UNIVERSIDADE FEDERAL DE SANTA CATARINA

CENTRO TECNOLÓGICO

DEPTO. DE INFORMÁTICA E ESTATÍSTICA

CURSO DE SISTEMAS DE INFORMAÇÃO

DATA WAREHOUSE

PROF. JOSÉ LEOMAR TODESCO

Trabalho Final de Data Warehouse

Data Mart do Vestibular da UFSC de 2010 a 2012

Emerson Demetrio Plácido

Florianópolis, 06 de Dezembro de 2015

1 RESUMO

Este trabalho visa analisar os resultados obtidos nos concursos vestibulares realizados pela Universidade Federal de Santa Catarina - UFSC nos anos de 2010, 2011 e 2012, assume-se um papel de uma empresa de recrutamento interessada em buscar talentos nas áreas da engenharia, física e química. Para tanto, são feitas analises de desempenho individual dos candidatos, principalmente nas áreas exatas (matemática, física e química), a fim de financiar bolsas de estudo em universidades estrangeiras. O escopo se limitará aos alunos que tiveram melhor desempenho nos anos de 2010, 2011 e 2012. Serão feitos também filtros por renda e localização, origem do candidato e a renda per capita de sua família. Os cursos de interesse são física (bacharelado), química, engenharia química e engenharia de materiais.

Palavras-chave: vestibular, UFSC, data warehouse, data mart, formulário socioeconômico, Universidade Federal de Santa Catarina, Comissão Permanente do Vestibular, COPERVE.

2 INTRODUÇÃO

Atualmente, as pessoas que pretendem cursar alguma faculdade na Universidade Federal de Santa Catarina precisam realizar o UFSC/COPERVE. A concorrência nos cursos de engenharia é extremamente alta e além disso, as disciplinas exatas sempre são barreiras para os candidatos.

Este trabalho tem por objetivo buscar a relação de alunos que se saíram bem nas disciplinas de exatas, tais como química, física, matemática e, além disso, língua estrangeira (especialmente, inglês). Uma vez que serão ofertadas bolsas de estudo aos candidatos que foram melhor nas disciplinas acima, será feita uma análise no banco de dados oferecido pela disciplina afim de filtrar os resultados. A seguir, mostra-se o passo a passo para a construção do *data-mart* que será utilizado para transformar os dados da COPERVE em informações, abordando as etapas de planejamento, projeto, modelagem dimensional, perguntas estratégicas, ETL e front-end.

3 OBTENÇÃO DOS DADOS

Os dados utilizados neste trabalho foram disponibilizados pelo professor da disciplina em parceria com a COPERVE/UFSC. É importante ressalvar que os eventos analisados serão aqueles que aconteceram nos anos de 2010, 2011 e 2012. Os dados do vestibular são obtidos através de preenchimento de um cadastro socioeconômico por parte dos alunos, no momento da inscrição. Os formulários não interferem na classificação/admissão dos candidatos. O objetivo principal destes formulários é a compreensão dos aspectos socioeconômico e culturais dos candidatos.

4 MODELO RELACIONAL (COPERVE)

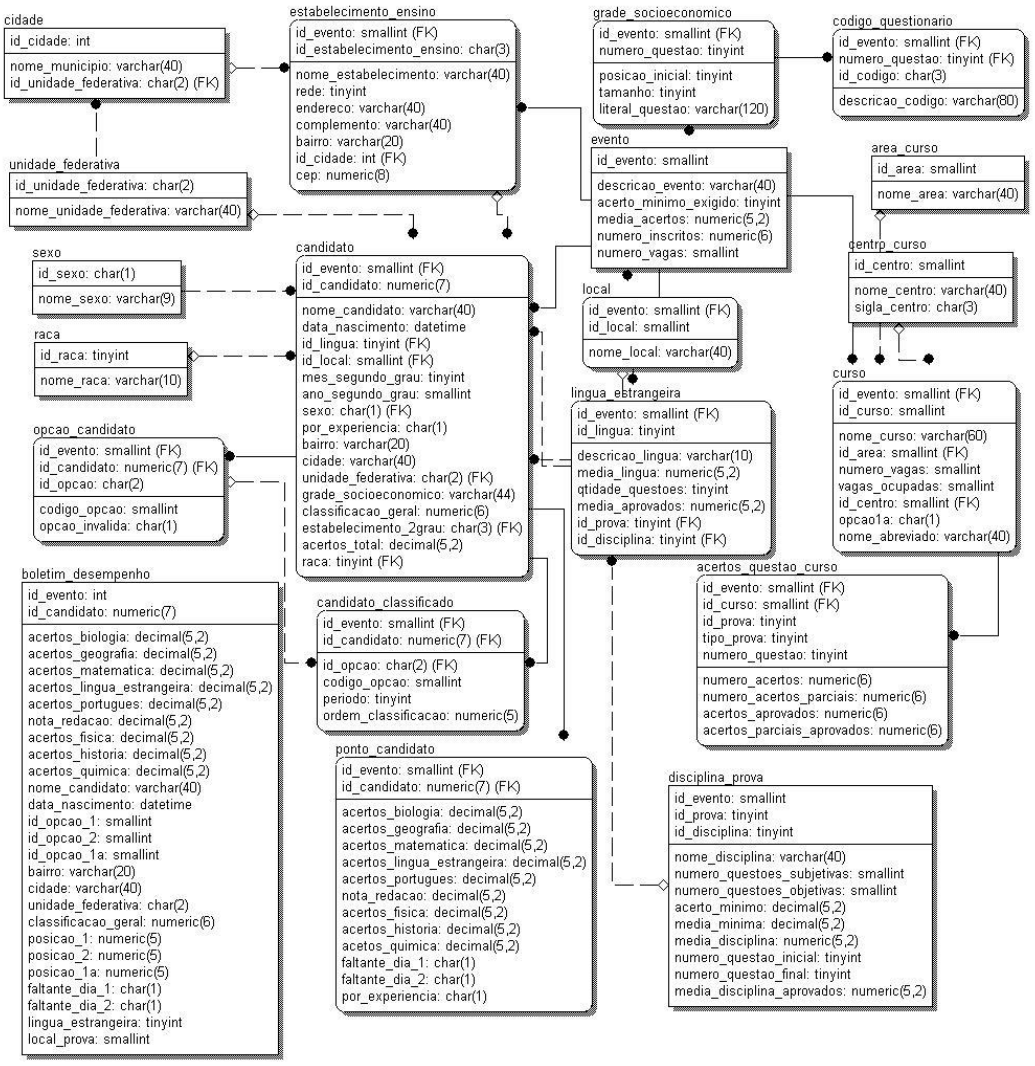

Figura 1 – Modelagem do Banco de Dados Vestibular

5 MÉTODOS

Para cruzamento dos dados segundo a especificação definida, optou-se por utilizar um *data-mart*, (um nodo de um projeto *data wharehouse*), buscando uma base robusta e confiável para obtenção dos dados e para análise de resultados. O ferramental e as técnicas utilizadas são descritas em seguida.

6 METODOLOGIA

Para construção do data-mart, são necessários os seguintes passos:

1. Planejamento do projeto
2. Administração do projeto
3. Definição dos requisitos de negócio
4. Modelagem dimensional
5. Projeto físico
6. Desenvolvimento e projeto da área de transição
7. Especificação da aplicação do usuário final
8. Desenvolvimento da aplicação do usuário final
9. Projeto e arquitetura técnica
10. Instalação e seleção de produtos
11. Implantação e manutenção

Este trabalho foca nas etapas de planejamento, análise, modelagem dimensional, projeto físico, ETL e front-end.

6.1 PLANEJAMENTO

Escopo: traçar um perfil dos candidatos dos vestibulares de 2010 a 2012 da UFSC buscando talentos nas áreas exatas (matemática, física e química) e como eles dividem-se dentro das opções oferecidas pela UFSC.

Justificativa: O trabalho servirá para gerar informações relacionadas ao perfil do candidato, a fim de premiar aqueles que melhor se saíram com bolsas de estudo.

Exclusões: Candidatos que prestaram o vestibular por experiência; Todas as avaliações dos anos que não estão na relação.

Riscos: Dados fornecidos podem apresentar inconsistências ou defeitos.

Fatores críticos para o sucesso: desenvolver um *data-mart* que responda as perguntas propostas abaixo.

6.2 ANÁLISES E PERGUNTAS

Com o objetivo de gerar informações sobre os candidatos, instituições de ensino e desempenho no vestibular, foram levantadas as seguintes questões para análise:

1. Quem são os candidatos que melhor se saíram nas disciplinas de física, química, matemática e inglês, juntas?
2. Quem são os candidatos que melhor se saíram nas disciplinas de física, química, matemática e inglês, separadamente?
3. Qual é o percentual de acertos dos candidatos nas áreas exatas observando origem socioeconômica?
4. Qual é a média de nota das disciplinas das áreas exatas por origem socioeconômica?
5. Qual é a renda familiar dos candidatos que melhor se saíram nas áreas exatas?

6.3 MODELAGEM DIMENSIONAL

A modelagem dimensional foi definida conforme a especificação da disciplina, sendo:

Definição do Processo de negócio:
O projeto teve como objetivo analisar dados de desempenho e origem socioeconômica dos candidatos.

Definição do grão:
Para poder responder às nossas perguntas estratégicas, foi decidido como grão o desempenho anual de cada candidato em cada matéria.

Definição do fato:
O fato “ft_desempenho”, nele é descriminado cada desempenho em cada matéria, e nota total. Também contém uma chave estrangeira (fk) para relacionamento com as dimensões.

Definição das dimensões:

O modelo proposto conta com cinco dimensões:

- **Dimensão Candidato (dm_evento)**

  Contém dados sobre o evento vestibular de cada ano.

- **Dimensão Curso (dm_curso)**

  Contém as informação sobre o curso escolhido pelo vestibulando.

- **Dimensão Socioeconômico (dm_socioeconomico)**

  Contém dados sobre os cadastros socioeconômicos dos candidatos.

- **Dimensão Candidato (dm_candidato):**

  Contém as representações das informações gerais de um candidato.

- **Dimensão Curso (dm_regiao)**

  Contém as informação sobre o curso escolhido pelo vestibulando.

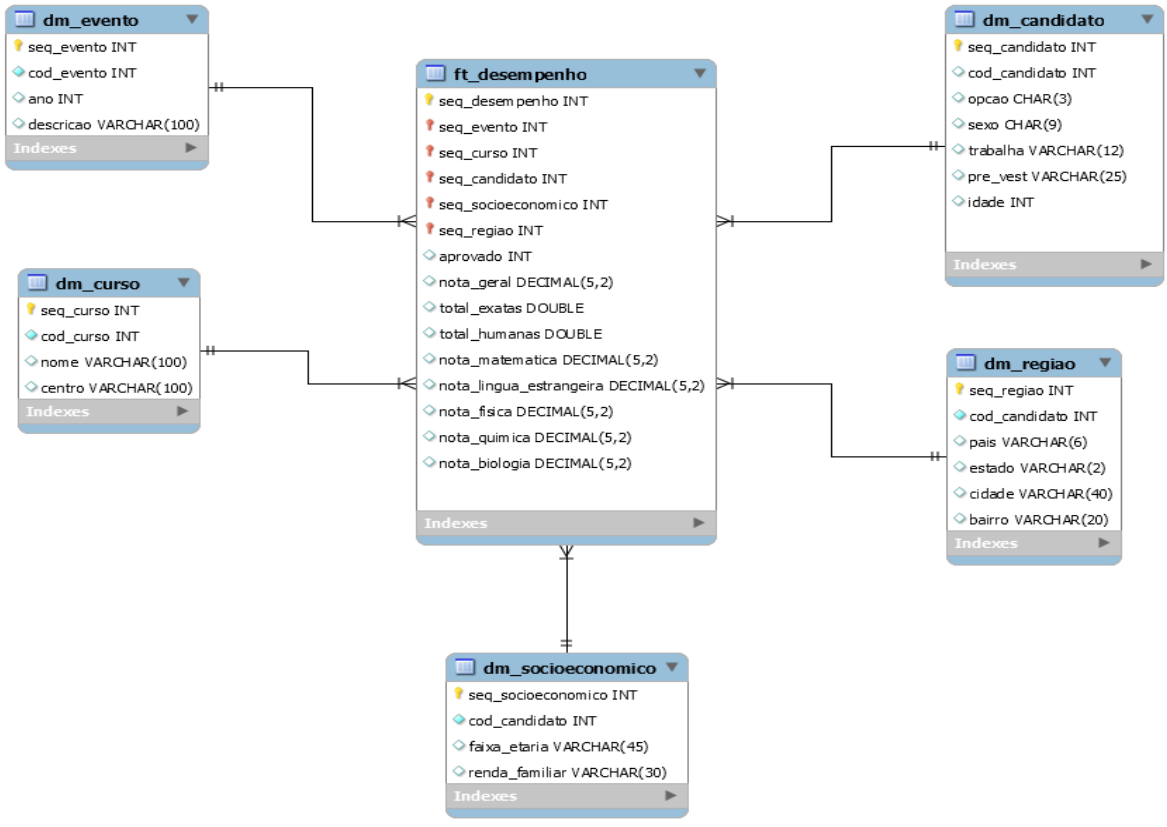

Figura 2 – Modelagem Dimensional

7 PROJETO FÍSICO

O projeto físico foi implementando no bando de dados *MySql*. O *Script* de criação está disponível no anexo 1.

7.1 FERRAMENTAS UTILIZADAS

Para modelagem física, dimensional, ETL, back-end e front-end foram utilizadas as seguintes ferramentas:

Modelagens: Software MySQL Workbench

Carregamento do projeto físico, avaliação e validação da modelagem física e dimensional:

Software Navicat Premium

Extração, *back-end*, transformações e carga de dados: Spoon(Pentaho), *Java script*

Front-End: Software Tableau

As seguintes dimensões e fatos foram criadas para o *data-mart*:

Dimensão Evento:

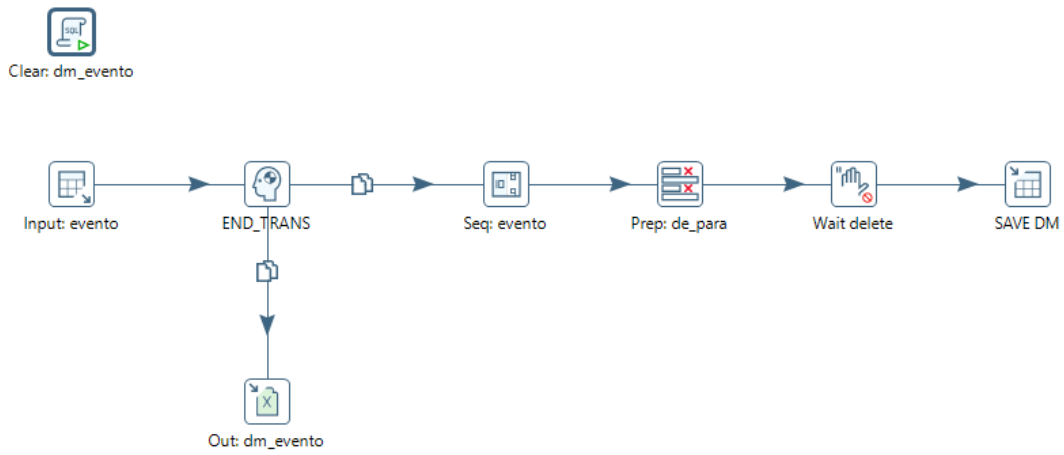

Figura 3 – Dimensão Evento

Nesta dimensão, são carregados os eventos cujos anos interessam na modelaegem (2009, 2010 e 2011).

Dimensão Curso:

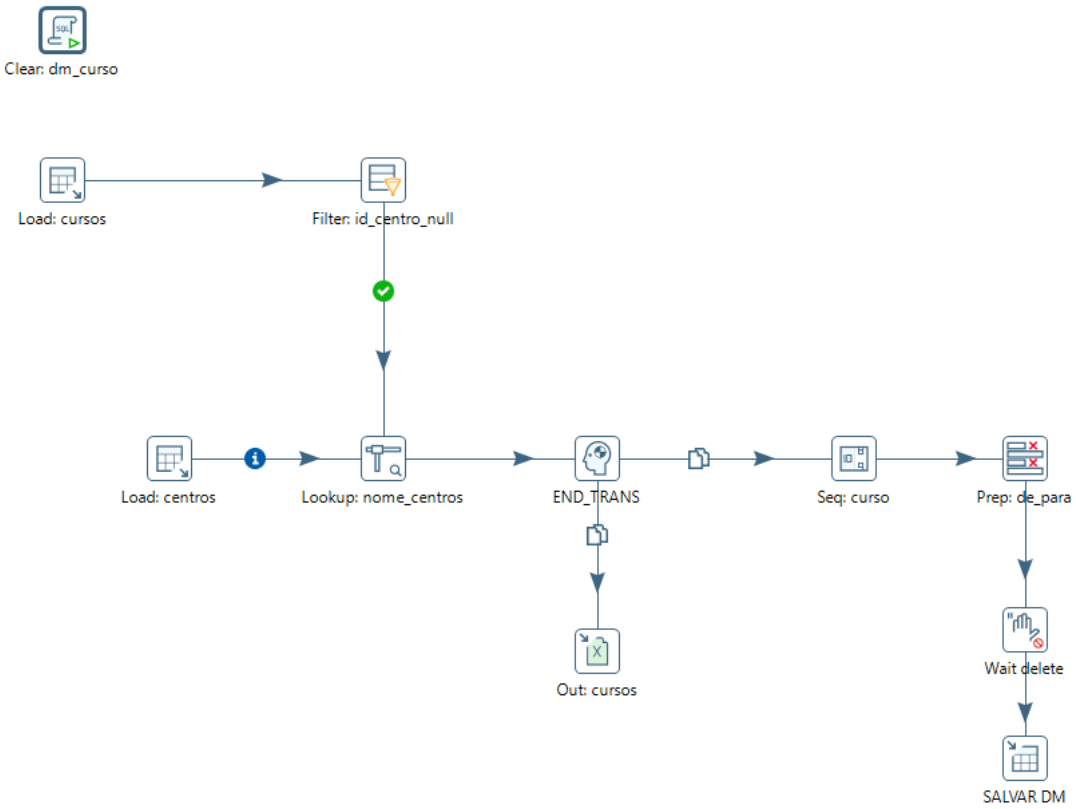

Figura 4 – Dimensão Curso

Esta dimensão é responsável por obter junto ao banco de dados original (no projeto, chamado de “DB_VESTIBULAR”) os cursos que interessam para analise. É possível observar na figura o padrão de nomenclatura implementado em cada passo, sendo “Load” um evento de consulta, “Lookup” um passo aonde se faz uma procura por valores, etc.

Dimensão Região

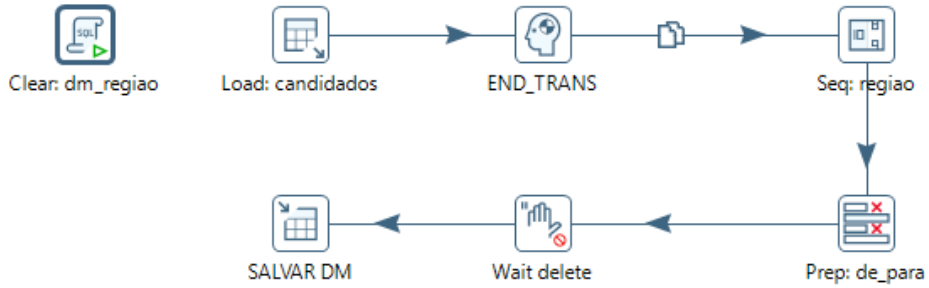

Figura 5 – Dimensão Região

Nesta Dimensão são carregados as regiões de onde vem os candidatos (cidade, estado, etc).

Dimensão Socioeconômico

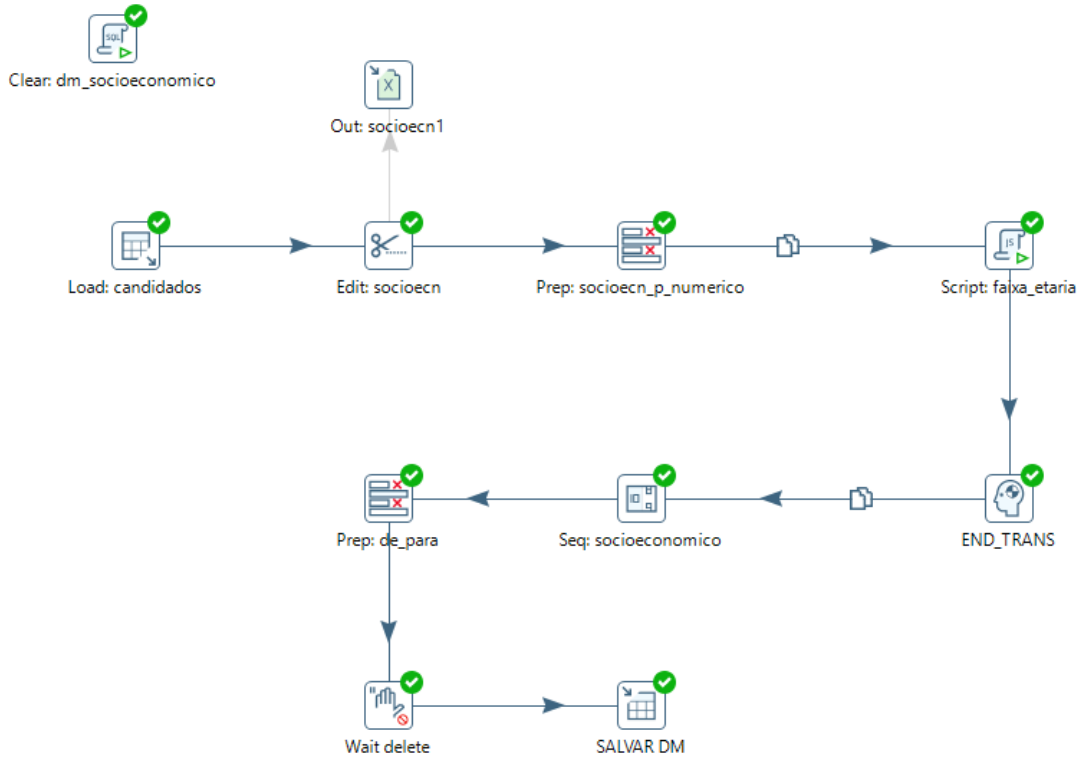

Figura 6 – Dimensão Socioeconômico

Nesta Dimensão são carregados faixa-etária e renda familiar dos candidatos. É interessante notar que para "calcular" a faixa etária foram necessários dois passos, sendo um deles implícito: Num primeiro momento, calcula-se a idade via *SQL* (implícito no passo *Load*: candidatos) e num segundo momento, há uma função escrita em Java script que traz a faixa-etária baseada no critério: Até 18 anos: Adolescente, Acima de 18 anos e menor do que 60 anos: Adulto, acima de 60 anos: Terceira Idade.

Dimensão Candidato:

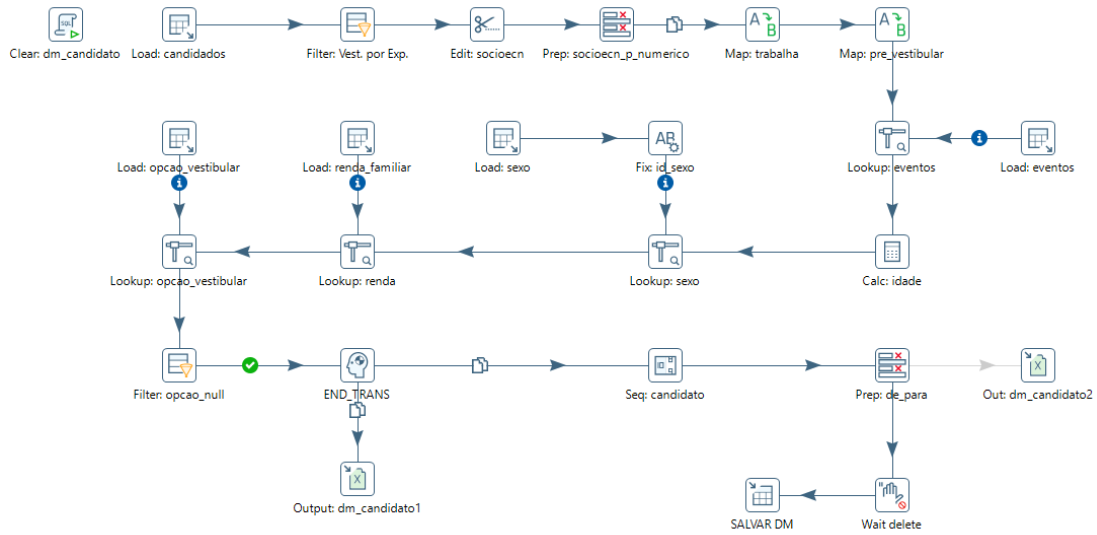

Figura 7 – Dimensão Candidato

Nesta dimensão são carregados os dados do candidato. São realizados alguns filtros, quais sejam: Se fez vestibular por experiência, se trabalha, sexo e remoção de resultados *null*.

Fatos Desempenho

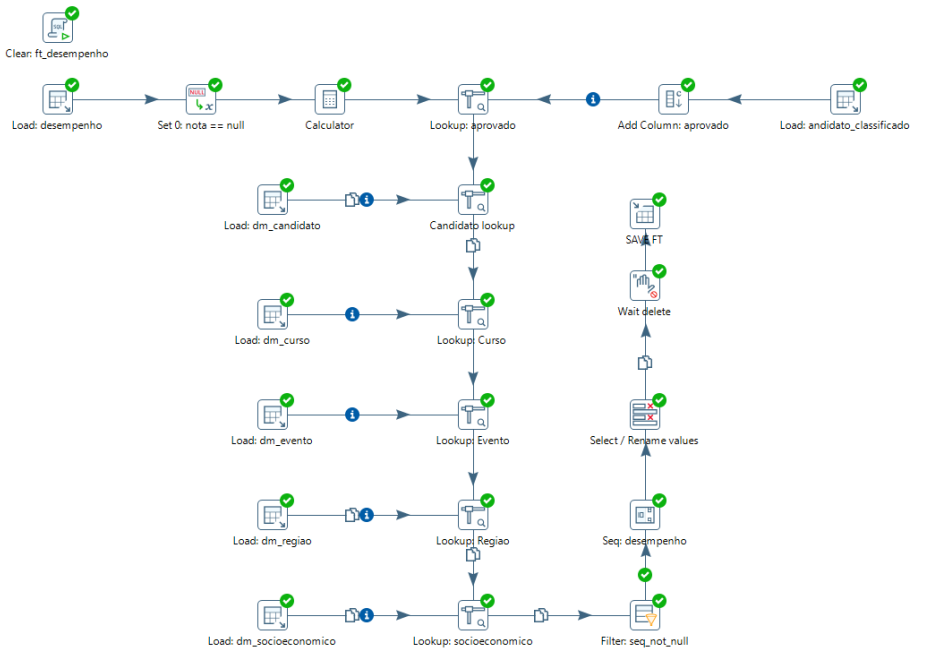

Figura 8 – Fato Desempenho

NA tabela de fatos “Desempenho” é possível observar o resultado de toda a analise sendo feita. Esta é a parte mais normatizada da modelagem, sendo que diversos campos são apenas chaves-estrangeiras para outras tabelas. Observa-se a presença de diversos *Lookup’s*, cuja finalidade é justamente buscar os valores em outras tabelas. É importante ressaltar que esta foi a tabela cuja concepção requereu um enorme esforço para ser concluída, tendo em vista as diversas restrições de integridade que se fizeram necessárias e que, por diversas vezes, foram quebradas por dados ausentes no banco original.

8 FRONT-END

Para o front-end do projeto, diversas ferramentas foram experimentadas mas apenas uma trouxe um resultado satisfatório, o software *Tableau.*

Para a concepção do trabalho, foi criado um novo “livro”, como é chamado internamente a pasta de trabalho dentro do software, conforme figura abaixo:

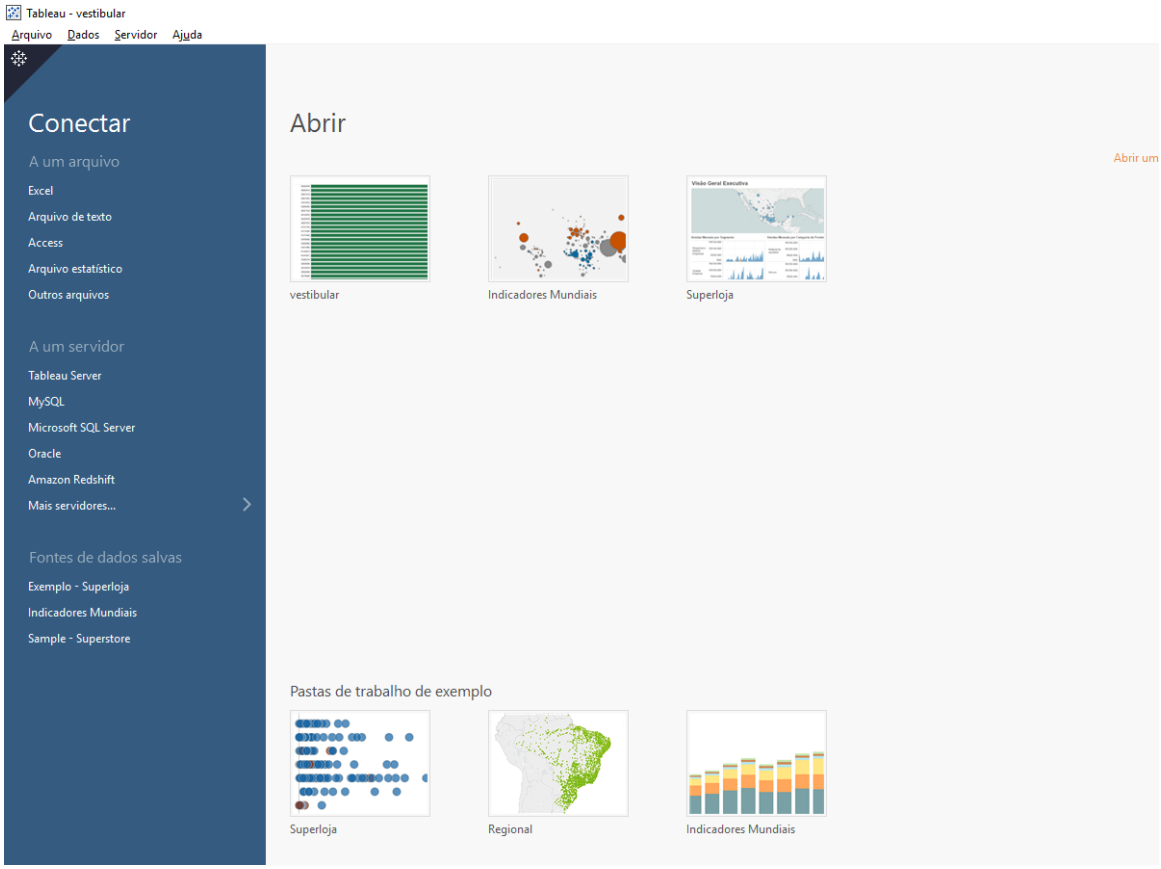

Figura 8 – Livro “vestibular” – Tableau

Para construção dos resultados e as analises, observa-se que o Tableau é um software de fácil acesso e de manipulação amigável. Obviamente que com suas peculiaridades. Para concepção do trabalho, foram necessárias algumas horas de leitura a respeito do Tableau.

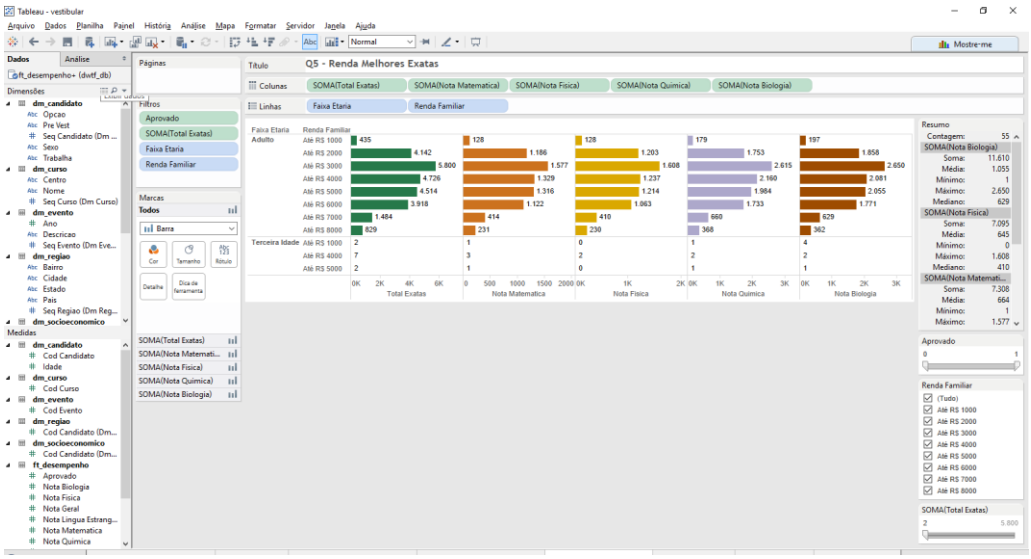

Figura 9 – Resultado da análise realizada no Tableau

9 RESULTADOS

Para responder as perguntas definidas na metodologia, foram realizadas diversas consultas nos dados extraídos através do Tableau.

Cada uma delas respondida individualmente, conforme imagens abaixo:

Questão 1 - Quem são os candidatos que melhor se saíram nas disciplinas de física, química, matemática e inglês, juntas?

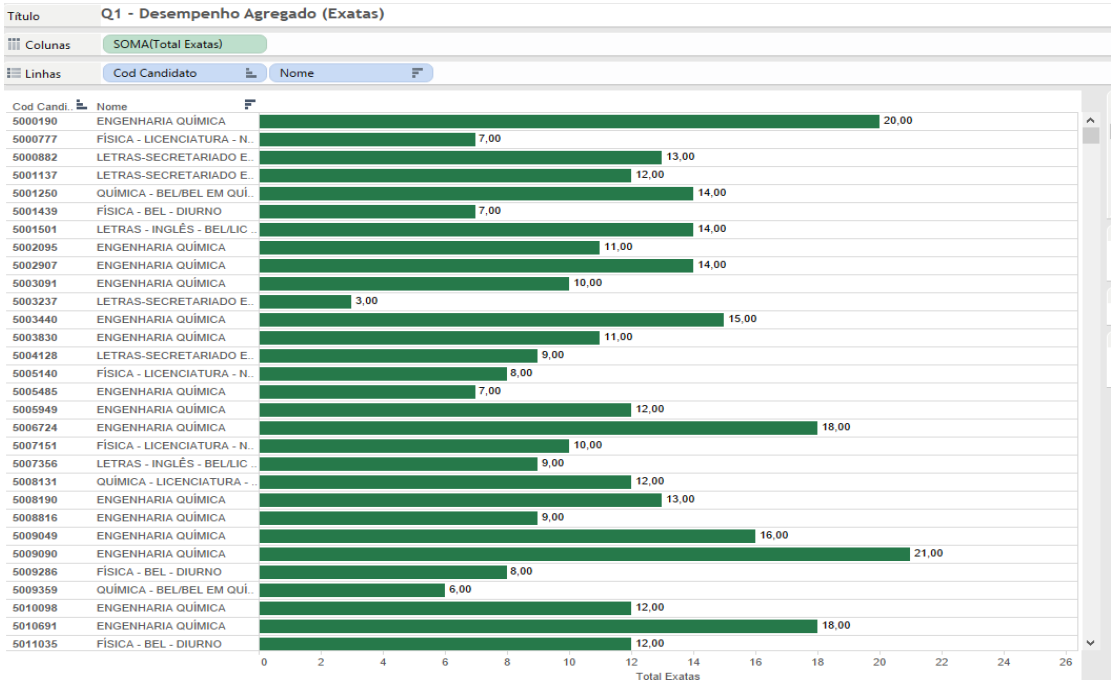

Figura 10 - Questão 1

Com o software tableau, é possível realizar filtragens dinâmicas, baseado num critério definido pelo usuário, entretanto, para esta questão, é possível observar que os candidatos que melhor se saíram foram os de Engenharia Química.

Questão 2 - Quem são os candidatos que melhor se saíram nas disciplinas de física, química, matemática e inglês, separadamente?

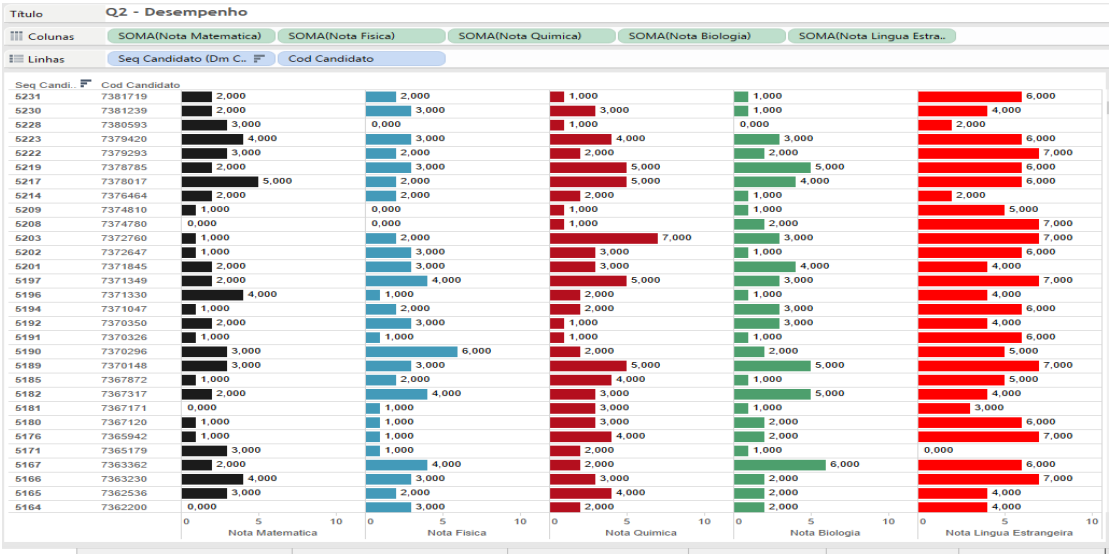

Figura 11 - Questão 2

Nesta análise, é possível observar uma relação direta entre os resultados obtidos em matérias exatas, ou seja, se um candidato vai bem em física, é provável que ele vá bem também em química, por exemplo.

Questão 3 - Qual é o percentual de acertos dos candidatos nas áreas exatas observando origem socioeconômica?

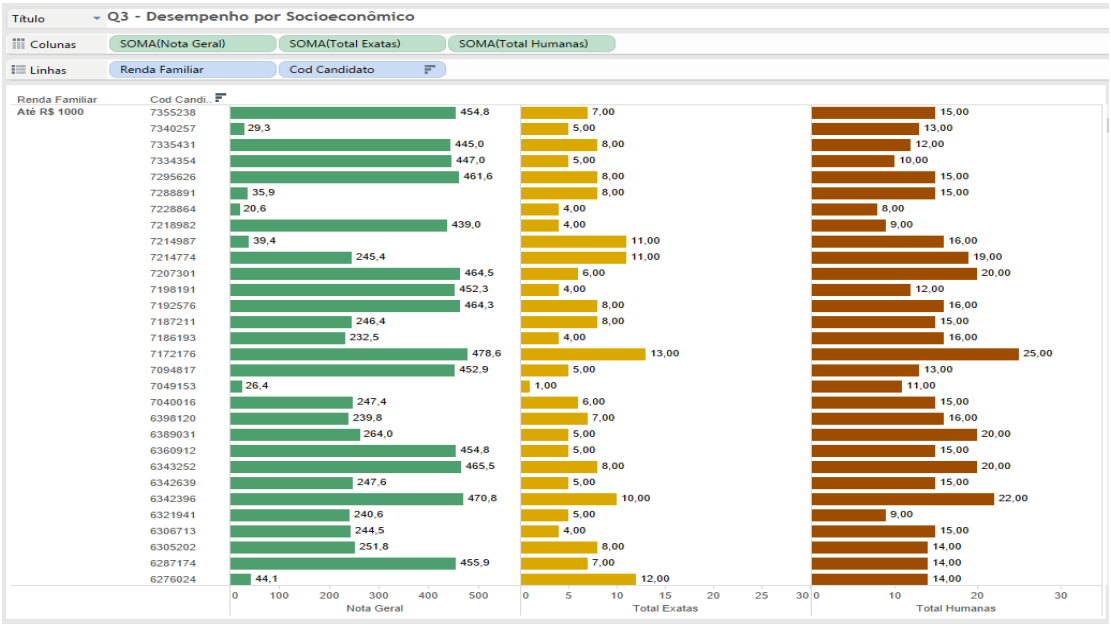

Figura 12 - Questão 3

Para esta resposta, foram utilizados campos da dimensão socioeconômico, observando o desempenho resumido dos candidatos baseado na renda familiar.

Questão 4 - Qual é a média de nota das disciplinas das áreas exatas por origem socioeconômica?

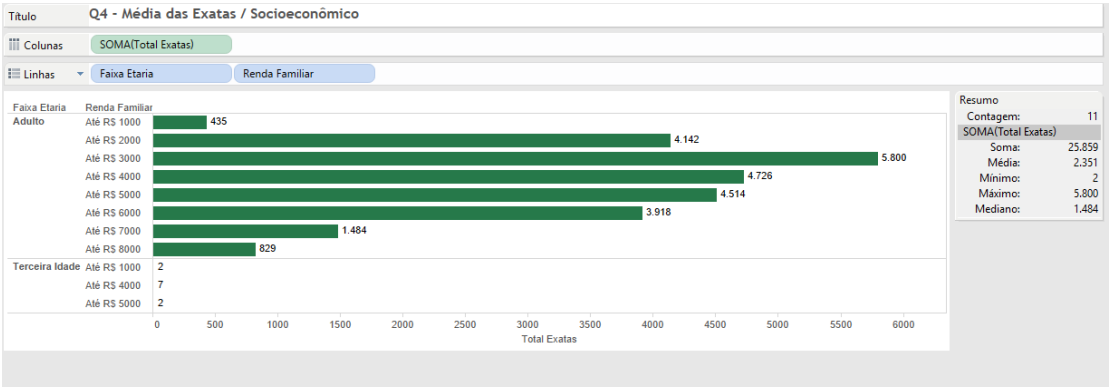

Figura 13 - Questão 4

Para esta análise, foram necessários dados de desempenho e dados de renda, é possível observar que o melhor resultado vem dos candidatos cuja renda familiar é de até 3000 reais, observando também a distribuição dos demais resultados.

Questão 5 - Qual é a renda familiar dos candidatos que melhor se saíram nas áreas exatas?

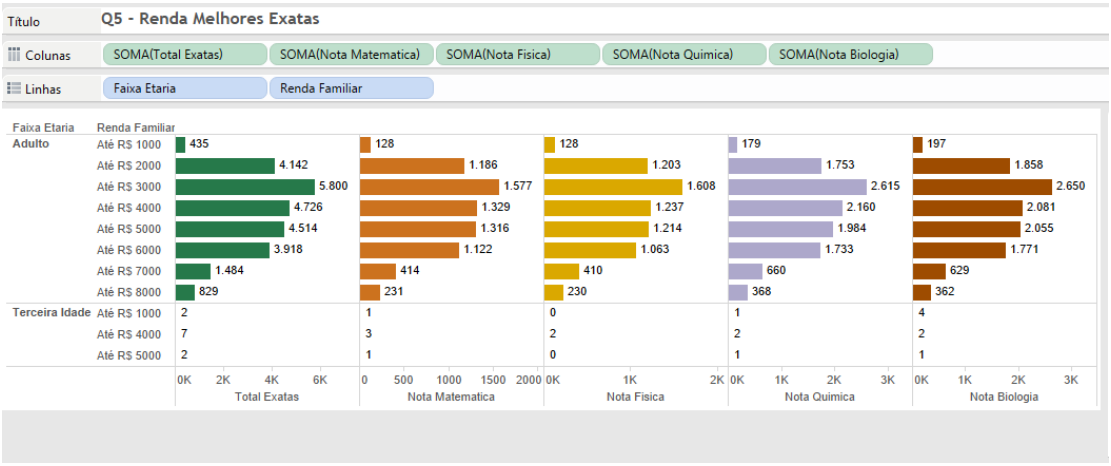

Fig. 14 - Questão 5

Esta questão trata das melhores notas, baseado apenas na renda familiar. É possível ver que a correlação com a pergunta anterior é mantida, ou seja, as famílias com renda até 3000 reais ainda se mantém como provedoras dos candidatos cujo resultado se destaca.

9.1 DESAFIOS E DIFICULDADES ENCONTRADAS

Para concepção do trabalho foram necessárias horas de dedicação e estudo das ferramentas. Embora houvesse domínio da maioria das ferramentas, os softwares Spoon e Tableau trouxeram novos desafios. Num primeiro momento, foi necessário configurar o Spoon para que ele se conectasse com o banco de dados MySql, o que em teoria deveria ser automático, entretanto, não foi. Para isso, foi necessário acrescentar um driver de conexão, um arquivo *jar* fornecido pela *Oracle*, listado nas referências deste trabalho. A ferramenta Spoon é amigável, entretanto, é difícil de encontrar os erros, quando acontecem, mesmo quando executando no modo *debug*, ao final dos procedimentos, entretanto, o resultado é satisfatório, observando o modo como as tabelas são preenchidas automaticamente e respeitando as regras estabelecidas pelo usuário.

Ao partir para o front-end, observou-se também o mesmo problema de driver para Mysql, sendo necessário instalar uma outra versão, desta vez para o sistema operacional Windows 10. Dentro da ferramenta, é possível realizar diversas analises baseado nos mais diversos critérios, também é interessante ver como as chaves estrangeiras são conectadas automaticamente, bastando apenas arrastar e soltar os elementos que representam as tabelas.

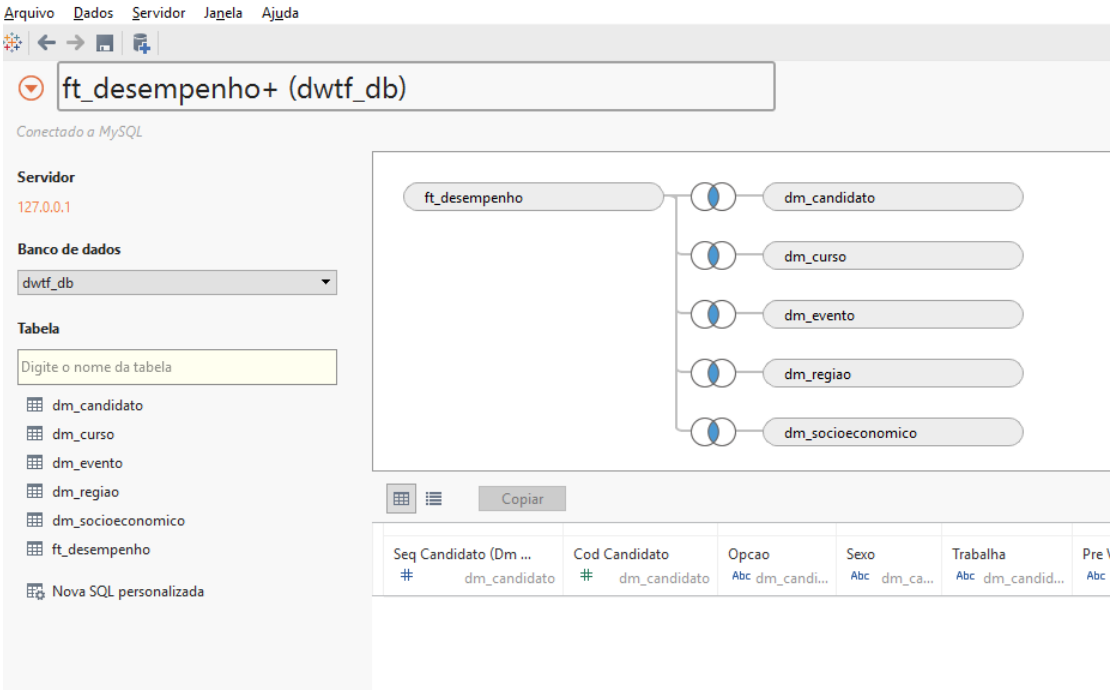

Figura 15 – Software Tableau

10 VERSIONAMENTO

Embora não fosse parte da especificação do trabalho, foi criado um repositório com as diversas versões do trabalho, assim como a sua evolução, permitindo assim um acompanhamento do mesmo. Como trata-se de um trabalho acadêmico, o acesso é público, disponível no endereço a seguir:

< https://github.com/emersondemetrio/20152dw >

11 CONCLUSÕES

Este trabalho serviu para concretizar diversos conhecimentos aprendidos de maneira teórica nas disciplinas de Banco de Dados e Data Wharehouse do curso de Sistemas de Informação da Universidade Federal de Santa Catarina.

Num primeiro momento, houve um impulso de fazer as análises via sql puro, por conta da incerteza a respeito do entendimento tanto do software Spoon quanto do software Tableau, mas ao decorrer dos trabalhos, com o lento, porém constante entendimento dos aspectos de cada sistema, passou a existir um interesse por ambos os softwares e o resultado das consultas trouxe um entendimento maior do porquê das dimensões e fatos serem do jeito que são.

Considera-se este um trabalho valioso para a formação dos envolvidos.

12 REFERÊNCIAS

Ferramentas:

Tableau – Front End Software

Download:
< http://www.tableau.com/pt-br> (Acesso: 06/12/2015)

Aulas e exemplos (Acesso: 06/12/2015)

- < http://onlinehelp.tableau.com/current/pro/online/windows/en-us/buildexamples_line.html>
- <https://www.youtube.com/watch?v=pXYgsd9xOZI>
- <https://www.youtube.com/watch?v=x56ipAMMmLA>
- <https://www.youtube.com/watch?v=6BWPoccQatI>

Mysql Workbench – Modelagem
< https://www.mysql.com/products/workbench/> (Acesso: 06/12/2015)

Navicat – Projeto Físico, validações, consultas, etc
< http://www.navicat.com> (Acesso: 06/12/2015)

Drivers

Mysql para Spoon
< https://dev.mysql.com/downloads/connector/j/5.0.html> (Acesso: 06/12/2015)

Driver MySql ODBC
<https://dev.mysql.com/downloads/connector/odbc> (Acesso: 06/12/2015)

13 ANEXOS

Anexo 1 – Script de Criação do projeto físico:

```
-- MySQL Script generated by MySQL Workbench
-- 12/06/15 17:55:58
-- Model: New Model    Version: 1.0
-- MySQL Workbench Forward Engineering

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 dwtf_db
-- -----------------------------------------------------
-- Trabalho Final - Data Warehouse UFSC

-- -----------------------------------------------------
-- Schema dwtf_db
--
-- Trabalho Final - Data Warehouse UFSC
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `dwtf_db` DEFAULT CHARACTER SET utf8 COLLATE
utf8_general_ci ;
USE `dwtf_db` ;

-- -----------------------------------------------------
-- Table `dwtf_db`.`dm_evento`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `dwtf_db`.`dm_evento` (
  `seq_evento` INT NOT NULL COMMENT '',
  `cod_evento` INT NOT NULL COMMENT '',
  `ano` INT NULL COMMENT '',
  `descricao` VARCHAR(100) NULL COMMENT '',
  PRIMARY KEY (`seq_evento`)  COMMENT '')
ENGINE = InnoDB;



-- -----------------------------------------------------
-- Table `dwtf_db`.`dm_candidato`
```

```
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `dwtf_db`.`dm_candidato` (
  `seq_candidato` INT NOT NULL COMMENT '',
  `cod_candidato` INT NULL COMMENT '',
  `opcao` CHAR(3) NULL COMMENT '',
  `sexo` CHAR(9) NULL COMMENT '',
  `trabalha` VARCHAR(12) NULL COMMENT '',
  `pre_vest` VARCHAR(25) NULL COMMENT '',
  `idade` INT NULL COMMENT '',
  PRIMARY KEY (`seq_candidato`)  COMMENT '')
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `dwtf_db`.`dm_curso`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `dwtf_db`.`dm_curso` (
  `seq_curso` INT NOT NULL COMMENT '',
  `cod_curso` INT NOT NULL COMMENT '',
  `nome` VARCHAR(100) NULL COMMENT '',
  `centro` VARCHAR(100) NULL COMMENT '',
  PRIMARY KEY (`seq_curso`)  COMMENT '')
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `dwtf_db`.`dm_socioeconomico`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `dwtf_db`.`dm_socioeconomico` (
  `seq_socioeconomico` INT NOT NULL COMMENT '',
  `cod_candidato` INT NOT NULL COMMENT '',
  `faixa_etaria` VARCHAR(45) NULL COMMENT '',
  `renda_familiar` VARCHAR(30) NULL COMMENT '',
  PRIMARY KEY (`seq_socioeconomico`)  COMMENT '')
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `dwtf_db`.`dm_regiao`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `dwtf_db`.`dm_regiao` (
```

```
  `seq_regiao` INT NOT NULL COMMENT '',
  `cod_candidato` INT NOT NULL COMMENT '',
  `pais` VARCHAR(6) NULL COMMENT '',
  `estado` VARCHAR(2) NULL COMMENT '',
  `cidade` VARCHAR(40) NULL COMMENT '',
  `bairro` VARCHAR(20) NULL COMMENT '',
  PRIMARY KEY (`seq_regiao`)  COMMENT '')
ENGINE = InnoDB;



-- -----------------------------------------------------
-- Table `dwtf_db`.`ft_desempenho`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `dwtf_db`.`ft_desempenho` (
  `seq_desempenho` INT NOT NULL COMMENT '',
  `seq_evento` INT NOT NULL COMMENT '',
  `seq_curso` INT NOT NULL COMMENT '',
  `seq_candidato` INT NOT NULL COMMENT '',
  `seq_socioeconomico` INT NOT NULL COMMENT '',
  `seq_regiao` INT NOT NULL COMMENT '',
  `aprovado` INT NULL COMMENT '',
  `nota_geral` DECIMAL(5,2) NULL COMMENT '',
  `total_exatas` DOUBLE NULL COMMENT '',
  `total_humanas` DOUBLE NULL COMMENT '',
  `nota_matematica` DECIMAL(5,2) NULL COMMENT '',
  `nota_lingua_estrangeira` DECIMAL(5,2) NULL COMMENT '',
  `nota_fisica` DECIMAL(5,2) NULL COMMENT '',
  `nota_quimica` DECIMAL(5,2) NULL COMMENT '',
  `nota_biologia` DECIMAL(5,2) NULL COMMENT '',
  PRIMARY KEY (`seq_desempenho`, `seq_evento`, `seq_curso`, `seq_candidato`,
`seq_socioeconomico`, `seq_regiao`)  COMMENT '',
  INDEX `fk_ft_desempenho_dm_evento_idx` (`seq_evento` ASC)  COMMENT '',
  INDEX `fk_ft_desempenho_dm_curso1_idx` (`seq_curso` ASC)  COMMENT '',
  INDEX `fk_ft_desempenho_dm_candidato1_idx` (`seq_candidato` ASC)  COMMENT '',
  INDEX `fk_ft_desempenho_dm_socioeconomico1_idx` (`seq_socioeconomico` ASC)
COMMENT '',
  INDEX `fk_ft_desempenho_dm_regiao1_idx` (`seq_regiao` ASC)  COMMENT '',
  CONSTRAINT `fk_ft_desempenho_dm_evento`
    FOREIGN KEY (`seq_evento`)
    REFERENCES `dwtf_db`.`dm_evento` (`seq_evento`)
    ON DELETE NO ACTION
```

```
  ON UPDATE NO ACTION,
 CONSTRAINT `fk_ft_desempenho_dm_curso1`
  FOREIGN KEY (`seq_curso`)
  REFERENCES `dwtf_db`.`dm_curso` (`seq_curso`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
 CONSTRAINT `fk_ft_desempenho_dm_candidato1`
  FOREIGN KEY (`seq_candidato`)
  REFERENCES `dwtf_db`.`dm_candidato` (`seq_candidato`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
 CONSTRAINT `fk_ft_desempenho_dm_socioeconomico1`
  FOREIGN KEY (`seq_socioeconomico`)
  REFERENCES `dwtf_db`.`dm_socioeconomico` (`seq_socioeconomico`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
 CONSTRAINT `fk_ft_desempenho_dm_regiao1`
  FOREIGN KEY (`seq_regiao`)
  REFERENCES `dwtf_db`.`dm_regiao` (`seq_regiao`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;


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