UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
BACHARELADO CIÊNCIA DA COMPUTAÇÃO

ADSON WHOBBERT DA LUZ

COMPARAÇÃO DE FERRAMENTAS DE DATA WAREHOUSE: ESTUDO DE CASO COM DADOS DO IBGE

TRABALHO DE CONCLUSÃO DE CURSO

PONTA GROSSA
2017

ADSON WHOBBERT DA LUZ

COMPARAÇÃO DE FERRAMENTAS DE DATA WAREHOUSE: ESTUDO DE CASO COM DADOS DO IBGE

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação, do Departamento Acadêmico de Informática, da Universidade Tecnológica Federal do Paraná.

Orientadora: Prof. Drª. Simone de Almeida

PONTA GROSSA
2017

TERMO DE APROVAÇÃO

COMPARAÇÃO DE FERRAMENTAS DE DATA WAREHOUSE: ESTUDO DE CASO COM DADOS DO IBGE

por

ADSON WHOBBERT DA LUZ

Este Trabalho de Conclusão de Curso (TCC) foi apresentado em 23 de novembro de 2017 como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

_____
Profa..Drª Simone de Almeida
Orientadora

_____
Prof. Esp. Marcos Vinicius Fidelis
Membro titular

_____
Prof. Drª. Helyane Bronoski Borges
Membro titular

_____
Prof(a). Drª Helyane Bronoski Borges
Responsável pelo Trabalho de Conclusão de Curso

_____
Prof. MSc Saulo Jorge Beltrão de Queiroz
Coordenador do curso

- A Folha de Aprovação assinada encontra-se arquivada na Secretaria Acadêmica -

RESUMO

LUZ, Adson W. **Utilização de ferramentas de business intelligent para tomada de decisão em uma base data warehouse.** 2017. 61f. Trabalho de Conclusão de Curso em Bacharelado em Ciência da Computação – Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2017.

Com a necessidade de obter informações sobre o mercado de trabalho, como os melhores clientes, riscos que a empresa enfrenta, perspectivas da empresa e dos concorrentes, entre outros fatores que pode influenciar na tomada de decisão de um gestor, surgiu a necessidade de um banco de dados que permitisse armazenar e dar acesso rápido a história da empresa, e também que essa informação fosse recuperada de uma forma que facilitasse a análise estatística, surgiu assim o *Data Warehouse*, onde o resultado de seu projeto é, informação disponível para gestão; visão de curvas de comportamento; agilidade de ferramentas para apoio à decisão; segurança de informações para a decisão; maior abrangência de visão de indicadores; recursos mais abrangentes para análise de negócios; necessidades e expectativas atendidas por tecnologia da informação. Ou seja, *Data Warehouse* é um banco de dados voltado para o suporte à decisão de usuários finais. Isso fornece a informação aos gestores, mas é preciso saber como filtrar, como buscar essas informações no DW e esse é o objetivo das ferramentas de BI (*Business Inteligence*). Estas, implementam as técnicas de BI que tem como objetivo evitar surpresas; reduzir a incerteza na tomada de decisão; prever mudanças na indústria, prevenir surpresas tecnológicas; ter melhor entendimento sobre a capacidade atual e futura dos concorrentes, clientes, entidades governamentais, fornecedores, entre outros; avaliar, de forma objetiva e contínua, a posição competitiva atual e futura da empresa; identificar ameaças e oportunidades antes que seus concorrentes o façam. Independente aos diferentes níveis de sofisticação, o processo de IC visa transformar dados em informações. Para isso o IC conta com o apoio de diversas áreas, ou melhor, é formada por várias áreas, sendo elas a tecnologia da informação, produção de inteligência, contra inteligência, Ciência da informação e administração.

Palavras-chave: Data Warehouse. Gestão da Informação. Business Intelligence.

ABSTRACT

LUZ, Adson W. **Use of business intelligent tools for decision making in a data warehouse database.** 2017. 61p. Work of Conclusion Course Graduation in Technology in Systems Analysis and Development - Federal Technology University of Parana. Ponta Grossa, 2017.

With the need to get information about the Market, like best clients, risks the company is taking, the company and competitors prospects, between another factors that can influence the manager decisions, appears a need of a data base that allow to store and give a quick access to the company history, also that these information would be retrieved in a way that make it easy to do the data analysis, that's how DW (Data Warehouse) emerged, where the project result is, information available for management, behavior curve view, agility tools for decision support, and a lot more. Data Warehouse is a data base made for decision support of final users. It gives information to managers, but there is the need to know how to filter these, how to get the information from the DW and thats the point of BI (Business Intelligence) tools. They implement BI techniques that intend to avoid surprises; reduce uncertainty on decision take, predict industry changes, among other functionalities. Apart from the sophistication different levels, the BI process is about to transform data in information. BI has the support from several areas, like Information Technology, Science of information and Management.

Keywords: Data Warehouse. Business Intelligence. Science of Information.

LISTA DE QUADROS

LISTA DE FIGURAS

LISTA DE SIGLAS

BD	Banco de Dados
BI	*Business Intelligence*
CSV	*Comma-separated values*
DM	*Data Mart*
DS	*Data Source*
DW	*Data Warehouse*
ETL	*Extract Transform and Load*
HTML	*HyperText Markup Language*
IBGE	Instituto Brasileiro de Geografia e Estatística
OLAP	*On-Line Analytical Processing*
OLPT	*On-Line Transactional Processing*
SQL	*Structured Query Language*
TI	Tecnologia da Informação
TIC	Tecnologia da Informação e Comunicação
XML	*Extensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO

A obtenção de informações estratégicas, relativas ao contexto de tomada de decisão, é de suma importância para o sucesso de uma empresa. Tais informações permitem à empresa um planejamento rápido frente às mudanças nas condições do negócio, essencial na atual conjuntura de um mercado globalizado. Dessas necessidades surgiram ferramentas para o apoio à decisão, como o *Data Warehouse* (DW), que pode ser definido como uma coleção de dados, orientados por assunto, integrados, variáveis com o tempo e não voláteis, para dar suporte ao processo decisório (INMON, 1993).

Os DWs surgiram com o objetivo de fornecer os subsídios necessários para a transformação de uma base de dados OLTP (*On-Line Transaction Processing*) para OLAP (*On-Line Analytical Processing*) e, assim, prover os elementos necessários a quem toma as decisões nas organizações.

Enquanto Bancos de Dados (BD) transacionais são projetados para trabalhar com informações recentes e operacionais, um DW armazena o histórico empresarial. Este trabalha com informações de longo prazo, normalmente o horizonte fica entre 5 e 10 anos. É formado por várias “partições”, sendo cada uma um conjunto de informações. Em uma empresa uma “partição” pode ser a representação de um setor, sendo chamadas de *Data Mart* (DM) (MACHADO, 2006).

Com a melhoria e grande difusão das Tecnologias da Informação e Comunicação (TICs), a quantidade de informação e velocidade com que é possível acessa-la acaba dificultando sua sintetização por parte dos gestores das empresas, sendo necessário a utilização de recursos tecnológicos que visem facilitar tal atividade, assim como DW e ferramentas de *Business Intelligence* (BI).

BI é o processo responsável pela coleta e interpretação das informações, permitindo filtrar os aspectos relevantes de um determinado problema, auxiliando o decisor, por meio de informações estratégicas, nas suas atividades de planejamento (MENDES, et al 2010). Ferramentas de BI permitem que os usuários construam aplicativos que ajudem as organizações em seus processos de decisão (GIOIA, 2008). Este segmento da Tecnologia da Informação (TI) é bastante dinâmico, uma vez que cada empresa tem necessidades distintas, resultando em um modelo de negócio específico para cada uma, o que define a arquitetura das estratégias empresariais.

Devido a essas particularidades é impossível construir uma aplicação genérica que atenda adequadamente a todas as empresas, surgindo diversas ferramentas de BI disponíveis no mercado, dificultando ao gestor saber qual ferramenta se adequa à sua empresa. Pretende-se neste trabalho selecionar e aplicar ferramentas de BI em um estudo de caso, representado por um *Data Mart* no qual serão realizados experimentos com as ferramentas escolhidas, extraindo as principais caraterísticas de cada uma.

1.1 OBJETIVOS

Esta seção apresenta o objetivo geral como também os objetivos específicos do trabalho.

1.1.1 Geral

Aplicar ferramentas de *Business Intelligence* em um *Data Mart* construído a partir da base de dados do IBGE.

1.1.2 Específicos

- Identificar formas de modelagem de um DW;
- Utilizar a ferramenta *Knowage*, *Pentaho* e *Talend* para apresentar os dados e mostrar o seu funcionamento, identificando suas características principais;
- Extrair informações do DM apresentando os resultados obtidos.

1.2 JUSTIFICATIVA

Sabe-se que esta é a era da Informação, tornando indispensável seu domínio dentro de uma empresa, para que estas auxiliem os gestores na avaliação de mercado (de seus produtos, fornecedores, clientes, concorrentes, etc), gerando vantagem sobre os seus competidores.

Com a difusão das TICs, a quantidade de informação que se pode alcançar é imensa, o que faz com que o mercado fique muito mais dinâmico e complexo. No meio de tanta informação é necessária uma forma de armazena-las, recupera-las, analisa-

las, e a partir desta gerar novos conhecimentos que ajudem a empresa a evoluir e alcançar seus objetivos.

Com a utilização de um DW é possível armazenar essa informação de uma maneira que facilite a recuperação desses dados. As ferramentas de BI filtram e apresentam essas informações de uma forma objetiva, como gráficos e tabelas. Porém, existem diversas ferramentas no mercado, sendo difícil para o gestor saber qual é a ferramenta mais adequada para a sua empresa. Com esse trabalho se espera facilitar essa decisão, realizando o estudo e apresentando as características de duas dessas ferramentas que estão disponíveis.

As ferramentas eleitas para a realização do trabalho são Knowage (SPAGO, 2017) e Pentaho (PENTAHO, 2017), pois são ferramentas gratuitas que possuem grande visibilidade no mercado, com isso pretende-se ter um maior alcance de público, uma vez que não terá a restrição de preço, sendo de fácil acesso a quem pretende implementar uma solução BI.

1.3 ESTRUTURA DO TRABALHO

No Capítulo 2 será mostrado o referencial teórico deste trabalho, apresentando as principais características de um DW, seus elementos e um estudo de caso onde foi implementado um DW. No Capítulo 3 é apresentado a implementação do DW e a aplicação das ferramentas de BI, detalhando as tecnologias escolhidas, e o passo a passo do desenvolvimento. No Capítulo 4 é mostrado os resultados obtidos com a implementação do DW e a aplicação das ferramentas escolhidas.

2 DATA WAREHOUSE

Nesse Capítulo é detalhado o processo de construção de um DW, estando organizado em 3 Seções, sendo que na Seção 2.1 é apresentado os principais elementos do DW, na Seção 2.2 é discutida as abordagens mais utilizadas para o desenvolvimento do DW. Na Seção 2.3 é feito um estudo de caso para exemplificar o conteúdo das duas seções anteriores. A última Seção 2.4 apresenta as considerações do capítulo identificando os conceitos a serem utilizados neste trabalho.

2.1 PRINCIPAIS ELEMENTOS

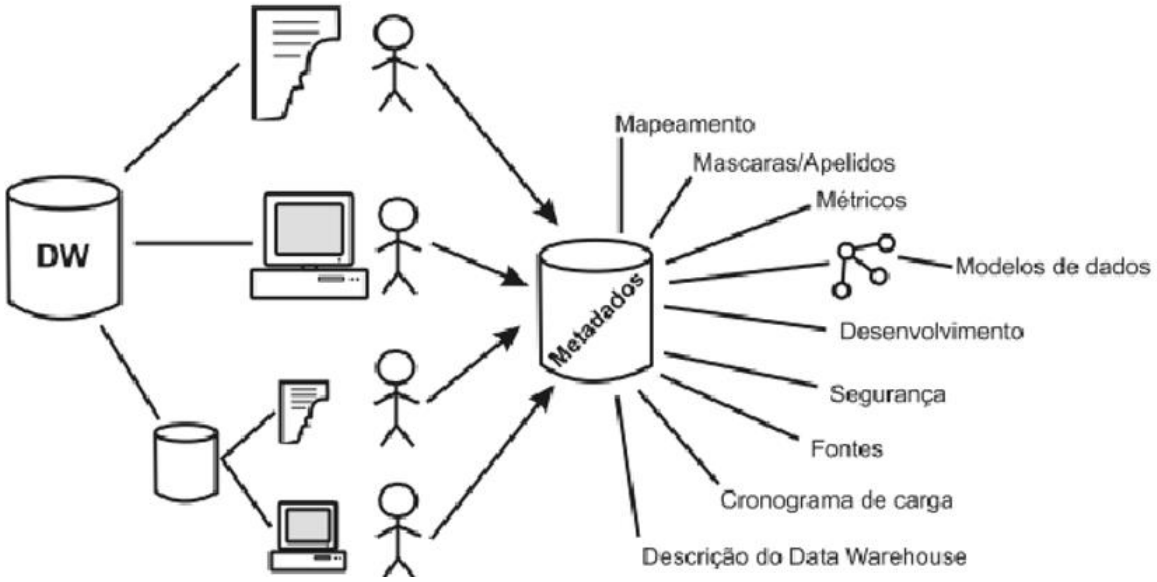
Segundo Machado (2004) um DW é constituído, basicamente, por três elementos:

- Fatos: São coleções de itens de dados, sendo representado com valores aditivos. Fatos são na verdade registros das transações ocorridas nos bancos de dados transacionais. Por exemplo, em uma revendedora de carros um fato seria as vendas ocorridas.
- Dimensões: são agrupadas em 4 tipos de elementos, tais como: “quando foi realizado”, “onde foi realizado”, “quem realizou” e “o que foi realizado”. Esses elementos são implementados em tabelas de dimensões. Na mesma revendedora de carros citada anteriormente, uma dimensão seria o intervalo de tempo em que elas foram efetuadas, como exemplo as vendas que ocorreram mensalmente.
- Medidas: são os atributos numéricos que representam os fatos. Uma medida é determinada pela combinação das dimensões que participam de um fato e estão localizados como atributos deste. Por exemplo, o valor em reais das vendas, o número vendido de unidades de produtos e a quantidade em estoque.

Outros itens importantes de um DW são os metadados e a granularidade. Metadado pode ser considerado o índice de um DW, seu repositório deve conter dados como a origem dos dados armazenados no DW, regras de negócio, regras de transformações dos dados, apelidos e formatos dos dados; metadados nada mais são

que dados sobre os dados (KIMBALL, ROSS, 2011). Na Figura 1 é possível visualizar a relação entre o DW e os metadados.

Figura 1 - Relação entre o DW e os Metadados



Fonte: Machado (2004, p.27)

Segundo (BARQUIN,1997), as vantagens dos metadados são a redução de complexidade, redução da possibilidade de erros e uma melhor captura do ambiente de dados operacionais e fatoração do tempo.

Granularidade, que é o nível de detalhamento do DW, se este vai conter dados diário, semanal, quinzenal etc. É mais um aspecto relevante do projeto de um DW, pois uma granularidade baixa, aumentará a quantidade de dados armazenados no DW, e uma granularidade alta limitará as pesquisas por ter poucos detalhes. Por exemplo, se é escolhida uma faixa de tempo de um mês para a coleta de dados, mas o gestor precisa realizar uma comparação entre a primeira e segunda quinzena de cada mês, o DW não será capaz de apresentar a informação correta, afinal para obter a informação solicitada o DW faria a média do valor mensal.

Por outro lado, se é escolhida uma faixa de tempo diária, a quantidade de informação cresceria muito, deixando o DW mais lento e contendo praticamente a mesma informação que um banco de dados transacional, perdendo assim a vantagem proposta pelo DW (INMON, 1999).

2.2 IMPLEMENTAÇÃO DE UM DW

Nessa Seção será abordado os principais elementos para a implementação de um DW. Na Seção 2.2.1 é explicada as abordagens para a construção de um DW, na Seção 2.2.2 é discorrido sobre como realizar o processo de ETL a partir de uma planilha eletrônica, na Seção 2.2.3 é explanado os modelos mais comuns de DW, por fim na Seção 2.2.4 são apresentados alguns tipos de DW.

2.2.1 Abordagens

Nas abordagens tratadas a seguir será usado o processo de Extração, Transformação e Carregar/*Extract Transform Load* (ETL), fase essencial para criar um DW, devido aos dados poderem ser obtidos de diferentes fontes, isso pode ser feito tanto comprando uma ferramenta que realize a tarefa do ETL ou construir uma própria. A primeira alternativa economiza tempo a segunda dinheiro, a situação intermediária seria selecionar uma ferramenta de código aberto e adapta-la de acordo com o contexto do problema em questão. Trujillo (2003) propôs 6 passos para a execução do ETL:

1) selecionar fontes;
2) transformar as fontes;
3) unir as fontes;
4) selecionar alvos;
5) mapear as fontes para os atributos alvo;
6) carregar dados.

Para melhor entender as etapas acima, suponha que uma construtora deseja abrir um novo empreendimento, para saber qual a melhor localização e características deste. Assim resolvem fazer um DW, a primeira etapa acontece quando a empresa precisa escolher quais as fontes de dados que irão alimentar o DW. Nesse processo foi eleito dados obtidos da prefeitura da cidade. Outros dados como por exemplo o perfil de consumo dos cidadãos, são identificados por meio de uma pesquisa. A segunda etapa serve para estabelecer um padrão nos dados, ou seja, normalizando-os. Os dados de sexo podem estar descritos como “Feminino” ou “Masculino”, ou

ainda como “F” ou “M”, esta etapa irá definir uma nomenclatura padrão para representar o conteúdo do atributo sexo.

A terceira etapa une todas as fontes, passando em seguida para a quarta etapa que define quais serão os DMs desenvolvidos, que comporão o DW. A quinta estará concluída quando estiver definida a forma de mapeamento das fontes dos dados e seu atributo alvo, a última etapa consiste em popular os DMs.

Pode-se implementar um DW por meio de 3 abordagens: *top down, botton up, combine* (MACHADO, 2006).

- *Top down* é a que exige maior planejamento, iniciando pela extração, transformação e integração dos dados, passando então para o DW onde são extraídos os dados e metadados para os DMs. A vantagem dessa abordagem é que há consistência do DW, com uma padronização dos DMs.
- *Botton up* é requisitada pelos gestores por obter um produto em um tempo menor em relação a abordagem *Top down*, nela é realizado o processo de extração, transformação e integração dos dados para um DM específico implementando-o. Esse processo é repetido para cada DM até que o DW seja concluído. Essa forma incremental, realizando vários DMs para então integrar em um único DW, pode despadronizar os DMs ameaçando assim a consistência do DW.
- *Combine* nada mais é que uma combinação das duas abordagens anteriores, sendo realizado o processo de extração, transformação e integração dos dados para o DW completo e então desenvolvido os DMs individualmente.

2.2.2 ETL a Partir de uma planilha eletrônica

Devido aos usuários se sentirem confortáveis ao utilizar o Excel, ou outros aplicativos que não fornecem a opção de armazenar a informação, muitas fontes de dados estão em tabelas do Excel. Isso acaba gerando algumas complicações na etapa do ETL, como tipos incorretos de dados, devido ao tratamento de tipos de dados do Excel ser menos sofisticado que em bancos relacionais ou em ferramentas de ETL, a possibilidade de esconder dados no Excel pode fazer com que as ferramentas não encontrem esses dados.

Como não é possível declarar o tipo do dado no Excel, quando esses são lidos pelas ferramentas, os tipos são definidos de acordo com o conteúdo contido nas células. De acordo com Ramos (2011), os tipos são definidos de 4 maneiras:

DT_R8: dado numérico sem outras distinções quanto ao tipo (*integer*, *currency*, *value*) ou tamanho;

DT_DATE: datas e horas;

DT_WSTR: *strings* de até 4.000 caracteres;

DT_NTEXT: *strings* com mais de 4.000 caracteres;

Apesar de não ter encontrado nenhuma documentação oficial afirmando tal, segundo (RAMOS, 2011), é assim que acontece o critério de assinatura:

- Se todos os valores são número, DT_R8 é assinado.
- Se todos os valores são datas e/ou horas, DT_DATE é assinado.
- Se contém alguma *string* de até 255 caracteres, DT_WSTR é assinado.
- Se contém alguma *string* com mais de 255 caracteres, DT_NTEXT é assinado.

Para Ramos (2011), os passos apresentados a seguir devem ser seguidos para previnir tipos incorretos:

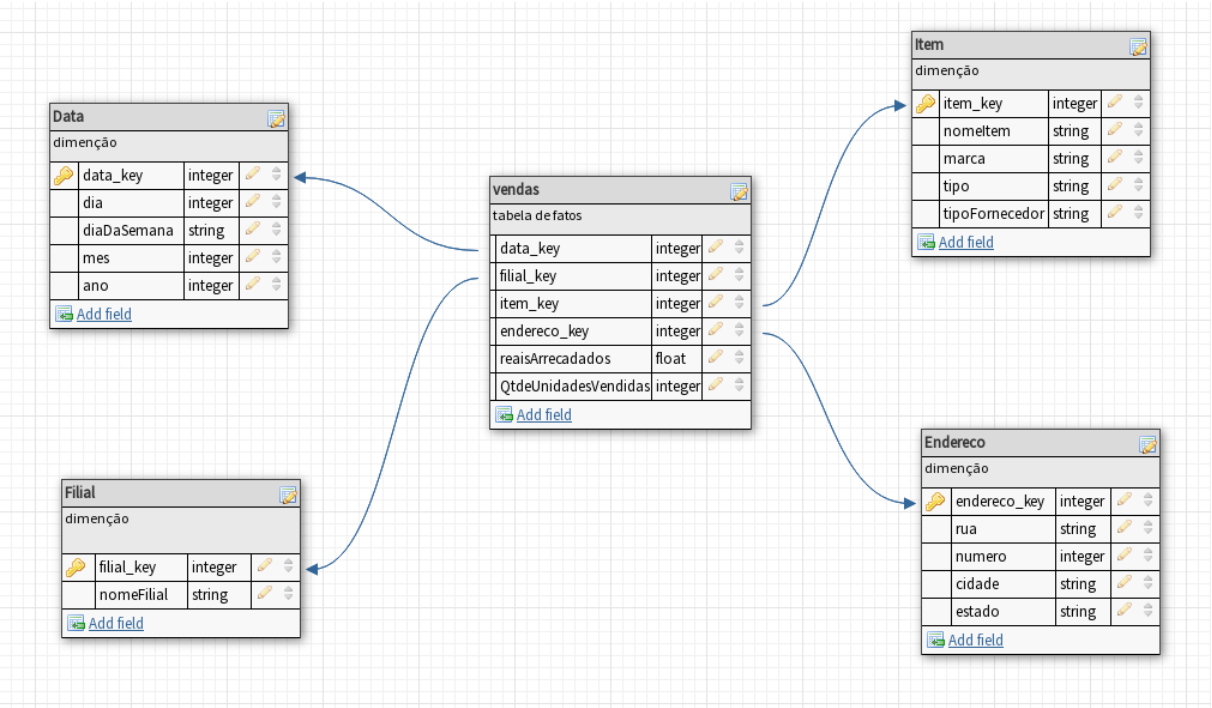
- Criar um arquivo de exemplo do excel para ajudar no processo de assinatura automática do tipo de dados;
- Realizar qualquer mudança necessária nas definições de tipo de dados adicionando fonte OLE DB e Componentes de fonte do Excel;
- Utilizar Conversão de dados e transformações de coluna derivada para fazer qualquer outra mudança necessária para obter os tipos de dados corretamente.

2.2.3 Modelos

Além das abordagens apresentadas na Seção 2.2.1 é necessário também definir qual o modelo que será usado para a implementação do DW, como este estará organizado. A seguir serão descritos os principais modelos/*schemas:*

- Modelo estrela/*star schema:* é constituído de uma tabela de fatos rodeado por tabelas de dimensões, cada dimensão nesse modelo é representada por apenas uma tabela. Isso pode ser visualizado na Figura 2.

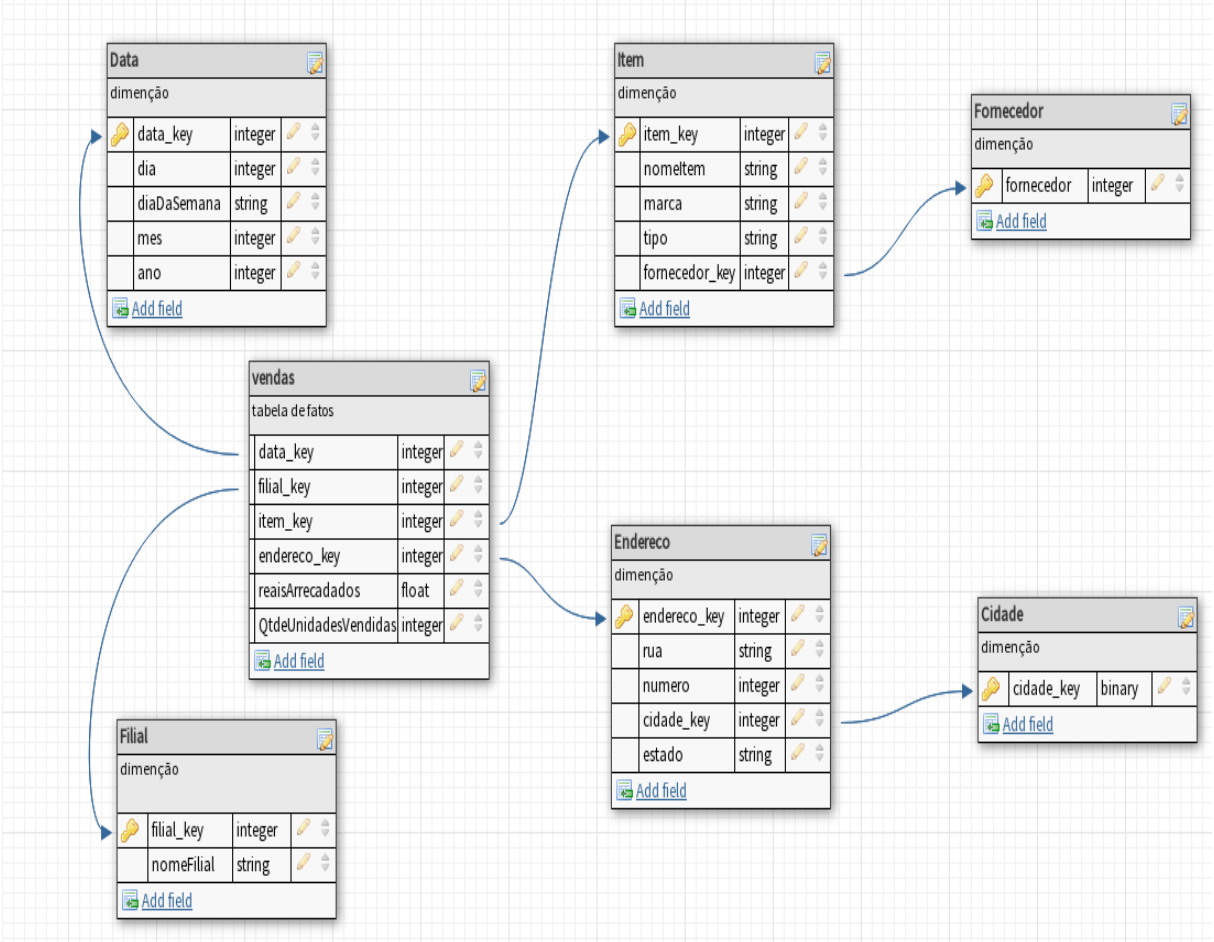
Figura 2 - Exemplo modelo estrela/star schema



Fonte: Autoria própria (2017)

- O modelo floco de neve/*snowflake schema* é similar ao modelo estrela, contém apenas uma tabela de fatos, mas as suas tabelas de dimensões podem ser normalizadas. Na prática significa que uma dimensão pode ser definida por meio de mais de uma tabela, como pode ser observado na Figura 3. A normalização da tabela ITEM fez com que esta fosse dividida em mais uma tabela dimensão, a FORNECEDOR, no caso as duas tabelas apresentam o produto que foi vendido, o mesmo ocorre com a tabela ENDERECO.

Figura 3 - Exemplo modelo floco de neve/snowflake schema

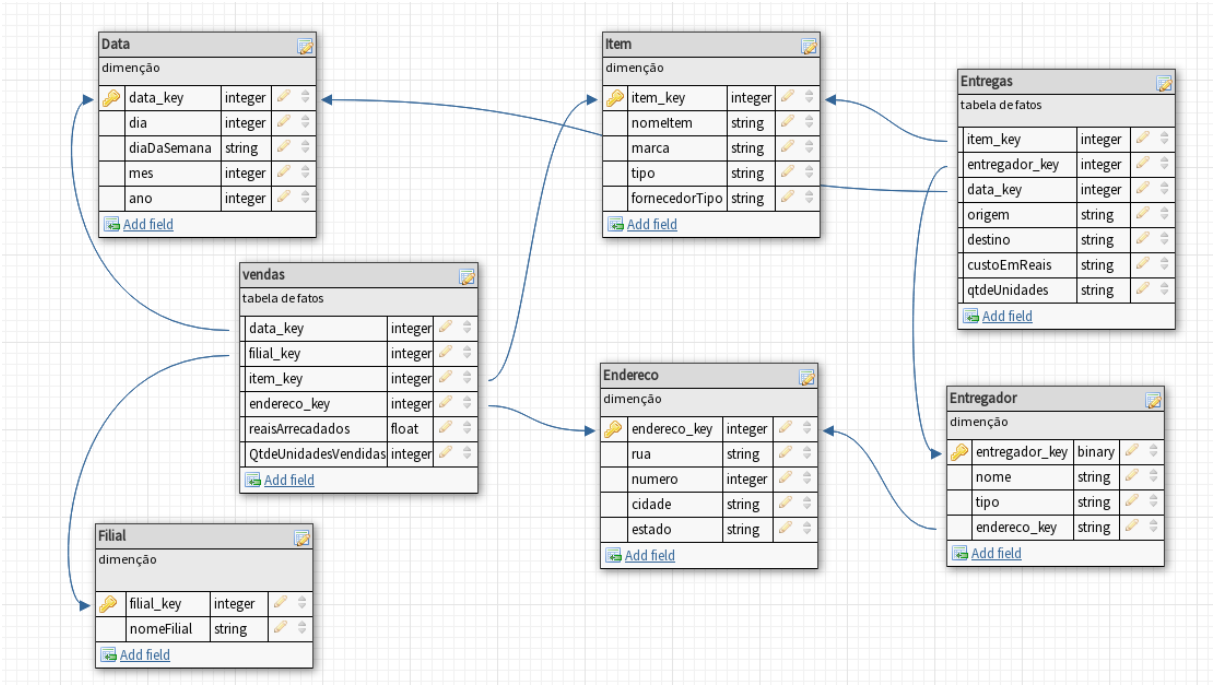


Fonte: Autoria própria (2017)

A normalização faz com que a redundância seja diminuída, facilitando a sua manutenção e economizando espaço de armazenamento.

- O modelo galaxia/*galaxy schema* ou também modelo constelação de fatos/*fact constellation schema* diferencia dos anteriores pois permite mais de uma tabela de fato. A Figura 4 exemplifica esse modelo, onde a tabela “Vendas” e “Entrega” são tabelas fato.

Figura 4 - Exemplo do Modelo Galaxia/galaxy schema



Fonte: Autoria própria (2017)

2.2.4 Tipo de *Data Warehouse*

Os 3 tipos de DW que serão discutidos são: processamento de informação/*information processing*, processamento analítico/*analytical processing* e mineração de dados/*data mining*.

- Processamento de informação: DW que permite processar os dados nele contido. Os dados podem ser processados por *queries*, análises estatísticas básicas, relatórios usando *crosstabs*, tabelas, gráficos ou grafos;
- Processamento analítico: o DW suporta processamento analítico dos dados nele contido. Os dados podem ser analisados por operações OLAP básicas, incluindo *drill-down*, *drill-up*, *slice-and-dice* e pivoteamento.
  - *Drill-down* refere-se a operação de sair do topo de uma hierarquia em direção ao dado mais detalhado.
  - *Drill-up* faz o caminho inverso ao *drill-down*.
  - *Slice-and-dice* é a operação de rearranjar os dados, para que se tenha diferentes perspectivas ao analisa-los. Pivoteamento, ou rotação, ocorre

a mudança dos eixos das dimensões para fim de visualização (VASCONCELOS et al., 2010).

- Pivoteamento é a rotação do eixo das dimensões afim de mudar a visualização dos dados.

- Mineração de dados: permite que se descubra informações através da busca de padrões e associações escondidas, construção de modelos analíticos, realizar classificações e predições. Os resultados dessa mineração podem ser apresentados por ferramentas de visualização, como SpagoBI (atualmente Knowage) ou Pentaho.

2.3 ESTUDO DE CASO

Para exemplificar as informações fornecidas nas Seções 2.1 e 2.2 será utilizado um estudo de caso realizado por Gura e Benck (2011), desenvolveram um DM para uma farmácia de manipulação que faz medicamentos personalizados, ou seja, o médico receita o princípio ativo e a quantidade que o paciente necessita, podendo até misturar mais de um princípio ativo em uma mesma cápsula.

O primeiro passo foi realizar um *brainstorming* com o cliente para levantar quais eram as informações relevantes que a farmácia desejava com o DW, para isso foram identificadas as questões:

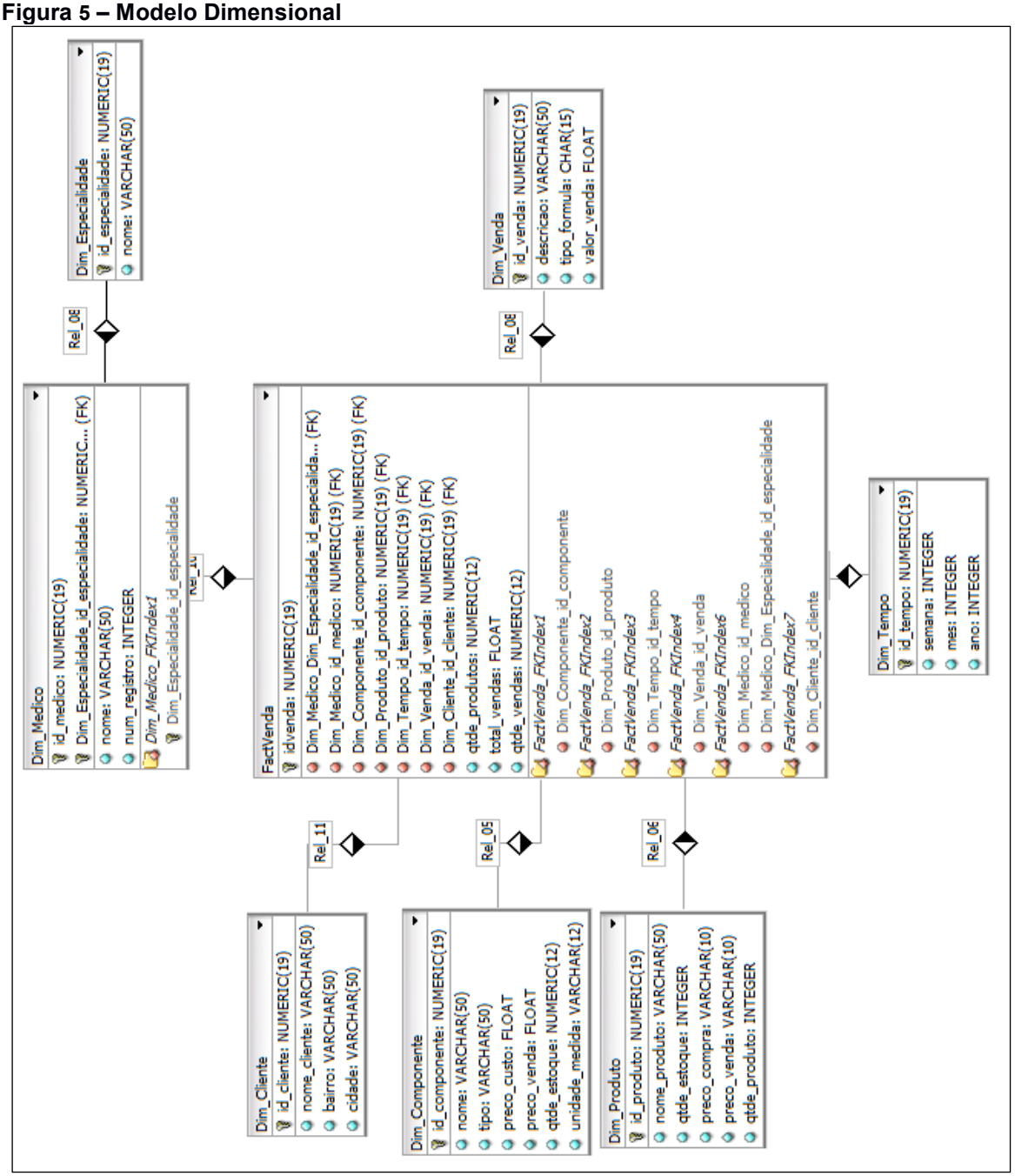
- Qual o faturamento por médico, em um mês ou um ano?
- Qual o faturamento por especialidade médica?
- Qual o faturamento de um tipo de componente?
- Qual o faturamento por cliente em um mês ou um ano?
- Qual o componente mais vendido por especialidade médica?
- Qual o mês de maior e menor faturamento em 2011?
- Qual o faturamento em 2010?

Com essas perguntas e com uma análise no modelo relacional do banco de dados da empresa, foi possível realizar o modelo dimensional apresentado na Figura 5. No modelo pode-se observar que a topologia usada foi a *Snow Flake,* pois a dimensão “Médico” é normalizada com a dimensão especialidade.

2.4 CONSIDERAÇÕES

Com os estudos abordados neste Capítulo, foi possível identificar e escolher que para o presente trabalho será utilizado a abordagem *Bottom Up.*

Figura 5 – Modelo Dimensional



Fonte: Gura e Benck(2011)

A plataforma Pentaho foi utilizada para auxiliar em alguns processos, como a criação do xml (*Extensible Markup Language*) do modelo dimensional e também para o processo de ETL, este foi feito com o pacote *data-integration* do Pentaho.

No processo de transformação, algumas dimensões foram transformadas individualmente, outras como “médico” e “especialidade”, foram tratadas em conjunto. O processo de transformação, na maioria dos casos consiste em ordenar os dados e depois carregar os mesmos nas dimensões do DW, em alguns casos mais de uma tabela incide em um mesmo alvo, questão que é resolvida na etapa 5 (mapear as fontes para o atributo alvo) do processo ETL descrito na Seção 2.2.1.

Durante a verificação dos dados, notou-se que alguns registros estavam incompletos, como por exemplo datas não registradas, por isso alguns desses atributos foram removidos. Além disso, também foi necessária uma transformação complexa na data da dimensão tempo, pois no BD esta estava armazenada em uma única *string*, por isso foi extraído o ano, mês e dia dessa *string*, e ainda transformar os dias em semanas para a recuperação de informações semanais.

Como dito anteriormente a dimensão médico e a dimensão especialidade foram agrupadas no mesmo processo. O primeiro passo é selecionar, na tabela de origem os dados que serão carregados no DW, a SQL. O Quadro 1 exemplifica a seleção dos dados referente a dimensão médico:

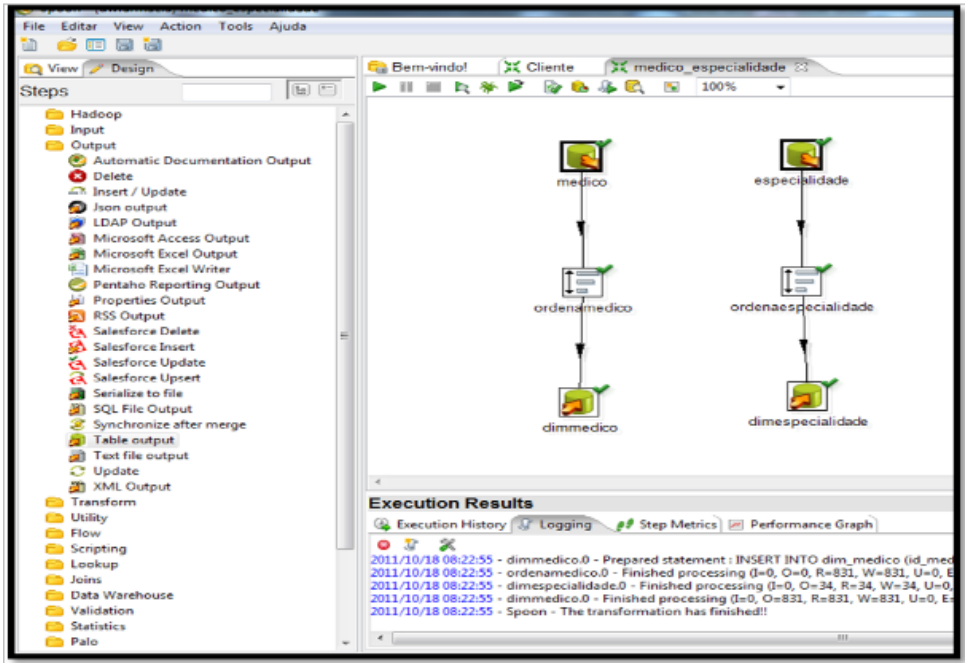
Quadro 1 - Comando de Seleção da Dimensão Médico

```
select m.cod_medico as id_medico,
m.nome as nome_medico,
m.nro_registro as num_registro,
e.cod_especialidade as id_especialidade
from medico m
join especialidade e on e.cod_especialidade =
m.cod_especialidade
order by m.cod_medico
```

Fonte: Gura e Benck (2011)

Os dados resultantes dessa consulta são usados em um comando *insert* na tabela destino. O processo realizado para a dimensão médico e especialização pode ser observado na Figura 6.

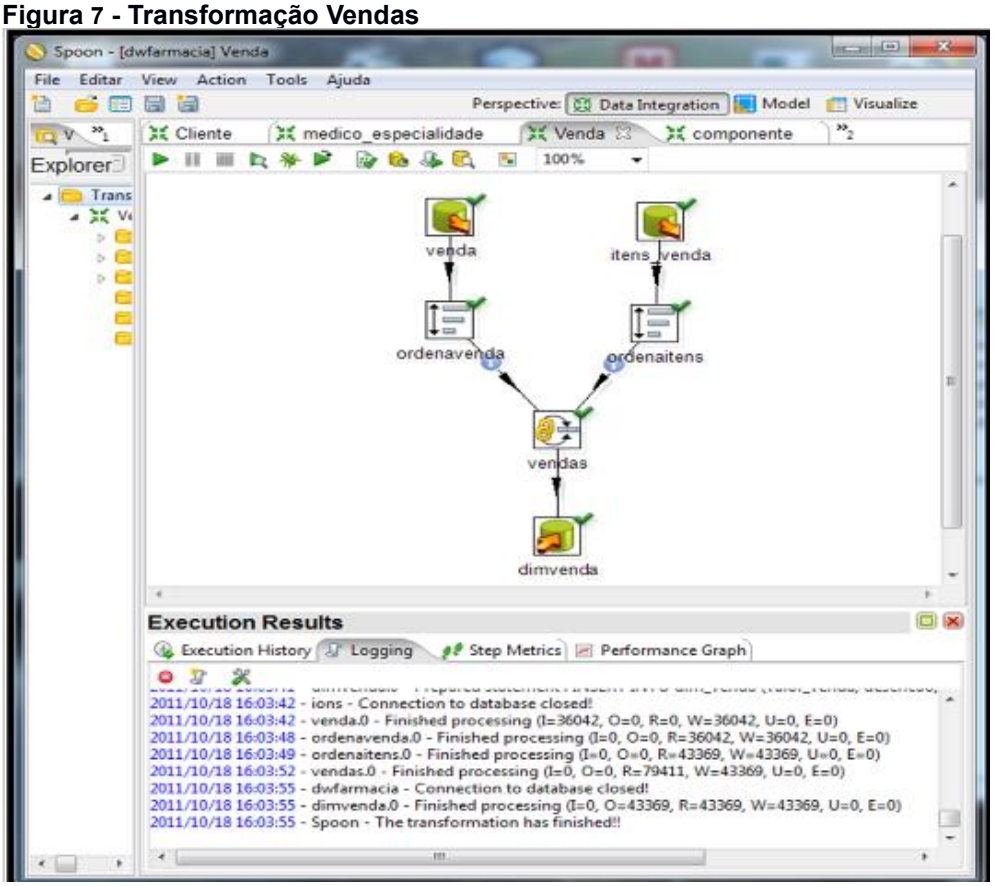
Figura 6 - Transformação Médico e Especialidade



Fonte: Gura e Benck (2011)

Em casos como o da dimensão venda é necessário utilizar o *mergejoin,* pois a fonte dessa dimensão é obtida de duas tabelas, venda e itens_venda. Essa transformação pode ser observada na Figura 7.

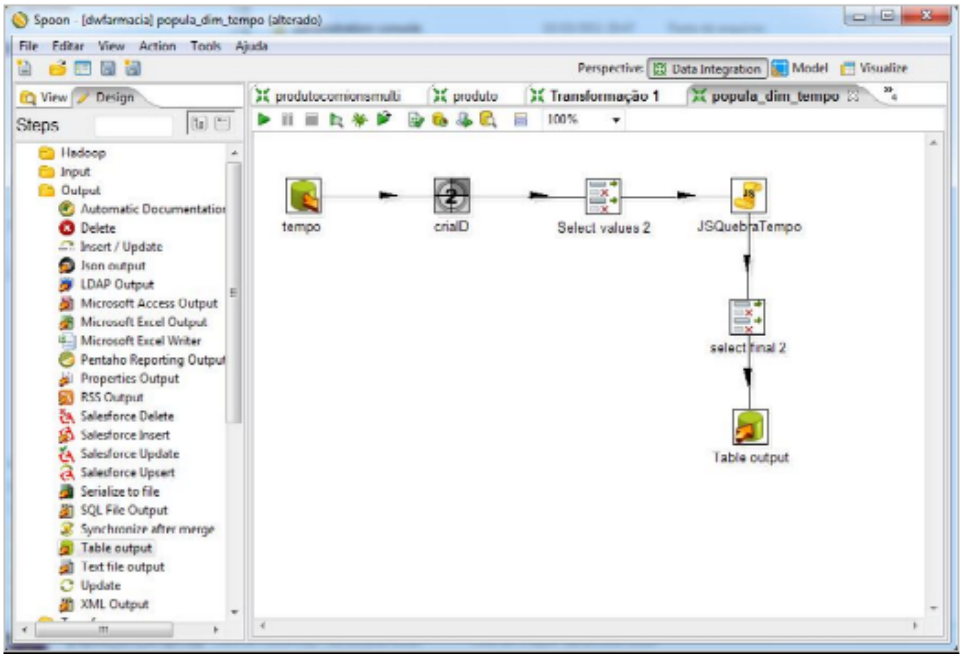
Figura 7 - Transformação Vendas



Fonte: Gura e Benck (2011)

Outra transformação importante a ser observada é a da dimensão tempo, por ser necessário dividir a *string* data. Na Figura 8 observa-se que foi criado o *id* da dimensão, após isso são selecionados os valores, então passa pelo processo quebra tempo, que consiste em dividir a data. Por fim os campos são selecionados e inseridos na dimensão.

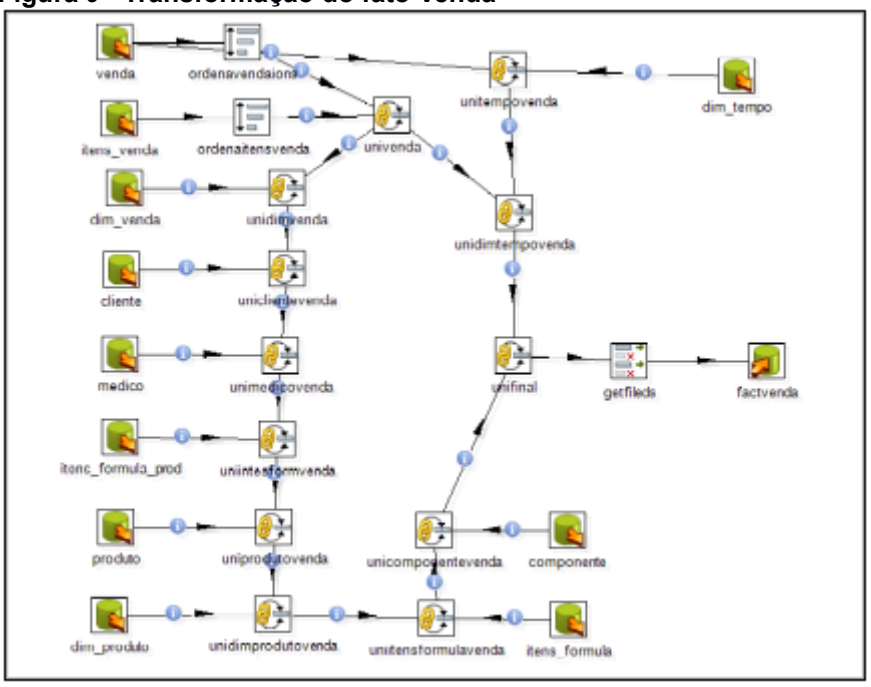
Figura 8 - Transformação da dimensão Tempo



Fonte: Gura e Benck (2011)

A Figura 9 representa a transformação do fato venda, onde são feitos os *joins* de todas as dimensões e das tabelas que foram necessárias para a composição do mesmo.

Figura 9 - Transformação do fato Venda



Fonte: Gura e Benck (2011)

Após finalizado o processo de ETL, os dados estão carregados no DM e este está pronto para que sejam realizadas as análises.

3 FERAMENTAS DE DW

No mercado existem várias ferramentas de DW disponíveis, dentre elas estão ferramentas pagas, gratuitas e de código aberto, cada uma com suas características e recursos que as diferem. Esta variedade é um ponto positivo, uma vez que dificilmente não haverá uma ferramenta que atenda aos requisitos específicos da empresa. Assim, este capítulo é dividido em três seções, sendo que a Seção 3.1 apresenta a ferramenta Talend, a Seção 3.2 aborda a ferramenta Pentaho, a Seção 3.3 mostra a ferramenta Knowage e a seção 3.4 faz as considerações do capítulo.

3.1 TALEND

Talend Open Studio provê soluções para integração de dados, *Big Data* entre outros, foi escolhida para o presente estudo por possuir uma curva de aprendizagem acelerada, facilitando o processo de ETL. A empresa foi fundada em 2005, hoje é aceita por boa parte da comunidade, sendo utilizada por grandes empresas como Air France e Lenovo, além de ser ganhadora de vários prêmios como DBTA 100 e *Big Data Award* (TALEND, 2017)

3.2 PENTAHO

Pentaho é uma plataforma que permite acessar, integrar, manipular, visualizar e analisar os dados. Estes dados podem estar armazenados em um banco de dados relacional, bancos analíticos, *clusters hadoop* ou bancos NoSql (PENTAHO, 2017).

Os produtos da Pentaho consistem nos componentes de BA (*Business Analytics*) e DI (*Data Integration*):

- *Business Analytics*: Permite a criação de relatórios, e *dashboards*, baseado no seu modelo de dados.
- *Data Integration*: Permite a execução completa do processo de ETL, usando um formato consistente e uniforme que é acessível e relevante para usuários finais e tecnologias de internet das coisas.

3.3 KNOWAGE

Knowage é a nova versão da conhecida ferramenta SpagoBI, a versão antiga (Spago) possuiu 5 versões, sendo assim a ferramenta Knowage, visando mostrar que é a nova versão desta, iniciou sua versão no número 6. Knowage possui duas versões, uma paga e uma gratuita e de código aberto, seguindo o legado deixado por sua predecessora (SPAGO, 2017).

Knowage na verdade é um conjunto de ferramentas possuindo duas suítes:

Knowage Server: um servidor com uma suíte completa, que permite criar desde Data Sources até relatórios, é a ferramenta principal do conjunto.

Knowage Report Designer: Utilizado para criar relatórios e depois publica-los no servidor

3.4 CONSIDERAÇÕES

Uma vez que um dos objetivos do presente trabalho é facilitar a introdução ao *Data Warehouse*, foi escolhido apenas ferramentas gratuitas ou pelo menos versões gratuitas de ferramentas pagas, não impedindo o uso por questões financeiras. Além disto, foi levado em consideração a aceitação das ferramentas pela comunidade, pois assim também não seria difícil encontrar apoio de outros usuários, caso seja necessário (Devmedia, 2017).

4 IMPLEMENTAÇÃO

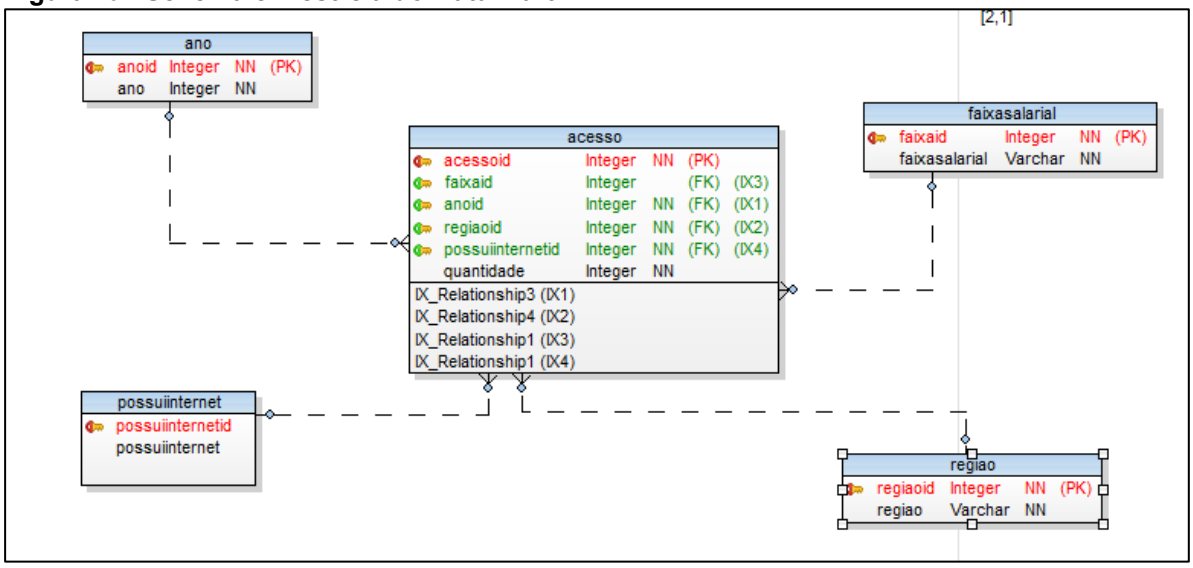
Este capítulo apresenta a implementação do DW, na Seção 4.1 é explicado como foi feita a criação do DM, a Seção 4.2 mostra o processo de ETL executado na ferramenta *Talend Open Studio*, a Seção 4.3 apresenta a aplicação das plataformas de BI Pentaho, e na Seção 4.4 está descrito a aplicação da plataforma de BI Knowage. Por fim a Seção 4.5 apresenta a conclusão da aplicação das ferramentas de BI.

4.1 CRIAÇÃO DO DM

Utilizou-se como fonte de dados, as bases de pesquisas disponibilizadas pelo IBGE em relação a conexão à internet pela população brasileira. Nas fontes disponibilizadas é possível saber a quantidade de domicílios com acesso à internet em cada região brasileira e pela faixa salarial. O instituto disponibiliza dados sobre o acesso à internet dos anos 2005, 2008, 2009, 2011, 2013, 2014 e 2015.

Com os dados descritos acima, foi criado o modelo de um DM em estrela, no qual a tabela fato seria o acesso à internet, nela estão contidas as informações de quantos domicílios estão agrupados, nas tabelas de dimensão estão alocados dados de ano, região, e faixa salarial, e se esse grupo possui ou não acesso à internet. O Esquema pode ser visualizado na Figura 11:

Figura 10 - Schema em estrela do Data Mart



Fonte: Autoria própria (2017)

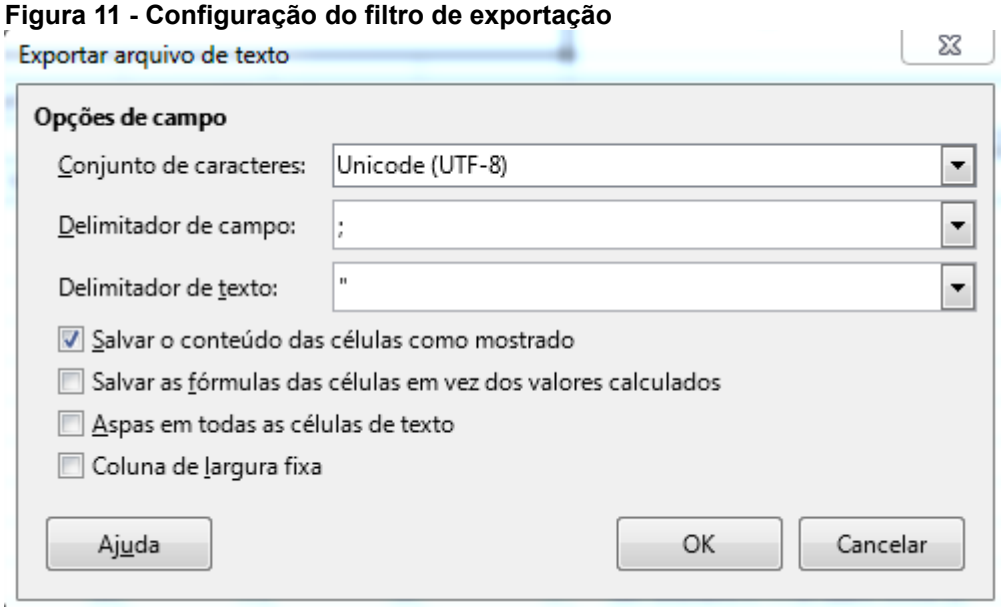
Ao criar o DM, uma vez que os dados de ano, região e faixa salarial são pré-definidos em todas as tabelas que foram utilizadas, as tabelas de dimensão foram carregadas já no *script* de criação. Além do DM foi criado um BD onde foi importado todos os dados obtidos do *site* do IBGE.

4.2 EXTRAÇÃO, TRANSFORMAÇÃO E CARREGAMENTO

Os dados, adquiridos no *site* do IBGE (2017), estão apresentados em formato ODS, que é um formato de planilha eletrônica, esses dados, a partir de 2013 estão organizados por assunto. Todas as tabelas de um mesmo assunto estão contidas em um único arquivo, nomeado pelo assunto. Os arquivos anteriores a 2013, estão organizados também por assunto, porém cada tabela está em um arquivo separado, nomeado por um código numérico.

Para poder transferir esses dados para o Banco de dados PostgreSQL, foi realizada a conversão dos arquivos para o formato de texto CSV, e então alterada a disposição dos dados para facilitar o processo de extração. A conversão dos arquivos foi feito pela ferramenta Calc(*Libre Office*), no momento da conversão é importante que selecione a opção “Editar as configurações do filtro”. O filtro de conversão foi configurado como descrito na Figura 12.

Figura 11 - Configuração do filtro de exportação

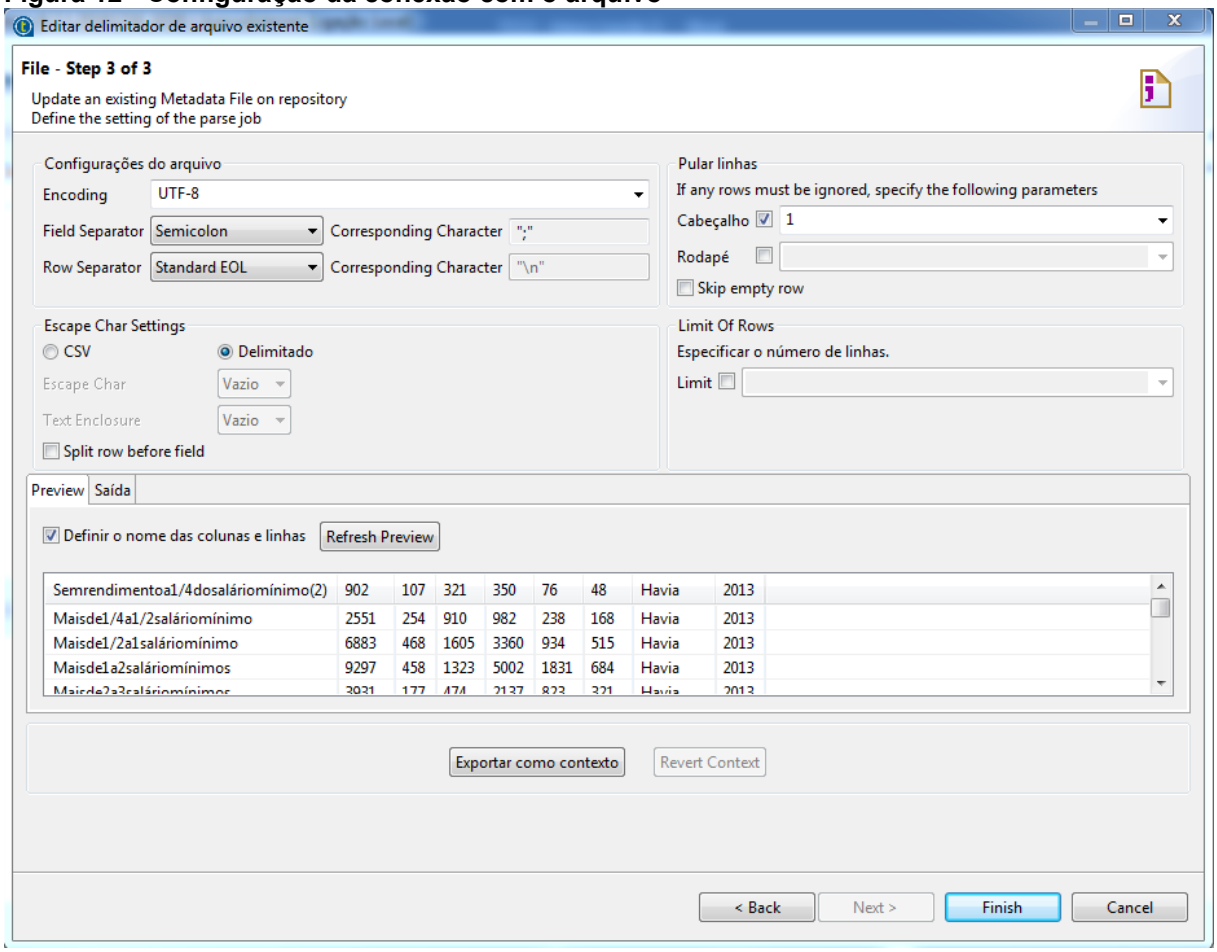


Fonte: Autoria própria (2017)

Após finalizada a conversão, foi necessário realizar algumas alterações nos arquivos, pois continham várias informações que não eram úteis para o estudo, além de campos numéricos utilizando vírgula como ponto decimal.

O processo de ETL foi feito a partir da ferramenta *Talend*. O primeiro passo foi criar um *link* com o arquivo CSV e com o DM criados na etapa anterior. Para a criação do *link* com o arquivo, foi necessário criar dentro da ferramenta um arquivo delimitado, setar o caminho até o arquivo CSV, e as configurações são feitas parecidas com o que foi feito ao gerar o arquivo na etapa anterior, como pode ser visto na Figura 13.

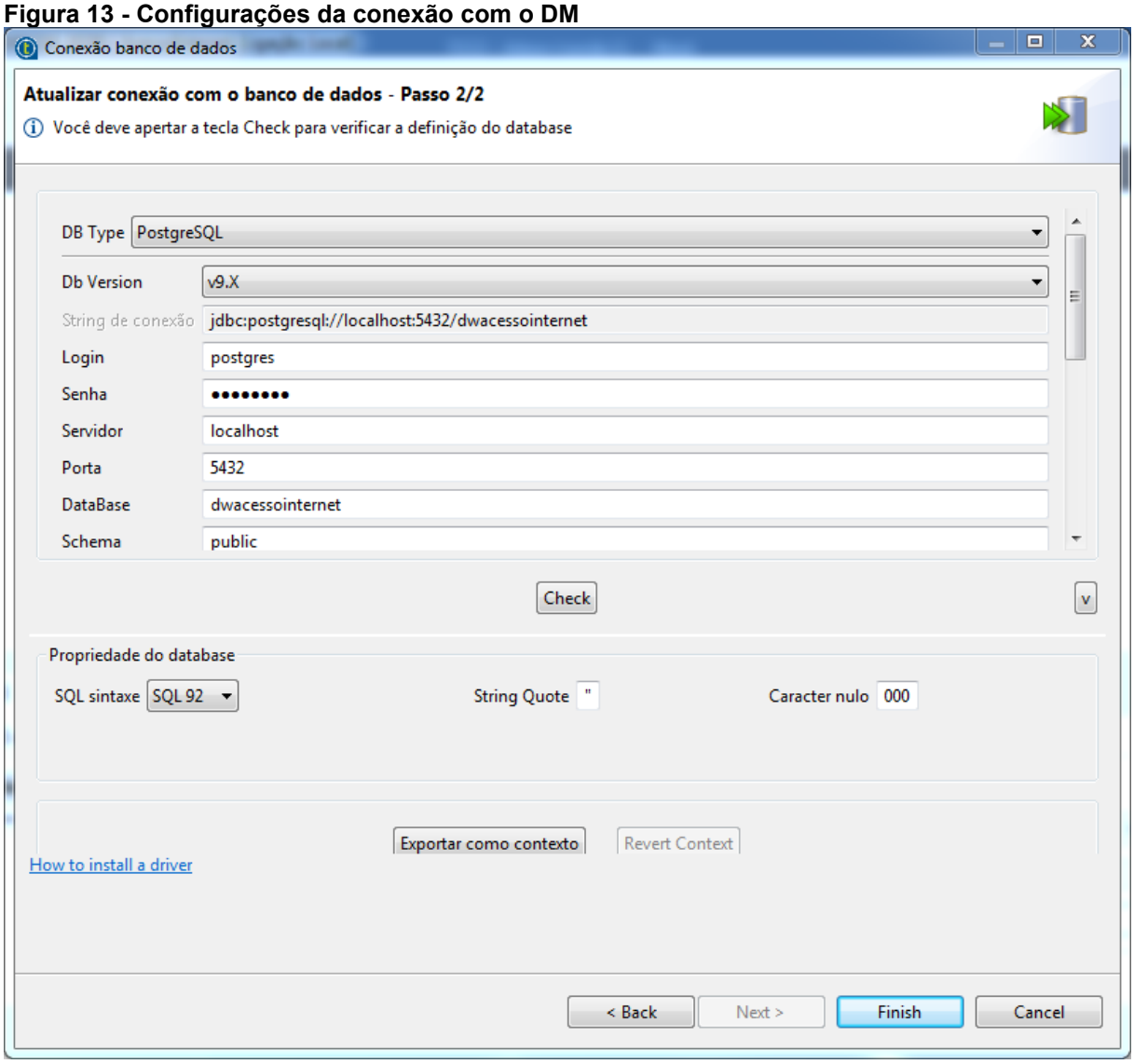
Figura 12 - Configuração da conexão com o arquivo



Fonte: Autoria própria (2017)

A conexão com o DM segue um processo similar, basta apenas criar uma conexão com o banco de dados, denominada Db *Connection*, e colocar os dados de acesso, as configurações podem ser vistas na Figura 14.

Figura 13 - Configurações da conexão com o DM

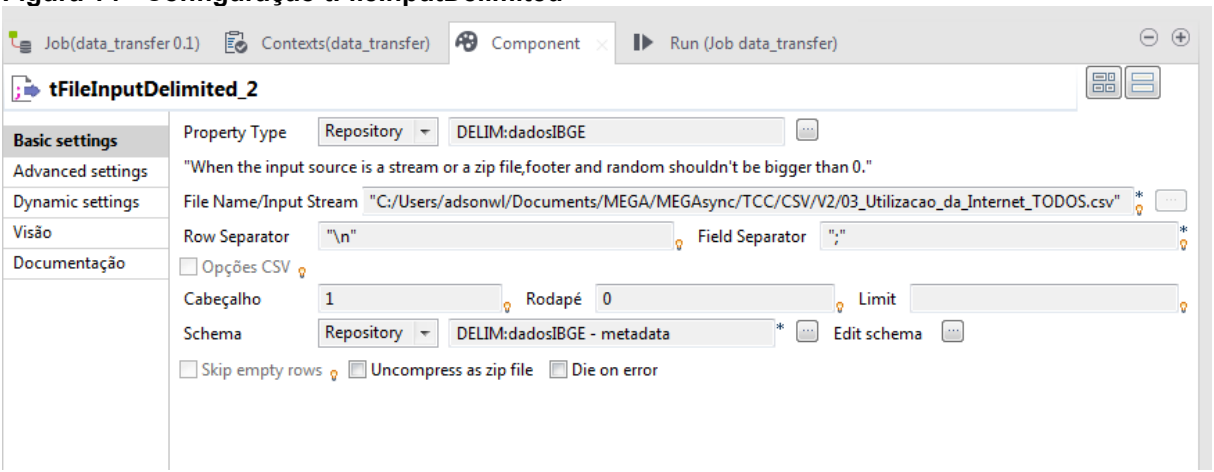


Fonte: Autoria própria (2017)

Após a configuração da conexão com o arquivo CSV, a próxima etapa foi criar os *jobs* que fazem o processo de ETL. No *job* o primeiro passo é criar um tFileInputDelimited, que é um componente de entrada de arquivos CSV para o *job*. A configuração do tFileInputDelimited, começa pela seleção do campo *property type* para *repository*, isso habilita um campo que permite selecionar o *link* com o arquivo fonte, isso trará todas as configurações do *link* para o componente.

Feito isto, falta configurar o *schema* do arquivo, no combo *schema* seleciona a opção *repository*, que habilitará novamente um campo permitindo a seleção da tabela gerada pelo *link* criado anteriormente. A configuração do componente é ilustrada na Figura 15.

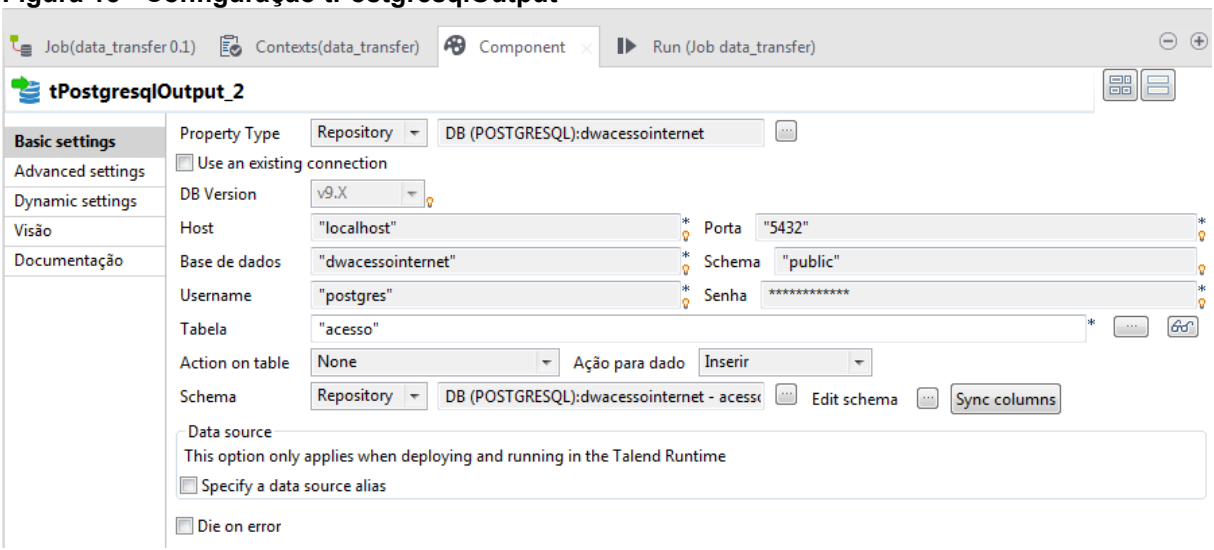
Figura 14 - Configuração tFileInputDelimited



Fonte: Autoria própria (2017)

Como os dados serão destinados a um DM criado no Postgres, que é o banco de dados relacional utilizado para armazenar os dados do DM, é necessário utilizar um componente chamado de tPostgresqlOutput, seguindo o mesmo processo do componente de entrada, a configuração da saída foi realizada como mostra a Figura 16.

Figura 15 - Configuração tPostgresqlOutput

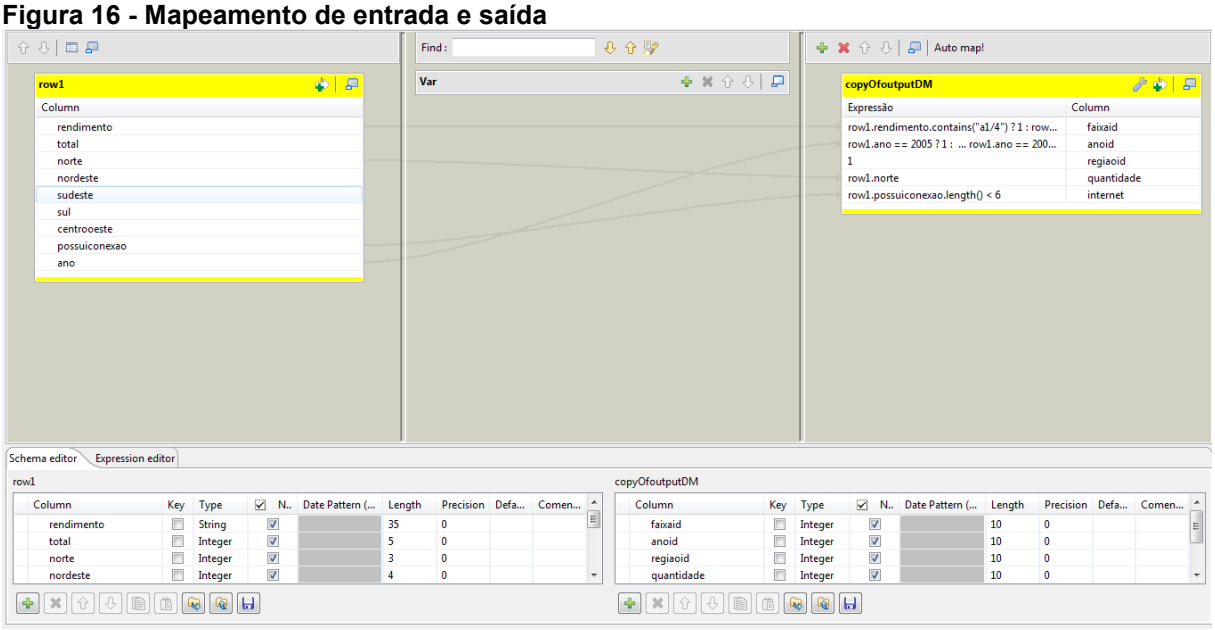


Fonte: Autoria própria (2017)

A última etapa para realizar o processo ETL é criar um componente que vai fazer o mapeamento entre a entrada e saída de dados, para isso foi utilizado o componente tMap, que permite filtrar, transformar, e mapear os dados vindos do arquivo de entrada, utilizando de expressões lógicas.

Na Figura 17, mostra o mapeamento dos dados, onde a tabela *row1* representa a entrada, e a tabela *copyOfoutputDM* representa a saída, as setas entre as duas tabelas mostra qual coluna da entrada está sendo mapeada para qual coluna da saída.

Figura 16 - Mapeamento de entrada e saída



Fonte: Autoria própria (2017)

Na tabela *copyOfOutputDM*, é possível visualizar a coluna Expressão, nesta coluna são realizadas as operações de transformação necessárias. Um exemplo de transformação, é o campo do DM internet, que é *boolean*; na tabela de entrada contém apenas o campo nomeado de possuiconexao, que é um tipo *string* contento sempre os valores “Havia” e “NãoHavia”, com isso foi feita a expressão do Quadro 2.

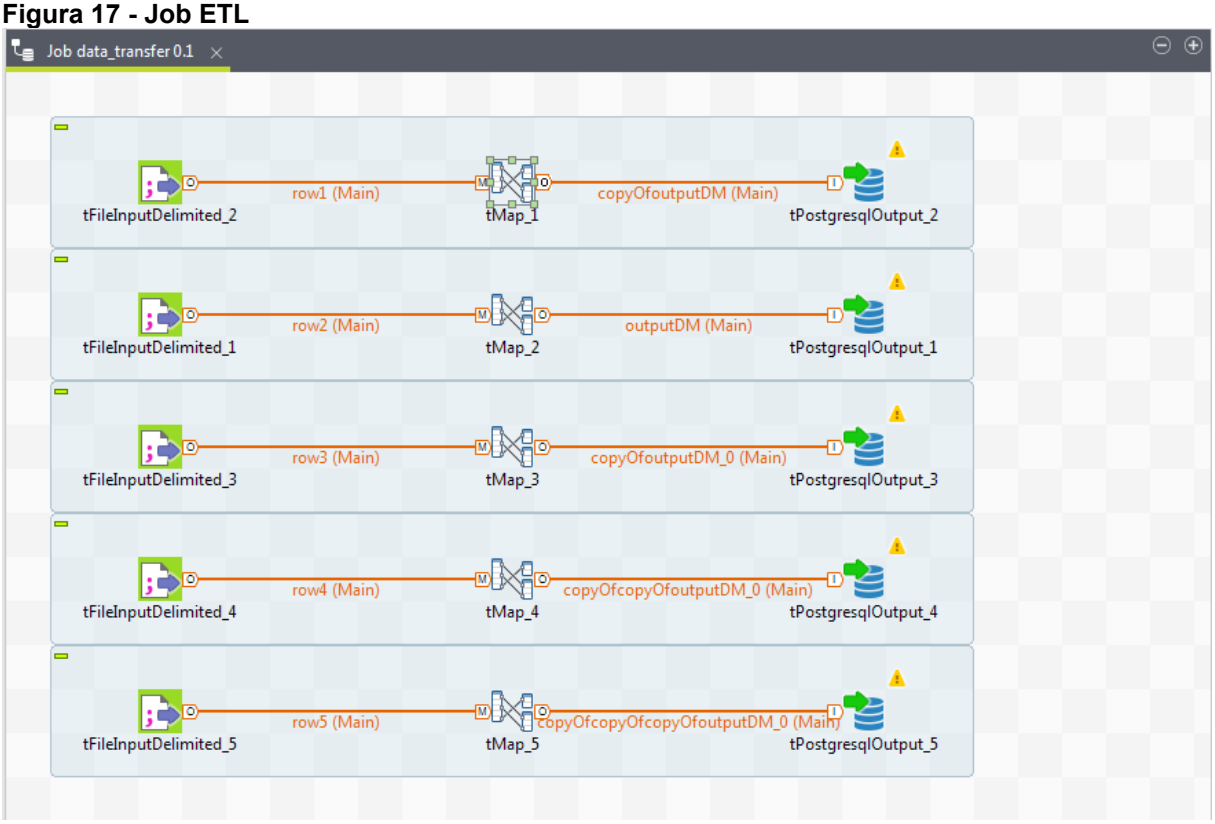
Quadro 2 - Expressão de conversão

row1.possuiconexao.length() < 6

Fonte: Autoria própria (2017)

Como cada linha da entrada possui os valores de cada uma das 5 regiões, e cada região se torna uma linha na tabela de saída, e o mapeamento permite mapear apenas uma linha de entrada para uma linha de saída por vez, foi necessário repetir este processo 5 vezes, uma para cada região, no final o *job* ficou como mostra a Figura 18.

Figura 17 - Job ETL

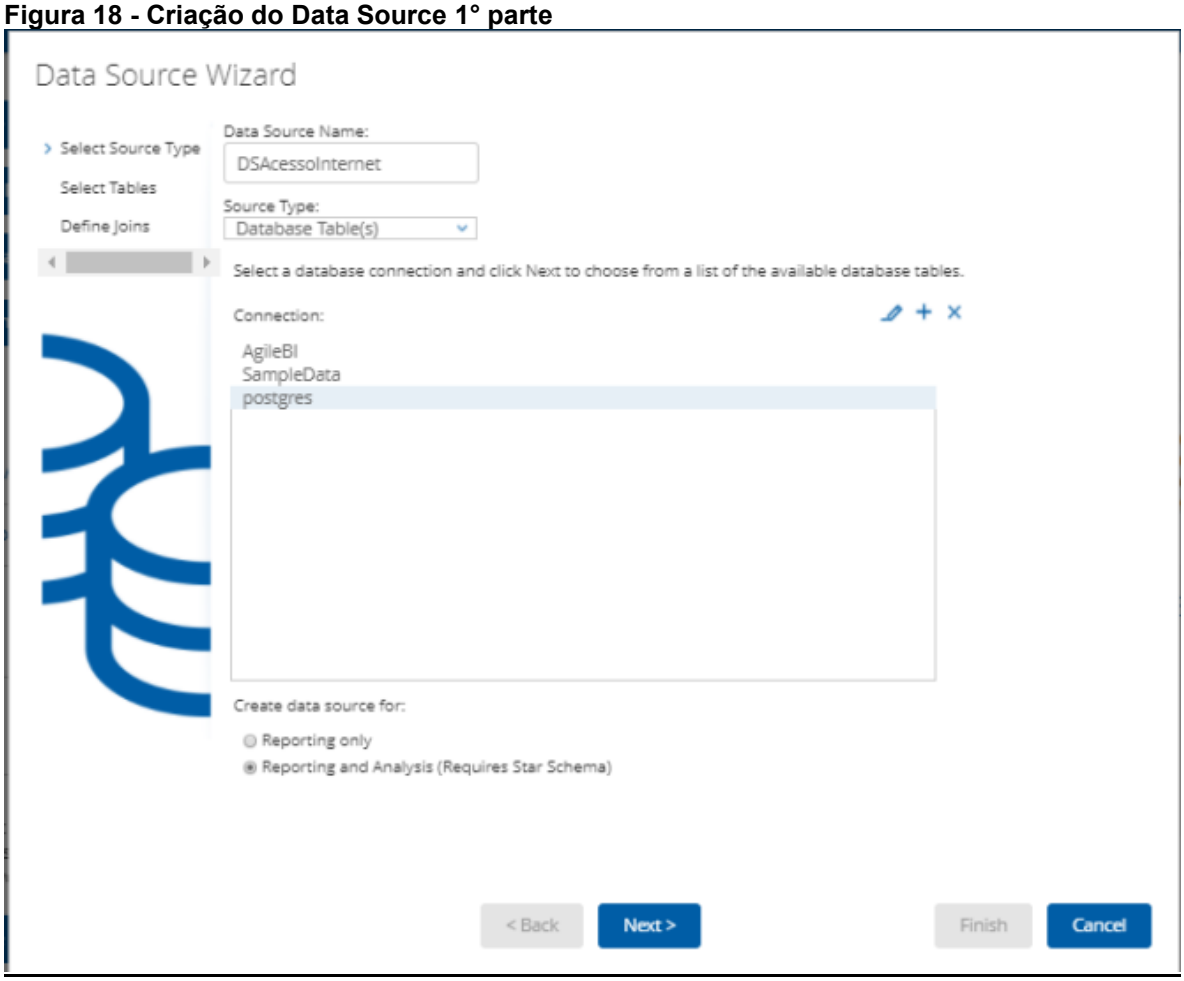


Fonte: Autoria própria (2017)

4.3 PENTAHO

Após a etapa de ETL, o primeiro passo é criar um DS (*Data Source*) na Ferramenta *Pentaho Analytics Plataform*. Uma vez que esteja autenticado como *Admin*, basta selecionar a opção *File* > *New* > *Data Source*, localizado no canto superior direito da tela principal. Isto abrirá um *Wizard* para a criação de tal, a primeira tela deste foi configurada como mostra a Figura 19.

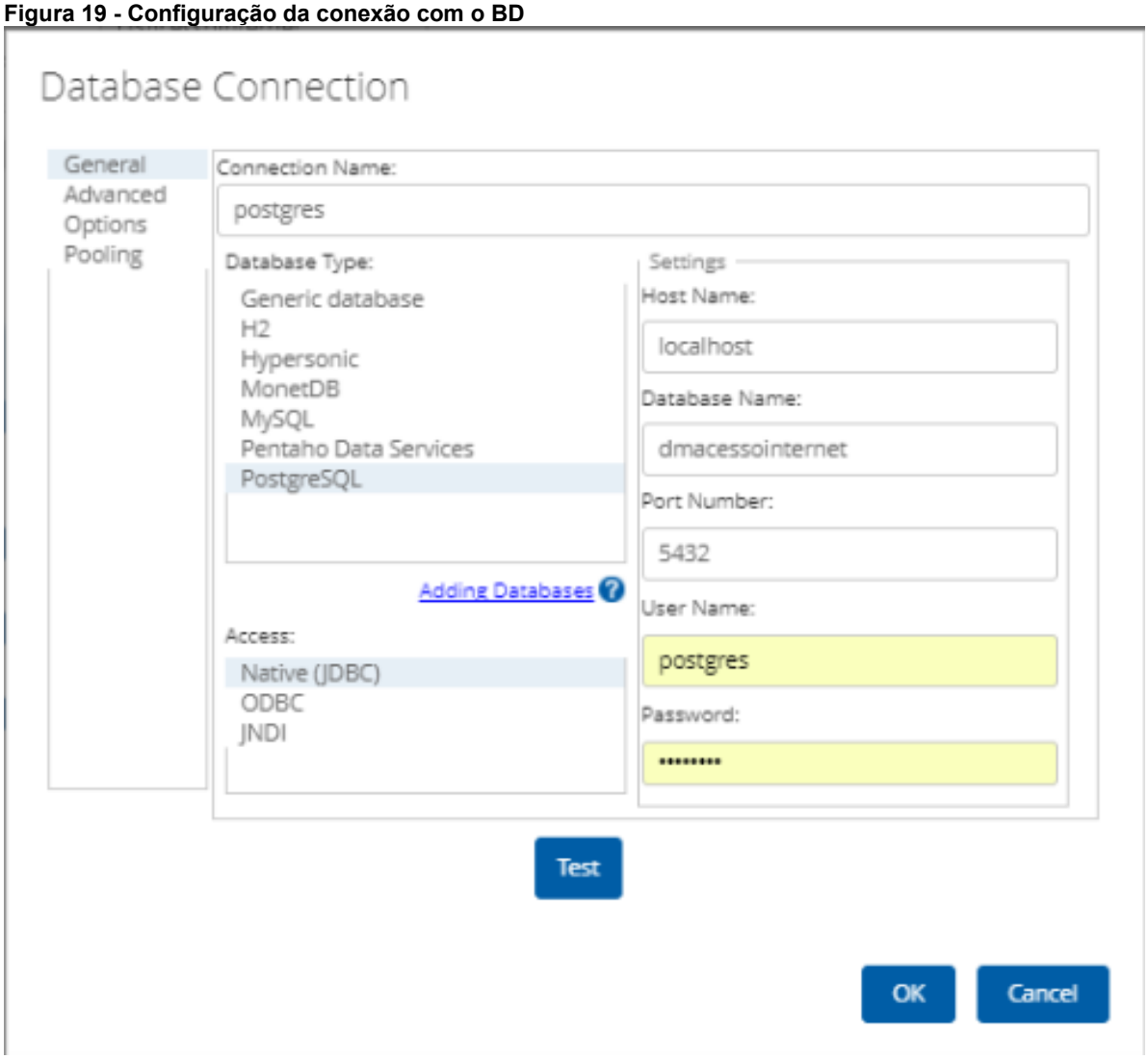
Figura 18 - Criação do Data Source 1° parte



Fonte: Autoria Própria (2017)

No campo *connection* é apresentada algumas conexões já existente. Nesta opção foi criada uma nova conexão apontando para o BD criado na etapa de ETL, descrito na seção 4.2. Para criar a nova conexão basta clicar no botão “+” localizado no canto superior direito do campo *Connection*. A conexão com o BD foi configurada como mostra a Figura 20.

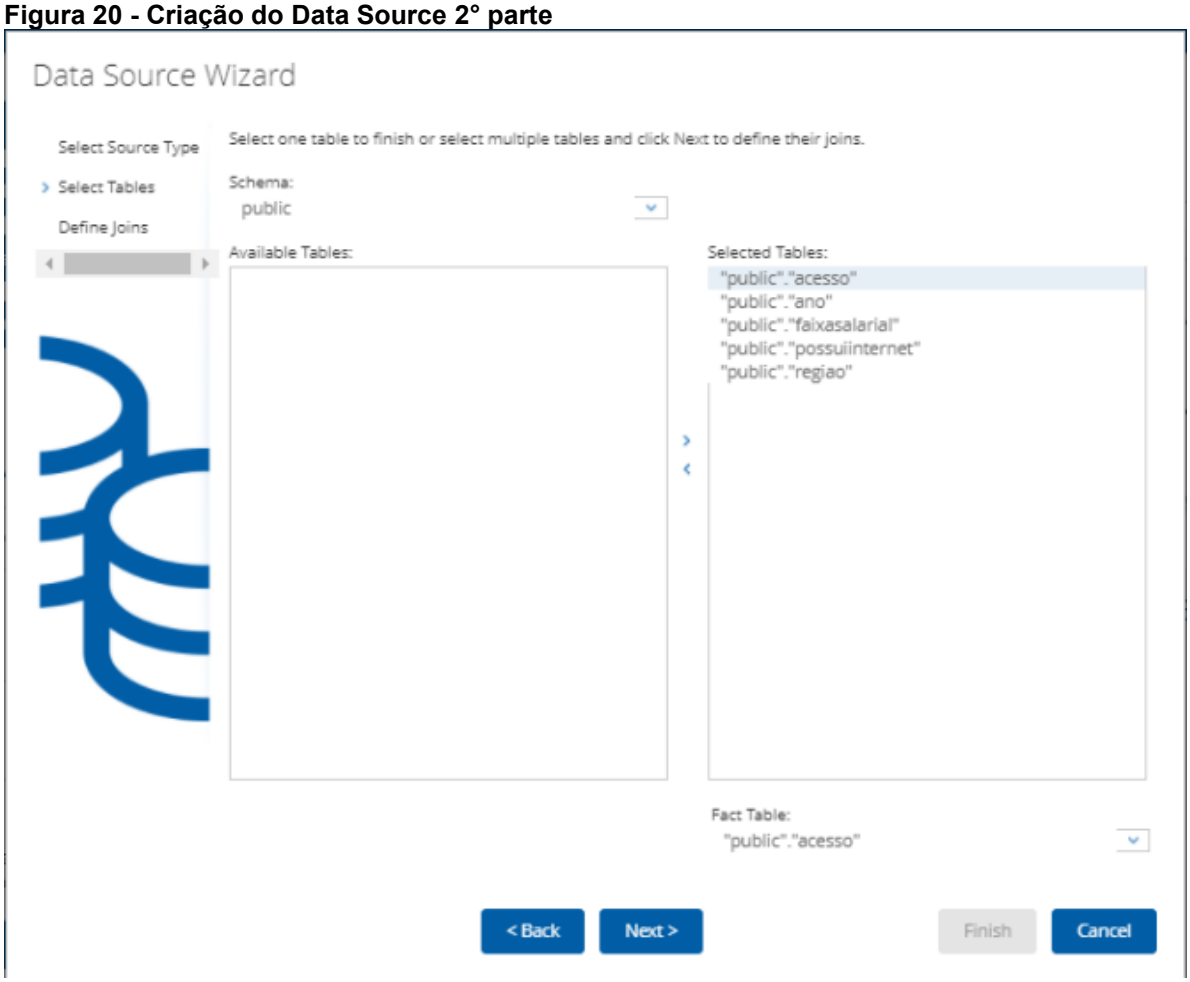
Figura 19 - Configuração da conexão com o BD



Fonte: Autoria própria (2017)

Configurada a conexão a passo seguintes é selecionar as tabelas que participariam do modelo, e a definição da tabela Fato, a seleção é visualizada na Figura 21.

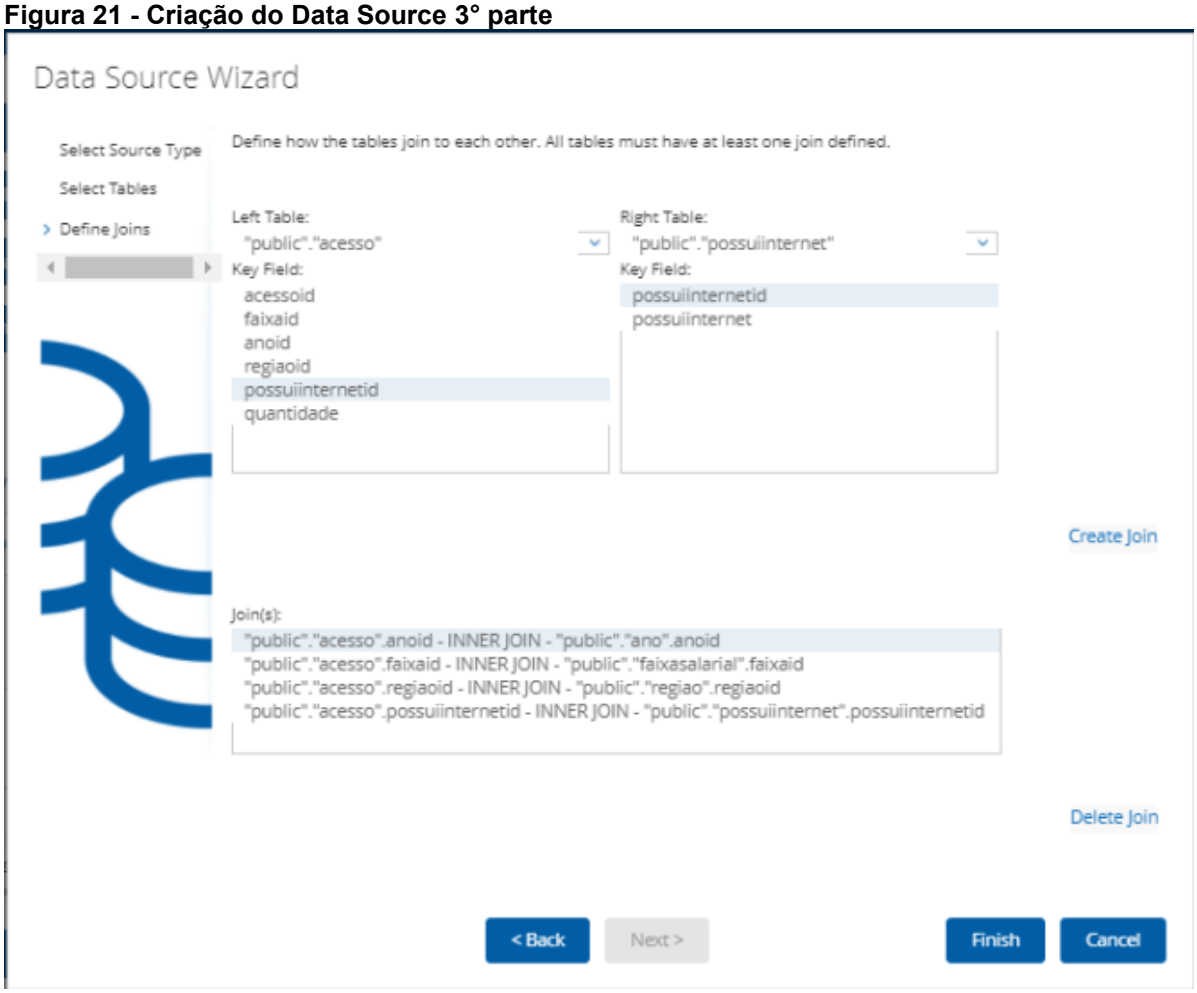
Figura 20 - Criação do Data Source 2° parte



Fonte: Autoria própria (2017)

A última etapa da criação do DS é definir os *Joins* ligando a tabela fato às tabelas dimensões, isto é feito na terceira página do *wizard.* Para cada *Join,* é selecionada a chave estrangeira da tabela fato no campo *Left Table,* e a tabela dimensão e sua chave primária no campo *Right Table*, a configuração final é exibida na Figura 22.

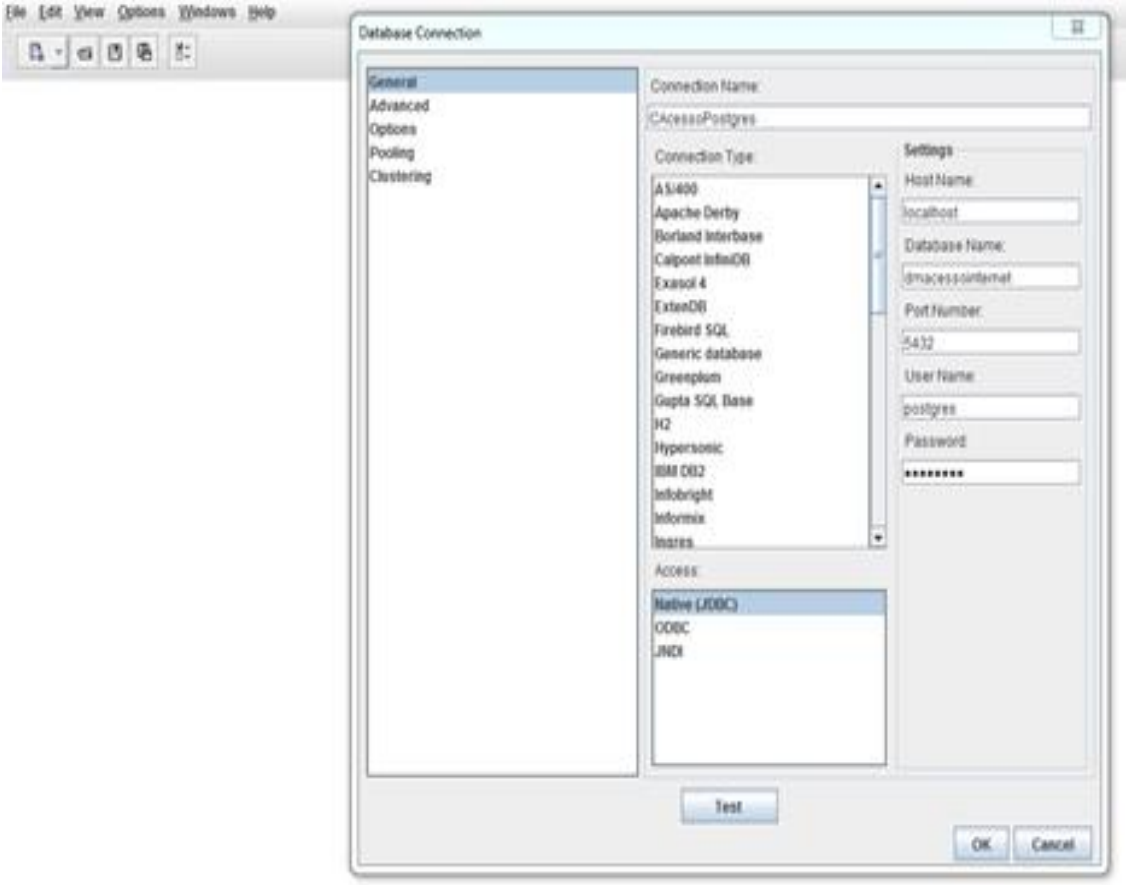
Figura 21 - Criação do Data Source 3° parte



Fonte: Autoria própria (2017)

Com isso o DS está concluído, agora é preciso criar o *Schema* de Cubo na ferramenta *Pentaho Schema Workbench*. O primeiro passo é criar uma conexão com o BD, que é realizado no menu *Options* > *Connection*. A configuração usada neste trabalho é visualizada na Figura 23.

Figura 22 - Criando conexão com o BD na ferramenta *Workbench*

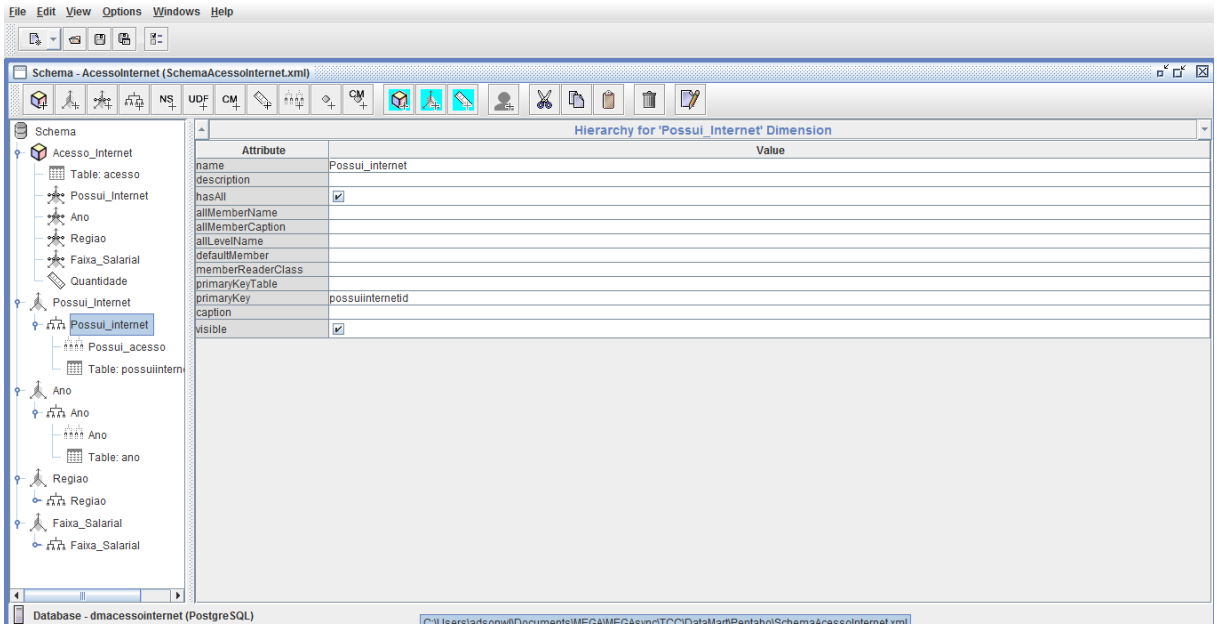


Fonte: Autoria própria (2017)

Com a conexão com o BD estabelecida é possível iniciar a criação do *Schema.* Para criar um novo *Schema* basta ir no menu *File* > *New* > *Schema.* Inicia-se a configuração criando um “cubo” que é a representação da tabela Fato, neste cubo é incluída as *Measures,* que são os atributos da tabela, e também o relacionamento com as dimensões.

Além do cubo, as dimensões também são configuradas, em cada dimensão é informada a tabela que fará parte da dimensão. O *Schema* final pode ser visualizado na Figura 24.

Figura 23 - Schema criado na ferramenta *Workbench*



Fonte: Autoria própria (2017)

Ao criar o *Schema* é gerado um arquivo XML, contendo as informações usadas em sua configuração. O conteúdo do arquivo gerado pode ser visualizado no Quadro 3.

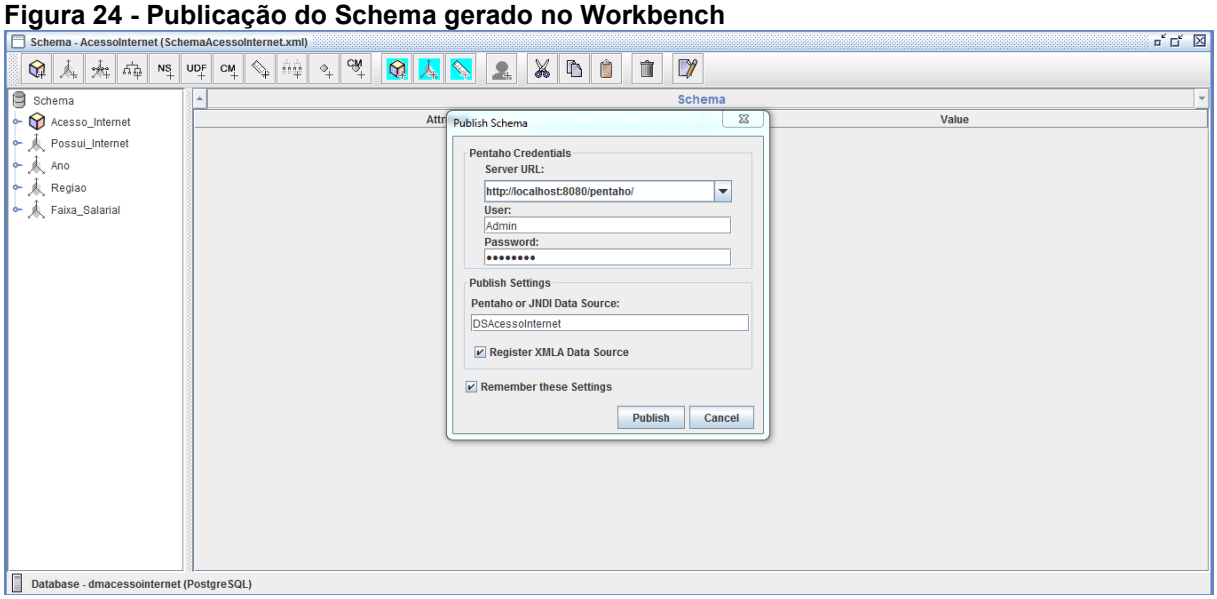
Quadro 3 - Arquivo XML do Cubo gerado no Workbench

```
<Schema name="AcessoInternet">
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Possui_Internet"
description="Possui acesso a internet">
    <Hierarchy name="Possui_internet" visible="true" hasAll="true" primaryKey="possuiinternetid">
      <Table name="possuiinternet" schema="public">
      </Table>
      <Level name="Possui_acesso" visible="true" column="possuiinternet" type="Boolean"
uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" visible="false" highCardinality="false" name="Ano"
description="Ano">
    <Hierarchy name="Ano" visible="true" hasAll="true" primaryKey="anoid">
      <Table name="ano" schema="public">
      </Table>
      <Level name="Ano" visible="true" table="ano" column="ano" type="Integer" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Regiao"
description="Regi&amp;#227;o do acesso">
    <Hierarchy name="Regiao" visible="true" hasAll="true">
      <Table name="regiao" schema="public">
      </Table>
      <Level name="Regiao" visible="true" column="regiao" type="String" uniqueMembers="true"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Faixa_Salarial">
    <Hierarchy name="Faixa_Salarial" visible="true" hasAll="true" primaryKey="faixasalarial">
      <Table name="faixasalarial" schema="public">
      </Table>
      <Level name="Faixa_Salarial" visible="true" column="faixasalarial" type="String" uniqueMembers="true"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Cube name="Acesso_Internet" caption="Acesso Internet" visible="true" description="Acesso Internet"
cache="true" enabled="true">
    <Table name="acesso" schema="public">
    </Table>
    <DimensionUsage source="Possui_Internet" name="Possui_Internet" visible="true"
foreignKey="possuiinternetid" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Ano" name="Ano" visible="true" foreignKey="anoid" highCardinality="false">
    </DimensionUsage>
<DimensionUsage source="Regiao" name="Regiao" visible="true" foreignKey="regiaoid"
highCardinality="false">
    </DimensionUsage>
<DimensionUsage source="Faixa_Salarial" name="Faixa_Salarial" visible="true"
foreignKey="faixaid" highCardinality="false">
</DimensionUsage>
    <Measure name="Quantidade" column="quantidade" datatype="Integer" aggregator="sum"
description="Quantidade de acesso a internet" visible="true">
    </Measure>
  </Cube>
</Schema>
```

Fonte: Autoria própria (2017)

Para poder utilizar esse *Schema* gerado no *Pentaho Analytics Plataform* para criar relatórios, é necessário publica-lo selecionando a opção no menu *File* > *Publish.* A ferramenta solicita que seja informado o endereço do *server Pentaho*, assim como *login*, senha e qual o DS de destino. Estas informações foram dadas como mostra a Figura 25.

Figura 24 - Publicação do Schema gerado no Workbench



Fonte: Autoria própria (2017)

Uma vez publicado o *Schema*, é possível criar os relatórios. Para fazê-lo basta criar um *JPivot View* no menu *File* > *New* > *JPivot View.* A ferramenta requisitará o *Schema* e *DS* a ser usado, informando o *Schema* e DS criados nas etapas anteriores. Com estes passos será possível visualizar os dados do DM, ao clicar no ícone de gráfico acima da tabela que aparece na Figura 26 mostrando os dados, aparecerá um gráfico de barras.

Para manipular as informações exibidas no gráfico, basta expandir ou recolher as colunas da tabela. A Figura 26 mostra a expansão da tabela, visando mostrar a quantidade de casas possuíam acesso por cada região e quantas não.

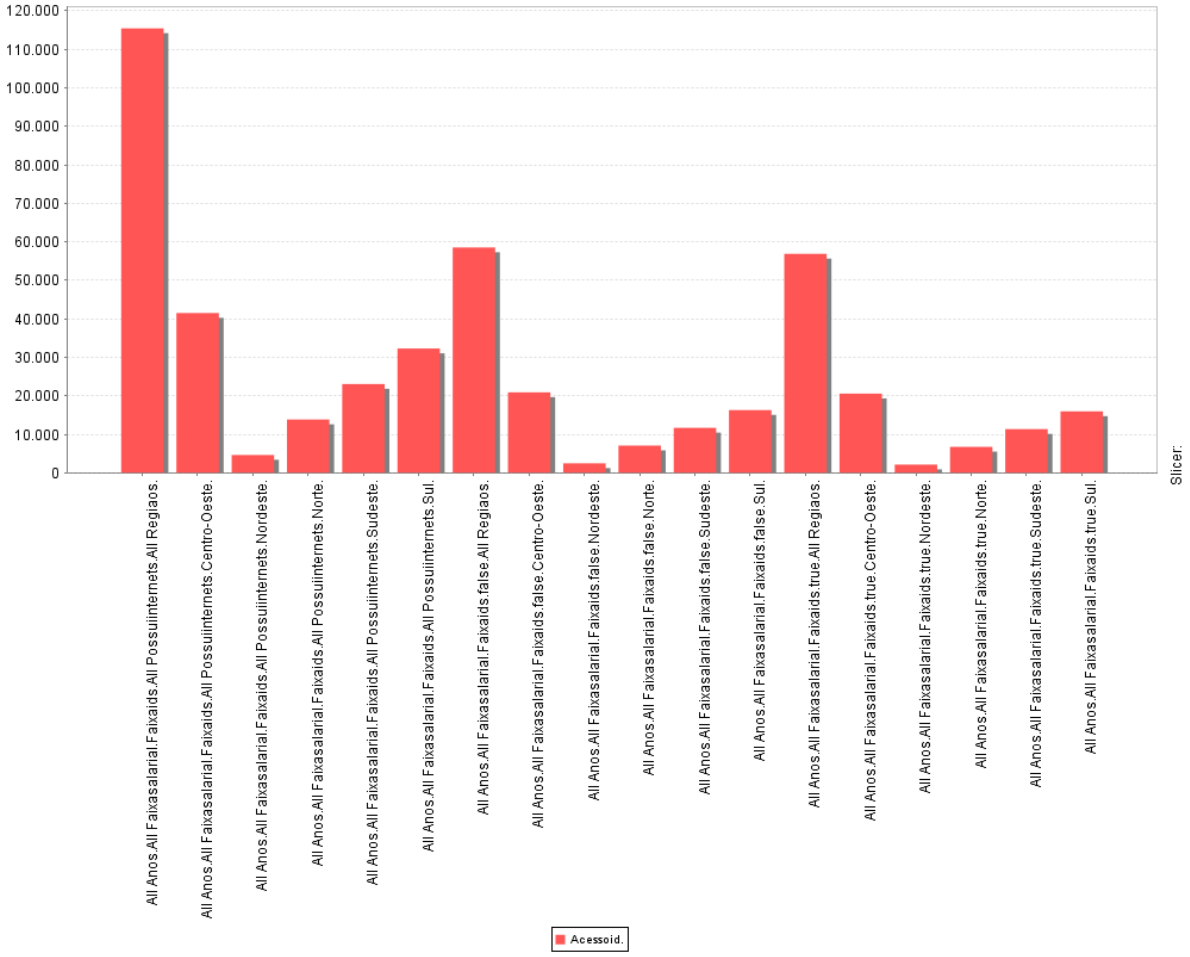
Figura 25 - Tabela de dados detalhando por região

Ano	Faixaid	Possuiinternet	Regiao	Measures Acessoid
All Anos	All Faixasalarial.Faixaids	All Possuiinternets	All Regiaos	115.440
			Centro-Oeste	41.520
			Nordeste	4.656
			Norte	13.872
			Sudeste	23.088
			Sul	32.304
		false	All Regiaos	58.545
			Centro-Oeste	20.925
			Nordeste	2.493
			Norte	7.101
			Sudeste	11.709
			Sul	16.317
		true	All Regiaos	56.895
			Centro-Oeste	20.595
			Nordeste	2.163
			Norte	6.771
			Sudeste	11.379
			Sul	15.987

Fonte: Autoria própria (2017)

Apesar dos controles para mostrar e esconder as colunas, não foi possível esconder os totalizadores das colunas, por exemplo o totalizador “All Regiaos”, isso acaba poluindo o gráfico com informações que podem não agregar conhecimento ao estudo e dificultando a visão das informações que se deseja obter, a exemplo a Figura 26, gerada a partir da tabela da Figura 25.

Figura 26 - Gráfico gerado com a ferramenta Pentaho



Fonte: Autoria própria (2017)

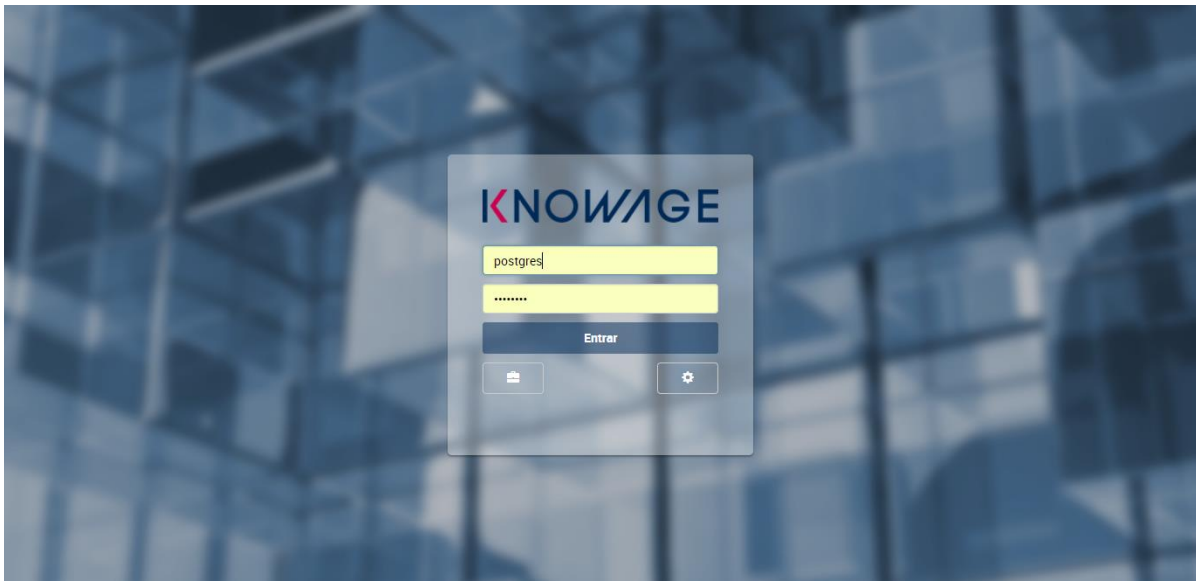
Na parte direita do gráfico estão separados por regiões, a quantidade de residências que possuíam acesso à internet na data da pesquisa, a esquerda destes, estão as residências que não possuíam acesso à internet. Como pode ser observado, o gráfico fica poluído pelos acumuladores de medidas, como exemplo a coluna de quantas residências possuem acesso em todas as regiões.

4.4 KNOWAGE

A ferramenta Knowage, diferente do Pentaho, necessita que seja feito previamente a sua instalação, e como pré-requisito a ferramenta exige que possua instalado uma versão do MySQL versão 5.5 ou superior, JAVA (JRE ou JDK) versão 1.7 ou 1.8, e TOMCAT versão 7 ou superior. Uma vez que a instalação é um processo simples e padrão, não será abordado nesse trabalho como fazê-lo, bastando garantir que os pré-requisitos citados aqui sejam atendidos.

Uma vez que a ferramenta esteja instalada, aparecerá no menu um ícone do Knowage e com a descrição “Start Knowage”, que será responsável por iniciar o servidor Knowage. Após iniciada a ferramenta, entrar no link “http://localhost:8080/knowage/”, onde deverá aparecer a tela de login, como mostra a Figura 28, clicando no ícone de engrenagem, abaixo do botão “Entrar”, fará o login como Administrador.

Figura 27 - Tela de *login* Knowage

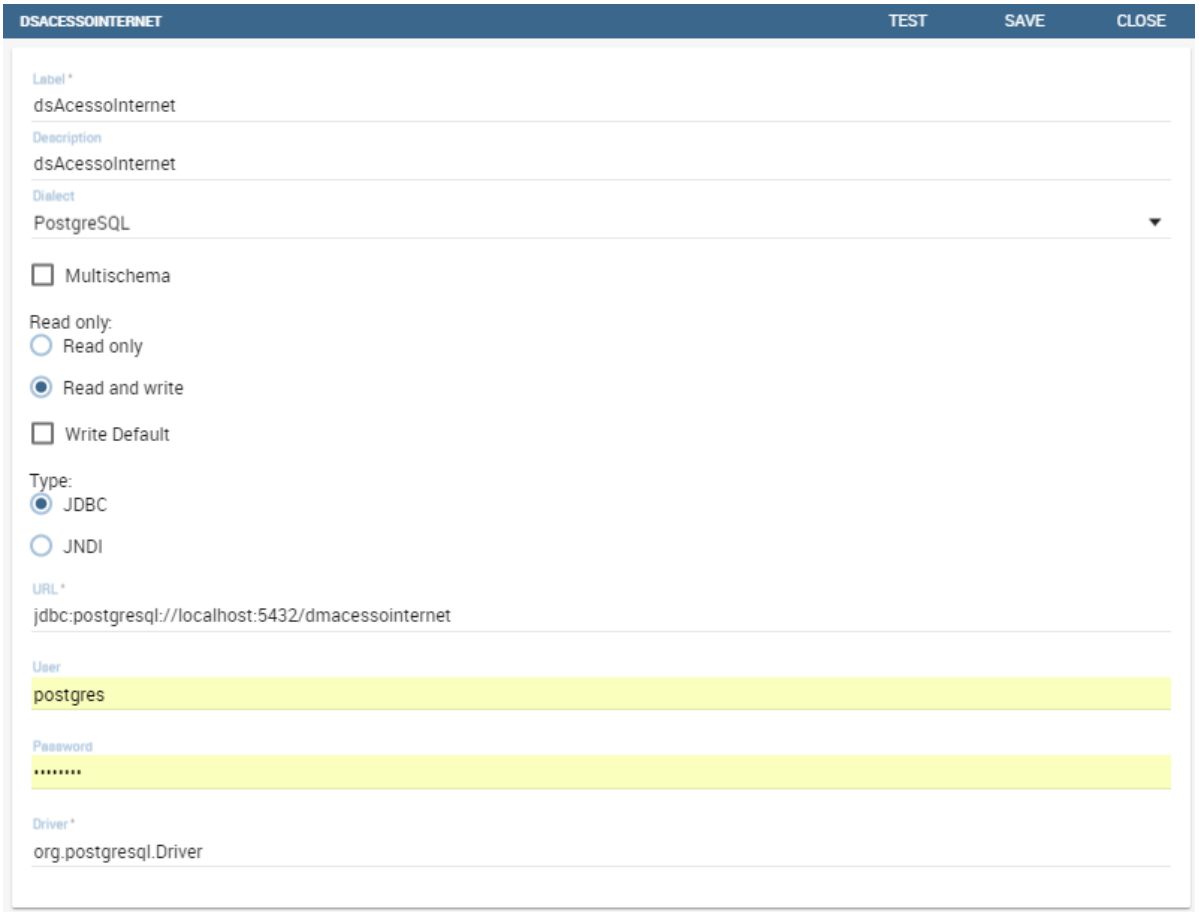


Fonte: Autoria própria (2017)

O primeiro passo é criar um DS, para isto clica no ícone de menu, apontado pelo texto em vermelho no canto superior da tela, o menu está organizado em várias seções, para criar o DS, deve ir até a seção “*Data Providers*” e clicar em “*Data Source*”, isto abrirá uma tela mostrando alguns DS de exemplos, que já vem

configurados com a ferramenta, para criar um novo DS, basta clicar no botão vermelho com o símbolo “+” no meio, a tela de criação foi configurada como mostra a figura 29.

Figura 28 - Criação do DS na ferramenta *Knowage*



Fonte: Autoria própria (2017)

Para verificar se a configuração está correta, deve-se clicar no botão “Test” localizado no canto superior direito da tela, se estiver correto, será exibido uma mensagem informando o sucesso no teste.

O próximo passo é criar um modelo OLAP, porém antes de cria-lo no servidor, antes é necessário criar um arquivo XML, com o *schema* do DM com seu cubo e dimensões. Esse arquivo é feito utilizando uma estrutura de marcação em bocós, semelhante ao HTML. Abaixo segue as TAGs e os seus campos que foram utilizado neste trabalho em particular, não é apresentado outras TAGs ou campos disponíveis para a criação de um *schema*:

- *Schema*: É a tag principal, pois toda a configuração deve estar delimitada por ela, o único campo que é informado nela é o nome do *schema*.

- *Dimension*: Usada para definir as dimensões, deve ser preenchido os campos *type*, *visible*, *highCardinality*, *name* e *description*.
- *Hierarchy*: Define a hierarquia dentro da dimensão, deve ser definido os campos *name*, *visible*, *hasAll* e *primaryKey*.
- *Table*: Define qual a tabela usada pela dimensão, deve ser definido os campos *name* e *schema*(schema do BD que será a fonte dos dados).
- *Level*: Define os níveis da hierarquia, deve ser definido os campos *name*, *visible*, *column*, *type*, *uniqueMembers*, *levelType* e *hideMemberIf*.
- Cube: Define o cubo do schema, deve ser definido os campos *name*, *caption*, *visible*, *description*, cache e *enabled*.
- *DimensionUsage*: define as relações entre o cubo e as dimensões, deve ser definido os campos *source*, *name*, *visible*, *foreignKey* e *highCardinality*.

Considerando as Tags acima, foi desenvolvido o arquivo XML com o conteúdo apresentado no Quadro 4.

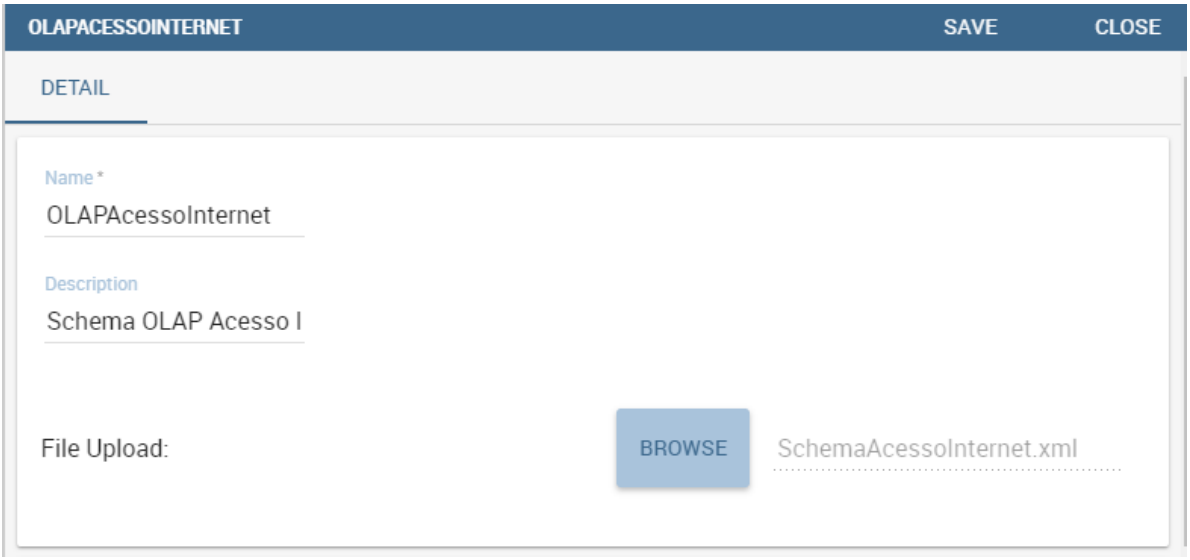
Quadro 4 - Schema desenvolvido para a ferramenta Knowage

```
<Schema name="AcessoInternet">
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Possui_Internet"
description="Possui acesso a internet">
    <Hierarchy name="Possui_internet" visible="true" hasAll="true" primaryKey="possuiinternetid">
      <Table name="possuiinternet" schema="public">
      </Table>
      <Level name="Possui_acesso" visible="true" column="possuiinternet" type="Boolean"
uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" visible="false" highCardinality="false" name="Ano"
description="Ano">
    <Hierarchy name="Ano" visible="true" hasAll="true" primaryKey="anoid">
      <Table name="ano" schema="public">
      </Table>
      <Level name="Ano" visible="true" table="ano" column="ano" type="Integer" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Regiao"
description="Regi&amp;#227;o do acesso">
    <Hierarchy name="Regiao" visible="true" hasAll="true" primaryKey="regiaoid">
      <Table name="regiao" schema="public">
      </Table>
      <Level name="Regiao" visible="true" column="regiao" type="String" uniqueMembers="true"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Faixa_Salarial">
    <Hierarchy name="Faixa_Salarial" visible="true" hasAll="true" primaryKey="faixaid">
      <Table name="faixasalarial" schema="public">
      </Table>
      <Level name="Faixa_Salarial" visible="true" column="faixasalarial" type="String" uniqueMembers="true"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Cube name="Acesso_Internet" caption="Acesso Internet" visible="true" description="Acesso Internet"
cache="true" enabled="true">
    <Table name="acesso" schema="public">
    </Table>
    <DimensionUsage source="Possui_Internet" name="Possui_Internet" visible="true"
foreignKey="possuiinternetid" highCardinality="false">
</DimensionUsage>
    <DimensionUsage source="Ano" name="Ano" visible="true" foreignKey="anoid" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Regiao" name="Regiao" visible="true" foreignKey="regiaoid"
highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Faixa_Salarial" name="Faixa_Salarial" visible="true" foreignKey="faixaid"
highCardinality="false">
    </DimensionUsage>
    <Measure name="Quantidade" column="quantidade" datatype="Integer" aggregator="sum"
description="Quantidade de acesso a internet" visible="true">
    </Measure>
  </Cube>
</Schema>
```

Fonte: Autoria própria (2017)

Com o arquivo pronto, é possível criar o modelo OLAP no servidor. De volta a ferramenta Knowage, abre-se o menu e na seção "*Catalogs*" clica-se no item "*Modrian schemas catalog*", ao fazê-lo será exibido uma tela com alguns *schemas* de exemplo, para criar um novo, clica-se no botão vermelho com o símbolo "+" no meio, na tela de criação deverá ser preenchido o campo "*Name*" e selecionar o arquivo XML contendo o schema OLAP, e opcionalmente pode ser preenchido o campo "*Description*", a Figura 30 mostra a configuração utilizada nesse trabalho.

Figura 29 - Criação do schema OLAP na ferramenta Knowage



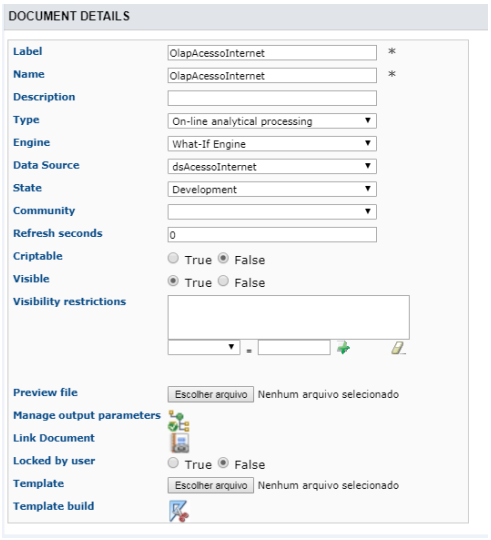
Fonte: Autoria própria (2017)

Depois de salvar o schema OLAP, o próximo passo é abrir o "*Document Browser*", para isso deve-se abrir o menu e clicar no ícone de pasta, no canto superior esquerdo da tela, logo abaixo da identificação do usuário logado, isto fará abrir uma tela com exibindo uma estrutura de pastas, onde conterá a pasta "*Multidimensional Analysis*" clicando neste item, será mostrado um exemplo de documento usando o modo OLAP.

Para criar um novo documento, clica-se no botão vermelho com um símbolo "+" no meio, abrirá duas opções, para o estudo de caso deste trabalho foi selecionado a opção "*Generic document*". Na tela de criação deve ser informado os campos *Type*, *Engine*, *Data Source* além de *Label* e *Name,* após informado esses campos, clicar no ícone de disquete no canto superior da tela. Após salvar, ainda na mesma tela

aparecerá algumas opções, para o próximo passo, é clicado no ícone a direita do texto "*Template Build*". A Figura 31 mostra esta tela de criação após salvar as configurações.

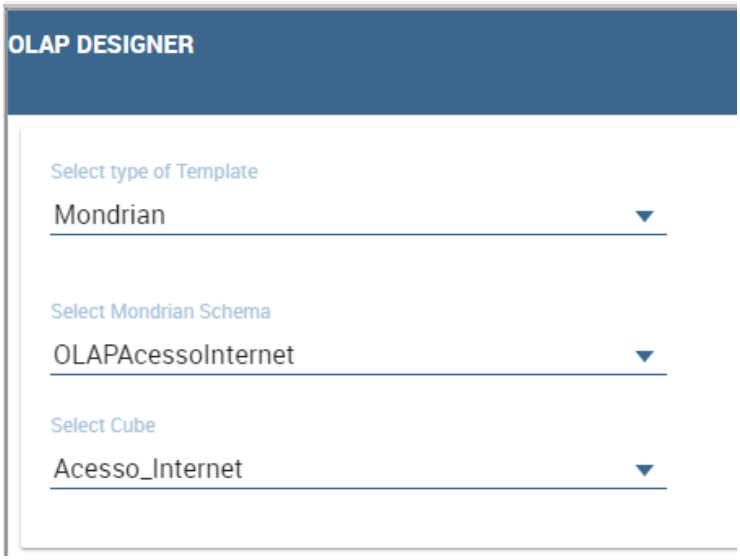
Figura 30 - Criação do documento na ferramenta Knowage



Fonte: Autoria própria (2017)

Dentro do "*Template Build*", a primeira tela exibirá um campo para selecionar o tipo de *template* que será criado, seleciona-se a opção "*Mondrian*", depois seleciona o *schema mondrian* criado na etapa anterior, e por fim o cubo que foi descrito no arquivo XML, a Figura 32 exibe tais campos preenchidos.

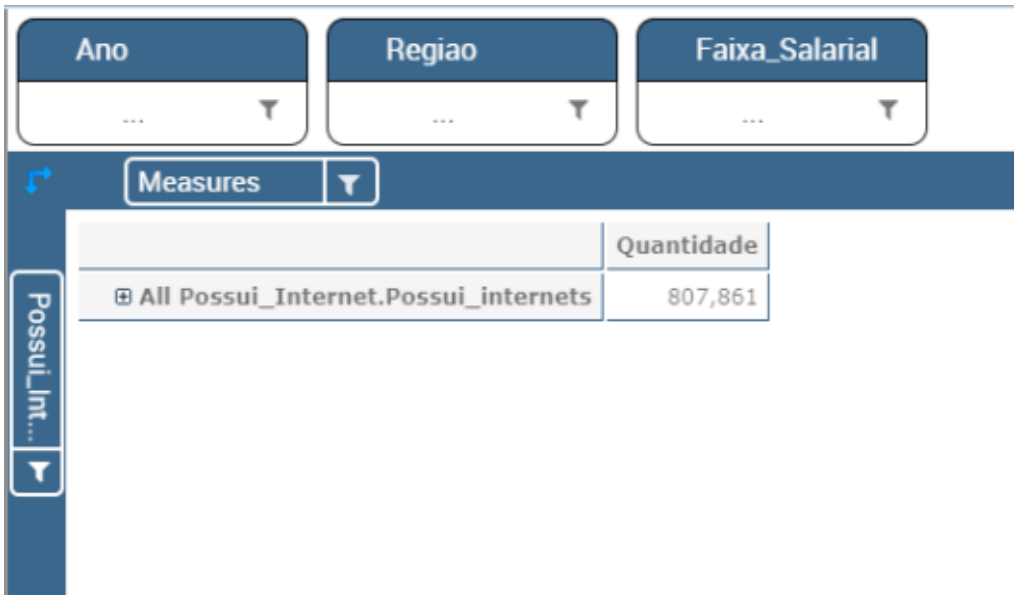
Figura 31 - Olap Desginer na ferramenta Knowage



Fonte: Autoria própria (2017)

Dado estas configurações, clica-se no botão “*START*” localizado no canto superior direito. Será exibido uma tela com uma tabela exibindo uma medida e uma dimensão, as outras dimensões aparecem no topo da tela, para incluí-las basta escolher uma e arrastar até onde está a dimensão que já faz parte da tabela, a Figura 33 exibe a tela inicial gerada neste trabalho.

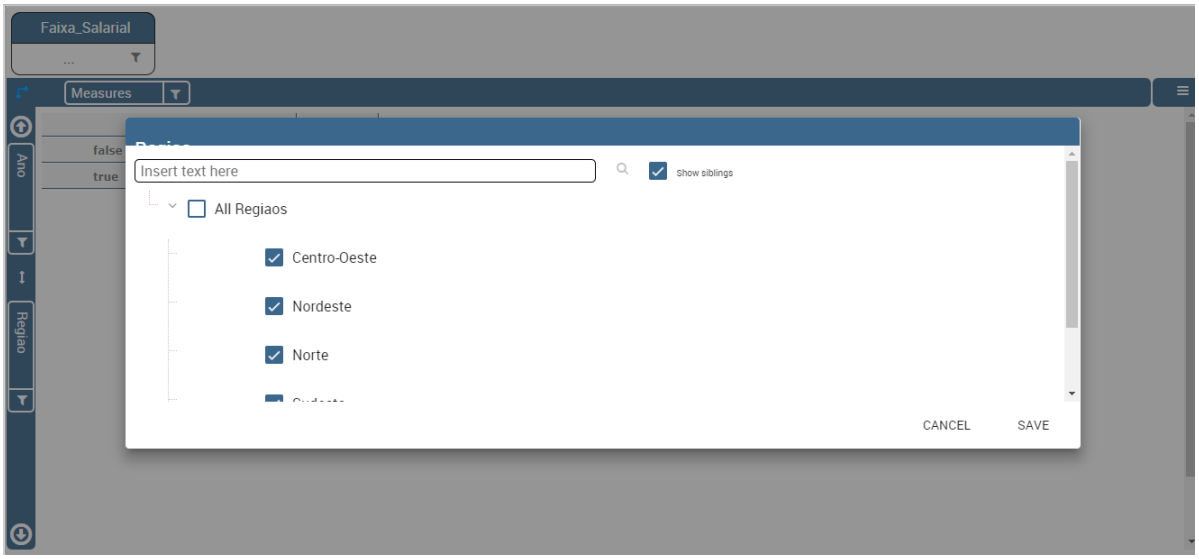
Figura 32 - Tela inicial do relatório



Fonte: Autoria própria (2017)

Clicando no ícone de Filtro no canto das dimensões, é possível filtrar as informações que desejadas daquela dimensão, a Figura 34 exibe o filtro aplicado a dimensão “Região”.

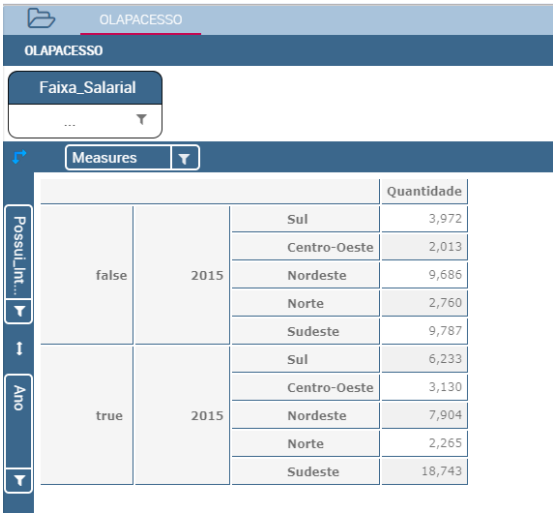
Figura 33 - Filtro da dimensão Região



Fonte: Autoria própria (2017)

Para exemplificar as configurações possíveis, foi filtrado os acessos apenas do ano 2015, detalhando por região e se possuía acesso à internet ou não, a Figura 35 mostra como ficou a tabela.

Figura 34 - Tabela com os filtros aplicados



			Quantidade
false	2015	Sul	3,972
		Centro-Oeste	2,013
		Nordeste	9,686
		Norte	2,760
		Sudeste	9,787
true	2015	Sul	6,233
		Centro-Oeste	3,130
		Nordeste	7,904
		Norte	2,265
		Sudeste	18,743

Fonte: Autoria própria (2017)

Ao selecionar uma medida para aparecer na tabela, basta desmarcar a caixa do acumulador, a exemplo na Figura 33, onde foi desmarcado a caixa do acumulador “All Regiaos”, isso facilita a visualização dos dados, permitindo mostrar apenas as informações que são relevantes ao estudo.

4.5 CONSIDERAÇÕES DO CAPÍTULO

Na Seção 4.5.1 é apresentado as considerações do capítulo quanto a utilização das ferramentas de *Data Warehouse*. Por fim na Seção 4.5.2 é apresentado as considerações sobre os resultados obtidos nos relatórios.

4.5.1 Ferramentas

A ferramenta *Talend Open Studio*, utilizada no processo de ETL, mostrou-se bastante eficaz, permitindo realizar todo o processo de ETL de forma rápida e intuitiva, provendo mensagens claras e objetivas, facilitando a localização e correção de eventuais erros.

A ferramenta *Pentaho* mostrou grande facilidade ao desenvolver o Data Mart, uma vez que foi possível realizar todo o processo de forma gráfica, não necessitando assim conhecimento de programação.

Uma dificuldade encontrada na ferramenta *Pentaho* foi nos filtros do JPivot *View*, pois não possibilitou a escolha de mostrar ou não os acumuladores de medidas, poluindo as informações nos casos em que estes dados não são desejados.

A ferramenta *Knowage*, mostrou grande versatilidade na hora de exibir os dados, apresentando de forma simples a seleção e filtros dos dados, porém é necessário um conhecimento prévio de programação para criar o *schema* OLAP.

4.5.2 Acesso à internet

Nos dados analisados foi possível perceber que o problema do acesso à internet é maior nas regiões Norte e Nordeste, sendo as únicas onde a quantidade de residências que não possuíam acesso à internet superou aa que possuíam, sendo respectivamente 2.760 e 9.686 as residências que possuíam acesso em 2015 e 2.265 e 7.904 as que possuíam.

As regiões Sul e Sudeste obtiveram as melhores medias na pesquisa, com as residências que possuíam acesso alcançando em torno de dois terços do total de residências da região, sendo respectivamente 3.972 e 9.787 as residências sem acesso à internet em 2015, e 6.233 e 18.743 as residências que possuíam.

A Região Centro-Oeste ficou entre as duas situações, sendo 2.013 as residências sem acesso à internet e 3,130 as com acesso à internet na data da pesquisa realizada pelo IBGE.

5 CONSIDERAÇÕES FINAIS

Na Seção 5.1 é apresentado a conclusão do trabalho, e por fim na Seção 5.2 é mostrado os trabalhos futuros.

5.1 CONCLUSÃO

Como formas de construção de um DW, foram identificadas diversas estruturas, assim como modelos de implementa-las. Algumas das arquiteturas estudadas como a estrela e floco de neve são mais simples de implementar em relação a arquitetura constelação, uma vez que consistem de apenas uma tabela fato. Obviamente apenas isto não é o suficiente para escolher qual arquitetura usar, precisando fazer um estudo da necessidade da empresa, e do objetivo que o gestor ao adotar uma estratégia como um DW.

Algumas das estratégias para criar o DW seria o *top down* e *bottom up,* cujo o maior impacto no desenvolvimento seria que a segunda traz um retorno mais rápido para a empresa, porém não dá a garantia de um DW consistente. A primeira estratégia da a garantia de um DW consistente, porém por exigir que todos os DMs estejam prontos para iniciar o seu uso, demora mais tempo para que o gestor possa usufruir do produto.

A ferramenta *Pentaho*, mostrou-se bastante prática, permitindo toda a configuração e uso, desde a criação do modelo dimensional até a geração dos relatórios apenas com ferramentas visuais, o ponto fraco seria a usabilidade do *JPivot View*, mostrando alguns controles em botões muito pequenos, dificultando seu uso, o uso não foi impedido por esse motivo, apenas dificultado.

A ferramenta *Knowage*, mostrou uma grande preocupação com a usabilidade do usuário ao criar os relatórios, apresentando controles intuitivos e fáceis de utilizar, o ponto fraco da ferramenta foi por não prover uma maneira visual de criar o modelo dimensional, exigindo um conhecimento prévio de programação para que seja possível cria-lo.

Utilizando as duas ferramentas foi possível extrair informações do DM desenvolvido, ambas proveram maneiras rápidas de alterar quais dados são mostrados, ordem e agrupamento, possibilitando ao usuário visualizar as informações de diferentes perspectivas rapidamente.

A aplicação de ferramentas diferentes em um mesmo DM, possibilitou ver as características de cada uma, mostrando onde se apresentam seus pontos fortes. Em ambos foi possível retirar e manipular a informação desejada, sendo assim, o gestor não teria uma perda de recursos optando por qualquer uma das duas, e a caraterística de ambas serem gratuitas, no caso a versão gratuita das duas ferramentas, permite que sua adoção seja feita indiferente do planejamento financeiro para a construção do DW que a empresa tenha adotado.

5.2 TRABALHOS FUTUROS

Como trabalhos futuros, pode ser aplicado as versões pagas das ferramentas *Pentaho* e *Knowage*, a fim de mostrar as diferenças e recursos a mais elas oferecem. A aplicação dessas ferramentas em um estudo de caso que requeira um DW com diversos DMs, para ver como se comportam em um ambiente de maior complexidade. Outra abordagem seria realizar uma análise mais profunda nos resultados obtidos a partir das ferramentas, a fim de mostrar o conhecimento que elas permitem extrair.

REFERÊNCIAS

BARQUIN, R.; EDELSTEIN, S. **Planning and Designing the Data Warehouse**. Prentice Hall, 1997.

BERTOLINI, Ana Virgínia A. G.; CHIAPPIN, Márcia Almeida; MAYOLO, Vianei Roberto; D'ARRIGO, Fernanda Pauletto; BARCELLOS, Paulo Fernando Pinto; DIAS, Deise Taiana de Ávila. Soluções business intelligence open source no suporte à estratégia organizacional. **Revista Inteligência competitiva**. São Paulo, v. 5, n. 2, 2015. p. 40-59.

Devmedia. **Business Intelligence: Conhecendo algumas ferramentas Open Source**. Disponível em: <https://www.devmedia.com.br/business-intelligence-conhecendo-algumas-ferramentas-open-source/31963>. Acesso em: 07 mai. 2017.

GIOIA, A.; CAZZIN, G.; DAMIANI, E. SpagoBI: a distinctive approach in open source business intelligence. **Digital Ecosystems and Technologies**. Phitsanulok, Tailândia, 2008.

GURA, Emanoély Fernanda; BENCK, Larissa Lourenço Nunes. **Construção de um *Data Warehouse*, Aliado a uma Ferramenta *Open Source Ireport* na Geração de Informações para Tomada de Decisão.** 2011.

IBGE. **Site do Instituto Brasileiro de Geografia e Estatística**. Disponível em: <http://downloads.ibge.gov.br/downloads_estatisticas.html>. Acesso em: 07 mai. 2017.

INMON, W.H. **Information System Arquiteture:** Development in 90's. Nova York: John Wiley & Sons Inc., 1993. Partes do livro disponível em: www.billinmon.com.

KIMBALL, Ralph; ROSS, Margy. **The Data Warehouse Toolkit:** The Complete Guide to Dimensional Modeling. John Wiley & Sons, 2011.

MACHADO, Felipe Nery Rodrigues. **Tecnologia e projeto de Data Warehouse:** uma visão multidimensional. 2. ed. São Paulo: Érica, 2006. 318 p. ISBN 85-365-0012-3

MENDES, Andréa; MARCIAL, Elaine Coutinho; FERNANDES, Fernando do Carmo. **Fundamentos da inteligência competitiva**. 1. ed. Brasília: Thesaurus, 2010. 133 p. (Inteligência competitiva; 1). ISBN 9788570629524.

PENTAHO. **Documentação da ferramenta Pentaho**. Disponível em: <https://help.pentaho.com/Documentation/7.1>. Acesso em 04 set. 2017.

RAMOS, Salvador. **Using excel in ETL Processes**. Disponível em <http://blogs.solidq.com/en/businessanalytics/using-excel-etl-processes>. Acesso em 04 set. 2017.

SPAGO. **Documentação da ferramenta SpagoBi**. Disponível em: <https://www.spagobi.org/homepage/services/documentation/>. Acesso em: 04 set. 2017.

TALEND. Site da ferramenta Talend Open Studio. Dísponível em: <https://www.talend.com>. Acesso em: 04 set. 2017.

TRUJILLO, Juan; MORA, Sergio Luján. **A UML Based Approach for Modeling ETL Proccess in Data Warehouses**. Departamento de Lenguajes y Sistemas Informáticos. Universidad de Alicante. 2003

VASCONCELOS, Alexandre Antônio de; PINHEIRO, André Caribé; ALVES, Ivani Aparecida, SANTOS, Rafael Tardelli Pacheco dos; SILVA, Wantuil. ***Business Inteligence:*** O papel do Data Warehouse no Processo de Suporte a Tomada de Decisão. Pós-Graduação Lato Sensu – Instituto de Educação Tecnológica. Belo Horizonte, 2010.