

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
ESPECIALIZAÇÃO EM BANCO DE DADOS**

**HERCULES ALEX BORGES**

**DATA WAREHOUSE E ETL APLICADOS AO SETOR AGRÍCOLA  
PARA OBTENÇÃO DE INFORMAÇÕES ESTRATÉGICAS**

**MONOGRAFIA DE ESPECIALIZAÇÃO**

**PATO BRANCO  
2017**

**HERCULES ALEX BORGES**

**DATA WAREHOUSE E ETL APLICADOS AO SETOR AGRÍCOLA  
PARA OBTENÇÃO DE INFORMAÇÕES ESTRATÉGICAS**

Trabalho de Conclusão de Curso, apresentado ao II Curso de Especialização em Banco de Dados, da Universidade Tecnológica Federal do Paraná, campus Pato Branco, como requisito parcial para obtenção do título de Especialista.

Orientador: Dr. Claudio Leones Bazzi.

**PATO BRANCO  
2017**



Ministério da Educação  
**Universidade Tecnológica Federal do Paraná**  
Câmpus Pato Branco

Diretoria de Pesquisa e Pós-Graduação  
II Curso de Especialização em Banco de Dados – Administração e  
Desenvolvimento



---

## **TERMO DE APROVAÇÃO**

**DATA WAREHOUSE E ETL APLICADOS AO SETOR AGRÍCOLA PARA OBTENÇÃO  
DE INFORMAÇÕES ESTRATÉGICAS**

por

**HERCULES ALEX BORGES**

Este Trabalho de Conclusão de Curso foi apresentado em 24 de fevereiro de 2017 como requisito parcial para a obtenção do título de Especialista em Banco de Dados. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

---

**Dr. Claudio Leones Bazzi**  
Prof. Orientador

---

**Marcelo Teixeira**  
Membro titular

---

**Viviane Dal Molin de Souza**  
Membro titular

“O Termo de Aprovação assinado encontra-se na Coordenação do Curso”

Dedico este trabalho ao meu filho Kevin Alex Borges e a minha esposa Gisele Ana da Silva, por estarem sempre me apoiando e principalmente pela paciência, amor e respeito aos meus objetivos.

## **AGRADECIMENTOS**

Ao Prof. Dr. Claudio Leones Bazzi, pela orientação, apoio, confiança, e que foi o principal responsável por minha especialização em banco de dados.

Ao Prof. Dr. Marcelo Teixeira, pelo apoio, pela confiança e pela oportunidade em participar deste incrível curso de especialização e desta fantástica turma de Pato Branco.

Ao Prof. Dr. Marco Antonio Barbosa, por estar sempre disponível, acompanhando e tirando dúvidas com relação a metodologia deste trabalho.

Você nunca sabe que resultados virão da sua ação. Mas se você não fizer nada, não existirão resultados.

Mahatma Gandhi

## RESUMO

BORGES, Hercules Alex. *Data Warehouse e ETL* aplicados ao setor agrícola para obtenção de informações estratégicas, 2017. 75 f. Monografia (II Curso de Especialização em Banco de Dados) - Universidade Tecnológica Federal do Paraná. Pato Branco, 2017.

O setor comercial de grãos de cooperativas agrícola e empresas cerealistas no Paraguai, é responsável também por gerenciar os contratos de grãos, acompanhando as operações de recebimento físico e de faturamento das compras de grãos, mediante a fixação de preços destes contratos. Conhecer resultados atingidos e confronta-los com metas pré-estabelecidas pela empresa, verificando a sua evolução ao longo do tempo e percebendo a sua tendência, irá gerar informações para tomada de decisão que é fundamental em qualquer processo de gestão. Verificou-se a possibilidade em implementar uma solução de Data Warehouse aplicando o ETL, tendo como origem o banco de dados Oracle, utilizado pelo sistema ERP Dolphin que foi desenvolvido pela Datapar S.A., buscando assim transformar os dados em informações através da modelagem dimensional e apresentação dos indicadores de desempenho. Optou-se pela ferramenta de ETL da Pentaho o PDI e pelo banco de dados Postgres para armazenar o DW.

**Palavras-chave:** *Data Warehouse. ETL. PDI. Pentaho.*

## **ABSTRACT**

BORGES, Hercules Alex. Data warehouse and ETL applied to the agricultural sector to obtain strategic information, 2017. 75 f. Monography (II Specialization Course in Database) - Federal University of Technology - Parana. Pato Branco, 2017.

The commercial grain sector of agricultural cooperatives and cereal companies in Paraguay is responsible for managing grain contracts, following the physical receipt and billing of grain purchases, by means of price fixing. Knowing the results achieved and confronting them with goals established by the company, verifying its evolution over time and realizing its tendency, will generate information for decision making that is fundamental in any management process. It was verified the possibility of implementing a Data Warehouse solution applying the ETL, originating from the Oracle database, used by the Dolphin ERP system that was developed by Datapar SA, thus seeking to transform the data into information through dimensional modeling and presentation Performance indicators. The Pentaho ETL tool PDI and the Postgres database were used to store the DW.

**Keywords:** Data Warehouse. ETL. PDI. Pentaho.



## LISTA DE SIGLAS

DM	<i>Data Mart</i>
DW	<i>Data Warehouse</i>
ETL	<i>Extract Transform Load</i>
OLAP	<i>Online Analytical Processing</i>
OLTP	<i>Online Transaction Processing</i>
PDI	<i>Pentaho Data Integration</i>
XML	<i>Extensible Markup Language</i>

## LISTA DE FIGURAS

Figura 1 - Elementos básicos do DW. (Kimball e Ross, 2002).....	18
Figura 2 - Tabela Fato Vendas .....	20
Figura 3 - Tabela Dimensão Cliente .....	21
Figura 4 - Modelo Estrela.....	22
Figura 5 - Modelo Floco de Neve.....	22
Figura 6 - Etapas adotadas para a Modelagem do DW.....	30
Figura 7 - Etapas de construção do DW.....	30
Figura 8 - Visão geral da Modelagem DW Agrícola.....	37
Figura 9 - Esquema estrela do recebimento de grãos. ....	38
Figura 10 - Esquema estrela do fechamento de preço. ....	38
Figura 11 - Esquema estrela do saldo a fixar preço.....	39
Figura 12 - Tabelas Junk e Stages utilizadas no ETL.....	40
Figura 13 - Jobs de carga geral do DW Agrícola .....	41
Figura 14 - Menu de componentes do <i>Dashboard Designer</i> .....	41
Figura 15 - Seleção de métricas e propriedades para componentes do <i>Dashboard</i> .....	42
Figura 16 - Comparativo de recebimento por filial .....	43
Figura 17 - <i>Dashboard Designer</i> , criando painel de recebimento de grãos.....	43
Figura 18 - Curva ABC de filiais e Top 5 de produtores rurais com contratos de grãos fechados.....	44
Figura 19 - <i>Dashboard Viewer</i> em ASP.NET, visualizando painel de saldo a fixar.....	45
Figura 20 - <i>Job</i> que realiza a carga da dimensão tempo do DW.....	46
Figura 21 - Criação de consulta multidimensional Recebimento de Grãos, com <i>Dashboard Designer</i> . ....	47
Figura 22 - <i>Dashboard</i> Cubo Recepção de Grãos, sendo executado a partir do IIS. ....	48

## LISTA DE TABELAS

Tabela 1 - Matriz de barramento do DW .....	31
Tabela 2 - Matriz de Barramento Detalhada, ou Matriz de Necessidades.....	31
Tabela 3 - Fonte de Dados Dimensões do DW Agrícola.....	32
Tabela 4 - Fonte de Dados para Fatos do DW Agrícola .....	34

## SUMÁRIO

1 INTRODUÇÃO .....	14
1.1 CONSIDERAÇÕES INICIAIS.....	14
1.2 OBJETIVOS .....	14
1.2.1 Objetivo Geral.....	14
1.2.2 Objetivos Específicos .....	15
1.2.3 JUSTIFICATIVA.....	16
1.2.4 ORGANIZAÇÃO DO TEXTO.....	16
2 DATA WAREHOUSE.....	17
2.1 ARQUITETURA DO DATA WAREHOUSE .....	18
2.2 MODELAGEM DIMENSIONAL .....	19
2.2.1 Tabelas Fatos.....	20
2.2.2 Tabelas Dimensão.....	21
2.2.3 Modelo Estrela ( <i>star schema</i> ) .....	21
2.2.4 Modelo Floco de Neve ( <i>snow flake schema</i> ) .....	22
2.2.5 Arquitetura de Barramentos .....	23
2.2.6 ETL.....	23
3 MATERIAIS E MÉTODO. ....	25
3.1 MATERIAIS .....	25
3.1.1 MÉTODO.....	26
4 RESULTADOS E DISCUSSÃO .....	29
4.1 APRESENTAÇÃO DO DATA WAREHOUSE .....	29
4.2 MODELAGEM DO SISTEMA.....	29
4.2.1 Etapas para Modelagem.....	30
4.2.2 Matriz de Necessidades.....	30
4.2.3 Fonte de Dados.....	32
4.2.4 Modelagem Dimensional .....	36
4.3 DESCRIÇÃO DO SISTEMA .....	40
4.4 IMPLEMENTAÇÃO DO SISTEMA.....	45
5 CONCLUSÃO .....	49
REFERÊNCIAS .....	50

## 1 INTRODUÇÃO

Este capítulo apresenta as considerações iniciais, com uma visão geral do trabalho, os objetivos e a justificativa do mesmo e a organização do texto.

### 1.1 CONSIDERAÇÕES INICIAIS

A gestão moderna busca monitorar o desempenho de suas organizações por meio de indicadores, apurando os resultados a serem comparados com as metas, verificando o desvio ocorrido e seu respectivo desempenho.

Os clientes de empresas agrícolas da Datapar S.A. buscam monitorar estes indicadores elegendo os principais para suas tomadas de decisões, exportando dados do ERP e construindo informações por meio de planilhas.

Com propósito de apresentar de forma gráfica a produção do recebimento agrícola, foi implementado um *Data Warehouse* (DW) utilizando-se de modelagem dimensional, para que todas as informações possam ser facilmente identificadas e utilizadas em *dashboards* (*painéis*), os quais facilitam a tomada de decisão de gestores das organizações.

A principal origem dos dados é o banco de dados Oracle, tendo como alvo o DW construído no banco de dados Postgres, todo o processo de ETL foi realizado por meio da ferramenta *Pentaho Data Integration* (PDI).

### 1.2 OBJETIVOS

O objetivo geral apresenta o resultado principal do trabalho realizado e os objetivos específicos o complementam, no sentido de valores agregados.

#### 1.2.1 Objetivo Geral

Aplicar ferramentas de *ETL* em um *DW* que será construído à partir de dados agrícolas de recebimento de grãos, buscando indicadores de desempenho, exemplificando técnicas utilizadas para a modelagem dimensional, extração, transformação e cargas dados, a fim de obter informações estratégicas sobre o segmento agrícola.

### 1.2.2 Objetivos Específicos

- Realizar a modelagem dimensional para os dados relacionados a compra e recebimento de grãos (soja, milho, trigo, etc.), incluindo a fixação de preços de mercados futuros, em busca de indicadores de desempenho;
- Criar um DW utilizando o banco de dados Postgres fazendo uso dos dados definidos na etapa anterior;
- Extrair dados de um banco de dados, realizar as transformações necessárias para que os dados persistam corretamente no banco de dados destino DW Postgres, para essa tarefa será utilizado o Pentaho PDI;
- Apresentar as informações em *dashbord* (painel) através das soluções da DevExpress que permite visualizar dados além de Windows e mobile, também em Web, o qual será utilizado neste trabalho através da tecnologia ASP.NET.
- Analisar dados de entrada de grãos em silos pela pesagem de caminhões, buscando o volume de recebimentos por cultura de cereal e por filial da empresa agrícola, para que possa ser realizado um comparativo ao longo dos anos das três maiores culturas (soja, milho e trigo).
- Analisar dados de cereais já recebidos nos silos e que ainda não estão com o preço fixado, para que possa ser realizado comparativos entre filiais no ano atual e no ano anterior, através de Curva ABC.
- Após analisar os dados das duas etapas anteriores, existe uma terceira etapa onde será verificado por meio de curva ABC as filiais que mais receberam grãos, e que já estão com seus preços fixados. Outra informação importante que será buscada nesta terceira etapa, são os 5 (cinco) produtores que mais entregaram grãos e já realizaram a fixação de preço.

### 1.2.3 JUSTIFICATIVA

A construção de um DW para organizar as informações em busca de indicadores de desempenho no recebimento de grãos, tais como soja, milho e trigo, justifica pelo fato do grande risco em gerir as organizações agrícolas por meio de suposições, passando então a visualizar de forma simples informações relevantes para sua tomada de decisão.

A utilização do Postgres como banco de dados para o DW, além de não onerar o projeto por ser um banco de dados *open source*, possui muitos recursos que podem contribuir no ETL. Neste sentido, utilizou-se a ferramenta *Pentaho Data Integration (PDI)* para extrair os dados de origem, armazenado em um banco de dados Oracle. Para visualização dos dados, optou-se pela ferramenta Devexpress Dashboard Visualization, pela praticidade e simplicidade, em gerar os Dashboards.

### 1.2.4 ORGANIZAÇÃO DO TEXTO

Este texto está organizado em capítulos, dos quais este é o primeiro e apresenta a ideia e o contexto da solução desenvolvida, incluindo os objetivos e a justificativa.

O Capítulo 2 contém o referencial teórico que fundamenta a proposta conceitual da solução desenvolvida. O referencial teórico está centrado em DW, modelagem dimensional e ETL.

No Capítulo 3 estão os materiais e o método utilizados no desenvolvimento deste trabalho, incluindo a elaboração da monografia ferramentas utilizadas.

O Capítulo 4 contém a solução desenvolvida, com exemplos de documentação da modelagem dimensional e de desenvolvimento do DW através do PDI realizando ETL.

No Capítulo 5 está a conclusão com as considerações finais.

## 2 DATA WAREHOUSE

O DW é banco de dados com informações recuperadas do ambiente operacional de uma empresa. Em teoria, os dados de um DW não são voláteis, exceto quando necessário realizar correções de dados já carregados. Estes dados normalmente não devem ser alterados, pois estão disponíveis somente para consulta. Uma das características marcantes do DW é a sua capacidade em consolidar dados de várias origens, não somente de um ambiente transacional OLTP, mas também de planilhas eletrônicas documento texto, entre outros. Inmom define o D. W. como sendo uma coleção de dados orientados por assuntos, integrados, variáveis com o tempo e não voláteis, para dar suporte ao processo de tomada de decisão. Kimball descreve que, por meio de um conjunto de ferramentas e técnicas de projeto, quando aplicadas às necessidades específicas dos usuários e aos bancos de dados, permitirá que planeje e construa DW.

As propriedades de um DW, são quem o diferencia de convencionais de apoio a decisão. São estas:

- ✓ **Orientado a assunto:** os sistemas transacionais normalmente, fala-se de módulos. Ex. modulo de faturas, modulo de PDV, no caso do DW, a abordagem é por assunto, visando sempre o setor das empresas, tentando responder a perguntas específicas como: “Quais são os produtores rurais que mais fornecem soja nos últimos 5 anos?”.
- ✓ **Integrado:** o DW é constituído de fontes de dados de várias origens, sendo obrigatório o tratamento das informações, formatando, realizando formatação de datas, moedas, conflitos de nomes, conversão de medidas, para que seja possível relacionar as informações.
- ✓ **Não volátil,** diferente dos sistemas transacionais, onde operações de inclusão, exclusão e alteração de registros é realizado a todo instante, o DW só possui duas operações que devem ser realizadas, uma é a carga, a outra é a consulta. A carga realizada, normalmente é realizada de uma determinada data em diante, garantindo o histórico dos assuntos.
- ✓ **Variante no tempo:** Ser variável no tempo, significa poder acessar o DW, mantém históricos da empresa por período muito longos, permitindo por exemplo análises de tendências, ou mesmo auditorias confrontada por diferentes assuntos.



## 2.1 ARQUITETURA DO DATA WAREHOUSE

As duas maiores referências sobre o assunto de DW, Bill Inmon e Ralph Kimball, divergem quanto a construção do DW, enquanto Inmon aborda *top-down*, Kimball aborda *botton-up*.

Ralph Kimball defende que o DW deve ser construído a partir de varios Data Marts orientados por assunto, e depois integrar os diversos assuntos para obter-se o DW. Tendo concluído o DW, haveria pontos de conexões entre dos DM, que no caso são as tabelas Fatos e Dimensões. Este conceito foi chamado por Kimball de DW Bus Architecture. O Kimball Group ficou conhecido pelas centenas de DWs bem-sucedidos.

Bill Inmon pensa de maneira oposta a Kimball. Segundo Inmon primeiro devemos construir o DW, criando um único modelo corporativo, seguindo depois para os *Data Marts* que devem ser construídos por departamentos ou assuntos. Ele acredita que o ponto de partida deva ser o CIF – *Corporate Information Factory*, o qual seria alimentado por sistemas transacionais, e também sugere a construção de *Operational Data Store* – ODS, sendo facultativo a sua utilização, mas segundo ele ajuda a reduzir a complexidade na construção do DW, pois os sistemas transacionais da empresa teriam seus esforços de integração depositados no ODS.

Segundo Kimball e Ross (2002), o ambiente de um DW é formado por quatro componentes: iniciando pela Fonte de Dados, seguido pela *Data Staging Area*, depois os Dados de Apresentação e por último as Ferramentas para usuário Final. A Figura 1 demonstra a iteração entre os elementos.

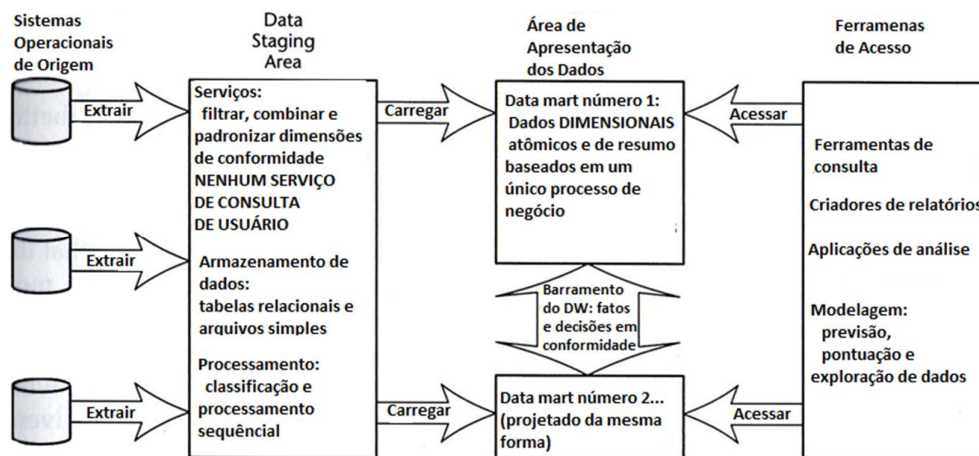


Figura 1 - Elementos básicos do DW. (Kimball e Ross, 2002)

Fonte de dados é o componente do DW que abrange todos os dados de origem que irão compor as informações do DW. Compreende os sistemas OLTP, os arquivos em diversos formatos (XLS, TXT, etc.), os sistemas de CRM, ERP e outras possíveis origens (Kimball 2004).

A área de armazenamento intermediária utilizada no ETL é denominada *Data Staging Area*, que é de extrema importância na transição dos dados de origem até o destino final no DW (Kimball 2004).

Área de apresentação de dados (*Data Presentation Area*) é o local em que os dados ficam organizados, armazenados e tornam-se disponíveis a serem consultados diretamente pelo usuário final, por desenvolvedores de relatórios analíticos e ferramentas de mineração de dados. Lembrando-se de que a *Staging Area* deve ser inacessível ao usuário final.

Ferramentas de acesso a dados (*Data Access Tools*) são por definição, todas as ferramentas de acesso a dados consultam os dados na área de apresentação do DW, como exemplo temos as ferramentas com capacidade de análises multidimensionais das informações armazenadas denominadas de OLAP, do inglês *On-line Analytical Processing*, e também as ferramentas responsáveis por descoberta de informações relevantes, por meio de padrões, associações, mudanças e anomalias encontradas nos dados analisados, chamadas de *Data Mining*, ou Mineração de Dados.

Para entender melhor o fluxo desta arquitetura proposta por Kimball, começamos pela extração dos dados de origens que normalmente são arquivos texto, planilhas, banco de dados relacionais de ERP's, CRM's e demais ambientes operacionais. Estes dados extraídos são armazenados temporariamente em uma área denominada *stage area*, onde devem ser tratados, validados e formatados até a sua entrega no DW, por meio do processo ETL. Estando estas informações em conformidade com suas origens, serão consumidas por ferramentas OLAP ou ferramentas de Data Mining.

## **2.2 MODELAGEM DIMENSIONAL**

Visando melhorar o desempenho de consultas e provendo facilidades de utilização a partir de um conjunto de eventos básicos de medição, surge a metodologia chamada Modelagem dimensional.

“Em um ambiente Sistema Gerenciador de Banco de Dados Relacional (SGBDR),

a tabela de fatos, é construída com um registro para cada medida distinta. Esta tabela de fatos é então envolvida por um conjunto de tabelas de dimensão que descrevem precisamente o que é conhecido no contexto de cada registro de medição. Devido à estrutura característica de um modelo dimensional, normalmente ele é denominado *Star schema*. Os modelos dimensionais mostram serem compreensíveis, previsíveis, ampliáveis e altamente resistentes ao ataque específico de grupos de usuários de negócio devido à sua natureza simétrica previsível. Os modelos dimensionais são a base de muitos aprimoramentos de desempenho do SGDBR, inclusive agregações e métodos de indexação avançados. Também são a base do desenvolvimento incremental e distribuído do DW através do uso de tabelas dimensões e tabelas fatos em conformidade e a base lógica de todos os sistemas OLAP”. (Kimball e Ross, 2002)

São três os esquemas que utilizamos na modelagem dimensional: - *Star schema*, de onde se derivam os próximos dois esquemas, *Snowflake schema* e *Facts Constallation schema* (Kimball et al., 2002).

### 2.2.1 Tabelas Fatos

A principal tabela em um modelo dimensional é a tabela de fatos, na qual as medições numéricas de desempenho da empresa estão armazenadas, como mostra a Figura 2.

Segundo Kimball, todas as medições em uma tabela de fatos devem estar alinhadas na mesma granularidade. Os fatos mais interessantes são os numéricos e aditivos, assim como o exemplo de volume de vendas em dólares. Existem também tabelas fatos numéricos e semi-aditivos, que podem ser sumarizadas apenas algumas dimensões, pois na maioria dos casos são saldos e só podem ser somados ao longo de uma dimensão ou outra, mas não de todas. Já os fatos que não são aditivos, nestes não é possível sumarizar nenhuma dimensão.

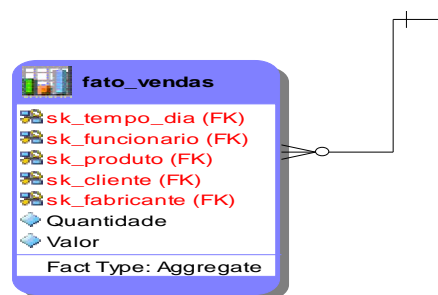


Figura 2 - Tabela Fato Vendas

## 2.2.2 Tabelas Dimensão

Tabelas de dimensão sempre estão ligadas a uma tabela de fatos. Elas possuem as características da tabela fato.

Utiliza-se de chaves simples, em geral as chaves artificiais, ou SK (surrogate keys), estas tabelas normalmente não dependem do tempo, pois existe uma dimensão tempo em quase todos os DWs.

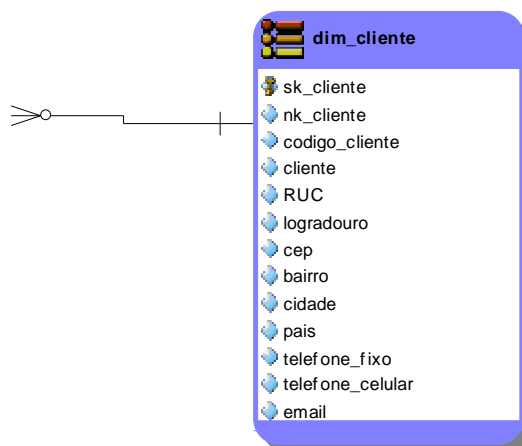


Figura 3 - Tabela Dimensão Cliente

## 2.2.3 Modelo Estrela (*star schema*)

A reunião destes dois blocos a tabela de fatos e as tabelas de dimensão, denomina-se de esquema de junção em estrela. Segundo Kimball, a primeira coisa que observamos neste modelo é a simplicidade e a simetria, esta mesma simplicidade proporciona melhor desempenho nas consultas. Outra vantagem é a flexibilidade deste modelo quando houver mudanças. Quanto maior a granularidade ou atômica, maior será a sua capacidade de dimensionalidade.

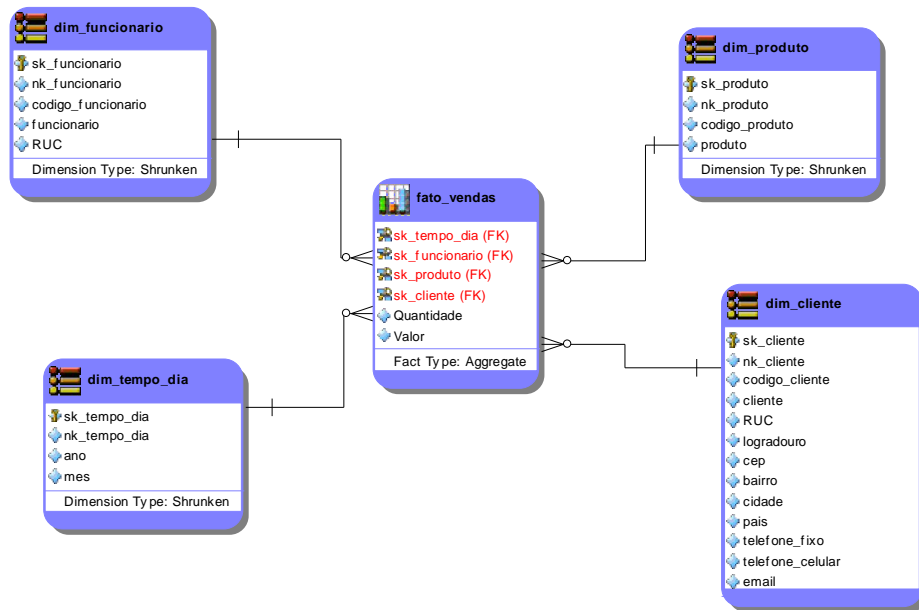


Figura 4 - Modelo Estrela

#### 2.2.4 Modelo Floco de Neve (Snow Flake Schema)

Este modelo tem como prática a remoção dos atributos redundantes da tabela de dimensão desnormalizada, simples, colocando em uma tabela de dimensão secundária normalizada (Kimball, Ross 2002.). O que ocorre neste caso é um aumento na complexidade do modelo e a baixa performance nas consultas devido ao aumento de junções (joins) que serão necessários para trazer a informação desejada.

Kimball, recomenda que apesar de ser aceita como modelo dimensional, em geral, deve ser evitada, em virtude dos nossos dois orientadores de projeto: facilidade de uso e desempenho. A figura 5 demonstra a dimensão produto sendo detalhada por mais de um fabricante.

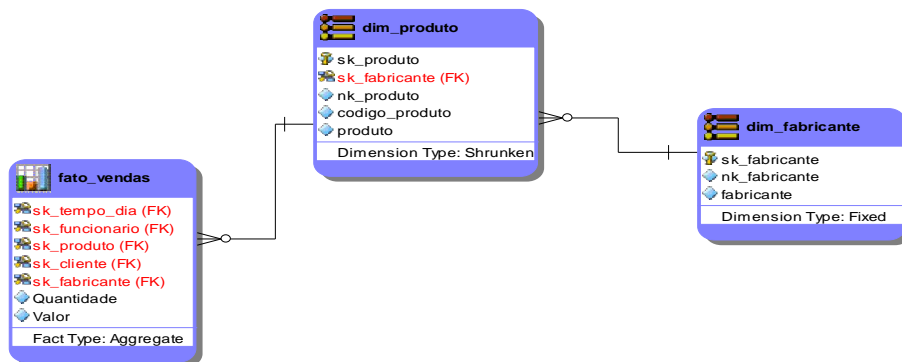


Figura 5 - Modelo Floco de Neve

### 2.2.5 Arquitetura de Barramentos

Kimball, deixa claro que a criação do DW em partes isoladas inviabiliza o objetivo maior que é a consistência. Ele sugere a utilização de um método incremental e arquitetônico denominado arquitetura de barramento de DW (Kimball, Ross 2002).

#### Equação

Data Marts podem estar conectados e coexistirem de modo muito útil se obedecerem a um mesmo padrão, pois a reutilização de dimensões torna a ETL mais otimizada, reduzindo também as manutenções. A Figura 6 demonstra a matriz de barramento DW, que é a ferramenta utilizada para criar, documentar e comunicar a arquitetura de barramento.

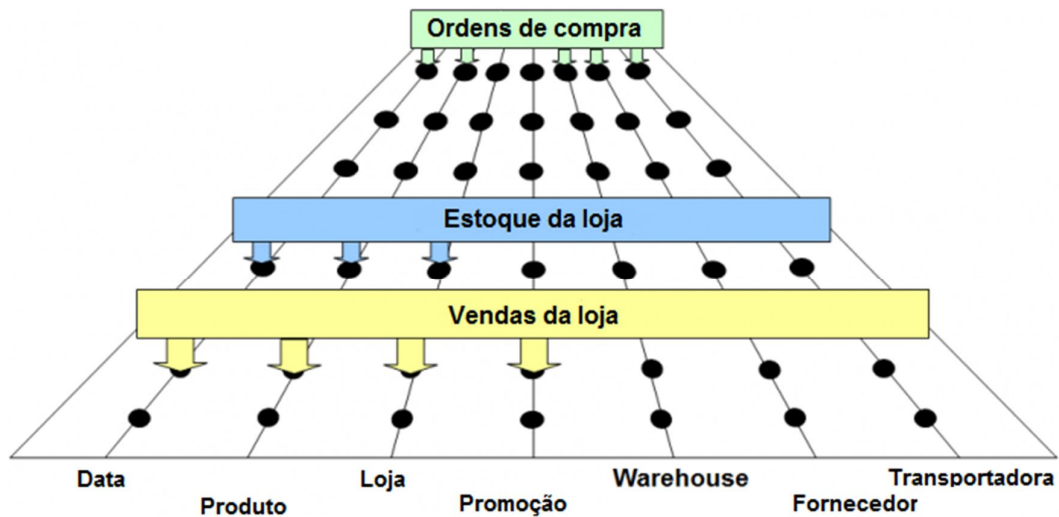


Figura 6 - Bus Matrix (Kimball e Ross, 2002)

### 2.2.6 ETL

A ETL é fundamental para que o DW tenha sua implantação bem-sucedida. De acordo com Inmon (2002,p.295): “ O processo de integração e transformação de dados tipicamente consome até 80% dos recursos no desenvolvimento”. Na prática isso é verdade, pois se analisarmos que, boa parte da regra de negócio se resolve na ETL, fazendo com que as tabelas fato e dimensão, tenha as informações em conformidade com os sistemas de origem, precisamos então, “perder” tempo com o entendimento dos sistemas origem para carregar no DW a informação correta.

Nesta área, realizam-se quatro importantes atividades no processo de ETL, são elas:

Extração de dados: Quando lemos os dados da origem para colocarmos na área de estágio, corrigindo os conflitos e realizando tratamentos de elementos ausentes, ajustando nomenclaturas e formatando se necessário para um formato específico.

Limpeza de dados: Sempre é realizada quando necessitamos remover informações não desejadas, vindas dos sistemas de origem, aqui também removemos duplicações, garantindo a coesão com a regra de negócio.

Colocar em conformidade: Ocorre quando obtemos dados de origens distintas, arquivos formato texto, JSON, XML, planilhas eletrônicas e banco de dados relacional, entre outras origens possíveis. Como estas fontes de dados podem estar “misturadas” em um DW, precisamos criar um padrão a nível corporativo, a fim de integrar os dados, para não invalidar a informação apresentada.

Entregar dados para o Front Room: A área de staging deve disponibilizar os dados preparados para a execução de Queries. O produto final entregue pelos processos de ETL, são os dados estruturados fisicamente, de maneira “simétrica”, em esquemas dimensionais denominados esquemas estrela.

## 2.4 OLTP VS. OLAP

Os sistemas de TI podem ser divididos em sistema analítico (OLAP) e sistema transacional (OLTP). Geralmente dizemos que o OLTP é o fornecedor da fonte de dados para o DW, enquanto o OLAP auxilia a análise destes dados.

OLTP (On-line Transaction Processing) é identificada pelo seu grande número de transações utilizando as operações (*INSERT*, *UPDATE*, *DELETE*). O foco principal do banco de dados OLTP é realizar estas operações rapidamente, mantendo a integridade de dados em ambientes multi-acesso.

OLAP (On-line Analytical Processing) possui um volume bem baixo de transações. O foco é o tempo de resposta nas consultas. No banco de dados OLAP, existem dados históricos sumarizados, agregados, armazenados em esquemas multidimensionais com o modelo estrela, ou também com o modelo floco de neve, este último é menos recomendado por Ralph Kimball, pois torna o modelo mais complexo e com menor desempenho nas consultas.

### 3 MATERIAIS E MÉTODO.

Este capítulo apresenta os materiais e o método utilizados na realização deste trabalho. Os materiais se referem às tecnologias como linguagens e ferramentas para a modelagem e a construção do DW. O método contém as etapas com os principais procedimentos utilizados para o desenvolvimento do sistema.

#### 3.1 MATERIAIS

As ferramentas e as tecnologias utilizadas para as atividades de modelagem, ETL e construção do DW são:

- a) Embarcadero ER/Studio Version 10.0 (Embarcadero Technologies, 2014) para modelagem dimensional;
- b) Oracle Database 10g Release 10.2.0.5 (Oracle, 2001, 2006) para a banco de dados origem do ambiente operacional;
- c) PL/SQL Developer Versão 9.0.0.1601(Allround Automations, 1997 - 2011), como ferramenta para construção de consultas SQL e Views no Oracle, banco de dados origem;
- d) Pentaho Data Integration Community(PDI) também chamado de Kettle Versão pdi-ce-5.4.0.1-130(Pentaho Corporation 2007 - 2012) utilizado para realização da ETL;
- e) Máquina Virtual Java (JVM) Versão jdk1.7.0\_05 (Oracle, 1993, 2012);
- f) PostgreSQL Versão 9.4.10 (The PostgreSQL Global Development Group, 1996-2017) que servirá como DW, deste trabalho;
- g) pgAdmin PostgreSQL Tools Version 1.20.0 (The pgAdmin Development Team, 2002-2014) utilizada para acessar, e realizar execução de scripts e consultas SQL no banco de dados Postgres;
- h) Microsoft Visual Studio Community 2015 Version 14.0.25431.01 Update 3 (Microsoft Corporation, 2017) para implementação dos Dashboards;
- i) Microsoft .NET Framework Version 4.6.01586(Microsoft Corporation, 2017);
- j) IIS Versão 10.0.14393.0(Microsoft Corporation, 2017) rodando no Windows 10, servidor Web para disponibilizar os Dashboards em ASP.NET;



- k) Data Visualization Dashboard versão 15.2.5 (Developer Express Inc., 1998-2017), ferramenta utilizada para criar e publicar os *dashboards*;

### 3.1.1 MÉTODO

O método utilizado neste trabalho está baseado na abordagem *bottom up* e no Ciclo de vida do DW, caracterizado por uma sequência de etapas a serem seguidas. Tudo começa no planejamento do Programa ou Projeto, em seguida realiza-se levantamento e também a definição dos requisitos de negócios. A partir daí, vamos ter três caminhos a seguir. O primeiro seria pela parte de Arquitetura e Design Técnico, o segundo seguimos trabalhando a definição da Modelagem Dimensional, o Projeto físico e a parte de ETL com a utilização da *Data Staging*, e o terceiro que planeja e desenvolve uma aplicação para as consultas analíticas, que são realizadas pelos gestores e executivos, buscando indicadores de desempenho para o seu negócio, o front end do DW. (Kimball et al 1998)

Essas fases foram utilizadas como base, mas sofreram inclusões e alterações para adaptar-se aos interesses deste projeto. Ressalta-se que as atividades realizadas não foram estritamente sequenciais.

- a) Planejamento: no planejamento foram definidas as principais etapas e atividades para a realização do projeto. Foi realizada pesquisa dentre os clientes da Datapar S.A., também foi avaliado a forma que seria disponibilizado o sistema. Tanto o processo de ETL, DW utilizando Postgres, como a servidor de aplicação que neste caso será o MSIIS, estarão em um servidor Windows, com conexão para o servidor em ambiente Linux onde se encontra dados transacionais de Origem no banco de dados Oracle.
- b) Definição de requisitos do negócio: foi realizada entrevista com quatro grandes clientes selecionados pela direção da empresa Datapar S.A., após os gestores falarem sobre os seus “desejos”, no cruzamento de informações de vários módulos do ERP Dolphin, foi solicitado para que informassem o assunto de maior interesse, tanto as cooperativas agrícolas, quanto as empresas agrícolas, reforçaram o interesse nos assuntos de recebimento de grãos, bem como o seu faturamento e saldos.
- c) Projeto da arquitetura técnica: considerada uma das mais importantes fases do

projeto de DW. Em reuniões internas com a direção da Datapar S.A., foi definida a equipe responsável pela criação do DW, também foi definido um ambiente no Sistema operacional Windows Server 2012, para configurar ferramentas necessárias para a construção do DW.

- d) A fase de seleção e instalação de produtos: nesta fase foi realizada uma pesquisa referente as ferramentas de ETL disponíveis no mercado. Cogitou-se em criar a própria ferramenta, porém não é viável pelo tempo que isso poderia levar. Optou-se pelo PDI (Kettle).
- e) Modelagem dimensional: Utilizamos uma matriz de necessidades proposta pelo professor (Oliveira, Grimaldo, 2016, pag. 26) para realizar o levantamento de requisitos referente a definição do DM Agrícola. Esta matriz é tem como objetivo assegurar que o DW seja utilizável e extensível a todos os assuntos (DM) do DW. A modelagem dimensional adotada neste projeto foi a Star Schema. Nesta fase também é analisada as fontes de dados para que atendam ao modelo de dados.
- f) Projeto físico: neste momento, foi definida a nomenclatura a ser utilizada no DW, e criado o desenho físico das estruturas lógicas do modelo dimensional. Realizando a criação do database no banco de dados Postgres, em seguida a criação das tabelas fato e dimensão por meio de script DDL (*Data Definition Language*), também foram criados, atributo de tabelas, campos chaves, relacionamentos, índices e restrições.
- g) Projeto e desenvolvimento da área de organização de dados: nesta fase demonstramos as atividades fundamentais a serem realizadas no DW, que são: a extração, transformação e carga de dados. Foi criado dentro o Pentaho Data Integration (PDI) duas conexões, sendo uma para o sistema transacional de origem (banco de dados Oracle) e outra para o DW (banco de dados Posgres). Iniciou-se a implementação carregando as *stage* áreas, após isso utilizou-se da informação nas *stages* para gerar dos dados para as dimensões e fatos por meio de transformações e assim que estas foram concluídas e testadas, foi criado *jobs* para agrupar as transformações, criando uma sequência, estas eram selecionadas por *jobs* do PDI.
- h) Especificações de aplicações de usuário final: como o projeto de DW é extenso, foi necessário além de identificar os assuntos mais prioritários, identificar também as informações que estarão no Dashboard e facilitar o acesso através de abas por assunto.

- i) Desenvolvimento de aplicações para usuário final: devido a portabilidade, elegemos a Web como o ambiente perfeito para a apresentação dos painéis com os indicadores, baseados em ferramentas da Devexpress utilizando ASP.NET.
- j) Disponibilização do DW: A princípio foi solicitado pela Datapar S.A., a disponibilidade em seu *Data Center*, viabilizando um cenário ideal.
- k) Manutenção e crescimento do DW: Quando implantado em algum cliente, será necessário o contínuo acompanhamento, realizando monitoramento, suporte e atendimento, sobre performance e possíveis dúvidas quanto a utilização das ferramentas dimensionais consumidas pelo DW.

## 4 RESULTADOS

Este capítulo apresenta o sistema que foi desenvolvido como resultado deste trabalho. Inicialmente é apresentada a descrição do mesmo. Em seguida é apresentada a sua modelagem. Por fim, o processo de ETL é apresentado, por meio de telas e explicação do seu funcionamento e a implementação é exemplificada por meio de partes da codificação SQL, ou transformações do PDI, e terminamos apresentando os painéis (*dashboards*) a ferramenta de desenvolvimento utilizada e a aplicação em ASP.NET com indicadores consultados pela Web.

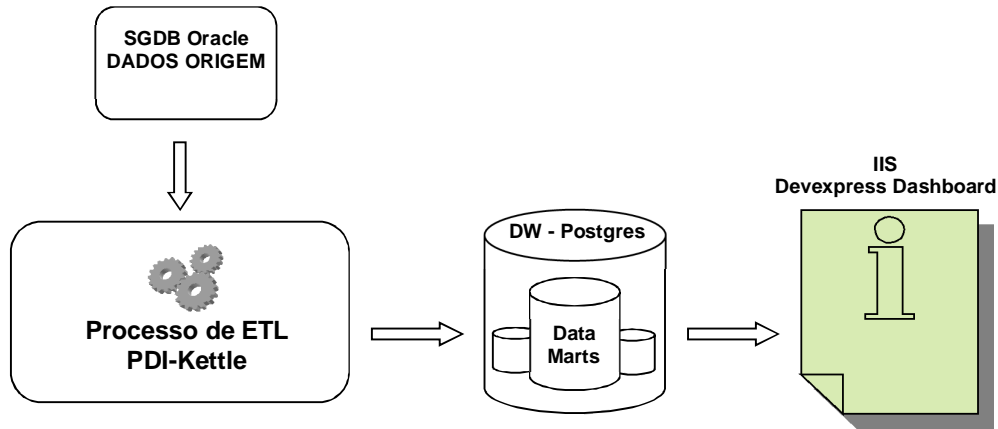
### 4.1 APRESENTAÇÃO DO DATA WAREHOUSE

O DW modelado e desenvolvido como resultado deste trabalho, permite o armazenamento de informações originadas de um ambiente transacional para o setor agrícola. Estas informações já estão modeladas no esquema estrela, que é otimizado para consultas analíticas. Os dados são agrupados e armazenados por dia, utilizando a dimensão *dim\_tempo\_dia*.

A interface com o usuário é via Web, onde o usuário terá acesso a três abas, sendo que cada aba tem um dashboard sobre determinado contexto dentro do assunto Agrícola. Cada Dashboard possui gráficos, tabelas e outros componentes necessários para a apresentação dos indicadores.

### 4.2 MODELAGEM DO SISTEMA

Inicialmente foi definido um diagrama com a visão geral da construção do DW, com etapas integrando-se entre si, demonstrada na Figura 9. O propósito deste diagrama é poder demonstrar esta integração dos processos, que serão realizados para a construção de informações analíticas, desde os dados da origem, até a apresentação destas informações por meio de painéis (*dashboards*) na Web.

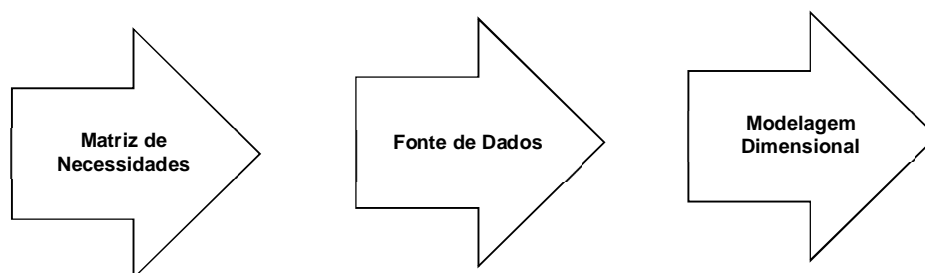


**Figura 6 - Etapas adotadas para a Modelagem do DW**

#### 4.2.1 Etapas para Modelagem

Para diminuir a complexidade no processo de entrevistas, dispensando os possíveis mapas mentais e planilhas contendo dados desejados, vamos utilizar uma técnica mais eficaz (Oliveira, Grimaldo, 2016, pag. 25) que nos cita três etapas para minimizar o tempo necessário de desenvolvimento de um projeto de DW.

Estas métricas foram inicialmente identificadas na Matriz de Necessidades e encontradas no banco de dados relacional do ERP Dolphin, e posteriormente informadas na Fonte de Dados.



**Figura 7 - Etapas de construção do DW**

#### 4.2.2 Matriz de Necessidades

Ao realizar-se o levantamento e identificar os indicadores, adicionamos na Matriz de Necessidades todos os requisitos do DW Agrícola, incluindo as métricas e suas

características. Normalmente armazenamos as métricas nas tabelas Fatos, e as propriedades nas tabelas de Dimensão.

**Tabela 1 - Matriz de barramento do DW**

<i>Dimensions / Facts</i>	Recebimento de Grãos	Fechamento de Preço	Saldo Fechamento de Preço
<b>Tempo</b>	*	*	*
<b>Cliente e Fornecedor</b>	*	*	*
<b>Empresa e Filial</b>	*	*	*
<b>Produto Agrícola</b>	*	*	*

**Tabela 2 - Matriz de Barramento Detalhada, ou Matriz de Necessidades.**

	Recebimento de Grãos	Fechamento de Preço	Saldo Fechamento de Preço
	Quantidade de Grãos Recebida	Qtd. Recebida com Preço Fechado	Saldo da Quantidade Recebida para Fechamento de Preço
		Valor Recebida com Preço Fechado	
<i>Dimensions / Facts</i>	Recebimento de Grãos	Fechamento de Preço	Saldo Fechamento de Preço
SK_TEMPO_DIA			
DATA			
MÊS			
ANO			
<b>Tempo</b>	*	*	*
SK_CLIENTE			
Nome do Cliente			
RUC do Cliente			
Endereço do Cliente			
Bairro do Cliente			
Cidade do Cliente			
Estado do Cliente			
País do Cliente			
Cliente é Associado			
Data Início Parceria			
Data da Inativação			
Código do Cliente			
NK do Cliente			
<b>Cliente e Fornecedor</b>	*	*	*
SK_EMPRESA			

Nome da Empresa			
Nome da Filial			
Nome do Grupo Empresarial			
Código Empresa			
Código Filial			
Código Grupo Empresarial			
NK Empresa Filial			
<b>Empresa e Filial</b>	*	*	*
<i>SK_PRODUTO_AGRICOLA</i>			
Nome do Produto			
Nome da Família			
Nome da Cultura			
Safra			
Código Produto			
Código Família			
Semente			
NK Produto Agrícola			
<b>Produto Agrícola</b>	*	*	*

### 4.2.3 Fonte de Dados

Após trabalharmos com a Matriz de Necessidades, criamos a planilha de Fonte de Dados que é a principal guia para projetarmos o DW, com ela estão mapeadas todas as tabelas, visões e atributos, bem como seus relacionamentos quando necessários, facilitando a identificação nos sistemas transacionais.

O objetivo desta ferramenta é facilitar a comunicação aumentando a nossa compreensão sobre os assuntos a serem modelados do DW Agrícola.

A tabela de fonte de dados para dimensões do assunto agrícola é comum termos a coluna de relacionamento das tabelas de origem, porém como a solução adotada foi a utilização de visões (*views*) do banco de dados origem, não foi necessário apresentar as junções entre tabelas.

**Tabela 3 - Fonte de Dados Dimensões do DW Agrícola**

DIMENSÕES	ORIGEM		OBSERVAÇÃO
	Visões (Views)	CAMPO	
<b>CLIENTE</b>			
Nome do Cliente	v_stg_cliente	cliente	
RUC do Cliente	v_stg_cliente	ruc	RUC no Paraguai, é equivalente ao CPF no Brasil
Endereço do Cliente	v_stg_cliente	direccion	

Bairro do Cliente	v_stg_cliente	localidad	
Cidade do Cliente	v_stg_cliente	ciudad	
Estado do Cliente	v_stg_cliente	departamento	
País do Cliente	v_stg_cliente	pais	
Cliente é Associado	v_stg_cliente	situacion	Utilizado apenas por Cooperativas
Data Início Parceria	v_stg_cliente	data_ini_parceria	Alguns clientes já eram parceiro antes da implantação do ERP.
Data da Inativação	v_stg_cliente	data_inativacao	
Código do Cliente	v_stg_cliente	cod_cliente	
NK do Cliente	v_stg_cliente	nk_cliente	
<b>EMPRESA E FILIAL</b>			
Nome da Empresa	v_stg_empresa_filial	empresa	
Nome da Filial	v_stg_empresa_filial	filial	
Nome do Grupo Empresarial	v_stg_empresa_filial	grupoempresarial	
Código Empresa	v_stg_empresa_filial	cod_empresa	
Código Filial	v_stg_empresa_filial	cod_filial	
Código Grupo Empresarial	v_stg_empresa_filial	cod_grupoempresarial	
NK Empresa Filial	v_stg_empresa_filial	nk_empresa	
<b>PRODUTO AGRÍCOLA</b>			
Nome do Produto	v_stg_dim_produto_agricola	produto	
Nome da Família	v_stg_dim_produto_agricola	familia	Família é o subgrupo do produto
Nome da Cultura	v_stg_dim_produto_agricola	cultura	Cultura, SOJA, MILHO(MAIZ), TRIGO, necessário, pois o subgrupo está abrangendo mercadorias que não são cereais.
Safra	v_stg_dim_produto_agricola	safra	
Código Produto	v_stg_dim_produto_agricola	cod_produto	
Código Família	v_stg_dim_produto_agricola	cod_familia	Família é o subgrupo do produto
Semente	v_stg_dim_produto_agricola	sementes	
NK Produto Agrícola	v_stg_dim_produto_agricola	nk_produto_agricola	
<b>Tempo (Data Registro Primeira Diária)</b>			
Ano			A dimensão TEMPO não tem origem. Ela é gerada pela ETL.
Mês			
Data			

Fonte de Dados para Fatos do DW, aqui realizamos as junções, ou seja, os relacionamentos entre as tabelas fatos que armazenam as métricas com as dimensões que apresenta suas características. Para este trabalho identificamos três tabelas fatos, uma para recebimento de grãos, outra para o fechamento de preços de grãos já recebido, e por fim o saldo a fechar preços dos grãos já recebidos.



Tabela 4 - Fonte de Dados para Fatos do DW Agrícola

FATOS				
Recebimento de Grãos	ORIGEM			OBSERVAÇÃO
	TABELA	CAMPO	RELACIONAMENTO	
Métricas				
Quantidade Recebida em Kg	fato_recebimento_de_graos	quantidade		A quantidade será inserida no DW em Kg, porém poderá ser apresentada ao usuário em Toneladas.
DIMENSÕES RECEBIMENTO DE GRÃOS				
Cliente	dim_cliente	sk_cliente	from fato_recebimento_de_g raos inner join dim_cliente using (sk_cliente)	O Cliente neste contexto, pode ser uma cooperativa, um produtor rural, com uma empresa agrícola.
Empresa	dim_empresa	sk_empresa	from fato_recebimento_de_g raos inner join dim_empresa using (sk_empresa)	Alguns clientes da Datapar são multi empresas, além disso a maioria possui filiais, necessitando de análises por filiais.
Produto agrícola	dim_produto_agricola	sk_produto_agricola	from fato_recebimento_de_g raos inner join dim_produto_agricola using (sk_produto_agricola)	Produto agrícola são considerados os cereais, soja, milho, trigo, sorgo, etc.
Tempo	dim_tempo_dia	sk_tempo_dia	from fato_recebimento_de_g raos inner join dim_tempo_dia using (sk_tempo_dia)	
Fechamento de Preços de Grãos	ORIGEM			OBSERVAÇÃO
	TABELA/VISÃO	CAMPO	RELACIONAMENTO	
Métricas				
Quantidade em Kg de preço fechado	fato_fechamento_de_preco	quantidade		O valor dos fechamentos são todos em dólar (US\$). A quantidade será inserida no DW em Kg, porém poderá ser apresentada ao usuário em Toneladas.
Valor referente ao Quantidade em Kg Fechada		valor		

DIMENSÕES FECHAMENTO DE PREÇOS DE GRÃOS				
Ciente	dim_cliente	sk_cliente	from fato_fechamento_de_preco inner join dim_cliente using (sk_cliente)	O Cliente neste contexto, pode ser uma cooperativa, um produtor rural, com uma empresa agrícola.
Empresa	dim_empresa	sk_empresa	from fato_fechamento_de_preco inner join dim_empresa using (sk_empresa)	Alguns clientes da Datapar são multi empresas, além disso a maioria possui filiais, necessitando de análises por filiais.
Produto agrícola	dim_produto_agricola	sk_produto_agricola	from fato_fechamento_de_preco inner join dim_produto_agricola using (sk_produto_agricola)	Produto agrícola são considerados os cereais, soja, milho, trigo, sorgo, etc.
Tempo	dim_tempo_dia	sk_tempo_dia	from fato_fechamento_de_preco inner join dim_tempo_dia using (sk_tempo_dia)	
Tempo	dim_tempo_dia	sk_tempo_dia	from fato_saldo_de_fechamento_de_preco inner join dim_tempo_dia using (sk_tempo_dia)	

Saldo de Fechamento de Preços de Grãos	ORIGEM			OBSERVAÇÃO
	TABELA	CAMPO	RELACIONAMENTO	
Métricas				
Saldo de Fechamento de Preços de Grãos	fato_saldo_de_fechamento_de_preco	saldo		
<b>DIMENSÕES SALDO DE FECHAMENTO DE PREÇOS DE GRÃOS</b>				
Cliente	dim_cliente	sk_cliente	from fato_saldo_de_fechamento_de_preco inner join dim_cliente using (sk_cliente)	O Cliente neste contexto, pode ser uma cooperativa, um produtor rural, com uma empresa agrícola.
Empresa	dim_empresa	sk_empresa	from fato_saldo_de_fechamento_de_preco inner join dim_empresa using (sk_empresa)	Alguns clientes da Datapar são multiempresas, além disso a maioria possui filiais, necessitando de análises por filiais.
Produto agrícola	dim_produto_agricola	sk_produto_agricola	from fato_saldo_de_fechamento_de_preco inner join dim_produto_agricola using (sk_produto_agricola)	Produto agrícola são considerados os cereais, soja, milho, trigo, sorgo, etc.

#### 4.2.4 Modelagem Dimensional

Nesta etapa, nós formatamos o banco no modelo estrela (*star schema*), facilitando localizar as informações desejadas pelos usuários, navegando com maior eficiência através de ferramentas de consulta. Esta mesma simplicidade dos modelos dimensionais, é que aumenta a velocidade de acesso a uma informação no DW.

Na figura 11 é possível ver a reutilização das tabelas de dimensão nas tabelas fatos dentro do DW Agrícola, isso é muito importante quando envolvemos mais assuntos, facilitando a integração destes.



**Figura 8 - Visão geral da Modelagem DW Agrícola**

O primeiro esquema estrela que subgrupamos (figura 9) é o Recebimento de Grãos, considerado o mais importante no Assunto Agrícola, responsável por armazenar informações do estoque físico recebido nos silos ao longo dos anos:

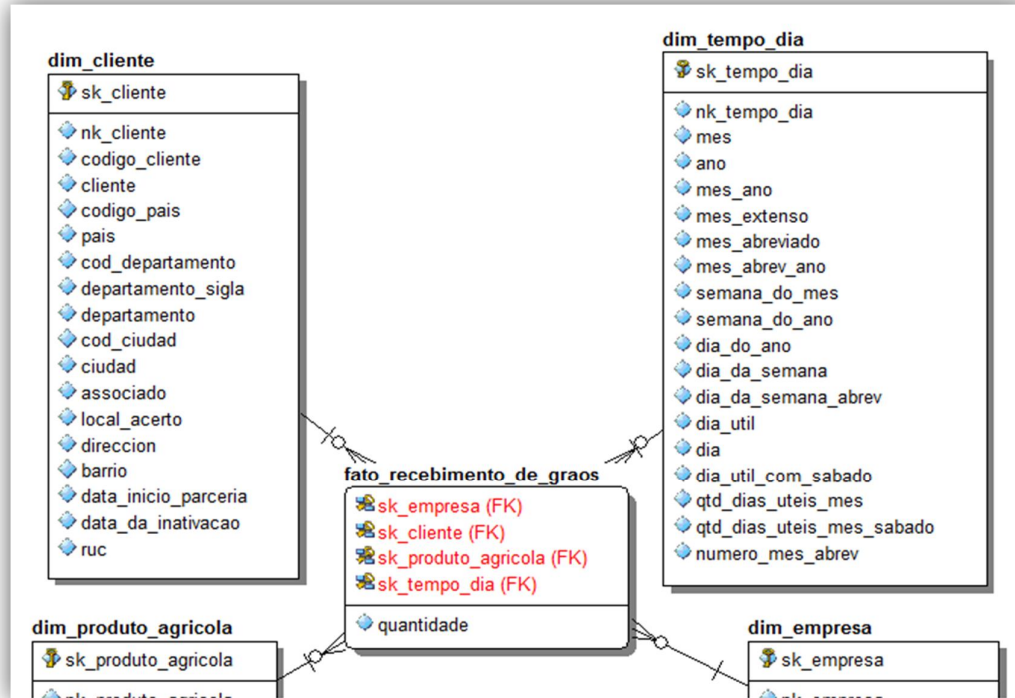


Figura 9 - Esquema estrela do recebimento de grãos.

O segundo esquema estrela que subgrupamos (figura 10) é o Fechamento de Preço, que tem como responsabilidade guardar as informações pertinentes ao preço fechado no mercado internacional e mercado interno.

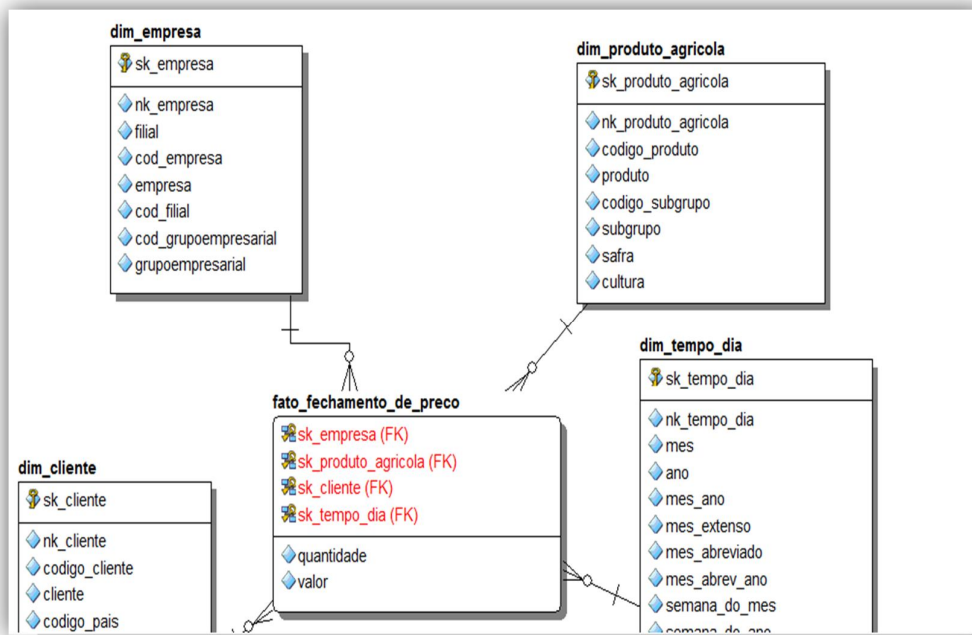


Figura 10 - Esquema estrela do fechamento de preço.

Por fim o terceiro é o esquema (figura 11) de saldos do que se recebeu e ainda falta fechar preço.

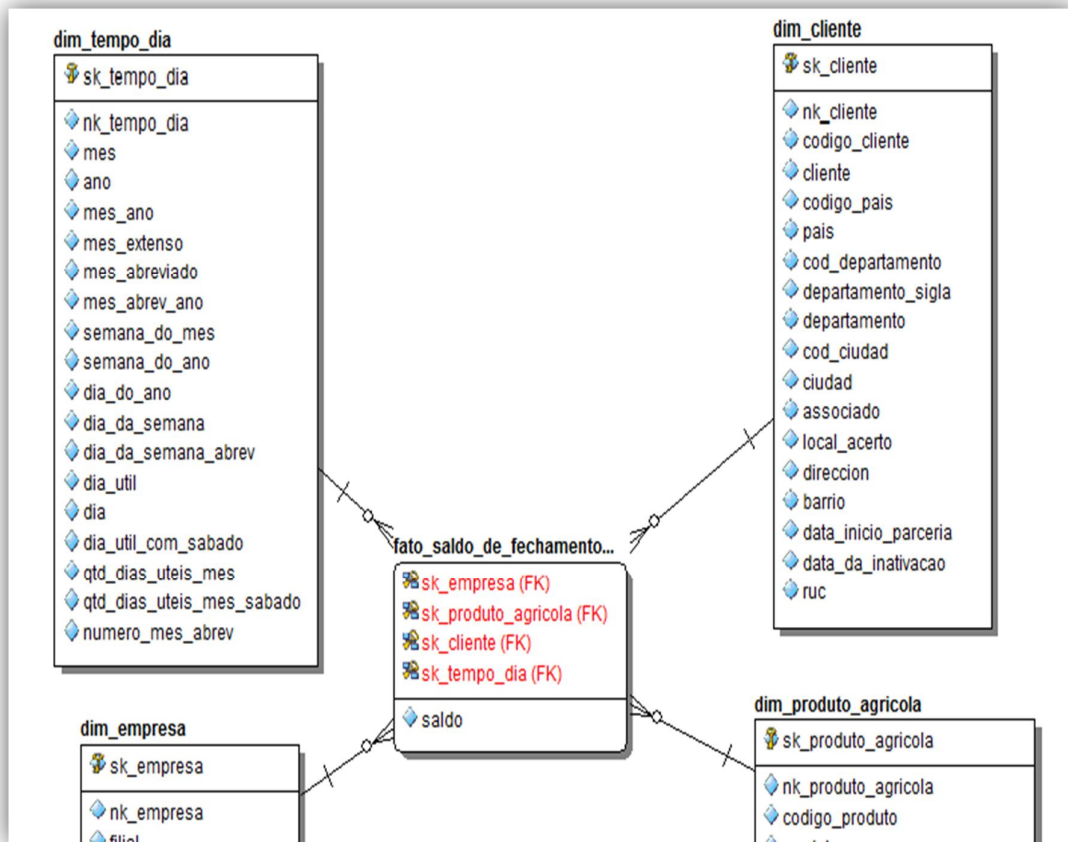


Figura 11 - Esquema estrela do saldo a fixar preço.

Percebam que a modelagem dimensional é autoexplicativa, de fácil entendimento, que facilita a comunicação com os usuários chave das aplicações analíticas e entre todos os profissionais envolvidos no projeto.

Outras tabelas também criadas para a solução do DW, forma tabelas *Junk*, e tabelas de *stage* que auxiliam no processo de carga do ETL, porém elas não se relacionam dentro de DW.



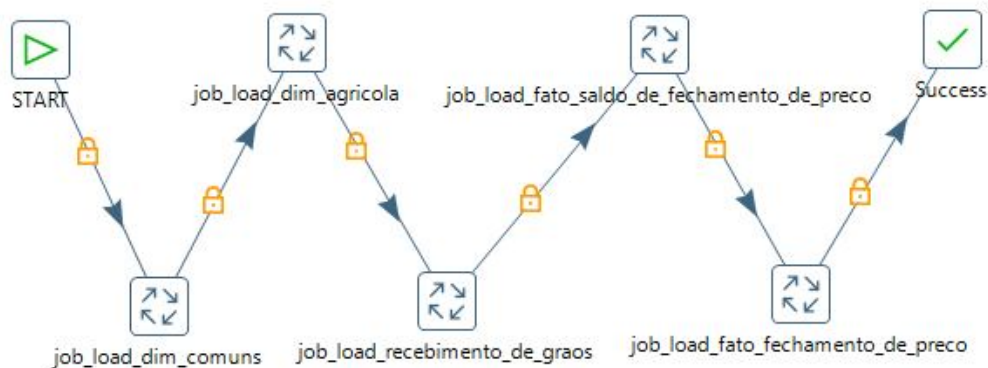
Figura 12 - Tabelas Junk e Stages utilizadas no ETL.

### 4.3 DESCRIÇÃO DO SISTEMA

O *DW*, depois de modelado e estruturado, precisa ser alimentado com informações originadas do sistema operacional Dolphin ERP que tem como base de dados o SGDB Oracle. Para realizar esta tarefa a interface de usuário que realiza este trabalho é o PDI, antigamente também chamado de Kettle. Ele é o responsável por extrair os dados da origem, trata-los, validá-los e finalmente gravar nas tabelas do *DW*.

Este processo de carga realizado pelo PDI, não precisa ser executado a todo instante, normalmente é agendado pelo sistema operacional que o executa em horários fora do expediente das organizações, visando não comprometer o desempenho o sistema transacional.

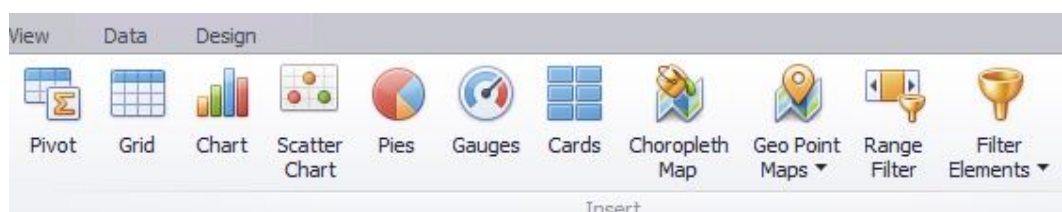
Através do arquivo `spoon.bat` que se encontra na pasta `pdi-ce-5.4.0.1-130\data-integration` do PDI, selecionamos o job de carga geral (figura 13) que nos demonstra a interatividade da carga dos *steps*, neste caso *Jobs* da *ETL*.



**Figura 13 - Jobs de carga geral do DW Agrícola**

Após as informações estarem em conformidade no DW, prontos para serem consumidos por ferramentas de análise, podemos então apresentar os painéis com indicadores. Estes painéis desenvolvidos com a componentes *Devexpress* dentro da IDE do *MS Visual Studio 2015 community*, demonstram de forma rápida e visual alguns indicadores necessários para a gestão agrícola na tomada de decisão.

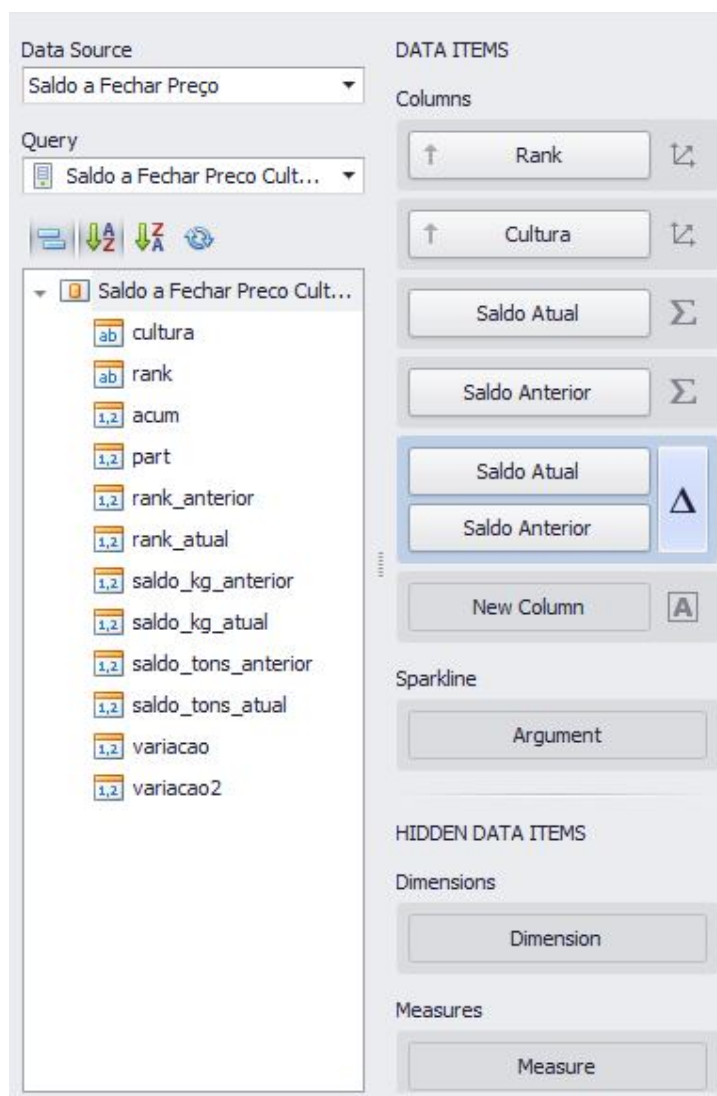
Após criado os Painéis do Sistema por meio do *Dashboard Designer*, nós salvamos o projeto em um arquivo XML, que é reutilizado na aplicação *Web ASP.NET*. A figura 14 apresenta o menu de componentes *Dashboard Designer* com opções de *Grid*, *Pivot Grid*, *Chart*, *Gauges* entre outros componentes que podem ser utilizados em *Designer* para montarmos um determinado *Dashboard*.



**Figura 14 - Menu de componentes do Dashboard Designer**

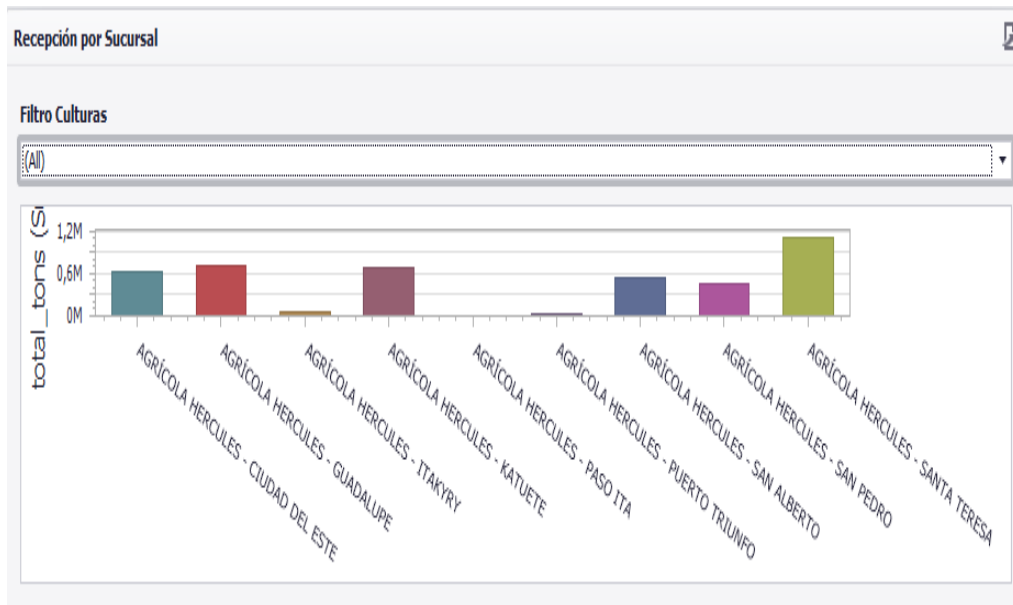
Para construção do *Dashboard* é necessário a utilização de uma consulta ao DW, esta consulta fica disponível no Designer para que possa ser selecionado os campos necessários e aplicado algum tipo de fórmula disponível quando a informação for uma métrica. A figura 15 demonstra algumas colunas sendo selecionada por meio de arrastar e soltar.





**Figura 15 - Seleção de métricas e propriedades para componentes do *Dashboard***

Dentre os componentes utilizado na construção dos painéis, o componente filtro é de grande importância, para que o usuário interaja com o *Dashboard*, possibilitando visualizar por filial o total de grãos recebidos ou ainda, filtrar uma cultura específica realizando a mesma análise, porém com foco agora na cultura selecionada conforme podemos ver na figura 16.



**Figura 16 - Comparativo de recebimento por filial**

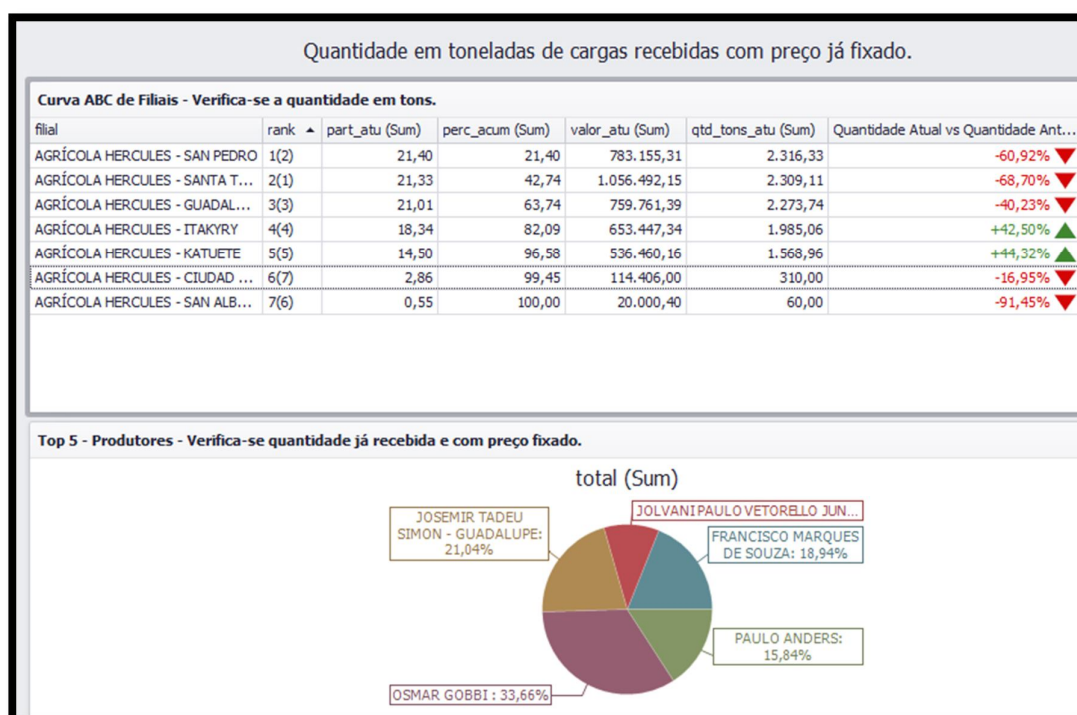
Na figura 18 é apresentada a cultura soja e seu recebimento por ano, podendo ser comparado com as outras principais culturas milho e trigo que se encontram no mesmo *Dashboard*, importante para apresentar ao banco e financeiras como forma de comprovação de solicitações de crédito.



**Figura 17 - Dashboard Designer, criando painel de recebimento de grãos.**

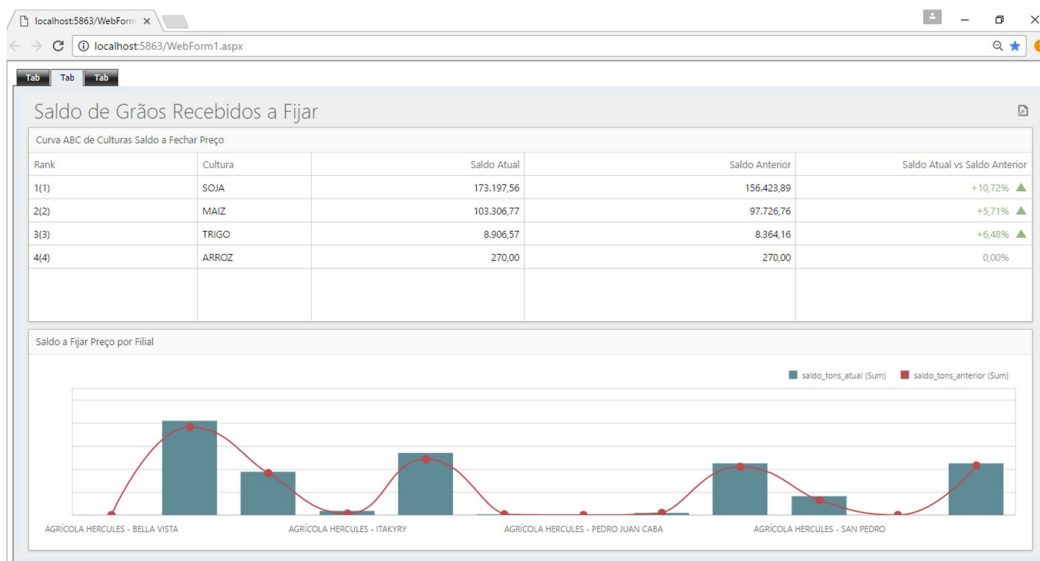
Para uma correta tomada de decisões, quanto abrir uma nova filial ou fechar uma existente é necessário um acompanhamento de seu desempenho ano a ano, para isso utilizamos a curva ABC de filiais que demonstra um rank da sua posição atual e a que estava no ano anterior por meio da notação 1(2), sendo que 1 é o ano atual e (2) é sua posição no ano anterior (figura 18).

O top 5 de produtores é muito importante para a organização identificar e também bonificar os que entregam no prazo e que fecham os preços com antecedência (figura 18).



**Figura 18 - Curva ABC de filiais e Top 5 de produtores rurais com contratos de grãos fechados.**

O Saldo de Grãos Recebidos a Fixar (figura 19), nos apresenta uma curva ABC, que é uma ferramenta gerencial no auxílio de identificação de itens que necessitam de maior atenção por parte do gestor, neste caso temos um Ranking onde é verificado o produto que mais tem saldo a fixar, e como era este saldo no mesmo mês do ano anterior, ainda é possível ver a variação destes saldos demonstrado de forma gráfica, que pode ser configurada no *Dashboard Designer*.



**Figura 19 - Dashboard Viewer em ASP.NET, visualizando painel de saldo a fixar.**

#### 4.4 IMPLEMENTAÇÃO DO SISTEMA

Começamos a implementação pela estrutura física do banco de dados, escolhemos o *Postgres* para essa tarefa.

As *Surrogate Keys* (SK) garantem a unicidade de identificadores por todo o ambiente do DW, normalmente de tipo inteiro, que ocupam pouco espaço e possibilitam junções mais rápidas do que outros tipos de dados.

Uma SK é uma chave substituta que geralmente é um valor numérico arbitrário atribuído pelo processo de carga ou pelo sistema de banco de dados. A grande vantagem em utilizá-la é que ela pode ser estruturada de forma que seja sempre única por toda a extensão de integração para o DW.

A *Natural Key* (NK) é formada pela concatenação das PK's das tabelas do sistema de origem, que foram utilizadas na geração dos dados para a dimensão. Por conveniência, melhor leitura e validação de dados costuma-se concatenar as PK's das tabelas de origem com um separador (#,\$,|,etc).

No Anexo 1, temos a DDL do dim\_empresa, note que colocamos como *UNIQUE* a NK, para que possamos garantir no PDI que os registros da origem não seja duplicados no DW.

No Anexo 2, é apresentada uma *DDL* de uma tabela fato do DW, notem que a Chave Primária é composta pelas Chaves Estrangeiras, que neste caso são as SK.

Outro ponto importante neste *DDL* é a utilização dos *DEFERRABLE CONSTRAINT*. Por padrão, as *CONSTRAINTS* são aplicadas e validadas quando uma instrução DML é executada mesmo sem executar o commit, neste caso para cada *INSERT*, *DELETE* e *UPDATE* o banco de dados faz essa verificação, diminuindo a performance de inserção em massa. A fim de melhorar a performance do processo de carga na ETL, estas *CONSTRAINTS* são postergadas para que o banco realize o *commit* das inserções somente no final de suas transações.

O PDI – Kettle, é a ferramenta que utilizamos neste trabalho para extrair os dados do Sistemas de Origem, transformar os dados para que possamos carregar nos modelos dimensionais do nosso DW.

Para facilitar nosso trabalho configuramos no arquivo “kettle.properties” as variáveis de conexão do banco de origem e destino, garantindo a reutilização das transformações sem necessitar mudar parâmetros de conexão do banco de dados.

Ao abrir o PDI temos na pasta BI/ETL do nosso projeto uma transformação da dimensão tempo no arquivo *trans\_dim\_tempo\_dia.ktr* (Figura 20), utilizada para não carregar a dimensão tempo toda hora, nesta transformação verificar se ela já está com a quantidade de 21.916 registros, caso não esteja é por que os dados não foram carregados ainda, então o step dá um erro e cai no step *job\_carga\_dim\_tempo*, executando a carga completa da *dim\_tempo\_dia*.

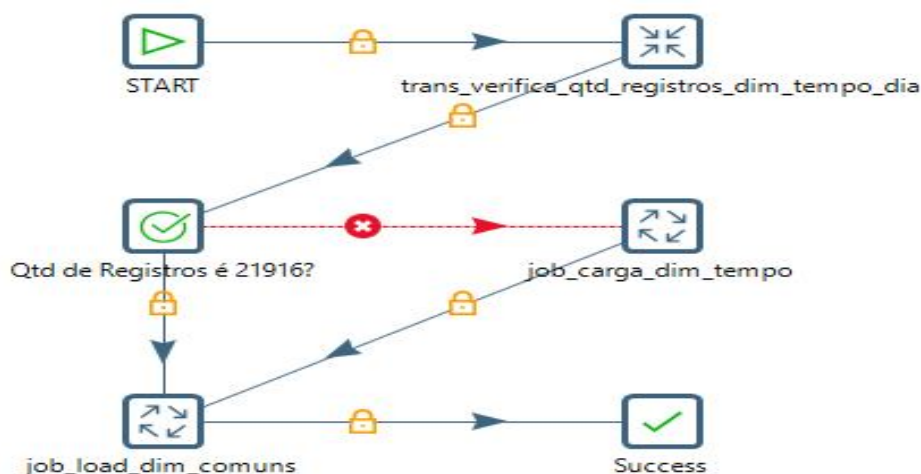


Figura 20 - Job que realiza a carga da dimensão tempo do DW.

Após o DW criado e com a carga completa carregada pelo PDI, e implementado a ferramenta de *Dashboard Designer* (figura 21), demonstramos a criação de uma consulta multidimensional, que assim como os demais Dashboards são salvos em um arquivo XML.

The screenshot shows the Microsoft PivotTools interface. The main window displays a multidimensional query configuration for 'Cubo Recebimento de Grãos'. The data table is as follows:

	2003		2004		2005		2006		
	Quantidade em Kg	Quantidade em Tons	Quantidade em Kg	Quantidade em Tons	Quantidade em Kg	Quantidade em Tons	Quantidade em Kg	Quantidade em Tons	C...
ARROZ			2.037,34	2.037.340,00	8.046,00	8.046.000,00			
MAIZ	118.513,75	118.513.746,00	75.594,66	75.594.664,00	48.023,23	48.023.230,00	145.675,35	145.675.346,00	
SOJA	157.831,02	157.831.020,00	123.329,97	123.329.972,00	140.457,38	140.457.384,00	113.357,71	113.357.712,00	
TRIGO			8.257,69	8.257.686,00	13.304,49	13.304.488,00	5.331,61	5.331.612,00	
Grand Total	276.344,77	276.344.766,00	209.219,66	209.219.662,00	209.831,10	209.831.102,00	264.364,67	264.364.670,00	

**Figura 21 - Criação de consulta multidimensional Recebimento de Grãos, com *Dashboard Designer*.**

Para fazer com que nosso *Dashboard* possa ser consultado pela Web, será necessário que o arquivo XML criado na Figura 21.

O resultado será visto por meio do *Dashboard Viewer* implementado para plataforma ASP.NET (figura 22).

Cubo - Recepción de Granos							
Pivot 1							
	▼ 2003		2003 Total	▼ 2004			2004 Total
	MAIZ	SOJA		ARROZ	MAIZ	SOJA	
AGRÍCOLA HERCULES - CIUDAD DEL ESTE	41.410,48	25.771,99	67.182,47	2.037,34	8.113,46	28.879,47	39.030,27
AGRÍCOLA HERCULES - GUADALUPE	32.026,12	41.889,51	73.915,63		23.715,48	21.389,18	45.747,21
AGRÍCOLA HERCULES - ITAKYRY							
AGRÍCOLA HERCULES	23.099,61	36.794,87	59.894,47		12.815,00	21.680,69	37.017,17

Figura 22 - Dashboard Cubo Recepción de Grãos, sendo executado a partir do IIS.

## 5 CONCLUSÃO

O objetivo buscado com esse trabalho foi alcançado. Diante de tantas necessidades dos gestores e investidores não basta mais relatórios impressos, é preciso de informação para conseguir o conhecimento desejado, e com este conhecimento ser possível tomar decisões que podem fazer a diferença no mercado competitivo de hoje.

Por meio deste trabalho foi desenvolvido um *DW* com modelagem dimensional, *star schema*, utilizando o banco de dados Postgres, realizando a extração dos dados de um ERP com banco de dados Oracle. A carga de dados foi realizada pelo PDI *Pentaho Data Integration*, que provou ser uma ferramenta muito eficiente no que se propõe a fazer, de fácil aprendizado, e grande flexibilidade. E por fim, foi desenvolvido uma ferramenta de designer com componentes *Devexpress* que proporcionou a geração de gráficos e sendo possível compartilhar suas configurações por meio de arquivo *XML*, sendo este arquivo utilizado para publicar sua visualização na Web por meio de aplicação ASP.NET.

Com este processo de modelagem, ETL e ferramentas de análise gerencial, os gestores e investidores poderão ter um acesso mais rápido aos indicadores de desempenho, com maior confiabilidade, pois não estará mais montando a informação, pois ela já estará pronta.



## REFERÊNCIAS

ALEXANDRE, R. N. Fundamentos e Modelagem de Bancos de Dados Multidimensionais. 2007. Disponível em: <<https://msdn.microsoft.com/pt-br/library/cc518031.aspx>> Acesso em: 09 de janeiro de 2017.

DEVEXPRESS. Dashboard. Disponível em: <<https://documentation.devexpress.com/#Dashboard/CustomDocument12049>>. Acesso em 04 de dezembro de 2016.

GREENWALD, R., STACKOVIK, S., STERN, J. Oracle Essencial Base de Dados Oracle 11g, 2009, p.193-236. ISBN - 978-85-7608-223-1.

INMON, W.H. Building Data Warehousing - Como transformar informações em oportunidades de negócios, Primeira edição. Wiley Publishing Inc, 2000. ISBN 0-7645-9944-5. Editora Berkeley 2001. ISBN 85-7251-586-0.

KIMBALL, R., MERZ, R. Data Webhouse. Construindo o data warehouse para a Web, primeira edição 2000. Editora CAMPUS. ISBN- 85-352-0655-8.

KIMBALL, R., ROSS, M. The Data Warehouse Toolkit. Guia completo para modelagem dimensional, segunda edição 2002. Editora CAMPUS. ISBN - 0-471-20024-7.

KIMBALL, R., ROSS, M., THORNTHWAITE, W., MUNDY, J., BECKER, B. The Data Warehouse Lifecycle Toolkit: Practical Techniques Building Data Warehouse and Business Intelligence Systems, 2nd edition. Wiley Publishing Inc., 2008. ISBN 0-470-14977-9.

MACHADO, F. N. R. Tecnologia e Projeto de Data Warehouse: Uma Visão Multidimensional. 4ª Ed. São Paulo: Érica, 2008. 318 p. ISBN: 8536500123.

OLIVEIRA, G., OLIVEIRA E., BI Como Deve Ser, 2016, p.45-121

PASSOS, E.; e GOLDSCHMIDT, R Data Mining: Um Guia Prático, 4ª ed. Elsevier Editora, 2005.

Pentaho Community. Data Integration – Kettle Documentation.. Última documentação atualizada disponível sobre o aplicativo Data Integration da Pentaho Open Source Business Intelligence. Disponível em <<http://wiki.pentaho.com/display/EAI/Latest+Pentaho+Data+Integration+%28aka+Kettle%29+Documentation>>. Acesso em: 12 de janeiro 2017.

PostgreSQL. PostgreSQL. Disponível em: <<https://www.postgresql.org.br/>>. Acesso em 15 de novembro de 2016.

**ANEXOS**

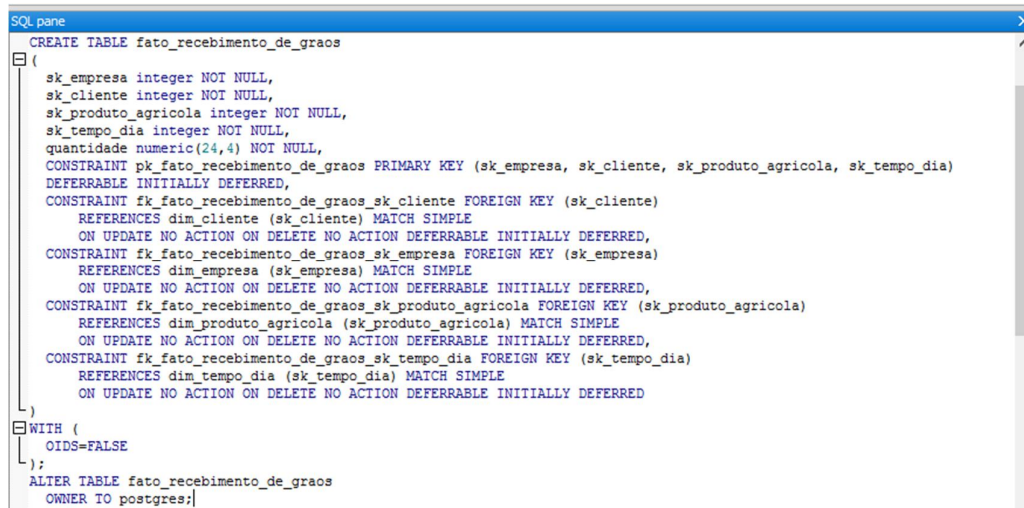
## ANEXO 1 – Comando DDL para criar a tabela dim\_empresa

```

CREATE TABLE dim_empresa
(
  sk_empresa serial NOT NULL, -- Primary Key da dimensão Empresa, Surrogate Key (chave substituta)
  nk_empresa character varying(255) NOT NULL,
  filial character varying(255) NOT NULL,
  cod_empresa integer NOT NULL,
  empresa character varying(255) NOT NULL,
  cod_filial integer NOT NULL,
  cod_grupoempresarial integer NOT NULL,
  grupoempresarial character varying(255) NOT NULL,
  CONSTRAINT dim_empresa_pkey PRIMARY KEY (sk_empresa),
  CONSTRAINT dim_empresa_nk_empresa_key UNIQUE (nk_empresa)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE dim_empresa
  OWNER TO postgres;
GRANT ALL ON TABLE dim_empresa TO public;
GRANT ALL ON TABLE dim_empresa TO postgres;
COMMENT ON COLUMN dim_empresa.sk_empresa IS 'Primary Key da dimensão Empresa, Surrogate Key (chave substituta)';

```

## ANEXO 2 - Comando DDL para criar a tabela fato\_recebimento\_de\_graos



```

CREATE TABLE fato_recebimento_de_graos
(
  sk_empresa integer NOT NULL,
  sk_cliente integer NOT NULL,
  sk_produto_agricola integer NOT NULL,
  sk_tempo_dia integer NOT NULL,
  quantidade numeric(24,4) NOT NULL,
  CONSTRAINT pk_fato_recebimento_de_graos PRIMARY KEY (sk_empresa, sk_cliente, sk_produto_agricola, sk_tempo_dia)
  DEFERRABLE INITIALLY DEFERRED,
  CONSTRAINT fk_fato_recebimento_de_graos_sk_cliente FOREIGN KEY (sk_cliente)
    REFERENCES dim_cliente (sk_cliente) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION DEFERRABLE INITIALLY DEFERRED,
  CONSTRAINT fk_fato_recebimento_de_graos_sk_empresa FOREIGN KEY (sk_empresa)
    REFERENCES dim_empresa (sk_empresa) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION DEFERRABLE INITIALLY DEFERRED,
  CONSTRAINT fk_fato_recebimento_de_graos_sk_produto_agricola FOREIGN KEY (sk_produto_agricola)
    REFERENCES dim_produto_agricola (sk_produto_agricola) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION DEFERRABLE INITIALLY DEFERRED,
  CONSTRAINT fk_fato_recebimento_de_graos_sk_tempo_dia FOREIGN KEY (sk_tempo_dia)
    REFERENCES dim_tempo_dia (sk_tempo_dia) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION DEFERRABLE INITIALLY DEFERRED
)
WITH (
  OIDS=FALSE
);
ALTER TABLE fato_recebimento_de_graos
  OWNER TO postgres;

```