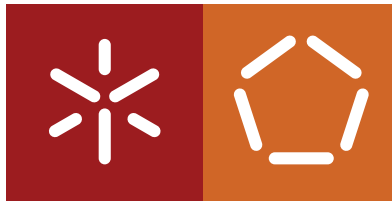


2016/2017



UNIVERSIDADE DO MINHO

MESTRADO INTEGRADO EM ENGENHARIA INFORMÁTICA

PROCESSAMENTO ANALÍTICO DE DADOS

Implementação e Exploração de um Sistema Multidimensional

Gil Gonçalves - A67738

Luis Paulo Ferreira Pedro - A70415

José Pedro Santos Monteiro - A73014

Bruno Manuel Gonçalves Ribeiro - A73269

15 de Maio de 2017

Conteúdo

1	Introdução	5
1.1	Contextualização	5
1.2	Apresentação do Caso de Estudo	5
1.3	Motivação	6
1.4	Objetivos	6
1.5	Estrutura do relatório	7
2	Levantamento e Análise de Requisitos para o Sistema Multidimensional	8
3	Mockups para discussão com as partes interessadas	10
4	Processo de Geração de Cubo	12
4.1	Descrição do processo	12
4.2	Estrutura do cubo	12
4.2.1	Medidas	13
4.2.2	Dimensões	13
4.2.3	Hierarquias	14
4.2.4	Atributos	14
4.3	Povoamento do Cubo	15
5	Implementação dos <i>Dashboards</i>	17
5.1	Contextualização Teórica	17
5.1.1	O que é um <i>Dashboards</i> ?	17
5.1.2	Tipos de <i>Dashboards</i>	17
5.1.3	<i>KPIs</i> e sua relevância	18
5.1.4	Planeamento prévio à conceção de um <i>dashboard</i>	18
5.1.5	Funcionalidades de um <i>Dashboard</i>	19
5.2	Implementação final das <i>dashboards</i>	20
6	Conclusões e Trabalho Futuro	23
	Anexos	25
i	Tabela de factos <i>Data Warehouse</i>	25
ii	Caracterização das dimensões do cubo <i>Dimensão Cliente</i>	26
iii	Caracterização das dimensões do cubo <i>Dimensão Data</i>	27
iv	Caracterização das dimensões do cubo <i>Dimensão Jogo</i>	28
v	Código XML Cubo	28

Lista de Figuras

1.1	Processo de desenvolvimento de um sistema multidimensional.	6
3.1	<i>Mockup</i> representativa do <i>dashboard</i> da área financeira.	10
3.2	<i>Mockup</i> representativa do <i>dashboard</i> da área de gestão de <i>stock</i> /negociação com produtores.	11
3.3	<i>Mockup</i> representativa do <i>dashboard</i> da área <i>marketing</i>	11
4.1	Star Schema	13
4.2	Transformação limpa cache Kettle	15
4.3	Invocação HTTP para limpeza da cache	16
5.1	<i>Dashboard</i> para a área financeira.	21
5.2	<i>Dashboard</i> para a área de gestão de stock.	22
5.3	<i>Dashboard</i> para a área de de <i>Marketing</i>	22
6.1	Caracterização da tabela de factos	25
6.2	Caracterização da Dimensão Cliente	26
6.3	Caracterização da Dimensão Data	27
6.4	Caracterização da Dimensão Jogo	28

Resumo

Ao longo do semestre anterior, nas unidades curriculares de Sistemas Operacionais e *Data Warehousing*, implementou-se um *Data Warehouse* e respetivo processo ETL para duas empresas distintas de venda de jogos online. Nesse sentido, desenvolveu-se um sistema multidimensional orientado aos requisitos das várias partes interessadas, i.e., aos vários perfis de utilização desse sistema, já apresentados no primeiro semestre.

Por conseguinte, e tendo em conta o que foi implementado no primeiro semestre, foi necessário construir um *cubo OLAP* sobre o *Data Warehouse* armazenado em *MySQL*. Assim, visto não ser possível implementar o cubo diretamente no DBMS, foi necessária a utilização da ferramenta Pentaho Mondrian, bem como o Pentaho Schema Workbench para criar o ficheiro xml com a definição do cubo. No entanto, devido a problemas relativos a não refrescamento do cubo por parte do motor analítico, foi necessária a inclusão de uma nova funcionalidade ao processo de ETL de forma a cache do Mondrian seja refrescada.

Por fim, e após serem levantados todos os requisitos impostos pelos gestores e de serem apresentadas as propostas visuais (*mockups*) para os *dashboards*, implementou-se então o cubo e posteriormente as *dashboards*, sendo que para a implementação destas foi usada a ferramenta Pentaho Business Analytics.

Área de aplicação: Construção de um sistema multidimensional para um grupo empresarial.

Palavras-Chave: *Data Warehouse*, *Dashboard*, Cubo OLAP, Sistema Multidimensional, Mondrian, Video-jogos.

Capítulo 1

Introdução

1.1 Contextualização

Recordando que outrora foi implementado um Sistema de *Data Warehousing* com o seu respetivo processo de ETL para as duas fontes, sendo uma delas o *Site XXI* que é destinada à venda de jogos online, e outra loja virtual também destinada à venda de jogos online, mas focada exclusivamente na venda de jogos retro. Atualmente a interação dos gestores com o *Data Warehouse* é assegurada pelos elementos do grupo de trabalho que executam diariamente as interrogações ao Sistema que os gestores querem ver respondidas, enviando por email para os mesmos os resultados obtidos. Logicamente que todo este processo não agrada aos gestores, sendo que o processo é efetuado com intermediários tornando-se demoroso e pouco conveniente.

Numa retrospectiva do que foi desenvolvido no semestre passado, houve uma necessidade pela parte do *Site XXI* de reunir a informação dos seus dois sites distintos de forma a ter a informação centralizada, e de facto o que foi desenvolvido respondeu a essa necessidade. O triunfo da empresa foi notório nos últimos 6 meses, o que levou a que empresa voltasse a negociar com o grupo no sentido de desenvolver plataformas para suporte à visualização de dados, de forma a que eles não fiquem tão dependentes de um engenheiro sempre que necessitam de interrogar o sistema para tomarem decisões.

Em conclusão, e depois de apresentado o contexto atual da empresa e de se recordar um pouco o setor de atividade, os gestores com o nosso auxílio concluíram acerca da necessidade de, em tempo real, se monitorizar os dados provenientes das duas fontes de dados heterogêneas da empresa, armazenados no *Data Warehouse*.

1.2 Apresentação do Caso de Estudo

De forma a tornar a interação com o *Data Warehouse* mais célere e menos dependente de terceiros, surgiu então a necessidade de implementar um sistema multidimensional que permitirá aos gestores do grupo uma melhor avaliação do desempenho das empresas em questão, isto através da definição de um conjunto de indicadores chave requeridos por cada gestor.

Pretende-se então facilitar todo o processo de gestão do grupo empresarial, na qual anteriormente dependia de um engenheiro que fazia as interrogações requeridas pelos gestores. Sendo este processo demorado, partiu-se então para a implementação de um sistema multi-

dimensional, evolução natural de um *Data Warehouse*, que permite aos vários gestores da companhia avaliar, com a frequência que acharem necessária, os indicadores chave das várias empresas que compõem o grupo. Assim, recorrendo aos seus computadores pessoais, tablets ou smartphones, estes podem aceder à *dashboard* e analisar os indicadores de desempenho da empresa, facilitando assim todo o processo de tomada de decisão.

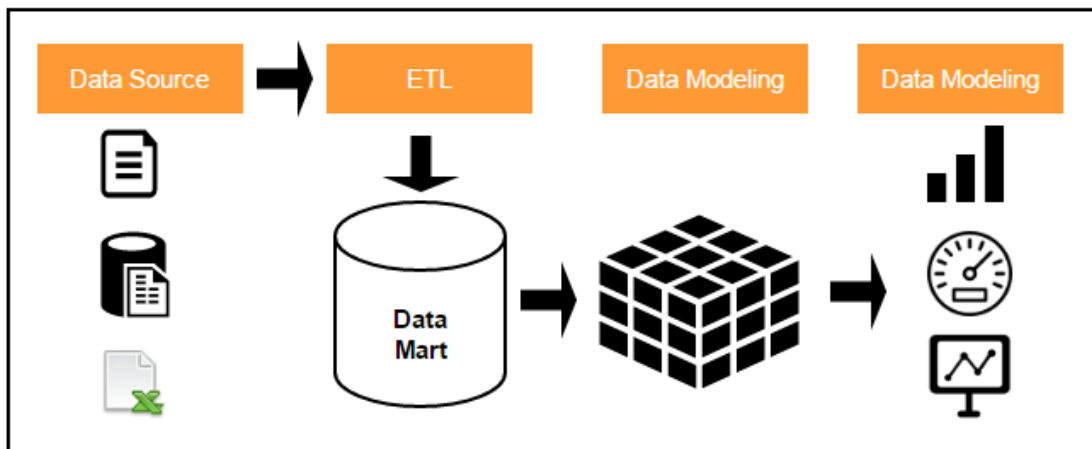


Figura 1.1: Processo de desenvolvimento de um sistema multidimensional.

1.3 Motivação

Do segundo contacto com o grupo empresarial foi desenvolvimento um *Data Warehouse* de forma a satisfazer o principal requisito imposto, que se centrava na congregação dos dados das duas empresas num só sistema de forma a facilitar todo o processo de análise à empresa, algo que anteriormente era limitativo devido ao tempo necessário para obter os dados necessários ao processo de decisão.

No entanto, surgiu um novo contacto do grupo empresarial que se mostraram preocupados com o facto de não estarem a tirar partido do sistema, visto que ainda precisavam de um engenheiro para executar manualmente todas as interrogações relevantes aos gestores. Assim, o grupo decidiu que iria ser necessário rever todo este processo de forma automatizar-lo, algo que consideraram importante para o crescimento do grupo.

Com isto pretendem ter sistema multidimensional completamente automatizado que seja capaz de trabalhar com os dados disponíveis no sistema, de forma a fornecer dados atualizados sobre o contexto atual da empresa, tornando assim o processo de tomada de decisão mais eficiente.

1.4 Objetivos

O objetivo deste projeto prende-se com a implementação das últimas camadas do processo de desenvolvimento de um sistema multidimensional. Assim, será necessário o desenvolvimento do cubo OLAP e também de uma camada de apresentação dos dados, mais propri-

amente os *dashboards*. Visto que o grupo empresarial é composto por três gestores de três diferentes áreas, irão ser desenvolvidos três diferentes *dashboards* orientados à necessidade de cada gestor. Cada *dashboard* terá de representar de forma rápida e clara o contexto atual da empresa em relação à área para a qual foi desenhado, esperando que estes sejam uma mais valia para o processo da tomada de decisão.

Assim, os principais objetivos do projeto são:

- Implementar um cubo OLAP com base no esquema multidimensional que possibilite a visualização da informação requerida graficamente;
- Garantir uma vista sobre os indicadores chave em tempo real;
- Possibilitar uma melhor interação dos gestores com os dados relevantes ao processo de tomada de decisão;
- Garantir um acesso facilitados ao indicadores chave a todas as partes interessadas;
- Garantir uma maior flexibilidade na forma como os dados relevantes são apresentados;
- Dotar o grupo empresarial de um sistema multidimensional com a capacidade de ser adaptado ao longo do tempo.

Em suma, o cumprimento dos objetivos explicitados anteriormente levará a um aumento da satisfação do cliente para com o novo paradigma da empresa, uma vez que facilitará significativamente todo o processo de análise e tomada de decisão, ao qual se espera um melhor desempenho da empresa no mercado.

1.5 Estrutura do relatório

Introdução: Capítulo introdutório do documento onde é apresentado a contextualização do trabalho, a motivação e os objetivos a que nos propomos.

Levantamento e Análise de Requisitos para o Sistema Multidimensional: Capítulo onde será feito o levantamento de requisitos de forma apurar o que irá ser implementado no projeto.

Mokups para discussão com as partes interessadas: Capítulo onde se apresenta as *mockups* das *dashboards* com base nos requisitos impostos.

Processo de Geração de Cubo: Capítulo onde é demonstrado todo o processo relativo à geração do cubo.

Implementação dos Dashboards: Capítulo onde se detalha a forma e tipo de gráficos que utilizado em cada *dashboard*.

Conclusões e trabalho futuro: Capítulo onde se apresenta o trabalho futuro e respectivas conclusões gerais do projeto.

Capítulo 2

Levantamento e Análise de Requisitos para o Sistema Multidimensional

Para o levantamento e análise de requisitos houve a necessidade de várias reuniões com os gestores e partes interessadas do grupo empresarial de forma a perceber as necessidades da empresa. Só assim se conseguiu de facto identificar as necessidades cada gestor, ao qual identificamos necessidades distintas referentes a três áreas também distintas, cada uma composta por um gestor: área financeira, área de gestão de *stock*/negociação com os produtores e a área de *marketing*.

Em relação à área financeiro, ficou claro nas reuniões realizadas que informações relativas a volumes de vendas e a lucros obtidos pelos diversos períodos de tempo possíveis de observar são importantes, isto para analisar o desempenho das empresas. Assim sendo, as informações mais importantes a representar na *dashboard* são:

- Saber qual lucro da empresa por ano/trimestre/mês;
- Saber qual lucro da empresa por dia de semana;
- Saber qual o volume de vendas por ano/trimestre/mês;
- Ter um indicador de "bem-estar" do volume de vendas.

Em relação à área de gestão de negócios e *stock*, ficou claro que era crucial negociar com os produtores de jogos, uma vez que o funcionamento do *stock* desta empresa é virtual e os jogos são adquiridos a custo zero, sendo que com a venda de um jogo a empresa entrega 10% do lucro aos produtores do mesmo. Assim sendo, e após várias reuniões chegou-se aos seguintes requisitos:

- Ter um indicador de necessidade de adquirir *stock*;
- Saber em forma de lista os jogos com falta de *stock*;
- Saber quais os produtores mais requisitados;

- Perceber quais os jogos mais requisitados.

Na última área, e entendendo que esta área é focalizada na área de *marketing*, após várias reuniões com o gestor desta percebemos quais as ferramentas que este necessita para a sua função na empresa, e com isto definiram-se o os seguintes requisitos:

- Saber quais os clientes com mais compras efetuadas/dinheiro gasto
- Quais são os jogos mais comprados
- Ter acesso a um Mapa geográfico com a dispersão dos clientes
- Saber a dispersão por género

Capítulo 3

Mockups para discussão com as partes interessadas

Sendo o resultado final do projeto apresentar ao cliente um conjunto de três *dashboards*, é necessário desenhar as *mockups* de cada *dashboard* que não vão ser mais que a representação do resultado esperado com o projeto. Assim, é notória a importância das mesmas de modo a que o cliente possa ter uma visão final do projeto, ainda que estejamos numa fase inicial.

Outro ponto importante que justifica a criação das *mockups* prende-se com o sucesso do projeto, ou seja, se de facto o projeto vai de encontro com as necessidades do cliente. Com estas é possível discutir com o cliente que tipo de representação prefere para cada requisito que impôs, bem como aspetos visuais dos próprios *dashboards*, o que permitirá reduzir a probabilidade de se ter refazer todo o processo de implementação dos *dashboards* por falta de sintonia entre o cliente e a equipa de desenvolvimento.

Assim sendo, foram desenhadas e prontamente validadas as seguintes *mockups* para cada área:

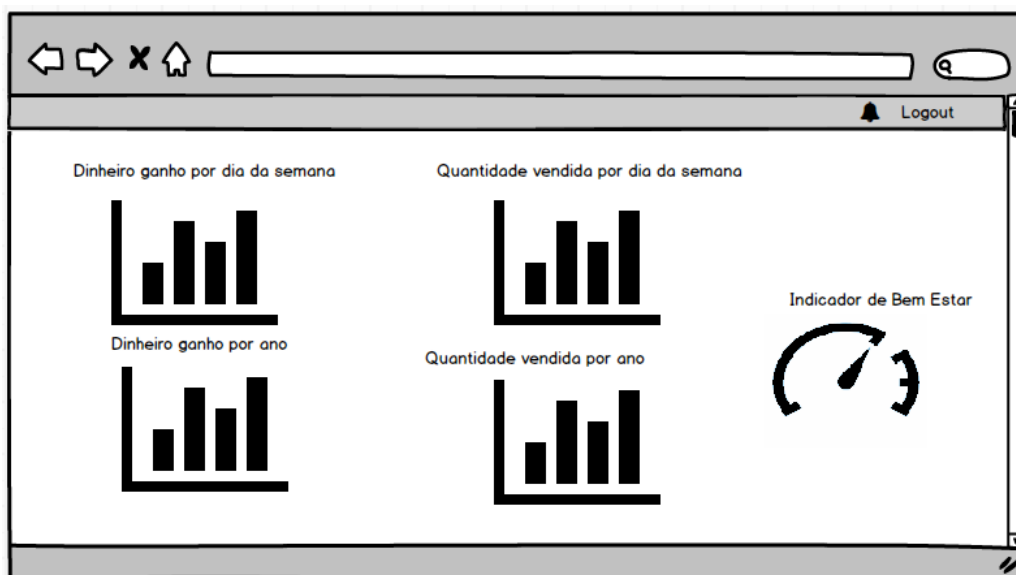


Figura 3.1: *Mockup* representativa do *dashboard* da área financeira.

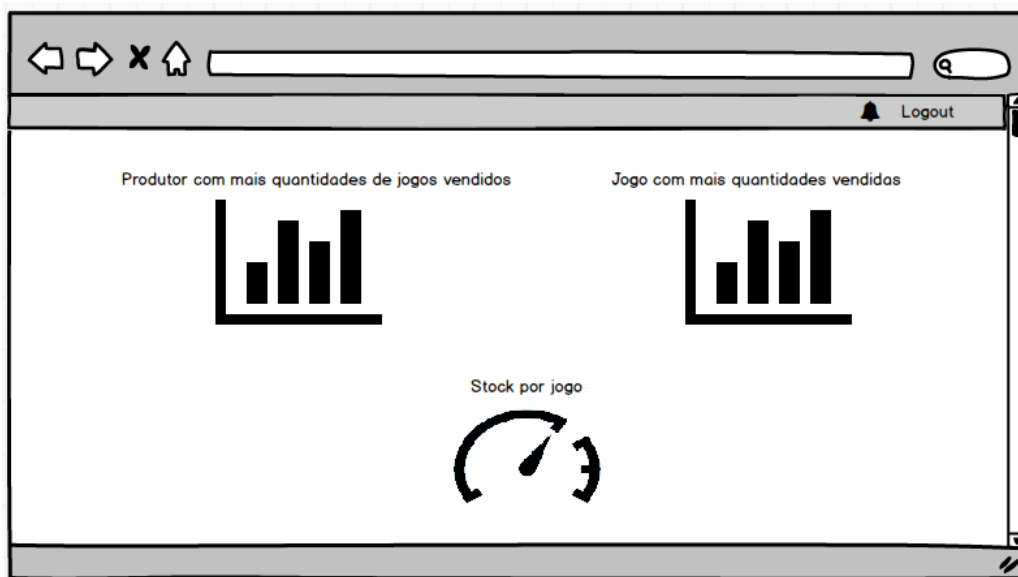


Figura 3.2: *Mockup* representativa do *dashboard* da área de gestão de *stock*/negociação com produtores.

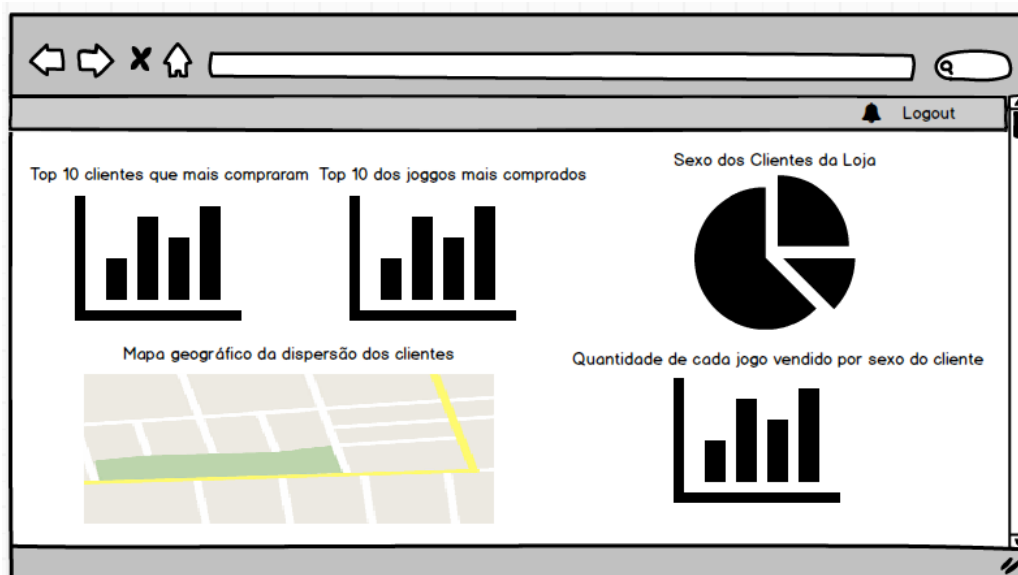


Figura 3.3: *Mockup* representativa do *dashboard* da área *marketing*.

Capítulo 4

Processo de Geração de Cubo

4.1 Descrição do processo

Numa fase inicial, anterior à implementação propriamente dita do cubo, houve por parte do grupo de trabalho uma discussão sobre qual a ferramenta a ser utilizada nesta fase da implementação do cubo e posteriormente para a criação das *dashboards*. Sendo assim, estabelecemos que estas ferramentas tinham de ser gratuitas e que tinham de garantir que realizavam de forma satisfatória o trabalho proposto.

O sistema de base de dados utilizado para este foi o *MySQL*, não havendo motivo para alterações decidiu-se manter-se tal e qual como se encontrava. Decidimos então utilizar as ferramentas *Pentaho*, mais precisamente *Pentaho Business Analytics*, uma vez que já havíamos utilizado ferramentas *Pentaho* anteriormente, como o *Kettle*, e assim achamos que seria a melhor opção continuar a explorar o software disponível. Foi ainda utilizado o *Pentaho Business Analytics* para integrar sistemas multidimensionais, no qual utiliza um motor analítico, o *Mondrian*. Este apenas necessita de um ficheiro *.xml* que explicita a estrutura do cubo.

Para a criação do ficheiro *.xml* usou-se o *Pentaho Schema Workbench* para definir as várias hierarquias presentes no *Data Warehouse* e também para definir a relação presente entre as várias dimensões e tabela de factos. Na fase final era necessário que as *dashboards* fossem em tempo real, isto é, quando o *data warehouse* sofresse alguma modificação, houvesse essa alteração nos *dashboards*. Para tal, foi necessário acrescentar um passo ao ETL implementado anteriormente na ferramenta *Kettle*, processo esse que será descrito na fase de povoamento do cubo.

4.2 Estrutura do cubo

A camada de processamento analítico (o cubo OLAP) instala-se sobre o *Data Warehouse* previamente construído, e de forma a que seja mais perceptível como se estruturou o cubo, é necessário rever o Esquema dimensional do *Data Warehouse* que está apresentado na imagem abaixo:

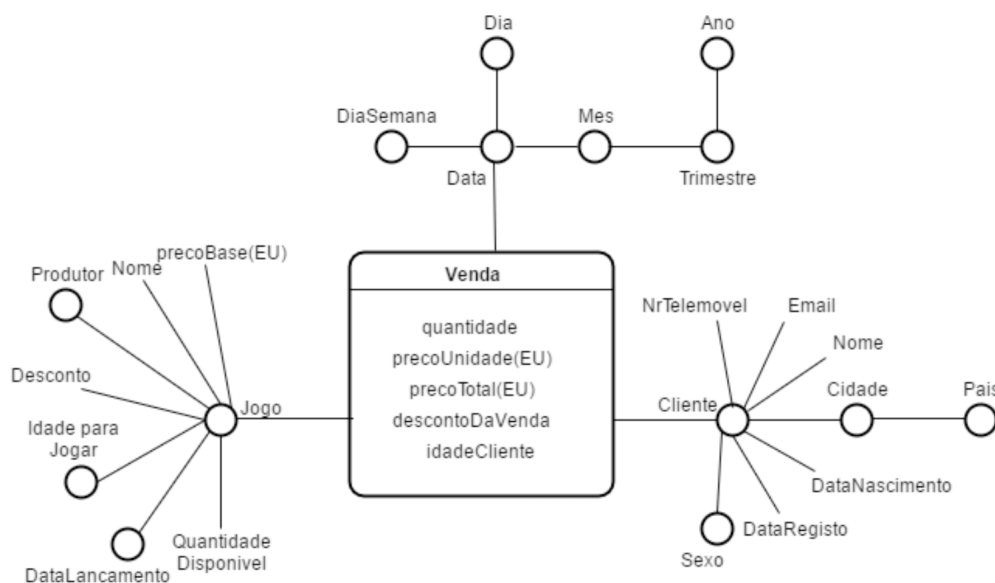


Figura 4.1: Star Schema

Na secção anterior, o processo de criação do cubo foi explicado mais direccionado para a tecnologia utilizada, sendo que neste capítulo vai ser explicado qual a estrutura do cubo e como este foi povoado. Assim sendo, e observando o esquema dimensional do *Data Warehouse*, surge a questão mais pertinente na estrutura do cubo, que é se a construção do cubo contempla a utilização de todas as dimensões e atributos. Analisando os requisitos definidos pelos gestores implicam, de uma forma ou de outra, todas as dimensões de análise que estão presentes no esquema dimensional irão ser necessárias, pelo que a estrutura do cubo irá seguir a estrutura do *Data Warehouse*. Os atributos descritivos, mesmo não sendo agregáveis, também irão ser incluídos no cubo para detalhar melhor o nível da hierarquia ao qual estejam associados, quando necessário.

4.2.1 Medidas

As medidas que vão estar presentes no cubo são as mesmas que estão presentes na tabela de factos do *Data Warehouse* (TF-Vendas). Estas medidas são:

- Quantidade
- Preço Unidade (EU/€)
- Preço Total (EU/€)
- Desconto da Venda
- Idade do Cliente

4.2.2 Dimensões

As dimensões disponíveis serão três, sendo estas:

- **Jogo:** Esta dimensão descreve todos os jogos disponíveis em ambas as lojas online.
- **Cliente:** Esta dimensão representa os dados dos clientes ao qual a venda foi efetuada.
- **Data:** Dimensão temporal.

4.2.3 Hierarquias

Todas as dimensões presentes no *Data Warehouse* têm hierarquias, pelo que é necessários representar as mesma na estrutura do cubo.

No caso da *Dimensão Cliente*:

- Hierarquia 1: id \rightarrow Cidade \rightarrow Pais \rightarrow ALL
- Hierarquia 2: id \rightarrow Sexo

No caso da *Dimensão Data*:

- Hierarquia 1: id \rightarrow Data \rightarrow Dia \rightarrow ALL
- Hierarquia 2: id \rightarrow Data \rightarrow Dia da Semana \rightarrow ALL
- Hierarquia 3: id \rightarrow Data \rightarrow Mês \rightarrow Trimestre \rightarrow Ano \rightarrow ALL

No caso da *Dimensão Jogo*:

- Hierarquia1: id \rightarrow Produtor
- Hierarquia2: id \rightarrow Idade para Jogar
- Hierarquia3: id \rightarrow Data de Lançamento

4.2.4 Atributos

Em relação aos atributos das dimensões, necessariamente estas serão as que já compunham as dimensões do esquema dimensional do *Data Warehouse*. Para melhor compreensão ver anexos.

No caso da *Dimensão Cliente* possui os seguintes atributos:

- NrTelemovel
- Email
- Nome
- Cidade
- Pais
- DataNascimento
- DataRegisto
- Sexo

No caso da *Dimensão Jogo* possui os seguintes atributos:

- PrecoBase
- Nome
- Produtor
- Desconto
- Idade para jogar
- DataLançamento
- Quantidade Disponivel

No caso da *Dimensão Data* possui os seguintes atributos:

- DiaSemana
- Dia
- Mes
- Trimestre
- Ano

4.3 Povoamento do Cubo

A fase de povoamento do cubo é feita necessariamente através do *ETL* sendo os dados extraídos dos sistemas operacionais processados e posteriormente carregados no *data warehouse*. Para corresponder à realidade, os dados ilustrados nos painéis de controlo tem que acompanhar a janela de oportunidade do *ETL* da empresa, sendo crucial que os dados sejam atualizados num processo contínuo, logo após a execução do *ETL*.

Por conseguinte, as interrogações ao cubo são feitas através da interface gráfica fornecida pelo *Pentaho*, sendo que esta ferramenta utiliza o motor analítico *Mondrian*, que recorre a mecanismos de *caching* que permitem respostas mais rápidas às *queries*. No entanto, poderão acontecer casos em que a resposta dada a uma determinada interrogação não esteja de acordo com os dados presentes no sistema multidimensional, erros estes que acontecem quando este sistema de *caching* não funciona corretamente.

Assim, tivemos a necessidade de dar a volta ao mecanismo de *caching* quando novos dados são inseridos no *data warehouse*. Desta forma, procedemos à alteração do processo de *ETL*, acrescentando uma nova transformação. Aproveitando o facto de o *Pentaho Business Analytics* disponibilizar um serviço web para limpeza da *cache*, procedemos à invocação desse mesmo serviço, como demonstrado na imagem abaixo:



Figura 4.2: Transformação limpa cache Kettle

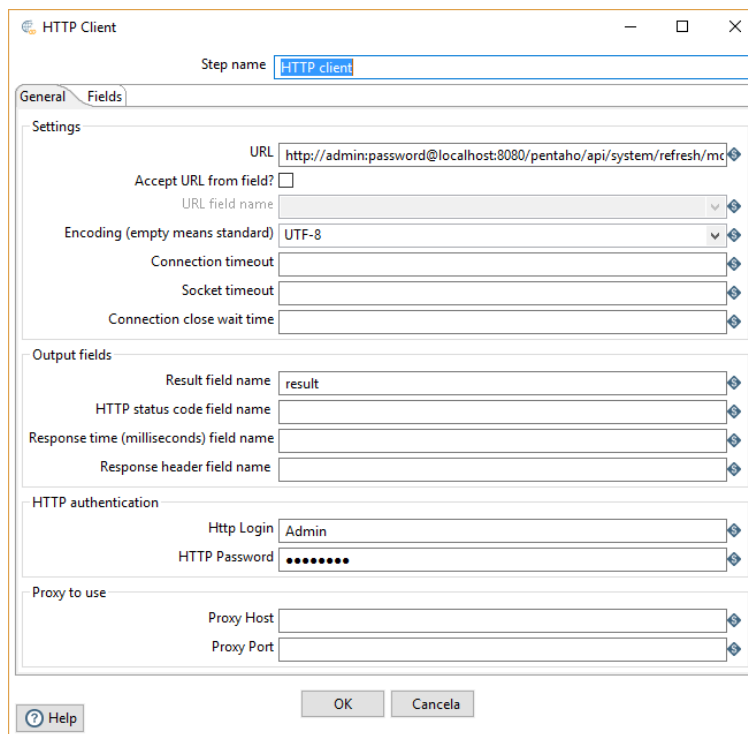


Figura 4.3: Invocação HTTP para limpeza da cache

Para testar que tudo funcionava de forma correta, foi criado um *script* de forma a injetar dados nos sistemas operacionais e perceber se tudo corria de acordo com o estipulado. Após este processo de teste, foi então iniciado o processo de instalação do cubo no sobre o *Data Warehouse* do grupo empresarial, podendo assim a equipa de desenvolvimento começar a implementação dos *dashboards*.

Capítulo 5

Implementação dos *Dashboards*

Numa tentativa de elucidar o leitor sobre as decisões tomadas e o porquê dessas decisões, vai ser apresentado neste capítulo uma contextualização teórica de onde se aplicam as *dashboards* de forma a justificar a aplicação das mesmas no trabalho prático, assim irá ser descrito o que produto final elaborado.

5.1 Contextualização Teórica

5.1.1 O que é um *Dashboards*?

Um *dashboard*, orientado à área de *Business Intelligence*, é uma ferramenta de visualização de dados que exhibe o estado atual das métricas e dos indicadores chaves de desempenho (*KPIs*) para uma dada empresa. Devido às diversas áreas em que este pode ser utilizado, um dado *dashboard* pode ser adaptado de modo a exibir métricas específicas orientadas a um determinado ponto de vista ou departamento. Qualquer que seja a utilização de um determinado *dashboard*, este deve possuir um conjunto de características fundamentais, entre as quais se salientam:

- **Simples**, permitindo comunicar com clareza os dados para o qual foi construído;
- **Focado**, apresentando informação bem orientada, não se desviando dos seus princípios e não distraindo o utilizador com outra informação;
- **Organizado em termos da sua informação** de negócio de forma a preservar o seu significado e a sua utilização;
- **Atual** em relação aos últimos meios de compreensão em termos de perceção humana relativos à apresentação visual da informação;
- **Agradável à vista**.

5.1.2 Tipos de *Dashboards*

Relativamente aos tipos de *dashboards* existentes, estes podem ser divididos em três grandes grupos:

- ***Dashboards* estratégicos**, que monitorizam o progresso de uma empresa para atingir metas pré-definidas, onde a sua visualização é acompanhada com os KPI's mais relevantes;
- ***Dashboards* analíticos/táticos** Ao contrário do *Dashboards* estratégico, estes *Dashboards* são preparados para propósitos mais detalhados, geralmente sendo usados para traçar as tendências em relação aos objetivos e iniciativas da empresa;
- ***Dashboards* operacionais** que monitorizam atividades ou variáveis que necessitam de um acompanhamento em tempo real, fornecendo informações detalhadas.

A distinção entre cada tipo de *dashboards* não resulta apenas das diferentes funções que cada um apresenta, mas também da audiência a que cada um se destina. Esta distinção em relação ao público-alvo de cada um dos tipos de *dashboards* pode ser analisada nos exemplos que se seguem:

- ***Dashboards* estratégicos**, orientados para executivos seniores;
- ***Dashboards* analíticos/táticos**, orientado para chefes de departamentos;
- ***Dashboards* operacionais**, orientado para funcionários dos vários departamentos.

5.1.3 *KPIs* e sua relevância

Um *KPI*, i.e., *Key Performance Indicator*, é um valor mensurável que avalia a posição da empresa relativamente aos objetivos chave do negócio. Os *KPIs* podem ser de mais alto nível e, daí, focarem-se no desempenho global da empresa ou, pelo contrário, focar-se num mais baixo nível e ir, por exemplo, ao nível dos vários departamentos que constituem uma companhia.

Relativamente ao *KPI* e sua relevância, este pode ter um impacto relativamente grande, se adequado ao negócio, ou, pelo contrário, um impacto diminuto, ao ponto de se questionar a razão da sua existência, se pura e simplesmente não se tiver pensado no negócio em primeiro lugar e, por isso, adotado aquilo que o mercado utiliza mais frequentemente, invés daquilo que seria mais adequado à realidade da empresa. Assim sendo, para que o cenário seja o primeiro, deve-se começar por analisar a organização e respetivos objetivos, seguido de um planeamento da forma como os mesmos poderão ser alcançados.

O processo, como em qualquer processo no contexto de *dashboarding* deverá ser iterativo.

5.1.4 Planeamento prévio à conceção de um *dashboard*

Existem algumas “regras” que devem ser tidas em atenção antes da conceção de um determinado *dashboard*, a destacar:

- 1- Identificar quem é que se pretende impressionar;
- 2- Selecionar o tipo mais correto de *dashboard*;
- 3- Agrupar os dados de forma lógica, fazendo uma utilização sensata do espaço;

- 4- Fazer com que os dados sejam relevantes para a sua audiência;
- 5- Organizar da melhor forma o *dashboard* apresentando apenas as métricas mais importantes;
- 6- Determinar a frequência com que os dados precisam de ser realmente refrescados.

O correto cumprimento destas regras fará com que qualquer *dashboard* se torne num produto útil para as partes interessadas, capaz de responder às suas necessidades.

5.1.5 Funcionalidades de um *Dashboard*

Como os *dashboards* são elementos de visualização de dados interativos, estes possuem diversas funcionalidades que possibilitam a:

- Incorporação de filtros de seleção;
- Definição de linhas temporais;
- Seleção e alteração do modelo de representação gráfica da informação;
- Navegação, de diferentes tipos, sobre os dados;
- Operação em aplicações em tempo real ou próximas de tempo real.

Com as funcionalidades apresentadas anteriormente, facilmente se percebe o porquê dos *dashboards* serem maciçamente usados por uma grande parte das empresas, facilitando o processo de visualização dos dados e, consequentemente, os processos de tomada de decisão.

5.2 Implementação final das *dashboards*

Depois de toda esta contextualização teórica foram criados, tal como já mencionado anteriormente, três *dashboards* uma para cada área da empresa. Os *dashboards* criados tiveram como base os requisitos definidos anteriormente, i.e., os perfis de utilização, bem como os *mockups* prévios à implementação.

Assim, descreve-se seguidamente, de uma forma particular, os componentes que integram cada *dashboard*.

A *dashboard* para o área financeira contempla as seguintes características:

- 1- Um gráfico de barras onde o gestor financeiro consegue visualizar os dias da semana onde é gasto maior volume de dinheiro;
- 2- Um gráfico de barras onde o gestor financeiro consegue visualizar os dias da semana onde é comprado uma maior quantidade de jogos;
- 3- Um gráfico de barras onde o gestor financeiro consegue visualizar o dinheiro gasto ao longo dos anos;
- 4- Um gráfico de barras onde o gestor financeiro consegue visualizar a quantidade de jogos comprados ao longo dos anos;
- 5- Um *KPI* onde o gestor financeiro é capaz de escolher o ano e assim ver a atual situação da empresa de uma maneira mais agradável.

A *dashboard* para a área de gestão de stock contempla as seguintes características:

- 1- Um gráfico de barras onde o gestor de stock consegue visualizar os dez produtores com mais jogos vendidos;
- 2- Um gráfico de barras onde o gestor de stock consegue visualizar os dez jogos com mais quantidade vendida;
- 3- Um *KPI* onde o gestor de stock consegue visualizar a quantidade disponível de um jogo.

A *dashboard* para a área de *marketing* contempla as seguintes características:

- 1- Um gráfico de barras onde o gestor de *marketing* consegue visualizar o dez clientes que mais compram fazem;
- 2- Um gráfico de barras onde o gestor de *marketing* consegue visualizar os jogos mais comprados;
- 3- Um gráfico circular onde o gestor de *marketing* consegue visualizar a dispersão de géneros dos clientes que fazem compras na loja online;
- 4- Um mapa que permite ao gestor de *marketing* visualizar a dispersão por áreas dos clientes que fazem compras na loja online;
- 5- Um gráfico de barras onde o gestor de *marketing* consegue visualizar os dez jogos mais comprados por género dos clientes.

Para terminar, apresentam-se *screenshots* dos *dashboards* finalmente implementados.

Com as *dashboards* elaboradas para a área de gestão financeira, fica claro que o gestor terá agora acesso à informação que pretendia quando nos abordou. Gráficos de barras para fácil compreensão sobre os volumes de vendas e lucro efetuado. Para além disso também lhe é apresentado um indicador de bem estar por ano tornando assim mais apelativa e simples saber como está a correr o ano a nível monetário.

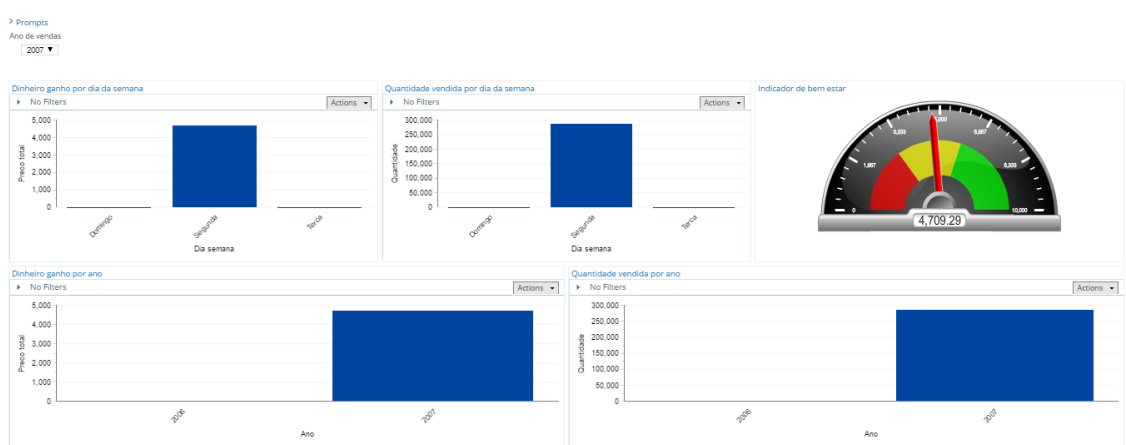


Figura 5.1: *Dashboard* para a área financeira.

O gestor de stock consegue visualizar de forma expressiva quando um dado jogo precisa de ser reabastecido. Se em stock o jogo tiver menos que quinhentas unidade, então o indicador irá estar posicionado na cor vermelha. Se tiver entre quinhentas e mil e quinhentas unidades então o indicador irá se posicionar na cor amarela. A cima desse valor irá se posicionar na cor verde.

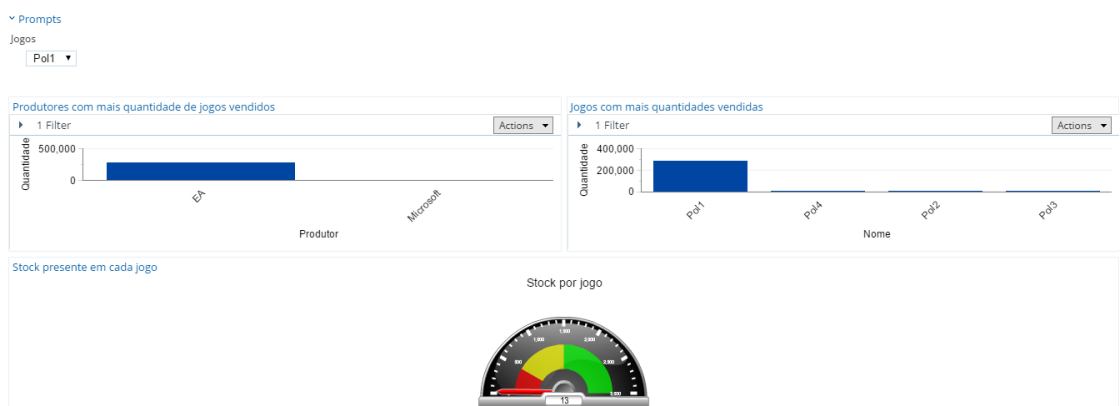


Figura 5.2: *Dashboard* para a área de gestão de stock.

O gestor de marketing vai conseguir agora ter as informações relativas aos países dos clientes, para saber onde deve promover os seus *websites*, vai aumentar a precisão desta área de negócios de forma a ajudar criar campanhas publicitárias. Vai-lhes permitir saber quais os clientes que mais compram e qual o sexo, isto de forma bastante simples e de fácil compreensão, indo de acordo com o discutido nas reuniões onde houve o levantamento dos requisitos.

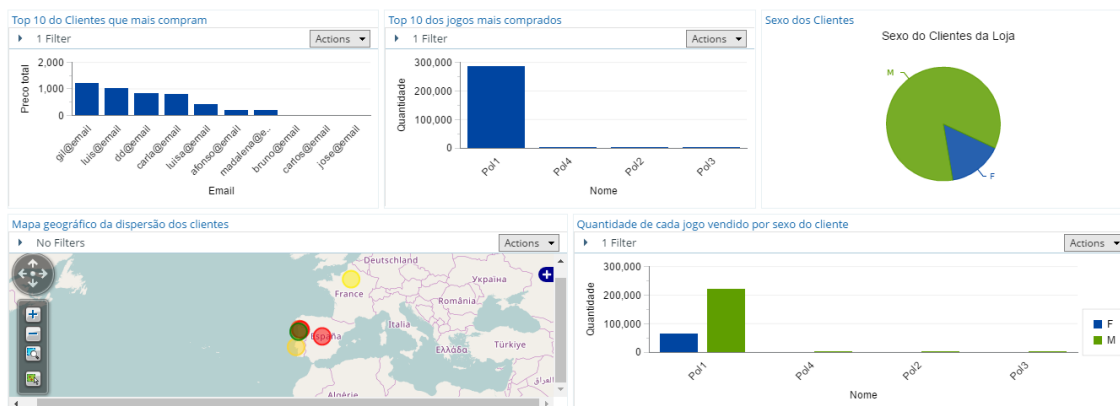


Figura 5.3: *Dashboard* para a área de de Marketing.

Capítulo 6

Conclusões e Trabalho Futuro

Após a realização do projeto, é necessário identificar os aspetos positivos e negativos detetados ao longo de todo o processo. Um dos pontos positivos prende-se ao facto de termos utilizado apenas ferramentas Pentaho, pois facilitou todo o processo de desenvolvimento, desde adquirir licenças de teste, facilidade de instalação, um vasto conjunto de documentação e também uma grande comunidade de utilizadores que fornecem um conjunto alargado de tutoriais, entre outros. Assim, foi possível, por exemplo, a inclusão de um mapa geográfico, algo que era difícil de desenvolver se a documentação e a comunidade online não fossem relevantes.

Outro ponto negativo prende-se com a utilização do MySQL como motor de base de dados, pois este não permite geração automática de estruturas multidimensionais como os cubos OLAP. No entanto, este ponto foi facilmente ultrapassado através da utilização do Pentaho Schema Workbench, que permitiu não só construir de forma simples e intuitiva o ficheiro .xml com a estrutura do cubo, bem como corrigir ficheiros já existentes, o que se revelou crucial para o sucesso do projeto.

Já com o cubo OLAP embebido no servidor, todo o processo de execução de *queries* MDX mostrou-se bastantes simples, bastando de forma bastante intuitiva arrastar as medidas e respetivos atributos das hierarquias para rapidamente se obter os resultados pretendidos. No entanto, este apenas se mostra útil para modelos de negócio pouco complexos, visto que não conta com os *widgets* mais recentes na área dos *dashboards*. Ainda assim esta pode ser uma limitação do facto de estar a ser utilizada uma licença de teste, tendo este ponto de ser posteriormente explorado.

Em suma, o trabalho correu como esperado, sendo que tanto o cubo OLAP como as *dashboards* cumpriram os requisitos impostos pelos gestores do grupo empresarial.

Bibliografia

Pentaho, 2016a. Data Integration — Pentaho Data Integration Platform. Available at: <https://www.pentaho.com/product/data-integration>

Pentaho, 2016b. Pentaho Mondrian Documentation. Available at: <http://mondrian.pentaho.com/documentation/schema.php>

<http://www.bidashboard.org/types.html>

Refresh mondrian cache on the ba server. Available at: <http://helicaltech.com/mondrian-cache-refresh-automation-mechanism-in-pentaho-ba-server/>

Ribeiro, B., Monteiro, J., Gonçalves, G., and Pedro, L. (2017). BIGS.

Anexos

i Tabela de factos *Data Warehouse*

Caracterização da tabela de factos					
Identificação		Ft_VendasJogos			
Descrição		Tabela que acolhe os vários registos de vendas de jogos realizados nas várias lojas do “site XXI”.			
Data <i>mart</i>		Comercial			
Tipo		Transacional			
Utilidade estratégica		Incentivar a venda de jogos. Estabelecer um ranking de clientes para ações promocionais. Identificar quais os jogos mais vendidos. Identificar quais as faixas etárias que mais compram os nossos jogos.			
Povoamento		Realizado diariamente entre a uma horas e as sete horas.			
Dimensão inicial					
Crescimento		0.05% mês.			
Período de dados		Desde o ano de 2013. Os anos anteriores ficarão em arquivos.			
Atributos					
Dimensões					
Nr	Identificação	Chave	Domínio	Descrição	Exemplo
1	IdCliente	S	Inteiro	Código interno do cliente da loja “site XXI”.	1
2	IdJogo	S	Inteiro	Código interno referente ao jogo da loja “site XXI”.	1
3	IdData	S	Inteiro	Código da data referente a data em que o jogo foi comprado.	1
Medidas					
Nr	Identificação	Domínio	Descrição	Exemplos	
1	QuantidadeComprada	Inteiro	Número de jogos vendidos.	2	
2	PrecoCompradoUnidade	Decimal(10,5)	Preço individual do jogo.	10	
3	PrecoTotal	Decimal(10,5)	Preço total da compra.	12	
4	DescontoVenda	Decimal(10,5)	Desconto que obteve com a compra.	0.3 ou seja teve um desconto de 30%.	
5	IdadeCliente	Inteiro	Idade que o cliente tinha quando efetuou a compra.	20	
Índice					
Nr	Identificação		Tipo	Descrição	
1	IdVenda		Primário	Único, ordenado fisicamente(<i>clustered</i>) de forma crescente.	
2	IdCliente		Secundário	Ordenado de forma crescente.	
3	IdJogo		Secundário	Ordenado de forma crescente.	
4	IdData		Secundário	Ordenado de forma crescente.	
Perfis de Utilização					
Administrador da base de dados e gestores da loja					
Observações					
Todos os valores considerados nos atributos medida são em Euros (€). Qualquer valor relativo a venda na loja do “site XXI” que não seja situada na zona euro, deve primeiramente ser convertida em euros.					

Figura 6.1: Caracterização da tabela de factos

ii Caracterização das dimensões do cubo *Dimensão Cliente*

Caracterização da dimensão					
Identificação	DimCliente				
Descrição	Clientes do siteXXI				
Tipo	Com variação e história(tipo 4)				
Crescimento	18250/ano((10+15+30)*365)				
Atributos					
Nr	Identificação	Descrição	Dominio (Tamanho)	Variação [(S)im/(N)ão]	Exemplos
1	nrTelemovel	Telemóvel do Cliente	String	S	91999998
2	Email	Email do Cliente	String	N	email@email.com
3	Nome	Nome do Cliente	String	N	Gil Marques
4	Cidade	Cidade do Cliente	String	S	Paços de Ferreira
5	País	País do Cliente	String	S	França
6	DataNascimento	Data Nascimento do Cliente	Date	N	23/04/99
7	DataRegisto	Data em que o cliente se Registou na Loja	Date	N	22/08/15
8	Sexo	Sexo do Cliente	Char	N	M
9	id	Código interno para a identificação do cliente	inteiro	N	12
Hierarquia(Ramos)					
Nr	Identificação		Esquema		
1	H1		Id->Cidade -> País -> ALL		
2	H2		Id->Sexo->ALL		
Perfis de utilização					
Administradores gerais da loja					
Observações					
Nada a assinalar					

Figura 6.2: Caracterização da Dimensão Cliente

iii Caracterização das dimensões do cubo *Dimensão Data*

Caracterização da dimensão					
Identificação	DimData				
Descrição	Data das Vendas da loja do site XXI				
Tipo	Sem variação e sem história				
Crescimento	18250/ano((10+15+30)*365)				
Atributos					
Nr	Identificação	Descrição	Dominio (Tamanho)	Variação [(S)im/(N)ão]	Exemplos
1	Id_Data	Identificador único da data	Int	N	1
2	Dia da Semana	Dia da semana que foi feita a compra	String	N	Segunda
3	Mês	Mês em que foi feito a compra	Int	N	1
4	Trimestre	Trimestre em que foi feita a compra	Int	N	2
5	Dia	Dia em que foi feito a compra	Int	N	1
6	Ano	Ano em que foi feita a compra	Int	N	2008
Hierarquia(Ramos)					
Nr	Identificação	Esquema			
1	H1	Data->Dia-> ALL			
2	H2	Data->Mês->Trimeste->Ano>ALL			
3	H3	Data->Dia da Semana->ALL			
Perfis de utilização					
Administradores gerais da loja					
Observações					
Nada a assinalar					

Figura 6.3: Caracterização da Dimensão Data

iv Caracterização das dimensões do cubo *Dimensão Jogo*

Caracterização da dimensão					
Identificação	DimJogo				
Descrição	Jogos do siteXXI				
Tipo	Com variação e sem história (tipo 1)				
Crescimento	18250/ano((10+15+30)*365)				
Atributos					
Nr	Identificação	Descrição	Dominio (Tamanho)	Variação [(S)im/(N)ão]	Exemplos
1	Id_jogo	Identificador único do jogo	Int	N	1
2	PrecoBase	Preço do jogo	Decimal(5,3)	S	10.4
3	Nome	Nome do Jogo	String	N	Fifa 17
4	Produtor	Produtor do Jogo	String	N	EA
5	Desconto	Desconto do Jogo	Decimal(5,3)	S	0.4
6	IdadeParaJogar	Idade mínima para o jogador comprar o jogo	Int	N	10
7	DataLancamento	Data em que o jogo saiu	Date	N	22/08/15
8	QuantidadeDisponivel	Quantidade disponível do jogo apresenta na loja	Int	S	20
Hierarquia(Ramos)					
Nr	Identificação	Esquema			
1	H1	IdJogo->DataLancamento -> ALL			
2	H2	IdJogo->IdadeParaJogar->ALL			
3	H3	IdJogo->Produtor->ALL			
Perfis de utilização					
Administradores gerais da loja					
Observações					
Nada a assinalar					

Figura 6.4: Caracterização da Dimensão Jogo

v Código XML Cubo

```
<Schema name="localhost">
  <Dimension visible="true" highCardinality="false" name="Dim cliente">
```

```

<Hierarchy name="Cidade" visible="true" hasAll="true"
  ↳ primaryKey="id_cliente">
    <Table name="dim_cliente">
    </Table>
    <Level name="pais" visible="true" column="pais" type="String"
      ↳ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        <Annotations>
          <Annotation name="Data.Role">
            <![CDATA[Geography]]>
          </Annotation>
          <Annotation name="Geo.Role">
            <![CDATA[country]]>
          </Annotation>
        </Annotations>
      </Level>
    <Level name="cidade" visible="true" column="cidade" type="String"
      ↳ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        <Annotations>
          <Annotation name="Data.Role">
            <![CDATA[Geography]]>
          </Annotation>
          <Annotation name="Geo.Role">
            <![CDATA[city]]>
          </Annotation>
          <Annotation name="Geo.RequiredParents">
            <![CDATA[country]]>
          </Annotation>
        </Annotations>
      </Level>
    <Level name="id_cliente" visible="true" column="id_cliente"
      ↳ type="Numeric" uniqueMembers="false" levelType="Regular"
      ↳ hideMemberIf="Never">
    </Level>
  </Hierarchy>
<Hierarchy name="Data nascimento" visible="true" hasAll="true"
  ↳ primaryKey="id_cliente">
    <Table name="dim_cliente">
    </Table>
    <Level name="Data nascimento" visible="true" column="data_nascimento"
      ↳ type="Date" uniqueMembers="false" levelType="Regular"
      ↳ hideMemberIf="Never">
    </Level>
  </Hierarchy>
<Hierarchy name="Data registro" visible="true" hasAll="true"
  ↳ primaryKey="id_cliente">
    <Table name="dim_cliente">

```

```

</Table>
<Level name="Data registro" visible="true" column="data_registro"
  ↳ type="Date" uniqueMembers="false" levelType="Regular"
  ↳ hideMemberIf="Never">
</Level>
</Hierarchy>
<Hierarchy name="Email" visible="true" hasAll="true"
  ↳ primaryKey="id_cliente">
  <Table name="dim_cliente">
  </Table>
  <Level name="Email" visible="true" column="email" type="String"
    ↳ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
  </Level>
</Hierarchy>
<Hierarchy name="Last update" visible="true" hasAll="true"
  ↳ primaryKey="id_cliente">
  <Table name="dim_cliente">
  </Table>
  <Level name="Last update" visible="true" column="last_update"
    ↳ type="Date" uniqueMembers="false" levelType="Regular"
    ↳ hideMemberIf="Never">
  </Level>
</Hierarchy>
<Hierarchy name="Nome" visible="true" hasAll="true"
  ↳ primaryKey="id_cliente">
  <Table name="dim_cliente">
  </Table>
  <Level name="Nome" visible="true" column="nome" type="String"
    ↳ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
  </Level>
</Hierarchy>
<Hierarchy name="Nr telemovel" visible="true" hasAll="true"
  ↳ primaryKey="id_cliente">
  <Table name="dim_cliente">
  </Table>
  <Level name="Nr telemovel" visible="true" column="nr_telemovel"
    ↳ type="String" uniqueMembers="false" levelType="Regular"
    ↳ hideMemberIf="Never">
  </Level>
</Hierarchy>
<Hierarchy name="Sexo" visible="true" hasAll="true"
  ↳ primaryKey="id_cliente">
  <Table name="dim_cliente">
  </Table>
  <Level name="Sexo" visible="true" column="sexo" type="String"
    ↳ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">

```

```

    </Level>
    <Level name="id_cliente" visible="true" column="id_cliente"
      ↪ type="Numeric" uniqueMembers="false" levelType="Regular"
      ↪ hideMemberIf="Never">
    </Level>
  </Hierarchy>
</Dimension>
<Dimension visible="true" highCardinality="false" name="Dim data">
  <Hierarchy name="Data" visible="true" hasAll="true" primaryKey="id_data">
    <Table name="dim_data">
    </Table>
    <Level name="Ano" visible="true" column="ano" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
    <Level name="Trimeste" visible="true" column="trimestre" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
    <Level name="Mes" visible="true" column="mes" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
    <Level name="Data" visible="true" column="data" type="Date"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Dia" visible="true" hasAll="true" primaryKey="id_data">
    <Table name="dim_data">
    </Table>
    <Level name="Dia" visible="true" column="dia" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
    <Level name="Data" visible="true" column="data" type="Date"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Dia semana" visible="true" hasAll="true"
    ↪ primaryKey="id_data">
    <Table name="dim_data">
    </Table>
    <Level name="Dia semana" visible="true" column="dia_semana"
      ↪ type="String" uniqueMembers="false" levelType="Regular"
      ↪ hideMemberIf="Never">
    </Level>
    <Level name="Data" visible="true" column="data" type="Date"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>

```

```

</Dimension>
<Dimension visible="true" highCardinality="false" name="Dim jogo">
  <Hierarchy name="Data lancamento" visible="true" hasAll="true"
    ↪ primaryKey="id_jogo">
    <Table name="dim_jogo">
    </Table>
    <Level name="Data lancamento" visible="true" column="data_lancamento"
      ↪ type="Date" uniqueMembers="false" levelType="Regular"
      ↪ hideMemberIf="Never">
    </Level>
    <Level name="id_jogo" visible="true" column="id_jogo" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Desconto" visible="true" hasAll="true"
    ↪ primaryKey="id_jogo">
    <Table name="dim_jogo">
    </Table>
    <Level name="Desconto" visible="true" column="desconto" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Idade minima" visible="true" hasAll="true"
    ↪ primaryKey="id_jogo">
    <Table name="dim_jogo">
    </Table>
    <Level name="Idade minima" visible="true" column="idade_minima"
      ↪ type="Numeric" uniqueMembers="false" levelType="Regular"
      ↪ hideMemberIf="Never">
    </Level>
    <Level name="id_jogo" visible="true" column="id_jogo" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Last update" visible="true" hasAll="true"
    ↪ primaryKey="id_jogo">
    <Table name="dim_jogo">
    </Table>
    <Level name="Last update" visible="true" column="last_update"
      ↪ type="Date" uniqueMembers="false" levelType="Regular"
      ↪ hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Nome" visible="true" hasAll="true" primaryKey="id_jogo">
    <Table name="dim_jogo">
    </Table>

```



```

    <Level name="Nome" visible="true" column="nome" type="String"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Preco" visible="true" hasAll="true"
    ↪ primaryKey="id_jogo">
    <Table name="dim_jogo">
    </Table>
    <Level name="Preco" visible="true" column="preco" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Produtor" visible="true" hasAll="true"
    ↪ primaryKey="id_jogo">
    <Table name="dim_jogo">
    </Table>
    <Level name="Produtor" visible="true" column="produtor" type="String"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
    <Level name="id_jogo" visible="true" column="id_jogo" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
  <Hierarchy name="Stock" visible="true" hasAll="true"
    ↪ primaryKey="id_jogo">
    <Table name="dim_jogo">
    </Table>
    <Level name="Stock" visible="true" column="stock" type="Numeric"
      ↪ uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
    </Level>
  </Hierarchy>
</Dimension>
<Cube name="localhost" visible="true" cache="true" enabled="true">
  <Table name="tf_venda">
  </Table>
  <DimensionUsage source="Dim cliente" name="Dim cliente" visible="true"
    ↪ foreignKey="id_cliente" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim data" name="Dim data" visible="true"
    ↪ foreignKey="id_data" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim jogo" name="Dim jogo" visible="true"
    ↪ foreignKey="id_jogo" highCardinality="false">
  </DimensionUsage>
  <Measure name="Desconto" column="desconto" formatString="###,###,###.#"
    ↪ aggregator="avg">

```

```
</Measure>
<Measure name="Idade cliente" column="idade_cliente"
  ↪ formatString="###,###,###.#" aggregator="count">
</Measure>
<Measure name="Preco total" column="preco_total" aggregator="sum">
</Measure>
<Measure name="Preco unidade" column="preco_unidade"
  ↪ formatString="###,###,###.#" aggregator="sum">
</Measure>
<Measure name="Quantidade" column="quantidade" aggregator="sum">
</Measure>
</Cube>
</Schema>
```