADMAE	Idade da mãe, em anos.
INSTRMAE	Instrução da mãe, conforme codificação de INSTRUCAO.
FILHVIVOS	Número de filhos vivos.
FILHMORT	Número de filhos mortos, não incluindo o próprio.
SEMANGEST	Semanas de gestação, conforme as tabelas: Para os anos de 1979 a 1994: 0 ou 9: Ignorado 1: Menos de 20 semanas 2: 20 a 27 semanas 3: 28 e mais semanas Para os anos a partir de 1995: 0 ou 9: Ignorado 4: Menos de 21 semanas 5: 22 a 27 semanas 6: 28 a 36 semanas 7: 37 a 41 semanas 8: 42 semanas e mais
TIPOGRAV	Tipo de gravidez, conforme a tabela: 0 ou 9: Ignorado 1: Única 2: Dupla 3: Trílice 4: Mais de 3
TIPOPARTO	Tipo de parto, conforme a tabela: 0 ou 9: Ignorado 1: Espontâneo 2: Operatório 3: Fórceps

	4: Outro
PESONASC	Peso ao nascer, em gramas.
ASSISTMED	Indica se houve assistência médica, conforme a tabela: 0 ou 9: Ignorado 1: Com assistência 2: Sem assistência
ATESTANTE	Indica o médico atestante, conforme a tabela: 0 ou 9: Ignorado 1: Sim, atendeu ao falecido 2: Substituto 3: Instituto Médico Legal 4: Serviço de Verificação de Óbitos 5: Outro
EXAME	Indica se houve exame complementar, conforme a tabela: 0 ou 9: Ignorado 1: Sim 2: Não
CIRURGIA	Indica se houve cirurgia, conforme a tabela: 0 ou 9: Ignorado 1: Sim 2: Não
NECROPSIA	Indica se houve necrópsia, conforme a tabela: 0 ou 9: Ignorado 1: Sim 2: Não
OBITOFE1	Para óbitos femininos em idade fértil, indica se estava grávida no momento da morte, conforme a tabela: 0 ou 9: Ignorado 1: Sim 2: Não Este campo é válido somente a partir de 1995.
OBITOFE2	Para óbitos femininos em idade fértil, indica se esteve grávida nos 12 meses anteriores a morte, conforme a tabela: 0 ou 9: Ignorado 1: Sim 2: Não Este campo é válido somente a partir de 1995.
CAUSABAS	Causa básica, conforme a Classificação Internacional de Doença (CID), 9ª Revisão, para os dados de 1979 a 1995, e 10ª Revisão, para os dados de 1996. Observe-se que, quando a causa básica da 9ª Revisão não contiver a subcategoria de 4 algarismos, a quarta posição deve conter "X". Na mesma situação, na 10ª Revisão, a quarta posição deve conter um espaço em branco.
TIPOVIOL	Indica o tipo de violência, se cabível, conforme a tabela: 0 ou 9: Ignorado 1: Homicídio 2: Suicídio 3: Acidente 4: Outros tipos de violência Este campo é válido a partir de 1995.

TIPOACID	Indica o tipo de acidente, se cabível, conforme a tabela: 0 ou 9: Ignorado 1: Atropelamento 2: Demais acidentes de trânsito 3: Queda 4: Afogamento 5: Outros tipos de acidente Este campo é válido a partir de 1995.
FONTINFO	Fonte da informação, conforme a tabela: 0 ou 9: Ignorado 1: Boletim de Ocorrência 2: Hospital 3: Família 4: Outro
ACIDTRAB	Indica se foi acidente de trabalho, conforme a tabela: 0 ou 9: Ignorado 1: Sim 2: Não
LOCACID	Indica o local do acidente, se cabível, conforme a tabela: 0 ou 9: Ignorado 1: Via Pública 2: Domicílio 3: Outro 4: Local de trabalho
CRITICA	Dado para controle interno.
NUMEXPORT	Dado para controle interno.
CRSOCOR	Regional de Saúde de ocorrência. (não faz parte da base nacional).
CRSRES	Regional de Saúde de residência. (não faz parte da base nacional).
RACACOR	Indica a raça/cor do falecido: 1: Amarela 2: Branca 3: Indígena 4: Parda 5: Preta 0 ou 9: Ignorado
ETNIA	Código para etnia indígena, se RACACOR seja 3, conforme a <i>tabela de etnias</i> .
UFINFORM	UF que coletou e informou o óbito, conforme a <i>Tabela de UFs</i> .

Municípios

Descrição: Contém registros correspondentes a cada município, com seu código, nome e região administrativa (ERSA, delegacia, distrito etc).

NOME	DESCRIÇÃO
MUNICIPIO	Código do município segundo o IBGE, com a estrutura udddddv, onde: uu: código numérico da UF

	dddd: seqüencial do município dentro da UF v: dígito verificador
DESCRICAÇÃO	Nome do município. Eventualmente, este nome pode estar desatualizado, dada a frequência de alteração de nomes de municípios no Brasil, sendo isto uma competência municipal
REGIAO	Região administrativa estadual a que o município pertence. Eventualmente, pode também estar desatualizado, por ser esta divisão administrativa uma competência estadual.

País

Descrição: Contém registros correspondentes a cada país.

NOME	DESCRIÇÃO
CODIGO	Código do país
DESCRICAÇÃO	Nome do país

Estado

Descrição: Contém registros correspondentes a cada Unidade da Federação.

NOME	DESCRIÇÃO
SIGLA_UF	Sigla da Unidade da Federação
CODIGO	Código da Unidade da Federação, segundo o IBGE
DESCRICAÇÃO	Nome da Unidade da Federação

Ocupação

Descrição: Contém registros correspondentes a cada ocupação da Classificação Brasileira de Ocupações.

NOME	DESCRIÇÃO
CODIGO	Código da ocupação.
DESCRICAÇÃO	Descrição da ocupação.

Etnia

Descrição: Contém registros correspondentes a cada etnia indígena.

NOME	DESCRIÇÃO
CODIGO	Código da etnia
DESCRICAÇÃO	Nome da etnia

CID9

Descrição: Contém registros correspondentes a cada categoria da Classificação Internacional de Doenças - CID, 9ª Revisão (a 3 dígitos).

NOME	DESCRIÇÃO
DESCRICA0	Descrição da categoria.
CAUSAS	Código da categoria.

CID10

Descrição: Contém registros correspondentes a cada categoria e subcategoria da Classificação Internacional de Doenças, 10ª Revisão (a 3 e 4 caracteres).

NOME	DESCRIÇÃO
CID10	Código da categoria ou subcategoria, alinhado à esquerda, sendo o primeiro caractere uma letra, os dois seguintes dígitos e o quarto dígito ou espaço em branco. Note-se que este campo não contém o ponto.
OPC	Indicativo do sistema "cruz" e "asterisco": "+": classificação por etiologia "*": classificação por manifestação em branco: não tem dupla classificação
CAT	Se igual a "S", indica que é uma categoria (3 caracteres)
SUBCAT	Se igual a "S", indica que é uma subcategoria (4 caracteres). Como existem categorias sem subcategorias, pode haver códigos marcados como categoria e subcategoria simultaneamente.
DESCR	Descrição da categoria ou subcategoria, sendo que as 5 primeiras posições contém o seu código, com o ponto, e a 6ª posição está em branco.
RESTRSEXO	Indica se a categoria/subcategoria está limitada a homens ou mulheres, de acordo com os seguintes códigos: 1: válida apenas para homens; 3: válida apenas para mulheres; e 5: válida tanto para homens como para mulheres. As restrições correspondem às indicadas no Manual de Instrução da CID-10 (Volume II), às páginas 26 e 27.

CIDCAPXX

CIDCAP - Descrição: Contém registros correspondentes a cada capítulo da Classificação Internacional de Doenças, 9ª Revisão (a 3 dígitos).

CIDCAP10 - Descrição: Contém registros correspondentes a cada capítulo da Classificação Internacional de Doenças, 10ª Revisão (a 3 caracteres).

NOME	DESCRIÇÃO
DESCRICAÇÃO	Descrição do capítulo.
CAUSAS	Código das categorias vinculados a este capítulo, no formato iii-fff, onde: iii: categoria inicial fff: categoria final

CIDBRXX

CIDBR - Descrição: Contém registros correspondentes a cada elemento da Lista Tabular CID-BR2.

CIDBR-10 - Descrição: Contém registros correspondentes a cada elemento da Lista Tabular CID-BR.

CIDBR2 - Descrição: Contém registros correspondentes a cada elemento da Lista Tabular CID-BR2.

NOME	DESCRIÇÃO
DESCRICAÇÃO	Descrição do elemento.
CAUSAS	Código das categorias vinculados a este elemento, no formato iii-fff,iii- fff,..., onde: * iii: categoria inicial fff: categoria final

* As vírgulas permitem indicar várias faixas de categorias. Se as categorias iniciais e finais forem as mesmas, o traço e a categoria final são dispensadas. Exemplo: 500-506,508

VIII.2 Consulta gerada para população do Data Staging

A consulta básica para geração dos registros da tabela usada no Data Staging é a seguinte:

```
SELECT If([SEXO]='2','F',If([SEXO]='1','M','I')) AS Sexo,
If([ESTCIVIL]='1','Solteiro',If([ESTCIVIL]='2','Casado',If([ESTCIVIL]='3','Viúvo',If([ESTCIVIL]='4','Separado',If([ESTCIVIL]='5','Outro','Ignorado'))))) AS Estado_Civil,
Mid([DATANASC],7,2)+'/'+Mid([DATANASC],5,2)+'/'+Mid([DATANASC],1,4) AS Nascimento,
Obito.IDADE,
TAOCUPAC.DESCRICAÇÃO AS ocupacao,
TABPAIS.DESCRICAÇÃO AS Natural,
If([INSTRUCAO]='1','Nenhum',If([INSTRUCAO]='2','1º. grau',If([INSTRUCAO]='3','2º. grau',If([INSTRUCAO]='4','Superior','Ignorado'))))) AS Instrucao,
If([RACACOR]='1','Amarela',If([RACACOR]='2','Branca',If([RACACOR]='3','Indígena',If([RACACOR]='4','Parda',If([RACACOR]='5','Preta','Ignorado'))))) AS Raca,
tabetnia.DESCRICAÇÃO AS Etnia,
TaMunicResid.CIDADE AS Munic_Resid,
```

```

TaMunicResid.UF AS UF_Resid,
TaMunicResid.Regiao AS Regiao_Resid,
Obito.CRSRES,
Mid([DATAOBITO],5,2) AS Dia,
Mid([DATAOBITO],3,2) AS Mes,
Left([DATAOBITO],2) AS Ano,
If([LOCOCOR]='1','Hospital',If([LOCOCOR]='2','Via Publica',If([LOCOCOR]='3','Domicilio',
If([LOCOCOR]='4','Outro','Ignorado')))) AS LocOcorr,
TaMunicipio.CIDADE AS Mun_Ocorr,
TaMunicipio.UF AS UF_Ocorr,
TaMunicipio.Regiao AS Regiao_Ocorr,
Obito.CRSOCOR,
TAOCUP_pai.DESCRICAO AS ocupacao_pai,
If([INSTRPAI]='1','Nenhum',If([INSTRPAI]='2','1o. grau',If([INSTRPAI]='3','2o.
grau',If([INSTRPAI]='4','Superior','Ignorado')))) AS Instrucao_Pai,
TAOCUP_mae.DESCRICAO AS ocupacao_mae,
If([INSTRMAE]='1','Nenhum',If([INSTRMAE]='2','1o. grau',If([INSTRMAE]='3','2o.
grau',If([INSTRMAE]='4','Superior','Ignorado')))) AS Instrucao_Mae,
Obito.IDADEMAE,
Obito.FILHVIVOS,
Obito.FILHMORT,
Obito.SEMANGEST,
If([TIPOGRAV]='1','Unica',If([TIPOGRAV]='2','Dupla',If([TIPOGRAV]='3','Triplice',
If([TIPOGRAV]='4','+ de 3','Ignorado')))) AS Gravidez,
If([TIPOPARTO]='1','Espontaneo',If([TIPOPARTO]='2','Operatorio',If([TIPOPARTO]='3',
'Forceps', If([TIPOPARTO]='4','Outro','Ignorado')))) AS Parto,
Obito.PESONASC,
Obito.CAUSABAS,
If([OBITOF1]='1','S',If([OBITOF1]='2','N','I')) AS ObFeto1,
If([OBITOF2]='1','S',If([OBITOF2]='2','N','I')) AS ObFeto2,
If([TIPOVIOL]='1','Homicidio',If([TIPOVIOL]='2','Suicidio',If([TIPOVIOL]='3','Acidente',
If([TIPOVIOL]='4','Outra Violencia','Ignorado')))) AS TipoViol,
If([TIPOACID]='1','Atropelamento',If([TIPOACID]='2','Acidente Transito',If([TIPOACID]='3',
'Queda', If([TIPOACID]='4','Afogamento',If([TIPOACID]='5','Outro Acidente','Ignorado')))) AS
TAcidente,
If([FONTINFO]='1','Boletim Ocorrencia',If([FONTINFO]='2','Hospital', If([FONTINFO]='3',
'Familia', If([FONTINFO]='4','Outro','Ignorado')))) AS FonteInfo,
If([ACIDTRAB]='1','S',If([ACIDTRAB]='2','N','I')) AS AcidTrab,
If([LOCACID]='1','Via Publica',If([LOCACID]='2','Domicilio', If([LOCACID]='3','Outro',
If([LOCACID]='4','Local Trabalho','Ignorado')))) AS LocalAcid,

```



```

IIf([ATESTANTE]='1','atendeu o falecido',IIf([ATESTANTE]='2','Substituto', IIf([ATESTANTE]='3',
'IML', IIf([ATESTANTE]='4','Serv Verif Obitos', IIf([ATESTANTE]='5','Outro','Ignorado')))) AS
Atest,
IIf([ASSISTMED]='1','Com Assit',IIf([ASSISTMED]='2','Sem Assist','Ign')) AS Assit1,
IIf([EXAME]='1','S',IIf([EXAME]='2','N','I')) AS Exame,
IIf([CIRURGIA]='1','S',IIf([CIRURGIA]='2','N','I')) AS Cirurg,
IIf([NECROPSIA]='1','S',IIf([NECROPSIA]='2','N','I')) AS Necrop
FROM ((((((Tabela Obitos AS Obito LEFT JOIN TABPAIS ON Obito.NATURAL =
TABPAIS.CODIGO) LEFT JOIN tabetnia ON Obito.ETNIA = tabetnia.CODIGO) LEFT JOIN
TaUF ON Obito.UFINFORM = TaUF.CODIGO) LEFT JOIN TaMunicipio ON Obito.MUNIOCOR
= TaMunicipio.COD_MUN) LEFT JOIN TaMunicipio AS TaMunicResid ON Obito.MUNIRES =
TaMunicResid.COD_MUN) LEFT JOIN TAOCUPAC ON Obito.OCUPACAO =
TAOCUPAC.CODIGO) LEFT JOIN TAOCUPAC AS TAOCUP_pai ON Obito.OCUPPAI =
TAOCUP_pai.CODIGO) LEFT JOIN TAOCUPAC AS TAOCUP_mae ON Obito.OCUPMAE =
TAOCUP_mae.CODIGO;

```

Foram criadas três consultas a partir da consulta acima, pois ocorrem diferenças entre as tabelas de óbitos; mais especificamente, as tabelas do ano de 1996 fazem parte da 10ª Revisão do CID e contém tabelas com as causas-morte diferentes das demais tabelas do SIM, a partir de 1995 alguns campos foram adicionados no esquema da tabela de óbitos, as demais tabelas (entre 90 e 94) contêm o mesmo esquema.

VIII.3 Descrição da Tabela utilizada no processo de Data Staging

Nome	Tipo	Tamanho
sexo	Texto	1
data_nasc	Texto	10
idade	Texto	3
estadocivil	Texto	10
ocupacao	Texto	30
natural	Texto	30
instrucao	Texto	10
raca_cor	Texto	10
etnia	Texto	10
causa_basica	Texto	40
obito_feto1	Texto	1
obito_feto2	Texto	1
tipo_violencia	Texto	10
tipo_acidente	Texto	15
fonte_inform	Texto	15
acidente_trab	Texto	1

local_acidente	Texto	15
atestante	Texto	20
assist_medica	Texto	1
exame	Texto	1
cirurgia	Texto	1
necropsia	Texto	1
dia_obito	Texto	2
mes_obito	Texto	2
ano_obito	Texto	4
munic_resid	Texto	40
UF_resid	Texto	2
Regiao_resid	Texto	2
ocupacao_pai	Texto	30
instrucao_pai	Texto	10
ocupacao_mae	Texto	53
instrucao_mae	Texto	10
idade_mae	Número (Byte)	1
filhos_vivos	Número (Byte)	1
filhos_mortos	Número (Byte)	1
semana_gest	Texto	10
tipo_gravidez	Texto	10
tipo_parto	Texto	10
peso_nasc	Texto	6
local_ocorr	Texto	10
munic_ocorr	Texto	40
UF_ocorr	Texto	2
Regiao_ocorr	Texto	2
Morto	boolean	1

VIII.4 Descrição das Tabela do DW

VIII.4.1 Fato

Nome	Tipo	Tamanho
COD_PESS	Número (Longo)	4
COD_LRESID	Número (Longo)	4
COD_FETO	Número (Longo)	4
COD_CAUSA	Número (Longo)	4
COD_LOCORR	Número (Longo)	4
COD_DATA	Número (Longo)	4
Morto	Boolean	1

VIII.4.2 Dimensões

Causa

Nome	Tipo	Tamanho
COD_CAUSA	Número (Longo)	4
CAUSA_BASICA	Texto	40
OBITO_FETO1	Texto	1
OBITO_FETO2	Texto	1
TIPO_VIOLENCIA	Texto	10
TIPO_ACIDENTE	Texto	15
FONTE_INFORM	Texto	15
ACIDENTE_TRAB	Texto	1
LOCAL_ACIDENTE	Texto	15
ATESTANTE	Texto	20
ASSIST_MEDICA	Texto	1
EXAME	Texto	1
CIRURGIA	Texto	1
NECROPSIA	Texto	1

Data

Nome	Tipo	Tamanho
COD_DATA	Número (Longo)	4
DIA_OBITO	Texto	2
MES_OBITO	Texto	2
ANO_OBITO	Texto	4

Feto

Nome	Tipo	Tamanho
COD_FETO	Número (Longo)	4
OCUPACAO_PAI	Texto	30
INSTRUCAO_PAI	Texto	10
OCUPACAO_MAE	Texto	30
INSTRUCAO_MAE	Texto	10
IDADE_MAE	Número (Byte)	1
FILHOS_VIVOS	Número (Byte)	1
FILHOS_MORTOS	Número (Byte)	1
SEMANA_GEST	Texto	10
TIPO_GRAVIDEZ	Texto	10

TIPO_PARTO	Texto	10
PESO_NASCIM	Texto	6

Local Ocorrência

Nome	Tipo	Tamanho
COD_LOCORR	Número (Longo)	4
LOCAL_OCORR	Texto	10
MUNIC_OCORR	Texto	40
UF_OCORR	Texto	2
REGIAO_OCORR	Texto	2

Local Residência

Nome	Tipo	Tamanho
COD_LRESID	Número (Longo)	4
MUNIC_RESID	Texto	40
UF_RESID	Texto	2
REGIAO_RESID	Texto	2

Pessoal

Nome	Tipo	Tamanho
COD_PESS	Número (Longo)	4
SEXO	Texto	1
ESTADO_CIVIL	Texto	10
DATA_NASC	Texto	10
IDADE	Texto	3
OCUPACAO	Texto	30
NATURAL	Texto	30
INSTRUCAO	Texto	10
RACA_COR	Texto	10
ETNIA	Texto	10