Pipeline de RANU: Modelação Open EHR, criação de um Data Warehouse e criação de indicadores numa plataforma de Bussiness Inteligence

Lara Vaz, Mariana Lindo and Tiago Novais

Informatics Department, University of Minho, Braga,Portugal.

Abstract

A perda auditiva permanente é uma das perturbações congénitas mais comuns, com uma incidência estimada de um a três por cada mil recém-nascidos. No contexto do Rastreio Auditivo Neonatal Universal, existe uma série de informações que devem ser registadas, nomeadamente dados relacionados com a gravidez, o nascimento e o recém-nascido, a identificação dos fatores de risco de surdez e os resultados da pontuação do Apgar. Por conseguinte, um dos objetivos deste estudo foi desenvolver um *template* de OpenEHR para ser preenchido no âmbito do RANU, com essas mesmas informações. De seguida, partindo dos campos desse *template*, foi possível efetuar o seu mapeamento com campos semelhantes presentes num dataset privado e posteriormente armazenar estes dados mapeados numa base de dados NoSQL, o MongoDB. Após isto, procedeu-se ao estabelecimento da conexão entre a Base de Dados NoSQL com uma Base de Dados Relacional, o MySQL. Posteriormente, procedeu-se à modelação de um Data Warehouse que foi povoado com todas as informações presentes no dataset privado, mas de forma estruturada, consistente e processada. Finalmente, foram construídos gráficos para a visualização de indicadores relativos ao RANU, utilizando uma plataforma de *Bussiness Inteligence*, o Tableau, tendo sido este alimentado pelos dados presentes no DW.

Keywords: openEHR, RANU, Data Warehouse, MongoDB, MySQL, Tableau, Bussiness Inteligence

1 Introdução

Sempre que um paciente chega ao hospital, seria útil ter acesso ao seu histórico médico, a fim de aumentar a assertividade do diagnóstico e, consequentemente, obter um tratamento mais adequado e atempado. Desta forma, os profissionais de saúde teriam à sua disposição mais informação na qual poderiam basear as suas decisões. No entanto, uma parte importante desta informação está dispersa pelos vários sistemas de saúde em que os doentes são tratados quando, idealmente, deveria estar reunida e presente em todos, para ser disponibilizada sempre que necessária. Este é precisamente o objetivo que a fundação OpenEHR [1] propôs alcançar, uma vez que, com a abordagem OpenEHR, os dados do paciente podem ser estruturados, armazenados, geridos e trocados de forma segura e fiável entre diferentes prestadores de cuidados de saúde e outros grupos de interesse [2].

1.1 Processo Clínico Eletrónico

Um Processo Clínico Eletrónico (EHR) é uma coleção eletrónica de informação sobre uma pessoa, que é armazenada numa máquina, geralmente um computador, e inclui documentação clínica, resultados de testes, imagens e informação de apoio à decisão [3]. A qualidade do EHR pode melhorar a qualidade dos cuidados de saúde [4] e facilitar a investigação para fins académicos, ao contribuir para uma prática médica mais baseada em provas[5]. Além disso, existe um consenso sobre a necessidade de um sistema que permita a manutenção e a interoperabilidade destes registos [6][7][8].

1.2 Processo ETL

O processo ETL (*Extract-Transform-Load*) é responsável pela integração de dados num repositório designado emphData Warehouse (DW) e envolve três fases de tratamento de dados: extração, transformação e carregamento, conforme representado na Figura 1. O primeiro passo em qualquer processo ETL é a extração de dados, que consiste na recuperação de dados provenientes de uma ou mais fontes de dados heterogéneas e na migração dos mesmos para um repositório de dados temporário. Assim, o processo precisa de integrar eficazmente sistemas presentes em diferentes plataformas, tais como diferentes sistemas de gestão de bases de dados, sistemas operacionais e protocolos de comunicação [9]. A segunda fase é a transformação, na qual os dados extraídos são transformados para um formato específico com base em regras, funções e condições, de modo a facilitar a fase final [10]. Por último, o carregamento de dados para a estrutura multidimensional alvo é o passo final do ETL. Nesta etapa, os dados extraídos e transformados são guardados num DW para que possam ser analisados e utilizados adequadamente [9].

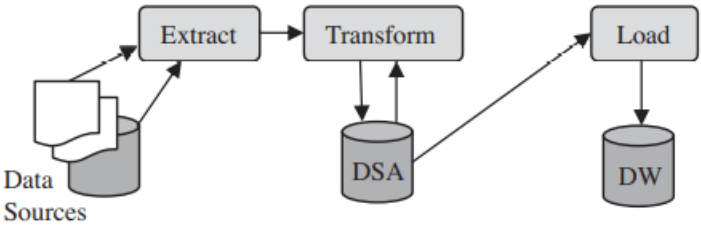

Fig. 1 Esquema geral do processo ETL.

1.3 Objetivos e Motivações

A perda auditiva permanente é uma das perturbações congénitas mais comuns, com uma incidência estimada de um a três por cada mil recém-nascidos [11][12]. Hoje em dia, muitos fatores de risco de perda auditiva em recém-nascidos são conhecidos. No entanto, se os fatores de risco fossem utilizados como o único critério para levar a cabo o rastreio auditivo neonatal, só seria possível identificar 50% dos casos de surdez. Por outro lado, existem provas clínicas de que a intervenção precoce influencia positivamente o prognóstico linguístico e cognitivo da criança e desenvolvimento social. Portanto, em 1992, foi criado o Rastreio Auditivo Neonatal Universal (RANU), que visa testar todas as crianças ao nascimento ou, no máximo, até 30 dias após o nascimento e, em caso de perda auditiva confirmada, o bebé deve iniciar a intervenção precoce e adequada, até os 6 meses de idade [13].

No contexto do RANU, existe uma série de informações que devem ser registadas, nomeadamente dados relacionados com a gravidez, o nascimento e o recém-nascido, a identificação dos fatores de risco de surdez e os resultados da pontuação do Apgar. [14] [15]. Por conseguinte, um dos objetivos deste estudo é desenvolver um *template* de EHR para ser preenchido no âmbito do RANU, contendo informações relativas ao registo do nascimento (incluindo os resultados do Apgar), ao processo do rastreio auditivo e à identificação dos fatores de risco de surdez. Para a construção do *template* serão seguidas as especificações do openEHR, como o primeiro passo de um pipeline de investigação do RANU, representado na Figura 2. O próximo passo é simular uma situação real em que seriam extraídos diferentes campos do *template*, preenchidos com dados reais de pacientes. Contudo, como o *template* não foi realmente preenchido por pacientes, utilizaram-se dados provenientes de um dataset privado que continha os mesmos campos presentes no *template* desenvolvido. Estes dados foram posteriormente mapeados com os campos do *template* e guardados num ficheiro JSON no MongoDB. Com os dados armazenados no MongoDB, é possível ir para o passo 5, que é a modelação do DW. Isso inclui a criação de tabelas de factos e dimensões e a escolha de uma arquitetura apropriada. Nas etapas 6 a 8, será executado o processo ETL, que termina com o povoamento do DW. Por fim, a partir dos dados presentes no DW é

possível extrair informações e desenvolver indicadores estatísticos relativos ao RANU, que podem ser incorporados numa interface de *Business Intelligence* (BI). Assim sendo, o objetivos deste trabalho passam pela realização de cada uma das etapas descritas no pipeline.

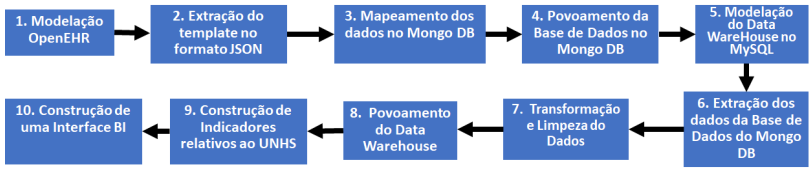

Fig. 2 Esquema do Pipeline de Investigação do RANU.

Este artigo inclui seis secções. Após a Introdução, a 2ª secção, Background, apresenta alguns estudos semelhantes a este. Na 3ª secção, as ferramentas e métodos utilizados neste estudo são mencionados e descritos. Na secção 4, são apresentadas e explicadas todas as etapas que fazem parte do pipeline de investigação que culmina na obtenção de indicadores estatísticos relativos ao RANU. A 5ª e última secção contempla a conclusão e trabalho futuro.

2 Background

A livre circulação dos cidadãos europeus através dos estados membros da União Europeia acrescenta um importante nível de complexidade aos esforços estratégicos de interoperabilidade sanitária. A utilização de dados de saúde eletrónicos tem sido marcada como uma importante atividade e política estratégica para melhorar os cuidados de saúde nos países europeus. Os cuidados de saúde transfronteiriços dependem da capacidade de estabelecer práticas partilhadas no que diz respeito ao intercâmbio de dados dos doentes em todos os países. O objetivo do estudo de Gavrilov et al.[16] foi propor um novo design de armazenamento de dados de cuidados de saúde baseado no processo reestruturado de ETL. Assim, desenvolveram um modelo que oferece um conjunto abrangente de serviços de interoperabilidade para permitir que as plataformas nacionais de saúde criem redes de informação transfronteiriças, em conformidade com os Serviços Abertos Inteligentes para os Pacientes Europeus. A abordagem apresentada incorpora a interoperabilidade técnica e organizacional, interligando a norma HL7 e a estrutura de Pontos de Contacto Nacionais Abertos, a fim de proporcionar uma arquitetura modular, escalável e interoperacional.

Wulff et ai. [17] desenvolveram uma abordagem baseada em openEHR para alcançar a interoperabilidade em Sistemas de Apoio à Decisão Clínica (CDSS), através da projeção e implementação de um sistema para a deteção automática da Síndrome da Resposta Inflamatória Sistémica (SIRS) nos cuidados intensivos pediátricos. Para criar e implementar o CDSS, os autores utilizaram bases de conhecimento e interfaces interoperáveis, reutilizaram arquétipos

acordados internacionalmente, incorporaram a terminologia LOINC e desenvolveram *queries* AQL, que permitiram recuperar factos dinâmicos de uma forma padronizada e inequívoca. Os autores consideraram a utilização dos arquétipos openEHR e das *queries* AQL uma abordagem viável para colmatar a lacuna de interoperabilidade entre as infraestruturas locais e o CDSS. O conceito concebido foi transferido com sucesso para um CDSS baseado no openEHR clinicamente avaliado.

3 Ferramentas e Métodos

3.1 SCRUM

De forma a facilitar a organização do trabalho ao longo do semestre e a divisão das tarefas entre os membros do grupo, foi utilizada a *framework* Scrum, que permite que as pessoas possam lidar com problemas adaptativos complexos, enquanto cumprem várias tarefas, de forma produtiva e criativa. Para isso, a equipa Scrum possui um *Backlog*, uma lista dinâmica e definida de acordo com as prioridades de execução das tarefas, que contém todo o trabalho que poderá ser necessário efetuar até se obter o produto final. Esta lista é preenchida pela equipa Scrum e validada pelo *Product Owner*, que representa os clientes, neste caso os docentes. O trabalho presente no *Backlog* é dividido, de forma a ser executado em diferentes Sprints, que neste projeto em concreto consistiam num período de 2 semanas.

De forma que o grupo soubesse o que realizar em cada Sprint, era efetuado previamente a este, um Sprint *Planning*, no qual o grupo selecionava tarefas do *Backlog* que pretendia realizar nesse período de tempo. Durante a realização do Sprint, o grupo reunia-se frequentemente, normalmente 15 minutos por dia, de forma a analisar o estado atual do trabalho, a identificar os problemas e propor soluções. Essas reuniões eram lideradas pelo Scrum *Master*, que era o responsável pela equipa durante esse Sprint, estando consciente de todo o trabalho que estava em curso, assim como das dificuldades e desafios. No fim do Sprint, a equipa reunia-se com o *Product Owner*, onde o Scrum *Master* apresentava uma retrospetiva sobre todo o trabalho que foi efetuado nesse Sprint. Por último, era ainda efetuado um novo Sprint *Planning*, para planear o próximo Sprint. No caso particular deste projeto, o Scrum *Master* era rotativo, de forma que cada um dos elementos da equipa tivesse a oportunidade de assumir esta função. Ao logo do semestre foram efetuados 5 Sprints e um total de 37 tarefas, uniformemente distribuídas pelos Sprints, o que permitiu uma execução eficaz, atempada e produtiva de todo o trabalho [18].

3.2 OpenEHR

O OpenEHR é um *open standard* [19] mantido pela openEHR Foundation, que visa converter dados de saúde de uma forma física para uma forma eletrónica e garante a interoperabilidade universal entre dados eletrónicos [20].

Este também facilita a interoperação de sistemas EHR, porque um *dataset* de EHR pode ser totalmente representado através de arquétipos compartilháveis.

A abordagem openEHR tem 4 pilares principais: o Modelo de Referência (RM), o Modelo de Arquétipos (AM), a terminologia e os Tipos de Fundação (FT). O RM é um modelo de informação formal e estável que se concentra nas estruturas lógicas de um EHR. O RM tmabém define as estruturas e os atributos básicos necessários para expressar instâncias de dados EHR, incluindo tipos de dados, estruturas de dados e componentes de um EHR [21].

O AM consiste em arquétipos e *templates*. Os arquétipos são os artefactos formais e semânticos que facilitam a coleta, armazenamento, recuperação, representação, comunicação e análise de dados clínicos. Estes podem ser modelados por profissionais clínicos e especialistas em informática médica, por meio da restrição do RM. Os arquétipos desempenham um papel importante na abordagem openEHR, pois não só suportam a representação da semântica, mas também facilitam a manutenção [22], escalabilidade, interoperabilidade [23] e o *input* dos profissionais clínicos [24]. Existem muitos tipos de arquétipos, sendo um deles a *evaluation*, que representa opiniões e avaliações sobre o paciente, como diagnóstico, avaliação de risco, objetivos e recomendações. Outros dois tipos de arquétipos comummente utilizados são a *observation* que representa todas as observações objetivas, medidas de alguma forma, de fenómenos e fenómenos relatados pelo paciente, por exemplo, dor e ainda o *encounter*, que serve para registar os detalhes de uma única interação, contacto ou evento de atendimento entre um indivíduo e um(ns) prestador(es) de saúde, para a prestação de serviço(s) de saúde. Essa interação pode ser presencial ou remota [25]. Em relação aos *templates* openEHR, estes são normalmente *compositions*, uma vez que são constituídos por diversos arquétipos, que podem (ou não) ser alterados para fins específicos de contexto. Os *templates* são normalmente usados para gerar interfaces de programação de aplicações (APIs), definições de esquema XML (XSDs), formulários de interface do utilizador e esquemas de armazenamento. Como o openEHR é uma abordagem terminológica neutra, então, permite que terminologias externas como o SNOMED-CT possam ser referidas em arquétipos [21].

Por fim, os FT são tipos genéricos de baixo nível assumidos e utilizados em todos os componentes e especificações do openEHR. Os FT incluem todos os tipos de dados que podem ser usados nos arquétipos e *templates* [26]. Alguns exemplos de tipos de dados são o *text* (texto livre), o *coded text* (representação textual codificada de termos, como strings, números, símbolos ou outros), a *quantity* (quantidades físicas mensuráveis como massa e comprimento), a *duration* (duração de um evento ou (in)atividade, como dias, horas, minutos e segundos), um *boolean* (verdadeiro ou falso), a *datetime* e o *count* (quantidades contáveis, por exemplo, o número de cigarros fumados num dia). Neste projeto, o padrão openEHR foi adotado para o desenvolvimento dos arquétipos e do *template* final, o que incluiu o uso do RM, do AM, de uma terminologia e também dos FT [27].

3.3 SNOMED-CT

O SNOMED-CT é uma terminologia clínica multilingue de termos de saúde, com conteúdo clínico cientificamente validado e lançado mensalmente e que permite a representação consistente do conteúdo clínico em EHRs. Este fornece uma maneira padronizada de representar frases clínicas capturadas pelo clínico e permite a sua interpretação automática. Além disso, o SNOMED-CT permite que sistemas de suporte verifiquem os registos dos pacientes e forneçam conselhos em tempo real. Para conseguir tudo isso, o modelo lógico do SNOMED-CT define a forma pela qual cada tipo de componente e derivado do SNOMED-CT está relacionado e representado. Os principais tipos de componentes do SNOMED-CT são conceitos, descrições e relacionamentos. O modelo lógico especifica como esses componentes podem ser geridos numa configuração de implementação para atender a uma variedade de usos primários e secundários [28].

3.4 Clinical Knowledgment Manager

O principal trabalho da comunidade openEHR Internacional é realizado por 4 'programas' que se concentram respetivamente em especificações, modelação clínica, software e educação. Ao nível da modelação clínica, existe o Clinical Modelling Program, que é realizado por profissionais clínicos e especialistas em informática médica e que trabalham no ambiente Clinical Knowledge Manager (CKM) [20]. O CKM é um sistema para desenvolvimento colaborativo, gestão e publicação de uma ampla gama de recursos de conhecimento clínico, como arquétipos, *templates* e conjuntos de termos [29]. Neste projeto, os arquétipos que já existiam no CKM foram usados e alterados para construir o *template* final.

3.5 Archetype Designer

Como parte de um esforço para expandir a comunidade global openEHR, a Better e a openEHR International lançaram o Archetype Designer, um ambiente de modelação clínica gratuito, baseado na web, para o desenvolvimento de arquétipos openEHR. Este permite a autoria visual de arquétipos e *templates*, incluindo a análise completa de arquétipos, validação, *flattening* e serialização. Com o Archetype Designer, os utilizadores têm acesso a vários repositórios a partir dos quais podem inserir e obter *templates* e modelos de arquétipos com várias opções de exportação [30]. Neste trabalho, a *framework* Archetype Designer foi utilizada para pesquisar, modificar e usar arquétipos para construir um *template* final sobre o RANU.

3.6 MongoDB

O MongoDB é uma base de dados NoSQL orientada a documentos e utilizada para o armazenamento de dados de alto volume. Em vez de usar tabelas

e linhas como nas bases de dados relacionais tradicionais, o MongoDB utiliza coleções e documentos. Os documentos consistem em pares chave-valor que são a unidade básica de dados no MongoDB. As coleções contém conjuntos de documentos e funções que são equivalentes às tabelas de uma base de dados relacional. Como vantagens do Mongo DB em relação a bases de dados relacionais destaca-se o facto deste permitir representar relacionamentos hierárquicos, armazenar arrays e outras estruturas mais complexas com mais facilidade e também a elevada escalabilidade dos seus ambientes [31]. Neste projeto, utilizou-se o MongoDB para armazenar, em formato JSON, um *array* com dados de vários pacientes, que foram mapeados, de forma a corresponderem aos campos presentes no *template* desenvolvido sobre o RANU.

3.7 MySQL

O MySQL é uma base de dados relacional desenvolvida pela Oracle e é baseado em linguagem de *query* estruturada (SQL). Sendo uma base de dados relacional, este armazena dados em tabelas separadas, em vez de colocar todos os dados num grande repositório. O MySQL fornece também um ambiente de programação flexível, uma vez que permite o desenvolvimento de modelos lógicos no MySQL Workbench, através da utilização de tabelas, colunas e de regras que regem os relacionamentos entre diferentes campos de dados, como por exemplo, *one-to-one, one-to-many, unique*, obrigatório ou opcional e "apontadores" entre diferentes tabelas. O MySQL impõe estas regras para que, com uma base de dados bem projetada, não existam dados inconsistentes, duplicados ou desatualizados [32]. Para este projeto, utilizou-se o MySQL para a modelação do *Data Warehouse* e o seu posterior armazenamento num *schema* do MySQL.

4 Resultados e Discussão

A primeira etapa realizada neste projeto foi a construção de um *template*, representado na Figura 3, segundo as especificações openEHR. Para isto, foram utilizados 6 arquétipos, em que 5 são do tipo *evaluation* e um último do tipo *observation*.

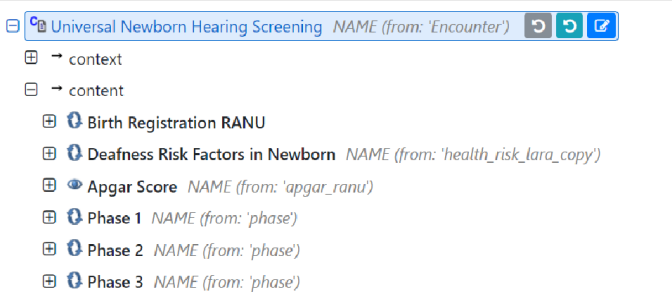

Fig. 3 Esquema do *template* 'Universal Newborn Hearing Screening'.

O primeiro arquétipo, *Birth Registration RANU* é referente aos dados que são preenchidos no momento do nascimento e resulta de alterações efetuadas no arquétipo já existente no CKM openEHR-EHR-EVALUATION.pregnancy summary.v0. As alterações consistiram fundamentalmente na eliminação ou adição de alguns campos, de modo a que apenas ficassem presentes aqueles que são preenchidos no momento do nascimento [33]. Além disso, alguns campos também tiveram que ser alterados, pois a maioria era de texto livre e na secção de detalhes do arquétipo, os autores sugeriram a alteração desses campos para *coded text* de acordo com uma determinada terminologia. Neste caso, a terminologia escolhida foi SNOMED-CT [28], que foi utilizada para codificar o elemento puerpério como '*Normal*' ou '*Complicated*'. Concluídas todas as mudanças, o arquétipo final ficou constituído por 2 partes fundamentais: uma parte destinada aos dados que inclui os campos data/hora de nascimento, tempo de gestação, puerpério da mãe do recém-nascido, local de nascimento, peso, sexo e outra parte destinada ao protocolo no qual estão integrados o id de processo e o id do paciente.

O segundo arquétipo utilizado foi o *Deafness Risk Factors in Newborn*, derivado do openEHR-EHR-EVALUATION.health_risk.v1, existente no CKM. As alterações efetuadas foram idênticas às realizadas com o 1º arquétipo, sendo que o resultado final apresenta na mesma 2 partes principais. A parte dos dados integra um conjunto de fatores de risco de surdez do tipo *coded text*, também codificados de acordo com a terminologia SNOMED-CT, como se pode observar na Tabela 1. Para além destes, a parte dos dados contém também a data em que foi realizado o rastreio destes mesmo fatores de risco e ainda um campo *coded text* que identifica se o paciente não apresenta fatores de risco, se apresenta apenas fatores de risco não tardio ou se apresenta fatores de risco tardios e não tardios.

Table 1 Representação dos termos utilizados no campo 'Risk Factors' em SNOMED-CT.

Code System Concept Name	Code System Concept Code
Family history of congenital sensorineural hearing loss	456531000124109
Congenital infection caused by Herpes virus	715337002
Congenital syphilis	35742006
Congenital toxoplasmosis	73893000
Congenital rubella syndrome	1857005
Cytomegalovirus	407444007
Congenital anomaly of face	398302004
Low birth weight infant ($< 1500g$)	276610007
Hyperbilirubinemia (with exchange transfusion)	14783006
Bacterial meningitis	95883001
Apgar at 1 minute (0 to 4)	169895004
Apgar at 5 minute (0 to 6)	169909004
Deafness symptom	272033007
Artificial respiration	40617009
History of therapy with ototoxic medication	441899004

Para além destes arquétipos, foi elaborada o único arquétipo do tipo observation, o Apgar Score, sendo este constituído por dois *point events*, um correspondente ao minuto 1 e outro correspondente ao minuto 5. Cada um destes apresentava um valor inteiro correspondente ao valor total do Apgar Score em cada momento. Os restantes arquétipos, *Phase 1, Phase 2* e *Phase 3*, seguem todos a mesma lógica em que é registado a avaliação feita no formato *coded text* utilizando-se a terminologia *Pass*,*Refer Right*, *Refer Left*, *Refer Both* em cada fase assim como a respetiva data da realização e o ID do avaliador da mesma.

De seguida, foi efetuado o mapeamento dos campos presentes no dataset privado com os *paths* correspondentes no *template* e seguindo as especificações OpenEHR. Para isso, recorreu-se a uma *script python* que por cada linha do ficheiro CSV, onde se encontrava o dataset privado fornecido, dividiria cada linha por ponto e vírgula. O resultado deste processo foi posteriormente tratado, de forma a que cada campo estivesse de acordo com as especificações OpenEHR adequadas para esse campo e formato específicos. Por exemplo, para o mapeamento dos campos do tipo *coded text*, presentes no *template*, foi necessário inserir a 'terminology_id' assim como o 'code_string' associados. Todas estas correspondências foram realizadas através de chaves e valores de um dicionário, de tal modo que o resultado final foi um dicionário por linha com vários dicionários no seu interior. Finalmente, cada dicionário, que representa uma linha, foi adicionado a uma lista que por sua vez através do comando insert_many() foi inserida numa coleção do MongoDB.

O passo seguinte consistiu na modelação do Data Warehouse em formato estrela no MYSQL Workbench (Figura 4). Para isso, foram construídas as tabelas de dimensões e uma tabela de factos. É importante salientar que foi necessário construir uma tabela de dimensões para cada fator de risco, uma

vez que se existisse apenas uma única tabela para todos os fatores, então existiria também uma única chave estrangeira a fazer correspondência à tabela de factos e no contexto do problema tal não faria sentido, pois um paciente pode conter mais do que um fator de risco. Tendo as tabelas já criadas seguiu-se, então, o povoamento do DW, em que se começou pela inserção de valores nas tabelas de dimensões. A ordem pelo qual as inserções foram efetuadas é de extrema importância, visto que os atributos inseridos nas tabelas de dimensões seriam depois chaves estrangeiras na tabela de factos. Assim sendo, começando pelo povoamento das tabelas de dimensões, este foi efetuado através do editor de *queries* do MYSQL Workbench, no qual foram introduzidas as *queries* responsáveis por este povoamento. Cada tabela seria constituída por 'id', um identificador auto incrementado que representava a *primary key*, e 'descricao', uma breve descrição dos diferentes valores que cada id representaria.

Seguidamente foi necessário povoar a tabela de factos que apresenta todos os campos que são chaves estrangeiras para as tabelas de dimensões, alguns campos relativos ao paciente em si, assim como alguns campos que podem ser obtidos através de cálculos com outros campos. Entre estes campos calculados destacam-se 'idade1', 'idade2', 'idade3', que representam as idades em dias de cada recém-nascido aquando a avaliação das fases 1, 2, 3, respetivamente e os campos 'tempo1_2' e 'tempo2_3', que representam o tempo decorrido em dias entre a primeira e a segunda avaliações e o tempo entre as segunda e terceira avaliações, respetivamente. Quanto aos restantes campos, estes contém o 'num_seq': número sequencial do paciente, 'num_processo': número identificador do processo, 'data_nascimento': data de nascimento do recém-nascido, 'gestacao': tempo de gestação, 'peso': peso em gramas, 'data_avaliacao1', 'data_avaliacao2', 'data_avaliacao3', 'data_avaliação_r': datas nas quais foram realizadas as avaliações 1, 2, 3 e o rastreio, 'nmec_avaliador1', 'nmec_avaliador2', 'nmec_avaliador3', 'nmec_avaliador_r': números mecanográficos dos avaliadores que realizaram a avaliação 1, 2, 3 e o rastreio, 'sexo': chave estrangeira do sexo, 'local_nascimento': chave estrangeira do local de nascimento, 'apgar1': chave estrangeira do valor do Apgar Score no primeiro minuto, 'apgar5': chave estrangeira do valor do Apgar Score no quinto minuto, 'puerperio': chave estrangeira do puerpério, 'risco': chave estrangeira do valor do risco, 'avaliacao1', 'avaliacao2', 'avaliacao3': chaves estrangeiras dos valores das avaliações 1, 2 e 3, 'fator1', 'fator2', fator3', 'fator4', 'fator5', 'fator6', 'fator7', 'fator8', 'fator9' e 'fator10': traduzem a ausência ou presença dos diversos fatores. O povoamento desta tabela foi efetuado recorrendo a uma *script python*, utilizando o driver *mysql.connector* para fazer a ligação com o MySQL e o *pymongo* para fazer a ligação com o MongoDB. Quanto à *script*, a maioria dos campos das tabelas de facto foram facilmente preenchidos sem necessidade de pré-processamento, contudo, tal não se verificou em todos os campos. Para o campo 'peso' foi necessária a conversão do valor do mesmo para gramas, caso as unidades estivessem em libras. Além deste campo, todas as datas tiveram de ser instanciadas para o tipo *Datetime*, pois, no Mysql o campo da data era do tipo *Datetime*. Como foi referido anteriormente as idades e os

tempos teriam de ser calculados e por isto foi criada a função *'days_between'* que calcula o número de dias entre duas datas. Em adição, os valores dos campos que se apresentavam como chaves estrangeiras necessitaram da adição de uma condição que visa transformar os valores que no MongoDB se apresentam como nulos, na string 'Unknown'.

Concluído o pré-processamento, procedeu-se à criação de uma *script* SQL, que através do cursor do *mysql.connector* poderia ser processada pelo MySQL e assim preencher o DW. De notar que as chaves estrangeiras foram obtidas através de *queries* SQL que selecionavam o id da tabela de dimensões, no qual o valor da respetiva coluna 'descricao' coincidia com os valores obtidos na base de dados do MongoDB. É também relevante salientar que foi necessária a utilização de *queries* SQL para atualizar os valores que representam nulos do DW para *NULL*, uma vez que o *mysql.connector* impossibilitava a introdução do valor *NULL*.

No final, foi feita a conexão da ferramenta utilizada para a construção da interface de BI com a Base de Dados, MySQL, para desta forma permitir a visualização dos indicadores estatísticos de uma forma mais dinâmica e intuitiva. Esta foi possível utilizando a funcionalidade de conexão direta do Tableau, que permite, então a conexão do mesmo com o MySQL. Esta requer apenas que sejam introduzidas a porta, servidor, banco de dados, nome do utilizador e senha.

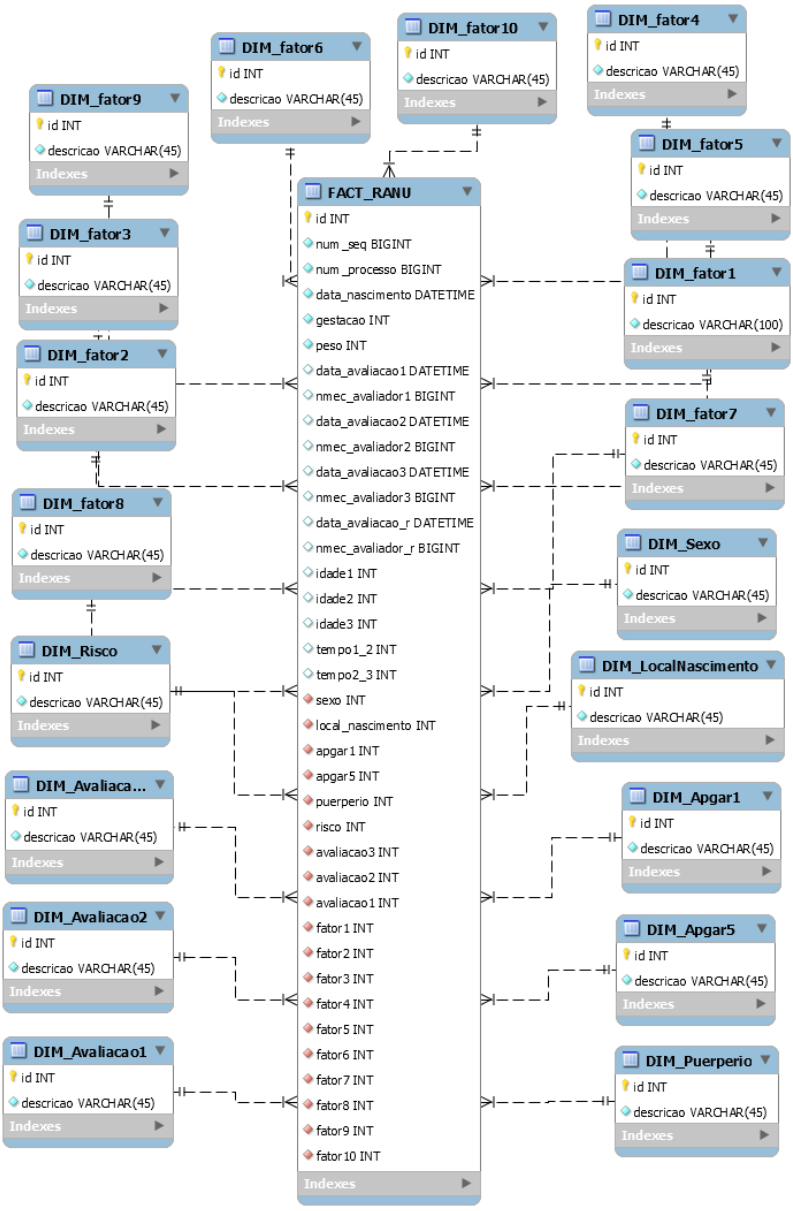

Fig. 4 Modelo do *Data Warehouse* em formato estrela.

5 Conclusão e Trabalhos Futuros

Os objetivos do trabalho foram cumpridos com sucesso, uma vez que foi possível construir um *Data Warehouse* com todas as informações essenciais que poderiam ser recolhidas pelo preenchimento do *template* elaborado pelo grupo. Além disto, também foi possível a construção de gráficos para a visualização de indicadores utilizando uma ferramenta de BI, conectada diretamente com o DW, e consequentemente, possibilitando o acesso aos dados contidos no mesmo. Estes indicadores são bastante relevantes para a fácil avaliação da qualidade de um serviço ou instituição e também possibilitam a realização de uma avaliação estatística dos dados essenciais. Com este projeto, concluiu-se, também, que o DW tem um papel extremamente relevante na análise dos dados ao facilitar a criação de *dashboards* de BI, uma vez que permite centralizar dados de diversas fontes e de forma consistente e estruturada.

Contudo, algumas melhorias poderiam ter sido efetuadas, como a implementação de uma alternativa para o tratamento dos valores nulos na transição dos valores do MongoDB para o MySQL, visto que isto se apresentou como uma impossibilidade. Outra proposta de melhoria seria a alteração de dados do tipo "Datetime" para "Date" no DW, onde a hora associada apresenta o formato "00:00:00", uma vez que esta informação é irrelevante. Por último, seria também relevante o desenvolvimento de mais indicadores de forma a haver uma maior amplitude na avaliação dos dados.

References

[1] Dolin, R.H., Alschuler, L., Beebe, C., Biron, P.V., Boyer, S.L., Essin, D., Kimber, E., Lincoln, T., Mattison, J.E.: The hl7 clinical document architecture. Journal of the American Medical Informatics Association **8**(6), 552–569 (2001)

[2] Beale, T.: Archetypes: Constraint-based domain models for future-proof information systems. In: OOPSLA 2002 Workshop on Behavioural Semantics, vol. 105, pp. 1–69 (2002). Citeseer

[3] Institute, N.C.: Eletronical Medical Record. https://www.cancer.gov/publications/dictionaries/cancer-terms/def/electronic-medical-record

[4] Delpierre, C., Cuzin, L., Fillaux, J., Alvarez, M., Massip, P., Lang, T.: A systematic review of computer-based patient record systems and quality of care: more randomized clinical trials or a broader approach? International Journal for Quality in Health Care **16**(5), 407–416 (2004)

[5] Zeng, Q., Cimino, J.J.: Evaluation of a system to identify relevant patient information and its impact on clinical information retrieval. In: Proceedings of the AMIA Symposium, p. 642 (1999). American Medical Informatics Association

[6] Garde, S., Knaup, P., Schuler, T., Hovenga, E., *et al.*: Can openehr archetypes empower multi-centre clinical research? Studies in health technology and informatics **116**, 971–976 (2005)

[7] Garde, S., Knaup, P., Hovenga, E.J., Heard, S.: Towards semantic interoperability for electronic health records. Methods of information in medicine **46**(03), 332–343 (2007)

[8] Xiao, L., Cousins, G., Courtney, B., Hederman, L., Fahey, T., Dimitrov, B.D.: Developing an electronic health record (ehr) for methadone treatment recording and decision support. BMC medical informatics and decision making **11**(1), 1–10 (2011)

[9] El-Sappagh, S.H.A., Hendawi, A.M.A., El Bastawissy, A.H.: A proposed model for data warehouse etl processes. Journal of King Saud University-Computer and Information Sciences **23**(2), 91–104 (2011)

[10] Hamoud, A., Hashim, A.S., Awadh, W.A.: Clinical data warehouse: a review. Iraqi Journal for Computers and Informatics **44**(2) (2018)

[11] Hyde, M.L., *et al.*: Newborn hearing screening programs: Overview. Journal of Otolaryngology **34**(2), 70 (2005)

[12] Nelson, H.D., Bougatsos, C., Nygren, P.: Universal newborn hearing screening: systematic review to update the 2001 us preven-tive services task force recommendation. Pediatrics **122**(1), 266–276 (2008)

[13] Monteiro, L., Calado, V.: Como organizar um rastreio universal da audição neonatal, pp. 27–38 (2001)

[14] Saúde, D.G: Boletim de Saúde Infantil e Juvenil. https://www.dgs.pt/paginas-de-sistema/saude-de-a-a-z/boletim-de-saude-infantil.aspx

[15] Hawley, G.: The use of an electronic health record (ehr) in a maternity shared-care environment (2015)

[16] Gavrilov, G., Vlahu-Gjorgievska, E., Trajkovik, V.: Healthcare data warehouse system supporting cross-border interoperability. Health informatics journal **26**(2), 1321–1332 (2020)

[17] Wulff, A., Haarbrandt, B., Tute, E., Marschollek, M., Beerbaum, P., Jack, T.: An interoperable clinical decision-support system for early detection of sirs in pediatric intensive care using openehr. Artificial Intelligence in Medicine **89**, 10–23 (2018)

[18] Scrum.org: What Is Scrum. https://www.scrum.org/resources/what-is-scrum

[19] Pahl, C., Zare, M., Nilashi, M., de Faria Borges, M., Weingaertner, D., Detschew, V., Supriyanto, E., Ibrahim, O.: Role of openehr as an open source solution for the regional modelling of patient data in obstetrics. J Biomed Inform. **55**, 174–187 (2015)

[20] Foundation, O.: What Is openEHR. https://www.openehr.org/about/what_is_openehr

[21] Min, L., Tian, Q., Lu, X., et al.: Modeling ehr with the openehr approach: an exploratory study in china. BMC Med Inform Decis (2018)

[22] Atalag, K., Yang, H., Tempero, E., Warren, J.: Evaluation of software maintainability with openehr–a comparison of architectures. Int J Med Inform. **83**(11), 849–859 (2014)

[23] Garde, S., Knaup, P., Hovenga, E., Heard, S.: Towards semantic interoperability for electronic health records–domain knowledge governance for open ehr archetypes. Inf Med. **46**(3), 332–343 (2007)

[24] Foundation, O.: openEHR Architecture Overview. https://www.openehr.org/releases/BASE/latest/docs/architecture_overview/architecture_overview.html

[25] Specifications, O.: Class Descriptions. https://specifications.openehr.org/

[26] Foundation, O.: openEHR Foundation Types Specification. https://specifications.openehr.org/releases/BASE/Release-1.1.0/foundation_types.html

[27] Foundation, O.: Data Types Information Model. https://specifications.openehr.org/releases/RM/latest/data_types.html

[28] SNOMED: Who We Are. https://www.snomed.org/snomed-international/who-we-are

[29] Manager, C.K.: About CKM. https://ckm.openehr.org/ckm/

[30] Better: Modelling Tool Available as a Free-of-charge Cloud Service for openEHR Archetype Development. https://blog.better.care

[31] MongoDB: What Is MongoDB? https://www.mongodb.com/pt-br/what-is-mongodb

[32] MySQL: What Is MySQL? https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html

[33] NHS: Your Baby's Health and Development Reviews. https://www.

nhs.uk/conditions/baby/babys-development/height-weight-and-reviews/baby-reviews/