Especialização em *Business Intelligence*
Unidade Curricular de Data Warehouse

Ano Letivo de 2017
1º Semestre

BIGS

Business Intelligence Gaming Seller

Bruno Ribeiro (a73269)
Gil Gonçalves (a67738)
Luís Paulo Pedro (a70415)
José Pedro Monteiro (a73014)

Janeiro, 2017

Data de Recepção	
Responsável	
Avaliação	
Observações	

BIGS

Bruno Ribeiro (a73269)
Gil Gonçalves (a67738)
Luís Paulo Pedro (a70415)
José Pedro Monteiro (a73014)

Janeiro, 2017

Resumo

Este projeto foi desenvolvido no âmbito do perfil de Business Intelligence do Mestrado em Engenharia Informática na Universidade do Minho e visa produzir um sistema operacional para empresa *Site XXI*, modelar e construir um sistema de apoio à decisão, bem como apresentar todo o processo de implementação e migração de dados para o sistema de Data Warehousing.

A elaboração deste sistema é de extrema relevância desenvolver um projeto para dar suporte ao negocio da empresa que neste momento se encontra em rápido crescimento. Assim sendo, foi proposto a equipa desenvolver este projeto cujo qual a equipa denominou BIGS.

Neste relatório segue-se toda a descrição de todo o processo, decisões e planeamentos que forma tomados durante a realização do mesmo.

Área de Aplicação: Business Intelligence, Data Warehousing e Sistemas Operacionais

Palavras-Chave: Bases de Dados Relacionais, Sistemas Operacionais, Business Intelligence, Entidades, Relacionamentos, Metodologia, Modelo Conceptual, Modelo Lógico, Modelo Físico, SGBD, DBDL.

Índice

Índice de Figuras

Índice de Tabelas

1. Introdução

1.1. Enquadramento

A informação empresarial é considerada um ativo de elevada importância independentemente da dimensão da organização. No âmbito da sua atividade, seja no processo produtivo, no contacto com fornecedores e clientes ou outra etapa, as organizações geram dados que poderão ser posteriormente sistematizados e utilizados para fins diversos.

Em grandes empresas com informação em suporte digital, em modo geral, todos os seus sistemas informáticos são suportados por bases de dados, isto leva á necessidade de centralizar toda a informação relacionada com o negócio, para assim aprimorar o negócio, e ter de um acesso muito mais célere e eficiente a toda essa informação. Conseguir combinar todos estes dados, permite com que o gestor ter novas perspetivas de negocio, e assim, conseguir tomar decisões para melhor o rendimento da empresa. Assim surge a necessidade da implementação da criação de um sistema de apoio à decisão, esse sistema tem de nome *Data Warehouse.*

Um *Data Warehouse* é um sistema de computação utilizado para armazenar informações relativas às atividades de uma organização em bases de dados, de forma consolidada. O desenho da base de dados favorece os relatórios, a análise de grandes volumes de dados e a obtenção de informações estratégicas que podem facilitar a tomada de decisão. O *Data Warehouse* possibilita a análise de grandes volumes de dados, recolhidos dos sistemas tradicionais. São as chamadas séries históricas que possibilitam uma melhor análise de eventos passados, oferecendo suporte às tomadas de decisões presentes e a previsão de eventos futuros. Um sistema como este, de suporte à decisão, tem a capacidade de oferecer à empresa visão de negócio quase em tempo real, o que vai proporcionar uma melhor gestão e prospeção nos indicadores que se pretendem analisar.

A empresa optou pela criação desta base de dados devido a estar a entrar no mercado online de venda de jogos, o que levou a que o aumentou de vendas tenha sido exponencial. A empresa bracarense pretende estar presente em vários domínios, onde vê a necessidade de ter toda a informação relacionada com as vendas centralizada. A empresa *Site XXI* tem neste momento dois domínios, sendo um deles mais destinado para jogos Retro, os dados são

suportados em MongoDB, e em MySQL, os investidores por parte da empresa *Site XXI* acreditam que este é o caminho a seguir para melhorar o seu negócio. No entanto, para que todo este processo seja bem sucedido é preciso perceber a motivação e avaliar a utilidade do sistema de modo a perceber se este é o caminho a seguir para a realização do projeto.

1.2. Motivação e objetivos

Na atualidade, a empresa *Site XXI* sediada em Braga, já tem a sua relevância no mercado de venda de jogos online, já dispõe de uma vasta gama de clientes de todos os géneros e idades, com isto o volume de dados produzidos tem vindo a aumentar.

O mercado tecnológico está em constante mudança e a toda hora surgem novas empresas concorrentes à empresa Bracarense. Com a nova loja de venda de jogos Retro, a empresa *Site XXI* deixou de ter a informação centralizada numa única base de dados. Assim sendo houve a necessidade de centralizar e armazenar os dados para previsões ou correções no processo de tomadas de decisão inerentes a esta grandeza de negócio. Tendo isto em mente os gestores da empresa de Braga contactaram a equipa de gestão BIGS, com o objetivo focado no lucro global das suas duas filiais. Esta necessidade revela a disposição necessária, pela parte da chefia da empresa, para o investimento num projeto de implementação de um sistema de *Data Warehouse*. Delineou-se então suscitamento os objetivos pretendidos com a sua aplicação. Espera-se que a loja de venda de jogos retro tenha um aumento do número de clientes na ordem dos 40% no primeiro ano e 20%, e que a loja principal tenha um crescimento 20% de lucro por ano. Apesar do aumento de clientes ser importante para o *Site XXI,* tenciona-se acima de tudo, aumentar o lucro da empresa. Um objetivo crucial, é através do *Data Warehouse* incentivar os clientes atuais a fazer mais compras, levando de novo ao que é primordial em todo este processo, o aumento de lucro da empresa. Surge assim a necessidade de criar mecanismos de acesso aos dados, que permitam fazer filtragem, colocar fatores em evidência, denunciar variações, e que permitam mostrar apenas os dados que são realmente importantes. Espera-se encontrar padrões de modo a perceber que tipos de campanhas publicitárias a empresa deve investir de modo a ter o retorno financeiro que se objetiva. Para assegurar que as medidas tomadas são adequadas, existem duas reuniões intermedias no fim dos primeiros dois meses para avaliar os resultados obtidos e planear os passos seguintes, é perentório que a empresa consiga apresentar provas que todo os passos a tomar são realmente bem fundamentados.

1.3. Avaliação da utilidade do sistema

Depois das enumeras sugestões dos seus utilizadores para adicionar jogos retro a empresa *site XXI* decidiu atender ao pedido dos seus clientes. Então criou uma nova loja online e para não misturar os jogos decidiu armazenar os dados numa outra base de dados. A base de dados escolhida era uma base de dados não relacional, mais propriamente em *MomgoDB*.

Posto isto a empresa viu-se então na necessidade de criar um *Data Warehouse*, deste modo permitiu a empresa ter a informação centralizada e disponível para os gestores. Como essa informação será possível analisar o consumo dos clientes em relação aos jogos, podendo assim sugerir ao utilizador novos jogos a comprar que se enquadram nos seus hábitos de consumo aumentando assim o número de vendas e consequentemente o lucro gerado.

Será possível também saber qual a faixa etária que mais utiliza o nosso sistema podendo assim enquadrar as promoções com a idade dos utilizadores.

Será também possível saber quais são as produtoras que mais jogos vendem podendo fazer alguma parceria de forma a reduzir os preços de jogos comprados, permitindo baixar o preço de venda e tornando a empresa mais competitiva para o mercado.

1.4. Plano de desenvolvimento do sistema

Contruir um sistema *data warehouse* é deveras complexo e exige um estudo pormenorizado em relação aos custos associados ao projeto, ferramentas a usar durante o desenvolvimento do projeto, o objetivo para o qual é contruído e se em termos de investimento retorno se compensa a construção do mesmo.

Primeiramente estabelecemos um diálogo com o gerente da empresa para definir a duração do projeto, os custos associados ao projeto, as ferramentas que o gerente estava a utilizar na empresa e as ferramentas as quais teríamos de implementar o projeto.

Posto a primeira fase de negocio passamos para a segunda fase do projeto que consiste em definir o objetivo da construção do data warehouse.

Depois procedemos a terceira fase do projeto que é uma analise das fontes para constatar se os objetivos pretendidos pelo cliente eram possíveis de serem satisfeitos.

Após mais uma conversa com o gerente da loja conseguimos definir os objetivos, objetivos esses que já foram definidos em cima.

Procedemos então para a quarta fase que é a construção da tabela de factos que irá ser a tabela central do data warehouse que irá permitir aceder a toda a informação. Posto isto partimos para a quinta fase de desenvolvimento que é definir a tabelas de dimensão que irá conter a matéria associada ao processo de negócio.

O sexto passo será definir as medidas que representam as conclusões que iremos retirar do data warehouse.

Posto isto podemos proceder a construção a construção do ETL para passar a extração dos dados e tratamento dos mesmos que irão ser inseridos no nosso data warehouse.

1.5. Justificação do sistema em termos de negócio

Um D*ata Warehouse* é necessário para a empresa *Site XXI* para dar suporte à tomada de decisões estratégicas da empresa. As vendas iram ser examinadas para entender o desempenho das vendas para assim aumentar os lucros da empresa *Site XXI* no futuro e melhorar o negócio da empresa. Com a informação recolhida a empresa conseguirá fazer planos de marketing de modo atrair mais clientes, isto só é possível devido a conseguir identificar melhor o publico alvo. A identificação do publico alvo para empresa *Site XXI* é muito importante, pois no mercado jogos um jogo tem muita probabilidade de agradar um grupo muito restrito de idades, ao identificar esse o grupo através dos dados fornecidos pelo *Data Warehouse* a equipa de marketing conseguirá fazer campanhas publicitarias para apelar o interesse desses clientes.

2. Planeamento do Projeto

Este capítulo irá se focar nas considerações, nas atividades associadas com o planeamento do projeto e na forma como foi distribuído pelos diferentes elementos da equipa de trabalho. Esta parte do relatório é voltada para as pessoas que são responsáveis pelo projeto *Data Warehouse*, independentemente de serem ou não parte dos sistemas informação (SI) ou organização empresarial.

2.1. Estabelecimento da identidade do projeto

Todos os projetos necessitam de uma identidade, para isso reunimos-mos com os responsáveis pelo projeto e definiu-se que o nome seria *BIGS* (Business Intellengence Gamming Seller).

Ilustração 1: Logotipo da Empresa

2.2. Identificação dos recursos necessários

Sem os recursos necessários nunca seria possível executar o projeto, logo numa fase inicial é preciso analisar se existe os recursos disponíveis para a equipa. Uma maquina por elemento é indispensável com o software *kettle*, *word*, *mysql* e *mongoDB*. Um recurso importante é existência de transporte ou pagamento de combustível para as deslocações dos membros da equipa com o cliente.

2.3. Preparação do plano de desenvolvimento

A preparação do plano de desenvolvimento é um dos aspetos mais importantes a ter em conta, pois um bom planeamento levará a melhor distribuição da carga de trabalho, por conseguinte, horas de trabalho poupadas no futuro. Para se fazer um bom planeamento tem que se definir em primeiro lugar quais tarefas são necessárias fazer no desenvolvimento, consoante cada dificuldade apresentada por cada tarefa e o conhecimento dos membros da equipa deve-se fazer sempre um período de tempo de conclusão da tarefa alargado, para caso haja acidentes de percurso.

Fazendo este processo de reflexão em relação as tarefas necessárias chegamos ao seguinte planeamento:

Task	Start Date	Days to Complete
Construção da matriz de decisão	05/dez	1
Seleção do data mart a desenvolver	06/dez	2
Escolha do Grão	08/dez	4
Escolha das dimensões de análise	12/dez	3
Desenvolvimento do diagrama das tabelas de factos	15/dez	1
Documentar as tabelas de factos	15/dez	1
Projetar o detalhe das dimensões	16/dez	3
Identificar os possiveis candidatos de agregados armazenados previamente	19/dez	2
Desenvolver a estratégia de desenvolvimento para as tabelas de agregados	19/dez	2
Projecção do ETL com o modelo BPMN	28/dez	4
Construção do ETL com a ferramenta Kettel	01/jan	12
Documentação do processo ETL	13/jan	4

Tabela 1: Planeamento do projeto

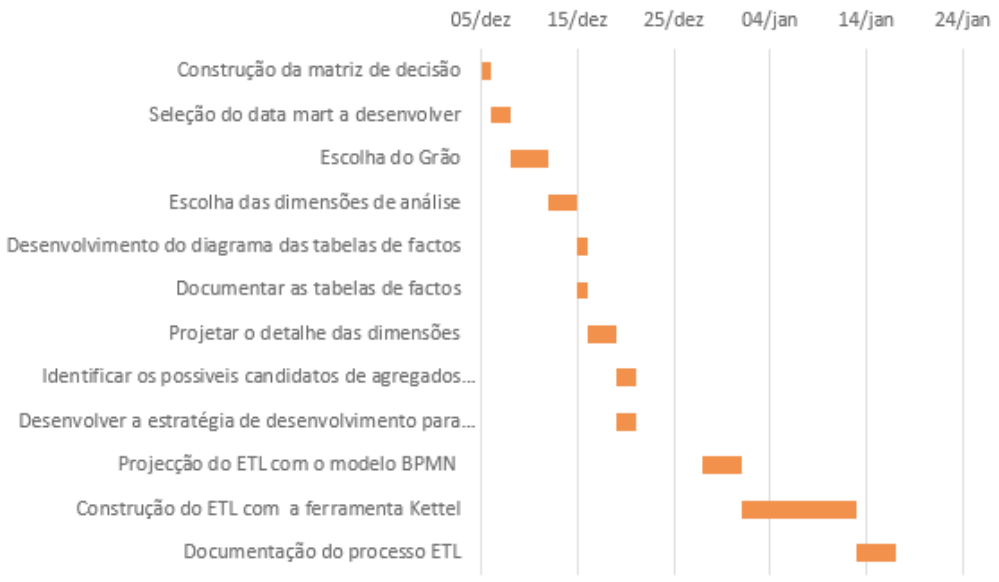


Ilustração 2: Diagrama de Gantt

2.4. Definição da equipa de desenvolvimento

A implementação de um Sistema de *Data Warehouse* numa organização obriga à constituição de equipas de desenvolvimento com formação e sensibilidade diversas, sendo esta heterogeneidade difícil de gerir. Para isso definimos uma equipa e os elementos da equipa são 4 alunos que frequentam o perfil de BI, e que no seguimento da cadeira *Data Warehouse* houve a proposta de fazer o seguinte trabalho descrito para implementação dos conhecimentos lecionados.

2.5. Definição da equipa de desenvolvimento

Depois de definido o planeamento do trabalho era necessário distribuir o trabalho pelos membros da equipa de modo distribuído para se obter sucesso na conclusão do trabalho.
O membro da equipa *Gil Gonçalves* é o proprietário da área do negócio do projeto e muitas vezes têm a responsabilidade financeira do projeto.
Os membros da equipa *Luís Pedro* e *José Pedro Monteiro* são os responsáveis pela conceção e desenvolvimento dos dados no *Data Warehouse*.
O membro da equipa *Bruno Ribeiro* é modelador dos dados do projeto sendo responsável pela performance dos detalhes da analise dos dados e é quem é responsável pelo desenvolvimento do modelo de dados dimensional.
A fase do ETL, vai ser implementada por todos os elementos do grupo de trabalho.

2.6. Estabelecimento de um conjunto de medidas de sucesso

Um *Data Warehouse* bem sucedido resulta de um negócio conjunto e de um esforço em grupo que compartilham a responsabilidade pela iniciativa. Nenhum dos grupos consegue obter um resultado com sucesso se tentarem construir um *Data Warehouse* sem o outro. De forma a desenvolver um projeto com esta envergadura, torna-se necessário definir algumas medidas de organização de forma a poder terminar o projeto com sucesso.

Desta forma, foram estabelecidas as medidas listas de seguida:

- O Data Warehouse é entregue a tempo;
- O Data Warehouse é útil;
- O preço do Data Warehouse encontra-se dentro do investimento;
- O Data Warehouse permite o retorno do investimento;
- O Data Warehouse é utilizado;
- Todos os requisitos são incorporados no Data Warehouse.

2.7. Revisão do plano de trabalho com o cliente

O planeamento do projeto foi feito em paralelo com o cliente através de várias reuniões marcadas durante o desenvolvimento, como foi referido na motivação e objetivos, existem duas reuniões intermédias no fim dos primeiros dois meses para avaliar os resultados obtidos e planear os passos seguintes havendo discussões com os responsáveis pelo Sistema de informação, negócio e os desenvolvedores de modo a todas as partes chegarem a consenso para que o projeto final agrade ambas as partes.

3. Identificação e análise das fontes de informação

De modo a se perceber o tipo de dados com qual se irá lidar e definir quais as restrições que cada fonte impõe ao processo de ETL, é feita uma caracterização das fontes onde se explora os aspetos mais importantes das mesmas.

3.1. Identificação e caracterização das fontes

Para o desenvolvimento do nosso *Data Warehouse*, a empresa *Site XXI* disponibilizou-nos duas fontes de informação uma pertencente ao site principal e outro ao site de venda de jogos retro, sendo uma relacional e uma não relacional.

3.1.1. Primeira loja de jogos online

A base de dados da loja irá conter todos os registos dos utilizadores que tenham criado conta, assim como as compras que eles efetuaram na loja e também as informações relativas aos jogos disponíveis na loja online.

A Relacional é representada pelo seguinte modelo Lógico:

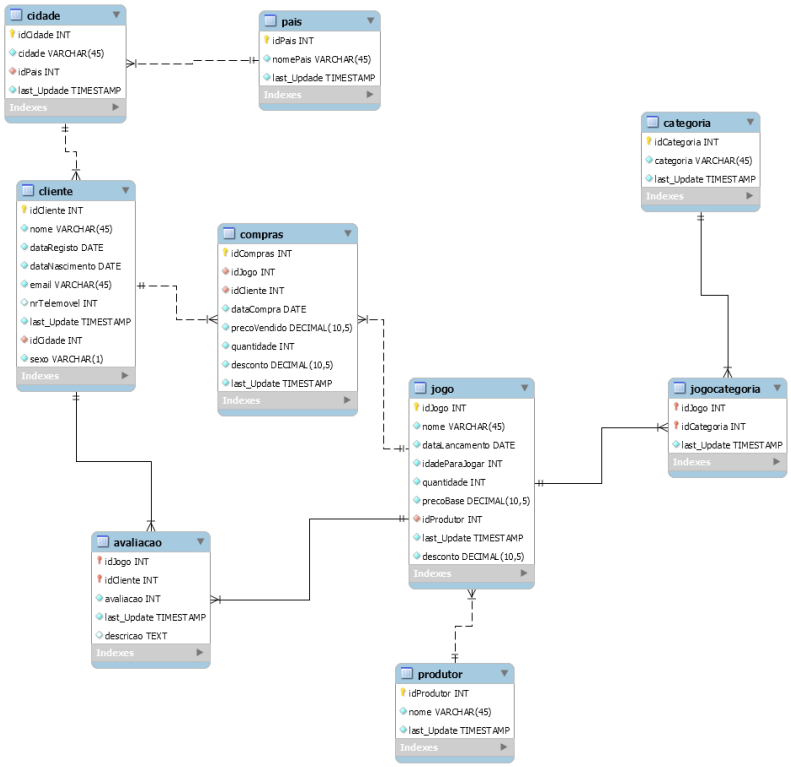


Ilustração 3: Modelo Lógico da Base de Dados Relacional

3.1.1. Segunda loja de jogos online

A base de dados da loja de jogos retro irá conter todos os registos dos utilizadores que tenham criado conta, assim como as compras que eles efetuaram na loja e também as informações relativas aos jogos retro disponíveis na loja. A fonte não-relacional (MongoDB) é representada por quatro coleções: Avaliação, Compras, Jogos e Utilizadores.

```
{
   _id: 1,
   nome: "Gil Goncalves",
   dataNascimento: "1992-09-1",
   email : "gil@mail.com",
   cidade : "Fafe",
   pais   : "Portugal",
   sexo : "M",
   numeroTelemovel : 123456709,
   dataRegisto : "2011-04-2"
}

{
   _id: 2,
   nome: "Jose Pedro",
   dataNascimento: "1993-03-12",
   email : "jose@mail.com",
   cidade : "Lousada",
   pais   : "Portugal",
   sexo : "M",
   numeroTelemovel : 234567890,
   dataRegisto : "2012-01-2"

}
```


Ilustração 4: Coleção Utilizadores


```
{
   _id: 1,
   nome : "Pacman",
   dataLancamento : "1980-05-22",
   idadeParaJogar : 5,
   quantidade : 100,
   precoBase : 5,
   produtor : "Namco",
   desconto : 0.5,
   categoria : ["Labirinto"]
}

{
   _id: 2,
   nome : "Super Mario World",
   dataLancamento : "1990-11-21",
   idadeParaJogar : 5,
   quantidade : 100,
   precoBase : 6,
   produtor : " Nintendo EAD",
   desconto : 0,
   categoria : ["Plataforma","Aventura"]
}
```


Ilustração 5:Coleção Jogo

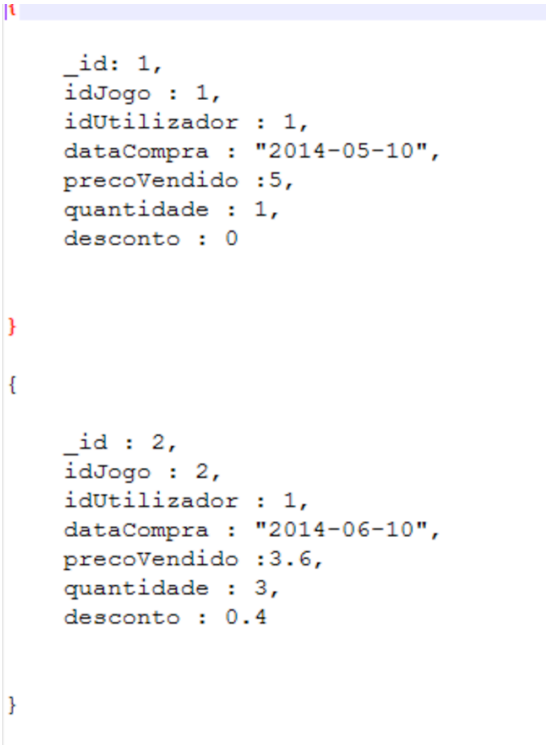


Ilustração 6: Coleção Compra

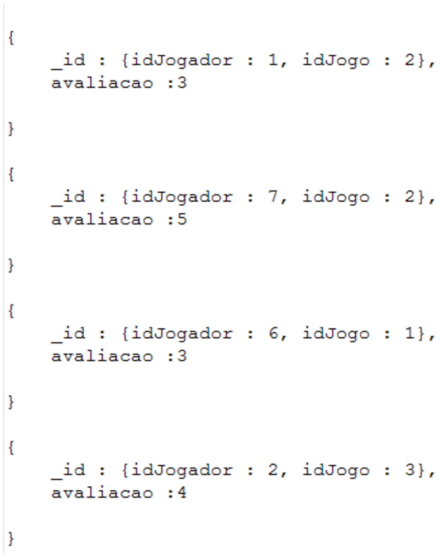


Ilustração 7: Coleção Avaliação

3.2. Análise das fontes com elaboração de perfis de dados (profiling)

Ao longo do processo de modelação, as equipas precisam desenvolver uma compreensão da estrutura, conteúdo, relações e derivação dos dados.

É preciso verificar se os dados existentes estão num estado utilizável, ou pelo menos verificar se as suas falhas podem ser geridas. É preciso entender o que é preciso para converte-lo num modelo dimensional.

O *Data profiling* utiliza recursos de consulta para explorar o conteúdo real e as relações no sistema de origem, em vez de depender de informações incompletas ou desatualizadas documentação. O *data profiling* pode ser tão simples como escrever algumas instruções em SQL ou tão sofisticado como uma ferramenta de criada para o efeito *Data Cleaner*.

3.2.1. Fonte de dados relacional

Analisando a fonte de informação da primeira loja rapidamente percebemos que o cliente possui um atributo número de telemóvel que pode tomar valor nulo. Como foi definido que o sexo ou era *M* ou era *F* é necessário averiguar se de facto estes dois valores estão presentes na base de dados e não um terceiro valor.

Em relação as vendas é necessário analisar o facto de o cliente não ter comprado quantidade menor ou igual a zero é preciso analisar também o facto da data de registo ser maior que a data de compras que ele efetuou, caso isso não aconteça é necessário o contacto com o gerente da loja para tentar resolver este problema. O desconto praticado nunca deverá ser maior que um, visto que o desconto se encontra na gama de maior ou igual que zero e menor ou igual a um.

Em relação a avaliação é necessário verificar se existe algum cliente que não tenha comprado o jogo e o tenha avaliado, caso isso aconteça é necessário falar com o gerente da loja para averiguar o assunto. É possível verificar a partida que existe um atributo na tabela avaliação que se encontra a nulo.

Nas tabelas cidade, país e categoria é necessário verificar se não existem nomes de cidade e de países que apesar de estarem escritos de maneira diferente significam o mesmo.

3.2.2. Fonte de dados não relacional

Como a base de dados da loja segunda loja esta alocada numa base de dados *NoSQL* e como não existem uma estrutura prévia das tabelas presentes é necessário analisar cada linha da tabela para averiguar a estrutura presente em casa uma delas.

Existem também os casos enumerados a cima que é preciso também tomar em conta quando se for a extrair os dados.

3.3. Desenvolvimento do mapa de dados fontes-*Data Warehouse*

No Anexos (Anexo 1) apresenta-se o mapa de dados para as dimensões jogo, cliente e tabela de factos. Em relação á dimensão data, o mapeamento é direto.

3.4. Definição de um plano de recuperação de dados

Para evitar a perda total de informação, semanalmente é carregado para uma base de dados as informações que estão presentes tanto no *data warehouse* como nas fontes de informação, este carregamento deverá ser feito na hora de menos utilização da base de dados e não deverá interferir com o carregamento para o *data warehouse*.

É também proposto que seja guardado a informação em papel para o caso das base de dados falharem todas.

3.5. Estimativa do volume de dados a migrar

Inicialmente iremos exportar poucos dados, visto de se tratar de uma empresa mais o menos jovem, a medida que a empresa for ganhando destaque é estimado que haja um grande volume de compras, logo o valor estimado seria de cinquenta mil registos diários.

Tabela	**Tamanho de Cada Ocorrência**
Cliente	196 bytes
Categoria	31 bytes
Cidade	45 bytes
Jogo	78 bytes
Pais	41 bytes
Produtor	56 bytes
*Avaliação	65 552 bytes
Compra	31 bytes
JogoCategoria	8 bytes

Tabela 2: MySQL tamanho por registo

*cálculo efetuado tendo em conta uso de todos os caracteres, ou seja, utilização do tamanho máximo possível do tipo

No carregamento inicial da base SQL nós temos 100 clientes, 10 categorias, 20 cidades, 50 jogos, pais 5, 30 produtores, 30 avaliações, 250 compras, 75 jogoCategoria.

Somando os valores a cima representados estima-se que o tamanho inicial necessário ronde 2 001 505 *bytes*, ou seja, aproximadamente 2MB (*megabytes*).

Na seguinte tabela encontra-se o tamanho de uma ocorrência na Fonte de Dados Não-Relacional.

Tabela	**Tamanho de Cada Ocorrência**
Utilizador	50 bytes
Compras	30 bytes
*Jogo	38+(6*N)
Avaliação	12 bytes

Tabela 3: MongoDB tamanho por registo

*N = Numero de elementos do array Strings

No carregamento inicial da base SQL nós temos 50 utilizadores, 400 compras, 25 jogos, 15 avaliações.

Somando os valores a cima representados estima-se que o tamanho inicial necessário ronde 15 630+(150+N) *bytes*.

3.6. Revisão da informação recolhida com o cliente

Através de um dialogo com o cliente estas foram as informações recolhidas:

- Saber quais foram os clientes que mais compraram na sua loja;
- Saber quais são as faixas etárias que mais compraram na sua loja;
- Saber qual o jogo mais vendido e consequentemente o menos vendido;
- Saber quais sãos os países que mais compram na sua loja;
- Saber quais as cidade que mais compram na sua loja;

Estas são as perguntas bases que o cliente quer ver respondidas para depois através de campanhas de *marketing* aumentas as vendas dos jogos e consequentemente o seu lucro.

4. Modelação Dimensional

4.1. Construção da matriz de decisão

4.1.1. Data Mart

De encontro as necessidades do nosso cliente, basta "apenas" implementar um Data Mart. Este Data Mart contém toda a informação que é necessária para a análise detalhada de todas as vendas feitas pelas duas lojas, assim o nosso cliente poderá ter uma visão geral sobre o negócio.

4.1.2. Matriz de Decisão

Caraterização de Data Mart de Vendas	
Identificação: Vendas	
Descrição Geral: Informação para suporte à tomada de decisão na área de venda da loja "*Século XXI*" providenciando elementos de dados selecionados acerca das vendas de jogos em todas lojas, para gestão e controlo das vendas realizadas.	
Estrutura base	
Tabela de Factos	TF-Vendas
Dimensões	
DataDeVenda	**X**
Jogo	**X**
Cliente	**X**
Numero Dimensões	3
Tipo	Transacional
Periodicidade	Diária
Descrição	Venda de jogos
Utilidade estratégica	Avaliação comercial da loja online. Definição estratégicas promocionais sobre a idade do publico alvo. Identificar jogos que mais sucesso tem tido na loja. Identificar o período de maior sucesso de vendas. Verificar a atitude dos clientes perante a aplicação de um certo desconto num dado jogo.
Utilizadores	Administrador da loja e gestor da Base de Dados

Tabela 4: Matriz de Decisão

4.2. Definição do grão

O grão do Data Warehouse corresponde a uma compra de um jogo por um cliente numa certa data.

4.3. Escolha e caracterização das Dimensões

Os modelos dimensionais devem ser estruturados de acordo com os processos de negócio e de tomada de decisão, e foi exatamente assim que definimos e tomamos a decisão de escolher as dimensões que futuramente serão carregadas para o nosso *Data Warehouse*.

Na secção anterior definimos o *grão* assim todos os factos contidos numa única tabela de factos respeitam o *grão* definido, no mesmo nível de detalhe. Tendo isto em mente, e sabendo que a modelação dimensional de dados é uma atividade que se rege pelos requisitos de processos de tomada de decisão e não por requisitos de processos de suporte operacional, assim iremos explicar abaixo todo o processo realizado com as mesmas.

4.3.2. Dimensão Cliente

Esta dimensão guarda os dados dos clientes ao qual a venda foi efetuada, esta dimensão tem como atributos, o identificador da dimensão, número de telemóvel, o nome, o sexo, a data que o cliente se registou, a data de nascimento, o email, a cidade e o país, destas temos a possibilidade de agrupar por:

- Sexo
- Data de Registo
- Cidade
- País

4.3.2. Dimensão Data

Com a dimensão data guarda a data em que a venda foi efetuada, esta dimensão tem como atributo o identificador da dimensão, atributo *data* que identifica a data em si "dia/mês/ano", o atributo *ano* que identifica o ano da data, o atributo *mês* que identifica o mês da data, o atributo *dia_mes* que identifica o dia do ano e *trimestre* que identifica o trimestre do ano.

Além de ser possível analisar os dados de apenas um dia, é possível agrupar os alugueres em relação à sua data sobre os seguintes níveis:

- Mês do ano, com possibilidade de roll-up para Trimestre, e por sua vez com possibilidade de roll-up para ano.

- Por Dia
- Por Dia da Semana

4.3.3. Dimensão Jogo

A dimensão Jogo guarda todas as informações relativas ao jogo que foi comprado, e tem como atributos o identificador da dimensão, o Preço Base, o Produtor, o Nome, a Quantidade Disponível, a Idade mínima para um jogador jogar, a Data de lançamento, e o Desconto a que o jogo está.
Destes atributos, podem ser agrupados por:

- Produtor
- Idade Mínima para Jogar
- Data de lançamento

4.3.3. Atributos com variação

As dimensões que tem atributos com variação são a **Dimensão Cliente** e **Dimensão de Jogo**.
Na **Dimensão Cliente** os atributos de variação por método de substituição *sem registo histórico* são:

- Número de telemóvel

E *com registo histórico*

- Cidade
- País

Em relação á **Dimensão Jogo** os atributos de variação por método de substituição *sem registo histórico*:

- Quantidade Disponível
- Preço Base
- Desconto

4.3.4. Síntese das Dimensões

Dimensões do Data Mart "Vendas"			
Nr	Identificação	Descrição	Esquema(Tipo)
1	Cliente	Identificação e caracterização dos clientes das diferentes lojas.	DIM_CLIENTE (com Variação, com História, numa periocidade Diária)
2	Jogo	Informação sobre o catálogo geral de jogos disponíveis em stock.	DIM_JOGO (com Variação, com História, numa periocidade Diária)
3	Data	Esta é a dimensão temporal. Acolhe todos os atributos que sustentam análises ao longo do tempo, como data, dia, dia da semana, mês, trimestre e ano.	DIM_DATA (com diferentes papeis)

Tabela 5: Síntese das Dimensões

4.3.4. Esquema Dimensional

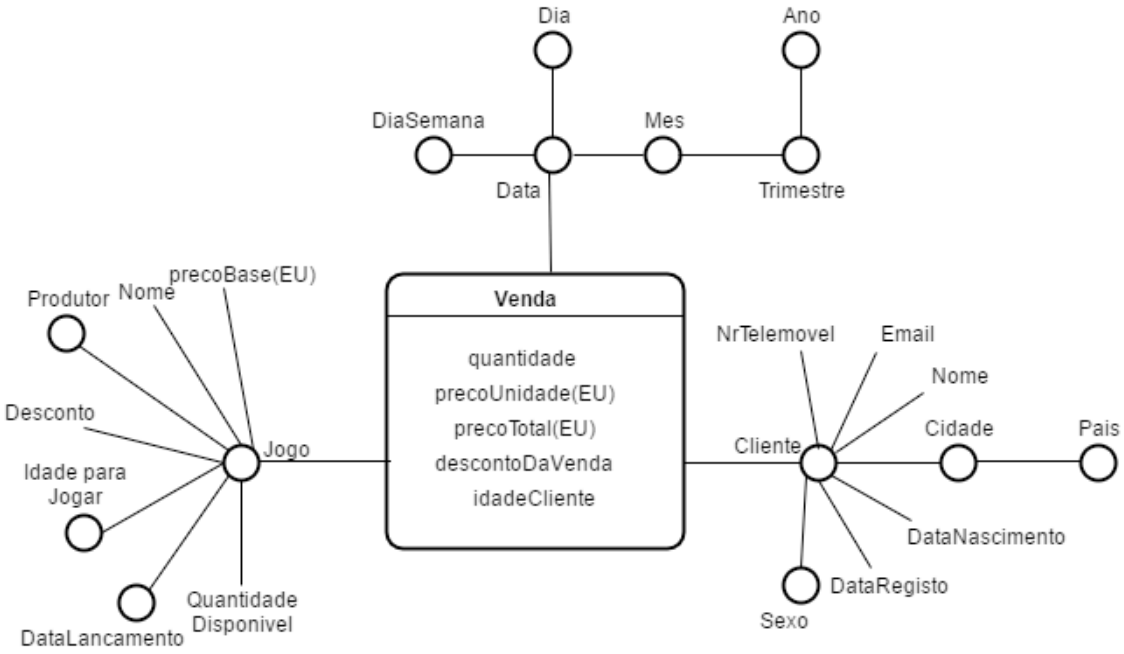


Ilustração 8:Star Schema

4.4. Escolha e caracterização das tabelas de factos

Todos os processos de negócios são representados por um modelo dimensional que consiste numa tabela de factos que irá conter os eventos de medições numéricas envolvendo as tabelas de dimensões que irá conter o contexto textual que era verdadeiro quando foi carregado. O primeiro objetivo da tabela de factos é que seja simples e simétrica. Os utilizadores beneficiam da simplicidade porque os dados são fáceis de entender e de navegar.

A tabela de facto escolhida dirá respeito a uma venda. Essa venda guardará a quantidade vendida para perceber quanto o cliente comprou, o preço por unidade, o preço total, o desconto da venda, e a idade do cliente. A idade do cliente vai permitir uma análise das compras por perfis etários.

4.4.1. Esquema conceptual

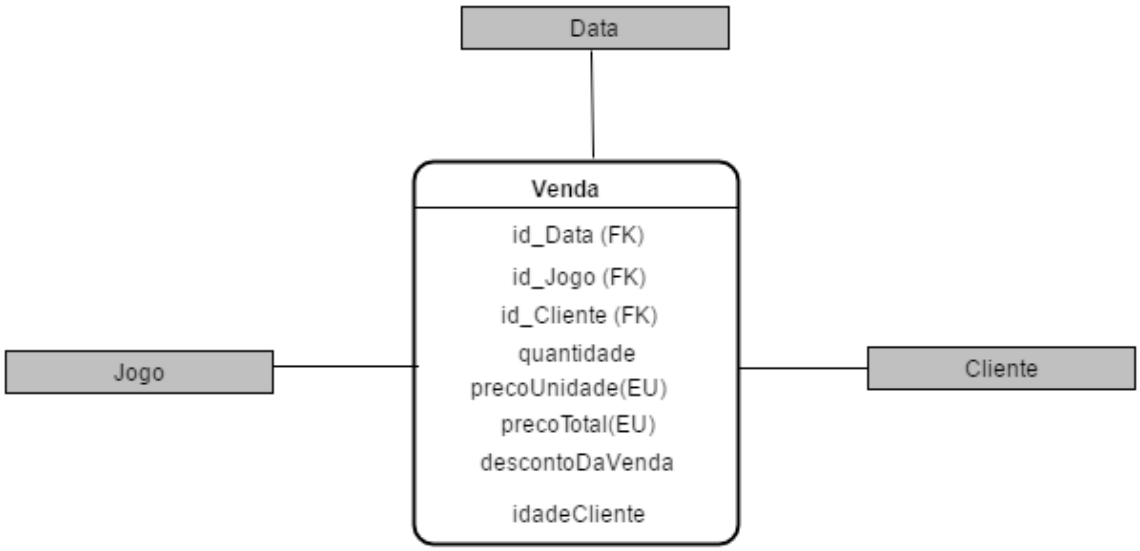


Ilustração 9: Esquema conceptual para o DataMart Vendas

4.4.1. Tabela de facto

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 *mart*			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(*clustered*) 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.			

Tabela 6: Síntese tabela de facto

4.5. Desenvolvimento e documentação dos esquemas multi- dimensionais

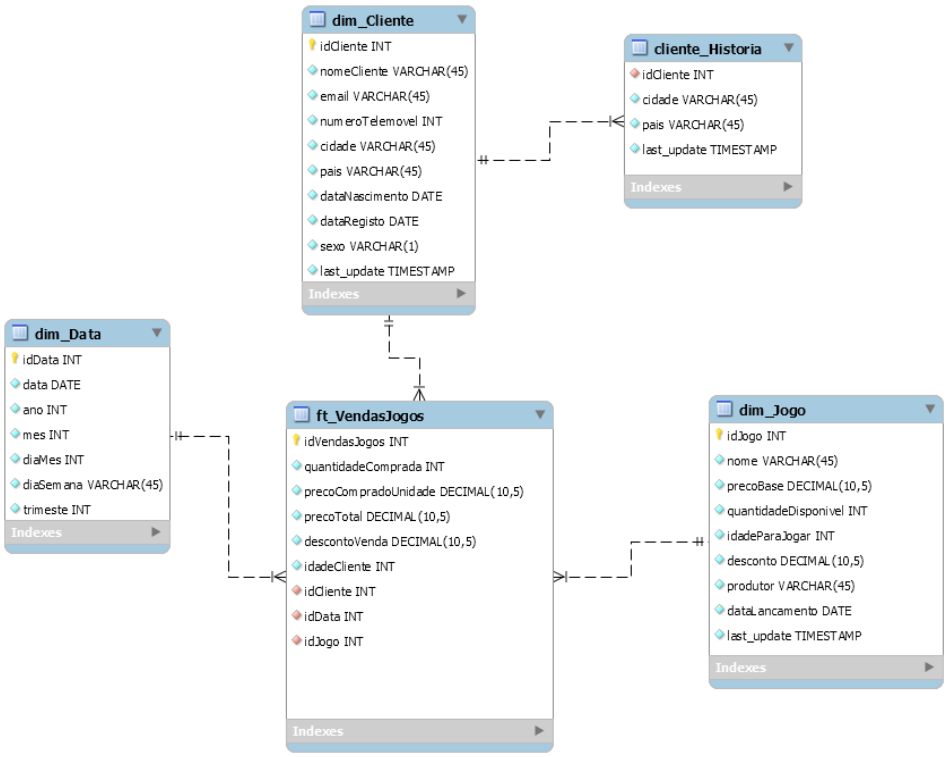


Ilustração 10: Modelo Dimensional

4.6. Revisão do modelo dimensional com o cliente

Na sequência de reuniões com o cliente após estar concluído o modelo dimensional apresentamos o modelo ao cliente para verificar se o modelo se encontrava segundo os requisitos que a empresa site XXI apresentou.

O cliente queria obter a informação a cerca das suas vendas para assim fazer profiling dos dados obtidos, no sentido de adaptar a sua empresa a aumentar o lucro da mesma, ao atrair mais clientes para a compra dos jogos à venda.

Na sequência da reunião onde foi introduzido o modelo dimensional foi aprovado pela empresa, podendo assim começar a implementação do ETL (Extraction, Transform, Load).

5. Extração, Transformação e Carregamento de Dados

5.1. Definição e caracterização do processo de povoamento do Data Warehouse

Após a definição do modelo dimensional do nosso Data Warehouse, é necessário proceder à definição e caracterização do processo que irá povoar este, mais propriamente o processo ETL associado a este projeto.

Inicialmente, é necessário proceder à identificação dos diferentes carregamentos, que neste caso irão ser dois, o carregamento inicial e regular do Data Warehouse.

O carregamento inicial é realizado apenas uma vez, no arranque do sistema, que irá permitir efetuar um povoamento inicial das tabelas de dimensão e da tabela de factos do nosso Data Mart, preparando assim o sistema para receber posteriores carregamentos de dados provocados por inserções e/ou modificações a dados nas fontes envolvidos no projeto.

Posteriormente, é necessário definir um processo ETL que represente o carregamento regular do nosso sistema, que irá ocorrer diariamente na janela de oportunidade definida anteriormente, que se situa entre duas e as seis horas da manhã. Este carregamento caracteriza-se por atualizar a informação do Data Warehouse com as novas informações resultantes de posteriores inserções e/ou modificações realizadas em cada uma das fontes de dados.

5.2. Modelação conceptual do processo de ETL

De seguida, irá ser apresentada a modelação conceptual do processo de ETL para cada um dos tipos de carregamento, documentado cada uma das fases que o compõem: extração, limpeza, conformidade e conciliação dos dados. Por fim, após a fase de conciliação e de na área de retenção já possuirmos as pré-dimensões e a pré-tabela de factos, fazemos o carregamento desses mesmos dados para o Data Warehouse.

5.2.1 Extração

Nesta primeira fase do processo de ETL, é efetuada a extração dos dados presentes em cada uma das fontes e o posterior carregamento destes para a área de retenção.

Para cada um dos carregamentos, são extraídos de cada uma das fontes os dados necessários às tabelas de dimensão e à tabela de facto, ocorrendo esta extração em paralelo, sendo que os dados serão mantidos em tabelas de auditoria para cada dimensão de cada fonte.

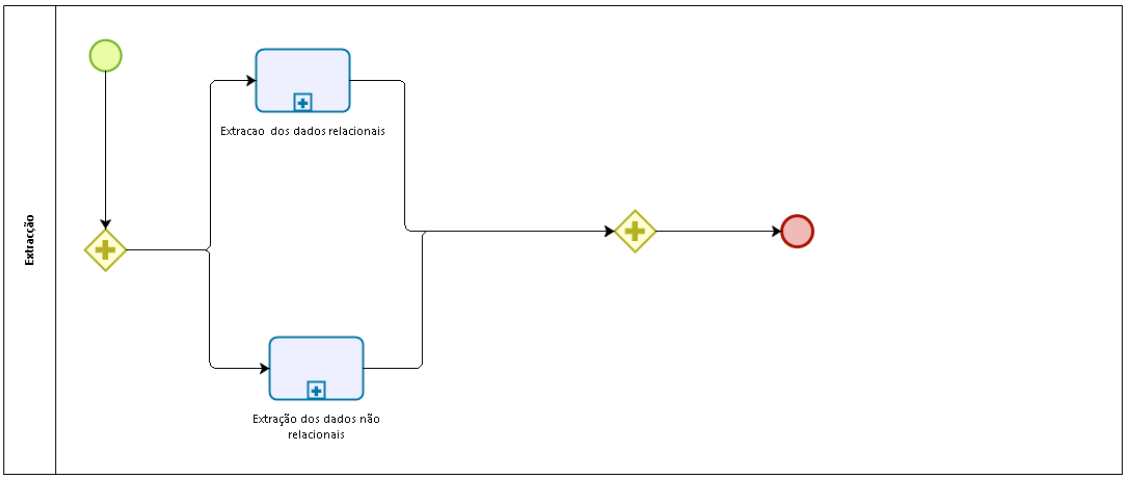


Ilustração 11: Extração dos dados dos clientes, jogos e vendas para cada fonte de dados.

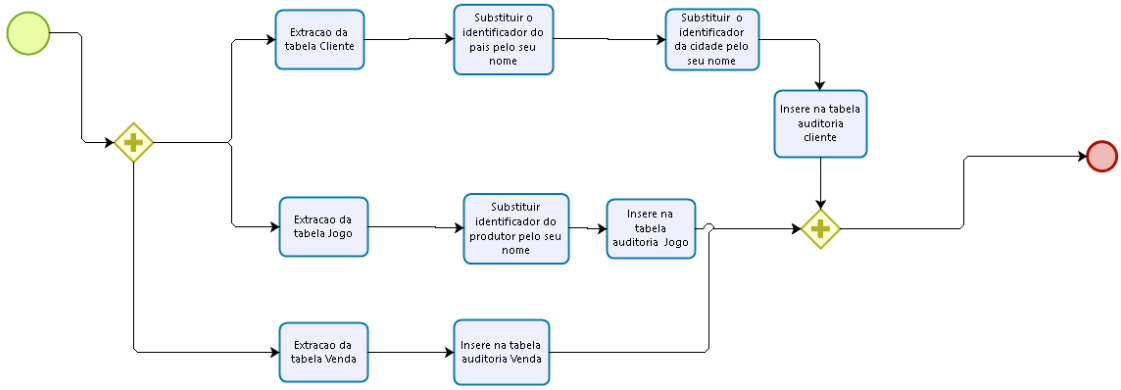


Ilustração 12: Processo de extração dos clientes, jogos e vendas da fonte de dados MySQL.

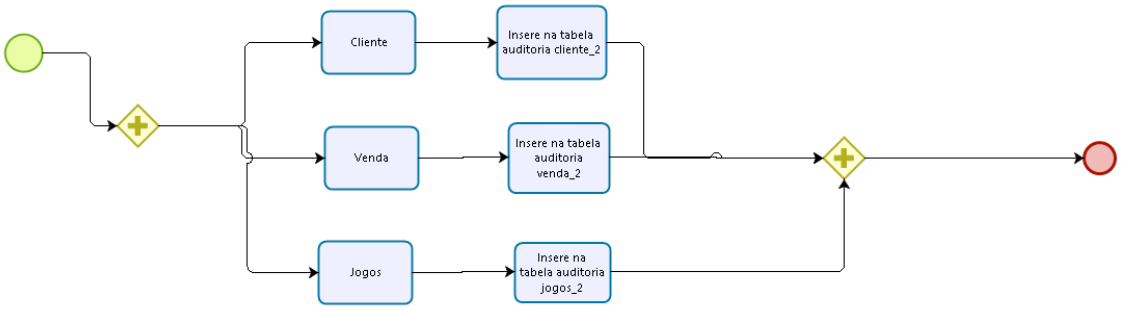


Ilustração 13: Processo de extração dos clientes, jogos e vendas da fonte de dados NoSQL.

No entanto, para a dimensão Data, esta é excecionalmente carregada de imediato para o Data Warehouse, isto no carregamento inicial do processo de ETL, pois é uma dimensão sem variação e que se sabe à partida que não terá qualquer tipo de falhas, visto que foram geradas automaticamente pela equipa de trabalho sob a forma de um ficheiro *csv*.

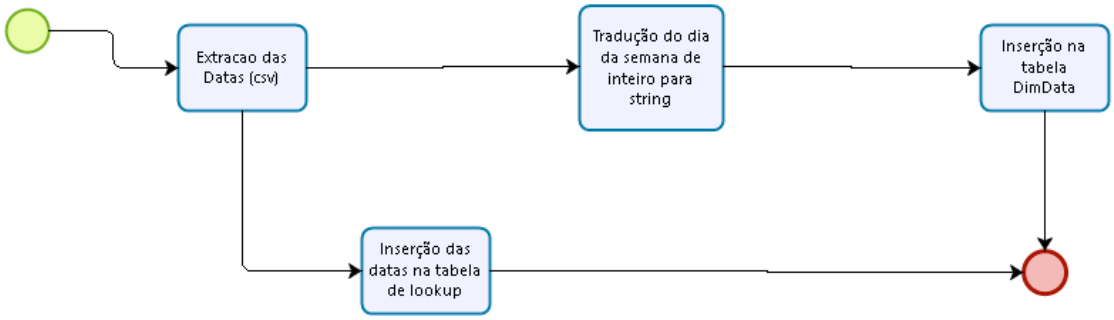


Ilustração 14: Processo de extração e carregamento da dimensão Data no Data Warehouse.

5.2.2 Limpeza

Já numa fase de limpeza, é necessário analisar e limpar os dados previamente extraídos das fontes de informação, eliminando os dados que não exprimam informação relevante ou concisa para o sistema de apoio à decisão.

Anteriormente, identificámos que apenas atributos relacionados com os clientes necessitam de ser limpos, sendo o caso do nome dos clientes e também o seu número de telemóvel, visto que a política da empresa deixa que os clientes decidam a forma como inserem o sue nome e também se inserem ou não o seu contacto pessoal.

Posto isto, é necessário que os registos dos clientes passem por uma fase de limpeza que irá capitalizar o nome do cliente e posteriormente verificar se o número de telemóvel deste é nulo, que em caso afirmativo será substituído por “Desconhecido”. Tal como acontece na fase anterior, os dados são migrados das tabelas de auditoria da fase de extração para as tabelas de auditoria da fase de limpeza, sendo que apenas os dados das tabelas de auditoria de clientes irão passar pela seguinte fase de limpeza:

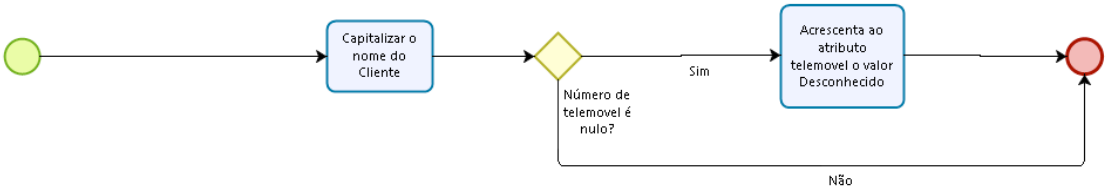


Ilustração 15: Processo de limpeza dos clientes para os dados extraídos de cada fonte.

5.2.3 Conformidade

Nesta etapa, e já com a informação extraída e limpa das etapas anteriores, será definitivamente transformada para o seu devido formato no Data Warehouse Para tal, o formato das tabelas irá ter que ser alterado tanto ao nível do nome do atributo como o seu tipo e a informação que contém, alterações estas que foram identificadas anteriormente nas tabelas *source-target*.

Para tal, é necessário proceder a um processo de inserção da operação a ser efetuada (inserção ou modificação), o tempo em que esta ocorreu (data de extração) e a fonte de dados origem (fonte de dados MySQL ou NoSQL), que irá ocorrer para o carregamento inicial para os dados de cada fonte. Já para o carregamento regular, não será preciso efetuar este passo para os dados provenientes da fonte de dados MySQL, visto que serão usadas tabelas de auditoria no sistema operacional para proceder a esta extração dos dados já contendo esta informação. Ainda para o carregamento regular e em relação aos dados provenientes da fonte de dados NoSQL, não será preciso adicionar o tempo de operação, visto que nessa fonte cada registo passará a conter esse mesmo tempo de inserção ou modificação para cada registo.

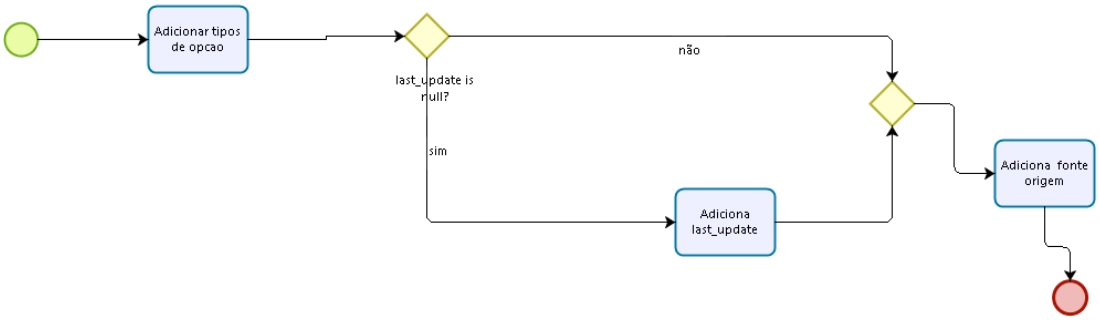


Ilustração 16 - Processo de conformidade para os registos.

5.2.4 Conciliação

A fase de conciliação remete para a integração dos dados das diferentes fontes entre si, preparando estes para a fase final de carregamento dos dados para o Data Warehouse.

Sendo as fontes heterogéneas, é necessário o recurso ao uso de chaves de substituição tanto para os clientes como para os jogos, tendo em conta que é necessária a

conciliação desses tipos de dados de várias fontes e é necessário criar uma chave sem significado para que não perca a integridade e previna a duplicação de chaves.

As chaves de substituição são geradas aquando de uma operação de inserção, verificando se o registo é duplicado. Em caso afirmativo, o registo é enviado para quarentena, caso contrário é gerada uma chave de substituição e o registo é inserido na tabela de pré-dimensão respetiva. Este processo é comum tanto ao carregamento inicial como o regular.

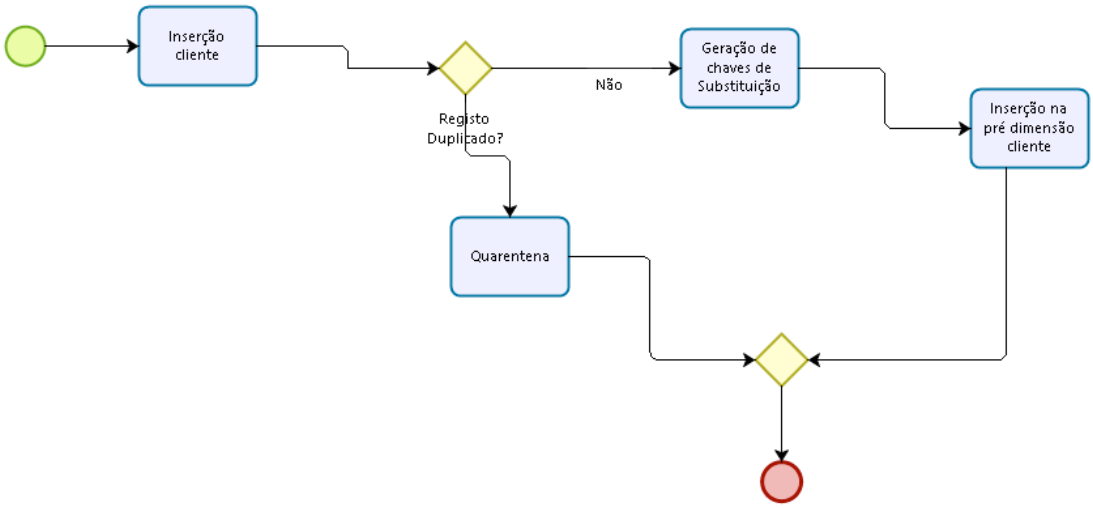


Ilustração 17 - Processo de conciliação de inserção de clientes.

Já para o carregamento regular, e visto que este tem de tratar de *updates*, para cada registo deste tipo faz-se um *lookup* da respetiva chave de substituição, e caso esta exista, este é guardado na tabela de *update* respetiva, caso contrário é inserido na respetiva tabela de quarentena.

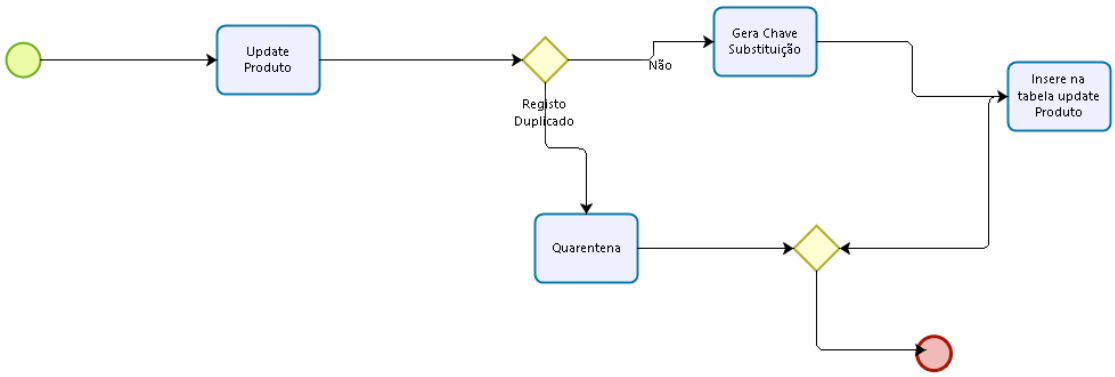


Ilustração 18 - Processo de conciliação de modificação dos produtos.

Por fim, é feito o processamento das vendas, ao qual substituímos os identificadores do cliente e jogo pelas suas respetivas chaves de substituição, inserindo os registos numa pré tabela de factos.

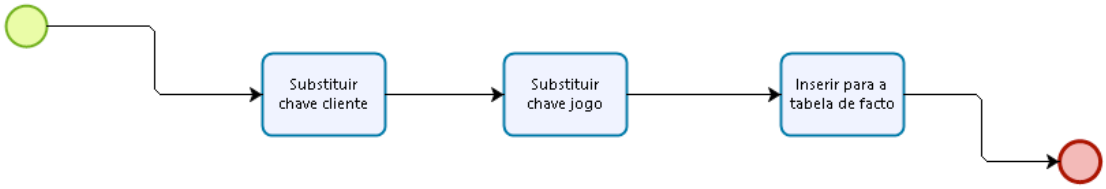


Ilustração 19 - Processo de conciliação das vendas.

5.2.5 Carregamento

Após a execução de todos os processos anteriormente referidos, segue-se o passo final que se caracteriza por carregar todas as pré dimensões e a pré tabela de factos para o Data Warehouse.

Este processo é praticamente direto, uma vez que os dados são do mesmo tipo que do sistema de suporte à decisão. No entanto, possui casos especiais pois é necessário tratar dos *updates* às dimensões com variação, sendo que este processo de modificação fica ao cargo do próprio Data Warehouse que necessita de possuir um conjunto de procedimentos para tratar destes casos.

5.3. Definição e implementação da área de retenção do sistema

Na área de retenção foram criadas três tabelas de auditoria para cada fonte de dados que albergará com os dados da extração, limpeza e conformidade. Esta tabelas vão ser usadas para cada tipo de carregamento, seja inicial ou regular, e irão conter os dados resultados de cada uma das fases acima referenciadas.

Ilustração 20: Tabelas de Extração(Retenção)

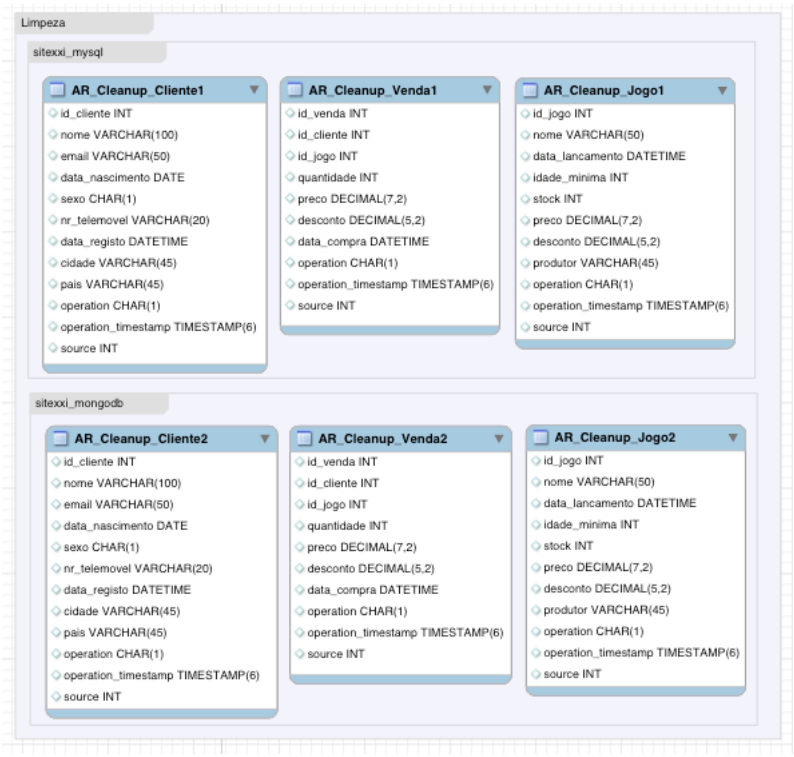


Ilustração 21:Tabelas de Limpeza(Retenção)

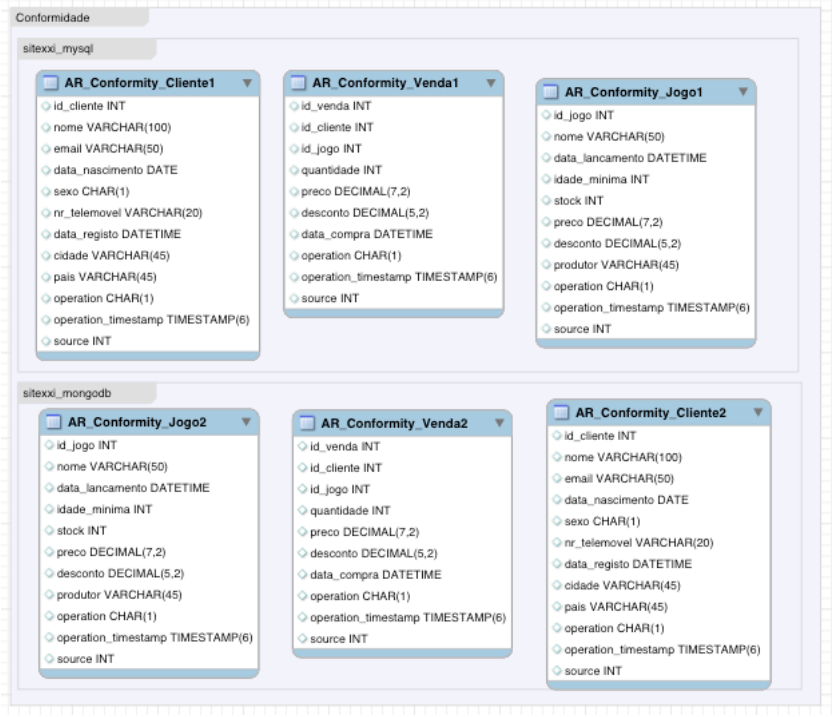


Ilustração 22: Tabelas de Conformidade (Retenção)

De seguida, foram criadas as tabelas para a fase de conformidade, ao qual criamos as pré dimensões Cliente e Jogo, bem como a pré tabela de factos Venda, que já corresponderam ao formato das dimensões e tabela de factos do nosso sistema de suporte à decisão.

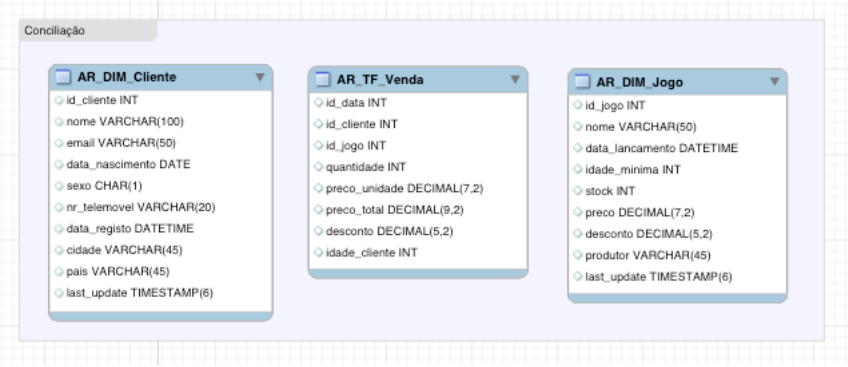


Ilustração 23: Tabelas de Conciliação

De seguida, foram criadas as tabelas de geração de chaves substituição, que possuem os atributos desambiguadores para cada uma das dimensões, de forma a que seja possível a gerar chaves de substituição para uma cada dimensão, sendo usadas também para tratar de registos duplicados.

Ilustração 24: Tabelas de Surrogate Key

Foram também criadas as tabelas de quarentena que permitiram guardar os registos duplicados, bem como os registos que possuam parâmetros inválidos, como o um email de um cliente que não é válido, entre outros.

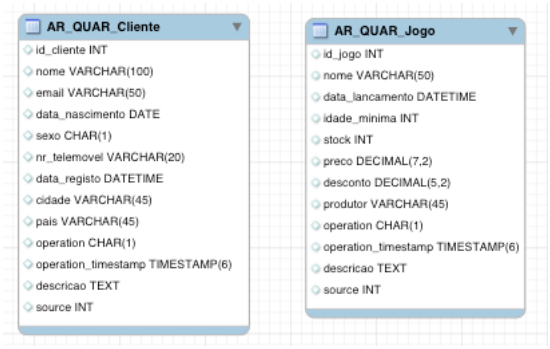


Ilustração 25: Tabelas de quarentena

Por fim, foram criadas as tabelas de *update*, que vão ser usadas para carregar para o Data Warehouse as modificações ocorridas nas fontes de dados.

Ilustração 26: Tabelas de Update

5.4. Implementação do sistema de povoamento

Para a implementação do sistema de povoamento, utilizamos a ferramenta Kettle ao qual desenvolvemos duas *jobs* que irão executar o carregamento inicial e final.

No carregamento inicial, vamos extrair os dados diretamente das fontes de dados para a área de retenção da extração. Posteriormente, aplicamos a limpeza dos dados extraídos de cada uma das fontes, e aplicamos a estas a conformidade para que os dados de diferentes fontes possam ser conciliáveis, ou seja, sejam idênticos. Após esta fase, vamos aplicar a conciliação dos dados, sendo que estes irão ser carregados para as tabelas de pré dimensões e a tabela de factos, realizando-se posteriormente o carregamento dos dados para o Data Warehouse.

Para o carregamento regular, o processo difere inicialmente na forma como extraímos os dados. Para a fonte de dados MySQL, vamos utilizar tabelas de auditoria para extrairmos os dados, sendo que estes já vêm com informações como o tipo de operação, quando ocorreu, entre outros. Posteriormente irão passar por uma fase de limpeza e conformidade, estando assim os dados prontos a serem conciliados. Esta fase de conciliação é diferente da aplicada no carregamento inicial, visto que agora os dados têm dois tipos diferentes de operações, *insert* e *update*. Para o caso dos *inserts*, o processo é idêntico ao carregamento inicial. Já em relação às operações de *update*, o processo difere na medida em que os dados têm de ser carregados para tabelas de modificação, que possuem os dados dos atributos a modificar. Estes dados irão ser processados do lado do Data Warehouse por procedimentos que irão fazer a substituição correta, variando conforme possua história ou não.

Por fim, optámos pela implementação de tabelas entre cada passo do processo de ETL para que caso ocorra algum tipo de falha (uma falha de energia, por exemplo), este consiga recuperar o estado anterior e continue o processo de ETL do ponto onde a falha ocorreu. Assim, entre cada fase do processo de povoamento, sempre que uma fase ocorre com sucesso, é feito a limpeza da tabela respetiva à fase anterior.

5.5. Testes e validação do sistema

Após a implementação do sistema ETL, é necessário criar um conjunto de testes que validem o comportamento do sistema, de forma a podermos rever o processo com o cliente e a finalizar o desenvolvimento deste sistema.

Para tal, foram executados dois testes: o primeiro teste verifica se jogos válidos são de facto inseridos no Data Warehouse; o segundo teste verifica se clientes repetidos são enviados para a quarentena.

Primeiro teste:

id_jogo	nome	data_lancamento	idade_minima	stock	preco	desconto	id_produtor
1	GTA	2014-02-12	18	100	60.00	5.00	4
2	Max Payne	2014-01-12	18	100	65.00	0.00	10
3	Hitman	2013-02-12	18	100	50.00	80.00	1

id_jogo	nome	data_lancamento	idade_minima	stock	preco	desconto	produtor	last_update
1	GTA	2014-02-12	18	100	60.00	2.75	Ubisoft	2017-01-20 03:22:14.000000
2	Max Payne	2014-01-12	18	100	65.00	0.00	Electronic Arts	2017-01-20 03:22:14.000000
3	Hitman	2013-02-12	18	100	50.00	40.40	Capcom	2017-01-20 03:22:14.000000

Como podemos verificar, o carregamento de jogos válidos da fonte de dados MySQL ocorreu de forma correta, pelo que o sistema está a fazer o processo de carregamento dos jogos. Conseguiu com sucesso extrair os dados da fonte MySQL e carregar com sucesso no Data Warehouse.

Segundo teste:

id_cliente	nome	email	data_nascimento	sexo	nr_telemovel	data_registo	cidade	pais
1	Gil Goncalves	gidl@mail.com	1992-09-01	M	123456709	2011-04-02 00:00:00	Fafe	Portugal
2	Jose Pedro	jose@mail.com	1993-03-12	M	234567890	2012-01-02 00:00:00	Lousada	Portugal
3	Bruno Ribeiro	bruno@mail.com	1991-01-14	M	345678901	2012-05-02 00:00:00	Braga	Portugal
4	Luis Pedro	luis@mail.com	1990-04-15	M	456789012	2009-04-15 00:00:00	Braga	Portugal
5	Celia Figueiredo	celia@mail.com	1993-12-24	F	567890123	2012-01-02 00:00:00	Lisboa	Portugal
6	Marcia Costa	marcia@mail.com	1992-02-12	F	Desconhecido	2011-12-02 00:00:00	Porto	Portugal
7	Daniel Rodrigues	daniel@mail.com	1990-09-01	M	Desconhecido	2010-04-02 00:00:00	Fafe	Portugal
8	Ricardo Lopes	ricardo@mail.com	1990-09-01	M	678901234	2009-04-02 00:00:00	Paris	Franca
9	Carlos Faria	carlos@mail.com	1993-12-01	M	Desconhecido	2012-11-02 00:00:00	Barcelona	Espanha
10	Gil Goncalves	gid@mail.com	2012-09-01	M	1234116709	2011-04-02 00:00:00	Fafe	Portugal

id_cliente	nome	email	data_nascimento	sexo	nr_telemovel	data_registo	cidade	pais
1	Gil Goncalves	gil@mail.com	1992-09-01	M	123456709	2011-04-02 00:00:00	Fafe	Portugal
2	Jose Pedro	jose@mail.com	1993-03-12	M	234567890	2012-01-02 00:00:00	Lousada	Portugal
3	Bruno Ribeiro	bruno@mail.com	1991-01-14	M	345678901	2012-05-02 00:00:00	Braga	Portugal
4	Luis Pedro	luis@mail.com	1990-04-15	M	123456789	2009-04-15 00:00:00	Braga	Portugal
5	Celia Natalia Figueiredo	celia@mail.com	1993-12-24	F	567890123	2012-01-02 00:00:00	Braga	Portugal
6	Alberto Fernandes	alberto@mail.com	1994-02-02	M	567890120	2012-01-02 00:00:00	Fafe	Portugal
7	Carla Afonso	carla@mail.com	1994-05-02	F	567890130	2013-01-02 00:00:00	Londres	Inglaterra

id_cliente	nome	email	data_nascimento	sexo	nr_telemovel	data_registo	cidade	pais	operation	operation_timestamp	descricao	source
2	Jose Pedro	jose@mail.com	12/03/93	M	234567890	02/01/12 00:00	Lousada	Portugal	I	2017-01-20 03:22:14.427000	Cliente repetido	2
3	Bruno Ribeiro	bruno@mail.com	14/01/91	M	345678901	02/05/12 00:00	Braga	Portugal	I	2017-01-20 03:22:14.427000	Cliente repetido	2
4	Luis Pedro	luis@mail.com	15/04/90	M	123456789	15/04/09 00:00	Braga	Portugal	I	2017-01-20 03:22:14.427000	Cliente repetido	2
5	Celia Natalia Figue	celia@mail.com	24/12/93	F	567890123	02/01/12 00:00	Braga	Portugal	I	2017-01-20 03:22:14.427000	Cliente repetido	2

Como podemos verificar neste caso de teste, verificámos que existem registos de clientes repetidos nas duas primeiras imagens, que correspondem aos clientes das fontes de dados MySQL e NoSQL, respetivamente. Podemos concluir que o nosso processo ETL previne este caso, colocando os registos repetidos na tabela de quarentena.

5.6. Revisão do sistema desenvolvido com o cliente

Terminada a fase de implementação do ETL, houve mais uma vez uma reunião com o cliente, de forma a confirmar que o processo de ETL ia de acordo com o que a empresa havia estipulado. Para ajudar a perceção deste processo utilizamos os diagramas BPMN para justificar todo processo de construção do Data Warehouse pedido. No fim desta reunião, a chefia mostrou-se satisfeita com o que havia sido implementado.

6. Instalação do Sistema

6.1. Definição do plano de instalação do sistema – área de retenção, *Data Warehouse* e sistema de povoamento

De forma a que o nosso sistema de povoamento funcione de forma correta, é necessário proceder a algumas alterações aos sistemas operacionais que suportam o sistema de suporte à decisão.

Para o sistema operacional MySQL, é necessária a inclusão de tabelas de auditoria de forma a albergar com as posteriores inserções e modificações de registos, de forma a que sempre que haja um *insert*/*update*, seja despoletado um *trigger* que atualize a respetiva tabela de auditoria, para que depois estas alterações ocorram no Data Warehouse.

Já para o sistema operacional NoSQL, é necessária a adição de um atributo *last_update* que registe a data do *insert*/*update*, para que depois nos carregamentos regulares seja possível extrair apenas o estritamente necessário.

A área de retenção e o Data Warehouse foram implementados e testados no SGDB MySQL, e o sistema ETL foi desenvolvido em Kettle. É necessária a instalação do MySQL e do Kettle para que o processo desenvolvido esteja operacional, sendo que o Kettle deverá ser instalado no sistema onde se encontra a área de retenção.

6.2. Implementação do sistema de Data Warehousing

A implementação do Data Warehouse deve seguir o processo apresentado na secção 4, sendo necessário seguir ao pormenor todas as etapas definidas.

As dimensões necessárias a criar são as dimensões cliente e jogo, sendo que a ambas possui atributos com variação, sendo que a primeira possui história. Como tal, é necessária e indispensável a criação da tabela de história para os atributos cidade e pais, caso contrário o modelo não se encontrará válido e não respeitará os requisitos definidos. Indispensável ainda é a criação da tabela de factos Venda, que albergará com todas as vendas efetuadas pela empresa.

Outro dos cuidados a ter é a inclusão do atributo *last_update* nas dimensões cliente e jogo, visto que estas possuem atributos com variação, como por exemplo número de telemóvel e preço, respetivamente.

Por fim, é estritamente necessário a criação dos seguintes procedimentos:

```
sp_update_cliente(id_cliente,nr_telemovel,cidade,pais,last_update);
sp_update_produto(id_produto,stock,preco,desconto,last_update);
```

Estes procedimentos têm de ser criados do lado do Data Warehouse de forma a que seja possível manter as variações em cada uma das dimensões com variação, pelo que a inexistência destes procedimentos afetará negativamente a coerência do sistema de suporte à decisão, fazendo com que este não trate dos *updates* realizados nos sistemas operacionais.

6.3. Carregamento inicial do Data Warehouse

Após a definição do plano de instalação do sistema e da implementação do Data Warehouse, segue-se o carregamento inicial dos dados no sistema de suporte à decisão.

Este processo é crítico visto que se este carregamento inicial falhar, é necessário rever toda a implementação anteriormente descrita, acarretando custos extra a nível monetário para o cliente.

De seguida, é apresentado o resultado do carregamento inicial do Data Warehouse, que ao que tudo indica correu de forma favorável. No entanto, é necessário proceder a uma posterior validação pormenorizada de forma a validar se de facto tudo ocorreu como esperado.

id_data	data	dia_semana	dia	mes	ano	trimestre
1	2013-01-01	Terca	1	1	2013	1
2	2013-01-02	Quarta	2	1	2013	1
3	2013-01-03	Quinta	3	1	2013	1
4	2013-01-04	Sexta	4	1	2013	1
5	2013-01-05	Sabado	5	1	2013	1
6	2013-01-06	Domingo	6	1	2013	1
7	2013-01-07	Segunda	7	1	2013	1
8	2013-01-08	Terca	8	1	2013	1
9	2013-01-09	Quarta	9	1	2013	1
10	2013-01-10	Quinta	10	1	2013	1

Ilustração 27: Carregamento inicial da dimensão Data.

id_cliente	nome	email	data_nascimento	sexo	nr_telemovel	data_registo	cidade	pais	last_update
1	Gil Goncalves	gidl@mail.com	1992-09-01	M	123456709	2011-04-02 00:00:00	Fafe	Portugal	2017-01-20 03:22:14.000000
2	Jose Pedro	jose@mail.com	1993-03-12	M	234567890	2012-01-02 00:00:00	Lousada	Portugal	2017-01-20 03:22:14.000000
3	Bruno Ribeiro	bruno@mail.com	1991-01-14	M	345678901	2012-05-02 00:00:00	Braga	Portugal	2017-01-20 03:22:14.000000
4	Luis Pedro	luis@mail.com	1990-04-15	M	456789012	2009-04-15 00:00:00	Braga	Portugal	2017-01-20 03:22:14.000000
5	Celia Figueiredo	celia@mail.com	1993-12-24	F	567890123	2012-01-02 00:00:00	Lisboa	Portugal	2017-01-20 03:22:14.000000
6	Marcia Costa	marcia@mail.com	1992-02-12	F	Desconhecido	2011-12-02 00:00:00	Porto	Portugal	2017-01-20 03:22:14.000000
7	Daniel Rodrigues	daniel@mail.com	1990-09-01	M	Desconhecido	2010-04-02 00:00:00	Fafe	Portugal	2017-01-20 03:22:14.000000
8	Ricardo Lopes	ricardo@mail.com	1990-09-01	M	678901234	2009-04-02 00:00:00	Paris	Franca	2017-01-20 03:22:14.000000
9	Carlos Faria	carlos@mail.com	1993-12-01	M	Desconhecido	2012-11-02 00:00:00	Barcelona	Espanha	2017-01-20 03:22:14.000000

Ilustração 28: Carregamento inicial da dimensão Cliente.

id_jogo	nome	data_lancamento	idade_minima	stock	preco	desconto	produtor	last_update
1	GTA	2014-02-12	18	100	60.00	2.75	Ubisoft	2017-01-20 03:22:14.000000
2	Max Payne	2014-01-12	18	100	65.00	0.00	Electronic Arts	2017-01-20 03:22:14.000000
3	Hitman	2013-02-12	18	100	50.00	40.40	Capcom	2017-01-20 03:22:14.000000
4	Gears of War	2014-02-12	18	10	30.00	5.05	Electronic Arts	2017-01-20 03:22:14.000000
5	Monkey Island	2010-02-12	12	50	20.00	20.00	Square Enix	2017-01-20 03:22:14.000000
6	Tomb Raider	2015-02-12	18	30	30.00	60.00	Konami	2017-01-20 03:22:14.000000
7	NASCAR	2014-02-12	18	100	15.30	54.00	Ubisoft	2017-01-20 03:22:14.000000
8	GTR	2015-10-12	18	10	13.00	0.00	SEGA	2017-01-20 03:22:14.000000
9	Need For Speed	2013-12-12	18	60	30.00	28.00	Activision Blizzard	2017-01-20 03:22:14.000000
10	Alone in The Dark	2014-02-12	18	100	30.00	39.00	Nintendo	2017-01-20 03:22:14.000000

Ilustração 29: Carregamento inicial da dimensão Jogo.

id_venda	id_data	id_cliente	id_jogo	quantidade	preco_unidade	preco_total	desconto	idade_cliente
1	495	1	1	1	30.00	30.00	50.00	21
2	528	2	1	3	39.00	117.00	40.00	21
3	377	2	2	1	65.00	65.00	0.00	20
4	132	3	2	1	25.00	25.00	50.00	22
5	408	1	3	1	60.00	60.00	0.00	21
6	517	4	3	1	15.00	15.00	50.00	24
7	409	4	4	3	30.00	90.00	0.00	23
8	425	1	4	1	58.00	58.00	20.00	21
9	801	1	5	1	12.00	12.00	8.00	22
10	132	5	5	1	10.00	10.00	50.00	19

Ilustração 30: Carregamento inicial da tabela de factos Venda.

6.4. Validação do povoamento realizado

Após a efetuação do carregamento inicial, é necessário proceder à validação do mesmo, tarefa esta que não se caracteriza por ser simples devido ao grande volume de dados carregados para o sistema de suporta à decisão, aliado ao facto da informação ser proveniente de duas fontes de dados distintas.

Anteriormente foram realizados testes à implementação do sistema de ETL de forma a que estes cobrissem na totalidade os diferentes casos que poderiam ocorrer aquando da inserção de registos no Data Warehouse, ao qual o sistema correspondeu de forma correta. No entanto, o volume de dados usados para testar este não tinha o volume dos dados que efetivamente irão ser inseridos neste, pelo que necessitamos de recorrer a outra técnica de verificação.

Posto isto, usámos as contagens dos registos de cada tabela extraída de cada sistema operacional, e comparámos com os registos presentes no Data Warehouse e também com os respetivos registos presentes nas tabelas de quarentena, e verificámos que estas contagens se encontravam iguais, pelo que assim pudemos concluir que o processo de ETL se encontra válido e responde às necessidades impostas pelo cliente.

6.5. Revisão do sistema desenvolvido com o cliente

Nesta última fase, foi planeada toda a instalação do sistema final no Cliente. Todo este processo foi de acordo com o orçamento que o que tinha sido estipulado e é de fácil instalação. Conseguiu-se então ir de acordo com o estabelecido pela chefia da empresa *Site XXI.*

Assim e com toda a confirmação dos meios necessários para a implementação do Data Warehouse, podemos dar início à instalação e carregamento inicial, para garantir que tudo estava de acordo, houve a validação do primeiro povoamento mostrando que este ia de acordo com o pretendido pelo cliente. Finalizando assim o processo de Data Warehousing.

7. Conclusões e Trabalho Futuro

7.1. Avaliação do processo de trabalho

O planeamento do processo de trabalho foi bem estipulado e a equipa de desenvolvimento conseguiu cumprir todos os datas estipuladas.

Todos os membros da equipa conseguiram executar as suas tarefas com sucesso. As reuniões foram feitas todas com forme o planeado, o que fez com que a comunicação da equipa com a empresa fosse fluida havendo assim um favorável desenvolvimento do projeto.

7.2. Avaliação do sistema desenvolvido

Desenvolver um sistema *data warehouse* é um processo trabalhoso que exige uma constante monotorização e adaptável as alterações que forem ocorrendo ao longo do tempo.
Graças ao sistema de *data warehouse* os vários utilizadores possuem agora um sistema centralizado com as informações provenientes das várias lojas.
O sistema desenvolvido permite que o gerente da loja retire conclusões sobre o número de jogos vendidos e quais as suas produtoras que mais jogos vendas nas duas empresas, assim como quais são as faixas etárias que mais compras jogos na sua empresa.

7.3. Evolução do Sistema

A construção do Data Warehouse, começou por uma detalhada análise das fontes, bem como reuniões com o cliente de forma a recolher os requisitos.
De seguida, quando o plano de projeto já se encontrava totalmente definido, foi criado o modelo dimensional e iniciada a construção do ETL. Durante a construção do ETL, foi criada uma área de retenção responsável por garantir a integridade do Data Warehouse.
Por ultimo, foi implementado o Data Warehouse sendo seguidamente povoado.
Deste modo, podemos concluir que o sistema evoluiu de forma fluida.

Bibliografia

Kimball, R. (2008). The data warehouse lifecycle toolkit. 1st ed. Indianapolis, IN: Wiley Pub.

Kimball, R. and Caserta, J. (2004). The data warehouse ETL toolkit. 1st ed. Indianapolis, IN: Wiley.

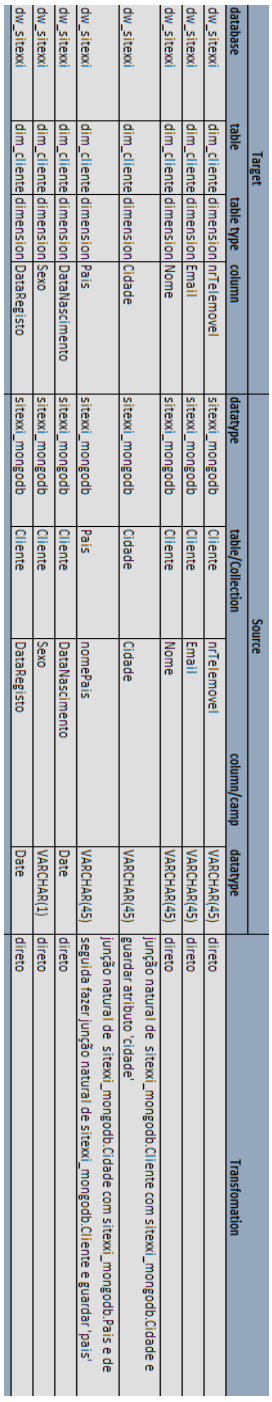
Golfarelli, M. and Rizzi, S. (2009). Data warehouse design: modern principles and methodologies. 1st ed. New York: McGraw-Hill.

Lista de Siglas e Acrónimos

BD	Base de Dados
AR	Área de Retenção
DW	Data Warehouse
ETL	Extract Transform Load
DSA	Data Staging Area

Anexos

I. Anexo 1



Target				Source				
database	table	table type	column	datatype	table/Collection	column/camp	datatype	Transformation
dw_sitexi	dim_cliente	dimension	nrTelemovel	sitexi_mongodb	Cliente	nrTelemovel	VARCHAR(45)	direto
dw_sitexi	dim_cliente	dimension	Email	sitexi_mongodb	Cliente	Email	VARCHAR(45)	direto
dw_sitexi	dim_cliente	dimension	Nome	sitexi_mongodb	Cliente	Nome	VARCHAR(45)	direto
dw_sitexi	dim_cliente	dimension	Cidade	sitexi_mongodb	Cidade	Cidade	VARCHAR(45)	junção natural de sitexi_mongodb.Cliente com sitexi_mongodb.Cidade e guardar atributo 'cidade'
dw_sitexi	dim_cliente	dimension	Pais	sitexi_mongodb	Pais	nomePais	VARCHAR(45)	junção natural de sitexi_mongodb.Cidade com sitexi_mongodb.Pais e de seguida fazer junção natural de sitexi_mongodb.Cliente e guardar 'pais'
dw_sitexi	dim_cliente	dimension	DataNascimento	sitexi_mongodb	Cliente	DataNascimento	Date	direto
dw_sitexi	dim_cliente	dimension	Sexo	sitexi_mongodb	Cliente	Sexo	VARCHAR(1)	direto
dw_sitexi	dim_cliente	dimension	DataRegisto	sitexi_mongodb	Cliente	DataRegisto	Date	direto

Tabela 7: Ilustração 15: Mapa de fontes de dados dimensão cliente

Target				Source				Transfomation
database	table	table type	column	datatype	table/Collection	column/camp	datatype	
dw_sitexxi	dim_cliente	dimension	nrTelemovel	sitexxi_mysql	Cliente	nrTelemovel	VARCHAR(45)	direto
dw_sitexxi	dim_cliente	dimension	Email	sitexxi_mysql	Cliente	Email	VARCHAR(45)	direto
dw_sitexxi	dim_cliente	dimension	Nome	sitexxi_mysql	Cliente	Nome	VARCHAR(45)	direto
dw_sitexxi	dim_cliente	dimension	Cidade	sitexxi_mysql	Cidade	Cidade	VARCHAR(45)	junção natural de sitexxi_mysql.Cliente com sitexxi_mysql.Cidade e guardar atributo 'cidade'
dw_sitexxi	dim_cliente	dimension	Pais	sitexxi_mysql	Pais	nomePais	VARCHAR(45)	junção natural de dw_seculoXII_V1.Cidade com dw_seculoXII_V1.Pais e de seguida fazer junção natural de dw_seculoXXI_V1.Cliente e guardar 'pais'
dw_sitexxi	dim_cliente	dimension	DataNascimento	sitexxi_mysql	Cliente	DataNascimento	Date	direto
dw_sitexxi	dim_cliente	dimension	Sexo	sitexxi_mysql	Cliente	Sexo	VARCHAR(1)	direto
dw_sitexxi	dim_cliente	dimension	DataRegisto	sitexxi_mysql	Cliente	DataRegisto	Date	direto
dw_sitexxi	dim_cliente	dimension	id_cliente					Surrogate Key
dw_sitexxi	dim_cliente	dimension	nrTelemovel	sitexxi_mongodb	Cliente	--	--	meter "desconhecido"
dw_sitexxi	dim_cliente	dimension	nrTelemovel	sitexxi_mysql	Cliente	--	--	meter "desconhecido"

Tabela 8: Ilustração 8: Mapa de fontes dados Dimensão cliente 2

Target				Source				Transfomation
database	table	table type	column	datatype	table/Collection	column/camp	datatype	
dw_sitexxi	dim_jogo	dimension	Nome	sitexxi_mysql	Jogo	nome	VARCHAR(45)	direto
dw_sitexxi	dim_jogo	dimension	dataLancamento	sitexxi_mysql	Jogo	dataLancamento	Date	direto
dw_sitexxi	dim_jogo	dimension	idadeParaJogar	sitexxi_mysql	Jogo	idadeParaJogar	INT	direto
dw_sitexxi	dim_jogo	dimension	quantidade	sitexxi_mysql	Jogo	quantidade	INT	direto
dw_sitexxi	dim_jogo	dimension	precoBase	sitexxi_mysql	Jogo	precoBase	Decimal(10,5)	direto
dw_sitexxi	dim_jogo	dimension	produtor	sitexxi_mysql	Produtor	nome	VARCHAR(45)	junção natural de sitexxi_mysql.Jogo com sitexxi_mysql.Produtor e guardar atributo 'Produtor.nome'
dw_sitexxi	dim_jogo	dimension	id_jogo					Surrogate Key

Tabela 9: Mapa de fontes dados dimensão jogo

Target				Source				Transfomation
database	table	table type	column	datatype	table/Collection	column/camp	datatype	
dw_sitexxi	dim_jogo	dimension	Nome	sitexxi_mongodb	Jogo	nome	VARCHAR(45)	direto
dw_sitexxi	dim_jogo	dimension	dataLancamento	sitexxi_mongodb	Jogo	dataLancamento	Date	direto
dw_sitexxi	dim_jogo	dimension	idadeParaJogar	sitexxi_mongodb	Jogo	idadeParaJogar	INT	direto
dw_sitexxi	dim_jogo	dimension	quantidade	sitexxi_mongodb	Jogo	quantidade	INT	direto
dw_sitexxi	dim_jogo	dimension	precoBase	sitexxi_mongodb	Jogo	precoBase	Decimal(10,5)	direto
dw_sitexxi	dim_jogo	dimension	produtor	sitexxi_mongodb	Produtor	nome	VARCHAR(45)	junção natural de sitexxi_mongodb.Jogo com sitexxi_mongodb.Produtor e guardar atributo 'Produtor.nome'

Tabela 10: Mapa de fontes dados Dimensão jogo 2

Target				Source				Transfomation
database	table	table type	column	datatype	table/Collection	column/camp	datatype	
dw_sitexxi	fact_venda	Fact	quantidade	sitexxi_mysql	Compras	quantidade	INT	direto
dw_sitexxi	fact_venda	Fact	quantidade	sitexxi_mongodb	Compras	quantidade	INT	direto
dw_sitexxi	fact_venda	Fact	precoUnidade	sitexxi_mysql	Jogo	precoBase	Decimal(10,5)	junção natural de sitexxi_mysql.Jogo com sitexxi_mysql.Compras e guardar atributo "precoBase"
dw_sitexxi	fact_venda	Fact	precoUnidade	sitexxi_mongodb	Jogo	precoBase	Decimal(10,5)	junção natural de sitexxi_mongodb.Jogo com sitexxi_mongodb.Compras e guardar atributo "precoBase"
dw_sitexxi	fact_venda	Fact	precoTotal	sitexxi_mysql	Compras	precoVendido	Decimal(10,5)	direto
dw_sitexxi	fact_venda	Fact	precoTotal	sitexxi_mongodb	Compras	precoVendido	Decimal(10,5)	direto
dw_sitexxi	fact_venda	Fact	descontoDaVenda	sitexxi_mysql	Compras	desconto	Decimal(10,5)	direto
dw_sitexxi	fact_venda	Fact	descontoDaVenda	sitexxi_mongodb	Compras	desconto	Decimal(10,5)	direto
dw_sitexxi	fact_venda	Fact	id_venda					Surrogate key
dw_sitexxi	fact_venda	Fact	idadeCliente	sitexxi_mysql	Cliente	idadeCliente	INT	junção natural de sitexxi_mysql.Cliente com sitexxi_mysql.Compras e guardar atributo "idadeCliente"
dw_sitexxi	fact_venda	Fact	idadeCliente	sitexxi_mongodb	Cliente	idadeCliente	INT	junção natural de sitexxi_mongodb.Cliente com sitexxi_mongodb.Compras e guardar atributo "idadeCliente"

Tabela 11: Fonte Mapa Destino Tabela Facto