



Universidade Federal de Itajubá

# **CONCEPÇÃO DE UM DATA WAREHOUSE E CRM PARA O SISTEMA DE OUVIDORIA DA ANEEL**

André Ruelli

Orientador: Prof. Germano Lambert Torres

Co-orientador: Prof. Manoel Eduardo Miranda Negrisoni

Dissertação apresentada à  
Universidade Federal de Itajubá,  
para obtenção do título de Mestre  
em Engenharia Elétrica.

Dezembro/02

*“Dedico este trabalho a Jesus Cristo e a minha filha Isabela, pela paciência e compreensão da minha ausência.”*

# SUMÁRIO

|   |     |
|---|-----|
| Resumo .....  | 05  |
| Abstract .....  | 06  |
| <br>  |     |
| <u>Capítulo 1</u> - Introdução e Objetivos                                  |     |
| 1.1 – CRM e Data Warehouse no Contexto Atual. ....                          | 07  |
| 1.2 - Avaliação da Estrutura da Ouvidoria ANEEL .....                       | 08  |
| 1.3- A Necessidade da ANEEL .....   | 12  |
| <br>  |     |
| <u>Capítulo 2</u> – Descrição da Base de Dados e Sistemas de Replicação     |     |
| 2.1- Descrição da Base de Dados em SQL na ANEEL.....                        | 14  |
| 2.2- Descrição da Replicação da Base de Dados em Oracle na Atento....       | 17  |
| 2.3- Descrição da Replicação da Base de Dados em SQL na Contax.....         | 18  |
| 2.4- Avaliação do Volume de Dados Existentes na ANEEL.....                  | 19  |
| <br>  |     |
| <u>Capítulo 3</u> – O Conceito de Data Warehouse e CRM                      |     |
| 3.1- Evolução dos Sistemas de Informação .....                              | 23  |
| 3.2- Ambiente de Data Warehouse .....                                       | 29  |
| 3.3- Arquitetura .....  | 37  |
| 3.4- Granularidade .....  | 51  |
| 3.5- Modelagem de Dados de um Data Warehouse .....                          | 55  |
| 3.6- Desenvolvimento de um Data Warehouse.....                              | 69  |
| 3.7- Extração de Informações do Data Warehouse .....                        | 82  |
| 3.8- CRM .....  | 89  |
| <br>  |     |
| <u>Capítulo 4</u> – Concepção do Projeto DW e CRM para a Ouvidoria da ANEEL |     |
| 4.1- Descrição do Projeto .....   | 93  |
| 4.2- Descrição da Ferramenta Utilizada .....                                | 93  |
| 4.3- Descrição da Estrutura Utilizada .....                                 | 95  |
| 4.4- Documentação do Projeto .....  | 96  |
| 4.5- Evolução do Data Mart para Atender Necessidades de CRM .....           | 111 |
| 4.6- Custos de Implementação do Projeto Data Mart .....                     | 112 |

|                                     |         |
|-------------------------------------|---------|
| <u>Capítulo 5</u> – Conclusão ..... | 114     |
| 5.1- Dificuldades Encontradas ..... | 116     |
| 5.2- Projetos Futuros.....          | 116     |
| <br>Referências .....               | <br>119 |
| <br>Anexos .....                    | <br>122 |

## **RESUMO**

A organização de informações representa hoje um dos maiores desafios para as organizações que geram grandes conjuntos de dados. A tendência dos volumes de dados é crescer ainda mais devido ao avanço de tecnologias incorporadas, praticamente em todos os setores das Instituições.

Atualmente, há toda uma recente indústria dedicada a Data Warehouse e Sistemas de Suporte à Decisão (DSS), destinada a solução de problemas relacionados ao armazenamento e integridade dos dados, como o caso encontrado na base de dados da Ouvidoria da ANEEL, objeto deste trabalho. O mundo dos sistemas de informação amadureceu ao ponto em que, muito mais do que sistemas de processamentos operacionais, são considerados válidos e úteis para o apoio gerencial.

Neste trabalho, além de ser apresentado todo o conceito de Data Warehouse e CRM, será apresentado um modelo desenvolvido para o Sistema de Ouvidoria da ANEEL, visando ao armazenamento, transformação e apresentação dos dados em uma formatação acessível à gerência.

Esta dissertação reforça ainda mais o conceito de que Data Warehouse é um dos avanços mais relevantes no projeto de banco de dados e, também, a solução para o problema de armazenamento de dados históricos.

## **ABSTRACT**

The organization of information represents today one of the greatest challenges for any corporation that generates great amount of data. The tendency of data volume is to increase even more due to the advance of incorporated technology, practically in all sectors of the institutions.

Nowadays, there is a brand new segment of the industry dedicated to Data Warehouse, and Decision Support Systems, aiming the solution of problems related to storing and the integrity of data, as the case found in the database of the Customer Service at ANEEL, main subject of this work. The world of information system has reached such a maturity level that they are considered only operational processing systems but valid and a lot useful to the management support.

In this work, in addition to presenting the whole concept of Data Warehouse and CRM, it will be presented a model developed but the Customer Service at ANEEL, aiming the storage, transformation and the presentation of data in such a way that is easily accessible by the management.

This dissertation reinforces even more the concept that Data Warehouse is one the the most relevant advances in the database project, and also the only solution to the problem of storing historical data.

# Capítulo 1

## INTRODUÇÃO E OBJETIVOS

### 1.1 CRM E DATA WAREHOUSE NO CONTEXTO ATUAL

Data Warehouse é um conceito que proporciona soluções de negócios às organizações e envolve a integração de diversas tecnologias e suas implementações, destinado a uma nova concepção de armazenamento e disposição de dados. O Data Warehouse proporciona um novo entendimento a questões relacionadas a sistemas de apoio a decisão. A década de 90 trouxe um crescente problema de supersaturação ao mundo da ciência, dos negócios e do governo. Nossa capacidade de coletar e armazenar dados ultrapassou em muito nossa condição de analisar, resumir e extrair conhecimento destes dados.

Os métodos tradicionais de análise de dados, baseados principalmente no controle direto dos dados, simplesmente não permitem a manipulação de conjuntos volumosos de dados. Embora a tecnologia voltada para o armazenamento de dados tenha nos oferecido as ferramentas básicas para armazenar e extrair com eficiência grandes conjuntos de dados, a questão crucial é como ajudar um gerente a entender e analisar esses conjuntos. Para lidar com esta questão, faz-se necessária uma nova geração de ferramentas inteligentes para extração automatizada de dados e descoberta de conhecimento.

Neste contexto, o sistema de Ouvidoria da ANEEL aparece como um gerador de dados que são originados de ligações efetuadas por consumidores de energia elétrica ao telefone 0800 da Instituição. No caso da base de dados da Ouvidoria pode-se, facilmente através de ferramentas destinadas a extração de dados, identificar padrões e disponibilizar conhecimentos para a superintendência responsável. No item a seguir será apresentada a estrutura da Superintendência de Mediação Administrativa Setorial – SMA, responsável pela Ouvidoria da ANEEL.

Por outro lado, existe um conceito chamado Customer Relationship Management – CRM, conhecido pela tradução como “Gerência do Relacionamento com

o Cliente” que se encaixa perfeitamente nos objetivos da ANEEL, destinado a entender e influenciar o comportamento dos clientes, por meio de comunicações significativas para melhorar e estreitar a relação entre a Instituição e os consumidores finais de energia elétrica.

O CRM pode focalizar diretamente os objetivos da ANEEL, entretanto, nada seria possível se não existisse um mecanismo que possibilite a apresentação das informações necessárias para suprir a carência de conhecimento por parte da Instituição.

Obter sucesso na utilização das tecnologias de relacionamento é um processo de aprendizagem em contínua evolução. Os estágios que compõem o processo de maturidade do gerenciamento são definidos em:

- Início – Montagem e aprendizagem
- Crescimento – Aplicação, tecnologia e disseminação dos dados
- Controle – Intervenção e posicionamento gerencial
- Integração – Info-estrutura empresarial
- Distribuição – Divisão interfuncional
- Estratégia – Táticas, planejamento.

Nota-se que os dois primeiros estágios estão diretamente ligados ao sucesso da implantação de um Data Warehouse onde, logo na fase inicial de um projeto, já se deve conhecer a necessidade das informações. No capítulo 4 será mostrado um breve levantamento das informações necessárias para gerar um modelo de dados no Data Warehouse, formatado para atender às exigências de um CRM para a ANEEL.

## **1.2 AVALIAÇÃO DA ESTRUTURA DA OUVIDORIA DA ANEEL**

A Agência Nacional de Energia Elétrica - ANEEL tem a seguinte estrutura organizacional de acordo com o DECRETO 2.335 / 1997:

I - Diretoria:

a) Secretaria-Geral

b) Gabinete do Diretor-Geral

II – Procuradoria-Geral

III – Superintendências de Processos Organizacionais

Neste nível se encontra a Superintendência de Mediação Administrativa Setorial - SMA, responsável por uma das atribuições mais importantes da agência que é “o atendimento das reclamações de agentes e consumidores”.

Para tornar possível a eficiência da operacionalização desta atribuição, a SMA conta com uma estrutura montada de acordo com os grupos definidos a seguir:

a) Serviço de Atendimento ao Consumidor – SAC

Serviço de teleatendimento do 0800 da ANEEL, realizado pela empresa ATENTO, especializada em atendimento a consumidores.

b) Grupo de Apoio Operacional – GAO

Responsável pelo tratamento de Solicitações de Ouvidoria, encaminhadas pelos consumidores, através do SAC - Serviço de Atendimento ao Consumidor, e-mail, internet, carta e fax.

c) Grupo de Apoio Técnico – GAT

Responsável pela análise e parecer técnico dos documentos encaminhados pelas equipes do GAO e do COD.

d) Controle Documental – COD

Responsável pelo recebimento, expedição e circulação de documentos na SMA e ANEEL, através do Sistema Integrado de Controle de Processos e Documentos - SIC.

e) Apoio às Agências Estaduais – AGE

Responsável pela uniformização das ações de elaboração e acompanhamento do Plano de Atividades e Metas, das Agências descentralizadas, em cooperação com a SMA.

f) Serviços de Informática – INF

Responsável pelo apoio técnico operacional e manutenção do Sistema de Gerenciamento de Ouvidoria SGO.

É importante destacar cada grupo e suas necessidades. Como exemplo, verifica-se que o grupo AGE, responsável pelo controle das agências conveniadas, já apresenta uma necessidade de informação relacionada à produtividade que pode facilmente ser obtida através de dados históricos armazenados em um Data Warehouse.

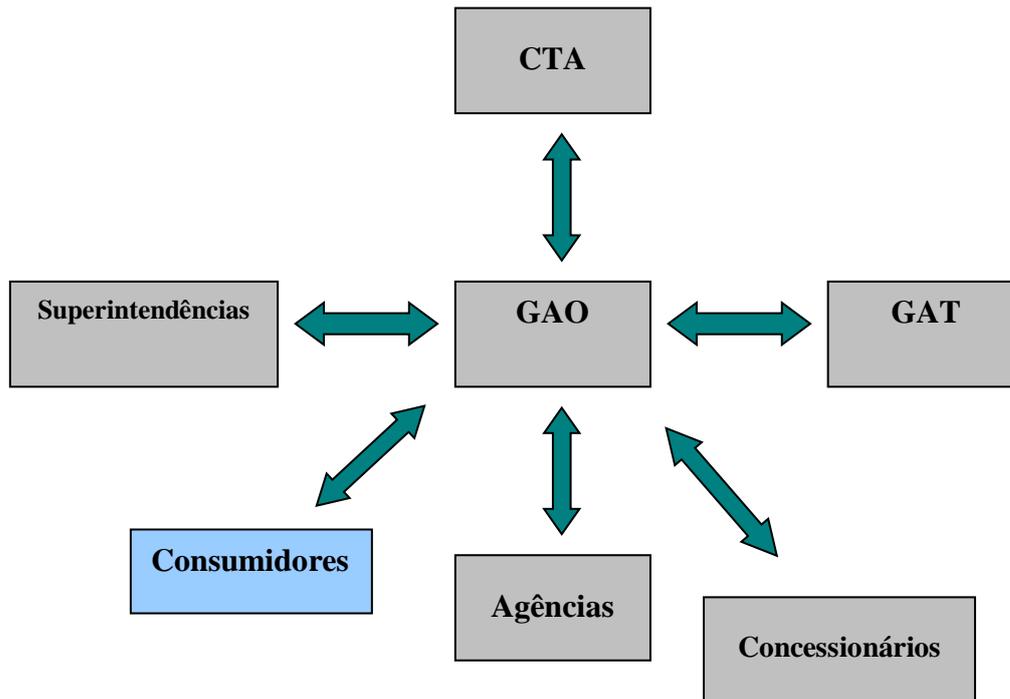
No capítulo 3 serão apresentadas várias tecnologias que compõem um projeto de Data Warehouse e CRM, proporcionando a apresentação de informações completas e precisas.

### **1.2.1 Descrição do Sistema SGO**

O Sistema de Gestão da Ouvidoria - SGO estabelece um elo entre os consumidores, as concessionárias, as agências reguladoras estaduais e a ANEEL, tornando possível registrar as solicitações de informações, reclamações e denúncias, obtendo-se elementos para avaliação, auditoria e intermediação entre usuários e concessionários de energia, bem como o gerenciamento de informações do setor elétrico.

O processo de tratamento de informações é agilizado através das agências estaduais, utilizando tecnologia de ponta (Internet), de forma a centralizar as informações na ANEEL. Relatórios gerenciais/administrativos atuais, permitem a análise da situação em nível nacional.

A figura 1.1 mostra o fluxo simplificado do trâmite de uma solicitação de ouvidoria registrada na CTA ANEEL e o anexo 1 contém o fluxo completo do tratamento da informação.



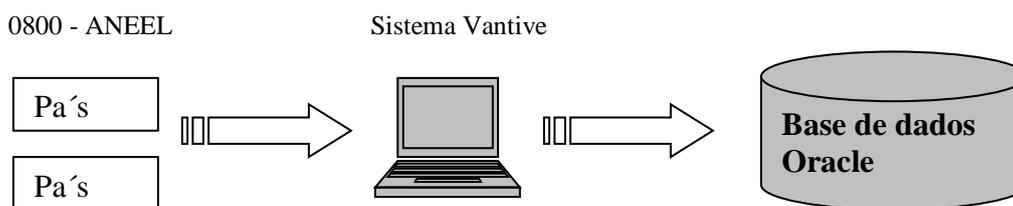
**Fig. 1.1 – Fluxo simplificado do SGO**

Estão integradas ao sistema as agências estaduais ARCON (PA), ARSEP (RN), ARCE (CE), AGERBA (BA), CSPE (SP), AGERS (RS), AGER (MT), ARPE (PE), ARSAL (AL), AGR (GO), AGEPAN (MS), AGEEL (PB), ARSAM (AM), as concessionárias destes estados e outras concessionárias de estados ainda sem agências conveniadas.

### **1.2.2 Descrição da Estrutura da CTA**

Os serviços de teleatendimento da ANEEL são executados pela empresa Atento Brasil S/A especializada no ramo de call center, com uma infra-estrutura de 80 Postos de Atendimento – PA’s e 30 Unidades de Respostas Audíveis - URA’s.

A Atento possui um sistema de atendimento chamado “Vantive” que registra as solicitações em uma base de dados Oracle. No capítulo 2 será apresentada a rotina de importação necessária para converter esses registros e exportá-los para a base de dados da ANEEL.



**Fig. 1.2 – Sistema CTA ANEEL**

Recentemente, uma nova central de teleatendimento entrou em funcionamento apenas para atender ligações efetuadas por consumidores do estado de São Paulo. A Contax S/A conta com uma estrutura semelhante à da Atento do Brasil S/A, entretanto sua base de dados é padronizada com a existente na ANEEL, fator que torna mais fácil a troca de dados (replicação) com a ANEEL.

### **1.3 A Necessidade da ANEEL**

Apresentado anteriormente, o SGO foi desenvolvido para atender as necessidades da Ouvidoria da ANEEL, buscando a qualidade dos serviços oferecidos à sociedade. Entretanto, no que se refere a informações estratégicas e mais completas o sistema torna-se insuficiente e a base de dados relacional pobre em informações.

O Data Warehouse para Ouvidoria complementa os serviços oferecidos pelo sistema de informação existente, gerando informações mais sólidas e estáveis devido à preservação dos dados históricos importantes, armazenados em ambiente estático.

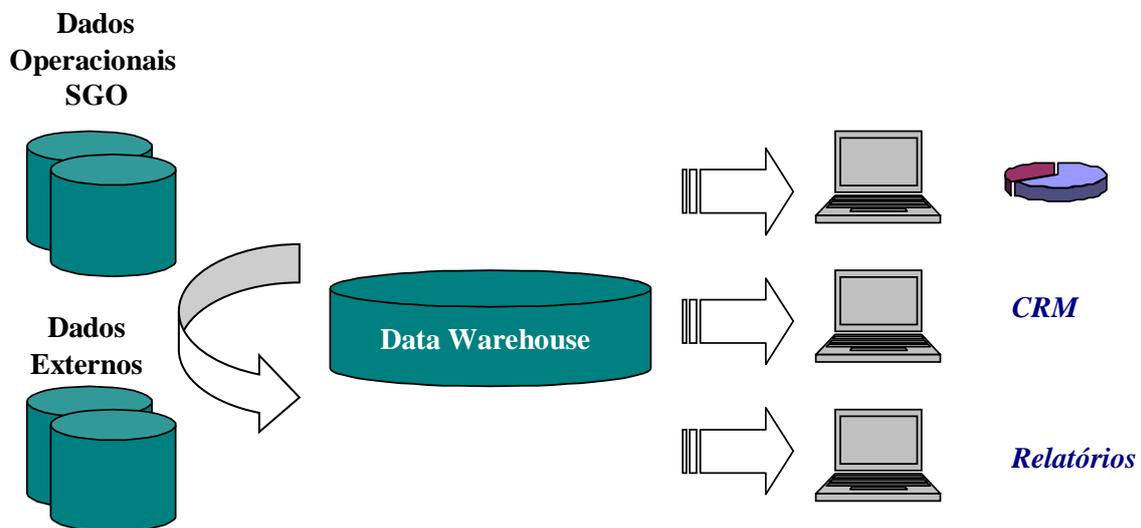
Os dados apresentados atualmente nos relatórios de Ouvidoria são extraídos da base de dados operacional (relacional) existente, ocasionando uma perda nas informações históricas devido ao expressivo volume de transações diárias no banco de dados da Ouvidoria.

Logo, um sistema de apoio à decisão concretizaria um novo mecanismo de apresentação de informações gerenciais para a ANEEL, através de um único repositório de dados denominado Data Warehouse (armazém de dados), que vem se mostrando a

melhor opção para transformar dados residentes em grandes bases em informações estratégicas.

No capítulo 4, serão apresentados alguns relatórios gerados a partir do ambiente Data Warehouse com a utilização de um “front end” desenvolvido em ferramenta OLAP.

Resumindo, o objetivo deste trabalho é criar um ambiente estático para armazenar dados históricos extraídos a partir da base de dados relacional do Sistema de Gestão da Ouvidoria e tornar possível a apresentação de informações úteis para o gerenciamento do processo, como representado na figura 1.3.



**Fig. 1.3 – Objetivo final do projeto**

## Capítulo 2

### DESCRIÇÃO DA BASE DE DADOS E SISTEMAS DE REPLICAÇÃO

#### 2.1 Descrição da Base de Dados em SQL na ANEEL

A base de dados da Ouvidoria da ANEEL armazena todas as solicitações registradas pelas centrais de teleatendimento através de importação de dados, pelas agências conveniadas através do SGO e pela internet através do site da ANEEL. Este armazenamento é efetuado em plataforma Microsoft SQL 2000 em um servidor de grande porte residente na própria agência.

Esta base é composta por 59 tabelas, com diferentes conteúdos: movimentação diária, cadastro, auxiliares e controle do Sistema. O anexo 2 mostra o modelo de dados completo, destacando em amarelo as tabelas que são alimentadas diariamente.

O conteúdo de maior importância para a Ouvidoria está armazenado nestas quatro tabelas básicas, que são replicadas diariamente entre as centrais de teleatendimento (ATENTO-DF e CONTAX-SP). Para entender a lógica do modelo, veja a seguir a estrutura das tabelas básicas:

DBO.SOLICITACAO

|                               |          |   |
|-------------------------------|----------|---|
| Numero_da_Solicitacao         | varchar  | Armazena o número da solicitação                |
| Codigo_do_Concessionario      | varchar  | Armazena o código da concessionária             |
| Codigo_do_Tipo_da_Unidade     | varchar  | Armazena a classificação da unidade consumidora |
| Anonimo                       | bit      | Registra uma solicitação anônima                |
| Sigilo                        | bit      | Registra uma solicitação com sigilo             |
| Codigo_Sequencial             | int      | Armazena a natureza da reclamação               |
| Numero_da_Unidade_Consumidora | varchar  | Armazena o número da unidade consumidora        |
| Registro_Novo                 | bit      | Registro novo                                   |
| Registro_Alterado             | bit      | Registro alterado                               |
| Agencia_Responsavel           | varchar  | Armazena o código da agência responsável        |
| Data_Criacao                  | datetime | Armazena a data de criação                      |
| Data_Alteracao                | datetime | Armazena a data da alteração                    |

|                          |          |  |
|--------------------------|----------|--|
| Unidade_Responsavel      | char     | Armazena o número da unidade consumidora             |
| Protocolo_Concessionario | varchar  | Armazena o número do processo na concessionária      |
| codigo_do_tipo_decisao   | int      | Armazena a procedência da solicitação                |
| Codigo_do_Tecnico        | int      | Armazena o código do técnico                         |
| Codigo_do_BO             | int      | Armazena o código do BO                              |
| Codigo_da_situacao       | int      | Armazena o código da situação                        |
| Prazo                    | datetime | Armazena o prazo concedido a concessionária          |
| Primeiro_Contato         | datetime | Armazena a data do primeiro contato                  |
| Situacao_Concessionaria  | int      | Registra a abertura da solicitação na concessionária |
| Situacao_agencia         | int      | Registra a abertura da solicitação na agência        |
| Codigo_do_CTA            | int      | Armazena o código da CTA origem                      |
| Codigo_Sup               | varchar  | Armazena o código da superintendência responsável    |

#### DBO.ENVOLVIDO\_NA\_SOLICITACAO

|                       |          |                                  |
|-----------------------|----------|----------------------------------|
| Numero_da_Solicitacao | varchar  | Armazena o número da solicitação |
| Codigo_do_Envolvido   | int      | Armazena o código do envolvido   |
| Codigo_do_Tipo        | bit      | Armazena o tipo de envolvido     |
| Registro_Alterado     | bit      | Registra alteração               |
| Data_Criacao          | datetime | Armazena a data de criação       |
| Data_Alteracao        | datetime | Armazena a data de alteração     |

#### DBO.ENVOLVIDO

|                      |          |                                 |
|----------------------|----------|---------------------------------|
| Codigo_do_Envolvido  | int      | Armazena o código do envolvido  |
| Pais                 | int      | Armazena o País                 |
| UF                   | varchar  | Armazena o estado               |
| Classe_de_Envolvido  | smallint | Armazena a classe do envolvido  |
| Codigo_do_Tratamento | int      | Armazena o tratamento           |
| PJ                   | bit      | Armazena o CNPJ                 |
| Nome                 | varchar  | Armazena o nome do envolvido    |
| CPF                  | char     | Armazena o CPF                  |
| Orgao_Expedidor      | varchar  | Armazena o órgão                |
| Empresa              | varchar  | Armazena a empresa do envolvido |
| Logradouro           | varchar  | Armazena a rua                  |
| Complemento          | varchar  | Armazena o complemento          |
| Bairro               | varchar  | Armazena o bairro               |

|                           |          |  |
|---------------------------|----------|--|
| Cidade                    | varchar  | Armazena a cidade  |
| CEP                       | char     | Armazena o CEP   |
| Telefone_Residencial      | varchar  | Armazena telefone  |
| Telefone_Comercial        | varchar  | Armazena telefone  |
| Telefone_Celular          | varchar  | Armazena telefone  |
| Fax                       | varchar  | Armazena o número do fax   |
| E_mail                    | varchar  | Armazena o e-mail  |
| Hora_de_Registro          | datetime | Armazena a hora de criação   |
| Forma_de_Contato_Desejada | int      | Registra a forma de contato desejada   |
| Registro_Alterado         | bit      | Registra a alteração   |
| Data_Alteracao            | datetime | Armazena a data de alteração   |
| Data_Criacao              | datetime | Armazena a data de criação   |
| codigo_dosexo             | char     | Armazena o código do sexo  |
| codigo_dofaixa_etaria     | char     | Armazena a faixa etária  |
| codigo_da_escolaridade    | char     | Armazena a escolaridade  |
| codigo_da_ocupacao        | char     | Armazena e ocupação  |
| CodigoMunicipio           | numeric  | Armazena o código do município – criado para facilitar a busca da dimensão local do DW |

#### DBO.CONTATO

|                                |          |                                       |
|--------------------------------|----------|---------------------------------------|
| Codigo_do_Contato              | int      | Armazena o código do contato          |
| Numero_da_Solicitacao          | varchar  | Armazena o número da solicitação      |
| Codigo_do_Encerramento         | int      | Armazena o Código da Situação         |
| Codigo_do_Envolvido            | int      | Armazena o código do envolvido        |
| Contato_Inicial                | int      | Armazena o contato inicial            |
| Duracao                        | char     | Armazena o tempo da chamada           |
| Contato_Anterior_com_Concessio | bit      | Registra contato com o concessionário |
| Anonimo                        | bit      | Registra anônimo                      |
| Sigilo                         | bit      | Registra sigilo                       |
| Mensagem                       | text     | Armazena a mensagem                   |
| Registro_Alterado              | bit      | Registra alteração                    |
| Data_Alteracao                 | datetime | Armazena a data alteração             |
| Data_Criacao                   | datetime | Armazena e data da criação            |

É importante destacar a tabela DBO.HISTORICO que é gerada pelo SGO e registra todos os fluxos das solicitações de ouvidoria, como visto anteriormente no fluxo

simplificado representado pela figura 1.1. Por exemplo, quando uma solicitação é tramitada do grupo operacional – GAO para uma agência qualquer, é inserido um registro nesta tabela com a data e os possíveis responsáveis pela tramitação.

Esta tabela será de muita importância na migração dos dados para o ambiente Data Warehouse. Resumindo, é na tabela HISTORICO que o SGO registra todos os eventos ocorridos em uma solicitação.

#### DBO.HISTORICO

|                        |          |                                   |
|------------------------|----------|-----------------------------------|
| Data                   | datetime | Armazena a data da alteração      |
| Numero_da_Solicitacao  | varchar  | Armazena o numero da solicitação  |
| Codigo_do_Tecnico      | int      | Armazena o código do técnico      |
| Codigo_do_Encerramento | int      | Armazena o código do encerramento |
| Codigo_do_BO           | int      | Armazena o código do BO           |

Na ANEEL, o servidor SQL que contém a base da Ouvidoria está montado em uma maquina Dell com 4 processadores Xeon 700 M, 4GB de memória RAM, 18 Gb de espaço interno, 3 Terabytes de espaço externo e montada em cluster com outra maquina com a mesma configuração.

A rede da ANEEL possui um link de conexão dedicada com o servidor de base de dados na Atento e outro com o servidor de base de dados na Contax, ambas as conexões são de 1M Bit de banda de transmissão de dados. Esta estrutura é suficiente para possibilitar a importação e exportação de dados (replicação diária) entre todas as bases envolvidas neste contexto.

## 2.2 Descrição da Replicação da Base de Dados em Oracle na Atento

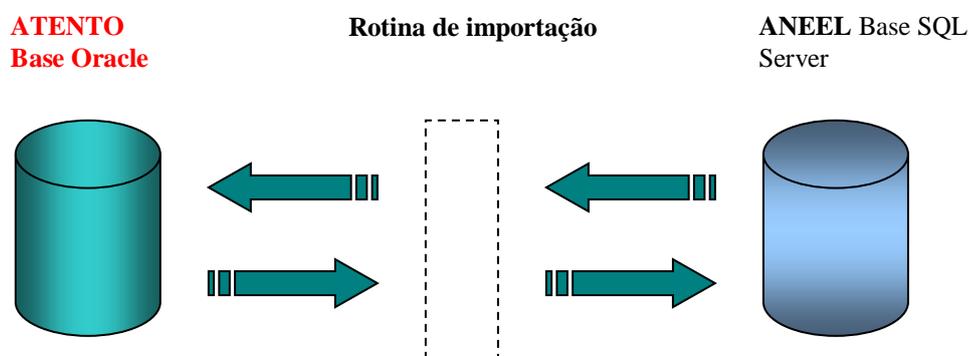
Os registros gerados pela Atento são armazenados em plataforma Oracle, conseqüentemente as bases de dados da ANEEL e da Atento não seguem a mesma plataforma, tornando-se incompatíveis para efetuar uma troca direta de dados. Rotinas de importação de dados fazem a conversão do conteúdo da base de dados da Atento, que

está na plataforma ORACLE, para o conteúdo da base de dados da ANEEL, que está na plataforma SQL.

Para uma análise mais profunda, o anexo 3 apresenta o modelo de dados incorporado pela Atento, notando-se que o padrão adotado é bem diferente do modelo de dados da ouvidoria, apresentado no anexo 2, com isso torna-se complexa a tarefa das rotinas de importação, que são:

- a) Converter os registros gerados na base ORACLE em SQL;
- b) Converter os registros gerados na base SQL em ORACLE;
- c) Atualizar registros na base ORACLE;
- d) Atualizar registros na base SQL.

A figura 2.1 simboliza a replicação de dados entre a base da ANEEL e a base de dados da Atento.



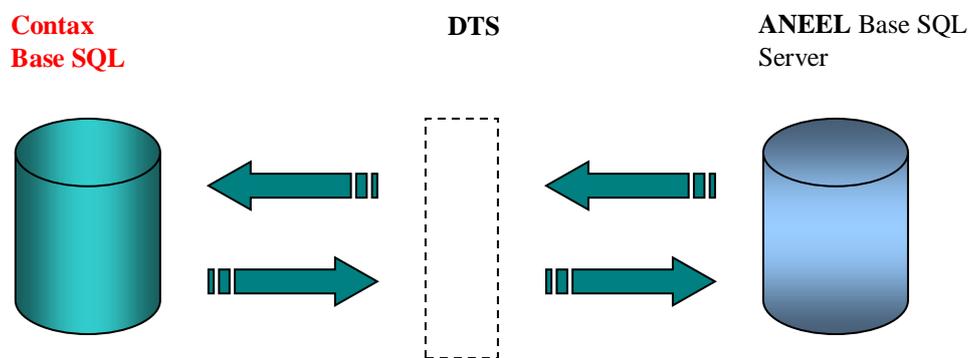
**Fig. 2.1 – Replicação de dados entre ANEEL e Atento**

### **2.3 Descrição da Replicação da Base de Dados em SQL na Contax**

Os registros gerados pela Contax são padronizados com os existentes na base da ANEEL, são geradas em plataforma SQL e estrutura das quatro tabelas básicas são exatamente iguais as contidas na base da ANEEL.

Neste caso, a replicação das bases é executada através de um software convencional fornecido pelo próprio fabricante. Conhecido como DTS, este software é um conjunto de componentes usado para importar, exportar e transformar os dados, possibilitando a simples criação de soluções para transferir, transformar e consolidar dados. Com apenas alguns cliques do mouse e usando o DTS Designer, um usuário pode tornar o que já foi uma tediosa tarefa de importar e exportar dados em uma tarefa que exige o mínimo esforço [29].

A figura 2.2 simboliza a replicação de dados entre a base da ANEEL e a base de dados da Contax.



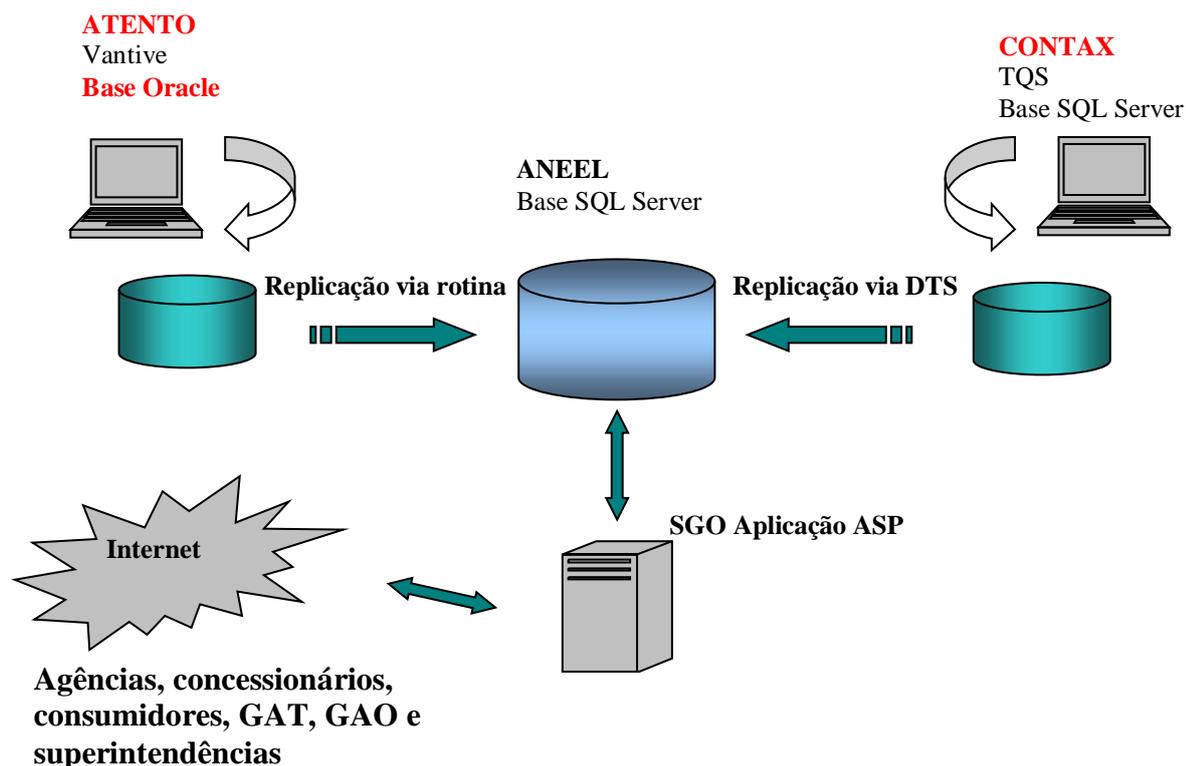
**Fig. 2.2 – Replicação de dados entre ANEEL e Contax**

## **2.4 Avaliação do Volume de Dados Existente na ANEEL**

Esta complexidade do sistema foi apresentada para mostrar a importância em organizar o armazenamento de dados de forma segura e consistente. No caso da ANEEL certificar a integridade dos dados onde existem vários agentes externos transacionando na base, não é uma tarefa fácil. A figura 2.3 mostra todos os agentes que geram ou atualizam registros na base:

- a) Atento, através de replicação de dados;
- b) Contax, através de replicação de dados;
- c) Consumidores, através da internet;
- d) Concessionários, através do SGO via web;

- e) Agências, através do SGO via web;
- f) GAO, através do SGO via web;
- g) GAT, através do SGO via web;
- h) Superintendências, através do SGO via web.



**Fig. 2.3 – Estrutura completa dos agentes transacionais na base ANEEL**

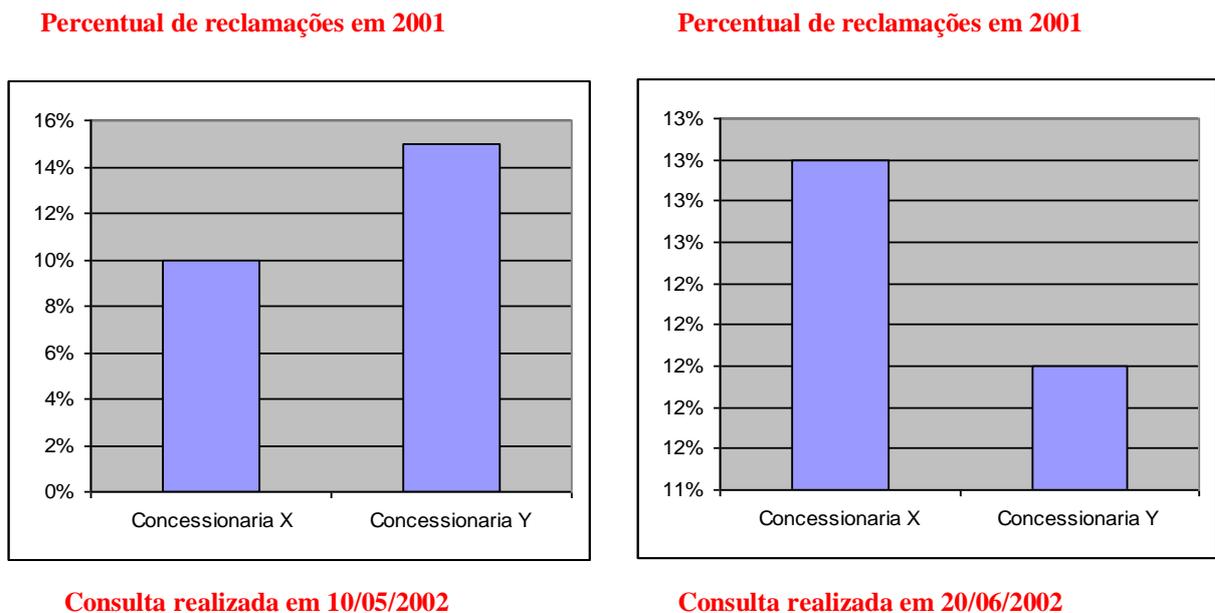
Em uma estrutura como esta, deve-se tomar alguns cuidados e adotar procedimentos para que as replicações não prejudiquem a performance do sistema. No caso apresentado a conexão com a Contax apenas será iniciada após o término da replicação com a base da Atento. É importante lembrar que estes procedimentos são executados durante a madrugada para não prejudicar o acesso de outros agentes através do aplicativo SGO.

Neste caso, para um Data Warehouse não importa de onde se originaram os dados e sim a consistência deles. O Data Warehouse irá extrair registros apenas da base de dados da ANEEL, sem interferir em nenhum processo de replicação.

### 2.4.1 Inconsistência na Extração de Informações do Ambiente Operacional

A base de dados da Ouvidoria possui hoje cerca de 4 milhões de registros, gerados não apenas por solicitações de consumidores, mas por vários complementos que envolvem o tratamento das informações. É importante lembrar que enquanto uma solicitação não é finalizada ela sofre centenas de alterações, como exemplo, mudança de status, acréscimo de contatos, e outros. Logo, um registro pode sofrer uma ou mais alterações diárias.

É possível extrair informações gerenciais do próprio ambiente relacional porém, em praticamente todos os casos, as informações se alteram ao longo do tempo, pois os dados históricos não são preservados. No caso da ANEEL, as tabelas básicas são frequentemente alteradas no ambiente relacional, causando inconsistência nas consultas realizadas periodicamente. A figura 2.4 mostra o resultados da mesma “Query SQL” efetuada em datas distintas.



**Fig. 2.4 – Consultas efetuadas em datas distintas**

Nota-se que esta variação pode ter sido causada por uma replicação ou algum outro tipo de atualização não documentada, por exemplo, uma rotina de correção de dados.

Isto não ocorre com informações extraídas de um Data Warehouse pois nele ficará armazenado a posição daquele determinado período sem sofrer qualquer alteração futura.

## Capítulo 3

### O CONCEITO DE DATA WAREHOUSE E CRM

#### 3.1 EVOLUÇÃO DOS SISTEMAS DE INFORMAÇÃO

##### 3.1.1 Histórico

No início da década de 1960 o mundo da computação consistia no desenvolvimento de aplicações individuais que eram executadas com base em arquivos mestres<sup>1</sup>, caracterizadas pelo uso de relatórios e programas geralmente escritos em COBOL.

O uso de cartões perfurados ainda era comum e os arquivos mestres eram armazenados em fitas magnéticas. As fitas se adequavam muito bem a grandes volumes de dados, a um baixo custo, mas possuíam o inconveniente de terem de ser acessadas sequencialmente. Dessa forma havendo a necessidade de ler determinado registro era necessário percorrer toda a fita magnética podendo gerar uma perda de tempo de até vinte ou trinta minutos [3].

Com o passar do tempo a quantidade de arquivos mestres foi crescendo junto com o número de acervos de fitas magnéticas. Logo haviam arquivos mestres por toda parte gerando uma enorme quantidade de dados redundantes. Tal redundância de dados ocasionava problemas em vários aspectos como [3]:

- Necessidade de sincronizar dados a serem atualizados, já que os mesmos apareciam em formatos diferentes;
- Manutenção de programas;
- Complexidade de desenvolvimento de novos programas;
- Quantidade de hardware necessário para se manter todos os arquivos mestres.

---

<sup>1</sup> Arquivos usados para armazenamento de dados e informações de controle

Por volta de 1970 surgiu uma nova tecnologia de armazenamento e acesso a dados chamado de *Direct Access Storage Device* (DASD), dispositivo de armazenamento de acesso direto, que permitia acessar diretamente os dados sem haver a necessidade de se passar pelos registros 1,2,3,...n para chegar ao registro n + 1. Uma vez que o endereço do registro fosse conhecido, bastava ir diretamente ao registro procurado. Assim, o tempo de acesso aos dados em DASD era extremamente menor do que o tempo para se varrer uma fita.

Surgiu com o DASD, um novo tipo de software, o SGBD (Sistema Gerenciador de Banco de Dados). Este, realmente solucionou os problemas de armazenamento e acesso aos dados no DASD, surgindo a idéia de “banco de dados”, o que tornaria os dados mais confiáveis, sem redundância, pois se tratava de um único repositório de dados para todo o processamento. [3]

### **3.1.2 OLTP**

Aproximadamente em 1975 surgiu o processamento de transações on-line (OLTP). O computador pôde ser usado para tarefas que antes não eram possíveis como: controlar sistemas de reservas de passagens aéreas, sistemas de caixas bancários, sistemas de controle de produção dentre outros. Tais aplicativos foram desenvolvidos como sistemas de captura, processamento e armazenamento de dados, de forma rápida, eficiente e segura, cujas rotinas e processos internos se limitavam a atualizar e disponibilizar os dados através de transações. [8]

“A finalidade do processamento de transações on-line é processar grandes quantidades de transações minúsculas e atômicas, sem perder nenhuma delas”. [8]

Em sistemas de OLTP são processados milhares de transações diárias e cada uma destas contendo uma pequena porção de dados. Os usuários de um sistema de processamento de transações freqüentemente lidam com um registro por vez, solicitam consultas de registros únicos que permitem continuar a entrada de dados que estavam realizando. No dia a dia, as tarefas de inclusão, alteração e exclusão são repetidas inúmeras vezes e as consultas à banco de dados são sempre as mesmas. Os fatores de grande importância para os usuários de OLTP são a alta performance e dados atualizados no momento de acesso.

O ciclo de desenvolvimento de um sistema OLTP, segue determinadas fases, as quais se baseiam em exigências e requisitos bem definidos pelos usuários. Através das metodologias existentes, procura-se exercer o máximo controle sobre a redundância de dados.

Um exemplo de aplicativo OLTP é a gestão de recursos humanos de uma organização, onde são armazenados todos os dados dos funcionários. Se um funcionário sair da organização ou mudar de endereço, seus dados serão modificados no banco operacional, possibilitando, também, consultas como uma relação de todos os funcionários da organização ou um ficha pessoal de um empregado específico. [3]

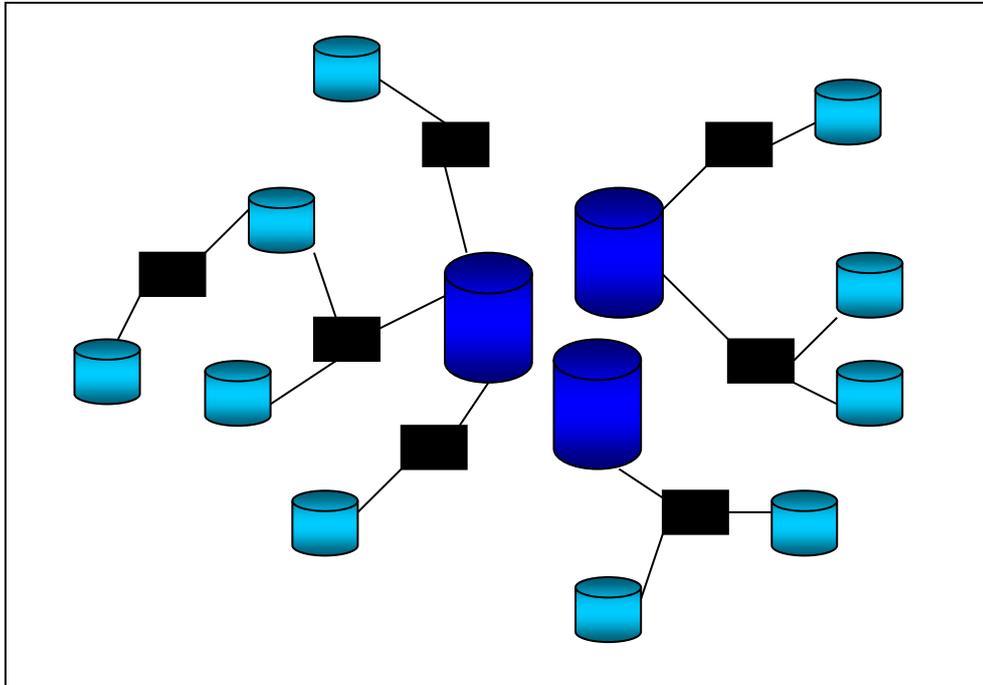
### **3.1.3 Sistemas de Informação**

No início da década de 1980, novas tecnologias, como os computadores pessoais (PCs) e as linguagens de Quarta Geração (L4Gs), começaram a aparecer. Estas novas tecnologias permitiram ao usuário final assumir um papel mais ativo, controlando diretamente os sistemas e dados, descobrindo que era possível utilizar os dados para outros objetivos além de atender ao processamento de transações on-line de alta performance.

### **3.1.4 Arquitetura de Desenvolvimento Espontâneo ou Teia de Aranha**

Surgiu em meados dos anos 80 um programa muito simples chamado de processamento de “extração”, cujo objetivo era varrer um arquivo ou banco de dados usando alguns critérios de seleção, e, ao encontrar dados que atendiam a tais critérios, transportava-os para outro arquivo ou banco de dados. [3].

Os programas de extração se tornaram rapidamente muito populares no ambiente de processamento de informações, existindo extrações das extrações, ou seja, eram realizadas extrações de banco menores que já eram produto já de outras extrações e assim, sucessivamente, formando uma verdadeira “teia de aranha”, como mostrado na figura 3.1 a seguir.



**Fig. 3.1 – Arquitetura de desenvolvimento espontâneo [3]**

Este processamento de extrações produziu o que pode ser chamado de “arquitetura de desenvolvimento espontâneo” ou “sistemas herdados” que ocorre quando uma organização lida com todo o processo de arquitetura de hardware e software com uma postura totalmente aberta.

### **3.1.5 Problemas com a Arquitetura de Desenvolvimento Espontâneo**

A arquitetura de desenvolvimento espontâneo trouxe diversos problemas para as organizações, como mostrado a seguir:

#### **3.1.5.1 Falta de credibilidade dos dados**

Dificulta o processo de tomada de decisão na medida em que dois departamentos apresentam resultados totalmente divergentes, levando a gerência a não saber o que pensar e a tomar decisões baseadas em interesses políticos e pessoais. Era o que ocorria por exemplo, quando se queria saber quanto foi o lucro de determinada empresa nos últimos quatro anos e cada departamento gerava um relatório que tinha como resultado dados divergentes. Isto acontecia devido à falta de sincronismo entre os departamentos e na definição de parâmetros com valores diferentes para a geração de tais relatórios. Os relatórios eram gerados em datas e horários diferentes

comprometendo desta forma, a integridade dos relatórios devido ao parâmetro de tempo ter sido diferente para cada relatório.

Outro fator crítico era o diferencial algorítmico, que consistia na utilização de parâmetros diferentes para a seleção e análise dos dados. O excesso de extrações, desde o momento em que os dados ingressavam no sistema até o momento em que era preparada uma análise para a gerência, aumentava a probabilidade de discrepâncias em virtude do *timing*<sup>2</sup> e diferencial algorítmico, além de que os dados poderiam, ainda, vir de fontes externas aumentando sensivelmente a falta de credibilidade dos dados devido à falta de sincronismo entre tais fontes. A falta de um banco comum a todas extrações fazia com que um departamento extraísse, por exemplo, dados de um banco A e o outro departamento de um banco totalmente diferente. Todas as características descritas contribuía para a falta de credibilidade dos dados.

### **3.1.5.2 Os problemas da produtividade**

A recuperação de dados nos diversos arquivos e conjunto de dados de uma empresa, para a produção de relatórios corporativos, era um processo muito maçante devido à necessidade de se localizar e analisar os dados, pois podem existir vários elementos de nomes iguais mas com finalidade e significado totalmente diferentes um do outro. Era necessária a verificação do nome, da descrição e da fórmula de cálculo de cada elemento de dados. Em seguida, muitos programas deviam ser escritos para recuperar os dados de suas diversas fontes, sendo que cada programa deste, devia ser customizado e ainda tinha pela frente a difícil tarefa de passar por todas as tecnologias que a empresa dispunha. Portanto, o tempo e os custos para se produzir tais relatórios eram muito altos. Assim sendo, tornava-se improvável que o primeiro relatório corporativo se encaixasse de alguma forma nos requisitos dos futuros relatórios.

### **3.1.5.3 Impossibilidade de transformar dados em informações**

A falta de integração das muitas aplicações de um ambiente legado e a falta de dados históricos, impossibilitava a geração de informações gerenciais para o Sistema de Apoio a Decisão (SAD).

---

<sup>2</sup> Intervalo de tempo considerado entre acontecimentos distintos

Com todos esses problemas, se fez necessária a criação de uma base de dados central, onde fosse possível armazenar todos os dados dos sistemas transacionais e possibilitasse a integração dos mesmos.

### **3.1.6 O Ambiente Projetado**

A arquitetura de desenvolvimento espontâneo não era suficiente para atender às necessidades com relação às visões futuras da organização. O grande volume de dados e a diversidade dos sistemas implantados faziam com que a busca de informações fosse muito lenta e difícil. Além disso, não era possível usar tais informações para apoio à decisão, pois não existiam dados históricos. Como solução desses problemas, foi desenvolvida uma nova arquitetura de armazenamento de dados, chamado de ambiente projetado.

Neste ambiente existem dois tipos de dados:

- Primitivos: são dados detalhados utilizados na condução das operações cotidianas da organização. Podem ser atualizados e são operados por rotinas executadas de forma repetitiva.
- Derivados: são dados resumidos ou calculados de forma a atender às necessidades da gerência da organização. Não podem ser atualizados e geralmente são dados históricos.

O ambiente projetado se apresenta em quatro níveis distintos:

- Operacional: contém apenas dados primitivos e atende a comunidade de transações de alta performance [3]. É baseada em aplicações responsáveis por manter os dados do negócio de uma organização de forma detalhada e atualizada e possui alta probabilidade de acesso. Esse ambiente é fonte de dados para o Data Warehouse (DW).
- Atômico ou Data Warehouse: contém dados primitivos que não são atualizados e ainda alguns dados derivados. Ele é composto por dados integrados mais granulares, variáveis no tempo, baseados em negócios e com algum nível de resumo.

- **Departamental:** praticamente só contém dados derivados e informações úteis aos diversos departamentos de uma empresa, armazenados em um banco departamental - o Data Mart<sup>3</sup> - cuja fonte de dados é o Data Warehouse.

- **Individual:** é composto, geralmente, por pequenos volumes de dados temporários que são utilizados para análises heurísticas e onde, também, é tradicionalmente processado o EIS (Executive Information Systems) [3].

O processamento de Sistemas de Apoio a Decisão necessitam de dados que estejam no nível do Data Warehouse devido a várias características que eles possuem. Estas características serão destacadas no item a seguir [3].

### **3.2 AMBIENTE DE DATA WAREHOUSE**

O ambiente de Data Warehouse é uma proposta para prover de forma centralizada e organizada acesso a dados de vários sistemas de origem. Esta proposta tem desafiado várias tecnologias.

O Data Warehouse é uma infra-estrutura de informações de apoio à decisão estável e confiável. Esta estrutura não pode ser obtida diretamente dos sistemas transacionais pois, a necessidade de integridade e correção faz com que os dados sejam muito normalizados e de difícil consulta. O Data Warehouse utiliza ferramentas que auxiliam na carga e transformação dos dados e sua consulta posterior proporciona agilidade e precisão na tomada de decisões. O Data Warehouse contempla os recursos necessários para um SAD (Sistema de Apoio à Decisão) e EIS.

Os dados atendem desde informações resumidas para alta direção até informações mais detalhadas destinadas a gerências hierárquico inferior.

A seguir serão abordados alguns aspectos fundamentais para o entendimento do ambiente de um Data Warehouse. No próximo item serão apresentados alguns conceitos de autores renomados no assunto, as principais características de um Data Warehouse, uma comparação entre sistema de informação transacional e sistema de apoio à decisão e uma introdução sobre o processo de Data Warehousing.

---

<sup>3</sup> Bancos de Dados Departamentais que reúnem dados específicos de um assunto ou setor da empresa

### 3.2.1 Conceitos

Atualmente é extremamente complexo dar uma definição precisa para Data Warehouse. Isso ocorre não pela falta de informação sobre assunto, mas pela quantidade abundante de informações. Não existe padronização ou norma sobre o assunto [11]. Desta forma, seguem algumas definições que em conteúdo dizem o mesmo, sendo diferentes apenas na forma.

“Data Warehouse é uma coleção de dados orientados por assunto, integrados, variáveis com o tempo e não voláteis, para dar suporte ao processo de tomada de decisão, onde cada unidade dos dados é relevante em determinado momento do tempo” [3];

“É uma cópia de dados transacionais especificamente estruturados para consulta e análise” [7];

“Abrange todos os dados que são considerados recursos da empresa, incluindo dados operacionais e informacionais, entretanto focando em necessidades informacionais” [18];

“Data Warehouse é um único, completo e consistente depósito de dados obtidos de uma variedade de fontes e disponível aos usuários finais de maneira que eles possam entendê-los e usá-los no contexto dos negócios da empresa” [14].

Portanto, Data Warehouse é um conjunto de dados, obtidos de sistemas transacionais, que são tratados e estruturados de maneira a facilitar as consultas deixando-as mais precisas e eficientes, fornecendo ao usuário um mecanismo mais confiável na análise e na tomada de decisão.

Tendo apresentado os conceitos é necessário seguir adiante em mais um aspecto primordial para o entendimento do ambiente de Data Warehouse: as características inerentes ao mesmo.

### 3.2.2 Características do Data Warehouse

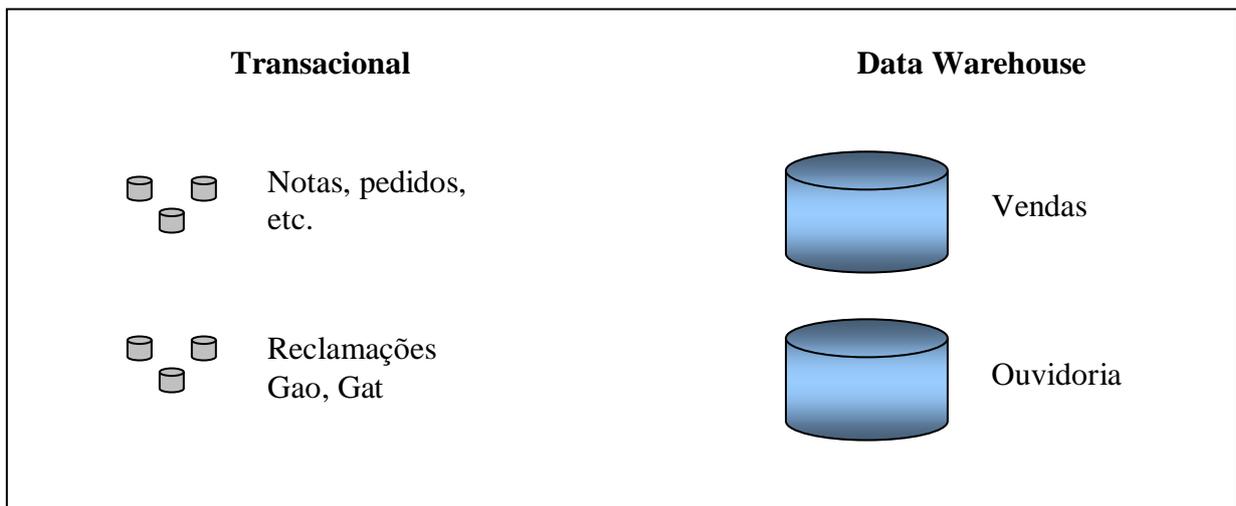
As características dos dados de um Data Warehouse são: orientados por assunto, integrados, variáveis no tempo e não voláteis. Outras características importantes são a granularidade dos dados e os metadados.

Devido à importância da granularidade dos dados, o assunto será mais detalhado no capítulo 3.4. A seguir são descritas as demais características.

#### 3.2.2.1 Orientado por assunto

A orientação por assunto se dá devido ao fato do foco dos dados serem os assuntos de interesse no grupo de negócios da organização. De acordo com o negócio da empresa o Data Warehouse terá uma orientação específica para atender àquela necessidade.

Existe aí uma grande diferença relativa aos sistemas transacionais, pois estes têm como foco os dados necessários para o trabalho diário, para cada setor ou departamento, enquanto o Data Warehouse tem como foco os assuntos globais de interesse da empresa inteira.



**Fig. 3.2 – Orientação por assunto adaptada de Inmon [3]**

### 3.2.2.2 Integrado

Como um Data Warehouse é obtido através da coleção de dados de vários sistemas de origem pode haver uma discordância na representação de alguns dados. Em um determinado sistema o atributo data pode ser definido pelo formato 'dd/mmm/yyyy', enquanto em outro por 'dd/mm/yy' ou 'dd/mm/yyyy'. Assim como podem ser usadas unidades de medidas diferentes para um mesmo dado: quilômetros, metros, centímetros, milímetros.

Para a passagem dos dados para o Data Warehouse é necessário que esses dados sejam convertidos para uma representação única. Neste caso a representação de dados que era feita através de vários tipos é unificada. Esta unificação é chamada de integração dos dados.

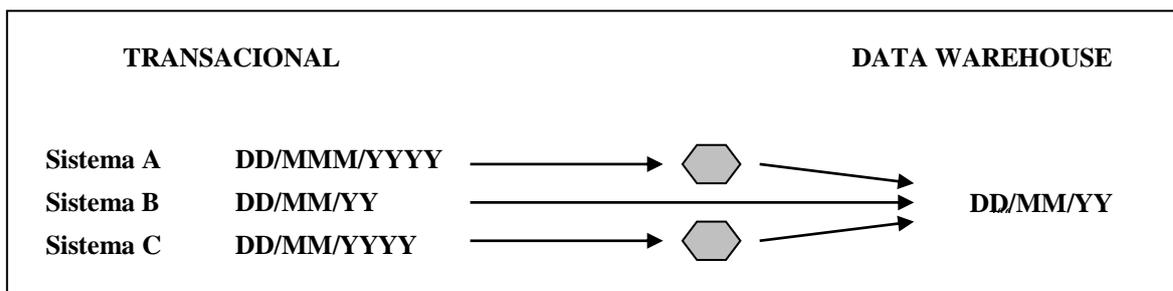


Fig. 3.3 – Integração adaptada de [3]

### 3.2.2.3 Variante no tempo

Como um Data Warehouse é composto de dados históricos, o horizonte de tempo apresentado por ele é maior quando comparado à dados transacionais. Desta maneira a atualização dos dados é feita apenas em alguns momentos no tempo e não na ocasião do acesso.

A variância no tempo está patente de várias formas:

Grande horizonte temporal – a maneira mais simples é a de que os dados estão representados ao longo de um grande horizonte temporal, de 5 a 10 anos, enquanto que no ambiente operacional esse horizonte é muito mais curto, por exemplo do presente

momento até 60 ou 90 dias. Aplicações que deverão ter uma boa performance e estarem preparadas para processos transacionais não poderão ser muito sobrecarregadas de informação para manter alguma flexibilidade. Logo, aplicações transacionais não poderão ter grande horizonte temporal.

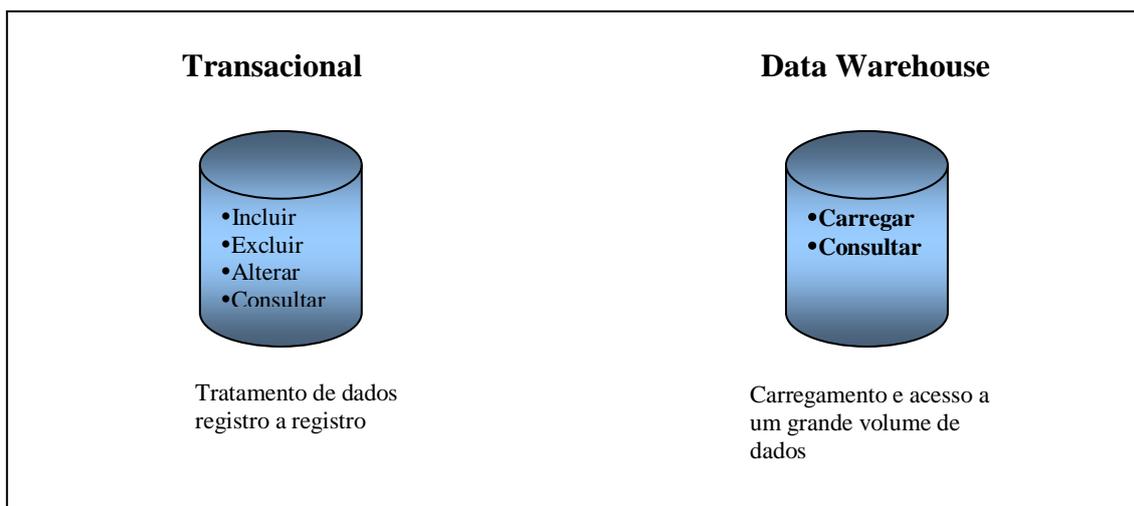
Estrutura de chave – outra forma patente de variação temporal é a estrutura de chave. Toda chave no data warehouse contém de forma implícita ou explícita um elemento de data, como dia, semana, mês, etc.

#### 3.2.2.4 Não volátil

Um Data Warehouse possui duas operações básicas, carga dos novos dados e consultas posteriores. Os dados não são trazidos do ambiente operacional diretamente. Primeiro eles são filtrados, tratados e transformados para somente depois serem carregados.

Operações como alterações e exclusões não são realizadas. Isso ocorre por que muitas vezes dados do ambiente operacional não passam para o Data Warehouse e outros são resumidos de tal forma que só são encontrados dentro do Data Warehouse.

Assim, os dados armazenados corretamente no DW não serão mais atualizados, tendo-se uma imagem fiel da época em que foram gerados.



**Fig. 3.4 – Não volátil adaptado de [3]**

### 3.2.2.5 Metadados

Uma vez que Data Warehouse reflete o modelo de negócio de uma organização, um elemento essencial do mesmo é o gerenciamento de metadados. Os metadados são definidos como “dados dos dados”, “informações das informações”. Dada a complexidade das informações de um Data Warehouse, a documentação dos sistemas e dos bancos de dados tornou-se de vital importância pois, sendo um projeto gigantesco, se não houver documentação não será possível analisá-lo.

Em um ambiente de suporte à decisão, as consultas não são fixas e rotineiras como no ambiente operacional, onde cada setor cria e manipula seus dados de acordo com suas necessidades. Portanto os usuários de DW precisam conhecer bem a estrutura e a definição dos dados, para poder examiná-los. Seguem alguns exemplos de metadados [7]:

- Frequência de atualização das fontes de dados;
- Estruturas dos bancos de dados fontes;
- Métodos e regras de agregação;
- Frequência de atualização do Data Warehouse etc.

Os metadados podem vir de várias fontes, tais como:

- Repositórios de ferramentas CASE, pois delas pode-se extrair informações sobre origem, fluxo e formato;
- Documentação dos sistemas transacionais;
- Entrevistas com especialistas da organização que entendam do negócio ou entrevistas com os usuários. (Estas entrevistas tornam-se muitas vezes uma fonte preciosa, pois algumas dessas informações não estão documentadas em nenhum outro lugar).

Segundo Inmon, “os metadados englobam o DW e mantém informações sobre o que está e aonde?” [7]. O autor ainda define as informações que o DW mantém:

- A estrutura dos dados segundo a visão do programador;
- A estrutura dos dados segundo a visão do analista de SAD;

- A fonte de dados que alimenta o DW;
- A transformação sofrida pelos dados no momento de sua migração para o DW;
- Modelo de dados;
- Relacionamento entre modelo de dados e o DW;
- Histórico das extrações de dados.

Existem hoje algumas ferramentas OLAP<sup>4</sup> que fazem o gerenciamento dos metadados. Cada organização os utilizará de forma a suprir suas necessidades, afim de exercer o controle efetivo sobre os dados armazenados.

Tendo por base as características apresentadas é interessante fazer um paralelo entre o sistema transacional e o sistema de apoio à decisão.

### 3.2.3 Sistema de Informação Transacional X Sistema de Apoio à Decisão

Algumas características diferem os Sistemas de Informação Transacionais dos Sistemas de Apoio à Decisão. Na tabela a seguir, pode-se visualizar as principais diferenças desses sistemas. [3] [13] [7] [14].

| Características       | Sistemas Transacionais        | SAD                          |
|-----------------------|-------------------------------|------------------------------|
| 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 <i>ad-hoc</i> |
| Condições dos dados   | Dados transacionais           | Dados Analíticos             |
| Volume                | Megabytes – gigabytes         | Gigabytes – terabytes        |
| Histórico             | 60 a 90 dias                  | 5 a 10 anos                  |

<sup>4</sup> Ferramenta utilizada no ambiente analítico para suporte à decisão

|                      |                       |                              |
|----------------------|-----------------------|------------------------------|
| Granularidade        | Detalhados            | Detalhados e resumidos       |
| Redundância          | Não ocorre            | Ocorre                       |
| Estrutura            | Estática              | Variável                     |
| Manutenção desejada  | Mínima                | Constante                    |
| 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        |

**Tab. 3.1 – Tabela comparativa entre Sistemas Transacionais e SAD**

O Data Warehouse suporta processamento analítico (OLAP) e os requisitos funcionais e de performance são bastante diferentes das aplicações transacionais (OLTP), tradicionalmente suportadas por uma Base de Dados Transacionais.

Os Sistemas de Suporte à Decisão e Sistemas de Informações Executivas possuem funcionalidade e desempenho diferentes dos sistemas de produção da organização. Estes sistemas recuperam e atualizam um registro por vez, usualmente atendendo a muitos usuários de forma concorrente, exigindo também um tempo de resposta imediato. Aqueles normalmente lidam com poucos usuários por vez e os requisitos em termos de tempo de resposta podem não ser críticos. No entanto, lidam com consultas complexas, não antecipadas ou previstas, envolvendo grandes quantidades de registros básicos referentes aos processos transacionais da organização.

O Sistema de Suporte à Decisão geralmente requer dados consolidados de muitas fontes heterogêneas e estas podem incluir fontes externas em adição a várias bases de dados transacionais. As diferentes fontes podem conter qualidade variável ou usar representações inconsistentes, códigos e formatos, os quais necessitam ser reconciliados.

A importância em separar dados que dão suporte aos sistemas de caráter transacional da organização daqueles que são suporte aos processos gerenciais e de suporte à decisão é que cada tipo de aplicação pode se concentrar naquilo que faz de melhor, oferecendo melhor funcionalidade e desempenho para seu caso específico.

Outro problema existente na passagem dos dados do ambiente transacional para o DW diz respeito ao acesso eficiente dos dados. Existem três tipos de carga que podem ser feitos do ambiente transacional para o DW:

- O carregamento de dados históricos;
- O carregamento de dado de valor corrente no ambiente operacional;
- O carregamento de alterações do Data Warehouse a partir das alterações que tenham ocorrido no ambiente transacional desde a última atualização do Data Warehouse.

Outra importante questão a ser abordada durante a passagem dos dados é referente à necessidade de gerenciar o volume de dados. É preciso efetuar a condensação desses dados, caso contrário o DW ficará grande demais para ser controlado.

### **3.2.4 O Processo de Data Warehouse**

Data Warehouse é um ambiente tecnológico que permite recolher, organizar e agregar uma grande quantidade de dados em informação utilizável, transformando um banco de dados transacional em um ambiente que permite o uso estratégico dos dados, sendo um ambiente e não um produto.

As fases desse processo serão abordadas e desenvolvidas nos itens seguintes.

## **3.3 ARQUITETURA**

O primeiro passo a ser seguido no processo de Data Warehousing é a definição de uma arquitetura que possibilite a coleta, manipulação e apresentação dos dados de forma eficiente, fazendo com que o usuário final não visualize os detalhes do processamento e sim a realização das suas necessidades.

É preciso considerar que a arquitetura do DW não é somente a representação da transferência de dados de bancos operacionais para um outro de maior volume. Devem ser seguidos métodos e padrões pré-definidos para que haja sucesso.

Como as necessidades de cada organização são bastante heterogêneas, o mesmo acontece com a arquitetura do DW que dependerá do negócio da organização. Abaixo serão abordadas duas arquiteturas de duas grandes autoridades da área que são Ralph Kimball [KIM98a] e Bill Inmon [6].

### 3.3.1 Arquitetura Segundo Kimball

A arquitetura de um Data Warehouse visa principalmente coletar os dados de origem, transformá-los e transportá-los para os servidores de aplicação e disponibilizá-los de maneira útil e eficiente para que sejam analisados pelo usuário final [7].

Essa arquitetura seria um plano geral para a construção de um DW, podendo servir como uma base funcional para o mesmo, pois ela descreve o fluxo de dados dos sistemas fontes até os usuários finais, assim como as transformações realizadas em todas as etapas de sua construção, especificando também as técnicas para que isso aconteça.

A figura 3.5 abaixo apresenta uma visão geral de como essa arquitetura é organizada e servirá como base para a apresentação de conceitos, técnicas e ferramentas utilizadas nos próximos tópicos deste item.

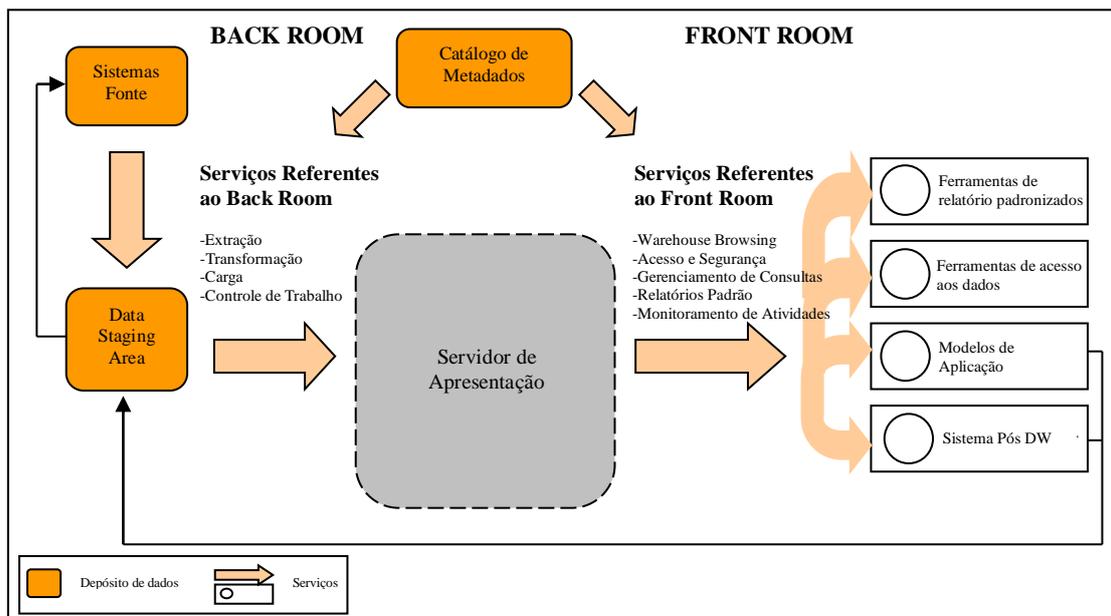


Fig 3.5 – Figura sobre Arquitetura adaptada de Kimball [7]

Observando, existem duas partes principais na figura que são: Back Room e Front Room, além de dois outros componentes que são os serviços e os depósitos de dados. Os serviços são responsáveis por realizar as tarefas requeridas pelo DW, e os depósitos de dados são locais temporários ou permanentes onde os dados serão depositados.

Acompanhando a estrutura da figura 3.5 pode ser observado claramente o processo que se passa desde os sistemas fonte até a apresentação final para o usuário. Os dados são movidos dos sistemas fonte (sistemas transacionais) para uma área de armazenamento temporário (Data Staging Area) através de serviços oferecidos pelo Back Room. Após esse processo, os dados são transformados e organizados apoiados por outros serviços dessa área, e então são carregados no servidor de apresentação.

Depois de armazenados os dados nesses servidores, eles estão disponíveis para o acesso dos usuários, que é feito através de ferramentas disponíveis no mercado ou softwares produzidos pelos próprios programadores.

Nos próximos tópicos serão expostos detalhadamente cada um dos componentes, tanto do Back Room como do Front Room, apresentando conceitos, funcionalidades e serviços de cada uma dessas áreas.

### **3.3.1.1 Back room**

Fase responsável por todo o processo de organização dos dados, sendo de grande importância pois a qualidade dos dados depende muito dela. Esse processo pode ser conduzido utilizando ferramentas disponíveis no mercado ou através de programas desenvolvidos pelos próprios componentes da equipe de DW. Mesmo empregando essas ferramentas para facilitar esse processo as atividades que ocorrem no Back Room são complexas e normalmente consomem uma parcela considerável de tempo no projeto.

A seguir serão vistos os componentes, serviços, gerenciamento de recursos e metadados referentes a essa área.

#### **3.3.1.1.1 Componentes do back room**

De acordo com a figura 3.5 o Back Room é composto pelos seguintes componentes:

- **Sistemas Fonte:** são sistemas de dados internos ou externos de onde são retiradas as informações para que seja construído o DW, podendo ser planilhas, sistemas transacionais, arquivos entre outros.

- **Data Staging Area:** local onde é feita a preparação dos dados para que possam ser carregados para o DW. Essa preparação é feita através dos serviços oferecidos por essa área que serão discutidos no próximo tópico.

- **Servidor de Apresentação:** é compartilhado tanto pelo Back Room quanto pelo Front Room (figura 3.5). Esse servidor é a plataforma de destino dos dados do Back Room no qual eles serão armazenados.

### **3.3.1.1.2 Serviços do back room**

#### a) Extração

A extração dos dados operacionais é a primeira tarefa a ser executada. Esses dados devem ser inseridos em uma área de trabalho temporária para a qual os dados serão colocados e tratados antes de serem carregados para o Data Warehouse.

Deve-se determinar quais os dados devem ser extraídos e que tipos de filtros devem ser aplicados. Deve-se entender os objetivos do processo de extração determinando quais tipos de serviços serão necessários [7].

#### b) Transformação

Depois que o dado é extraído então ele deve ser tratado de modo que seja apresentado na forma correta para o usuário e adequados ao negócio para o qual o Data Warehouse dará apoio.

O tratamento dos dados poderá envolver um ou mais dos seguintes processos, dependendo da necessidade e situação:

- **Integração** – envolve a geração de chaves substitutas para cada registro de modo a evitar dependência de chaves definidas no sistema legado, onde o processo de geração dessa chave substituta obriga a integridade referencial as tabelas.

- Limpeza – Constitui no conjunto de atividades realizadas sobre os dados extraídos, de modo a corrigir o uso incorreto ou inconsistente de códigos e caracteres especiais; resolver problemas de conflito de domínios; tratar dados perdidos; corrigir valores duplicados ou errados. Essa limpeza deve deixar os dados em formato padrão, não duplicado, correto e consistente espelhando a realidade.
- Eliminação – consiste em eliminar os campos e dados provenientes dos sistemas legados que não serão úteis ao DW.
- Combinação – é realizada quando fontes de dados possuem os mesmos valores de chaves, representando registros iguais ou complementares ou atributos não chave iguais, incluindo equivalência textual de códigos de sistemas legados distintos.
- Verificação da integridade referencial – significa verificar se os dados de uma tabela são iguais aos dados correspondentes em outra tabela.
- Desnormalização e renormalização – consiste em reunificar as hierarquias de dados, separadas pela normalização dentro de uma tabela desnormalizada.
- Conversão de tipos de dados – envolve transformação de baixo nível, de forma a converter um tipo de dado em um outro formato.
- Cálculos, derivação e alocação – são transformações a serem aplicadas sobre as regras do negócio identificadas durante o processo de levantamento de requisitos.
- Agregação – a agregação pode ser realizada em qualquer parte do processo de carga com a finalidade de melhorar o desempenho das consultas.
- Auditoria no conteúdo dos dados – o processo de transformação deve realizar constantes verificações de somas, contagem de linhas e testes.
- Valores nulos – para evitar valores nulos em um DW, são utilizados artifícios em suas ocorrências como 09/09/9999 para datas, -1 para números ou qualquer outra atribuição referente ao negócio.

#### c) Carga

Depois de serem transformados, a etapa final é a carga de dados no Servidor de Apresentação. Os utilitários de carga devem permitir que o administrador do sistema

possa monitorar toda a operação, cancelar quando necessário e reiniciar após uma falha sem a perda da integridade dos dados.

#### d) Controle de trabalho

Esse serviço envolve o fluxo completo dos trabalhos e de organização de dados. Para que isso ocorra é essencial que seja montado um ambiente para a criação, gerência e monitoramento desse fluxo de dados de modo a facilitar sua manutenção e desenvolvimento. Esse ambiente deve ter as seguintes funcionalidades:

- Definição de tarefas – deve-se planejar a seqüência de passos ou trabalhos a serem realizados e especificar as relações entre eles, cujo conjunto de atividades define o fluxo de dados no DW.

- Agendamento de tarefas – consiste em agendar as operações de carga no DW que geralmente são baseadas em tempo ou eventos. Para isso é necessário monitorar *flags* do banco de dados, verificar a existência de arquivos, comparar a criação de datas, etc.

- Monitoramento – os sistemas devem oferecer informações do andamento de todos os seus processos, como qual passo do processo de carga está sendo executado, quando iniciou, quanto tempo levou, dentre outras informações. As ferramentas utilizadas devem apresentar as informações do que ocorreu, está ocorrendo ou ocorrerá, preferencialmente, através de recursos visuais.

- Tratamentos de exceções – os sistemas necessitam de um local para a manipulação de dados rejeitados durante o processo de carga, naturalmente limitados a uma quantidade de erros aceitáveis e, também, de um processo para manipulá-los. Esses processos podem ser simples como inspeção manual, correção de arquivos e execução de nova carga. Quando possível, os dados que foram rejeitados, devido a erros de conteúdo, como caracter em lugar de dado numérico, deveriam ser destinados diretamente ao responsável pelo sistema fonte para a correção. A finalidade dessa atitude é, a curto prazo, aumentar o grau de cooperação entre os diversos profissionais com vistas a melhoraria da qualidade dos dados para o DW, o que é fundamental.

- Arquivos de Log – funcionam como um registro de todas as informações do processo de carga. Arquivos de Log permitem recuperar dados e reiniciar o processo

em caso de erro durante a execução de um determinado trabalho. Estes arquivos podem ser gerados em arquivos textos ou *log* para um banco de dados, facilitando a criação de gráficos e relatórios sobre as transações efetuadas, usados também para realização de estudos que permitem otimizar os processos de carga.

- Controle de erros – normalmente ocorrem erros durante o processo de carga, devido a essas ocorrências deve-se elaborar um plano e disponibilizar ferramentas para tratar esses erros irrecuperáveis, quando ocorrerem. Essas ferramentas deveriam antecipar e providenciar recuperação de falhas, parada do processo de carga e reinício do sistema. Além disso, as ferramentas deveriam realizar atividades para minimizar o impacto de falhas, como por exemplo realizar a operação de *commit* (confirmar) sobre um pequeno conjunto de registros ao mesmo tempo e manter histórico do que foi confirmado.

- Notificação – são alertas emitidos para determinadas pessoas com a finalidade de informar sobre o processo de carga, como por exemplo, a ocorrência de falhas e necessidade de reiniciar o sistema, a necessidade de tomadas ações críticas para dar continuidade à carga, dentre outras. Os tipos de notificações podem ser vários, como envio de mensagem pelo sistema para a console, envio de e-mail, alerta através de *pager*, mensagem pré-gravada para telefone convencional ou celular, etc.

### **3.3.1.1.3 Gerenciamento de recursos do back room**

O DW está sujeito aos mesmos riscos que qualquer outro sistema computacional, como falha ou quebra de disco, falta de energia elétrica, danos físicos na memória principal, etc. Não obstante estes riscos, o DW necessita manter os dados por períodos de tempo mais longos que os sistemas operacionais. Desta forma é de vital importância que existam controles como:

- *Back Up e Recovery* – são processos que permitem a recuperação de um trabalho após a ocorrência de uma falha.

- *Archive e Retrieve* – são processos que permitem aos usuários de negócio acessar dados antigos que foram movidos do DW para outros meios de armazenamento como fita e CD-ROM.

#### **3.3.1.1.4 Metadados do back room**

Todas as atividades realizadas no Back Room são realizadas com o auxílio dos metadados, que em linhas gerais, para esta área, são guias para os processos de extração, transformação e carga.

Esse tipo de metadado ajuda a equipe de banco de dados a trazer as informações sobre eles para dentro do DW, além de interessar aos usuários do negócio quando há uma necessidade dessas informações.

#### **3.3.1.2 Front room**

O Front Room é o local onde ocorre a interface do usuário com o DW. Nesta fase, já com os dados tratados e armazenados, eles devem ser disponibilizados de forma inteligível ao usuário final, seja através de gráficos, planilhas ou outro requisito apontado por ele.

Como o DW deverá ser usado geralmente pela diretoria e gerência das organizações, o acesso aos dados deve ser o mais simples possível, sendo construído com uma interface bastante amigável para que seja escondida toda a complexidade envolvida na sua concepção. Agora serão abordados os componentes, serviços e acesso aos dados dessa área assim como os seus metadados.

##### **3.3.1.2.1 Componentes do front room**

- Servidor de Apresentação: os dados são armazenados nesse servidor para que sejam acessados normalmente pelos usuários finais, que poderão requisitá-los através de ferramentas ou aplicações específicas.
- Ferramentas de acesso aos dados: o usuário, utilizando ferramentas de acesso aos dados, pode realizar consultas *ad-hoc*<sup>5</sup> e gerar relatórios, os quais tipicamente são centrados em necessidades específicas, tais como investigação de anomalias, determinação de padrões e tendências, etc. Normalmente o conjunto resposta de uma consulta é armazenado temporariamente pela própria ferramenta de acesso aos dados. Algumas dessas ferramentas possibilitam também a transferência do conjunto de

---

<sup>5</sup> Consultas elaboradas pelo próprio usuário

respostas obtidas, para outros aplicativos, como por exemplo, planilhas eletrônicas, de modo que os dados obtidos possam ser analisados e trabalhados posteriormente.

- Ferramentas geradoras de relatórios padrão: utilizam o DW como fontes primárias de dados. Estas ferramentas normalmente geram relatórios padronizados e muitas vezes possuem internamente um cache ou bibliotecas de relatórios que armazenam um conjunto de relatórios pré-executados que, possibilitam rápido tempo de resposta e apresentação ao usuário.

- Modelos de Aplicação: *Data Mining* (Mineração de Dados) é um exemplo de modelo de aplicação. Consiste da aplicação de análise de dados e algoritmos que, sob aceitáveis limitações de eficiência computacional, produz uma enumeração particular de padrões sobre os dados. O objetivo é identificar novos, válidos, potenciais, úteis e compreensíveis padrões nos dados [25].

- Sistema pós DW: é um sistema onde o sistema fonte de dados é o próprio DW. A proposta básica destes sistemas ainda é a geração de relatórios.

### **3.3.1.2 Serviços do front room**

Geralmente os serviços a seguir são disponibilizados em ferramentas de *front end*<sup>6</sup>. Há uma tendência da migração dos serviços de acesso aos dados das ferramentas de *front-end* para os servidores de aplicação. Isso se deve a dois motivos principais: a “pressão” exercida pelos grandes compradores de produtos de DW sobre os vendedores de bancos de dados forçando-os a melhorar seus produtos, especificamente os destinados ao DW; e o grande crescimento da demanda por ferramentas baseadas na Web. A seguir serão descritas as funcionalidades oferecidas por essa área.

#### a) Warehouse Browsing (Navegador)

Têm a finalidade de auxiliar os usuários em seus esforços para acessar e encontrar as informações que necessitam. Para possibilitar essa ajuda, qualquer tipo de ferramenta *browser* (pesquisa) deveria ser ligada dinamicamente ao catálogo de metadados dessa fase, que será abordado posteriormente, para mostrar aos usuários as áreas subordinadas do DW e os elementos de dados dentro dessas áreas.

Também devem ser capazes de encontrar as definições e derivações dos vários elementos de dados e mostrar um conjunto padronizado de relatórios que inclui esses elementos. Uma vez que o usuário encontre o item de interesse, o browser deveria prover uma ligação (*link*) para o recurso apropriado.

#### b) Serviços de Segurança e Acesso

Segurança e Acesso utilizam os serviços de autenticação e autorização, para a identificação do usuário e permitindo, ou não, a utilização de recursos.

Autenticação é o método de verificar se um usuário é realmente quem diz ser. Normalmente consiste em uma simples senha, que habitualmente é obrigada a ser alterada em intervalos de tempos definidos por um sistema. Além disto, o sistema pode requerer algum tipo de identificação física, como por exemplo, o uso de cartões magnéticos. No lado do banco de dados deve ser adotado um sistema de autenticação no qual cada usuário receba uma identificação única.

Autorização consiste em determinar quem tem acesso a que recurso. Depende de inúmeros fatores tais como cultura organizacional, segurança, etc.

#### c) Monitoramento de Atividades

As atividades de monitoramento envolvem a obtenção de informações sobre o uso do DW. Os serviços podem ser centrados em torno das seguintes áreas:

- Desempenho – pode-se utilizar as informações provenientes do uso do DW para melhorar o tempo de resposta de consultas, como por exemplo verificando-se quais tabelas e colunas mais freqüentemente são juntadas, selecionadas e agregadas. Isso pode conduzir a modificações e criação de novos agregados, índices e, inclusive, a modificações no esquema do projeto.
- Suporte ao usuário – pode-se acompanhar os usuários, principalmente os que foram treinados recentemente e estão iniciando o uso do sistema. Pode-se verificar, também, o que os usuários estão fazendo e onde necessitam de ajuda.

---

<sup>6</sup> Ferramentas responsáveis pela exibição de dados ao usuário final

- Marketing – publicar simples estatísticas de modo a prestar informações aos administradores, tais como o uso do DW por parte dos usuários, onde e como estão sendo empregados os recursos financeiros despendidos para o DW, etc.

- Planejamento – os dados obtidos do DW, tais como crescimento do uso, tempo médio de consulta, tamanho do banco de dados, tempo de carga, dentre outros inúmeros fatores possibilitam orientar novos esforços de planejamento. Há softwares específicos no mercado projetados para monitorar as atividades do DW. Há, também, ferramentas de gerenciamento de consultas que oferecem o monitoramento de consultas como um subproduto e outras ferramentas de Front-End que oferecem atividades rudimentares de monitoramento.

#### d) Gerenciamento de consultas

Consiste em um conjunto de funcionalidades que gerencia as modificações realizadas entre a formulação de uma consulta e a sua execução no banco de dados e o retorno do conjunto resposta ao usuário.

#### e) Padronização de relatórios

Esse serviço possibilita a criação de relatórios em formatos pré-definidos, limitando a interação com o usuário, uma grande audiência e podem ser executados em horários previamente programados.

### **3.3.1.2.3 Metadados do front room**

Os metadados dessa área são mais descritivos e auxiliam as ferramentas de consulta e geradoras de relatórios. Esses metadados são utilizados também como um tipo de dicionário de negócios, representado por todos os elementos de dados.

### 3.3.2 Arquitetura Segundo Inmon

Inmon propõe um modelo funcional chamado *Corporate Information Factory* (CIF), ou seja, uma fábrica de informações corporativas baseado em uma arquitetura definida de acordo com a figura 3.6, onde as atividades da empresa são armazenadas em bancos de dados transacionais, a partir dessas bases os dados passam por diversas integrações e transformações sendo armazenadas no Data Warehouse ou no ODS (será explicado a seguir), ficando disponíveis para seu uso posterior pelos usuários finais.

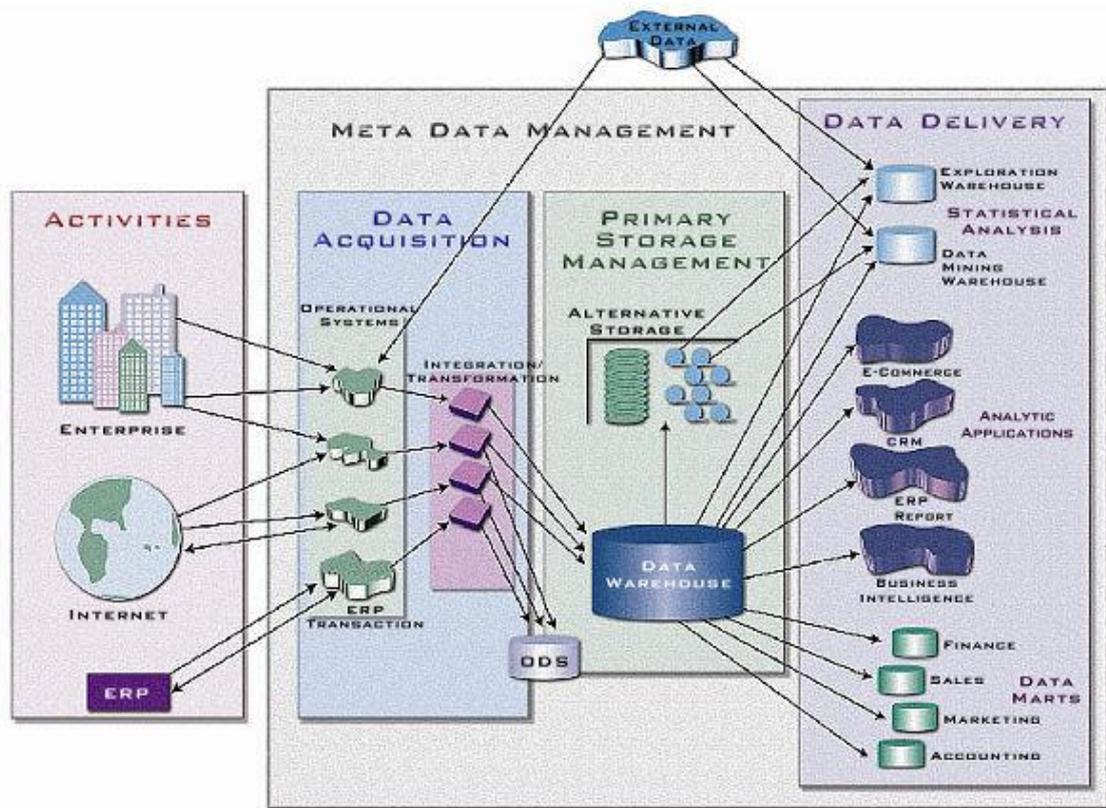


Fig. 3.6 – Figura sobre CIF [6]

A seguir serão definidos todos os componentes da figura.

**Activities** – São os eventos capturados pelos legados da empresa e/ou Sistemas ERP<sup>7</sup> assim como transações externas.

**Data Acquisition** – é um conjunto de processos responsável pela captura, integração, transformação, reengenharia e carga do Data Warehouse e da Operational

<sup>7</sup> Enterprise Resource Planning – é um sistema responsável pela integração de todas as áreas de uma determinada organização

Data Store (ODS). Reengenharia de dados é o processo de investigação, padronização e limpeza de dados.

**Operational Systems (sistemas operacionais)** – São os sistemas internos e externos que suportam as operações do negócio no dia a dia. São a fonte de dados para o Data Warehouse e o Operational Data Store. São exemplos de sistemas operacionais: ERP, sistemas legados e relacionais.

**Operational Data Store (ODS)** - Orientado por assunto, integrado, corrente, consiste em uma coleção de dados volátil usado para suporte tático ao processo de tomada de decisão para a empresa. É o ponto central de integração de dados para o gerenciamento, distribuição e visão comum de dados da empresa, pode ser considerado intermediário entre o DW e os sistemas transacionais pois tem características tanto de um como de outro.

**Primary Storage Management** – Consiste no processo de gerência de dados dentro e a caminho do Data Warehouse e do Operational Data Store. Inclui processos de *back-up* and *recovery*, particionamento, sumarização, agregação, *archival* e *retrieval*. Os processos são incluídos no Alternative Storage.

**Alternative Storage** – É o conjunto de dispositivos usados para efetivar o custo de armazenar o Data Warehouse e explorar os dados que são necessários mas não são acessados frequentemente. Esses dispositivos são mais baratos que os discos e fornecem performance adequada quando há necessidade de usá-los.

**Data Warehouse** – como foi visto anteriormente é orientado por assunto, integrado, variante com o tempo, não volátil, uma coleção de dados usados para suporte tático/estratégico de tomada de decisão da empresa. Este é o ponto central de integração de dados para inteligência do negócio e a origem de dados dos Data Marts, disponibiliza uma visão comum dos dados da empresa.

**Data Delivery (liberação de dados)** – É um conjunto de processos que permitem aos usuários finais e grupos de suporte construir e gerenciar visões do Data Warehouse dentro de seus Data Marts. Isto envolve três passos ou processos que são: filtrar, formatar e liberar dados do Data Warehouse para os Data Marts.

**Data Mart** – É a customização e/ou sumarização de dados liberados do Data Warehouse e estruturado para suportar a especificação analítica requerida de uma unidade ou função de negócio. É utilizado como uma visão comum de dados estratégicos da empresa e fornece um maior controle e responsabilidade do negócio. O Data Mart pode ou não estar no mesmo servidor ou local do Data Warehouse (conceitos e classificações de Data Marts serão abordados no capítulo 3.6.1).

**Exploration Warehouse** – É uma estrutura DSS com o propósito de prover um completo processamento de consultas *ad-hoc*. Um *Exploration Warehouse* utiliza compressão de dados para prover respostas mais rápidas com a capacidade de acessar a base de dados inteira.

**Data Mining Warehouse** – É um ambiente criado para analisar teste de hipóteses, incertezas e suposições desenvolvidas em um *Exploration Warehouse*. Ferramentas especializadas de *Data Mining* contêm agentes inteligentes para serem usados na execução dessas tarefas.

**Statistical Applications** – Conjunto de execuções complexas, dificuldades de análises estatísticas como exceções, recursos, médias e modelos de análises. O Data Warehouse é a origem desses dados para análise. Essas aplicações requerem um razoável ambiente executável.

**Analytic Applications** – São pré-determinados, aplicações de suporte à decisão. Eles geralmente requerem alguma customização para combinar os requisitos específicos da empresa. A origem dos dados é o Data Warehouse. Exemplos dessas aplicações são análises de ricos, base de dados análise de marketing (CRM<sup>8</sup>), etc.

**Meta Data Management** – É um processo utilizado para gerenciar informações necessárias para promover a legibilidade dos dados usados na administração. O conteúdo ou informação do dados é descrito no termo de dados sobre os dados, atividades e conhecimento.

**External Data** – é qualquer dado fora da coleção de dados de aplicações internas da empresa. Eles podem ser dados demográficos, créditos, competitivos e

---

<sup>8</sup> Customer Relationship Management – utilizado para conhecer o perfil dos clientes de uma organização baseado em uma base confiável de dados

informações financeiras. Geralmente, dados externos são adquiridos pela empresa de um vendedor de informações.

Depois de conhecidos os componentes do modelo, pode-se navegar facilmente pelo fluxo descrito na figura 3.6 e assim entender o funcionamento dessa arquitetura.

Tendo entendido o processo de Data Warehousing, serão descritos nos próximos itens os passos para a criação de um DW através de outros conceitos importantes.

### **3.4 GRANULARIDADE**

Um dos fatores mais importantes da modelagem física de dados, independente da arquitetura e implementação a serem utilizadas, é a definição da granularidade de dados. Este ponto deve ser analisado profundamente, visto que a definição não acurada do nível de granularidade pode ocasionar um custo elevado desnecessário, ou ainda, fazer com que o projeto de DW não cumpra plenamente as suas atribuições.

#### **3.4.1 A Importância da Granularidade de Dados de um Data Warehouse**

Granularidade de dados em um Data Warehouse, refere-se ao nível de sumarização dos elementos de dados e ao nível de detalhe disponível nos dados. Quanto mais detalhes existem nos dados, menor é a granularidade e conseqüentemente, quanto menos nível de detalhe, maior é a granularidade.

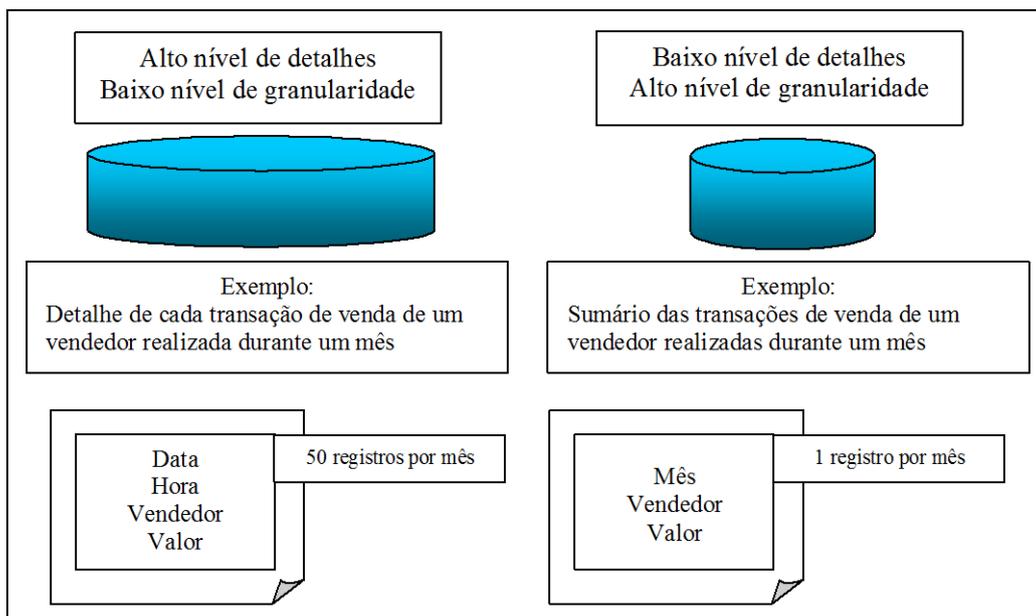
Segundo Inmon, o aspecto mais importante de um projeto de data warehouse é a questão da granularidade, e sua definição é uma questão fundamental para o projeto de Data Warehouse, pois afeta diretamente o volume de dados a ser armazenado, o tipo de informações que poderão ser obtidas e o esforço computacional necessário para a obtenção das informações desejadas, por ocasião da realização de consultas [3].

Outro aspecto importante da granularidade em um projeto é entender que ela não se limita somente a tempo, mas a todos os fatores de classificação da informação que estiverem sendo utilizados [22].

“A granularidade baixa é utilizada quando o projeto quer manter o Data Warehouse como uma base histórica de recuperação e análise detalhada das transações dos sistemas operacionais”[22].

Com um nível de granularidade muito baixo, é possível responder a praticamente qualquer consulta. Entretanto uma grande quantidade de recursos computacionais se faz necessário, entre os quais pode ser citado o grande espaço necessário para o armazenamento dos dados no DW, podendo ser este um fator crítico para o sucesso do projeto.

Quando se adota um nível alto de granularidade, o número de questões a que os dados podem responder é limitado. A vantagem é que com um nível mais alto de granularidade, não somente é necessário menos espaço como são necessários menos índices obtendo-se uma considerável economia de espaço de armazenamento no banco de dados analítico [3].



**Fig. 3.7 – Granularidades diferentes em um mesmo assunto, adaptado [22]**

Uma forma possível de se minimizar este problema é criar níveis duais de granularidade.

### 3.4.2 Níveis Duais de Granularidade

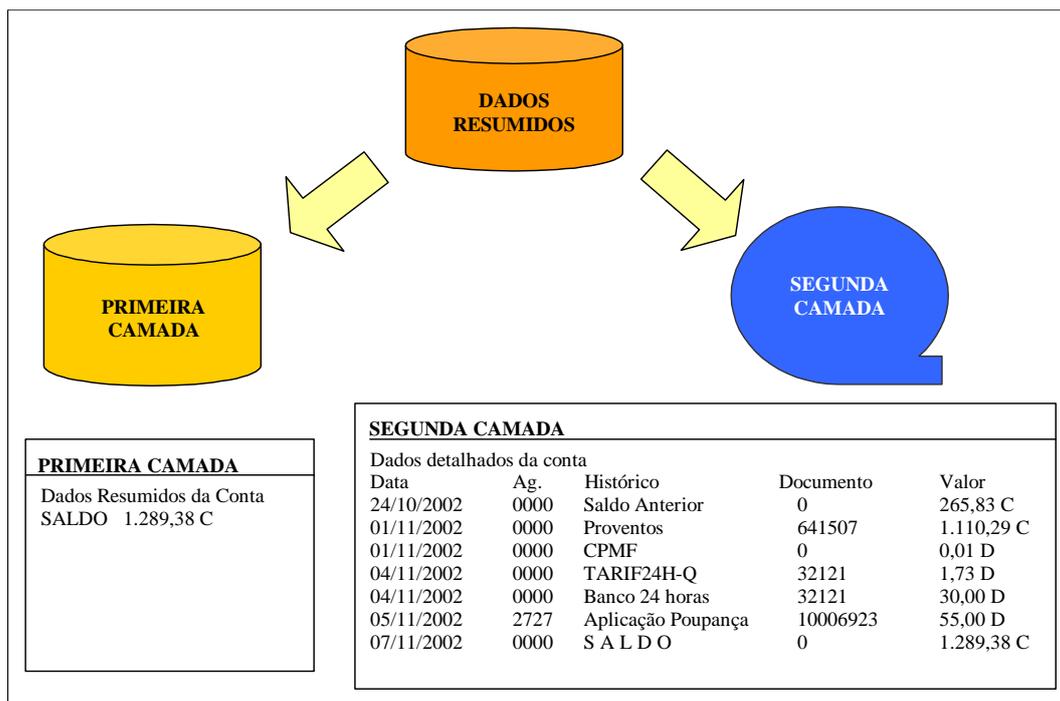
O balanceamento do nível de granularidade é um dos aspectos mais críticos no planejamento de um DW, pois na maior parte do tempo, há uma grande demanda por eficiência no armazenamento e no acesso aos dados, bem como pela possibilidade de analisar dados em maior nível de detalhes. Quando uma organização possui uma grande quantidade de dados, faz sentido pensar em mais de um nível de granularidade, pois em algum momento será necessário o acesso a detalhes de um determinado dado. Na realidade, a necessidade de que exista mais de um nível de granularidade é tão grande que a opção de projeto que consiste em duplos níveis de granularidade deveria ser o padrão para quase todas as empresas [3].

Descobrir o equilíbrio entre a real necessidade dos níveis de granularidade do usuário final do projeto e o custo envolvido nesta implantação, é um dos principais pontos da definição de um DW. Definir níveis duais de granularidade baseia-se simplesmente em definir um conjunto de dados a ser disponibilizado com um nível mais baixo de granularidade e o conjunto histórico ser armazenado agrupado com um nível mais alto de granularidade.

Todos os dados provenientes dos sistemas transacionais e fontes externas são inicialmente carregados no nível de detalhes corrente. Os dados correntes são colocados em disco sendo mais rápido o acesso, mas tornando-se mais caro e complexo a sua gerência.

Em seguida, os dados são sumarizados e armazenados em um nível levemente resumido, onde haverá um volume de dados menor, visando atender a pré-consultas disponibilizadas para usuários finais. São normalmente armazenados em disco.

Os dados altamente sumarizados são obtidos a partir da compactação dos dados levemente resumidos tendo a finalidade de atender a gerentes e administradores, pois as informações são disponibilizadas de forma compacta e de fácil acesso. Por outro lado, os dados provenientes dos sistemas transacionais também são armazenados no nível de detalhes antigos quando atingirem uma idade limite, normalmente especificada no projeto, sendo movidos para um meio de armazenamento de massa, tipicamente mais barato como fita, disco ótico, *compact disc* (CD), DVD etc.



**Fig. 3.8 – Níveis duplos de granularidade**

Desta forma, o nível “dual” de granularidade possibilita alta flexibilidade ao DW, sendo possível a obtenção de informações tanto de forma eficiente e com tempo de reposta adequado, quanto através de atividades complexas e custosas. Entretanto, o mais importante é que em qualquer uma dessas situações, as informações desejadas serão obtidas, pois os dados se encontram no DW.

Analisando as diversas abordagens sobre granularidade a colocação de Kimball [8], que defende a adoção do mais baixo nível de granularidade possível em um DW, parece ser a mais adequada, pois, dessa forma, seria possível capturar todos os detalhes dos dados. Deve-se levar em consideração que as necessidades que os usuários tem hoje não necessariamente serão as mesmas que eles terão amanhã. Tal afirmação ganha mais importância quando se observa o tempo de duração do DW, de 5 a 10 anos [3]. Dessa forma, é necessário que uma solução de DW seja flexível e escalonável, de modo a poder atender qualquer necessidade do usuário, independente de envolver dias, meses ou anos, o que somente será possível se os dados existirem.

### 3.5 MODELAGEM DE DADOS DE UM DATA WAREHOUSE

Tendo visto as abordagens a respeito da arquitetura e definido qual a melhor estratégia para a escolha da granularidade dos dados, é necessário conhecer as principais técnicas de modelagem de dados de um Data Warehouse.

A modelagem de dados de um Data Warehouse é completamente diferente da modelagem de dados de um sistema transacional. Isso ocorre porque estes são feitos para atenderem a necessidades distintas.

Em um sistema transacional é necessário um modelo de dados que evite a redundância e a inconsistência dos dados. Esta necessidade é garantida através da utilização das formas normais<sup>9</sup>.

Já no modelo de dados de um DW o aspecto mais importante é que as consultas sejam realizadas de maneira ágil e eficiente. Não há necessidade de atender às formas normais, pois isso deixaria as consultas muito lentas, pela necessidade de vários *joins* entre as tabelas, inviabilizando o modelo do DW.

Além disso, o modelo de dados de um ambiente transacional é muito complexo para o entendimento de um gerente, diretor ou outro usuário que tenha a necessidade de informações gerenciais. Com a modelagem de um DW os usuários finais facilmente entendem e navegam pela solução da estrutura de dados resultante [22].

Nos tópicos a seguir serão apresentados os modelos propostos por Ralph Kimball e Bill Inmon e algumas estratégias de conversão do modelo transacional para o modelo de dados do DW.

#### 3.5.1 Modelo de Dados Segundo Ralph Kimball

Os modelos Entidade/Relacionamento<sup>10</sup> não podem ser utilizados como base para um Data Warehouse pois as consultas são muito difíceis de serem entendidas tanto pelo usuário quanto pelos softwares de SGBD (Sistemas Gerenciadores de Banco de Dados). Essas dificuldades podem ser resolvidas através da implantação de modelos dimensionais.

---

<sup>9</sup> Formas normais são regras para evitar que um dado apareça mais de uma vez no banco de dados

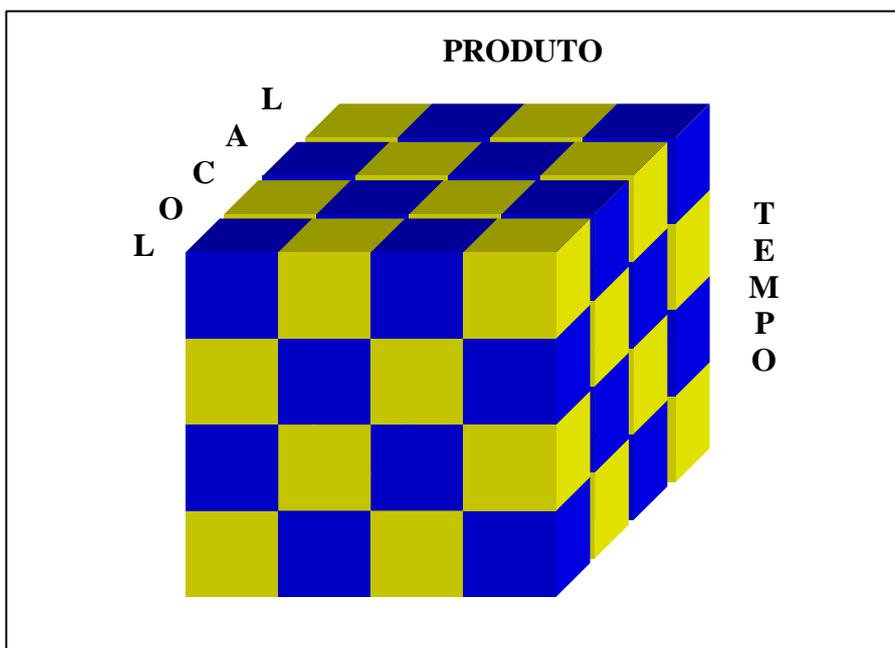
<sup>10</sup> Tipo de modelagem do ambiente transacional

A modelagem dimensional permite-nos visualizar dados. A habilidade de visualização dos dados é algo bastante abstrato, principalmente quando se trata desse tipo de modelagem.

Resumindo, dados que se encaixam perfeitamente em uma espécie de cubos de dados, onde qualquer ponto no interior dele está na intersecção de suas coordenadas, definidas nas arestas do cubo. Este é o modelo dimensional, ou seja cada ponto interno ao cubo contém as medições para aquela combinação específica.

Perguntas complexas que envolvam questões de análise dos negócios de uma empresa, geralmente requerem uma visão dos dados de várias perspectivas. As respostas a esse tipo de pergunta podem levar a tomada de decisões acertadas ou não.

Um bom exemplo seria uma rede de supermercados que estejam querendo melhorar o desempenho de suas vendas ou saber se suas promoções estão trazendo resultado. Para responder a essas perguntas, faz-se necessário examinar os dados sobre as vendas disponíveis nos vários supermercados. Uma análise deste tipo requer uma visão histórica baseada no volume de vendas, sob múltiplas perspectivas, como por exemplo: a quantidade de vendas por produto, total de vendas por um determinado período de tempo em um determinado supermercado, quais os produtos mais vendidos, entre várias outras perguntas possíveis de serem feitas.



**Fig. 3.9 – Cubo de dados, adaptado [8]**

A seguir será detalhada a estrutura do modelo dimensional, suas variações e como construí-la.

### **3.5.1.1 Modelo estrela (Star Join Schema)**

Um outro nome para o modelo dimensional é o **Star Join Schema** (Esquema em Estrela). É chamado assim por se tratar um modelo bem assimétrico, tendo uma tabela no centro com múltiplas junções conectando-se às tabelas auxiliares exibidas em um padrão radial. Todas as tabelas secundárias possuem apenas uma junção com a tabela central.

Essas tabelas são definidas como:

- Tabela Central – Tabela de Fatos
- Demais tabelas – Tabelas de Dimensão

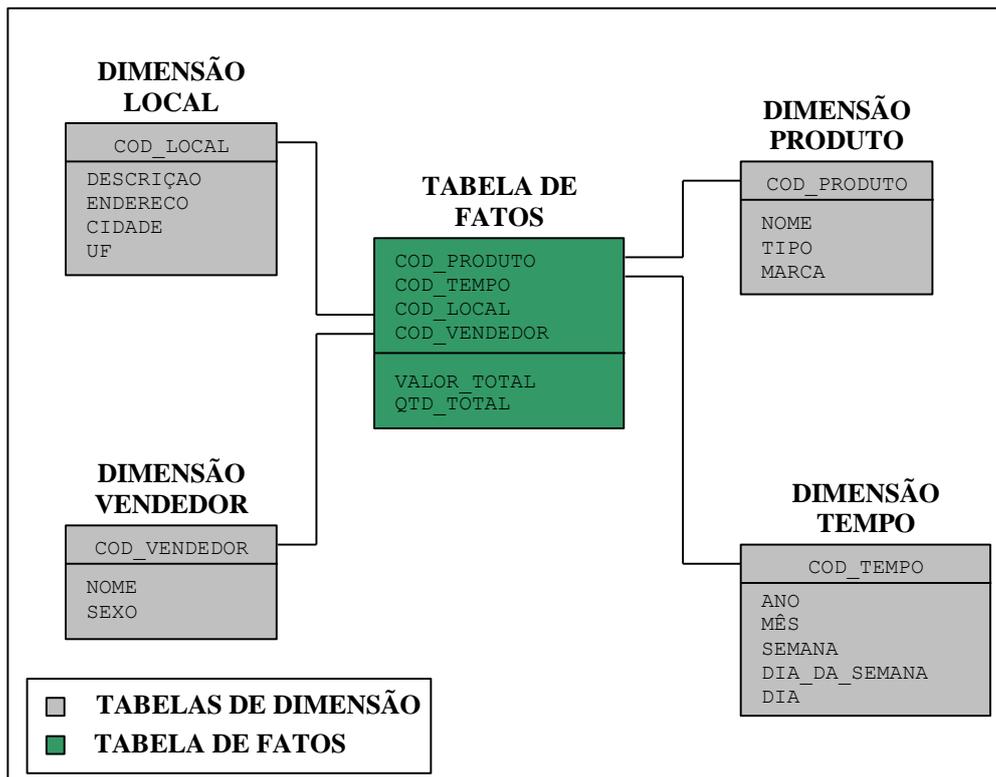
Qualquer combinação das dimensões, gera um registro diferente na tabela de fatos. Esses dois tipos de tabela serão descritos a seguir.

### **3.5.1.2 Tabela de fatos**

A tabela de fatos representa medições numéricas do negócio e estas medidas podem ser obtidas pela intersecção de todas as dimensões.

Os fatos melhores e mais úteis são numéricos, continuamente valorados (diferente a cada medida) e aditivos (podem ser adicionados às diversas dimensões). O motivo dessa utilização é que em praticamente todas as consultas feitas a essa tabela de fatos, serão solicitados centenas, dezenas ou milhares de registros para construir o conjunto resposta. A única forma viável de possibilitar este processo é o adcionamento de dados e se as medições forem numéricas, isso pode ser facilmente obtido.

Os fatos devem ser valorados justamente para distinguir a tabela de fatos das tabelas dimensionais.



**Fig. 3.10 – Modelo estrela (Star Join Schema)**

### 3.5.1.3 Tabela de fatos sem fatos

Alguns processos mercedores de um Data Warehouse produzem tabelas de maneira análoga ao que ocorre com uma tabela fato normal, contendo algumas chaves cada qual representando uma dimensão do evento, mas que não contem fatos mensuráveis. Estas tabelas são chamadas de Tabelas de Fatos sem Fatos (Factless Tables).

Kimball [8] sugere duas variações principais de tabelas sem fatos: Tabelas de Rastreamento de Eventos (Event Tracking Tables) e Tabelas de Cobertura (Coverage Tables).

As Tabelas de Rastreamento de Eventos freqüentemente não estão associadas a fatos numéricos óbvios e têm como principal objetivo à contagem de fatos. Por exemplo, em uma escola verificar quais alunos foram mais assíduos (eventos de freqüência). Estas tabelas freqüentemente não estão associadas a fatos numéricos óbvios.

As Tabelas de Cobertura são frequentemente tabelas de eventos que não ocorreram. Um bom exemplo deste caso é a verificação de quantos itens em uma promoção não foram vendidos.

De modo geral Tabelas de Cobertura não contêm fatos da mesma forma ocorre com as Tabelas de Rastreamento de Eventos [8].

No capítulo 4 será mostrado uma aplicação de tabela de fatos sem fatos no Data Warehouse da Ouvidoria.

#### **3.5.1.4 Tabelas dimensionais**

As tabelas dimensionais representam as descrições textuais das dimensões do negócio, onde para cada registro de uma dimensão existe uma descrição específica para ela. Como por exemplo, em uma dimensão produto em que cada linha dessa tabela representa um produto específico.

Para tabelas dimensionais os melhores atributos são textuais e discretos. Dependendo da combinação desses valores, pode ser definida a consulta com linhas de comando SQL onde pode ser obtido o conjunto resposta, utilizando esses atributos de dimensão para criar os cabeçalhos de linha e definir restrições.

Segundo Kimball [7], modelos dimensionais reais no mundo dos negócios contêm entre 4 e 15 dimensões, sendo raros modelos com 2 ou 3 dimensões. Ainda segundo esse autor, modelos com 20 ou mais dimensões devem ser estudados para se verificar as dimensões supérfluas e/ou combiná-las.

#### **3.5.1.5 Relação entre tabela de fatos e tabelas dimensionais**

Quando é projetado um banco de dados, pode surgir a dúvida sobre um campo numérico extraído de uma fonte de dados de produção: se ele é fato ou atributo. Será um fato se variar constantemente a cada amostragem e será um atributo de dimensão se for estável, como a descrição de um determinado item.

O modelo Estrela normalmente é utilizado em grande parte dos projetos de banco de dados analíticos, no entanto existem algumas situações que se deve abdicar de seu uso, como por exemplo, quando uma tabela dimensional com uma grande quantidade de registros e atributos, pois isso prejudicaria o desempenho das consultas.

Para solucionar esse problema é necessária a adoção de modelos alternativos, como a variação do modelo estrela e tabela de fatos sem fatos (Factless Table), detalhados a seguir.

### 3.5.1.6 Variação do modelo estrela - Snowflake

O modelo *Snowflake* (flocos de neve), é uma variação do modelo estrela no qual as tabelas de dimensão estão normalizadas de acordo com a 3ª Forma Normal<sup>11</sup>, ou seja, além de continuarem ligadas à tabela fato, passam a se ligar a outra tabela de dimensão [7].

Existem alguns motivos para a escolha desse modelo:

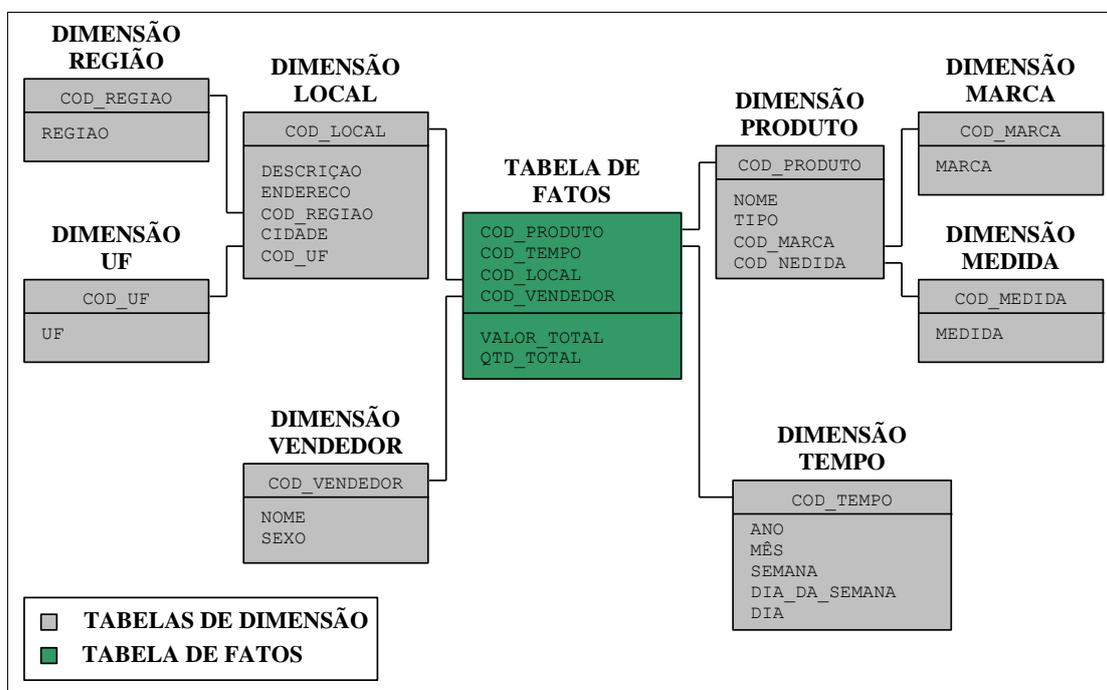
- Representação de hierarquia de atributos apresentada explicitamente pela normalização das tabelas;
- Fácil manutenção;
- Economia de espaço de armazenamento de dados;
- Redução no tamanho das tabelas de dimensão.

Esse tipo de modelagem também possui algumas desvantagens como:

- Aumento da complexidade da arquitetura devido ao aumento do número de tabelas, comprometendo a compreensão por parte do usuário;
- Redução na eficiência da recuperação de dados;
- A economia de espaço de armazenamento, segundo Kimball, não chega a 0,1% [7];
- Criação e gerenciamento de chaves artificiais entre as dimensões.

Kimball aconselha os projetistas a resistirem à tentação de transformar o modelo estrela em modelos *Snowflake* (flocos de neve), devido ao impacto da complexidade deste tipo de estrutura sobre o usuário final e também tendo em vista a performance das consultas que podem ser afetadas, enquanto que o ganho em termos de espaço de armazenamento seria pouco relevante [7]. Entretanto, ele aconselha a

utilização de “subdimensões” em casos particulares, como pode ser observado na figura 3.11 as dimensões Marca e Medida ligadas à dimensão Produto e as dimensões UF e Região, ligadas a Dimensão Local.



**Fig. 3.11 – Modelo Snowflake**

### 3.1.5.7 Passos para a criação das tabelas de fato

Deve-se seguir quatro passos para o desenvolvimento de um modelo de dados dimensional:

1) Escolha do processo de negócio a modelar: A escolha do processo deve ser feita de acordo com o tipo de atividade exercida pelo mesmo, como por exemplo: vendas, faturamento, inventários de estoque, administração de contas entre outros. Um processo de negócio é uma operação importante na organização suportada por algum tipo de sistema (ou sistemas) legado(s) de onde é possível coletar os dados para o DW.

2) Escolha do grão do processo de negócio: O grão é o nível fundamental atômico de dados que representará esse processo na tabela de fatos [8]. O grão também

<sup>11</sup> Uma tabela não pode possuir referência transitiva e sim a própria chave

é importante pois determina a dimensionalidade do modelo assim como o seu tamanho. Alguns exemplos de grãos: transações individuais; instantâneos individuais diários, semanais ou mensais. Esse grão pode ser obtido através da definição do problema do usuário com base em reuniões e entrevistas onde será exposto o seu problema e as suas necessidades reais sobre as informações gerencias.

3) Escolha das dimensões: Uma escolha cuidadosa do grão determina as dimensões primárias das tabelas de fatos. As dimensões constituem a base para se definir os cabeçalhos de linhas dos relatórios finais do usuário. Devem estar de acordo com as denominações tradicionalmente utilizadas pelos usuários, afim de que sejam rápida e amplamente compreendidas. Algumas dimensões típicas são: tempo, produto, cliente, loja, local, promoção, vendedor. Para cada dimensão escolhida, deve-se descrever todos os atributos dessa dimensão.

4) Escolha dos fatos: Essa escolha deve ser feita com base nos fatos mensuráveis que irão popular a tabela de fatos. Alguns fatos típicos são: quantidade vendida, vendas em dólar, custo em dólar.

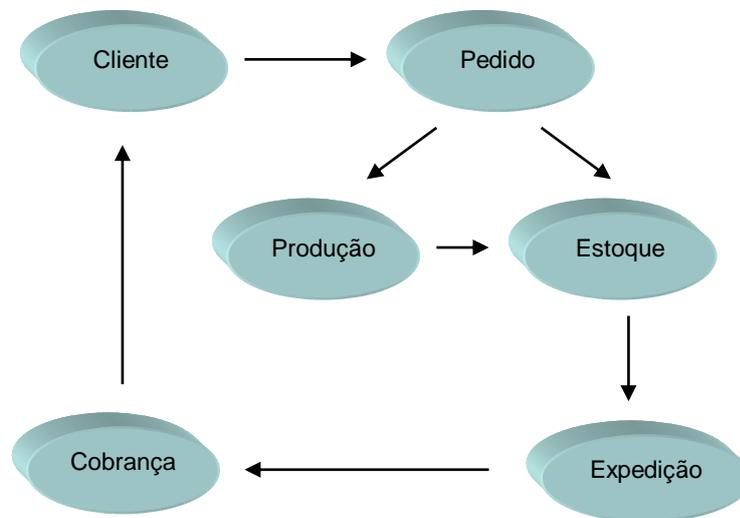
### **3.5.2 Modelo de Dados Segundo Inmon**

Este modelo que será resumidamente apresentado neste trabalho, pode ser pesquisado mais detalhadamente em [6]. Este modelo é composto por três níveis de modelagem: a modelagem de alto nível, a modelagem de nível intermediário e a modelagem de baixo nível.

#### **3.5.2.1 Modelo de dados de alto nível**

O alto nível de modelagem apresenta as entidades e seus relacionamentos, conforme mostrado na Figura 3.12. As entidades exibidas neste nível encontram-se no nível mais alto de abstração. Para determinar quais entidades participam deste nível é necessário estabelecer o “escopo de integração” que é quem define as fronteiras do modelo de dados e deve ser definido antes do início do processo de modelagem. O escopo de integração pode ser definido pelos analistas, pela gerência e pelo usuário final

e deve ser redigido em não mais do que cinco páginas e em uma linguagem de fácil entendimento para as pessoas de negócios [16].



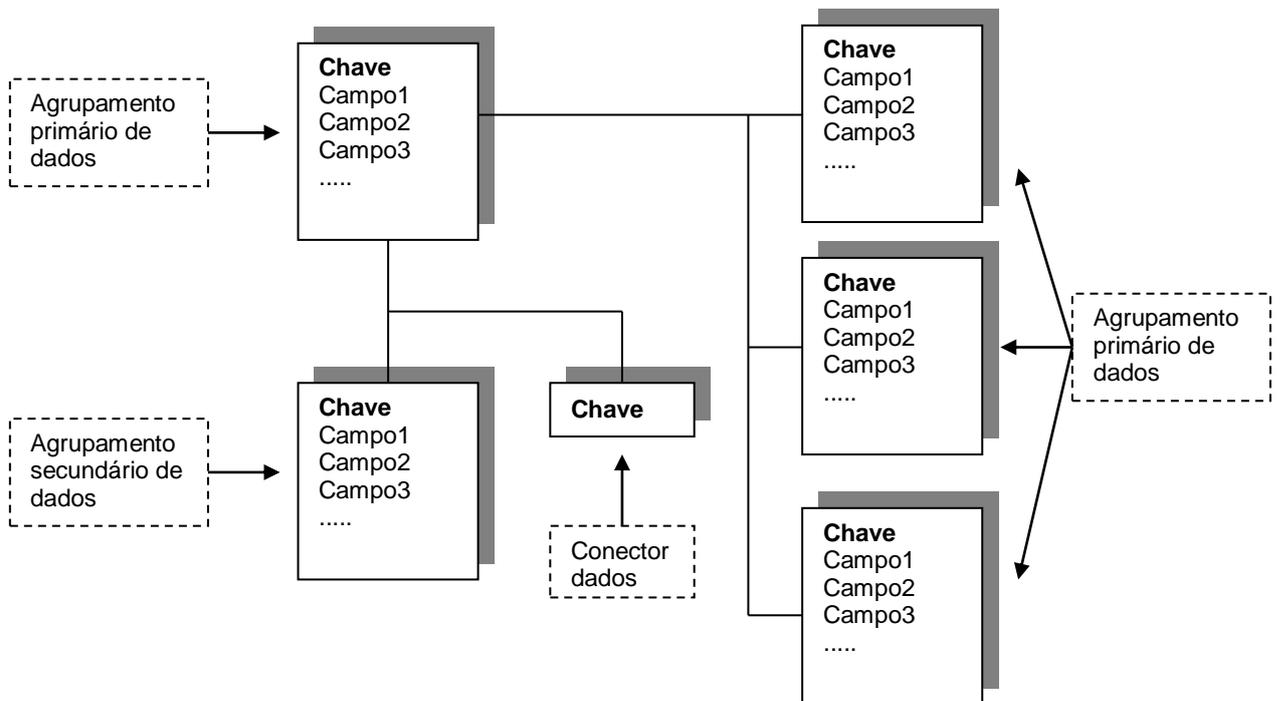
**Fig. 3.12 – Representação da modelagem de alto nível**

### 3.5.2.2 Modelo de dados de nível intermediário

O modelo de dados de nível intermediário é criado a partir das áreas de interesse ou entidades identificadas no nível alto de modelagem, para cada uma destas áreas ou entidades é desenvolvido um nível intermediário próprio. Este nível é composto por quatro elementos apresentados na Figura 3.13.

- ◆ Um **agrupamento primário** de dados, o qual é composto pelos atributos que aparecem uma única vez em cada área de interesse. Como todos os agrupamentos de dados o agrupamento primário contem atributos e chaves;
- ◆ Um **agrupamento secundário** de dados que engloba os atributos que podem aparecer mais de uma vez na mesma área de interesse;
- ◆ Um **conector** que representa os relacionamentos dos dados entre as áreas de interesse;
- ◆ O **tipo dos dados**.

Esses quatro elementos de modelagem são usados para identificar os atributos de dados de um modelo e os relacionamentos entre tais atributos. Normalmente cada agrupamento de dados existente no modelo de dados resulta na definição de uma tabela durante o processo de projeto do banco de dados.



**Fig. 3.13 – Os elementos do modelo de dados de nível intermediário [3]**

### 3.5.2.3 Modelo baixo nível

O modelo físico de dados é gerado a partir do modelo intermediário mediante a simples expansão deste de forma que ele passe a apresentar chaves e características físicas. A essa altura, o modelo físico de dados se parece com uma série de tabelas relacionais.

Neste momento aparentemente o projeto físico do banco de dados poderia ser dado por concluído, entretanto falta ainda uma última etapa deste processo, a inclusão de características de performance.

Neste caso para um Data Warehouse, o primeiro passo seria então decidir sobre a granularidade e o particionamento dos dados. Depois que a granularidade e o particionamento tiverem sido incluídos, várias outras atividades de projeto físico serão adicionadas ao projeto. Essas atividades serão descritas mais adiante no item sobre estratégias de conversão do modelo E-R para o DW.

No centro destas atividades encontra-se a utilização de E/S (entrada/saída) física que é a atividade que realiza o tráfego de dados entre os meios de armazenamento e o computador.

O que torna este evento E/S tão importante para performance é que a transferência de dados entre meio de armazenamento e o computador ocorre de forma aproximadamente duas ou três vezes mais lenta do que a velocidade de funcionamento do computador. A nível de comparação, a velocidade de processamento do computador é medida em nanos-segundos e a velocidade de E/S é medida em milissegundos.

A tarefa do projetista consiste em organizar o banco de forma à retornar o maior número de registros em uma E/S, minimizando o número de E/S. Dessa forma se o projetista tem ciência de que uma consulta é realizada frequentemente e esta busca informações de cinco blocos diferentes, ele poderia agrupar estas informações para minimizar o custo.

Outro fator que permite esses agrupamentos é que os dados de um Data Warehouse, normalmente, não são atualizados. Isso deixa o projetista livre para utilizar técnicas de projeto físico que não seriam viáveis caso os dados fossem atualizados constantemente.

### **3.5.3 Estratégias de Conversão do E-R para o Modelo de Dados Data Warehouse**

Na construção do modelo do DW, Inmon recomenda a utilização do modelo de dados corporativo<sup>12</sup>, pois este modelo já possui todos os atributos necessários para registrar as informações dos sistemas operacionais da empresa. Contudo, um considerável número de alterações é feito no modelo corporativo de dados, quando este é aplicado ao DW. Para este procedimento Inmon oferece uma série de passos que

---

<sup>12</sup> Modelo de dados de uma determinada organização

podem ser seguidos considerando que as decisões de transformações tem que ser baseadas nos requisitos da empresa [3].

Esses passos são: remoção de dados que são usados apenas no ambiente operacional, adição do elemento de tempo nas estruturas de chaves, inclusão de dados derivados, transformação dos relacionamentos em artefatos no Data Warehouse, inclusão de características de performance e realização de uma análise de estabilidade dos dados.

### **3.5.3.1 Remoção dos dados puramente operacionais**

A primeira ação tomada na passagem do modelo corporativo para o Data Warehouse consiste na remoção dos dados puramente operacionais. Esses dados dizem respeito apenas às atividades diárias envolvidas na execução do negócio da empresa, ou seja, dados como mensagens, descrições e status, não são utilizados no contexto de um SAD. Dessa forma esses dados serão removidos ao serem passados para o DW.

### **3.5.3.2 Adição do elemento tempo na estrutura da chave**

O segundo passo a ser tomado é a adição do elemento tempo na estrutura da chave das tabelas, se este já não fizer parte das mesmas.

Como em um data warehouse o elemento tempo tem papel fundamental nas pesquisas e agrupamentos realizados é importantíssimo que este elemento esteja atrelado ao dados como parte da chave. Dessa forma seria possível verificar no Data Warehouse quais transformações um determinado dado sofreu durante um período de tempo dentro do ambiente operacional.

O elemento tempo pode ser adicionado de duas maneiras: adicionando um campo marcando a data da condição do dado ou adicionando dois campo um com o início e fim de condição. Neste caso poderia ser representada tanto uma data fixa no tempo como período de ocorrência de uma situação.

### **3.5.3.3 Inclusão de dados derivados**

O passo seguinte é a inclusão de dados derivados ao modelo do Data Warehouse, sendo que estes não existem no modelo operacional. Os campos derivados

são aqueles que, no ambiente operacional, precisam ser calculados cada vez que tem que ser exibidos.

No Data Warehouse então estes são calculados apenas uma vez e armazenados, economizando assim o processamento despendido na consulta desses dados. Outra vantagem é a diminuição no risco do uso de algoritmos de cálculos falhos, pois após ser armazenado o dado não precisa mais ser recalculado aumentando assim a integridade dos dados.

#### **3.5.3.4 Transformação de relacionamentos entre dados em artefatos de dados**

Os relacionamentos das modelagens clássicas assumem que há somente um valor de negócio no relacionamento. Levando em consideração que o mais importante para os sistemas operacionais é o dado estar íntegro durante a transação, esta abordagem é correta.

Em um Data Warehouse devido ao seu caráter histórico e de apoio à decisão, os relacionamentos podem ter muitos valores para um mesmo par de tabelas. Assim sendo, a melhor maneira de representar um relacionamento é através da criação de artefatos.

Um artefato é composto pela parte do relacionamento que é óbvia e tangível no momento da extração. Dessa forma o artefato pode incluir chaves estrangeiras e outros dados importantes, tais como colunas associativas, ou somente os dados relevantes, sem incluir as chaves estrangeiras.

O problema desta definição é que as alterações que ocorreram até o momento da extração seriam perdidas, pois depois de várias alterações o que valeria é a situação atual. Uma opção para resolver este problema seria guardar registros históricos dessa forma teria todos as alterações dos dados.

#### **3.5.3.5 Acomodação em diferentes níveis de granularidade**

A acomodação em diferentes níveis de granularidade é o primeiro passo tomado na inclusão das características de performance. Quando a granularidade apresentada pelo Data Warehouse é diferente da apresentada pelo sistema transacional é necessário que os dados sejam acomodados à granularidade do DW.

Por exemplo, em um caso que um registro é armazenado diariamente, ao ser passado para o DW este pode ser agregado mensalmente, mudando assim sua granularidade de dia para mês.

### **3.5.3.6 União dos dados comuns de diferentes tabelas**

Acomodados os dados nos níveis de granularidade é necessário considerar a possibilidade de combinar duas ou mais tabelas do modelo corporativo em uma única tabela do modelo do Data Warehouse. Para essa combinação ser realizada, devem ser atendidas algumas condições: as tabelas devem compartilhar uma chave comum, parcial ou não; os dados das diferentes tabelas normalmente devem ser usados juntos e o padrão de inclusão nas tabelas deve ser o mesmo.

Com a combinação destas tabelas o número de E/Ss é diminuído e a performance melhorada, pois acaba a necessidade da junção (*join*) de várias tabelas para realizar uma consulta.

### **3.5.3.7 Criação de array de dados**

Os dados do modelo corporativo normalmente estão bastante normalizados, onde a existência de grupos repetitivos não é permitida. Entretanto, em algumas situações no ambiente de DW pode haver redundância controlada. As condições para isso acontecer são: o número de ocorrências de um dado é previsível, a ocorrência dos dados é relativamente pequena (tamanho físico), as ocorrências dos dados geralmente são usadas juntas e a inserção dos dados é estável.

Por exemplo, em um caso em que é guardada apenas informações de valores mensais sobre um determinado dado, pode ser criado um *array* no Data Warehouse para o valor anual, de forma que cada mês represente uma ocorrência neste *array* ao invés de representar um registro. Esta abordagem gera economia de espaço e torna mais fácil a organização das ocorrências anuais em uma única localização física, gerando assim a possibilidade do aumento de performance. É claro que esta melhoria depende do SGBD usado e da organização física dos registros dentro deste.

### **3.5.3.8 Separação dos atributos de dados de acordo com sua estabilidade**

A análise de estabilidade é uma tarefa que consiste em agrupar atributos de dados segundo a propensão a alterações. Esse agrupamento pode ser feito da seguinte maneira: os atributos que raramente são alterados, os atributos que são alterados algumas vezes e os atributos que são alterados frequentemente. O resultado final da análise de estabilidade é a criação de grupos de dados que apresentam características semelhantes.

## **3.6 DESENVOLVIMENTO DE UM DATA WAREHOUSE**

Tendo definido a arquitetura, a granularidade e a modelagem a ser utilizada, pode ser dado seguimento ao processo de desenvolvimento do DW.

O conceito de Data Warehouse é relativamente novo, e por isso ainda há poucas e diferentes metodologias de desenvolvimento a se consultar, sendo ainda a melhor forma de aprendizado, a tentativa e erro [5]. Desta forma, não existe uma padrão ou uma receita a ser seguida para o desenvolvimento de um DW e sim, um conjunto de ferramentas que contemplam desde a etapa de extração e análise de dados até a construção propriamente dita e o gerenciamento do Data Warehouse.

Determinar o tamanho do projeto e o ponto inicial é um fator crítico para o desenvolvimento do DW, já que devido o projeto estar ligado aos negócios da organização, o valor do investimento geralmente situa-se na casa dos milhões de dólares. A maioria dos projetos de sucesso imediato, começou focalizando as necessidades de uma comunidade de usuários bem definida, e por isso é extremamente importante, que além da participação da equipe responsável pela parte técnica, a colaboração constante dos profissionais da área executiva, pois qualquer passo em falso pode causar graves prejuízos e levar a organização a consultar informações não confiáveis, tendo como principal consequência, a tomada de decisões erradas com base nestes dados.

Este item aborda as fases de um projeto de um DW e os problemas geralmente encontrados no seu desenvolvimento além das funções desempenhadas dentro do ambiente do DW. Para entender as fases de um projeto de DW é necessário primeiro conhecer detalhadamente os conceitos e definições de Data Mart.

### 3.6.1 Data Marts

Segundo Kimball [7], Data Mart (DM) é um subconjunto lógico de um Data Warehouse completo customizado para um departamento em particular, em que o objetivo é o processamento analítico dos dados (Sistemas de Apoio a Decisão). Conforme Inmon [3] os “Data Marts são subconjuntos de dados de uma empresa armazenados fisicamente em mais de um local, geralmente divididos por departamento (Data Marts ‘departamentais’)”.

O Data Mart é a forma mais simples de Data Warehouse, constituindo-se de um pequeno DW, com limitado escopo de dados, projetado para representar uma função particular do negócio como vendas, marketing ou finanças. Frequentemente são constituídos e controlados por um único departamento dentro de uma organização.

Dado seu foco em um único assunto, os data marts geralmente são populados somente de algumas fontes de dados. As fontes podem ser de sistemas transacionais internos, um DW central ou dados externos [23]. As principais características de um Data Mart são:

- Rapidez na implementação;
- Baixo custo;
- Controle local em vez de centralizado;
- Redução do tempo de resposta a consultas tornando a relação entre custo-benefício muito favorável, em contraste com o esforço prolongado de modelagem, tempo de desenvolvimento e recursos financeiros exigidos pelo DW.

Data Marts possibilitam também às organizações analisarem dados históricos armazenados em bancos de dados multidimensionais e obterem informações adequadas no momento oportuno, se tornando mais competitivas perante outras organizações e garantindo desta forma a sua própria sobrevivência.

A tabela a seguir sumariza as diferenças básicas entre um DW e um Data Mart levando em consideração que nenhuma destas definições limita o tamanho de um DM ou a complexidade dos dados de suporte a decisão contidos nele.

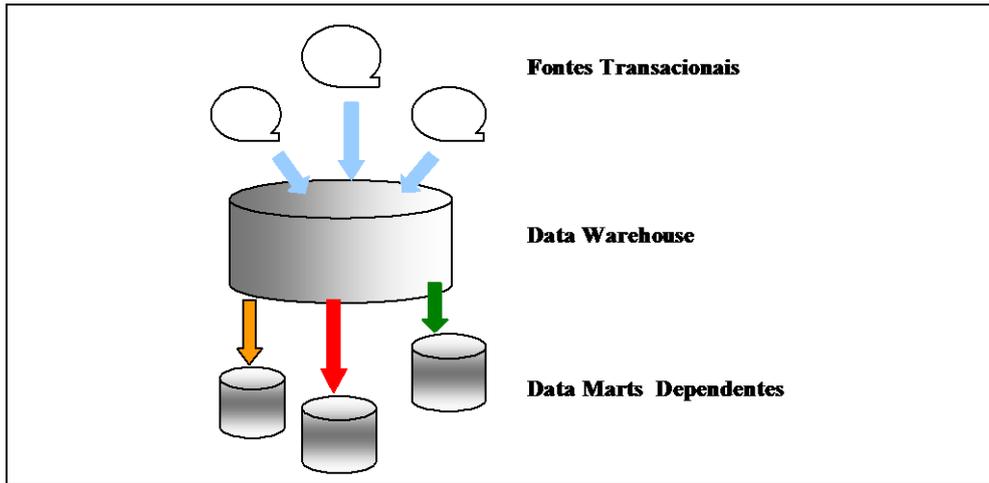
| <b>Propriedades</b>    | <b>Data Warehouse</b>                  | <b>Data Mart</b>                                    |
|------------------------|--|---|
| Escopo                 | Centralizado e compartilhado (Negócio) | Departamento ou área de usuário (Linha de Negócios) |
| Assuntos               | Vários assuntos                        | Poucos assuntos                                     |
| Origem dos Dados       | Várias origens                         | Poucas Origens                                      |
| Características        | Flexível                               | Restritivo  |
|                        | Nível Estratégico                      | Nível tático  |
| Tamanho Típico         | 100 GB – TB                            | Menor 100 GB  |
| Tempo de Implementação | Meses a Anos                           | Meses   |

**Tab. 3.2 – Comparativo entre DW e DM adaptado de [23]**

Os Data Marts são classificados quanto a sua arquitetura em Dependentes e Independentes como descrito a seguir.

### **3.6.1.1 Data Marts Dependentes**

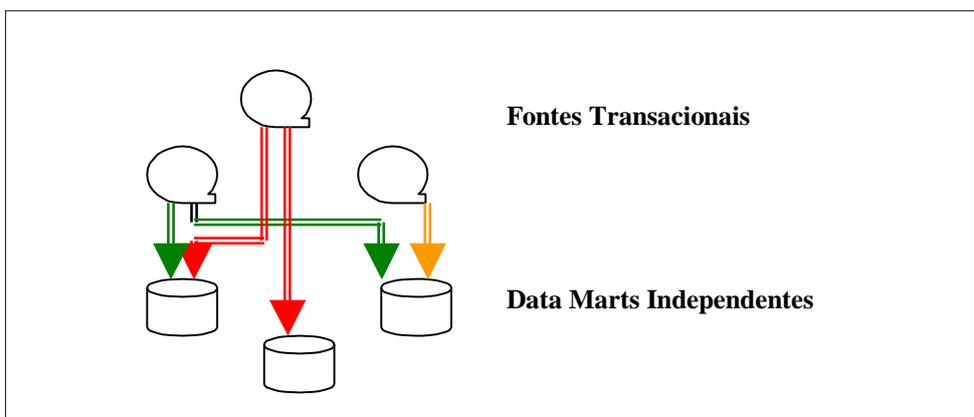
Os Data Marts dependentes são construídos diretamente a partir dos dados de um Data Warehouse central, que por sua vez, possui sua fonte de dados nos sistemas legados e em outras fontes externas à organização. Um Data Mart dependente permite a união de dados da organização em um DW, partindo para segmentação, ou seja, divide o DW em áreas menores gerando assim bancos orientados por assuntos departamentalizados. Isto dará as vantagens usuais de centralização dos dados, conforme ilustra a figura 3.14.



**Fig. 3.14 – Data Marts dependentes, adaptado [23]**

### 3.6.1.2 Data marts independentes

Um DM independente é construído diretamente de uma ou mais aplicações (geralmente mais de uma) sem o uso do DW central, tendo seus dados vindos diretamente das fontes transacionais, fontes externas ou até mesmo ambos. Este tipo de arquitetura pode ser desejável para grupos menores dentro de uma organização, em um projeto de escopo relativamente menor.



**Fig. 3.15 – Data Marts independentes, adaptado [23]**

### 3.6.1.3 Data mart dependente X data mart independente

A tecnologia usada tanto no DW como no DM é a mesma. As variações que ocorrem são mínimas, tendo no volume de dados e tempo de implementação as principais vantagens para sua construção. Porém, o processo de integração entre os diversos DM constituintes de um DW deve ser analisado.

Uma das diferenças mais importantes entre Data Mart dependente e independente é a existência de dados detalhados que residem ou não dentro dele. Em um DM independente, onde cada departamento vai diretamente à aplicação a fim de recolher os dados necessários, os dados detalhados são capturados e colocados dentro do mesmo. Com Data Marts dependentes, o DW captura e integra todos os dados detalhados que são necessários e não há a necessidade de trazer dados detalhados ao DM dependente desde que residam no DW.

O desenvolvimento de Data Marts independentes sem um planejamento global acarreta na fragmentação de dados de uma organização e inibe a utilização de informações de forma integrada na corporação, podendo fazer surgir e proliferar as antigas “ilhas<sup>13</sup>”.

O problema da integração de Data Marts pode, por exemplo, ser superado adotando-se uma solução onde, antes de iniciar o desenvolvimento, deve ser construído um projeto mestre contendo a arquitetura básica e geral de todo o DW, o que permitirá a sua integração no futuro em um empreendimento único ou, então, construir um DW completo na forma de DM distribuídos em unidades individuais [2].

Outra forma de resolver este problema de integração é através da implementação do conceito de “*Data Warehouse Bus*”, onde o objetivo é construir um esquema geral e padronizado de tabelas dimensão e tabelas fato. Por exemplo, supondo que no ambiente de um futuro DW serão utilizadas as dimensões “Tempo”, “Cliente” e “Produto”, deve-se especificar rigorosamente e cuidadosamente os seus atributos de forma que todo e qualquer Data Mart que for constituído e venha a ter uma dessas dimensões, obrigatoriamente obedeça ao padrão estabelecido (mesmos atributos e características). Nesse ambiente padronizado, as tabelas são chamadas de “tabelas de

dimensão conformadas” e “tabelas fato conformadas”, sendo possível a integração dos diversos DM em um DW.

Devido à necessidade de obtenção imediata de informações táticas e estratégicas dos negócios, assim como as restrições de recursos (custos, tempo e conhecimento tecnológico), cada vez mais as organizações estão optando por implementar primeiro os DM para somente depois agregá-los a um Data Warehouse. Cabe a cada organização avaliar a sua demanda e optar pelo melhor investimento a ser feito.

### **3.6.2 Fases de um projeto de Data Warehouse / Data MArts**

O Processo de implementação de um DW/DM tem alguns aspectos que se assemelham ao desenvolvimento tradicional de sistemas e deverá seguir uma metodologia básica semelhante. Entretanto existem diferenças sucintas que devem ser observadas com muito cuidado e detalhamento. Segundo Barbieri [25], os principais passos para o projeto de um DW/DM são:

- Planejamento;
- Levantamento das necessidades;
- Modelagem dimensional;
- Projeto físico dos banco de dados;
- Projeto de extração, transformação e carga;
- Desenvolvimento de aplicações;
- Validação e Teste;
- Treinamento;
- Implantação.

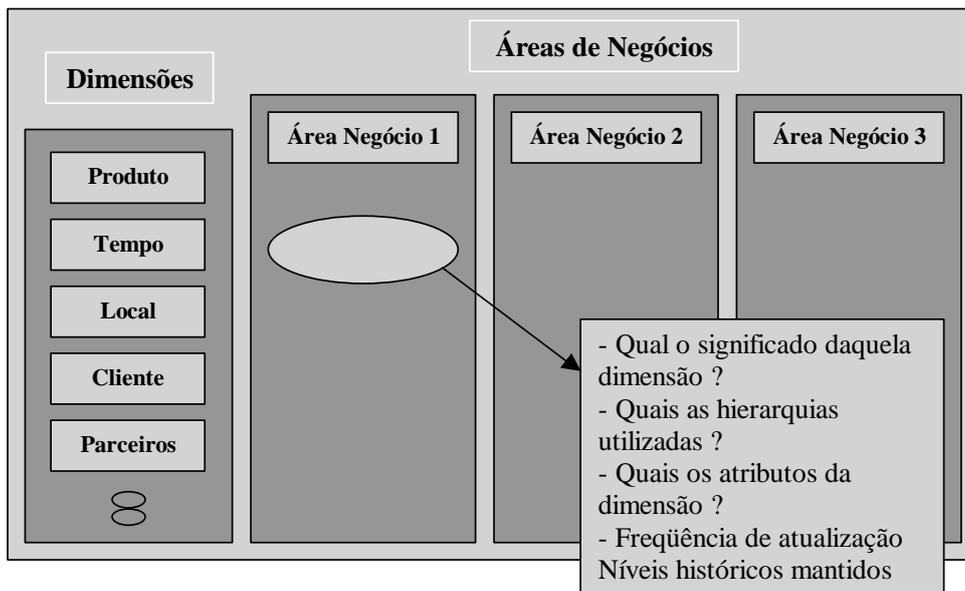
Todas as fases serão detalhadas a seguir.

---

<sup>13</sup>Termo oriundo da época dos sistemas legados e diz respeito à falta de integração e compartilhamento de dados entre os sistemas operacionais

### 3.6.2.1 Planejamento

Esta primeira etapa do projeto de DW/DM, define-se o escopo do projeto, sempre tendo em vista o foco do negócio, atentando-se para as áreas mais críticas da empresa e as necessidades mais prementes de informações gerenciais. A definição da abordagem corporativa vai desde a escolha de um DW monolítico, grande, fortemente integrado em nível de projeto do qual sairão os Data Marts posteriormente (abordagem monolítica ou *top-down*) até uma alternativa gradativa, onde os Data Marts evolutivos integrarão o DW na medida de suas implementações (abordagem evolutiva ou *bottom-up*). Definidas as áreas/assuntos do primeiro projeto, é feito o planejamento para a integração dos DM de forma gradativa em direção ao DW identificando os elos entre eles que possibilitarão as conexões futuras e integrações sem grandes traumas [Figura 3.16].



**Fig. 3.16 – Matriz para planejamento da integração evolutiva de Data Warehouse**

[25]

Antes de se iniciar o projeto de DW/DM é essencial ter atenção à arquitetura tecnológica que servirá de base para o projeto. É fundamental que os componentes

básicos de uma arquitetura sejam definidos antes do início do projeto de DW/DM, devido a fatores relacionados a performance e disponibilidade que podem interferir em níveis de serviço e compromissos variados durante o projeto. De acordo com Barbieri [25], os componentes tecnológicos básicos que deverão ser observados, além da rede corporativa, são:

- Sistema Gerenciador de Banco de Dados: deve residir em uma máquina robusta e dependendo do projeto, em um ambiente de alta disponibilidade, performance e segurança.
- Ferramentas de Desenvolvimento de Sistemas OLAP: produtos que desenvolvem e executam aplicações OLAP.
- Ferramentas para ETL (Extração Transformação e Carga): o aspecto de qualidade de dados de um Data Warehouse é de fundamental importância, e a escolha destas ferramentas, aliada a integridade dos dados fonte, se tornam fatores críticos de sucesso do projeto.
- Catálogo de controle de Metadados: ferramenta desenvolvida internamente ou adquirida no mercado que visa o controle dos metadados do projeto e dos processos de ETL.
- Mecanismos para transferência de dados entre ambientes heterogêneos: genericamente denominados *Middlewares*, facilitam o processo de transferência entre ambientes heterogêneos.
- Servidor de Data Mart/Cubos: ambiente onde reside o gerenciador dos Data Mart.

### **3.6.2.2 Levantamento de necessidades**

Nesta etapa deverão ser identificados dois modelos. O primeiro modelo é o modelo Dimensional, ou aquele que representa os blocos conceituais de dados necessários ao alcance dos objetivos do sistema de suporte a decisão. O outro modelo é relacionado com as fontes das informações. É o modelo Fonte dos Dados. Nele deverão ser registrados os blocos conceituais de dados existentes, com suas respectivas descrições e formas atuais de armazenamento e de uso nos sistemas.

### **3.6.2.3 Modelagem dimensional**

A modelagem de dados é seguramente um dos fatores críticos de sucesso num projeto de DW, e pode representar a fronteira entre o sucesso e o seu fracasso. Os volumes brutos dos dados deverão ser cuidadosamente considerados no projeto, visando processamento para obtenção das informações sumarizadas e consolidadas.

### **3.6.2.4 Projeto físico de banco de dados**

Nessa etapa serão desenhadas as estruturas lógicas do modelo dimensional, com as definições das tabelas fatos e dimensões, relacionamentos, indexação, atributos de tabelas e implantação de regras.

### **3.6.2.5 Projeto de extração, transformação e carga**

Nessa etapa deverão ser definidos os processos requeridos de transformação do modelo fonte para o modelo dimensional. Segundo Barbieri [25], os conceitos de extração dos dados e de seu tratamento podem ser divididos em :

- Filtro de Dados: relaciona os procedimentos e condições para se eliminar os elementos de dados indesejáveis no modelo dimensional.
- Integração dos Dados: define a forma de se correlacionar informações existentes em formas distintas, e que deverão ser integradas no sistema gerencial.
- Condensação dos Dados: define forma de se reduzir volumes de dados visando obter informações resumidas e sumarizadas.
- Conversão de Dados: define os procedimentos para se transformar dados em unidades, formatos e dimensões diferentes.
- Derivação de Dados: define os meios e formulas para se produzir dados virtuais, a partir de dados existentes.

### 3.6.2.6 Desenvolvimento de aplicações

Nessa etapa será projetado o sistema aplicativo, objeto do trabalho. As ferramentas devem ser de fácil utilização e devem priorizar a interface Web, facilitando o acesso aos dados via *browser*.

### 3.6.3 Problemas Encontrados no Desenvolvimento de um Data Warehouse

Durante o desenvolvimento de um Data Warehouse, com base nas etapas do projeto descritos anteriormente, podem ocorrer vários problemas. Para ajudar o gerente do DW a detectá-los Barquini cita os mais comuns [13].

1) Começar sem o envolvimento da diretoria da organização: Para que o projeto de DW tenha sucesso e continuidade é necessário que a alta diretoria esteja comprometida com o projeto, garantindo as verbas necessárias e ajudando a direcionar o foco do DW para o negócio da organização.

2) Criar expectativas que não poderão ser cumpridas: “O DW mostrará aos gerentes as melhores decisões”. Informações erradas como esta podem criar expectativas além do que o DW realmente poderá fazer, isto é um grave erro e pode comprometer a continuidade do projeto assim que os gerentes descobrirem que o DW não mostrará as melhores decisões, mas sim respostas as consultas efetuadas. Cabe aos usuários elaborar consultas inteligentes e analisar as respostas obtidas.

3) Carregar o DW com informações somente porque elas existem: Nem todos os dados disponíveis nos sistemas transacionais da organização são necessariamente úteis para o DW. O arquiteto dos dados deve analisar junto aos usuários quais os dados que realmente contém informações necessárias e desprezar aqueles que não fazem parte dos objetivos do DW.

4) Acreditar que o projeto do banco de dados do DW é o mesmo que o projeto de um sistema transacional: Em um processo transacional, o projeto deve fornecer velocidade de acesso e facilidades na atualização de registros. O DW é fundamentalmente diferente. A meta no DW são acessos agregados, ou seja, somas, médias, tendências, etc. Outra diferença entre os dois tipos de sistemas é o tipo de

usuários. Nos sistemas transacionais um programador desenvolve uma consulta que poderá ser utilizada milhares de vezes. No DW o usuário final desenvolve suas consultas que podem ser utilizadas somente uma vez.

5) Escolher um gerente para o DW com orientação técnica: O DW é essencialmente uma prestação de serviços e não um serviço de armazenamento de dados por isso é fundamental que o gerente do DW seja uma pessoa voltada aos interesses dos usuários e principalmente que fale a mesma “língua” deles.

6) Focalizar-se em dados do tipo registros: Muitas vezes os projetos de DW partem do princípio que as informações necessárias ao bom desempenho do DW estão somente em forma de registros nos arquivos dos sistemas transacionais da organização. Isto pode ser um equívoco, já que muitas informações podem estar armazenadas fora dos sistemas transacionais em forma de textos, imagens, sons e vídeos.

7) Acreditar nas promessas de performance, capacidade e escalabilidade dos fornecedores: A informática cresce de uma maneira muito rápida, isto também acontece com o tamanho do DW, portanto é interessante fazer um estudo de crescimento do DW antes de definir a configuração, que deve atender com folga o banco de dados do DW pelo menos até a conclusão do projeto inicial. É interessante que o servidor do banco de dados do DW seja fornecido por uma organização idônea e que garanta futuras expansões.

8) Acreditar que quando o DW estiver rodando seu problemas estarão terminados: Assim que o DW começar a rodar, os usuários começarão a criar mais consultas e estas consultas necessitarão de novos dados que resultarão em novas consultas. Assim, o projeto do DW precisa ser atualizado continuamente, não só com novos dados mas também com novas tecnologias.

### **3.6.4 Requisitos de Pessoal para o Data Warehouse**

Para criar e manter um Data Warehouse é necessário desenvolver uma série de funções. Dependendo do tamanho do projeto e o tipo de tecnologia utilizada, podem ser necessárias várias pessoas para realizar as diferentes funções previstas, e, ao mesmo tempo, deve-se manter a equipe de Data Warehouse relativamente pequena, para que

possa responder de maneira rápida às necessidades de informação em constante mudança em toda organização [5].

A maneira mais eficaz de arquitetar um DW que transforme dados em informação é ter na equipe, funcionários que já fazem parte da organização com habilidades de negócios necessárias para executar certas funções. Estas funções podem variar conforme o estágio em que se encontra o DW, bem como, podem ser agrupadas para que uma só pessoa realize várias delas ao mesmo tempo.

Segundo Inmon [5] o quadro sugerido para o desenvolvimento inicial de um Data Warehouse é:

| <b>Função</b>                           | <b>Descrição</b>  | <b>Ftes<sup>14</sup><br/>Mínimos<br/>Iniciais</b> |
|---|---|---|
| Administrador de Data Warehouse         | Gerencia o projeto total, desde incursão na implementação inicial ate manutenção e iterações adicionais   | 1   |
| Gerente de Mudança Organizacional de DW | Gerencia as expectativas e percepções da organização quanto ao Data Warehouse, suas capacidades, limitações e o impacto em todos os aspectos da organização                       | Menos de 1  |
| Administrador de Banco de Dados         | Cria e gerencia o(s) banco(s) de dados físicos que compõe(m) todos os níveis e aspectos do Data Warehouse   | 1   |
| Gerente de Metadados                    | Gerencia metadados de negócios e técnicos, assegurando que estejam atuais, precisos e adequadamente integrados  | Menos de 1  |
| Analista de necessidades de negócios    | Identifica e analisa as necessidades de informação de negócios da organização e auxilia no projeto do DW para satisfazer essas necessidades                                       | 2   |
| Arquiteto de Data Warehouse             | Cria e mantém os modelos de dados para todos os níveis de do DW. Analisa sistemas de origem para determinar sistema(s) de registro. Projeta aquisição de dados e ambiente técnico | 1   |
| Desenvolvedor de Aquisição de Dados     | Cria e mantém programas e processos que executam a  | 1 a 3   |

<sup>14</sup> Full Time Equivalents – Funcionários equivalentes em tempos integral.

|                                     |   |  |
|-------------------------------------|---|--|
|                                     | extração, transformação e carga de dados das origens até os alvos   |  |
| Desenvolvedor de Acesso a Dados     | Cria e mantém programas e processos e “caminhos” predefinidos que permitem que usuários finais acessem dados do DW                                    | 1 a 3                                  |
| Desenvolvedores de Manutenção de DW | Cria e mantém programas e processos e procedimentos para executar tarefas de manutenção como arquivamento, recuperação, segurança, monitoração e etc. | 1                                      |
| Responsável executivo de Si         | Responsável pelo suporte ao projeto de DW em termos de fornecimento de fundos, recursos e de representação frente aos seus colegas de alto escalão    | 1 (Não é uma tarefa de tempo integral) |
| Analista de Qualidade de Dados      | Monitora e assegura qualidade de dados no DW que satisfaça as necessidades da organização   | Mais do que 1                          |

**Tab. 3.3 - Quadro de funcionários sugeridos para desenvolvimento inicial de um Data Warehouse [5]**

| <b>Função</b>                           | <b>FTEs Adicionais por Iteração</b>   |
|---|---|
| Administrador de Data Warehouse         | 0 (Líderes de equipe podem ser requeridos se a funcionalidade do DW crescer a um tamanho que os torne necessários). |
| Gerente de Mudança Organizacional de DW | Possivelmente 1 após diversas iterações.  |
| Administrador de Banco de Dados         | Até 1 para cada 2 ou 3 iterações adicionais.  |
| Gerente de Metadados                    | Mínimo de 1 após a primeira interação e as seguintes.   |
| Analista de necessidades de negócios    | 2 adicionais após a implementação inicial; até 1 para cada 2 ou 3 iterações seguintes.                              |
| Arquiteto de data Warehouse             | Mínimo de 1 após a primeira interação e as seguintes.   |
| Desenvolvedor de Aquisição de Dados     | Até 1 para cada 1 ou 2 novas iterações e para a sua manutenção, dependendo do crescimento da demanda de usuários.   |
| Desenvolvedor de Acesso a Dados         | Até 1 para cada 1 ou 2 novas iterações e para a sua manutenção, dependendo do crescimento da demanda de usuários.   |
| Desenvolvedores de Manutenção de DW     | 0 a 1 após diversas iterações, dependendo da complexidade das   |

|                                |  |
|--------------------------------|--|
|                                | funções de manutenção.   |
| Responsável executivo de Si    | 0 (nenhum recurso adicional necessário)  |
| Analista de Qualidade de Dados | Possivelmente 1 ou mais após a iteração inicial, dependendo da qualidade de dados dos novos sistemas de registro e da qualidade apresentada por sistemas de registro existentes. |

**Tab. 3.4 – Quadro de funcionários de SI sugerido para Gerenciamento Contínuo de Data Warehouse [5]**

A equipe de projeto do Data Warehouse será tipicamente composta de 12 a 15 pessoas que serão responsáveis por criar um Data Warehouse na organização. Inmon [5] sugere que a equipe ideal deve ser composta de projetistas de banco de dados, de pessoas que conheçam bem os sistemas legados por dentro e por fora, para a extração dos dados necessários para o Data Warehouse (arqueologistas de dados), programadores de sistemas para assegurar que os dados dos sistemas legados movam-se de forma limpa e repetitiva pelo Data Warehouse, usuários, instrutores e administradores de Data Warehouse, que em geral, são responsáveis pela coordenação de todos os aspectos do projeto de DW e por equilibrar as necessidades e objetivos de todos os interessados.

### **3.7 EXTRAÇÃO DE INFORMAÇÕES DO DATA WAREHOUSE**

Depois de definido e projetado o escopo do projeto e depois de construído o repositório de dados do DW, é necessário realizar a extração das informações, utilizando ferramentas de acesso aos dados, conhecidas também por ferramentas *front end*.

As ferramentas não podem ser muito complexas porque não serão utilizadas por profissionais da área técnica, mas precisam ser robustas o suficiente para dar agilidade no acesso às informações estratégicas.

Existem várias maneiras de recuperar informações de um DW, uma das formas de recuperação é através do uso de ferramentas baseadas na tecnologia OLAP (On-Line Analytical Processing).

### 3.7.1 OLAP

Processamento Analítico On-line (OLAP - On-line Analytical Processing) é um termo usado para descrever uma abordagem dimensional de suporte a decisão, sendo um método importante na arquitetura do Data Warehouse através do qual os dados podem ser transformados em informação [8].

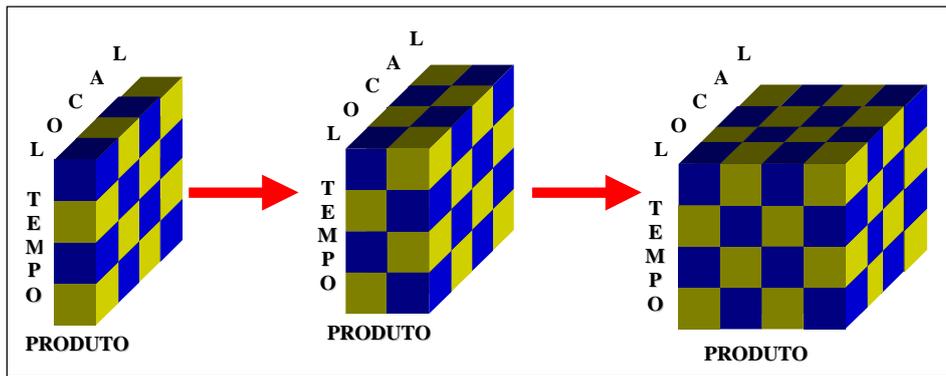
O OLAP é uma extensão natural do Data Warehouse e o Conselho de OLAP o define como uma “categoria da tecnologia de software que permite que analistas, gerentes e executivos obtenham, de maneira rápida, consistente e interativa, acesso a uma variedade de visualizações possíveis de informações que foram transformadas de dados puros para refletir a dimensão real do empreendimento do ponto de vista do usuário” [5].

Segundo Barquini [13], "OLAP é considerado uma categoria de software que permite analistas, gerentes e executivos obter respostas dentro dos dados, através de uma rápida, consistente e interativa forma de acesso a uma ampla variedade de possíveis visões".

Em um modelo de dados OLAP, a informação é conceitualmente organizada em cubos que armazenam valores quantitativos ou medidas. As medidas são identificadas por duas ou mais categorias descritivas denominadas dimensões que formam a estrutura de um cubo. Uma dimensão pode ser qualquer visão do negócio que faça sentido para sua análise. O modelo dimensional permite a visão dos dados de diversas formas, o que o torna extremamente flexível e faz com que seja de grande utilidade para os processos de consultas e análises, sendo uma poderosa ferramenta no suporte ao processo decisório. As figuras 3.17, 3.18 e 3.19 apresentam uma simulação de como visualizar, de diferentes formas, os mesmos dados.

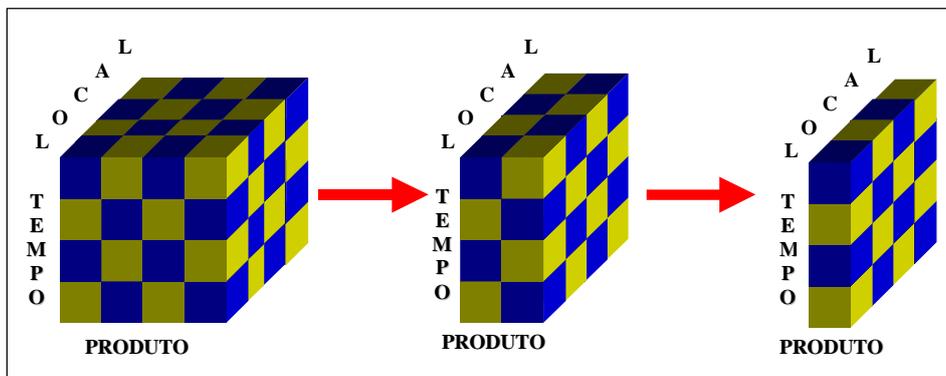
Basicamente existem quatro tipos de operações utilizadas em OLAP para analisar dados:

- *Drill Down* – Com a capacidade de *Drill Down* o usuário pode navegar do mais alto nível até o dado detalhado utilizando da granularidade para efetivar as operações [22].



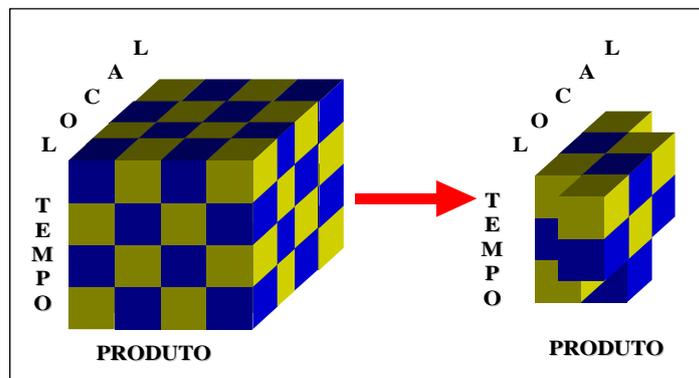
**Fig. 3.17 – Operação de Drill Down**

- *Roll Up* – Com capacidade de *Roll up* o usuário pode navegar do nível detalhado até o mais alto nível de sumarização de dados.



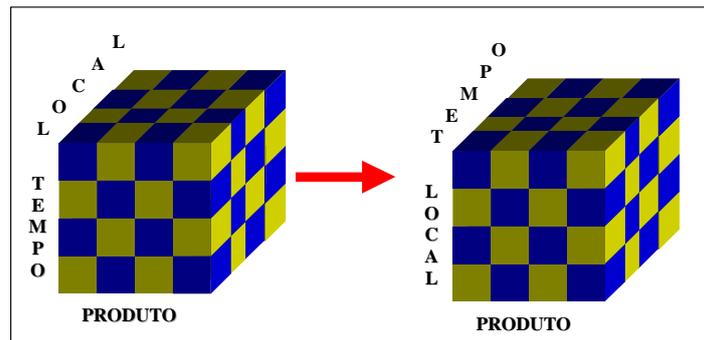
**Fig. 3.18 – Operação de Roll Up**

- *Slice* – É a operação que corta o cubo, mas mantém a mesma perspectiva de visualização dos dados. É como fatiar o cubo para visualizar apenas a fatia que nos interessa.



**Fig. 3.19 – Operação de Slice**

- *Dice* – É a mudança de perspectiva da visão. É como girar o cubo em nossas mãos.



**Fig. 3.20 – Operação de dice**

O conceito de OLAP foi citado pela primeira vez em um artigo escrito em 1992 por E.F.Cood e definia doze regras para avaliar as ferramentas OLAP. As doze regras são apresentadas a seguir [8]:

1) Visão Conceitual Multidimensional – Os usuários podem manipular os modelos multidimensionais de dados mais facilmente e intuitivamente.

2) Transparência – O acesso a qualquer nível de Data Warehouse, incluindo metadados, deveria ser transparente ao usuário. O OLAP deve poder interagir facilmente com os produtos habituais dos usuários, por exemplo: planilhas eletrônicas, processadores de texto ou aplicativos DSS.

3) Acessibilidade – a ferramenta OLAP tem que executar qualquer conversão necessária a fim de apresentar aos usuários uma única, coerente e consistente visão dos dados.

4) Performance de Relatório Consistente – mesmo com o aumento do número de dimensões ou do tamanho do banco de dados, o usuário não deve perceber uma degradação significativa no desempenho do fornecimento de informações.

5) Arquitetura Cliente/Servidor – é necessário que ferramenta seja capaz de operar em um ambiente cliente /servidor, pois atualmente os dados são armazenados em um mainframe ou em servidor de rede, cujos dados são acessados através de computadores pessoais.

6) Dimensionalidade Genérica – a estrutura básica dos dados e o formato dos relatórios não devem ser influenciados por qualquer dimensão de dados.

7) Manipulação Dinâmica da Matriz Esparsa<sup>15</sup> – para qualquer determinada matriz esparsa de dados, existe um e somente um esquema físico ótimo. Este esquema provê máxima eficiência e operacionalidade. A ferramenta deve poder ajustar seu esquema físico, para o máximo desempenho, baseando-se na densidade dos dados armazenados.

8) Suporte Multiusuário – quando vários usuários necessitarem trabalhar simultaneamente com o mesmo modelo analítico ou criar modelos novos, a ferramenta deverá ser capaz de prover esse acesso simultâneo, sem prejudicar a integridade e a segurança dos dados.

9) Operações Através de Dimensões – quaisquer duas ou mais células, apesar de pertencerem a dimensão diferentes dentro de um determinado modelo analítico, devem poder ser usadas para realizar quaisquer cálculos.

10) Manipulação Intuitiva dos Dados – todo o processo de criação de modelos, manipulação de dados e realização de cálculos deve acontecer da forma mais intuitiva possível.

11) Flexibilidade de Relatórios – relatórios devem ser capazes de apresentar os dados de forma sintetizada ou informações que são o resultado de cálculos de um modelo criado, de acordo com qualquer visão possível.

12) Dimensões e Agregações Ilimitadas – qualquer ferramenta OLAP deve poder acomodar pelo menos 15 e até vinte dimensões de dados e deve permitir um número ilimitado de níveis de agregação definidos pelo usuário.

Além destas doze regras o Gartner Group acrescentou mais nove características que permitem uma melhor avaliação das ferramentas OLAP [8]:

- Dados *Arrays* múltiplos;
- OLAP *joins*;

---

<sup>15</sup> Esparsa acontece quando uma pequena porção (arbitrariamente, menor que 0,1%) de célula de dados potencialmente atualizadas está ocupando uma estrutura multidimensional

- Ferramentas para gerenciar as bases de dados;
- Armazenar objetos;
- Seleção de subconjuntos;
- Detalhe *drill-down* em nível de linha
- Suporte a dados locais;
- *Refresh* incremental das bases de dados;
- Interface SQL.

De acordo com Kimball [8], o conjunto de 12 critérios OLAP propostos por E. F. Codd, que supostamente deveria representar um padrão de comparação de sistemas de suporte a decisão, são muito vagos para serem usados como diretrizes de avaliação de um sistema. A filosofia OLAP ainda carece de critérios mais específicos e sólidos.

Quanto à arquitetura OLAP, elas se classificam em: ROLAP, MOLAP, DOLAP, HOLAP e WOLAP.

### **3.7.1.1 Arquiteturas OLAP**

A ferramenta OLAP é constituída de um conjunto de tecnologias especialmente projetadas para dar suporte ao processo decisório através de consultas, análises e cálculos mais sofisticados nos dados corporativos, estejam armazenados em um Data Warehouse ou não, por parte dos seus usuários, geralmente analistas, gerentes e executivos. Existem quatro tipos de arquitetura que serão descritas a seguir:

- ROLAP (Processamento Analítico On Line Relacional): O ROLAP é uma simulação da tecnologia OLAP, feita em banco de dados relacionais, que, por utilizar a estrutura relacional, possui a grande vantagem de não ter restrições no volume de armazenamento de dados. Esta estrutura utiliza o modelo dimensional para gerenciar os dados. As ferramentas *front-end* permitem efetuar requisições multidimensionais, mas o programa de ROLAP transforma consultas em rotinas SQL. A principal vantagem de se adotar uma ferramenta ROLAP é a utilização de uma tecnologia estabelecida, de arquitetura aberta e padronizada como é a relacional, beneficiando-se da diversidade de plataformas, escalabilidade e paralelismo de hardware, além de que a arquitetura, permite analisar enormes volumes de dados. Em contra partida, uma grande quantidade

de usuários acessando simultaneamente, poderá causar sérios problemas de performance no servidor causando, inclusive, o travamento do mesmo.

- **MOLAP (Processamento Analítico On Line Multidimensional):** São OLAP's multidimensionais, onde no SGBD ficam os dados num formato simples, e no Servidor MOLAP, que é na verdade um banco de dados multidimensional, fica os dados consolidados. A arquitetura MOLAP se processa da seguinte forma: com um servidor multidimensional o acesso aos dados ocorre diretamente no banco, ou seja, o usuário trabalha, monta e manipula os dados do cubo diretamente no servidor. Isso traz grandes benefícios aos usuários no que diz respeito à performance, mas tem problemas com escalabilidade além de ter um alto custo para aquisição.

- **DOLAP (Processamento Analítico On Line Desktop):** É uma OLAP que se baseia numa arquitetura DESKTOP, ou seja, é uma ferramenta para usuários que possuam uma cópia de sua base multidimensional ou de um subconjunto dela ou que queiram acessar um repositório de dados central. Basicamente acessa os cubos já existentes no banco de dados ou um conjunto de cubos selecionados pelo usuário. O ganho com essa arquitetura é o pouco tráfego que se dá na rede, visto que todo o processamento OLAP acontece na máquina cliente, e a maior agilidade de análise, além do servidor de banco de dados não ficar sobrecarregado, sem incorrer em problemas de escalabilidade. A desvantagem é que o tamanho do micro-cubo não pode ser muito grande, caso contrário a análise passa a ser demorada e/ou a máquina do cliente pode não suportar em função de sua configuração.

- **HOLAP (Processamento Analítico On Line Híbrido):** A arquitetura HOLAP nada mais é do que uma mistura de tecnologias onde há uma combinação entre ROLAP e MOLAP. A vantagem é que com a mistura de tecnologias pode-se extrair o que há de melhor de cada uma, ou seja, a alta performance do MOLAP com a escalabilidade melhor do ROLAP.

- **WOLAP ou Web OLAP:** As facilidades desta arquitetura são: a possibilidade de plataformas independentes para dar suporte a usuários distantes, aplicações de groupware, facilidade de aprendizado e de manutenção. As dificuldades são: as limitações dos recursos da Internet, as interfaces e as funcionalidades quando comparados com o ambiente cliente/servidor. Novas tecnologias para Internet, como Java e ActiveX estão surgindo para tentar solucionar estas dificuldades.

- A escolha entre DOLAP, ROLAP, MOLAP, HOLAP e WOLAP depende de quais características do problema e de dados, que se deseja dar prioridade. A classificação da arquitetura dependerá do tipo de banco de dados utilizado, multidimensional ou relacional e quanto ao processamento, se realizado no cliente ou no servidor.

### **3.8 CRM – CUSTOMER RELATIONSHIP MANAGEMENT**

A gerência do relacionamento com o cliente é uma abordagem destinada a entender e influenciar o comportamento dos clientes, por meio de comunicações [30].

CRM é um processo interativo que transforma informações sobre os clientes em relacionamentos positivos com os mesmos.

A tecnologia, na forma de técnicas avançadas de transformação e apresentação gráfica de dados, acelera a utilidade e a velocidade da tomada de decisão administrativa. Ela aumenta muito mais o poder do pessoal de contato com clientes, empregados que trabalham com informações, funções de marketing e venda e empregados administrativos, com inteligência de informações sobre negócios significativamente melhores em relação a seus clientes atuais e potenciais.

CRM deve ser integrado em tudo o que a empresa faz, todos com os quais ela trabalha (até mesmo os fornecedores) e todos com os quais ela transaciona. Quando uma empresa diz que seu objetivo é um excelente serviço ao cliente, isso quer dizer toda a empresa.

#### **3.8.1 Visão Estratégica de Data Warehouse e CRM**

Quando sua estratégia é profunda e de longo alcance, então o que você obtém com seus cálculos é muito, de modo que você pode vencer antes mesmo de lutar. Quando seu pensamento estratégico tem pouca profundidade e pouco alcance, então o que você obtém com seus cálculos é pouco, de modo que você perde antes mesmo de lutar.

A estratégia maior prevalece sobre a estratégia menor, de modo que os que não têm estratégia somente podem ser derrotados. Pode-se dizer que os guerreiros vitoriosos vencem primeiro e então vão para a guerra, enquanto os guerreiros derrotados primeiro vão à guerra e então procuram vencer [30].

Inúmeras empresas em múltiplos setores estão obtendo consideráveis benefícios com o data warehouse. As razões típicas dadas são maior rapidez melhor tomada de decisões, aumento do poder dos empregados, integridade dos dados operacionais, análise de cenário, intimidade com o cliente, análise de qualquer coisa e de tudo e controle do processo.

Certamente, essas são boas razões, são motivações adequadas para maximizar o retorno do data warehouse.

Mas, para entender o data warehouse estrategicamente, primeiro deve-se entender a estratégia e o pensamento estratégico. Assim que estiver entendido esses conceitos, a lógica estratégica do data warehouse se tornará clara e surgirá o caminho para uma implementação ótima.

O retorno máximo do data warehouse ocorre quando ele é conceituado, implementado, gerenciado e evoluído dentro do contexto.

### **3.8.2 Ampliando CRM e Data Warehouse**

Ter sucesso na utilização das tecnologias de relacionamento é um processo de aprendizagem em contínua evolução. Os estágios que compõem o processo de maturidade do gerenciamento são independentes e inter-relacionados, e a aprendizagem continua torna-se muito valiosa para uma organização com o passar do tempo. Os períodos de tempo podem ser reduzidos por meio de um entendimento prévio de ações de maturação, experiências e combinação futuras da aprendizagem do gerenciamento. Com base em experiências históricas do passado, conhecer o futuro pode realmente acelerar o processamento de maturação e posicionar para oportunidades extremamente relevantes.

À medida que sua empresa atingir novas oportunidades e potenciais, suas informações proporcionaram muito mais benefícios, com as mudanças que estão

fornecendo mais do que acesso universal aos dados ou um novo sistema de relatórios de gerenciamento sofisticado. O conhecimento sobre o futuro orienta as pessoas esclarecidas no aproveitamento das oportunidades.

Existem seis estágios do crescimento:

- a) Início – Partida: Montagem e aprendizagem;
- b) Crescimento – Aplicação, tecnologia e disseminação dos dados;
- c) Controle – Intervenção/Posicionamento gerencial;
- d) Integração – Info-estrutura empresarial;
- e) Distribuição – Divisão interfuncional, utilizações e mercado;
- g) Estratégia/maturidade – Táticas, planejamento e estratégia compartilhados.

### **3.8.3 Aplicando Política de Privacidade ao Data Warehouse para CRM**

Muitos de nós recebemos ligações telefônicas de empresas de telemarketing citando informações que foram fornecidas por empresas contratadas por meio da Internet, por aplicações financeiras, por financiadoras, por malas-diretas e afins. Ficamos assombrados com certos fatos que essas empresas podem conhecer a nosso respeito. Durante a criação e a administração de bancos de dados, deve haver um conjunto claro de políticas, procedimentos e diretrizes para a coleta, a utilização, a disseminação e a administração de informações sobre clientes e de nossas escolhas sobre os dados.

A privacidade é o assunto de maior preocupação pública em todo o mundo. Ainda que tenha sido uma questão secundária para os administradores de bancos de dados, especialmente em relação a aplicações de bancos de dados de marketing, recentemente a privacidade passou para a linha de frente como um importante foco para a atenção do governo, da mídia e dos consumidores. Isso se deve em grande parte à Internet e, mais especificamente, ao “comércio pela Web”. Mas a questão da privacidade perpassa todas as formas de comércio e não somente no ambiente online.

Um exemplo da crescente coleta de dados pessoais está no varejo, onde recentemente se lançaram cartões de “associação” ou de “lealdade”. Os cartões oferecem aos clientes preços reduzidos para certos produtos, mas a cada vez que o cliente utiliza o cartão, são coletadas informações sobre os seus hábitos de consumo. Em alguns casos, esses dados pessoais são armazenados e processados pelos varejistas em bancos de dados ou *Data Warehouses* para finalidades próprias, sem conhecimento ou aprovação dos clientes. As mesmas informações pessoais também podem, com a tecnologia e os ambientes online atuais, ser coletadas a partir de compras feitas com cartões inteligentes (smart cards), cartões telefônicos e cartões de débito ou crédito.

Em relação a CRM, “privacidade” significa controle e proteção relacionados com a utilização de informações pessoais. “Informações pessoais” são quaisquer informações de natureza pessoal sobre um indivíduo (idade, sexo, renda etc.). Quando são armazenadas informações pessoais em bancos de dados, malas-diretas ou *Data Warehouses*, é obrigatório que os proprietários (isto é, “controladores de dados”) protejam dados de clientes contra abusos.

## Capítulo 4

# CONCEPÇÃO DO PROJETO DW E CRM PARA A OUVIDORIA DA ANEEL

### 4.1 DESCRIÇÃO DO PROJETO

O projeto inicial de Data Warehouse para a Ouvidoria não necessita disponibilizar todas as informações corporativas contidas na ANEEL, mas sim o que está relacionado às necessidades da Ouvidoria. Neste contexto e visando um projeto eficiente e de rápida implementação, partimos para uma solução simplificada, porém com as mesmas características de um Data Warehouse, o Data Mart (Data Warehouse Departamental). O objetivo foi implantar um sistema de apoio à decisão consistente e um projeto CRM que atendesse às necessidades dos usuários envolvidos no processo.

O desenvolvimento de um Data Mart atende a necessidade da Ouvidoria da ANEEL em obter informações precisas sobre a situação dos concessionários de energia elétrica, disponibilizando um ranking de acordo com o quantitativo de reclamações e alguns outros indicadores existentes na base de dados.

Como visto anteriormente, os Data Marts - DM possuem os mesmos conceitos e tecnologias aplicadas em um DW.

A escolha em desenvolver um Data Mart também partiu do interesse em criar um modelo padronizado para ser implantado nos demais departamentos existentes na ANEEL.

A metodologia utilizada para o desenvolvimento do Data Mart foi semelhante a do desenvolvimento tradicional de sistemas, segundo Barbieri [25].

## 4.2 DESCRIÇÃO DA FERRAMENTA UTILIZADA

A ferramenta “Sagent Provides” é mundialmente conhecida no ramo de business intelligence e conta com uma série de aplicativos que tornam possível o desenvolvimento e implantação de um Data Warehouse.

O Sagent efetua tanto as tarefas de “Back Room” quanto as de “Front Room”, já vistas anteriormente e detalhadas segundo Kimball [7].

A seguir, serão mostrados alguns dos aplicativos utilizados do DM da Ouvidoria.

- **Analytical Calculator Transform**

Aplicativo utilizado para transformar os dados ao carregar o modelo dimensional.

- **Analysis**

Aplicativo utilizado para desenvolver e disponibilizar “Front end” para o usuário final.

- **Admin**

Aplicativo utilizado para o controle de conexões ao banco.

- **Design Studio**

Aplicativo utilizado para desenvolver os planos de carga do modelo dimensional.

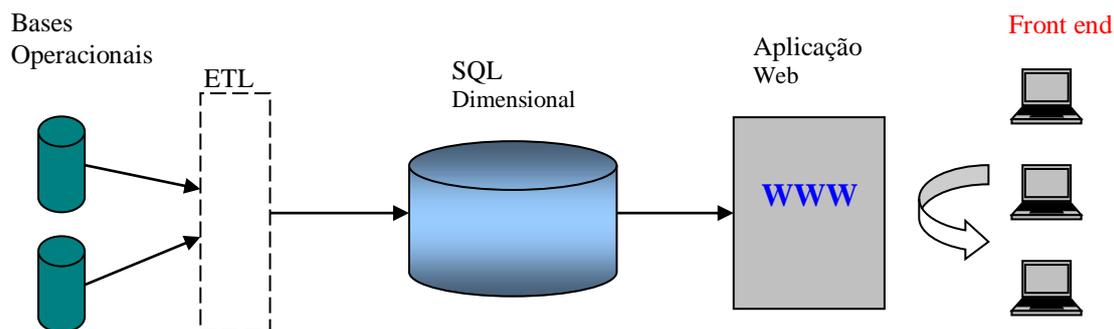
- **Automation**

Aplicativo utilizado para efetuar cargas automáticas, executa os planos de carga desenvolvidos no Design Studio de acordo com a programação estabelecida.

Este aplicativo é de grande importância para o projeto, no caso do Data Mart da Ouvidoria estas cargas são efetuadas na madrugada para não sobrecarregar as bases de dados operacionais.

### 4.2.1 Estrutura da Ferramenta Sagent

A figura 4.1 mostra a estrutura da ferramenta Sagent, para simplificar o sistema, nota-se que o Sagent executa praticamente todo o processo.



**Fig. 4.1 – Estrutura da ferramenta utilizada no projeto DM da Ouvidoria**

O Processo de ETL (Extração Transformação e Carga) é um dos requisitos mais importantes no desenvolvimento de um Data Mart. Com ele é possível integrar dados de várias bases de dados existentes na ANEEL. Neste projeto utilizou-se apenas dados contidos na base de dados da Ouvidoria, entretanto, será apresentado na conclusão a possibilidade de cruzar informações de outras bases existentes na ANEEL, com isso, será possível aumentar a qualidade dos indicadores para avaliar a situação dos concessionários do setor elétrico brasileiro.

### 4.3 DESCRIÇÃO DA ESTRUTURA UTILIZADA

A arquitetura OLAP utilizada foi o ROLAP, baseada em banco de dados relacional, adotando o modelo estrela segundo Inmon [3].

O modelo dimensional projetado para as necessidades da ouvidoria foi armazenado em banco de dados relacional, Microsoft SQL server 2000, montado em uma máquina Dell com 4 processadores Xeon 700 Mhz, 4GB de memória RAM, 18 GB

de espaço interno e 3 Terabytes de espaço externo operando em cluster com outra máquina com a mesma configuração.

O servidor de aplicação Sagent foi instalado provisoriamente no mesmo equipamento, porém, por medidas de segurança e padronização de arquiteturas, será movido para outro servidor cluster.

Além do modelo dimensional, foi criada uma área para armazenar os dados temporariamente, onde os dados serão tratados antes de ser carregados no Data Mart. Conhecida como “Data Staging Área”, este local foi de extrema importância para o caso da Ouvidoria, onde todos os registros foram validados previamente.

#### **4.4 DOCUMENTAÇÃO DO PROJETO**

A seguir, serão mostradas as fases desenvolvidas para o sucesso do projeto Data Mart da Ouvidoria da ANEEL.

As fases necessárias para o desenvolvimento do projeto foram:

- Planejamento
- Levantamento das necessidades
- Modelagem dimensional
- Projeto de extração, transformação e carga
- Desenvolvimento de aplicações “front end”

##### **4.4.1 Planejamento**

O escopo do projeto limita-se à Superintendência de Mediação Administrativa Setorial (SMA) onde se encontra o gerenciamento do Sistema de Ouvidoria da ANEEL. O Data Mart foi desenvolvido utilizando a abordagem evolutiva ou Bottom-up e poderá ser integrado futuramente com outros Data Marts dos demais departamentos, afim de formar um Data Warehouse corporativo da ANEEL, contendo todas as informações existentes na Instituição.

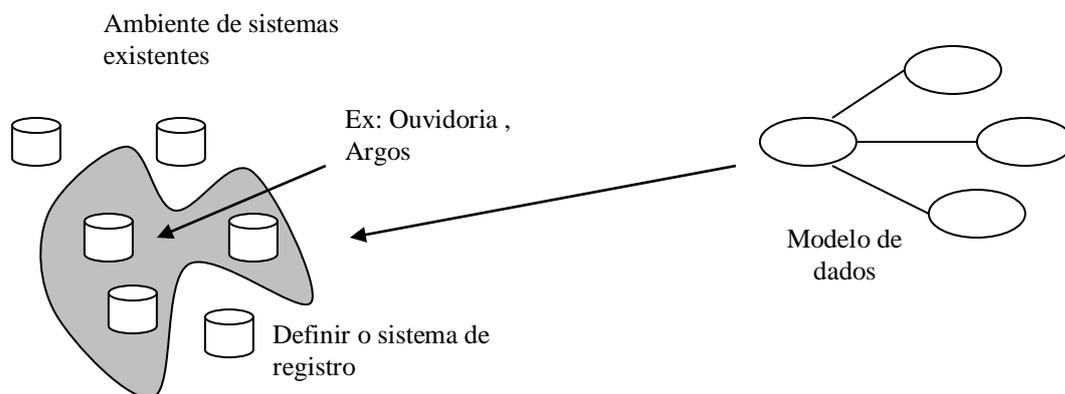
#### 4.4.2 Levantamento das Necessidades

Antes de qualquer elaboração do modelo dimensional destinado ao Data Warehouse, foi necessário efetuar um levantamento das necessidades existentes na Instituição em relação as informações gerenciais.

No caso da Ouvidoria, analisou-se a consistência das informações contidas na base transacional para avaliar o adicionamento ou não no modelo dimensional. Caso estes dados não tivessem consistência, seria necessário um tratamento destinado a correções, porém, em alguns casos isto não é possível. Como exemplo, se uma tabela na base de dados transacional não possui consistência, em hipótese alguma poderá gerar uma dimensão no modelo dimensional utilizado pelo Data Mart.

Para o sucesso do ambiente projetado deve-se analisar detalhadamente todos os dados que irão compor o modelo dimensional, para ter a certeza da integridade das informações extraídas e armazenadas no Data Mart.

Resumindo, a extração de dados da base operacional é um dos pontos chave no projeto de Data Mart para Ouvidoria. “O ponto de partida do plano de migração é o modelo de dados. O modelo de dados representa as necessidades de informações da empresa” [3]. A figura 4.2 simboliza a migração de dados para o ambiente projetado. Em tese, é possível construir um Data Mart sem um modelo de dados. Contudo, na prática, isso jamais é feito. Tentar construir um Data Mart sem um modelo de dados é como tentar navegar sem um mapa.



**Fig. 4.2 – Migração para o ambiente projetado para o DM da Ouvidoria**

É importante destacar dois atributos que foram criados após o desenvolvimento do modelo dimensional.

- Codigo\_Sup na tabela SOLICITACAO
- CodigoMunicipio na tabela ENVOLVIDO

Estes atributos foram criados no modelo relacional para atender as dimensões “SUPERINTENDENCIA” e “LOCAL” no modelo dimensional.

A seguir será mostrado o modelo fonte de dados utilizados pelo Data Mart da Ouvidoria. O modelo é baseado em quatro tabelas principais que possuem informações importantes do sistema de Ouvidoria.

#### DBO.SOLICITACAO

|                               |          |  |
|-------------------------------|----------|--|
| Numero_da_Solicitacao         | varchar  | Armazena o número da solicitação                     |
| Codigo_do_Concessionario      | varchar  | Armazena o código da concessionária                  |
| Codigo_do_Tipo_da_Unidade     | varchar  | Armazena a classificação da unidade consumidora      |
| Anonimo                       | bit      | Registra uma solicitação anônima                     |
| Sigilo                        | bit      | Registra uma solicitação com sigilo                  |
| Codigo_Sequencial             | int      | Armazena a natureza da reclamação                    |
| Numero_da_Unidade_Consumidora | varchar  | Armazena o número da unidade consumidora             |
| Registro_Novo                 | bit      | Registro novo  |
| Registro_Alterado             | bit      | Registro alterado                                    |
| Agencia_Responsavel           | varchar  | Armazena o código da agência responsável             |
| Data_Criacao                  | datetime | Armazena a data de criação                           |
| Data_Alteracao                | datetime | Armazena a data da alteração                         |
| Unidade_Responsavel           | char     | Armazena o número da unidade consumidora             |
| Protocolo_Concessionario      | varchar  | Armazena o número do processo na concessionária      |
| codigo_do_tipo_decisao        | int      | Armazena a procedência da solicitação                |
| Codigo_do_Tecnico             | int      | Armazena o código do técnico                         |
| Codigo_do_BO                  | int      | Armazena o código do BO                              |
| Codigo_da_situacao            | int      | Armazena o código da situação                        |
| Prazo                         | datetime | Armazena o prazo concedido a concessionária          |
| Primeiro_Contato              | datetime | Armazena a data do primeiro contato                  |
| Situacao_Concessionaria       | int      | Registra a abertura da solicitação na concessionária |
| Situacao_agencia              | int      | Registra a abertura da solicitação na agência        |
| Codigo_do_CTA                 | int      | Armazena o código da CTA origem                      |

|            |         |   |
|------------|---------|---|
| Codigo_Sup | varchar | Armazena o código da superintendência responsável |
|------------|---------|---|

#### DBO.ENVOLVIDO\_NA\_SOLICITACAO

|                       |          |                                  |
|-----------------------|----------|----------------------------------|
| Numero_da_Solicitacao | varchar  | Armazena o número da solicitação |
| Codigo_do_Envolvido   | int      | Armazena o código do envolvido   |
| Codigo_do_Tipo        | bit      | Armazena o tipo de envolvido     |
| Registro_Alterado     | bit      | Registra alteração               |
| Data_Criacao          | datetime | Armazena a data de criação       |
| Data_Alteracao        | datetime | Armazena a data de alteração     |

#### DBO.ENVOLVIDO

|                           |          |                                      |
|---------------------------|----------|--------------------------------------|
| Codigo_do_Envolvido       | int      | Armazena o código do envolvido       |
| Pais                      | int      | Armazena o País                      |
| UF                        | varchar  | Armazena o estado                    |
| Classe_de_Envolvido       | smallint | Armazena a classe do envolvido       |
| Codigo_do_Tratamento      | int      | Armazena o tratamento                |
| PJ                        | bit      | Armazena o CNPJ                      |
| Nome                      | varchar  | Armazena o nome do envolvido         |
| CPF                       | char     | Armazena o CPF                       |
| Orgao_Expedidor           | varchar  | Armazena o órgão                     |
| Empresa                   | varchar  | Armazena a empresa do envolvido      |
| Logradouro                | varchar  | Armazena a rua                       |
| Complemento               | varchar  | Armazena o complemento               |
| Bairro                    | varchar  | Armazena o bairro                    |
| Cidade                    | varchar  | Armazena a cidade                    |
| CEP                       | char     | Armazena o CEP                       |
| Telefone_Residencial      | varchar  | Armazena telefone                    |
| Telefone_Comercial        | varchar  | Armazena telefone                    |
| Telefone_Celular          | varchar  | Armazena telefone                    |
| Fax                       | varchar  | Armazena o número do fax             |
| E_mail                    | varchar  | Armazena o e-mail                    |
| Hora_de_Registro          | datetime | Armazena a hora de criação           |
| Forma_de_Contato_Desejada | int      | Registra a forma de contato desejada |
| Registro_Alterado         | bit      | Registra a alteração                 |

|                        |          |  |
|------------------------|----------|--|
| Data_Alteracao         | datetime | Armazena a data de alteração   |
| Data_Criacao           | datetime | Armazena a data de criação   |
| codigo_do_sexo         | char     | Armazena o código do sexo  |
| codigo_do_faixa_etaria | char     | Armazena a faixa etária  |
| codigo_da_escolaridade | char     | Armazena a escolaridade  |
| codigo_da_ocupacao     | char     | Armazena e ocupação  |
| CodigoMunicipio        | numeric  | Armazena o código do município – criado para facilitar a busca da dimensão local do DW |

DBO.HISTORICO

|                        |          |                                   |
|------------------------|----------|-----------------------------------|
| Data                   | datetime | Armazena a data da alteração      |
| Numero_da_Solicitacao  | varchar  | Armazena o numero da solicitação  |
| Codigo_do_Tecnico      | int      | Armazena o código do técnico      |
| Codigo_do_Encerramento | int      | Armazena o código do encerramento |
| Codigo_do_BO           | int      | Armazena o código do BO           |

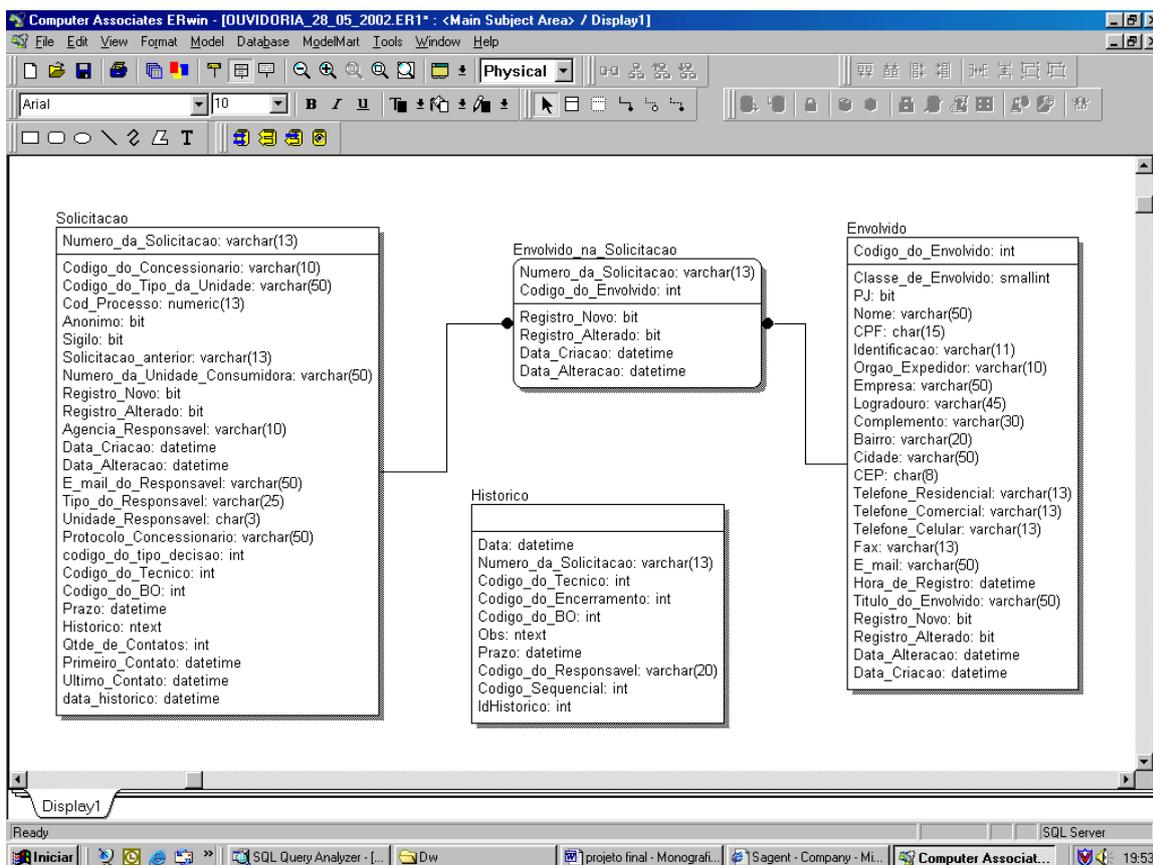
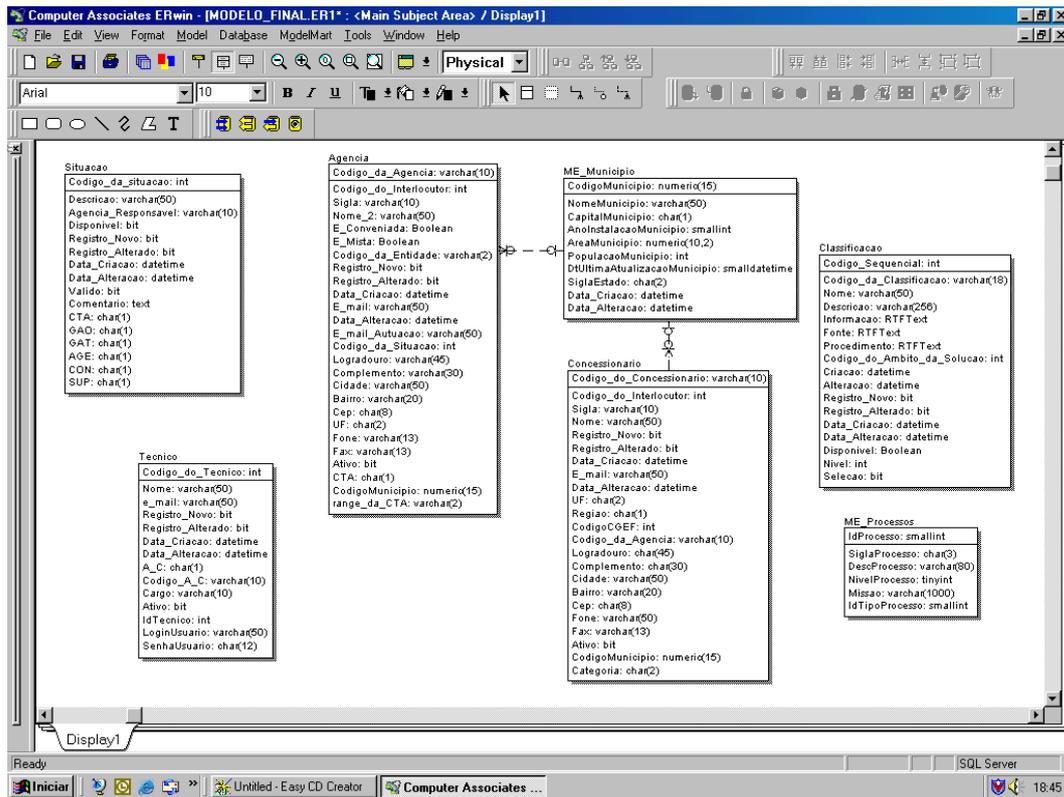


Fig. 4.3 – Tabelas que possuem informações sobre a Ouvidoria

### 4.4.3 Modelagem Dimensional

O modelo dimensional projetado, representa as principais informações existentes na base de dados da Ouvidoria voltadas para informações gerenciais.

A figura 4.4 representa as tabelas utilizadas para carregar as dimensões do modelo, nota-se que a dimensão tempo não consta na figura pois foi criada a partir da própria ferramenta.



**Fig. 4.4 – Tabelas retiradas do modelo relacional para gerar as dimensões**

Nota: As tabelas de fatos foram carregadas a partir das tabelas representadas na figura 4.3.

#### DIM\_SITUACAO

|                  |   |
|------------------|---|
| COD_SITUACAO     | Armazena o código da situação – representando o fluxo             |
| SI_DESCRICAO     | Armazena o nome da situação                                       |
| COD_SITUACAO_REL | Armazena o código da situação original do Modelo Relacional (MER) |

#### DIM\_LOCAL

|                 |  |
|-----------------|--|
| COD_LOCAL       | Armazena o código da localidade da solicitação |
| LC_REGIAO       | Armazena a região                              |
| LC_SIGLA_ESTADO | Armazena a sigla do estado                     |
| LC_ESTADO       | Armazena o nome do estado                      |
| LC_MUNICIPIO    | Armazena o nome do município                   |
| COD_LOCAL_REL   | Armazena o código da localidade do MER         |

#### DIM\_TECNICO

|                 |   |
|-----------------|---|
| COD_TECNICO     | Armazena o código do técnico responsável pela SO. |
| TE_NOME         | Armazena o nome do técnico                        |
| TE_SIGLA        | Armazena a sigla da agência do técnico            |
| COD_TECNICO_REL | Armazena o código do técnico original do MER      |

#### DIM\_CONCESSIONARIO

|                    |  |
|--------------------|--|
| COD_CONCESSIONARIO | Armazena o código do concessionário        |
| CO_SIGLA           | Armazena a sigla do concessionário         |
| CO_NOME            | Armazena o nome do concessionário          |
| CO_UF              | Armazena o estado do concessionário        |
| COD_CONCESSION_REL | Armazena o código do concessionário do MER |

#### DIM\_TEMPO

|            |  |
|------------|--|
| COD_TEMPO  | Armazena o código do tempo                           |
| ANO        | Armazena o ano referente                             |
| SEMESTRE   | Armazena o semestre referente                        |
| BIMESTRE   | Armazena o bimestre referente                        |
| MÊS        | Armazena o mês referente                             |
| MÊS_NOME   | Armazena o nome do mês referente                     |
| DIA_ANO    | Armazena o dia do ano                                |
| DIA_MÊS    | Armazena o dia do mês                                |
| DIA_SEMANA | Armazena o dia da semana                             |
| DATA       | Armazena a data com a formatação do banco relacional |

#### DIM\_CLASSIFICACAO

|                    |   |
|--------------------|---|
| COD_SEQUENCIAL     | Armazena o código da classificação                          |
| CL_NOME            | Armazena o nome da classificação                            |
| CL_NIVEL           | Registra o nível da classificação ex: info, recla, denuncia |
| COD_SEQUENCIAL_REL | Armazena o código da classificação do MER                   |

DIM\_SUPERINTENDENCIA

|             |   |
|-------------|---|
| COD_SUP     | Armazena o código da superintendência     |
| SU_SIGLA    | Armazena a sigla da superintendência      |
| SU_NOME     | Armazena o nome da Superintendência       |
| COD_SUP_REL | Armazena o código da sup. original do MER |

TAB\_FATO\_TOTALIZACOES

|                   |   |
|-------------------|---|
| COD_LOCAL         | Chave da dimensão LOCAL                           |
| COD_SEQUENCIAL    | Chave da dimensão CLASSIFICACAO                   |
| COD_TEMPO         | Chave da dimensão TEMPO                           |
| COD_AGENCIA       | Chave da dimensão AGENCIA                         |
| COD_CONCESIONARIO | Chave da dimensão CONCESSIONARIO                  |
| QT_SOLICITACAO    | Soma das solicitação da tabela SOLICITACAO do MER |

TAB\_FATO\_EVENTO

|                   |   |
|-------------------|---|
| COD_SITUACAO      | Chave da dimensão SITUACAO                      |
| COD_TEMPO         | Chave da dimensão TEMPO                         |
| COD_AGENCIA       | Chave da dimensão AGENCIA                       |
| COD_CONCESIONARIO | Chave da dimensão CONCESSIONARIO                |
| COD_SUP           | Chave da dimensão SUPERINTENDENCIA              |
| COD_TECNICO       | Chave da dimensão TECNICO                       |
| QT_SOLICITACAO    | Soma das solicitação da tabela HISTORICO do MER |



CO\_UF → retirada da do atributo UF da tabela CONCESSIONARIO

COD\_CONCESSIONARIO\_REL → retirada do atributo CODIGO\_CONCESSIONARIO da tabela CONCESSIONARIO. Servirá como base para as comparações para a carga da tabela de fatos.

#### **DIM\_AGENCIA**

COD\_AGENCIA → chave primaria da dimensão gerada pela ferramenta SAGENT

AG\_NOME → retirada do atributo nome da tabela AGENCIA

AG\_SIGLA → retirada do atributo sigla da tabela AGENCIA

COD\_AGENCIA\_REL → retirada do atributo Codigo\_Agencia da tabela AGENCIA. Servirá como base para as comparações para a carga da tabela de fatos.

#### **DIM\_CLASSIFICACAO**

COD\_SEQUENCIAL → chave primaria da dimensão gerada pela ferramenta SAGENT

CL\_CLASSE → retirada do atributo NOME da tabela CLASSIFICACAO

COD\_CLASSIFICACAO → retirada do atributo CODIGO\_CLASSIFICACAO da tabela CLASSIFICACAO

CL\_NIVEL → Feita utilizando derivação dos dados do campo COD\_CLASSIFICACAO, como já existia a classificação e o seu nível era obtido por uma substring, foi desenvolvida uma fórmula para que de acordo com o tamanho dessa string fosse gerado o seu nível. (Exemplo: COD\_CLASSIFICACAO = 010102 retornando o tamanho da substring = 6 então essa classificação é de Nível 3).

COD\_SEQUENCIAL\_REL → retirada do atributo CODIGO\_SEQUENCIAL da tabela CLASSIFICACAO. Servirá como base para as comparações para a carga da tabela de fatos.

#### **DIM\_LOCAL**

COD\_LOCAL → chave primária da dimensão gerada pela ferramenta SAGENT

LC\_SIGLA\_ESTADO → retirada do atributo NOME da tabela ME\_MUNICIPIO

LC\_MUNICIPIO → retirada do atributo MUNICIPIO da tabela ME\_MUNICIPIO

COD\_LOCAL\_REL → retirada do atributo CODIGO\_SEQUENCIAL da tabela ME\_MUNICIPIO. Servirá como base para as comparações para a carga da tabela de fatos.

Todas as dimensões foram carregadas a partir de tabelas existentes no modelo relacional. Para isso ser possível, foi criado um campo em cada dimensão contendo o código original do modelo relacional.

Depois de executados os planos para a carga das dimensões, houve a necessidade de armazenar dados em tabelas auxiliares (Data Staging Área) contendo os atributos necessários para a carga da tabela de fatos no qual foi realizando uma filtragem dos dados. Nesta fase se encontra a transformação propriamente dita, neste caso, a transformação mais importante foi a integração de datas.

Depois de carregados os dados nesta tabela auxiliar, foi executado um plano com diversas comparações entre as chaves relacionais e as chaves dimensionais, e para cada ocorrência de uma combinação entre local, tempo, classificação, concessionário e agência, seria referenciado então o código da dimensão àquela determinada ocorrência e assim seria possível realizar o quantitativo que é representado pelo campo QT\_SOLICITAES.

As comparações são feitas da seguinte maneira: se existe uma ocorrência do tipo Data (Relacional) = Data (Dimensional), então o atributo COD\_TEMPO referente àquela ocorrência segue o fluxo para a próxima comparação, senão o registro é descartado. Então são feitas as próximas comparações entre as demais dimensões até que se tenha na tabela fato todas as ocorrências daquela determinada ocorrência. Exemplo: Em São Paulo, no dia 25/10/02 o concessionário X, obteve duas reclamações de classificação 15.

#### 4.4.5 Desenvolvimento de aplicações

Após o sucesso e a validação da carga efetuada no modelo dimensional, o passo seguinte foi o desenvolvimento das consultas necessárias para atender às necessidades da Superintendência responsável pela Ouvidoria (SMA).

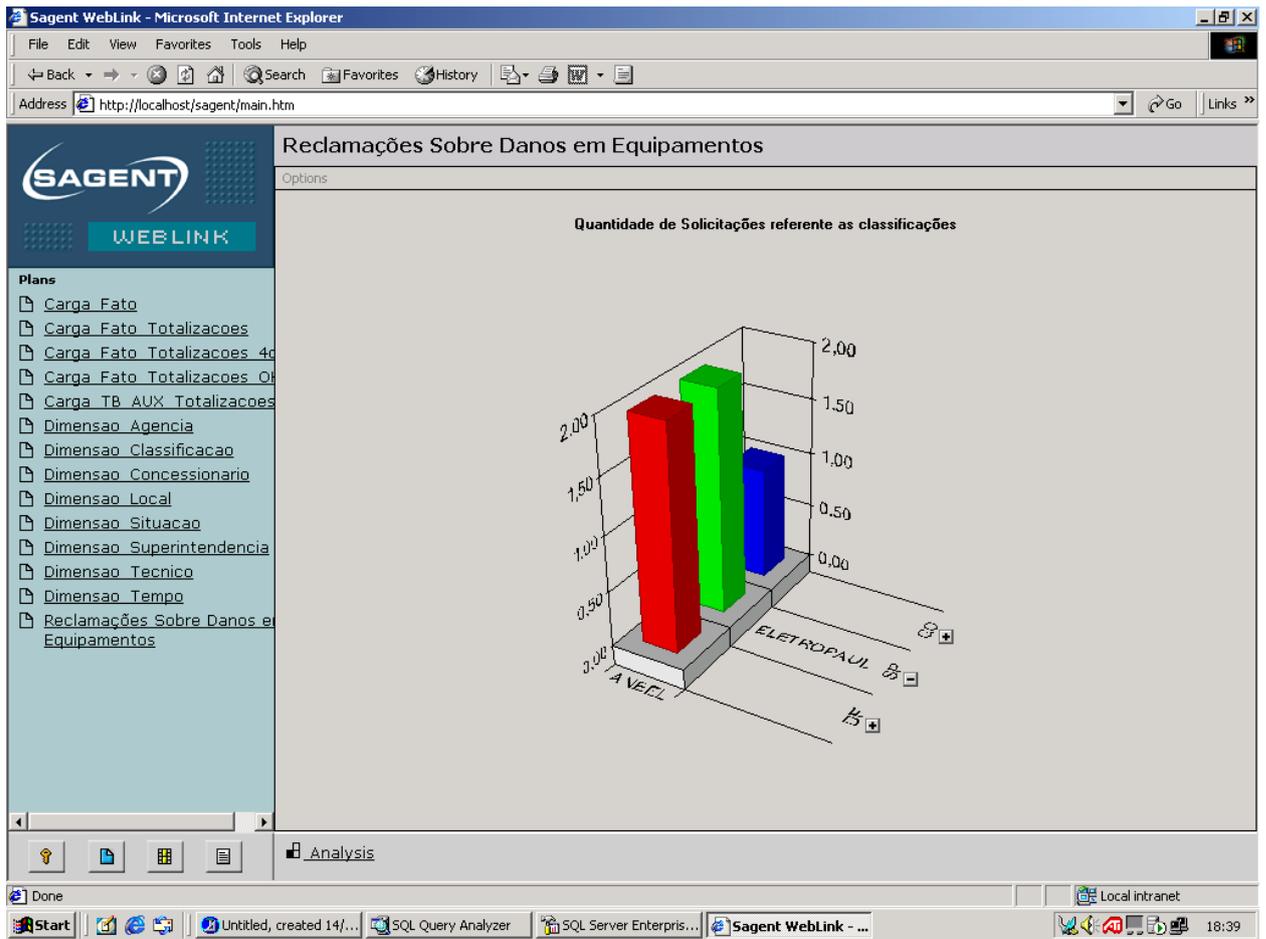
Utilizando novamente a ferramenta Sagent, agora como “front end”, foram desenvolvidas algumas consultas de acordo com as necessidades levantadas anteriormente.

O desenvolvimento de aplicações é uma das tarefas mais fáceis de todo o projeto, podendo até ser gerados e gerenciados pelos próprios usuários finais do sistema. Este benefício, é claro, não interfere nos aspectos relacionados ao gerenciamento de um Data Warehouse.

Apesar da simplicidade dos processos de elaboração de “front ends” é importante lembrar que um dos fatores-chave para o sucesso do gerenciamento do Data Warehouse ou de um Data Mart é manter uma equipe inicial pequena, porém, funcionalmente ágil [5].

A seguir, serão mostrados alguns relatórios, seguidos de visualizações gráficas. Este item é um dos mais importantes deste trabalho, pois aqui será apresentado o resultado de um projeto de Data Mart bem sucedido.

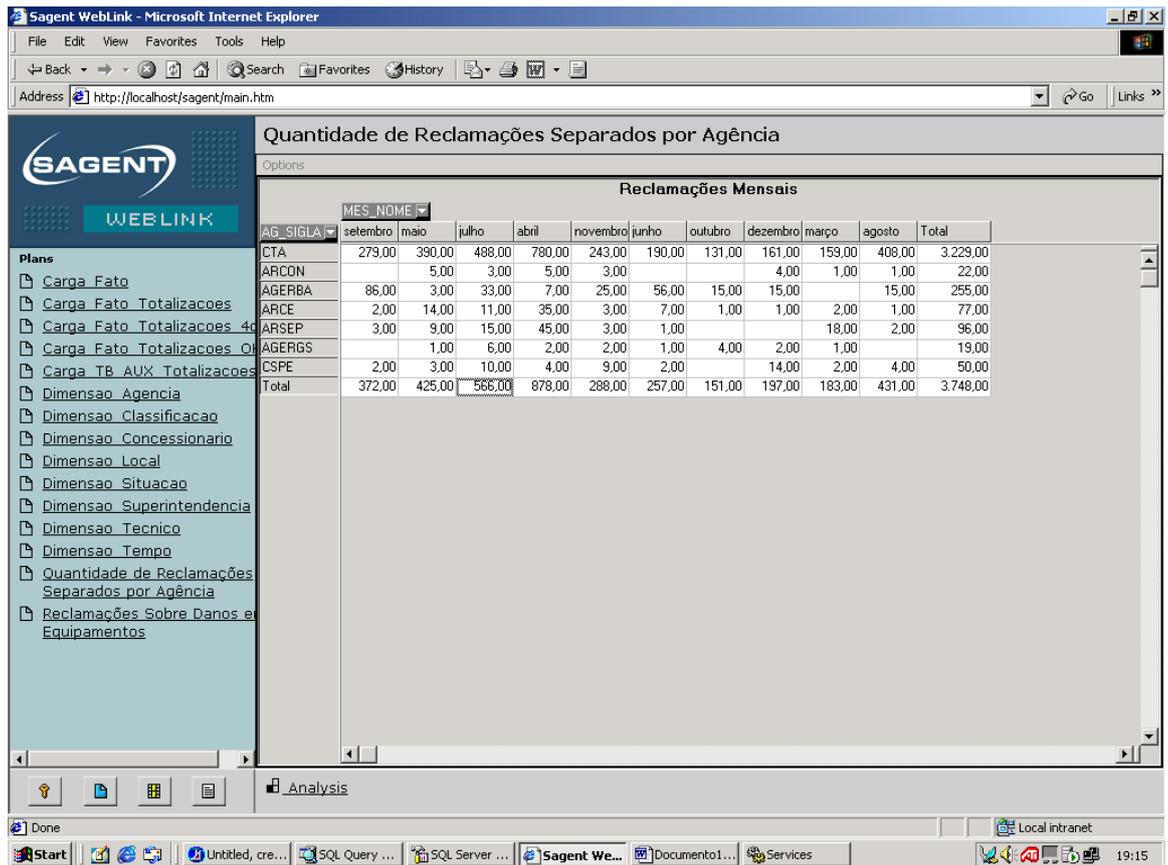
A figura 4.6 mostra um relatório contendo a quantidade de reclamações referentes a danos e ressarcimentos de um determinado concessionário. Onde pode ser visualizado o quantitativo de reclamações nos estados selecionados previamente.



**Fig. 4.6 – Front end 1**

A figura 4.7 mostra a quantidade de solicitações separadas por agências estaduais num determinado mês.

Este exemplo constitui uma planilha dinâmica onde poderão ser aplicadas as operações de “Slice” e “Dice” facilitando o entendimento da informação ou em formato gráfico como mostrado na figura 4.8.



**Fig. 4.7 – Front end 2**

É importante destacar que os relatórios e gráficos gerados já foram disponibilizados em ambiente web, através de um aplicativo da própria ferramenta Sagent, chamado “Web Link”, com isto, é possível disponibilizar na internet qualquer consulta gerada a partir de uma necessidade da Instituição.

Referindo-se à SMA, que é responsável pela Ouvidoria da ANEEL, esta tecnologia é muito válida, devido à descentralização de suas atividades nas Agências Estaduais. Como citado anteriormente, existem 13 Agências conveniadas, reforçando ainda mais a necessidade em disponibilizar informações pela internet, onde as próprias Agências podem efetuar consultas on-line no Data Mart da Ouvidoria da ANEEL.

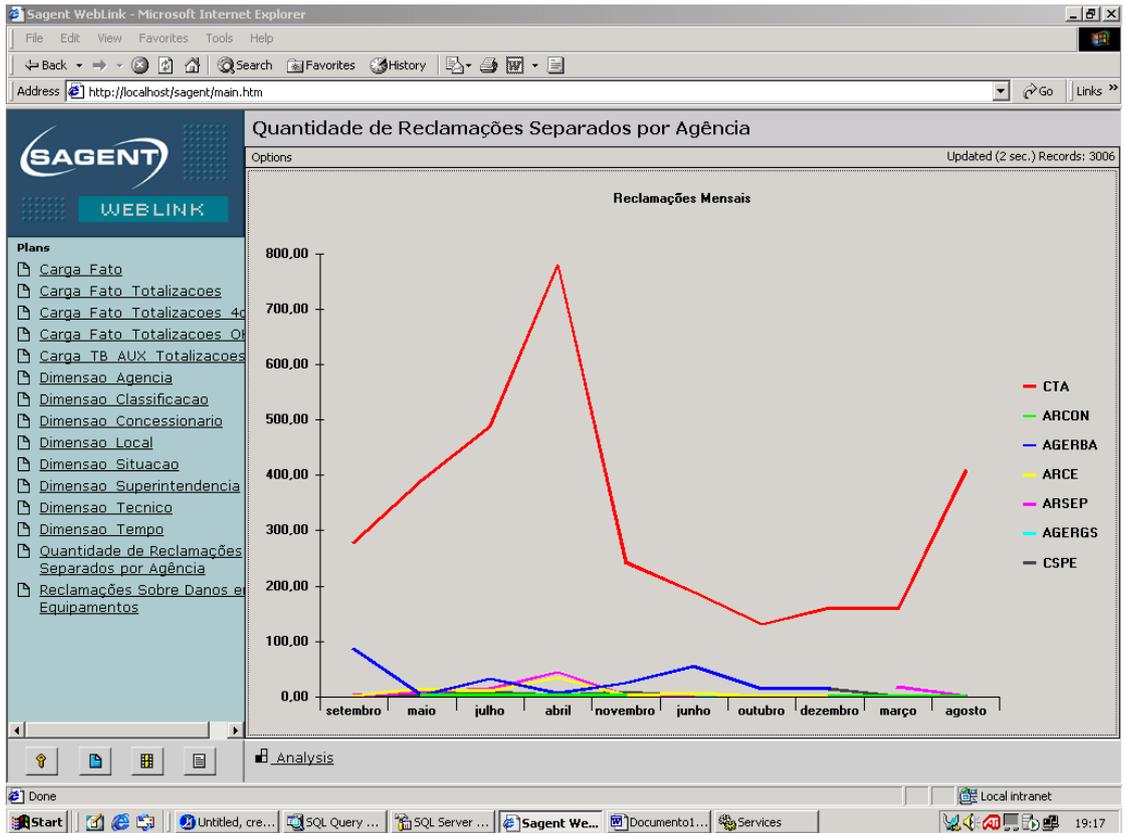


Fig. 4.8 – Front end 3

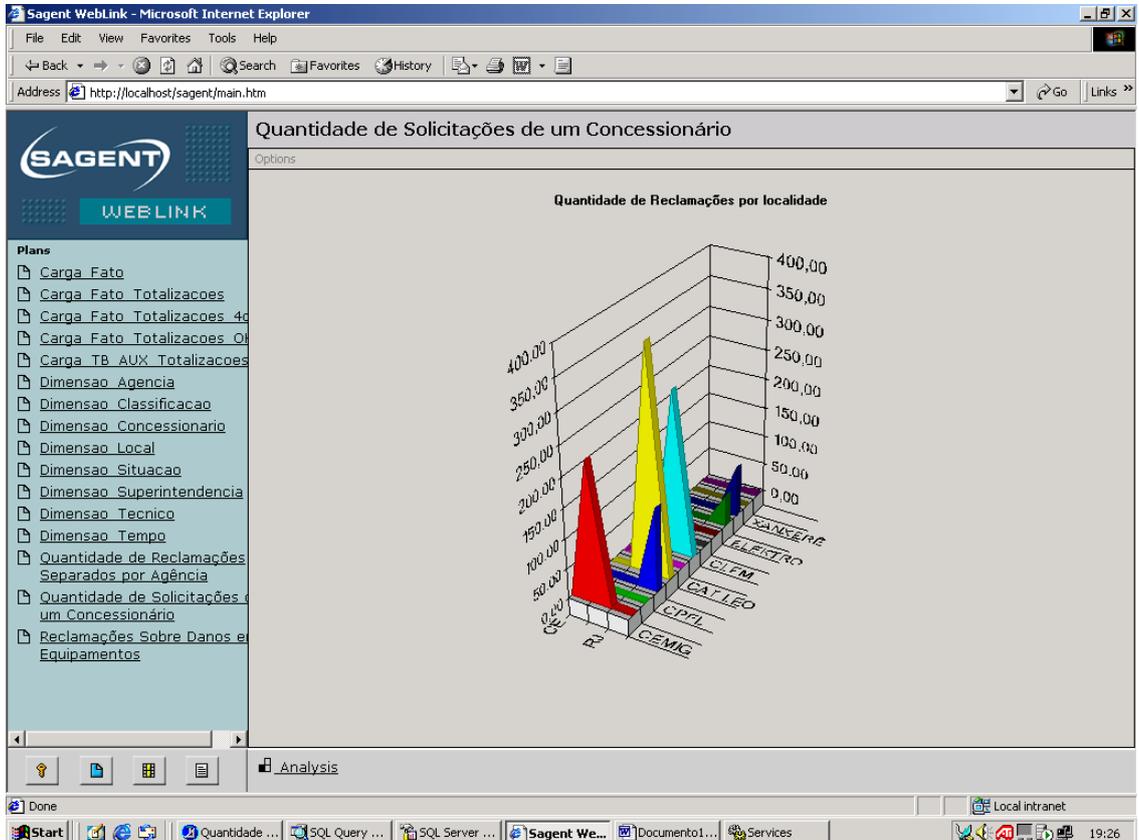


Fig. 4.9 – Front end 4

A figura 4.9 foi gerada apenas para mostrar os recursos gráficos disponíveis no Data Mart da Ouvidoria, onde é possível analisar informações sob várias perspectivas.

#### **4.5 EVOLUÇÃO DO DATA MART PARA ATENDER NECESSIDADES DE CRM**

Para enriquecer a utilidade do Data Mart da Ouvidoria, foi desenvolvida uma aplicação direcionada para analisar os consumidores e para esta avaliação ser possível, houve a necessidade de adicionar mais uma dimensão no modelo, chamada DIM\_ENVOLVIDO, como descrita a seguir.

##### **DIM\_ENVOLVIDO**

COD\_ENVOLVIDO → chave primária da dimensão

EN\_GRUPO → Relaciona o grupo do envolvido de acordo com a sua localidade, referindo-se a nível sócio-econômico e cultural.

EN\_NIVEL → Armazena o nível do envolvido diferenciado por governo, entidades, indústrias, etc

Nesta dimensão existe um atributo caracterizando grupos de consumidores de acordo com dados extraídos do Instituto Brasileiro de Geografia Estatística (IBGE), de acordo com as suas localidades.

Esta aplicação ainda não está ativa por não conter todos os campos devidamente preenchidos. Por exemplo, para se obter o nível do solicitante, será necessário que a central de teleatendimento registre este campo.

A vantagem de uma aplicação CRM eficiente está na agilidade das respostas aos consumidores. Na ANEEL ou em qualquer outra organização, o esforço é concentrado em atingir o consumidor com a resposta certa, no momento correto, oferecendo ao mesmo uma resposta satisfatória às suas demandas, de forma clara e coerente, mas, isto fica por conta do ambiente operacional. O Data Mart tem a função de acompanhar o gerenciamento do processo e medir os indicadores de sucesso.

Por exemplo, é possível saber quantas reclamações de solicitantes com nível de governo foram atendidas em um período de tempo.

Neste capítulo é oportuno dizer que o foco da Ouvidoria não é apenas avaliar ou influenciar consumidores e sim identificar problemas genéricos com os prestadores de serviços públicos do setor elétrico.

O principal interesse na aplicação CRM para a Ouvidoria é criar um modelo de comunicação estratégica com os resultados dos indicadores em questão, onde o maior desafio é enviar mensagens ou relatórios automaticamente para equipes internas da SMA. Na conclusão será apresentado um projeto que, futuramente, poderá ser desenvolvido, onde algoritmos inteligentes de busca fazem análise e coletam dados importantes.

No momento, para gerar um aplicativo “front end” deste processo, serão necessárias algumas alterações e atualizações nos conteúdos das tabelas relacionadas aos envolvidos. Por este motivo, neste momento, não será apresentada a solução final da aplicação CRM.

#### **4.6 CUSTOS DE IMPLEMENTAÇÃO DO PROJETO DATA MART**

Muitas organizações iniciam o desenvolvimento do Data Warehouse partindo com a solução departamental, Data Mart. No caso da Ouvidoria também foi adotado este conceito e as vantagens foram claras:

- Baixo Custo;
- Área de aplicação mais estreita, maior controle das informações;
- Tempo de desenvolvimento mais curto.

O tempo de desenvolvimento do projeto Data Mart foi de aproximadamente seis meses, sendo desconsiderado o tempo de levantamento dos processos do modelo relacional do Sistema de Ouvidoria, apresentado no anexo 2.

Os custos do desenvolvimento do DM da Ouvidoria serão apresentados na tabela abaixo:

| <b>Descrição</b> | <b>Valor aproximado em U\$</b> | <b>Período</b> |
|------------------|--------------------------------|----------------|
|------------------|--------------------------------|----------------|

|                  |                   |                     |
|------------------|-------------------|---------------------|
| SQL Server       | 40.000,00         | *                   |
| Sagent           | 250.000,00        | *                   |
| Hardware         | 100.000,00        | *                   |
| Equipe 5 pessoas | 120.000,00        | 6 meses - 960 horas |
| <b>Total:</b>    | <b>510.000,00</b> | <b>6 meses</b>      |

**Tab. 4.1 – Custos para implementação do Data Mart da Ouvidoria**

Historicamente, implementações de Data Warehouses Corporativos representam um grande esforço e exigem um tempo considerável. E, dependendo da alternativa de implementação escolhida, pode ter um impacto dramático no tempo necessário para se constatar um retorno do investimento. Apenas para fazer uma breve comparação, um Data Warehouse corporativo para ANEEL não ficaria pronto em menos de 18 meses.

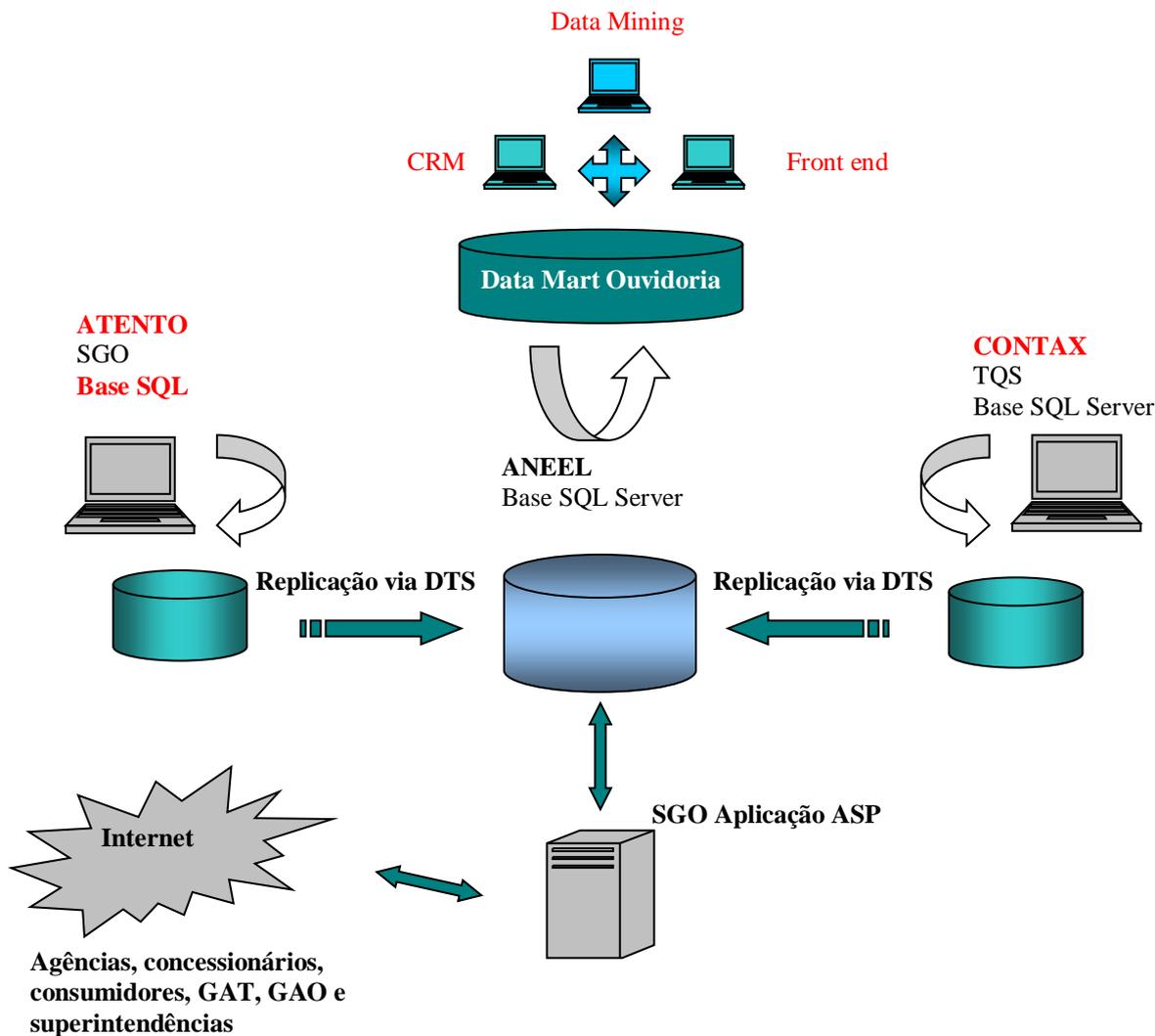
## Capítulo 5

### CONCLUSÃO

O conceito Data Warehouse se tornou mundialmente conhecido como a solução mais adequada para sistemas de apoio à decisão. Considerado um dos avanços mais relevantes em projetos de banco de dados, o Data Mart dimensional para a Ouvidoria aumentou significativamente a capacidade dos administradores em analisar rapidamente conjuntos de dados extensos e multidimensionais.

A base de dados operacional da Ouvidoria possui uma complexidade grande no que se refere a transações diárias, ou seja, muitos registros contidos na base de são modificados diariamente por agentes externos. Devido a esse fator, o objetivo deste trabalho foi aplicar uma solução que preservasse os históricos das modificações dos dados. Conforme simbolizada na figura 5.1 esta estrutura mostra algumas vantagens da aplicação:

- Extração de dados para o ambiente projetado, efetuando o armazenamento de informações históricas;
- Ambiente Read only, somente leitura;
- Apresentação de informações a partir do ambiente projetado, onde os dados não se alteram;
- Eliminação de consultas efetuadas na base transacional, gerando um aumento na performance do sistema SGO;
- Modelo dimensional adequado às necessidades das informações;
- Os dados armazenados no ambiente projetado, possuem qualidade e conteúdo para gerar novos tipos de consultas, ex: data mining.



**Fig. 5.1 Estrutura final após o sucesso do projeto**

Esta figura, já apresentada no capítulo 2, acrescida do Data Mart representa a aplicação desenvolvida para a Ouvidoria da ANEEL, notando-se que os agentes envolvidos continuaram os mesmos com exceção da base de dados da Atento, que foi migrada para SQL devido a necessidade de padronização nos processos de replicação, sendo este mais um fator que garante a confiabilidade nos registros migrados para a base da ANEEL.

Resumindo, este trabalho mostrou uma solução desenvolvida utilizando um Data Mart para resolver o problema relacionado à falta de integridade dos dados ocasionadas pelo grande volume de transações diárias na base operacional da Ouvidoria,

onde ficou claro que a solução nunca seria possível se não fosse a aplicação do conceito de Data Warehouse / Data Mart.

Em grande parte deste trabalho foi apresentada a ferramenta Sagent atuando como cerne de todo o processo, tanto na extração quanto na apresentação das informações em aplicações “front end”, entretanto, é válido destacar que o projeto conceitual de Data Mart para a Ouvidoria não depende de uma ferramenta específica, podendo-se considerar um produto genérico o qual pode ser adaptado em qualquer plataforma.

É importante lembrar que Data Warehouse não é um produto, mas uma estratégia que reconhece a necessidade de consolidar e armazenar separadamente dados em sistemas de informação dedicados a auxiliar os profissionais da Instituição a tomar decisões mais rápidas e com maior eficiência. Essa estratégia deve ter um objetivo: melhorar o desempenho da corporação.

## **5.1 DIFICULDADES ENCONTRADAS**

A principal dificuldade do projeto foi adequar o modelo e os dados contidos na base relacional para possibilitar o armazenamento no Data Mart.

Integrar registros é bem comum num projeto de Data Warehouse, porém, no caso da Ouvidoria foi necessário utilizar duas tabelas auxiliares para padronizar as datas numa formatação comum.

Os custos das ferramentas utilizadas num projeto de Data Warehouse são altos, hoje a ANEEL ainda não possui nenhuma ferramenta OLAP, conseqüentemente, para ser possível o desenvolvimento do projeto foi obtido com o fabricante da ferramenta Sagent uma versão Shareware. Para os testes efetuados em data mining, foi utilizado a ferramenta WizWhy demo, obtida gratuitamente por meio do site do fabricante.

## 5.2 PROJETOS FUTUROS

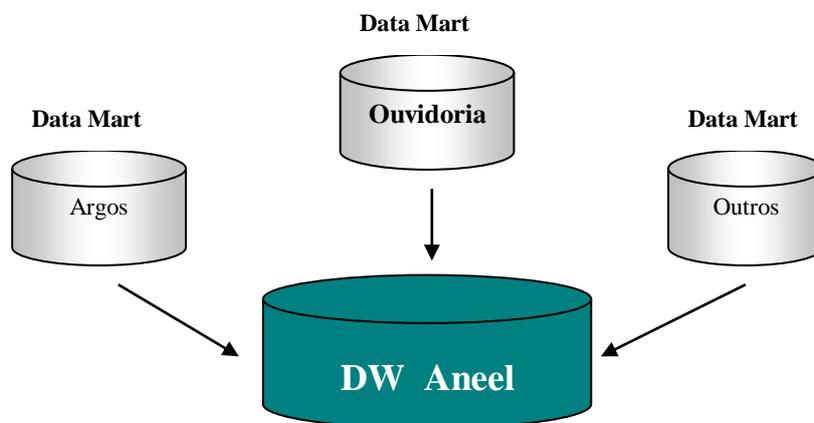
O Projeto futuro de maior interesse da ANEEL é o desenvolvimento do Data Warehouse corporativo, contendo todas as informações gerenciais da Instituição.

No projeto DM da Ouvidoria foi utilizada a abordagem evolutiva, “Bottom Up”, onde foi adotado um padrão de desenvolvimento adequado para todos os departamentos da ANEEL, tornando-se possível o desenvolvimento futuro de outros Data Marts que possam ser integrados ao Data Warehouse corporativo.

Possivelmente o próximo Data Mart a ser desenvolvido é do projeto Argos, que é um sistema de monitoramento de interrupção de energia elétrica. Estes equipamentos, instalados em diversas localidades, registram todas as interrupções do fornecimento e os tempos de duração, sendo os dados armazenados em uma base utilizada pela fiscalização.

Com a integração de um Data Mart do projeto Argos, pode-se facilmente cruzar informações importantes vindas da Ouvidoria com informações contidas na base de dados do Argos. Por exemplo, existe na Ouvidoria um quantitativo de reclamações referentes à interrupção do fornecimento de uma determinada localidade, então, futuramente poderá ser efetuada uma checagem dos registros de reclamações daquela localidade cruzando informações vindas da base Argos.

A figura 5.2 simboliza a evolução de um Data Warehouse para a ANEEL.



**Fig. 5.2 Data Warehouse Global**

Outro projeto de grande interesse da ANEEL está relacionado à extração de informações contidas no Data Warehouse através da tecnologia Data Mining. Ainda pouco utilizadas no Brasil, as ferramentas de Mineração de Dados (*Data Mining*) são consideradas a terceira geração de Sistemas de Suporte à Decisão (SSD). Elas possibilitam o cruzamento de informações na base de dados, com respostas impensadas pelo usuário sem o uso desta tecnologia.

As aplicações OLAP, vistas anteriormente neste trabalho, permitem que o usuário final possa extrair informações e conhecimento que são úteis para sua organização visualizando e manipulando de forma multidimensional os dados do DW.

Por outro lado, as ferramentas de mineração de dados são especializadas em procurar padrões nos conjuntos de dados. Esta busca pode ser efetuada automaticamente pelo sistema ou interativamente com um analista responsável pela geração de hipóteses. Diversas ferramentas distintas, como redes neurais, indução de árvores de decisão, sistemas baseados em regras e programas estatísticos, tanto isoladamente quanto em combinação, podem ser aplicados ao problema. Ao final do processo, o sistema de Mineração de Dados gera um relatório das descobertas, que passa então a ser interpretado pelos interessados.

Alguns testes com esta tecnologia já foram iniciados no Data Mart da Ouvidoria, onde através da ferramenta WizWhy [31] e utilizando lógica formal, foram gerados alguns algoritmos de busca de dados.

## Referências

- [1] GRAEML, Alexandre R.. Sistemas de Informação: o alinhamento da estratégia de TI com a estratégia corporativa. São Paulo, Editora Atlas S.A., 2000.
- [2] GRAY, Paul & Watson, Hugh J. **Decision Support in the Data Warehouse**. New Jersey, Prentice Hall PTR, 1998.
- [3] INMON, W.H. – Como Construir o Data Warehouse, Campus, Rio de Janeiro, 1997.
- [4] INMON, W.H. & HACKATHORN, Richard D. – Como Usar o Data Warehouse, Infobook, Rio de Janeiro, 1997.
- [5] INMON, W.H. – Gerenciando Data Warehouse. Rio de Janeiro, Makron Books, 1999.
- [6] INMON, W.H – Information System Architecture, John Wiley & Sons Inc, 1993
- [7] KIMBALL, Ralph & REEVES, Laura & ROSS, Margy & THORNTHWAITE, Warren - The Data Warehouse Lifecycle Toolkit, John Wiley & Sons, New York, 1998.
- [8] KIMBALL, Ralph – **The Data Warehouse Toolkit**. São Paulo: Makron Books, 1998. 388p.
- [9] HARRISON, T.H. – *Intranet Data Warehouse*, Berkeley, 1998.
- [10] GUPTA, C. R.. An Introduction to Data Warehouse, 1997, in <http://system.services.com/dwintro.htm>
- [11] MATTISON, Rob. Data Warehousing: strategies, technologies and techniques. New York, McGraw – Hill, 1996, 485p.

- [12] - Walton and Cline in Data Warehouses/Data Marts: Repositories for Data Mining
- [13] - BARQUINI, Ramon. **Planning and Designing the Warehouse**. New Jersey: Prentice-Hall, 1996. 311p.
- [14] ONEIL, B. – Oracle Data Warehousing. Indianapolis, Sams Publishing, 1997.
- [15] CAMPOS, Maria Luiza & FILHO, Arnaldo V. Rocha. **Data Warehouse**. Disponível na Internet via <http://genesis.nce.ufrj.br/dataaware/tutorial/indice.html>. Arquivo consultado em 20/04/2002.
- [16] DALALBA, Adriano. **Um Estudo Sobre Data Warehouse**. Caxias do Sul: UCS,1998.97p. Monografia.
- [17] GIOVINAZZO, A. William. **Object-Oriented Data Warehouse Design Bulding a Star Schema**. New Jersey: Prentice Hall PRT, 2000, 250p.
- [18] IBM. **Data Modeling Techniques for Data Warehousing**. Disponível na Internet via [www.readbooks.com](http://www.readbooks.com). Arquivo acessado em 2002.
- [19] IBM. **Data Modeling for a Data Warehouse**. Disponível na Internet via [www.redbooks.ibm.com/pubs/pafs/readbooks/sg242238.pdf](http://www.redbooks.ibm.com/pubs/pafs/readbooks/sg242238.pdf). Arquivo acesso em 2002.
- [20] INMON, William H. **Data Mart Does Not Equal Data Warehouse**. Disponível na Internet via [www.billinmon.com](http://www.billinmon.com). Arquivo consultado em 2002.
- [21] INMON, William H. **Enhancing Data Warehouse Performance**. Disponível na Internet via [www.billinmon.com](http://www.billinmon.com). Arquivo consultado em 2002.
- [22] MACHADO, Felipe Nery Rodrigues. **Projeto de Data Warehouse – Uma visão multidimensional**. São Paulo. Érica, 2000, 248p.

- [23] ORACLE. **Data Mart**. Disponível na Internet via <http://www.oracle.com.br>.  
Arquivo acesso em 2002.
- [24] SEIDLER, C. Dez mandamentos. **ComputerWorld**, Rio de Janeiro, v.20,n38,  
p.18-23, jan/1995.
- [25] BARBIERI, Carlos. **BI – Business Intelligence – Modelagem & Tecnologia**, Rio de Janeiro, Axcel Books, 2001.
- [26] COLANGELO, Lucio F.. **Implantação dos Sistemas ERP**. São Paulo, Editora Atlas, 2001.
- [27] STAIR, Ralph M.. **Princípios de Sistemas de Informação – Uma Abordagem Gerencial**. Rio de Janeiro, Editora LTC, 1998 2ª Edição.
- [28] TURBAN, Efraim & AROSON, Jay E.. **Decision Support Systems and Intelligent Systems**. New Jersey, Uppler Saddle River, 1998 5ª Edition.
- [29] PATTON, ROBERT. – **Projetando e Administrando Banco de Dados SQL Server**. Rio de Janeiro, Editora Altabooks, 2002.
- [30] SWIFT, RONALD. – **CRM (Customer Relationship Management)**,
- [31] **WizWhy** – Disponível em <http://www.wizsoft.com>
- [32] Coelho, Regina – **Proposta para o Desenvolvimento da Comunicação da ANEEL**, Brasília, 2001.

Anexos poderão ser obtidos em <http://www.dw-work.com.br>.



**AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA**

# **Concepção de um Data Warehouse e CRM para o Sistema de Ouvidoria da ANEEL**

**Aluno: André Ruelli**

**Orientador: Prof. Germano Lambert Torres**

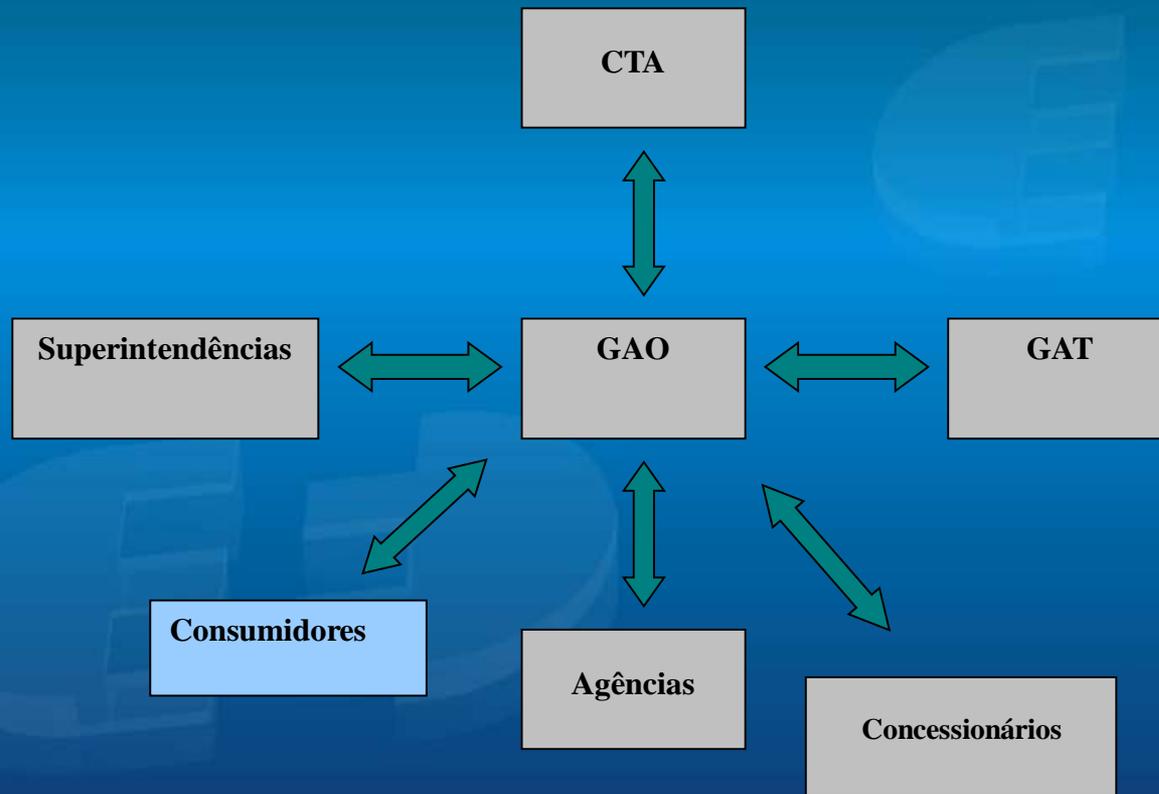
**Co-orientador: Prof. Manoel Eduardo Miranda Negrisoni**

# Ouvidoria da ANEEL

- ✓ Superintendência de Mediação Administrativa Setorial - SMA
- ✓ Responsável pelo atendimento das reclamações de agentes e consumidores
  
- ✓ CTA's
- ✓ Grupos internos na ANEEL
- ✓ Agências Estaduais

# Sistema de Gestão da Ouvidoria - SGO

✓ Estabelece o elo entre os agentes envolvidos no processo

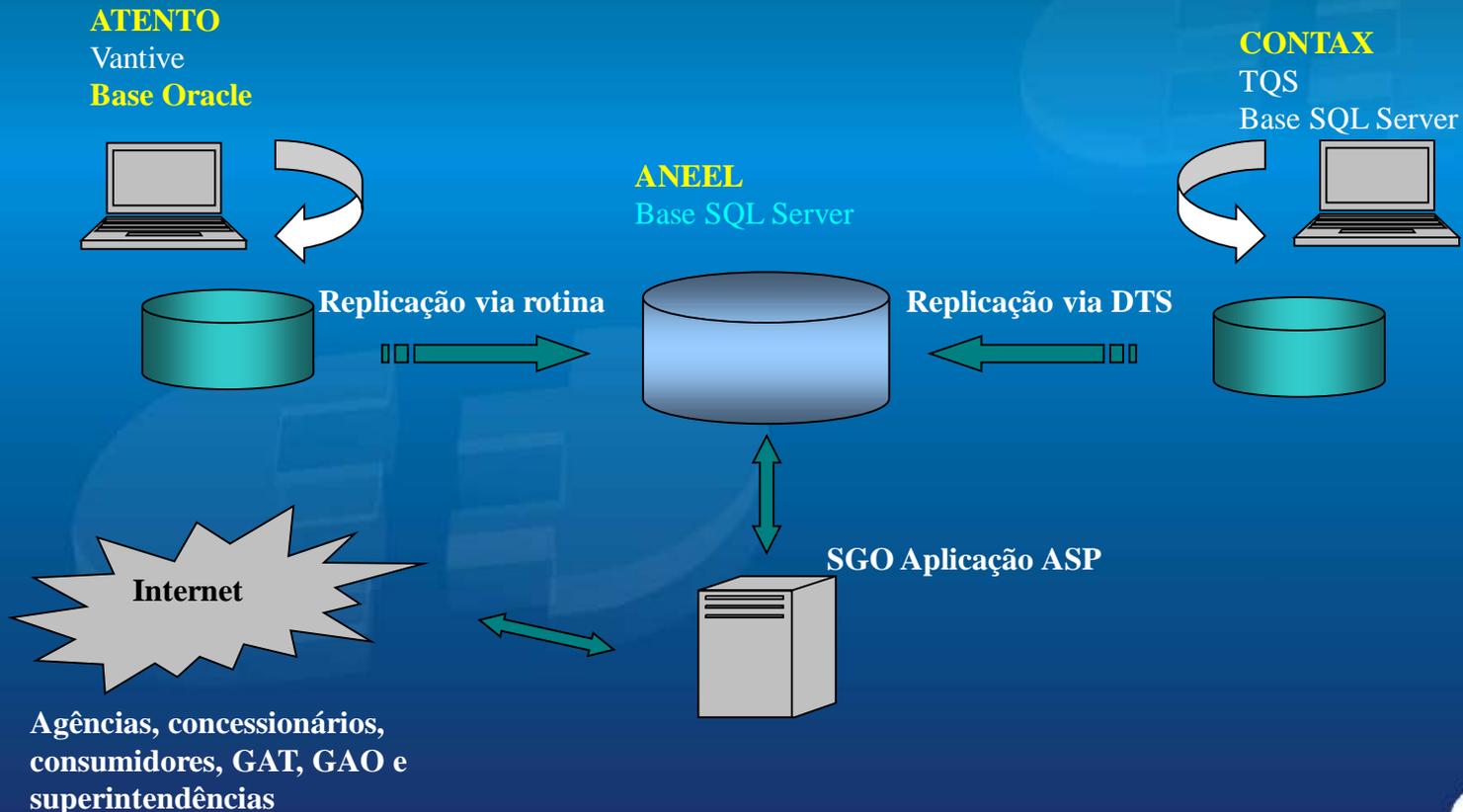


# Base de dados da Ouvidoria

- ✓ Volume de dados existentes na base, cerca de  $3.000 \times 10^3$  solicitações de consumidores
- ✓ Inclusão de 7.000 solicitações por dia através das CTA's
- ✓ 3.000 transações por dia, através do SGO

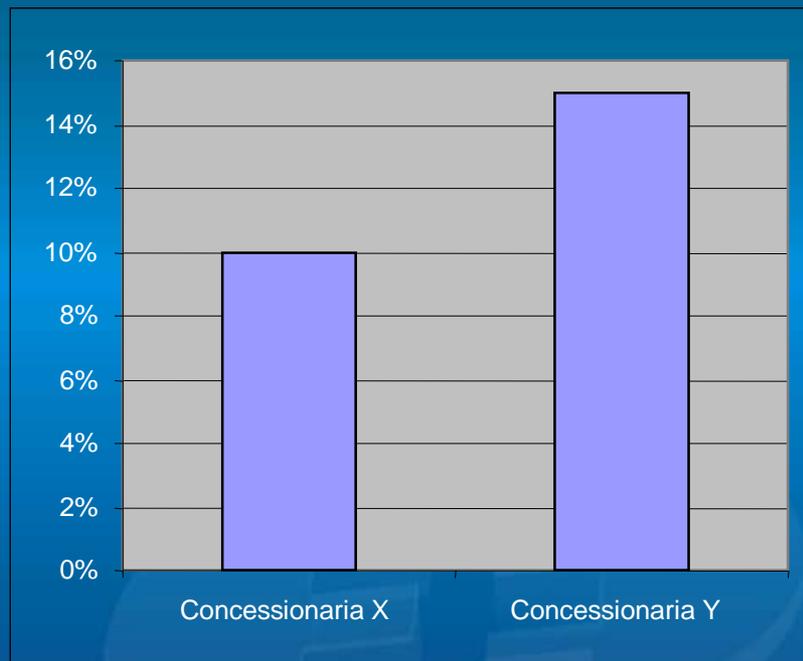
# Pontos Críticos

- ✓ Volatilidade
- ✓ Sobreposições
- ✓ Complexidade dos processos de replicação



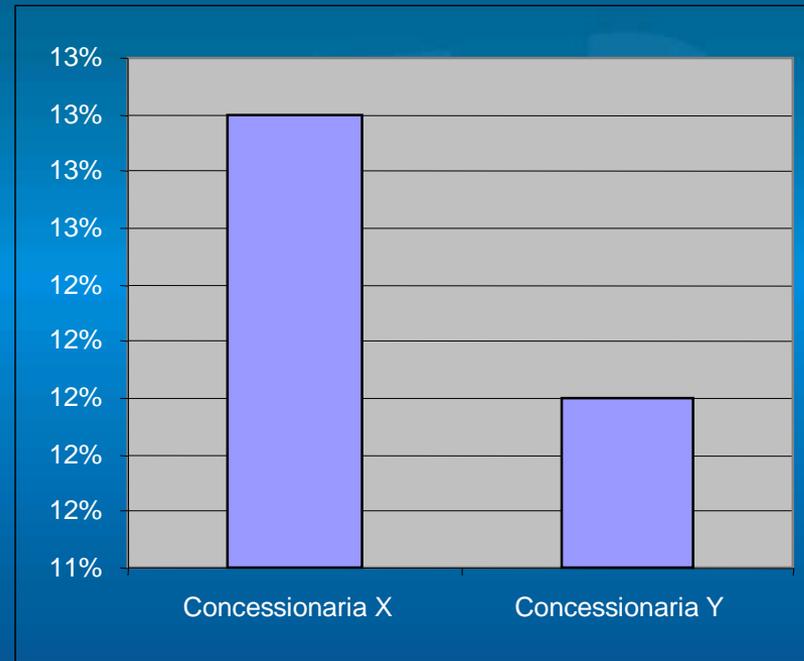
# Falta de Integridade nas Informações

Percentual de reclamações em 2001



Consulta realizada em 10/05/2002

Percentual de reclamações em 2001



Consulta realizada em 20/06/2002

# Necessidade da Ouvidoria

- ✓ Sistema de Apoio à Decisão – SAD
- ✓ Integridade das informações
- ✓ CRM
- ✓ Cruzar informações de outras áreas

*Solução: Data Warehouse*

# Evolução dos Sistemas de Informação

## ✓ Histórico

- 1960, Arquivos mestres;
- 1970, Direct Access Storage Device (DASD);
- 1975, Sistema Gerenciador de Banco de Dados (SGBD)

## ✓ *On Line Transaction Processing (OLTP)*

- “A finalidade do processamento de transações on-line é processar grandes quantidades de transações minúsculas e atômicas, sem perder nenhuma delas” [KIM98b]

# Evolução dos Sistemas de Informação

- ✓ Problemas com a Arquitetura de Desenvolvimento espontâneo
  - Falta de credibilidade
  - Problemas de produtividade
  - Impossibilidade de transformar dados em informações
- ✓ Ambiente Projetado

# Ambiente de Data Warehouse

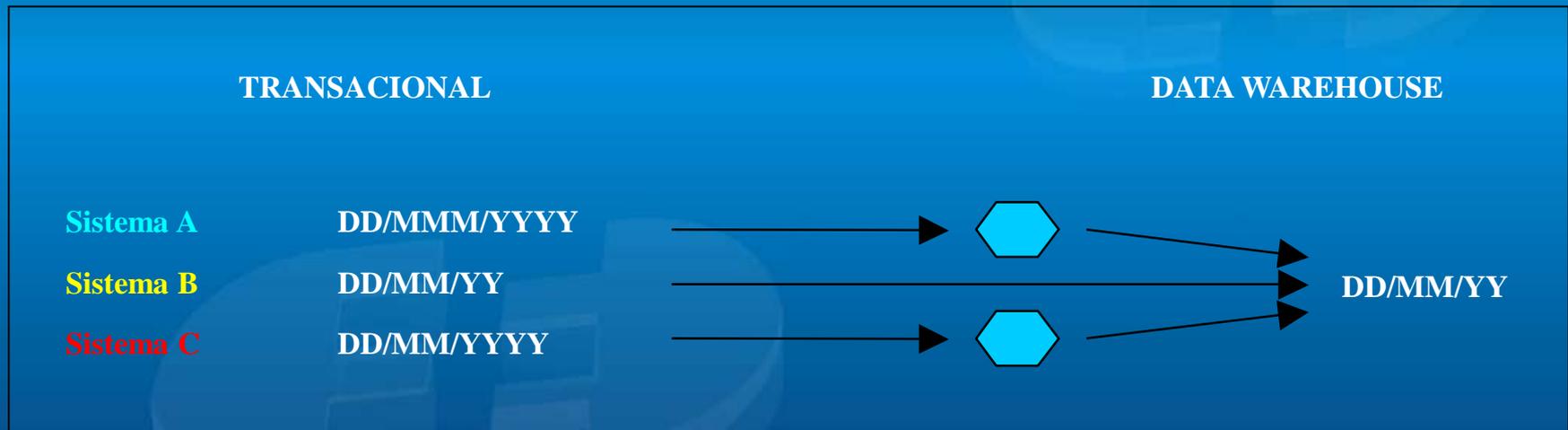
## ✓ Conceitos

- “Data Warehouse é uma coleção de dados orientados por assunto, integrados, variáveis com o tempo e não voláteis, para dar suporte ao processo de tomada de decisão, onde cada unidade dos dados é relevante em determinado momento do tempo”
- “É uma cópia de dados transacionais especificamente estruturados para consulta e análise”

# Ambiente de Data Warehouse

## ✓ Características

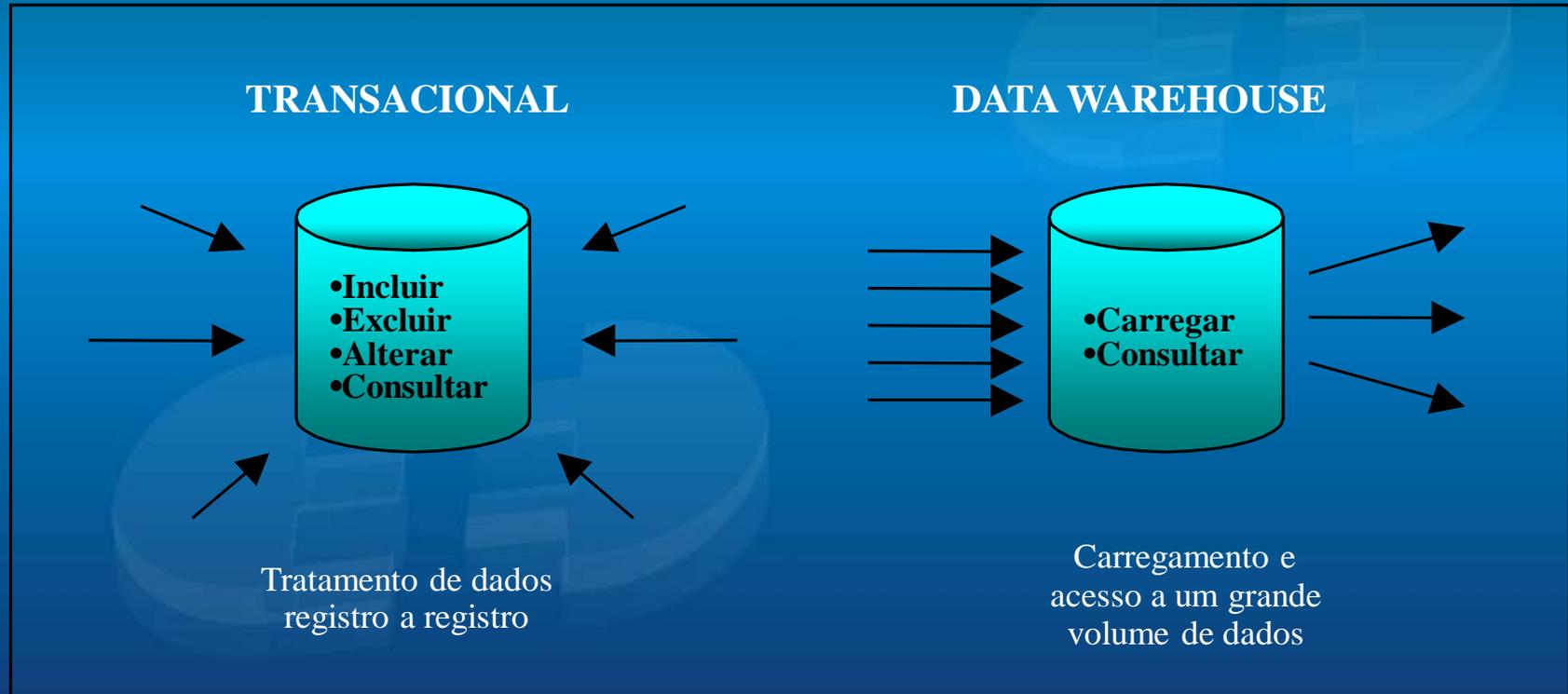
–Integrado



# Ambiente de Data Warehouse

## ✓ Características

– Não Volátil



# Ambiente de Data Warehouse

## ✓ Características

### – Variante no Tempo

- Grande horizonte temporal – os dados estão representados ao longo de um grande horizonte temporal, de 5 a 10 anos
- Estrutura de chave – Toda chave no Data Warehouse contém de forma implícita ou explícita um elemento de data, como dia, semana, mês, etc

# Ambiente de Data Warehouse

## ✓ Características

### – Metadados

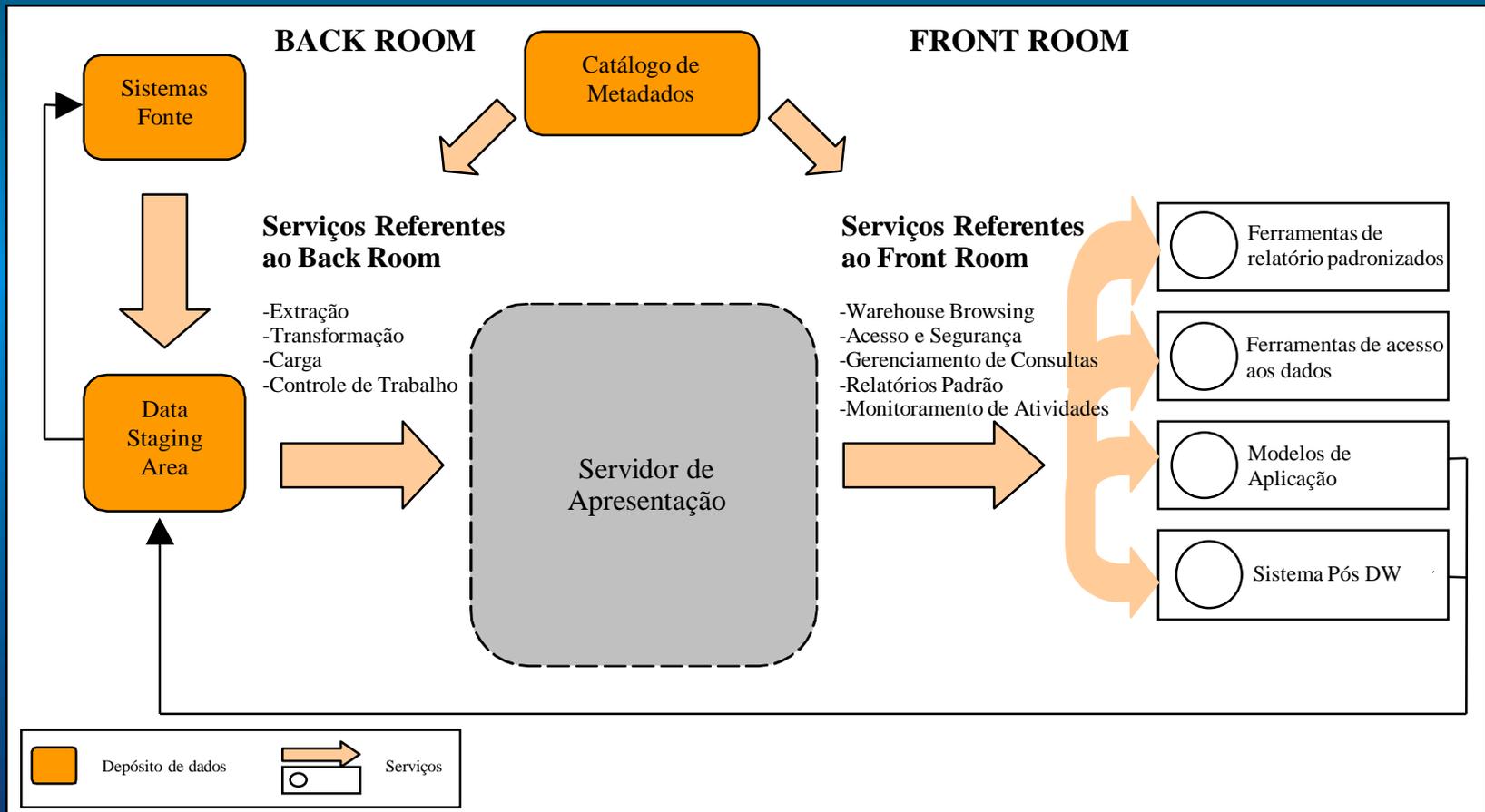
- São definidos como “dados dos dados”, “informações das informações”
- Vital importância na documentação dos sistemas e dos bancos de dados
- Segundo Inmon, “os metadados englobam o DW e mantêm informações sobre o que está aonde”

# Arquitetura do DW

- **Processo de construção de um Data Warehouse**
  - Coletar os dados dos sistemas fonte
  - Organizar e transportar esses dados para um outro repositório
  - Apresentar os dados de forma clara e inteligível para o usuário final escondendo toda a complexidade que envolve esse processo

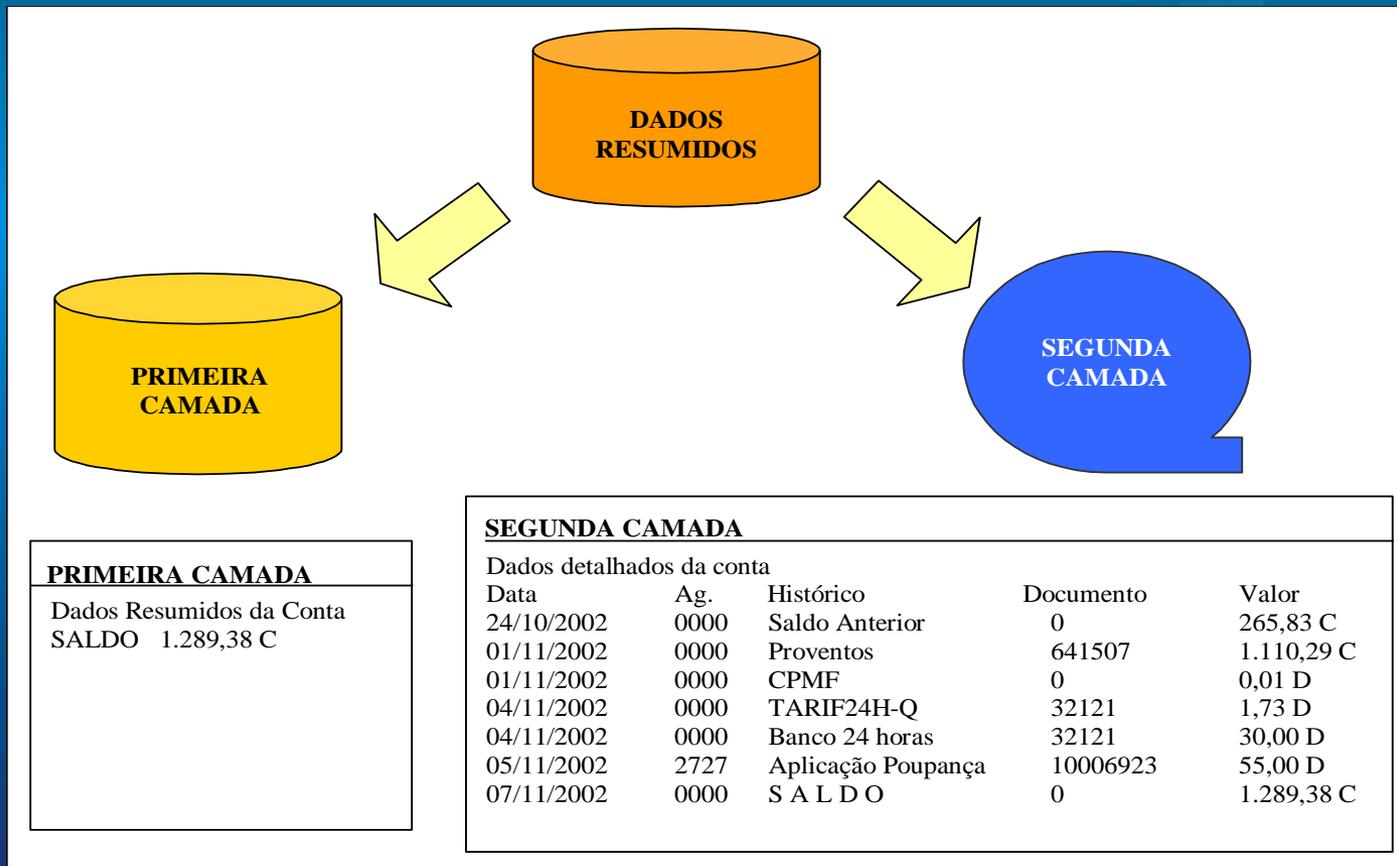
# Arquitetura do DW

- Organização da arquitetura segundo Kimball



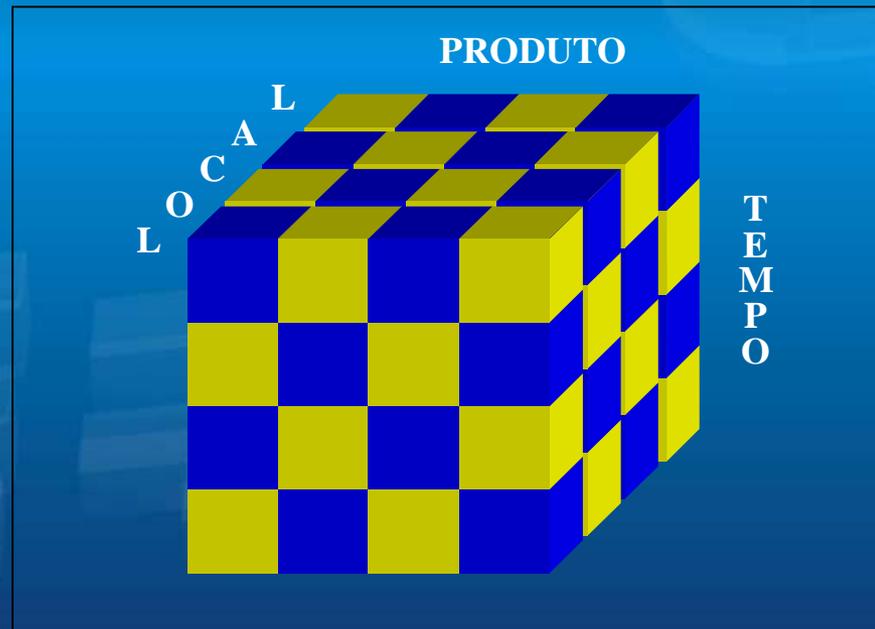
# Granularidade

- ↑ Granularidade ↓ Nível de Detalhe, ↓ Granularidade ↑ Nível de Detalhe
- Níveis Duais de Granularidade



# Modelagem de Dados no DW

- ✓ Visão dos dados de várias perspectivas
- ✓ Dados que se encaixam perfeitamente em uma espécie de cubos de dados

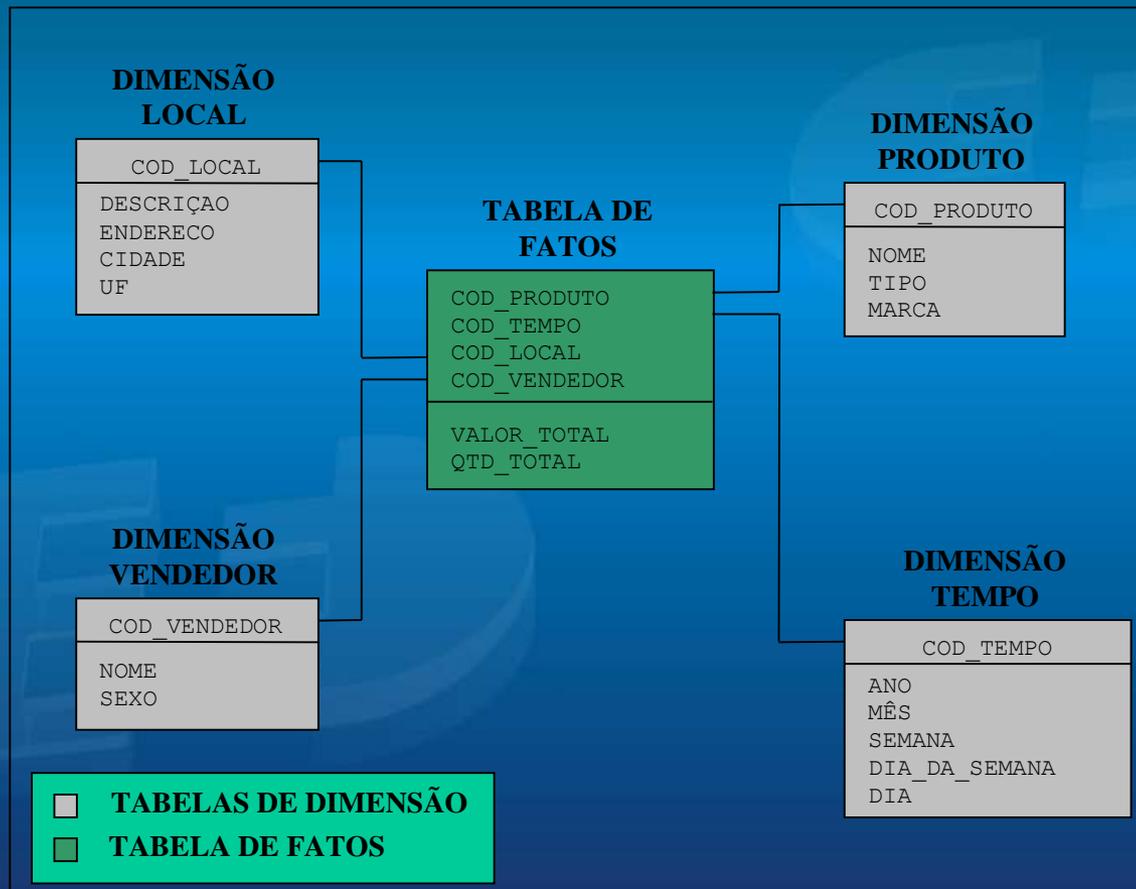


# Modelagem de Dados no DW

- Modelo Estrela (*Star Join Schema*)
  - Tabela de Fatos
    - Dados numéricos, continuamente valorados e aditivos
    - compactação de registros
  - Tabela de Dimensão
    - Melhores atributos são textuais e discretos
    - Utilizado como cabeçalhos de linha e para definir restrições.

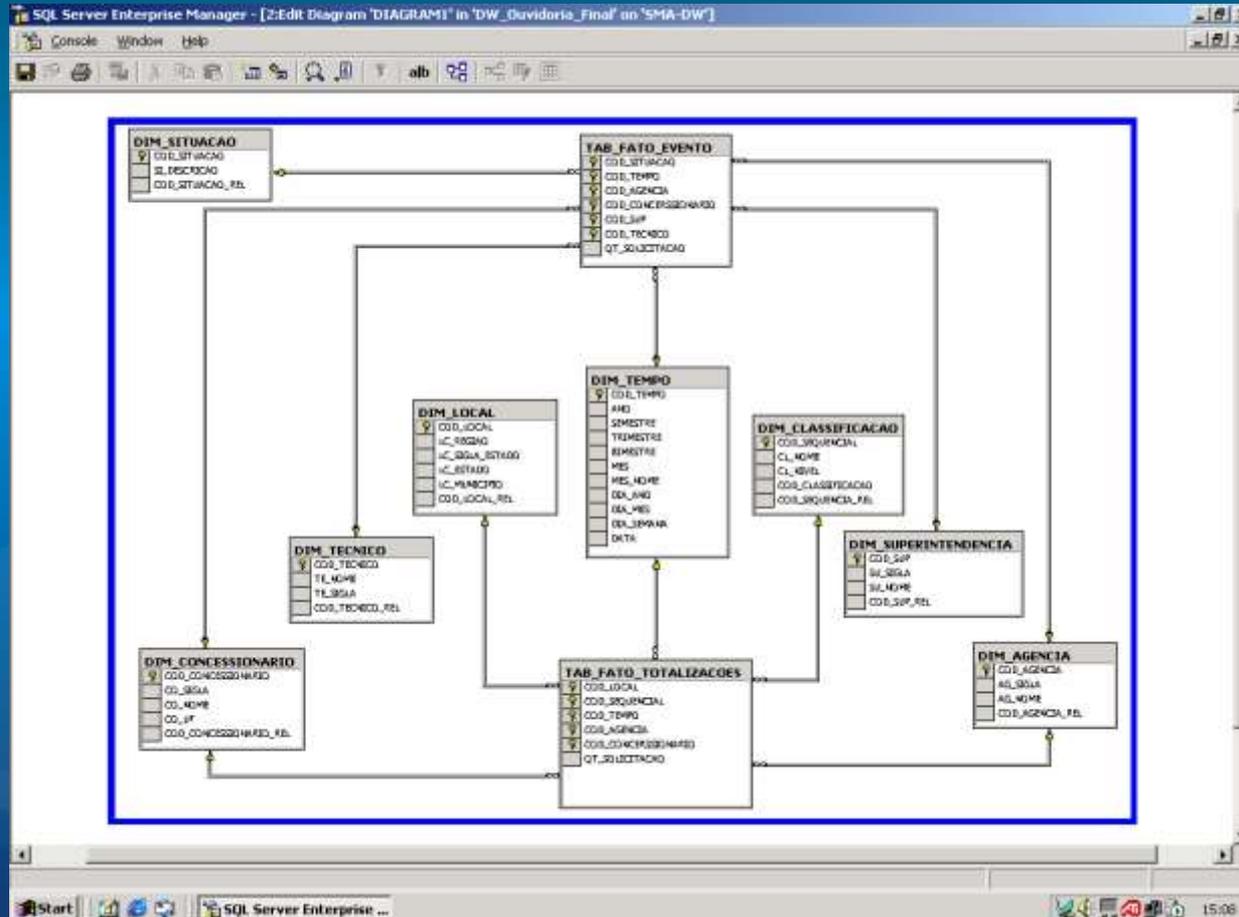
# Modelagem de Dados no DW

- Modelo Estrela (*Star Join Schema*)



# Modelagem de Dados no DW

- Modelo Estrela da Ouvidoria



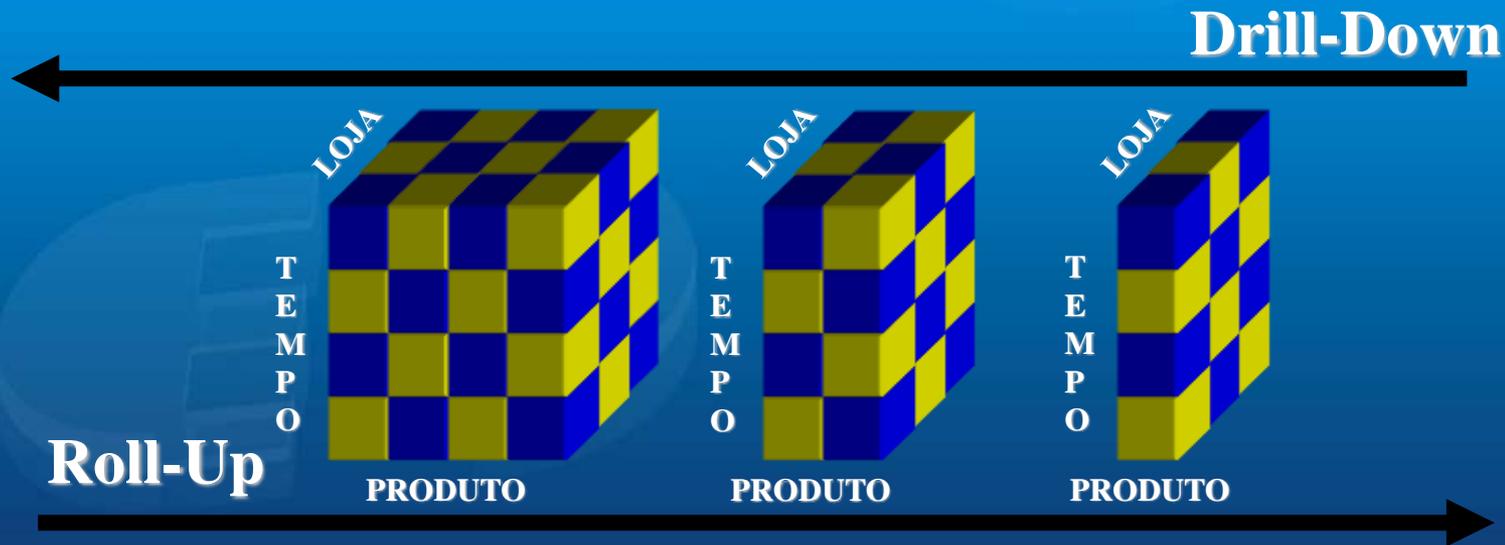
# Extração de Informações do DW

- **OLAP**

- Descreve uma abordagem dimensional para o suporte à decisão
- Processamento Analítico
- Constitui-se de todas as atividades de consulta e apresentação de dados numéricos e descritivos vindos do DW

# Extração de Informações do DW

- Operações básicas em OLAP
  - Drill-Down e Roll-Up
    - Capacidade de subir ou descer ao longo das hierarquias do cubo na direção de maior detalhamento(Drill-Down) ou menor (Roll-Up)



# Extração de Informações do DW

- **Arquiteturas OLAP**

- ROLAP (Relational OLAP)

- Conjunto de interfaces e aplicações que dão ao SGBD relacional características multidimensionais
    - Utiliza extensões de SQL para tratar as operações e dados multidimensionais

- MOLAP (Multidimensional OLAP)

- HOLAP (Hybrid OLAP)

- WOLAP (Web OLAP)

- DOLAP (Desktop OLAP)

# Data Mart

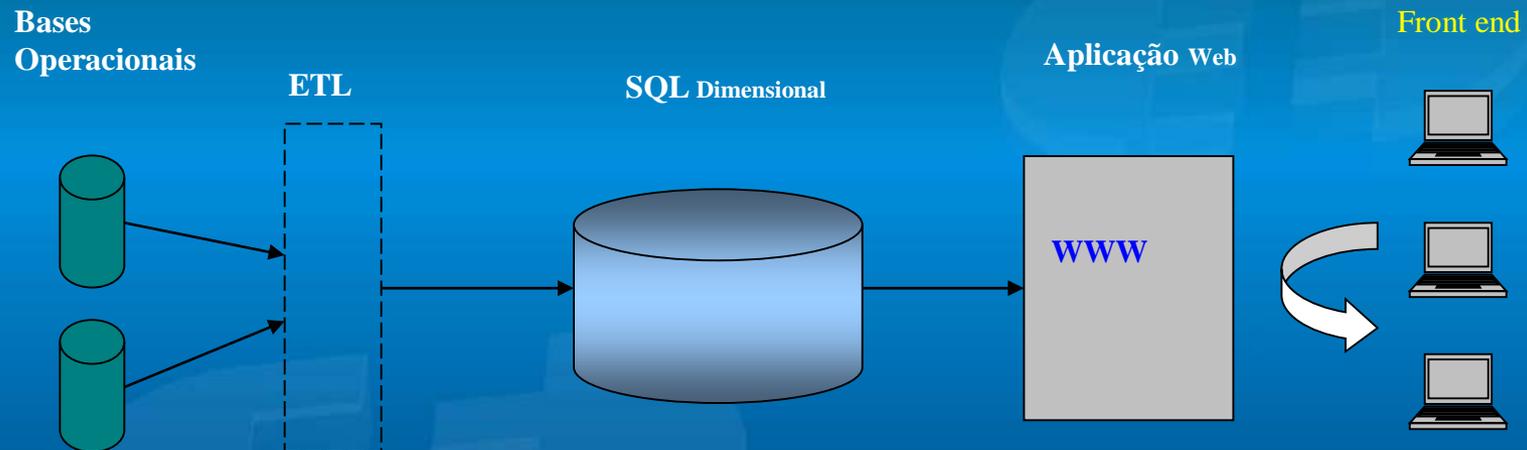
- Subconjunto lógico de um Data Warehouse completo customizado para um departamento em particular
- “Data Marts são subconjuntos de dados de uma empresa, armazenados fisicamente em mais de um local, geralmente divididos por departamento (Data Marts ’departamentais’)”
- Abordagem
  - Abordagem Monolítica ou *Top Down*
  - Abordagem Evolutiva *Bottom Up*

# Data Mart

- Principais características de um Data Mart
  - Rapidez na implementação
  - Baixo custo
  - Controle local em vez de centralizado
  - Redução do tempo de resposta a consultas, tornando a relação entre custo-benefício muito favorável

# Data Mart Ouvidoria

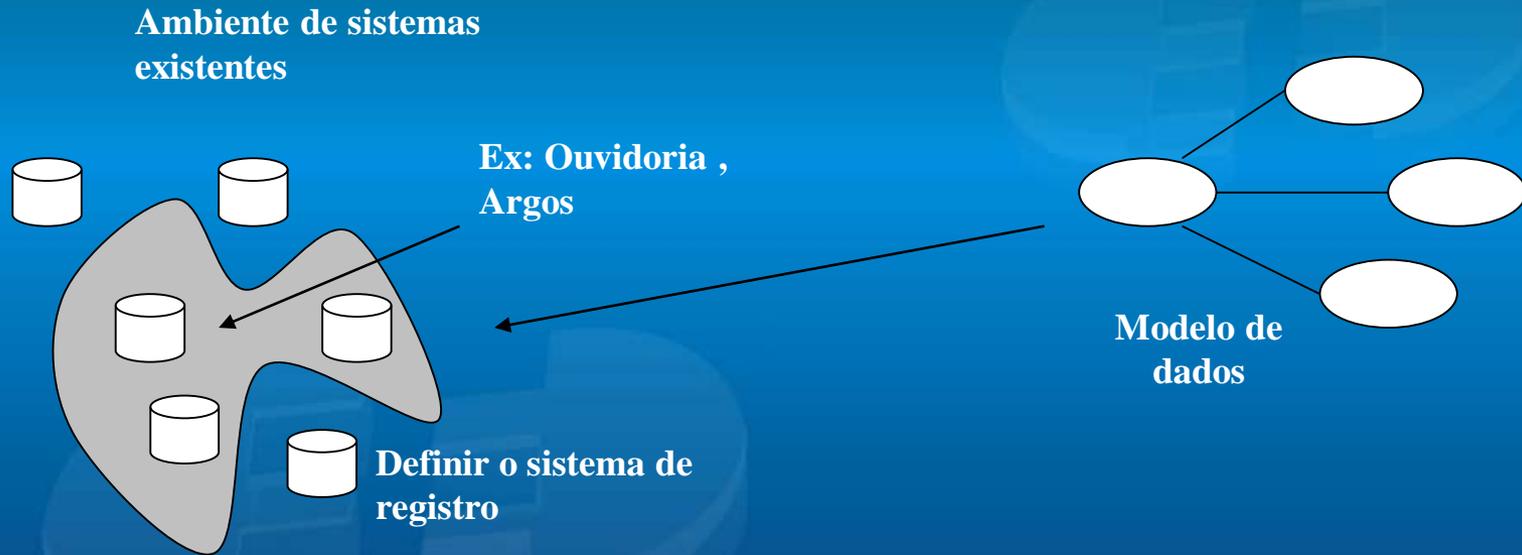
- Projeto lógico desenvolvido baseado na ferramenta Sagent



- Projeto genérico para qualquer ferramenta

# Data Mart Ouvidoria

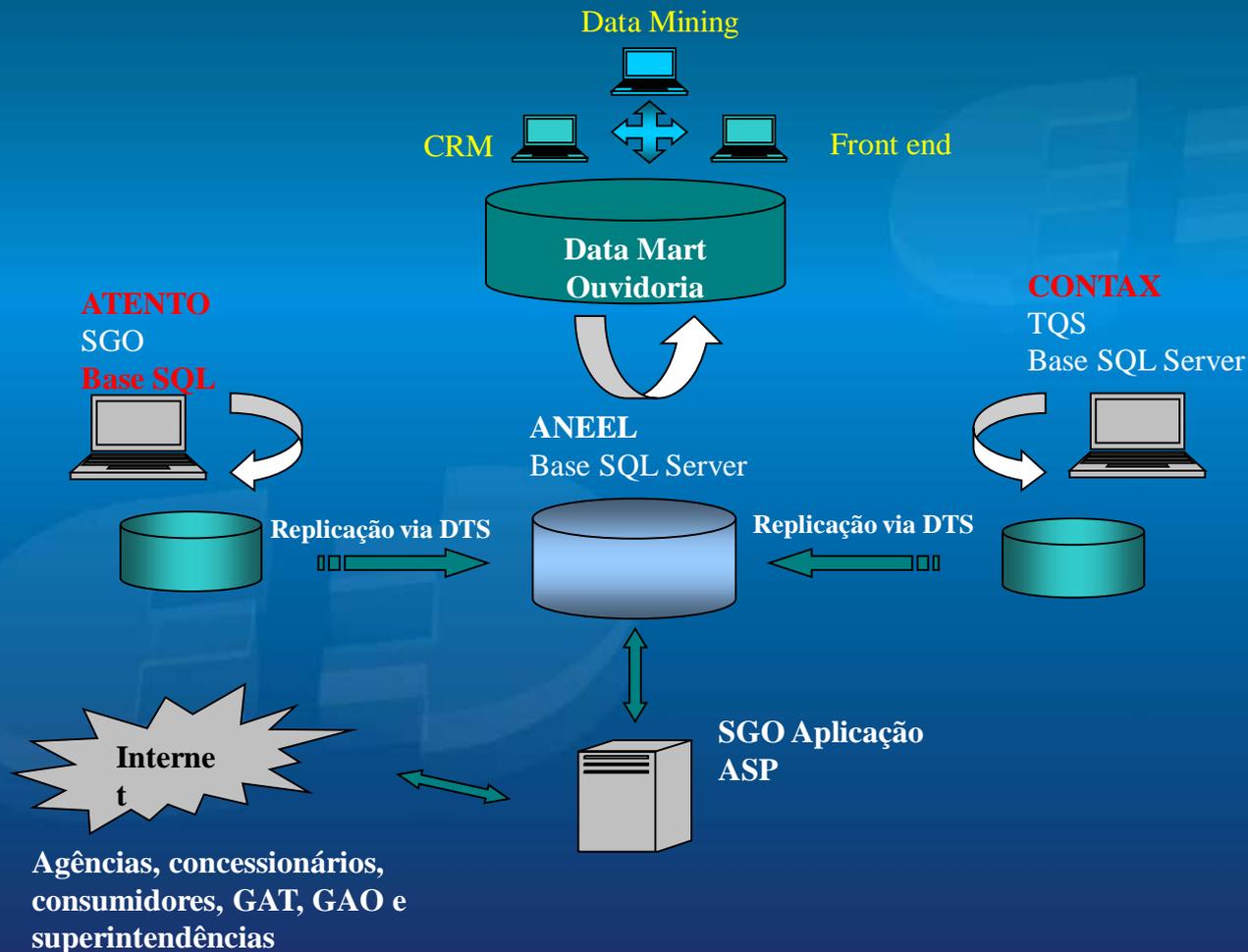
- Modelo dimensional representando a melhor informação existente na base



- Fotografia diária (granularidade)

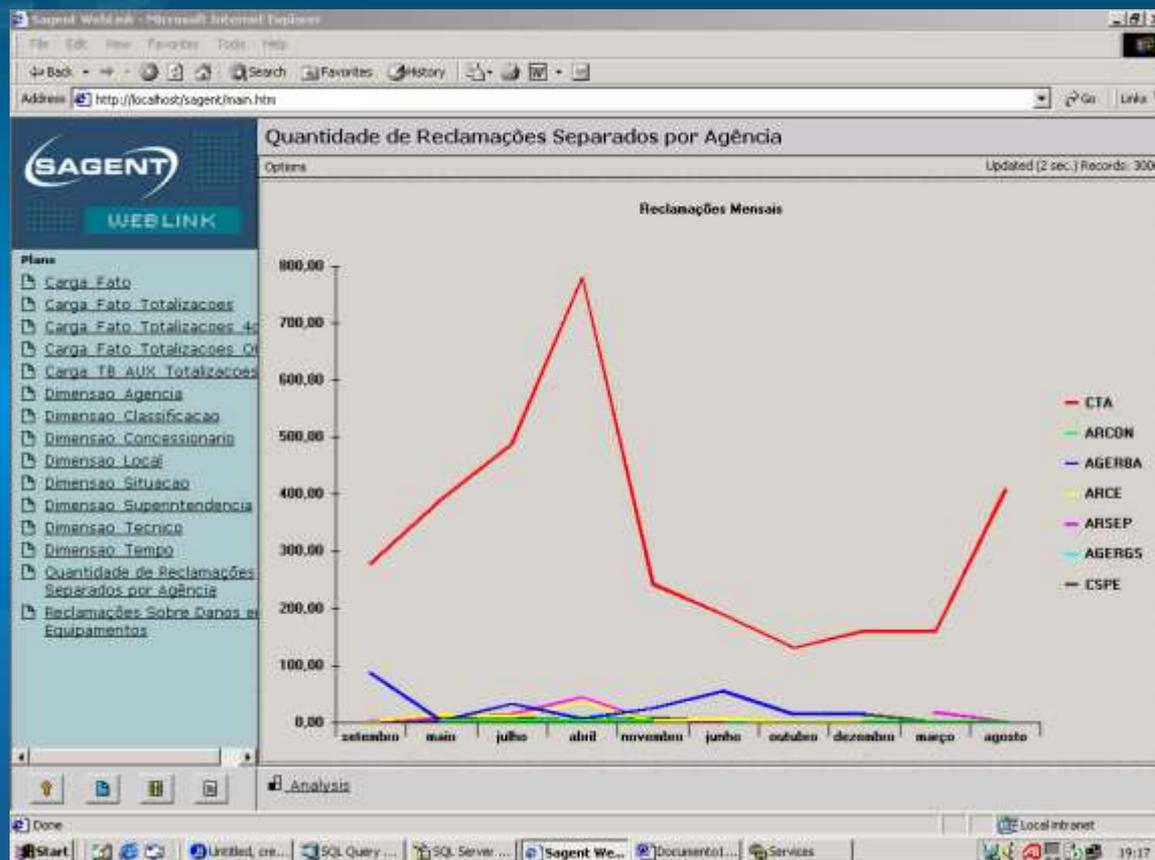
# Data Mart Ouvidoria

- Representação da solução final

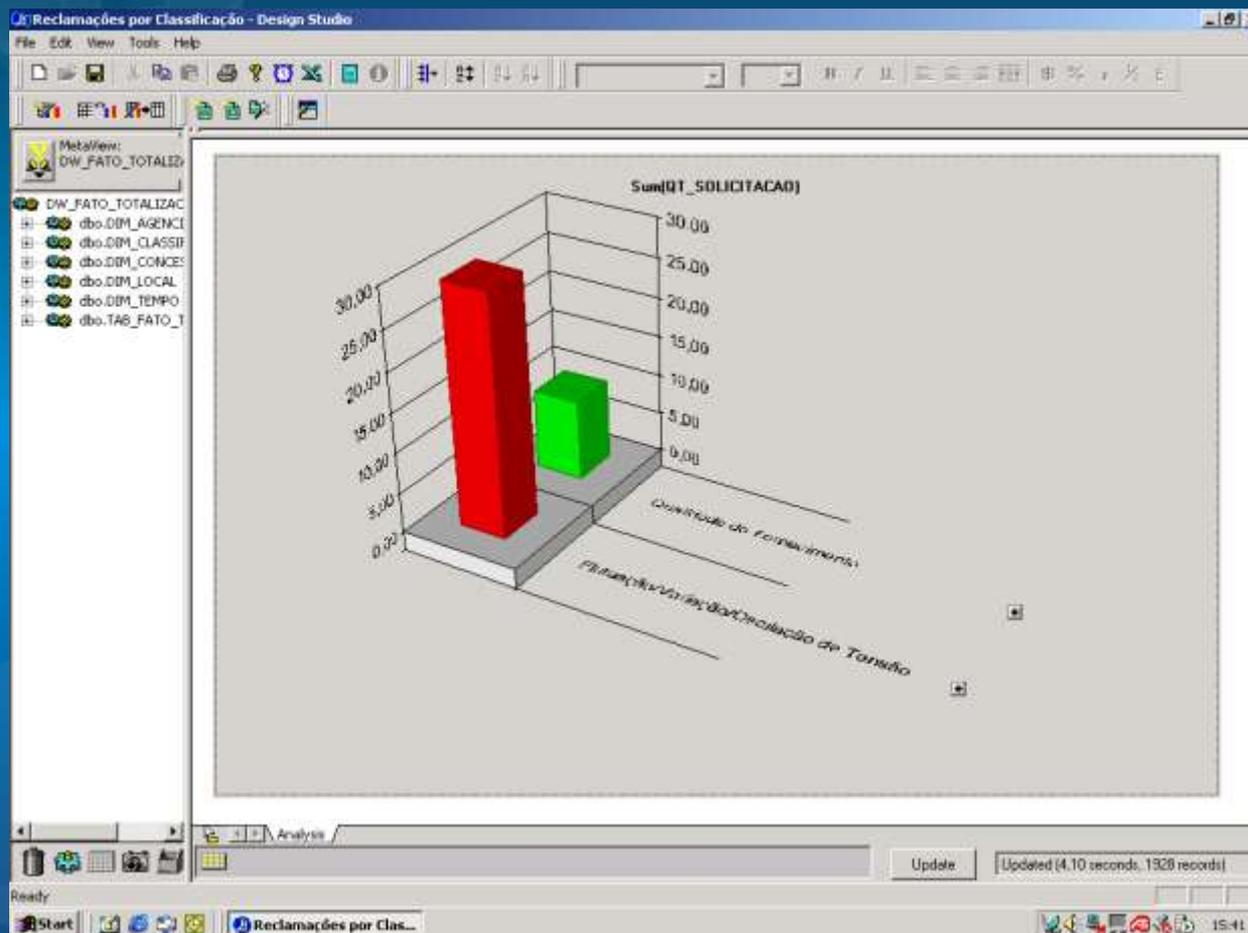




# Exemplos de consultas Front end



# Exemplos de consultas Front end



# Vantagens

- **Precisão nas informações**
- **Performance**
- **Visualização das informações**
- **Ambiente separado da base operacional**

# CRM

- **Gestão do conhecimento**
- **Perfil do consumidor (saber quem é o reclamante)**
- **Dificuldade em colher informações sobre o consumidor**
- **Influenciar**

# Conclusão

- **O conceito Data Warehouse se tornou mundialmente conhecido como a solução mais adequada para sistemas de apoio à decisão**
- **A única solução!**
- **Complexidade da Ouvidoria**
- **Custos**
- **Ferramentas**
- **Previsão aproximada U\$ 400.000**

# Conclusão

- Padronização para DW corporativo
- Integração com outras bases
- Data Mining

