TÉCNICO LISBOA

Data2Help: Integração e Limpeza de Dados

José Eduardo Alves Costa

Dissertação para obtenção do Grau de Mestre em

Engenharia Informática e de Computadores

Orientadores: Prof. Helena Isabel De Jesus Galhardas
Prof. Vasco Miguel Gomes Nunes Manquinho

Júri

Presidente: Prof. Daniel Jorge Viegas Gonçalves
Orientador: Prof. Helena Isabel De Jesus Galhardas
Vogal: Prof. Rui Miguel Carrasqueiro Henriques

Janeiro 2021

Agradecimentos

É com grande felicidade que termino este longo percurso e não poderia deixar de agradecer a todos aqueles que me ajudaram e apoiaram nesta jornada. Certamente alguém será esquecido, mas, desde já, fica o meu agradecimento a todos aqueles que me ajudaram tanto de forma direta ou indireta na conceção deste documento.

Em primeiro lugar agradecer muito aos meus orientadores, o Professor Vasco Manquinho e a Professora Helena Galhardas, por me escolherem para fazer esta dissertação num tema tão importante para Portugal e que poderá oferecer contributos benéficos para a sociedade. Agradecer-lhes também pela orientação e apoio contínuo durante um ano tão diferente e difícil como 2020. Uma palavra especial também ao professor Rui Henriques pelo apoio e colaboração direta no trabalho.

Ao Instituto Superior Técnico, por me proporcionar uma formação de excelência e por me ter dado tão bons colegas, companheiros de estudo e amigos.

Aos meus amigos por todo o apoio, amizade e bons momentos!

À minha família, que me educou e apoiou em todos os momentos da minha vida.

E por fim, mas não menos importante, à minha namorada pelo apoio, amor e carinho que me foi dado ao longo destes últimos anos e me permitiu superar todos os desafios.

Este trabalho foi parcialmente apoiado por fundos nacionais através da FCT, Fundação para a Ciência e Tecnologia, no âmbito dos projetos UIDB / 50021/2020, DSAIPA / AI / 0044/2018.

Resumo

Em Portugal Continental os serviços de emergências médicas são coordenados pelo Instituto Nacional de Emergência Médica (INEM). A produtividade operacional do INEM é muito importante para o país, visto que pode fazer a diferença entre a morte e a vida de um cidadão. O projeto Data2Help pretende criar ferramentas para otimizar a alocação de recursos, melhorando a qualidade e o tempo de resposta às emergências médicas. Uma das tarefas do projeto passa pela integração de fontes de dados do INEM com fontes de dados externas. Dessa forma podem ser correlacionados dados meteorológicos, de eventos desportivos ou eventos musicais com a quantidade de ocorrências em certos locais. O objetivo desta tese consistiu no desenvolvimento de um repositório integrado de dados, ou Data Warehouse, que armazena dados históricos que permitirá encontrar correlações entre os dados e a aplicação de modelos preditivos usando algoritmos de ciência de dados.

Para realizar a integração de dados foi necessário um levantamento de requisitos, onde se definiram as fontes de dados a utilizar no projeto, bem como um conjunto de consultas a que o Data Warehouse deve responder. De seguida, com base nas consultas identificadas, desenvolveu-se o Modelo Muldimensional do Data Warehouse. Por fim, foi desenvolvido um Processo ETL que permitiu carregar os dados para o Data Warehouse.

Palavras-chave: Integração de Dados, Emergências Médicas, Limpeza de Dados , Modelo Multidimensional, Data Mart, Processo ETL, Data Warehouse

Abstract

In mainland Portugal, emergency medical services are coordinated by the Instituto Nacional de Emergências Médicas (INEM). INEM's operational productivity is very important for the country, making the difference in saving citizens' lives. The Data2Help project intends to create tools to optimize the allocation of resources, improving the quality and the response time of medical emergencies. One of the project tasks is the integration of INEM data sources with external data sources. This way, weather data, sports events data or musical events data can be correlated with the number of occurrences in certain places. The goal of this thesis was the development of an integrated data repository, or Data Warehouse, which stores historical data that will allow to find data correlations between the data and to apply predictive models using data science algorithms.

To carry out the data integration task, a requirements analysis was necessary, in which the data sources to be used in the project were defined, as well as a set of queries to which the Data Warehouse should answer. Then, based on the identified queries, the Multidimensional Model of the Data Warehouse was developed. Finally, an ETL Process was developed to load data to the Data Warehouse.

Keywords: Data Integration, Data Cleaning, Medical Emergencies, Multidimensional Model, Data Mart, ETL Process, Data Warehouse

Conteúdo

Agradecimentos iii

Resumo v

Abstract vii

Lista de Figuras xvi

Lista de Tabelas xvii

1 Introdução **1**

1.1 Objetivos 2

1.2 Contribuições 2

1.3 Metodologia de Validação 2

1.4 Organização do Documento 3

2 Conceitos Básicos **5**

2.1 Conceitos Técnicos 5

2.2 Conceitos do Domínio 8

3 Trabalho Relacionado **11**

3.1 Projetos 11

3.1.1 Projeto DASH 11

3.1.2 Estudo de simulação para melhorar o desempenho dos serviços de emergência médica 16

3.1.3 Plataforma para Apoio à Tomada de Decisão em Situações de Emergência 18

3.1.4 Discussão 20

3.2 Software para Integração de Dados 21

3.2.1 Ferramentas de Software 21

3.2.2 Discussão 23

4 Solução **25**

4.1 Levantamento de Requisitos 25

4.1.1 Fontes de dados 25

4.1.2 Consultas apenas com dados do SIADEM 27

4.1.3 Consultas com integração de dados externos 28

4.2 Arquitetura da Solução 28
4.3 *Data Staging Area* 29
4.4 Modelo Multidimensional Data Warehouse 30
4.4.1 Data Mart Todas as Ocorrências 34
4.4.2 Data Mart Ocorrências com Meios 35
4.4.3 Data Mart Ocorrências com Informação Completa 36
4.4.4 Data Mart Futebol 37
4.4.5 Data Mart Concertos 38
4.4.6 Data Mart Festivais 39
4.4.7 Data Mart Meteorologia 41
4.5 Processo ETL 42
4.5.1 Dimensão Tempo 42
4.5.2 Dimensão Localização 43
4.5.3 Dimensão Tipo de Emergência 44
4.5.4 Data Mart Todas as Ocorrências 45
4.5.5 Data Mart Ocorrências com Meios 46
4.5.6 Data Mart Ocorrências com Informação Completa 50
4.5.7 Data Mart Futebol 51
4.5.8 Data Mart Concertos 55
4.5.9 Data Mart Festivais 55
4.5.10 Data Mart Meteorologia 55
4.5.11 *Jobs* 56

5 Validação Experimental **59**
5.1 Validação das Consultas 59
5.1.1 Validação das consultas apenas com dados do SIADEM 59
5.1.2 Validação das consultas com dados do SIADEM e dados externos 67
5.2 Desempenho das Consultas 75
5.3 Dados do Data Warehouse vs Dados da Base de Dados do SIADEM 75

6 Conclusões **77**
6.1 Trabalho Futuro 78

Bibliografia **80**

A Trabalho Relacionado **81**

B Modelo Multidimensional **83**
B.1 Programa em *Transact-SQL* para criar o Data Warehouse 83
B.2 Representação Gráfica do Modelo Multidimensional 91

C Transformações Data Staging Area **93**
C.1 Transformação para os dados de Futebol 93
C.2 Transformação para os dados de Concertos 94
C.3 Transformação para os dados de Festivais 94

D Processo ETL **95**
D.1 Programa para calcular distâncias através de coordenadas geográficas 95
D.2 Data Mart Todas as Ocorrências 96
D.3 Data Mart Ocorrências com Meios 97
D.4 Data Mart Ocorrências com Informação Completa 99
D.5 Data Mart Ocorrências Futebol 100
D.6 Data Mart Ocorrências Concertos 102
D.7 Data Mart Ocorrências Festivais 103
D.8 Data Mart Ocorrências Meteorologia 104

E Consultas realizadas à base de dados do SIADEM **107**

F Guia do Programador **113**
F.1 Acesso ao Data Warehouse / Data Staging Area 113
F.1.1 Alterações no Data Warehouse 113
F.2 Acesso aos ficheiros do Pentaho Data Integration 113

Lista de Figuras

2.1 Sistema virtual de integração de dados 5
2.2 Arquitetura da infraestrutura tecnológica desde a integração materializada de dados até à análise de dados 7

3.1 Ecrã inicial da aplicação web, mostrando a distribuição geográfica das ocorrências, e os possíveis filtros a aplicar para interagir com a aplicação [15] 20
3.2 Quadrante Mágico da Gartner para Ferramentas de Integração de Dados [8] 21

4.1 Arquitetura da Solução 29
4.2 Data Staging Area Data2Help 29
4.3 Matriz em Bus Data2Help 30
4.4 Modelo Muldimensional Data Mart com todas as ocorrências 34
4.5 Modelo Muldimensional Data Mart Ocorrências com Meios 35
4.6 Modelo Muldimensional Data Mart Ocorrências com Informação Completa 37
4.7 Modelo Muldimensional Data Mart Futebol 38
4.8 Modelo Muldimensional Data Mart Concertos 39
4.9 Modelo Muldimensional Data Mart Festivais 40
4.10 Modelo Muldimensional Data Mart Meteorologia 41
4.11 Transformação para inserir os tempos de início dos jogos de futebol na dimensão Tempo 43
4.12 Transformação para inserir as localizações das ocorrências na dimensão Localização . . 44
4.13 Transformação para carregar dados para a dimensão Tipo de Emergência 45
4.14 Transformação para carregar os dados na tabela de factos do Data Mart Todas as Ocorrências 46
4.15 Transformação para carregar dados na dimensão Unidade 47
4.16 Transformação para inserir o atributo estação quando este não existe 48
4.17 Transformação para carregar dados na dimensão Grupo de Unidades no Data Mart Ocorrências com Meios 48
4.18 Transformação para carregar dados na dimensão Destino 49
4.19 Transformação para carregar dados na tabela de factos do Data Mart Ocorrências com Unidades 49
4.20 Transformação para carregar dados na dimensão Equipas 51

4.21 Transformação para carregar dados na dimensão Competição 52
4.22 Transformação para carregar dados na dimensão Futebol 52
4.23 Transformação para carregar dados na tabela de factos do Data Mart Futebol 54
4.24 Job que executa as transformações para carregar os dados no Data Mart Todas as Ocorrências 56

5.1 Transformação para inserir num ficheiro *CSV* as ocorrências ao longo do tempo numa zona próxima a um estádio num intervalo de tempo em que tenha decorrido um jogo . . . 68
5.2 Gráfico para mostrar o número de ocorrências ao longo do tempo junto ao Estádio da Luz às sextas feiras entre as 19h:30m e as 23h:30m 69
5.3 Gráfico para mostrar o número de ocorrências ao longo do tempo junto à Altice Arena às sextas feiras entre as 18 horas e as 00 horas de sábado 70
5.4 Gráfico para mostrar o número de ocorrências ao longo do tempo junto à Quinta da Atalaia às sextas feiras, sabádos e domingos, entre dia 24 de Agosto de 2018 e dia 28 de Setembro de 2018 72
5.5 Gráfico para mostrar o número de ocorrências ao longo do tempo junto à estação 762 e que ocorreram à segunda feira, entre o dia 17 de Dezembro de 2018 e o dia 25 de Fevereiro de 2019 73
5.6 *Job para comparar os ficheiros obtidos pelas consultas que extraem todas as ocorrências, seus respetivos tipos e prioridades da Base de Dados do SIADEM e do Data Warehouse* 76

A.1 Modelo Conceptual dos Processo do SAMU 94 [1] 81

B.1 Representação Gráfica do Modelo Multidimensional do Data Warehouse do Projeto Data2Help 92

C.1 Transformação para extrair os dados de futebol da *API* e carregá-los na Data Staging Area 93
C.2 Transformação para carregar o conjunto de dados de concertos na Data Staging Area . . 94
C.3 Transformação para carregar o conjunto de dados de festivais na Data Staging Area . . . 94

D.1 Programa utilizado para calcular a distância entre as ocorrências e os estádios onde se realizam os jogos de futebol 95
D.2 Transformação para carregar os tempos de início das ocorrências na dimensão Tempo . 96
D.3 Transformação para carregar a localização das ocorrências na dimensão Localização . . 96
D.4 Transformação para inserir o Distrito, Concelho e Freguesia da localização das ocorrências na dimensão Localização 96
D.5 Transformação para carregar os dados na dimensão Tipo de Emergência 96
D.6 Transformação para carregar os dados na tabela de factos do Data Mart Todas as Ocorrências 97
D.7 Job que executa as transformações para carregar os dados no Data Mart Todas as Ocorrências 97

D.8 Transformação para carregar os tempos de acionamento do primeiro meio na dimensão Tempo . . . 97
D.9 Transformação para carregar unidades na dimensão Unidade . . . 97
D.10 Transformação para carregar as restantes unidades na dimensão Unidade . . . 97
D.11 Transformação para inserir o atributo estação quando este não existe . . . 98
D.12 Transformação para carregar os destinos dos primeiros meios enviados para cada ocorrência . . . 98
D.13 Transformação para carregar dados na tabela de factos do Data Mart Ocorrências com Meios . . . 98
D.14 Transformação para carregar os dados na dimensão Grupo de Unidades . . . 98
D.15 Job que executa as transformações para carregar os dados no Data Mart Ocorrências com Meios . . . 99
D.16 Transformação para carregar os tempos de chegada do primeiro meio na dimensão Tempo 99
D.17 Transformação para carregar os tempos de saída do primeiro meio do local da ocorrência na dimensão Tempo . . . 99
D.18 Transformação para carregar os tempos de chegada do primeiro meio ao destino na dimensão Tempo . . . 99
D.19 Transformação para carregar dados na tabela de factos do Data Mart Ocorrências com Informação Completa . . . 100
D.20 Transformação para carregar os dados na dimensão Grupo de Unidades . . . 100
D.21 Job que executa as transformações para carregar os dados no Data Mart Ocorrências com Informação Completa . . . 100
D.22 Transformação para carregar os tempos de início dos jogos de futebol dimensão Tempo . 100
D.23 Transformação para carregar as localizações dos estádios de futebol na dimensão Localização . . . 101
D.24 Transformação para carregar os dados na dimensão Competição . . . 101
D.25 Transformação para carregar os dados na dimensão Equipas . . . 101
D.26 Transformação para carregar os dados na dimensão Futebol . . . 101
D.27 Transformação para carregar os dados na tabela de factos do Data Mart Futebol . . . 101
D.28 Job que executa as transformações para carregar os dados no Data Mart Futebol . . . 102
D.29 Transformação para carregar os tempos de início dos concertos na dimensão Tempo . . 102
D.30 Transformação para carregar as localizações dos concertos na dimensão Localização . . 102
D.31 Transformação para carregar os dados na dimensão Concertos . . . 102
D.32 Transformação para carregar os dados na tabela de factos do Data Mart Concertos . . . 102
D.33 Job que executa as transformações para carregar os dados no Data Mart Concertos . . . 103
D.34 Transformação para carregar os tempos de início dos festivais na dimensão Tempo . . . 103
D.35 Transformação para carregar os tempos de fim dos festivais na dimensão Tempo . . . 103
D.36 Transformação para carregar as localizações dos festivais na dimensão Localização . . . 103
D.37 Transformação para carregar os dados na dimensão Festivais . . . 103

D.38 Transformação para carregar os dados na tabela de factos do Data Mart Festivais 104
D.39 Job que executa as transformações para carregar os dados no Data Mart Festivais 104
D.40 Transformação para carregar os tempos das medições meteorológicas na dimensão Tempo 104
D.41 Transformação para carregar as localizações das estações meteorológicas na dimensão Localização 104
D.42 Transformação para carregar os dados na dimensão Meteorologia 104
D.43 Transformação para carregar os dados na tabela de factos do Data Mart Meteorologia . . 105
D.44 Job que executa as transformações para carregar os dados no Data Mart Meteorologia . 105

Lista de Tabelas

5.1 Tabela de comparação dos tempos de execução das consultas listadas na Secção 4.1.2 75

Capítulo 1

Introdução

Em Portugal os serviços de emergências médicas são coordenados pelo Instituto Nacional de Emergência Médica (INEM)[1]. Por norma, as situações de emergência médica são comunicadas ao INEM através de chamadas telefónicas para o número 112, onde pessoal especializado toma conta da ocorrência e decide quais os meios apropriados para cada emergência médica. O Centro de Orientação de Doentes Urgentes (CODU)[2] é a divisão do INEM responsável pelo atendimento do 112.

A produtividade operacional do INEM é muito importante para Portugal, pelo que é necessário reduzir o máximo possível o tempo de resposta a uma emergência médica, que pode fazer a diferença entre a morte e a vida de um cidadão. Existem dois momentos cruciais nas operações realizadas pelo INEM: o primeiro momento está relacionado com o tempo de demora a atender cada chamada para o 112 e o segundo momento com o tempo que o veículo de emergência demora desde o despacho até que chega ao local da emergência.

Com o projeto Data2Help pretende-se criar ferramentas para otimizar a alocação de recursos, melhorando desse modo a qualidade e o tempo de resposta às emergências médicas em Portugal continental. Os principais objetivos do projeto são: (i) prever a carga de trabalho do CODU; (ii) otimizar as escalas do pessoal do CODU de acordo com a carga de trabalho esperada; (iii) desenvolver modelos preditivos para as solicitações de veículos de emergência em cada área geográfica; (iv) desenvolver software para otimizar o número de pessoas e veículos de emergência ativos em cada turno de trabalho.

Para conseguir otimizar as operações dos serviços de emergência médica, o projeto Data2Help propõe aplicar e desenvolver algoritmos avançados de análise de dados, bem como novos modelos e algoritmos eficientes para planeamento e escalonamento de recursos. Para além de diversas contribuições científicas, um protótipo funcional será integrado no fluxo operacional do INEM de forma a testar e validar as ferramentas desenvolvidas no projeto.

De modo a atingir os objetivos, o projeto Data2Help irá integrar dados do INEM relacionados com ocorrências com outros dados de informação pública, como dados dados meteorológicos ou dados de eventos desportivos e musicais. É no desenvolvimento da tarefa de integração de dados que este

[1] https://www.inem.pt/
[2] https://www.inem.pt/CODU/

documento se irá focar.

1.1 Objetivos

O principal objetivo da tarefa de integração de dados no projeto Data2Help é integrar fontes de dados do INEM com outras fontes (fontes de dados meteorológicos, de eventos musicais, de eventos desportivos, entre outras igualmente relevantes). Como resultado da integração, será construído um repositório integrado contendo dados históricos de emergências médicas, tempos de resposta das equipas médicas, veículos despachados, além de outras informações sobre a resposta operacional do INEM. Os dados externos relevantes que possam estar correlacionados com o número de ocorrências também estarão no repositório integrado.

O repositório integrado de dados irá conter dados que alimentam modelos preditivos e de correlação. Assim sendo, é importante garantir que o repositório permite que seja possível executar sobre si um conjunto bem delineado de consultas, desenvolvido em colaboração direta com os responsáveis da próxima etapa do projeto. A qualidade dos dados também é um fator importante a ter em conta, ou seja, não devem existir dados incorretos ou incompletos.

Na tarefa de integração de dados será também implementado um mecanismo para garantir que os dados são atualizados periodicamente.

1.2 Contribuições

O conjunto de contruibuições que a tarefa de integração de dados trouxe ao projeto Data2Help são:

- Identificação de um conjunto de consultas a que os dados guardados no repositório devem conseguir responder. As consultas foram obtidas com a colaboração dos responsáveis pela próxima fase do projeto, através da observação das fontes de dados.
- Desenho e implementação do repositório integrado de dados contendo dados do SIADEM e dados externos. A implementação do repositório permite que haja carregamento de novos dados para o atualizar.
- Desenvolvimento de um processo ETL que permite a extração dados das fontes e carregamento de dados para o repositório integrado.

1.3 Metodologia de Validação

Para validar a tarefa de integração de dados no projeto Data2Help é necessário verificar se os principais objetivos propostos para a tarefa foram concretizados. Na fase de levantamento de requisitos foram definidas um conjunto de consultas a que os dados integrados deveriam responder, é fundamental verificar se é possível executar essas consultas. Depois, é analisado o desempenho das consultas

efetuadas sobre os dados integrados em comparação com a execução dessas consultas sobre a base de dados do SIADEM. Por fim, é necessário verificar que os dados integrados estão de acordo com os dados da fonte original, que não houve perda de dados nem dados incorretos.

1.4 Organização do Documento

Este documento está organizado do seguinte modo: o Capítulo 2 descreve os conceitos básicos para um melhor enquadramento com o tema de integração de dados e com o tema das emergências médicas. O Capítulo 3 descreve o trabalho de pesquisa realizado para a melhor compreensão da tarefa de integração de dados, nesta parte são abordados trabalhos relacionados com os objetivos da tarefa de integração de dados do projeto Data2Help, e também é apresentada uma análise a ferramentas de software para realizar a integração. O capítulo 4 descreve a solução desenvolvida para realizar a integração de dados. O Capítulo 5 apresenta validação experimental levada a cabo assim como os resultados obtidos. No Capítulo 6, encontram-se as conclusões obtidas, bem como propostas de trabalho futuro a realizar.

Capítulo 2

Conceitos Básicos

Este capítulo apresenta os conceitos básicos relacionados com a tarefa de integração e limpeza de dados no projeto Data2Help. Os conceitos básicos apresentados serão divididos em conceitos técnicos relacionados com a integração de dados (Secção 2.1) e conceitos do domínio relacionados com emergências médicas (Secção 2.2).

2.1 Conceitos Técnicos

A integração de dados consiste num conjunto de técnicas que permitem um acesso uniforme a um conjunto de fontes de dados autónomas e heterogéneas, que podem ser controladas por diferentes pessoas ou organizações [6].

O sistema de integração de dados pode ter duas formas: virtual ou materializada. Ambas apresentam um esquema para consultas uniforme. O objetivo destes sistemas é uniformizar os diferentes formatos de dados presentes nas várias fontes.

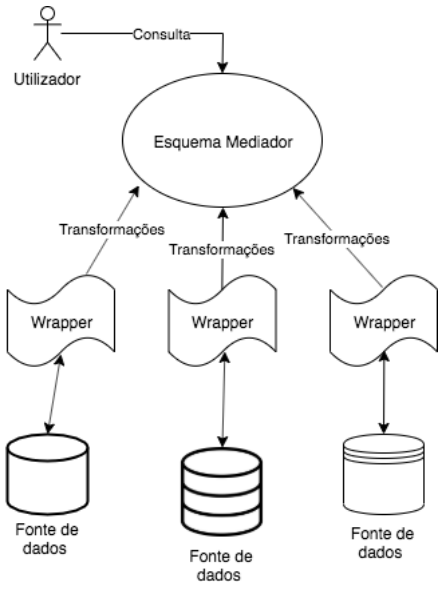

Figura 2.1: Sistema virtual de integração de dados

A Figura 2.1 representa um sistema virtual de integração de dados. Neste tipo de sistemas os utilizadores executam as consultas sobre um *esquema mediador*, que contém apenas um subconjunto de atributos relevantes para o seu domínio. Estes sistemas têm a capacidade de converter as consultas dos utilizadores em novas consultas colocadas diretamente às fontes de dados. As propriedades das fontes de dados necessárias para utilizar os dados são conhecidas por estes sistemas. Os sistemas conhecem também os mapeamentos semânticos que mostram a correspondência entre os atributos das fontes de dados e os atributos do esquema mediador.

A tarefa da comunicação entre o mediador e as fontes de dados é realizada por um software integrado no sistema, conhecido por *wrapper*. O wrapper funciona como um tradutor, enviando as consultas às fontes de dados e traduzindo as respostas de forma a que estas possam depois ser processadas pelo sistema de integração de dados.

A principal vantagem da integração virtual de dados é que existe a garantia de que os dados estão sempre atualizados, pois as consultas são executadas diretamente às fontes de dados. Já a principal desvantagem está relacionada com o custo das consultas às fontes de dados.

Um sistema de integração de dados materializado consegue-se através da integração de várias fontes de dados num único repositório de dados, que é chamado *Data Warehouse*. Este repositório tem como objetivo armazenar informação útil para uma dada organização, sendo uma parte do processo de apoio à tomada de decisão e facilitando o processo de análise de dados [19]. É suposto que um Data Warehouse guarde registos de uma perspetiva histórica, ou seja, deve ser possível armazenar dados de vários anos e uma dimensão tempo deve estar presente. O Data Warehouse deve ser refrescado de forma periódica, ao serem carregados novos dados existentes nas fontes.

Para definir a organização dos dados armazenados num Data Warehouse é utilizada a *Modelação Multidimensional* [23]. Um Modelo Multidimensional organiza-se em torno de factos, associados a um conjunto de atributos, que por sua vez estão organizados em diferentes dimensões. Os *factos* são o foco do que se pretende analisar, são normalmente valores numéricos mensuráveis, como por exemplo o número de unidades vendidas numa loja ou a temperatura registada. Os factos podem estar relacionados com um conjunto de atributos que os caracterizam sob várias perspetivas como, por exemplo, a temperatura de um dado local a uma dada hora. Quando vários atributos estão relacionados com a mesma propriedade do acontecimento estamos perante uma *dimensão*.

O Modelo Multidimensional pode ser implementado num sistema de gestão de bases de dados relacionais. Se assim for, o registo de dados é organizado em tabelas. Os factos e as dimensões são guardados em tabelas diferentes. As tabelas de factos correspondem ao assunto que pretendemos analisar e as tabelas de dimensão adicionam contexto à tabela de factos [19]. Existem vários tipos de esquemas para esta modelação: esquemas em estrela, em floco de neve ou em constelação.

O *esquema em estrela* é composto por uma única tabela de factos, que constitui o centro da estrela. Esta, por sua vez, liga-se a múltiplas tabelas de dimensão. As tabelas de dimensão estão apenas ligadas à tabela de factos. Os *esquemas em floco de neve* são também compostos por uma tabela de factos ligada a múltiplas tabelas de dimensão. Diferem do esquema em estrela na medida em que permitem que uma dimensão seja representada por mais do que uma tabela, ou seja, as tabelas de

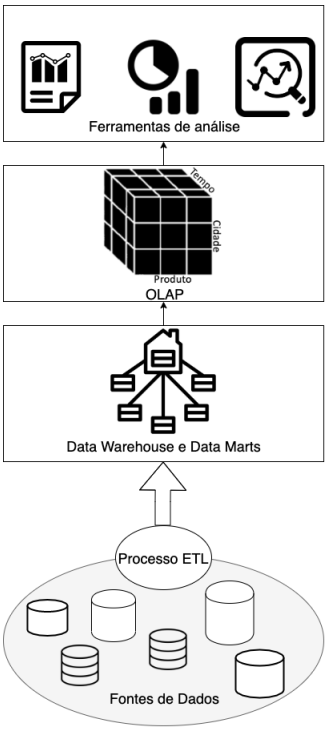

Figura 2.2: Arquitetura da infraestrutura tecnológica desde a integração materializada de dados até à análise de dados

dimensão contêm dados normalizados. Quando uma dimensão é normalizada, alguns atributos passam para uma nova tabela, que mantém uma ligação com a primeira. Um *esquema em constelação* consiste em várias tabelas de factos que partilham dimensões.

Na Figura 2.2 está representada a arquitetura típica desde o processo de construção de um Data Warehouse até à análise dos dados presentes nesse Data Warehouse.

Como está representado na Figura 2.2, o Data Warehouse é o resultado da execução de um processo sobre várias fontes de dados. O processo é conhecido por *Processo ETL* e é composto por 3 etapas: Extração, Transformação e Carregamento de dados [6]. No processo de *Extração* são extraídos dados de várias fontes heterogéneas. No processo de *Transformação* os dados são transformados e modificados para garantir que ficam com a qualidade necessária. Durante esta transformação os dados podem passar por diferentes tipos de processos: limpeza, agregação, formatação e até mesmo construção de novos dados. Por fim é feito o *Carregamento* dos dados já transformados para o Data Warehouse. Este processo ETL é realizado com o apoio de uma ferramenta de software apropriada, como por exemplo: Pentaho Data Integration [1] ou Oracle Data Integrator [2]. Como foi referido anteriormente, o resultado final deste processo é um Data Warehouse que abrange toda uma organização. Para além do processo ETL, os softwares permitem implementar um processo semelhante para atualização periódica do Data Warehouse. A principal diferença entre os processos está relacionada com a fase de extração, uma vez que o processo de atualização apenas extrai novos dados ou dados atualizados.

Recorrendo ao mesmo processo ETL do qual resulta o Data Warehouse, podem ser obtidos *Data*

[1] https://www.hitachivantara.com/en-us/products/data-management-analytics/pentaho-platform/pentaho-data-integration.html
[2] https://www.oracle.com/middleware/technologies/data-integrator.html

Marts. Data Marts são Data Warehouses de tamanho inferior, sendo que são especializados num determinado domínio ou departamento de uma organização, podendo fornecer apoio à tomada de decisão no seu domínio específico.

No nível "OLAP" da arquitetura, como representado na Figura 2.2, encontra-se representado o sistema *Online Analytical Processing (OLAP)*, utilizado para explorar o Data Warehouse. Os sistemas OLAP baseiam-se no modelo multidimensional que deu origem ao Data Warehouse para estruturar os dados na forma de um cubo. As operações dos sistemas OLAP permitem aos utilizadores analisar interativamente a informação contida no Data Warehouse, sob várias perspetivas, permitindo explorar a tabela de factos (as células do cubo) pelas diferentes dimensões (as arestas do cubo). Estas capacidades oferecem um acesso intuitivo e um alto desempenho na análise de grandes volumes de dados [23].

No nível "Ferramentas de Análise" da Figura 2.2 está representada a fase de análise de dados. Nesta fase são utilizadas ferramentas de software que permitem aos seus utilizadores explorar os dados. Existem ferramentas de vários tipos, cujas funcionalidades incluem: geração de relatórios, obter informações estatísticas e também *Data Mining* [19].

As ferramentas para gerar relatórios podem produzir, enviar e gerir relatórios, que podem ser simples ou interativos. As ferramentas para obter informações estatísticas são usadas para analisar e visualizar dados utilizando métodos estatísticos. As ferramentas de Data Mining permitem analisar dados de forma a identificar tendências e padrões, permitindo também que sejam feitas previsões com base nos dados atuais.

A principal vantagem da integração materializada de dados e dos Data Warehouses está relacionada com o alto desempenho das consultas na base materializada. Como principal desvantagem temos a necessidade de atualizar o Data Warehouse sempre que existirem mudanças significativas nas fontes de dados [6].

2.2 Conceitos do Domínio

Nesta secção serão abordados os conceitos relacionados com as Emergências Médicas, com o objetivo de estabelecer uma base de entendimento com o tema do domínio.

Em Portugal existe o *Sistema Integrado de Emergências Médicas (SIEM)* que se define como sendo o conjunto de ações e entidades coordenadas, que cooperam com o objetivo de intervir e prestar assistência a vítimas de acidente ou doença súbita. O SIEM é responsável toda a atividade de emergência, como o sistema de socorro pré-hospitalar, o transporte, a receção no hospital e a adequada referenciação do doente [12]. O *Instituto Nacional de Emergência Médica (INEM)* é o organismo responsável por coordenar o funcionamento do SIEM.

O SIEM é ativado quando alguém telefona para o número europeu de emergência (112) e o atendimento das chamadas cabe às Centrais 112. Sempre que o motivo da chamada esteja relacionado com um problema de saúde, a mesma é encaminhada para os *Centros de Orientação de Doentes Urgentes (CODU)* do INEM.

Os CODU são responsáveis por atender e avaliar, num curto espaço de tempo, os pedidos de socorro recebidos, com o objetivo de determinar os recursos adequados a cada ocorrência. Estes têm também a responsabilidade de dar instruções e aconselhamento de pré-socorro. Todo este processo está a cargo de médicos e técnicos com formação específica.

Os CODU têm à sua disposição diversos meios de comunicação, bem como meios para atuação no terreno, tais como: as ambulâncias, as motas, as Viaturas Médicas de Emergência e Reanimação (VMERs), os helicópteros de emergência médica, entre outros meios. Através de uma análise criteriosa efetuada pelos técnicos dos CODU, estes são responsáveis por acionar os diferentes meios de socorro, apoiá-los durante a prestação de socorro no local das ocorrências e, de acordo com as informações clínicas recebidas das equipas no terreno, selecionar e preparar a receção hospitalar dos diferentes doentes.

Na maior parte dos casos, os meios de socorro acionados são ambulâncias, que estão distribuídas por vários pontos do país, quer em delegações do INEM quer em postos de Bombeiros ou da Cruz Vermelha Portuguesa. Neste momento em Portugal continental existem dois tipos de ambulâncias coordenadas pelos CODU: ambulâncias SBV e ambulâncias SIV. As ambulâncias de *Suporte Básico de Vida (SBV)* são ambulâncias de socorro, destinadas à estabilização e transporte de doentes que necessitem de assistência durante o transporte e cuja tripulação e equipamento permitem a aplicação de medidas de SBV e *Desfibrilhação Automática Externa (DAE)*. As ambulâncias de *Suporte Imediato de Vida (SIV)* são um meio de socorro que para além da estabilização e transporte pré-hospitalar, pode transportar doentes em estado crítico entre hospitais. Estas ambulâncias para além do material das ambulâncias SBV, estão equipadas com diversos fármacos de emergência e acessórios para a sua preparação.

Outro meio para atuação no terreno utilizado pelo INEM são os *Motociclos de Emergência Médica*. Os motociclos são ideais para deslocações em cidade pois são bastante ágeis para combater os congestionamentos no trânsito, permitindo uma chegada mais rápida ao local da ocorrência. Apesar das limitações, a carga do motociclo inclui um DAE, oxigénio, adjuvantes da via aérea e ventilação, equipamento para avaliação de sinais vitais e glicemia capilar, entre outros. Este material permite realizar as medidas iniciais para estabilizar a vítima, até que se reúnam as condições ideais para o seu transporte.

A *Viatura Médica de Emergência e Reanimação (VMER)* é um veículo de intervenção pré-hospitalar, destinado ao transporte rápido de uma equipa médica até ao local da ocorrência. Tem como principal objetivo a estabilização pré-hospitalar e o acompanhamento médico durante o transporte das vítimas.

Os *helicópteros de emergência médica* têm como missão o transporte de doentes graves entre unidades de saúde ou entre o local da ocorrência e uma unidade de saúde. Estão equipados com material de SAV.

Todos os meios para atuação no terreno têm uma base fixa e estão distribuídos em vários pontos de Portugal continental.

Capítulo 3

Trabalho Relacionado

Neste capítulo analisaremos trabalhos publicados cujos temas estão relacionados com o projeto Data-2Help e com a integração de dados (Secção 3.1). O capítulo também aborda e compara ferramentas de software com capacidade para executar a integração de dados na Secção 3.2.

3.1 Projetos

Na Secção 3.1.1 é apresentado sobretudo um projeto realizado em Londres para melhorar os serviços de emergência médica. Na Secção 3.1.2 é apresentado um estudo de simulação para melhorar o desempenho dos serviços de emergência médica em França. Na Secção 3.1.3 é apresentada uma plataforma desenvolvida em Espanha para apoio à tomada de decisão em situações de emergência. Na Secção 3.1.4 são abordados os projetos das secções anteriores, e são extraídas ilações que podem ser aplicadas ao projeto Data2Help.

3.1.1 Projeto DASH

O objetivo do projeto DASH [7] é melhorar o Serviço de Ambulâncias de Londres (*LAS, London Ambulance Service*). O projeto explorou o potencial impacto da integração de novas fontes e tecnologias de dados na resposta a emergências médicas, sendo que estas fontes de dados poderiam ser externas aos serviços de emergências médicas. A equipa que realizou o projeto começou por perceber que dados eram recolhidos e utilizados pelos serviços de emergência médica e também que novos dados poderiam ser interessantes explorar. Depois de selecionar os novos dados a utilizar, verificaram como se iriam integrar esses dados de forma a que estes pudessem apoiar a tomada de decisão.

No relatório do projeto são referidas seis sugestões de novas iniciativas integrando dados para melhorar o despacho das ambulâncias do LAS. As sugestões foram: (i) integração de dados de saúde e assistência social com o LAS; (ii) dados de transporte; (iii) dados sobre a qualidade do ar; (iv) dados que indicam a localização da população em tempo real; (v) dados da tecnologia de comunicação por vídeo; (vi) dados de previsões da meteorologia e de condições climáticas.

Dados de saúde e assistência social

A equipa do projeto DASH estudou a possibilidade de integrar dados de saúde e de assistência social, de forma a que estes influenciassem de forma positiva o despacho de veículos do LAS.

O LAS não recolhia informações sobre o que acontece antes e depois do envolvimento dos meios de emergências médicas. A equipa do projeto DASH concluiu que a integração dessas informações podia ser interessante para avaliar a abordagem do LAS a ocorrências anteriores, e assim melhorar a eficácia na tomada de decisão do despacho de ambulâncias. Com este propósito surgiu o projeto de Partilha de Dados do Departamento de Emergência Pré-Hospitalar (PHED) [5]. Este projeto procurou relacionar dados das ambulâncias com dados do hospital, para perceber o que acontece aos pacientes depois de entrarem nas urgências do hospital. A equipa do projeto DASH sugeriu também estender a abordagem do projeto PHED a ocorrências não emergenciais, ou seja, ocorrências em que os pacientes não foram encaminhados para o hospital.

Uma outra sugestão do projeto DASH está relacionada com o acesso da equipa da ambulância a registos de saúde do paciente. Assim, poderia haver melhorias na qualidade do atendimento e nas decisões da equipa no local da ocorrência, como um melhor diagnóstico e o encaminhamento dos pacientes para um local adequado ao seu caso, ao invés de serem encaminhados para os departamentos de urgências dos hospitais. Um exemplo da utilização de registos de saúde, foi o projeto SAFER 2 [21] que procurou desenvolver um novo protocolo clínico que permitisse aos paramédicos avaliar as pessoas idosas após uma queda e, se apropriado, encaminhá-las para os serviços comunitários de quedas. Para além de pacientes idosos, outros segmentos como: pacientes com doenças mentais, pacientes com problemas respiratórios, pacientes com problemas de abuso de álcool e drogas, entre outros podem ser alvo de melhores decisões por parte das equipas de emergências médicas.

Um outro estudo realizado em Inglaterra [24], no Serviço de Ambulâncias do Sudoeste (SWASFT)[1], mostra também que um aumento na quantidade de informação disponível sobre um paciente pode levar as equipas de emergência médica a ter melhorias no processo decisão e, consequentemente, um atendimento mais apropriado.

No projeto Smart Emergency Medical Service (SEMS) [13] procurou aplicar-se a tecnologia de *IoT* (Internet das Coisas) para otimizar os processos dos serviços de emergência médica. Neste projeto, dados das ocorrências foram também integrados com registos históricos médicos dos pacientes, para apoiar as equipas responsáveis pela assistência do paciente.

O projeto DASH mostrou que também poderia ser interessante para as equipas de emergência integrar informações de caráter social dos pacientes, em particular, informar: o local de residência dos pacientes, a possível falta de aquecimento ou de condições na habitação, necessidades de cuidados sociais, o reconhecimento de adultos vulneráveis, entre outras informações, que possam ser relevantes. Por exemplo, considerando que a equipa tinha conhecimento que um idoso residia num lar da terceira idade, seria considerada uma situação de maior urgência, uma vez que num lar os idosos encontram-se teoricamente em melhores condições do que em casa.

[1] https://www.swast.nhs.uk

Dados de transporte

Esta iniciativa tinha como objetivo fazer uma parceria com os Transportes de Londres para mobilidade inteligente no trânsito dos veículos de emergências médicas.

Nos últimos anos, a população londrina bem como uma grande parte da população mundial acostumou-se a usar aplicações de transporte inteligente como Google Maps[2] ou Waze[3] para planear as suas viagens diariamente. As tecnologias de transportes e mobilidade inteligente usam dados de trânsito fornecidos em tempo real por meio de *crowdsourcing*, através de dispositivos equipados com GPS como smartphones ou veículos.

O projeto DASH considerou que, entre as várias oportunidades que a mobilidade inteligente poderia trazer, ter um mecanismo capaz de recomendar o veículo que demoraria menos tempo a chegar ao local da ocorrência seria a oportunidade mais promissora para melhorar a tomada de decisão no processo de despacho. Atualmente, em Londres, estimativas do custo da rota ou informações históricas de velocidades de resposta são usadas para estimar os tempos de resposta do veículo, o que leva a uma incerteza sobre se o veículo chegará dentro do tempo estimado. Uma maior certeza de que os meios chegariam no tempo estimado tornaria o despacho mais eficiente e reduziria atrasos imprevistos, que podem afetar a assistência dos pacientes. O mecanismo capaz de recomendar o veículo, que chegaria em menos tempo ao local da ocorrência permite também melhorar os modelos de relocalização de recursos, que consideram o posicionamento ideal de veículos.

Num outro estudo realizado sobre o LAS [16] desenvolveu-se uma estrutura de simulação e foi introduzido um novo método para definir rotas e tratar o despacho. O novo método usa estimativas para definir rotas mais precisas e sensíveis às variações do trânsito em tempo real. Concluiu-se que utilizando o novo método era possível diminuir o tempo de resposta na assistência a pacientes.

Outra abordagem que pode ser seguida usando os dados de mobilidade inteligente, está relacionada com a localização dinâmica de ambulâncias ao longo do dia. Um estudo com este propósito [20] concluiu que levando em consideração velocidades e os tempos de viagem em tempo real, o reposicionamento de veículos seria uma boa solução para conseguir manter um tempo de resposta reduzido. A mesma conclusão foi obtida num estudo realizado na Alemanha [17].

Dados sobre a qualidade do ar

Nesta iniciativa pretendia-se uma interação do LAS com a *London Air Quality Network* para previsões de utilização de recursos.

Como é sabido, a poluição do ar tem um impacto significativo na saúde da população, especialmente nas zonas urbanas onde o elevado tráfego automóvel é uma realidade. As pessoas com problemas respiratórios, como asma ou com um Distúrbio Pulmonar Obstrutivo Crónico (DPOC), são particularmente vulneráveis à poluição do ar e, quando sentem dificuldade em respirar, por vezes têm de pedir socorro. Como os problemas respiratórios representam uma grande parte das ocorrências (doenças respiratórias como a asma são comuns), incluir melhorias no despacho para este tipo de ocorrências deve ser

[2] https://www.google.pt/maps/preview
[3] https://www.waze.com

uma prioridade.

Na cidade de Londres são extraídos dados relevantes sobre a qualidade do ar através de sensores. Integrar os dados extraídos para conseguir melhorias no processo de despacho é algo bastante desafiador, pois existe uma enorme variedade de tipos de poluição e a qualidade do ar pode variar até com as condições atmosféricas. O foco principal desta iniciativa foram os casos extremos, onde existe uma especial atenção no planeamento de recursos.

O projeto DASH sugere duas abordagens a serem consideradas pelo LAS: (i) desenvolver uma aplicação de suporte ao paciente para pacientes com asma, com base em previsões de qualidade do ar, informações de localização do paciente, e que seja adequadamente projetada e avaliada para maximizar a melhoria dos resultados; e (ii) comunicações públicas sobre planos para minimizar a poluição da frota de veículos do LAS.

Para além da poluição, a variação das partículas transportadas pelo ar também tem influência na quantidade de ocorrências, isso veio a ser confirmado num estudo realizado em Itália [18].

Dados que indicam a localização da população em tempo real

Esta iniciativa passava por usar dados e informações dos operadores de redes móveis para localizar a população em tempo real.

Em Londres utilizam-se dados históricos para prever a localização de ajuntamentos da população. Esta abordagem não é a mais correta, pois seria mais correto saber a localização da população em tempo real, o que normalmente não é possível. O uso de dados da localização em tempo real permitiria uma cobertura de ambulâncias ajustada à mobilidade da população, através de métodos de análise espacial e *data mining*.

Apesar da utilização de dados em tempo real ser a maneira mais correta para abordar a cobertura de ambulâncias, os padrões associados a dados históricos são por norma precisos e suficientes para organizar os recursos. Os benefícios a curto prazo do uso de dados em tempo real seriam limitados a eventos imprevistos no qual os ritmos típicos de mobilidade espácio-temporal de Londres são interrompidos por períodos relativamente longos, como por exemplo: interrupções não planeadas de transportes ou desastres naturais.

Num estudo realizado sobre os dados das redes móveis [4] concluiu-se que as operadoras de redes móveis obtêm dados de mobilidade da população em tempo real com um grande nível de detalhe. Neste momento, as operadoras utilizam esses dados para fins comerciais e para marketing baseado na localização.

Num trabalho de 2014 [3] estudou-se o potencial uso dos dados da operadora Telefonica[4]. Embora o tema abordado tenha sido na área da criminalidade e não na área da saúde, os autores observaram que os dados de localização da população das redes móveis fornecem uma previsão espácio-temporal significativamente mais precisa do que dados históricos. No artigo mencionou-se uma divisão da cidade em células. Para cada hora, o conjunto de dados contém uma estimativa de quantas pessoas estão

[4]https://www.telefonica.com/es/home

na célula, a percentagem das que estão em casa, no trabalho ou apenas na célula de passagem, bem como o sexo e idade se estivessem corretamente associados ao operador.

Um estudo realizado em Israel [14] apresentou um modelo de simulação utilizando um Sistema de Informação Geográfica. O modelo de simulação apresentado no estudo mostra que os serviços de emergência médica poderiam ser mais eficazes, se fosse aplicada uma implementação dinâmica de ambulâncias de forma a dar resposta à carga, isso resultaria em maiores casos de assistência com sucesso.

Para se utilizar este tipo de dados, o público deve ser informado sobre o uso pretendido, incluindo riscos e oportunidades, e é natural que haja alguma oposição. Apesar dos riscos, esta abordagem pode otimizar o despacho de veículos de emergência médica, logo há motivos para prosseguir com perspetivas de benefício da saúde pública.

Dados da tecnologia de comunicação por vídeo

Esta iniciativa centra-se na utilização de dados da tecnologia de comunicação por vídeo.

Nos dias de hoje, uma grande parte da população está familiarizada com aplicações com funcionalidades de comunicação por vídeo, que podem ser utilizadas em smartphones, como o Skype[5]. O projeto DASH procurou benefícios do uso deste tipo de aplicações para otimizar o despacho de ambulâncias.

Existem casos em que é benéfico para o paciente recorrer a consultas por videochamada, pois são poupados da viagem, dos tempos de espera e do ambiente hospitalar. Todavia, é necessário perceber se é clinicamente seguro para o paciente e se não existem problemas técnicos que ponham em causa este tipo de comunicação. Para o lado do LAS a realização de consultas de forma remota traria uma diminuição da quantidade de ambulâncias enviadas e consequentemente mais recursos disponíveis.

O projeto DASH sugere a utilização deste tipo de comunicação ao LAS, que deve ser implementado de uma forma lenta e incremental para o LAS se ir familiarizando e ajustando. Ao implementar este tipo de comunicação seria sempre necessário a autorização do paciente.

Dados de previsões da meteorologia e de condições climáticas

Esta iniciativa procura utilizar dados de previsões da meteorologia e de condições climáticas, de forma a terem impacto no despacho do LAS.

Em Londres já eram utlizados dados climáticos para planear a estratégia do LAS, principalmente para preparar o Inverno, a altura em que o clima fica mais frio e há uma maior tendência para o aumento de ocorrências. A preparação é efetuada sobretudo por planeamento de pessoal e recursos.

O projeto DASH sugeriu ao LAS procurar tendências associadas às diferentes condições climatéricas. Essas tendências foram confirmadas em 2014 com referência a dados de Birmingham [22] e em 2017 usando dados de Londres [11]. Confirmando que, do lado do paciente, é natural que ocorram mais fraturas no clima frio e tonturas ou desmaios em temperaturas mais quentes. Do lado do despacho, no Inverno é natural encontrar condições difíceis na estrada (por exemplo, estradas com gelo ou

[5]https://www.skype.com

baixa visibilidade) e doenças do pessoal que podem deixar os serviços de emergências médicas com mão de obra insuficiente.

Outros artigos referem pesquisas para relacionar dados do calendário (estação do ano, semana do mês, dia da semana, feriados, etc.) com dados relativos à meteorologia (altas e baixas temperaturas, chuvas, queda de neve, etc.), isto para prever o volume de chamadas de emergência médica. Como é o caso de um estudo realizado nos Estados Unidos da América [9], no estado de Kentucky ou um outro realizado na Coreia do Sul [10], que nos mostram evidências sobre esta relação.

O projeto DASH concluiu que apesar de haver correlações entre as condições climatéricas e o despacho de ambulâncias, as previsões meteorológicas não devem ser muito relevantes para o planeamento da distribuição de recursos, dado a não serem consideradas totalmente fiáveis.

3.1.2 Estudo de simulação para melhorar o desempenho dos serviços de emergência médica

Um estudo de simulação para melhorar o desempenho dos processos dos Serviços de Emergências Médicas em França (SAMU) decorreu no departamento de Val-de-Marnede (SAMU 94) [1].

O modelo proposto foi um modelo de Simulação de Eventos Discretos (DES) [2]. Neste modelo foram analisadas simulações com possíveis alterações nos processos do SAMU 94, que poderiam levar a melhorias na eficiência operacional. Estas simulações envolviam cinco estratégias, descritas mais à frente, em que o nível dos recursos ia sendo alterado bem como a posição das equipas de emergência médica em toda a área de serviço. Nestas simulações foram consideradas duas medidas base para avaliar o desempenho em cada tipo de estratégia: (i) Cobertura, que é definida como a percentagem de chamadas para as quais o tempo de resposta não excede um dado período de tempo, (ii) a Taxa de Utilização dos Recursos Humanos, que é definida como a carga de trabalho total dividida pelo tempo total de operação. Estas duas medidas são muito importantes pois estão relacionadas com o objetivo do SAMU, que é salvar vidas com recursos limitados.

Os recursos humanos envolvidos nas operações são: (i) Operadores, responsáveis por atender chamadas, identificar chamadas inadequadas e criar um arquivo médico; (ii) Reguladores do SAMU, que são médicos especializados em emergências e estão responsáveis por realizar uma avaliação médica das chamadas e determinar a melhor solução para pacientes de alta prioridade (prioridade 1); (iii) Reguladores do PDS (“PDS” é a sigla em francês para cuidados permanentes) que são médicos de clínica geral, e são responsáveis por realizar uma avaliação médica das chamadas mas para os restantes pacientes (prioridade 2). As equipas SMUR (“SMUR” é a sigla francesa para Serviço Móvel de Emergência e Reanimação) são compostas por um médico, um motorista, uma enfermeira e/ou um técnico médico de emergência, incluindo também um veículo.

Os dados usados para construir o modelo de DES foram extraídos das bases de dados do sistema SAMU 94 num período de 15 meses. Os dados foram posteriormente analisados com o objetivo de extrair informação relacionada com o despacho de veículos, a quantidade de recursos alocados e os tempos de processamento e deslocação.

O processo e dados recolhidos pelo SAMU 94 associados a cada tipo de ocorrência foram descritos e representados num modelo conceptual, que está visível em anexo na Figura A.1. O Modelo Conceptual do SAMU 94 foi utilizado para comparar o desempenho das várias estratégias, sendo que cada estratégia considerou vários cenários. De seguida é apresentada a descrição das estratégias e cenários mais relevantes para o SAMU94.

Estratégia A: Variando o número e a carga de trabalho dos recursos

Os principais recursos necessários para atender a pedidos de emergências médicas são profissionais médicos qualificados. Assim sendo, a tarefa de dimensionar a equipa de assistência é crucial para garantir um atendimento pré-hospitalar de alta qualidade, em termos de cobertura eficiente e redução de custos.

Na estratégia A, para avaliar o desempenho do SAMU 94 em relação a mudanças no nível de recursos, foram testados alguns cenários com alterações no nível de recursos em períodos críticos. Períodos críticos são caracterizados por um alto volume de chamadas e despachos de ambulâncias ou por períodos com uma quantidade de recursos disponíveis reduzida.

Nos cenários testados houve variações no número de operadores, no número de equipas SMUR, de reguladores PDS e reguladores SAMU. Houve também um cenário diferente onde foram enviadas equipas SMUR sem um médico para tratar ocorrências com menor prioridade.

Os resultados obtidos para cada cenário tiveram três medidas de desempenho: (i) a taxa de utilização de recursos; sendo que a medida de cobertura foi dividida da seguinte forma: (ii) a cobertura em 20 minutos para chamadas de prioridade 1; (iii) a cobertura em 20 minutos para chamadas de prioridade 2.

Os resultados dos cenários que variam o número de operadores, de reguladores PDS e reguladores SAMU bem como o cenário que remove médicos para ocorrências de baixa prioridade mostram que não existem alterações significativas nas medidas de cobertura, apenas a taxa de utilização de recursos se altera significativamente. Os cenários em que aumentaram o número de operadores e reguladores SAMU mostraram um sistema saturado, na medida em que não há melhorias significativas na cobertura.

Não existem impactos associados à eficiência do sistema nos cenários em que a quantidade de recursos diminui, o que mostra que se pode diminuir o número de operadores e reguladores PDS sem que aumente o tempo de espera da chamada.

Quando houve variações do número de equipas SMUR, o desempenho na medida de cobertura foi afetado significativamente, o que prova a importância destas equipas.

Estratégia B: Melhorias na implementação das equipas SMUR

Tanto o número de equipas SMUR como a localização dessas equipas são fatores importantes na eficiência do atendimento pré-hospitalar. Na estratégia B foram testadas várias localizações de equipas SMUR para obter melhorias no desempenho geral da cobertura. Nesta estratégia foi observado o efeito de relocalizar as equipas SMUR que inicialmente estavam localizadas na base central, para outras três

bases localizadas ao longo do departamento de Val-de-Marne.

Foi avaliado o impacto de relocalizar gradualmente de uma até três equipas SMUR para as outras bases, no horário das 8 às 20 horas. De salientar que no estudo foi tido em atenção a distinção entre dias úteis e fins de semana.

Os resultados obtidos mostraram que a relocalização de equipas SMUR em bases distribuídas pelo departamento de Val-de-Marne, traria melhorias significativas na cobertura.

Estratégica C: Resposta regionalizada

Quando existe uma ocorrência, o sistema do SAMU 94 tem como regra enviar a equipa SMUR disponível mais próxima do local da ocorrência. Na estratégia C foi testada uma regra de despacho alternativa, chamada Resposta Regionalizada, que consistia em atribuir a cada equipa SMUR uma dada área. Se essa equipa estivesse ocupada, a equipa disponível mais próxima ficaria responsável pela operação. A principal vantagem desta estratégia consiste em minimizar os tempos de viagem devido ao tamanho limitado da área geográfica que as equipas SMUR percorrem entre os locais das chamadas.

A Resposta Regionalizada foi aplicada ao cenário tradicional do SAMU 94 e aos cenários de relocalização obtidos na Estratégia B (Secção 3.1.2). A conclusão obtida é que apenas para o cenário em que são relocalizadas três equipas SMUR se verificou um impacto positivo no desempenho da cobertura.

Estratégia D: Relocalização multiperíodo

Como é lógico o volume de chamadas de emergência, bem como a sua localização muda dinamicamente com base nas atividades da população durante o dia. A estratégia D sugeria que a atribuição de equipas SMUR às bases deveria ser ajustada, de forma a cobrir adequadamente as áreas onde há maior quantidade de pessoas.

Os resultados obtidos mostraram que as melhorias mais significativas associadas a esta estratégia estão relacionadas com a cobertura de ocorrências com prioridade 2.

Estratégia E: Melhoria de processos

Na estratégia E foram testadas novas tecnologias para realizar a triagem de chamadas e a criação de arquivos médicos.

Os resultados obtidos mostraram que a estratégia E permitiu reduzir o tempo de despacho de ambulâncias e consequentemente trouxe melhorias na cobertura.

3.1.3 Plataforma para Apoio à Tomada de Decisão em Situações de Emergência

Num projeto que decorreu em Espanha [15] foi desenvolvida uma plataforma web analítica que apresenta vários resultados estatísticos de ocorrências de emergências, permitindo aos utilizadores obter várias informações sobre as ocorrências. Esta plataforma foi desenvolvida para as Ilhas Canárias. A aplicação incorpora dados da ocorrência em si, como a sua localização geográfica ou temporal, com

dados de fontes externos, como dados sociais e económicos, para permitir aos utilizadores estudar relações entre fatores externos e as ocorrências.

Na plataforma foram utilizados mais de 7 milhões de registos de ocorrências que foram extraídos na última década. O Centro de Coordenação de Emergência e Segurança (CECOES) é o serviço público responsável pela gestão de todos as ocorrências registadas nas Ilhas Canárias.

A utilização desta aplicação permitiu melhorias na análise de dados, o que ajudou o CECOES a melhorar o seu desempenho nos processos de tomada de decisão. A plataforma permite a integração futura de novos dados com o objetivo de fornecer ainda mais informações, que possam estar relacionadas com as ocorrências.

Seleção de Dados e Processo ETL

Como já foi referido, este estudo foi realizado usando dados históricos de ocorrências tratadas pelo CECOES, referentes aos anos de 2010 a 2014. Os dados mais relevantes extraídos das ocorrências são: (i) Informações de data e hora, de início e fim da ocorrência, (ii) Idade e género das pessoas envolvidas; (iii) Localização; (iv) Tipo de ocorrência (acidente de viação, incêndio, etc); (v) Recursos alocados para resposta (polícia, bombeiros, etc.); (vi) Avaliação da gravidade da ocorrência (Baixa, Média, Alta, etc.).

Para obter uma plataforma com os dados das ocorrências, foi necessário realizar um processo ETL, para extrair, transformar e, por fim, carregar os dados para a plataforma. A processo de integração periódica de novos dados também foi desenvolvido.

Os dados em bruto apresentavam alguns problemas, por exemplo, um grande número de categorias nos vários parâmetros ou formas diferentes de se referir à mesma localização. Para resolver este tipo de problemas, no processo ETL foi necessário dar ênfase ao processo de transformação, para garantir a qualidade dos dados.

Depois do carregamento dos dados para a plataforma, foi possível obter um conjunto de dados pronto para análise. A etapa seguinte do projeto consistiu no desenvolvimento da aplicação para análise de dados, que permitirá aos utilizadores da plataforma analisar as ocorrências e melhorar o desempenho do CECOES na tomada de decisão.

Desenvolvimento da Plataforma Web

A linguagem de programação utilizada para desenvolver o design da plataforma foi R [6], com o apoio de várias bibliotecas. A utilização destas tecnologias permitiu aos utilizadores uma interação intuitiva, a Figura 3.1 mostra o ecrã inicial da plataforma. A aplicação permite que os utilizadores visualizem dados geográficos das ocorrências num painel apelativo e filtrem informações.

[6] https://www.r-project.org/

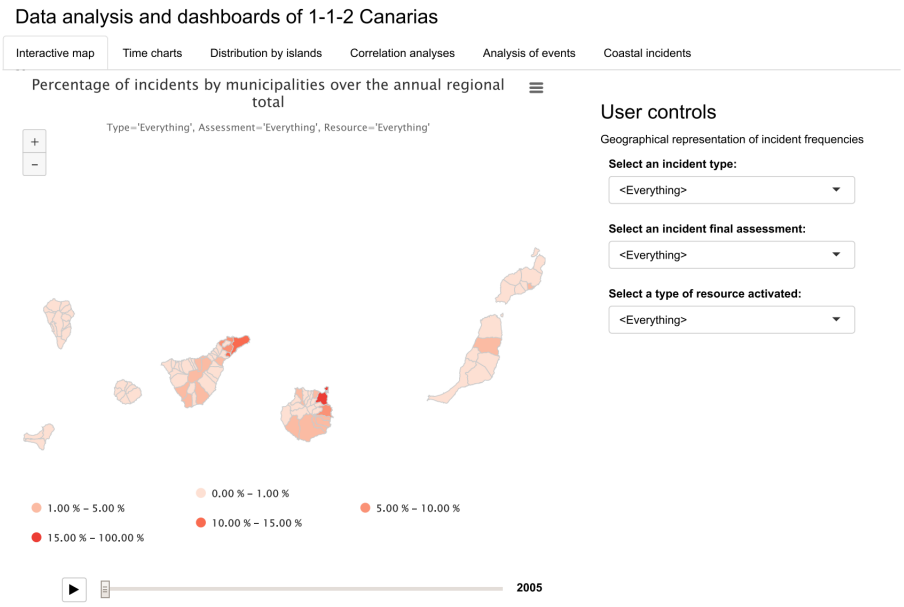

Figura 3.1: Ecrã inicial da aplicação web, mostrando a distribuição geográfica das ocorrências, e os possíveis filtros a aplicar para interagir com a aplicação [15]

3.1.4 Discussão

O projeto DASH (Secção 3.1.1) tem um objetivo semelhante ao do projeto Data2Help, otimizar os processos dos serviços de emergências médicas. Em ambos os projetos serão integrados novos dados para atingir esse mesmo objetivo. Para o projeto Data2Help podem ser interessantes algumas iniciativas patentes no projeto DASH. A integração de dados de saúde, assistência social, localização da população, qualidade do ar ou dados meteorológicos no Data Warehouse do projeto Data2Help pode possibilitar a deteção de correlações entre o número de ocorrências e esses tipos de dados.

O estudo de simulação (Secção 3.1.2) realizado em França também procura melhorar o desempenho dos serviços de emergências médicas, mas utiliza estratégias ligeiramente diferentes do projeto Data2Help. Apesar disso, algumas ideias podem ser transpostas para o projeto Data2Help. Por exemplo, poderão ser detetadas correlações entre a localização das equipas de emergência, os recursos disponíveis, a hora das ocorrências ou o tempo de chegada da equipa. A deteção destas correlações poderá trazer melhorias no planeamento e escalonamento dos recursos, tal como no estudo de simulação onde foram propostas novas localizações das ambulâncias ou um escalonamento diferente dos recursos.

O projeto de desenvolvimento de uma plataforma de análise de dados para apoio à tomada de decisão em situações de emergência (Secção 3.1.3) também pretende realizar uma integração de dados das ocorrências com dados de fontes externas, tal como o projeto Data2Help. As fontes de dados externas utilizadas na plataforma são muito semelhantes às que se pretendem utilizar no Data Warehouse do projeto Data2Help. Informações sobre o processo ETL executado no projeto da plataforma também poderão ser interessantes, devido ao facto das informações presentes nos dados de ambos os projetos estarem relacionadas.

3.2 Software para Integração de Dados

Nesta secção serão analisadas e comparadas várias ferramentas de software para integração de dados (Secção 3.2.1). As ferramentas serão analisadas com o objetivo de perceber qual a apropriada para realizar o Data Warehouse no projeto Data2Help (Secção 3.2.2).

3.2.1 Ferramentas de Software

A escolha das ferramentas a analisar foi baseada no *Quadrante Mágico da Gartner para Ferramentas de Integração de Dados* [8], representado na Figura 3.2, um relatório que analisa e avalia várias ferramentas bem como as empresas que as produzem em termos de capacidade de execução e abrangência da visão. Para as comparações realizadas foi utilizado o relatório de 2019.

Figura 3.2: Quadrante Mágico da Gartner para Ferramentas de Integração de Dados [8]

No Quadrante Mágico as ferramentas são dividas em quatro quadrantes: (i) os Líderes, que é o quadrante ambicionado por todos, mas apenas estão presentes os softwares que demonstram ter compreensão sólida das necessidades do mercado bem como uma completa capacidade de desenvolvimento e abrangência da visão; (ii) os Desafiantes, são caracterizados pela capacidade de executar as suas estratégias, contudo, ainda não têm um planeamento capaz de forma a garantirem uma direção bem definida no mercado; (iii) os Visionários, são caracterizados por apresentarem uma boa visão de desenvolvimento do mercado, mas ainda não executam corretamente essa abordagem; (iv) os Competidores de Nicho, são caracterizados pelos resultados excelentes num segmento específico, mas são limitados para atuar noutros segmentos.

Nas subsecções seguintes iremos focar-nos essencialmente nas ferramentas presentes no quadrante dos Líderes, sendo que também abordaremos o Pentaho e o SQL Server, pertencentes respetivamente ao quadrante de Competidores de Nicho e Desafiantes.

Informatica

A ferramenta *Informatica PowerCenter* é atualmente líder do mercado, oferece capacidades de integração de dados através de processos ETL e é usada essencialmente para construir Data Warehouses e Data Marts para empresas. Está disponível quer para sistemas operativos Windows quer para sistemas operativos baseados em *UNIX*.

A utilização do produto permite extrair, tansformar e carregar grandes volumes de dados. Tem a capacidade de se conectar com um grande conjunto de tipos diferentes de fontes de dados, por exemplo: *MySQL*, *SQL Server* ou Oracle. A interação com o produto é feita através de uma interface gráfica que permite configurar e realizar todo o processo ETL de uma forma intuitiva, através de uma tecnologia "drag and drop". O software tem também algumas funcionalidades para análise de dados.

IBM InfoSphere Information Server

A *IBM InfoSphere Information Server* é uma plataforma para integração de dados que oferece vasta gama de ferramentas para trabalhar dados e é essencialmente voltada para a área Business Intelligence. Está disponível para sistemas operativos *Windows*, *Linux* e *AIX*.

A ferramenta utilizada para a integração de dados é a *IBM InfoSphere DataStage*, suporta grandes volumes de dados e permite projetar as etapas do processo ETL através de uma interface gráfica, por meio de uma tecnologia "drag and drop". A IBM InfoSphere DataStage consegue também ela conectar-se com um vasto conjunto de diferentes tipos fontes de dados, como fontes de dados *Oracle* ou *Sybase*.

Oracle Data Integrator

O *Oracle Data Integrator* é um software para integração de dados que permite desenvolver Data Warehouses. O Oracle Data Integrator usa uma abordagem ELT que difere da clássica abordagem ETL na medida em que os dados depois de extraídos são imediatamente carregados no Data Warehouse, só depois ocorre a etapa de transformação de dados. Esta abordagem permite um melhor desempenho quando estamos perante um grande volume de dados. Está disponível para sistemas *Windows* e *Linux*.

A ferramenta permite configurar todo o processo ELT através de uma interface gráfica, utiliza uma tecnologia "drag and drop" para configurar todas as etapas. O Oracle Data Integrator consegue conectar-se a vários tipos de diferentes de fontes de dados e pode associar-se a outras ferramentas da Oracle para permitir diversas interações com os dados.

Pentaho Data Integration

O Pentaho consiste num conjunto de ferramentas *Open Source* que se podem interligar de forma a realizar integração, análise e apresentação de dados. A ferramenta utilizada para a integração de dados é o *Pentaho Data Integration (PDI)*, este permite criar Data Warehouses através de processos ETL. O PDI está disponível para vários sistemas operativos: *Windows*, *Linux* ou *MacOS*.

O software apresenta uma interface gráfica intuitiva, e utiliza a tecnologia de “drag and drop” para projetar os processos ETL. O PDI tem a capacidade de se conectar a diferentes tipos de fontes de dados.

Microsoft SQL Server

O *Microsoft SQL Server*, é composto por conjunto de softwares que permitem integração, análise e apresentação de dados. O *SQL Server Integration Services (SSIS)* é a ferramenta utilizada para a realizar a integração de dados, sendo que esta está apenas disponível para o sistema operativo *Windows*.

O SSIS permite criar Data Warehouses através do desenvolvimento de processos ETL por meio de uma interface gráfica, bem como complementar os processos ETL com código para definir de forma personalizada as suas tarefas. A ferramenta suporta a integração de vários tipos diferentes de fontes de dados.

3.2.2 Discussão

Para escolher a ferramenta a utilizar de forma a realizar a integração de dados, é necessário perceber quais as ferramentas de software que atendem às necessidades do projeto, e se possível as ferramentas deveriam ser *open source* e suportadas por vários sistemas operativos.

Todas ferramentas descritas anteriormente, têm a capacidade de desenvolver o processo ETL do projeto Data2Help e consequentemente carregar os dados transformados para o Data Warehouse. Mas, o Pentaho Data Integration é o software descrito anteriormente capaz de preencher todos os requisitos e preferências do Projeto Data2Help.

As outras ferramentas Pentaho poderão ser também interessantes numa fase posterior de forma a complementar a integração de dados com a sua análise e apresentação. Outro ponto importante e decisivo na escolha da ferramenta está relacionado com a familiarização já existente, assim a fase de implementação pode iniciar-se mais rapidamente.

Capítulo 4

Solução

Neste capítulo é descrita a solução de integração de dados. Na Secção 4.1 é apresentada a tarefa de Levantamento de Requisitos, onde se definem claramente as consultas às quais o repositório de dados integrado deve conseguir responder. A Secção 4.2 apresenta a arquitetura da solução definida após o Levantamento de Requisitos. A Secção 4.3 apresenta a base de dados intermédia na qual os dados sofrem transformações antes de serem carregados para o Data Warehouse. Na Secção 4.4 é detalhado o processo que deu origem ao modelo do Data Warehouse e são descritos todos os Data Marts presentes no Data Warehouse. Na Secção 4.5 é descrito o processo ETL realizado para obter o Data Warehouse.

4.1 Levantamento de Requisitos

O Levantamento de Requisitos ocorreu em colaboração direta com o professor Rui Henriques, que será responsável pela próxima tarefa do projeto (desenvolver algoritmos avançados de ciência de dados para encontrar correlações de dados e aplicar modelos preditivos), tendo como principal objetivo estabelecer um entendimento entre as partes interessadas relativamente aos objetivos da integração de dados.

Nesta etapa foram definidas as fontes de dados a utilizar na integração de dados. As diversas fontes de dados estão apresentadas na Secção 4.1.1. De seguida, com as fontes a utilizar definidas, foi identificado um conjunto de consultas ao qual deve ser possível responder com base nos dados integrados (Secção 4.1.2 e Secção 4.1.3). Definidas as consultas, concluiu-se que o melhor método a utilizar seria uma integração de dados materializada. Desta forma, os dados serão armazenados num Data Warehouse com a qualidade necessária para que se efetue a sua análise posteriormente.

4.1.1 Fontes de dados

Nesta secção estão apresentadas as várias fontes de dados utilizadas no projeto.

SIADEM

Antes dar início à especificação de requisitos foi necessário reunir com colaboradores da empresa Hexagon [1] (empresa responsável pela gestão da base de dados do Sistema Integrado de Atendimento e Despacho de Emergência Médica (SIADEM)). A reunião serviu para apresentar e prestar esclarecimentos sobre a base de dados relacional da SIADEM.

Foi fornecido pelo INEM um conjunto de dados relativos ao período compreendido entre o dia 9 de Maio de 2012 e o dia 26 de Fevereiro de 2020. O Sistema de Gestão de Base de Dados (SGBD) utilizado pelo SIADEM é o Microsoft SQL Server.

A base de dados do SIADEM contém diversos tipos de informações: dados sobre as ocorrências (chamada, descrição da ocorrência, tempos de processamento, localização ou meios alocados), dados sobre os meios, dados sobre as estações onde estão os meios, informações sobre hospitais, entre outros.

Dados Meteorológicos

O conjunto de dados meteorológicos foram fornecidos pelo Instituto Português do Mar e da Atmosfera [2]. Foram obtidos através de medições efetuadas por três sensores colocados em diferentes estações na cidade de Lisboa e são referentes ao período compreendido entre 12 de Dezembro de 2018 e 30 de Janeiro de 2019.

O conjunto de dados fornecido permite obter diversos tipos de informações relativos às medições (a hora da medição, medidas de temperatura, humidade, intensidade, direção do vento ou precipitação acumulada). Também é possível obter as coordenadas da localização das estações onde estão os sensores que efetuaram as medições.

Dados relativos a eventos de jogos de futebol

Os dados relativos a jogos de futebol foram extraídos da API-FOOTBALL [3]. Esta API (*Application Programming Interface*) permite extrair, em formato JSON [4], dados relativos a várias ligas e taças de futebol nacionais e internacionais. É de referir que a API permite extrair dados atualizados, um ponto que se enquadra com as necessidades do Data Warehouse do projeto Data2Help.

O conjunto de dados extraído da API refere-se à primeira liga portuguesa, no período compreendido entre a época 2010/2011 e a época 2019/2020.

Dados relativos a eventos musicais

Os dados relacionados com eventos musicais foram extraídos da API Songkick [5]. Esta API permite a extração de dados relativos a concertos e festivais, em formato JSON. Os concertos e festivais decor-

[1] https://hexagon.com
[2] https://www.ipma.pt/pt/
[3] https://rapidapi.com/api-sports/api/api-football
[4] https://www.json.org
[5] https://www.songkick.com/developer

ridos dos quais é pretendido extrair dados, necessitam de ser selecionados manualmente no website por um utilizador registado.

No conjunto de dados extraído estão presentes concertos e festivais realizados em Portugal com uma popularidade e quantidade de público considerável, no período compreendido entre Maio de 2012 e Fevereiro de 2020.

4.1.2 Consultas apenas com dados do SIADEM

Nesta secção está listada um conjunto de consultas, identificadas com o apoio do professor Rui Henriques, que deve ser possível efetuar sobre o repositório de dados integrado utilizando apenas dados fornecidos pelo SIADEM.

1. **Ocorrências/Tempo**
   (a) Número de ocorrências numa data
   (b) Número de ocorrências num intervalo de tempo
   (c) Número de ocorrências por ano
   (d) Número de ocorrências por ano / mês
   (e) Número de ocorrências por dia da semana
2. **Ocorrências/Localização**
   (a) Número de ocorrências por Distrito
   (b) Número de ocorrências por Concelho
   (c) Número de ocorrências por Freguesia
3. **Ocorrências/Prioridade**
   (a) Número de ocorrências por Prioridade
4. **Ocorrências/Tipo de Ocorrência**
   (a) Número de ocorrências por Tipo de Ocorrência
5. **Ocorrências/Destino de Ocorrência**
   (a) Número de ocorrências por Destino de Ocorrência
6. **Tempos máximos/médios**
   (a) Tempo médio até acionamento
   (b) Tempo máximo até acionamento
   (c) Tempo médio até à chegada do meio
   (d) Tempo máximo até à chegada do meio

(e) Tempo médio até à saída do meio

(f) Tempo máximo até à saída do meio

(g) Tempo médio de chegada do meio ao destino

(h) Tempo máximo de chegada do meio ao destino

4.1.3 Consultas com integração de dados externos

Nesta secção serão listadas consultas que devem ser possíveis de executar utilizando dados do SIADEM e dados de fontes externas.

1. **Ocorrências** / **Jogos de Futebol**

   (a) Variação do número de ocorrências ao longo do tempo no local em que se realiza o jogo

   (b) Tipo de ocorrências em eventos de futebol

2. **Ocorrências** / **Concertos**

   (a) Variação do número de ocorrências ao longo do tempo no local em que se realiza o concerto

   (b) Tipo de ocorrências em concertos

3. **Ocorrências** / **Festivais**

   (a) Variação do número de ocorrências ao longo do tempo no local em que se realiza o festival

   (b) Tipo de ocorrências em festivais

4. **Ocorrências** / **Meteorologia**

   (a) Variação do número de ocorrências de acordo com a variação das condições meteorológicas numa determinada área

   (b) Tipo de ocorrências associadas a diferentes condições meteorológicas

4.2 Arquitetura da Solução

A arquitetura da solução está apresentada na figura 4.1. Como referido na secção anterior, será utilizada uma integração de dados materializada, ou seja, os dados serão armazenados num Data Warehouse. A solução irá iniciar-se com um processo ETL que irá extrair, transformar e carregar dados das várias fontes para a Data Staging Area (apresentada detalhadamente na Secção 4.3). De seguida, os dados voltam a passar por um processo ETL (apresentado detalhadamente na Secção 4.5) mas, desta vez, são extraídos da Data Staging Area, com o objetivo de serem carregados no Data Warehouse. É importante referir que, antes de ser executado o segundo processo ETL, é necessário que seja realizada a modelação multidimensional do Data Warehouse (apresentada detalhadamente na Secção 4.4).

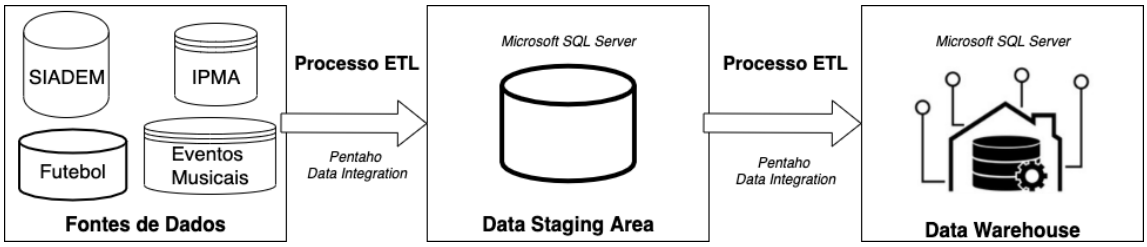

Figura 4.1: Arquitetura da Solução

4.3 *Data Staging Area*

Os dados poderiam ser extraídos, transformados e carregados diretamente para o Data Warehouse. No entanto, tendo em conta a qualidade e as diversas fontes de dados existentes no projeto Data2Help, definiu-se que os dados iriam ser guardados numa Data Staging Area. A Data Staging Area é uma base de dados intermédia e materializada na qual os dados sofrem algumas transformações e validações antes de serem carregados para o Data Warehouse. Os dados externos foram extraídos e transformados com o auxílio do Pentaho Data Integration e o SGBD utilizado foi o Microsoft SQL Server. As transformações que deram origem à extração e ao carregamento desses dados na Data Staging Area estão apresentadas em anexo ao documento, na Secção C.

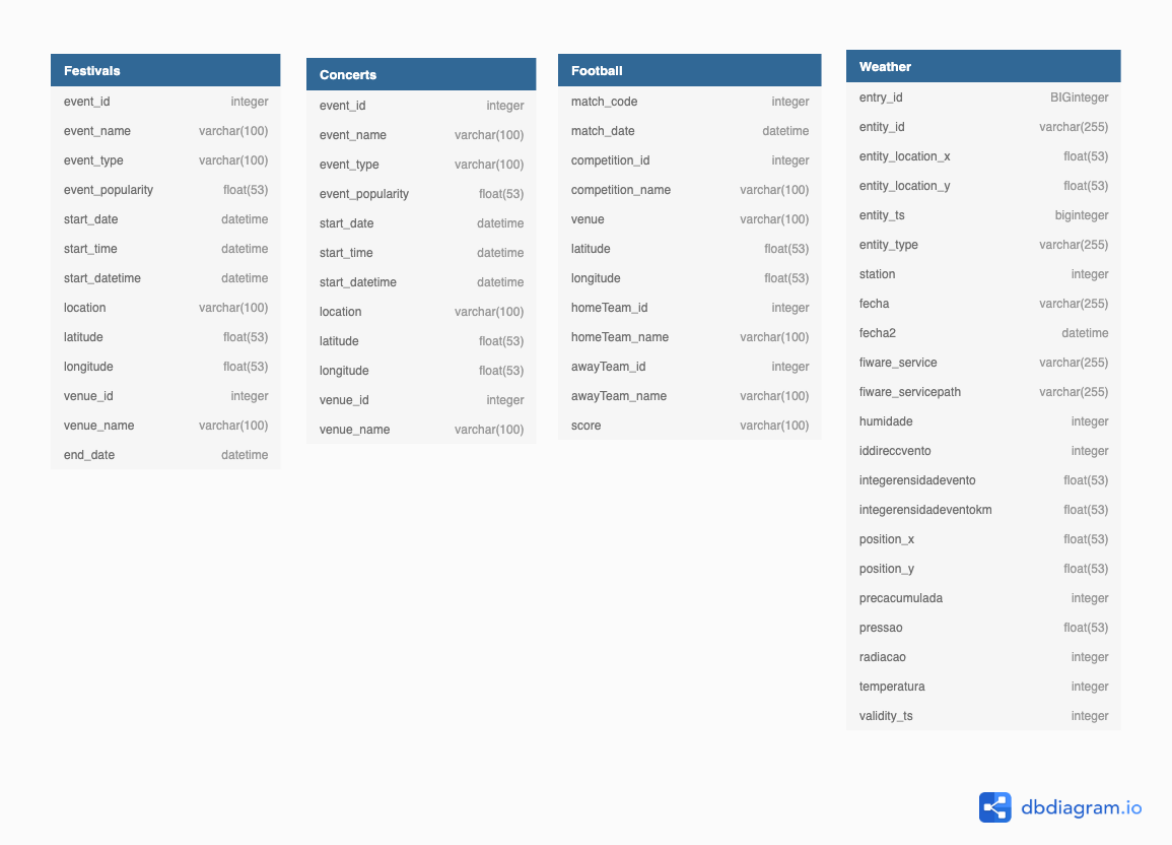

Figura 4.2: Data Staging Area Data2Help

A Data Staging Area é formada pelo conjunto de dados fornecido pelo SIADEM e pelos dados extraídos de fontes externas. Na Figura 4.2 estão representadas as tabelas da Data Staging Area que contêm os dados externos. Nesta base de dados intermédia foi criada uma tabela para cada tipo de dados externos: (i) uma tabela com dados de eventos de futebol; (ii) uma tabela com dados de concertos; (iii) uma tabela com dados de festivais; (iv) uma tabela com dados meteorológicos.

4.4 Modelo Multidimensional Data Warehouse

Com base nas consultas da Secção 4.1.2 e da Secção 4.1.3 foi desenvolvida uma *bus matrix*, representada na Figura 4.3, que é uma ferramenta utilizada com o objetivo de tornar a modelação do Data Warehouse num processo incremental. A matriz consiste num quadro em que os processos de negócio estão representados nas linhas.

Dimensões	Tempo	Localização	Tipo de Ocorrência	Unidade	Destino	Futebol	Competição	Equipas	Concerto	Festivais	Meteorologia
Processos de Negócio											
Todas as Ocorrências	X	X	X								
Ocorrências c/ Meios	X	X	X	X	X						
Ocorrências c/ informação completa	X	X	X	X	X						
Ocorrências / Futebol	X	X	X			X	X	X			
Ocorrências / Concertos	X	X	X						X		
Ocorrências / Festivais	X	X	X							X	
Ocorrências / Meteorologia	X	X	X								X

Figura 4.3: Matriz em Bus Data2Help

No enquadramento do projeto Data2Help, considerámos que os processos de negócio são ocorrências (factos), e que podemos inserir as ocorrências em vários subconjuntos consoante o tipo de informação que pretendemos extrair do Data Warehouse. As colunas representam as diferentes dimensões do Data Warehouse. A *bus matrix* foi desenvolvida incrementalmente, de processo de negócio em processo de negócio, o que facilitou a associação dos processos de negócio às dimensões. Também é importante destacar que as dimensões têm uma representação uniforme para todo o Data Warehouse, ou seja, são partilhadas pelas diferentes tabelas de factos. As células da matriz assinaladas com um "X" refletem uma relação lógica entre os processos de negócio e as dimensões.

Cada processo de negócio deu origem a um diferente Data Mart. Assim, a matriz permitiu construir o Modelo Multidimensional do Data Warehouse, representado graficamente em anexo na Figura B.1. O esquema obtido para o Data Warehouse foi um esquema em constelação. O esquema em constelação permite que existam várias tabelas de factos, de forma a extrair os diferentes tipos de informação pretendidos, sendo que as várias tabelas de factos partilham tabelas de dimensão. De realçar também que cada entrada das tabelas de factos corresponde a uma ocorrência. O esquema relacional do Data Warehouse está representado de seguida (as chaves primárias estão sublinhadas e as chaves estrangeiras indicadas por "FK"):

- *dimension_time (id_time, year, quarter, month, day, hour, minute, second, dayofweek, weekday)*
- *dimension_location (id_location, location_name, city, locality, street, reference_points, door_nr, ZIP, latitude, longitude, local_desc)*
- *dimension_emergency _type (id_emergency _type, priority, type, description)*
- *fact_occurrence_all (id_occurrence, id_start_time, id_occurrence_location, id_emergency_type)*

  id_start_time: FK (dimension_time)

  id_occurrence_location: FK (dimension_location)

  id_emergency_type: FK (dimension_emergency _type)
- *dimension_unit (id_unit, cod_unit, type_unit, station, d_group)*
- *dimension_destination (id_destination, destination_name)*
- *fact_occurrence_w_units (id_occurrence, id_start_time, id_occurrence_location, id_emergency_type, id_destination, id_activation_time, time_to_activation)*

  id_start_time: FK (dimension_time)

  id_occurrence_location: FK (dimension_location)

  id_emergency_type: FK (dimension_emergency _type)

  id_destination: FK (dimension_destination)

  id_activation_time: FK (dimension_time)
- *dimension_group_unit_w_units (id_occurrence, id_unit)*

  id_occurrence: FK (fact_occurrence_w_units)

  id_unit: FK (dimension_unit)
- *fact_occurrence_complete (id_occurrence, id_start_time, id_occurrence_location, id_ emergency_ type, id_destination, id_activation_time, id_arrive _time, id_leave_time, id_destination_time, time_ to_ activation, time_to_arrive, time_to_leave, time_to_destination)*

  id_start_time: FK (dimension_time)

  id_occurrence_location: FK (dimension_location)

id_emergency_type: FK (dimension_emergency _type)

id_destination: FK (dimension_destination)

id_activation_time: FK (dimension_time)

id_arrive_time: FK (dimension_time)

id_leave_time: FK (dimension_time)

id_destination_time: FK (dimension_time)

- *dimension_group_unit_complete (id_occurrence, id_unit)*

  id_occurrence: FK (fact_occurrence_complete)

  id_unit: FK (dimension_unit)

- *dimension_competition (competition_id, competition_code, competition_name)*

- *dimension_teams (team_id, team_code, team_name)*

- *dimension_football (match_id, match_code, id_date_match, competition_id, location_stadium_id, homeTeam_id, awayTeam_id, score)*

  id_date_match: FK (dimension_time)

  competition_id: FK (dimension_competition)

  location_stadium_id: FK (dimension_location)

  homeTeam_id: FK (dimension_teams)

  awayTeam_id: FK (dimension_teams)

- *fact_football_occurrence (id_occurrence, id_emergency_type, id_event_football, id_occurrence_start_ time, id_occurrence _start_time„ _location)*

  id_emergency_type: FK (dimension_emergency _type)

  id_event_football: FK (dimension_football)

  id_occurrence_start_time: FK (dimension_time)

  id_occurrence_location: FK (dimension_location)

- *dimension_concerts (event_id, event_code, event_name, event_id_start_time, event_id_location, event_popularity)*

  event_id_start_time: FK (dimension_time)

  event_id_location: FK (dimension_location)

- *fact_concerts_occurrence (id_occurrence, id_emergency_type, id_event_concert, id_ occurrence_ start_ time, id_occurrence _start_time, _location)*

  id_emergency_type: FK (dimension_emergency _type)

  id_event_concert: FK (dimension_concerts)

id_occurrence_start_time: FK (dimension_time)

id_occurrence_location: FK (dimension_location)

- *dimension_festivals (event_id, event_code, event_name, event_id_start_time, event_id_end_time, event_id_location, event_popularity)*

  event_id_start_time: FK (dimension_time)

  event_id_end_time: FK (dimension_time)

  event_id_location: FK (dimension_location)

- *fact_festivals_occurrence (id_occurrence, id_emergency_type, id_event_festival, id_ occurrence_ start_ time, id_occurrence _location)*

  id_emergency_type: FK (dimension_emergency _type)

  id_event_festival: FK (dimension_festivals)

  id_occurrence_start_time: FK (dimension_time)

  id_occurrence_location: FK (dimension_location)

- *dimension_weather (id_weather_measure, id_time_measure, id_ station, id_ location_ station_ weather, humidity, wind_direction_id, wind_intensity, wind_intensity_km, accumulated_precipitation, pressure, radiation, temperature)*

  id_time_measure: FK (dimension_time)

  id_location_station_weather: FK (dimension_location)

- *fact_weather_occurrence(id_occurrence, id_emergency_type, id_weather_measure, id_occurrence_ start_ time, id_occurrence_location)*

  id_emergency_type: FK (dimension_emergency _type)

  id_weather_measure: FK (dimension_weather)

  id_occurrence_start_time: FK (dimension_time)

  id_occurrence_location: FK (dimension_location)

Em todos os Data Marts estão presentes as dimensões referentes a localização, tempo e tipo de emergência.

A dimensão Localização, representada pela tabela *dimension_location* guarda as localizações utilizadas no Data Warehouse. Tem como chave primária o atributo *id_location*, uma chave artificial ou *Surrogate Key*, gerada internamente no Data Warehouse e auto-incremental. Esta chave vai ser gerada a partir dos atributos *latitude* e *longitude*, ou seja, para cada conjunto diferente dos atributos latitude e longitude é gerada uma chave diferente. Outros atributos como a cidade, o nome da localização, os pontos de referência ou o código postal estão guardados nesta tabela e qualquer um destes atributos pode estar a NULL.

A dimensão Tempo, representada pela tabela *dimension_time* guarda as várias medições de tempo do Data Warehouse. Tem como chave primária o atributo *id_time*, guardado em formato *Datetime* [6]. Os restantes atributos da dimensão são gerados pela normalização da chave primária, que resulta nos seguintes atributos: ano (*year*), trimestre(*quarter*), mês (*month*), dia (*day*), hora (*hour*), minuto (*minute*), segundo (*second*), dia da semana (*dayofweek*) e o atributo, gerado em formato booleano, denominado dia da semana (*weekday*), em que 1 corresponde aos dias úteis da semana e 0 aos dias não úteis da semana, ou seja, fim-de-semana (não inclui feriados).

A dimensão Tipo de Emergência (*dimension_emergency_type*) é também comum a todos os Data Marts e tem como chave primária uma chave artificial denominada *id_emergency_type* gerada a partir dos atributos prioridade da ocorrência (*priority*) e tipo de ocorrência (*type*), ou seja, cada entrada da tabela guarda um conjunto diferente dos atributos prioridade e tipo.

4.4.1 Data Mart Todas as Ocorrências

A modelação multidimensional do Data Mart com todas as ocorrências, cujo resultado está representado na Figura 4.4, foi realizada com o objetivo de responder a consultas da lista da Secção 4.1.2. Este Data Mart inclui todas as ocorrências, tenham sido ativados meios de socorro ou não.

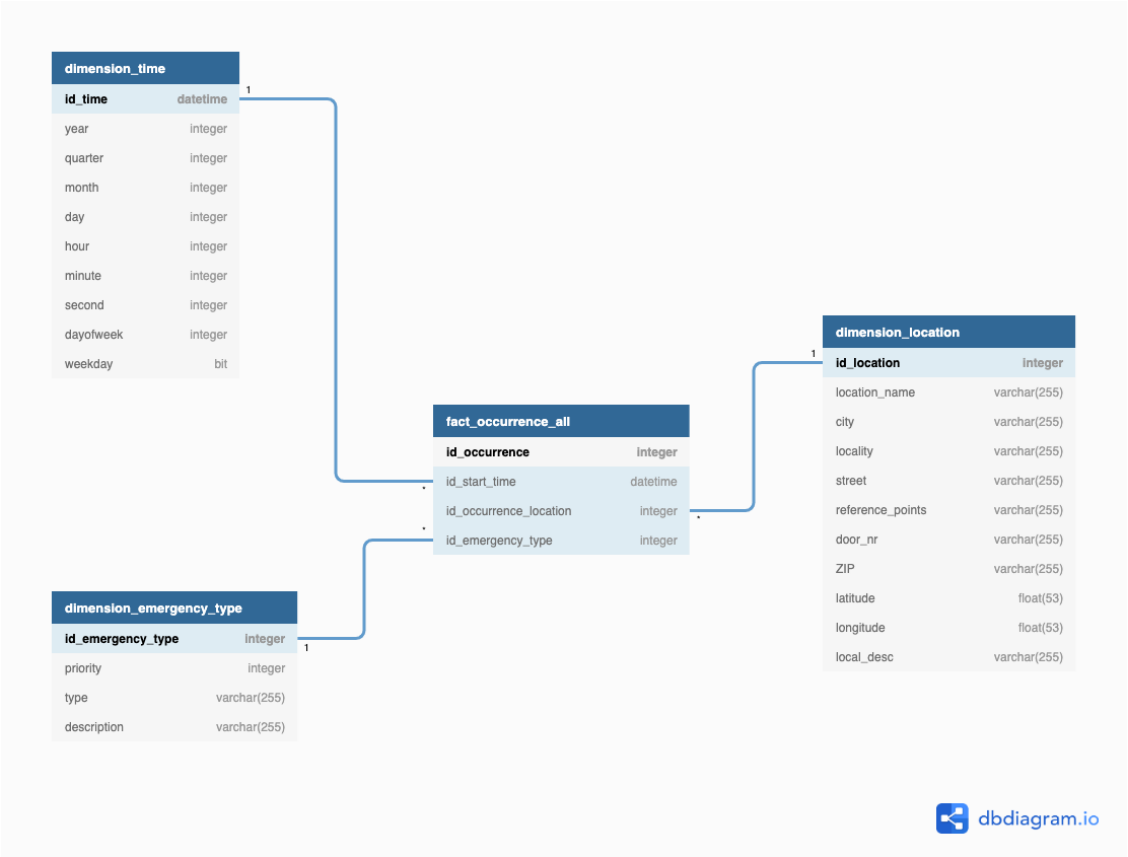

Figura 4.4: Modelo Muldimensional Data Mart com todas as ocorrências

[6] https://docs.microsoft.com/en-us/sql/t-sql/data-types/datetime-transact-sql?view=sql-server-ver15

O Data Mart é composto pelas dimensões Tempo, Localização e Tipo de Emergência. A tabela de factos (*fact_occurrence_all*) tem como chave primária o identificador da ocorrência *id_occurrence*. Os restantes atributos são as chaves estrangeiras das três dimensões que compõem o Data Mart, ou seja, o identificador do tempo de início da ocorrência, o identificador da localização e o identificador do tipo de emergência.

4.4.2 Data Mart Ocorrências com Meios

A modelação do Data Mart com as ocorrências para as quais foram atribuídos meios, como é demonstrado na Figura 4.5, para além das informações presentes no Data Mart com todas as ocorrências (Secção 4.4.1), permite-nos extrair informações relacionadas com as unidades de socorro enviadas e o tempo de despacho ou o destino da primeira unidade enviada.

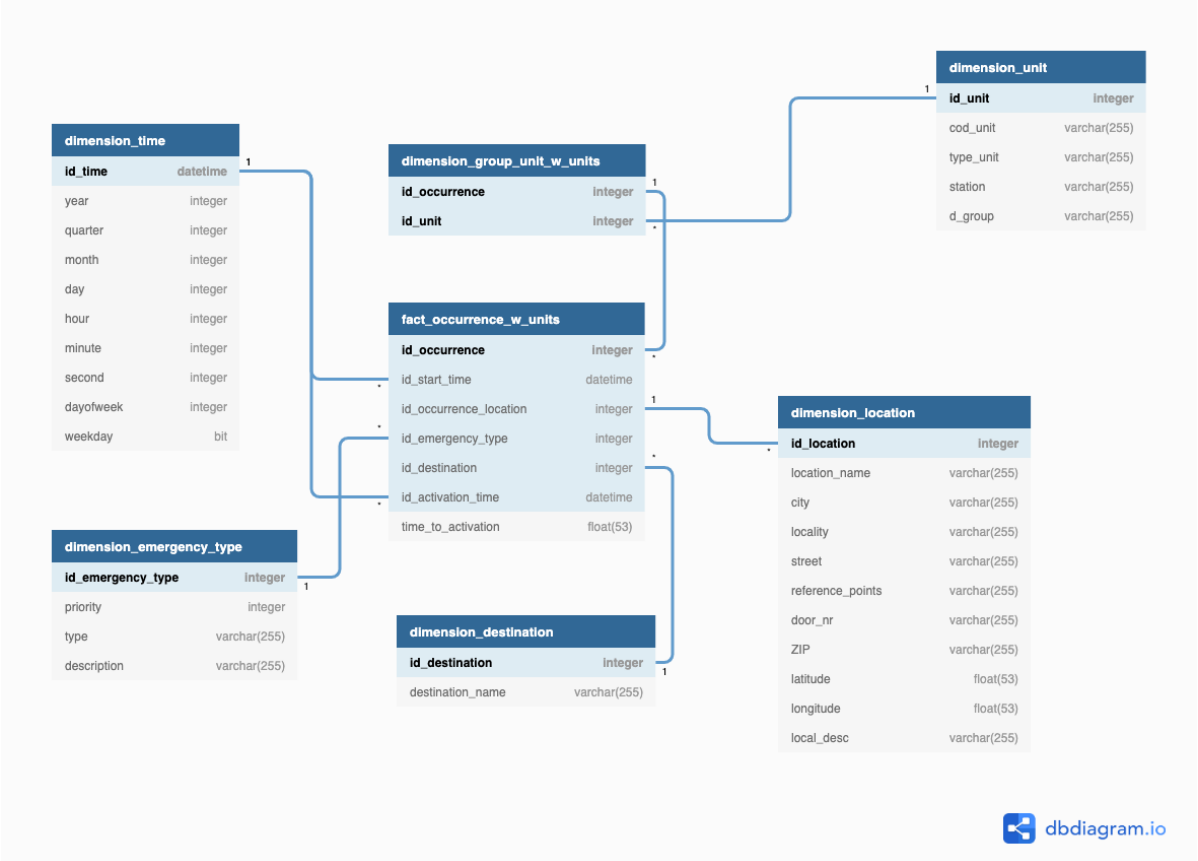

Figura 4.5: Modelo Muldimensional Data Mart Ocorrências com Meios

Para além da dimensão Tempo, Localização e Tipo de Emergência o Data Mart inclui as dimensões: (i) Unidade(*dimension_unit*); (ii) Grupo Unidades (*dimension_group_unit_w_units*); (iii) Destino (*dimension_destination*). Tanto a dimensão Unidade como a dimensão Destino têm como chave primária uma chave artificial auto-incremental.

A dimensão Unidade contém a lista de meios, cada linha da tabela representa um meio e cada meio é representado pela chave primária (*id_unit*), gerada a partir do identificador utilizado na base de dados

do SIADEM (*cod_unit*). Atributos como tipo da unidade (*type_unit*), a sua estação (*station*) ou grupo de despacho (*d_group*) também estão presentes na tabela.

A dimensão Grupo de Unidades é uma tabela ponte. Num Data Warehouse é assumido que as tabelas de dimensão têm uma relação com as tabelas de factos de um para muitos (1:n). Assim, cada registo de uma tabela de dimensão pode estar ligado a mais do que um registo da tabela de factos mas o mesmo não pode acontecer no sentido contrário, já que uma linha da tabela de factos apenas se pode ligar a uma linha na tabela de dimensão. Neste caso específico o Data Warehouse necessita de uma relação de muitos para muitos (n:n), visto que um meio pode estar associado a mais do que uma ocorrência, e uma ocorrência pode necessitar de mais do que um meio. Para obter a modelação correta é necessário transformar esta relação de n:n em duas ligações de 1:n utilizando uma tabela ponte denominada tabela Grupo de Unidades. Esta tabela é constituída por uma chave composta, formada pelo identificador da ocorrência (*id_occurrence*) e pelos identificadores dos meios associados a essa ocorrência (*id_unit*).

A dimensão Destino guarda os destinos dos primeiros meios enviados para cada ocorrência, a sua chave artificial (*id_destination*) é gerada a partir do seu atributo nome do destino (*destination_name*).

A tabela de factos (*fact_ocurrence_w_units*) tem como chave primária o identificador da ocorrência. É também composta pelas chaves estrangeiras das tabelas de dimensão, a não ser para a dimensão Grupo de Unidades que está associada apenas ao identificador da ocorrência. Assim sendo, é composta pela chave do tempo de início da ocorrência, pela chave do tempo de acionamento do primeiro meio, pela chave da localização da ocorrência, pela chave do tipo de emergência e pela chave do destino da primeira unidade. Finalmente, o último atributo da tabela é a medição do tempo de demora (em segundos) até ao acionamento do primeiro meio, obtido através subtração do tempo de início da ocorrência ao tempo de acionamento do primeiro.

4.4.3 Data Mart Ocorrências com Informação Completa

O Data Mart das ocorrências com informação completa, representado na Figura 4.6, é composto pelas ocorrências para as quais foram ativados meios. Neste Data Mart, para além dos tempos de início da ocorrência e de ativação do primeiro meio temos também os tempos de chegada ao local da ocorrência, saída do local da ocorrência e chegada ao local de destino por parte do primeiro meio. As informações relacionadas com as unidades de socorro enviadas e o destino da primeira unidade estão igualmente presentes neste Data Mart.

As tabelas de dimensão deste Data Mart são as mesmas do Data Mart Ocorrências com Meios (Secção 4.4.2), e também precisa de uma tabela ponte (*dimension_group_unit_complete*).

Já a tabela de factos (*fact_occurrence_complete*), para além de conter atributos armazenados na tabela de factos do Data Mart Ocorrências com Meios, tem também em conta os tempos de chegada ao local da ocorrência, saída do local da ocorrência e chegada ao local de destino por parte do primeiro meio. Adicionalmente, a medição do tempo de demora (em segundos) até ao local da ocorrência, até à saída do local da ocorrência e até à chegada ao local de destino também são armazenados na tabela,

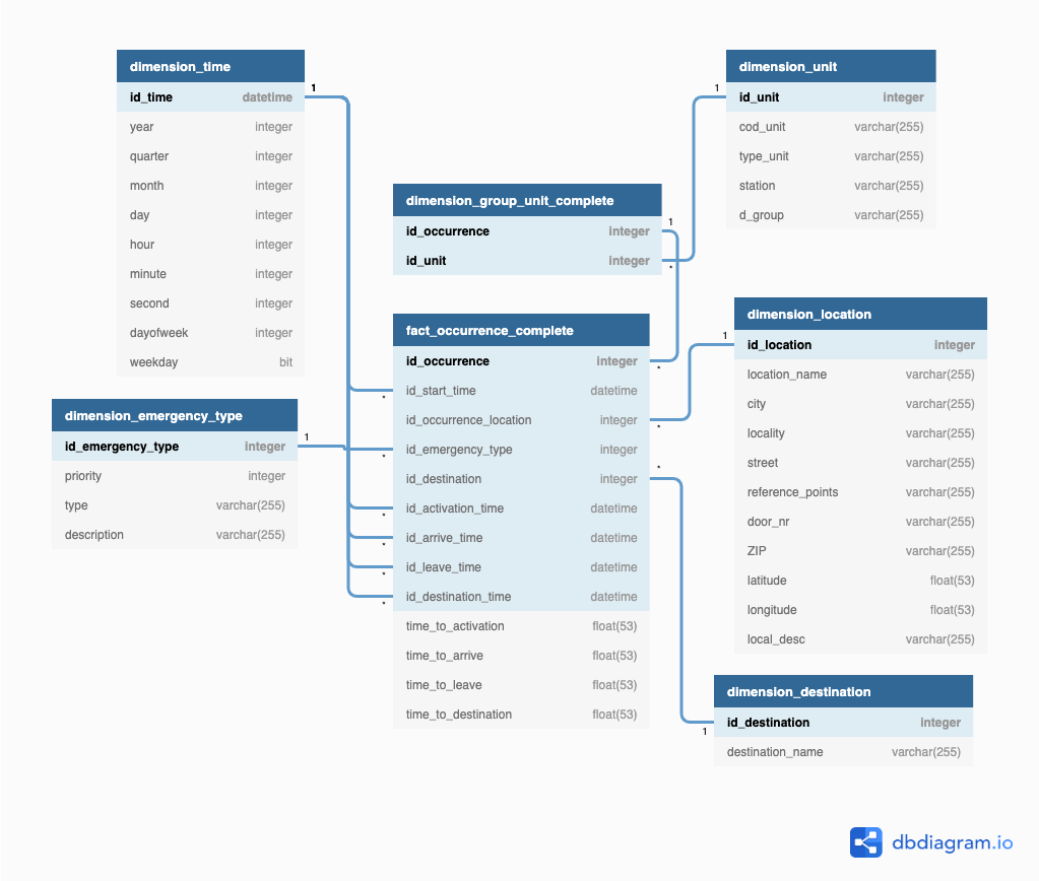

Figura 4.6: Modelo Muldimensional Data Mart Ocorrências com Informação Completa

e são obtidos através da subtração do tempo de início da ocorrência a cada um destes tempos.

4.4.4 Data Mart Futebol

O Data Mart Futebol, representado na Figura 4.7, foi modelado para permitir responder a consultas da Secção 4.1.3 relacionadas com ocorrências em zonas próximas a locais onde decorram eventos desportivos. Este Data Mart é composto pelas ocorrências localizadas num raio de 2 quilómetros a partir do estádio onde se realiza o evento e num intervalo de tempo de quatro horas, que se inicia 60 minutos antes da hora marcada para o início do jogo.

Para além das habituais dimensões Tempo, Localização e Tipo de Emergência este Data Mart inclui as dimensões Futebol (*dimension_football*), Competição (*dimension_competition*) e Equipas (*dimension_teams*).

A dimensão Futebol guarda dados sobre cada jogo de futebol. É composta pelo identificador do jogo (*match_id*), por uma chave artificial gerada através da chave original da fonte de dados de onde foram extraídos (*match_code*), pela data/hora do início do jogo (*id_date_match*), pela chave estrangeira *id_competition* que é referente à competição à qual está inserido o jogo, pela chave estrangeira correspondente à localização do estádio onde se realizou o jogo (*location_stadium_id*), pelas chaves estrangeiras de cada uma das equipas (*homeTeam_id* e *awayTeam_id*), e pelo resultado final do en-

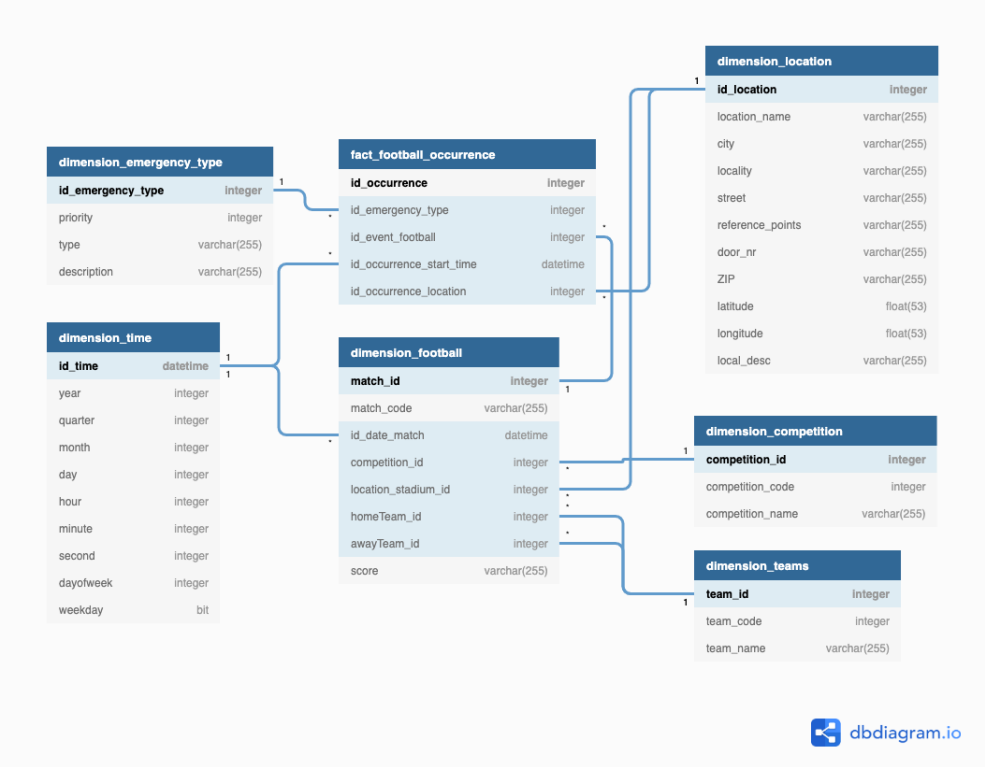

Figura 4.7: Modelo Muldimensional Data Mart Futebol

contro (*score*).

Cada entrada da dimensão Competição representa, logicamente, uma diferente competição. A dimensão é composta pela sua chave artificial identificadora (*competition_id*), obtida pela chave original da sua fonte (*competition_code*) que, por sua vez, também é um atributo da dimensão e pelo nome da competição (*competition_name*). O objetivo desta dimensão é verificar, através de agregações, se existem variações do número de ocorrências de acordo com a competição.

A dimensão Equipas é composta pelos vários clubes. A dimensão guarda o identificador de cada equipa (*team_id*), uma chave artificial gerada pela chave original da fonte de dados (*team_code*) e o nome da equipa (*team_name*). O objetivo da dimensão Equipas é verificar a existência de variações do número de ocorrências de acordo com as equipas presentes em cada jogo.

A tabela de factos do Data Mart Futebol (*fact_football_occurrence*) é composta pelo identificador da ocorrência (*id_occurrence*), e por chaves estrangeiras das várias dimensões que correspondem ao tipo da emergência (*id_emergency_type*), ao evento desportivo mais próximo da localização da ocorrência (*id_event_football*), o tempo de início da ocorrência (*id_occurrence_start_time*) e a localização da mesma (*id_occurrence_location*).

4.4.5 Data Mart Concertos

O Data Mart Concertos, representado na Figura 4.8, foi modelado de forma a permitir a realização das consultas da Secção 4.1.3, relacionadas com ocorrências em zonas próximas a locais onde decorram concertos. Este Data Mart é composto por ocorrências localizadas num raio de 1,5 quilómetros a partir

local onde se realizou o evento, num intervalo de tempo de seis horas, que se inicia três horas antes da hora marcada para início do concerto e termina três horas após a hora marcada para o início do concerto.

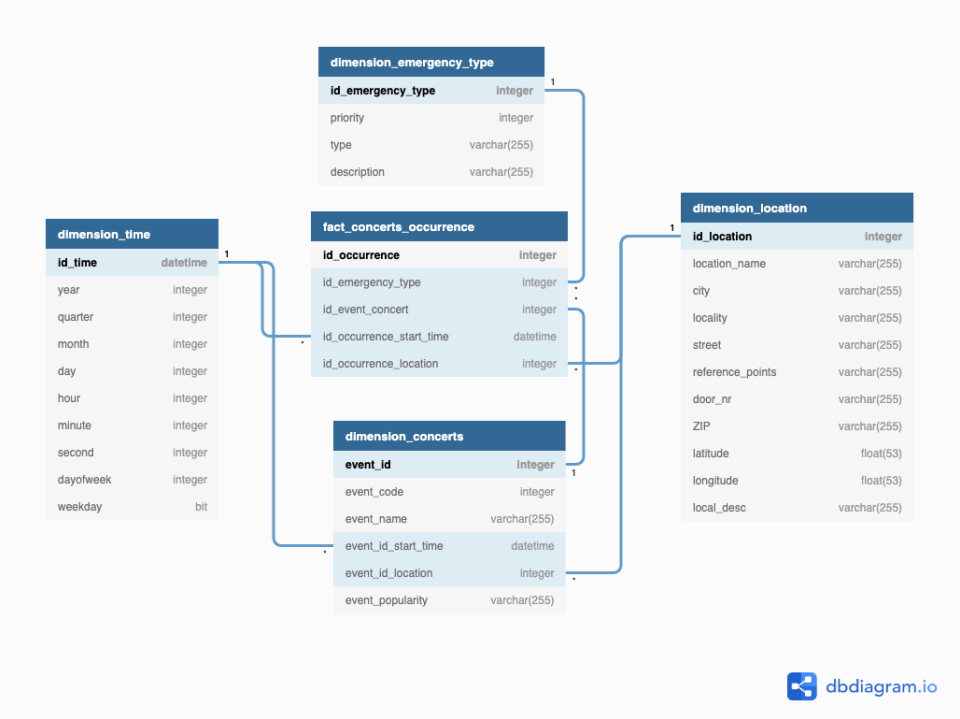

Figura 4.8: Modelo Muldimensional Data Mart Concertos

Neste Data Mart, às dimensões Tempo, Localização e Tipo de Emergência é adicionada a dimensão Concertos *dimension_concerts*.

A dimensão Concertos guarda dados relativos a cada espetáculo. A dimensão tem como chave primária o identificador do concerto (*event_id*), que é uma chave artificial obtida através da chave original da fonte de onde foram extraídos os dados (*event_code*). Os restantes atributos que compõem a dimensão são: a data/hora do início do concerto (*event_id_start_time*), a chave da localização do local onde se realizou o concerto (*event_id_location*) e o atributo popularidade (*event_popularity*), que é uma medida atribuída pela fonte de dados original e poderá trazer alguma informação relacionada com a variação do número de ocorrências de acordo com a popularidade do evento.

A tabela de factos do Data Mart Concertos (*fact_concerts_occurrence*) é composta pelo identificador da ocorrência (*id_occurrence*), que é a sua chave primária. A tabela é também composta pelas chaves estrangeiras correspondentes às várias dimensões: tipo da emergência (*id_emergency_type*), pelo evento próximo da localização da ocorrência (*id_event_concert*), pelo tempo de início da ocorrência (*id_occurrence_start_time*) e pela localização da mesma (*id_occurrence_location*).

4.4.6 Data Mart Festivais

O Data Mart Festivais, representado na Figura 4.9, foi modelado para que se executem as consultas da Secção 4.1.3 relacionadas com ocorrências em zonas próximas de festivais. Para este Data Mart serão

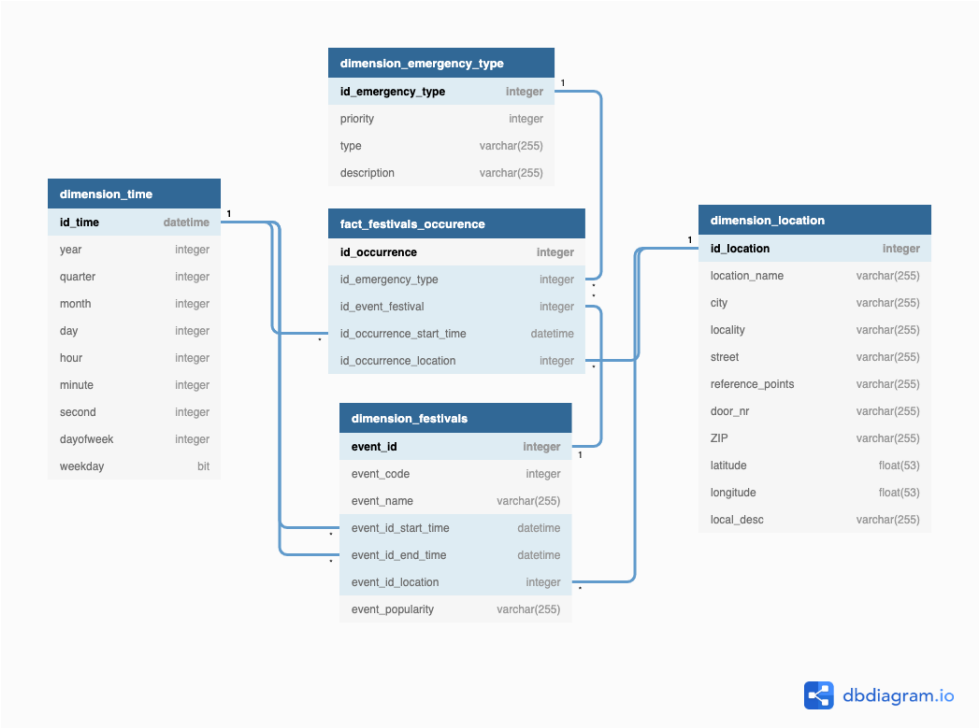

Figura 4.9: Modelo Muldimensional Data Mart Festivais

carregadas as ocorrências localizadas num raio de 3 quilómetros a partir do local de onde se realiza o festival, num intervalo de tempo que vai desde o início do dia marcado para o evento (00:00:00 horas) até às 23:59:59 do dia marcado para o fim do festival.

O raio de três quilómetros foi escolhido porque, por norma, os espaços dos festivais têm grandes áreas que incluem, por vezes, locais para acampamento. Também o intervalo de tempo escolhido está relacionado com as eventuais pernoitas que decorram durante a realização do festival.

O Data Mart Festivais é composto pelas dimensões Tempo, Localização, Tipo de Emergência e Festivais (*dimension_festivals*).

A dimensão Festivais guarda dados relativos a esses mesmos eventos. A dimensão é composta por: o identificador do festival (*event_id*), que é uma chave artificial gerada através da chave original da fonte de dados de onde foram extraídos os dados (*event_code*), pela chave estrangeira da data correspondente ao início do festival (*event_id_start_time*), pela chave estrangeira da data correspondente ao fim do festival (*event_id_end_time*), pela chave estrangeira da localização do local onde se realizou o festival (*event_id_location*) e pelo atributo popularidade (*event_popularity*) que é semelhante ao da dimensão Concerto da Secção 4.4.5.

A tabela de factos do Data Mart Festivais (*fact_festivals_occurrence*) é semelhante à tabela de factos do Data Mart Concertos (Secção 4.4.5). É composta pela sua chave primária que é o identificador da ocorrência (*id_occurrence*), e pelas seguintas chaves estrangeiras: identificador do tipo de emergência (*id_emergency_type*), chave da localização da ocorrência (*id_occurrence_location*), chave do festival associado à ocorrência (*id_event_festival*) e chave do tempo de início da ocorrência (*id_occurrence_start_time*).

Figura 4.10: Modelo Muldimensional Data Mart Meteorologia

4.4.7 Data Mart Meteorologia

O Data Mart Meteorologia, representado na Figura 4.10, deve permitir a execução das consultas da Secção 4.1.3 relacionadas com ocorrências próximas de sensores que fazem medições meteorológicas, num intervalo de tempo de uma hora, que se inicia 30 minutos antes da hora da medição e termina 30 minutos depois da hora da medição.

O Data Mart é composto pela dimensão Meteorologia (*dimension_weather*), e pelas dimensões Tempo, Localização e Tipo de Emergência.

A dimensão Meteorologia guarda dados meteorológicos relativos às medições efetuadas. A dimensão tem como chave primária o identificador da medição (*id_weather_measure*), que é uma chave artificial gerada através dos atributos estação (*station*) e hora da medição (*id_time_measure*). Os outros atributos da dimensão são: a chave estrangeira correspondente à localização da estação onde foi efetuada a medição (*id_location_station_weather*) e por várias medidas obtidas pelo sensor (humidade (*humidity*), identificador da direção do vento (*wind_direction_id*), intensidade do vento (*wind_intensity*), intensidade do vento em quilómetros por hora (*wind_intensity_km*), precipitação acumulada (*accumulated_precipitation*), pressão atmosférica (*pressure*), radiação (*radiation*) e temperatura (*temperature*).

A tabela de factos do Data Mart Meteorologia (*fact_weather_occurrence*) tem como chave primária

o identificador da ocorrência (*id_occurrence*). A tabela também é composta pelas chaves estrangeiras das dimensões: Tipo da Emergência (*id_emergency_type*), Meteorologia (medição à qual está associada a ocorrência) (*id_weather_measure*), Tempo (o tempo de início da ocorrência) (*id_occurrence_ start_ time*) e Localização (localização da ocorrência) (*id_occurrence_location*).

4.5 Processo ETL

Neste capítulo estão apresentadas as transformações que compõem o processo ETL do projeto. Na Secção 4.5.1 são apresentadas transformações que permitem carregar dados na dimensão Tempo. Na Secção 4.5.2 são apresentadas as transformações que inserem dados na dimensão Localização. A Secção 4.5.3 apresenta a transformação que carrega dados na dimensão Tipo de Emergência. As secções seguintes apresentam as transformações necessárias para executar o carregamento de dados em cada um dos Data Marts. Por fim, a Secção 4.5.11, apresenta os *Jobs*: processos que permitem orquestrar as transformações.

Algumas das transformações que formam o processo ETL são muito semelhantes, pelo que, no documento, apenas serão apresentadas algumas em detalhe, sendo que todas as transformações utilizadas em cada um dos Data Marts estão apresentadas em anexo ao documento na Secção D. As transformações representadas nas figuras foram desenvolvidas com o Pentaho Data Integration e são compostas por steps que, se possível, são executados paralelamente e que se ligam entre si.

É importante referir que, caso se opte por carregar em primeiro lugar os dados para o Data Mart Todas as Ocorrências, os tempos de início, as localizações e os tipos de emergência de todas as ocorrências serão inseridos no Data Warehouse, pelo que, quando posteriormente forem carregados os dados nos restantes Data Marts, não é necessário executar as transformações que inserem esses dados específicos.

Devido às semelhanças no modelo multidimensional dos Data Marts Futebol, Concertos, Festivais e Meteorologia apenas será apresentado detalhadamente o Data Mart Futebol (Secção 4.5.7).

4.5.1 Dimensão Tempo

As medições de tempo carregadas para a dimensão Tempo foram: (i) tempos de início das ocorrências; (ii) tempos de acionamento da primeira unidade; (iii) tempos de chegada da unidade ao local da ocorrência; (iv) tempos de saída da unidade do local da ocorrência; (v) tempos de chegada da unidade ao seu destino; (vi) tempos de início dos jogos de futebol; (vii) tempos de início dos concertos; (viii) tempo de início dos festival; (ix) tempo de fim dos festival; (x) tempo das medições meteorológica.

O principal objetivo das transformações que carregam dados para a dimensão Tempo passa por preencher todos os atributos da dimensão através da normalização da chave primária.

Numa fase inicial dos fluxos é necessário extrair os dados da Data Staging Area, convertê-los para formato *Datetime* e homogeneizar o fuso horário, neste caso para o formato *Tempo Universal Coordenado (UTC)*, que é o fuso horário de referência, a nível mundial, e que corresponde à hora de inverno

em Portugal Continental. Numa segunda fase é necessário normalizar as medições de tempo, de forma a obter todos os atributos que constituem a dimensão Tempo e, já numa última fase, carregam-se os atributos obtidos para a dimensão. Os fluxos das transformações que permitiram normalizar os tempos e inseri-los na dimensão foram idênticos para as várias fontes, independentemente da medição de tempo a que correspondem.

Na Figura 4.11 está representado o fluxo da transformação que permitiu carregar os dados relativos ao tempo de início do jogo de futebol na dimensão Tempo. A descrição dos steps deste fluxo está apresentada é a seguinte:

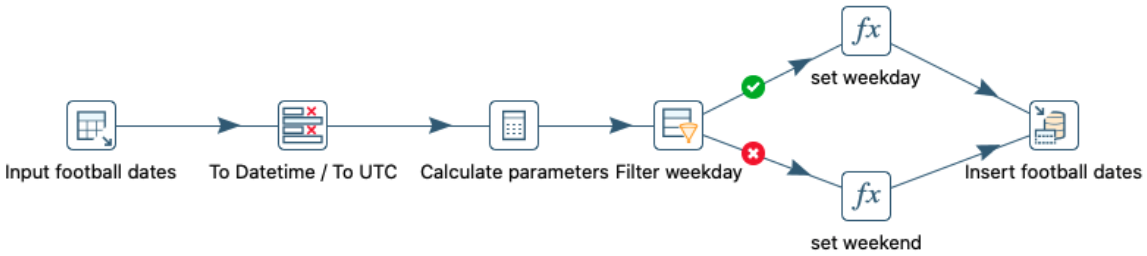

Figura 4.11: Transformação para inserir os tempos de início dos jogos de futebol na dimensão Tempo

Input football dates Extrai os dados relativos aos jogos da Data Staging Area.

To Datetime / To UTC Seleciona os dados relativos ao tempo de início do jogo e faz a conversão para o formato *Datetime* e para o fuso horário *UTC*.

Calculate parameters Normaliza os tempos, calculando os vários parâmetros extraíveis de uma data, como o ano, o mês, o dia, a hora ou o dia da semana.

Filter weekday Verifica se a data em questão corresponde a um dia útil ou a fim de semana. Caso seja dia útil, essa data passa ao *step* “set weekday"e, caso contrário, passa a “set weekend".

set weekday Atribui o valor Booleano 1 aos dias úteis da semana

set weekend Atribui o valor Booleano 0 aos dias não úteis da semana.

Insert football dates Insere na dimensão Tempo os tempos de início dos jogos, em formato *Datetime* (a chave primária), bem como todos os atributos calculados ao longo do fluxo. Caso este tempo já tenha sido inserido no Data Warehouse não efetua quaisquer alterações na dimensão.

4.5.2 Dimensão Localização

A dimensão localização é composta por dados referentes a várias localizações: (i) localização das ocorrências; (ii) localização dos estádios de futebol; (iii) localização dos concertos; (iv) localização dos festivais; (v) localização das estações que efetuam as medições meteorológicas.

O principal objetivo das transformações que carregam dados para a dimensão Localização é preencher o máximo de atributos possíveis em cada entrada da tabela, de acordo com os dados existentes

na fonte de dados original. Numa primeira fase da transformação, é necessário extrair os dados da Data Staging Area. De seguida, caso seja necessário, são realizadas transformações nos dados para que se enquadrem da melhor forma possível com os atributos da dimensão. Depois, é necessário criar uma chave artificial para cada conjunto diferente dos atributos latitude e longitude. Por fim, a transformação carrega os dados para o Data Warehouse.

Os fluxos das transformações que carregam dados para a dimensão Localização são muito semelhantes apesar de serem provenientes de várias tabelas da Data Staging Area. Na Figura 4.12 está representado o exemplo do fluxo que termina com a inserção da localização das ocorrências na dimensão. O fluxo é composto por:

Figura 4.12: Transformação para inserir as localizações das ocorrências na dimensão Localização

Input data Extrai os dados relativos à localização das ocorrências da Data Staging Area, como a latitude, a longitude, rua, nome do local, ou referências da localização.

Concat fields Une (se existirem) os campos tipo de via (Avenida, Rua, Praça...) e nome (Por exemplo: Av. + Almirante Reis = Av. Almirante Reis)

Create/Insert Surrogate Key Cria uma chave artificial para cada conjunto diferente de latitude e longitude. Insere a chave na dimensão bem como as coordenadas geográficas. Se esse conjunto de latitude e longitude já estiver presente na dimensão não efetua alterações.

Insert Occurrence Location Atualiza as localizações, inseridas no step anterior inserindo dados como: o nome da rua, o código DICOFRE ou referências da localização. Caso não existam novos dados para inserir ou atualizar a dimensão, não são efetuadas alterações.

4.5.3 Dimensão Tipo de Emergência

Todas as ocorrências existentes no Data Warehouse têm um tipo e uma prioridade associada. O objetivo da dimensão Tipo de Emergência é que nela estejam todas combinações existentes entre os atributos tipo de ocorrência e prioridade da ocorrÊncia, para cada combinação a dimensão terá uma diferente chave artificial.

Para carregar dados na dimensão Tipo de Emergência é apenas necessário uma transformação, representada na Figura 4.13. Numa primeira fase da transformação é necessário extrair os dados da Data Staging Area. De seguida, criam-se as chaves artificiais a partir dos atributos prioridade e tipo. Posteriormente associa-se a cada atributo tipo a descrição que lhe corresponde, para que seja mais facilmente identificável o motivo da ocorrência. Já numa fase final os dados são inseridos na dimensão.

A descrição do fluxo que deu origem à transformação representada na Figura 4.13 é a seguinte:

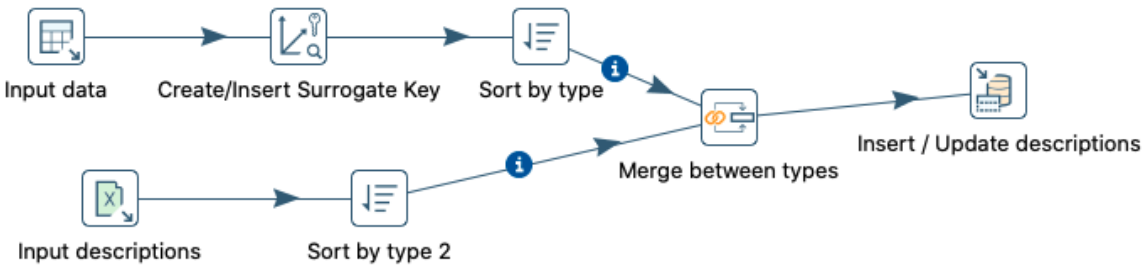

Figura 4.13: Transformação para carregar dados para a dimensão Tipo de Emergência

Input data Extrai os dados relativos aos tipos e prioridades das ocorrências da Data Staging Area.

Create/Insert Surrogate Key Cria uma chave artificial para cada conjunto diferente de prioridade de ocorrência e tipo de ocorrência. Insere a chave artificial gerada, a prioridade e o tipo da ocorrência na dimensão. Caso esse conjunto de prioridade e tipo de emergência já esteja presente na dimensão Tipo de Emergência não efetua alterações.

Sort by type Organiza os registos existentes por ordem alfabética do atributo tipo da emergência.

Input descriptions Extrai uma tabela com duas colunas, em formato *Excel*. Uma das colunas tem os diferentes tipos das ocorrências e a outra tem a descrição dos tipos.

Sort by type 2 Organiza a tabela por ordem alfabética do atributo tipo.

Merge between types Faz uma junção entre as duas tabelas provenientes dos dois subfluxos, utilizando uma operação *INNER JOIN* [7] no atributo tipo. Permite associar os tipos das ocorrências vindos do step "Sort by type" à sua descrição.

Insert / Update descriptions Procura a chave artificial de cada linha e insere as descrições dos tipos das ocorrências caso ainda não tenham sido carregadas.

4.5.4 Data Mart Todas as Ocorrências

Para inserir os dados no Data Mart Todas as Ocorrências são necessárias 4 transformações: (i) A transformação representa em anexo na Figura D.2 para a dimensão Tempo; (ii) A transformação representada na Figura 4.12 e descrita na Secção 4.5.2 para a dimensão Localização; (iii) A transformação descrita na Secção 4.5.3 e representada na Figura 4.13 para a dimensão Tipo de Emergência; (iv) A transformação para carregar os dados para a tabela de factos do Data Mart, representa na Figura 4.14.

Para a tabela de factos pretendemos que sejam carregadas todas as ocorrências registadas na fonte de dados do SIADEM. A transformação que origina o carregamento dos dados para a tabela de factos inicia-se com a extração de dados das ocorrências da Data Staging Area. De seguida, é necessário procurar e associar as chaves artificiais correspondentes a cada ocorrência. Para finalizar, são inseridas as ocorrências na tabela de factos: a sua identificação, bem como as chaves estrangeiras que

[7] https://www.w3schools.com/sql/sql_join_inner.asp

compõem cada entrada da tabela de factos. O fluxo que dá origem à transformação está apresentado seguidamente:

Figura 4.14: Transformação para carregar os dados na tabela de factos do Data Mart Todas as Ocorrências

Input data Extrai os dados relativos às ocorrências da Data Staging Area.

To Datetime / **To UTC** Seleciona os dados relativos ao tempo de início da ocorrência em formato *String*, converte-os para o formato *Datetime* e fuso horário *UTC*.

Lookup_emergency_type Procura na dimensão Tipo de Emergência a chave artificial que corresponde a cada ocorrência.

Lookup_location Procura na dimensão Localização a chave artificial da localização da ocorrência.

Insert / **Update** Insere na tabela de factos todas as ocorrências, ou seja, o identificador da ocorrência e os seus atributos. Caso a ocorrência já tenha sido inserida, o step atualiza as chaves estrangeiras se estas sofrerem alterações.

4.5.5 Data Mart Ocorrências com Meios

Para carregar os dados correspondentes ao Data Mart Ocorrências com Meios, são necessárias as seguintes transformações: (i) Duas transformações idênticas à descrita na Secção 4.5.1 para a dimensão Tempo (uma para inserir o tempo de início da ocorrência, representada na Figura D.2 e outra para inserir o tempo de acionamento do primeiro meio, representada na Figura D.8); (ii) Uma transformação como a da Figura 4.12 para a dimensão Localização; (iii) A transformação descrita na Secção 4.5.3 para a dimensão Tipo de Emergência; (iv) Três transformações a carregar dados na dimensão Unidade; (v) Uma transformação para carregar dados na dimensão Grupo de Unidades; (vi) Uma transformação para carregar dados na dimensão Destino; (vii) Uma transformação para carregar os dados na tabela de factos.

Em relação às transformações utilizadas para carregar dados na dimensão Unidade, duas dessas delas têm como principal objetivo o carregamento de dados para a dimensão. A terceira transformação tem como principal objetivo fazer limpeza dos dados carregados, neste caso atribuir uma “estação” a meios que na base de dados do SIADEM não tenham estações atribuídas.

As duas transformações que carregam dados para a dimensão são idênticas, a diferença entre elas é apenas as tabelas da Data Staging Area de onde são extraídas. Essas transformações iniciam-se com a extração de dados. Em segundo lugar, é atribuído um valor padrão a todas as entradas da fonte original que se encontrem a NULL nos atributos estação e tipo de meio. Seguidamente, são criadas as

chaves artificiais para as unidades a partir do identificador utilizado na base de dados do SIADEM. Por fim, os dados são inseridos na dimensão.

Na Figura 4.15 está representado um dos fluxos que dá origem a uma das transformações que carregam dados para a dimensão Unidade. Os steps do fluxo estão descritos de seguida:

Figura 4.15: Transformação para carregar dados na dimensão Unidade

Input data Extrai os dados relativos às unidades da Data Staging Area.

Station and type not null Atribui um valor padrão a todas as entradas da fonte original que se encontrem a NULL nos atributos estação e tipo de meio, neste caso utiliza-se "no_station" para estação e "NA" para tipo de unidade.

Create/Insert Surrogate Key Cria e insere na dimensão a chave artificial gerada a partir do identificador utilizado na fonte de dados original. Caso a chave já esteja presente na dimensão não efetua alterações.

Insert / Update Insere ou atualiza os atributos de cada unidade.

Em relação à transformação utilizada para limpeza dos dados carregados, esta inicia-se com dois subfluxos com extrações de dados da dimensão Unidade. De seguida, há uma filtragem nos subfluxos, num deles mantêm-se os meios com estação associada e no outro os que não têm estação associada. Depois, unindo os dois fluxos, e porque o identificador dos meios é formado a partir do identificador da estação a que o meio corresponde, é utilizado o algoritmo de comparação de cadeias de caracteres (ou strings) *Jaro* para obter os valores Jaro mais altos no identificador dos meios. Depois, excluem-se da transformação os meios que tenham um valor de similaridade inferior a 0,7. Para finalizar, assume-se que as unidades que se mantêm no fluxo pertencem à mesma estação e atualizamos o atributo estação na dimensão. O fluxo que dá origem a esta transformação está representado na Figura 4.16, e descrito de seguida:

Input data 1 Extrai dados relativos às unidades da dimensão Unidade.

Filter by station != no_station Filtra as entradas da dimensão Unidade, exclui as que têm o valor do atributo estação igual a "no_station".

Input data 2 Extrai dados relativos às unidades da dimensão Unidade.

Select Values Atribui um nome diferente ao atributo de identificação do meio e estação. Isto para facilitar a distinção ao longo do fluxo.

Filter by station = no_station Filtra as entradas da dimensão Unidade, exclui as que têm o valor do atributo estação diferente de "no_station".

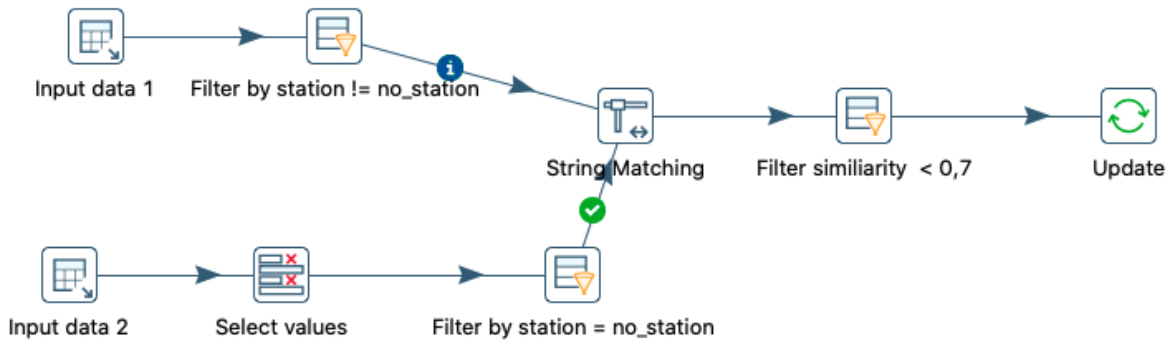

Figura 4.16: Transformação para inserir o atributo estação quando este não existe

String Matching Utiliza o algoritmo de comparação de strings *Jaro* para obter as similaridades com valor mais alto no atributo de identificação da unidade.

Filter similiarity > 0,7 Exclui as entradas com valor de similaridade inferior a 0,7.

Update Atualiza o atributo estação na dimensão Unidade para os registos filtrados.

Para efetuar o carregamento de dados na dimensão Grupo de Unidades é necessária uma transformação, representada na Figura 4.17. A transformação inicia-se com a extração de dados da Data Staging Area. Posteriormente são filtradas ocorrências para que fiquem no fluxo apenas as ocorrências que tiveram meios ativados. De seguida, é necessário procurar e associar as chaves artificiais das unidades que foram ativadas em cada ocorrência. Finalmente, inserem-se na dimensão as ocorrências e os meios associados a cada uma delas. A descrição dos steps da transformação é a seguinte:

Figura 4.17: Transformação para carregar dados na dimensão Grupo de Unidades no Data Mart Ocorrências com Meios

Input data Extrai dados relativos às ocorrências da Data Staging Area.

Filter occurrences Filtra as ocorrências extraídas, excluindo aquelas para as quais não existem unidades associadas.

Lookup_unit Procura na dimensão Unidade a chave artificial das unidades associadas a cada ocorrência.

Insert / **Update** Insere na dimensão as ocorrências e as suas unidades associadas, isto é, os identificadores das ocorrências e a chave artificial das unidades.

Em relação à transformação utilizada para efetuar o carregamento de dados na dimensão Destino, esta inicia-se com a extração dos dados relativos aos destinos dos meios. De seguida, para todas as

entradas da fonte original em que as ocorrências não tenham um destino associado, isto é o atributo destino a NULL, é inserido o valor padrão “sem destino”. Por fim, são geradas na transformação as chaves artificiais a partir do identificador utilizado para o atributo destino na base de dados do SIADEM. A transformação está representada na Figura 4.18, e a descrição dos steps do fluxo é a seguinte:

Figura 4.18: Transformação para carregar dados na dimensão Destino

Input data Extrai os dados relativos aos destinos das ocorrências da Data Staging Area.

If destination is null Atribui o valor padrão “sem destino” a todas as entradas que se encontrem a NULL no atributo destino.

Create/Insert Surrogate Key Cria uma chave artificial para o atributo destino a partir do identificador utilizado na fonte de dados original. Insere na dimensão a chave artificial e o identificador que a originou. Caso o destino já esteja inserido na dimensão, não efetua alterações.

A transformação que permite inserir os dados na tabela de factos, está representada na Figura 4.19. Inicia-se com a extração das ocorrências da Data Staging Area. De seguida, há uma filtragem e são excluídas do fluxo as ocorrências sem meios associados. Depois, é atribuído o valor “sem destino” ao atributo destino nas ocorrências que não tenham um destino associado e a conversão nas medições de tempo para formato *Datetime*. A seguir, é necessário procurar as chaves artificiais e associa-las a cada ocorrência. Por fim, é calculado o tempo (em segundos) até ao acionamento da primeira unidade e são inseridos os dados na tabela de factos. Os steps que constituem a transformação são:

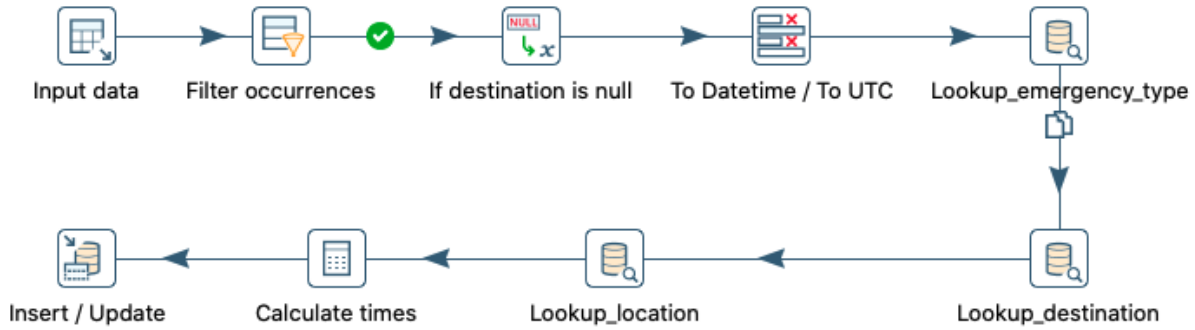

Figura 4.19: Transformação para carregar dados na tabela de factos do Data Mart Ocorrências com Unidades

Input data Extrai os dados das ocorrências da Data Staging Area.

Filter occurrences Filtra as ocorrências extraídas, excluindo aquelas para as quais não existem unidades associadas.

If destination is null Atribui o valor padrão "sem destino" às entradas em que o valor do atributo destino seja NULL.

To Datetime / **To UTC** Seleciona os dados relativos ao tempo de início da ocorrência em formato *String*, converte-os para formato *Datetime* e para o fuso horário *UTC*.

Lookup_emergency_type Procura na dimensão Tipo de Emergência a chave artificial correspondente a cada ocorrência.

Lookup_destination Procura na dimensão Destino a chave artificial do destino de cada ocorrência.

Lookup_location Procura na dimensão Localização a chave artificial correspondente à localização de cada ocorrência.

Calculate times Executa o cálculo do tempo de duração até ao acionamento da primeira unidade, isto é, subtrai o tempo de início da ocorrência ao tempo de acionamento da primeira unidade.

Insert / **Update** Insere na tabela de factos o identificador de cada ocorrência e os seus restantes atributos. Caso a ocorrência já tenha sido inserida, o step atualiza os restantes atributos se estes sofrerem alterações.

4.5.6 Data Mart Ocorrências com Informação Completa

Para carregar os dados correspondentes ao Data Mart Ocorrências com Informação Completa, são necessárias as seguintes transformações: (i) Cinco transformações para a dimensão Tempo (para o tempo início da ocorrência (Figura D.2), para o tempo de acionamento do primeiro meio (Figura D.8), para o tempo de chegada ao local da ocorrência (Figura D.16), para o tempo de saída do local (Figura D.17) e para o tempo de chegada ao destino (Figura D.18)); (ii) A transformação descrita na Secção 4.5.2 para a dimensão Localização; (iii) A transformação descrita na Secção 4.5.3 para a dimensão Tipo de Emergência; (iv) Três transformações a carregar dados na dimensão Unidade (Figuras D.9, D.10 e D.11); (v) Uma transformação para carregar dados na dimensão Grupo de Unidades; (vi) Uma transformação para carregar dados na dimensão Destino (Figura D.12); (vii) Uma transformação para carregar os dados na tabela de factos.

O Data Mart com Informação Completa é composto por um subconjunto de dados presentes Data Mart Ocorrências com Meios e naturalmente também subconjunto do Data Mart Todas as Ocorrências. Visto isto, caso já tenham sido carregados os dados para estes Data Marts apenas é necessário executar as transformações para carregar os novos tempos na dimensão Tempo, para carregar dados na dimensão Grupo de Unidades e para carregar dados na tabela de factos.

A transformação que executa o carregamento de dados para a dimensão Grupo de Unidades, representado na Figura D.20), é semelhante à do Data Mart Ocorrências com Meios descrito na secção 4.5.5 e representada na Figura 4.17. O que difere as transformações é a filtragem das ocorrências, neste caso apenas são inseridas no Data Mart as ocorrências que tenham meios associados e que tenham

registados os tempos de início da ocorrência, de ativação do primeiro meio, de chegada ao local da ocorrência, saída do local da ocorrência e chegada ao local de destino.

Por sua vez, a transformação que permite inserir dados na tabela de factos, representado na Figura D.19, é muito semelhante à da Figura 4.19. As diferenças entre as transformações ocorrem na fase de filtragem das ocorrências e na fase de cálculo dos tempos, já que nesta transformação para além do tempo até ao acionamento do primeiro meio, é calculado também o tempo até à chegada ao local da ocorrência, até à saída do local da ocorrência e até à chegada ao local de destino.

4.5.7 Data Mart Futebol

As transformações necessárias para carregar os dados no Data Mart Futebol são: (i) Duas transformações para a dimensão Tempo; (ii) Duas transformações para a dimensão Localização; (iii) A transformação descrita na Secção 4.5.3 para a dimensão Tipo de Emergência; (iv) Uma transformação para a dimensão Equipas; (v) Uma transformação para a dimensão Competição; (vi) Uma transformação para carregar os dados na tabela de factos.

Em relação à dimensão Tempo, são utilizadas duas transformações porque é necessário inserir o tempo de início das ocorrências (Figura D.2) e o tempo de início dos jogos (Figura D.22). O mesmo acontece na dimensão Localização, é necessário inserir a localização das ocorrências (Figura D.3) e a localização dos estádios (Figura D.23).

Para executar o carregamento de dados para a dimensão Equipas é necessário executar uma transformação, que está representada na Figura 4.20. A transformação inicia-se com a extração de dados da Data Staging Area. Depois, são criadas e inseridas as chaves artificiais correspondentes a cada equipa. Para finalizar, são inseridos na dimensão os nomes dessas equipas. O fluxo que origina a transformação é o seguinte:

Figura 4.20: Transformação para carregar dados na dimensão Equipas

Input data Extrai os dados relativos às equipas de futebol da Data Staging Area.

Create/Insert Surrogate Key Cria e insere na dimensão uma chave artificial para cada uma das equipas através da chave identificadora de cada equipa na fonte de dados original, que também é inserida na dimensão. Caso a chave artificial da equipa já tenha sido inserida na dimensão, não são efetuadas alterações.

Insert Team names Insere o nome de cada uma das equipas na dimensão.

A transformação que executa o carregamento de dados para a dimensão Competição, representada na Figura 4.21, inicia-se com a fase de extração de dados. Depois são criadas e inseridas na

dimensão as chaves artificias de cada competição. Finalmente, são inseridos os nomes nas respetivas competições. A descrição dos steps da transformação é a seguinte:

Figura 4.21: Transformação para carregar dados na dimensão Competição

Input data Extrai os dados relativos às competições de futebol da Data Staging Area.

Create/Insert Surrogate Key Cria e insere na dimensão uma chave artificial para cada uma das competições através do identificador utilizado para cada competição na fonte de dados original, que por sua vez também é inserido na dimensão. Caso a chave artificial da competição já tenha sido inserida na dimensão, não há alterações.

Insert Team names Insere o nome de cada competição na dimensão.

A transformação executada para inserir dados na dimensão Futebol inicia-se com a extração de dados da tabela *Football* da Data Staging Area. Depois, há uma conversão no fuso horário da hora marcada para o início do jogo para *UTC*. É criada uma chave artificial para cada jogo, a partir do identificador do jogo na fonte original. De seguida, procuram-se as chaves artificiais correspondentes aos restantes atributos do encontro: a chave da competição, da localização do estádio e das equipas. Por fim, são inseridos os dados dos atributos de cada jogo. A transformação está representada na Figura 4.22, e de seguida estão apresentados os steps do fluxo:

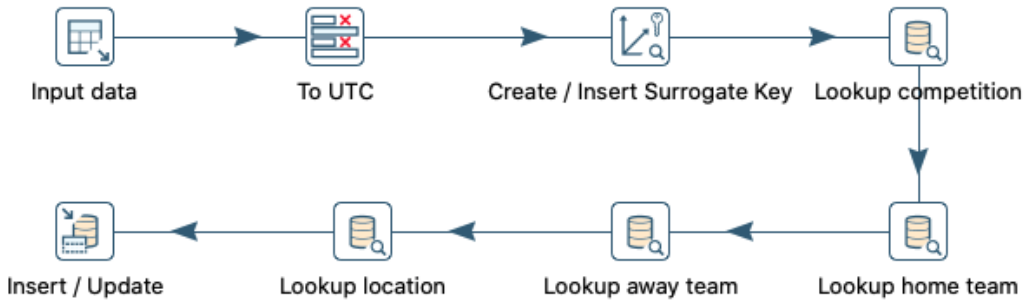

Figura 4.22: Transformação para carregar dados na dimensão Futebol

Input data Extrai os dados dos jogos de futebol da Data Staging Area.

To UTC Seleciona os dados relativos ao tempo de início dos jogos de futebol, já em formato *Datetime* e passa-os para o fuso horário *UTC*.

Create/Insert Surrogate Key Gera e insere na dimensão uma chave artificial para cada jogo através do identificador do jogo utilizado na fonte de dados original, sendo que esse identificador também é inserido na dimensão. Caso a chave artificial do jogo já tenha sido inserida na dimensão, não são efetuadas alterações.

Lookup_competition Procura na dimensão Competição a chave artificial da competição a que cada jogo corresponde.

Lookup_home_team Procura na dimensão Equipas a chave artificial da equipa que joga em casa.

Lookup_away_team Procura na dimensão Equipas a chave artificial da equipa que joga fora.

Lookup_location Procura na dimensão Localização a chave artificial da localização dos estádios onde se realizaram os jogos.

Insert / **Update** Insere na tabela de factos os atributos de cada jogo. Caso o jogo já tenha sido inserido, o step atualiza os atributos de cada jogo se estes sofrerem alterações.

Para finalizar, a transformação executada para inserir os dados na tabela de factos está representada na Figura 4.23. A transformação inicia-se com extração de todas as ocorrências, e de todos os jogos de futebol da Data Staging Area. De seguida, são excluídas as ocorrências que se iniciem a horas em que não existam jogos de futebol (entre as 0 e as 8h). Com os jogos de futebol, calculamos o intervalo em que pretendemos encontrar ocorrências relacionadas com cada jogo, isto é o intervalo de tempo de 4 horas que se inicia uma hora antes do encontro e termina 4 horas depois. A seguir, já com o intervalo de tempo associado a cada jogo definido, fazemos um produto cartesiano cruzando o intervalo de tempo definido para os jogos de futebol com a hora de início das ocorrências, que resultará numa tabela com a combinação de todas essas linhas. De seguida, para cada entrada dessa tabela calculamos a distância entre a ocorrência e o estádio onde se realizou o jogo. Com as distâncias calculadas, é executada uma filtragem na tabela e são excluídas todas as ocorrências que estejam a uma distância superior a 2 quilómetros do estádio. Por fim, já com as ocorrências que provavelmente estão associadas a cada jogo de futebol, procuram-se as chaves estrangeiras que correspondem aos restantes atributos da tabela de factos, e são inseridas na tabela de factos. A descrição dos steps da transformação é a seguinte:

Input data Extrai os dados das ocorrências da Data Staging Area.

To Datetime / **To UTC** Seleciona os dados relativos ao tempo de início da ocorrência em formato *String*, converte-os para o formato *Datetime* e para o fuso horário *UTC*.

Calculate hours Extrai do atributo tempo de início da ocorrência o valor referente à hora.

Filter occurrences by hours Executa uma filtragem nas ocorrências excluindo as que têm início entre as 0h e as 8h.

Input data 2 Extrai da Data Staging Area os dados relativos a jogos de futebol.

To UTC Seleciona os dados relativos ao tempo de início dos jogos de futebol, já em formato *Datetime* e passa-os para o fuso horário *UTC*.

Calculate start/end hour Calcula a partir da hora marcada para o início a hora inicial e final do intervalo de tempo em que procuramos relacionar ocorrências.

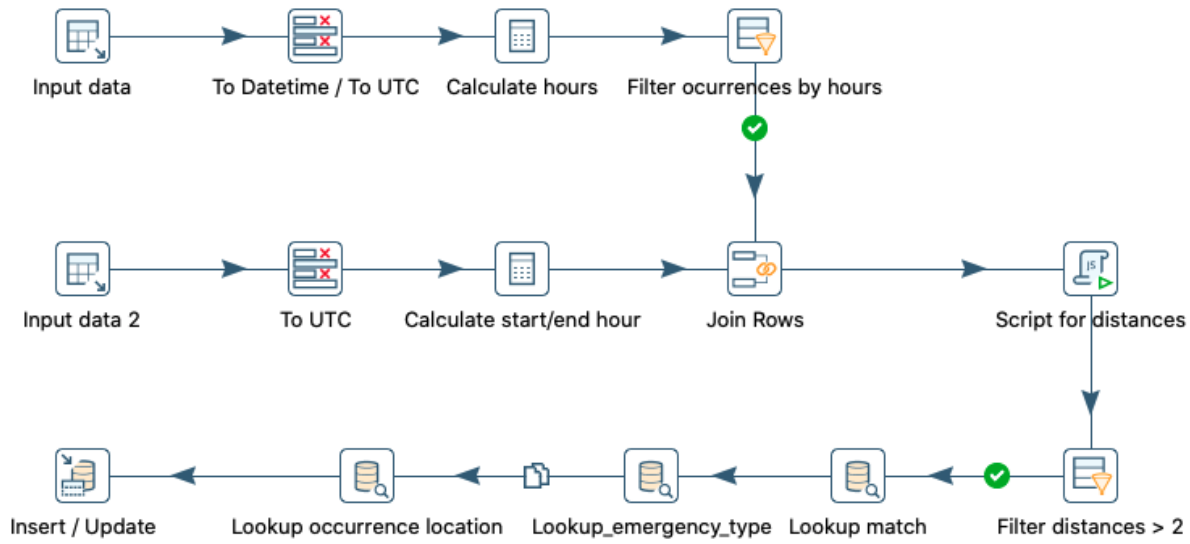

Figura 4.23: Transformação para carregar dados na tabela de factos do Data Mart Futebol

Join Rows Faz o produto cartesiano entre o fluxo que provém do step "Filter occurrences by hours" e do step "Calculate start/end hour", com a condição de que a hora de ocorrência tem de estar no intervalo de tempo definido para o jogo.

Script for distances Executa um programa, representado em anexo na Figura D.1, que a partir das coordenadas geográficas (latitude e latitude) do local da ocorrência e do estádio, calcula a distância em quilómetros entre os dois pontos.

Filter distances > 2 Filtra as ocorrências, mantendo no fluxo apenas as que estejam a uma distância inferior a 2 quilómetros do estádio onde se realiza o jogo.

Lookup_match Procura na dimensão Futebol a chave artificial do jogo a que cada ocorrência está associada.

Lookup_emergency_type Procura na dimensão Tipo de Emergência a chave artificial correspondente a cada ocorrência.

Lookup_occurrence_location Procura na dimensão Equipas a chave artificial da equipa que joga fora.

Lookup_location Procura na dimensão Localização a chave artificial da localização da ocorrência.

Insert / Update Insere na tabela de factos o identificador de cada ocorrência, bem como as chaves estrangeiras correspondentes. Caso a ocorrência já tenha sido inserida, o step atualiza as chaves estrangeiras se estas sofrerem alterações.

4.5.8 Data Mart Concertos

Para carregar os dados no Data Mart Concertos são necessárias as seguintes transformações: (i) Duas transformações para a dimensão Localização; (ii) Duas transformações como a descrita na Secção 4.5.1 para a dimensão Tempo; (iii) A transformação da Secção 4.5.3 para a dimensão Tipo de Emergência; (iv) Uma transformação para a dimensão Concertos; (v) Uma transformação para carregar os dados na tabela de factos.

Como referido na introdução do capítulo, as transformações que executam o carregamento de dados para o Data Mart Concertos são muito semelhantes às da Secção 4.5.7. Em relação à dimensão Tempo, são necessárias duas transformações porque é necessário inserir o tempo de início das ocorrências (Figura D.2) e o tempo de início dos concertos (Figura D.29). Para a dimensão Localização é necessário inserir a localização das ocorrências (Figura D.3) e a localização dos concertos (Figura D.30). A transformação que insere os dados na dimensão Concertos, representado na Figura D.31, é semelhante à que insere dados na dimensão Futebol. Em relação à tabela de factos, representada na Figura D.32, apesar de seguir o mesmo modelo do Data Mart Futebol, o intervalo de tempo e a distância que associa as ocorrências aos concertos é diferente. O intervalo de tempo passa a 6 horas, iniciando-se 3 horas antes da hora marcada para o início do concerto e a distância passa a 1,5 quilómetros.

4.5.9 Data Mart Festivais

As transformações utilizadas para carregar os dados no Data Mart Festivais são: (i) Duas transformações para a dimensão Localização; (ii) Três transformações para a dimensão Tempo; (iii) Uma transformação para dimensão Tipo de Emergência; (iv) Uma transformação para a dimensão Festivais; (v) Uma transformação para carregar os dados na tabela de factos.

Para a dimensão Localização, é necessário inserir as localizações das ocorrências (Figura D.3) e dos festivais (Figura D.36). A dimensão Tempo necessita de uma transformação para inserir o tempo de início das ocorrências (Figura D.2), uma para a data de início dos festivais (Figura D.34) e uma para a data do final dos festivais (Figura D.34). A transformação que insere os dados na dimensão Festivais, representada na Figura D.37 é idêntica à que insere dados na dimensão Concertos. Na tabela de factos, representada na Figura D.38, o intervalo de tempo é o compreendido entre as datas de início e fim do festival e a distância é de 3 quilómetros.

4.5.10 Data Mart Meteorologia

As transformações utilizadas para carregador os dados no Data Mart Meteorologia são: (i) Duas transformações para a dimensão Tempo; (ii) Duas transformações para a dimensão Localização; (iii) Uma transformação para a dimensão Tipo de Emergência; (iv) Uma transformação para a dimensão Meteorologia; (v) Uma transformação para carregar os dados na tabela de factos.

Apesar de o conceito ser um pouco diferente, o Data Mart Meteorologia apresenta um modelo muito semelhante aos Data Marts que relacionam ocorrências com eventos. Sendo que, neste Data Mart

considerou-se que uma medição seria como que um evento.

Portanto, para a dimensão Tempo, são necessárias duas transformações, uma para o tempo de início das ocorrências (Figura D.2) e uma para o tempo da medição executada pelo sensor (Figura D.40). Na dimensão Localização, são inseridas as localizações das ocorrências (Figura D.3) e as localizações dos sensores (Figura D.41). A transformação que carrega os dados para a dimensão Meteorologia está representada na Figura D.42, e é semelhante às transformações que carregam dados para as dimensões Futebol, Concertos e Festivais. Para a transformação que insere dados na tabela de factos, representada na Figura D.43 o intervalo de tempo associado a cada medição é de uma hora, inicia-se 30 minutos antes da medição e termina 30 minutos depois da medição, as distâncias variam consoante a localização das estações.

4.5.11 *Jobs*

No processo ETL do projeto Data2Help, os *Jobs* foram utilizados para executar de forma sequencial as transformações que carregam os dados em cada Data Mart. Os *Jobs* tal como as transformações são compostos por steps que se ligam entre si, e podem ser executados várias vezes para efetuar o refrescamento dos dados do Data Warehouse. A execução dos *Jobs* pode ser agendada para que se inicie de forma automática.

Em relação à ordem sequencial da execução das transformações num *Job*, é importante referir que as dimensões sem chaves estrangeiras devem ser executas em primeiro lugar, e que os dados só podem ser carregados para uma dimensão que tenha uma chave estrangeira de outra dimensão após os dados terem sido carregados para a segunda dimensão.

Os *Jobs* que permitem carregar dados em cada Data Mart estão representados no anexo na Secção D, logo após a representação das transformações que os compõem. Na Figura 4.24 está representado o *Job* que permite carregar os dados no Data Mart Todas as Ocorrências. A descrição dos steps do Job é a seguinte:

Figura 4.24: Job que executa as transformações para carregar os dados no Data Mart Todas as Ocorrências

START O step que dá início ao *Job*, é este step que permite o agendamento para execução automática das transformações.

dim_time Executa a transformação que carrega os tempos de início das ocorrências na dimensão Tempo.

dim_location Executa a transformação que insere a localização das ocorrências na dimensão Localização.

insert DICOFRE Executa a transformação que insere o nome do Distrito, Concelho e Freguesia correspondentes à localização da ocorrência.

dim_emerg _type Executa a transformação que carrega os dados na dimensão Tipo de Emergência.

fact_all_occurrences Executa a transformação que carrega os dados referentes a todas as ocorrências na tabela de factos.

Capítulo 5

Validação Experimental

Neste capítulo, é apresentada e discutida a validação da Solução descrita no Capítulo 4. Na Secção 5.1 está apresentada a validação das consultas definidas para a integração de dados. Na Secção 5.2, está apresentado o desempenho do Data Warehouse em comparação com o desempenho da base de dados do SIADEM. Na Secção 5.3 apresenta-se a validação das ocorrências carregadas para o Data Warehouse.

5.1 Validação das Consultas

Para validar o Data Warehouse, é necessário verificar se é possível responder às consultas identificadas para a integração de dados (Secção 4.1.2 e Secção 4.1.3). Na Secção 5.1.1 está apresentada a validação das consultas que utilizam apenas dados do SIADEM. A Secção 5.1.2 apresenta a validação das consultas que utilizam dados do SIADEM e dados externos.

5.1.1 Validação das consultas apenas com dados do SIADEM

De seguida, para as consultas da Secção 4.1.2, estão apresentadas as consultas em português e na linguagem *Transact-SQL* [1], bem como o resultado obtido da execução das consultas em SQL.

1. **Ocorrências/Tempo**

   (a) Número de ocorrências numa data

   Por exemplo: 28/03/2018

```
SELECT CONVERT (date, id_start_time) as DATE, COUNT (*) as TOT_OCO
FROM fact_occurrence_all
WHERE CONVERT (date, id_start_time) = '2018/03/28'
GROUP by CONVERT (date, id_start_time)
```

[1] https://docs.microsoft.com/en-us/sql/t-sql/language-reference?view=sql-server-ver15

DATE	TOT_OCO
2018-03-28	3583

(b) Número de ocorrências num intervalo de tempo

Por exemplo: Entre 28/03/2018 e o final do dia 31/03/2018

```
SELECT Count (*) as TOT_OCO
FROM fact_occurrence_all
WHERE id_start_time BETWEEN '2018-03-28' AND '2018-03-31 23:59:59.000'
```

TOT_OCO
14093

(c) Número de ocorrências por ano

```
SELECT year,
       COUNT(*) as TOT_OCO
FROM  dimension_time
INNER JOIN fact_occurrence_all ON
fact_occurrence_all.id_start_time = dimension_time.id_time
GROUP BY year
ORDER BY year
```

year	TOT_OCO
2012	651529
2013	1120101
2014	1170343
2015	1215521
2016	1316129
2017	1337265
2018	1400791
2019	1415393
2020	226844

(d) Número de ocorrências por ano / mês

```
SELECT year,
   month,
       COUNT(*) as TOT_OCO
FROM  dimension_time
INNER JOIN fact_occurrence_all ON
```

```
fact_occurrence_all.id_start_time = dimension_time.id_time
GROUP BY year,
 month
ORDER BY year, month
```

year	month	TOT_OCO
2012	5	63187
2012	6	79883
2012	7	83092
2012	8	84699
2012	9	80577
...	...	...
2019	10	120628
2019	11	116241
2019	12	122954
2020	1	128392
2020	2	98452

(e) Número de ocorrências por dia da semana

```
SELECT dayofweek,
COUNT (id_occurrence) as  TOT_OCO
FROM  dimension_time
INNER JOIN fact_occurrence_all ON
fact_occurrence_all.id_start_time = dimension_time.id_time
GROUP BY dayofweek
ORDER BY dayofweek
```

dayofweek	TOT_OCO
1	1367220
2	1476013
3	1398063
4	1394621
5	1410406
6	1430397
7	1377196

2. **Ocorrências/Localização**

(a) Número de ocorrências por Distrito

```
SELECT Distrito,
COUNT(id_occurrence) as TOT_OCO
FROM   dimension_location
INNER JOIN fact_occurrence_all ON
fact_occurrence_all.id_occurrence_location=dimension_location.id_location
GROUP BY Distrito
ORDER BY Distrito
```

Distrito	TOT_OCO
NULL	4535
Aveiro	654263
Beja	182343
Braga	685300
Bragança	128976
Castelo Branco	207121
Coimbra	472578
Évora	147496
Faro	616070
Guarda	158951
Leiria	490644
Lisboa	2199143
Portalegre	120422
Porto	1667375
Santarém	513388
Setúbal	818264
Viana do Castelo	221168
Vila Real	195912
Viseu	369967

(b) Número de ocorrências por Concelho

```
SELECT Distrito,
   Concelho,
   COUNT(id_occurrence) as TOT_OCO
FROM   dimension_location INNER JOIN fact_occurrence_all ON
 fact_occurrence_all.id_occurrence_location=dimension_location.id_location
GROUP BY Distrito,
 Concelho
ORDER BY Distrito,
 Concelho
```

Distrito	Concelho	TOT_OCO
NULL	NULL	4535
Aveiro	Águeda	40908
Aveiro	Albergaria-a-Velha	27019
Aveiro	Anadia	30364
Aveiro	Arouca	10627
Aveiro	Aveiro	77484
Aveiro	Castelo de Paiva	14979
...	...	...
Viseu	Vila Nova de Paiva	5357
Viseu	Viseu	82915
Viseu	Vouzela	9016

(c) Número de ocorrências por Freguesia

```
SELECT Distrito,
   Concelho,
   Freguesia
   COUNT(id_occurrence) as TOT_OCO
FROM   dimension_location INNER JOIN fact_occurrence_all ON
 fact_occurrence_all.id_occurrence_location=dimension_location.id_location
GROUP BY Distrito,
 Concelho,
 Freguesia
ORDER BY Distrito,
 Concelho,
 Freguesia
```

Distrito	Concelho	Freguesia	TOT_OCO
NULL	NULL	NULL	4535
Aveiro	Águeda	Agadão	299
Aveiro	Águeda	Aguada de Baixo	1046
Aveiro	Águeda	Aguada de Cima	2736
Aveiro	Águeda	Águeda	9306
...	...	...	...
Viseu	Vouzela	Ventosa	660
Viseu	Vouzela	Vouzela	1489

3. **Ocorrências/Prioridade**

   (a) Número de ocorrências por Prioridade

```
 SELECT priority,
 COUNT (id_occurrence) as TOT_OCO
 FROM   dimension_emergency_type
INNER JOIN fact_occurrence_all
ON fact_occurrence_all.id_emergency_type =
dimension_emergency_type.id_emergency_type
GROUP BY priority
ORDER BY priority
```

priority	TOT_OCO
0	33
1	1180367
2	2
3	6927740
4	8825
5	849445
6	25279
7	631
8	861570
9	24

4. **Ocorrências/Tipo de Ocorrência**

   (a) Número de ocorrências por Tipo de Ocorrência

```
SELECT type,
COUNT (id_occurrence) as TOT_OCO
FROM   dimension_emergency_type
INNER JOIN fact_occurrence_all
ON fact_occurrence_all.id_emergency_type =
dimension_emergency_type.id_emergency_type
GROUP BY type
ORDER BY type
```

TYPE	TOT_OCO
ACD	348695
AEC	1271751
AFO	4083
AGR	130848
ALR	39363
CAP	16784
...	...
xChamadaFalsa	2871
xNaoOcorrencia	164546
zTST	822

5. **Ocorrências/Destino de Ocorrência**

   (a) Número de ocorrências por Destino de Ocorrência

```
SELECT destination_name,
COUNT (id_occurrence) as TOT_OCO
FROM   dimension_destination INNER JOIN fact_occurrence_w_units
ON dimension_destination.id_destination =
fact_occurrence_w_units.id_destination
GROUP BY destination_name
```

Destination	TOT_OCO
CENTRO DIAGNOSTICO PNEUMOLOGICO CDP COIMBRA ANTIGO	4
CENTRO HOSPITALAR ALTO MINHO	133710
CENTRO HOSPITALAR ALTO MINHO (HOSPITAIS# 152103)	3695
CENTRO HOSPITALAR ALTO MINHO - URGREF	977
CENTRO HOSPITALAR CALDAS RAINHA	53683
...	...
URGENCIA PEDIATRICA HOSPITAL FARO	15
URGENCIA PEDIATRICA HOSPITAL FARO (HOSPITAIS# 151987)	65
URGENCIA PEDIATRICA HOSPITAL FARO - URGREF	17

6. **Tempos médios/máximos** (em segundos)

   (a) Tempo médio até ao acionamento

```
SELECT AVG(time_to_activation) as AVG_TIME
 FROM fact_occurrence_w_units
```

AVG_TIME
253.3319612222835

(b) Tempo máximo até ao acionamento

```
SELECT MAX(time_to_activation)
FROM fact_occurrence_w_units
```

MAX_TIME
3377425.0

(c) Tempo médio até à chegada do meio

```
SELECT AVG(time_to_arrive)
FROM fact_occurrence_complete
```

AVG_TIME
1156.1025983754646

(d) Tempo máximo até à chegada do meio

```
SELECT MAX(time_to_arrive)
FROM fact_occurrence_complete
```

MAX_TIME
37593.0

(e) Tempo médio até à saída do meio

```
SELECT AVG(time_to_leave)
FROM fact_occurrence_complete
```

AVG_TIME
2567.7593601739104

(f) Tempo máximo até à saída do meio

```
SELECT MAX(time_to_leave)
FROM fact_occurrence_complete
```

MAX_TIME
43308.0

(g) Tempo médio de chegada do meio ao destino

```
SELECT AVG(time_to_destination)
FROM fact_occurrence_complete
```

AVG_TIME
3208.7351390255726

(h) Tempo máximo de chegada ao destino do meio

```
SELECT MAX(time_to_destination)
FROM fact_occurrence_complete
```

MAX_TIME
43311.0

Como é possível comprovar pelos resultados apresentados anteriormente, o Modelo Multimensional desenvolvido para os Data Marts que utilizam apenas dados do SIADEM permite responder às consultas definidas no Levantamento de Requisitos.

5.1.2 Validação das consultas com dados do SIADEM e dados externos

Para validar as consultas definidas na Secção 4.1.3, são utilizados dois métodos distintos, visto que é necessário utilizar o programa representado na Figura D.1, para calcular distâncias entre dois pontos através das coordenadas geográficas. Os dois métodos são: (i) transformações do Pentaho Data Integration que geram um ficheiro em formato *CSV* para mostrar a variação do número de ocorrências ao longo do tempo num local definido; (ii) consultas em *Transact-SQL* para mostrar os tipos de ocorrências mais comuns em cada Data Mart.

1. **Ocorrências** / **Jogos de Futebol**

   (a) Variação do número de ocorrências ao longo do tempo no local em que se realiza o jogo

   A transformação desenvolvida está representada na Figura 5.1. O objetivo da transformação passa por escrever num ficheiro em formato *CSV* as ocorrências ao longo do tempo numa zona próxima a um estádio num intervalo de horas em que tenha decorrido um jogo. O jogo que serviu de exemplo foi um jogo realizado no Estádio da Luz, em Lisboa, entre o Sport Lisboa e Benfica e o Moreirense Futebol Clube no dia 2 de Novembro de 2018 (sexta feira) pelas 20 horas e 30 minutos. Foram inseridas no ficheiro as ocorrências à sexta feira, num raio de 2 km do Estádio da Luz, que tiveram início entre as 19h:30m e as 23h:30m, entre o dia 1 de Setembro de 2018 e o dia 1 de Março de 2019. A descrição dos *steps* está apresentada de seguida:

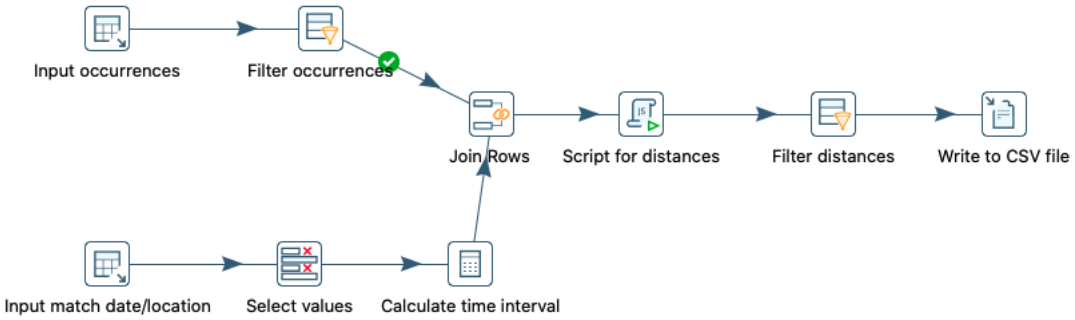

Figura 5.1: Transformação para inserir num ficheiro *CSV* as ocorrências ao longo do tempo numa zona próxima a um estádio num intervalo de tempo em que tenha decorrido um jogo

Input occurrences Extrai do Data Warehouse dados relativos a todas as ocorrências

Filter occurrences Exclui do fluxo as ocorrências que não decorreram entre o dia 2 de Novembro de 2018 e o dia 2 de Novembro de 2019.

Input match date/location Extrai do Data Warehouse a hora de início do jogo e o localização do estádio.

Select Values Atribui um nome diferente à data da realização do jogo e à localização para que seja mais facilitar a sua identificação no fluxo.

Join Rows Faz o produto cartesiano entre o fluxo que provém do step "Select Values" e do step "Filter occurrences", com a condição de que a hora de início da ocorrência tem de estar no intervalo de tempo definido para o jogo.

Script for distances Executa um programa, que a partir das coordenadas geográficas (latitude e latitude) do local da ocorrência e do estádio, calcula a distância em quilómetros entre os dois locais.

Filter distances Filtra as ocorrências, mantendo no fluxo apenas as que estejam a uma distância inferior a 2 quilómetros do estádio onde se realiza o jogo.

Write to CSV file Insere no ficheiro CSV as ocorrências obtidas.

A partir do ficheiro *CSV* foi gerado um gráfico, representado na Figura 5.2, que mostra a variação do número de ocorrências no local onde se realizou o encontro. O intervalo de tempo definido para a amostra presente no gráfico iniciou-se no dia 7 de Setembro de 2018 e terminou no dia 14 de Dezembro de 2018. A barra a laranja representa o dia em que se realizou o jogo.

O gráfico mostra que no dia em que se realizou o encontro, houve um claro aumento do número de ocorrências na zona do Estádio da Luz em relação à normalidade.

(b) Tipo de ocorrências em eventos de futebol

```
SELECT dimension_emergency_type.type,
dimension_emergency_type.description,
```

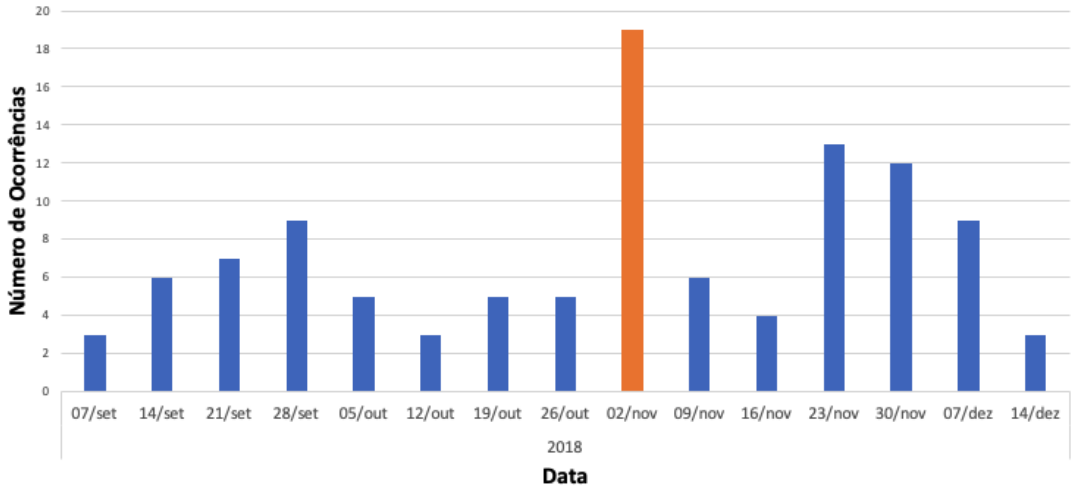

Figura 5.2: Gráfico para mostrar o número de ocorrências ao longo do tempo junto ao Estádio da Luz às sextas feiras entre as 19h:30m e as 23h:30m

```
count (id_occurrence) AS TOT_OCO
FROM fact_football_occurrence INNER JOIN dimension_emergency_type
ON fact_football_occurrence.id_emergency_type =
dimension_emergency_type.id_emergency_type
GROUP BY dimension_emergency_type.type, dimension_emergency_type.description
ORDER BY count (id_occurrence) DESC
```

type	description	TOT_OCO
TRA	Trauma	922
IND	Outros Problemas	814
AEC	Alteração de Estado de Consciência	760
DPN	Dispneia	496
TOX	Intoxicação	307
...	...	...
AFO	Afogamento/Acidente Mergulho	7
xChamadaFalsa	Chamada Falsa	4
RCN	Recém Nascidos/SAVP	3

2. **Ocorrências / Concertos**

(a) Variação do número de ocorrências ao longo do tempo no local em que se realiza o concerto

A transformação desenvolvida no Pentaho Data Integration para mostrar a variação do número de ocorrências ao longo do tempo em locais próximos de concertos, é semelhante à apresentada para as zonas próximas a estádios de futebol. O principal objetivo da transfor-

mação é escrever num ficheiro *CSV* as ocorrências próximas ao local de um concerto no intervalo de horas em que decorreu o concerto. O concerto que serviu de exemplo foi um concerto do cantor Roberto Carlos realizado na Altice Arena, em Lisboa na sexta feira dia 17 de Maio de 2019 pelas 21 horas. As ocorrências inseridas no ficheiro tiveram início entre as 18 horas de sexta feira e as 00 horas de sábado, entre dia 17 de Feveiro de 2019 e dia 18 de Agosto de 2019 num raio de 1,5 quilómetros da Altice Arena.

A partir do ficheiro *CSV* gerado pela transformação foi criado um gráfico, representado na Figura 5.3, que mostra a variação do número de ocorrências no zona onde da Altice Arena. A amostra definida para o gráfico iniciou-se no dia 22 de Março de 2019 e terminou no dia 12 de Julho de 2019. A barra a laranja representa o dia em que se realizou o concerto.

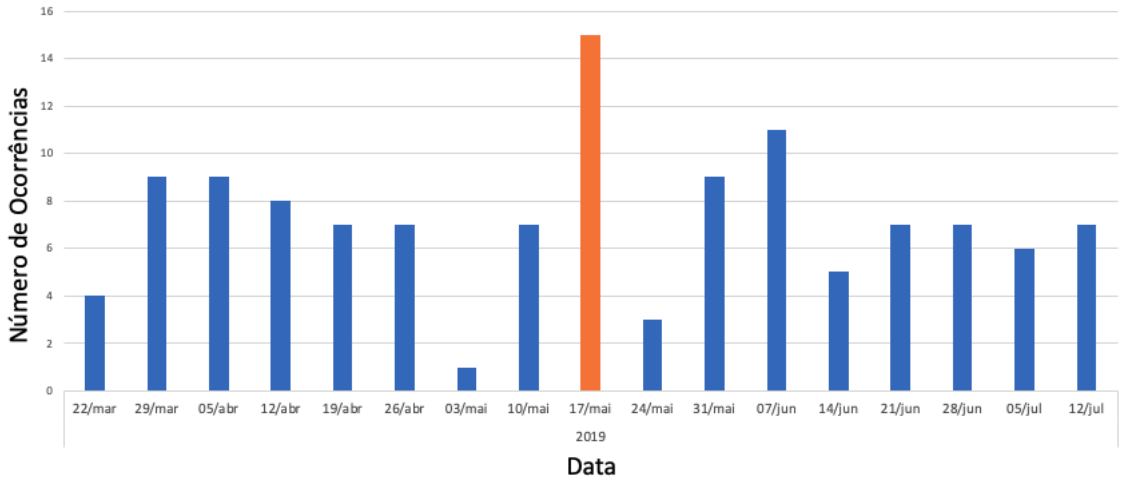

Figura 5.3: Gráfico para mostrar o número de ocorrências ao longo do tempo junto à Altice Arena às sextas feiras entre as 18 horas e as 00 horas de sábado

O gráfico mostra que no dia em que se realizou o concerto, houve um aumento do número de ocorrências na zona da Altice Arena em relação à normalidade.

(b) Tipo de Ocorrência em concertos

```
SELECT dimension_emergency_type.type,
dimension_emergency_type.description,
count (id_occurrence) as TOT_OCO
FROM fact_concerts_occurrence INNER JOIN dimension_emergency_type
ON fact_concerts_occurrence.id_emergency_type =
dimension_emergency_type.id_emergency_type
GROUP BY dimension_emergency_type.type, dimension_emergency_type.description
ORDER BY COUNT (id_occurrence) DESC
```

type	description	TOT_OCO
TRA	Trauma	163
IND	Outros Problemas	124
AEC	Alteração de Estado de Consciência	117
TOX	Intoxicação	65
...	...	...
RCN	Recém Nascidos/SAVP	1
NEG	Negligência/Violência Doméstica/Maus Tratos	1
CAP	CAPIC	1
xChamadaFalsa	Chamada Falsa	1

3. **Ocorrências** / **Festivais**

(a) Variação do número de ocorrências ao longo do tempo no local em que se realiza o festival

A transformação desenvolvida no Pentaho Data Integration para mostrar a variação do número de ocorrências ao longo do tempo em zonas onde decorrem festivais é também semelhante à apresentada para as zonas próximas de estádios de futebol. Esta transformação tem como principal objetivo escrever num ficheiro *CSV* as ocorrências próximas ao local de um festival nos mesmos dias da semana em que decorreu o festival. O festival que foi utilizado para mostrar a variação foi o Festival do Avante de 2018 realizado na Quinta da Atalaia, concelho do Seixal, entre sexta feira dia 7 de Setembro de 2018 e domingo dia 9 de Setembro de 2018. As ocorrências inseridas no ficheiro tiveram início à sexta, sábado ou domingo, entre dia 5 de Agosto de 2018 e dia 10 de Outubro de 2018 num raio de 3 quilómetros da Quinta da Atalaia.

A partir do ficheiro *CSV* gerado na transformação foi criado um gráfico, representado na Figura 5.4, que mostra a variação do número de ocorrências no zona onde da Quinta da Atalaia. A amostra definida para o gráfico iniciou-se no dia 24 de Agosto de 2018 e terminou no dia 28 de Setembro de 2018. As barras a laranja representam os dia em que decorreu o festival.

O gráfico mostra que nos dias em que decorreu o festival, houve um aumento do número de ocorrências na zona da Quinta da Atalaia.

(b) Tipo de ocorrência em festivais

```
SELECT type, description, count(id_occurrence)
FROM fact_festivals_occurence
INNER JOIN dimension_emergency_type ON
fact_festivals_occurence.id_emergency_type =
```

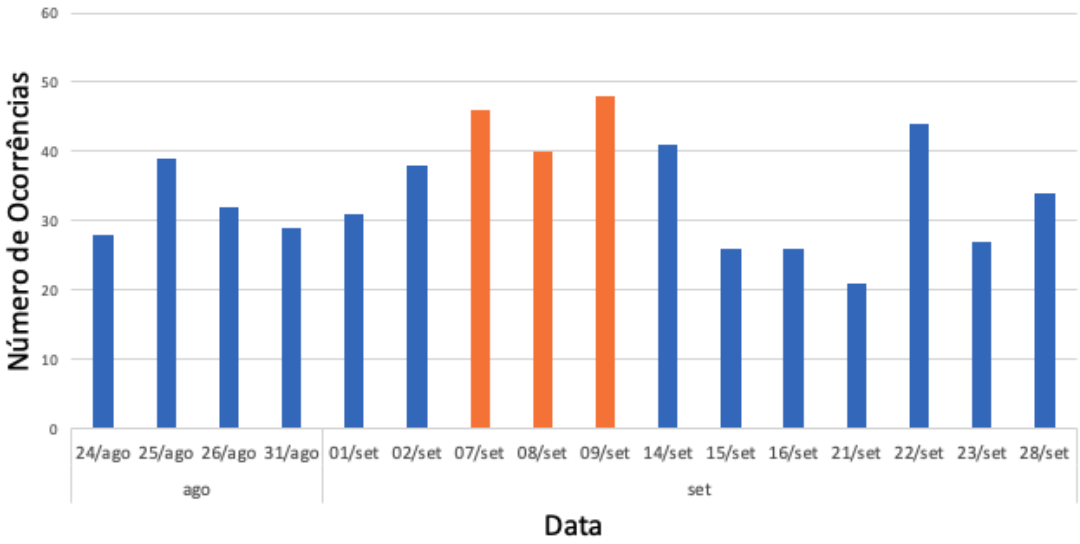

Figura 5.4: Gráfico para mostrar o número de ocorrências ao longo do tempo junto à Quinta da Atalaia às sextas feiras, sabádos e domingos, entre dia 24 de Agosto de 2018 e dia 28 de Setembro de 2018

```
dimension_emergency_type.id_emergency_type
GROUP BY type, description
ORDER BY count(id_occurrence) DESC
```

type	description	TOT_OCO
TRA	Trauma	783
IND	Outros Problemas	762
AEC	Alteração de Estado de Consciência	587
DPN	Dispneia	364
...	...	...
NEG	Negligência/Violência Doméstica/Maus Tratos	1
AFO	Afogamento/Acidente Mergulho	1
x112PT_Notificar	Ocorrência Transferida do 112PT (Notificação)	1

4. **Ocorrências / Meteorologia**

   (a) Variação do número de ocorrências de acordo com a variação das condições meteorológicas numa determinada área

   A transformação utilizada para mostrar variações no número de ocorrências com diferentes condições meteorológicas, volta a ser semelhante à apresentada para as zonas próximas de estádios de futebol. Neste caso a transformação teve como principal objetivo escrever num ficheiro *CSV* as ocorrências próximas de uma estação metereológica num determinado dia da semana. O dia que serviu de base para mostrar variações foi o dia 19 de Janeiro de

2019, um dos dias em que a estação com o identificador (*id_station*) 762 (localizada junto ao aeroporto de Lisboa) registou as temperaturas mais baixas. As ocorrências carregadas para o ficheiro ocorreram à segunda feira, entre o dia 12 de Dezembro de 2018 e o dia 15 de Abril de 2019 num raio de 3,6 quilómetros da estação.

O ficheiro CSV originado pela transformação, representado na Figura 5.5, permitiu a criação de um gráfico que mostra a variação do número de ocorrências junto da estação. Para o gráfico foi definida uma amostra que se iniciou no dia 17 de Dezembro de 2018 e terminou no dia 25 de Fevereiro de 2019. A barra representada com a cor laranja representa o dia utilizado como base.

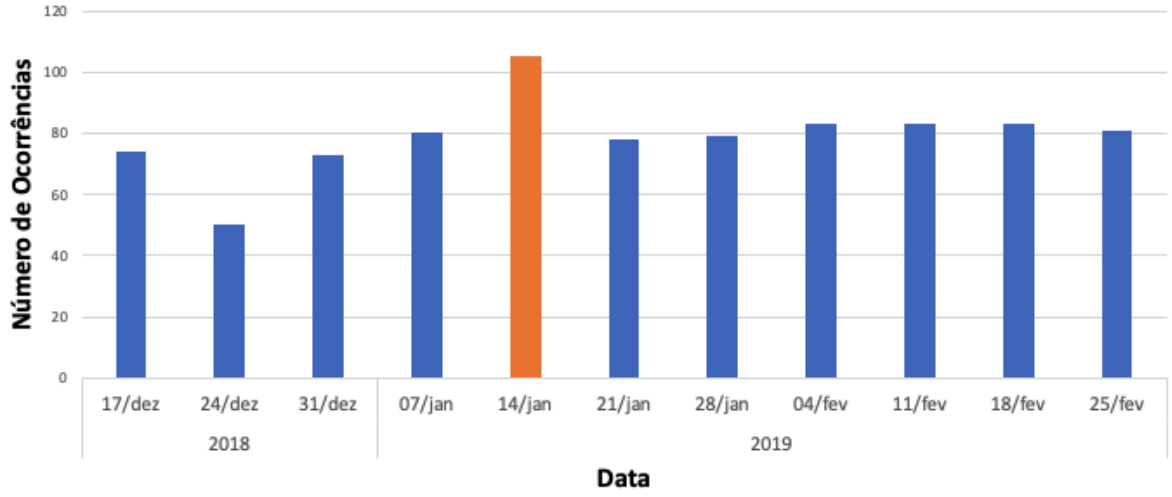

Figura 5.5: Gráfico para mostrar o número de ocorrências ao longo do tempo junto à estação 762 e que ocorreram à segunda feira, entre o dia 17 de Dezembro de 2018 e o dia 25 de Fevereiro de 2019

O gráfico mostra um aumento no número de ocorrências no dia 19 de Janeiro de 2019, de salientar que nesse dia a estação registou uma temperatura média mais baixa do que nos restantes dias representados no gráfico.

(b) Tipo de ocorrências associadas a diferentes condições meteorológicas

Para mostrar os tipos de ocorrências mais comuns associados a diferentes condições meteorológicas foram utilizadas duas consultas. A primeira consulta procurou os tipos de ocorrências mais comuns quando o sensor registava temperaturas inferirores a 10 graus centígrados. A segunda consulta procurou os tipos de ocorrências mais comuns quando o sensor registava temperaturas superiores a 20 graus centígrados.

i.
```
SELECT dimension_emergency_type.type,
dimension_emergency_type.description,
count(id_occurrence) as TOT_OCO
FROM fact_weather_occurrence
INNER JOIN dimension_emergency_type
ON fact_weather_occurrence.id_emergency_type=
```

```
dimension_emergency_type.id_emergency_type
INNER JOIN dimension_weather
ON fact_weather_occurrence.id_weather_measure=
dimension_weather.id_weather_measure
WHERE temperature < 10
GROUP BY dimension_emergency_type.type,
dimension_emergency_type.description
ORDER BY count(id_occurrence) desc
```

type	description	TOT_OCO
TRA	Trauma	14
IND	Outros Problemas	13
DPN	Dispneia	12
DTC	Dor Torácica	11
...	...	...

ii.
```
SELECT dimension_emergency_type.type,
dimension_emergency_type.description,
count(id_occurrence) as TOT_OCO
FROM fact_weather_occurrence
INNER JOIN dimension_emergency_type
ON fact_weather_occurrence.id_emergency_type=
dimension_emergency_type.id_emergency_type
INNER JOIN [D2H_DW].[dbo].[dimension_weather]
ON fact_weather_occurrence.id_weather_measure=
dimension_weather.id_weather_measure
WHERE temperature > 20
GROUP BY dimension_emergency_type.type,
dimension_emergency_type.description
ORDER BY count(id_occurrence) desc
```

type	description	TOT_OCO
AEC	Alteração de Estado de Consciência	38
TRA	Trauma	37
IND	Outros Problemas	28
ACD	Acidente Viação	17
...	...	...

5.2 Desempenho das Consultas

Outro aspeto importante para a validação do Data Warehouse é a comparação do desempenho das consultas da Secção 4.1.2 executadas sobre a base de dados relacional do SIADEM e sobre o Data Warehouse. Na Tabela 5.1, está apresentada a comparação entre o tempo de processamento das consultas a cada uma das bases de dados. A unidade de tempo utilizada foi milissegundos e cada entrada na tabela é o resultado da média de 5 execuções de cada consulta. A lista de consultas em *Transact SQL* executadas sobre a base de dados relacional do SIADEM encontra-se no Anexo E.

Consulta	DB SIADEM (ms)	D2H DW (ms)
1(a)	53377	156
1(b)	55379	117
1(c)	56029	244
1(d)	16657	410
1(e)	8455	161
2(a)	2332	161
2(b)	2595	911
2(c)	2874	990
3(a)	1158	161
4(a)	2374	213
5(a)	17365	187
6(a)	6724	141
6(b)	6766	136
6(c)	4359	41
6(d)	4351	41
6(e)	4286	52
6(f)	4431	34
6(g)	4246	41
6(h)	4487	28

Tabela 5.1: Tabela de comparação dos tempos de execução das consultas listadas na Secção 4.1.2 à base de dados relacional do SIADEM e ao Data Warehouse.

Os resultados apresentados na tabela mostram que o tempo de processamento das consultas realizadas ao Data Warehouse é claramente inferior. Isto acontece por vários motivos, por exemplo: para realizar as consultas quando estas são executadas sobre a base de dados relacional são necessárias várias conversões nos dados.

5.3 Dados do Data Warehouse vs Dados da Base de Dados do SIADEM

O terceiro tipo de validação experimental está relacionado com a necessidade de provar que as ocorrências presentes na base de dados relacional do SIADEM são as mesmas que estão carregadas no Data Warehouse, ou seja, que não existem perdas nem incorreções. Para realizar esta validação, foram realizadas consultas SQL às duas bases de dados que extraem todas as ocorrências, bem como os seus tipos e prioridades. O resultado das consultas foi escrito para ficheiro em formato *txt*. De seguida com o apoio do Pentaho Data Integration foi realizado um *Job*, representado na Figura 5.6, que com-

para o conteúdo dos dois ficheiros. Se o conteúdo dos dois ficheiros for igual o *Job* é finalizado com sucesso, caso os ficheiros sejam diferentes o *Job* é finalizado sem sucesso.

Execution Results

History | Logging | Job metrics | Metrics

Job / Entrada do Job	Coment◆rio	Resultado	Raz◆o	Filename	Nr	Data do Log
compare						
Job: compare	Start of job execution		start			2020/12/22 17:36:01
START	Start of job execution		start			2020/12/22 17:36:01
START	Job execution finished	Success			0	2020/12/22 17:36:01
File Compare	Start of job execution		Followed unconditional link			2020/12/22 17:36:01
File Compare	Job execution finished	Success			0	2020/12/22 17:42:18
Job: compare	Job execution finished	Success	finished		0	2020/12/22 17:42:18

Figura 5.6: *Job para comparar os ficheiros obtidos pelas consultas que extraem todas as ocorrências, seus respetivos tipos e prioridades da Base de Dados do SIADEM e do Data Warehouse*

A Figura 5.6 mostra que o *Job* foi finalizado com sucesso, pelo que podemos concluir que os ficheiros são iguais e consequentemente podemos conlcuir que as ocorrências presentes em ambas as bases de dados são as mesmas, e que têm os seus atributos tipo e prioridade corretamente associados.

Capítulo 6

Conclusões

Como foi referido no Capítulo 1, a produtividade das operações do INEM é muito importante para Portugal, pelo que quaisquer melhorias na produtividade podem salvar a vida de um cidadão. O objetivo do projeto Data2Help passa por melhorar o desempenho dos processos do INEM.

O foco desta tese foi a tarefa de Integração e Limpeza de Dados no projeto Data2Help. O principal objetivo da tarefa de integração de dados passou por integrar dados históricos do SIADEM com fontes de dados externas, num repositório que permitisse responder a conjunto definido de consultas. Para concretizar a tarefa, optou-se por uma integração materializada de dados que culminou na criação de um Data Warehouse. Na próxima etapa do projeto serão aplicados modelos preditivos e de correlação de dados. Com base nos modelos preditivos e nas correlações detetadas, serão posteriormente desenvolvidos modelos e algoritmos eficientes para planeamento e escalonamento de recursos do INEM.

O documento iniciou-se com uma introdução ao projeto (Capítulo 1), com ênfase para a tarefa de Integração de Dados. De seguida, são apresentados os conceitos básicos de integração de dados e do domínio das emergências médicas (Capítulo 2). Segue-se uma análise a trabalhos realizados relacionados com o tema do projeto, bem como uma análise a ferramentas de software para efetuar a integração materializada de dados (Capítulo 3). Por fim, é apresentada a solução efetuada para integração de dados (Capítulo 4) e a sua correspondente validação experimental (Capítulo 5).

É possível concluir que os principais objetivos da tarefa de integração de dados foram atingidos com sucesso. Visto que o Data Warehouse permite responder às consultas identificadas em colaboração com os responsáveis pela próxima etapa do projeto. Em relação aos dados carregados no projeto, também é possível verificar que não houve perdas de dados, que foram executadas transformações para melhorar a qualidade dos mesmos e foi implementado um mecanismo que permite realizar a atualização periódica dos dados. Por sua vez, em relação ao desempenho do Data Warehouse em relação à base de dados relacional do SIADEM, também é possível concluir que foram alcançadas melhorias significativas no que toca às consultas que utilizam apenas dados do SIADEM.

Na Secção 6.1 estão apresentadas propostas de trabalho futuro que pode ser realizado no Data Warehouse.

6.1 Trabalho Futuro

A integração de dados realizada ainda tem alguns aspetos que podem ser melhorados, sobretudo em relação aos dados externos. Em relação aos dados de eventos musicais, a *API* utilizada não permite que os dados sejam extraídos de forma automática, pelo que foi extraído apenas um conjunto limitado de dados sobre eventos, que decorreram no mesmo intervalo de tempo em que existem dados do SIADEM. Utilizar uma API que permitisse adicionar automaticamente novos dados sobre eventos musicais em Portugal seria interessante.

O conjunto de dados meteorológicos extraído também é bastante limitado, apenas existem dados de sensores localizados no Concelho de Lisboa e no intervalo de tempo de 1 ano. Para uma melhor deteção de correlações das ocorrências com este tipo de dados, devem futuramente ser adicionados dados meteorológicos de outras zonas de Portugal e num maior intervalo de tempo, pois tal permitiria um melhor correlacionamento entre os dados meteorológicos e os dados das ocorrências. Uma API interessante a ser utilizada, seria a *meteo|Técnico*[1] fundado pelo Grupo de Previsão Numérica do Tempo (GPNT) da Secção de Ambiente e Energia do Instituto Superior Técnico.

Ainda em relação a dados externos, outras fontes de dados, como dados sobre epidemias, por exemplo integrando dados sobre a pandemia que afeta o mundo atualmente, a *COVID 19*[2]. Infelizmente, o conjunto de dados do SIADEM que está inserido no Data Warehouse é anterior ao registo da primeira pessoa infetada com o vírus *SARS-CoV2* em Portugal. Decerto que os dados da pandemia trariam informações muito interessantes para correlacionar com as ocorrências. Ainda sobre dados de epidemias, integrar dados sobre a Gripe trará informações relevantes para o INEM. Infelizmente, os dados da Gripe disponibilizados pela API da Direção Geral de Saúde [3] quanto à localização não têm uma granularidade apropriada para o projeto, visto que são recolhidos por Associação Regional de Saúde.

Outra proposta importante para efetuar numa camada superior ao Data Warehouse, seria o desenvolvimento de uma aplicação ou portal com uma abordagem "amiga do utilizador". Esta aplicação deveria funcionar como um sistema de apoio à decisão, que estruturasse a informação armazenada no Data Warehouse e a mostrasse de forma clara e rápida ao utilizador. O sistema iria auxiliar a tomada de decisão e iria permitirir que o utilizador analisasse os dados de uma forma intuitiva.

[1] https://meteo.tecnico.ulisboa.pt/
[2] https://covid19.min-saude.pt/category/perguntas-frequentes/
[3] https://transparencia.sns.gov.pt

Bibliografia

[1] L. Aboueljinane, E. Sahin, Z. Jemai, and J. Marty. A simulation study to improve the performance of an emergency medical service: application to the french val-de-marne department. *Simulation modelling practice and theory*, 47:46–59, 2014.

[2] J. Banks, J. S. Carson, B. L. Nelson, D. M. Nicol, et al. *Discrete-event system simulation*, volume 3. Prentice hall Upper Saddle River, NJ, 1996.

[3] A. Bogomolov, B. Lepri, J. Staiano, N. Oliver, F. Pianesi, and A. Pentland. Once upon a crime: towards crime prediction from demographics and mobile data. In *International Conference on Multimodal Interaction*, pages 427–434. ACM, 2014.

[4] F. Calabrese, L. Ferrari, and V. D. Blondel. Urban sensing using mobile phone network data: a survey of research. *Acm computing surveys (csur)*, 47(2):25, 2015.

[5] S. Clark, M. Damiani, H. Dorning, M. Halter, and A. Porter. Patient-level data linkage across ambulance services and acute trusts: assessing the potential for improving patient care. *International Journal of Population Data Science*, 1(1), 2017.

[6] A. Doan, A. Halevy, and Z. Ives. *Principles of data integration*. Elsevier, 2012.

[7] A. Drake, A. Pollitt, E. Sklar, L. Smith, S. Parsons, and E. Schneider. Data for ambulance dispatch. Technical report, Policy Institute at King's College London, 2018.

[8] E. T. Ehtisham Zaidi, Nick Heudecker. Gartner magic quadrant for data integration tools, 2019.

[9] D. R. Holleman, R. L. Bowling, and C. Gathy. Predicting daily visits to a waik-in clinic and emergency department using calendar and weather data. *Journal of General Internal Medicine*, 11(4):237–239, 1996.

[10] H. J. Kam, J. O. Sung, and R. W. Park. Prediction of daily patient numbers for a regional emergency medical center using time series analysis. *Healthcare informatics research*, 16(3):158–165, 2010.

[11] M. Mahmood, J. Thornes, F. Pope, P. Fisher, and S. Vardoulakis. Impact of air temperature on london ambulance call-out incidents and response times. *Climate*, 5(3):61, 2017.

[12] A. P. G. Martins. Emergência pré-hospitalar. 2011.

[13] E. Park, J. H. Kim, H. S. Nam, and H.-J. Chang. Requirement analysis and implementation of smart emergency medical services. *IEEE Access*, 6:42022–42029, 2018.

[14] K. Peleg and J. S. Pliskin. A geographic information system simulation model of ems: reducing ambulance response time. *The American journal of emergency medicine*, 22(3):164–170, 2004.

[15] C. J. Pérez-González, M. Colebrook, J. L. Roda-García, and C. B. Rosa-Remedios. Developing a data analytics platform to support decision making in emergency and security management. *Expert Systems with Applications*, 120:167–184, 2019.

[16] M. Poulton and G. Roussos. Towards smarter metropolitan emergency response. In *24th IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications, PIMRC 2013, London, United Kingdom, September 8-11, 2013*, pages 2576–2580, 2013. doi: 10.1109/PIMRC.2013.6666581. URL https://doi.org/10.1109/PIMRC.2013.6666581.

[17] M. Reuter and W. Michalk. Towards the dynamic relocation of ambulances in germany: The risk of being too late. In *2012 Annual SRII Global Conference, San Jose, CA, USA, July 24-27, 2012*, pages 642–649, 2012. doi: 10.1109/SRII.2012.78. URL https://doi.org/10.1109/SRII.2012.78.

[18] S. Z. Sajani, E. Alessandrini, S. Marchesi, and P. Lauriola. Are day-to-day variations of airborne particles associated with emergency ambulance dispatches? *International journal of occupational and environmental health*, 20(1):71–76, 2014.

[19] M. Y. Santos and R. Isabel. *Business Intelligence da informação ao conhecimento*. Lisboa: FCA, 2017.

[20] V. Schmid and K. F. Doerner. Ambulance location and relocation problems with time-dependent travel times. *European Journal of Operational Research*, 207(3):1293–1303, 2010. doi: 10.1016/j.ejor.2010.06.033. URL https://doi.org/10.1016/j.ejor.2010.06.033.

[21] H. A. Snooks, R. Anthony, R. Chatters, J. Dale, R. Fothergill, S. Gaze, M. Halter, I. Humphreys, M. Koniotou, P. Logan, et al. Support and assessment for fall emergency referrals (safer) 2: a cluster randomised trial and systematic review of clinical effectiveness and cost-effectiveness of new protocols for emergency ambulance paramedics to assess older people following a fall with referral to community-based care when appropriate. *Health technology assessment*, 21(13), 2017.

[22] J. E. Thornes, P. A. Fisher, T. Rayment-Bishop, and C. Smith. Ambulance call-outs and response times in birmingham and the impact of extreme weather and climate change. *Emerg Med J*, 31(3):220–228, 2014.

[23] A. Vaisman and E. Zimányi. *Data warehouse systems*. Springer, 2014.

[24] O. Zorab, M. Robinson, and R. Endacott. Are prehospital treatment or conveyance decisions affected by an ambulance crew's ability to access a patient's health information? *BMC emergency medicine*, 15(1):26, 2015.

Apêndice A

Trabalho Relacionado

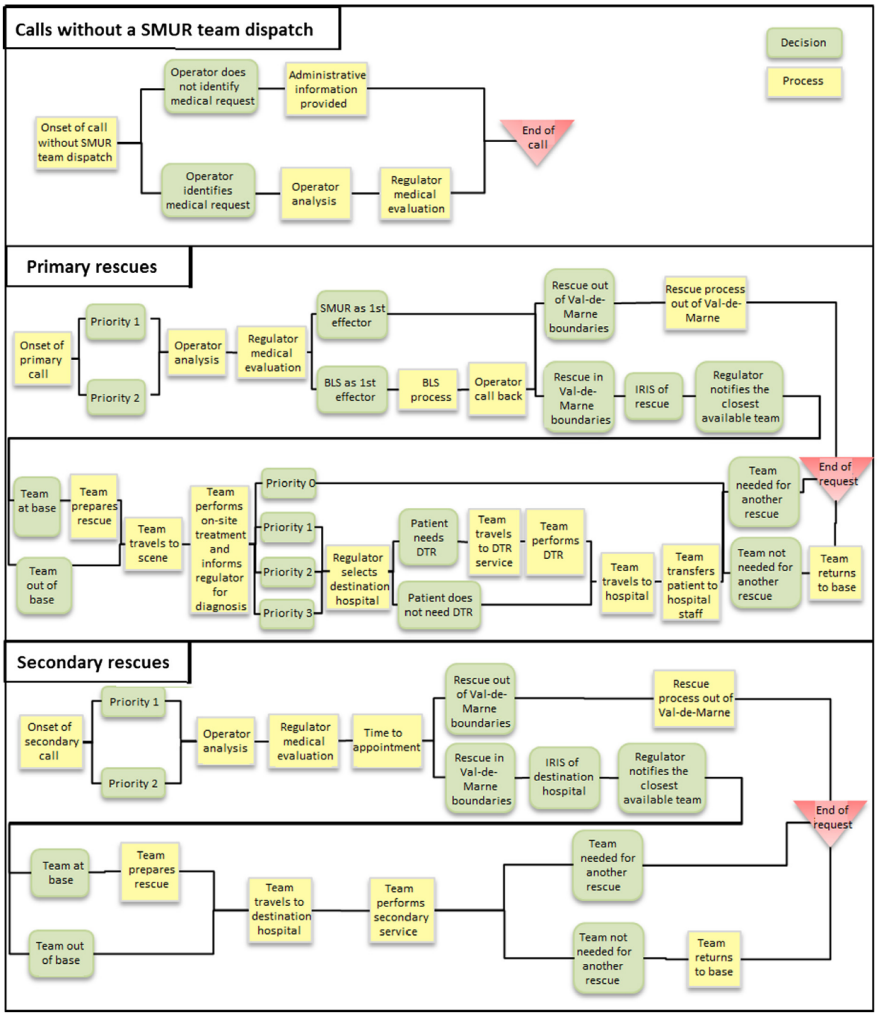

Figura A.1: Modelo Conceptual dos Processo do SAMU 94 [1]

Apêndice B

Modelo Multidimensional

B.1 Programa em *Transact-SQL* para criar o Data Warehouse

```
USE D2H_DW;

CREATE TABLE [dimension_time] (
  [id_time] datetime PRIMARY KEY NOT NULL,
  [year] integer,
  [quarter] integer,
  [month] integer,
  [day] integer,
  [hour] integer,
  [minute] integer,
  [second] integer,
  [dayofweek] integer,
  [weekday] BIT
)
GO

CREATE TABLE [dimension_location] (
  [id_location] integer PRIMARY KEY NOT NULL,
  [location_name] VARCHAR(450),
  [city] VARCHAR(255),
  [locality] VARCHAR(255),
  [street] VARCHAR(255),
  [reference_points] VARCHAR(255),
  [door_nr] VARCHAR(255),
  [ZIP] VARCHAR(255),
```

```
  [latitude] FLOAT(53),
  [longitude] FLOAT(53),
  [local_desc] VARCHAR(255),
  [DICOFRE] VARCHAR(255),
  [DISTRITO] VARCHAR(255),
  [CONCELHO] VARCHAR(255),
  [FREGUESIA] VARCHAR(450)
)
GO

CREATE TABLE [dimension_unit] (
  [id_unit] integer PRIMARY KEY NOT NULL,
  [cod_unit] VARCHAR(255),
  [type_unit] VARCHAR(255),
  [station] VARCHAR(255),
  [d_group] VARCHAR(255)
)
GO

CREATE TABLE [dimension_destination] (
  [id_destination] integer PRIMARY KEY NOT NULL,
  [destination_name] VARCHAR(255),
)
GO

CREATE TABLE [dimension_emergency_type] (
  [id_emergency_type] integer PRIMARY KEY,
  [priority] integer,
  [type] VARCHAR(255),
  [description] VARCHAR(255)
)
GO

CREATE TABLE [fact_occurrence_all] (
  [id_occurrence] integer PRIMARY KEY NOT NULL,
  [id_start_time] datetime NOT NULL,
  [id_occurrence_location] integer NOT NULL,
  [id_emergency_type] integer,
  FOREIGN KEY ([id_start_time])
```

```
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_occurrence_location])
  REFERENCES [dimension_location] ([id_location]),
  FOREIGN KEY ([id_emergency_type])
  REFERENCES [dimension_emergency_type] ([id_emergency_type])
)
GO

CREATE TABLE [fact_occurrence_w_units] (
  [id_occurrence] integer PRIMARY KEY NOT NULL,
  [id_start_time] datetime NOT NULL,
  [id_occurrence_location] integer NOT NULL,
  [id_emergency_type] integer NOT NULL,
  [id_destination] integer NOT NULL,
  [id_activation_time] datetime,
  [time_to_activation] float(53),
  FOREIGN KEY ([id_start_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_occurrence_location])
  REFERENCES [dimension_location] ([id_location]),
  FOREIGN KEY ([id_emergency_type])
  REFERENCES [dimension_emergency_type] ([id_emergency_type]),
  FOREIGN KEY ([id_destination])
  REFERENCES [dimension_destination] ([id_destination]),
  FOREIGN KEY ([id_activation_time])
  REFERENCES [dimension_time] ([id_time])
)
GO

CREATE TABLE [fact_occurrence_complete] (
  [id_occurrence] integer PRIMARY KEY NOT NULL,
  [id_start_time] datetime NOT NULL,
  [id_occurrence_location] integer NOT NULL,
  [id_emergency_type] integer NOT NULL,
  [id_destination] integer NOT NULL,
  [id_activation_time] datetime,
  [id_arrive_time] datetime,
  [id_leave_time] datetime,
  [id_destination_time] datetime,
```

```
  [time_to_activation] float(53),
  [time_to_arrive] float(53),
  [time_to_leave] float(53),
  [time_to_destination] float(53)
  FOREIGN KEY ([id_start_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_occurrence_location])
  REFERENCES [dimension_location] ([id_location]),
  FOREIGN KEY ([id_emergency_type])
  REFERENCES [dimension_emergency_type] ([id_emergency_type]),
  FOREIGN KEY ([id_destination])
  REFERENCES [dimension_destination] ([id_destination]),
  FOREIGN KEY ([id_activation_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_arrive_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_leave_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_destination_time])
  REFERENCES [dimension_time] ([id_time])
)
GO

CREATE TABLE [dimension_group_unit_w_units] (
  [id_occurrence] integer NOT NULL,
  [id_unit] integer NOT NULL,
  PRIMARY KEY ([id_occurrence], [id_unit]),
  FOREIGN KEY ([id_occurrence])
  REFERENCES [fact_occurrence_w_units] ([id_occurrence]),
  FOREIGN KEY ([id_unit])
  REFERENCES [dimension_unit] ([id_unit])
)
GO

CREATE TABLE [dimension_group_unit_complete] (
  [id_occurrence] integer NOT NULL,
  [id_unit] integer NOT NULL,
  PRIMARY KEY ([id_occurrence], [id_unit]),
  FOREIGN KEY ([id_occurrence])
```

```
  REFERENCES [fact_occurrence_complete] ([id_occurrence]),
  FOREIGN KEY ([id_unit])
  REFERENCES [dimension_unit] ([id_unit])
)
GO

CREATE TABLE [dimension_competition] (
  [competition_id] integer PRIMARY KEY NOT NULL,
  [competition_code] integer,
  [competition_name] VARCHAR(255)
)
GO

CREATE TABLE [dimension_teams] (
  [team_id] integer PRIMARY KEY NOT NULL,
  [team_code] integer,
  [team_name] VARCHAR(255)
)
GO

CREATE TABLE [dimension_football] (
  [match_id] integer PRIMARY KEY NOT NULL,
  [match_code] VARCHAR(255),
  [id_date_match] datetime,
  [competition_id] integer,
  [location_stadium_id] integer,
  [homeTeam_id] integer,
  [awayTeam_id] integer,
  [score] VARCHAR(255),
  FOREIGN KEY ([id_date_match])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([competition_id])
  REFERENCES [dimension_competition] ([competition_id]),
  FOREIGN KEY ([location_stadium_id])
  REFERENCES [dimension_location] ([id_location]),
  FOREIGN KEY ([homeTeam_id])
  REFERENCES [dimension_teams] ([team_id]),
  FOREIGN KEY ([awayTeam_id])
  REFERENCES [dimension_teams] ([team_id])
```

```
)
GO

CREATE TABLE [fact_football_occurrence] (
  [id_occurrence] integer PRIMARY KEY NOT NULL,
  [id_emergency_type] integer NOT NULL,
  [id_event_football] integer NOT NULL,
  [id_occurrence_start_time] datetime NOT NULL,
  [id_occurrence_location] integer,
  FOREIGN KEY ([id_emergency_type])
  REFERENCES [dimension_emergency_type] ([id_emergency_type]),
  FOREIGN KEY ([id_event_football])
  REFERENCES [dimension_football] ([match_id]),
  FOREIGN KEY ([id_occurrence_start_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_occurrence_location])
  REFERENCES [dimension_location] ([id_location])
)
GO

CREATE TABLE [dimension_concerts] (
  [event_id] integer PRIMARY KEY NOT NULL,
  [event_code] integer,
  [event_name] VARCHAR(255),
  [event_id_start_time] datetime,
  [event_id_location] integer,
  [event_popularity] VARCHAR(255),
  FOREIGN KEY ([event_id_start_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([event_id_location])
  REFERENCES [dimension_location] ([id_location])
)
GO

CREATE TABLE [fact_concerts_occurrence] (
  [id_occurrence] integer PRIMARY KEY NOT NULL,
  [id_emergency_type] integer NOT NULL,
  [id_event_concert] integer NOT NULL,
  [id_occurrence_start_time] datetime NOT NULL,
```

```
  [id_occurrence_location] integer,
  FOREIGN KEY ([id_emergency_type])
  REFERENCES [dimension_emergency_type] ([id_emergency_type]),
  FOREIGN KEY ([id_event_concert])
  REFERENCES [dimension_concerts] ([event_id]),
  FOREIGN KEY ([id_occurrence_start_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_occurrence_location])
  REFERENCES [dimension_location] ([id_location])
)
GO

CREATE TABLE [dimension_festivals] (
  [event_id] integer PRIMARY KEY NOT NULL,
  [event_code] integer,
  [event_name] VARCHAR(255),
  [event_id_start_time] datetime,
  [event_id_end_time] datetime,
  [event_id_location] integer,
  [event_popularity] VARCHAR(255),
  FOREIGN KEY ([event_id_start_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([event_id_end_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([event_id_location])
  REFERENCES [dimension_location] ([id_location])
)
GO

CREATE TABLE [fact_festivals_occurence] (
  [id_occurrence] integer PRIMARY KEY NOT NULL,
  [id_emergency_type] integer NOT NULL,
  [id_event_festival] integer NOT NULL,
  [id_occurrence_start_time] datetime NOT NULL,
  [id_occurrence_location] integer,
  FOREIGN KEY ([id_emergency_type])
  REFERENCES [dimension_emergency_type] ([id_emergency_type]),
  FOREIGN KEY ([id_event_festival])
  REFERENCES [dimension_festivals] ([event_id]),
```

```
  FOREIGN KEY ([id_occurrence_start_time])
  REFERENCES [dimension_time] ([id_time]),
  FOREIGN KEY ([id_occurrence_location])
  REFERENCES [dimension_location] ([id_location])
)
GO

CREATE TABLE [dimension_weather] (
  [id_weather_measure] int PRIMARY KEY NOT NULL,
  [id_time_measure] datetime NOT NULL,
  [id_station] int NOT NULL,
  [id_location_station_weather] int,
  [humidity] int,
  [wind_direction_id] int,
  [wind_intensity] FLOAT(53),
  [wind_intensity_km] FLOAT(53),
  [accumulated_precipitation] int,
  [pressure] float,
  [radiation] int,
  [temperature] int,
  FOREIGN KEY ([id_location_station_weather])
  REFERENCES [dimension_location] ([id_location]),
  FOREIGN KEY ([id_time_measure])
  REFERENCES [dimension_time] ([id_time])
)
GO

CREATE TABLE [fact_weather_occurrence] (
  [id_occurrence] integer PRIMARY KEY NOT NULL,
  [id_emergency_type] integer NOT NULL,
  [id_weather_measure] integer NOT NULL,
  [id_occurrence_start_time] datetime NOT NULL,
  [id_occurrence_location] integer NOT NULL,
  FOREIGN KEY ([id_emergency_type])
  REFERENCES [dimension_emergency_type] ([id_emergency_type]),
  FOREIGN KEY ([id_occurrence_location])
  REFERENCES [dimension_location] ([id_location]),
  FOREIGN KEY ([id_occurrence_start_time])
  REFERENCES [dimension_time] ([id_time]),
```

```
  FOREIGN KEY ([id_weather_measure])
  REFERENCES [dimension_weather] ([id_weather_measure])


)
GO
```

B.2 Representação Gráfica do Modelo Multidimensional

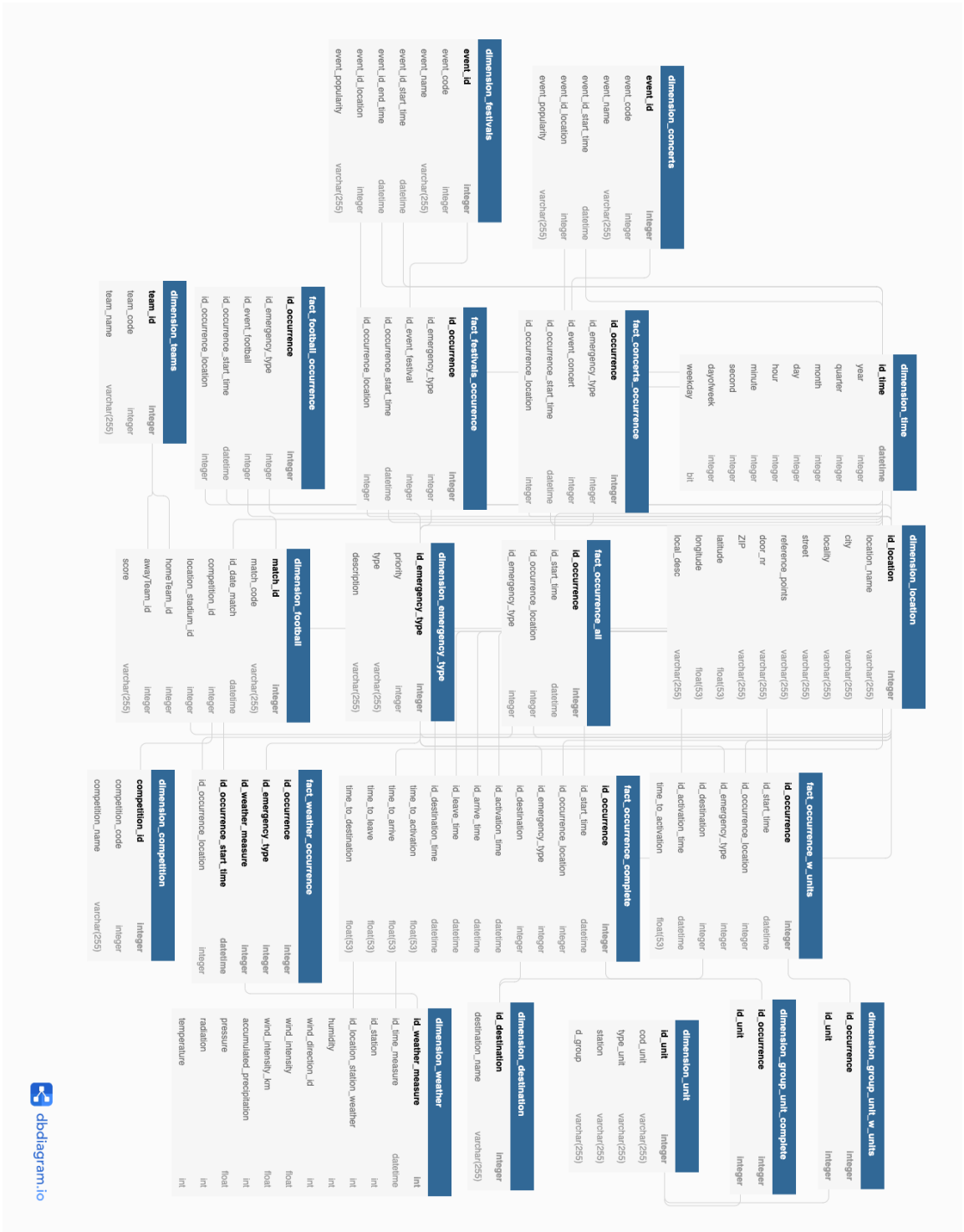

Figura B.1: Representação Gráfica do Modelo Multidimensional do Data Warehouse do Projeto Data2Help

Apêndice C

Transformações Data Staging Area

C.1 Transformação para os dados de Futebol

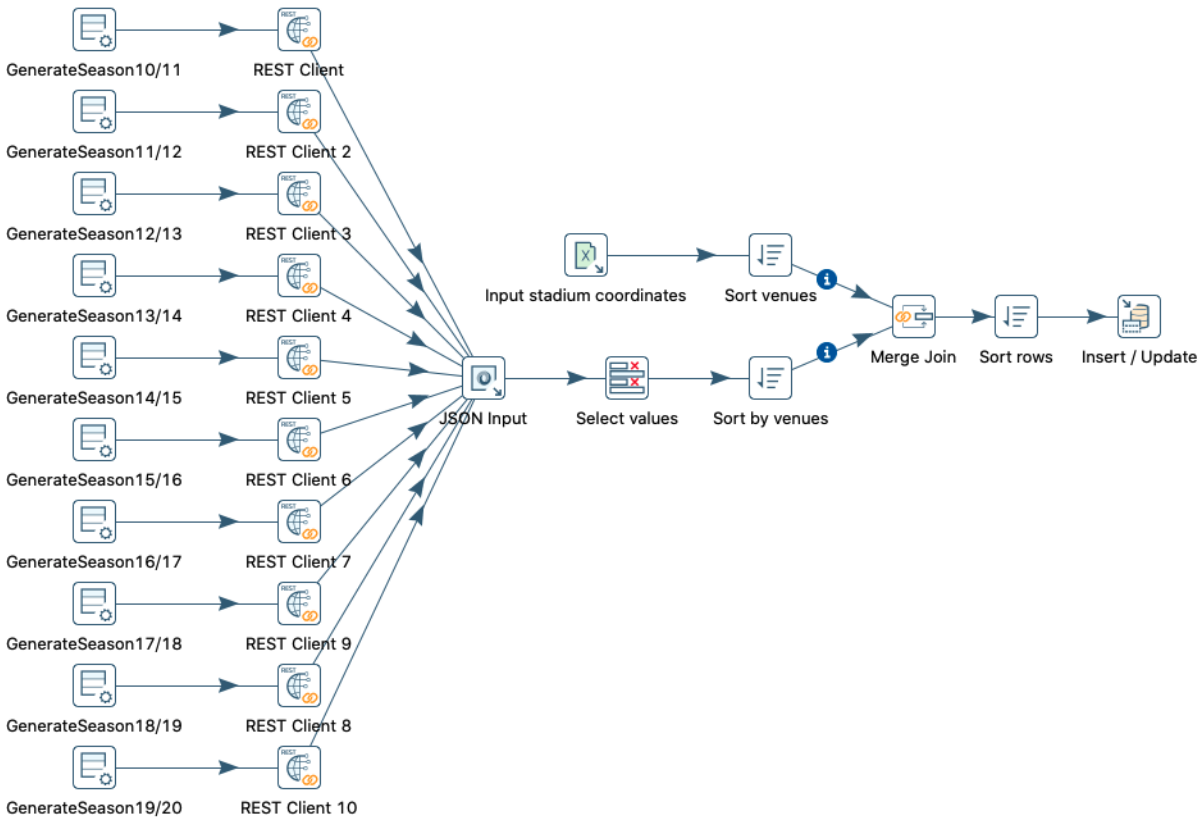

Figura C.1: Transformação para extrair os dados de futebol da *API* e carregá-los na Data Staging Area

C.2 Transformação para os dados de Concertos

Figura C.2: Transformação para carregar o conjunto de dados de concertos na Data Staging Area

C.3 Transformação para os dados de Festivais

Figura C.3: Transformação para carregar o conjunto de dados de festivais na Data Staging Area

Apêndice D

Processo ETL

D.1 Programa para calcular distâncias através de coordenadas geográficas

```
  var R = 6371; // Radius of the earth in km
  var dLat = deg2rad(latitude_jogo-latitude);  // deg2rad below
  var dLon = deg2rad(longitude_jogo-longitude);
  var a =
    Math.sin(dLat/2) * Math.sin(dLat/2) +
    Math.cos(deg2rad(latitude)) * Math.cos(deg2rad(latitude_jogo)) *
    Math.sin(dLon/2) * Math.sin(dLon/2)
    ;
  var c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
  var d = R * c; // Distance in km


function deg2rad(deg) {
  return deg * (Math.PI/180)
}
```

Figura D.1: Programa utilizado para calcular a distância entre as ocorrências e os estádios onde se realizam os jogos de futebol

D.2 Data Mart Todas as Ocorrências

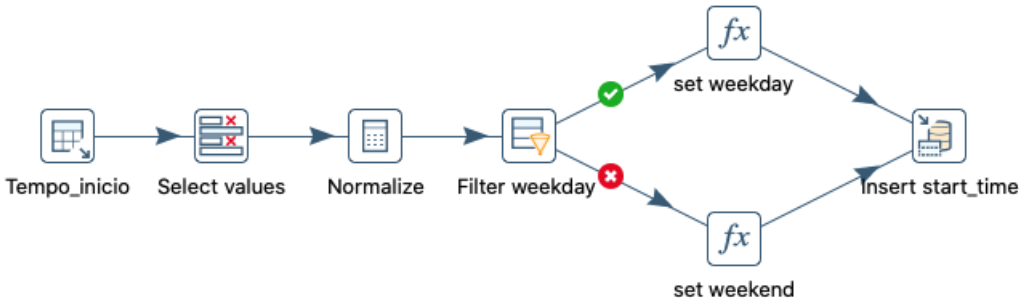

Figura D.2: Transformação para carregar os tempos de início das ocorrências na dimensão Tempo

Figura D.3: Transformação para carregar a localização das ocorrências na dimensão Localização

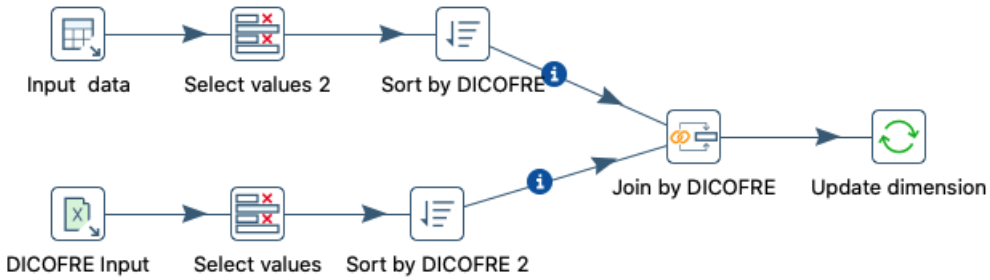

Figura D.4: Transformação para inserir o Distrito, Concelho e Freguesia da localização das ocorrências na dimensão Localização

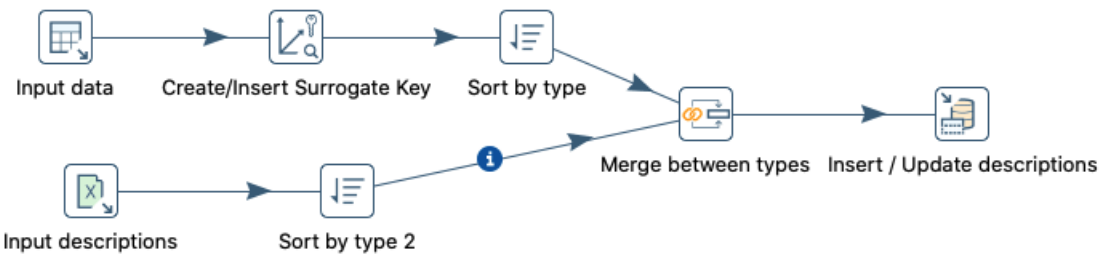

Figura D.5: Transformação para carregar os dados na dimensão Tipo de Emergência

Figura D.6: Transformação para carregar os dados na tabela de factos do Data Mart Todas as Ocorrências

Figura D.7: Job que executa as transformações para carregar os dados no Data Mart Todas as Ocorrências

D.3 Data Mart Ocorrências com Meios

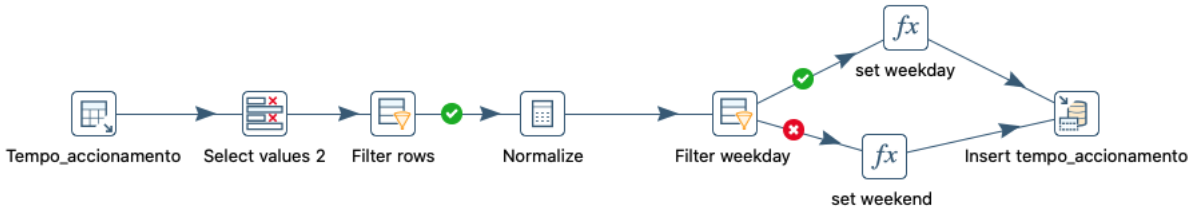

Figura D.8: Transformação para carregar os tempos de acionamento do primeiro meio na dimensão Tempo

Figura D.9: Transformação para carregar unidades na dimensão Unidade

Figura D.10: Transformação para carregar as restantes unidades na dimensão Unidade

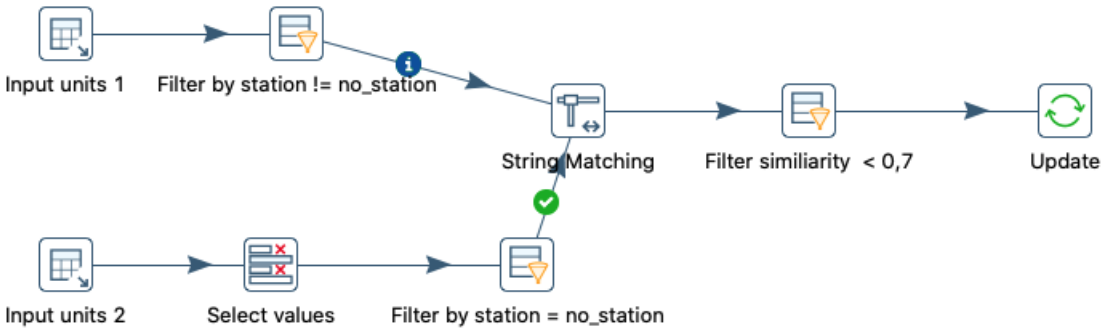

Figura D.11: Transformação para inserir o atributo estação quando este não existe

Figura D.12: Transformação para carregar os destinos dos primeiros meios enviados para cada ocorrência

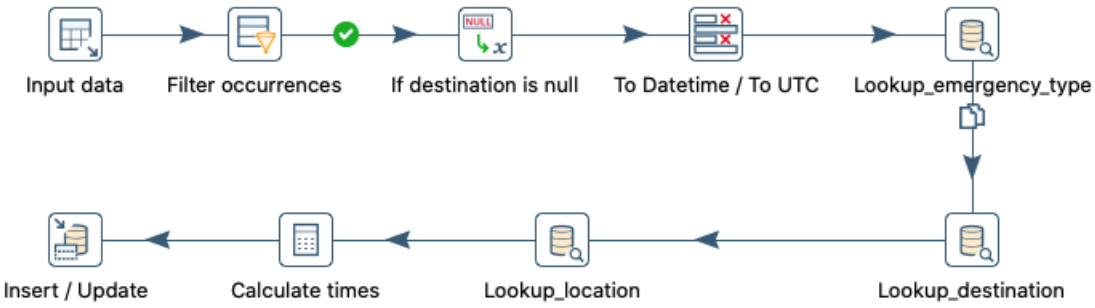

Figura D.13: Transformação para carregar dados na tabela de factos do Data Mart Ocorrências com Meios

Figura D.14: Transformação para carregar os dados na dimensão Grupo de Unidades

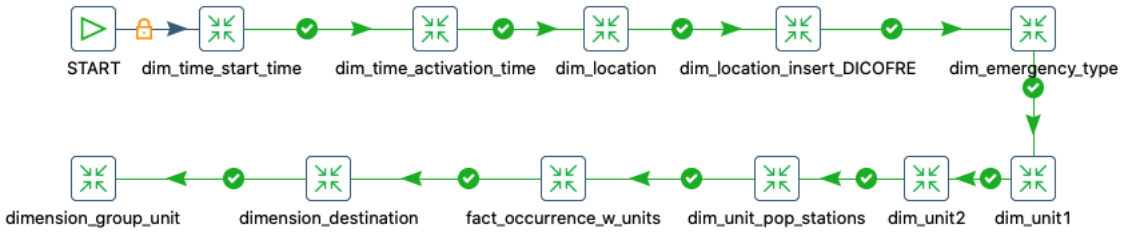

Figura D.15: Job que executa as transformações para carregar os dados no Data Mart Ocorrências com Meios

D.4 Data Mart Ocorrências com Informação Completa

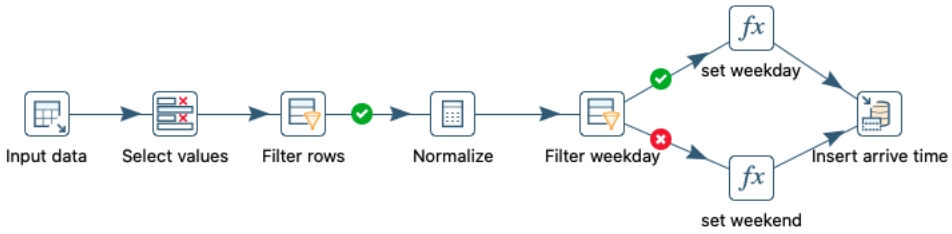

Figura D.16: Transformação para carregar os tempos de chegada do primeiro meio na dimensão Tempo

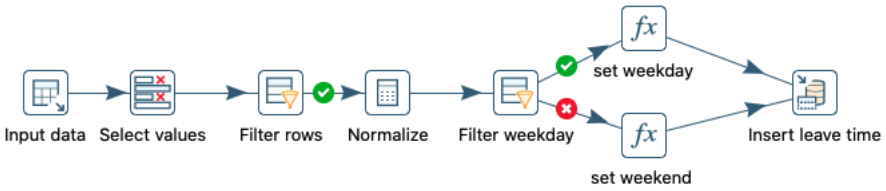

Figura D.17: Transformação para carregar os tempos de saída do primeiro meio do local da ocorrência na dimensão Tempo

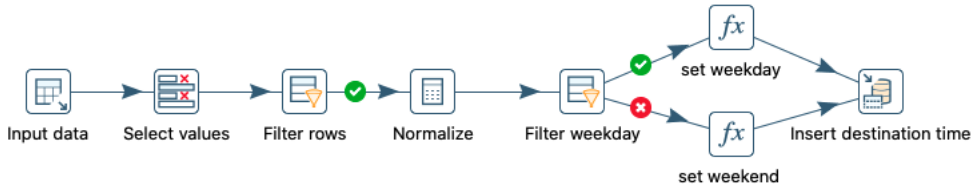

Figura D.18: Transformação para carregar os tempos de chegada do primeiro meio ao destino na dimensão Tempo

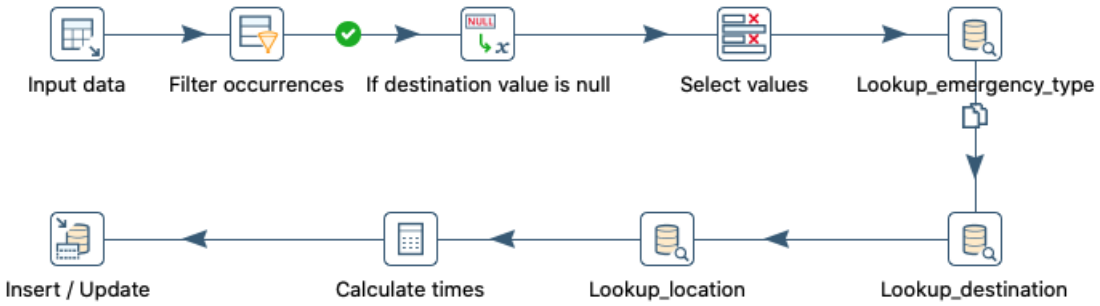

Figura D.19: Transformação para carregar dados na tabela de factos do Data Mart Ocorrências com Informação Completa

Figura D.20: Transformação para carregar os dados na dimensão Grupo de Unidades

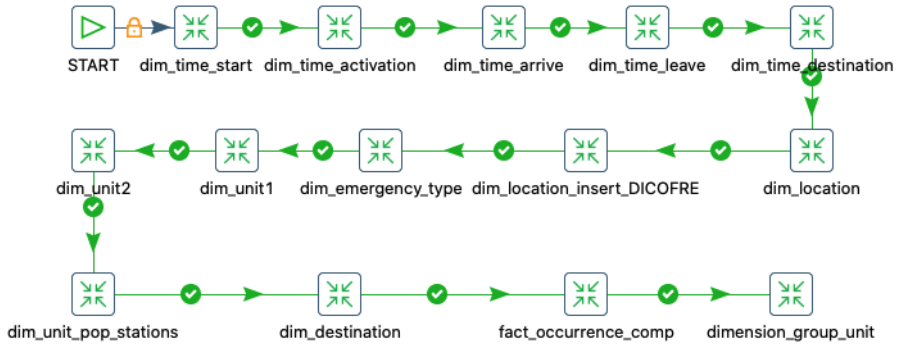

Figura D.21: Job que executa as transformações para carregar os dados no Data Mart Ocorrências com Informação Completa

D.5 Data Mart Ocorrências Futebol

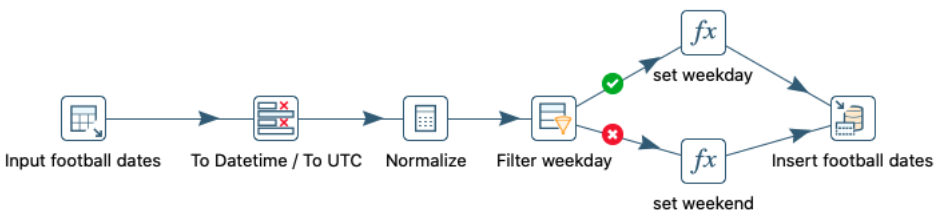

Figura D.22: Transformação para carregar os tempos de início dos jogos de futebol dimensão Tempo

Figura D.23: Transformação para carregar as localizações dos estádios de futebol na dimensão Localização

Figura D.24: Transformação para carregar os dados na dimensão Competição

Figura D.25: Transformação para carregar os dados na dimensão Equipas

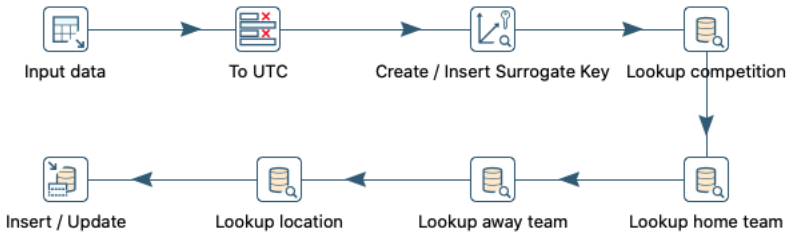

Figura D.26: Transformação para carregar os dados na dimensão Futebol

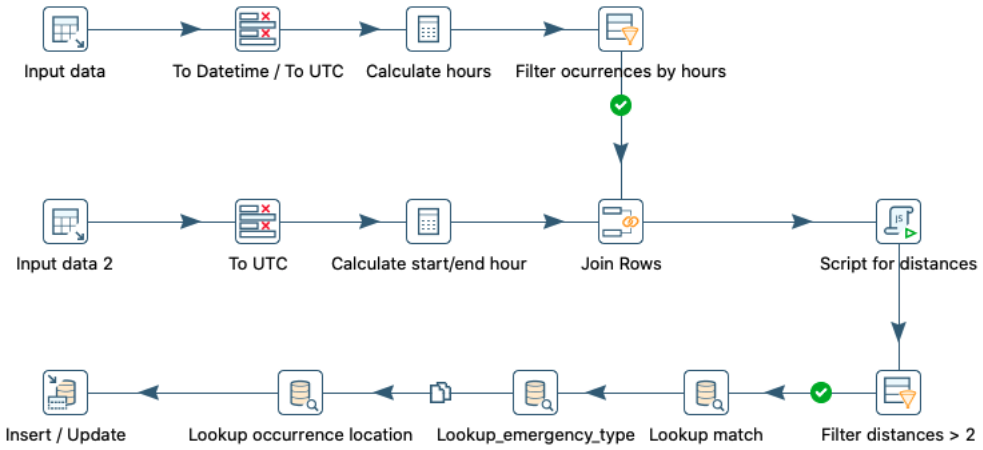

Figura D.27: Transformação para carregar os dados na tabela de factos do Data Mart Futebol

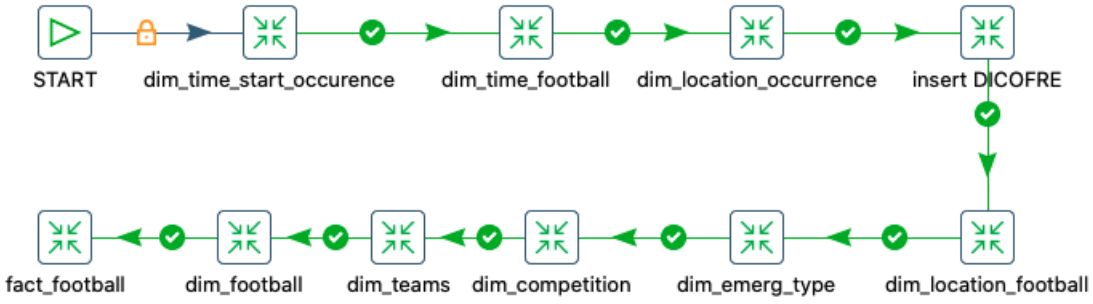

Figura D.28: Job que executa as transformações para carregar os dados no Data Mart Futebol

D.6 Data Mart Ocorrências Concertos

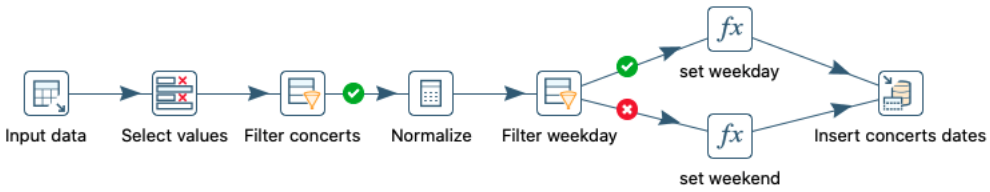

Figura D.29: Transformação para carregar os tempos de início dos concertos na dimensão Tempo

Figura D.30: Transformação para carregar as localizações dos concertos na dimensão Localização

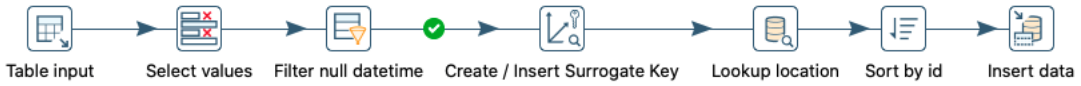

Figura D.31: Transformação para carregar os dados na dimensão Concertos

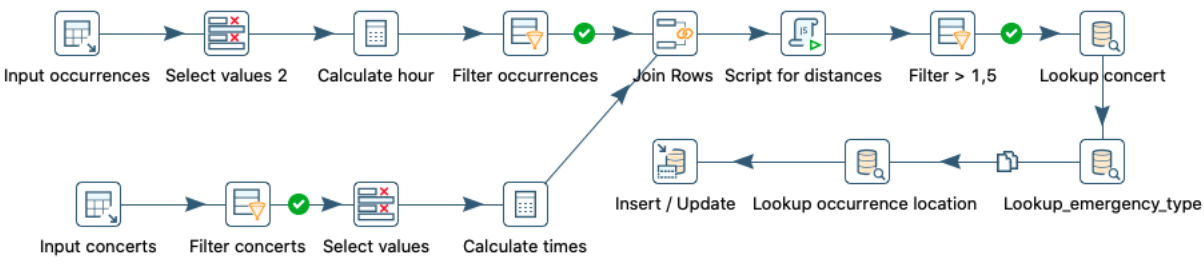

Figura D.32: Transformação para carregar os dados na tabela de factos do Data Mart Concertos

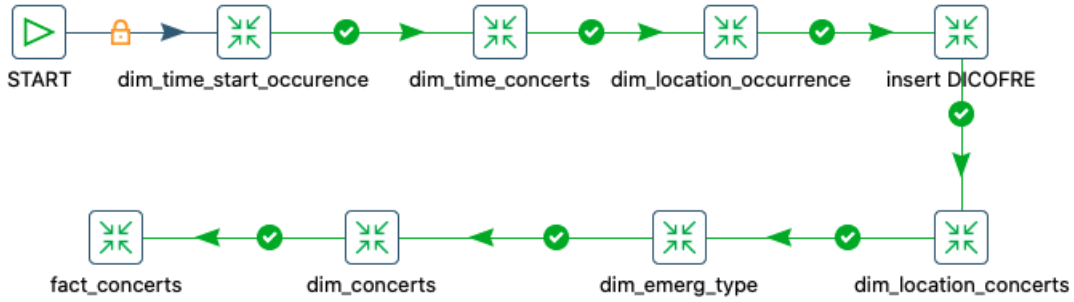

Figura D.33: Job que executa as transformações para carregar os dados no Data Mart Concertos

D.7 Data Mart Ocorrências Festivais

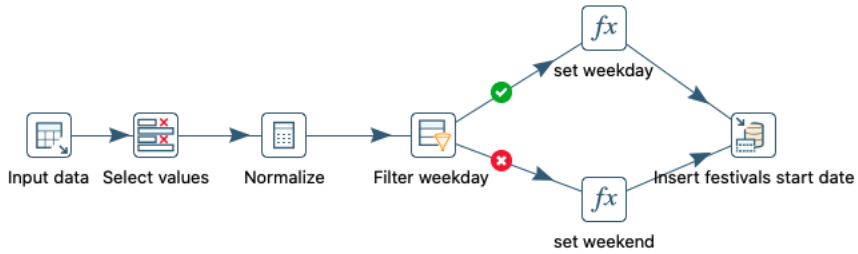

Figura D.34: Transformação para carregar os tempos de início dos festivais na dimensão Tempo

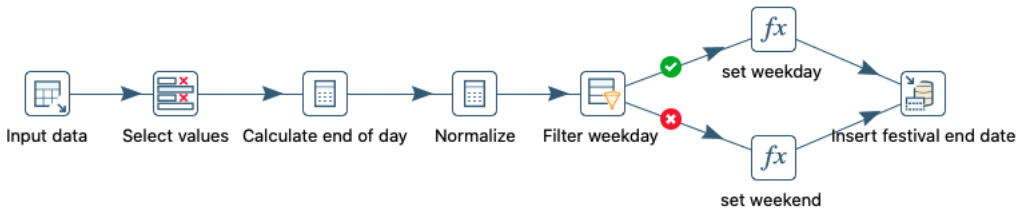

Figura D.35: Transformação para carregar os tempos de fim dos festivais na dimensão Tempo

Figura D.36: Transformação para carregar as localizações dos festivais na dimensão Localização

Figura D.37: Transformação para carregar os dados na dimensão Festivais

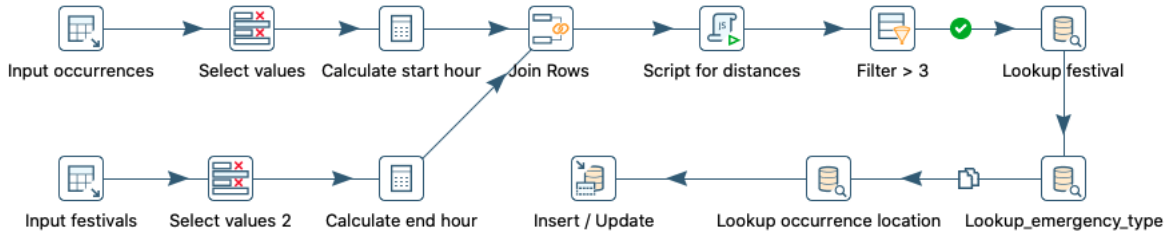

Figura D.38: Transformação para carregar os dados na tabela de factos do Data Mart Festivais

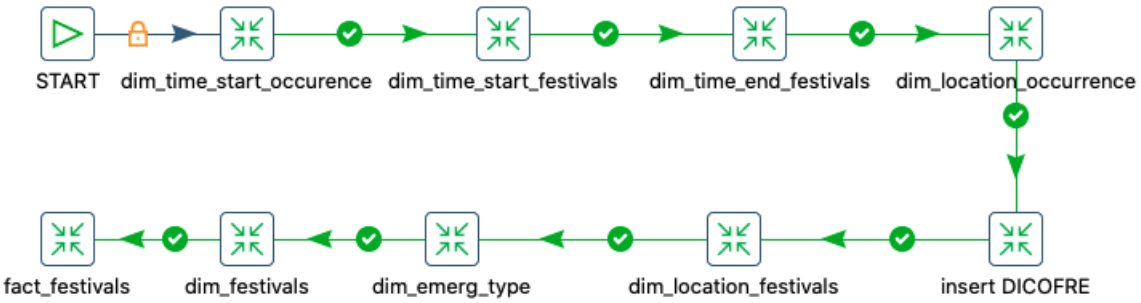

Figura D.39: Job que executa as transformações para carregar os dados no Data Mart Festivais

D.8 Data Mart Ocorrências Meteorologia

Figura D.40: Transformação para carregar os tempos das medições meteorológicas na dimensão Tempo

Figura D.41: Transformação para carregar as localizações das estações meteorológicas na dimensão Localização

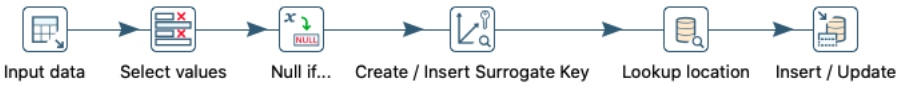

Figura D.42: Transformação para carregar os dados na dimensão Meteorologia

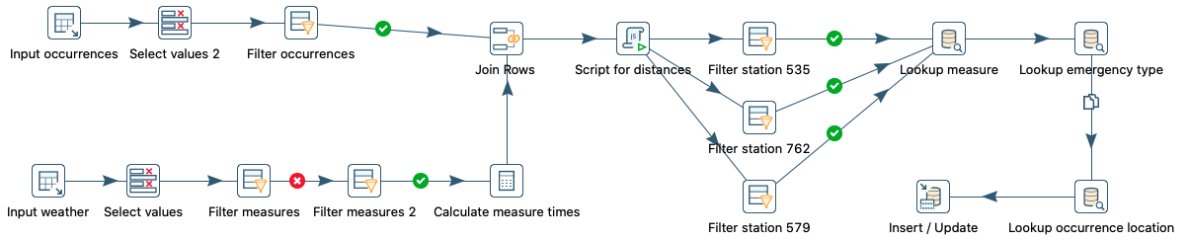

Figura D.43: Transformação para carregar os dados na tabela de factos do Data Mart Meteorologia

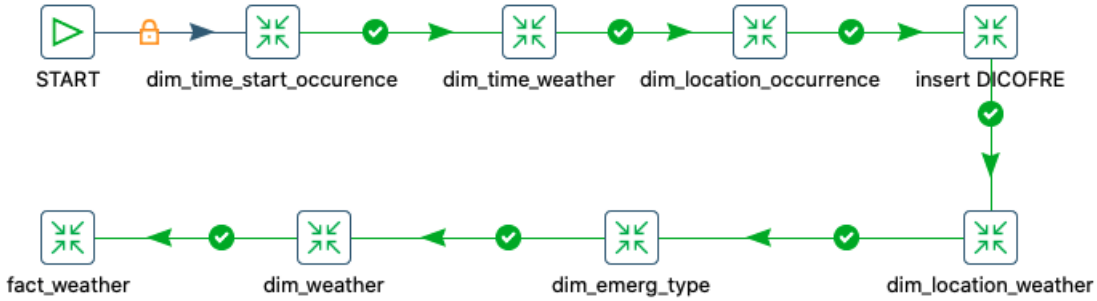

Figura D.44: Job que executa as transformações para carregar os dados no Data Mart Meteorologia

Apêndice E

Consultas realizadas à base de dados do SIADEM

1. **Ocorrências/Tempo**

   (a) Número de ocorrências numa data

   Por exemplo: 28/03/2018

```
SELECT CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101)
, COUNT (DISTINCT eid)
AS TOTOCORRENCIAS
FROM agency_event
WHERE CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101)
= '2018-03-28'
GROUP BY CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101)
```

   (b) Número de ocorrências num intervalo de tempo

   Por exemplo: Entre 28/03/2018 e o final do dia 31/03/2018

```
SELECT COUNT (DISTINCT eid) AS TOTOCORRENCIAS
FROM agency_event
WHERE CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101)
BETWEEN '2018-03-28' AND '2018-03-31'
```

(c) Número de ocorrências por ano

```
SELECT YEAR(CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101)) AS YYYY
, COUNT (DISTINCT eid) AS TOTOCORRENCIAS
FROM agency_event
GROUP BY YEAR(CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101))
ORDER BY YYYY
```

(d) Número de ocorrências por ano / mês

```
SELECT YEAR(CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101))
AS YYYY,
MONTH(CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101))
AS MM
, COUNT (DISTINCT eid) AS TOTOCORRENCIAS
FROM agency_event
GROUP BY YEAR(CONVERT(date, STUFF(STUFF(STUFF
(REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101)),
MONTH(CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101))
ORDER BY YYYY, MM
```

(e) Número de ocorrências por dia da semana

```
SELECT DATEPART(weekday, (CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101)))
AS weekday
, COUNT (DISTINCT eid) AS TOTOCORRENCIAS
FROM agency_event
GROUP BY DATEPART(weekday, (CONVERT(date, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ') , 101)))
ORDER BY weekday
```

2. **Ocorrências/Localização**

(a) Número de ocorrências por Distrito

```
SELECT SUBSTRING(lev3,1,2) as Distrito
, COUNT (DISTINCT eid) AS TOTOCORRENCIAS
FROM agency_event
GROUP BY SUBSTRING(lev3,1,2)
ORDER BY Distrito
```

(b) Número de ocorrências por Concelho

```
SELECT SUBSTRING(lev3,1,4) as Concelho
, COUNT (DISTINCT eid) AS TOTOCORRENCIAS
FROM agency_event
GROUP BY SUBSTRING(lev3,1,4)
ORDER BY Concelho
```

(c) Número de ocorrências por Freguesia

```
SELECT SUBSTRING(lev3,1,6) as Freguesia
, COUNT (DISTINCT eid) AS TOTOCORRENCIAS
FROM agency_event
GROUP BY SUBSTRING(lev3,1,6)
ORDER BY Freguesia
```

3. **Ocorrências/Prioridade**

(a) Número de ocorrências por Prioridade

```
SELECT priority,
COUNT (DISTINCT eid)
FROM agency_event
GROUP BY priority
ORDER BY priority
```

4. **Ocorrências/Tipo de Ocorrência**

(a) Número de ocorrências por Tipo de Ocorrência

```
SELECT tycod,
COUNT (DISTINCT eid)
FROM agency_event
GROUP BY tycod
```

```
ORDER BY tycod
```

5. **Ocorrências/Destino de Ocorrência**

   (a) Número de Ocorrências por Destino de Ocorrência

```
SELECT aecust1,
COUNT (DISTINCT eid)
FROM agency_event
GROUP BY aecust1
ORDER BY aecust1
```

6. **Tempos médios/máximos** (em segundos)

   (a) Tempo médio até ao acionamento

```
SELECT AVG(CAST(DATEDIFF(SS,
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))),
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ds_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ')))AS bigint))
as avgtime
FROM agency_event INNER JOIN common_event
ON agency_event.eid=common_event.eid
```

   (b) Tempo máximo até ao acionamento

```
SELECT MAX(CAST(DATEDIFF(SS,
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))),
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ds_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ')))AS bigint))
as avgtime
FROM agency_event INNER JOIN common_event
ON agency_event.eid=common_event.eid
```

   (c) Tempo médio até à chegada do meio

```
SELECT AVG(CAST(DATEDIFF(SS,
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))),
```

```
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ar_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))))AS bigint))
as avgtime
FROM agency_event INNER JOIN common_event
ON agency_event.eid=common_event.eid
```

(d) Tempo máximo até à chegada do meio

```
SELECT MAX(CAST(DATEDIFF(SS,
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))),
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ar_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))))AS bigint))
as avgtime
FROM agency_event INNER JOIN common_event
ON agency_event.eid=common_event.eid
```

(e) Tempo médio até à saída do meio

```
SELECT AVG(CAST(DATEDIFF(SS,
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))),
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (tr_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))))AS bigint))
as avgtime
FROM agency_event INNER JOIN common_event
ON agency_event.eid=common_event.eid
```

(f) Tempo máximo até à saída do meio

```
SELECT MAX(CAST(DATEDIFF(SS,
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))),
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (tr_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))))AS bigint))
as avgtime
FROM agency_event INNER JOIN common_event
ON agency_event.eid=common_event.eid
```

(g) Tempo médio de chegada do meio ao destino

```
SELECT AVG(CAST(DATEDIFF(SS,
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))),
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ta_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ''))))AS bigint))
as avgtime
FROM agency_event INNER JOIN common_event
ON agency_event.eid=common_event.eid
```

(h) Tempo máximo de chegada ao destino do meio

```
SELECT MAX(CAST(DATEDIFF(SS,
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ad_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' '))),
(CONVERT(datetime, STUFF(STUFF(STUFF(
REPLACE (ta_ts, 'UT', ''),13,0,':'),11,0,':'),9,0,' ''))))AS bigint))
as avgtime
FROM agency_event INNER JOIN common_event
ON agency_event.eid=common_event.eid
```

Apêndice F

Guia do Programador

F.1 Acesso ao Data Warehouse / Data Staging Area

Para aceder às bases de dados é necessário ter o Microsoft SQL Server instalado, bem como os dados de acesso às bases de dados.

F.1.1 Alterações no Data Warehouse

Para fazer alterações no Data Warehouse deve-se atualizar em primeiro lugar o *script* "*D2H_DW*" (apresentado na Secção B.1) que contém o modelo multidimensional do Data Warehouse, e de seguida deve executar-se o script.

F.2 Acesso aos ficheiros do Pentaho Data Integration

Para aceder ao PDI e aos seus ficheiros é necessário ter instalado o *Java*. Preferencialmente a versão de 64 bits do *Java 8*. Tanto com o *Java Development Kit (JDK)* como com o *Java Runtime Environment (JRE)* o PDI irá funcionar.

Os ficheiros do PDI estão divididos em dois tipos: as transformações (formato ktr) e os *jobs* (formato kjb). É importante salientar que nas transformações os *steps* são executados em paralelo sempre que possível, e realizam transformações sobre os dados a um baixo nível. Por sua vez os *jobs* executam os *steps* de forma sequencial e uma das suas possíveis funcionalidades é executar sequências de jobs.

Para alterar as transformações e os *jobs* é necessário entrar na aplicação do PDI e aceder à pasta onde estão guardados.

Para executar as transformações e os *jobs* é necessário abrir a aplicação do PDI, aceder aos ficheiros pretendidos e carregar no botão *"Run"*. Para que as transformações sejam executadas com sucesso é necessário ter acesso tanto à Data Staging Area como ao Data Warehouse.

Para realizar um refrescamento do Data Warehouse é necessário em primeiro lugar executar as transformações que carregam dados para a Data Staging Area, e de seguida executam-se os *jobs* que carregam dados para cada um dos Data Marts.