



CENTRO UNIVERSITÁRIO UNIVATES
CURSO DE SISTEMAS DE INFORMAÇÃO

**UM ESTUDO DE CASO SOBRE A IMPLANTAÇÃO DE BUSINESS
INTELLIGENCE NA COUROS BOM RETIRO**

Leandro Görgen Portz

Lajeado, novembro de 2014

Leandro Görgen Portz

UM ESTUDO DE CASO SOBRE A IMPLANTAÇÃO DE BUSINESS INTELLIGENCE NA COUROS BOM RETIRO

Trabalho de Conclusão de Curso II apresentado ao
Centro de Tecnologia da Informação do Centro
Universitário UNIVATES, como parte dos requisitos
para a obtenção do título de Bacharel em Sistemas de
Informação.

Orientador: Prof. Ms. Pablo Dall'Oglio

Lajeado, novembro de 2014

Leandro Görgen Portz

UM ESTUDO DE CASO SOBRE A IMPLANTAÇÃO DE BUSINESS INTELLIGENCE NA COUROS BOM RETIRO

Este trabalho foi julgado adequado para a obtenção do título de bacharel em Sistemas de Informação do CETEC e aprovado em sua forma final pelo Orientador e pela Banca Examinadora abaixo:

Prof. Pablo Dall'Oglia – UNIVATES, Mestre em computação aplicada pela UNISINOS – São Leopoldo, Brasil.

Prof. Vílson Cristiano Gärtner, UNIVATES, Mestre em computação aplicada pela UNISINOS – São Leopoldo, Brasil.

Prof. Juliano Dertzbacher, UNIVATES, Mestre em computação pela UFRGS, Porto Alegre, Brasil.

Lajeado, novembro de 2014

DEDICATÓRIA

Dedico este trabalho aos meus pais, em especial pela dedicação e apoio em todos os momentos.

AGRADECIMENTOS

A Couros Bom Retiro pelo apoio, os colegas de trabalho pelo tempo disponibilizado para a realização deste trabalho.

Aos professores da UNIVATES pelos conhecimentos e experiências transmitidos, principalmente ao meu orientador Pablo pela oportunidade de realização deste trabalho, paciência e auxílio.

A minha noiva, Datini, pelos puxões de orelha nos momentos de distração e pelo apoio e compreensão em todos os momentos. A minha prima Joice Junqueira pela revisão do presente documento.

A todos meus colegas de curso pelas experiências compartilhadas e principalmente pelas amizades construídas ao longo do curso.

RESUMO

Atualmente, o mercado muda rapidamente e as organizações devem adequar-se ao novo cenário na mesma velocidade. Para que isso aconteça, as organizações devem possuir um processo de tomada de decisão eficaz e eficiente. Este fator torna a informação um dos bens mais valiosos das organizações. Para que esse processo seja possível, este deve estar baseado em informações integrais e confiáveis. Uma das estratégias que vem sendo muito utilizada para obter informações com estas qualidades é a implantação de um *Data Warehouse*, que integra as informações de várias fontes e disponibiliza tais informações para os tomadores de decisão, através de uma ferramenta de *Business Intelligence*, com o intuito de gerar informações relevantes para o processo de tomada de decisão. O presente trabalho tem como objetivo a modelagem de um *Data Warehouse*, bem como a implantação de uma ferramenta de *Business Intelligence*, assim como executar todos os processos necessários para seu correto funcionamento, na empresa Couros Bom Retiro.

Palavras-chave: *Informação, Decisão, Business Intelligence, Data Warehouse, OLAP, ETL.*

ABSTRACT

Currently the market is rapidly changing and organizations must adapt to the new environment at the same speed. To make this possible, organizations must have a process for making effective and efficient decisions. This makes the information one of the most valuable assets of organizations. For this process to be possible it must be based on reliable and accurate information. One strategy that has been widely used for information with these qualities is the implementation of a data warehouse, where it integrates information from multiple sources and makes it available to decision makers, through a Business Intelligence tool, in order to generate information relevant to the process of decision making. The present work goal is to model a Data Warehouse, and implement a Business Intelligence tool, as well as all to perform all the processes required for correct operation, on the company Couros Bom Retiro.

Keywords: **Information, Decision, Business Intelligence, Data Warehouse, OLAP, ETL.**

LISTA DE FIGURAS

Figura 1 – Pirâmide organizacional.....	23
Figura 2 – Estrutura básica de um sistema de apoio a decisão.....	25
Figura 3 – Cubo de dados com 4 dimensões.....	28
Figura 4 – Ambiente de Data Warehouse.....	33
Figura 5 – Representação gráfica de um DW com ODS.....	33
Figura 6 – Cubo de dados.....	35
Figura 7 – Representação gráfica dos esquemas estrela, flocos de neve e multiestrela.....	38
Figura 8 – Cubo de compras.....	54
Figura 9 – Cubo de consumo.....	56
Figura 10 – Cubo de Ajuste.....	58
Figura 11 – Cubo de estoque.....	60
Figura 12 – Cubo de Faturas.....	63
Figura 13 – Cubo de Comissões.....	66
Figura 14 – Diagrama de atividade do processo de ETL da área de compras.....	70
Figura 15 – Diagrama de atividade do processo de ETL da área comercial.....	73
Figura 16 – Arquitetura do ambiente.....	75
Figura 17 – Tela de principal do Pentaho.....	79
Figura 18 – Configuração do DB.....	80
Figura 19 – Marketplace.....	81
Figura 20 – Pentaho Schema Workbench editando o schema estoque.....	82
Figura 21 – Tabela e seus atributos.....	83
Figura 22 – Atributos da dimensão.....	84

Figura 23 – Atributos do nível hierárquico.....	85
Figura 24 – Atributos da métrica.....	86
Figura 25 – Saiku com o cubo de compras.....	87
Figura 26 – Tela de publicação.....	88
Figura 27 – Tela principal do CDE.....	89
Figura 28 – Tela de criação das consultas.....	90
Figura 29 – Tela de configuração dos gráficos.....	91
Figura 30 – Primeira parte do painel da área de compras.....	93
Figura 31 – Segunda parte de painel da área de compras.....	94
Figura 32 – Painel da área comercial.....	95

LISTA DE CÓDIGOS

Listagem 1 – Código SQL para criação das dimensões tempo, produto e filial.....	50
Listagem 2 – Código SQL da função PopulaTempo.....	51
Listagem 3 – Código SQL para criação do fato compra e a dimensão fornecedor.....	54
Listagem 4 – Código SQL para criação do fato compra e a dimensão requisição.....	56
Listagem 5 – Código SQL para criação do fato ajuste.....	58
Listagem 6 – Código SQL para criação do fato estoque.....	60
Listagem 7 – Código SQL para criação do fato fatura e as dimensões cliente, tipo de fatura e banco.....	63
Listagem 8 – Código SQL para criação do fato comissão e as dimensões representante e tipo de comissão.....	66

LISTA DE TABELAS

Tabela 1 – Mapa de transformação das dimensões compartilhadas.....	52
Tabela 2 – Mapa de extração.....	67
Tabela 3 – Tabela de Informação volumétrica da área de compras.....	68
Tabela 4 – Mapa de transformação da área de compras.....	69
Tabela 5 – Tabela de Informação volumétrica da área comercial.....	71
Tabela 6 – Tabela de transformação da área comercial.....	72

LISTA DE ABREVIATURAS

ACID –	Atomicidade, Consistência, Isolamento e Durabilidade
BD –	Banco de Dados
BI –	<i>Business Intelligence</i>
CDE –	<i>Community Dashboard Editor</i>
CNPJ –	Cadastro Nacional de Pessoa Jurídica
CRM –	Customer Relationship Management
DER –	Diagrama Entidade e Relacionamento
DSS –	<i>Decision Support System</i>
DW –	<i>Data Warehouse</i>
EIS –	<i>Executive Information System</i>
ER –	Entidade e Relacionamento
ERP –	<i>Enterprise Resource Planning</i>
ETL –	<i>Extract, Transform e Load</i>
FAD –	Ferramentas de Apoio a Decisão
JNDI –	<i>Java Naming and Directory Interface</i>
ODS –	<i>Operational Data Store</i>
OLAP –	<i>Analytic Processing On-Line</i>
PHP –	<i>Hypertext Preprocessor</i>
PSW –	<i>Pentaho Schema Workbench</i>
SAD –	Sistema de Apoio a Decisão
SGBD –	Sistema Gerenciador de Banco de Dados
SI –	Sistemas de Informação

SIG –	Sistemas de Informações Gerenciais
SIT –	Sistemas de Informação Transacionais
SO –	Sistema Operacional
SQL –	<i>Structured Query Language</i>
TI –	Tecnologia da Informação

SUMÁRIO

1 INTRODUÇÃO.....	15
1.1 Motivação.....	17
1.2 Objetivos.....	18
1.3 Organização do Trabalho.....	19
2 REFERENCIAL TEÓRICO.....	21
2.1 Tomada de decisão.....	21
2.2 Sistemas de apoio a decisão.....	23
2.3 Business Intelligence.....	25
2.4 Ferramentas OLAP.....	27
2.4.1 Operações OLAP.....	30
2.5 Data Warehouse.....	31
2.5.1 Características de um Data Warehouse.....	31
2.5.2 Arquitetura de um Data Warehouse.....	32
2.5.3 Modelagem dimensional.....	34
2.5.4 Passos para modelagem dimensional.....	39
2.6 ETL.....	40
3 METODOLOGIA.....	42
4 TRABALHO PROPOSTO.....	45
4.1 Visão geral.....	45
4.1.1 Caracterização da empresa.....	45
4.1.2 Justificativa.....	46
4.1.3 Áreas de negócio.....	46
4.2 Processos modelados.....	47
4.2.1 Levantamento de requisitos.....	47
4.2.2 Grupos de informações identificados.....	47
4.3 Modelagem dimensional.....	48
4.3.1 Informações compartilhadas.....	49
4.3.2 Área de Compras.....	52

4.3.3 Área comercial.....	61
4.4 Transformação dos dados.....	67
4.4.1 Área de compras.....	68
4.4.2 Área comercial.....	71
4.5 Escolha da ferramenta de BI.....	73
4.6 Arquitetura.....	75
4.7 Tecnologias utilizadas.....	76
5 AVALIAÇÃO DO BI.....	78
5.1 Implantação.....	78
5.1.1 Criação dos cubos de dados.....	81
5.1.2 Criação dos dashboards.....	88
5.1.3 Dashboards criados.....	92
5.2 Disponibilização.....	95
5.3 Questionário de avaliação.....	96
6 CONCLUSÕES.....	98
REFERÊNCIAS.....	100
APÊNDICES.....	102

1 INTRODUÇÃO

Com o passar das décadas, a informação vem obtendo uma maior relevância no contexto das organizações. Segundo Freitas et al. (1997), no ambiente organizacional, a informação já é considerada como um recurso vital, assim como a mão de obra e a matéria-prima. A informação é vista como um elemento decisivo que pode determinar o sucesso ou o fracasso de uma organização.

A informação obtém uma crescente importância, o aumento da concorrência e a globalização dos negócios, obrigam as organizações a enfrentarem novas situações, onde a informação torna-se fundamental. A qualidade da decisão tomada pelos gestores vai depender diretamente da qualidade e relevância das informações que forem disponibilizadas pelos sistemas da organização. Segundo Colaço (2004), para levar as corporações a um lugar de destaque, os administradores precisam ter a capacidade de analisar os dados disponíveis e tomar decisões rápidas e seguras. Para Freitas et al. (1997), as organizações que conseguem trabalhar a informação de uma maneira satisfatória certamente terão maior competitividade.

Há algumas décadas, as organizações vêm utilizando Sistemas de Informação Transacionais (SIT) para registrar suas transações de negócio, tais como: entradas e saídas no estoque, emissões de nota fiscal, gerações de duplicatas, folha de pagamento, lançamentos contábeis, ordens de produção e de compra, pedidos, dentre outros. Este tipo de sistema é caracterizado principalmente pelo seu grande volume de transações e alta taxa de atualizações. As organizações também fazem uso de Sistemas de Informações Gerenciais

(SIG) para extrair informações que contribuam para o processo de tomada de decisão, o que nem sempre é uma tarefa fácil.

O SIG reúne informação de um SIT, apresentando as informações integradas e sumarizadas que auxiliam na análise, planejamento e suporte à decisão, além de permitir o acompanhamento do desempenho de um ou mais departamentos (COLAÇO, 2004). Geralmente estas duas categorias de sistemas são englobadas nos famosos pacotes *Enterprise Resource Planning* (ERP).

Embora forneça informações importantes para a tomada de decisões, um SIG possui algumas limitações tais como: raramente busca informações de várias fontes de dados; possui restrições quanto à capacidade de análise e cruzamento de informações, bem como não permite simulações com cenários futuros. Em geral, o SIG é eficaz no auxílio de decisões para problemas estruturados, que são caracterizados por sua linearidade, repetitividade e facilidade de definição do problema.

Para resolver problemas que estão além da capacidade de um SIG, foi criado o conceito de Sistema de Apoio a Decisão (SAD). Podemos conceituar um SAD como um ambiente projetado para apoiar, contribuir e influenciar o processo de tomada de decisão (COLAÇO, 2004). Segundo Freitas et al. (1997), os SAD são sistemas que contribuem para o processo de decisão, onde o foco está em suportar e não em automatizar as decisões.

Um SAD é um sistema que se caracteriza por permitir uma análise histórica dos dados da organização, pesquisar informações de várias fontes e permitir a criação de cenários para que uma decisão possa ser simulada. Em geral, este tipo de sistema é eficaz para auxiliar na resolução de problemas estruturados ou não estruturados, sendo que os não estruturados tem como característica a falta de completude e precisão nas informações necessárias para sua solução.

Uma tendência muito forte nos últimos anos são os *softwares de Business Intelligence* (BI), que podem ser entendidos de uma forma mais ampla, como a utilização de várias fontes de informação para definir estratégias de competitividade nos negócios da empresa (BARBIERI, 2011). Onde a ideia é olhar para os dados já existentes na organização, analisar e gerar conhecimento.

Para o correto funcionamento do BI se faz necessário o uso de uma base de dados específica para este fim, com uma modelagem dimensional, a qual denomina-se *Data Warehouse* ou simplesmente DW. Esse tipo de banco armazena as informações históricas da organização orientadas por assunto. Além disso, as informações de um DW não são voláteis, ou seja, uma vez carregadas, não sofrem alterações. Os dados encontrados dentro do DW são obtidos a partir dos sistemas da empresa, planilhas, dados externos a empresas, entre outras fontes. Com todos os dados unificados em uma nova base pode-se gerar consultas unificadas, cruzamentos de informações, entre outras operações (COLAÇO, 2004).

Para serem armazenados de forma íntegra, os dados devem passar por um processo chamado *Extract, Transform, Load* (ETL), o que traduzindo obtemos: extração, transformação e carga, para só então serem armazenados no DW.

A partir de então, esses dados ficam disponíveis para análises, consultas, acompanhamento de indicadores, entre outras possibilidades dentro de uma ferramenta de BI. Esse tipo de ferramenta proporciona uma maior liberdade para os gestores, pois estes ganham a possibilidade de visualizar, cruzar, gerar gráficos com os dados da organização sem a dependência da área de desenvolvimento para confecção de novos relatórios.

1.1 Motivação

Muitas empresas, atualmente, mantêm vários sistemas de informação para administrar suas atividades. Na grande maioria das vezes, estes sistemas são desenvolvidos por diferentes fornecedores, e assim, não compartilham um mesmo repositório de dados, o que pode gerar problemas de consistência de informações. Como exemplo, pode-se citar a dificuldade na tomada de decisão com informações distribuídas em bases de dados diferentes, o que configura um problema de integração. Para Colaço (2004) “os executivos sempre sofrem ao emitir relatórios gerenciais em sistemas diferentes e encontrarem resultados diferentes sobre o mesmo assunto”.

Um *Business Intelligence* é um Sistema de Apoio a Decisão, que resolve este tipo de problema por meio da carga de dados centralizada em um *Data Warehouse*, reunindo e integrando dados contidos em vários sistemas, a até mesmo fora deles, possibilitando que

gestores realizem consultas, façam simulações, entre outras operações, em uma base de dados íntegra (COLAÇO, 2004).

O BI apesar de ser um conceito já estabelecido, ainda não é realidade para todas empresas, pois algumas organizações desconhecem este tipo de solução, ou não acreditam que isso possa agregar algum resultado a seu negócio e em outros casos o custo do projeto é maior que os seus benefícios.

O grande desafio na implantação de um BI está na escolha das informações a disponibilizar, o que varia muito de empresa para empresa, pois, mesmo empresas do mesmo ramo estão inseridas em ambientes de negócios diferentes, tem enfoques diferentes, isso faz com que a necessidade de informações para tomada de decisão seja diferente. Neste sentido, um projeto de implantação de um sistema de BI em uma organização possui características que o tornam único.

A estratégia de um BI precisa estar sempre alinhada ao plano estratégico e interesses de cada empresa, e direcionada às metas previamente estabelecidas pelos gestores. Segundo Primak (2009, texto digital) “de nada adianta a empresa possuir todas as ferramentas disponíveis no mercado para a extração das informações e não saberem ao certo interpretá-las”. É fundamental que a organização saiba dar um rumo ao seu capital intelectual para que dessa forma as informações obtidas através do *Business Intelligence* atendam às expectativas esperadas (PRIMAK, 2009).

Assim, o presente trabalho propõe um estudo de caso para a implantação de uma ferramenta de BI na empresa Couros Bom Retiro LTDA, com sede na cidade de Teutônia-RS, com o objetivo de melhorar a qualidade dos dados sobre quais são baseadas as decisões necessárias para o bom desenvolvimento e gerenciamento da organização.

1.2 Objetivos

Este trabalho tem como objetivo geral implantar um sistema de *Business Intelligence* na empresa Couros Bom Retiro, visando integrar as informações e consequentemente contribuir para melhorar o processo de tomada de decisões.

Já como objetivos específicos podemos citar:

- Levantamento das necessidades de tomada de decisão;
- Projeto e construção do *Data warehouse*;
- Escolha de uma ferramenta de BI;
- Implementação das rotinas dos processos de ETL;
- Elaboração dos painéis e relatórios para os usuários;
- Treinamento para utilização do ambiente;
- Disponibilização da ferramenta para análise;
- Avaliação dos resultados obtidos.

1.3 Organização do Trabalho

O presente trabalho encontra-se subdividido da seguinte maneira: no capítulo dois apresenta-se o referencial teórico, cujo objetivo é explanar os conceitos utilizados para a construção do conhecimento necessário para a elaboração e entendimento do estudo de caso proposto.

Já no terceiro capítulo, encontra-se a descrição da metodologia utilizada para o desenvolvimento do presente trabalho, como se dará a pesquisa, bem como o levantamento de informações sobre o processo decisório, além das informações necessárias para a implantação do BI.

No quarto capítulo, serão apresentadas as técnicas de modelagem dimensional, assim como sua documentação, e a descrição dos processos de ETL.

No quinto capítulo, estão descritos os métodos utilizados para a disponibilização do ambiente para os usuários assim como a avaliação realizada por eles e a interpretação dos resultados obtidos.

No sexto capítulo, encontram-se as considerações finais do presente trabalho.

2 REFERENCIAL TEÓRICO

Neste capítulo apresentam-se os conceitos que são inerentes a total compreensão do estudo de caso proposto.

2.1 Tomada de decisão

Segundo Freitas et al. (1997) o ato de tomar decisão é inerente a todos os seres humanos, acontecendo nas mais variadas circunstâncias, faixas etárias e classes sociais.

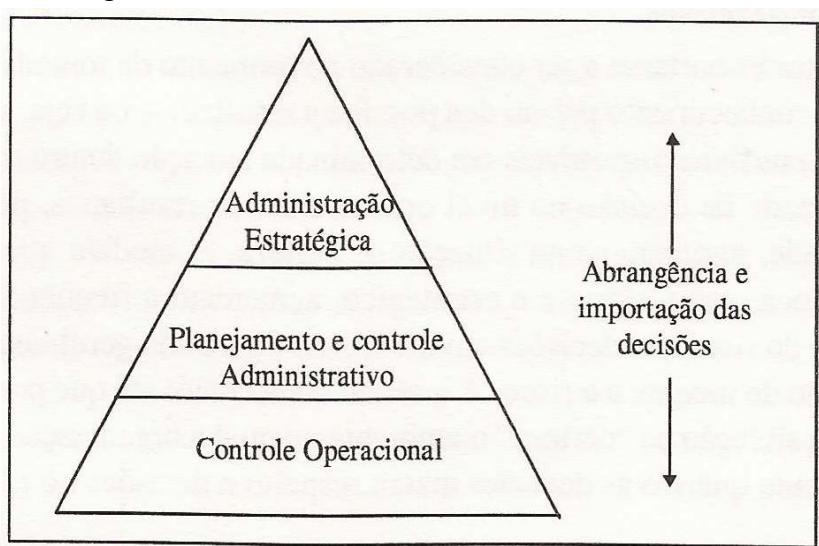
Nas organizações não é diferente, decisões são tomadas a todo momento sejam elas para resolver os problemas estruturados e os não estruturados. Os problemas estruturados, são caracterizados por sua linearidade, repetitividade, facilidade de definição e estrutura definida, é o tipo de situação que ocorre no cotidiano da organização. Já os problemas não estruturados são novos para a organização, onde a informação é incompleta e imprecisa e sem uma estrutura definida, este tipo de decisão ocorre esporadicamente nas organizações (FREITAS et al., 1997).

Além da sua estrutura, as decisões pode ser classificadas quanto à atividade administrativa a que pertencem, segundo três níveis apresentados abaixo (ANTHONY apud FREITAS et al., 1997, p. 54):

- **Nível operacional:** significa o uso eficaz e eficiente das instalações e recursos da organização para executar as operações da melhor forma possível; a decisão neste nível é um processo pelo qual se assegura que as atividades operacionais serão bem executadas; grande parte das decisões que ocorrem neste nível são estruturadas e os procedimentos a serem seguidos são geralmente muito consistentes; as decisões operacionais e suas ações geralmente geram uma resposta imediata;
- **Nível tático:** as decisões deste nível são normalmente relacionadas com o controle administrativo e são utilizadas para definir as ações de controle, formular novas regras de decisão que irão ser utilizadas no nível operacional, englobando também a aquisição de recursos, localização de projetos e novos produtos; neste nível são necessárias informações sobre o funcionamento planejado tomando por base normas, expectativas e pressupostos;
- **Nível estratégico:** engloba a definição de objetivos, políticas e critérios gerais para planejar o curso da organização; o propósito das decisões no nível estratégico é desenvolver estratégias para que a organização seja capaz de atingir seus objetivos; as atividades neste nível não possuem um período com ciclo uniforme; estas atividades podem ser irregulares, ainda que alguns planos estratégicos se façam dentro de planejamentos anuais ou em períodos preestabelecidos.

Na pirâmide da Figura 1, apresentada a seguir, encontra-se representado graficamente os três níveis apresentados anteriormente, dando a ideia clara de hierarquia entre os níveis. Por meio desta representação, se torna claro também o nível de abrangência das decisões que são tomadas em cada nível da organização, ou seja, quanto mais alto o nível maior será a abrangência das consequências da decisão tomada.

Figura 1 – Pirâmide organizacional.



Fonte: FREITAS et al.1997, p.55.

Segundo Leme (2004) os executivos vivem da tomada de decisões, e o processo que envolve a ação gera inúmeras incertezas. A solução para a mitigação das incertezas é, dispor do maior número de informações relevantes possíveis, para tornar isso uma realidade, é preciso fazer uso de um Sistema de Apoio a Decisão.

2.2 Sistemas de apoio a decisão

Segundo Leme (2004) é possível dividir os Sistemas de Informação (SI) das organizações em dois grandes grupos, onde no primeiro encontram-se os Sistemas de Informação Transacionais, que são responsáveis pelo controle das funções da empresa, como faturamento, contas a pagar e receber, folha de pagamento, entre outros. Já no segundo grupo, estão localizados os sistemas de possibilidades a análise e mensuração do desempenho da organização, estas ferramentas são os Sistemas de Apoio a Decisão.

Para Freitas et al. (1997) os SADs são sistemas que contribuem para o processo de decisão, onde o foco está em suportar e não em automatizar as decisões, ou seja, os sistemas são criados para apresentarem o maior número de informações relevantes a tomada de decisão para o decisor e não para automatizar o seu trabalho. Busca-se definir uma ferramenta que dê suporte ao usuário final, e não que venha substituir o seu julgamento e, por consequência, a sua decisão.

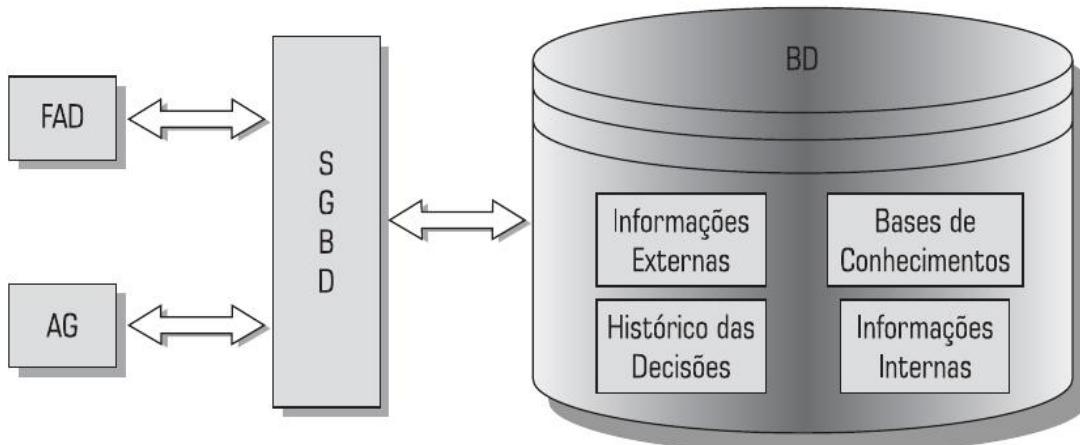
Sistema de Apoio a Decisão é um sistema computacional, que permite o decisor ampliar a sua capacidade na identificação e na resolução de problemas mal estruturados (COURBON apud FREITAS, 1997, p. 84).

Segundo Colaço (2004), muitos conceitos de Sistema de Apoio a Decisão o rotula como Sistema de Informação (SI) que apoia qualquer processo de tomada de decisão em todos os níveis da organização. Porém, isso não é o suficiente, a diferença está no fato de um SAD não apenas fornecer informações, mas também contribuir e influenciar o processo de tomada de decisão. Então, o conceito que melhor define um SAD está relacionado a um ambiente complexo, projetado para fornecer subsídios para que a alta gerência tome decisões. Este ambiente é composto pelos seguintes componentes:

- **Banco de Dados (BD):** Este pode ser visto como uma coleção de dados inter-relacionados. Nos ambientes de suporte a decisão estes dados pode ser tanto interno ou externos a organização, podem ser de conhecimentos e experiências de especialistas e de um histórico de decisões tomadas;
- **Sistema Gerenciador de Banco de Dados (SGBD):** Software desenvolvido para que usuários possam definir, criar e manipular Banco de Dados para as mais diversas finalidades, devem ser integrados e compartilhados para que as redundâncias de dados sejam mitigadas;
- **Aplicativos com características gerenciais:** São aplicativos com funções gerências de análise, ou seja, com a capacidade de analisar alternativas e fornecer sugestões para resolução dos problemas. Softwares com estas funcionalidades podem fazer parte do grande ambiente de suporte à decisão;
- **Ferramentas de Apoio a Decisão (FAD):** São aplicativos desenvolvidos para apresentar graficamente as informações, auxiliar a simulação de cenários, capacitar a análise ou descobrimento de conhecimento, entre outras funcionalidades. Estes aplicativos também são conhecidos como ferramentas de BI.

A Figura 2 apresenta como os componentes apresentados estão relacionados.

Figura 2 – Estrutura básica de um sistema de apoio a decisão.



Fonte: COLAÇO, 2004, p.10.

Como geralmente essas ferramentas são utilizadas por gestores, a sua interface deve ser de fácil utilização e flexível, pois devem possibilitar inúmeras formas de consultas e cruzamento de dados sem a necessidade de um conhecimento muito apurado em informática.

2.3 Business Intelligence

Business Intelligence (Inteligência Empresarial), ou apenas BI, é um termo do *Gartner Group*. É um conceito, um modelo que atende os gestores que ocupam posições estratégicas dentro das organizações, que estão diretamente ligadas ao negócio e que possuem poder de decisão e influência sobre os rumos da empresa.

O conceito surgiu na década de 80 e refere-se ao processo de coleta, organização, análise, compartilhamento e monitoramento de informações que oferecem suporte a gestão de negócios, desenvolvendo percepções e entendimentos a seu respeito, o que lhes permite tornar o processo de tomada de decisão mais consciente.

Para Leme (2004) o termo BI reúne todos os assuntos ligados à tomada de decisão, armazenamento de dados, publicação de informações, até a produção de “cubos” multidimensionais de dados, o que será apresentado na subseção 2.5. Segundo Barbieri (2011) o BI, pode se compreendido como a utilização de várias fontes de dados para transformá-los

em informação que possibilitem a definição de estratégias competitivas, análises e tomadas de decisões.

Com o passar do tempo, BI ganhou maior dimensão e importância, dentro de um processo natural de aperfeiçoamento e crescimento, anexando uma grande quantidade de ferramentas, tais como: o Executive Information System (EIS), o Decision Support System (DSS), planilhas eletrônicas, geradores de consultas e relatórios, Data Mining, ferramentas *Analytic Processing On-Line* (OLAP), entre tantas outras, que permite às empresas transformar dados gerados por seus sistemas de informação transacionais em informação qualitativa e relevante para a tomada de decisão (PRIMAK, 2009, texto digital).

Mas as empresas que implantaram o *Enterprise Resource Planning* (ERP), cuja função é controlar e gerenciar o operacional das empresas, registrando, processando e documentando cada novo fato do cotidiano da organização e distribuindo essas informações de maneira clara e segura, em tempo real, chegaram à conclusão de que apenas armazenar grande quantidade de dados de nada adianta se estes se encontram repetidos, incompletos e pulverizados em vários sistemas dentro da organização. Chegou-se à conclusão que era necessário a utilização de ferramentas que possibilitassem o agrupamento destes dados em uma base única e integrá-los de forma que possibilitem a realização de análises sob diferentes ângulos (PRIMAK, 2009, texto digital).

Para Barbieri (2011) as ferramentas de BI são baseadas em processos fortemente transformadores de dados, tornando a qualidade dos dados que serve como base para a ferramenta, cada vez mais, de extrema importância, pois, a qualidade dos dados de entrada vai refletir diretamente na qualidade da informação utilizada na tomada de decisão.

Com a implantação de uma ferramenta de BI, os gestores poderão ter acesso às informações de uma maneira ágil, abreviando assim o tempo de resposta a problemas ou à mudanças no mercado. Segundo Leme (2004) a implantação de um sistema de BI apresenta vários benefícios, entre os quais estão os citados a seguir:

- Possibilidade de reconhecer, criar, organizar e usufruir dos ativos informacionais de uma organização;
- Antecipação às mudanças de mercado;
- Conhecimento sobre o negócio;

- Aprendizado pelos sucessos e falhas internos e dos concorrentes;
- Visão clara sobre novos mercados;
- Compreender as tendências dos negócios, melhorando a consistência no momento de decisão de estratégias e ações a serem tomadas;
- Facilitar a identificação de riscos;
- Planejamento corporativo mais amplo;
- Facilitar o acesso e distribuir informação de modo mais amplo para obter envolvimento de todos dentro da empresa;
- Oferecer dados estratégicos para análise com um mínimo de atraso em relação a uma transação ou evento dentro da empresa.

2.4 Ferramentas OLAP

Segundo Colaço (2004) uma das tarefas mais solicitadas da equipe de Tecnologia de Informação (TI) é a construção de consultas ou relatórios, sejam estas simples listagens ou complexos mapas de custos, que apresentem informações sobre a organização. Como este tipo de demanda é difícil de ser prevista na concepção ou aquisição de sistemas, a solução da indústria de TI foi o desenvolvimento de ferramentas para geração de relatórios, porém, com o surgimento do *Data Warehouse* e a elevação da relevância da informação no ambiente organizacional, surge a necessidade de ferramentas com maior poder de análise, surge então em 1993 através do Dr. Codd¹, a tecnologia *Analytic Processing On-Line* (OLAP) ou processamento analítico on-line, que caracteriza o conjunto de técnicas utilizadas para navegar por informações contidas em *Data Warehouse*.

As ferramentas de OLAP são capazes de navegar pelos dados de um *Data Warehouse*, com o objetivo final de transformar dados em informações capazes de dar suporte ao processo

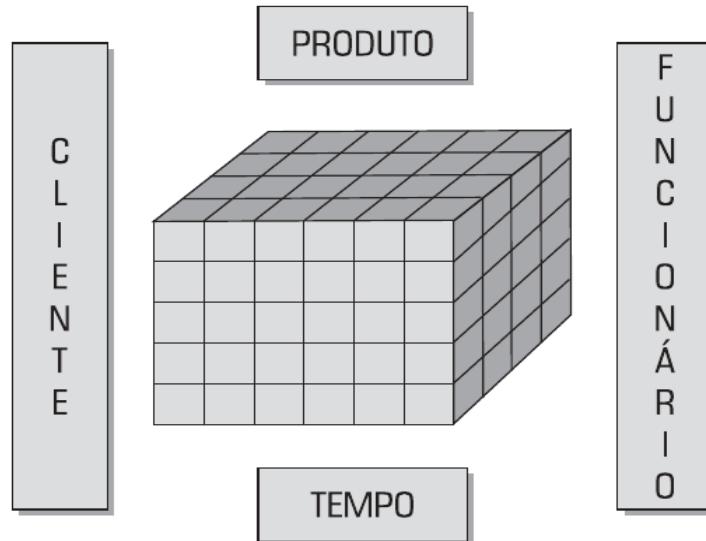
¹Cr. E.F. (Ted) Codd escreveu em 1993 um ensaio intitulado *Providing OLAP to User-Analysts: An IT Mandate*, onde foi criado o termo OLAP.

de tomada de decisão de forma amigável e flexível ao usuário em tempo hábil (Araújo; Batista; Magalhães, texto digital).

Atualmente as linguagens de programação e as principais organizações mantenedoras de SGBDs oferecem APIs² e componentes prontos para criação de softwares OLAP, causando em alguns gestores de projetos de TI a falsa impressão de simplicidade no que envolve o desenvolvimento deste tipo de software os encorajando a desenvolvê-los, o que geralmente resulta em projetos fracassados ou em ferramentas com carência interminável de manutenção. O ideal é escolher uma ferramenta OLAP que venha pronta, seja comprando-a ou buscando em comunidades de softwares livres, este tipo de ferramenta possui algumas características, sendo a principal delas, a possibilidade de visualização multidimensional dos dados.

Os dados são modelados na estrutura denominada cubo, onde em cada dimensão fica representado um tema que representa um assunto importante da empresa. Na Figura 3, pode ser visto um exemplo de cubo.

Figura 3 – Cubo de dados com 4 dimensões.



Fonte: COLAÇO, 2004, p.29.

Para Colaço (2004) as ferramentas de OLAP possuem doze regras, que foram criadas por Dr. Codd, para sua avaliação, estas são descritas a seguir:

- **Visão conceitual multidimensional:** os dados são estruturados em várias dimensões, desta maneira, possibilitando o cruzamento de informações de diversas formas;

²Application Program Interface – Um conjunto de funções predefinidas, documentadas e disponibilizadas por um software que serve de interface para outras aplicações interagirem com o mesmo.

- **Transparência:** as solicitações enviadas a uma ferramenta OLAP devem ser atendidas, não importando a origem dos dados, de forma transparente ao usuário final;
- **Acessibilidade:** os softwares OLAP devem suportar conexão com todas as bases de dados onde as informações foram mapeadas;
- **Desempenho de informações consistentes:** significa o conhecimento de todas as informações armazenadas a fim de que a disponibilização das consultas, para o usuário seja simples;
- **Arquitetura cliente/servidor:** as aplicações OLAP deve ser concebidas na arquitetura cliente/servidor, com o intuito de atender a qualquer usuário independentemente do local em que ele se encontre;
- **Dimensionalidade genérica:** deve possuir a capacidade de tratar os dados com a quantidade de dimensões que forem necessário para geração da informação;
- **Manipulação de dados dinâmicos:** deve tratar as células nulas, ou sem valor, que usualmente ocorrem com o grande volume de dados registrados nas diversas dimensões;
- **Suporte a multiusuários:** habitualmente a ferramenta OLAP vai ser acessada por vários usuários e estes podem estar analisando o mesmo cubo de dados. As ferramentas devem suportar este acesso simultâneo;
- **Operações ilimitadas em dimensões cruzadas:** os softwares OLAP devem possuir a capacidade de navegação em todas as dimensões que os dados que estão sendo analisados possuírem;
- **Manipulação intuitiva dos dados:** a ferramenta deve ser intuitiva, ou seja, o usuário deve poder manipular os dados livremente, ou seja, sem nenhum tipo de auxílio;
- **Flexibilidade nas consultas:** a ferramenta deve possibilitar que o usuário possa efetuar qualquer tipo de consulta sobre os dados disponibilizados a ele;
- **Níveis de dimensão e agregação ilimitados:** devido ao grande volume de dados gerado pelas dimensões, a ferramenta deve prover vários níveis de associações de dados.

Fundamentalmente, ferramentas OLAP são softwares direcionados a usuários finais para fazerem uso dos dados registrados no DW, gerando relatórios que são capazes de auxiliar a tomada de decisão (LEME, 2004).

2.4.1 Operações OLAP

Para que a geração das informações seja possível, as ferramentas OLAP devem permitir a montagem e manipulação de cenários. Atualmente estas ferramentas suportam funções de derivação de dados complexos, o que é nomeado *Slice and Dice*. Estas funções servem para alterar a ordem das dimensões, alterar linhas por colunas, com o intuito de permitir a análise das informações por outros ângulos, permitindo a investigação de diferentes inter-relacionamentos. O *Slice and Dice* engloba quatro operações, que são (COLAÇO, 2004):

- **Ranging:** operação responsável por modificar o resultado das consultas incluindo ou removendo dados, para que isso ocorra é necessário a intervenção do usuário, como por exemplo a inclusão de um novo fornecedor em uma consulta, o resultado refletirá nas demais operações;
- **Drilling:** operação responsável por permitir a navegação entre diferentes níveis de detalhamento de um cubo de dados. Esta operação possui um subconjunto de operações que possibilitam a navegação, que são: *Drill down*, que permite aumentar o nível de detalhamento, *Drill Across*, que permite a navegação transversal dos dados, e *Drill up* ou *Roll up*, que permite diminuir o nível de detalhamento dos dados;
- **Rotation:** operação responsável por permitir a alteração da forma de visualização dos dados, sem a alteração de conteúdo visualizado. A grande vantagem no *rotation* é que a operação é processada apenas em memória sem novas consultas no servidor;
- **Ranking:** operação responsável por permitir que os usuários filtrem os dados apresentados, eliminando as informações desnecessárias para o processo de tomada de decisão.

2.5 Data Warehouse

Surgiu na década 90 um conceito cujo foco é a grande base de dados informacionais, ou seja, um repositório único de dados, chamado de *Data Warehouse* (Armazém de dados). Para Leme (2004), DW está entre os principais componentes para a execução prática de um projeto de *Business Intelligence*.

Segundo Colaço (2004) *Data Warehouse* pode ser conceituado como um Banco de Dados projetado para armazenar informações integradas de toda organização, separado lógica e fisicamente dos sistemas da organização, formando assim um histórico destas informações.

2.5.1 Características de um Data Warehouse

O *Data Warehouse* segundo Leme (2004) possui quatro características principais, que são:

- **Não volatilidade:** Os dados armazenados dentro de um *Data Warehouse* não são voláteis, ou seja, estes representam uma fotografia da empresa no momento de sua extração. Os dados gravados neste repositório não devem ser alterados após sua gravação, com apenas algumas exceções, como por exemplo, uma falha no processo de carga. Assim é possível fazer a comparação entre dois momentos distintos da organização sem o risco de que uma alteração no sistema transacional da empresa tenha efeito nos dados históricos armazenados dentro do *Data Warehouse*;
- **Estrutura por assunto:** As informações armazenadas dentro desta base de dados estão agrupadas por um determinado assunto, como por exemplo, faturamento, contas a pagar, contas a receber, entre outros. Cada assunto é considerado um cubo de dados ou cubo de conhecimento. Cada assunto é representado por uma ou mais tabelas fato, na qual são registradas as transações da organização;
- **Integrado:** Para Leme (2004), esta possivelmente seja a característica mais importante em um *Data Warehouse*. Os dados registrados em um *Data Warehouse* são

provenientes de sistemas distintos, com padrões e formatos diferentes. Porém antes do seu efetivo registro, os dados devem passar por um processo que os tornem totalmente íntegros para que assim assumam um único padrão;

- **Variável com o tempo:** Os sistemas de informação transacional fazem alterações nos dados de suas bases a sempre que forem necessário, perdendo assim o histórico da informação, pois em uma consulta o valor retornado será sempre o último registrado. Já em um *Data Warehouse* quando uma alteração é feita gera um novo registo, mantendo assim o histórico da informação.

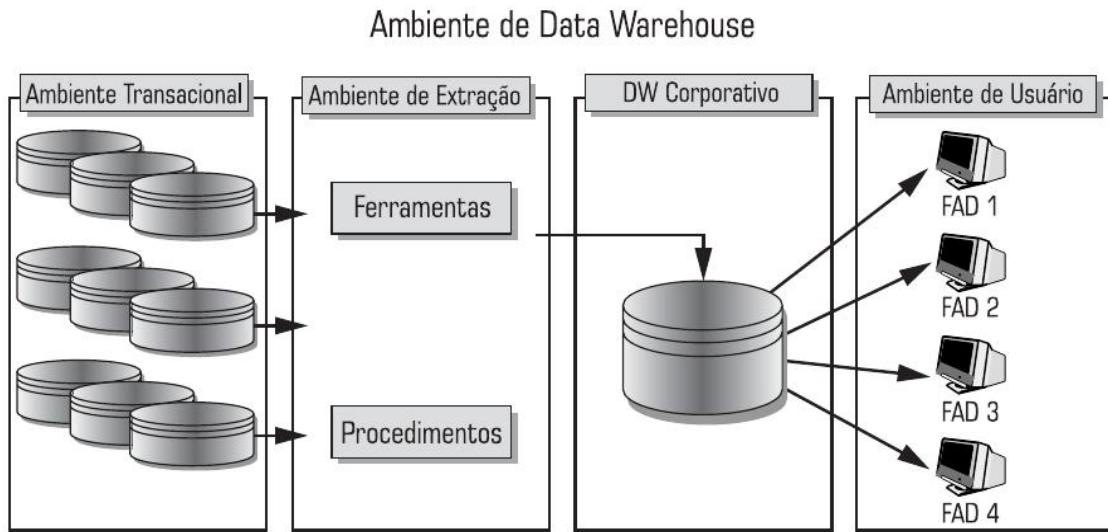
Para que o *Data Warehouse* mantenha as características citadas acima, os dados dos sistemas transacionais da organização, bem como os dados externos, precisam passar por um processo de extração e transformação para depois serem armazenados. A seguir, apresenta-se a arquitetura do sistema onde estes dados são armazenados.

2.5.2 Arquitetura de um Data Warehouse

Segundo Colaço (2004) os dados são inicialmente extraídos das aplicações da organização, sendo armazenado em uma área intermediária chamada *Staging Area*, ou área de estágio. Neste local, os dados são integrados e limpos, para só então serem exportados para o DW. No processo de integração, os dados passam por um processo que os colocam em conformidade com um padrão. Já no processo de limpeza, as chaves repetidas, os valores inválidos ou outros tipos de erros são eliminados.

Na Figura 4 podemos ver o caminho trilhado pelos dados até serem armazenados no DW para então podem ser acessados por aplicativos de BI.

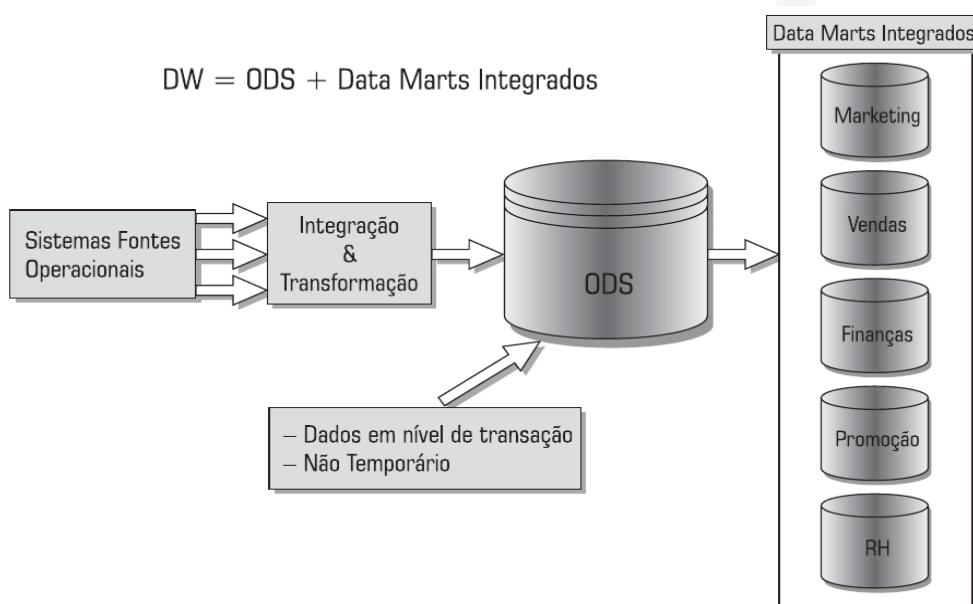
Figura 4 – Ambiente de *Data Warehouse*.



Fonte: COLAÇO, 2004, p. 19.

Para Kimball apud Colaço (2004), as soluções de DW podem ter outra área intermediária de armazenagem de dados, além da *Staging Area*, denominada *Operational Data Store* (ODS), que fica posicionada logicamente após o processo de ETL, como apresentado na Figura 5. Nesta área, se encontram dados parcialmente estruturados e analíticos, organizados por área de negócio, os dados são uma cópia do processo de ETL. A área de dados ODS, pode ser compreendida como um cadastro consolidador de informações, porém mantendo a granularidade e estrutura não dimensional oriundas dos sistemas de extração (BARBIERI, 2011).

Figura 5 – Representação gráfica de um DW com ODS.



Fonte: KIMBALL apud COLAÇO, 2004, p. 20.

Os dados contidos em um ODS são mantidos como uma cópia de um curto intervalo de tempo dos dados operacionais, podendo então servir para recuperação de cargas defeituosas e em alguns casos para dar suporte a operações em tempo real, como no caso de aplicativos *Customer Relationship Management* (CRM). O *Data Warehouse*, não está preparado para atender a demanda de sistemas que necessitem de dados em tempo real ou quase real (KIMBALL; ROSS, 2002). Por questão de economia de espaço, alguns projetos são implementados sem o ODS, o que não acarreta problemas graves, pois cargas problemáticas podem ser refeitas com uma nova extração, porém isso demanda mais tempo (COLAÇO, 2004).

Não basta apenas a implementação desta arquitetura para o correto armazenamento das informações em um *Data Warehouse*. Para que isso seja possível, é necessário um esforço dos administradores de banco de dados, já que a modelagem dos dados de um DW é diferente da modelagem do ambiente operacional, este assunto será abordado no capítulo a seguir.

2.5.3 Modelagem dimensional

Na década de 70 surgiu o conceito de Diagrama Entidade e Relacionamento (DER). Este foi o responsável pela significativa melhora na performance das bases de dados, pois seu intuito é eliminar em sua totalidade a redundância de dados, com isso garantindo que inclusões, alterações ou remoções de dados sejam feitas em um único local (COLAÇO, 2004).

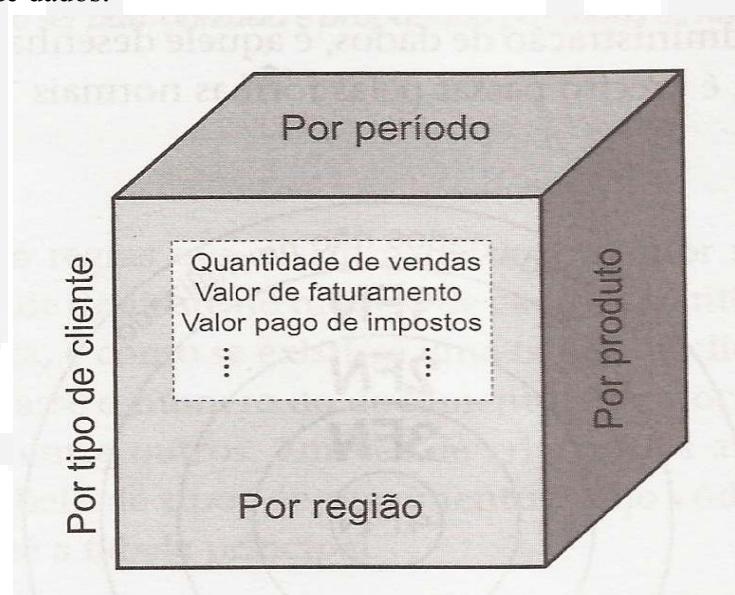
Este tipo de modelagem é eficiente para os registros de transações nos SITs, porém, em consultas gerenciais, o modelo perde sua eficiência, devido à complexidade das consultas, e também da alta taxa de atualização dos dados, fazendo com que os dados históricos da organização sejam perdidos. A realização de consultas gerenciais, que são geralmente complexas devido ao grande número de conexões entre tabelas, são impraticáveis em um banco de dados construído a partir de um modelo Entidade e Relacionamento (ER). Para diminuir esta complexidade e aumentar o desempenho das consultas, surgem os modelos dimensionais.

A modelagem dimensional para Barbieri (2011) é a simples alteração na ordem de distribuição de campos por entre as tabelas do modelo ER, modificando a estrutura para que

se torne mais voltada a muitos pontos de entrada, que são as dimensões, e menos para os dados mensuráveis, que são os fatos. Isso significa que os dados estão organizados em um formato diferente do proposto pelo modelo ER, assemelhando-se a uma estrela, onde no centro encontra-se a tabela fato e ao seu redor, ligados a ela, estão as tabelas de dimensões com informações que facilitam a visualização dos dados contidos na tabela fato.

O resultado da modelagem dimensional pode ser associada a um cubo, onde dentro do cubo se encontram os dados mensuráveis da organização, como quantidade de venda, valor de faturamento, entre muitas outras medidas. Já nas faces do cubo estão localizadas as formas que os dados podem ser analisados, como por cliente, por período, por produto, entre outras, como mostra a Figura 6. Para Barbieri (2011), o conceito de cubo representa o conjunto de dados de dimensões e de fatos, extraídos dos sistemas, com o objetivo de atender necessidades específicas.

Figura 6 – Cubo de dados.



Fonte: LEME, 2004, p. 176.

2.5.3.1 Tabela Fato

Para Barbieri (2011) nas tabelas fato concentram-se os dados de interesse, passíveis de manipulação numérica e estatística da organização, cada linha nesta tabela representa um fato. Este pode, ou não, estar agrupado, dependendo do nível de granularidade da tabela fato. Em

suas linhas estão registradas uma ou mais medidas numéricas. Geralmente estas tabelas possuem como chave primária uma combinação entre as chaves de todas as tabelas de dimensões e possuem uma quantidade de linhas muito superior a estas tabelas. Possuem uma quantidade inferior de colunas, em comparação com as dimensões, porém ocupam mais de 90% do espaço de um *Data Warehouse* (LEME, 2004).

Segundo Kimball (2002) a “tabela fato”, é a primeira tabela em um modelo dimensional, onde as medidas de desempenho do negócio são armazenadas. Nas tabelas fato, apenas métricas numéricas devem ser armazenadas, sendo que informações textuais necessárias devem ser armazenadas nas tabelas de dimensões.

Os fatos armazenados nesta tabela podem ser de três tipos: aditivos, onde os fatos armazenados geram somas coerentes em todas as dimensões, como por exemplo a quantidade vendida, semi-aditivas, são os fatos que geram medidas coerentes em algumas dimensões da tabela fato, como, por exemplo, uma temperatura, onde não faz sentido sua soma e sim sua média, e por fim os fatos não-aditivos, que são as métricas em que sua agregação não fazem sentido em nenhuma das dimensões, como, por exemplo, um campo onde é armazenado o total de vendas até o momento da extração, este tipo de fato é utilizado para contagens para resumo de registros e não pode ser somado ao longo do tempo.

2.5.3.2 Tabela Dimensão

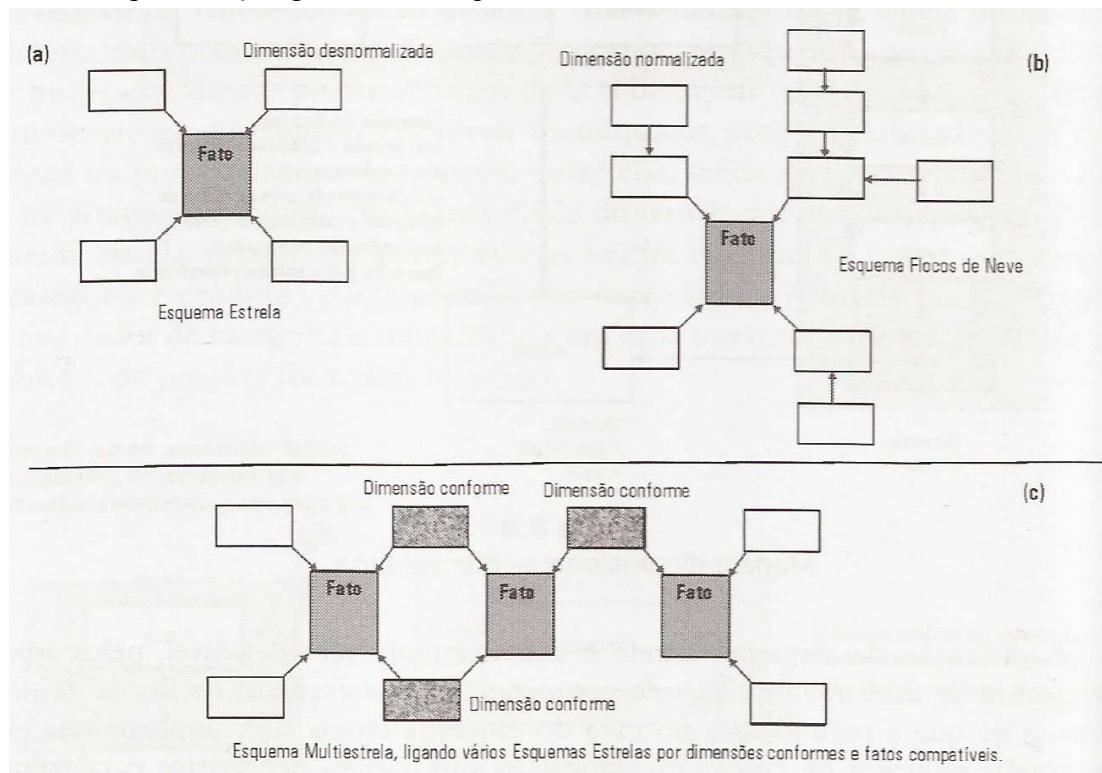
Na órbita de todas as tabelas fato existem algumas tabelas, as quais são denominadas tabelas dimensões. Por meio delas são feitos os filtros e agrupamentos dos dados. Para Barbieri (2011), estas são as tabelas que possuem as chaves de entrada do modelo, como tempo, fornecedores, produtos, entre outras. Essas tabelas possuem uma relação de 1:N com a tabela fato, ou seja, cada linha da tabela de dimensão pode estar relacionada com várias linhas na tabela fato. Uma dimensão possui um número significativamente inferior de linhas em relação a tabela fato e possui as informações descritivas relacionadas a dimensão. Sua chave primária deve ser única, e sempre diferente da sua chave no sistema de origem, este conceito é chamado de chave artificial.

Existem duas formas de criar uma tabela dimensão, que são o *Star Schema* ou esquema estrela e o *Snowflake Schema* ou esquema flocos de neve, onde o último prega a normalização das informações contidas nas tabelas de dimensão e o outro não. Para Barbieri (2011) a utilização do esquema estrela é altamente recomendável, pois o ganho de desempenho deste esquema é muito superior, devido à quantidade de junções necessárias para obter informações de uma dimensão modelada no esquema flocos de neve. Este, por sua vez, leva vantagem na economia de espaço gerada pela normalização, porém, a quantidade de espaço ocupada pelas tabelas de dimensão é muito inferior a ocupada pelas tabelas fato, cujo espaço economizado será desprezível se comparado com o ganho de desempenho gerado pelas consultas sem junções realizados pelo esquema estrela.

Dentro de um *Data Warehouse* podem ser armazenadas várias tabelas fato e suas respectivas dimensões que podem ser compartilhadas entre as tabelas fatos, para que isso seja possível, deve haver uma compatibilidade entre as dimensões e as tabelas fato envolvidas devem possuir uma granularidade compatível. No interior de um DW se encontram muitos cubos de dados em formato de estrela. Quando as dimensões começam a ser reutilizadas em novos fatos temos um novo esquema chamado multiestrela (BARBIERI, 2011).

A Figura 7, a divisão ‘a’ apresenta como as tabelas dimensões desnormalizadas, no esquema estrela, relacionam-se com a tabela fato, já na divisão ‘b’ está representada como as dimensões conectam-se com suas tabelas auxiliares, para atender a normalização exigido pelo esquema flocos de neve, na parte ‘c’ apresenta-se uma representação de como as tabelas fato podem compartilhar dimensões compatíveis com outras tabelas fato, formando o esquema multiestrela.

Figura 7 – Representação gráfica dos esquemas estrela, flocos de neve e multiestrela.



Fonte: BARBIERI, 2011, p.170.

Durante a evolução natural dos dados da organização pode exigir que alguns dados sejam atualizadas nas tabelas de dimensões, com por exemplo, clientes podem alterar seu estado civil, fornecedores podem alterar seu endereço, entre outros casos possíveis. Isso exige que uma estratégia para controle das atualizações dos dados seja adotada, existem três maneiras disponíveis na literatura, que são classificados por tipo:

- **Tipo 1:** A atualização dos campos desta tabela é um processo mais simples, porém os dados históricos são perdidos, assim como nos sistemas de informação transacional. Esta abordagem é geralmente utilizada quando a informação não possui um fator relevante ao processo de negócio;
- **Tipo 2:** Criação de um novo registro na tabela de dimensão. Com esta abordagem, se mantêm o histórico dos dados, porém é necessário a inclusão de campos para controle da validade do registro na tabela dimensão, quando um novo registro é inserido na tabela fato, este se relaciona apenas com o registro válido no momento da inclusão;
- **Tipo 3:** Criação de uma nova coluna na tabela dimensão, onde é registrado o último valor válido, antes da atualização, mantendo assim a informação atual e a informação anterior.

2.5.3.3 Granularidade

Granularidade é um termo utilizado para definir o nível de detalhamento dos dados armazenados nas tabelas fato de um *Data Warehouse*, ou seja, quanto maior for o nível de granularidade, menor será o detalhamento da informação, e quanto menos detalhe maior será o nível da granularidade. Nos sistemas transacionais, por exemplo, o nível de granularidade é o menor possível, já que os dados se encontram no mais puro detalhe. A granularidade é um dos pontos-chave no projeto de um *Data Warehouse*, pois este influí diretamente na quantidade de dados que serão armazenadas em disco e também no tipo de consulta que poderão ser realizadas (LEME, 2004).

2.5.3.4 Hierarquia dimensional

Hierarquias são formadas através do relacionamento entre atributos de uma mesma dimensão, estes são responsáveis diretamente pela possibilidade de realização das operações OLAP, mais especificamente das funções que envolvem *Drilling*, criando níveis de agregação entre alguns atributos de uma dimensão e desta maneira influenciando a granularidade dos dados (COLAÇO, 2004). Para exemplificar a utilização de hierarquias pode-se citar a mais simples delas, na dimensão que representa o tempo, claramente existe uma relação hierárquica entre os atributos ano, semestre, mês e dia, onde o maior nível de agregação é o ano e o menor é o dia.

2.5.4 Passos para modelagem dimensional

Segundo Kimball e Ross (2002) o processo de modelagem dimensional possui quatro passos, que são:

- **Escolha do processo de negócio a ser modelado:** um processo é uma atividade da organização, que normalmente é suportada por um SIT, como faturamento, compras, pedidos, entre outros. A maneira mais eficiente para escolha dos processos é através de entrevistas com os usuários, concentrando-se em processos de negócios, em vez de áreas de negócio;
- **Definição do nível de granularidade dos dados:** definir o nível de granularidade é especificar exatamente o que uma linha na tabela de fatos representa individualmente. Este representa o nível de detalhe associado com as medidas da tabela de fatos. Ele fornece a resposta para a pergunta: “Como você descreve uma única linha na tabela de fato?”;
- **Escolha as dimensões que se aplicam a cada linha da tabela fato:** atributos que são dimensões normalmente são as respostas para o questionamento “Como são descritos os dados que resultam do processo de negócio?”. Pode-se citar como exemplos de dimensões: data, produto, cliente, tipo de transação, entre outras;
- **Identificar os fatos numéricas que serão registrados na tabela fato:** atributos que estão entre os fatos são respostas para a pergunta “O que estamos medindo?”, todos os fatos devem ser compatíveis com o nível de granularidade definido no segundo passo, como exemplo de fato temos: quantidade vendida, valor de custo, valor da venda, entre outros.

2.6 ETL

Antes de os dados serem armazenados no *Data Warehouse*, eles devem passar por um conjunto de processos chamados *Extract, Transform e Load* (ETL). Para Colaço (2004), durante o projeto de um DW, os processos de ETL consomem mais de 70% do tempo de desenvolvimento, devido ao alto grau de customização exigido por estes processos, pois, cada empresa possui uma estrutura de dados específica em suas fontes de dados e também à falta de conhecimento e documentação destas. Nas organizações, em alguns casos, os dados estão armazenados fora dos sistemas de informação, como legislações, dados estratégicos armazenados em planilhas eletrônicas, entre outras situações, o processo de ETL também é

responsável por integrar essas informações com os dados extraídos dos sistemas da organização.

Segundo Kimball e Caserta (2004) o processo de ETL é a base do *Data Warehouse*, pois este processo, quando devidamente projetado, é responsável pela extração dos dados nos sistemas de origem, imposição de padrões de qualidade e consistência e a oferta dos dados no formato próprio para os softwares com o objetivo de auxílio no processo de tomada de decisão.

Geralmente a equipe de desenvolvimento do projeto tem uma escolha a fazer neste ponto, que é a escolha, no mercado, de uma ferramenta pronta para este fim ou o desenvolvimento interno destas rotinas, seja com a utilização de scripts³ de banco de dados ou o desenvolvimento de uma ferramenta completa.

O processo de ETL é dividido em fases, onde cada uma executa uma tarefa específica. As fases são:

- **Extração:** primeira fase do processo, onde os dados são extraídos dos sistemas fonte e armazenados em uma área temporária. Alguns fatores devem ser observados neste processo, como: frequência da extração, os critérios de seleção dos dados a serem extraídos e o melhor momento para a extração;
- **Transformação:** segunda fase do processo, fase esta que demanda maior empenho e tempo da equipe de desenvolvimento. Nesta etapa, são aplicadas as regras de negócio, ajustes de granularidade, regras de agregações, integração e limpeza, entre outras. A integração consiste na consolidação dos dados de diversas origens, que geralmente envolve diferentes codificações. Os dados devem ser perfeitamente integrados para que ao serem armazenados assumam uma única convenção. A limpeza é a rejeição de valores inválidos, chaves repetidas ou registros com outros tipos de erro;
- **Carga:** Fase final do processo de ETL, onde os dados são transferidos da área temporária utilizada pelo processo para o ambiente de produção.

³ Script é um texto com instruções que deve ser seguidas, seja por um sistema computacional ou por pessoas.

3 METODOLOGIA

Para o desenvolvimento do presente trabalho, a definição da metodologia se faz necessário. Segundo Gerhardt e Silveira (2009, p.11) pode-se compreender metodologia como “o estudo do método, ou seja, é o corpo de regras e procedimentos estabelecidos para realizar uma pesquisa”.

A metodologia tem como atividade principal a pesquisa, que no presente trabalho pode ser caracterizada como de natureza exploratória. Para Wainer (2007, p. 29), uma pesquisa deve ser caracterizada como exploratória quando “além de descrever o fenômeno, faz propostas para novas teorias, ou novas observações, novas métricas para medir o fenômeno.”

Segundo Gerhardt e Silveira (2009, p. 35) a pesquisa exploratória tem “como objetivo proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou a construir hipóteses.”

Quanto à abordagem da pesquisa utilizada para o desenvolvimento do presente trabalho, esta pode ser classificada como qualitativa. Segundo Wainer (2007, p. 5), a pesquisa qualitativa “baseia-se na observação cuidadosa do ambiente onde o sistema será utilizado, do entendimento das várias perspectivas dos usuários ou potenciais usuários do sistema”. Segundo Gerhardt e Silveira (2009, p. 31) “a pesquisa qualitativa não se preocupa com a representação numérica, mas, sim, com o aprofundamento do comportamento de um grupo social, de uma organização, etc”, ou seja, esta abordagem de pesquisa visa o entendimento do ambiente pesquisado e não de sua mensuração.

Ainda segundo Wainer (2009, p. 28) “de um modo geral, métodos qualitativos em ciência da computação são métodos que se caracterizam por ser um estudo aprofundado de um sistema no ambiente onde ele está sendo usado, ou, em alguns casos, onde se espera que o sistema seja usado.”

Sendo esta uma pesquisa muito específica, podemos reconhecer que ela adota a forma de um estudo de caso, o que vem de acordo com Wainer (2009, p. 31) o qual afirma que, as “áreas da computação que estão interessadas em processos, decisões e opiniões optam por um estilo de estudo de caso.”

Segundo Ventura (2007, p.2) o estudo de caso “visa à investigação de um caso específico, bem delimitado, contextualizado em tempo e lugar para que se possa realizar uma busca circunstanciada de informações”. O estudo de caso caracteriza-se, segundo Wainer (2009), pela forma que o pesquisador interage com os sujeitos pesquisados, que geralmente é de forma semiformal, utilizando entrevistas e conversas programadas, e normalmente tem acesso a documentos, dados e outros materiais formais da organização, tendo como objetivo descobrir as práticas formais desta e os valores, opiniões e atitudes dos sujeitos pesquisados.

Para Yin (2010, p.24) “o método de estudo de caso permite que os investigadores retenham as características globais e significativas dos eventos da vida real, tais como, comportamento dos pequenos grupos, os processos organizacionais e administrativos”, entre outros. Este método permite ao investigador a visualização do cenário pesquisado como um todo, e não apenas da perspectiva dos entrevistados, podendo, assim, propor melhorias globais nos processos.

Após a caracterização do presente trabalho quanto aos aspectos metodológicos, é necessário definir os principais passos para realização do estudo. Este trabalho se caracteriza como um Estudo de Caso, com o objetivo de compreender o cenário que envolve o processo de tomada de decisão e acompanhamentos de indicadores de desempenho da empresa Couros Bom Retiro LTDA.

Para dar suporte ao assunto abordado no estudo de caso, outras fontes se fazem necessárias, sendo que para o presente trabalho, foram utilizadas pesquisas bibliográficas e de campo.

Segundo Fonseca apud Gerhardt e Silveira (2009, p.36) pesquisa bibliográfica se caracteriza como,

[...] o levantamento de referências teóricas já analisadas, e publicadas por meios escritos e eletrônicos, como livros, artigos científicos, páginas de web sites. Qualquer trabalho científico inicia-se com uma pesquisa bibliográfica, que permite ao pesquisador conhecer o que já se estudou sobre o assunto.

Ainda segundo Fonseca apud Gerhardt e Silveira (2009, p.36) “a pesquisa de campo caracteriza-se pelas investigações em que, além da pesquisa bibliográfica e/ou documental, se realiza coleta de dados junto a pessoas, com o recurso de diferentes tipos de pesquisa”.

O primeiro passo do Estudo de Caso será o levantamento das necessidades de tomada de decisão, que será realizado através de uma pesquisa qualitativa com os tomadores de decisão da organização e os responsáveis pelo acompanhamento dos indicadores do setor selecionado para implantação.

Após ter definido as necessidades de tomada de decisão, com base na análise das respostas da pesquisa qualitativa, as informações sobre os processos serão sintetizadas e se tornarão a base para o projeto do *Data Warehouse* e sua posterior implementação.

Para a apresentação dos dados para os administradores se faz necessária a utilização de uma ferramente de BI, que será selecionada nesta fase do projeto. A escolha da ferramenta será baseada na política de software da organização, que basicamente consiste na utilização de softwares livres.

A próxima fase, envolverá o projeto e implementação das rotinas de ETL, que serão responsáveis pela população de dados nos cubos de dados contidos no DW, com a ferramenta de BI já definida, estes processos devem contemplar as suas possíveis particularidades.

Após os testes das rotinas de ETL e o DW estarem com uma certa quantidade de informações, a ferramenta de BI será disponibilizada aos usuários para análise, onde estes avaliarão a qualidade e confiabilidade das informações obtidas por meio da utilização da ferramenta.

Para que seja realizada a avaliação do trabalho realizado, uma nova pesquisa qualitativa será realizada com os usuários, coletando assim as suas opiniões e sugestões, as quais serão registradas e utilizadas para eventuais melhorias no *Data Warehouse*, bem como analisadas para a redação das conclusões do presente trabalho.

4 TRABALHO PROPOSTO

4.1 Visão geral

Este capítulo apresenta o processo de modelagem e criação do *Data Warehouse* assim como a apresentação das ferramentas utilizadas para a preparação do ambiente para os usuários.

4.1.1 Caracterização da empresa

Fundado em 1989, o GRUPO BOM RETIRO tem história marcada pela simplicidade e seriedade com que trata seus negócios transformando a relação com seus clientes em pessoal, fiel e duradoura.

No início das atividades, terceirizava-se o beneficiamento de peles em outros curtumes. Em 1994, foi alugado um prédio em Taquari onde se produzia 400 peles por dia. Atualmente, o Grupo conta com 11 unidades produtivas no Brasil e uma na Argentina, somando uma área construída superior a 170.000 m², com a capacidade de industrializar 15.000 couros por dia de salgado a acabado e gera aproximadamente 3.500 empregos diretos.

O grupo é formado por cinco empresas, sendo elas, Medicouros indústria e comércio de couros, Couros Bom Retiro, Future Indústria de Couros LTDA, Curtume CBR e Bom Retiro Transporte e Logística.

O grupo tem como missão oferecer soluções em couro com qualidade, segurança e responsabilidade, visando resultados aos clientes, funcionários, fornecedores, comunidade e acionistas.

4.1.2 Justificativa

A justificativa do presente trabalho se dá pelo aumento da dinamicidade do mundo dos negócios, forçando assim as organizações a serem mais ágeis em seus processos internos de tomada de decisão, também pelo interesse dos gestores da organização em *softwares* que lhes proporcionem acesso aos dados da empresa de uma forma mais ágil e dinâmica, sem a dependência da equipe de TI.

4.1.3 Áreas de negócio

Para a realização do presente trabalho, entre as diversas áreas de negócio, foram selecionadas duas para a aplicação do presente estudo de caso. Para a seleção utilizou-se como critério balizador a experiência do autor nas áreas disponíveis e optou-se pelas áreas: compras e comercial.

4.2 Processos modelados

O primeiro passo para a modelagem dos processos foi a descoberta dos mesmos. Este processo de descoberta foi realizado através de uma pesquisa com os usuários responsáveis pelas decisões das áreas selecionadas da organização.

4.2.1 Levantamento de requisitos

Para que as informações fossem modeladas de maneira correta foi elaborado um questionário, que pode ser encontrado no Apêndice A deste documento, cujo objetivo era obter os processos específicos mais relevantes dentro da sua esfera de trabalho, onde as informações utilizadas para a tomada de decisão eram encontradas.

Os colaboradores selecionados para responder o questionário estão integrados ao corpo de colaboradores da organização há mais de 15 anos, sendo então: um Gerente Administrativo de 41 à 43 anos que possui de 15 à 20 anos de experiência e uma Gerente de Compras de 30 à 34 anos que possui de 8 a 12 anos de experiência.

4.2.2 Grupos de informações identificados

Com a análise das respostas dos questionários nas áreas selecionadas encontrou-se a necessidade de modelar os processos. Na área de compras são: compra, consumo de produtos, ajuste de estoque e níveis de estoque, já na área comercial os processos são faturas de exportação e comissão dos agentes comerciais.

As principais regras de negócio envolvidas nos processos são:

- **Compras:** As compras são realizadas com base na programação de produção que são informadas pelas unidades, com posse destas informações são geradas as ordens de compra para os fornecedores com os quais os preços já foram negociados;
- **Consumo:** O processo de industrialização da matéria-prima é baseado em fórmulas de produção. Através destas e da quantidade de matéria-prima presente em cada lote, o sistema calcula a quantidade de material e realiza as baixas no estoque, as quais são consideradas baixas por lote e são os consumos reais de insumos;
- **Ajustes:** No final de cada mês é realizado o balanço de almoxarifado. Quando for encontrada uma diferença de estoque onde a quantidade física é maior que a quantidade que o sistema está registrando, este processo de ajuste é realizado. Nos casos em que a situação é inversa, ou seja, o sistema registra maior quantidade do que existe fisicamente, o ajuste é feito através de uma requisição que gera no sistema um consumo de material, que é tratado como consumo manual;
- **Níveis de estoque:** Registro dos níveis de estoque diários para cada item de estoque;
- **Faturas:** Documento responsável pela venda de exportação, podendo assumir duas formas, sendo estas comerciais ou de amostras. A forma comercial é responsável pelo real faturamento com exportação da organização. As faturas de amostra, embora possuam valor, não são efetivamente pagas, visto que nestes casos o material é remetido para confecção de amostras;
- **Comissão:** Os clientes entram em contato com a empresa através dos representantes comerciais, estes recebem uma porcentagem do valor faturado dependendo da relevância do cliente e do próprio agente.

4.3 Modelagem dimensional

Para o processo de modelagem dimensional foram utilizados os quatro passos sugeridos por Kimball e Ross (2002), que são: a escolha do processo a ser modelado, definição da granularidade, escolha das dimensões e a identificação dos fatos.

Para auxiliar nos passos dois e três da modelagem foi utilizada a técnica do 5W2H, porém não há a necessidade da utilização dos dois H, que são o *how* que indica como será feito e o *how much* indicando quanto vai custar para ser feito, uma vez que o *how much* é a própria métrica escolhida. Os cinco W são:

- **What:** consiste no que será registrado como fato;
- **Why:** porque será feito, aqui é analisada a real necessidade do item anterior, eventualmente outra estratégia pode atender a necessidade levantada;
- **Where:** onde será feito, este W consiste em informações sobre a localização;
- **When:** representa a informação de quando será feito, utilizado para obter informações sobre as dimensões de tempo;
- **Who:** indica por quem será feito, possui informações do responsável pelo fato.

Durante a terceira fase do processo de modelagem percebeu-se que algumas dimensões poderiam ser utilizadas em mais de um cubo de informações, criando assim um esquema multiestrela. Desta forma gerando uma economia de espaço em disco e no processo de ETL.

4.3.1 Informações compartilhadas

As seguintes dimensões são compartilhadas por dois ou mais cubos de dados dentro do *Data Warehouse*:

- **Dimensão Tempo (DimTempo):** Esta dimensão contém a informação do tempo em que os fatos ocorreram. A dimensão possui um registro por dia, sendo inserida conforme necessário. Como o fato estoque é registrado diariamente isso evita que existam lacunas na dimensão. Ela possui as informações de ano, semestre, trimestre, mês, dia e dia da semana;

- **Dimensão Produto (DimProduto):** Esta dimensão possui as informações dos produtos consumidos e comprados na organização. Possui os seguintes dados: código, descrição do produto, estrutural, unidade de medida, categoria e subcategoria;
- **Dimensão Filial (DimFilial):** Esta dimensão possui as informações da filial onde o fato ocorreu. Possui as seguintes informações: código, nome fantasia, descrição completa da filial, Cadastro Nacional de Pessoa Jurídica (CNPJ), cidade e estado em que está situada a filial.

Na Listagem 1 encontra-se o código escrito em *Structured Query Language* (SQL) para a criação das dimensões de tempo, produtos e filiais no banco de dados.

Listagem 1 – Código SQL para criação das dimensões tempo, produto e filial.

```
1  -- -----
2  -- Tabela DimTempo
3  --
4  CREATE TABLE DimTempo (
5      idDimTempo serial,
6      Data        DATE,
7      Dia         INT,
8      Mes         INT,
9      Ano         INT,
10     DiaSemana  VARCHAR(15),
11     MesExtenso VARCHAR(15),
12     Semestre   VARCHAR(15),
13     Trimestre  VARCHAR(15),
14     CONSTRAINT PK_dimTempo_id PRIMARY KEY (idDimTempo));
15
16 --
17 -- Tabela DimProduto
18 --
19 CREATE TABLE DimProduto (
20     idDimProduto serial,
21    Codigo        VARCHAR(6),
22     Descricao    VARCHAR(45),
23     Estrutural   VARCHAR(25),
24     NCM          VARCHAR(15),
25     Unidade      VARCHAR(3),
26     CategoriaN1 VARCHAR(35),
27     CategoriaN2 VARCHAR(35),
28     CategoriaN3 VARCHAR(35),
29     CONSTRAINT PK_dimProduto_id PRIMARY KEY (idDimProduto));
30
31 --
32 -- Tabela DimFilial
33 --
34 CREATE TABLE DimFilial (
35     idDimFilial serial,
36    Codigo        INT,
37     Fantasia     VARCHAR(25),
38     Descricao    VARCHAR(40),
39     CNPJ         CHAR(18),
40     Cidade       VARCHAR(32),
```

```

41 Estado      VARCHAR(30),
42 CONSTRAINT PK_dimFilial_id PRIMARY KEY (idDimFilial));

```

Fonte: Elaborado pelo autor.

Para criar os registros da dimensão tempo foi desenvolvida uma função na linguagem pl/pgsql⁴, basicamente esta linguagem permite que sejam agrupados códigos escritos em SQL em uma função, e esta pode ser chamada em uma consulta SQL ou separadamente. A função que está transcrita na Listagem 2 é responsável por transformar a data passada por parâmetro em um registro na dimensão tempo.

Listagem 2 – Código SQL da função PopulaTempo.

```

1  CREATE OR REPLACE FUNCTION PopulaTempo(dataIns date)
2  RETURNS date
3  AS $$
4  DECLARE
5      mes integer;
6      dias integer;
7      sem integer;
8      tri integer;
9      trimestre varchar[] := '{"1 trimestre", "2 trimestre",
10                           "3 trimestre", "4 trimestre"}';
11     semestre varchar[] := '{"1 semestre", "2 semestre"}';
12     diaSemana varchar[] := '{"Domingo", "Segunda-Feira",
13                           "Terca-Feira", "Quarta-Feita",
14                           "Quinta-feira", "Sexta-Feira",
15                           "Sabado"}';
16     Meses     varchar[] := '{"Janeiro", "Fevereiro", "Marco",
17                           "Abril", "Maio", "Junho", "Julho",
18                           "Agosto", "Setembro", "Outubro",
19                           "Novembro", "Dezembro"}';
20 BEGIN
21     IF NOT EXISTS(SELECT * FROM DimTempo WHERE data=dataIns) THEN
22         SELECT date_part('month', dataIns) INTO mes;
23         SELECT date_part('dow', dataIns) INTO dias;
24         SELECT trunc((mes-1)/3, 0)+1    INTO tri;
25         SELECT trunc((mes-1)/6, 0)+1    INTO sem;
26         INSERT INTO DimTempo VALUES (default, dataIns,
27                                         date_part('day', dataIns), mes,
28                                         date_part('year', dataIns),
29                                         diaSemana[dias+1],
30                                         Meses[mes], semestre[sem],
31                                         trimestre[tri]);
32     END IF;
33     return dataIns;
34 END;
35 $$ LANGUAGE plpgsql;

```

Fonte: Elaborado pelo autor.

A função apresentada na Listagem 2 basicamente decompõem a data passada por parâmetro, utilizando funções SQL, nas suas três informações, ou seja em ano, mês e dia, com base nestes três dados é calculado o semestre, trimestre e dia da semana, para posterior registro na dimensão.

⁴ Plpgsql é uma linhagem procedural para desenvolvimento de funções e gatilhos dentro do SGBD PostgreSQL.

Para a documentação das transformações realizadas para a formação das tabelas dimensão e fato de um DW utiliza-se uma matriz que registra os campos de origem e de destino de cada coluna das tabelas, assim como as transformações sofridas pelos dados antes da carga no DW. Esta matriz se denomina mapa de transformação. Na Tabela 1 encontra-se o mapa de transformação das dimensões compartilhadas.

Tabela 1 – Mapa de transformação das dimensões compartilhadas.

Banco de dados	Origem			Destino					S C D	Transformação
	Tabela Origem	Campo Origem	Tipo campo	Tabela Destino	Campo Destino	Tipo Campo	Tipo Tabela			
SQL	dfiliais	codi	Char(2)	dimfilial	codigo	Integer	Dimensão	2		
SQL	dfiliais	fant	Char(25)	dimfilial	fantasia	VARCHAR(25)	Dimensão	1		
SQL	dfiliais	desc	Char(40)	dimfilial	descricao	VARCHAR(40)	Dimensão	1		
SQL	dfiliais	cnpj	Char(18)	dimfilial	cnpj	Char(18)	Dimensão	1		
SQL	dsce045	nome	Char(32)	dimfilial	cidade	VARCHAR(32)	Dimensão	1		
SQL	dscep060	desc	Char(30)	dimfilial	estado	VARCHAR(40)	Dimensão	1		
SQL	dsce100	redu	Char(6)	dimproduto	codigo	Char(6)	Dimensão	2		
SQL	dsce100	desc	Char(35)	dimproduto	descricao	VARCHAR(45)	Dimensão	1		
SQL	dsce100	estrutural	Char(26)	dimproduto	estrutural	VARCHAR(25)	Dimensão	1		
SQL	dsce100	ncm	Char(14)	dimproduto	ncm	VARCHAR(15)	Dimensão	1		
SQL	dsce100	unid	Char(2)	dimproduto	unidade	VARCHAR(3)	Dimensão	1		
SQL	dscp067	nivel	Char(34)	dimproduto	categorian1	VARCHAR(35)	Dimensão	1		
SQL	dscp067	nivel	Char(34)	dimproduto	categorian2	VARCHAR(35)	Dimensão	1		

Fonte: Elaborado pelo autor.

A coluna *Slowly Changing Dimensions* (SCD) do mapa de transformação registra a estratégia de atualização da dimensão, onde o número 2 representa a criação de uma nova linha na dimensão caso o campo seja alterado e o número 1 representa a atualização do campo na tabela de dimensão.

4.3.2 Área de Compras

A área de compras é responsável pelas compras de insumos para produção, material para embalagem, material de segurança, entre outros tipos de materiais. A compra de matéria-prima é realizada por um setor especializado, não entrando no escopo do presente trabalho.

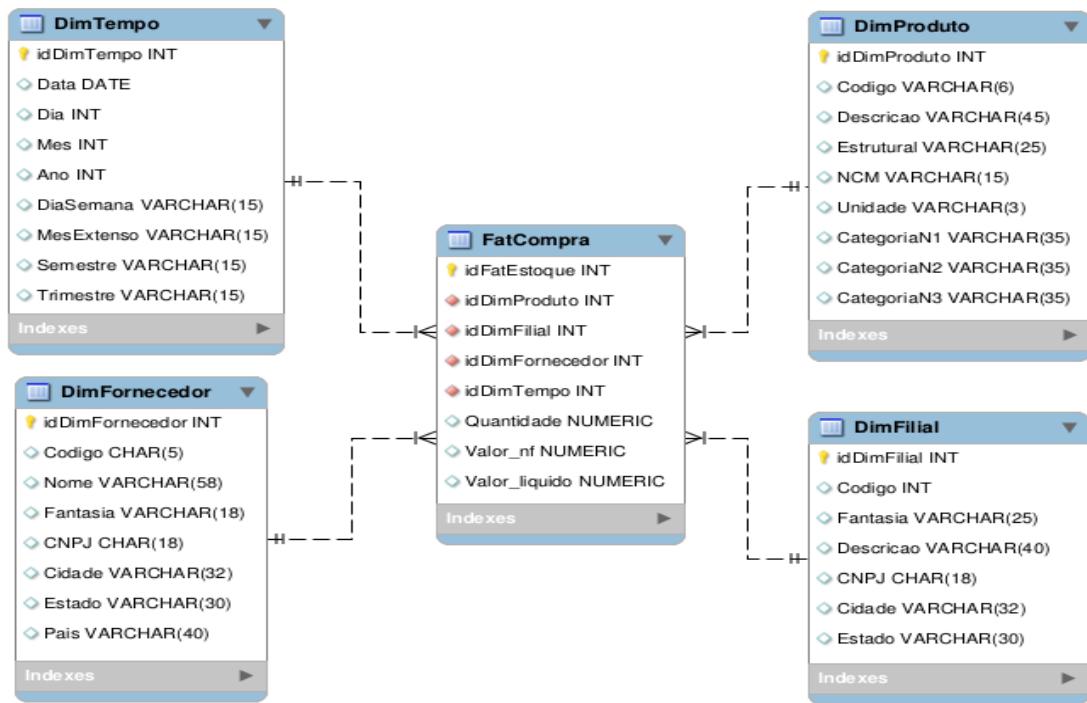
4.3.2.1 Cubo de Compras

O cubo de compra tem como objetivo o registro das compras realizadas pela organização junto a seus fornecedores, este fato possui as seguintes métricas: Quantidade adquirida, valor líquido e valor da nota fiscal. Os cinco W para o fato de compras são:

- **What:** Valores e quantidades relativas as compras de materiais;
- **Why:** Uma compra de insumos ocorre para reposição de estoques para produção. Como é uma constante, não é necessário armazenar esta informação;
- **Where:** Representa o local onde esta compra foi realizada, e está relacionada com a dimensão DimFilial;
- **When:** Representa o momento em que a compra foi realizada, e está relacionado com a dimensão DimTempo;
- **Who:** Representa o fornecedor do pedido de compra, e está relacionado com a dimensão DimFornecedor.

O cubo de compras possui quatro dimensões, sendo as três dimensões compartilhadas (DimTempo, DimFilial e DimProduto), detalhadas no item 4.3.1, e a dimensão fornecedor (DimFornecedor). Esta última possui as informações dos fornecedores da organização, contendo as seguintes informações: código, nome, nome fantasia, CNPJ, Cidade, Estado e País. Na Figura 8 encontra-se o diagrama da tabela fato com suas dimensões.

Figura 8 – Cubo de compras.



Fonte: Elaborado pelo autor.

Na Listagem 3 encontra-se o código SQL para a criação da tabela fato (FatCompra) e da dimensão fornecedor, que é específica deste cubo (DimFornecedor), no banco de dados.

Listagem 3 – Código SQL para criação do fato compra e a dimensão fornecedor.

```

1  -- -----
2  -- Table DimFornecedor
3  -- -----
4  CREATE TABLE DimFornecedor (
5      idDimFornecedor serial,
6      Codigo          char(5),
7      Nome            VARCHAR(58),
8      Fantasia        VARCHAR(18),
9      CNPJ            CHAR(18),
10     Cidade          VARCHAR(32),
11     Estado          VARCHAR(30),
12     Pais            VARCHAR(40),
13     PaisIngles     VARCHAR(40),
14     CONSTRAINT PK_dimFornecedor_id PRIMARY KEY (idDimFornecedor)
15 );
16
17 -- -----
18 -- Table FatCompra
19 -- -----
20 CREATE TABLE FatCompra (
21     idFatEstoque    serial,
22     idDimProduto    INT NOT NULL,
23     idDimFilial     INT NOT NULL,
24     idDimFornecedor INT NOT NULL,
25     idDimTempo      INT NOT NULL,
26     Quantidade      DECIMAL(15,2) DEFAULT 0,
27     Valor_liquido   DECIMAL(15,2) DEFAULT 0,

```

```
28     Valor_NF      DECIMAL(15,2) DEFAULT 0,  
29     CONSTRAINT PK_FatCompra_id PRIMARY KEY (idFatEstoque),  
30     CONSTRAINT FK_FatCompra_DimProduto      FOREIGN KEY  
31         (idDimProduto) REFERENCES DimProduto(idDimProduto),  
32     CONSTRAINT FK_FatCompra_DimFilial      FOREIGN KEY  
33         (idDimFilial) REFERENCES DimFilial(idDimFilial),  
34     CONSTRAINT FK_FatCompra_DimFornecedor FOREIGN KEY  
35         (idDimFornecedor) REFERENCES DimFornecedor(idDimFornecedor),  
36     CONSTRAINT FK_FatCompra_DimTempo      FOREIGN KEY  
37         (idDimTempo) REFERENCES DimTempo(idDimTempo)  
38 );
```

Fonte: Elaborado pelo autor.

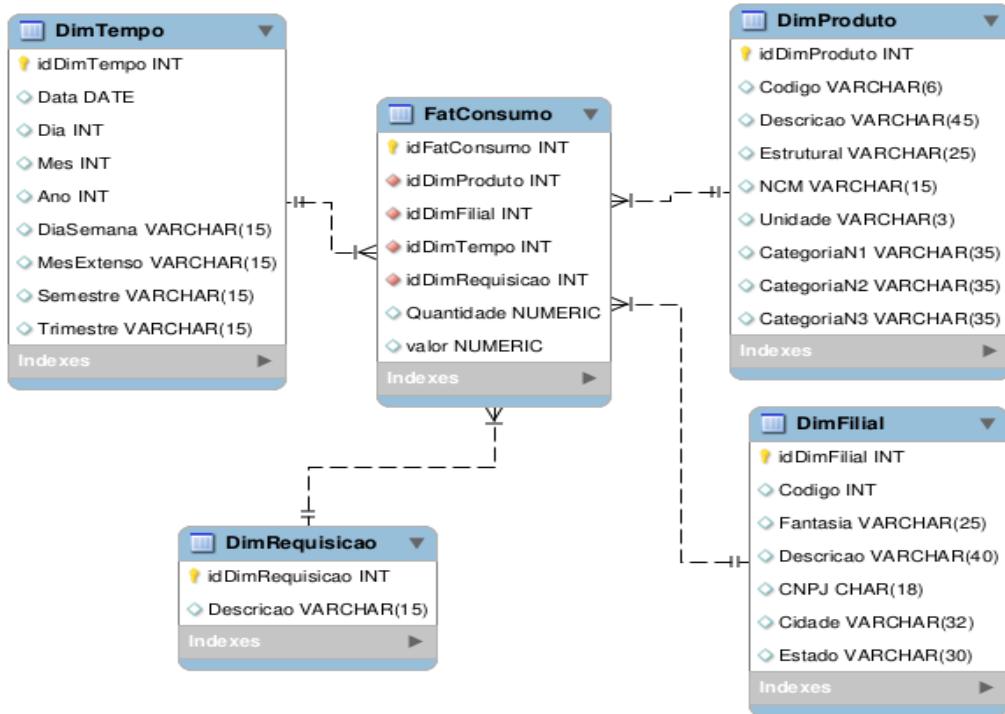
4.3.2.2 Cubo de Consumo

O cubo de consumo tem como objetivo o registro dos consumos de materiais pela organização. Este fato possui as métricas de valor consumido e quantidade consumida. Os cinco W para o fato de compras são:

- **What:** Valores e quantidades relativos aos consumos de materiais;
- **Why:** Um consumo ocorre por motivos diversos. Não é uma necessidade categorizar os motivos de um consumo;
- **Where:** Representa o local onde o consumo foi realizado, e está relacionado com a dimensão DimFilial;
- **When:** Representa o momento da realização do consumo, e está relacionado com a dimensão DimTempo;
- **Who:** Apesar do ERP armazenar o usuário que realizou o a baixa, esta informação é irrelevante para a realização de análises no BI.

O cubo de consumo possui quatro dimensões, sendo as três dimensões compartilhadas (DimTempo, DimFilial e DimProduto), detalhadas no item 4.3.1, e a dimensão requisição (DimRequisicao). Esta última possui a função de tipificar o consumo, contendo apenas a descrição da requisição. Na Figura 9 encontra-se o diagrama da tabela fato com suas dimensões.

Figura 9 – Cubo de consumo.



Fonte: Elaborado pelo autor.

Na Listagem 4 encontra-se o código SQL para a criação da tabela fato (FatConsumo) e da dimensão requisição, que é específica deste cubo (DimRequisicao), no banco de dados.

Listagem 4 – Código SQL para criação do fato compra e a dimensão requisição.

```

1  -- --
2  -- Table DimRequisicao
3  --
4  CREATE TABLE DimRequisicao (
5      idDimRequisicao serial,
6      Descrição      VARCHAR(15),
7      CONSTRAINT PK_dimRequisicao_id PRIMARY KEY (idDimRequisicao)
8  );
9
10 --
11 -- Table FatConsumo
12 --
13 CREATE TABLE FatConsumo (
14     idFatConsumo    serial,
15     idDimProduto    INT NOT NULL,
16     idDimFilial     INT NOT NULL,
17     idDimRequisicao INT NOT NULL,
18     idDimTempo      INT NOT NULL,
19     Quantidade      DECIMAL(15,2) DEFAULT 0,
20     valor           DECIMAL(15,2) DEFAULT 0,
21     CONSTRAINT PK_FatConsumo_id PRIMARY KEY (idFatConsumo),
22     CONSTRAINT FK_FatConsumo_DimProduto FOREIGN KEY
23         (idDimProduto) REFERENCES DimProduto(idDimProduto),
24     CONSTRAINT FK_FatConsumo_DimFilial FOREIGN KEY
25         (idDimFilial) REFERENCES DimFilial(idDimFilial),
26     CONSTRAINT FK_FatConsumo_DimRequisicao FOREIGN KEY
27         (idDimRequisicao) REFERENCES DimRequisicao(idDimRequisicao),

```

```
28  CONSTRAINT FK_FatConsumo_DimTempo      FOREIGN KEY
29      (idDimTempo) REFERENCES DimTempo(idDimTempo)
30 );
```

Fonte: Elaborado pelo autor.

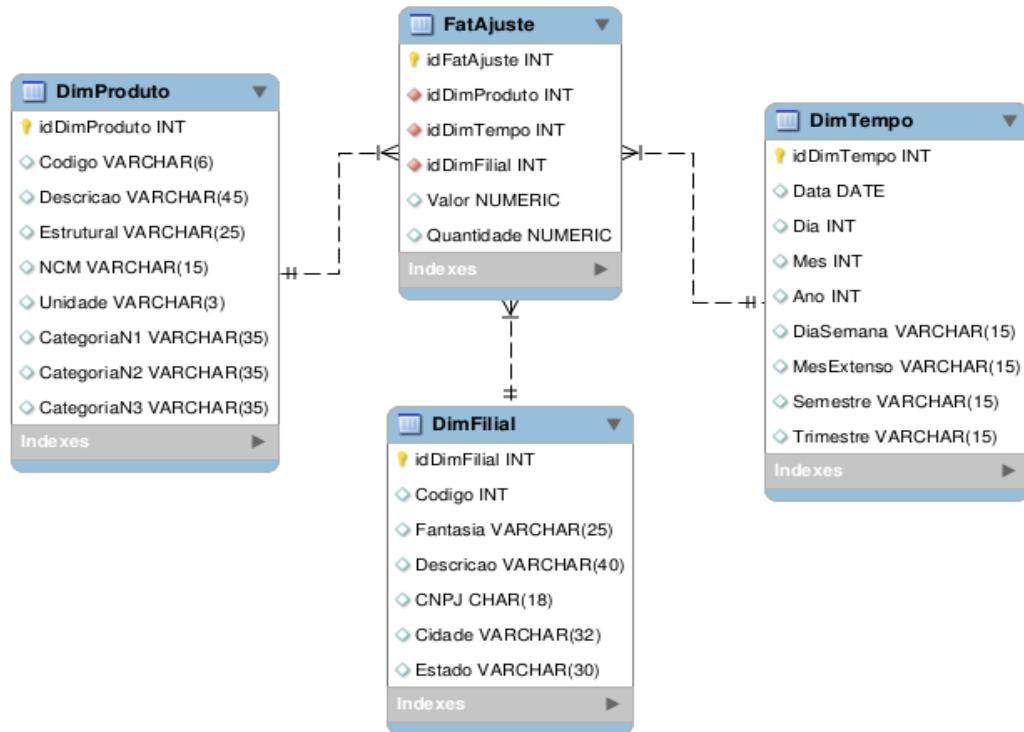
4.3.2.3 Cubo de Ajuste

O cubo de ajuste tem como objetivo o registro dos ajustes positivos de materiais, estes ajustes são necessários quando é encontrada uma divergência no balanço de estoque realizado mensalmente. Este fato possui as métricas de valor ajustado e quantidade ajustada. Os cinco W para o fato de compras são:

- **What:** Valores e quantidades relativos aos ajustes de materiais;
- **Why:** As causas de um ajuste são diversas. Não é necessidade do presente trabalho o mapeamento das mesmas;
- **Where:** Representa o local onde o ajuste foi realizado, e está relacionado com a dimensão DimFilial;
- **When:** Representa o momento em que foi realizado o ajuste, e está relacionado com a dimensão DimTempo;
- **Who:** Apesar do ERP armazenar o usuário que realizou o ajuste, esta informação é irrelevante para a realização de análises no BI.

O cubo de ajuste possui apenas as três dimensões compartilhadas (DimProduto, DimTempo e DimFilial), detalhadas no item 4.3.1. Na Figura 10 encontra-se o diagrama da tabela fato (FatAjuste) com suas dimensões.

Figura 10 – Cubo de Ajuste.



Fonte: Elaborado pelo autor.

Na Listagem 5 encontra-se o código SQL para a criação da tabela fato (FatAjuste) no banco de dados.

Listagem 5 – Código SQL para criação do fato ajuste.

```

1  -- --
2  -- Table FatAjuste
3  --
4  CREATE TABLE FatAjuste (
5      idFatAjuste    serial,
6      idDimProduto  INT NOT NULL,
7      idDimFilial   INT NOT NULL,
8      idDimTempo    INT NOT NULL,
9      Quantidade    DECIMAL(15,2) DEFAULT 0,
10     valor        DECIMAL(15,2) DEFAULT 0,
11     CONSTRAINT PK_FatAjuste_id PRIMARY KEY (idFatAjuste),
12     CONSTRAINT FK_FatAjuste_DimProduto FOREIGN KEY
13         (idDimProduto) REFERENCES DimProduto(idDimProduto),
14     CONSTRAINT FK_FatAjuste_DimFilial FOREIGN KEY
15         (idDimFilial) REFERENCES DimFilial(idDimFilial),
16     CONSTRAINT FK_FatAjuste_DimTempo FOREIGN KEY
17         (idDimTempo) REFERENCES DimTempo(idDimTempo)
18 );

```

Fonte: Elaborado pelo autor.

4.3.2.4 Cubo de Estoque

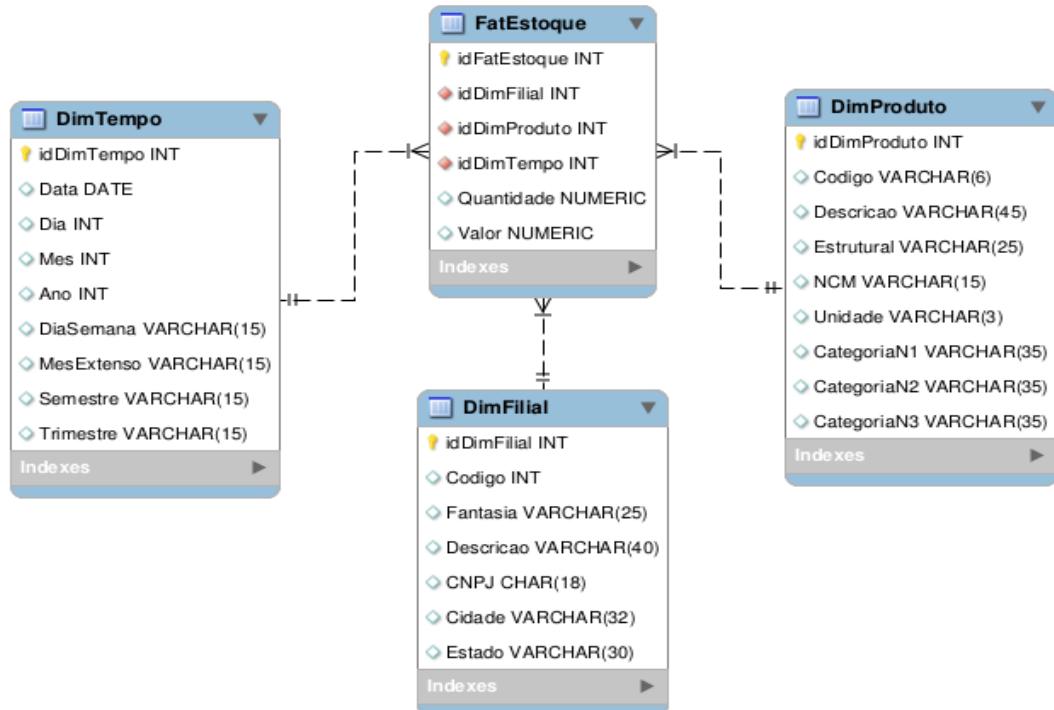
O cubo de estoque tem como objetivo o registro dos níveis de estoque dos materiais mantidos pela organização. Esta tabela fato contém uma linha por dia para cada produto movimentado pela empresa. Este fato possui as métricas de valor em estoque e quantidade em estoque. Os cinco W para o fato de compras são:

- **What:** Valores e quantidades relativos aos estoques de materiais;
- **Why:** Não se aplica;
- **Where:** Representa o local onde o estoque está alocado, e está relacionado à dimensão DimFilial;
- **When:** Representa o momento em que foi registrado os níveis de estoque está relacionado com a dimensão DimTempo;
- **Who:** Não se aplica.

Diferentemente de outros cubos que representam transações de negócios, e que possuem carga diferencial, ou seja, somente as transações ocorridas em um determinado período são carregadas para o *Data Warehouse*, o cubo de estoque possui carga total. Cargas totais, ou *snapshots* são realizadas sempre que precisamos ter uma “fotografia” ou monitoramento de um certo dado ao longo do tempo.

O cubo de estoque possui apenas as três dimensões compartilhadas (DimTempo, DimFilial e DimProduto), detalhadas no item 4.3.1. Na Figura 11 encontra-se o diagrama da tabela fato(FatEstoque) com suas dimensões.

Figura 11 – Cubo de estoque.



Fonte: Elaborado pelo autor.

Na Listagem 6 encontra-se o código SQL para a criação da tabela fato (FatEstoque) no banco de dados.

Listagem 6 – Código SQL para criação do fato estoque.

```

1  -- -----
2  -- Table FatEstoque
3  -- -----
4  CREATE TABLE FatEstoque (
5      idFatEstoque serial,
6      idDimFilial  INT NOT NULL,
7      idDimProduto INT NOT NULL,
8      idDimTempo   INT NOT NULL,
9      Quantidade   DECIMAL(15,2) DEFAULT 0,
10     Valor        DECIMAL(15,2) DEFAULT 0,
11     CONSTRAINT PK_FatEstoque_id PRIMARY KEY (idFatEstoque),
12     CONSTRAINT FK_FatEstoque_DimFilial FOREIGN KEY
13         (idDimFilial) REFERENCES DimFilial(idDimFilial),
14     CONSTRAINT FK_FatEstoque_DimTempo FOREIGN KEY
15         (idDimTempo) REFERENCES DimTempo(idDimTempo),
16     CONSTRAINT FK_FatEstoque_DimProduto FOREIGN KEY
17         (idDimProduto) REFERENCES DimProduto(idDimProduto)
18 );

```

Fonte: Elaborado pelo autor.

4.3.3 Área comercial

A área de comercial é responsável pelo recebimento dos pedidos de compra, feitos diretamente pelos clientes ou com intermédio de representantes, também chamados de agentes. Neste setor são tratados tanto os pedidos de exportação quanto os de mercado interno, para o presente trabalho serão abordados apenas os pedidos de exportação.

4.3.3.1 Cubo de Faturas

O cubo de faturas tem como objetivo o registro das faturas de exportação efetuadas pela organização junto a seus clientes e representantes, este fato possui as seguintes métricas: Valor faturado, valor da comissão em conta gráfica⁵. Os valores registrados neste cubo estão em dólares. Os cinco W para o fato de compras são:

- **What:** Valores da fatura e valor da comissão em conta gráfica;
- **Why:** Representa o motivo da fatura. Não se aplica;
- **Where:** Representa o local onde a fatura foi realizada, e está relacionado com a dimensão DimFilial;
- **When:** Representa o momento em que a fatura foi emitida, e está relacionado com a dimensão DimTempo.
- **Who:** Representa o *applicant* (responsável pelo pagamento), que está relacionado com a dimensão DimCliente.

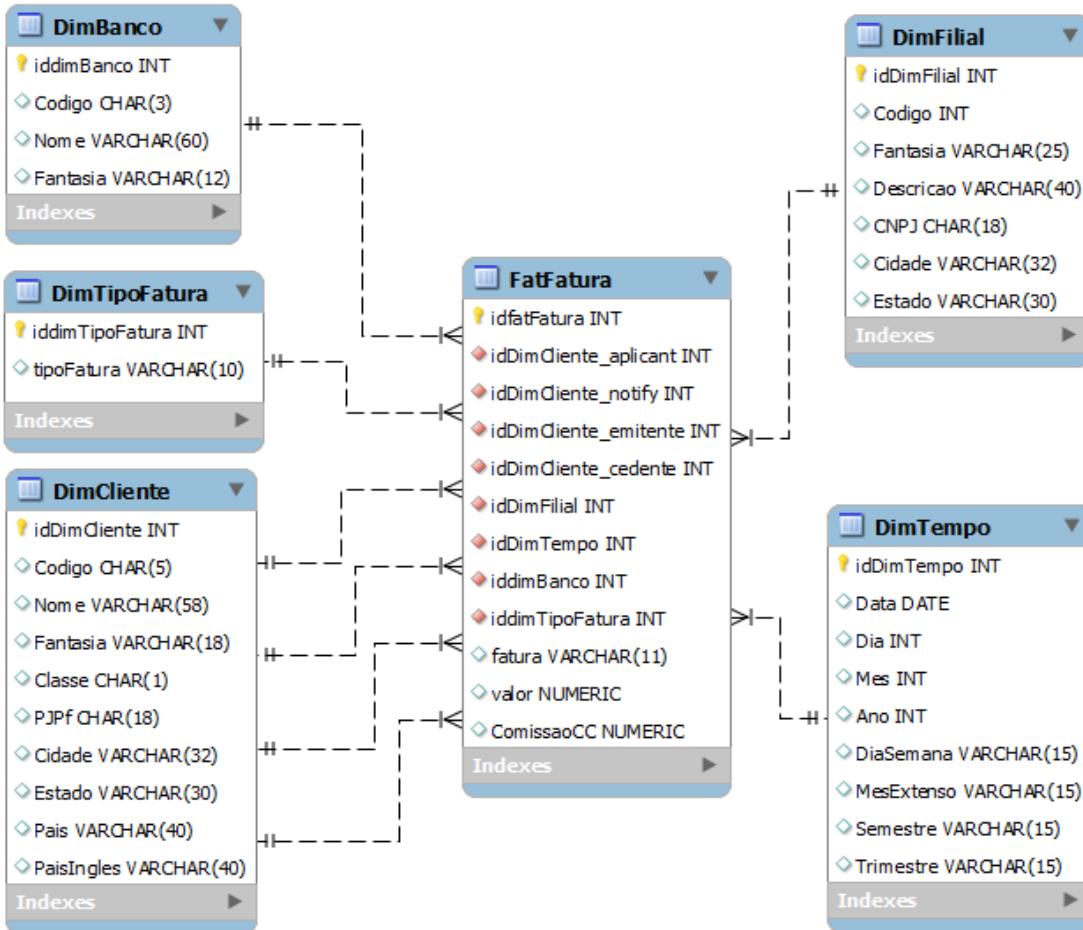
O cubo de fatura possui cinco dimensões, sendo duas dimensões compartilhadas (DimTempo e DimFilial), detalhadas no item 4.3.1, e as outras dimensões são:

⁵ Comissão em conta gráfica ocorre quando o representante é do exterior, então o valor da comissão é informado ao Banco Central, junto com outras informações através do Registro de Exportação. No momento que o *applicant* efetuar o pagamento da fatura o próprio Banco Central retém o valor da comissão e o envia diretamente a conta no exterior do representante.

- **Dimensão banco (DimBanco):** Contém os dados referente ao banco por onde vai ser realizado o pagamento da fatura, contendo as seguintes informações: código do banco, nome, nome fantasia;
- **Dimensão tipo de fatura (DimTipoFatura):** Esta dimensão tem como função a segregação das faturas em basicamente dois tipos: comercial e amostra;
- **Dimensão cliente (DimCliente):** Contém dos dados referente aos clientes da organização, as seguintes informações: código, nome, nome fantasia, Classe do cliente, CPF ou CNPJ dependendo se o cliente for pessoa física ou jurídica, Cidade, Estado e País.

Na Figura 12 encontra-se o diagrama da tabela fato (FatFatura) com suas dimensões. Pode-se notar que existem quatro ligações entre a tabela fato e a dimensão cliente, isso representa que para a emissão de uma fatura são necessários quatro clientes, que são o emitente, o cedente, o *applicant* e o *notify*, os dois primeiros são para controle interno da organização, por outro lado o *applicant* representa o cliente que vai pagar pela mercadoria e o *notify* representa as informações do cliente que será apenas notificado sobre a fatura. Existe ainda uma quinta informação que é o *consignee*, que representa quem vai receber a mercadoria no exterior, porém, esta informação foi julgada irrelevante para o presente trabalho.

Figura 12 – Cubo de Faturas.



Fonte: Elaborado pelo autor.

Na Listagem 7 encontra-se o código SQL para a criação da tabela fato (FatFatura) e das dimensões (DimBanco, DimTipoFatura e DimCliente) no banco de dados.

Listagem 7 – Código SQL para criação do fato fatura e as dimensões cliente, tipo de fatura e banco.

```

1  -- 
2  -- Table DimCliente
3  --
4  CREATE TABLE DimCliente (
5      idDimCliente serial,
6     Codigo      CHAR(5),
7      Nome       VARCHAR(58),
8      Fantasia   VARCHAR(18),
9      Classe     CHAR(1),
10     PJPf       CHAR(18),
11     Cidade     VARCHAR(32),
12     Estado     VARCHAR(30),
13     Pais       VARCHAR(40),
14     PaisIngles VARCHAR(40),
15     CONSTRAINT PK_dimCliente_id PRIMARY KEY (idDimCliente)
16 );
17
18 --
19 -- Table DimTipoFatura

```

```

20 -- -----
21 CREATE TABLE DimTipoFatura (
22   iddimTipoFatura serial,
23   tipoFatura      VARCHAR(10),
24   CONSTRAINT PK_dimTipoFatura_id PRIMARY KEY (iddimTipoFatura)
25 );
26
27 -- -----
28 -- Table DimBanco
29 -- -----
30 CREATE TABLE DimBanco (
31   idDimBanco serial,
32   codigo      CHAR(3),
33   nome        VARCHAR(60),
34   fantasia    VARCHAR(12),
35   CONSTRAINT PK_dimBanco_id PRIMARY KEY (idDimBanco)
36 );
37
38 -- -----
39 -- Table FatFatura
40 -- -----
41 CREATE TABLE FatFatura (
42   idfatFatura      serial,
43   idDimCliente_applicant INT NOT NULL,
44   idDimCliente_notify    INT NOT NULL,
45   idDimCliente_emitente  INT NOT NULL,
46   idDimCliente_cedente   INT NOT NULL,
47   idDimFilial        INT NOT NULL,
48   idDimBanco          INT NOT NULL,
49   idDimTempo          INT NOT NULL,
50   iddimTipoFatura    INT NOT NULL,
51   fatura              VARCHAR(11),
52   valor               DECIMAL(15,2) DEFAULT 0,
53   ComissaoCC          DECIMAL(15,2) DEFAULT 0,
54   CONSTRAINT PK_fatFatura_id PRIMARY KEY (idfatFatura),
55   CONSTRAINT FK_fatFatura_DimCliente_Applicant FOREIGN KEY
56     (idDimCliente_applicant) REFERENCES DimCliente(idDimCliente),
57   CONSTRAINT FK_fatFatura_DimCliente_notify   FOREIGN KEY
58     (idDimCliente_notify) REFERENCES DimCliente(idDimCliente),
59   CONSTRAINT FK_fatFatura_DimCliente_Emitente FOREIGN KEY
60     (idDimCliente_emitente) REFERENCES DimCliente(idDimCliente),
61   CONSTRAINT FK_fatFatura_DimCliente_Cedente  FOREIGN KEY
62     (idDimCliente_cedente) REFERENCES DimCliente(idDimCliente),
63   CONSTRAINT FK_fatFatura_DimFilial        FOREIGN KEY
64     (idDimFilial) REFERENCES DimFilial(idDimFilial),
65   CONSTRAINT FK_fatFatura_DimBanco          FOREIGN KEY
66     (idDimBanco) REFERENCES DimBanco(idDimBanco),
67   CONSTRAINT FK_fatFatura_DimTempo         FOREIGN KEY
68     (idDimTempo) REFERENCES DimTempo(idDimTempo),
69   CONSTRAINT FK_fatFatura_DimTipoFatura   FOREIGN KEY
70     (iddimTipoFatura) REFERENCES DimTipoFatura(iddimTipoFatura)
71 );

```

Fonte: Elaborado pelo autor.

4.3.3.2 Cubo de comissões

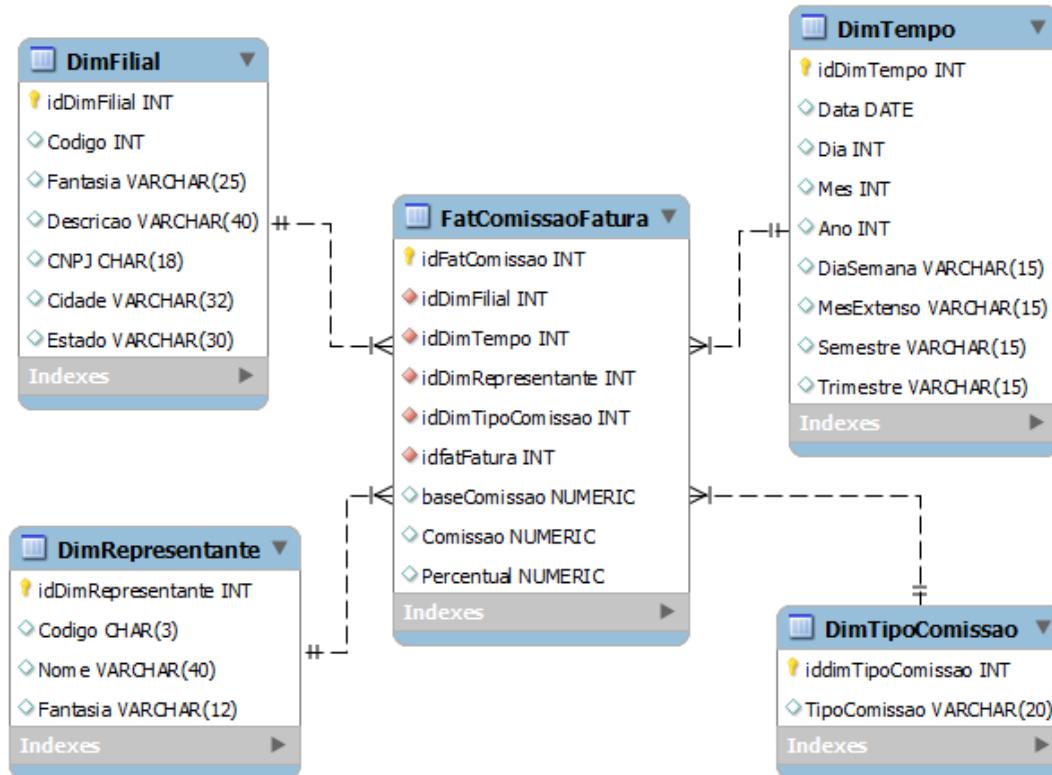
O cubo de comissões tem como objetivo o registro das comissões relativas as faturas de exportação efetuadas pela organização, este fato possui as seguintes métricas: Base de comissão, percentual de comissão e o valor da comissão, os valores registrados neste cubo estão em dólares. Os cinco W para o fato de compras são:

- **What:** Base de comissão, percentual da comissão e valor da comissão;
- **Why:** Representa a razão do comissionamento. Como é uma constante, não se aplica;
- **Where:** Representa o local onde a comissão foi gerada, e está relacionada com a dimensão DimFilial;
- **When:** Representa o momento em que a comissão foi gerada, e está relacionado com a dimensão DimTempo;
- **Who:** Representa o agente que recebeu a comissão, e está relacionado com a dimensão DimRepresentante.

O cubo de fatura possui quatro dimensões, sendo duas dimensões compartilhadas (DimTempo e DimFilial), detalhadas no item 4.3.1, e as outras dimensões são: tipo de comissão (DimTipoComissao), onde esta dimensão tem como função a segregação das comissões em três tipos: comissão em conta gráfica, comissão a remeter⁶ e por nota fiscal. Além disso, também possui a dimensão representante (DimRepresentante), onde está contém dos dados referente ao representante da organização junto ao cliente, com as seguintes informações: código, nome e nome fantasia. Na Figura 13 encontra-se o diagrama da tabela fato com suas dimensões.

⁶ Comissão a remeter é semelhante a conta gráfica, porém é o departamento financeiro da organização que decide o momento de realizar o pagamento do representante.

Figura 13 – Cubo de Comissões.



Fonte: Elaborado pelo autor.

Na Listagem 8 encontra-se o código SQL para a criação da tabela fato (FatComissaoFatura) e das suas dimensões específicas (DimRepresentante e DimTipoComissao) no banco de dados.

Listagem 8 – Código SQL para criação do fato comissão e as dimensões representante e tipo de comissão.

```

1  -- --
2  -- Table DimRepresentante
3  --
4  CREATE TABLE DimRepresentante (
5      idDimRepresentante serial,
6      Codigo             CHAR(3),
7      Nome               VARCHAR(40),
8      Fantasia           VARCHAR(12),
9      CONSTRAINT PK_dimRepresentante_id PRIMARY KEY (idDimRepresentante)
10 );
11
12 --
13 -- Table DimTipoComissao
14 --
15 CREATE TABLE DimTipoComissao (
16     idDimTipoComissao serial,
17     tipoComissao       VARCHAR(20),
18     CONSTRAINT PK_dimTipoComissao_id PRIMARY KEY (idDimTipoComissao)
19 );
20
21 --
22 -- Table FatComissaoFatura

```

```

23 -----
24 CREATE TABLE fatComissaoFatura (
25   idFatComissao      serial,
26   idDimFilial        INT NOT NULL,
27   idDimTempo          INT NOT NULL,
28   idDimRepresentante INT NOT NULL,
29   idDimTipoComissao  INT NOT NULL,
30   idfatFatura        INT NOT NULL,
31   baseComissao        DECIMAL(15,2) DEFAULT 0,
32   Comissao            DECIMAL(15,2) DEFAULT 0,
33   Percentual          DECIMAL(15,2) DEFAULT 0,
34   CONSTRAINT PK_fatComissaoFatura PRIMARY KEY (idFatComissao),
35   CONSTRAINT FK_FatComissaoFatura_DimRepresentante FOREIGN KEY
36     (idDimRepresentante) REFERENCES DimRepresentante(idDimRepresentante),
37   CONSTRAINT FK_FatComissaoFatura_DimTempo      FOREIGN KEY
38     (idDimTempo) REFERENCES DimTempo(idDimTempo),
39   CONSTRAINT FK_FatComissaoFatura_DimFilial    FOREIGN KEY
40     (idDimFilial) REFERENCES DimFilial(idDimFilial),
41   CONSTRAINT FK_FatComissaoFatura_DimTipoComissao FOREIGN KEY
42     (idDimTipoComissao) REFERENCES DimTipoComissao(idDimTipoComissao),
43   CONSTRAINT FK_FatComissaoFatura_fatFatura    FOREIGN KEY
44     (idfatFatura) REFERENCES FatFatura(idfatFatura)
45 );

```

Fonte: Elaborado pelo autor.

4.4 Transformação dos dados

Para a documentação dos processos ETL utilizam-se alguns documentos que indicam detalhes do processo de extração, como o mapa de extração, que é responsável pela documentação dos sistemas de origem das informações, os responsáveis pela informação, tipo de banco de dados, SO, tamanho do banco de dados da aplicação e a quantidade de transações diárias. A Tabela 2 apresenta o mapa de extração dos grupos de informação modelados no presente trabalho.

Tabela 2 – Mapa de extração

Grupo de informação	Área de negócio	Nome do aplicativo	Responsável pelo setor	SGDB	Quantidade de usuários	Tamanho do DB	Transações por dia
Compra	Compras	SCE	Maria Cecilia de Vargas	PostgreSQL	50	23 GB	180
Estoque	Compras	SCE	Maria Cecilia de Vargas	PostgreSQL	50	23 GB	5000
Ajuste	Compras	SCE	Maria Cecilia de Vargas	PostgreSQL	50	23 GB	7
Consumo	Compras	SCE	Maria Cecilia de Vargas	PostgreSQL	50	23 GB	1518
Fatura	Comercial	COM	André Gustavo Hass	xBase	15	8 GB	2
Comissões	Comercial	COM	André Gustavo Hass	xBase	15	8 GB	4

Fonte: Elaborado pelo autor.

Devido ao fato de algumas informações sobre a extração serem as mesmas para todas as linhas da tabela, as respectivas colunas foram suprimidas do mapa de extração para facilitar a visualização. Essas informações são: Linux (Sistema Operacional), e Leandro Portz (Responsável técnico).

4.4.1 Área de compras

Outro documento utilizado para a documentação do processo ETL é a tabela de informação volumétrica, esta é responsável por auxiliar nas decisões referentes ao armazenamento físico dos dados, mostrando uma previsão de crescimento do BD. Na Tabela 3 encontram-se as informações volumétricas para a área de compras.

Tabela 3 – Tabela de Informação volumétrica da área de compras.

Nome da tabela	Contagem inicial de linhas	Média de comp. de linha	Cresce com	Expectativa mensal de linhas	Expectativa mensal de Bytes	Tamanho inicial da tabela	Tamanho para seis meses MB
DimTempo	0	80	Novo dia	30	2400	0	0,014
DimFilial	14	153	Nova filial	0	0	2142	0,002
DimProduto	4000	233	Novo produto	10	2330	932000	0,924
DimRequisicao	2	19	Novo tipo de requisição	0	0	38	0,000
FatAjuste	0	136	Novo ajuste	207	28152	0	0,165
FatCompra	0	200	Nova compra	5382	1076400	0	6,307
FatConsumo	0	140	Novo consumo	45557	6377980	0	37,371
FatEstoque	0	136	Novo dia	110000	14960000	0	87,656

Fonte: Elaborado pelo autor.

Devido ao fato de alguns dados sobre a informação volumétrica serem os mesmos para todas as linhas da tabela, as respectivas colunas foram suprimidas do mapa de extração para facilitar a visualização. Essas informações são: inserir (Estratégia atualização), diária (Frequência de carga), etl_estoque (Nome do processo de ETL),

Na Tabela 4 encontra-se o mapa de transformação para as tabelas fato e dimensões específicas da área de compras. O mapa de transformação indica a origem e o destino das informações no processo de carga, bem como possíveis transformações realizadas sobre os dados.

Tabela 4 – Mapa de transformação da área de compras.

Banco de dados	Origem			Destino				S C D	Transformação
	Tabela Origem	Campo Origem	Tipo campo	Tabela Destino	Campo Destino	Tipo Campo	Tipo Tabela		
SQL	dfiliais	codi	Char(2)	dimfilial	codigo	integer	Dimensão	2	
SQL	dfiliais	fant	Char(25)	dimfilial	fantasia	VARCHAR(25)	Dimensão	1	
SQL	dfiliais	desc	Char(40)	dimfilial	descricao	VARCHAR(40)	Dimensão	1	
SQL	dfiliais	cnpj	Char(18)	dimfilial	cnpj	Char(18)	Dimensão	1	
SQL	dsce045	nome	Char(32)	dimfilial	cidade	VARCHAR(32)	Dimensão	1	
SQL	dscp060	desc	Char(30)	dimproduto	estado	VARCHAR(40)	Dimensão	1	
SQL	dsce100	redu	Char(6)	dimproduto	codigo	Char(6)	Dimensão	2	
SQL	dsce100	desc	Char(35)	dimproduto	descricao	VARCHAR(45)	Dimensão	1	
SQL	dsce100	estrutural	Char(26)	dimproduto	estrutural	VARCHAR(25)	Dimensão	1	
SQL	dsce100	ncm	Char(14)	dimproduto	ncm	VARCHAR(15)	Dimensão	1	
SQL	dsce100	unid	Char(2)	dimproduto	unidade	VARCHAR(3)	Dimensão	1	
SQL	dscp067	nivel	Char(34)	dimproduto	categorian1	VARCHAR(35)	Dimensão	1	
SQL	dscp067	nivel	Char(34)	dimproduto	categorian2	VARCHAR(35)	Dimensão	1	
SQL	dscp067	nivel	Char(34)	dimproduto	categorian3	VARCHAR(35)	Dimensão	1	
SQL	dsce200	forn	Char(5)	dimcliente	codigo	Char(5)	Dimensão	2	
SQL	dsce200	nome	Char(58)	dimcliente	nome	VARCHAR(58)	Dimensão	1	
SQL	dsce200	fant	Char(18)	dimcliente	fantasia	VARCHAR(18)	Dimensão	1	
SQL	dsce200	classecli	Char(1)	dimcliente	classe	Char(1)	Dimensão	1	
SQL	dsce200	cgc/cpf	Char(18)	dimcliente	pjpf	Char(18)	Dimensão	1	
SQL	dsce045	nome	Char(32)	dimcliente	cidade	VARCHAR(32)	Dimensão	1	
SQL	dscp060	desc	Char(30)	dimcliente	estado	VARCHAR(30)	Dimensão	1	
SQL	dscp220	desc	Char(40)	dimcliente	pais	VARCHAR(40)	Dimensão	1	
SQL	dscp220	descing	Char(40)	dimcliente	paisingles	VARCHAR(40)	Dimensão	1	
SQL	dsce200	forn	Char(5)	dimfornecedor	codigo	Char(5)	Dimensão	2	
SQL	dsce200	nome	Char(58)	dimfornecedor	nome	VARCHAR(58)	Dimensão	1	
SQL	dsce200	fant	Char(18)	dimfornecedor	fantasia	VARCHAR(18)	Dimensão	1	
SQL	dsce200	cgc	Char(18)	dimfornecedor	cnpj	Char(18)	Dimensão	1	
SQL	dsce045	nome	Char(32)	dimfornecedor	cidade	VARCHAR(32)	Dimensão	1	
SQL	dscp060	desc	Char(30)	dimfornecedor	estado	VARCHAR(30)	Dimensão	1	
SQL	dscp220	desc	Char(40)	dimfornecedor	pais	VARCHAR(40)	Dimensão	1	
SQL	dscp220	descing	Char(40)	dimfornecedor	paisingles	VARCHAR(40)	Dimensão	1	
SQL	dsce400	qtde	numeric	fatajuste	quantidade	numeric	Fato	2	
SQL	dsce400	valr	numeric	fatajuste	valor	numeric	Fato	2	
SQL	dsce400	qtde	numeric	fatcompra	quantidade	numeric	Fato	2	
SQL	dsce400	valr	numeric	fatcompra	valor_liquido	numeric	Fato	2	
SQL	dsce400	vlnf	numeric	fatcompra	valor_nf	numeric	Fato	2	
SQL	dsce400	obse	Char(20)	dimrequisicao	descricao	VARCHAR(15)	Dimensão	2	Se obse contiver 'MANUAL' então 'Req manual' se não 'Req. Lote'
SQL	dsce400	qtde	numeric	fatconsumo	quantidade	numeric	Fato	2	
SQL	dsce400	valr	numeric	fatconsumo	valor	numeric	Fato	2	
SQL	dsce400	fatestoque	numeric	fatestoque	quantidade	numeric	Fato	2	
SQL	dsce400	fatestoque	numeric	fatestoque	valor	numeric	Fato	2	

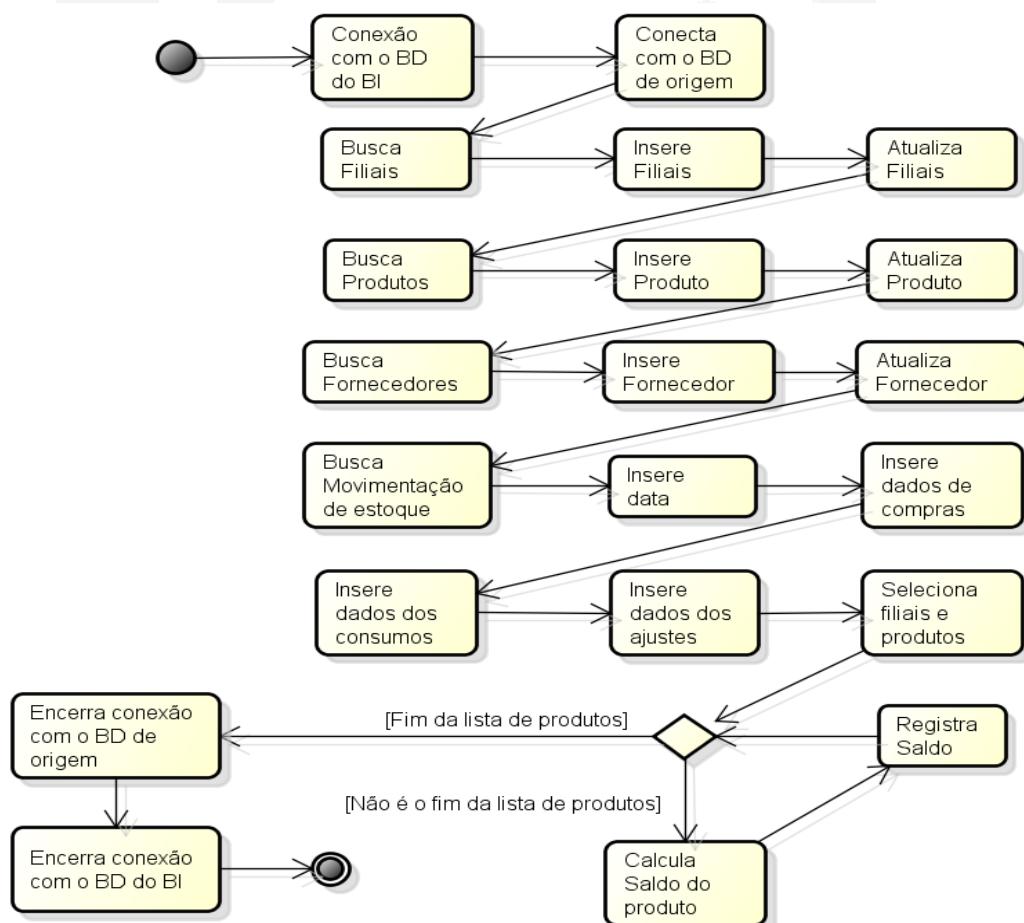
Fonte: Elaborado pelo autor.

A tabela FatEstoque não possui informações de origem, pois os dados registrados nesta são o resultado de uma função e não dados encontrados em outras tabelas. A função utilizada para gerar os níveis de estoques é a “saldorazao115”, esta recebe como parâmetro o código da filial, o código do produto e a data em que se deseja os valores de estoque.

Para que a transformação dos dados, descrita na Tabela 4, ocorra estes devem passar pelo processo ETL. Na área de compras este processo é executado todos os dias a 1 hora e 5 minutos da madrugada.

O processo está representado pelo diagrama de atividade encontrado na Figura 14. Este diagrama tem a função de demonstrar graficamente os passos realizados por um algoritmo para facilitar o seu entendimento. Como pode ser visto na Figura 14, após a conexão com os respectivos BD, vários grupos de informação são inseridos e atualizados, tais como: Filiais, produtos, fornecedores e datas. Para essas informações somente serão importados os dados inexistentes na base do BI, mediante a alguma relação com o fato importado. As únicas informações a serem importadas integralmente são as referentes a movimentação dos estoque da organização.

Figura 14 – Diagrama de atividade do processo de ETL da área de compras.



Fonte: Elaborado pelo autor.

No Apêndice B pode-se encontrar o código fonte, escrito em PHP, do processo ETL da área de compras.

4.4.2 Área comercial

Na Tabela 5 encontram-se as informações volumétricas para a área comercial, com o objetivo de auxiliar nas decisões referentes ao armazenamento físico dos dados, mostrando uma previsão de crescimento do BD.

Tabela 5 – Tabela de Informação volumétrica da área comercial.

Nome da tabela	Contagem linhas iniciais	Média de comp. de linha	Cresce com	Expectativa mensal de linhas	Expectativa mensal de Bytes	Tamanho inicial da tabela	Tamanho para seis meses MB
DimBanco	4	79	Nova Banco	1	79	316	0,001
DimCliente	724	246	Novo Cliente	5	1230	178104	0,181
DimRepresentante	60	49	Novo Representante	1	49	2940	0,003
DimTipocomissao	3	24	Novo tipo de comissão	0	0	72	0,000
DimTipofatura	2	14	Novo tipo de fatura	0	0	28	0,000
fatcomissaofatura	0	204	Nova comissão	50	10200	0	0,060
fatfatura	0	167	Nova fatura	50	8350	0	0,049

Fonte: Elaborado pelo autor.

Devido ao fato de alguns dados sobre a informação volumétrica serem os mesmos para todas as linhas da tabela, as respectivas colunas foram suprimidas do mapa de extração para facilitar a visualização. Essas informações são: inserir (Estratégia atualização), diária (Frequência de carga), cargaBI (Nome do processo de ETL).

Na Tabela 6, encontra-se o mapa de transformação para as tabelas fato e dimensões específicas da área comercial. O mapa de transformação indica a origem e o destino das informações no processo de carga, bem como possíveis transformações realizadas sobre os dados.

Tabela 6 – Tabela de transformação da área comercial.

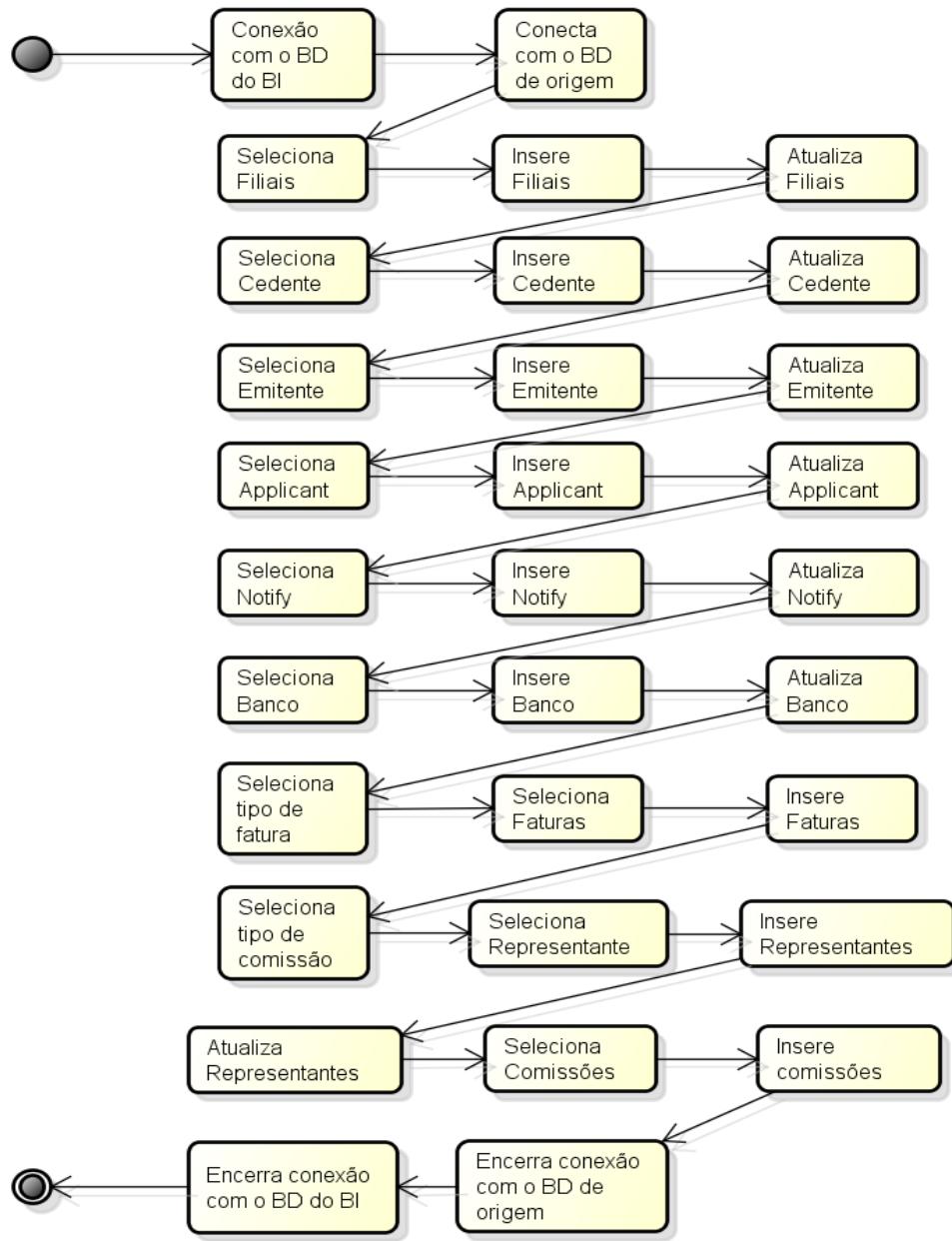
Banco de dados	Origem			Destino				S C D	Transformação
	Tabela Origem	Campo de Origem	Tipo campo	Tabela Destino	Campo Destino	Tipo Campo	Tipo Tabela		
SQL	dspr130	bco	Char(3)	dimbanco	codigo	Char(3)	Dimensão	2	
SQL	dspr130	nome	Char(59)	dimbanco	nome	VARCHAR(60)	Dimensão	1	
SQL	dspr130	fant	Char(12)	dimbanco	fantasia	VARCHAR(12)	Dimensão	1	
SQL	dscp210	repr	Char(3)	dimrepresentante	codigo	Char(3)	Dimensão	2	
SQL	dscp210	nome	Char(40)	dimrepresentante	nome	VARCHAR(40)	Dimensão	1	
SQL	dscp210	fant	Char(12)	dimrepresentante	fantasia	VARCHAR(12)	Dimensão	1	
DBF	dspr800	tfat	numeric	fatcomissaofatura	basecomissao	numeric	Fato	2	
DBF	dspr910	vcom	numeric	fatcomissaofatura	comissao	numeric	Fato	2	
DBF	dspr910	comi	numeric	fatcomissaofatura	percentual	numeric	Fato	2	
DBF	dspr800	invo	Char(10)	fatfatura	fatura	VARCHAR(11)	Fato	2	
DBF	dspr800	tfat	numeric	fatfatura	valor	numeric	Fato	2	
DBF	dspr800	ccgr	numeric	fatfatura	comissaocc	numeric	Fato	2	
DBF	dspr910	tcon	Char(1)	dimtipocomissao	descricao	VARCHAR(20)	Dimensão	2	Se tcom='N' então 'Nota Fiscal' se não se tcom='S' então 'Conta Gráfica' se não 'A Remeter' Se cosa='S' então 'Amostra' se não 'Comercial'
DBF	dspr800	cosa	Char(1)	dimtipofatura	tipofatura	VARCHAR(10)	Dimensão	2	

Fonte: Elaborado pelo autor.

Para que a transformação dos dados, descrita na Tabela 5 ocorra, estes devem passar pelo processo ETL. Na área comercial este processo é executado manualmente, pois os dados referente as faturas estão armazenados em um formato que não é mais possível acessar utilizando o PHP. O processo está representado pelo diagrama de atividade encontrado na Figura 15.

Como pode ser visto na Figura 15, após a conexão com os respectivos Bancos de Dados, vários grupos de informação são inseridos e atualizados, tais como: filiais, clientes, bancos e representantes. Para essas informações somente serão importados os dados inexistentes na base do BI, mediante a alguma relação com o fato importado. As únicas informações a serem importadas integralmente são as referentes as faturas e suas comissões.

Figura 15 – Diagrama de atividade do processo de ETL da área comercial.



Fonte: Elaborado pelo autor.

4.5 Escolha da ferramenta de BI

Nesta etapa realizou-se uma pesquisa para a escolha de uma ferramenta de BI para implantação na organização, para a avaliação mais criteriosa foram selecionados apenas os *softwares* com versões gratuitas para teste, ficando para um segundo momento apenas as

ferramentas *open source*, uma vez que esta era uma restrição imposta pela organização. Neste caso, então os seguintes aplicativos foram avaliados:

- **QlikView:** Esta ferramenta permite as empresas começarem com uma única versão do *QlikView Desktop* e expandirem para múltiplos servidores que suportam milhares de usuários ou milhões de registros. Acessa informações em dados armazenados em Oracle, Microsoft SQL Server ou MySQL. Trabalha com o conceito de pesquisa associativa, que possibilita uma experiência familiar que apresenta respostas rápidas. A sua interface intuitiva sublinha relações importantes nos seus dados, mostrando não só as associações, como também dados que não estão relacionados. A visualização é a parte divertida, em apenas alguns segundos, o usuário visualiza seus dados da forma que quiser, colocando em tabelas, quadros ou gráficos de todos os tipos imagináveis, ampliando ou minimizando a visualização. Para disponibilizar a ferramenta para múltiplos usuários é necessário a aquisição do *Qlikview server*. A forma de licenciamento da versão *server* é por usuário, sendo que para 5 usuários finais e 5 desenvolvedores o licenciamento custa por volta de R\$ 60.000,00 (QLIK, 2014, texto digital).
- **Pentaho:** É um software *open source* criado pela Pentaho, fundada em 2004, tendo sua primeira versão anunciada em 2005 nos Estados Unidos. Abordagem moderna, simplificada e interativa da Pentaho permite que os usuários possam acessar, descobrir e misturar todos os tipos e dados. Com um espectro de análise cada vez mais avançadas, os usuários podem analisar e visualizar dados em várias dimensões, ao mesmo tempo, minimizando a dependência de TI (PENTAHO, 2014, texto digital).
- **SpagoBI:** É um software *open source* criado pela *SpagoBI Labs*, na Itália. Abrange todas as áreas de análises de BI, com temas e motores inovadores. SpagoBI oferece uma ampla gama de ferramentas analíticas, como: Gerador de relatórios, ferramenta OLAP, gráficos, ETL, entre outras (SPAGOBI, 2014, texto digital).

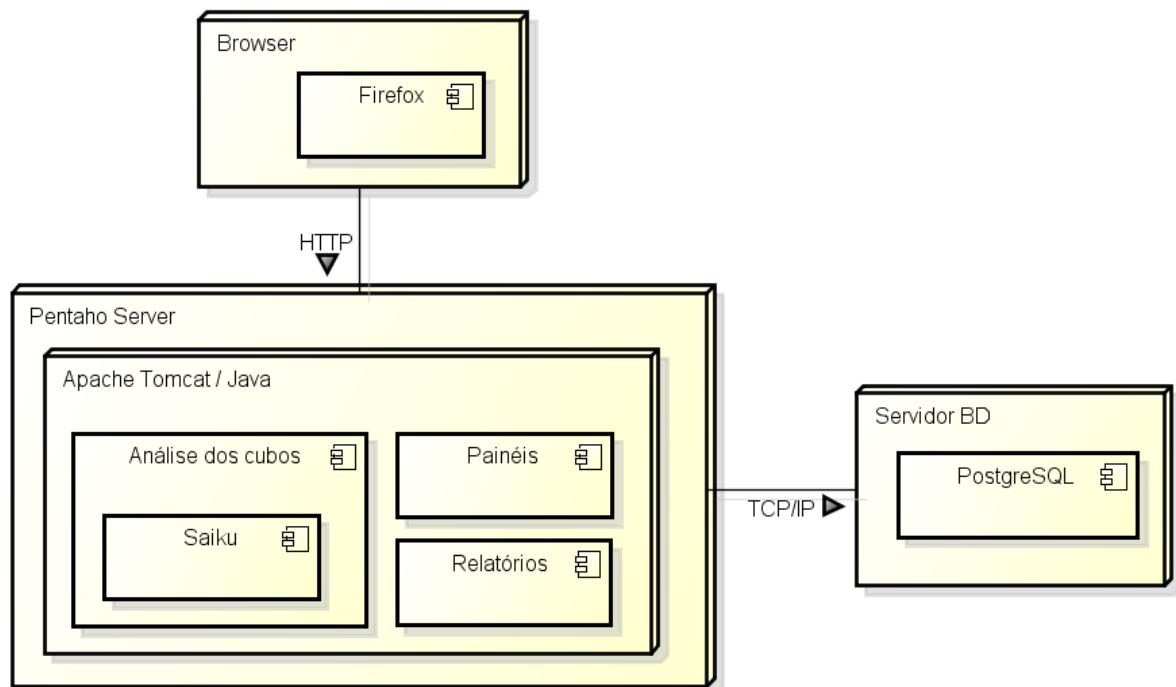
Após a análise das ferramentas, optou-se pela utilização do Pentaho, pois este se mostrou uma ferramenta de configurações iniciais mais simples, possui vasta documentação em Português, além de ser *open source*, o que está de acordo com a diretriz estabelecida pela organização.

O QlikView foi descartado por apenas poder ser utilizado na plataforma *Windows*, o que não ocorre no parque de máquinas da organização. Já o SpagoBI apesar de ser uma ferramenta *web*, assim como o Pentaho, logo após a instalação mostrou-se um aplicativo de configuração complexa e extensa, além de possuir uma documentação escassa e apenas no idioma Inglês.

4.6 Arquitetura

Para a disponibilização do ambiente de apoio a decisão para os usuários efetuarem os testes foi utilizado um computador com o Pentaho e o SGBD instalados, realizando o papel de servidor. Porém, quando o sistema entrar em produção, sendo utilizado por um grupo maior de usuários, a arquitetura de disponibilização do serviço será semelhante o encontrado na Figura 16, tendo um serviço de Banco de dados e outro para a aplicação, aumentando assim o desempenho do ambiente.

Figura 16 – Arquitetura do ambiente.



Fonte: Elaborado pelo autor.

4.7 Tecnologias utilizadas

Para a realização das tarefas necessárias para a concretização dos objetivos propostos algumas tecnologias foram necessárias, dentre elas podemos citar:

- **Banco de dados PostgreSQL:** PostgreSQL é um poderoso sistema de banco de dados *open source*. Ele tem mais de 15 anos de desenvolvimento ativo e uma arquitetura que ganhou uma forte reputação de confiabilidade, integridade de dados e correção. Funciona em todos os principais sistemas operacionais, tais como Linux, UNIX, MAC e Windows. É totalmente compatível com o conceito de Atomicidade, Consistência, Isolamento e Durabilidade (ACID), tem suporte completo para chaves estrangeiras, junções, *views*, *triggers* e *stored procedures* em várias linguagens. Suporta o armazenamento de grandes objetos binários, incluindo imagens, sons ou vídeos. Possui interfaces de programação nativas para C/C++, Java, PHP, entre várias outras linguagens. O tamanho máximo de uma base de dados é ilimitada, cada tabela pode possuir até 32 terabytes de tamanho, cada linha de uma tabela pode possuir o tamanho de 1.6 terabytes (POSTGRESQL, 2014, texto digital).
- **DBlink:** É um módulo que oferece suporte a conexão com outros bancos de dados PostgreSQL de dentro de uma conexão ativa (POSTGRESQL, 2014, texto digital).
- **PHP:** Utilizado neste trabalho para criação dos scripts ETL. É uma linguagem de script *open source* de uso geral, muito utilizada para o desenvolvimento de aplicações Web. O PHP pode ser utilizado na maioria dos sistemas operacionais, incluindo Linux, várias variantes Unix (incluindo HP-UX, Solaris e OpenBSD), Windows, Mac OS X, e provavelmente outros. O PHP também é suportado pela maioria dos servidores web atuais, incluindo Apache, Microsoft Internet Information Server, e muitos outros (PHP, 2014, texto digital).
- **Schema Workbench:** O Schema Workbench é uma ferramente visual que permite a criação, modificação e teste de cubos de dados. Os cubos de dados criados nesta ferramenta podem ser exportados diretamente para o Pentaho, possibilitando a utilização imediata dos mesmos (PENTAHO, 2014, texto digital).

- **Report Designer:** O Report Designer é uma ferramenta visual para criação de relatórios. Através desta ferramenta os relatórios podem ser exportados em diversos formatos, entre eles PDF, planilha eletrônica, HTML, entre outros. Os relatórios desenvolvidos com esta ferramenta podem ser exportados diretamente para o servidor Pentaho (PENTAHO, 2014, texto digital).

5 AVALIAÇÃO DO BI

Neste capítulo será apresentado como foi realizado o processo de implantação e disponibilização do ambiente e ainda o questionário de avaliação e suas respostas.

5.1 Implantação

Como plataforma de suporte para o ambiente de apoio a decisão foi utilizado um computador com o Sistema Operacional (SO) Linux, mais precisamente com Ubuntu 14.04, para a instalação do SGBD e do Java foi utilizado o gerenciador de pacotes padrão da distribuição.

Para a instalação do Pentaho o primeiro passo foi realizar o *download* do pacote de instalação, que pode ser encontrado no site da comunidade do Pentaho⁷. Este pacote é um arquivo comprimido com todos os arquivos necessários para a inicialização da ferramenta.

O processo de instalação da ferramenta é simples, basta apenas executar a ferramenta de descompressão disponível no sistema operacional utilizado como servidor e escolher o diretório onde os arquivos serão descompactados, no presente trabalho o sistema foi instalado no diretório */usr/local/*.

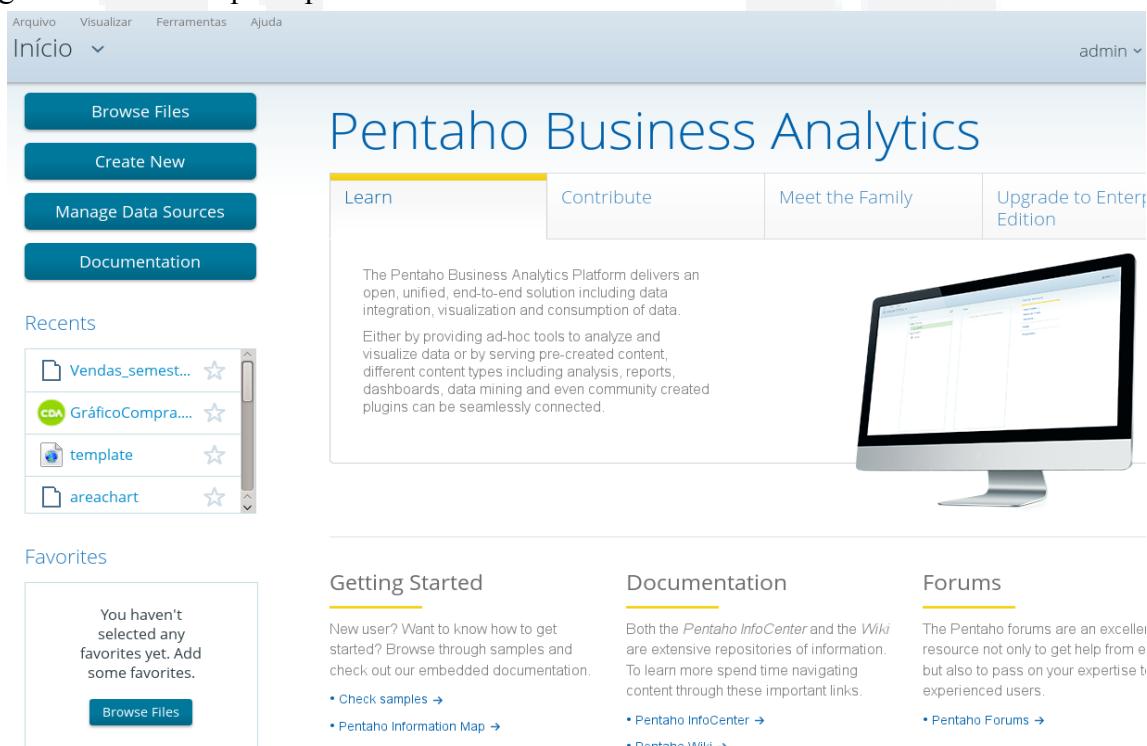
⁷ A comunidade do Pentaho fica hospedada no seguinte endereço: <http://community.pentaho.com/>

Para iniciar o serviço do ambiente basta executar o arquivo de inicialização, no processo de instalação são criados dois arquivos, sendo um para execução no *Windows*, que é o arquivo *start-pentaho.bat* e outro para MAC, Linux e UNIX, que é o arquivo *start-pentaho.sh*. O encerramento do serviço deve ser feito utilizando o arquivo *stop-pentaho*, utilizando a extensão específica para o SO do servidor.

Com o servidor em execução o serviço já está disponível para a utilização, para acessar o sistema utiliza-se um *software* cliente *web*, na barra de endereços deve-se utilizar o endereço do servidor seguido do símbolo “:” e a porta do servidor, como padrão o Pentaho está executando na porta 8080, esta configuração pode ser alterada no arquivo *server.xml*, que pode ser encontrado no diretório */usr/local/biserver-ce/tomcat/conf/*.

A Figura 17 apresenta a tela inicial do Pentaho, esta tela é apresentada após o usuário passar pelo processo de autenticação, como padrão o usuário de administrador é “*admin*” como a senha “*password*”, com este usuário é possível a manutenção dos usuários do sistema e a configuração das conexões com o SGBD, entre outras atividades.

Figura 17 – Tela de principal do Pentaho.

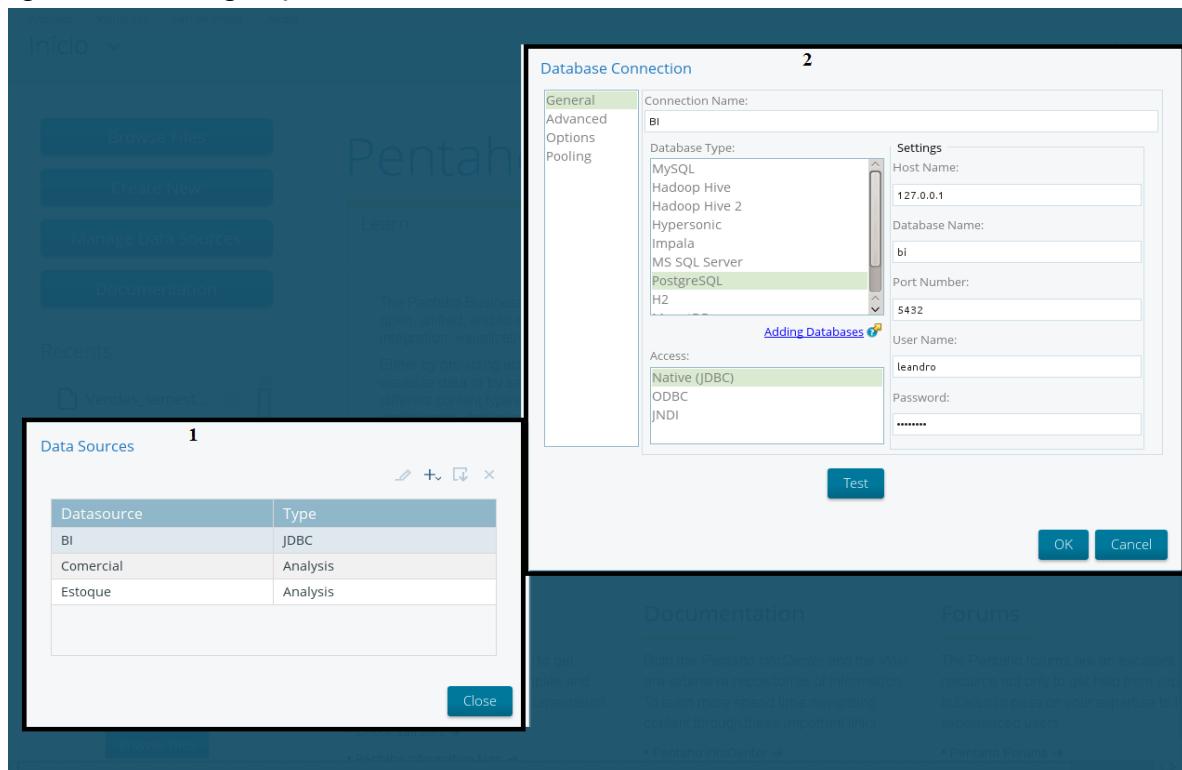


Fonte: Elaborado pelo autor.

Para que o Pentaho obtenha acesso aos dados, é necessário a configuração da conexão com o SGBD, sendo que isso é feito através do “*Manage Data Source*”. A Figura 18 apresenta as telas de configuração, onde o primeiro passo está representado pelo quadro

identificado com o número 1, sendo que nesta tela está listado as conexões já configuradas. Para criação de uma nova conexão, é necessário clicar no símbolo “+”, então será apresentada a tela identificada pelo número 2. Para criação da nova conexão é necessário definir o nome da conexão, escolher o SGBD, definir o endereço do servidor, nome do BD, a porta de conexão, usuário e senha.

Figura 18 – Configuração do DB.



Fonte: Elaborado pelo autor.

Para acrescentar novas funcionalidades ao Pentaho é possível a instalação de complementos, como o *Community Dashboard Editor* (CDE), responsável pela criação e edição dos painéis e o Saiku, ferramenta para criação de análises dos cubos de dados. A instalação destas ferramentas é realizada através do *Marketplace*, ferramenta disponível a partir da versão 4.8 do Pentaho.

A Figura 19 apresenta a tela do *Marketplace* que lista os complementos que estão instalados no servidor. Para realizar a instalação do *software* desejado basta apenas clicar no botão *install*, disponível ao lado de cada complemento. O Pentaho realiza o processo de *download* dos arquivos necessários e procederá com a instalação. Ao término do processo a reinicialização do servidor é solicitada para que a nova funcionalidade esteja disponível.

Figura 19 – *Marketplace*.

Component	Provider	Installed Version	Status
Pentaho Marketplace	Pentaho	Installed Version: TRUNK-SNAPSHOT (TRUNK)	Up to Date
Community Dashboards Framework	Webdetails	Installed Version: 14.07.29 (STABLE)	Up to Date
Community Data Access	Webdetails	Installed Version: 14.07.29 (STABLE)	Up to Date
Community Dashboard Editor	Webdetails	Installed Version: 14.07.29 (STABLE)	Up to Date
Community Graphics Generator	Webdetails	Installed Version: TRUNK-SNAPSHOT (TRUNK)	Upgrade
Saiku Analytics	Analytical Labs	Installed Version: 2.6-PENTAHO5-SNAPSHOT (TRUNK)	Up to Date
Saiku Chart Plus	IT4biz	Installed Version: RC4 (TRUNK)	Up to Date
Community Data Browser	Webdetails	Installed Version: 14.07.29 (STABLE)	Up to Date
Startup Rule Engine	Webdetails	Installed Version: 14.06.18 (RELEASE)	Up to Date
Portuguese (Brazilian variant) Language Pack Installer	Oncase, Open Consulting, IT4biz	Installed Version: RELEASE-0.7.3 (RELEASE)	Upgrade

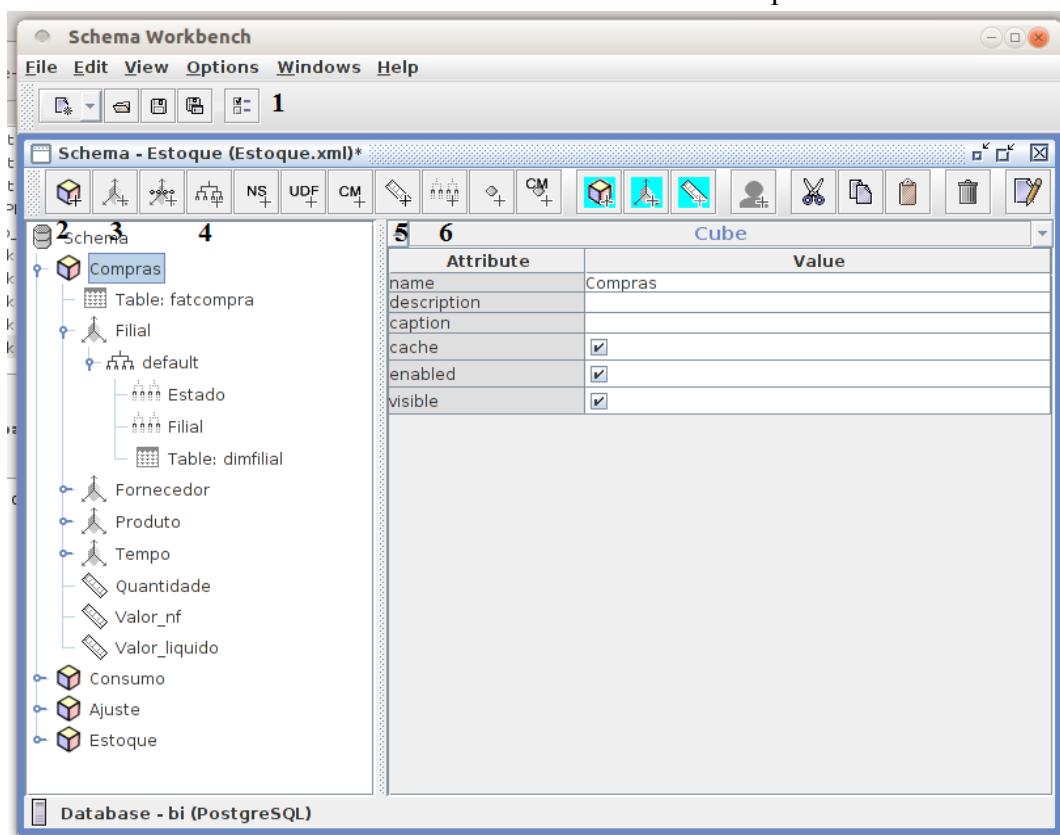
Fonte: Elaborado pelo autor.

Durante o acesso ao *Marketplace*, é realizada uma verificação dos complementos instalados em busca de atualização para os mesmos. Ao encontrar uma atualização disponível, o sistema apresenta o botão *Upgrade* ao lado do nome do complemento, como mostra a Figura 19.

5.1.1 Criação dos cubos de dados

Com a modelagem dimensional realizada e a criação das tabelas físicas no SGBD, é necessária a configuração dos cubos de dados. Este processo foi realizado com o auxílio da ferramenta *Pentaho Schema Workbench* (PSW). Na Figura 20, encontra-se a tela principal da ferramenta já com um *schema* sendo criado. Para que a criação do *schema* seja possível, é necessária a configuração de acesso ao SGBD. Isso pode ser feito através do ícone marcado pelo número 1. A tela de configuração apresentada pelo *software* é muito semelhante a encontrada no quadro 2 da Figura 18.

Figura 20 – Pentaho Schema Workbench editando o schema estoque.



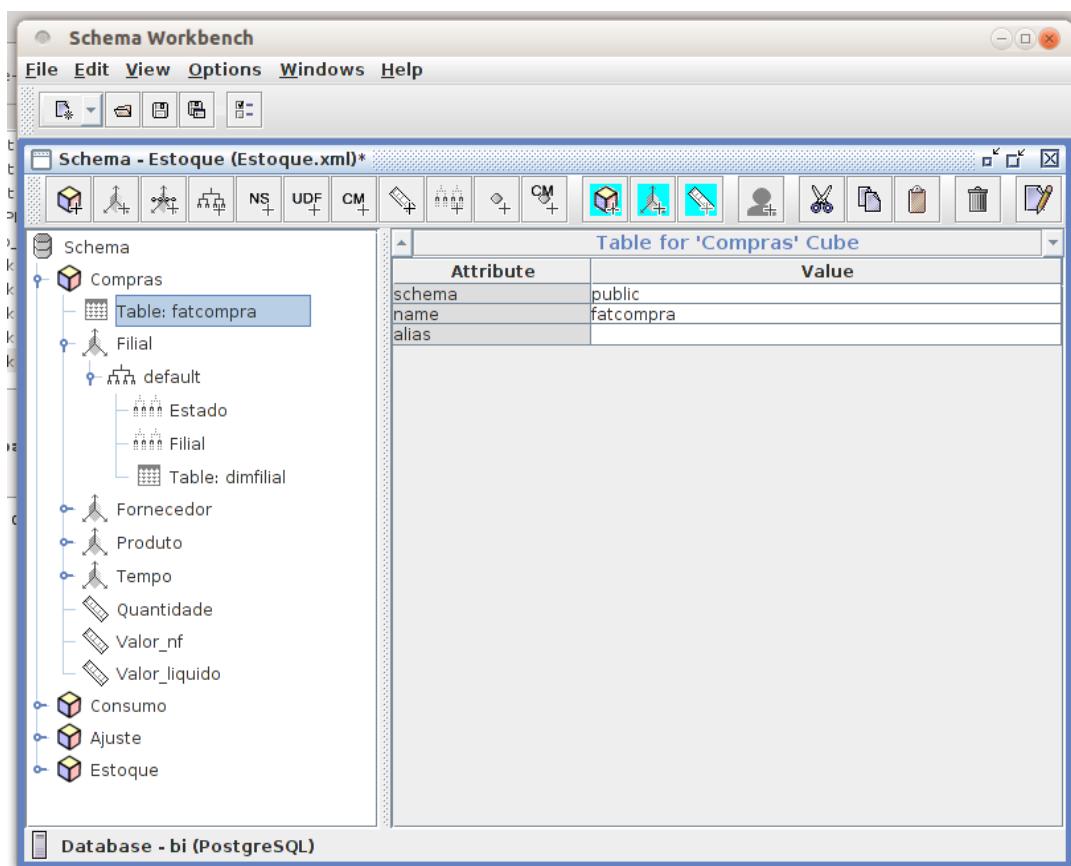
Fonte: Elaborado pelo autor.

Na Figura 20 pode-se observar a barra de ferramentas, onde os botões marcados com os números de 2 a 6 tem as seguintes funções, respectivamente: Novo cubo de dados, nova dimensão, nova hierarquia, nova métrica e novo nível de hierarquia.

Para a criação de um novo cubo de dados, o primeiro passo é a inserção de um novo cubo através do ícone marcado com o número 2. Após esta ação, o novo cubo é apresentado e seus atributos ficam disponíveis para edição, assim como está representado pela Figura 20. Ao deixar o ponteiro do *mouse* sobre o nome do atributo, a ferramenta apresenta uma breve descrição com a funcionalidade do mesmo.

Após a criação do cubo e a alteração do atributo “*name*” para o nome desejado, o próximo passo é a definição da tabela fato onde os dados referentes ao cubo estão armazenados. Este procedimento é realizado através do botão direito do mouse sobre o nome do cubo selecionando então a opção “*Add Table*”. Na Figura 21 apresenta-se a tabela já incluída e seus atributos, para esta operação apenas o atributo “*name*” precisa ser alterado.

Figura 21 – Tabela e seus atributos.



Fonte: Elaborado pelo autor.

Após a definição da tabela fato, o próximo passo é a criação das dimensões. Para realizar este procedimento, pode-se utilizar o ícone na barra de ferramentas ou o botão direito do *mouse* e a opção “*Add Dimension*”. Ao selecionar uma das duas opções, a ferramenta cria então, uma nova dimensão já com uma hierarquia. Neste momento, a ferramenta exigirá a criação de um nível na hierarquia, porém, isso só é possível após a definição da tabela que contém os dados da dimensão. Para realizar esta operação o usuário deve clicar com o botão direito do *mouse* sobre o nome da hierarquia e selecionar a opção “*Add Table*”, sendo que os atributos a ser alterado são os mesmos da tabela fato.

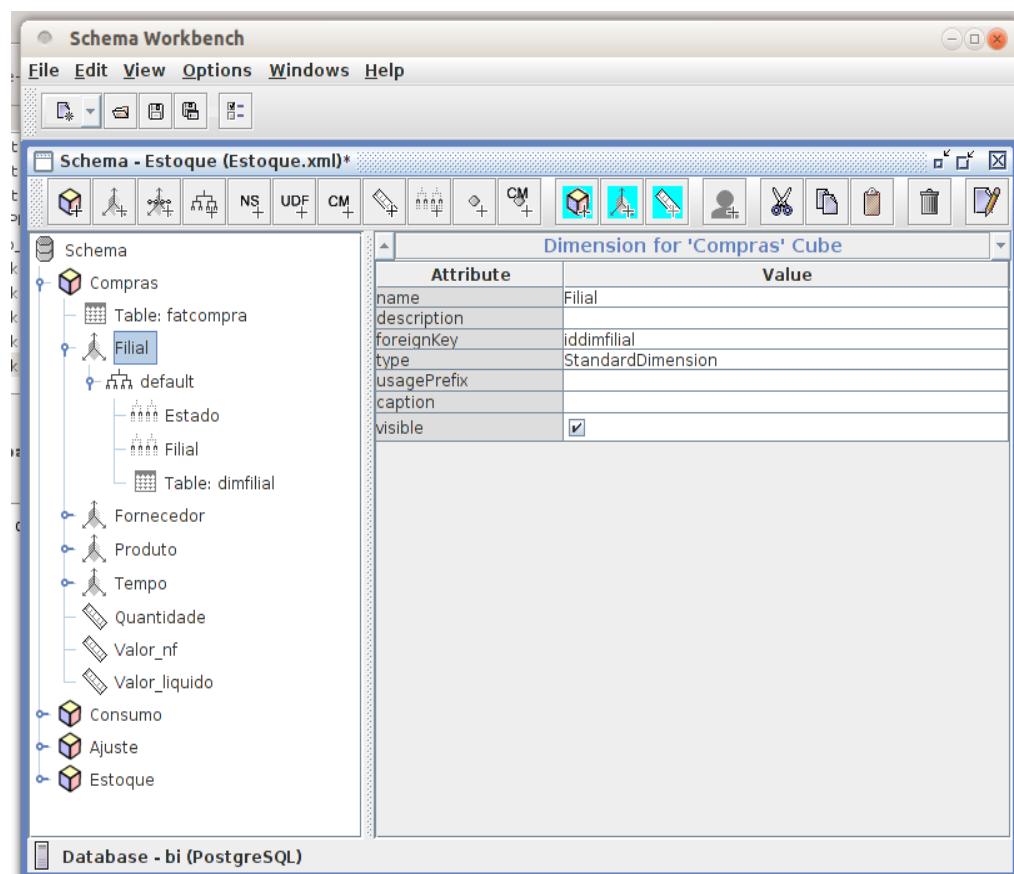
Após a seleção da tabela que representa a dimensão, o usuário deve selecionar a dimensão para editar os seus atributos, como mostra a Figura 22, os campos que devem ser alterados são:

- **Name:** para definir o nome da dimensão;
- **ForeignKey:** indica qual é a coluna que serve como chave estrangeira para a tabela fato. A configuração correta deste campo é muito importante. Caso isso seja feito de

maneira incorreta durante a análise do cubo no Pentaho, utilizando o Saiku por exemplo, ocorre um erro de tipo de dados incompatíveis na consulta e o Pentaho não deixa claro o motivo do erro;

- **Type:** este atributo tem a função de definir se a dimensão é normal ou de tempo.

Figura 22 – Atributos da dimensão

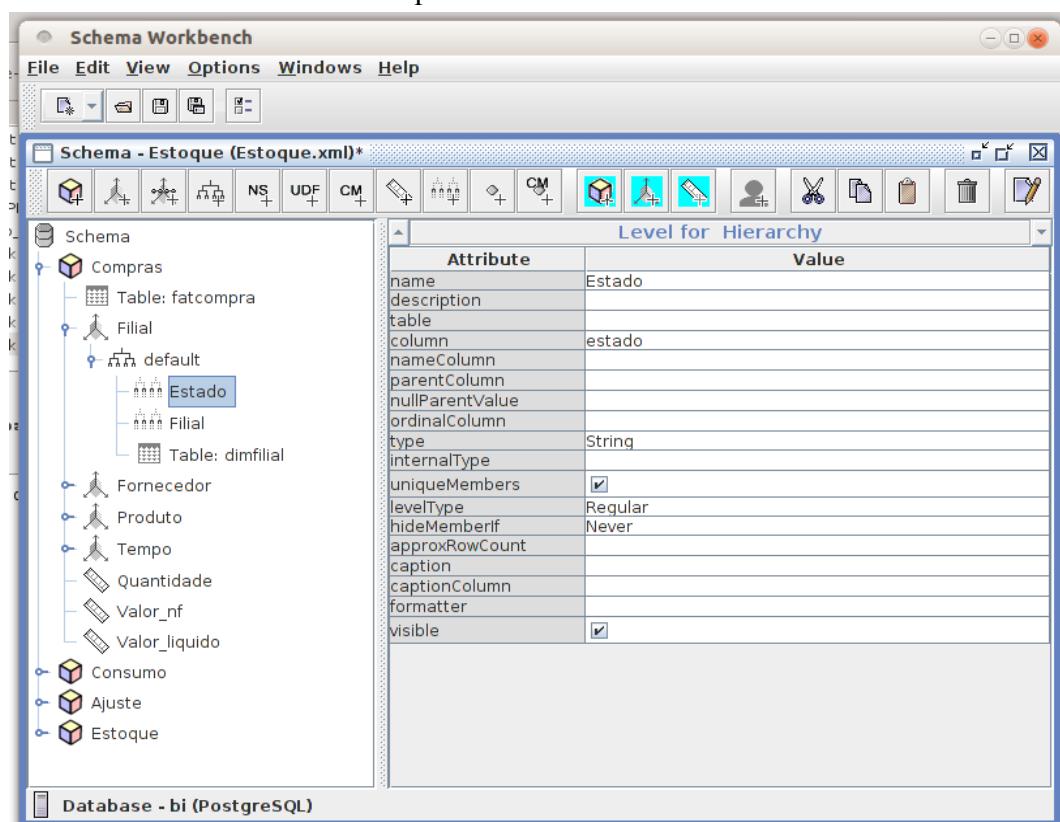


Fonte: Elaborado pelo autor.

Outra configuração importante a ser realizada é a alteração do atributo “*primaryKey*” na hierarquia, este parâmetro é responsável por informar a hierarquia qual é a coluna da tabela no banco de dados que é a chave primária.

Com a hierarquia configurada, o próximo passo é adicionar os níveis. Isso pode ser feito utilizando a barra de ferramentas ou o botão direito do *mouse* sobre a hierarquia selecionando a opção “*Add Level*”. A Figura 23 apresenta a tela da ferramenta com os atributos do nível disponíveis para edição.

Figura 23 – Atributos do nível hierárquico.



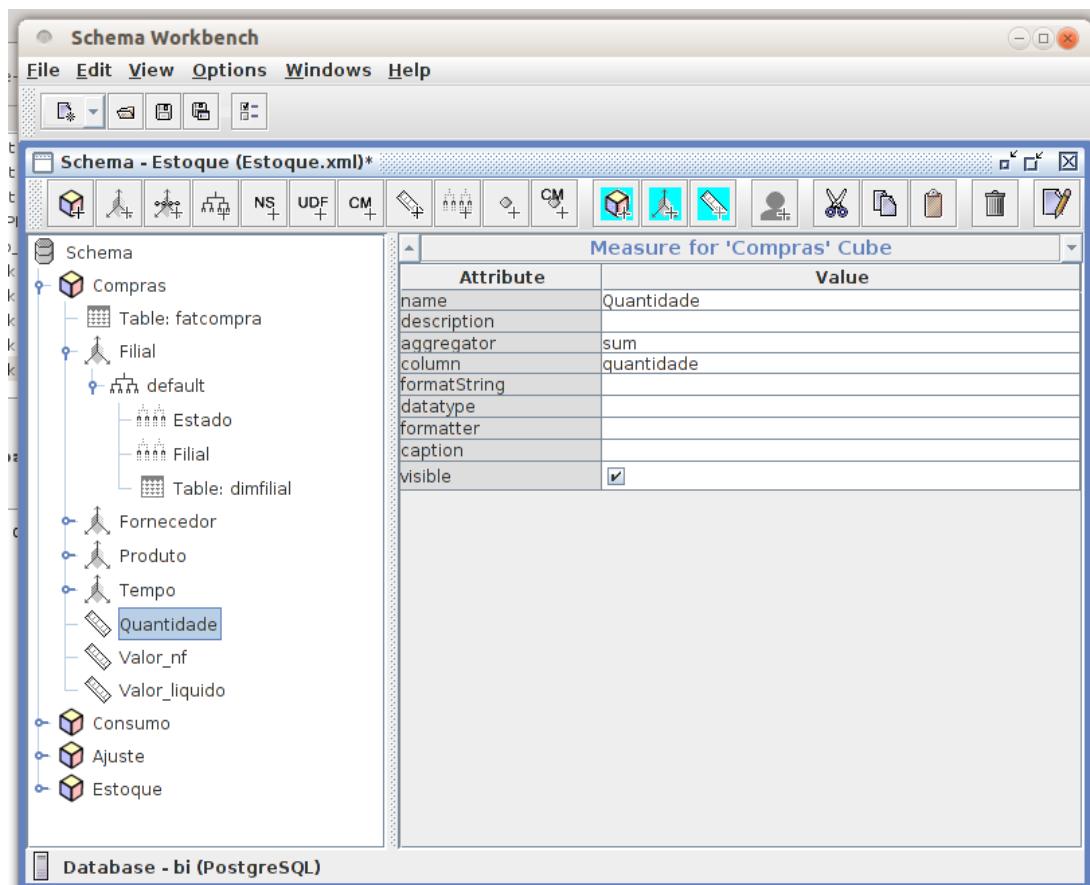
Fonte: Elaborado pelo Autor.

Para cada nível hierárquico adicionado, os seguintes atributos devem ser alterados:

- **Name:** Indica o nome do nível. O nome informado neste parâmetro é o que vai aparecer na ferramenta de análise;
- **Column:** Indica qual coluna da tabela na BD vai ser vinculada a este nível;
- **Type:** Indica o tipo de dados que está armazenado no BD;
- **LevelType:** Indica o tipo de nível que está sendo criado. Neste ponto o atributo *type* da dimensão faz diferença. Caso este atributo tenha sido configurado como “*StandardDimension*”, a única opção válida é “*Regular*”, caso contrário, em que o *type* for “*TimeDimension*”, deve-se escolher o parâmetro que melhor se enquadra ao conteúdo da coluna da tabela.
- **HideMemberIf:** Indicando em que condições o nível da hierarquia deve ser oculto.

O último tipo de objeto a ser incluído do cubo é a métrica, isso pode se feito através da barra de ferramentas ou pelo botão direito do *mouse* sobre o cubo selecionando a opção “*Add Measure*”. Na Figura 24 encontra-se representado a métrica com seus atributos.

Figura 24 – Atributos da métrica.



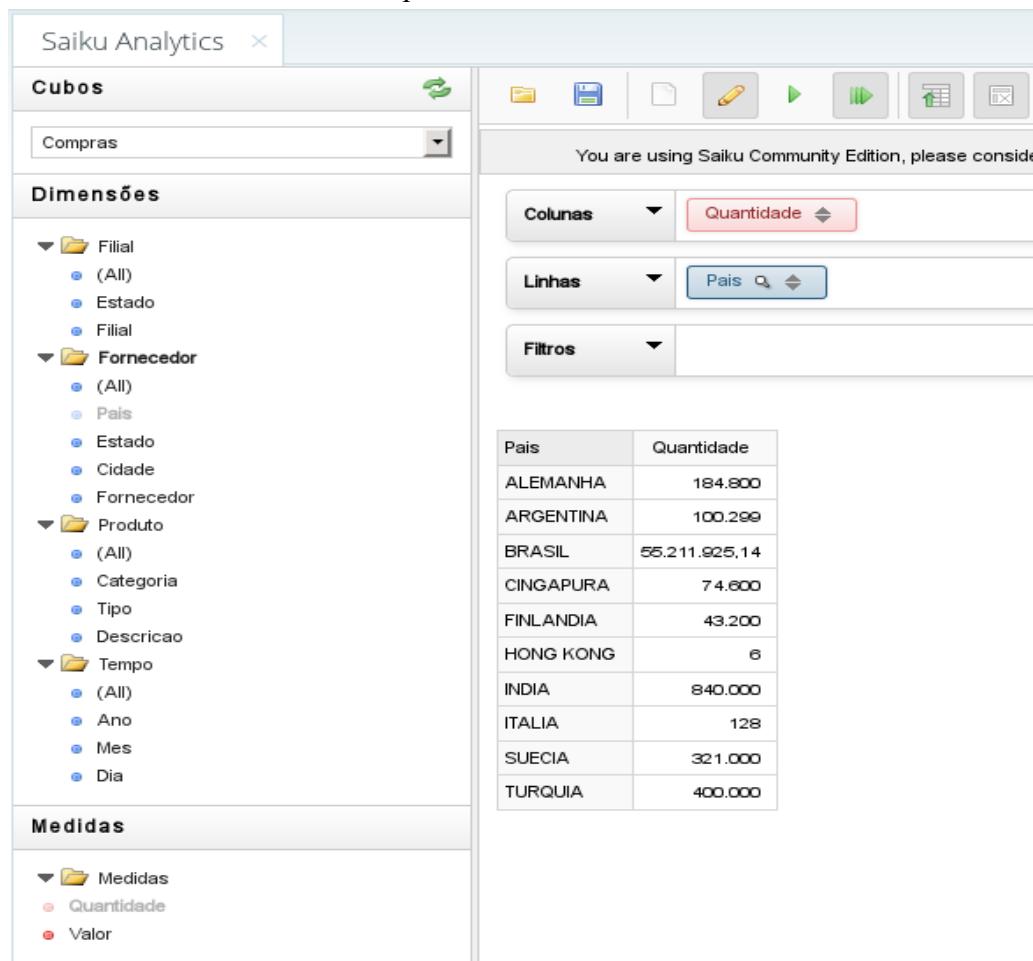
Fonte: Elaborado pelo autor.

Os atributos a serem alterados nesta etapa são:

- **Name:** Representa o nome da métrica;
- **Aggregator:** indica qual a função de agrupamento que será executado sobre a métrica durante o processo de análise. Dentre as funções podemos citar: soma, média, máximo, mínimo, entre outras;
- **Column:** Indica qual é o nome da coluna da tabela fato associada a métrica.
- **Data Type:** Indica o tipo de dado que está armazenado no Banco de Dados.

Em todos os objetos configurados existe o atributo “visible” que indica se o objeto vai ser apresentado na ferramenta de análise. A figura 25 apresenta a ferramenta de análise Saiku com o cubo de compras selecionado para análise.

Figura 25 – Saiku com o cubo de compras.

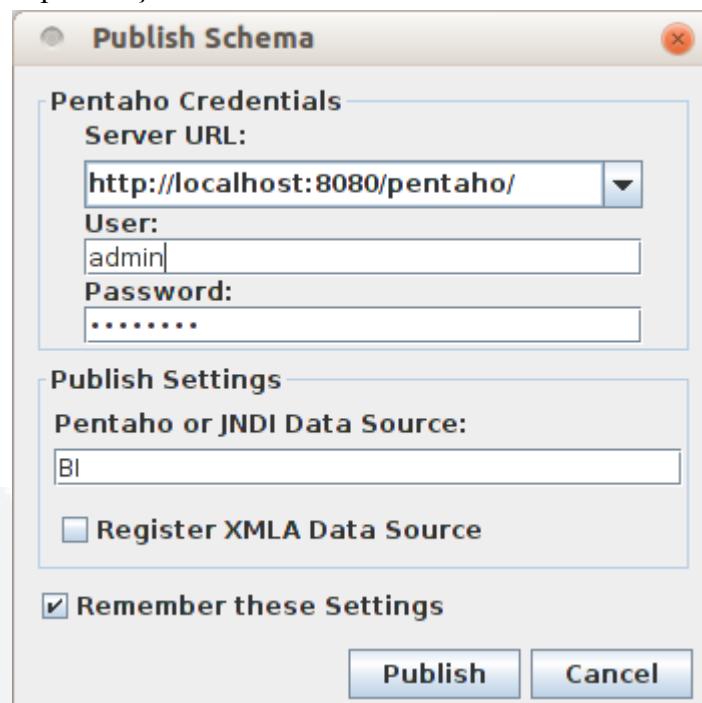


Fonte: Elaborado pelo autor.

Para que o cubo esteja disponível para análises, relatórios e painéis, este deve ser publicado no Pentaho. Para realizar esta operação, após salvar todas as alterações, o usuário deve clicar em *file* e selecionar a opção *Publish*. Ao realizar este procedimento será apresentada a tela representada pela Figura 26. Onde o usuário deve informar o endereço do servidor, usuário e senha, pra realizar a publicação.

Outro passo muito importante a ser realizado é a limpeza dos caches do servidor, após cada publicação, para que o servidor utilize a versão mais recente do arquivo publicado. Este processo é realizado pela tela principal do Pentaho, acessando o menu “Ferramentas” e após a opção “Atualizar”, clicando em seguida na opção “limpar cache do mondriam”.

Figura 26 – Tela de publicação.



Fonte: Elaborado pelo autor.

5.1.2 Criação dos dashboards

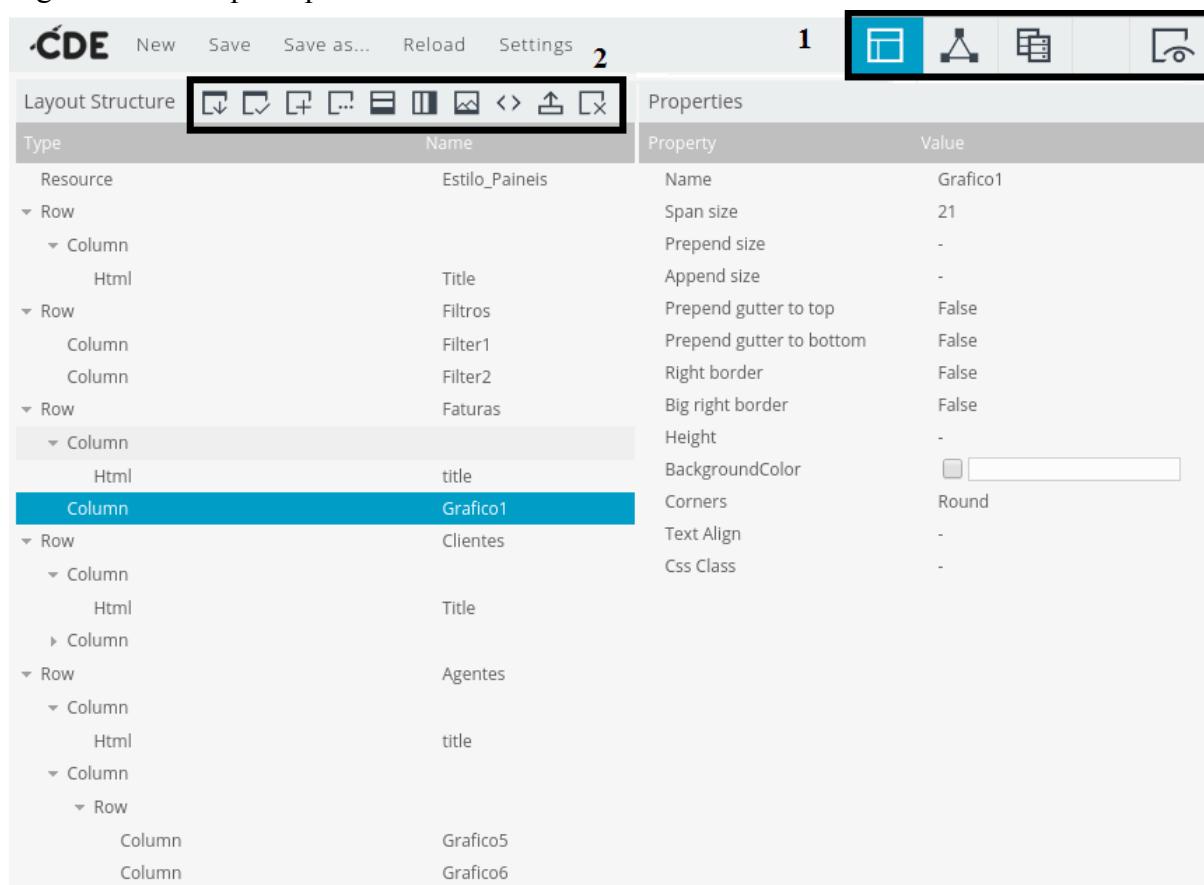
Com os cubos configurados e publicados, o Pentaho já está pronto para ser utilizado para criação de análises, porém um recurso da ferramenta ainda pode ser explorado, que é a utilização de *dashboards* ou painéis. Estes são responsáveis pelo monitoramento das informações registradas no *Data Warehouse* através de inúmeros recursos, como gráficos, tabelas, mapas, entre outros.

Para a criação dos painéis disponibilizados no presente trabalho foi utilizado a ferramenta CDE, esta é uma ferramenta livre desenvolvida pela *webdetails*⁸ e está disponível no *Marketplace*.

Após a instalação da ferramenta no Pentaho e a reinicialização do servidor, a opção de criar painéis já estará disponível no menu “*Create New*”, na tela principal do ambiente. A opção a ser escolhida é *CDE Dashboard*, ao realizar esta operação a tela representada pela Figura 27 será apresentada, porém sem nenhuma configuração de *Layout*.

⁸ Mais detalhes e demonstrações estão disponíveis no endereço <http://www.webdetails.pt/>.

Figura 27 – Tela principal do CDE.



Fonte: Elaborado pelo autor.

Na Figura 27, na parte superior esquerda pode-se ver os botões responsáveis por salvar e configurar o painel. No lado direito, marcado pelo quadro 1, encontram-se os botões para alternância entre o layout do painel, configuração dos gráficos e outros objetos do painel, a criação das consultas no banco de dados por último temos o botão responsável pela pré-visualização do painel, que pode ser realizada a qualquer momento do desenvolvimento.

A Figura 27 apresenta o *layout* desenvolvido para os painéis criados para o presente trabalho. A ferramenta permite que o layout possa ser salvo como modelo, reduzindo assim o trabalho para criação de novos painéis.

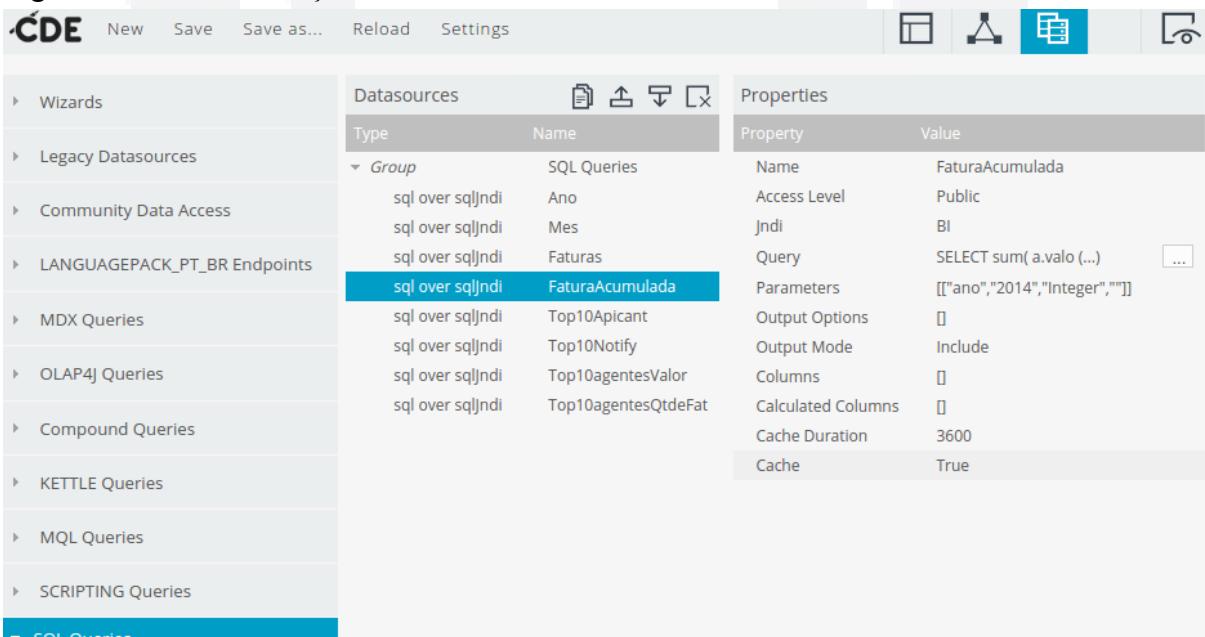
A tela de criação do *layout* é dividida em duas partes, onde a da esquerda apresenta os objetos criados, e a da direita os atributos do objeto selecionado. Para inserir novos objetos no *layout* utiliza-se os botões marcados pelo quadro 2, na Figura 27. Os botões têm diversas funções, como adicionar nova linha, nova coluna, entre outras. A cada tipo de objeto selecionado, a configuração e quantidade de botões é diferente.

No quadro da direita pode-se alterar as propriedades dos objetos inseridos no *layout*, entre as disponíveis, as seguintes têm uma maior relevância.

- **Name:** Representa o nome do objeto. Esta propriedade é utilizada para referenciar o local onde um gráfico será apresentado;
- **Span size:** Representa a largura do objeto;
- **Height:** Representa a altura do objeto, caso não seja informado um valor para esta propriedade, o sistema adequará a altura do objeto ao seu conteúdo;
- **BackgroundColor:** Representa a cor de fundo do objeto.

Com o *layout* definido, o próximo passo é a criação das consultas no BD. A tela de criação das conexões é dividida em três partes, onde temos: os componentes disponíveis para utilização, os componentes adicionados ao painel e por último as propriedades do objeto selecionado. A Figura 28 apresenta a tela de criação das consultas.

Figura 28 – Tela de criação das consultas.



Datasources		Properties	
Type	Name	Property	Value
Group	SQL Queries	Name	FaturaAcumulada
sql over sqlJndi	Ano	Access Level	Public
sql over sqlJndi	Mes	Jndi	BI
sql over sqlJndi	Faturas	Query	SELECT sum(a.valor ...)
sql over sqlJndi	FaturaAcumulada	Parameters	[{"ano": "2014", "Integer": ""}]
sql over sqlJndi	Top10Aplicant	Output Options	
sql over sqlJndi	Top10Notify	Output Mode	Include
sql over sqlJndi	Top10agentesValor	Columns	
sql over sqlJndi	Top10agentesQtdeFat	Calculated Columns	
		Cache Duration	3600
		Cache	True

Fonte: Elaborado pelo autor.

Os componentes disponíveis estão organizados em grupos. Para o desenvolvimento do presente trabalho foi utilizado apenas o grupo *SQL Queries*, onde é possível criar consultas utilizando apenas comandos SQL. Ao selecionar o componente desejado, este é inserido na parte central da página. A partir desse momento suas propriedades serão disponibilizadas para edição. As propriedades alteradas para a criação das consultas utilizadas neste trabalho foram:

- **Name:** Representa o nome da consulta, através deste que é feito a ligação com o componente que vai ser exibido no painel;
- **JNDI:** Representa o nome da conexão com o Banco de Dados;
- **Query:** Representa o código SQL responsável pela obtenção dos dados;
- **Parameters:** Representa os parâmetros utilizados como filtro de dados para a consulta SQL.

Com as consultas criadas, o próximo passo é a criação dos gráficos e parâmetros de filtro do painel, quando estes existirem. A tela de criação e configuração é semelhante a vista na Figura 28, como pode ser verificado na Figura 29. A separação em três também existe e mantém a mesma configuração da tela de criação de consultas.

Figura 29 – Tela de configuração dos gráficos.

Type	Name	Property	Value
Group	Selects	Name	Fatura
Group	Select Component	Title	Fatura X dia
Group	Select Component	Listeners	['Ano', 'Mes']
Group	Generic	Parameters	[{"ano", "Ano"}, {"mes", "Mes"}]
Simple parameter	Ano	Datasource	Faturas
Simple parameter	Mes	Width	-
Simple parameter	FaturaAcumulada	Height	250
Group	Charts	HtmlObject	Grafico1
CCC Line Chart	Fatura	clickable	False
CCC Bar Chart	TopAplicant	clickAction	
CCC Bar Chart	TopNotify	compatVersion	2
CCC Bar Chart	TopAgenteValor	crosstabMode	True
CCC Bar Chart	TopAgenteQtdeFat	legend	True
		seriesInRows	False
		timeSeries	False
		timeSeriesFormat	%Y-%m-%d

Fonte: Elaborado pelo autor

Os componentes utilizados nos painéis desenvolvidos para o presente trabalho pertencem aos seguintes grupos:

- **Generic:** Componente *simple parameter*. Pode ser comparado com uma variável, utilizada para registrar o conteúdo selecionado em um filtro e envia para o gráfico;
- **Selects:** Componente *Select Component*. Cria um *combobox* no painel, utilizado para filtrar as informações que serão apresentadas. Aqui, deve se alterar o nome, e indicar

qual o componente *simple parameter* que vai receber o seu conteúdo e a consulta responsável pelos dados apresentados no componente;

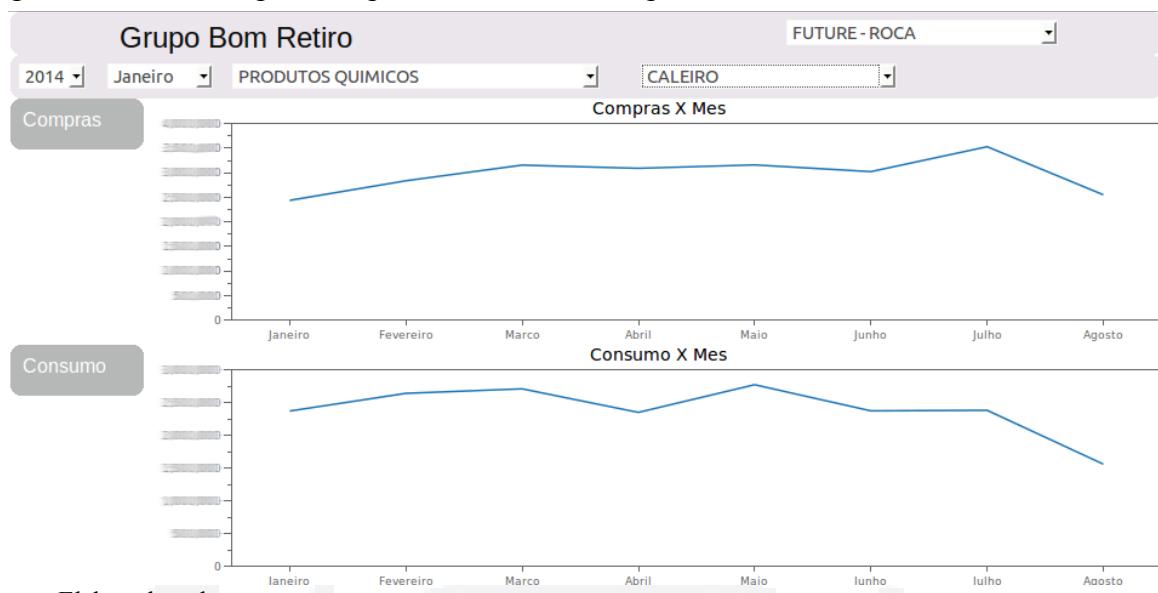
- **Charts:** Os componentes utilizados foram: *bar chart*, *pie chart* e *line chart*. Para todos estes componentes, as seguintes propriedades foram alteradas: *Name* (representa o nome do objeto), *Title* (representa o título do gráfico, onde este título será apresentado no painel), *Listeners* (representa quais são os parâmetros que o gráfico vai ficar “ouvindo”, ou seja, assim que ocorrer uma alteração em seu conteúdo o gráfico reexecuta a consulta no Banco de Dados e altera o seu conteúdo), *Parameters* (representa quais componentes do tipo *simple parameter* serão repassados para a consulta), *DataSource* (representa o nome da consulta responsável pelos dados apresentados no gráfico) e *HtmlObject* (representa o nome do objeto do *layout* onde o componente será exibido).

5.1.3 Dashboards criados

Após a definição dos cubos, foram elaborados alguns painéis, sendo que estes são responsáveis pela visualização dos dados contidos no *Data Warehouse* de uma maneira sumarizada, através de uma representação gráfica, com a utilização de elementos visuais, como mapas, gráficos, relógios de desempenho, entre outros. Para a realização deste trabalho foram desenvolvidos dois painéis, um para cada área de negócio modelada.

Na Figura 30, encontra-se a primeira parte do painel responsável pelo monitoramento das informações da área de compras. Este painel possui dois gráficos de linha que apresentam as informações relativas as compras e consumos de materiais na organização. Estes gráficos sofrem influência apenas do ano selecionado, já que apresentam as informações totalizadas mensalmente.

Figura 30 – Primeira parte do painel da área de compras.



Fonte: Elaborado pelo autor.

Este painel ainda possui mais seis gráficos onde estes representam as informações de ajustes e requisições manuais, como pode ser observado na Figura 31. Os gráficos estão organizados da seguinte maneira: um gráfico de barras e um em formato de pizza para ajustes apresentando os valores em quantidade, um gráfico de barras e um de pizza para ajustes apresentando as informações em valor, e por fim, dois gráficos de barras para as informações de requisições manuais, sendo um para quantidade e outro para valores. Esses últimos gráficos sofrem influência de todos os filtros do painel.

Figura 31 – Segunda parte de painel da área de compras



Fonte: Elaborado pelo autor.

Na Figura 32, encontra-se o painel responsável pelo monitoramento das informações da área comercial. Este painel possui um gráfico de linha que apresenta as informações relativas as faturas de exportação e a comissão em dólares totalizado por dia.

Este painel ainda possui mais quatro gráficos que representam as informações de maiores clientes e representantes, organizados da seguinte maneira: um gráfico de barras apresentando os valores para os 10 maiores *applicants*, um gráfico de barras apresentando as informações em valores para os 10 maiores *notify*, e, por fim, dois gráficos de barras para as informações dos 10 maiores representantes, sendo um apresentando os maiores valores de comissão e outro apresentando a quantidade de faturas realizadas. Para esse painel os filtros alteram as informações em todos os gráficos.

Figura 32 – Painel da área comercial.



5.2 Disponibilização

O serviço foi disponibilizado em uma máquina de pequeno poder de processamento, utilizada apenas para os testes realizados para a execução deste trabalho, com acesso liberado apenas dentro da organização através do endereço 192.168.0.210:8080. Então, realizou-se um breve treinamento com os usuários, no qual foram seguidos os seguintes passos:

1. Realização de uma breve apresentação da ferramenta;

2. Apresentação das *Dashboards* criadas para cada usuários;
3. Apresentação da ferramenta de análise *Saiku*;
4. Apresentação do navegador de arquivos disponíveis no ambiente, exemplificando como uma análise poderia ser salva e recuperada posteriormente.

Por questão de disponibilidade de tempo dos usuários, o primeiro a ser treinado foi o responsável pelo setor de compras, seguido então pelo responsável pelo setor comercial. Após o treinamento, os usuários tiveram 14 dias para experimentar a ferramenta e elucidar eventuais dúvidas sobre o ambiente, sendo que ao término deste prazo os usuários foram submetidos a um questionário de avaliação.

5.3 Questionário de avaliação

Para a avaliação da ferramenta foi utilizado um questionário com oito perguntas qualitativas e seis quantitativas, disponível no Apêndice C. Foram entrevistadas duas pessoas da empresa que utilizaram a ferramenta desenvolvida, sendo uma da área de compras e outra da área comercial. Para melhor trabalhar as respostas, chamaremos os entrevistados de E1 e E2. A seguir será apresentada uma análise dos resultados das avaliações.

A primeira questão sobre o *Data Warehouse*, visa saber se o sistema atende a área de negócio, sendo que para o ambos os entrevistados, atende totalmente.

O entrevistado E1 visualiza a ferramenta como totalmente contributiva para a tomada de decisões, já o E2 visualiza a ferramenta como parcialmente contributiva.

Ambos os entrevistados acham totalmente adequada a forma de apresentação das informações. O entrevistado E1 considera a forma de criação de análises totalmente adequada, já o entrevistado E2 considerou como sendo parcialmente adequada.

Quanto à utilidade do DW no desempenho das funções dos entrevistados, ambos acreditam que contribui totalmente para o desempenho de suas funções. Ambos os entrevistados acharam as informações são totalmente fiéis as informações disponíveis no

ERP, assim como ambos acreditam totalmente que o ambiente pode ser utilizado em um cenário real e possui um tempo de resposta totalmente aceitável.

Sobre as melhorias que as novas informações trazem ao dia a dia, tanto em relação à diferença para o sistema atual quanto à facilidade de acesso a informação, E1 acredita que o acesso das informações tornou-se mais ágil em relação ao sistema atual possibilitando assim que uma ação corretiva possa ser tomada antes do fechamento do período. Já E2 acredita que a possibilidade de escolha das informações necessárias e a possibilidade de escolha da forma de visualização das informações são as principais melhorias.

Sobre eventuais problemas que impediriam a utilização da ferramenta, E1 acredita que o ambiente possa ser utilizado apenas para áreas de negócio onde as informações são instantâneas, como os movimentos de estoque relacionados a produtos químicos. Já para E2 não existe nenhum problema.

Quanto a expansão das informações na área de negócio do entrevistado, E2 acredita que as informações de pagamentos e recebimentos diários e as informações de custos poderiam ser incorporadas ao ambiente. Já E1 acredita que as informações disponibilizadas no ambiente contemplam totalmente a área de compras.

Sobre a expansão do ambiente para outras áreas de negócio da organização, E1 acredita que os setores comercial e programação de produção seriam beneficiados com o ambiente, já para E2, os setores financeiro e contábil seriam beneficiados.

Como sugestão de melhoria para o ambiente, E1 sugeriu a criação de novos relatórios para visualização de dados, com relação as maiores compras por fornecedor e produtos com maior valor comprado, porém não especificou se estas melhorias são essenciais para a utilização do ambiente. Já E2 sugeriu um “menu de opções de mais fácil entendimento”, porém deixa claro que com o treinamento adequado isso não chega a ser um problema, tornando-se então uma melhoria desejável apenas.

As respostas obtidas junto aos entrevistados podem ser observadas nos Apêndices D e E do presente trabalho.

6 CONCLUSÕES

As tomadas de decisões motivam, impulsionam e direcionam o andamento das organizações, tanto no controle diário das operações, como no planejamento estratégico. A informação, devido a sua importância no ambiente empresarial, acaba se tornando um dos ativos mais preciosos, pois com ela os gestores podem tomar as decisões fundamentadas na real situação da organização. Os sistemas transacionais operacionalizam somente o básico, porém, é necessário visualizar os dados de negócio sob diferentes aspectos, com este propósito surgem os sistemas de apoio a decisão que baseiam-se em *Data Warehouse*. Estes são responsáveis pela transformação dos dados brutos em informação e depois em conhecimento, desta maneira, os dados brutos da organização são capazes de agregar valor ao negócio.

Com o presente trabalho, foi possível verificar a importância e validade dos conceitos relacionados, a importância da construção de um *Data Warehouse* para agregar dados voltados para a tomada de decisão. A ferramenta apresentada mostrou-se adequada, oferecendo recursos valiosos referentes à análise dos dados, porém, é necessário que os gestores conheçam os seus indicadores de desempenho e que passem esta informação aos projetistas de *Business Intelligence* para que estes consigam projetar o sistema para responder as suas perguntas de maneira eficiente. Com o desenvolvimento do presente trabalho, foi possível verificar a carência que os gestores possuem de informações sumarizadas, pois, os sistemas utilizados disponibilizam aos gestores muitos dados, exigindo assim um grande esforço desses para obter a informação desejada.

A execução deste trabalho foi de acordo com a metodologia apresentada. Todas as etapas foram cumpridas com sucesso, e além disso a sua aplicação trouxe resultados positivos na organização, motivando uma continuidade da aplicação da ferramenta mesmo após o desenvolvimento do presente trabalho, pois ela, segundo a avaliação dos gestores, mostrou-se promissora, com grandes possibilidades dentro da organização. Os gestores entrevistados para a avaliação do ambiente sugeriram a implantação nas áreas, contábil, de produção, financeira e comercial, sendo que, esta última, já está parcialmente implementada, restando apenas a integração dos dados referentes a negociações nacionais. Outras áreas que poderiam se beneficiar com o ambiente de BI são o setor de custos, recursos humanos e transportes, sendo que este último já possui solicitação formal para ser integrado ao ambiente. Com os resultados positivos alcançados, pretende-se dar continuidade e ampliar o uso da solução dentro da organização.

REFERÊNCIAS

- ARAÚJO, Erika Maria Teixeira; BATISTA, Mônica de Lourdes Souza; MAGALHÃES, Teresinha Moreira de. **Um estudo sobre as ferramentas OLAP**. Disponível em: <<http://www.devmmedia.com.br/um-estudo-sobre-as-ferramentas-olap/6691>>. Acesso em Maio de 2014.
- COLACO JUNIOR, Methanias; ALMEIDA, Maria de Fatima – colab; NASCIMENTO, Andre Vinicius – colab. Projetando sistemas de apoio a decisão baseados em data warehouse. Rio de Janeiro: Axcel, 2004.
- FREITAS, Henrique et al. **Informação e decisão**: sistemas de apoio e seu impacto. Porto Alegre: Ortiz, 1997.
- GERHARDT, Tatiana Engel; SILVEIRA, Denise Tolfo. Métodos de Pesquisa. 1A Edição. Porto Alegre: UFRGS, 2009.
- KIMBALL, Ralph; CASERTA, Joe. **The data warehouse ETL toolkit**: practical techniques for extracting, cleaning, conforming and delivering data. São Paulo: Wiley, 2004.
- KIMBALL, Ralph; ROSS, Margy. **The data warehouse toolkit**: the complete guide to dimensional modeling. 2. ed. São Paulo: Wiley Computer, 2002.
- LEME FILHO, Trajano. Business intelligence no Microsoft Excel. Rio de Janeiro: Axcel Books, 2004.
- PENTAHO. Disponível em <<http://www.pentaho.com>>. Acesso em: 20 out. 2014.
- PHP. Disponível em: <<http://php.net>>. Acesso em: 27 set. 2014.
- POSTGRESQL. Disponível em: <<http://www.postgresql.org>>. Acesso em: 27 set. 2014.
- PRIMAK, Fábio Vinícius. **Quais são os Benefícios que o BI nos traz?**. 2009. Disponível em: <http://www.oficinadanet.com.br/artigo/2143/quais_sao_os_beneficios_que_o_bi_nos_traz>. Acesso em Maio de 2014.
- PRIMAK, Fábio Vinícius. **A Evolução do Business Intelligence**, 2009. Disponível em: <http://www.oficinadanet.com.br/artigo/2144/a_evolucao_do_business_intelligence>. Acesso em Maio de 2014.

QLIK. Disponível em: <<http://www.qlikview.com/>>. Acesso em: 20 out. 2014.

SPAGOBI. Disponível em: <<http://www.spagobi.org/>>. Acesso em 20 de out. 2014.

VENTURA, Magda Maria. **O Estudo de Caso como Modalidade de Pesquisa.** Disponível em: <http://www.polo.unisc.br/portal/upload/com_arquivo/o_estudo_de_caso_como_modalidade_de_pesquisa.pdf>. Acessado em Maio de 2014

WAINER, Jacques. **Métodos de pesquisa quantitativa e qualitativa para a Ciência da Computação.** In: KOWALTOWSKI, Tomasz; BREITMAN, Karin; organizadores. Atualizações em Informática 2007. Rio de Janeiro: Ed. PUC-Rio; Porto Alegre: Sociedade Brasileira de Computação, 2007.

YIN, Robert K. **Estudo de caso:** planejamento e métodos. 4. ed. Porto Alegre: Bookman, 2010.

APÊNDICES

Apêndice A – Questionário de levantamento de requisitos.

Questionário

1 Que tipo de decisões você toma no dia a dia e que impactos elas têm na organização?

2 Quais os relatórios que você utiliza no seu dia a dia e quais informações estão disponíveis neles?

3 As informações disponíveis nestes relatórios são suficientes para você? (Sim, Não). Se não aponte as principais deficiências e limitações deste relatórios.

4 Que tipo de análises, simulações, agrupamentos de informações você gostaria de ter e o sistema/relatórios não fornecem?

5 Que outras informações você usa ou gostaria de usar para enriquecer a análise dos dados?

6 Para cada grupo de informações que você considera importante para tomada de decisão preencha a seguir:

a – Exemplo: Venda.

b – O que? Exemplo: Produto.

c – Quando? Exemplo: Dia/mês/ano

d – Quem? Exemplo: Cliente

e – Onde? Exemplo: Cidade do cliente/loja

f – Como? Exemplo: Promoção?

g – Medida. Exemplo: Valor e quantidade vendidos.

7 Quais são os níveis de detalhamento que você precisa para analisar a informação? Exemplo: País->Estado->Cidade, Ano->semestre->mês->dia. (uma resposta para cada subitem 6).

8 Você costuma acompanhar a evolução histórica dos dados?

Se sim, quanto tempo você costuma analisar?

Apêndice B – Processo ETL

```
#!/usr/bin/php
<?php
    date_default_timezone_set( 'America/Sao_Paulo' );
    require_once( 'funcoes.php' );
    // Database connection
    $connDest = NULL;
    $result = false;
    // executar o script ./etl.php passwordDes="senha"
    for ( $i=1; $i< $argc; $i++ ) { parse_str( $argv[ $i ] ); }
    // conecta no banco de destino
    $connDest = pg_connect( "host='127.0.0.1' dbname='bi' user='leandro' password='".$passwordDes.""" );
    if ( !$connDest )
        exit;
    // Configurações
    $periodo = busca( $connDest, "Estoque" );
    $dataIni = $periodo['DataInicial'];
    $dataFina = $periodo['DataFinal'];

    $ipOrigem = getConfig( $connDest, "ipOrigem" );
    $bancoOrigem = getConfig( $connDest, "bancoOrigem" );
    $usuarioOrigem = getConfig( $connDest, "usuarioOrigem" );

    // Cria conexão entre com o banco de destino
    $result = pg_query( $connDest, "SELECT
        dblink_connect( 'conexao', 'hostaddr='".$ipOrigem." dbname='".$bancoOrigem." user='".$usuarioOrigem."' ) );
    if ( $result )
    {
        // inicia transação para controle atômico
        pg_query( $connDest, 'BEGIN' );
        // Cria tabela temporária
        $sql = "CREATE TEMP TABLE dimFilialtemp AS SELECT * FROM
        dblink( 'conexao', '' );
        $sql .= 'SELECT a.codigoint, trim(a.fant), trim(a.desc), a.cnpj,
        trim(c.nome), trim(d.desc) as estado FROM dfiliais a ';
        $sql .= "INNER JOIN dsce200 b ON ( b.codfilial=0 AND b.deleted IS
        NULL AND a.cliente=b.forn ) ";
        $sql .= "INNER JOIN dsce045 c ON ( c.cdcidade = b.cdcidade ) ";
        $sql .= "INNER JOIN dscp060 d ON ( b.uf=d.abre ) ";
        $sql .= "WHERE a.deleted IS NULL AND a.codigoint IN ( ";
        $sql .= "SELECT DISTINCT codfilial FROM dsce400 WHERE deleted IS NULL
        AND data BETWEEN '".dataIni."' AND '".dataFina."' AND oper:int IN
        ( 10, 17, 21 ) ";
        $sql .= " ) AS resultado( codigo int, fantasia varchar( 45 ),
        descricao varchar(50), cnpj char(18), cidade varchar(30), estado
        varchar(30) );
        Logger( "Sql Filiais: ".$sql."\n" );
        pg_query( $connDest, $sql );
        // Insere filiais
        $sql = "INSERT INTO dimfilial ( codigo, fantasia, descricao, cnpj,
        cidade, estado ) ";
        $sql .= "SELECT * FROM dimFilialtemp WHERE codigo NOT IN ( SELECT
        codigo FROM dimfilial ) ";
        pg_query( $connDest, $sql );
        // atualiza dimensão filial
        $sql = "UPDATE dimfilial a SET fantasia=b.fantasia,
```

```
descricao=b.descricao, cnpj=b.cnpj, cidade=b.cidade, estado=b.estado ";
    $sql .= "FROM dimFilialtemp b WHERE a.codigo = b.codigo ";
    pg_query( $connDest, $sql );
    // elimina tabela temporária
    pg_query( $connDest, "DROP TABLE dimFilialtemp" );

    // Cria tabela temporária
    $sql = "CREATE TEMP TABLE dimProdutotemp AS SELECT * FROM
dblink( 'conexao', ' ';
    $sql .= 'SELECT a.redu, trim(a."desc"), a.estrutural, a.clfi, a.unid,
b.nivel, c.nivel, d.nivel FROM dsce100 a ';
    $sql .= "INNER JOIN dscp067 b ON ( substr( a.estrutural, 1, 5 ) =
substr( b.codigo, 1, 5 ) AND b.grau='2' AND b.deleted IS NULL ) ";
    $sql .= "INNER JOIN dscp067 c ON ( substr( a.estrutural, 1, 8 ) =
substr( c.codigo, 1, 8 ) AND c.grau='3' AND c.deleted IS NULL ) ";
    $sql .= "LEFT JOIN dscp067 d ON ( substr( a.estrutural, 1, 11 ) =
substr( d.codigo, 1, 11 ) AND d.grau='4' AND d.deleted IS NULL ) ";
    $sql .= "WHERE a.deleted IS NULL AND a.codfilial=0 AND a.redu IN ( ";
    $sql .= "SELECT DISTINCT redu FROM dsce400 WHERE deleted IS NULL AND
data BETWEEN '".dataIni."' AND '".dataFina."' AND oper IN ( '10',
'17', '21' ) ) ) ";
    $sql .= "AS resultado( re char( 6 ), de varchar( 45 ), es varchar( 26 ),
cl char( 14 ), un char( 3 ), n1 varchar( 34 ), n2 varchar( 34 ), n3
varchar( 34 ) )";
    Logger( "Sql Produtos: ".$sql."\n" );
    pg_query( $connDest, $sql );
    // Insere filiais
    $sql = "INSERT INTO dimproduto ( codigo, descricao, estrutural, ncm,
unidade, categorian1, categorian2, categorian3 ) ";
    $sql .= "SELECT * FROM dimProdutotemp WHERE re NOT IN ( SELECT codigo
FROM dimproduto ) ";
    pg_query( $connDest, $sql );
    // atualiza dimensão Produto
    $sql = "UPDATE dimproduto a SET descricao=b.de, estrutural=b.es,
ncm=b.cl, unidade=b.un, categorian1=b.n1, categorian2=b.n2,
categorian3=b.n3 ";
    $sql .= "FROM dimProdutotemp b WHERE a.codigo = b.re ";
    pg_query( $connDest, $sql );
    // elimina tabela temporária
    pg_query( $connDest, "DROP TABLE dimProdutotemp" );

    // Cria tabela temporária
    $sql = "CREATE TEMP TABLE dimFornecedorTemp AS SELECT * FROM dblink(
'conexao', ' ';
    $sql .= 'SELECT a.forn, trim(a.nome), trim( a.fant ), CASE WHEN empty(
a.cgc ) THEN a.cpf ELSE a.cgc END, ';
    $sql .= 'trim( CASE WHEN empty(a.cdcidade) THEN a.cida ELSE b.nome
END ), trim( c."desc" ) AS estado, trim(d."desc" ) AS pais, trim(d.descing)
AS paisI FROM dsce200 a ';
    $sql .= "LEFT JOIN dsce045 b ON ( a.cdcidade=b.cdcidade ) ";
    $sql .= "INNER JOIN dscp060 c ON ( a.uf=c.abre ) ";
    $sql .= "INNER JOIN dsce220 d ON ( a.pais=d.cod ) ";
    $sql .= "WHERE a.deleted IS NULL AND a.codfilial=0 AND a.forn IN ( ";
// a.tipo ILIKE '%F%' AND
    $sql .= "SELECT DISTINCT forn FROM dsce400 WHERE deleted IS NULL AND
data BETWEEN '".dataIni."' AND '".dataFina."' AND oper='10' ) ";
    $sql .= "AS resultado( forn char(5), no varchar(58), fa
varchar(18), cnpj char(18), ci varchar(30), es varchar(30), pais
varchar( 40 ), paisIng varchar( 40 ) )";
    Logger( "Sql Fornecedores: ".$sql."\n" );
```

```
    pg_query( $connDest, $sql );
    // Insere filiais
    $sql = "INSERT INTO DimFornecedor ( codigo, nome, fantasia, CNPJ,
cidade, estado, pais, PaisIngles ) ";
    $sql .= "SELECT * FROM dimFornecedorTemp WHERE forn NOT IN ( SELECT
codigo FROM DimFornecedor ) ";
    pg_query( $connDest, $sql );
    // atualiza dimensao Fornecedor
    $sql = "UPDATE DimFornecedor a SET nome=b.no, fantasia=b.fa,
CNPJ=b.cnpj, cidade=b.ci, estado=b.es, pais=b.pais, PaisIngles=b.paisIng ";
    $sql .= "FROM dimFornecedorTemp b WHERE a.codigo = b.forn ";
    pg_query( $connDest, $sql );
    // elimina tabela temporária
    pg_query( $connDest, "DROP TABLE dimFornecedorTemp" );

    // Busca dados de consumo e compra no estoque
    // Cria tabela temporária
    $sql = "CREATE TEMP TABLE compraConsumoTemp AS SELECT * FROM dblink(
'conexao', ' ";
    $sql .= 'SELECT codfilial, oper, data, redu, forn, desp, qtde, vlnf,
valr FROM dsce400 ';
    $sql .= "WHERE deleted IS NULL AND data BETWEEN '" . $dataIni . "' AND
'" . $dataFina . "' AND ( oper::int=10 OR oper::int=17 OR oper::int=21 )
ORDER BY 3' ) ";
    $sql .= "AS resultado( fl int, op char(2), data date, re char(6),
forn char(5), desp char(20), qtde numeric, valorNF numeric, valor
numeric )";
    Logger( "Sql Consumos e compras: ".$sql."\n" );
    pg_query( $connDest, $sql );

    // Insere datas
    $result = pg_query( $connDest, 'SELECT DISTINCT data FROM
compraConsumoTemp ORDER BY data' );
    $arr = pg_fetch_all( $result );
    foreach( $arr as $row )
    {
        $sql = "SELECT PopulaTempo( '" . $row['data'] . "'::date ) ";
        pg_query( $connDest, $sql );
    }

    // Insere dados da compra
    $sql = "INSERT INTO fatcompra ( iddimfilial, iddimproduto,
iddimfornecedor, iddimtempo, quantidade, Valor_NF, valor_liquido ) ";
    $sql .= "SELECT ";
    $sql .= "( SELECT iddimfilial      FROM dimfilial      WHERE
codigo=fato.fl      ORDER BY iddimfilial      DESC LIMIT 1 ), ";
    $sql .= "( SELECT iddimproduto      FROM dimproduto      WHERE
codigo=fato.re      ORDER BY iddimproduto      DESC LIMIT 1 ), ";
    $sql .= "( SELECT iddimfornecedor FROM dimfornecedor WHERE
codigo=fato.forn ORDER BY iddimfornecedor DESC LIMIT 1 ), ";
    $sql .= "( SELECT iddimtempo      FROM dimtempo      WHERE
data=fato.data      ORDER BY iddimtempo      DESC LIMIT 1 ), ";
    $sql .= " fato.qtde, fato.valorNF, fato.valor ";
    $sql .= "FROM compraConsumoTemp fato WHERE fato.op='10' ";
    Logger( "Sql Fato compra: ".$sql."\n" );
    pg_query( $connDest, $sql );
    $erro = pg_last_error( );
    if ( !empty( $erro ) )
        Logger( "Erro : ".$erro."\n" );
```

```
// Insere dados do Ajuste+
$sql = "INSERT INTO fatAjuste ( iddimfilial, iddimproduto,
iddimtempo, quantidade, valor ) ";
$sql .= "SELECT ";
$sql .= "( SELECT iddimfilial FROM dimfilial WHERE codigo=fato.fl
ORDER BY iddimfilial DESC LIMIT 1 ), ";
$sql .= "( SELECT iddimproduto FROM dimproduto WHERE codigo=fato.re
ORDER BY iddimproduto DESC LIMIT 1 ), ";
$sql .= "( SELECT iddimtempo FROM dimtempo WHERE data=fato.data
ORDER BY iddimtempo DESC LIMIT 1 ), ";
$sql .= " fato.qtde, fato.valor ";
$sql .= "FROM compraConsumoTemp fato WHERE fato.op='17' ";
Logger( "Sql Fato ajuste+: ".$sql."\n" );
pg_query( $connDest, $sql );
$erro = pg_last_error( );
if ( !empty( $erro ) )
    Logger( "Erro : ".$erro."\n" );

// Insere dados do Consumo
$sql = "INSERT INTO fatConsumo ( iddimfilial, iddimproduto,
iddimtempo, idDimRequisicao, quantidade, valor ) ";
$sql .= "SELECT ";
$sql .= "( SELECT iddimfilial FROM dimfilial WHERE codigo=fato.fl
ORDER BY iddimfilial DESC LIMIT 1 ), ";
$sql .= "( SELECT iddimproduto FROM dimproduto WHERE codigo=fato.re
ORDER BY iddimproduto DESC LIMIT 1 ), ";
$sql .= "( SELECT iddimtempo FROM dimtempo WHERE data=fato.data
ORDER BY iddimtempo DESC
LIMIT 1 ), ";
$sql .= " CASE WHEN fato.desp ILIKE '%MANUAL%' THEN 2 ELSE 1 END,
fato.qtde, fato.valor ";
$sql .= "FROM compraConsumoTemp fato WHERE fato.op='21' ";
Logger( "Sql Fato Consum: ".$sql."\n" );
pg_query( $connDest, $sql );
$erro = pg_last_error( );
if ( !empty( $erro ) )
    Logger( "Erro : ".$erro."\n" );

pg_query( $connDest, "DROP TABLE compraConsumoTemp" );

$resultProdutos = pg_query( $connDest, "SELECT a.iddimfilial,
a.codigo AS filial, b.iddimproduto, b.codigo AS produto FROM dimfilial a,
dimproduto b ORDER BY 2, 4" );
$arrayProdutos = pg_fetch_all( $resultProdutos );
// Insere dados dos níveis de estoque
$result = pg_query( $connDest, "SELECT iddimtempo, data FROM DimTempo
WHERE data BETWEEN '".$dataIni."' AND '".$dataFina."' ORDER BY data" );
$Datas = pg_fetch_all( $result );
$sqlInser = "INSERT INTO fat estoque ( iddimfilial, iddimproduto,
iddimtempo, quantidade, valor ) VALUES ";
foreach( $arrayProdutos as $linha )
{
    foreach( $Datas as $row )
    {
        $sql = "SELECT * FROM dblink( 'conexao', 'SELECT (wtent-
wtsai), (wfent-wfsai) FROM saldoRazaol15( '".$linha[ 'filial' ];
        $sql .= ", '".$linha[ 'produto' ]."', ''1998-01-01'', ''".
$row[ 'data' ].'', NULL, false, false ')'";
        $sql .= "AS resultado( estoqueFisico numeric, estoqueFinanceiro
numeric )";
    }
}
```

```
$estoque = pg_query( $connDest, $sql );
$saldos = pg_fetch_array( $estoque, 0 );
if ( $estoque && ( $saldos[ '0' ] > 0 || $saldos[ '1' ] > 0 ) )
{
    $sql = $sqlInser." ( ".$linha[ 'iddimfilial' ].", ".
$linha[ 'iddimproduto' ].", ".$row[ 'iddimtempo' ].", ".$saldos[ '0' ].",
".$saldos[ '1' ]." );
    pg_query( $connDest, $sql );
    $erro = pg_last_error( );
    if ( !empty( $erro ) )
        Logger( "Erro : ".$erro."\n" );
    //Logger( "Sql Fato Estoque: ".$sql );
}
}

// Salva periodo
grava( $connDest, "Estoque", $periodo );
// Encerra transação
pg_query( $connDest, 'COMMIT' );
pg_query( $connDest, "SELECT dblink_disconnect('conexao')" );
}
// finaliza conexao dom os bancos
pg_close( $connDest );
// Grava hora final
Logger( 'Final do processo ETL. ' );
?>
```

Apêndice C – Questionário de Avaliação do ambiente.

1. O ambiente disponibilizado atende a sua área de negócios (vendas, faturas, estoque)?

- Atende totalmente
- Atende parcialmente
- Indiferente
- Não atende parcialmente
- Não atende totalmente

2. O ambiente contribui para uma melhor tomada de decisões?

- Contribui totalmente
- Contribui parcialmente
- Indiferente
- Não contribui parcialmente
- Não contribui totalmente

3. Você considera a forma de apresentação de informações adequada?

- Totalmente adequada
- Parcialmente adequada
- Indiferente
- Parcialmente inadequada
- Totalmente inadequada

4. Você considera a forma de criação de análises adequada?

- Totalmente adequada
- Parcialmente adequada
- Indiferente
- Parcialmente inadequada
- Totalmente inadequada

5. Você considera que o ambiente disponibilizado contribui para o desempenho de suas atividades?

- Contribui totalmente
- Contribui parcialmente
- Indiferente
- Não contribui parcialmente
- Não contribui totalmente

6. Você considera as informações disponibilizadas no ambiente fiéis à realidade (confiáveis)?

- Totalmente fiéis
- Parcialmente fiéis
- Indiferente
- Parcialmente não fiéis
- Totalmente não fiéis

7. Você acredita que o ambiente poderia ser utilizado em um cenário real (em produção)?

- Acredita totalmente
- Acredita parcialmente
- Indiferente
- Não acredita parcialmente

() Não acredita totalmente

8. Você considera aceitável o comportamento do ambiente em relação ao tempo de resposta?

- () Totalmente aceitável
- () Parcialmente aceitável
- () Indiferente
- () Parcialmente não aceitável
- () Totalmente não aceitável

9. Quais melhorias as novas informações disponibilizadas trazem para o seu dia a dia em relação ao sistema atual para a tomada de decisões?

10. Quais melhorias as novas informações disponibilizadas trazem para o seu dia a dia em relação a facilidade de acesso às informações?

11. Você poderia citar algum problema que impediria o uso imediato em produção (distorção de informações, dificuldade de uso)?

12. Você acredita que o ambiente disponibilizado possa ser expandido para trazer outras informações de sua área de negócios? Quais informações?

13. Que outros departamentos da organização você acredita que também poderiam se beneficiar deste ambiente?

14. Que melhorias você sugere para o ambiente caso ele seja utilizado em produção em curto prazo? Essas melhorias são essenciais para o uso do ambiente ou são somente desejáveis?

Apêndice D – Resposta do avaliador 1.

1. O ambiente disponibilizado atende a sua área de negócios (vendas, faturas, estoque)?

- Atende totalmente
 Atende parcialmente
 Indiferente
 Não atende parcialmente
 Não atende totalmente

2. O ambiente contribui para uma melhor tomada de decisões?

- Contribui totalmente
 Contribui parcialmente
 Indiferente
 Não contribui parcialmente
 Não contribui totalmente

3. Você considera a forma de apresentação de informações adequada?

- Totalmente adequada
 Parcialmente adequada
 Indiferente
 Parcialmente inadequada
 Totalmente inadequada

4. Você considera a forma de criação de análises adequada?

- Totalmente adequada
 Parcialmente adequada
 Indiferente
 Parcialmente inadequada
 Totalmente inadequada

5. Você considera que o ambiente disponibilizado contribui para o desempenho de suas atividades?

- Contribui totalmente
 Contribui parcialmente
 Indiferente
 Não contribui parcialmente
 Não contribui totalmente

6. Você considera as informações disponibilizadas no ambiente fiéis à realidade (confiáveis)?

- Totalmente fiéis
 Parcialmente fiéis
 Indiferente
 Parcialmente não fiéis
 Totalmente não fiéis

7. Você acredita que o ambiente poderia ser utilizado em um cenário real (em produção)?

- Acredita totalmente
 Acredita parcialmente
 Indiferente
 Não acredita parcialmente

() Não acredita totalmente

8. Você considera aceitável o comportamento do ambiente em relação ao tempo de resposta?

- (x) Totalmente aceitável
- () Parcialmente aceitável
- () Indiferente
- () Parcialmente não aceitável
- () Totalmente não aceitável

9. Quais melhorias as novas informações disponibilizadas trazem para o seu dia a dia em relação ao sistema atual para a tomada de decisões?

10. Quais melhorias as novas informações disponibilizadas trazem para o seu dia a dia em relação a facilidade de acesso às informações?

No sistema utilizado pela nossa empresa, as informações de evolução de estoque/compra/consumo são feitas somente no fechamento mensal, pois a compra é feita com base na ficha técnica de cada artigo. No modelo apresentado, podemos ir monitorando diariamente a evolução das compras, consumos e estoque e posso intervir antes do fechamento do período caso algum número não seja satisfatório.

11. Você poderia citar algum problema que impediria o uso imediato em produção (distorção de informações, dificuldade de uso)?

O sistema apresentado pode ser utilizado somente nos produtos químicos e sistema de faturamento, onde as informações são instantâneas. Não seria viável avaliar diariamente itens do tipo material de manutenção, segurança do trabalhado, escritório, uma vez que o registro de estoque destes materiais não é instantâneo, visto que as notas fiscais têm que ser lançadas pela matriz.

12. Você acredita que o ambiente disponibilizado possa ser expandido para trazer outras informações de sua área de negócios? Quais informações?

13. Que outros departamentos da organização você acredita que também poderiam se beneficiar deste ambiente?

Comercial e programação de produção.

14. Que melhorias você sugere para o ambiente caso ele seja utilizado em produção em curto prazo? Essas melhorias são essenciais para o uso do ambiente ou são somente desejáveis?

Relatório por fornecedor elencando a maior compra os produtos por ordem de valor comprado.

Apêndice E – Resposta do avaliador 2.

1. O ambiente disponibilizado atende a sua área de negócios (vendas, faturas, estoque)?

- Atende totalmente
 Atende parcialmente
 Indiferente
 Não atende parcialmente
 Não atende totalmente

2. O ambiente contribui para uma melhor tomada de decisões?

- Contribui totalmente
 Contribui parcialmente
 Indiferente
 Não contribui parcialmente
 Não contribui totalmente

3. Você considera a forma de apresentação de informações adequada?

- Totalmente adequada
 Parcialmente adequada
 Indiferente
 Parcialmente inadequada
 Totalmente inadequada

4. Você considera a forma de criação de análises adequada?

- Totalmente adequada
 Parcialmente adequada
 Indiferente
 Parcialmente inadequada
 Totalmente inadequada

5. Você considera que o ambiente disponibilizado contribui para o desempenho de suas atividades?

- Contribui totalmente
 Contribui parcialmente
 Indiferente
 Não contribui parcialmente
 Não contribui totalmente

6. Você considera as informações disponibilizadas no ambiente fiéis à realidade (confiáveis)?

- Totalmente fiéis
 Parcialmente fiéis
 Indiferente
 Parcialmente não fiéis
 Totalmente não fiéis

7. Você acredita que o ambiente poderia ser utilizado em um cenário real (em produção)?

- Acredita totalmente
 Acredita parcialmente
 Indiferente
 Não acredita parcialmente

() Não acredita totalmente

8. Você considera aceitável o comportamento do ambiente em relação ao tempo de resposta?

- (x) Totalmente aceitável
() Parcialmente aceitável
() Indiferente
() Parcialmente não aceitável
() Totalmente não aceitável

9. Quais melhorias as novas informações disponibilizadas trazem para o seu dia a dia em relação ao sistema atual para a tomada de decisões?

Poder filtrar e buscar os dados de acordo com a necessidade de informação que cada tomada de decisão requer.

10. Quais melhorias as novas informações disponibilizadas trazem para o seu dia a dia em relação a facilidade de acesso às informações?

Poder escolher quais dados, em determinado período de tempo, e de que forma quero visualizar as informações.

11. Você poderia citar algum problema que impediria o uso imediato em produção (distorção de informações, dificuldade de uso)?

Nenhum, considero a ferramenta pronta para uso, obviamente mediante treinamento e preparação para utilização dos usuários.

12. Você acredita que o ambiente disponibilizado possa ser expandido para trazer outras informações de sua área de negócios? Quais informações?

Pode ser expandido sim, principalmente informações diárias de pagamentos, cobranças, controle de custos.

13. Que outros departamentos da organização você acredita que também poderiam se beneficiar deste ambiente?

Financeira e contábil.

14. Que melhorias você sugere para o ambiente caso ele seja utilizado em produção em curto prazo? Essas melhorias são essenciais para o uso do ambiente ou são somente desejáveis?

Se possível um menu de opções de mais fácil compreensão, mas que com treinamento não chega a ser problema, somente melhoria desejável.