



Universidade Federal do Estado do Rio de Janeiro
Centro de Ciências Exatas e Tecnologia
Escola de Informática Aplicada

Análise de Dados Abertos Governamentais usando Técnicas de Business Intelligence:
um Estudo de Caso das Eleições 2014

Jéssica Câmara de Almeida Veiga
Julio Cesar Banharo Guimarães

Orientador
Astério Kiyoshi Tanaka

Rio de Janeiro, RJ – Brasil

MAIO de 2015

Análise de Dados Abertos Governamentais usando Técnicas de Business Intelligence:
um Estudo de Caso das Eleições 2014

Jéssica Câmara de Almeida Veiga

Julio Cesar Banharo Guimarães

Projeto de Graduação apresentado à Escola de
Informática Aplicada da Universidade Federal do
Estado do Rio de Janeiro (UNIRIO) para obtenção
do título de Bacharel em Sistemas de Informação.

Aprovada por:

Prof. Asterio Tanaka D.Sc (UNIRIO)

Prof. Leonardo Azevedo D.Sc (UNIRIO; IBM Research
Brasil)

Prof. Márcio Barros D.Sc (UNIRIO)

Rio de Janeiro, RJ – Brasil.

MAIO de 2015

Agradecimentos Jéssica

Agradeço a todos que me suportaram ao longo desses anos que estive na faculdade.

Agradeço à minha família, especialmente ao meu pai e meu irmão pelo apoio durante toda a minha vida e pelo incentivo nos momentos mais difíceis.

Agradeço aos meus amigos da UNIRIO que enfrentaram comigo as dificuldades, dividiram experiências e compartilharam momentos de alegria, que facilitaram a jornada durante todos esses anos.

Agradeço ao Julio pela paciência, parceria e incentivo durante nosso tempo de convivência e, principalmente, na conclusão desse trabalho.

Agradeço a todos os professores e funcionários da Escola de Informática Aplicada e, em especial, ao orientador Astério Tanaka, pela sabedoria, comprometimento e disponibilidade, que foram determinantes para a conclusão deste trabalho.

Agradecimentos à banca pela disponibilidade e dedicação em participar da minha defesa.

Agradecimentos Julio

Agradeço a todos aqueles que me apoiaram em toda minha jornada na faculdade e aos que acreditaram em mim em todos os momentos.

Agradecimentos especiais a toda a minha família, sem ela eu não seria quem sou, nem chegaria aonde cheguei hoje. Em especial ao meu tio, Jorge Marcelo que conviveu comigo no meu dia a dia durante o curso.

Agradecimentos aos meus amigos próximos, Izabella Barboza, Felipe Alfaia e Geovanni Soares pelo apoio moral e psicológico. Passamos por muitos momentos difíceis e felizes durante esse período e não esquecerei a força que me deram nesse momento.

Agradeço em especial a Jéssica Veiga pelo incentivo quando mais precisei, pois sempre acreditou em mim e nunca me deixou fraquejar nos momentos mais difíceis. Quando achei que tudo estava muito desfocado foi ela quem me trouxe de volta ao caminho certo.

Agradecimentos ao meu orientador Astério Tanaka, sem ele nada disso seria possível. Profissional que tem meu grande respeito e apreço.

Agradeço também a todo o corpo docente da UNIRIO, que contribuíram para a minha formação profissional e acadêmica.

Agradecimentos à banca pela disponibilidade e dedicação em participar da minha defesa.

RESUMO

Este trabalho aborda o uso de técnicas de Business Intelligence (BI) na análise de Dados Abertos Governamentais. O objetivo foi aplicar conhecimentos adquiridos ao

longo do curso de Bacharelado em Sistemas de Informação no desenvolvimento de uma aplicação completa de BI, desde a extração, transformação e carga de dados em um banco de dados relacional até a modelagem dimensional da aplicação e a sua análise através de cubos OLAP (On Line Analytical Processing). O assunto escolhido foi a Eleição de 2014, cujos dados estão publicados no Portal do Tribunal Superior Eleitoral. A finalidade da aplicação foi demonstrar a viabilidade do uso de técnicas de BI, usando a plataforma aberta Pentaho BI Suite, no auxílio ao entendimento dos Dados Abertos Governamentais, que são disponibilizados em formato bruto, de difícil compreensão pelo cidadão comum.

Palavras-chave: Business Intelligence, Dados Abertos Governamentais, On Line Analytical Processing, Dados Eleitorais, Pentaho BI Suite.

ABSTRACT

This work covers the use of Business Intelligence (BI) techniques in the analysis of Open Government Data. The objective was to apply the knowledge gained during the Bachelor of Information Systems course in the development of a complete BI application from the extract, transform and load data into a relational database to the dimensional modeling of the application and its analysis through OLAP cubes (On Line Analytical Processing). The chosen subject was the 2014 election, for which data are published on the website of the Superior Electoral Tribunal. The purpose of the application was to demonstrate the feasibility of using BI techniques to aid the understandability of the Open Government Data, which are available in raw format, difficult to understand for ordinary citizens.

Keywords: Business Intelligence, Open Government Data, On Line Analytical Processing, Electoral Data, Pentaho BI Suite.

Índice

1 Introdução	1
1.1 Motivação	3

<u>1.2</u> <u>Objetivos</u>	3
<u>1.3</u> <u>Organização do texto</u>	4
<u>2</u> <u>Dados Abertos Governamentais</u>	5
<u>2.1</u> <u>Introdução</u>	5
<u>2.2</u> <u>Definição de Dados Abertos</u>	6
<u>2.3</u> <u>As Leis</u>	6
<u>2.4</u> <u>Os Princípios</u>	7
<u>2.5</u> <u>Dados utilizados neste trabalho</u>	8
<u>3</u> <u>Conceitos de Business Intelligence</u>	9
<u>3.1</u> <u>Introdução</u>	9
<u>3.2</u> <u>Data Warehouse</u>	9
<u>3.3</u> <u>Modelagem Dimensional</u>	10
<u>3.3.1</u> <u>Fatos</u>	11
<u>3.3.1.1</u> <u>Aditivos</u>	11
<u>3.3.1.2</u> <u>Semiaditivos</u>	11
<u>3.3.1.3</u> <u>Não aditivos</u>	11
<u>3.3.2</u> <u>Dimensões</u>	12
<u>3.3.2.1</u> <u>Dimensões Clássicas</u>	12
<u>3.3.2.2</u> <u>Dimensões Degeneradas</u>	13
<u>3.3.2.3</u> <u>Dimensões Lixo (Junk)</u>	13
<u>3.3.3</u> <u>Esquema Floco de Neve</u>	13
<u>3.3.4</u> <u>Esquema Estrela</u>	14
<u>3.4</u> <u>ETC</u>	15
<u>3.5</u> <u>OLAP</u>	16
<u>3.6</u> <u>Principais Operações OLAP</u>	18
<u>3.6.1</u> <u>Drill-down</u>	18

3.6.2 Roll-up	18
3.6.3 Pivoting	19
3.6.4 Drill-across	19
3.6.5 Drill-through	19
4 Pentaho BI Suite	21
4.1 Introdução	21
4.2 Licenças e formas de obtenção	21
4.3 Arquitetura	22
4.3.1 Pentaho Data Integration (PDI)	22
4.3.2 Pentaho Schema Workbench (PSW)	23
4.3.3 Pentaho Aggregation Designer (PAD)	24
4.3.4 Pentaho Metadata Editor (PME)	25
4.3.5 Pentaho Report Designer (PRD)	25
4.3.6 Saiku Analytics	25
4.3.7 Weka	26
5 Estudo de Caso: Dados eleitorais de 2014	27
5.1 Introdução	27
5.2 Análises	27
5.3 Modelagem Dimensional (Esquema Estrela)	28
5.3.1 Dimensão Localidade	29
5.3.2 Dimensão Candidato	29
5.3.3 Dimensão Cargo	29
5.3.4 Dimensão Ocupação	30
5.3.5 Dimensão Perfil	30
5.3.6 Dimensão Partido	30
5.3.7 Tipo Despesa	31

5.3.8 Tipo Receita	31
5.3.9 Dimensão Tempo	31
5.3.10 Dimensão Tipo Bem	31
5.3.11 Dimensão Situação Turno	32
5.3.12 Fato Vagas	32
5.3.13 Fato Bens	32
5.3.14 Fato Votação	32
5.3.15 Fato Despesa	33
5.3.16 Fato Receita	33
5.4 Desenvolvimento no Pentaho	34
5.4.1 Processo de ETC	34
5.4.2 Criação do Repositório	35
5.4.3 Visualização dos dados	37
5.5 Considerações Gerais sobre o Estudo de Caso	40
6 Conclusão	41
6.1 Considerações Finais	41
6.2 Trabalhos Futuros	42
7 Referências Bibliográficas	43

Índice de Tabelas

Tabela 1: Descrição da Dimensão Localidade	29
Tabela 2: Descrição da Dimensão Candidato	29
Tabela 3: Descrição da Dimensão Cargo	30
Tabela 4: Descrição da Dimensão Ocupação	30
Tabela 5: Descrição da Dimensão Perfil	30
Tabela 6: Descrição da Dimensão Partido	31
Tabela 7: Descrição da Dimensão Despesa	31

<u>Tabela 8: Descrição da Dimensão Receita</u>	31
<u>Tabela 9: Descrição da Dimensão Tempo</u>	31
<u>Tabela 10: Descrição da Dimensão Tipo Bem</u>	32
<u>Tabela 11: Descrição da Dimensão Situação Turno</u>	32
<u>Tabela 12: Descrição do Fato Vagas</u>	32
<u>Tabela 13: Descrição do Fato Bens</u>	32
<u>Tabela 14: Descrição do Fato Votação</u>	33
<u>Tabela 15: Descrição do Fato Despesa</u>	33
<u>Tabela 16: Descrição do Fato Receita</u>	33

Índice de Figuras

<u>Figura 1: Ilustração de uma tabela fato</u>	11
<u>Figura 2: Ilustração de uma tabela de dimensão</u>	12
<u>Figura 3: Ilustração de uma tabela de dimensão clássica</u>	12
<u>Figura 4: Ilustração de uma tabela com dimensão degenerada</u>	13
<u>Figura 5: Ilustração de uma tabela de dimensão lixo</u>	13
<u>Figura 6: Modelagem de um esquema floco de neve</u>	14
<u>Figura 7: Modelagem de um esquema estrela</u>	14
<u>Figura 8: Ilustração da etapa de ETC</u>	16
<u>Figura 9: Ilustração da operação <i>Drill-Down</i></u>	18
<u>Figura 10: Ilustração da operação Roll-Up</u>	19
<u>Figura 11: Ilustração da operação <i>Pivoting</i></u>	19
<u>Figura 12: Ilustração das operações <i>Drill-Across</i> e <i>Drill-Through</i></u>	20
<u>Figura 13: Representação da arquitetura Pentaho</u>	22
<u>Figura 14: Representação de uma transformação no PDI</u>	23
<u>Figura 15: Tela inicial do Schema Workbench</u>	24
<u>Figura 16: Tela inicial do PAD</u>	25
<u>Figura 17: DW Bus Matrix</u>	29
<u>Figura 18: Conexão do PDI com o PostgreSQL via JDBC</u>	34
<u>Figura 19: Processo de ETC das dimensões</u>	35
<u>Figura 20: Processo de ETC da tabela fato Receita</u>	35
<u>Figura 21: Conexão PSW com PostgreSQL via JDBC</u>	36
<u>Figura 22: Criação do Cubo no PSW</u>	37

<u>Figura 23: Valor das receitas por candidato</u>	38
<u>Figura 24: Doadores dos principais candidatos</u>	38
<u>Figura 25: Gráfico com valor das receitas por candidato</u>	39
<u>Figura 26: Gráfico com os doadores dos principais candidatos</u>	39

•

• Introdução

Como muitos conceitos de sistemas de informação, é difícil definir o momento que "nasceu" o termo *Business Intelligence (BI)*. O conceito de BI não é recente, pois povos antigos como os Egípcios e os Persas já usavam os seus princípios para observar e analisar o comportamento das marés utilizando as informações vindas da natureza e da experiência adquirida com o tempo. Outro exemplo eram os índios do Brasil, que revezavam diferentes áreas de terra para manter toda a terra útil e fértil durante os períodos de secas e de chuvas. Ou seja, o conhecimento obtido através da observação era a base para decisões que facilitassem a vida desses povos.

Com a Revolução Industrial, vieram os avanços tecnológicos, o aumento da produtividade e a procura da sociedade por recursos até então não utilizados. Então,

começou-se o investimento para obter um conhecimento maior.

As informações obtidas serviam para alavancar de forma mais proeminente as empresas e auxiliar à tomada de decisão referente à mão de obra e maquinário. Com isso, as fábricas conseguiam confeccionar um produto com maior qualidade e de forma mais rápida.

No início dos anos 1960, a Tecnologia da Informação era baseada em cartões perfurados e transistores. Nessa época, já se podia observar máquinas mais elaboradas entrando em operação e auxiliando o homem na coleta e tratamento dos dados, posteriormente transformados em informação.

Na década de 1970, surgiram os Sistemas Gerenciadores de Banco de Dados (SGBD), portanto, os dados já podiam ser armazenados e manipulados de uma melhor forma. Antes deles, as aplicações usavam sistemas de arquivos do sistema operacional para armazenar as suas informações. Com esse avanço, as grandes corporações começaram a olhar de forma diferente esses novos recursos e investir em novas funcionalidades para os seus sistemas de informação.

Na década de 1980, os computadores começavam a surgir nas residências, graças às grandes campanhas publicitárias impulsionadas principalmente pela IBM. Ao mesmo tempo em que a tecnologia e a informação atravessavam novos níveis, o mercado começou a utilizar o termo *Business Intelligence*, termo escrito pela primeira vez, justamente por um funcionário da IBM em 1958, *Hans Peter Luhn*, autor do artigo “*A Business Intelligence System*”[Luhn, 1958], que abordava sobre o desenvolvimento de sistemas automáticos, baseados em máquinas de processamento de dados, com a função de indexar e codificar automaticamente documentos e distribuí-los dentro das organizações conforme um ponto de ação escolhido. Na época, a preocupação de *Luhn* era com a disseminação da informação dentro das organizações, devido à forma como eram divididas, criando barreiras para o fluxo de informação. Por isso, as empresas precisavam de métodos mais eficientes de recuperação e de disseminação de informação, e como solução, surgiu a proposta de *Luhn*.

Na década de 1990, as grandes corporações já contavam com centros de dados que trabalhavam com grandes fluxos de informação e ajudavam no apoio a decisão e, com isso, o mercado empenhou-se no avanço de tecnologias e criação de softwares para auxiliarem esses centros a obterem informações diferenciadas e mais precisas.

Nesse período, houve a criação de um dos mais importantes componentes de BI até os dias de hoje, o *Data Warehouse* [Inmon, 1992]. Para projetar o *Data Warehouse*, foram criadas técnicas de modelagem dimensional [Kimball, 1996], que permitiram o desenvolvimento de aplicações analíticas típicas da BI.

Desde então, as empresas passaram a se interessar ainda mais pela BI e a contar com ferramentas e técnicas como *Decision Support System* (DSS) [Power, 2007], *Dashboards* [Fews, 2006], *Data Mining* [Fayyad et al., 1996], *On-Line Analytical Processing* (OLAP) [Codd, 1993]. Todas tinham o objetivo de tornar mais rápida e precisa a tomada de decisão, possibilitar melhores estratégias de negócios, com foco em um produto ou serviço, e com base em processos organizacionais.

As técnicas de BI mencionadas acima podem ser aplicadas, nas organizações, tipicamente sobre dados provenientes dos seus sistemas de informação operacionais, através de processos de extração, transformação e carregamento em *Data Warehouses* ou *Data Marts*, projetados para se adequar às ferramentas de análise.

No presente trabalho, essas técnicas de BI foram aplicadas sobre dados abertos governamentais, isto é, dados públicos que são organizados, catalogados e divulgados pelo governo em Páginas de Acesso à Informação e Portais de Transparência. Esses dados, embora estabeleçam um canal direto entre a população e o governo, cumprindo exigências legais, nem sempre estão organizados e disponibilizados de forma adequada para o entendimento e a análise do cidadão comum.

- **Motivação**

O acesso à informação pública está previsto na Constituição Federal e na Declaração Universal dos Direitos Humanos. A implantação da Lei de Acesso à Informação brasileira (Lei 12.527/2011 [Brasil, 2011], regulamentada pelo Decreto 7.724/2012 [Brasil, 2012]) possibilitou a disponibilização de dados públicos através da chamada transparência ativa, em que o Governo publica, espontaneamente, dados de seus órgãos que sejam de interesse do cidadão. Entretanto, dada a característica primária dos dados abertos governamentais, os dados brutos publicados não estão prontos para o consumo do cidadão. Desenvolver uma solução tecnológica que permita ao cidadão comum analisar os dados publicados pelo governo, através da Internet, estimula a sociedade a buscar um melhor entendimento sobre assuntos de seu

interesse, como a política de investimentos, acesso a serviços e controle de contas públicas.

- **Objetivos**

Este trabalho tem como objetivo aplicar conhecimentos adquiridos no Curso de Bacharelado em Sistemas de Informação, em particular nas disciplinas de bancos de dados e de modelagem, projeto e construção de sistemas, no desenvolvimento de uma prova de conceito do uso de ferramentas de BI, utilizando como fonte dados abertos governamentais. A prova de conceito do uso da ferramenta Pentaho, ao mesmo tempo em que serve como prática dos conhecimentos adquiridos, permite futuramente disponibilizar a informação pública de forma acessível ao cidadão comum, com algum valor agregado, por meio de um sitio na web.

Para a prova de conceito , foi criado um modelo dimensional a partir dos dados brutos armazenados no Repositório de Dados Eleitorais, disponibilizados no sítio do Tribunal Superior Eleitoral (TSE) <http://www.tse.jus.br/hotSites/pesquisas-eleitorais/>. Esse modelo dimensional serviu de base para aplicações de análise executadas com o uso da plataforma Pentaho BI Suite Community Edition [Pentaho, 2014].

- **Organização do texto**

O presente trabalho está estruturado em capítulos e, além desta introdução, será desenvolvido da seguinte forma:

- Capítulo II: são apresentados os conceitos de dados abertos governamentais, bem como suas leis e princípios;
- Capítulo III: são apresentados conceitos e técnicas de BI (*Business Intelligence*), como: Data Warehouses, Modelagem Dimensional, ETC e OLAP;
- Capítulo IV: é apresentada uma descrição geral da plataforma Pentaho, abordando as licenças disponíveis e a sua arquitetura;
- Capítulo V: é apresentada, através de um estudo de caso, a construção da solução

de BI a fim de auxiliar na compreensão dos dados;

- Capítulo VI: reúne as considerações finais, assinala as contribuições do trabalho e sugere possibilidades de aprofundamento posterior.

• **Dados Abertos Governamentais**

• **Introdução**

O movimento na busca por transparência política, prevenção à corrupção e participação pública dos cidadãos e de forma igualitária antecede à implantação da Lei de Acesso à Informação (Lei 12.527/2011, regulamentada pelo Decreto 7.724/2012). Porém, apenas após sua entrada em vigor ficou evidente a falta de maturidade na disponibilização dos dados abertos governamentais.

Na Constituição de 1988, já era obrigação do Estado prestar contas de sua atuação, bem como era direito do cidadão o acesso à informação.

Em 2000, a Lei de Responsabilidade Fiscal (Lei Complementar nº 101, de 4 de maio de 2000) estabeleceu, em regime nacional, normas de finanças públicas voltadas para a responsabilidade na gestão fiscal, definindo padrões a serem seguidos nos gastos públicos pela União, pelos Estados, pelo Distrito Federal e pelos Municípios. Esta lei é um dos maiores instrumentos de transparência em relação aos gastos públicos no Brasil.

Em 2003, foi criada a Controladoria Geral da União (CGU), órgão federal

responsável por auxiliar direta e imediatamente o Presidente da República nos assuntos relativos à defesa do patrimônio público e ao incremento da transparência da gestão, por meio das atividades de controle interno, auditoria pública, correição, prevenção e combate à corrupção e ouvidoria, no âmbito do Poder Executivo.

A partir de novembro de 2004, foi permitido ao cidadão acompanhar o uso do dinheiro público na internet, em âmbito federal, através do Portal da Transparência. Nele podemos encontrar dados sobre a aplicação de recursos federais, como: despesas, receitas, convênios, informações sobre servidores públicos, etc.

Em 2009, a Lei Complementar nº 131, de 27 de maio de 2009, determinou que a União, os Estados, o Distrito Federal e os Municípios disponibilizem, em meio eletrônico e tempo real, informações pormenorizadas sobre sua execução orçamentária e financeira. Esta lei foi criada para alterar a Lei de Responsabilidade Fiscal (Lei Complementar nº 101, de 4 de maio de 2000).

Foi neste cenário que a Lei nº 12.527, a Lei de Acesso à Informação – LAI foi sancionada em 18/11/2011 e entrou em vigor em 16/5/2012. De acordo com a LAI e seu regulamento (Decreto nº 7.724, de 16/5/2012), “é dever dos órgãos e entidades promover, independentemente de requerimentos, a divulgação em seus sítios na Internet de informações de interesse coletivo ou geral por eles produzidas ou custodiadas”.

- **Definição de Dados Abertos**

De acordo com o W3C, “Dados Abertos Governamentais são a publicação e disseminação das informações do setor público na Web, compartilhadas em formato bruto e aberto, compreensíveis logicamente, de modo a permitir sua reutilização em aplicações digitais desenvolvidas pela sociedade.”

(<http://www.w3c.br/divulgacao/pdf/dados-abertos-governamentais.pdf>).

Segundo a [definição](#) da *Open Knowledge Foundation*, dados abertos são aqueles que a sociedade pode livremente usar, reutilizar e redistribuir, com a exigência máxima de dar crédito ao autor e compartilhá-los com a mesma licença.

(<https://okfn.org/opendata/>).

O Guia para Criação da Seção de Acesso à Informação nos Sítios Eletrônico dos Órgãos e Entidades Federais interpreta o termo “formato aberto” como uma

“especificação publicada para armazenar dados digitais, livre de limitações legais no uso” e tem como objetivo principal “garantir o acesso a longo prazo aos dados sem incertezas atuais ou futuras no que diz respeito às diretrizes legais ou à especificação técnica” (http://www.acessoainformacao.gov.br/lai-para-sic/sic-apoio-orientacoes/guias-e-orientacoes/guia_secaositos-sem-logo.pdf).

- **As Leis**

- As três “leis” dos dados abertos governamentais, elaboradas por David Eaves (<http://eaves.ca/about-david/>), especialista em políticas públicas e ativista dos dados abertos, propõem que: Se o dado não pode ser encontrado e indexado, ele não existe;
- Se não estiver aberto e disponível em formato compreensível por máquina, ele não pode ser reaproveitado;
- Se algum dispositivo legal não permitir sua reaplicação, ele não é útil.

- **Os Princípios**

Um grupo de trinta defensores públicos, em dezembro de 2007, na Califórnia, escreveu os oito princípios dos dados abertos governamentais, que são:

- **Completos:** Todos os dados públicos estão disponíveis, não podem estar sujeitos a limitações de privacidade, segurança ou de privilégios. São eletronicamente gravados, mas não se limitam a documentos, bancos de dados, transcrições e gravações audiovisuais;
- **Primários:** Os dados são publicados na forma que são coletados da fonte, com maior nível de granularidade possível, e não de forma agregada ou

transformada.

Mesmo que uma entidade opte por transformar os dados e utilizá-los em um sítio de Internet construído para usuários finais, esta ainda tem a obrigação de trazer a informação completa para preservar os dados para a posteridade e permitir que outros possam construir seus próprios sítios;

- **Atuais:** Os dados são disponibilizados tão rapidamente quanto necessário a fim de preservar o seu valor;
- **Acessíveis:** Os dados são disponibilizados para o mais amplo público e para as mais variadas finalidades.

Visto isso, estes devem ser disponibilizados na Internet e preparados para o acesso de pessoas com deficiência, além de acessíveis aos usuários de diferentes plataformas de hardware e software;

- **Compreensíveis por máquinas:** Os dados são minimamente estruturados para permitir o seu processamento automatizado;
- **Não discriminatórios:** Os dados estão disponíveis a qualquer pessoa, sem a necessidade de identificação ou registro;
- **Não proprietários:** Os dados estão disponíveis em um formato que nenhuma entidade tem controle exclusivo;
- **Livres de licenças:** Os dados não estão sujeitos a direitos autorais, marcas registradas, patentes ou segredo industrial. Restrições razoáveis de privacidade, segurança e controle de acesso podem ser permitidas desde que reguladas por estatutos.

- **Dados utilizados neste trabalho**

O sítio do Tribunal Superior Eleitoral (<http://www.tse.jus.br/hotSites/pesquisas-eleitorais/>) é o repositório fonte dos dados deste trabalho. Nele estão publicados dados eleitorais que contém dados brutos das eleições desde 1945. Estes dados são disponibilizados, em formato .zip, para pesquisadores, imprensa e cidadãos em geral, principalmente aqueles que tenham interesse em analisar os dados de candidatura, resultados e prestação de contas.

A manutenção do site está sob a responsabilidade do Tribunal Superior

Eleitoral (TSE), órgão máximo da Justiça Eleitoral, que exerce o papel fundamental na construção e no exercício da democracia brasileira.

Para este trabalho, foram utilizados os dados das eleições de 2014, sobre candidatos, bem como seus bens declarados, despesas e receitas, além de números de votação.

Do ponto de vista dos princípios de dados abertos governamentais, todos os dados trabalhados estão em formato .csv e podem ser baixados, sem nenhum controle de acesso, do sítio do Tribunal Superior Eleitoral. Portanto, atendem aos princípios 1, 4, 5, 6, 7 e 8, acima citados. Alguns dias após o término do segundo turno da eleição 2014, todos os dados já estavam disponíveis no sítio, respeitando o princípio 3.

Além disso, os dados disponibilizados são primários, estando de acordo com o princípio 2, e não são facilmente interpretados pelos cidadãos. Por isso, justifica-se a necessidade de desenvolver uma solução de BI a fim de facilitar a análise e entendimento desses dados. Os dados trabalhados são apresentados com um maior detalhamento no Capítulo 5.

• Conceitos de Business Intelligence

• Introdução

A Inteligência de Negócio ou *Business Intelligence* (BI) é um conceito que pode ser entendido como a utilização de várias aplicações e tecnologias envolvidas no processo de coleta, armazenamento, organização e análise a fim de oferecer suporte à tomada de decisão e definição de estratégias de competitividade nos negócios de uma empresa (Barbieri, Carlos. BI2 – Business Intelligence: modelagem e qualidade. Rio de Janeiro: Elsevier, 2011).

O conceito de BI contempla estruturas de dados, como *data warehouse* (DW) e *data marts* (DM), ferramentas de ETC (extração, transformação e carga), fundamentais na transformação de dados transacionais em informacional, além de aplicações, como OLAP (*On-Line Analytical Processing*) e *data mining*.

- **Data Warehouse**

Os primeiros projetos de DW tiveram como referência duas fontes de inspiração: Bill Inmon e Ralph Kimball. O primeiro, Bill Inmon, é considerado o “pai do Data Warehouse”, e definiu este termo como: “uma coleção de dados, orientada a assunto, integrada, variante no tempo e não volátil, para suporte ao gerenciamento dos processos de tomada de decisão” (Inmon, William. *Building the Data Warehouse*. Wiley Publishing, 2005).

Na sua visão, deve-se modelar um DW único para toda a empresa e posteriormente partir para Data Marts construídos por assuntos ou departamentos, ou seja, uma abordagem “*top-down*”. Esta requer uma visão geral da solução, o que possui grande abrangência e implica em longo prazo de implementação. É um método habitualmente indicado para projetos onde todo o escopo é facilmente identificado, compreendido e mais simples de ser implementado.

Já a visão de Ralph Kimball é justamente a contrária. Para ele, uma empresa deve construir primeiramente *Data Marts* separados que, conforme forem evoluindo, deverão ser integrados para se chegar a um único modelo corporativo. Esse conceito foi batizado de “*Data Warehouse Bus Architecture*”, em que o Data Warehouse é um conjunto de *Data Marts* (Kimball, Ralph; Ross, Margy. *The Data Warehouse Toolkit*. John Wiley, 2002.).

A abordagem defendida por Kimball é a “*bottom-up*” e é usualmente utilizada em projetos onde é difícil se enxergar os limites e sua complexidade total, se tornando importante a construção incremental de *Data Marts* até chegar ao *Data Warehouse* corporativo. É um método que tem maiores resultados de sucesso, pois dessa forma o processo tem um desenvolvimento evolutivo, possuindo menores riscos de erro. Este método também tem um retorno mais rápido, já que a estrutura pode ser utilizada antes mesmo que seja feita a completa construção do *Data Warehouse*. Com isso, o

usuário acompanha o processo de produção do sistema, facilitando o acompanhamento e ajudando no processo de desenvolvimento de novos *Data Marts*.

De modo geral, um DW deve possibilitar a análise de grandes volumes de dados, ser adaptável, apresentar dados íntegros e seguros e servir como suporte para a tomada de decisão.

- **Modelagem Dimensional**

Na modelagem dimensional, os dados são organizados em um formato estelar, onde a atenção está mais voltada para as tabelas de entrada (dimensões) do que para os dados transacionais particionados (fatos). Portanto, permite ao usuário trabalhar com os dados em uma forma mais próxima (transpondo problemas reais da melhor forma possível para modelos em que serão baseados os bancos de dados) de seu entendimento e possibilita a busca de informações (fato), de forma clara e direta, através das dimensões.

- **Fatos**

As tabelas de fatos servem para armazenamento de medidas associadas a algum evento de negócio. Ou seja, estas tabelas armazenam indicadores de uma empresa, demonstrado com o exemplo da figura 1.

Estas tabelas possuem como chave primária um campo *multi-key* constituído das chaves primárias das tabelas de dimensão com que se relacionam. Logo, devem apresentar a mesma granularidade das tabelas de dimensão.

Normalmente, as tabelas de fatos ocupam mais espaço do que as de dimensão. Aproximadamente 90% de um DW é ocupado por tabelas de fatos, pois estas armazenam um registro para cada acontecimento que representa.

Fato_Vendas
FK_Tempo
FK_Produto
FK_Loja
Quantidade
Preço de Venda
Preço de Custo

Figura 1: Ilustração de uma tabela fato

Abaixo, estão os tipos de medidas de uma tabela fato.

- **Aditivos**

São medidas que são passíveis de soma ao longo de todas as dimensões. Por exemplo, valores de venda de um produto.

- **Semiaditivos**

São medidas que não são passíveis de soma ao longo de todas as dimensões existentes. Por exemplo, quantidade em estoque só faz sentido ser somada na dimensão produto e não nas dimensões de data e loja.

- **Não aditivos**

Quando um valor não é passível de soma em qualquer dimensão ou só permitem contagem e estatísticas associadas a essa contagem. Por exemplo, o valor de porcentagem de lucro.

- **Dimensões**

As tabelas de dimensão representam entidades de negócio e armazenam informações que qualificam os fatos. Estas tabelas apresentam uma chave

primária que lhes dá unicidade e serve de chave estrangeira para as operações de junção com as tabelas de fatos.

As tabelas de dimensão (exemplo abaixo na figura 2) não possuem muitas linhas se comparadas com as de fatos, porém possuem diversas colunas e devem ser entendidas como tabelas de filtro para as consultas das análises e relatórios.

Produto
PK_Produto
Descrição
Categoria
Marca
Modelo
Fabricante
Peso
Cor

Figura 2: Ilustração de uma tabela de dimensão

- **Dimensões Clássicas**

Algumas dimensões estão presentes na maioria dos projetos de DW/DM e são consideradas clássicas. Dentre elas estão as dimensões de “Tempo” e “Localidade”. A primeira, armazena dados como data, mês, ano, trimestre, flag de feriado, etc. Já a segunda, armazena dados de cidade, estado, região, país, etc (como na figura 3, na próxima página).

Tempo
PK_Tempo
Data
Mês
Trimestre
Ano
Flag_Feriado

Figura 3: Ilustração de uma tabela de dimensão clássica

- **Dimensões Degeneradas**

Derivam de uma tabela fato e não tem sua própria Tabela de dimensão. São utilizadas quando se deseja manter identificadores específicos do sistema, como os números de ordem e similares. Servem de referência direta a um sistema transacional (exemplo abaixo na figura 4).

Fato_Vendas
FK_Tempo
FK_Produto
FK_Loja
Nota Fiscal (DD)
Quantidade
Preço de Venda
Preço de Custo

Figura 4: Ilustração de uma tabela com dimensão degenerada

- **Dimensões Lixo (Junk)**

Este conceito está relacionado com indicadores binários, flags e atributos de baixa cardinalidade, como sexo e estado civil. Estas características são utilizadas de forma combinada, a fim de evitar a criação de um número muito grande de dimensões (exemplo abaixo na figura 5).

Perfil
PK_Perfil
Estado Civil
Sexo
Raça

Figura 5: Ilustração de uma tabela de dimensão lixo

- **Esquema Floco de Neve**

Neste tipo de esquema, as dimensões são normalizadas, ou seja, múltiplas tabelas representam uma dimensão. Esta estrutura é chamada de “floco de neve” porque as tabelas fato e de dimensões se combinam, em formato similar a um floco de neve. Neste esquema, são necessários diversos comandos de junções para recompor as informações desejadas (exemplo abaixo na figura 6).

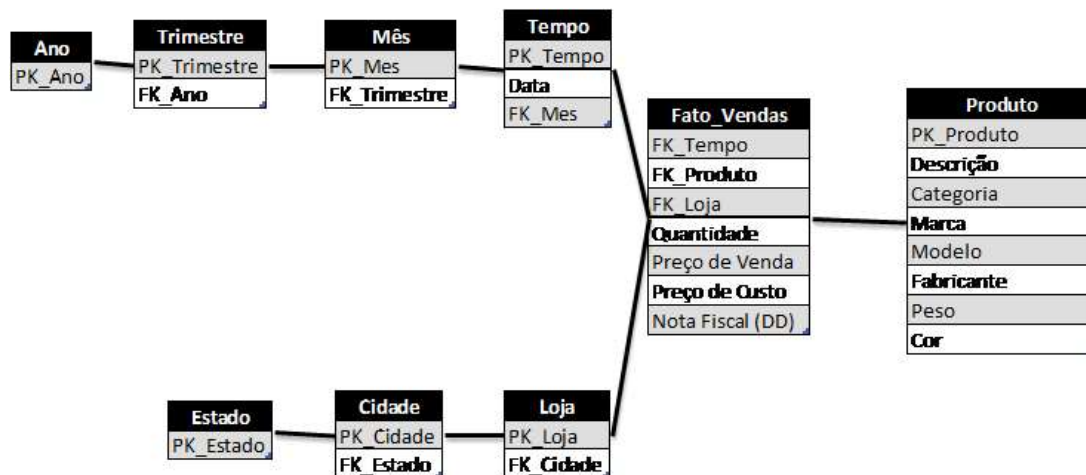


Figura 6: Modelagem de um esquema floco de neve

- **Esquema Estrela**

Neste modelo, todas as tabelas de dimensão se relacionam diretamente com a tabela de fatos. Com isso, essas tabelas devem conter todas as descrições necessárias de uma tabela fato como “Produto”.

Ao contrário do esquema floco de neve, essa abordagem não requer a normalização das tabelas dimensão. Logo, uma dimensão é representada por uma única tabela. Isso gera um aumento no tamanho das tabelas dimensão, visto que descrições textuais são repetidas em todos os registros. Em compensação, as junções só ocorrem entre tabelas de dimensões e de fatos (como na figura 7, na próxima página).

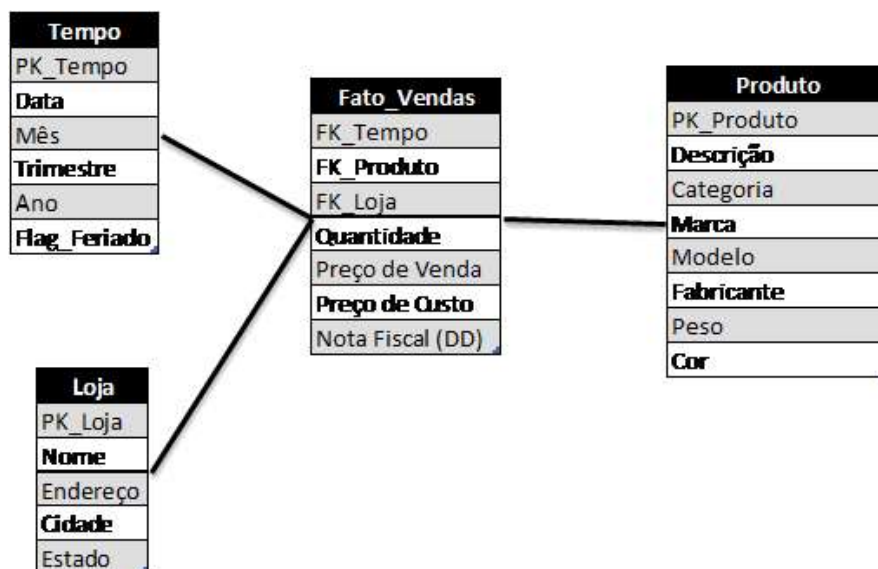


Figura 7: Modelagem de um esquema estrela

No presente trabalho, é utilizada a modelagem dimensional com esquema estrela, em função dos ganhos de desempenho quando comparado ao esquema floco de neve. Há um entendimento comum de que a redundância dos

dados é compensada pelas reduções dos comandos de junções.

- **ETC**

A etapa de ETC (Extração, Transformação e Carga), mais conhecida pelo seu nome em inglês ETL (*Extract, Transform, Load*), é considerada uma das mais críticas do *Data Warehouse* e/ou *Data Mart*. Esta etapa subdivide-se em três fases:

- Extração de dados de diversas fontes, como bancos de dados relacionais, arquivos texto, etc.;
- A transformação/limpeza desses dados, a fim de garantir a integridade dos dados;
- A carga destes dados comumente em um [*Data Mart*](#) ou um [*Data Warehouse*](#).

Na primeira fase, é preciso um extenso planejamento que deve contar com a clareza dos requisitos de negócio, uma forte análise da viabilidade dos dados a serem utilizados e as políticas de segurança adotadas em relação a esses dados. Logo em seguida, os dados são coletados dos seus sistemas de origem e transferidos para o ambiente onde ocorrerá a transformação dos dados.

Na segunda fase, é a etapa onde é feita a limpeza, conversão, ajustes e a consolidação dos dados e onde é aplicada uma série de regras e funções aos dados extraídos. Às vezes, os dados necessitam de pouca manipulação. Porém na maioria delas é necessário trabalhar algumas transformações, como, por exemplo, a junção de dados de diversas fontes, seleção apenas de determinadas colunas, tradução de valores codificados, formatação da data, quebra de informação e transposição, que consiste na técnica de transformar várias colunas em múltiplas linhas ou vice-versa.

A última fase do processo de ETC é a entrega ou carga dos dados. Nesta, é necessário estruturar e carregar os dados para as tabelas do *Data Warehouse* e/ou *Data Mart*, seguindo o modelo dimensional do projeto (exemplo abaixo na figura 8).

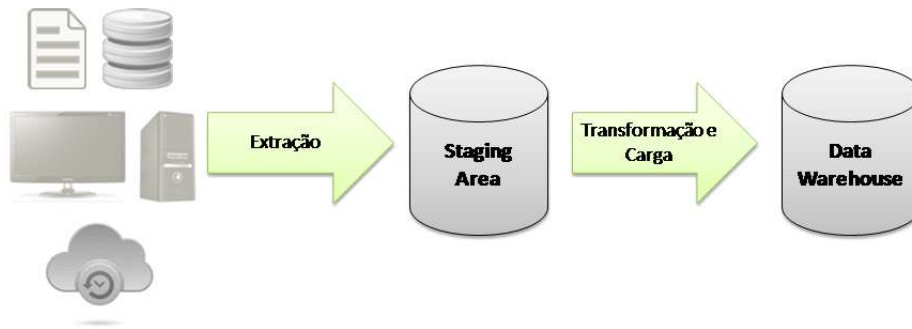


Figura 8: Ilustração da etapa de ETC

Antes de começar a etapa de ETC, é importantíssima a escolha de uma ferramenta adequada. Portanto, é necessário levar em consideração alguns pontos, como:

- Suporte a plataforma;
 - Diversidade de Conectores: apoiar a extração de dados de múltiplas fontes;
 - Execuções em paralelo;
 - Desempenho: utilizar métodos otimizados para lidar com grandes volumes de dados;
 - Escalabilidade: quando confrontada com cargas muito elevadas de trabalho, a ferramenta deve ser capaz de distribuir a tarefa entre múltiplos servidores;
 - Capacidade de visualização dos dados antes e depois de cada transformação;
 - Apoiar o agendamento de tarefas ETC, não necessitando de intervenção humana para completar uma tarefa particular, aproveitando assim o melhor tempo;
 - Suporte a programação de linha de comando;
 - Suportar a reutilização da lógica de transformação para que o usuário não precise reescrever várias vezes a mesma lógica de transformação.
-
- **OLAP**

On-Line Analytical Processing (OLAP) e as bases multidimensionais nos *Data Marts* possibilitam a consulta e análise de um grande volume de dados de forma mais rápida, sólida e interativa, a fim de suportar o processo decisório feito por analistas, gerentes e executivos. A estrutura de dados típica para OLAP é o cubo, formado por uma tabela de fato, as dimensões relevantes para a análise, com suas possíveis hierarquias, e as medidas a serem agregadas durante as consultas.

Os usuários que utilizam ferramentas OLAP veem os dados de maneira consolidada e esporadicamente lidam com um único registro. A complexidade aumenta à medida que as consultas dos usuários são alteradas.

Para cada método de armazenamento de um cubo OLAP, temos diferentes arquiteturas de aplicação a ser utilizada, cada uma com funções bem específicas. Quando o método de armazenamento e a aplicação escolhida se combinam, o rendimento das análises multidimensionais é mais ágil. Esses tipos de arquitetura são classificados de acordo com a tecnologia de banco de dados:

- *Relational On Line Analytical Processing* (ROLAP) utiliza a tecnologia de banco de dados relacionais para armazenar seus dados, bem como suas consultas são também processadas pelo gerenciador do banco de dados relacional;
- *Multidimensional On Line Analytical Processing* (MOLAP) são ferramentas que disparam suas requisições diretamente ao servidor de Banco de Dados multidimensional. Após o envio da requisição, o usuário continua manipulando os dados diretamente no servidor, tendo um ganho no desempenho;
- *Hybrid On Line Analytical Processing* (HOLAP) são ferramentas híbridas. É a combinação entre ROLAP e MOLAP, utilizando o melhor de ambas as categorias: a escalabilidade de ROLAP e o alto desempenho do MOLAP;
- *Desktop On Line Analytical Processing* (DOLAP) são ferramentas que disparam uma consulta de uma estação cliente para o servidor que, por sua vez, retorna enviando o cubo resultante de volta, para que possa ser analisado pelo usuário. Esse tipo de análise é utilizado para fornecer portabilidade aos dados. Ela emite uma consulta para o servidor e recebe as informações de volta para ser analisada na estação. Essa arquitetura oferece

a vantagem de reduzir o tráfego na rede e melhora o desempenho de processamento do servidor, pois todo o trabalho de análise das informações é feito pela máquina cliente;

- *Web On Line Analytical Processing* (WOLAP) são ferramentas que disparam uma consulta via um navegador web para o servidor, que por sua vez retorna enviando o cubo processado de volta, para que possa ser analisado pelo usuário.

- **Principais Operações OLAP**

As principais operações disponíveis nas interfaces OLAP são:

- **Drill-down**

Consiste em fazer uma exploração relacionada com o nível de granularidade dos dados armazenados. O *Drill-down* está relacionado com o fato de se partir de um nível mais alto de hierarquia até níveis menores, ou seja, pode-se “descer” para o dado mais detalhado. Por exemplo, analisar uma informação tanto anualmente quanto diariamente, partindo da mesma base de dados (exemplo na próxima página na figura 9).

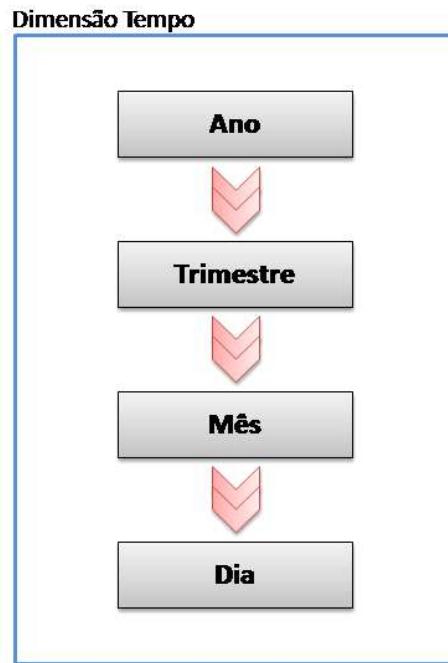


Figura 9: Ilustração da operação *Drill-Down*

- **Roll-up**

O *Roll-up* (ou *Drill-up*) consiste exatamente no conceito contrário ao *Drill-down*, ou seja, o *roll-up* consiste em “subir” de um nível mais baixo da hierarquia, mais detalhado do dado, para um nível mais alto. Por exemplo, informações de venda em nível diário para uma sumarização em um nível mensal (exemplo na próxima página na figura 10).

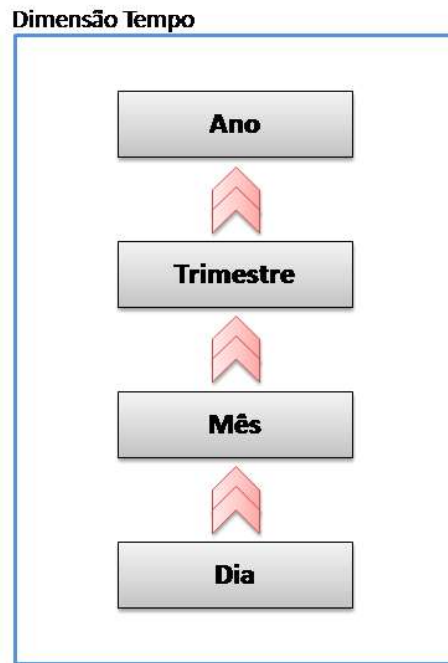


Figura 10: Ilustração da operação Roll-Up

- **Pivoting**

Consiste na rotação dos eixos de visualização das dimensões de um cubo OLAP, através da troca das colunas pelas linhas no resultado da consulta (exemplo abaixo na figura 11).

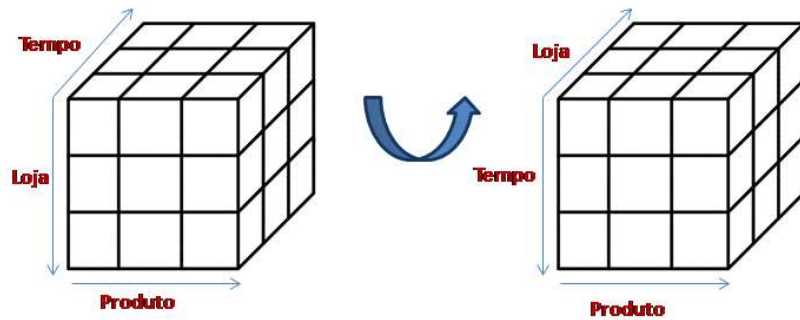


Figura 11: Ilustração da operação *Pivoting*

- **Drill-across**

O *Drill-across* consiste na ideia de passar de um esquema para outro, desde que ambos os esquemas possuam dimensões em comum. Por exemplo, comparar informações de vendas e entregas, como na Figura 12.

- **Drill-through**

O *Drill-through* consiste na ideia de poder “acessar” uma informação em um nível de detalhamento menor do que o disponibilizado na tabela fato e permitido pela sua granularidade. Tem a mesma lógica do *Drill-down*, só que o detalhe é buscado em uma estrutura além do esquema dimensional. Por exemplo, sair de um fato de vendas e buscar em um sistema transacional de origem, as informações de uma determinada venda através do número da nota fiscal (exemplo abaixo na figura 12).

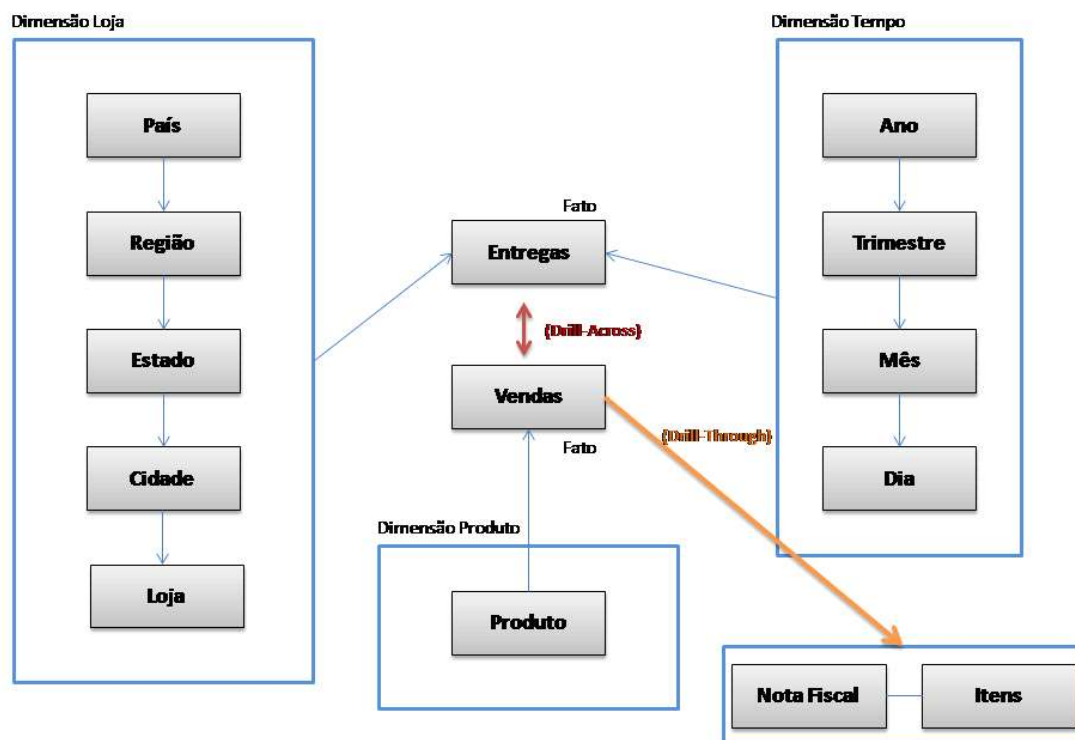


Figura 12: Ilustração das operações *Drill-Across* e *Drill-Through*

Após o processo teórico que é empregado no caso de uso, é preciso conhecer a ferramenta que será usada para processar os dados, o Pentaho BI Suite.

- **Pentaho BI Suite**

- **Introdução**

Pentaho BI é uma suíte completa de código aberto, desenvolvida em Java, e é o resultado da integração de diversas ferramentas de BI desenvolvidas de modo independente. A solução proporciona a transformação de dados (ETC), desenvolvimento de relatórios, análises OLAP, painéis (dashboards) e mineração de dados (data-mining).

Segundo o Gartner Group (www.gartner.com/), o Pentaho BI Community Edition é a plataforma de BI open source mais utilizada no mundo. No Brasil, há um crescente interesse pela sua utilização em soluções corporativas, inclusive na área de governo.

Nesta seção, serão apresentados os tipos de licenças, bem como as ferramentas necessárias para o desenvolvimento de um projeto de BI de ponta-a-ponta, desde a integração dos dados até os relatórios para os usuários finais.

- **Licenças e formas de obtenção**

A plataforma *Pentaho BI* oferece duas edições diferentes: uma é a *Community Edition* (CE) e a outra é a *Enterprise Edition* (EE).

A CE é distribuída como código aberto para desenvolvimento cooperativo pela comunidade, através da Licença Pública da Pentaho (PPL - Pentaho Public License). Existem diversas versões disponíveis e podem ser baixadas gratuitamente, através do endereço <http://sourceforge.net/projects/pentaho/files/>.

A EE é uma suíte de ferramentas já previamente integradas e algumas mais desenvolvidas que a versão livre. Além disso, conta com suporte técnico, manutenção com atualizações frequentes e acesso a uma melhor documentação e treinamento. O download da versão EE de avaliação pode ser efetuado no próprio sítio do Pentaho (<http://www.pentaho.com/>), para testes durante 30 dias. Após esse período, é necessário realizar a compra da licença, pois se trata de um software comercial.

- **Arquitetura**

Comumente chamado de Pentaho BI Platform, e recentemente renomeado para Business Analytics Platform (Plataforma BA), consiste em um servidor de BI, além de alguns clientes e *add-ons*.

A plataforma é executada em servidores compatíveis com J2EE e pode ser incorporada em outros servidores de aplicativos Java (exemplo abaixo na figura 13).

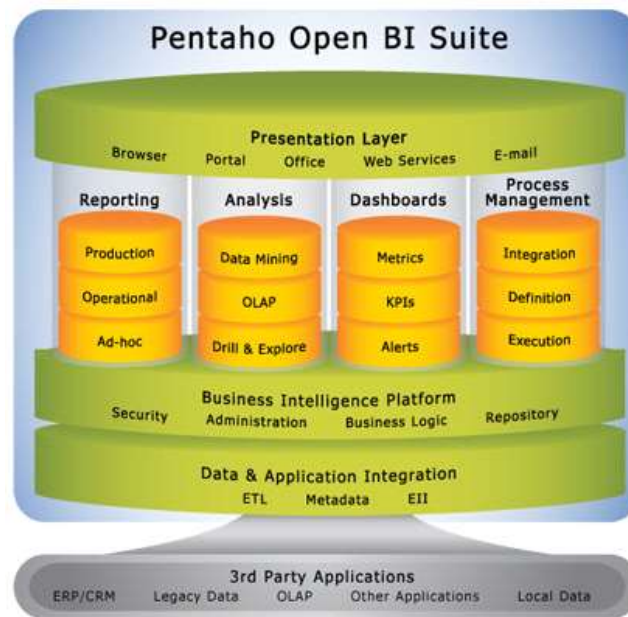


Figura 13: Representação da arquitetura Pentaho

A plataforma oferece algumas funcionalidades, como:

- Recursos de gerenciamento de usuários;
- Consultas e análises em tempo real, se utilizando de operações como *drill-down*, *roll-up* e *pivoting*;
- Relatórios e painéis;
- Workflows.

-

- **Pentaho Data Integration (PDI)**

O PDI, também conhecido como *Kettle*, é uma ferramenta de código aberto que auxilia ao usuário na integração dos dados e nas transformações realizadas no processo de ETC (exemplo abaixo na figura 14).

O *Kettle* possui quatro aplicações diferentes:

- *Spoon*: É uma interface gráfica que permite projetar o fluxo de dados desde os passos (*steps*) de entrada de dados (input), passando pela transformação e gerando a saída (output);
- *Pan*: É uma ferramenta que executa transformações modeladas com o *Spoon*, através de linha de comando;
- *Kitchen*: É uma ferramenta em linha de comando que executa jobs criados com o *Spoon*;
- *Carte*: É um servidor web simples que permite que transformações e trabalhos sejam executados remotamente.

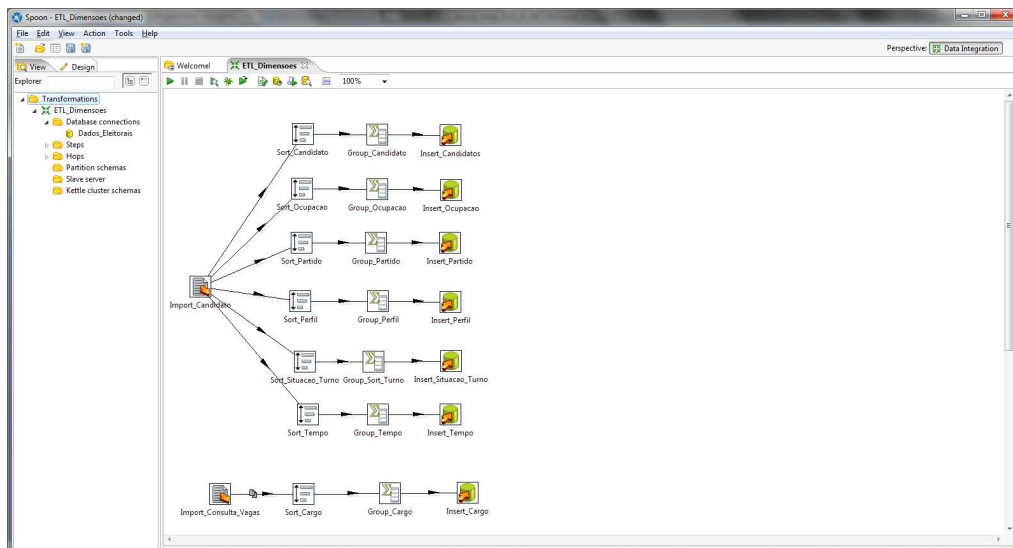


Figura 14: Representação de uma transformação no PDI

- **Pentaho Schema Workbench (PSW)**

O *Schema Workbench* é uma interface de design, desenvolvida em *Java*, que permite o desenvolvimento e testes de esquemas de cubos OLAP, através de arquivos XML, que são executados pelo Mondrian, o *engine* OLAP do Pentaho (exemplo na próxima página na figura 16).

O PSW fornece as seguintes funcionalidades:

- Editor de esquema integrado, para construção dos cubos OLAP, apresentando a fonte dos dados no rodapé para validação;
- Testes das consultas MDX (multi-dimensional expressions ou expressões multidimensionais) nos esquemas e nas bases de dados;
- Visualização da estrutura da base de dados.

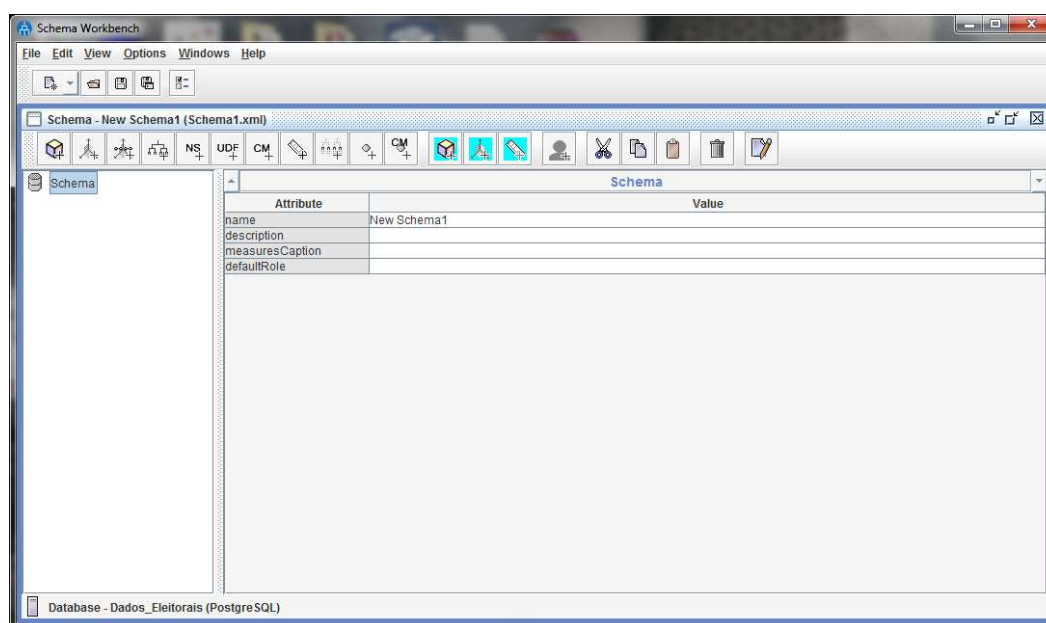


Figura 15: Tela inicial do Schema Workbench

- **Pentaho Aggregation Designer (PAD)**

É uma ferramenta gráfica, desenvolvida em *Java*, para simplificar a criação e implantação de tabelas agregadas, melhorando o desempenho das análises no Pentaho (*Mondrian*) e seus Cubos (OLAP). Ela fornece uma interface simples que permite criar tabelas agregadas de dimensões com níveis, de acordo com a especificação necessária (exemplo na próxima página na figura 16).

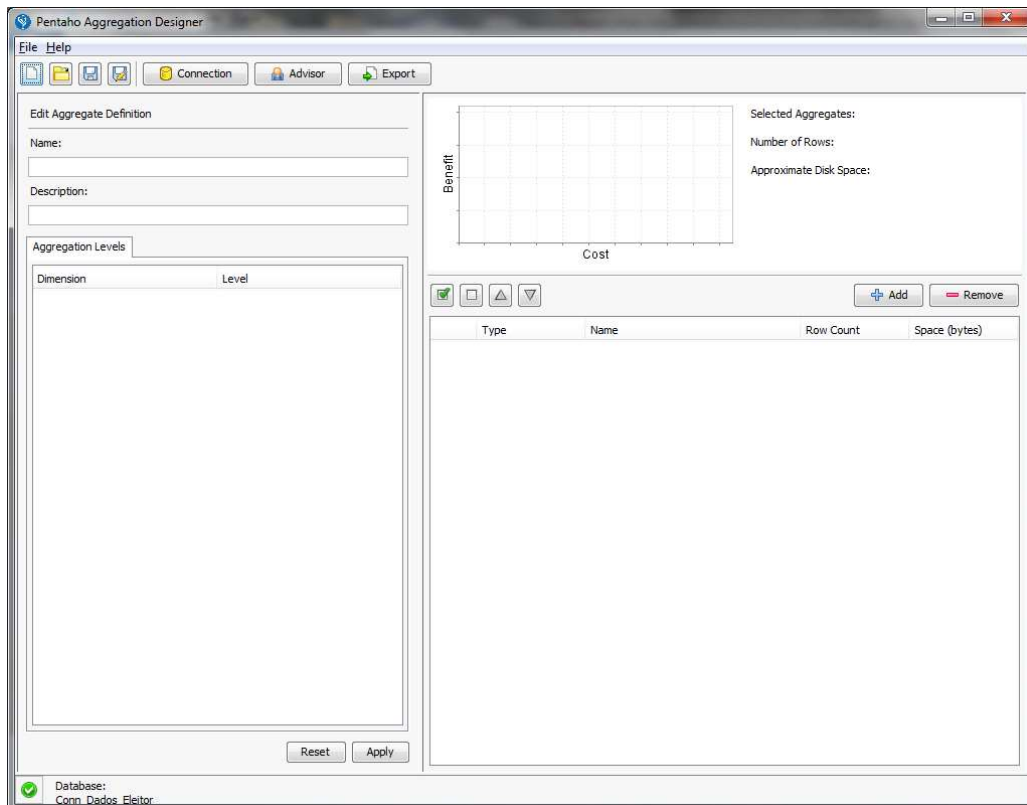


Figura 16: Tela inicial do PAD

- **Pentaho Metadata Editor (PME)**

Ferramenta para criar a camada de metadados. Auxilia o usuário mapeando a estrutura física da base de dados em um modelo lógico, onde campos extraídos do banco de dados recebem nomes para uma melhor visualização do usuário final.

- **Pentaho Report Designer (PRD)**

É uma ferramenta para geração de relatórios que tem a capacidade de se conectar a múltiplas fontes de dados, como, por exemplo, SQL, MDX e Community Data Access. Ela é capaz de gerar relatórios em PDF, Excel, HTML, CSV etc.

- **Saiku Analytics**

O *Saiku Analytics* é um cliente web disponível como *plug-in* para o *Pentaho BI Server*. Ele utiliza o *Mondrian* para proporcionar uma forma fácil de usar o recurso de Cubos OLAP através de uma experiência simples para usuário final.

- **Estudo de Caso: Dados eleitorais de 2014**

- **Introdução**

Este estudo de caso tem como foco disponibilizar, analisar e agregar valor à base de dados eleitorais, a fim de despertar o interesse do cidadão comum pelo aspecto da transparência política.

Sendo uma aplicação de BI, o estudo de caso segue as seguintes etapas de desenvolvimento:

- Levantamento dos requisitos sob a forma de questões de análise a

serem respondidas;

- Identificação e seleção das fontes de dados capazes de atender aos requisitos;
- Modelagem dimensional e criação do banco de dados;
- Desenvolvimento na plataforma de implementação, compreendendo:
 - Processo de ETC com a carga das tabelas do banco de dados;
 - Construção de cubos OLAP para análise dos dados;
 - Visualização das análises na ferramenta OLAP.

- **Análises**

Por se tratar de várias bases, é possível derivar diversas análises pertinentes. Este estudo de caso foi guiado por alguns questionamentos (feito por meio de perguntas informais a um pequeno grupo), como:

- Qual candidato recebeu o maior valor em doações?
- Qual foi seu maior doador?
- Qual candidato mais gastou com publicidade no Rio de Janeiro e qual o que menos gastou? Qual foi o valor?

Para responder aos questionamentos acima, foram utilizados os seguintes arquivos fontes:

- Dados de Candidaturas
("consulta_cand_<ano_eleicao>_<sigla_UF>");
- Bens declarados ("bem_candidato_<ano_eleicao>_<sigla_UF>");
- Vagas ("consulta_vagas_<ano_eleicao>_<sigla_UF>");
- Votação ("detalhe_votacao_munzona_<ano_eleicao>_<sigla_UF>");
- Receitas ("receitas_candidatos_<ano_eleicao>_<sigla_UF>");
- Despesas ("despesas_candidatos_<ano_eleicao>_<sigla_UF>");

- **Modelagem Dimensional (Esquema Estrela)**

Com os questionamentos especificados e os arquivos necessários rastreados e definidos, foi possível iniciar o trabalho para definir o esquema estrela usado no desenvolvimento desse projeto.

Após o planejamento da aplicação, foi definido que seria criado um esquema estrela com onze dimensões (Localidade, Candidato, Cargo, Ocupação, Perfil, Partido, Tipo Despesa, Tipo Receita, Tempo, Tipo Bem, Situação Turno) e cinco tabelas fato (Vagas, Bens, Votação, Despesa e Receita). Este esquema resultou na Matriz de Barramento de DW (DW Bus Matrix) da Figura 17, onde as linhas representam as tabelas de fatos e as colunas representam as dimensões que descrevem os fatos (exemplo abaixo na figura 17).

	Localidade	Candidato	Cargo	Ocupação	Perfil	Partido	Tipo Despesa	Tipo Receita	Tempo	Tipo Bem	Situação Turno
Vagas	X		X						X		
Bens	X	X	X	X	X	X			X	X	
Votação	X		X						X		
Despesa		X	X				X		X		X
Receita		X	X					X	X		X

Figura 17: DW Bus Matrix

- **Dimensão Localidade**

A dimensão Localidade possui atributos para descrever os locais onde, por exemplo, ocorreram as votações. O usuário final pode filtrar as consultas por município, estado ou região do país.

Coluna	Descrição
Id_Localidade	Identificador da localidade
Regiao	Agrupamentos das unidades da federação em regiões
UF	Sigla da Unidade da Federação
Cidade	Municípios das federações

Tabela 1: Descrição da Dimensão Localidade

- **Dimensão Candidato**

A dimensão Candidato possui atributos para descrever os candidatos que concorreram às vagas eleitorais, receberam receitas e declararam despesas nas eleições.

Coluna	Descrição
Id_Candidato	Identificador do candidato
Sequencial_Candidato	Número sequencial do candidato gerado internamente pelos sistemas eleitorais. Não é o número de campanha do candidato.
Nome_Candidato_Urna	Nome de urna do candidato
Nome_Candidato	Nome completo do candidato
Idade_Data_Eleicao	Idade do candidato da data da eleição
Numero_Titulo_Eleitoral	Número do título eleitoral do candidato

Tabela 2: Descrição da Dimensão Candidato

- **Dimensão Cargo**

A dimensão Cargo possui atributos para descrever os cargos para os quais os candidatos concorreram nas eleições.

Coluna	Descrição
Id_Cargo	Identificador do cargo
Codigo_Cargo	Código do cargo a que o candidato concorre
Descricao_Cargo	Descrição do cargo a que o candidato concorre

Tabela 3: Descrição da Dimensão Cargo

- **Dimensão Ocupação**

A dimensão Ocupação possui atributos para descrever as profissões que os candidatos declararam.

Coluna	Descrição
Id_Ocupacao	Identificador da ocupação
Codigo_Ocupacao	Código da ocupação do candidato
Descricao_Ocupacao	Descrição da ocupação do candidato

Tabela 4: Descrição da Dimensão Ocupação

- **Dimensão Perfil**

A dimensão Perfil possui atributos para descrever o perfil demográfico do candidato, como por exemplo, o grau de instrução, estado civil, raça, nacionalidade e sexo dos concorrentes.

Coluna	Descrição
Id_Perfil	Identificador do perfil

Codigo_Grau_Instrucao	Código do grau de instrução do candidato. Gerado internamente pelos sistemas eleitorais.
Descricao_Grau_Instrucao	Descrição do grau de instrução do candidato
Codigo_Estado_Civil	Código do estado civil do candidato
Descricao_Estado_Civil	Descrição do estado civil do candidato
Codigo_Cor_Raca	Código da cor/raça do candidato (auto declaração)
Descricao_Cor_Raca	Descrição da cor/raça do candidato (auto declaração)
Codigo_Nacionalidade	Código da nacionalidade do candidato
Descricao_Nacionalidade	Descrição da nacionalidade do candidato
Codigo_Sexo	Código do sexo do candidato
Descricao_Sexo	Descrição do sexo do candidato

Tabela 5: Descrição da Dimensão Perfil

- **Dimensão Partido**

A dimensão Partido possui atributos para descrever os partidos e as legendas concorrentes às vagas.

Coluna	Descrição
Id_Partido	Identificador do partido
Numero_Partido	Número do partido
Sigla_Partido	Sigla do partido
Nome_Partido	Nome do partido
Codigo_Legenda	Código sequencial da legenda gerado pela Justiça Eleitoral
Sigla_Legenda	Sigla da legenda
Nome_Legenda	Nome da legenda
Composicao_Legenda	Composição da legenda

Tabela 6: Descrição da Dimensão Partido

- **Tipo Despesa**

A dimensão Tipo Despesa possui atributos para descrever os setores econômicos com os quais foram gastos os recursos.

Coluna	Descrição
Id_Tipo_Despesa	Identificador do tipo da despesa
Tipo_Despesa	Tipo da despesa
Setor_Economico_Fornecedor	Setor econômico do fornecedor

Tabela 7: Descrição da Dimensão Despesa

- **Tipo Receita**

A dimensão Tipo Receita possui atributos para descrever os doadores dos recursos das campanhas eleitorais.

Coluna	Descrição
Id_Tipo_Receita	Identificador do tipo da receita
Tipo_Receita	Tipo da receita
Nome_Doador	Nome do doador

Tabela 8: Descrição da Dimensão Receita

- **Dimensão Tempo**

A dimensão Tempo possui atributos para descrever os anos e os turnos das eleições.

Coluna	Descrição
Id_Tempo	Identificador do tipo da receita
Ano	Ano da eleição
Numero_Turno	Número do turno da eleição

Tabela 9: Descrição da Dimensão Tempo

- **Dimensão Tipo Bem**

A dimensão Tipo Bem possui atributos para descrever os bens declarados pelos candidatos.

Coluna	Descrição
Id_Tipo_Bem	Identificador do tipo do bem
Codigo_Tipo_Bem	Código do tipo do bem
Descricao_Tipo_Bem	Descrição do tipo do bem (declarado)

Tabela 10: Descrição da Dimensão Tipo Bem

- **Dimensão Situação Turno**

A dimensão Situação Turno possui atributos para descrever os resultados finais das campanhas dos candidatos, como, por exemplo, candidato não eleito.

Coluna	Descrição
Id_Situacao_Turno	Identificador da situação do turno
Codigo_Situacao_Turno	Código da situação do turno
Descricao_Situacao_Turno	Descrição da situação do turno

Tabela 11: Descrição da Dimensão Situação Turno

- **Fato Vagas**

A tabela fato Vagas possui medidas sobre as vagas disputadas em cada

eleição.

Coluna	Descrição
Id_Localidade	Chave estrangeira para a tabela de dimensão Localidade
Id_Tempo	Chave estrangeira para a tabela de dimensão Tempo
Id_Cargo	Chave estrangeira para a tabela de dimensão Cargo
Qtd_Vagas	Quantidade de vagas disponíveis na eleição

Tabela 12: Descrição do Fato Vagas

- **Fato Bens**

A tabela fato Bens possui medidas sobre os bens declarados pelos candidatos.

Coluna	Descrição
Id_Tempo	Chave estrangeira para a tabela de dimensão Tempo
Id_Perfil	Chave estrangeira para a tabela de dimensão Perfil
Id_Localidade	Chave estrangeira para a tabela de dimensão Localidade
Id_Ocupacao	Chave estrangeira para a tabela de dimensão Ocupação
Id_Cargo	Chave estrangeira para a tabela de dimensão Cargo
Id_Tipo_Bem	Chave estrangeira para a tabela de dimensão Tipo Bem
Id_Partido	Chave estrangeira para a tabela de dimensão Partido
Id_Candidato	Chave estrangeira para a tabela de dimensão Candidato
Valor_Bem	Valor do bem declarado

Tabela 13: Descrição do Fato Bens

- **Fato Votação**

A tabela fato Votação possui medidas sobre os resultados das votações por localidade.

Coluna	Descrição
Id_Tempo	Chave estrangeira para a tabela de dimensão Tempo
Id_Localidade	Chave estrangeira para a tabela de dimensão Localidade
Id_Cargo	Chave estrangeira para a tabela de dimensão Cargo
Qtd_Aptos	Quantidade de eleitores aptos a votar
Qtd_Secoas	Quantidade de seções eleitorais existentes. Considera somente seções principais.
Qtd_Comparecimento	Quantidade de eleitores que compareceram às eleições
Qtd_Abstencoes	Quantidade de eleitores que não compareceram às eleições
Qtd_Votos_Nominais	Quantidade de votos nominais
Qtd_Votos_Brancos	Quantidade de votos brancos
Qtd_Votos_Nulos	Quantidade de votos nulos
Qtd_Votos_Legendas	Quantidade de votos em legenda

Tabela 14: Descrição do Fato Votação

- **Fato Despesa**

A tabela fato Despesa possui medidas sobre as despesas declaradas pelos concorrentes das eleições.

Coluna	Descrição
Id_Candidato	Chave estrangeira para a tabela de dimensão Candidato
Id_Cargo	Chave estrangeira para a tabela de dimensão Cargo
Id_Tipo_Despesa	Chave estrangeira para a tabela de dimensão Tipo Despesa
Id_Tempo	Chave estrangeira para a tabela de dimensão Tempo
Id_Situacao_Turno	Chave estrangeira para a tabela de dimensão Situação Turno
Valor_Despesa	Valor da despesa

Tabela 15: Descrição do Fato Despesa

- **Fato Receita**

A tabela fato Receita possui medidas sobre os recursos recolhidos pelos concorrentes das eleições.

Coluna	Descrição
Id_Tipo_Receita	Chave estrangeira para a tabela de dimensão Tipo Receita
Id_Candidato	Chave estrangeira para a tabela de dimensão Candidato
Id_Cargo	Chave estrangeira para a tabela de dimensão Cargo
Id_Tempo	Chave estrangeira para a tabela de dimensão Tempo
Id_Situacao_Turno	Chave estrangeira para a tabela de dimensão Situação Turno
Valor_Receita	Valor da receita

Tabela 16: Descrição do Fato Receita

- **Desenvolvimento no Pentaho**

- **Processo de ETC**

Os dados das eleições estão disponibilizados em sua forma primária bruta, a fim de permitir o maior número possível de levantamentos e análises. Portanto, diversas agregações e derivações podem surgir a partir destes dados, dependendo somente de cada utilizador.

Os dados estão organizados em 168 arquivos do tipo CSV e um arquivo PDF com a descrição de cada coluna/tabela em cada diretório. Para este trabalho, foram utilizados os dados das eleições de 2014.

Foi escolhido o sistema de gerência de banco de dados PostgreSQL

para armazenar as tabelas resultantes da aplicação, como mostra a conexão ao *database* Dados_Eleitorais na Figura 18.

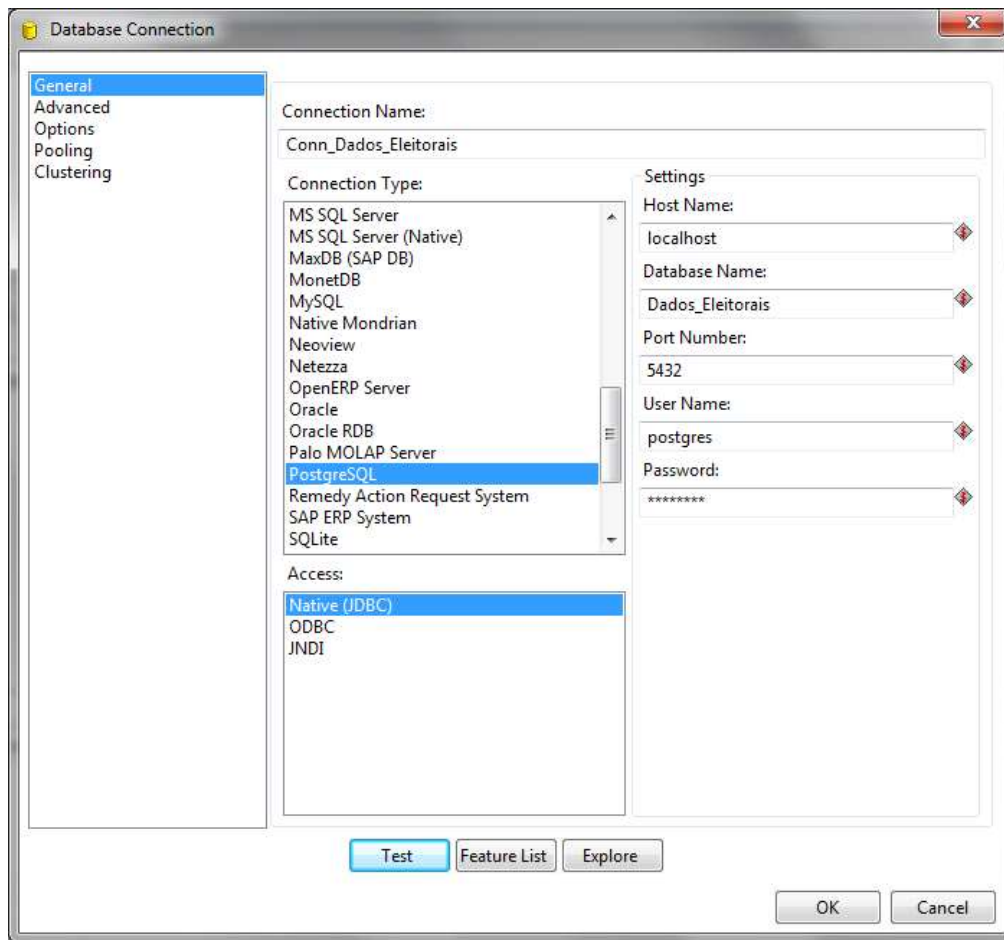


Figura 18: Conexão do PDI com o PostgreSQL via JDBC

Para a carga dos dados, primeiramente foi criada uma batch no Windows para juntar todos os arquivos CSV de uma tabela em um único arquivo. O PDI foi utilizado em seguida para popular o banco de dados, criado

no PostgreSQL, com os dados dos arquivos criados pela batch (como visto na próxima página na figura 19).

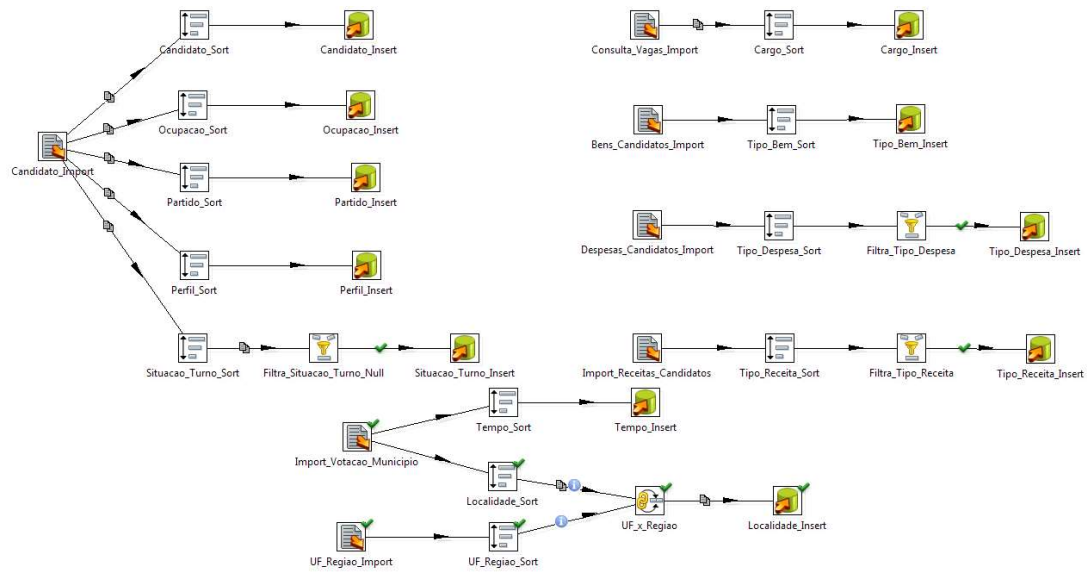


Figura 19: Processo de ETC das dimensões

A carga do banco de dados foi dividida em duas fases: na primeira foi realizada a carga das tabelas de dimensão e, em seguida, a carga das tabelas fato (como visto abaixo na figura 20).

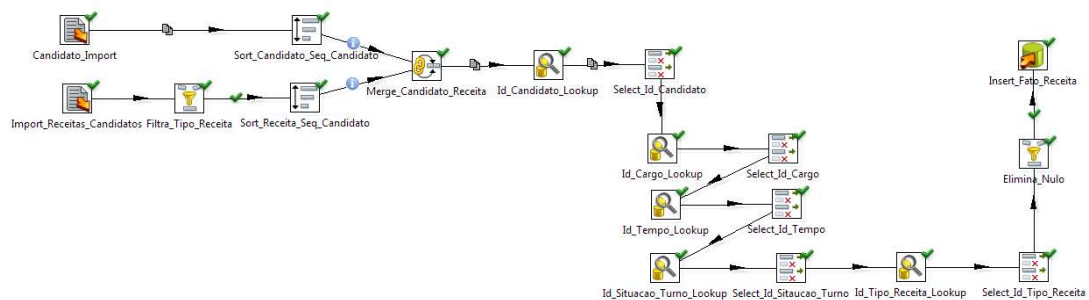


Figura 20: Processo de ETC da tabela fato Receita

- **Criação do Repositório**

A conexão do PSW com o PostgreSQL deve ser realizada antes da criação do repositório. Cabe ressaltar que, para que a conexão seja realizada com sucesso, deve existir um driver JDBC compatível com a versão do banco de dados utilizado, no diretório do Mondrian (como visto na próxima página na figura 21).

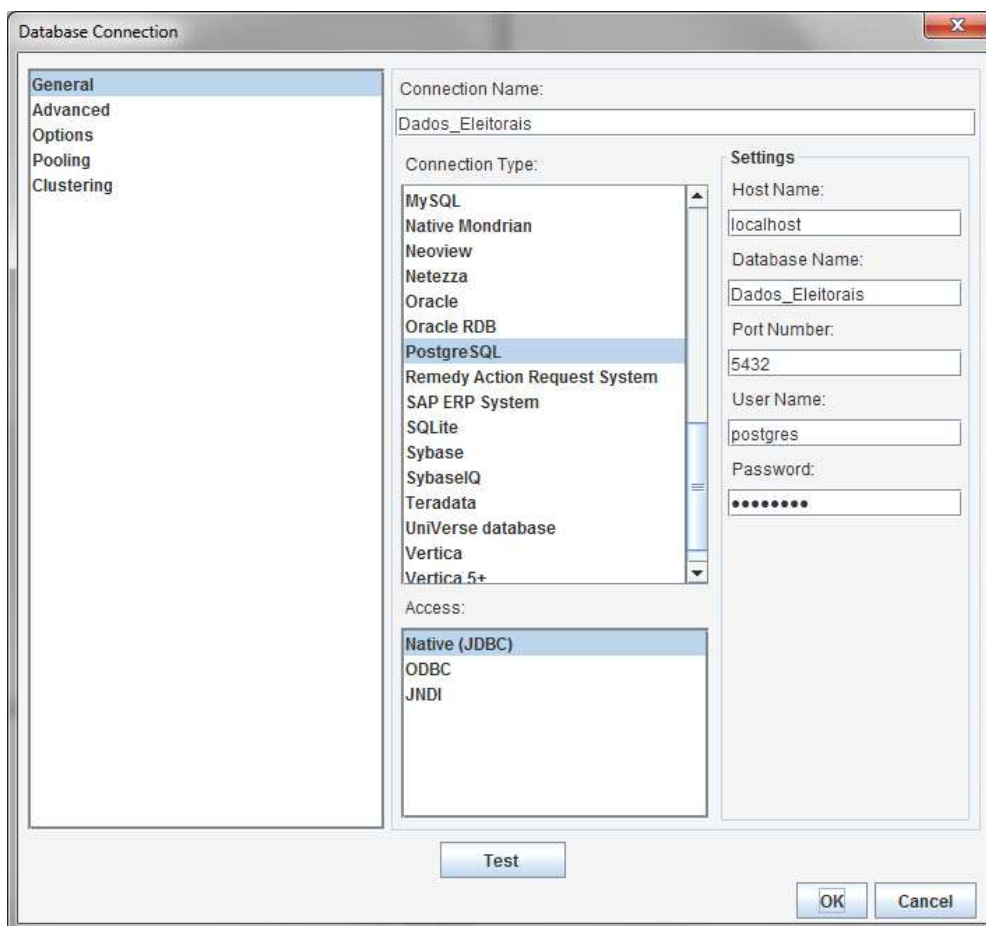


Figura 21: Conexão PSW com PostgreSQL via JDBC

O cubo foi criado tendo em vista uma tabela fato, neste caso (visto na próxima página na figura 22), o fato Receita. E a partir desta tabela, foram derivadas as dimensões mapeadas no modelo dimensional. Nesta etapa também são criadas as métricas, atributos que podem ser avaliados, como, por exemplo, receita, definidas no PSW.

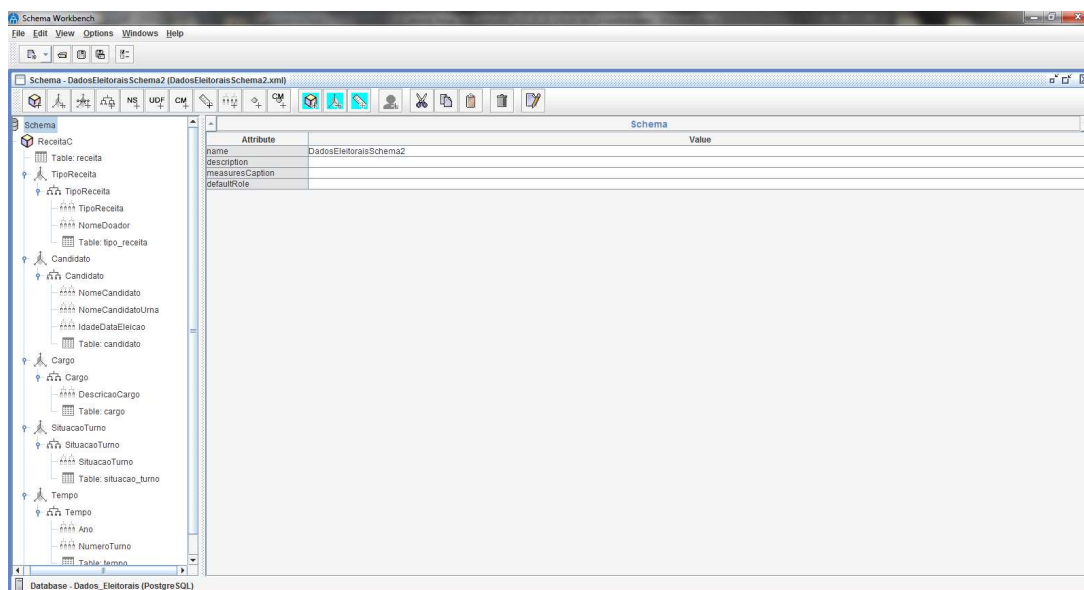


Figura 22: Criação do Cubo no PSW

- **Visualização dos dados**

As análises podem ser criadas no Pentaho CE de duas maneiras: através da ferramenta fornecida pelo próprio BI Server ou através de um plug-in denominado Saiku Analytics.

Para o desenvolvimento das análises deste trabalho, foi escolhida a ferramenta Saiku Analytics, por possuir uma interface mais amigável e de fácil manuseio. O cubo Receita foi utilizado para ilustrar as análises.

Primeiramente, seleciona-se o cubo publicado no BI Server pelo Mondrian. Após isso, será aberta uma interface com todas as dimensões e medidas criadas. Basta a definição dos campos que serão apresentados nas “Linhas” e “Colunas”, de acordo com a análise desejada. A consulta será executada automaticamente.

A adição de filtros também é possível, bastando “arrastar” a dimensão desejada para a aba “Filtros”.

Nas imagens abaixo, é respondido os seguintes questionamentos: Quais

candidatos a presidência receberam os maiores valores em doações? Quais foram seus os maiores doadores? (resultados vistos na próxima página nas figuras 23 e 24).

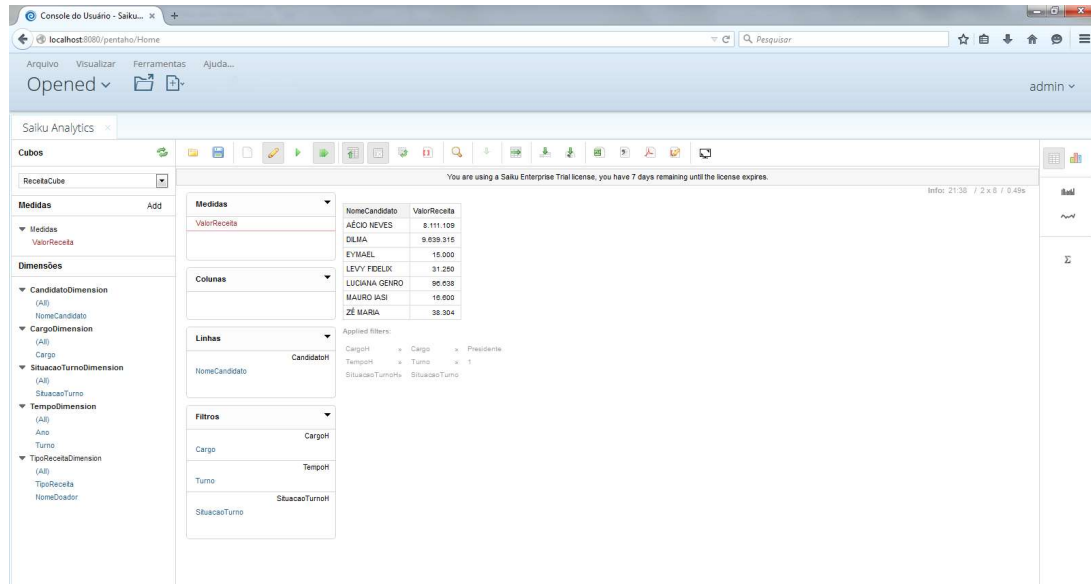


Figura 23: Valor das receitas por candidato

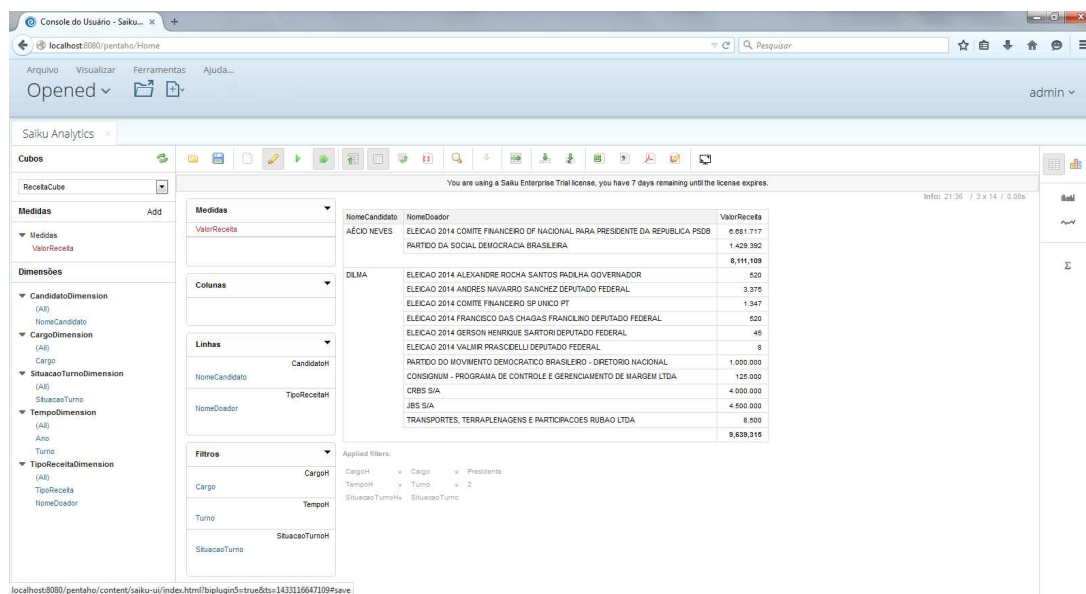


Figura 24: Doadores dos principais candidatos

No Saiku, pode-se alternar a visão de tabela, das análises previamente montadas, com a visão de gráficos. Neste plug-in, existem 10 tipos de gráficos disponíveis (vistos na próxima página nas figuras 25 e 26).

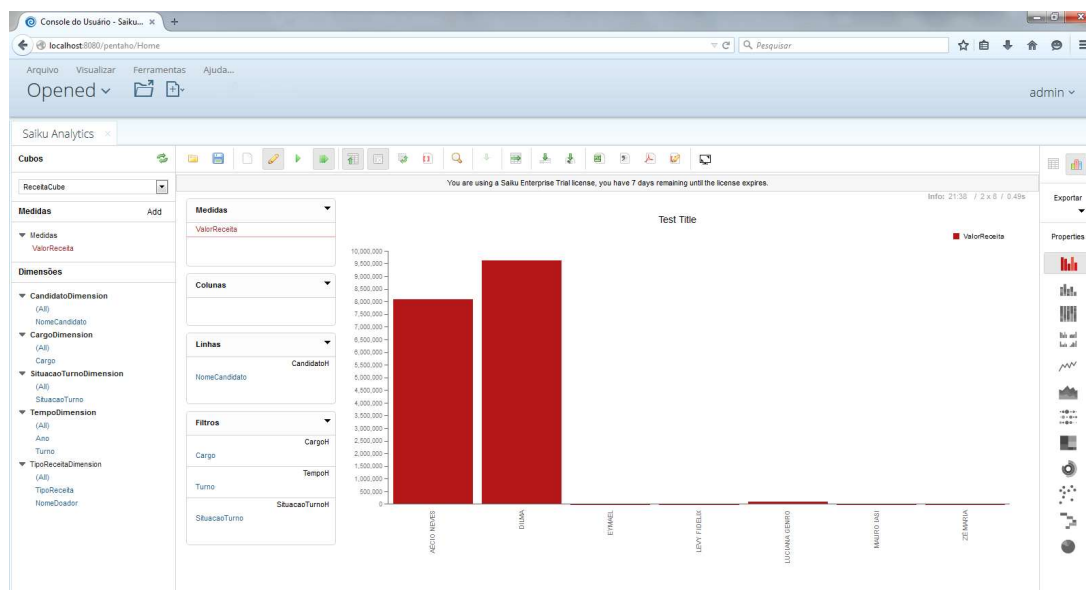


Figura 25: Gráfico com valor das receitas por candidato

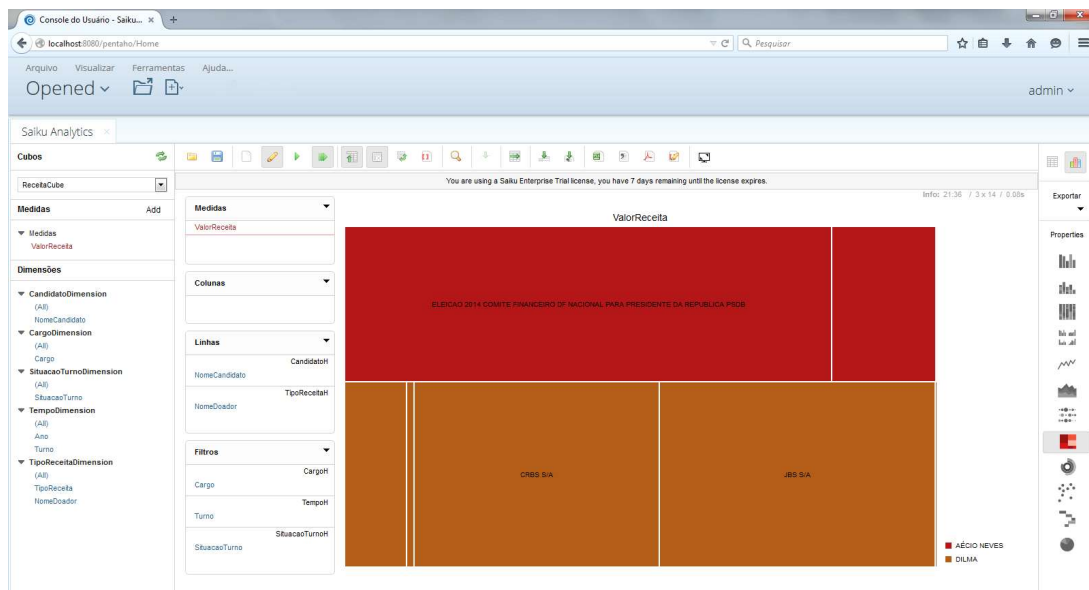


Figura 26: Gráfico com os doadores dos principais candidatos

Além disso, é possível exportar relatórios para visualização em planilhas XLS ou arquivos PDF.

Além do Saiku, existem outras ferramentas para criação de relatórios no Pentaho. Dentre elas, o Pentaho Report Designer e Dashboards no Pentaho Community. Para este último, é necessário conhecimento de linguagem HTML e CSS para desenvolver layouts.

- **Considerações Gerais sobre o Estudo de Caso**

O desenvolvimento do estudo de caso compreendeu todas as etapas previstas: levantamento de questões de análise, identificação e seleção das fontes de dados, modelagem dimensional e criação do banco de dados e a implementação na plataforma Pentaho BI Suite CE. Na etapa de implementação, o processo de ETC foi

realizado integralmente, usando o Pentaho Data Integration, assim como a carga das tabelas do banco de dados no PostgreSQL. Para a visualização das análises, foi construído, como exemplo, o cubo OLAP de Receita no Pentaho Schema Workbench, que foi publicado no Mondrian e visualizado na ferramenta Saiku Analytics disponível na suíte Pentaho CE. Outros cubos OLAP das demais tabelas fato podem ser implementado, assim como formas de visualização diferentes podem ser geradas usando outras ferramentas existentes.

Uma das maiores dificuldades na criação do DW foi entender como a ferramenta funcionava. Com a parte teórica bem fundamentada e após o entendimento da ferramenta, o projeto seguiu conforme o esperado. A ferramenta de ETL ajudou muito na importação dos dados, pois trabalhamos com diversos formatos de arquivos. A ferramenta de mais fácil aprendizado foi o Saiku, que de maneira fácil e rápida conseguimos gerar diferentes tipos de gráficos.

• Conclusão

• Considerações Finais

A área de BI vem crescendo cada vez mais dentro das organizações, tornando necessário um estudo mais aprofundado sobre a área. Ao longo desse trabalho, o assunto foi tratado em detalhes, desde a extração e o tratamento dos dados brutos, passando pelas transformações para carga, a construções de cubos OLAP para análise e chegando às informações geradas.

Transformar dados brutos disponíveis em portais governamentais em

informação não é trivial para um cidadão comum. No desenvolvimento desse projeto, foi necessária a utilização dos conhecimentos adquiridos em várias disciplinas do curso de Bacharelado em Sistemas de Informação. Na disciplina de Fundamentos de Sistemas de Informação, aprendemos o valor estratégico da informação e foi introduzido o tema de Sistemas de apoio à decisão; em Banco de Dados 1 e 2, aprendemos os conceitos, arquiteturas e linguagens de SGBD; a disciplina de Sistemas de Apoio à Inteligência de Negócios (SAIN) despertou o interesse desse projeto e foi fundamental do início ao fim, visto que nesta disciplina foram tratados temas como: modelagem dimensional, cubos OLAP, tendências de BI e ferramentas mais utilizadas.

Apesar dos conhecimentos adquiridos ao longo do curso, foram encontradas diversas dificuldades durante esse projeto, principalmente com a configuração e a utilização das ferramentas de BI. Essas dificuldades podem ser atribuídas ao fato de ter sido usada uma versão gratuita (Community Edition) da suíte Pentaho BI, cuja documentação disponível é incompleta. Neste sentido, foi essencial a utilização dos tutoriais apresentados na disciplina (SAIN).

Este trabalho teve como objetivo principal a prova de conceito do uso da ferramenta Pentaho em uma solução de BI que pudesse transformar dados governamentais brutos em gráficos de fácil entendimento, disponibilizados por meio do Portal da Transparência, em informação relevante e de fácil entendimento para a população, objetivo atingido por completo, com gráficos simples de serem observados por qualquer um. Ao final deste, foi possível visualizar de maneira amigável e compreensível informações que estimularão o senso crítico da sociedade perante as campanhas eleitorais.

Cabe ressaltar o escopo parcial da implementação, restrita ao cubo Receita. Porém, a mesma técnica pode ser replicada aos demais cubos.

• **Trabalhos Futuros**

Como sugestão de trabalho futuro, sugerimos o desenvolvimento de outras análises nos dados já utilizados, além da disponibilização dessas informações em um sítio na internet.

Com relação aos dados abertos governamentais, os conceitos, técnicas e o

processo do desenvolvimento apresentados podem ser replicados a outras bases de dados existentes no Portal da Transparência e em diversos sítios de Governos, das diferentes esferas.

• Referências Bibliográficas

Artigos e Livros

Barbieri, Carlos. BI2 – Business Intelligence: modelagem e qualidade. Rio de Janeiro: Elsevier, 2011.

Codd E.F., Codd S.B., and Salley C.T. (1993). "Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate" (PDF). Codd & Date, Inc.

D. J. Power (10 March 2007). "A Brief History of Decision Support Systems, version 4.0". DSSResources.COM.

Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (1996). "From Data Mining to Knowledge Discovery in Databases"

Inmon, W. Wiley and Sons. Building the Data Warehouse. 1st Edition, 1992.

Inmon, William. Building the Data Warehouse. Wiley Publishing, 2005.

Kimball, Ralph; Ross, Margy. The Data Warehouse Toolkit. John Wiley, 2002.

Luhn, Hans Peter. "A Business Intelligence System" (PDF). IBM Journal 2 (4): 314, 1958.

Pedroso, L., Tanaka, A. and Cappelli, C. (2013). "A Lei de Acesso à Informação brasileira e os desafios tecnológicos dos dados abertos governamentais", Position Paper – SBSI.

Steven Few, Information Dashboard Design: The Effective Visual Communication of Data (O'Reilly, 2006).

Links consultados

<http://blog.professorcoruja.com/2009/03/entendendo-o-pentaho-faq.html>, Acessado em 8 de dezembro de 2014.

<http://community.pentaho.com/projects/reporting/>, Acessado em 8 de dezembro de 2014.

<http://dados.gov.br/>, Acessado em 3 de dezembro de 2014.

<http://dados.gov.br/dados-abertos/>, Acessado em 3 de dezembro de 2014.

<http://dados.gov.br/faq/>, Acessado em 3 de dezembro de 2014.

<http://dados.gov.br/sobre/>, Acessado em 3 de dezembro de 2014.

<http://eaves.ca/2009/09>, Acessado em 3 de dezembro de 2014.

http://en.wikipedia.org/wiki/Data_warehouse, Acessado em 3 de dezembro de 2014.

http://en.wikipedia.org/wiki/Extract,_transform,_load, Acessado em 4 de dezembro de 2014.

<http://en.wikipedia.org/wiki/Pentaho> em 08/12/14, Acessado em 8 de dezembro de 2014.

http://governoaberto.cgu.gov.br/no_brasil/OGP_no_Brasil.asp, Acessado em 3 de dezembro de 2014.

<http://opengovdata.org/>, Acessado em 3 de dezembro de 2014.

<http://opendefinition.org/od/>, Acessado em 3 de dezembro de 2014.

<http://www.tse.jus.br/institucional/o-tse/o-tse>, Acessado em 8 de dezembro de 2014.

<http://wiki.pentaho.com/display/EAI/.01+Introduction+to+Spoon>, Acessado em 8 de dezembro de 2014.

<http://wiki.pentaho.com/display/EAI/Carte+User+Documentation>, Acessado em 8 de dezembro de 2014.

http://wiki.softwarelivre.org/pub/PentahoBrasil/Documentos/Pentaho_3_5.pdf, Acessado em 8 de dezembro de 2014.

http://www.acessoainformacao.gov.br/lai-para-sic/sic-apoio-orientacoes/guias-e-orientacoes/guia_secaositios-sem-logo.pdf, Acessado em 4 de dezembro de 2014.

<http://www.dextra.com.br/viabilizando-projetos-de-bi-com-pentaho/>, Acessado em 8 de dezembro de 2014.

http://www.planalto.gov.br/ccivil_03/leis/lcp/lcp101.htm, Acessado em 3 de dezembro de 2014.

<http://social.technet.microsoft.com/wiki/pt-br/contents/articles/10274.data-warehouse-x-data-mart.aspx>, Acessado em 3 de dezembro de 2014.

<http://www.transparencia.gov.br/sobre/Origem.asp>, Acessado em 3 de dezembro de 2014.

<http://www.tse.jus.br/hotSites/pesquisas-eleitorais/>, Acessado em 3 de dezembro de 2014.

<http://www.pentaho.com/>, Acessado em 3 de dezembro de 2014.

http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/112527.htm, Acessado em 3 de dezembro de 2014.

http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2012/Decreto/D7724.htm, Acessado em 3 de dezembro de 2014.

<http://www.uniriotec.br/~tanaka/SAIN/01-VisaoGeral.pdf>, Acessado em 3 de dezembro de 2014.

<http://www.uniriotec.br/~tanaka/SAIN/03-OLAP&ModelagemDimensional.pdf>, Acessado em 3 de dezembro de 2014.

<http://www.uniriotec.br/~tanaka/SAIN/04-ModelagemDimensional-Avancado.pdf>,
Acessado em 3 de dezembro de 2014.

<http://www.w3c.br/divulgacao/pdf/dados-abertos-governamentais.pdf>, Acessado em 3
de dezembro de 2014.

[http://www.w3c.br/pub/GT/GrupoDadosAbertos/PDF_-_Relatorio_Reuniao20110426
.pdf](http://www.w3c.br/pub/GT/GrupoDadosAbertos/PDF_-_Relatorio_Reuniao20110426.pdf), Acessado em 3 de dezembro de 2014.

[http://www.w3c.br/pub/Materiais/PublicacoesW3C/Manual_Dados_Abertos_WEB.pd
f](http://www.w3c.br/pub/Materiais/PublicacoesW3C/Manual_Dados_Abertos_WEB.pdf), Acessado em 3 de dezembro de 2014.

<http://www.wirelessmundi.inf.br/edicao-no-8/837-entrevista>, Acessado em 3 de
dezembro de 2014.

Anexo I DDL para as Tabelas de Dimensão e Fato

```
CREATE SEQUENCE public.tipo_receita_id_tipo_receita_seq;
```

```
CREATE TABLE public.Tipo_Receita (  
    Id_Tipo_Receita INTEGER NOT NULL DEFAULT  
nextval('public.tipo_receita_id_tipo_receita_seq'),  
    Tipo_Receita VARCHAR NOT NULL,  
    Nome_Doador VARCHAR NOT NULL,  
    CONSTRAINT tipo_receita_pk PRIMARY KEY (Id_Tipo_Receita)  
);
```

```
ALTER SEQUENCE public.tipo_receita_id_tipo_receita_seq OWNED BY  
public.Tipo_Receita.Id_Tipo_Receita;
```

```
CREATE SEQUENCE public.tipo_despesa_id_tipo_despesa_seq;
```

```
CREATE TABLE public.Tipo_Despesa (  
    Id_Tipo_Despesa INTEGER NOT NULL DEFAULT  
nextval('public.tipo_despesa_id_tipo_despesa_seq'),  
    Tipo_Despesa VARCHAR NOT NULL,  
    Setor_Economico_Fornecedor VARCHAR NOT NULL,  
    CONSTRAINT tipo_despesa_pk PRIMARY KEY (Id_Tipo_Despesa)  
);
```

```
ALTER SEQUENCE public.tipo_despesa_id_tipo_despesa_seq OWNED BY  
public.Tipo_Despesa.Id_Tipo_Despesa;
```

```
CREATE SEQUENCE public.tempo_id_tempo_seq;
```

```
CREATE TABLE public.Tempo (  
    Id_Tempo INTEGER NOT NULL DEFAULT  
nextval('public.tempo_id_tempo_seq'),  
    Ano INTEGER NOT NULL,  
    Numero_Turno INTEGER NOT NULL,  
    CONSTRAINT tempo_pk PRIMARY KEY (Id_Tempo)
```

```

);

ALTER SEQUENCE public.tempo_id_tempo_seq OWNED BY public.Tempo.Id_Tempo;

CREATE SEQUENCE public.situacao_turno_id_situacao_turno_seq;

CREATE TABLE public.Situacao_Turno (
    Id_Situacao_Turno INTEGER NOT NULL DEFAULT
nextval('public.situacao_turno_id_situacao_turno_seq'),
   Codigo_Situacao_Turno INTEGER NOT NULL,
   Descricao_Situacao_Turno VARCHAR NOT NULL,
    CONSTRAINT situacao_turno_pk PRIMARY KEY (Id_Situacao_Turno)
);

ALTER SEQUENCE public.situacao_turno_id_situacao_turno_seq OWNED BY
public.Situacao_Turno.Id_Situacao_Turno;

CREATE SEQUENCE public.perfil_id_perfil_seq;

CREATE TABLE public.Perfil (
    Id_Perfil INTEGER NOT NULL DEFAULT
nextval('public.perfil_id_perfil_seq'),
   Codigo_Grau_Instrucao INTEGER NOT NULL,
   Descricao_Grau_Instrucao VARCHAR NOT NULL,
   Codigo_Estado_Civil INTEGER NOT NULL,
   Descricao_Estado_Civil VARCHAR NOT NULL,
   Codigo_Cor_Raca INTEGER NOT NULL,
   Descricao_Cor_Raca VARCHAR NOT NULL,
   Codigo_Nacionalidade INTEGER NOT NULL,
   Descricao_Nacionalidade VARCHAR NOT NULL,
   Codigo_Sexo INTEGER NOT NULL,
   Descricao_Sexo VARCHAR NOT NULL,
    CONSTRAINT perfil_pk PRIMARY KEY (Id_Perfil)
);

ALTER SEQUENCE public.perfil_id_perfil_seq OWNED BY public.Perfil.Id_Perfil;

CREATE SEQUENCE public.ocupacao_id_ocupacao_seq;

CREATE TABLE public.Ocupacao (
    Id_Ocupacao INTEGER NOT NULL DEFAULT
nextval('public.ocupacao_id_ocupacao_seq'),
   Codigo_Ocupacao INTEGER NOT NULL,
   Descricao_Ocupacao VARCHAR NOT NULL,
    CONSTRAINT ocupacao_pk PRIMARY KEY (Id_Ocupacao)
);

ALTER SEQUENCE public.ocupacao_id_ocupacao_seq OWNED BY
public.Ocupacao.Id_Ocupacao;

CREATE SEQUENCE public.partido_id_partido_seq;

CREATE TABLE public.Partido (
    Id_Partido INTEGER NOT NULL DEFAULT
nextval('public.partido_id_partido_seq'),
   Numero_Partido INTEGER NOT NULL,

```

```

        Sigla_Partido VARCHAR NOT NULL,
        Nome_Partido VARCHAR NOT NULL,
       Codigo_Legenda VARCHAR NOT NULL,
        Sigla_Legenda VARCHAR NOT NULL,
        Nome_Legenda VARCHAR NOT NULL,
        Composicao_Legenda VARCHAR NOT NULL,
        CONSTRAINT partido_pk PRIMARY KEY (Id_Partido)
    );

ALTER SEQUENCE public.partido_id_partido_seq OWNED BY public.Partido.Id_Partido;

CREATE SEQUENCE public.tipo_bem_id_tipo_bem_seq;

CREATE TABLE public.Tipo_Bem (
    Id_Tipo_Bem INTEGER NOT NULL DEFAULT
nextval('public.tipo_bem_id_tipo_bem_seq'),
    Codigo_Tipo_Bem INTEGER NOT NULL,
    Descricao_Tipo_Bem VARCHAR NOT NULL,
    CONSTRAINT tipo_bem_pk PRIMARY KEY (Id_Tipo_Bem)
);

ALTER SEQUENCE public.tipo_bem_id_tipo_bem_seq OWNED BY
public.Tipo_Bem.Id_Tipo_Bem;

CREATE SEQUENCE public.localidade_id_localidade_seq;

CREATE TABLE public.Localidade (
    Id_Localidade INTEGER NOT NULL DEFAULT
nextval('public.localidade_id_localidade_seq'),
    Regiao VARCHAR NOT NULL,
    UF VARCHAR NOT NULL,
    Cidade VARCHAR NOT NULL,
    CONSTRAINT localidade_pk PRIMARY KEY (Id_Localidade)
);

ALTER SEQUENCE public.localidade_id_localidade_seq OWNED BY
public.Localidade.Id_Localidade;

CREATE SEQUENCE public.cargo_id_cargo_seq;

CREATE TABLE public.Cargo (
    Id_Cargo INTEGER NOT NULL DEFAULT
nextval('public.cargo_id_cargo_seq'),
    Codigo_Cargo INTEGER NOT NULL,
    Descricao_Cargo VARCHAR NOT NULL,
    CONSTRAINT cargo_pk PRIMARY KEY (Id_Cargo)
);

ALTER SEQUENCE public.cargo_id_cargo_seq OWNED BY public.Cargo.Id_Cargo;

CREATE TABLE public.Votacao (
    Id_Tempo INTEGER NOT NULL,
    Id_Localidade INTEGER NOT NULL,
    Id_Cargo INTEGER NOT NULL,
    Qtd_Aptos INTEGER NOT NULL,
    Qtd_Secoos INTEGER NOT NULL,

```

```

        Qtd_Comparecimento INTEGER NOT NULL,
        Qtd_Abstencoes INTEGER NOT NULL,
        Qtd_Votos_Nominais INTEGER NOT NULL,
        Qtd_Votos_Brancos INTEGER NOT NULL,
        Qtd_Votos_Nulos INTEGER NOT NULL,
        Qtd_Votos_Legenda INTEGER NOT NULL,
        CONSTRAINT votacao_pk PRIMARY KEY (Id_Tempo, Id_Localidade,
Id_Cargo)
);

CREATE SEQUENCE public.candidato_id_candidato_seq;

CREATE TABLE public.Candidato (
        Id_Candidato INTEGER NOT NULL DEFAULT
nextval('public.candidato_id_candidato_seq'),
        Sequencial_Candidato VARCHAR NOT NULL,
        Nome_Candidato VARCHAR NOT NULL,
        Numero_Candidato INTEGER NOT NULL,
        Nome_Candidato_Urna VARCHAR NOT NULL,
        Idade_Data_Eleicao INTEGER NOT NULL,
        Numero_Titulo_Eleitoral VARCHAR NOT NULL,
        CONSTRAINT candidato_pk PRIMARY KEY (Id_Candidato)
);

ALTER SEQUENCE public.candidato_id_candidato_seq OWNED BY
public.Candidato.Id_Candidato;

CREATE TABLE public.Receita (
        Id_Tipo_Receita INTEGER NOT NULL,
        Id_Candidato INTEGER NOT NULL,
        Id_Cargo INTEGER NOT NULL,
        Id_Tempo INTEGER NOT NULL,
        Id_Localidade INTEGER NOT NULL,
        Id_Situacao_Turno INTEGER NOT NULL,
        Valor_Receita NUMERIC NOT NULL,
        CONSTRAINT receita_pk PRIMARY KEY (Id_Tipo_Receita,
Id_Candidato, Id_Cargo, Id_Tempo, Id_Localidade, Id_Situacao_Turno)
);

CREATE TABLE public.Despesa (
        Id_Candidato INTEGER NOT NULL,
        Id_Cargo INTEGER NOT NULL,
        Id_Tipo_Despesa INTEGER NOT NULL,
        Id_Tempo INTEGER NOT NULL,
        Id_Situacao_Turno INTEGER NOT NULL,
        Id_Localidade INTEGER NOT NULL,
        Valor_Despesa NUMERIC NOT NULL,
        CONSTRAINT despesa_pk PRIMARY KEY (Id_Candidato, Id_Cargo,
Id_Tipo_Despesa, Id_Tempo, Id_Situacao_Turno, Id_Localidade)
);

CREATE TABLE public.Vagas (
        Id_Localidade INTEGER NOT NULL,
        Id_Tempo INTEGER NOT NULL,
        Id_Candidato INTEGER NOT NULL,
        Qtd_Vagas INTEGER NOT NULL,

```

```

        CONSTRAINT vagas_pk PRIMARY KEY (Id_Localidade, Id_Tempo,
Id_Candidato)
);

CREATE TABLE public.Bens (
    Id_Tempo INTEGER NOT NULL,
    Id_Perfil INTEGER NOT NULL,
    Id_Localidade INTEGER NOT NULL,
    Id_Ocupacao INTEGER NOT NULL,
    Id_Cargo INTEGER NOT NULL,
    Id_Tipo_Bem INTEGER NOT NULL,
    Id_Partido INTEGER NOT NULL,
    Id_Candidato INTEGER NOT NULL,
    Valor_Bem NUMERIC NOT NULL,
    CONSTRAINT bens_pk PRIMARY KEY (Id_Tempo, Id_Perfil,
Id_Localidade, Id_Ocupacao, Id_Cargo, Id_Tipo_Bem, Id_Partido, Id_Candidato)
);

ALTER TABLE public.Receita ADD CONSTRAINT tipo_receita_receita_fk
FOREIGN KEY (Id_Tipo_Receita)
REFERENCES public.Tipo_Receita (Id_Tipo_Receita)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Despesa ADD CONSTRAINT tipo_despesa_despesa_fk
FOREIGN KEY (Id_Tipo_Despesa)
REFERENCES public.Tipo_Despesa (Id_Tipo_Despesa)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Bens ADD CONSTRAINT tempo_bens_fk
FOREIGN KEY (Id_Tempo)
REFERENCES public.Tempo (Id_Tempo)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Vagas ADD CONSTRAINT tempo_vagas_fk
FOREIGN KEY (Id_Tempo)
REFERENCES public.Tempo (Id_Tempo)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Votacao ADD CONSTRAINT tempo_votacao_fk
FOREIGN KEY (Id_Tempo)
REFERENCES public.Tempo (Id_Tempo)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Despesa ADD CONSTRAINT tempo_despesa_fk
FOREIGN KEY (Id_Tempo)
REFERENCES public.Tempo (Id_Tempo)
ON DELETE NO ACTION
ON UPDATE NO ACTION

```



```

NOT DEFERRABLE;

ALTER TABLE public.Receita ADD CONSTRAINT tempo_receita_fk
FOREIGN KEY (Id_Tempo)
REFERENCES public.Tempo (Id_Tempo)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Despesa ADD CONSTRAINT situacao_turno_despesa_fk
FOREIGN KEY (Id_Situacao_Turno)
REFERENCES public.Situacao_Turno (Id_Situacao_Turno)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Receita ADD CONSTRAINT situacao_turno_receita_fk
FOREIGN KEY (Id_Situacao_Turno)
REFERENCES public.Situacao_Turno (Id_Situacao_Turno)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Bens ADD CONSTRAINT perfil_bens_fk
FOREIGN KEY (Id_Perfil)
REFERENCES public.Perfil (Id_Perfil)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Bens ADD CONSTRAINT ocupacao_bens_fk
FOREIGN KEY (Id_Ocupacao)
REFERENCES public.Ocupacao (Id_Ocupacao)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Bens ADD CONSTRAINT partido_bens_fk
FOREIGN KEY (Id_Partido)
REFERENCES public.Partido (Id_Partido)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Bens ADD CONSTRAINT tipo_bem_bens_fk
FOREIGN KEY (Id_Tipo_Bem)
REFERENCES public.Tipo_Bem (Id_Tipo_Bem)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Bens ADD CONSTRAINT localidade_bens_fk
FOREIGN KEY (Id_Localidade)
REFERENCES public.Localidade (Id_Localidade)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Vagas ADD CONSTRAINT localidade_vagas_fk
FOREIGN KEY (Id_Localidade)

```

```

REFERENCES public.Localidade (Id_Localidade)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Votacao ADD CONSTRAINT localidade_votacao_fk
FOREIGN KEY (Id_Localidade)
REFERENCES public.Localidade (Id_Localidade)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Receita ADD CONSTRAINT localidade_receita_fk
FOREIGN KEY (Id_Localidade)
REFERENCES public.Localidade (Id_Localidade)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Despesa ADD CONSTRAINT localidade_despesa_fk
FOREIGN KEY (Id_Localidade)
REFERENCES public.Localidade (Id_Localidade)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Bens ADD CONSTRAINT cargo_bens_fk
FOREIGN KEY (Id_Cargo)
REFERENCES public.Cargo (Id_Cargo)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Despesa ADD CONSTRAINT cargo_despesa_fk
FOREIGN KEY (Id_Cargo)
REFERENCES public.Cargo (Id_Cargo)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Receita ADD CONSTRAINT cargo_receita_fk
FOREIGN KEY (Id_Cargo)
REFERENCES public.Cargo (Id_Cargo)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Votacao ADD CONSTRAINT cargo_votacao_fk
FOREIGN KEY (Id_Cargo)
REFERENCES public.Cargo (Id_Cargo)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Bens ADD CONSTRAINT candidato_bens_fk
FOREIGN KEY (Id_Candidato)
REFERENCES public.Candidato (Id_Candidato)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

```

```
ALTER TABLE public.Vagas ADD CONSTRAINT candidato_vagas_fk
FOREIGN KEY (Id_Candidato)
REFERENCES public.Candidato (Id_Candidato)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Despesa ADD CONSTRAINT candidato_despesa_fk
FOREIGN KEY (Id_Candidato)
REFERENCES public.Candidato (Id_Candidato)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.Receita ADD CONSTRAINT candidato_receita_fk
FOREIGN KEY (Id_Candidato)
REFERENCES public.Candidato (Id_Candidato)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;
```