

UNIVERSIDADE FEDERAL DO MARANHÃO  
CURSO DE CIÊNCIA DA COMPUTAÇÃO

**JONATHAN MONTALVANE SILVA FERREIRA**

**UTILIZAÇÃO DA SUÍTE PENTAHO *COMMUNITY EDITION* PARA  
CONSTRUÇÃO DE PROJETOS DE *BUSINESS INTELLIGENCE***

São Luís

2016

**JONATHAN MONTALVANE SILVA FERREIRA**

**UTILIZAÇÃO DA SUÍTE PENTAHO *COMMUNITY EDITION* PARA  
CONSTRUÇÃO DE PROJETOS DE *BUSINESS INTELLIGENCE***

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Me. Antônio de Abreu Batista Júnior

São Luís  
2016

Ferreira, Jonathan Montalvane Silva

Utilização da Suíte Pentaho Community Edition para construção de projetos de business intelligence/ Jonathan Montalvane Silva Ferreira. - São Luis, 2016.

53 f.

Orientador: Antônio de Abreu Batista Júnior

Monografia (Graduação) - Universidade Federal do Maranhão, Curso de Ciência da Computação, 2016.

1. Dados 2. Camadas estratégicas 3. Tomadas de decisão 4. BI 5. Open source 6. Pentaho Community Edition

CDU 004.4:005

**JONATHAN MONTALVANE SILVA FERREIRA**

**UTILIZAÇÃO DA SUÍTE PENTAHO *COMMUNITY EDITION* PARA  
CONSTRUÇÃO DE PROJETOS DE *BUSINESS INTELLIGENCE***

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

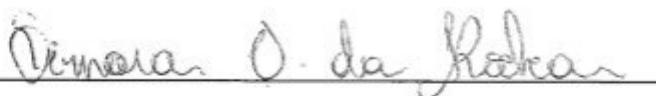
Orientador: Prof. Me. Antônio de Abreu Batista Júnior

Aprovada em: 22/03/2016

**BANCA EXAMINADORA**



Prof. Me. Antônio de Abreu Batista Júnior (Orientador)  
Universidade Federal do Maranhão



Prof. Dra. Simara Vieira da Rocha  
Universidade Federal do Maranhão



Prof. Me. Carlos Eduardo Portela Serra de Castro  
Universidade Federal do Maranhão

## **AGRADECIMENTOS**

À minha esposa Pollyanna e minha filha Luísa: minhas fontes diárias de alegria e motivação.

A todos os professores que contribuíram para minha formação acadêmica e profissional, em especial a meu orientador, professor Antônio, pela paciência e incentivo na elaboração desse trabalho.

E a todos que de alguma forma ajudaram nessa caminhada.

*"Tudo tem o seu tempo determinado, e há tempo para todo propósito debaixo do céu."*

Eclesiastes 3:1

## RESUMO

Recentemente, as organizações em geral têm se confrontado com uma quantidade cada vez maior de dados, advindos das mais variadas fontes, tais como, sistemas de gestão financeira, cadastros de vendas, entre outros. Para obter vantagens competitivas, as suas camadas estratégicas precisam avaliar o que estes dados querem lhes dizer, para embasarem o caminho que a organização precisa trilhar para se manter frente a uma concorrência cada vez mais especializada. Diante desse cenário, vem crescendo o conceito de *Business Intelligence* (BI), que se trata de um processo para transformar dados em conhecimento para auxiliar o processo de tomadas de decisão dentro das empresas.

Esta monografia tem como objetivo investigar o uso da suíte de BI *Pentaho Community Edition*, *open source* e gratuita, como uma alternativa de baixo custo às ferramentas de BI proprietárias existentes no mercado. Esta suíte se propõe a atender todas as fases de elaboração de um projeto de BI, desde a extração dos dados, até a apresentação ao usuário final.

A fim de estudar e investigar este problema um estudo de caso é proposto.

**Palavras-chaves:** Dados. Camadas estratégicas. Tomadas de decisão. BI. *Open source*. *Pentaho Community Edition*.

## ABSTRACT

Recently, organizations generally have grappled with an amount each increasing data arising from a variety of sources, such as financial management systems, sales records, among others. For competitive advantage, their strategic layers need to assess what these data mean for them, such restrictions the way that the organization needs to tread to stay ahead the competition increasingly specialized. In this scenario, it is growing the concept of Business Intelligence (BI) which is a process to transform data knowledge to assist the process of decision making within companies.

This monograph aims to investigate the use of the Pentaho BI Suite Community Edition, open source and free, as an alternative, low-cost, to proprietary BI tools on the market. This suite aims to meet all stages elaboration of a BI project, from the extraction of data to the presentation the end user.

In order to study this problem and investigate a case study be proposed.

**Keywords:** Data. strategic layers. decision making. BI. Open source. Pentaho Community Edition.



## LISTA DE FIGURAS

Figura 1: Arquitetura de um projeto de BI.....	15
Figura 2: Processo de ETL.....	17
Figura 3: Arquitetura de um Data Warehouse.....	19
Figura 4: Exemplo de granularidade.....	19
Figura 5: Modelo dimensional.....	20
Figura 6: Exemplo de Cubo Multidimensional.....	22
Figura 7: Data Mining.....	22
Figura 8: Suíte Pentaho.....	23
Figura 9: Pilha de Componentes Pentaho.....	24
Figura 10: Exemplo da utilização do Spoon (Kettle).....	24
Figura 11: Exemplo de job em Kettle.....	25
Figura 12: Componentes Pentaho OLAP.....	26
Figura 13: Software Schema Workbench.....	27
Figura 14: Interface do PAC.....	28
Figura 15: Interface do PUC.....	29
Figura 16: Fluxo do Pentaho CE.....	31
Figura 17: Site Inep.....	32
Figura 18: Diretório com a fonte de dados.....	32
Figura 19: Arquivo xls censo de 2010.....	33
Figura 20: Data warehouse censo.....	33
Figura 21: Interface de conexão mysql.....	34
Figura 22: Job de carga staging area.....	35
Figura 23: Transformação inicial staging area.....	36
Figura 24: Passo Cria Região.....	36
Figura 25: Transformação insere surrogate key Tipo.....	37
Figura 26: Resultado de carga do staging area.....	38
Figura 27: Job Carga do Data Warehouse.....	38
Figura 28: Transformação dimensão Tempo.....	39
Figura 29: Transformação dimensão Tipo.....	39



Figura 34: Criação do cubo multidimensional.....	42
Figura 35: Interface de publicação do cubo.....	42
Figura 36: Conexão bi server e Mysql.....	43
Figura 37: Visão Analítica no PUC.....	44
Figura 38: Exemplo de gráfico no PUC.....	44

## LISTA DE SIGLAS

BI – Business Intelligence

ETL – Extract, Transformation and Load

DW - Data Warehouse

OLTP - On-Line Transaction Processing

OLAP – On-Line Analytical Processing

MDX - Multi Dimension Expression

SQL – Structured Query Language

PAC – Pentaho Administration Console

PUC – Pentaho User Console

INEP – Instituto Nacional de Ensino e Pesquisa

XLS – Arquivo Excel

SGBD – Sistema de Gerencimento de Banco de Dados

PDS – Pentaho Design Studio

CDE – Community Dashboard Editor

ROLAP – Relational On-line Analytical Processing

RDBMS – Relational Data Base Management System

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	13
1.1 MOTIVAÇÃO.....	13
1.2 OBJETIVOS.....	14
1.3 ORGANIZAÇÃO DO TRABALHO.....	14
<b>2 REFERENCIAL TEÓRICO</b> .....	15
2.1 BUSINESS INTELLIGENCE.....	15
2.2 FASES DE UM PROJETO DE BI.....	15
2.2.1 <b>Definição do problema</b> .....	16
2.2.2 <b>Fontes de Dados e OLTP</b> .....	16
2.2.3 <b>ETL</b> .....	16
2.2.4 <b>Staging Area</b> .....	17
2.2.5 <b>Data Warehouse</b> .....	17
2.2.5.1 Granularidade.....	19
2.2.5.2 Modelagem Multidimensional.....	19
2.2.6 <b>Ferramentas OLAP</b> .....	21
2.2.7 <b>Data Mining</b> .....	22
2.3 SUÍTE PENTAHO.....	23
2.3.1 <b>Conceito</b> .....	23
2.3.2 <b>Pentaho Data Integration (Kettle ou spoon)</b> .....	24
2.3.2.1 Transformação.....	25
2.3.2.2 hop.....	25
2.3.2.3 job.....	25
2.3.3 <b>Mondrian OLAP</b> .....	26
2.3.3.1 Schema Workbench.....	26
2.3.4 <b>BI Server e Administration Console</b> .....	27
2.3.5 <b>PUC (Pentaho User Console)</b> .....	29
<b>3 ESTUDO DE CASO: CENSO SOBRE AS INSTITUIÇÕES DE ENSINO SUPERIOR</b> .....	30
3.1 OBJETIVO DO ESTUDO DE CASO.....	30
3.2 CENÁRIO.....	30
3.3 FONTE DE DADOS.....	31
3.4 INTEGRAÇÃO DE DADOS E ETL.....	32
3.5 STAGING AREA E DATA WAREHOUSE.....	33
3.6 PROCESSO DE ETL E CARGA DO DATA WAREHOUSE.....	34

3.6.1 <b>Conexão com o banco de dados</b> .....	34
3.6.2 <b>Job Carga Staging Area</b> .....	35
3.6.2.1 Carga inicial do <i>Staging Area</i> .....	35
3.6.2.2 Transformação Insere Tipo.....	37
3.6.3 <b>Job Carga Data Warehouse</b> .....	38
3.6.3.1 Transformação de carga da dimensão tempo.....	39
3.6.3.2 Transformação de carga da dimensão Tipo.....	39
3.6.3.3 Transformação de carga da dimensão Região.....	40
3.6.3.4 Transformação de carga da Tabela Fato.....	40
3.7 CRIAÇÃO DO CUBO CENSO.....	42
3.8 SERVIDOR E ADMINISTRAÇÃO DO PROJETO.....	43
3.9 ANÁLISE E GERAÇÃO DE RELATÓRIOS.....	43
3.9.1 <b>Cubo censo</b> .....	43
4 <b>CONCLUSÃO</b> .....	45
REFERÊNCIAS.....	46
APÊNDICES.....	48

## 1 INTRODUÇÃO

Em mundo cada vez mais competitivo, organizações de pequeno, médio e grande porte precisam gerenciar um volume de dados que crescem exponencialmente, dados esses que muitas vezes provém de fontes heterogêneas, e aparentemente sem relação entre si. Nesse contexto, surgiram novas formas de gerenciamento desses dados, reunindo as diversas fontes, e transformando-os em informações que possam ser usados para tomadas de decisões.

Apesar da importância de se otimizar a gerência de dados dentro das empresas, a grande maioria não dispõe de recursos para investir em projetos de solução de BI (*Business Intelligence*), uma vez que as ferramentas proprietárias existentes no mercado tem um custo elevado, inviabilizando a implantação de projetos, principalmente nas pequenas e médias empresas.

Visando reduzir os custos de implantações de projetos de BI, surgiram ferramentas *open source* e gratuitas, desenvolvidas e mantidas pela comunidade de programadores. Esses tipos de ferramentas podem servir como alternativa para as ferramentas proprietárias existentes no mercado, viabilizando a construção de projetos de BI, uma vez que não terão o custo com licenças de ferramentas pagas.

### 1.1 MOTIVAÇÃO

Hoje, sabe-se que a apesar de se reconhecer a importância de se otimizar a gerência de dados dentro das organizações, a grande maioria delas não sabe como lidar com esta questão. Os altos investimentos e a falta de conhecimento de muitos gerentes, tem sido uma barreira para implantação de processos de gestão dos dados.

No entanto, com o acirramento do mercado, muitas organizações estão aos poucos perdendo competitividade. A fim de se manterem vivas no mercado, as organizações precisam gerir eficientemente seus dados e impulsionar seus negócios. A sobrevivência de grande parte das organizações depende da existência de um ferramental adequado e de baixo custo para a gestão eficiente dos seus dados.

## 1.2 OBJETIVOS

O objetivo geral deste trabalho é investigar o uso da suíte Pentaho CE, como uma solução alternativa de baixo custo às ferramentas proprietárias de BI existentes no mercado.

Os objetivos específicos são listados:

- a. Apresentar as diversas fases de um ciclo de vida de um projeto de BI, desde a extração dos dados, até a apresentação das informações para tomadas de decisão;
- b. Demonstrar o uso das ferramentas da suíte Pentaho CE, para elaboração de projetos de BI;
- c. Elaborar um estudo de caso, aplicando a suíte estudada, e os conceitos que serão detalhados no capítulo anterior.

## 1.3 ORGANIZAÇÃO DO TRABALHO

O restante da monografia está estruturada da seguinte forma:

O segundo capítulo apresentará os principais conceitos sobre BI que serão utilizados para fundamentação teórica visando a construção do estudo de caso proposto.

Já o quarto capítulo apresenta o estudo de caso utilizando as ferramentas da suíte estudada.

Por fim, no quinto capítulo serão apresentadas as conclusões sobre o trabalho, além de propostas de trabalhos futuros.



## 2 REFERENCIAL TEÓRICO

Neste capítulo, serão apresentados os principais conceitos envolvidos na construção de um processo clássico de *Business Intelligence*, desde a delimitação do escopo do projeto, passando pelos processos de ETL, construção de um repositório do tipo *data warehouse*, até sua apresentação ao usuário final.

### 2.1 BUSINESS INTELLIGENCE

Conhecido em português como inteligência empresarial, trata-se de “um conjunto de ferramentas e aplicativos que oferece aos tomadores de decisão possibilidade de organizar, analisar, distribuir e agir, ajudando a organização a tomar decisões melhores e mais dinâmicas.” (BATISTA, 2004, p.121).

Segundo Silva (2011, p. 32), *Business Intelligence* “consiste na transformação metódica e consciente dos dados provenientes de quaisquer fontes de dados (estruturados e não estruturados) em novas formas de proporcionar informação e conhecimento dirigidos aos negócios e orientados aos resultados”.

### 2.2 FASES DE UM PROJETO DE BI

A seguir será descrito as principais fases de um projeto de *business intelligence*. A Figura 1 ilustra um exemplo de arquitetura de um projeto de bi, desde a extração dos dados até a apresentação ao usuário final.

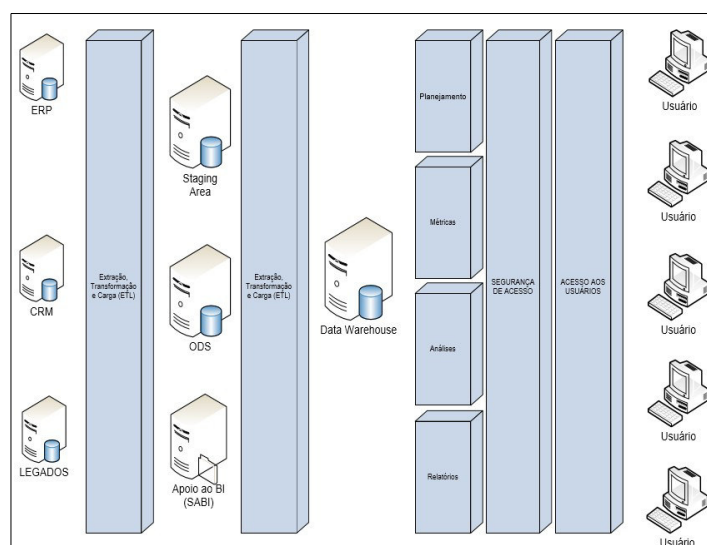


Figura 1: Arquitetura de um projeto de BI. Fonte: Auad, 2012, pag. 21

### 2.2.1 Definição do problema

Para usar inteligentemente a informação, os administradores de negócios necessitam, inicialmente, criar um plano de BI. “A implementação do plano requererá diversos processos. Muitos destes enquadram-se no conceito de gestão de conhecimento. Eles incluem identificar qual informação é importante para o processo de tomada de decisão.” (GORDON, GORDON, 2006, p. 252).

### 2.2.2 Fontes de Dados e OLTP

“O processo de integração de dados é um dos mais importantes relacionados com o ciclo de vida de um *data warehouse*, ele garante que os dados que alimentarão o DW sejam de qualidade e representativos para os assuntos escolhidos.” (CECI, 2012, p. 78).

Em um projeto de BI, os dados são extraídos das mais variadas fontes, tais como banco de dados, arquivos *excel*, *csv*, *access*, *txt*, OLTP (*on-line transaction processing*) etc, e reunidas no intuito de servirem de base para a construção do *data warehouse* do projeto.

Escolhidas as fontes que serão utilizadas nas análises, estas por muitas vezes advindas de fontes heterogêneas, são submetidas ao processo de ETL, para padronização dos dados, de forma que populem o *data warehouse* de forma correta.

### 2.2.3 ETL

Também conhecido por *cleansing* (limpeza), ETL (extração, transformação e carga/*load*) tem por objetivo melhorar a qualidade dos dados e gerar uma base separada para análise (um *data warehouse*) para não sobrecarregar as bases usadas pelos sistemas transacionais. A limpeza serve para eliminar inconsistências da base, completar dados, tratar valores nulos, eliminar registros duplicados, etc. (LOH, 2014, p. 33).

ETL é antes de tudo um processo. Os projetistas podem construir mecanismos para extração das fontes de dados, limpeza, e carga no DW, porém existem soluções automáticas no mercado que cumprem o papel de realizar esta etapa na construção de projetos de BI.

A Figura 2 mostra diversos fluxos de informações durante o processo de ETL, desde a extração, passando por vários processos, até a carga no repositório.

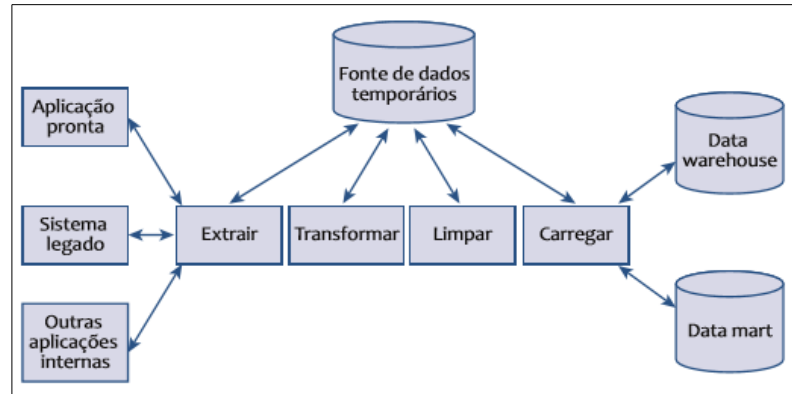


Figura 2: Processo de ETL. Fonte : Turban et al, 2009, pag. 72

#### 2.2.4 Staging Area

A *Staging area* é um local de armazenamento de dados intermediário, entre a fonte de dados e o *data warehouse*, cuja finalidade é facilitar a integração entre esses dois ambientes.

“Essa abordagem leva a uma outra vantagem na preparação de dados, uma vez que, os dados são armazenados em um sistema de banco de dados distintos, índices que podem ajudar a melhorar o desempenho do tratamento dos dados e podem ser livremente adicionados sem alterar o sistema de origem”. (BOUMAN, DONGEN, 2009, p. 119).

#### 2.2.5 Data Warehouse

Segundo Rezende e Abreu (2009, p. 213) *Data Warehouse* é um grande Banco de Dados que armazena dados de diversas fontes para futura geração de informações integradas, como base nos dados do funcionamento das funções empresariais operacionais de uma organização inteira.

Já, Corey apud Inmon (1997, p.12), definem que um *data warehouse* deve ter as seguintes propriedades:

**Orientado ao assunto:** Refere-se ao formato da organização das informações de modo a facilitar as consultas, ou seja, os dados serão

agrupados por assunto dos negócios da empresa, por exemplo: vendas, compras, produção, RH e etc.

**Integrado:** O *data warehouse* tem a função de armazenar os dados em um único ambiente, integrando dados de diversas fontes, arquivos XML, entre outros. No entanto, para a real integração, é necessário adotar alguns cuidados antecipadamente ao armazenamento no *data warehouse*.

**Não volátil:** Além de garantir a durabilidade das informações no tempo, essa propriedade também garante que os usuários somente terão acesso ao *data warehouse* com a possibilidade de somente leitura. Isso não significa que não haverá atualização dos dados, mas ocorrerá através de novas cargas de dados e, uma vez carregado, não mais poderá ser apagado. Diferentemente dos ambientes transacionais – OLTP, por intermédio das aplicações, os usuários podem executar: inclusão, alteração, exclusão e consulta dos dados.

**Variante no tempo:** Sem o elemento tempo, o *data warehouse* não teria muito sentido. O registro dos históricos das atualizações permite ao usuário conhecer qual era o estado de um determinado dado após uma atualização, uma vez que as novas entradas sempre serão mapeadas em um novo registro, ou seja, os dados contidos referem-se a algum momento de tempo específico. Para isso, os registros, quando carregados, recebem um atributo da unidade de tempo e nunca mais são atualizados. É essa característica que possibilita os analistas de negócios fazerem análises de tendências e visualizarem as variações das informações ao longo do tempo.

E a maior justificativa para os grandes volumes de dados dos *data warehouse* é exatamente a necessidade de manter os registros de históricos por tempo. A Figura 3 ilustra a arquitetura de um modelo de *data warehouse*. Nela pode ser visualizada as diversas fases de integração dos dados.

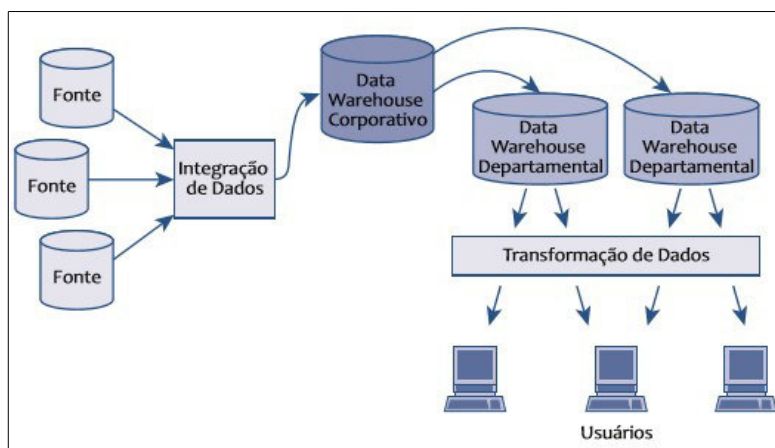


Figura 3: Arquitetura de um Data Warehouse. Fonte: Dill, 2002, pag. 18

### 2.2.5.1 Granularidade

Granularidade descreve o “nível de sumarização dos elementos e de detalhe disponíveis nos dados, considerando o mais importante aspecto no projeto de um *data warehouse*.” (MACHADO, 2008, p. 59).

A definição da granularidade constitui um dos aspectos mais importantes na construção do *data warehouse*, e por sua vez, de todo o projeto. Deve-se levar em conta, a quantidade de espaço em disco, e o tempo de resposta da busca pela informação.

A Figura 4 mostra um exemplo de níveis de granularidade. Em um nível mais baixo, os dados estão menos normalizados do que em um nível mais alto.

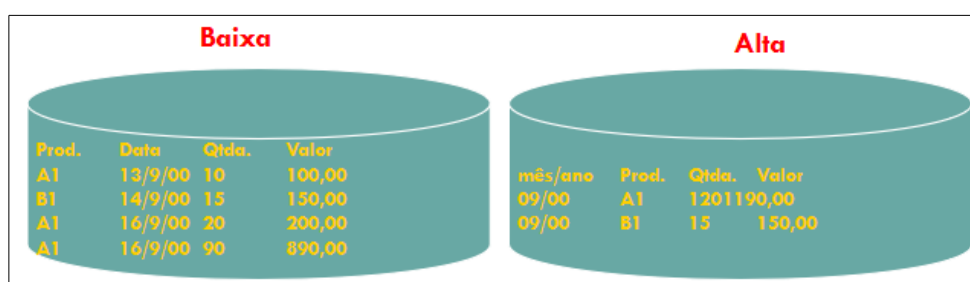


Figura 4: Exemplo de granularidade. Fonte: <http://slideplayer.com.br/slide/5648653/> (2016)

### 2.2.5.2 Modelagem Multidimensional

Para Goldschmidt e Passos (2005, p. 170) a “modelagem multidimensional é uma forma de modelagem de dados voltada para concepção e visualização de

conjuntos de medidas que descrevem aspectos comuns de um determinado assunto”.

Diferentemente de uma modelagem tradicional que tem como objetivo a normalização dos dados, este tipo de modelagem busca a facilidade de consulta dos usuários, admitindo redundância dos dados.

Ainda segundo Goldschmidt e Passos (2005, p. 171), uma modelagem dimensional é construído a partir de três componentes básicos:

**Fatos** - Um fato é uma coleção de itens de dados, composta de dados de medida e de contexto. Representa um item, ou uma transação ou um evento associado ao tema da modelagem.[...]

**Dimensões** – Uma dimensão é um tipo de informação que participa da definição de um fato[...]. Normalmente são descritivas ou classificatórias. Em geral, as perguntas “O quê? Quem? Onde? Quando?” ajudam a identificar as dimensões de um assunto.

**Medidas** - Uma medida é um atributo ou variável numérica que representa um fato[...]

A Figura 5 mostra um exemplo de modelagem dimensional do tipo estrela, em que temos a tabela de fatos no centro e suas dimensões ligadas diretamente a ela, ao contrário do tipo *snow flake*, em que uma ou mais tabelas de dimensões estão normalizadas.

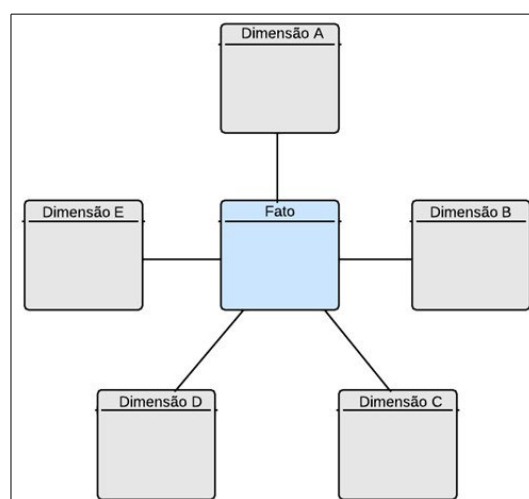


Figura 5: Modelo dimensional. Fonte: <http://imagens.canaltech.com.br/49726.68286-BIFato.png>

### 2.2.6 Ferramentas OLAP

Ferramentas OLAP (*On-line Analytical Processing*) são voltadas para a realização de consultas em bases de dados multidimensionais. As soluções OLAP apresentam uma alternativa para a publicação dos dados e informações vindas dos modelos dimensionais. A apresentação dessas informações pode ser de maneira tabular ou gráfica, tanto dos dados históricos armazenados nos repositórios *data warehouse*, bem como os dados reais para auxiliar o processo decisório.

Segundo Anzanello (2013), as principais características deste tipo de ferramentas, são:

**Consultas *ad-hoc*:** geradas pelos usuários finais de acordo com as suas necessidades de cruzar informações de uma forma não vista e que o levem a descoberta do que procuram. Segundo (Inmom, 2009) “são consultas com acesso casual único e tratamento de dados segundo parâmetros nunca antes utilizado de forma iterativa e heurística”.

***Slice and Dice*:** possibilita a alteração da perspectiva de visão. Serve para modificar a posição de uma informação, trocar linhas por colunas de maneira facilitar a compreensão dos usuários e girar o cubo sempre que houver necessidade.

***Drill down/up*:** consiste em realizar exploração em diferentes níveis de detalhes da informação. Com *drill down* dividi-se um item de resumo em seus componentes detalhados, como por exemplo, ano, semestre, trimestre, mensal e diário.

“A principal característica que está presente em todas as abordagens é o cubo multidimensional, capaz de filtrar os dados por diversas formas e modos customizados pelo usuário.” (Gouveia et al, 2011). A Figura 6 mostra um exemplo de representação para um cubo multidimensional, onde pode ser visualizado a face frontal com dados de vendas, e as dimensões compondo as outras faces do cubo.

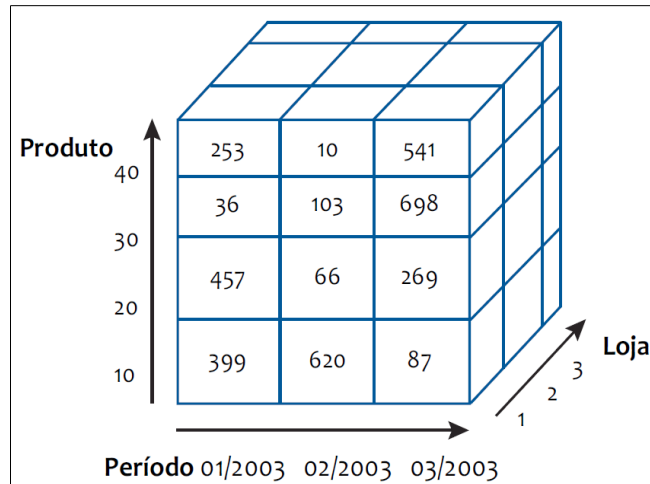


Figura 6: Exemplo de Cubo Multidimensional. Fonte: CECI *apud* CAMPOS, 2009, p. 85

### 2.2.7 Data Mining

“*Data Mining* (mineração de dados) é mais normalmente utilizado para análise estatística dos dados e descoberta de conhecimento. Análise estatística dos dados detecta padrões incomuns de dados e aplica técnicas de modelagem estatísticas e matemáticas para explicar os padrões.” (BALLARD, HERREMAN, 1998, pg 12, tradução nossa).

A Figura 7 ilustra algumas possibilidades de fontes de dados e resultados em um processo de mineração de dados.

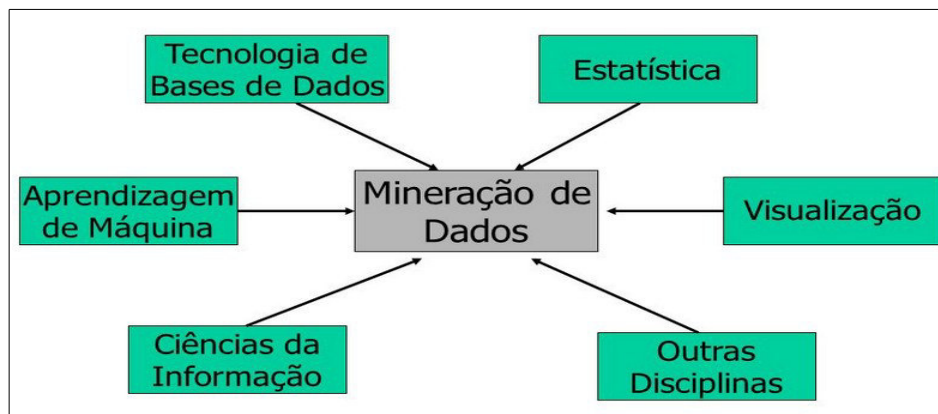


Figura 7: Data Mining. Fonte: [http://www.tutorialspoint.com/data\\_mining/images/dm\\_systems.jpg](http://www.tutorialspoint.com/data_mining/images/dm_systems.jpg) (2016)



## 2.3 SUÍTE PENTAHO

A seguir serão apresentadas algumas ferramentas da suíte Pentaho CE, para a elaboração de projetos de BI.

As ferramentas utilizadas neste trabalho podem ser adquiridas no endereço <http://sourceforge.net/projects/pentaho/>, todas de código aberto e gratuitas. A Figura 8 mostra o conjunto de ferramentas que compõem a suíte Pentaho.

Name	Modified	Size	Downloads / Week
Big Data Shims	2015-10-12		54
Business Intelligence Server	2015-10-12		2,990
Pentaho Metadata	2015-10-12		394
Data Integration	2015-10-12		10,039
Report Designer	2015-10-12		3,339
Big Data Preview	2012-01-30		16
Design Studio	2011-09-15		73
Report Design Wizard (Legacy)	2008-08-04		5
White Papers	2006-06-20		27

Figura 8: Suíte Pentaho. Fonte: <http://sourceforge.net/projects/pentaho/> (2016)

### 2.3.1 Conceito

De acordo com Bouman e Dongen (2009, p. 63) “Pentaho é uma suíte de *business intelligence*, em vez de um único produto: ele é feito por um conjunto de programas de computador que trabalham juntos para criar e oferecer soluções de *business intelligence*”.

Praticamente todos os programas da suíte Pentaho são feitos utilizando a linguagem de programação Java, no intuito de tornar as ferramentas portáteis e independentes de plataforma.

Na Figura 9, podem ser visualizadas as camadas de componentes da solução Pentaho e quais fases de um projeto de BI ela se propõe a atender.

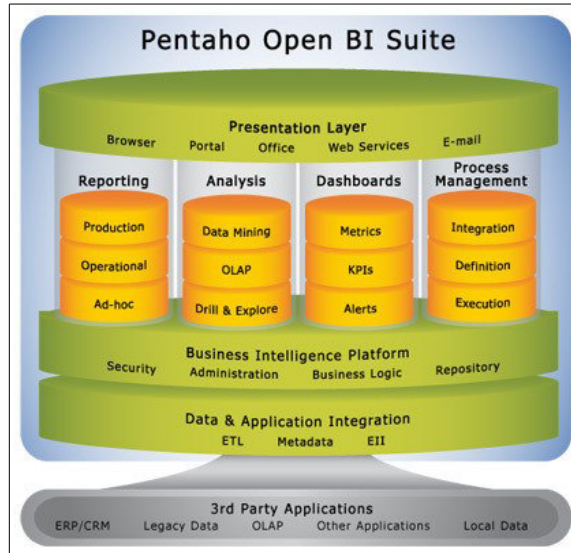


Figura 9: Pilha de Componentes Pentaho. Fonte: Bouman, Dongen, 2009, p. 64

### 2.3.2 Pentaho Data Integration (Kettle ou spoon)

“Kettle é um único produto, mas é composto por vários programas que são utilizados em diferentes fases de desenvolvimento do ciclo de implantação e de ETL.” (BOUMAN, DONGEN, 2009, p. 53).

A Figura 10 ilustra a interface da ferramenta Spoon, responsável pela implementação do processo de ETL da suíte Pentaho.

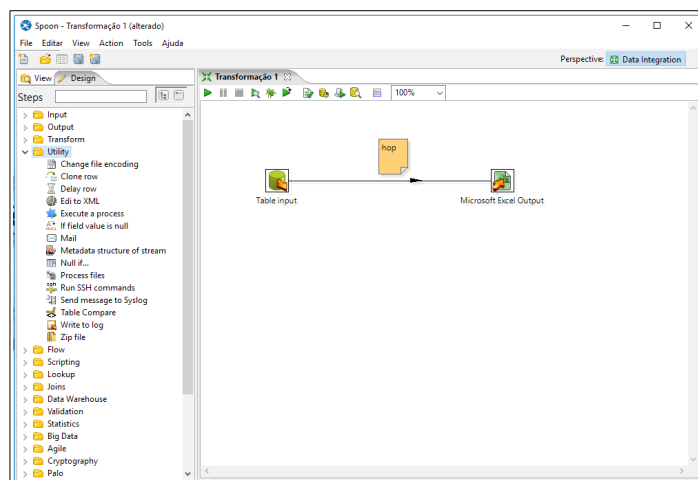


Figura 10: Exemplo da utilização do Spoon (Kettle). Fonte: Elaborado pelo autor utilizando o Spoon, (2016)

O software Spoon (Kettle) utiliza abstrações de transformações e *jobs* (conjunto de um ou mais transformações ou outros *jobs*) para implementar o processo de ETL. As seções seguintes apresentam os conceitos envolvidos na execução do Spoon.

### 2.3.2.1 Transformação

Segundo Bouman e Dongen (2009, p. 25), “a transformação é o carro-chefe de sua solução de ETL. Ele lida com a manipulação de linhas ou dados no sentido mais amplo possível da sigla de extração, transformação e carregamento. Ele consiste em uma ou mais etapas que realizam trabalhos ETL, tais como a leitura de dados a partir de arquivos, filtragem de linhas, limpeza de dados, ou carregar dados em um banco de dados.”

### 2.3.2.2 hop

Um hop, representado graficamente no Spoon por uma seta, liga um componente de origem a um de destino, e o caminho do fluxo de dados que transita entre eles.

### 2.3.2.3 job

Um *job* consiste em uma ou mais entradas de *jobs* que são executados em uma determinada ordem. A ordem de execução é determinada pelo emprego de saltos (*hops*) entre entradas de trabalho, bem como o resultado da execução propriamente dita (Bouman, Dougen, 2009, pag 30). A seguir, a Figura 11 mostra um exemplo de um *job* utilizando a ferramenta Spoon.

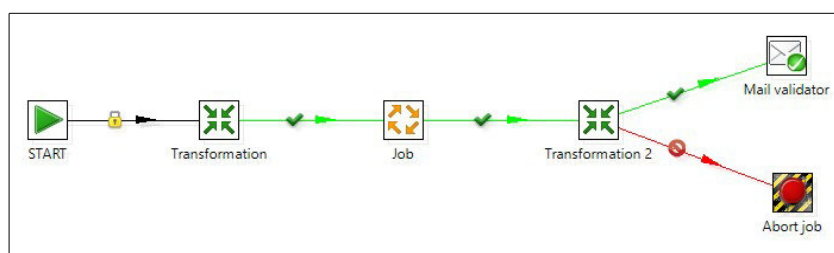


Figura 11: Exemplo de *job* em Kettle. Fonte: Construído pelo autor utilizando Spoon, (2016)

### 2.3.3 Mondrian OLAP

Mondrian é o motor Pentaho OLAP e traduz as consultas MDX (*Multi Dimension Expression*) para SQL baseado em um modelo multidimensional.

A Figura 12 mostra os componentes Pentaho OLAP, onde o cliente faz requisições via http, o engine Jpivot processa a requisição em MDX, e por sua vez, o engine Mondrian ROLAP(Relational OLAP) transforma a consulta MDX em consulta SQL, que busca na base de dados relacional, e retorna o resultado para ser exibido no *front-end*.

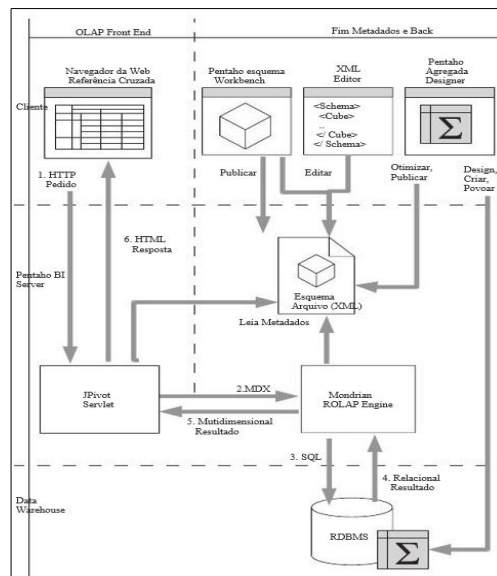


Figura 12: Componentes Pentaho OLAP. Fonte: BOUMAN, DONGEN, 2009, pg. 445

#### 2.3.3.1 Schema Workbench

Segundo Bouman e Dongen (2009, p. 442), o Schema Workbench “é a ferramenta visual para projetar e testar esquemas de cubos dimensionais. Mondrian usa esses esquemas de cubo dimensionais para interpretar MDX e traduzi-lo em consultas SQL para recuperar os dados de uma RDBMS”.

O Schema Workbench permite que o usuário crie visualmente, e teste cubos Mondrian OLAP.

Ele fornece as seguintes funcionalidades:

- Editor de esquema integrado com a fonte de dados subjacente para validação;
- Consultas MDX sobre o banco de dados;
- Navegue através das estruturas da base de dados. (BOUMAN, DONGEN, 2009, pg. 443)

A Figura 13 mostra a tela inicial do Schema Workbench, ferramenta para criação de cubos multidimensionais da suíte Pentaho.

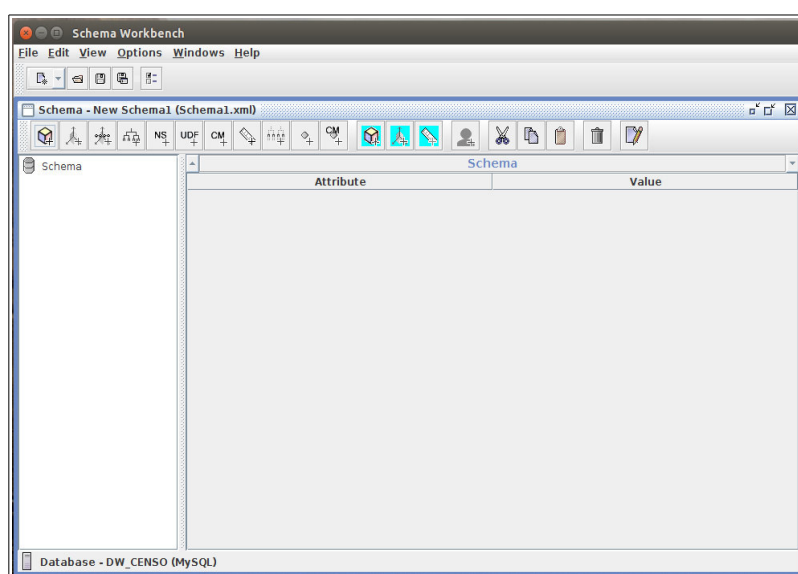


Figura 13: Software Schema Workbench. Fonte: Construído pelo autor utilizando o SW, (2016)

#### 2.3.4 BI Server e Administration Console

Segundo Bouman e Dongen (2009), Pentaho BI Server “é um conjunto de programas que trabalham para fornecer uma série de funcionalidades essenciais da suíte BI Pentaho”.

Ainda segundo (BOUMAN, DANGEN, 2009), o Pentaho BI Server pode ser dividido em três camadas:

**a) a plataforma:** As funcionalidades desta camada são, relativamente, de baixo nível e constituem uma infraestrutura básica da plataforma de BI. Essa camada fornece uma coleção de componentes que oferecem os seguintes serviços:

- repositório de soluções e motor de soluções;
- gerenciamento do pool de conexão com o banco de dados;

- autenticação de usuários e autorização de serviços;
- logging e serviços de auditoria;
- agendamento de tarefas;
- serviços de e-mail.

**b) componentes de BI:** Os seguintes componentes são encontrados nessa camada:

- camada de metadados;
- ad hoc serviço de relatório;
- motor ETL;
- motor Reporting;
- motor OLAP;
- motor de mineração de dados.

**c) a camada de apresentação:** Pentaho vem com uma interface web embutida, chamada de Console do Usuário. Esse forma um *front-end* que permite ao usuário humano interagir com o servidor. A camada de apresentação pode ser usada para navegação e, para abrir conteúdo existente como: relatórios, *dashboards* (painéis) e análises, porém em certa medida pode ser utilizado para criar novo conteúdo de BI.

As funcionalidades do *Bi Server* são executados através do seu *front-end web* chamada *Pentaho Administration console*, e é acessado através do endereço <http://localhost:8099/>, com usuário padrão “admin” e senha “password”.

Na Figura 14, pode ser visualizado a interface do PAC (Pentaho Administration Console).

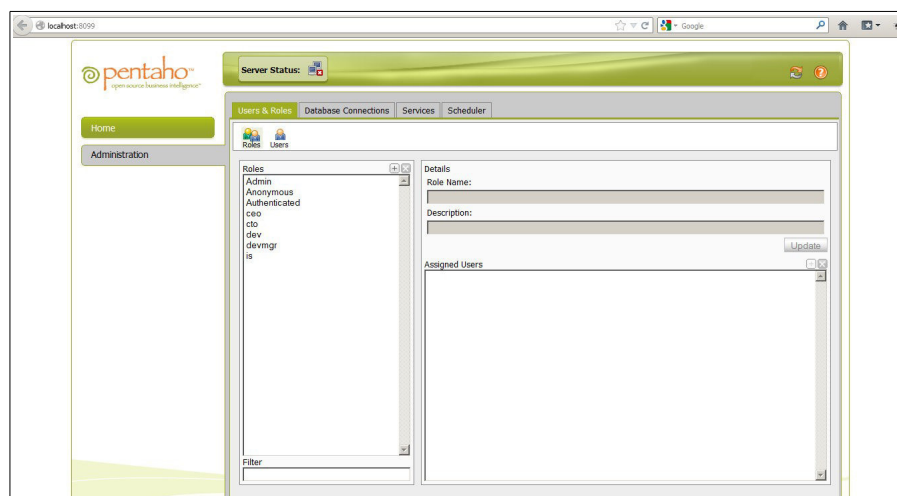


Figura 14: Interface do PAC. Fonte: Elaborado pelo autor utilizando o PAC, (2016)

### 2.3.5 PUC (Pentaho User Console)

“Pentaho vem com uma interface web chamada console do usuário. O console fornece um *front-end* que permite que um usuário humano interaja com o servidor.” (BOUMAN, DONGEN, 2009, p. 73).

O PUC é acessado através do endereço <http://localhost:8080/pentaho/> e tem como usuário padrão “joe” e senha “password”. Na Figura 15 pode ser visualizada a interface do PUC, ferramenta gráfica para gerar análises e relatórios.

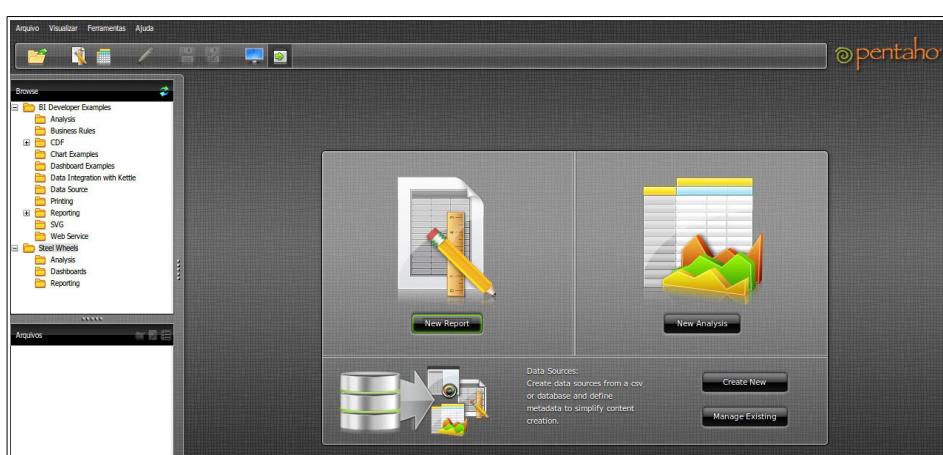


Figura 15: Interface do PUC. Fonte: Extraído pelo autor do PUC, (2006)

Neste capítulo foram apresentadas algumas das fundamentações teóricas que envolvem o tema *business intelligence*, além das ferramentas do Pentaho CE, que servirão de base para implementar o estudo de caso proposto. No próximo capítulo será elaborado o estudo de caso, utilizando as ferramentas estudadas.

### 3 ESTUDO DE CASO: CENSO SOBRE AS INSTITUIÇÕES DE ENSINO SUPERIOR

Neste capítulo é apresentado o estudo de caso, visando avaliar a utilização da suíte Pentaho CE como alternativa para a construção de projetos de BI. Para isso, serão utilizados dados históricos sobre o censo da quantidade de instituições de ensino superior nas diversas regiões do país, entre os anos de 2003 a 2013.

#### 3.1 OBJETIVO DO ESTUDO DE CASO

O estudo de caso que será apresentado neste trabalho, tem a finalidade de demonstrar que, a suíte de *Business Intelligence* Pentaho CE, ainda que *open source* e gratuita, é capaz de implementar as fases de um projeto de BI, desde a execução do processo de ETL, e carga do *data warehouse*, até a apresentação dos dados aos usuários finais.

Se bem-sucedido, organizações que dispõem de recursos escassos para implementação de projetos de BI, podem obter uma economia considerável, uma vez que, não necessitam pagar por licenças de ferramentas proprietárias, o que poderia impedir a viabilidade de projetos.

#### 3.2 CENÁRIO

O INEP é uma autarquia federal vinculada ao Ministério da Educação (MEC), cuja missão é promover estudos, pesquisas e avaliações sobre o Sistema Educacional Brasileiro com o objetivo de subsidiar a formulação e implementação de políticas públicas para a área educacional a partir de parâmetros de qualidade e equidade, bem como produzir informações claras e confiáveis aos gestores, pesquisadores, educadores e público em geral.<sup>1</sup>

Todos os anos, o INEP divulga em seu sítio, dados relativos a educação formal no Brasil. Os dados são públicos e estão disponíveis para download em uma de suas páginas dentro do portal.

Para o estudo de caso proposto neste trabalho, escolhemos dados sobre o censo da criação de diferentes instituições de ensino superior, públicas e privadas,

---

<sup>1</sup> Disponível : <<http://portal.inep.gov.br/conheca-o-inep>>. Acesso em 10 de set. 2015



entre os anos de 2003 e 2013. É interessante que os dados históricos, sejam analisados, no intuito de verificar a evolução dos resultados, comparar dados, para gerar conhecimento em eventuais tomadas de decisões.

O estudo de caso proposto tem o intuito de demonstrar os passos da construção de projeto de BI utilizando ferramentas *open source* da suíte Pentaho *Community Edition*. A Figura 16 mostra um fluxo de um projeto utilizando a suíte Pentaho CE.

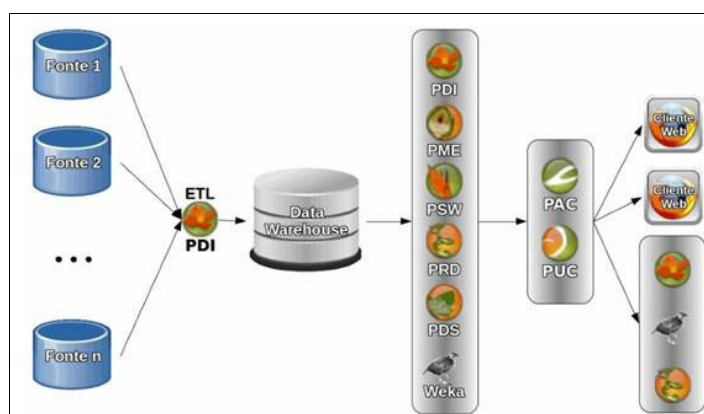


Figura 16: Fluxo do Pentaho CE. Fonte: <http://arquivo.devmedia.com.br/>

### 3.3 FONTE DE DADOS

A fonte de dados para a construção do projeto, foram extraídas do portal do Inep<sup>2</sup>, na página <http://portal.inep.gov.br/basica-levantamentos-acessar>, onde estão agrupados dados de cada ano do censo. Os dados estão armazenados em arquivos de extensão .xls, e compactados em arquivos .rar, que podem ser abertos através de programas descompactadores.

Para compor a base de análise, foram extraídos do site do INEP, 11 (onze) arquivos contendo a quantidade de instituições de ensino superior, entre os anos de 2003 e 2013.

<sup>2</sup> Disponível :<<http://portal.inep.gov.br/basica-levantamentos-acessar>> Baixado em 05 de ago. 2015

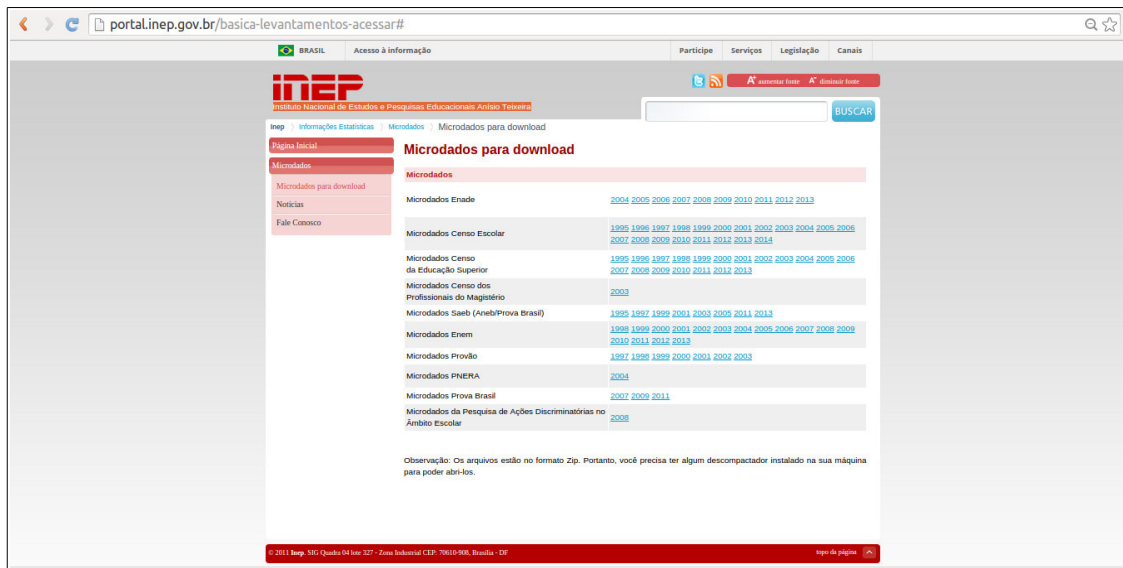


Figura 17: Site Inep. Fonte: <http://portal.inep.gov.br>. Acesso em Set. 2015

### 3.4 INTEGRAÇÃO DE DADOS E ETL

Realizados os downloads dos arquivos, eles foram agrupados em um diretório. Além disso, os arquivos foram renomeados no intuito de padronizar a nomenclatura.

O processo de ETL e *data quality* foi iniciado com a utilização do *software* Calc, da suíte LibreOffice, quando planilhas não utilizadas foram deletadas, e dados foram “limpos”, antes de serem utilizados como fonte de dados pelo Spoon. O resultado pode ser visualizado abaixo:

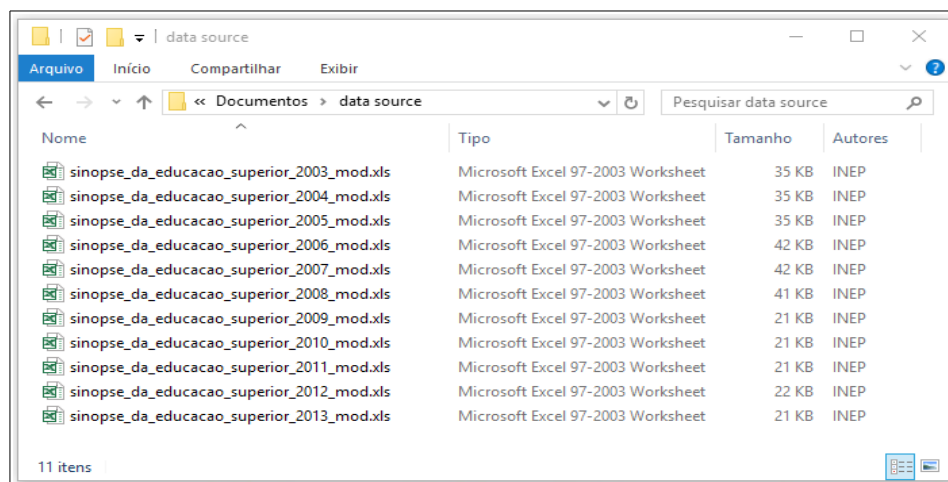


Figura 18: Diretório com a fonte de dados. Fonte: Construído pelo autor (2016)

A Figura 19 mostra a planilha sinopse\_da\_educacao\_superior\_2010\_mod.xls, que contém os dados referentes ao ano de 2010, do censo.

ESTADO	COMPETENCIA	UNIVERIDADES	CENTROS UNIVERSTARIOS	FACULDADES	# E CEFET
RONDÔNIA	FEDERAL	1	0	0	1
RONDÔNIA	ESTADUAL	0	0	0	0
RONDÔNIA	MUNICIPAL	0	0	0	0
RONDÔNIA	PARTICULAR	0	1	28	0
ACRE	FEDERAL	1	0	0	0
ACRE	ESTADUAL	0	0	0	0
ACRE	MUNICIPAL	0	0	0	0
ACRE	PARTICULAR	0	0	8	0
AMAZONAS	FEDERAL	1	0	0	1
AMAZONAS	ESTADUAL	1	0	0	0
AMAZONAS	MUNICIPAL	0	0	0	0
AMAZONAS	PARTICULAR	0	4	12	0
ROSIANA	FEDERAL	1	0	0	1
ROSIANA	ESTADUAL	1	0	0	0
ROSIANA	MUNICIPAL	0	0	0	0
ROSIANA	PARTICULAR	0	0	4	0
PARÁ	FEDERAL	3	0	0	1
PARÁ	ESTADUAL	1	0	0	0
PARÁ	MUNICIPAL	0	0	0	0
PARÁ	PARTICULAR	1	2	23	0
AMAPÁ	FEDERAL	1	0	0	0
AMAPÁ	ESTADUAL	1	0	0	0
AMAPÁ	MUNICIPAL	0	0	0	0
AMAPÁ	PARTICULAR	0	0	13	0
TOCANTINS	FEDERAL	1	0	0	1
TOCANTINS	ESTADUAL	1	0	0	0
TOCANTINS	MUNICIPAL	0	1	5	0

Figura 19: Arquivo xls censo de 2010. Fonte: Extraído do site do INEP, (2015)

### 3.5 STAGING AREA E DATA WAREHOUSE

A *Staging Area* e *Data warehouse* foram construídos utilizando o banco de dados *Mysql Community Server*, versão 5.7.10, e o SGBD *Mysql Workbench*, versão 6.3.5 CE. Tais softwares não serão abordados neste trabalho.

Optou-se por implementar a *Staging Area* utilizando uma tabela, dentro do *schema* do *data warehouse*. O resultado do modelo físico, pode ser visualizado na Figura 20:

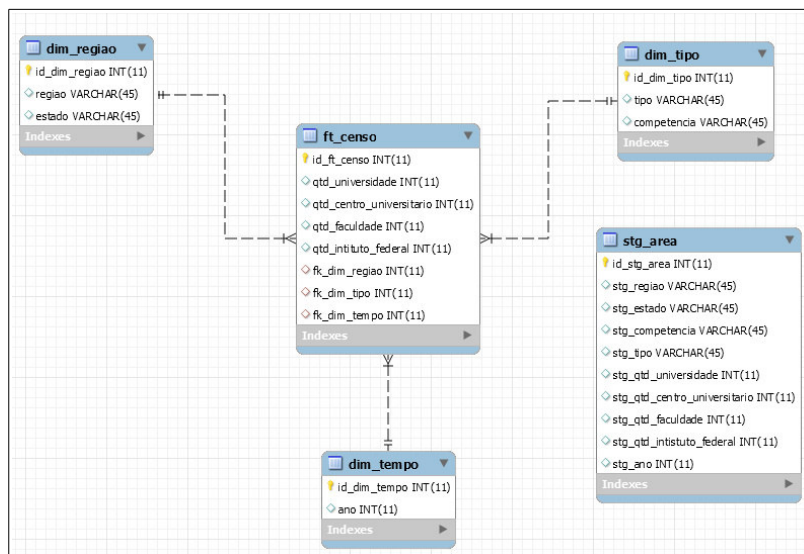


Figura 20: Data warehouse censo. Fonte: Construído pelo autor utilizando o *Mysql Workbench*, (2016)

O *script sql* de criação do *Staging Area* e *Data Warehouse*, por ser visualizado no Apêndice 1.

### 3.6 PROCESSO DE ETL E CARGA DO DATA WAREHOUSE

Uma vez projetado o *data warehouse*, inicia-se elaboração do processo de ETL(extração, transformação e carga) do DW, do estudo de caso proposto. Para isso, foi utilizada a ferramenta da suíte Pentaho CE, Spoon, versão 5.0.1 - stable.

Foram criados dois jobs, um para carga da *Staging Area* e outro para carga das dimensões e tabela fato do *data warehouse*.

#### 3.6.1 Conexão com o banco de dados

Uma vez que o *schema* do *data warehouse* e *staging area* foram implementados, precisamos conectar o Spoon à base de dados. A Figura 21 mostra a interface de conexão com o banco de dados MYSQL.

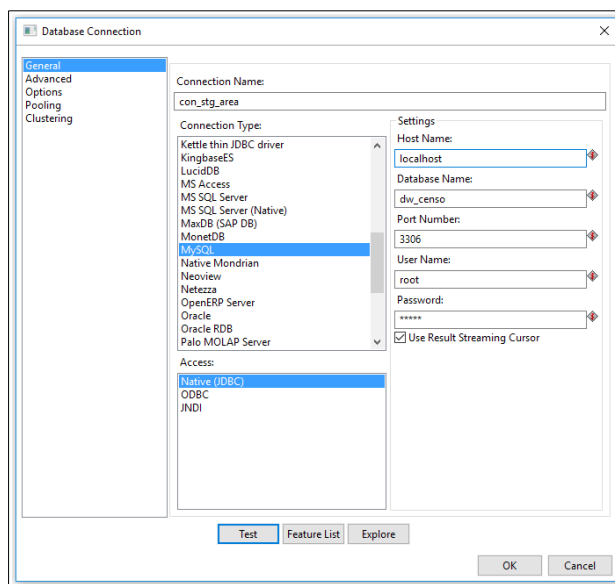


Figura 21: Interface de conexão mysql. Fonte: Criado pelo autor através do Spoon. (2016)

Para estabelecer uma conexão entre o Spoon e o banco de dados Mysql, é necessário adicionar o conector JDBC na pasta “/lib” do Spoon.

### 3.6.2 Job Carga Staging Area

O primeiro *job*, nomeado de “job\_carga\_staging\_area”, tem a finalidade de fazer a extração dos dados da fonte de dados, e finalizar com a carga do *staging area*, sendo composto por três transformações: a carga inicial dos dados advindos da fonte de dados, representada pela transformação “carga inicial staging area”, seguido da inserção das chaves das dimensões “tipo” e “região”, representadas pelas transformações “Inserere Tipo” e “Inserere Região”. Após a execução das transformações, caso a execução seja mal sucedida, o projetista receberá uma notificação. O resultado do processamento pode ser visualizado na Figura 22.



Figura 22: *Job* de carga *staging area*. Fonte: Construído pelo autor através do Spoon (2016)

Na próxima seção serão apresentadas as transformações que compõem o *job* “job\_carga\_staging\_area”.

#### 3.6.2.1 Carga inicial do *Staging Area*

A transformação se inicia com as entradas das fontes de dados em arquivos xls, extraídas do site do INEP. Após isso, colunas de tipos de faculdades são agrupadas e suas quantidades são somadas. Regiões são criadas a partir da coluna ESTADO, contido na fonte de dados. Feito isso, foi criada uma nova coluna intitulada ANO, que contém o ano respectivo à elaboração de cada censo. E por fim, são

excluídas colunas extras, e realizado o “append” (junção) dos dados e carga no *staging area*. Todo o processo pode ser visualizado na Figura 23:

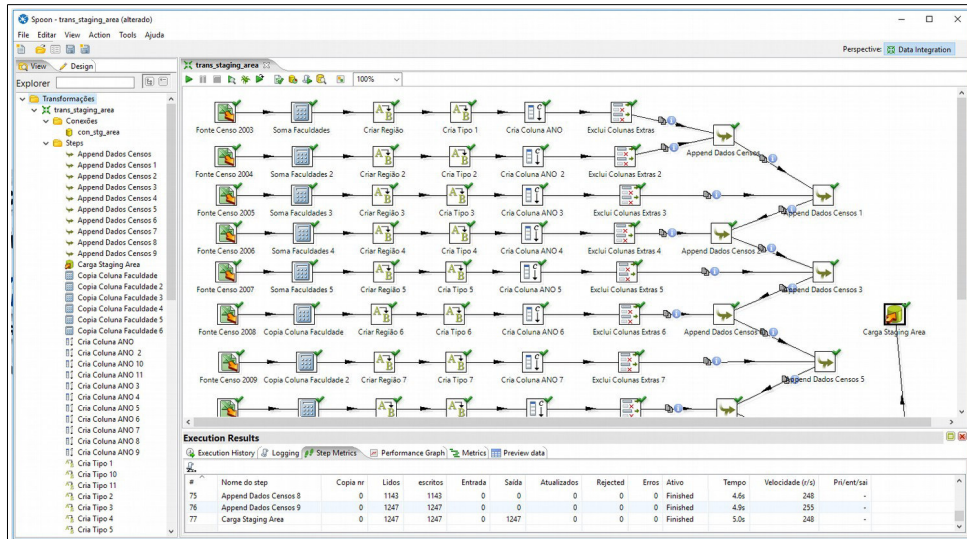


Figura 23: Transformação inicial staging area. Fonte: Elaborado pelo autor utilizando o Spoon (2016)

Na Figura 24, pode se visualizado o passo “Cria Região”, do tipo *value mapper*, responsável por ler os dados da coluna ESTADO e adicionar a uma nova coluna “TMP\_REGIAO”, a região ao qual pertence.

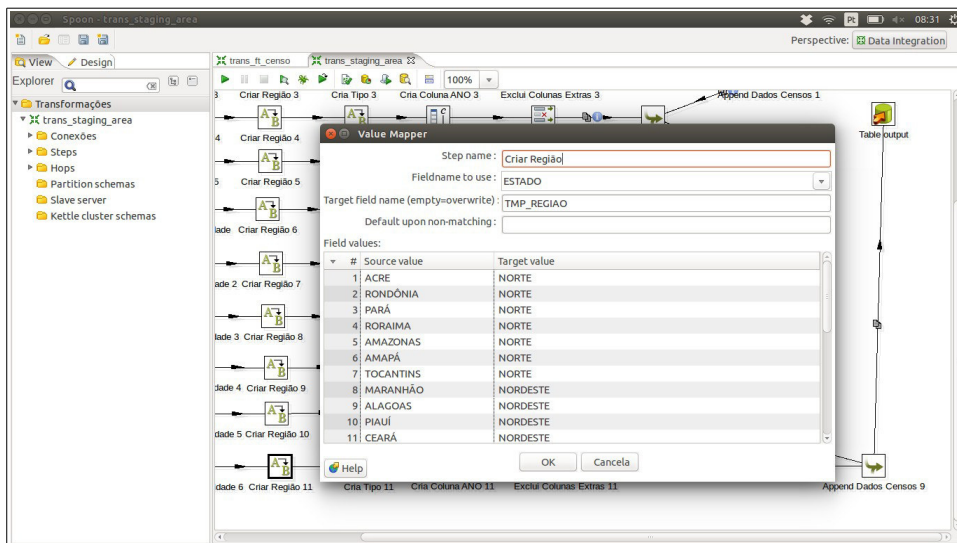


Figura 24: Passo Cria Região. Fonte: Elaborado pelo autor pelo Spoon (2016)

### 3.6.2.2 Transformação Inseere Tipo

A transformação “Inseere Tipo” inicia extraindo os dados do *Staging Area*, já carregados pela transformação carga inicial. O objetivo dessa transformação é popular o *Staging Area* com as *surrogate key*<sup>3</sup> da dimensão *dim\_tipo* para posterior carga do *data warehouse*. A Figura 25 ilustra o processo:

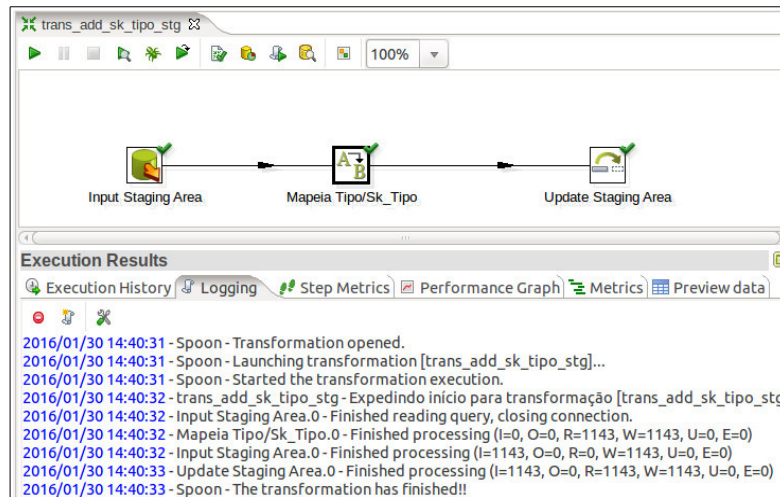


Figura 25: Transformação inseere *surrogate key* Tipo. Fonte: Elaborado pelo autor, através do Spoon (2016)

Uma vez que todos os dados foram carregados no *staging area* do projeto, podemos conectá-lo aos processos que executarão a carga das dimensões e tabela de fato, do *data warehouse*. A Figura 26 mostra o resultado da carga dos dados no banco de dados.

<sup>3</sup> Em um banco de dados, é uma chave de substituição. É um identificador único para cada entidade do mundo modelado ou um objeto no banco de dados.

The screenshot shows a MySQL Workbench window with a query result for 'stg\_area'. The query is 'SELECT \* FROM db\_censo.stg\_area;'. The result grid displays 28 rows of data with columns: id\_stg\_area, stg\_regiao, stg\_estado, stg\_competencia, stg\_boa, stg\_id\_universidade, stg\_id\_centro\_universitario, stg\_id\_faculdade, stg\_id\_instituto\_federal, and stg\_ano.

id_stg_area	stg_regiao	stg_estado	stg_competencia	stg_boa	stg_id_universidade	stg_id_centro_universitario	stg_id_faculdade	stg_id_instituto_federal	stg_ano
1	NORTE	RONDÔNIA	FEDERAL	PÚBLICA	1	0	0	0	2003
2	NORTE	RONDÔNIA	ESTADUAL	PÚBLICA	0	0	0	0	2003
3	NORTE	RONDÔNIA	MUNICIPAL	PÚBLICA	0	0	0	0	2003
4	NORTE	RONDÔNIA	PARTICULAR	PRIVADA	0	1	23	0	2003
5	NORTE	ACRE	FEDERAL	PÚBLICA	1	0	0	0	2003
6	NORTE	ACRE	ESTADUAL	PÚBLICA	0	0	0	0	2003
7	NORTE	ACRE	MUNICIPAL	PÚBLICA	0	0	0	0	2003
8	NORTE	ACRE	PARTICULAR	PRIVADA	0	0	5	0	2003
9	NORTE	AMAZONAS	FEDERAL	PÚBLICA	1	0	0	1	2003
10	NORTE	AMAZONAS	ESTADUAL	PÚBLICA	1	0	1	0	2003
11	NORTE	AMAZONAS	MUNICIPAL	PÚBLICA	0	0	0	0	2003
12	NORTE	AMAZONAS	PARTICULAR	PRIVADA	0	2	12	0	2003
13	NORTE	ROSIANA	FEDERAL	PÚBLICA	1	0	0	0	2003
14	NORTE	ROSIANA	ESTADUAL	PÚBLICA	0	0	0	0	2003
15	NORTE	ROSIANA	MUNICIPAL	PÚBLICA	0	0	0	0	2003
16	NORTE	ROSIANA	PARTICULAR	PRIVADA	0	0	5	0	2003
17	NORTE	PARÁ	FEDERAL	PÚBLICA	2	0	0	1	2003
18	NORTE	PARÁ	ESTADUAL	PÚBLICA	1	0	0	0	2003
19	NORTE	PARÁ	MUNICIPAL	PÚBLICA	0	0	0	0	2003
20	NORTE	PARÁ	PARTICULAR	PRIVADA	1	1	13	1	2003
21	NORTE	AMAPÁ	FEDERAL	PÚBLICA	1	0	0	0	2003
22	NORTE	AMAPÁ	ESTADUAL	PÚBLICA	0	0	0	0	2003
23	NORTE	AMAPÁ	MUNICIPAL	PÚBLICA	0	0	0	0	2003
24	NORTE	AMAPÁ	PARTICULAR	PRIVADA	0	0	7	0	2003
25	NORTE	TOCANTINS	FEDERAL	PÚBLICA	1	0	0	0	2003
26	NORTE	TOCANTINS	ESTADUAL	PÚBLICA	1	0	0	0	2003
27	NORTE	TOCANTINS	MUNICIPAL	PÚBLICA	0	0	1	0	2003
28	NORTE	TOCANTINS	PARTICULAR	PRIVADA	0	1	14	0	2003

Figura 26: Resultado de carga do *staging area*. Fonte: Elaborado pelo autor, utilizando Mysql Workbench (2016)

### 3.6.3 Job Carga Data Warehouse

Após a carga dos dados advindos da fonte de dados e carga do *staging area*, o *data warehouse* está pronto para ser carregado. Na Figura 27, o *job* de carga do *data warehouse* pode ser visualizado. O *job* inicia extraindo os dados contidos no *staging area*, carregando as dimensões *dim\_tipo*, *dim\_regiao* e *dim\_tempo* e finalizando com a carga da tabela fato *ft\_censo*.

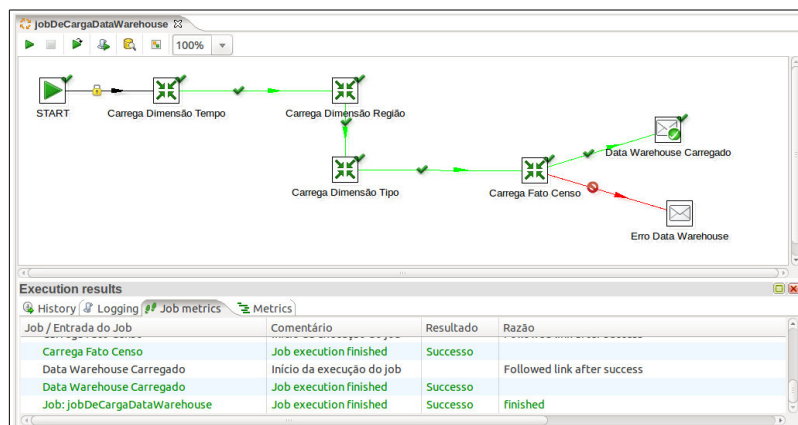


Figura 27: Job Carga do Data Warehouse. Fonte: Elaborado pelo autor pelo Spoon (2016)



### 3.6.3.1 Transformação de carga da dimensão tempo

A Figura 28 descreve a transformação de carga da dimensão tempo. Inicia extraindo os dados da coluna tempo, na tabela que contém a *staging area*, ordenando os dados por aquela coluna, excluindo duplicatas e por fim, carregando os dados na dimensão tempo, como pode ser visualizado na figura abaixo.

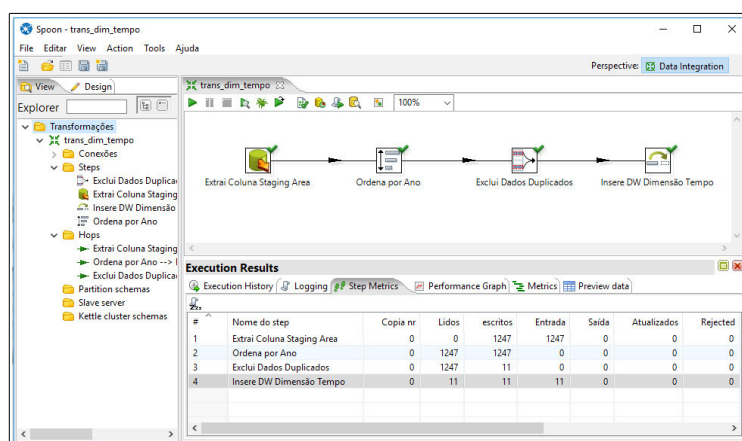


Figura 28: Transformação dimensão Tempo. Fonte: Elaborado pelo autor utilizando o Spoon, (2016)

### 3.6.3.2 Transformação de carga da dimensão Tipo

Após a carga da dimensão tempo, abaixo pode ser visualizada a transformação de carga da dimensão tipo. Essa transformação contém o mesmo fluxo de informações da carga de dimensão tempo, o que pode ser visualizado na Figura 29.

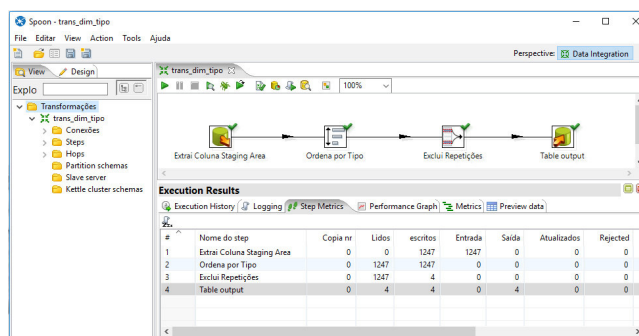


Figura 29: Transformação dimensão Tipo. Fonte: Elaborado pelo autor, (2016)

### 3.6.3.3 Transformação de carga da dimensão Região

A carga da dimensão Região segue o mesmo padrão das duas outras cargas de dimensões, como pode ser visualizado na Figura 30:

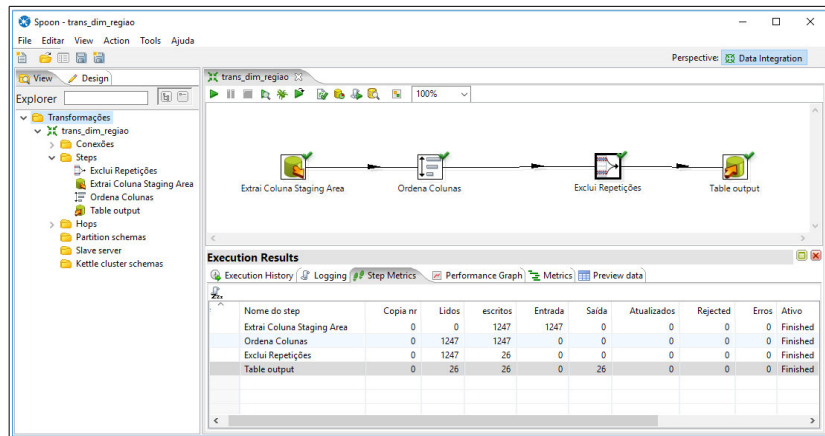


Figura 30: Transformação de dimensão Região. Fonte: Elaborado pelo autor utilizando o Spoon, (2016)

### 3.6.3.4 Transformação de carga da Tabela Fato

Uma vez que todas as dimensões foram carregadas, inicia-se a carga da tabela fato. A transformação extrai as medidas contidas no *staging area*, e as chaves primárias das dimensões tipo, tempo, e região. Os dados necessários são ordenados por seus "Ids", e as tabelas são comparadas e carregadas na tabela fato. Todo o fluxo de informações podem ser visualizado na Figura 31.

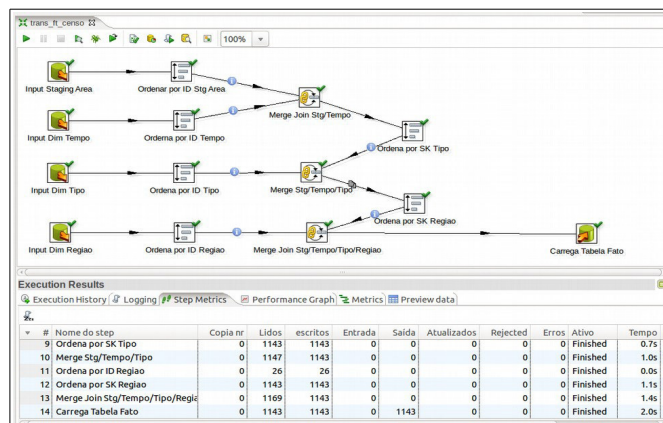


Figura 31: Transformação de carga Tabela Fato. Fonte: Elaborado pelo autor usando o Spoon, (2016)

Na Figura 32, pode ser visualizado o passo “Input Staging Area”, do tipo *input table*, responsável por ler os dados do *staging area* e carregar a tabela de fatos.

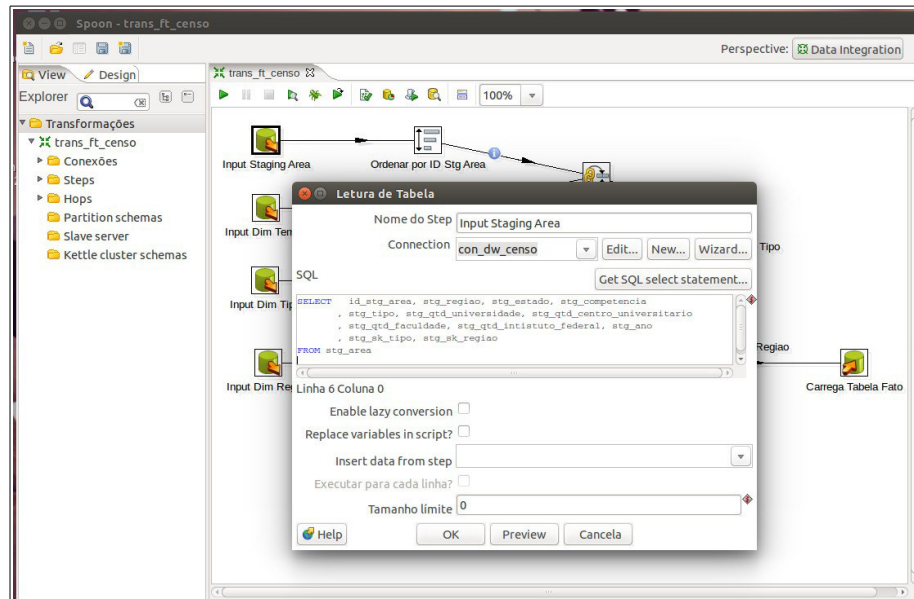


Figura 32: Leitura da *staging area*. Fonte: Elaborado pelo autor. (2016)

Ao final do processo de ETL utilizando o Spoon, os dados extraídos das fontes de dados são armazenadas no *data warehouse*, como pode ser visualizado na Figura 33:

The screenshot shows a SQL query result window with the following data:

id_ft_censo	qtd_universidade	qtd_centro_universitario	qtd_faculdade	qtd_instituto_federal	ft_dim_regiao	ft_dim_tipo	ft_dim_tempo
1049	2	2	33	0	22	4	5
1050	2	2	34	0	22	4	6
1051	2	2	34	0	22	4	7
1052	2	2	33	0	22	4	8
1053	2	2	29	0	22	4	9
1054	2	2	27	0	22	4	10
1055	2	2	29	0	22	4	11
1093	2	2	4	1	23	4	2
1236	0	2	12	0	26	4	1
710	0	3	12	5	15	4	2
711	0	3	84	7	15	4	3
712	0	3	84	7	15	4	3
713	0	3	83	6	15	4	4
714	0	3	79	4	15	4	5
715	0	3	81	0	15	4	6
716	0	3	81	0	15	4	7
717	0	3	81	0	15	4	8
718	0	3	81	0	15	4	9

Figura 33: Resultado da Carga do Data Warehouse. Fonte: Construído pelo autor, (2016)

### 3.7 CRIAÇÃO DO CUBO CENSO

Uma vez alimentado o *data warehouse* do estudo de caso, passamos a criar o cubo dimensional, através do Schema Workbench, que após publicado, poderá ser lido por ferramentas OLAP, para análises e geração de relatórios, o que pode ser visualizado na Figura 34.

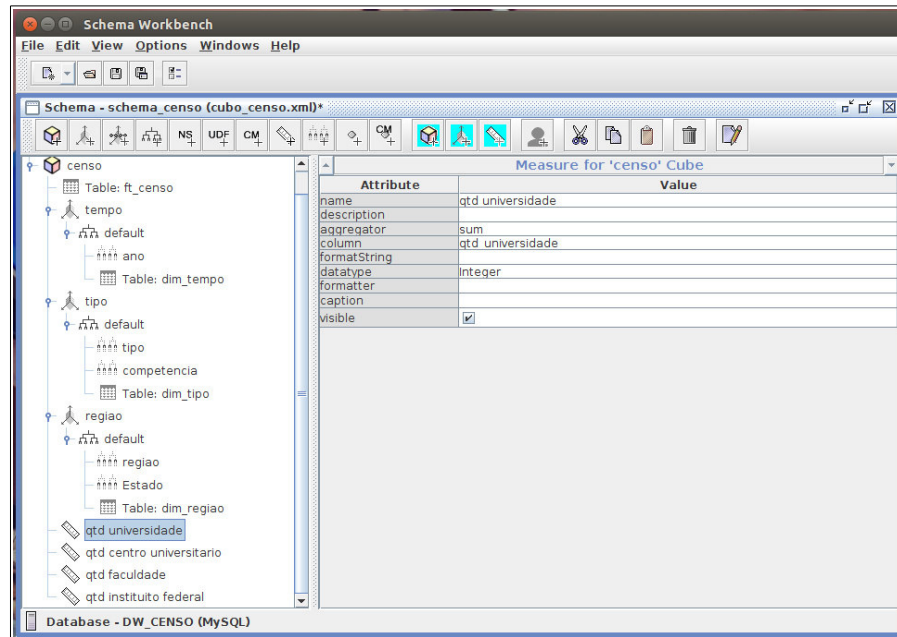


Figura 34: Criação do cubo multidimensional. Fonte: Elaborado pelo autor, utilizando o SW, (2016)

O resultado da criação do cubo “censo”, é um arquivo XML, intitulado **cubo\_censo.xml**, e pode ser visualizado no Apêndice 02.

Por fim, o cubo precisa ser publicado no servidor Bi server, para que possa ser acessado pelos usuários. A Figura 35 ilustra a interface de publicação do cubo no servidor.

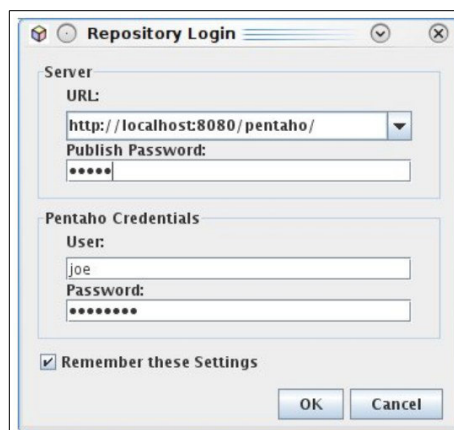


Figura 35: Interface de publicação do cubo. Fonte: Elaborado pelo autor, usando o SW, (2016)

### 3.8 SERVIDOR E ADMINISTRAÇÃO DO PROJETO

Uma vez o cubo criado, precisamos adicionar uma conexão da base de dados no administration console, para que possa ser visualizado no PUC. A Figura 36 mostra a criação da conexão entre o bi server e o banco de dados mysql.

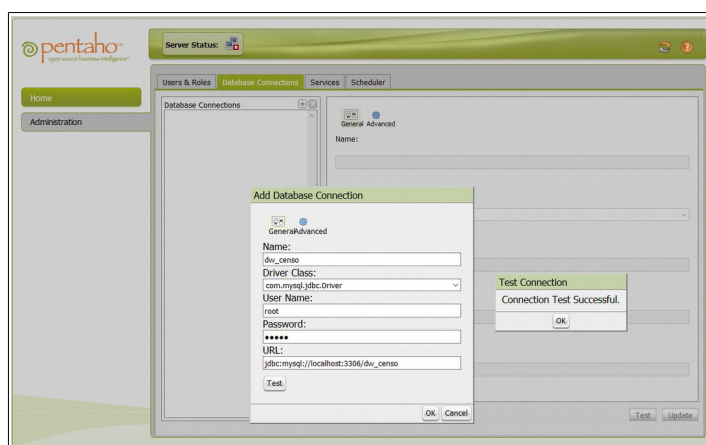


Figura 36: Conexão bi server e Mysql. Fonte: Construído pelo autor, utilizando o PAC (2016)

### 3.9 ANÁLISE E GERAÇÃO DE RELATÓRIOS

Após a publicação do cubo multidimensional e estar habilitado no PUC, o usuário final poderá fazer análises, gerar relatórios e criar *dashboards* que poderão subsidiar tomadas de decisões.

#### 3.9.1 Cubo censo

Uma vez que o cubo tenha sido publicado no Bi server, ele pode ser visualizado dentro do Console do Usuário(PUC), como mostrado na Figura 37.

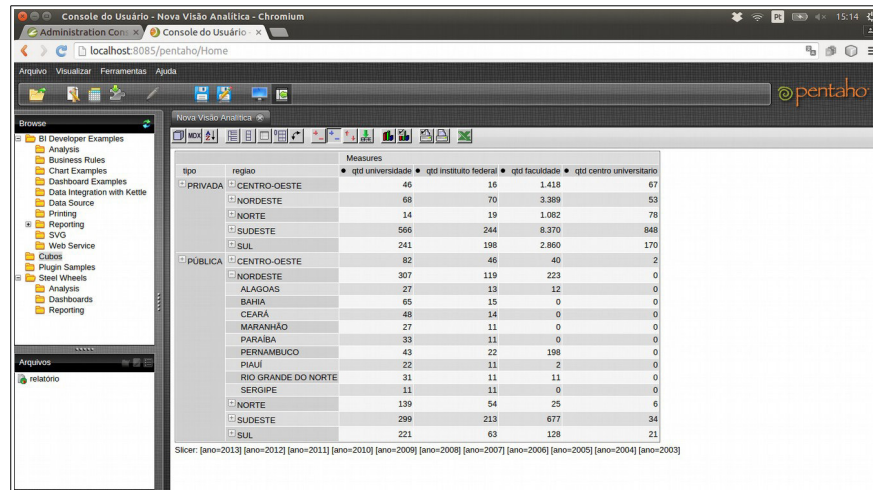


Figura 37: Visão Analítica no PUC. Fonte: Elaborado pelo autor utilizando o PUC, (2016)

Além de análises tabulares, o usuário final pode gerar gráficos, *dashboards* e relatórios. O resultado de uma consulta em forma de gráfico pode ser visualizado na Figura 38.

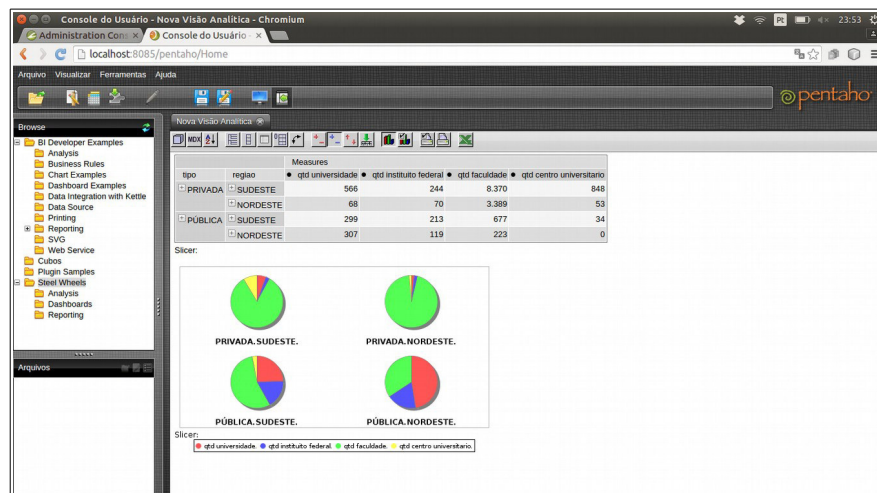


Figura 38: Exemplo de gráfico no PUC. Fonte: Elaborado pelo autor no PUC, (2016)

Outras ferramentas da suíte Pentaho se propõem a implementar interfaces para análises, como PDS (*Pentaho Design Studio*), para a confecção de como *dashboards*, ou para mineração de dados e análise estatística, como o WEKA. A escolha da ferramenta depende do escopo que ela se propõe a atender.

## 4 CONCLUSÃO

Este trabalho investigou o uso da suíte Pentaho *Community Edition* para a construção de projetos de BI. Demonstrou-se, através de um estudo de caso, que a suíte Pentaho atende todas as fases de construção de um projeto de BI. Também mostrou-se que a escolha da suíte diminui consideravelmente o custo da implantação de projetos de BI em organizações de qualquer porte, sendo uma alternativa às ferramentas proprietárias existentes no mercado.

Apesar dos últimos resultados, ainda há muito a se investigar. A suíte Pentaho CE possui muitas outras ferramentas que podem ser utilizadas, tais como, WEKA, para mineração de dados. Além da adição de bases de dados on-line, o que aproximaria de um projeto real de BI. Além disso, outras formas de visualização dos dados podem ser obtidas, tais como, *dashboards* e mineração de dados. Em relação ao BI Server, este possui controle de níveis de acesso de usuários, o que não foi trabalhado neste trabalho. O acesso se deu através de usuários-padrão, já gravados pela aplicação.

## REFERÊNCIAS

BOUMAN, Roland; DONGEN Jos van – **Pentaho Soluções: Business Intelligence e Data Armazenamento com Pentaho e Mysql** – Wiley Publishing, Inc., Indianapolis, Indiana, 2009.

BOUMAN, Roland; DONGEN Jos van – **Building Open Source ETL Solutions with Pentaho Data Integration** – Wiley Publishing, Inc., Indianapolis, Indiana, 2009.

GOLDSCHMIDT, Roland; PASSOS Emmanuel - **Data Mining: Um guia prático** - Ed. Elsevier, Rio de Janeiro, 2005.

SANTOS, Maribel Yasmina ; RAMOS, Isabel - **“Business Intelligence : Tecnologias da informação na gestão de conhecimento”**. Lisboa : FCA Editora de Informática, 2006. ISBN 972-722-405-9.

BATISTA, Emerson de Oliveira - **Sistemas de Informação : O uso consciente da tecnologia para o gerenciamento** - São Paulo: Editora Saraiva, 2004.

GORDON, Steven R.; GORDON R. Judith - **Sistema de Informação : Uma Abordagem Gerencial** - Editora LTC, 2006.

COREY, Michael et al - **Oracle 8i Data Warehouse** - Tradução de João Tortello. Rio de Janeiro: Campus, 2001.

Rezende, Denis Alcides; Abreu, Aline França de - **Tecnologia da Informação Aplicada a Sistemas de Informação Empresariais** - São Paulo: Editora Atlas, 2006.

MACHADO, Felipe Nery Rodrigues - **Tecnologia e Projeto de Data Warehouse** - 2. ed. São Paulo, SP: Érica, 2006.

LOH, Stanley - **BI na era do big data para cientistas de dados - indo além de cubos e dashboards na busca pelos porquês, explicações e padrões**. Porto Alegre, 2014.

Ceci, Flávio - **Business intelligence : livro digital / Flávio Ceci**; design instrucional Silvana Souza da Cruz Clasen. Palhoça : UnisulVirtual, 2012.

SILVA, Dhiogo Cardoso - **Uma arquitetura de business intelligence para processamento analítico baseado em tecnologias semânticas e em linguagem natural** - Florianópolis, 2011.

BALLARD Chuck, HERREMAN Dirk, SHAU Don – **Data Modeling Techniques for Data Warehousing** – Califórnia, 1998.

DILL, Sérgio Luis - **Uma metodologia para desenvolvimento de Data Warehouse e Estudo de Caso** - Dissertação submetida para obtenção do grau de Mestre em



Ciência da Computação. Universidade Federal de Santa Catarina, Florianópolis, 2002.

Auad, Arnald – **Conceitos de Business Intelligence: Guia definitivo** - São Paulo-SP, 2012.

Anzanello, Cynthia Aurora – **Artigo: OLAP Conceitos e Utilização** - Instituto de Informática, Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre – RS,

**MYSQL. JDBC Driver.** Disponível em:  
<<https://dev.mysql.com/downloads/connector/j/>>

**Mysql Workbench.** Disponível em: <<https://dev.mysql.com/downloads/workbench/>>

## APÊNDICES

## APÊNDICE 01 - Script do Data Warehouse DW\_CENSO

```

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';
-----
-- Schema DW_CENSO
-----
-----
-- Schema DW_CENSO
-----
CREATE SCHEMA IF NOT EXISTS `DW_CENSO` DEFAULT CHARACTER SET utf8 COLLATE
utf8_general_ci ;
USE `DW_CENSO` ;
-----
-- Table `DW_CENSO`.`dim_regiao`
-----
CREATE TABLE IF NOT EXISTS `DW_CENSO`.`dim_regiao` (
  `id_dim_regiao` INT NOT NULL AUTO_INCREMENT,
  `regiao` VARCHAR(45) NULL,
  `estado` VARCHAR(45) NULL,
  PRIMARY KEY (`id_dim_regiao`))
ENGINE = InnoDB;
-----
-- Table `DW_CENSO`.`dim_tempo`
-----
CREATE TABLE IF NOT EXISTS `DW_CENSO`.`dim_tempo` (
  `id_dim_tempo` INT NOT NULL AUTO_INCREMENT,
  `ano` INT NULL,
  PRIMARY KEY (`id_dim_tempo`))
ENGINE = InnoDB;
-----
-- Table `DW_CENSO`.`dim_tipo`
-----
CREATE TABLE IF NOT EXISTS `DW_CENSO`.`dim_tipo` (
  `id_dim_tipo` INT NOT NULL AUTO_INCREMENT,
  `tipo` VARCHAR(45) NULL,
  `competencia` VARCHAR(45) NULL,
  PRIMARY KEY (`id_dim_tipo`))
ENGINE = InnoDB;
-----
-- Table `DW_CENSO`.`ft_censo`
-----
CREATE TABLE IF NOT EXISTS `DW_CENSO`.`ft_censo` (
  `id_ft_censo` INT NOT NULL AUTO_INCREMENT,
  `qtd_universidade` INT NULL,
  `qtd_centro_universitario` INT NULL,
  `qtd_faculdade` INT NULL,
  `qtd_intituto_federal` INT NULL,
  `fk_dim_regiao` INT NULL,

```

```

`fk_dim_tipo` INT NULL,
`fk_dim_tempo` INT NULL,
PRIMARY KEY (`id_ft_censo`),
INDEX `fk_ft_censo_dim_regiao1_idx` (`fk_dim_regiao` ASC),
INDEX `fk_ft_censo_dim_tempo1_idx` (`fk_dim_tempo` ASC),
INDEX `fk_ft_censo_dim_tipo1_idx` (`fk_dim_tipo` ASC))
ENGINE = InnoDB;

-----
-- Table `DW_CENSO`.`stg_area`
-----
CREATE TABLE IF NOT EXISTS `DW_CENSO`.`stg_area` (
  `id_stg_area` INT NOT NULL AUTO_INCREMENT,
  `stg_regiao` VARCHAR(45) NULL,
  `stg_estado` VARCHAR(45) NULL,
  `stg_competencia` VARCHAR(45) NULL,
  `stg_tipo` VARCHAR(45) NULL,
  `stg_qtd_universidade` INT NULL,
  `stg_qtd_centro_universitario` INT NULL,
  `stg_qtd_faculdade` INT NULL,
  `stg_qtd_intstituto_federal` INT NULL,
  `stg_ano` INT NULL,
  PRIMARY KEY (`id_stg_area`))
ENGINE = InnoDB;

      SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

## APÊNDICE 02 – cubo\_censo.xml

```

<Schema name="schema_censo">
  <Cube name="censo" visible="true" cache="true" enabled="true">
    <Table name="ft_censo">
    </Table>
    <Dimension type="StandardDimension" visible="true" foreignKey="fk_dim_tempo"
highCardinality="false" name="tempo">
      <Hierarchy visible="true" hasAll="true">
        <Table name="dim_tempo">
        </Table>
        <Level name="ano" visible="true" column="ano" type="Integer" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
          </Level>
        </Hierarchy>
      </Dimension>
      <Dimension type="StandardDimension" visible="true" foreignKey="fk_dim_tipo"
highCardinality="false" name="tipo">
        <Hierarchy visible="true" hasAll="true" primaryKey="id_dim_tipo">
          <Table name="dim_tipo">
          </Table>
          <Level name="tipo" visible="true" column="id_dim_tipo" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
            </Level>
            <Level name="competencia" visible="true" column="competencia" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
              </Level>
            </Hierarchy>
          </Dimension>
          <Dimension type="StandardDimension" visible="true" foreignKey="fk_dim_regiao"
highCardinality="false" name="regiao">
            <Hierarchy visible="true" hasAll="true" primaryKey="id_dim_regiao">
              <Table name="dim_regiao">
              </Table>
              <Level name="regiao" visible="true" column="regiao" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
                </Level>
                <Level name="Estado" visible="true" column="estado" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
                  </Level>
                </Hierarchy>
              </Dimension>
              <Measure name="qtd universidade" column="qtd_universidade" datatype="Integer"
aggregator="sum" visible="true">
                </Measure>
                <Measure name="qtd centro universitario" column="qtd_centro_universitario" datatype="Integer"
aggregator="sum" visible="true">
                </Measure>
                <Measure name="qtd faculdade" column="qtd_faculdade" datatype="Integer" aggregator="sum"
visible="true">
                </Measure>

```

```
<Measure name="qtd instituito federal" column="qtd_intituto_federal" datatype="Integer"
aggregator="sum" visible="true">
  </Measure>
</Cube>
</Schema>
```