UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

PEDRO HENRIQUE STOLARSKI AUCELI

YAGO ARAUJO GARCIA

CRIAÇÃO DE GRAFOS DE CONHECIMENTOS SEMÂNTICOS COM DADOS ABERTOS DA ÁREA DA SAÚDE DE CURITIBA: UM ESTUDO DE CASO COM ATENDIMENTOS COM DESFECHO EM ÓBITO

CURITIBA

2022

PEDRO HENRIQUE STOLARSKI AUCELI
YAGO ARAUJO GARCIA

CRIAÇÃO DE GRAFOS DE CONHECIMENTOS SEMÂNTICOS COM DADOS ABERTOS DA ÁREA DA SAÚDE DE CURITIBA: UM ESTUDO DE CASO COM ATENDIMENTOS COM DESFECHO EM ÓBITO

Creation of Semantic Knowledge Graph with Curitiba Health Open Aata: A Case Study of Care With a Death Outcome

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Bacharel em Sistemas de Informação do Curso de Bacharelado em Sistemas de Informação da Universidade Tecnológica Federal do Paraná.

Orientador: Profª. Drª. Rita Cristina Galarraga Berardi

CURITIBA

2022

PEDRO HENRIQUE STOLARSKI AUCELI
YAGO ARAUJO GARCIA

CRIAÇÃO DE GRAFOS DE CONHECIMENTOS SEMÂNTICOS COM DADOS ABERTOS DA ÁREA DA SAÚDE DE CURITIBA: UM ESTUDO DE CASO COM ATENDIMENTOS COM DESFECHO EM ÓBITO

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Bacharel em Sistemas de Informação do Curso de Bacharelado em Sistemas de Informação da Universidade Tecnológica Federal do Paraná.

Data de aprovação: 20/junho/2022

Rita Cristina Galarraga Berardi
Doutora em Ciência da Computação
Universidade Tecnológica Federal do Paraná

Nádia Puchalski Kozievitch
Doutora em Ciência da Computação
Universidade Tecnológica Federal do Paraná

Heloise Manica Paris Teixeira
Doutora em Engenharia e Gestão do Conhecimento
Universidade Estadual de Maringá

CURITIBA
2022

AGRADECIMENTOS

Certamente estes parágrafos não irão atender a todas as pessoas que fizeram parte dessa importante fase de nossas vidas. Portanto, desde já pedimos desculpas àquelas que não estão presentes entre essas palavras, mas elas podem estar certas que fazem parte do nosso pensamento e de nossa gratidão.

Agradeçemos a nossa orientadora Profa. Dra. Rita Cristina Galarraga Berardi, por toda a paciência, por toda a inspiração que nos gerou antes e durante a produção desse trabalho, e pela amizade e os ensinamentos que levaremos durante a vida.

Gostaríamos também de agradecer às nossas famílias, tanto aqueles que ainda estão entre nós quanto aqueles que partiram, por todo o amor e confiança. O autor Pedro gostaria de agradecer em particular o seu pai Luiz, sua mãe Claudia e sua irmã Vitoria. O autor Yago gostaria de agradecer ao seu Pai Alberto em específico, por toda a inspiração e incentivo durante todo o período da faculdade.

Aos nossos queridos amigos, por nos aturar e nos ajudar durante esse período difícil e por não deixar a gente desistir. O autor Pedro gostaria de agradecer em particular os seus amigos Lucas, Marcelo, Mauro, Paula e Thiago.

Aos nossos pets Bisteca, Nina e Olivia que nos animaram no dia a dia.

Gostaríamos também de agradecer ao Dr. Gino Pigatto Filho por tratar um dos autores desse trabalho durante várias complicações que este teve.

A todos os Professores que compartilharam seu conhecimento durante nossas graduações.

A Secretaria do Curso, pela cooperação.

Enfim, a todos os que por algum motivo contribuíram para a realização desta pesquisa.

RESUMO

Com a criação da Lei de Acesso à Informação (LAI) a abertura de dados governamentais passou a receber uma maior atenção do governo brasileiro, uma vez que a liberação de dados pertinentes à população virou obrigatória, permitindo que novos trabalhos e pesquisas sejam realizados. Além disso, ela prevê uma forma de padronizar os dados liberados pelo governo, que deveria facilitar os trabalhos com essas bases, mas isso não é totalmente aplicado, devido a cada instituição criar a sua própria estrutura para padronização dos dados. O intuito desse trabalho é integrar bases de dados referentes à saúde pública do município de Curitiba, para que novas informações possam ser adquiridas. A escolha da área de saúde foi devido a mesma ser uma área relevante para sociedade e rica em dados e informações, assim trazendo relevância para os resultados obtidos com esse trabalho. Os resultados obtidos a partir da conexão criada entre as bases permite a extração de informações correlacionadas, no caso desse trabalho, apresentamos algumas perguntas que só podem ser respondidas caso as bases estejam conectadas. Um exemplo de extração que pode ser obtida com a conexão das bases, é a relação de doenças que atingem certas regiões com o tratamento e distribuição de água. Conexões de dados desse tipo podem auxiliar especialistas na área a realizar outros estudos referente a essas doenças nessas regiões.

Palavras-chave: grafo de conhecimento semântico; dados abertos; dados conectados; dados de saúde pública; integração semântica.

ABSTRACT

With the creation of the Brazilian law Lei de Acesso à Informação (LAI), the opening of government data began to receive greater attention from the government, since the release of relevant data to the population became mandatory, allowing new work and research to be carried out. In addition, it provides a way to standardize the data released by the government, which should facilitate work with these databases, but this is not fully applied, as each institution creates its own structure for data standardization. The purpose of this work is to integrate databases related to public health in the city of Curitiba, so that new information can be acquired. The choice of the health area was due to it being a relevant area for society and rich in data and information, thus bringing relevance to the results obtained with this work. The results obtained from the connection created between the data bases allow the extraction of correlated information, in the case of this work, we present some questions that can only be answered if the data is connected. An example of extraction that can be obtained with the connected data, is the relation of diseases that affect certain regions with the treatment and distribution of water. Data connections of this type can help experts in the field to carry out further studies regarding these diseases in these regions.

Keywords: semantic knowledge graph; open data; connected data; public health data; semantic integration.

LISTA DE FIGURAS

Figura 1 – Camadas da Web Semântica 15
Figura 2 – Classificação de Ontologias 19
Figura 3 – Tripla Genérica 20
Figura 4 – Exemplo de Inferência com RDFS 21
Figura 5 – Framework LDM 29
Figura 6 – Tarefa de Linkagem 30
Figura 7 – Framework Ontop 32
Figura 8 – Interface *Query* em SQL 32
Figura 9 – Mapeamento de Variáveis da *Query* para a Ontologia 33
Figura 10 – Diferença de Camadas entre *Frameworks* 33
Figura 11 – Conjuntos das Bases com seus Dados 35
Figura 12 – Exemplos de UIDs (Perfis) Gerados 39
Figura 13 – Resultado da Materialização das Triplas 42
Figura 14 – Tipos de abastecimento nos perfis 45
Figura 15 – Tipos de tratamento de água feito pelos perfis 46
Figura 16 – 5 Doenças mais comuns em perfis que não tratam sua água proveniente de rede pública 47
Figura 17 – Nível de escolaridade de perfis com registro de óbito que não tratam a água proveniente da rede pública 48
Figura 18 – Bairros com maior taxa de mortes em perfis que não tratam a água proveniente da rede pública 49

LISTA DE TABELAS

Tabela 1 – Particularidade entre *Frameworks* 34
Tabela 2 – Quantidade de Registros das Bases 36
Tabela 3 – Colunas Utilizadas no Perfil 37
Tabela 4 – Colunas E-Saude 37
Tabela 5 – Colunas SIHSUS 37
Tabela 6 – Colunas SIM 37
Tabela 7 – Ciclo de diagnósticos de um perfil 44

LISTA DE ABREVIATURAS E SIGLAS

Siglas

CID	Classificação Internacional de Doenças
CSV	*Comma-Separated Values*
DBC	*Database Container*
DL	*Description Logics*
FOAF	*Friend of a Friend*
HTML	*Hypertext Markup Language*
HTTP	*Hypertext Transfer Protocol*
IA	Inteligência Artificial
IBGE	Instituto Brasileiro de Geografia e Estatistica
LAI	Lei de Acesso a Informação
LDM	*Linked Data Mashup*
LOD	*Linked Open Data*
OWL	*Web Ontology Language*
PHO	*Public Healthcare Ontology*
RC	Representação de Conhecimento
RDF	*Resource Description Framework*
RDFS	*Resource Description Framework Schema*
SIACE	Sistemas Integrados de Atendimento de Consultas Especializadas
SIHSUS	Sistema de informações Hospitalares do Sistema Único de Saúde
SIM	Sistema de Informações de Mortalidade
SQL	*Structured Query Language*
UID	*Unique Identifier*

UMS	Unidades Municipais de Saúde
UPA	Unidade de Pronto Atendimento
URI	*Uniform Resource Identifier*
UTF-8	*8-bit Unicode Transformation Format*
XML	*Extensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO 13

2 REFERENCIAL TEÓRICO 15

2.1 Web Semântica 15

2.2 Dados Abertos e Conectados 16

2.3 Ontologia 17

2.4 Vocabulário 18

2.5 Representação de Grafos de Conhecimento Semânticos 19

2.5.1 RDF 20

2.5.2 RDFS 20

2.5.3 OWL 21

3 TRABALHOS CORRELATOS 23

4 METODOLOGIA 26

5 FERRAMENTAS 28

5.1 Framework LDM 28

5.2 Framework Ontop 31

5.3 Análise dos *Frameworks* 33

6 CRIAÇÃO DO GRAFO DE CONHECIMENTO SEMÂNTICO 35

6.1 Descrição das bases de dados utilizadas 35

6.2 Limpeza e Normalização 36

6.3 Criação do perfil 38

6.4 Construção da Ontologia 39

6.5 Mapeamento dos dados 42

7 AVALIAÇÃO E RESULTADOS 43

8 EVOLUÇÃO DO GRAFO DE CONHECIMENTO SEMÂNTICO 50

8.1 Análise das Bases 2020 50

8.2 Atualização das Bases 51

9 CONSIDERAÇÕES FINAIS 52

REFERÊNCIAS 54

ANEXO A CÓDIGO PYTHON UTILIZADO PARA LIMPEZA E NORMALIZAÇÃO DAS BASES 58

ANEXO B MAPEAMENTOS DE CLASSES 61

1 INTRODUÇÃO

Embora o governo brasileiro já disponibilizasse de forma aberta seus dados relacionados com os setores públicos para serem utilizados por quaisquer interessados, foi somente após a criação da Lei de Acesso a Informação (LAI)[1] em 2011 que a abertura de dados passou a receber maior atenção. A LAI foi criada com o intuito de ampliar e padronizar os dados disponibilizados pelas instâncias governamentais brasileiras (CIVIL, 2018).

Embora a LAI tenha sido criada com o propósito de padronizar os dados públicos, nem sempre os dados disponibilizados pelas instâncias estão de acordo com as regras de divulgação estipuladas, podendo causar transtornos para os interessados em utilizar esses dados. Por exemplo, caso se busque efetuar algum tipo de análise que necessite de informações ou dados de múltiplas bases, a falta de padronização será um desafio. As bases podem estar em formatos diferentes, a granularidade dos dados podem ser diferentes, ou até mesmo a informação que se pode obter delas é diferente, mesmo que elas pertençam ao mesmo domínio.

O maior problema das bases de dados abertos atualmente é a falta de padronização dos dados. Um problema que pode ser facilmente encontrado pela falta de padronização é o caso de dados que passam a mesma informação, mas estruturados de maneiras diferentes, de tal forma que para uma máquina esses dados são diferentes, é evidente como a falta de padronização é um obstáculo a ser enfrentado (JUNIOR *et al.*, 2018). Suponha que existam três bases com informações sobre a nacionalidade de pessoas, mas em uma base essa informação é adquirida como “brasileiro”, em outra como “Br”, e na última é obtida somente o país de origem, para esse exemplo será “Brasil”. Caso se tente efetuar uma análise dessas bases utilizando a informação de nacionalidade, se faz necessário toda uma etapa de tratamento desses dados, já que para uma máquina as informações extraídas delas são diferentes.

Para minimizar problemas como o apresentado, uma solução efetiva é a integração das bases de dados utilizando uma ontologia, criando assim um *Semantic Knowledge Graphs* (Grafo de Conhecimento Semântico). A ontologia é uma forma de padronização dos dados levando em consideração sua semântica, em outras palavras, o significado dos dados dentro de seu domínio. Uma ontologia é um modelo de dados utilizado para representar os conceitos, ou classes, de um domínio e seus relacionamentos (GUARINO; OBERLE; STAAB, 2009).

O objetivo principal deste trabalho é integrar bases de dados abertos de saúde por meio de ontologias com o intuito de se criar um grafo de conhecimento semântico e assim homogeneizar os dados, possibilitando análises com mais confiabilidade. O grafo de conhecimento especifico deste trabalho foi construído com base em duas perguntas de interesse que auxiliam a delimitar o escopo dos dados utilizados. As perguntas de interesse fizeram parte dos motivos para escolha do domínio das bases. Essas devem ser respondidas utlizando o grafo de conhecimento semântico, que deve facilitar a extração dessas informações. Com isso será

[1] http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/l12527.htm

possível observar os benefícios práticos da integração de bases de dados abertos por meio de ontologias e assim motivar o desenvolvimento de novas metodologias.

1. É possível analisar a trajetória de pacientes que passam por uma unidade pública de saúde e na sequência são internadas em hospitais e vêm a óbito?

2. É possível encontrar uma relação entre pacientes que são encaminhados para hospitais e acabam falecendo com o tipo de distribuição e tratamento de água de suas residências?

Existe a necessidade de utilizar bases de dados que contenham informações que sejam úteis para a sociedade, e como a área médica tem grande influência sobre os cidadãos, essa configuração sustenta o motivo de sua escolha (JUNIOR *et al.*, 2018).

Além disso, as ontologias devem ser criadas para domínios específicos, o que reforça a escolha das bases que serão utilizadas neste trabalho, uma vez que todas pertencem ao domínio de saúde pública (TOMA *et al.*, 2013).

A motivação de se conectar as bases é facilitar a geração de informações e conhecimentos sobre elas. Com as bases conectadas, pesquisas analíticas sobre o estado de saúde, por exemplo, poderiam ser feitas mais facilmente, o que ajuda na geração de informações sobre a saúde da cidade. Outro fator motivacional são possíveis inovações que podem ser realizadas com a utilização dessas informações, que para o caso de recursos públicos, é possível gerar uma melhoria na prestação de serviços. Esse tipo de iniciativa motiva as cidades a continuar a manutenção e distribuição desses dados, possibilitando um maior volume de trabalhos na área de dados abertos.

2 REFERENCIAL TEÓRICO

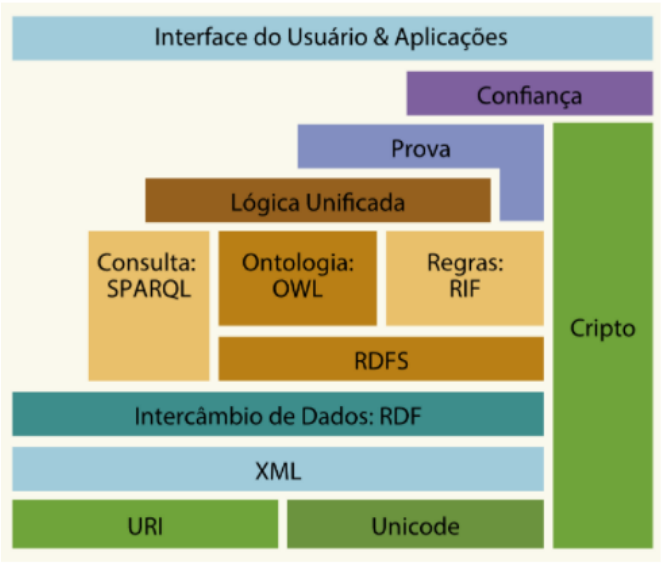
Para realização do projeto foram necessários conhecimentos dos conceitos por trás da área de Dados Abertos e Conectados, dentre eles as definições de ontologia, vocabulário, grafos de conhecimento semântico e até as diferentes formas de representação de conhecimento que podem ser utilizadas para estruturar os dados.

2.1 Web Semântica

A Web Semântica estrutura uma Web de Dados, com os mais variados tipos de dados que alguém pode disponibilizar. Contudo, para torná-la uma realidade, é necessário um grande volume de dados publicados na web de forma aberta, mas, além disso, eles devem estar em um formato padrão, acessível e gerenciável por ferramentas da Web Semântica. Ela não só precisa de acesso aos dados, como também as suas relações, para assim criar uma Web de Dados. Essas relações entre as bases de dados publicadas na Web são conhecidas como Dados Conectados (W3C, 2015a).

O termo Web Semântica surgiu pela primeira vez no artigo “Web Semântica: Um novo formato de conteúdo para a Web que tem significado para computadores vai iniciar uma revolução de novas oportunidades” (BERNERS-LEE; HENDLER; LASSILA, 2001). Neste artigo ele explica como a Web Semântica utiliza recursos de diversas áreas para alcançar seus objetivos, como agentes inteligentes e representação de conhecimento da Inteligência Artificial, por exemplo. Ainda neste artigo, ele também propôs um modelo de camadas para melhor exemplificar a Web Semântica, como visto na Figura 1.

Figura 1 – Camadas da Web Semântica



Fonte: (ISOTANI; BITTENCOURT, 2015).

O principal objetivo da Web Semântica é tornar legíveis os dados provenientes da web, tanto para humanos como para máquinas, e com isso, permitir que essas máquinas executem atividades na web, que antes só eram possíveis de serem efetuadas por humanos (ISOTANI; BITTENCOURT, 2015).

2.2 Dados Abertos e Conectados

A definição de dados abertos, segundo a *Open Definition*, são “dados que podem ser livremente utilizados, modificados e redistribuídos por qualquer um com qualquer propósito”. Aprofundando um pouco mais nessa definição, de acordo com (FOUNDATION, 2010), dados abertos podem ser resumidos em 3 pontos importantes.

- **Disponibilização e Acesso:** Os dados devem estar disponibilizados como um todo, preferencialmente possíveis de serem adquiridos pela web, e por um custo não maior que seu custo de reprodução. Eles também devem estar disponíveis em um formato conveniente e modificável.
- **Re-uso e Redistribuição:** Os dados devem estar disponíveis de forma que seja permitido o reuso e a redistribuição dos mesmos, incluindo a junção com outras bases de dados.
- **Participação Universal:** Todos devem ser capazes de utilizar, reutilizar e redistribuir os dados. Não deve haver nenhum tipo de restrição que impeça pessoas ou grupos de utilizar esses dados.

O conceito de Dados conectados, de acordo com Bizer apud (ISOTANI; BITTENCOURT, 2015), pode ser definido como um conjunto de boas práticas para publicar e conectar conjuntos de dados estruturados na web, com intuito de se criar uma “Web de Dados”. Tecnologias como *Hypertext Transfer Protocol* (HTTP) e *Uniform Resource Identifier* (URI) são utilizadas com o objetivo de permitir que os dados publicados possam ser lidos por pessoas e máquinas.

A Web de Dados é baseada em um conjunto de padrões, ou tecnologias, utilizadas na publicação e estruturação dos dados. Ela utiliza tecnologias como: URIs para identificação universal, HTTP para acesso universal, *Resource Description Framework* (RDF) como modelo padrão de representação de dados, e SPARQL como linguagem de consulta para acesso aos dados. São esses padrões que permitem que qualquer pessoa publique os dados de forma que possam ser lidos por humanos ou máquinas (ISOTANI; BITTENCOURT, 2015).

É possível traçar um paralelo entre a Web de Documentos e a Web de Dados. A Web de Documentos utiliza o padrão *Hypertext Markup Language* (HTML) como formato padrão para representação de conteúdo, ou seja, é por ele que é feito o acesso aos dados, enquanto na Web de Dados o acesso é feito através do modelo RDF. Além disso, na Web de Documentos são utilizados *hyperlinks* para transitar entre páginas, por exemplo o site www.wikipedia.org, no

qual todos os seus documentos são vinculados por *hyperlinks* colocados em palavras chave, enquanto na Web de Dados são utilizados *links* RDF para acessar dados de diversas fontes.

Vale lembrar que um dado aberto não necessariamente está conectado ou é conectável e vice-versa. Em 2006 foi proposto por Tim Berners-Lee o “Sistema de 5 Estrelas”, para classificar o grau de abertura dos dados. Nesse sistema quanto mais aberto e conectado um dado se encontra, mais estrelas ele recebe:

1. Disponível na internet em qualquer formato.
2. Disponível na internet de maneira estruturada.
3. Disponível na internet de maneira estruturada e em formato não proprietário.
4. Estar dentro dos padrões estabelecidos pela (W3C, 2017).
5. Conectar seus dados a outras bases.

Da 1ª à 3ª estrela, o objetivo é avaliar se a base está aberta, se seus dados estão disponíveis na internet de forma acessível para qualquer usuário e com licença de livre uso. A 4ª estrela é para que a base seja algo conectável, ou seja, outras bases de dados podem se conectar a ela. A 5ª e última estrela é para que ela esteja conectada a outras bases de dados.

2.3 Ontologia

A palavra ontologia tem origem grega a partir da junção de duas palavras: *ontos* (ser) e *logia* (estudos). Ela é um campo de estudo da filosofia, dentro do campo da metafísica, que abrange a natureza do ser, da existência e da própria realidade. Ontologias vêm sendo utilizadas no âmbito da computação como estrutura de Representação de Conhecimento (RC), uma sub-área da Inteligência Artificial (IA), e para criação de Grafos de Conhecimento Semântico. Uma ontologia é uma teoria representativa dos principais fatos e regras que governam parte da realidade, com fins computacionais (ALMEIDA, 2014).

De acordo com GRÜBER, 1993 apud (OLIVEIRA; LOSCIO, 2008), uma ontologia é uma especificação explícita e formal de uma conceitualização compartilhada. Entende-se por conceitualização os conceitos, objetos, entidades e relacionamentos de certo domínio. A conceitualização é o que dará o valor semântico para os objetos e entidades a partir de seus relacionamentos. No âmbito da computação, ontologias são compostas pelos seguintes itens:

- **Indivíduos:** Objeto básico da ontologia (representa um único ser).
- **Classes:** Representa um conjunto de objetos. Uma classe pode conter vários indivíduos e/ou outras classes.

- **Atributos:** São propriedades que os objetos podem ter e compartilhar entre si. Atributos podem tanto estar vinculados a um ou mais indivíduos como a uma ou mais classes.

- **Relacionamentos:** Representam as relações que os objetos podem ter entre si. As relações podem existir entre quaisquer objetos, sejam esses indivíduos ou classes.

Imagine a seguinte afirmação: ***"Todas as pessoas saindo de casa estão infectadas com covid19 e podem transmitir para outras pessoas"***. A frase pode ser quebrada para adquirir duas classes, **pessoa** e **doença**, um indivíduo explicíto pertencente a classe doença, **covid19**, um atributo, **sair**, e dois relacionamentos, **infecta** e **transmite**. Ela pode ser representada matematicamente da seguinte forma:

$$\forall \mathbf{x(Pessoa(x)} \wedge \mathbf{sair(x)} \rightarrow \forall \mathbf{y(Doença(covid19(y))} \wedge \mathbf{infecta(y,x)} \rightarrow \exists \mathbf{z(Pessoa(z)} \wedge \mathbf{transmite(x,z,y))))}$$

Essa representação é chamada de axioma e é desta forma que a ontologia carrega essa afirmação sobre o universo que ela domina. Axiomas são como regras para os relacionamentos, elas modelam restrições e regras inerentes às instâncias (SANTAREM; CONEGLIAN, 2016). Em resumo, com esse axioma criado e adicionado a uma ontologia, isso passará a ser uma regra do mundo criado por ela, ou seja, quaisquer pessoas que se encaixem nesta regra, a ontologia passa a afirmar que as mesmas estão com infectadas com covid19.

Quando se está criando uma ontologia o primeiro passo é decidir sobre qual domínio, área ou mundo ela será modelada, tendo isso definido o próximo passo é decidir o que a ontologia deve ser capaz de representar dentro desse domínio escolhido. O escopo serve para limitar a representação dentro da ontologia impedindo que ela se expanda muito e não tenha que carregar representações desnecessárias, que fogem do cenário definido.

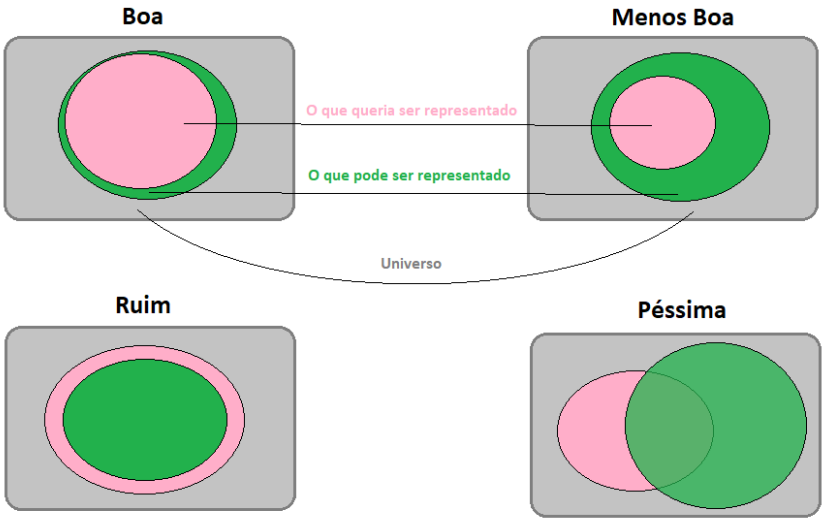
Como pode ser visto na Figura 2, uma ontologia pode ser classificada como sendo *good* (boa), *less good* (menos boa), *bad* (ruim) e *even worse* (péssima) de acordo com o quanto ela consegue representar dentro do esperado.

2.4 Vocabulário

Para resolver problemas envoltos de *Data Integration* (Integração de Dados), as ontologias são utilizadas como provedoras de um vocabulário comum para as aplicações [...] (KEET, 2020).

Vocabulários são definidos por termos que melhor descrevem o domínio que está sendo estudado (TOMA *et al.*, 2013). Um vocabulário é um documento apontado por uma URI. Esse documento é onde estão definidas as classes e as propriedades do vocabulário, a partir de termos que são retirados do domínio escolhido. Para referenciar essas classes e propriedades

Figura 2 – Classificação de Ontologias



Fonte: Elaborada pelos Autores.

é necessário efetuar a concatenação da URI do vocabulário com o nome da respectiva classe ou propriedade (ISOTANI; BITTENCOURT, 2015).

O vocabulário *Friend of a Friend* (FOAF)[1], por exemplo, é um projeto criado com o objetivo de conectar pessoas e informações na *Web* (BRICKLEY; MILLER, 2000). A URI "http://xmlns.com/foaf/0.1/" é o que permite a utilização desse vocabulário, e para chamar uma de suas propriedades, como "name" por exemplo, basta concatenar o nome da mesma ao final da chamada base, assim criando a URI da propriedade em si, "http://xmlns.com/foaf/0.1/name".

Conceitos são as representações dos termos utilizados pelo domínio, definidas a partir de atributos que as compõem, que podem ou não ser outras classes, e os relacionamentos são regras que permitem relacioná-las, dando valor semântico aos dados. Para a construção de ontologias são necessários conceitos de vocabulário e dicionário de dados, pois é por meio de sua utilização que são adquiridos os termos e as palavras-chave para definir as classes de uma ontologia.

2.5 Representação de Grafos de Conhecimento Semânticos

A modelagem de um grafo de conhecimento semântico começa pela representação dos dados em RDF e outras camadas semânticas como RDFS e OWL. Essa modelagem se faz necessária para tornar os dados conectados, deixando os dados padronizados de acordo com a Web de Dados.

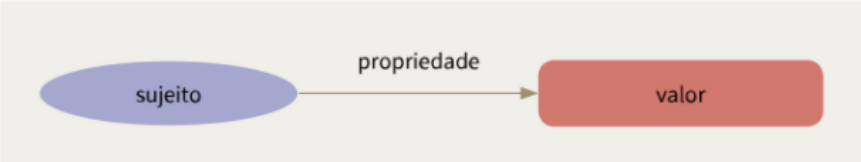
[1] http://xmlns.com/foaf/spec/

2.5.1 RDF

A RDF é uma série de especificações criadas pela (W3C, 2017), com a qual é possível modelar um domínio a partir da obtenção de seus conhecimentos em partes e da definição de regras sobre o significado ou a semântica dessas partes. Arquivos RDF são modelos que utilizam vocabulários baseados em URIs e sintaxes baseadas em *Extensible Markup Language* (XML).

O modelo RDF é formado pelo conjunto de declarações predicado, sujeito e objeto e essa combinação é conhecida como tripla. O objeto e o sujeito são recursos sobre os quais a RDF permite fazer afirmações. Uma afirmação na RDF expressa a relação do sujeito com o objeto, sendo que o predicado representa a natureza da relação, e é chamada em RDF de propriedade (W3C, 2010). O objeto pode ser tanto outro sujeito como também um literal, definindo uma propriedade para o recurso. A Figura 3 é um exemplo genérico de como as triplas funcionam.

Figura 3 – Tripla Genérica



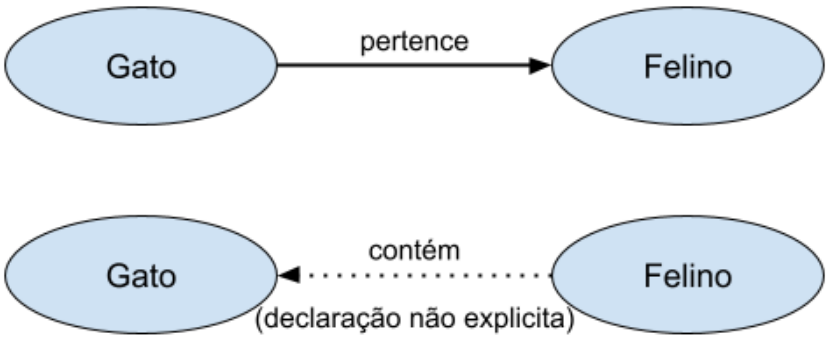
Fonte: (ISOTANI; BITTENCOURT, 2015).

2.5.2 RDFS

A *Resource Description Framework Schema* (RDFS) é um vocabulário que estende a RDF e permite especificar características que agregam semântica aos dados. Com ela é possível especificar que uma URI é uma propriedade de um recurso ou que um recurso pertence a uma determinada classe. Esse modelo utiliza o conceito de classes para criar categorias sobre os recursos, podendo até mesmo criar hierarquias de classes e subclasses e hierarquias de propriedades e subpropriedades. Além disso é possível criar restrições de tipos sobre os sujeitos e objetos das triplas, por meio da especificação de domínios e contradomínios para cada um dos tipos (W3C, 2015b).

Essas propriedades da RDFS são o que permite que se faça inferências sobre os dados, assim não sendo necessário a declaração explícita de todas as relações e nem de todos os atributos. A Figura 4 apresenta um exemplo construído para demonstrar essas propriedades. No exemplo apresentado foram declaradas duas classes, uma chamada “Felino” e outra chamada “Gato”. Ainda nesse examplo foi declarada uma relação entre elas chamada "Pertence", para indicar que a classe "Gato"está contida dentro da classe "Felino".

Figura 4 – Exemplo de Inferência com RDFS



Fonte: Elaborada pelos Autores.

No momento da definição da relação entre as classes apresentadas no exemplo da Figura 4 é possível declarar seu domínio e contradomínio, sendo essas as propriedades da relação que irão permitir realizar as inferências entre as classes. Sendo a relação “Pertence” da classe “Gato” e como contradomínio a classe “Felino”. Desta forma cria-se a relação “Gato pertence a Felino”, mas além disso, com essa relação criada, é possível inferir que “Felino contém Gato”, sem a necessidade de declarar uma nova tripla para explicitar essa relação. Dessa forma, é possível adquirir mais triplas do que realmente declarado.

Um outro tipo de inferência possível de ser feita é utilizando os conceitos de subclasse e subpropriedade. No momento em que um recurso é definido como sendo de uma subclasse, ele é automaticamente inferido como sendo da classe superior também, o mesmo ocorre para subpropriedades.

2.5.3 OWL

A *Web Ontology Language* (OWL) é uma linguagem da web semântica criada para representar de forma rica e complexa o conhecimento sobre os objetos, grupos de objetos e o relacionamento entre eles (W3C, 2012).

A OWL oferece uma série muito ampla de restrições para serem feitas sobre as triplas e diversos construtores que permitem a construção de triplas mais complexas. A base da OWL vem das *Description Logics* (DL), linguagem de representação de conhecimento muito utilizada na construção de ontologias.

DLs são sintaxes que permitem descrever as relações entre as entidades do domínio de interesse. As DLs tem três tipos de entidades: *Concepts* (Conceitos), *Roles* (Regras) e *Individual Names* (Nomes individuais). Conceitos se referem a conjuntos de indivíduos, regras a conjuntos de relações entre os indivíduos, e nomes individuais a indivíduos únicos no domínio. Uma ontologia baseada em DL consiste na definição de um conjunto de estados, os quais são chamados

de axiomas. Esses axiomas são apenas partes da situação descrita em uma ontologia e eles podem estar em diferentes estados do mundo em que ela consiste (KEET, 2020).

Axiomas podem criar relações entre indivíduos e/ou conceitos. Seguem alguns exemplos de axiomas (KEET, 2020):

- O axioma **Mother(Julia)** se refere ao indivíduo chamado **Julia** e explicita que ele é uma instância do conceito **Mother**.
- O axioma **parentOf(Julia, John)** se refere a dois indivíduos, um chamado **Julia** e outro chamado **John**, e explicita que eles são relacionados pela denotação **parentsOf**.
- O axioma **Mother** $\sqsubseteq$ **Parent** se refere a dois conceitos e explicita a relação de sub conceito entre eles, no caso **Mother** é um sub conceito de **Parent**.
- O axioma **parentOf** $\sqsubseteq$ **ancestorOf** se refere a duas regras e explicita a relação de sub regra entre elas, dizendo que **parentsOf** é uma sub regra de **ancestorOf**.

Na OWL baseada em DLs, os conceitos são as classes e as regras são as propriedades dos objetos adquiridos por ela, sendo uma extensão da RDF e RDF-S. Além disso, é adicionado o parâmetro data *type* ou *data property* para os atributos. Como o próprio nome sugere, ela permite definir o tipo do valor de um objeto.

3 TRABALHOS CORRELATOS

Existem diversas áreas que tangenciam este trabalho, porém, os artigos de maior relevância são aqueles que abrangem alguma dessas áreas de conhecimento: ontologias, dados abertos, integração de dados, dados médicos.

O trabalho de Queiroz, Lino e Motta (2016) é sobre uma ontologia para preservar a privacidade. Esse artigo se relaciona de várias formas, por ser uma pesquisa voltada à criação de uma ontologia e também por falar da necessidade de anonimizar dados pessoais antes de poder abrir uma base na internet, que é um dos problema que este trabalho tenta resolver por meio da criação dos perfis que serão explicados na Seção 4. O experimento do artigo é a criação de uma ontologia, onde o autor busca a interoperabilidade dos sistemas para a anonimização dos dados, uma vez que cada sistema tem a sua definição semântica para cada dado. Isso acaba se relacionando diretamente com este trabalho, uma vez que é prevista a interoperabilidade entre as bases contendo os dados de saúde de Curitiba.

No trabalho de Lopes, Vidal e Oliveira (2016) é criado um framework de ontologias com o intuito de integrar bases de dados abertos referentes à saúde, para se obter informações que apenas uma base não poderia suprir. No caso deste artigo a informação que os autores buscavam é se existe uma relação da causa de morte de recém-nascidos com possíveis hábitos de suas respectivas mães. Dentro desse artigo é descrita uma visão para integração de bases, a *Linked Data Mashup* (LDM), e como atingir essa visão dentro do framework, que foi o resultado obtido. A LDM acaba se diferenciando de outras visões pelo fato de utilizar múltiplas camadas ontológicas, uma que está ligada à base de dados, e outra que está ligada à camada de visão exportada. Nessa camada podem ser encontradas as ontologias, e um conjunto de regras que ajuda a mapear os conceitos da ontologia da base de dados para com a desta camada. Sendo que ambas são subconjuntos de uma grande ontologia, onde é feita toda a integração das bases. Esse artigo tangencia o presente trabalho nas suas áreas de aplicação, uma vez que é discutido sobre ontologias, dados abertos conectados e dados de saúde. Além disso, o artigo traz conceitos que foram utilizados neste trabalho, como é o caso da LDM. Ele também auxilia no método para a avaliação da ontologia deste trabalho, uma vez que é apresentado o método de questões de competência, que são perguntas criadas com o intuito de avaliar se a integração foi suficiente ou não.

No trabalho de Pereira, Wassermann e Salvador (2017) é discutido sobre a integração de bases de saúde pública, e como essa integração auxiliaria na geração de novas informações. Segundo Pereira, Wassermann e Salvador (2017) apud SMS-SP (2012) ”A falta de informações leva à subnotificação, o que impede a elaboração de indicadores demográficos e de saúde acurados, bem como o melhor desenvolvimento de sistemas de vigilância e estabelecimento de políticas de saúde”. Ou seja, esse artigo pretende resolver, por meio da integração das bases de dados de saúde, alguns problemas existentes, mais especificamente problemas em pacientes com anomalias congênitas. Para a integração, o artigo compara dois frameworks,

o LDM que já foi apresentado no estado da arte do presente trabalho pelos autores Lopes, Vidal e Oliveira (2016), e o framework Ontop. Ontop utiliza como base o programa Protégé (PEREIRA; WASSERMANN; SALVADOR, 2017) para a construção da ontologia juntamente do plugin Ontop para o mapeamento das bases de dados. Esse artigo traz relevância por ser uma pesquisa na área de dados abertos, dados de saúde e por tratar de integração de dados por meio de ontologias. Além disso, os frameworks avaliados nele foram analisados para verificar a viabilidade para o caso do presente trabalho.

No artigo de Klímek *et al.* (2018), foi apresentado uma metodologia para a criação de dados no formato de *Linked Open Data*, dos dados governamentais da República Tcheca referentes às pensões presentes no país. O objetivo inicial era a abertura dos dados, porém após discussões foi decidido que os dados seriam disponibilizados no formato de dados abertos conectados. Para isso foi necessário a extração dos dados, e a transformação desses para o formato RDF. Os autores não tiveram problemas para conectar os dados, uma vez que foi utilizado o código oficial de zona, o nosso CEP, para realizar a conexão entre classes similares. Como o resultado final foi um portal de dados abertos, esse também foi avaliado, através da usabilidade, portabilidade, disponibilidade e performance. Para finalizar foram criados 2 casos de uso, onde 2 usuários utilizavam os dados disponibilizados e os conectavam com outras bases de dados abertos conectados, demonstrando a facilidade para a conexão, e a riqueza das informações geradas a partir da integração. Esse artigo acaba se relacionando com o presente trabalho, uma vez que foram utilizadas ontologias e métodos para a triplificação de dados abertos para obter uma base no formato de triplas, que pode ser conectada a outras, agregando valor à utilização de ontologias para a integração de dados abertos.

No artigo de Nunes e Berardi (2020), foram explicitados os passos de uma futura pesquisa, essa que tem como premissa a integração de bases de dados heterogêneos através de um modelo semântico. Os dados dessa pesquisa são da área médica, com o foco em obesidade e cirurgias bariátricas, os dados são provenientes do setor privado. O foco deste trabalho é a pesquisa sobre a utilização de ontologias para integração de dados médicos, e a formulação de uma metodologia que será utilizada em uma pesquisa futura. Esse acaba se relacionando com o presente trabalho por utilizar dados da área médica, por se tratar de uma integração de bases de dados heterogêneos através de um modelo semântico e pela metodologia ter pontos similares, como o método para criação da ontologia e como avaliar a sua eficiência. Apesar de não terem sido apresentados resultados, essas semelhanças agregam valor, uma vez que valida partes da metodologia apresentada no atual trabalho.

A pesquisa realizada por Rolim *et al.* (2020) apresenta um modelo para a integração de dados abertos do domínio da Saúde através de grafos de conhecimento, com o foco em duas bases: uma com os registros de recém nascidos e outra com o registro de óbitos. Os autores explicam toda a metodologia necessária para se obter o grafo de conhecimento, alguns passos e ferramentas tangenciam a metodologia apresentada no presente trabalho. Alguns desses foram: a criação de uma ontologia para descrever a realidade dos dados, criação de mapea-

mentos semânticos dos dados presentes em um banco de dados relacional para o formato de triplas utilizando o Ontop como ferramenta, e a criação de questões de competência para avaliar se a integração foi bem sucedida. Esse trabalho também demonstra a criação de uma ontologia utilizada para descrever grafos de conhecimento, essa que foi responsável por uma virtualização dos dados triplificados. Isso ocorre através de vários mapeamentos semânticos e regras definidas na ontologia. Essa virtualização exclui a necessidade de uma triplestore para armazenamento de dados triplificados, por serem utilizadas ferramentas para a tradução de *queries* em SPARQL para uma consulta em SQL e os mapeamentos que são responsáveis pelas regras que irão gerar o grafo de conhecimento relativo a consulta SPARQL.

No trabalho de Iliadis *et al.* (2021) foi discutido sobre a dificuldade de centralização e integração de dados referentes à pandemia do covid-19, e como a utilização de grafos de conhecimento semântico é uma tecnologia viável para realizar essa integração. No artigo foram apresentados 2 exemplos de grafos onde foram integradas bases de múltiplos países, demonstrando como a utilização desses possibilita a integração dos dados. Os autores também demonstram uma preocupação sobre a manutenção dos grafos, uma vez que, os termos utilizados nas definições das propriedades do grafo, podem sofrer alterações com o decorrer de novas descobertas, o que poderia inviabilizar eles. Também foi discutido que o desenvolvimento de grafos de conhecimento devem ser realizados tentando englobar diferentes linguagens e quaisquer aspectos sociais referentes aos dados presentes neles.

No artigo de Wu (2021) é apresentado um modelo de grafo de conhecimento semântico para integrar e analisar dados de diferentes domínios, com o intuito de encontrar relações entre pessoas que foram infectadas com o covid-19, os locais que frequentaram e as interações interpessoais que tiveram. O autor utilizou diversas ferramentas, e algoritmos de *machine learning* para a validação e obtenção dos dados necessários, que foram sobre: pessoas, diagnósticos, transporte e dados geo-espaciais. Com todos os dados presentes no grafo de conhecimento foram realizados cálculos complexos para encontrar o relacionamento entre esses dados, assim possibilitando a visualização da trajetória de um pessoa, ou seja, os lugares que visitou, qual foi o meio de transporte e até se interagiu com outras pessoas. Essa pesquisa se relaciona com o atual trabalho por utilizar técnicas similares, como a utilização de grafos de conhecimento semântico para obter uma integração de dados heterogêneos e por ter utilizado dados da área da saúde.

4 METODOLOGIA

Como a construção do grafo de conhecimento semântico está baseado na necessidade de informações vindas da área da saúde, foram identificadas 3 bases de dados que juntas podem permitir a extração de informações para responder as questões de competência:

1. É possível analisar a trajetória de pacientes que passam por uma unidade pública de saúde e na sequência são internadas em hospitais e vêm a óbito?
2. É possível encontrar uma relação entre os pacientes que são encaminhados para hospitais e acabam falecendo, com o abastecimento e tratamento da água de suas residências?

Assim, as 3 bases de dados que conectadas permitem a extração de informações para responder a essas perguntas são:

- E-Saude, disponibilizada no formato *Comma-Separated Values* (CSV) pela Prefeitura de Curitiba (CURITIBA, 2020). Ela é construída utilizando dados referentes as Unidade de Pronto Atendimento (UPA), Unidades Municipais de Saúde (UMS), e dos Sistemas Integrados de Atendimento de Consultas Especializadas (SIACE), com uma granularidade trimestral.
- Sistema de informações Hospitalares do Sistema Único de Saúde (SIHSUS), disponibilizada no formato *Database Container* (DBC) pelo ministério de saúde brasileiro (SAúDE, 2017), obtida pelo portal do DATASUS. Ela é construída utilizando os dados referentes aos atendimentos feitos em hospitais, com uma granularidade de disponibilização mensal e por estado brasileiro.
- Sistema de Informações de Mortalidade (SIM), também disponibilizada no formato DBC pelo ministério de saúde brasileiro (SAúDE, 2017), obtida pelo portal do DATASUS. Ela é construída utilizando os dados referentes aos óbitos de pacientes, com uma granularidade anual e por estado brasileiro.

Após as definições já realizadas e a obtenção das bases que serão utilizadas nesse projeto, foram feitos os seguintes passos para alcançar os resultados esperados:

- Definir Escopos: Os primeiros passos, já realizados, envolve a definição do escopo dos dados, ou seja, as cidades que eles abrangem e a definição do domínio das bases.
- Coleta dos Dados: A coleta dos dados pode ser dividida nas subtarefas seguintes:
  - Obter as bases de dados: Etapa já realizada utilizando os portais das bases escolhidas com base nas perguntas de competência.

  - Converter as bases de DBC para CSV: As bases do SIHSUS e do SIM são adquiridas no formato *Database Container*, sendo necessário uma conversão das bases para CSV, padronizando os formatos entre todas as bases.
  - Aplicar filtros: Necessário aplicar filtros nas bases do SIHSUS e do SIM para manter somentes dados pertinentes ao município de Curitiba.

- Análise exploratória dos dados: É feita uma análise exploratória das três bases, para encontrar possíveis problemas que podem afetar os resultados esperados e para verificar quais dados seriam pertinentes para o trabalho. Problemas como linhas que não contenham algum dado utilizado na criação do perfil são localizados nessa etapa.

- Limpar e reduzir a granularidade dos dados: A limpeza começa pela remoção de duplicatas, de dados e colunas inconsistentes, de acentuação e normalização dos dados. A redução seguiu duas etapas, a remoção de registros das bases referentes aos hospitais e óbitos que não fossem do Paraná, e a remoção de registros de todas as bases que com certeza não teriam como estar em todas elas.

- Criar perfis com base na análise feita: Como as bases selecionadas não contêm nenhum tipo de ID para os registros, uma vez que as mesmas estão anonimizadas, foi necessário a criação dos perfis para serem utilizados como *Unique Identifier* (UID) para cada registro das bases. Dessa forma, eles podem ser utilizados para relacionar os registros entre as bases e ajudar na integração delas. Para definição do perfil foram utilizadas informações em comum do paciente que existem em todas as bases. As colunas selecionadas foram: sexo, data de nascimento e cidade do paciente.

- Criar a ontologia referente aos dados encontrados pelos perfis de pessoas das bases: É utilizada a ferramenta Protégé[1], com os vocabulários definidos por este trabalho, para conseguir conectar as três bases.

- Escolher um framework para mapear os dados: Foram analisados, testados e comparados os frameworks LDM e Ontop para realizar o mapeamento semântico dos dados.

- Realizar a integração semântica dos dados com a criação os indivíduos do grafo de conhecimento.

- Avaliar se a conexão permitiu responder as perguntas: Para a avaliação da ontologia serão utilizadas as questões de competência, ou seja, perguntas que deverão ser respondidas com a integração das bases.

[1] https://protege.stanford.edu/

5 FERRAMENTAS

Para manusear os arquivos das bases foi escolhida a linguagem de programação Python em conjunto da biblioteca Pandas, que possui funções de leitura e manipulação de arquivos CSVs (TEAM, 2021). Essa biblioteca permite fazer diversas operações sobre os arquivos CSVs, como cortes de linhas e colunas. Ela trata os arquivos carregados como se fossem tabelas em um banco de dados, ou seja, ela permite fazer consultas dos dados contidos nos arquivos utilizando funções que simulam uma query. Funções como *Loc*, utilizado para filtrar os dados a partir de uma condição, *Group By*, para agrupamento dos dados, e *Sort Values*, para ordenação dos dados, são somente alguns exemplos de operações que a biblioteca permite.

Além dessas ferramentas também foi utilizado o software Protégé. Ele é um programa baseado em Java para prototipação e desenvolvimento de ontologias (MUSEN, 2015). Esse programa foi utilizada para a criação da ontologia do trabalho, e foi escolhido por ser uma ferramenta *open source* com uma comunidade ativa, além de conter múltiplos *plugins* que facilitam a utilização de bases de dados.

Para fazer o mapeamento semântico dos dados contidos nos datasets de acordo com a ontologia criada, foram analisados 2 *frameworks* diferentes para realizar tal função. Esses foram descritos de acordo com um teste realizado, onde foram utilizados 2 registros de cada base para avaliar a viabilidade de ambos, e sua descrição será feita nas próximas duas subseções.

5.1 Framework LDM

Esse framework contém passos definidos para se alcançar a visão *Linked Data Mashup* (LDM) (LOPES; VIDAL; OLIVEIRA, 2016; VIDAL *et al.*, 2015; PEREIRA; WASSERMANN; SALVADOR, 2017; SCHULTZ *et al.*, 2012). São necessários 5 passos divididos entre 4 camadas.

- Extração dos dados abertos contidos nos arquivos CSV;
- Mapeamento e triplificação dos dados utilizando a ontologia criada neste trabalho;
- Criação dos *links owl:sameAs* entre as classes equivalentes dos arquivos do segundo passo;
- Fusão dos *links owl:sameAs* e dos arquivos contendo as triplas;
- Exportação do arquivo final para a inserção em uma *Triple Store*.

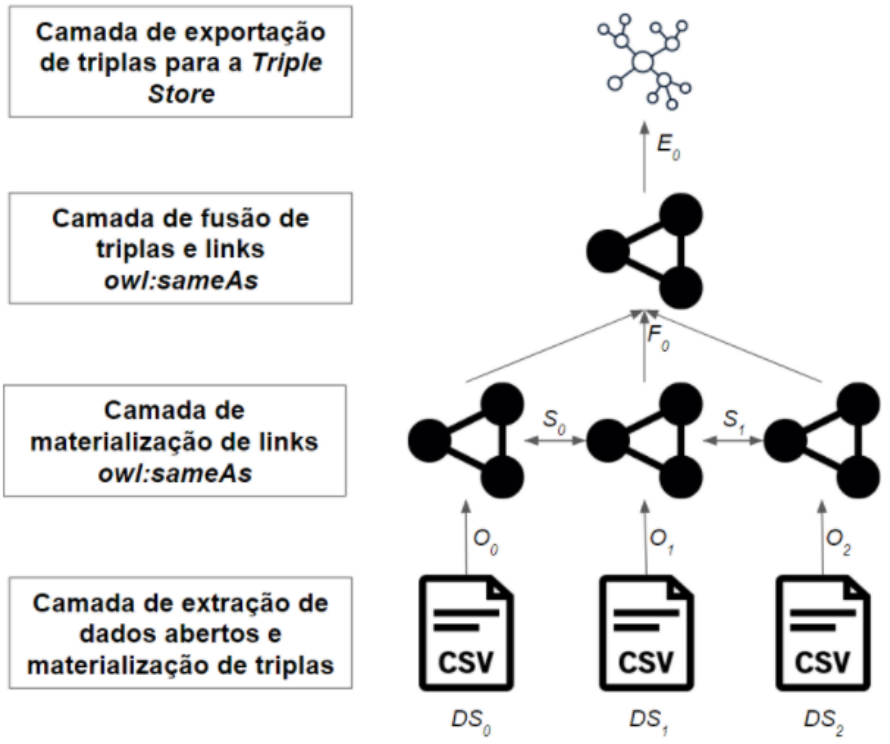
Enquanto que as quatro camadas foram:

- Camada de extração de dados abertos e materialização de triplas, que abrange o primeiro e o segundo passo;
- Camada de materialização de *links owl:sameAs*, que engloba o terceiro passo;

- Camada de fusão de triplas e *links owl:sameAs*, para o quarto passo;
- Camada de exportação de triplas para a *Triple Store*, para o quinto e último passo.

A arquitetura criada para a análise do *framework* LDM, adaptado para a realidade deste trabalho, foi resumida na Figura 5, que contém as 4 camadas previamente explicadas, representadas na parte esquerda da figura 5. Enquanto que os conjuntos são as entradas e saídas dos 5 passos explicitados anteriormente.

Figura 5 – Framework LDM



Fonte: Elaborada pelos Autores.

Para a primeira camada foi utilizada a ferramenta LOD-Refine[1], que é uma versão específica de outra ferramenta, o Open-Refine[2]. As duas foram criadas com o intuito de manusear dados em arquivos de vários formatos, com funções para alterar os dados presentes, como por exemplo mudar o formato de uma coluna de Int para String. A versão específica foi criada para trabalhos voltados ao *Linked Open Data* (LOD), e contém funções que permitem o mapeamento dos dados contidos em colunas (CSV) para os relacionamentos, tipos de dados, e URIs definidos na ontologia. No final é possível exportar os dados em um arquivo no formato *"turtle"*, contendo todas as triplas geradas através do mapeamento realizado na ferramenta. O conjunto DS={DS_0, DS_1, DS_2}, representado na Figura 5, são os datasets resultantes do primeiro passo do framework, sendo eles: "E-Saude", "SIHSUS" e "SIM", respectivamente. Eles se encontravam no formato CSV, e foram os dados de entrada para o segundo passo. Já o conjunto O={O_0,

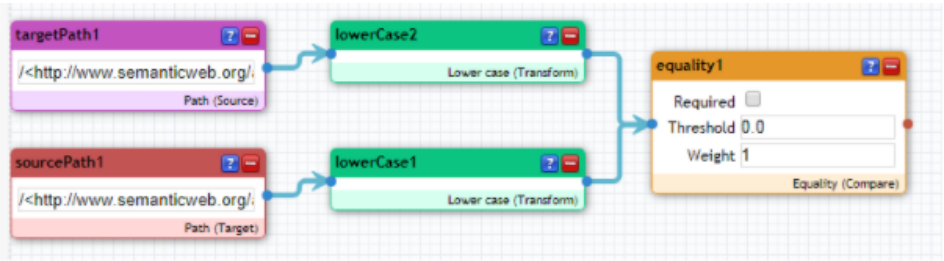
[1] https://github.com/sparkica/LODRefine
[2] http://openrefine.org/

O_1, O_2} é a saída do segundo passo, e são referentes aos arquivos com as triplas geradas, no formato *"turtle"*.

Para a segunda camada foi utilizada a ferramenta Silk (VOLZ *et al.*, 2009), para a criação dos links *owl:sameAs*, o terceiro passo do *framework*. Esses são responsáveis por criar uma equivalência entre as classes de ontologias e criar novas associações (DING *et al.*, 2010). O predicado *owl:sameAs* é de extrema importância para a inferência de classes, porém também se deve ter cuidado com o uso errôneo dele, podendo gerar inferências erradas (JAFFRI; GLASER; MILLARD, 2007). A ferramenta disponibiliza uma interface gráfica com várias funções, podendo ser funções de transformação dos dados, como tokenização ou transformar em lowercase. Além disso, também são disponibilizados comparadores, como igualdade e Jaccard, e até agregadores, como média, min e max. Todas essas funções fazem parte da tarefa de linkagem, e são necessárias para aumentar a acurácia dos links, ou seja, que o mesmo seja feito entre classes iguais. A acurácia é importante para evitar que links errados sejam criados, que foi o problema apresentado por (JAFFRI; GLASER; MILLARD, 2007).

Um exemplo da tarefa de linkagem é apresentada na Figura 6, onde os "Paths" são as classes equivalentes nos diferentes arquivos, *"lowerCase"* é uma função de transformação e *"equality"* é uma função de comparação. Para esta última função é possível verificar a presença de variáveis *threshold* e *weight*, a primeira define o quão restrito os links devem ser, por exemplo classes 100% iguais. Enquanto que a variável weight só será utilizada caso o retorno da função *"equality"* seja utilizada em um agregador, que não foi o caso no exemplo dado. Ao final são gerados arquivos contendo todos os links *owl:sameAs* entre as classes, representado pelo conjunto {S} na Figura 5.

Figura 6 – Tarefa de Linkagem



Fonte: Elaborada pelos Autores.

Para a terceira camada, e o quarto passo, foi utilizada a ferramenta Sieve (MENDES; MÜHLEISEN; BIZER, 2012), responsável por realizar a fusão entre os diferentes arquivos gerados no terceiro passo. Que foram os arquivos de triplas da primeira camada, conjunto {O}, e os links *sameAs* da segunda camada, conjunto {S}, além de retirar a redundância de classes geradas por essa fusão. A fusão junta todos os arquivos, que podem estar em formatos diferentes, para um arquivo final no formato "n-quad", representado pelo conjunto {E} na Figura 5.

Para a última camada, e passo, o arquivo da camada posterior foi exportado para uma *Triple Store*, para ser possível a criação de queries em SPARQL. A *Triple Store* escolhida foi

o GraphDB[3] que foi escolhida por aceitar arquivos em diversos formatos, por ter uma interface gráfica de fácil uso e conter algumas ferramentas de visualização das *queries* que facilitam o entendimento das triplas resultantes.

5.2 Framework Ontop

Esse *framework* foi baseado nos trabalhos de (PEREIRA; WASSERMANN; SALVADOR, 2017; CALVANESE *et al.*, 2015). O *framework* Ontop contém passos para realizar o mapeamento de um banco de dados relacional para uma ontologia, que é uma de suas diferenças quando comparado com o LDM. Foram necessários 4 passos divididos entre 4 camadas para a construção deste, sendo que os passos foram:

- Extração dos dados abertos contidos nos arquivos CSV;
- Inserção dos dados em um banco de dados relacional PostgreSQL;
- Criação do mapeamento semântico, que é a atribuição dos dados contidos nas colunas do banco de dados relacional para as classes, propriedades e relacionamentos da ontologia;
- Materialização das triplas geradas usando o mapeamento;
- Exportação do arquivo final para a inserção em uma *Triple Store*;

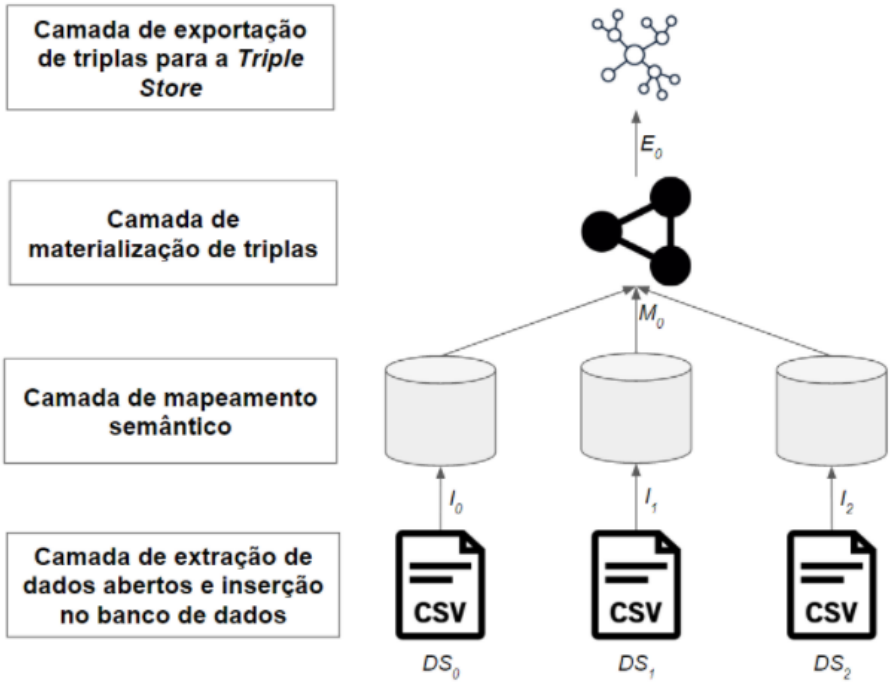
As quatro camadas definidas foram:

- Extração de dados abertos e inserção no banco de dados, onde a extração é a saída do primeiro passo, que serve de entrada para o segundo passo, a inserção no banco;
- Mapeamento semântico, que utiliza os dados do segundo passo para ser possível a realização do terceiro;
- Materialização de triplas, que engloba o quarto passo;
- Exportação de triplas para a *Triple Store*, que utiliza os dados do quarto passo para alimentar a *Triple Store*, que é o quinto passo.

Este *framework* foi resumido na Figura 7, que contém todas as camadas previamente apresentadas. Onde o conjunto DS={DS_0, DS_1, DS_2} é referente aos datasets “E-Saude”, “SIH-SUS” e “SIM”, todos em formato CSV, I={I_0, I_1, I_2} é a inserção dos dados abertos no banco de dados relacional, M={M_0,} é o mapeamento semântico e E={E_0,} é a exportação das triplas para a Triple Store.

[3] http://graphdb.ontotext.com/

Figura 7 – Framework Ontop

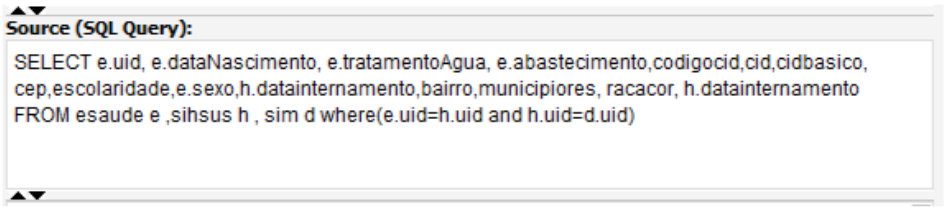


Fonte: Elaborada pelos Autores.

Para a primeira camada foi utilizado o banco de dados relacional PostgreSQL[4], com a ferramenta PgAdmin4[5]. Foram criados 3 bancos de dados, “E-Saude”, “SIHSUS”, “SIM”, para inserir os dados contidos nos arquivos CSV.

Para a segunda camada foi utilizada a ferramenta Protégé, que além de ser responsável pela criação da ontologia, contém o *plugin* Ontop que permite a criação dos mapeamentos na própria interface do programa. Esses são feitos por meio de consultas *Structured Query Language* (SQL), onde o retorno da *query* é armazenado em uma variável que será apontada para uma URI, um relacionamento ou propriedade de dados. Na Figura 8 é possível visualizar a interface da consulta SQL que deverá ser feita, enquanto que na Figura 9 é apresentado o mapeamento das variáveis da query para a ontologia.

Figura 8 – Interface *Query* em SQL

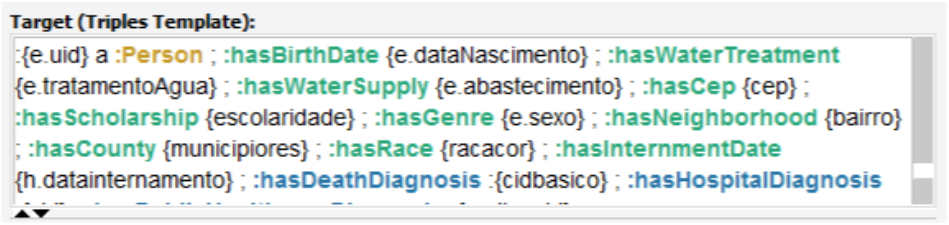


Fonte: Elaborada pelos Autores.

Para a terceira camada foi utilizado o Protégé junto do *plugin* Ontop, porém agora com a função de materializar triplas. Essa gera um arquivo com todas as triplas resultantes do ma-

[4] https://www.postgresql.org/
[5] https://www.pgadmin.org/download/pgadmin-4-windows/

Figura 9 – Mapeamento de Variáveis da *Query* para a Ontologia



Fonte: Elaborada pelos Autores.

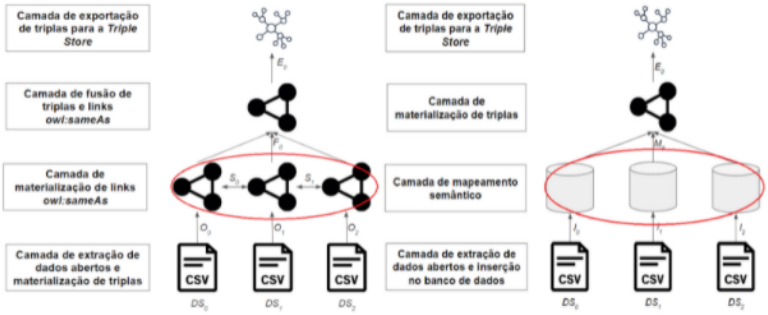
peamento. Além do Protégé, é possível realizar a mesma função utilizando o *command line interface* do Ontop (OntopCLI[6]), que utiliza a ontologia e o mapeamento para triplificar os dados contidos nos bancos de dados.

Para a última camada, similar ao *framework* LDM, o arquivo gerado na camada posterior foi exportado e inserido em uma *Triple Store*. Novamente foi utilizado o GraphDB, para ser possível a criação de *queries* em SPARQL, e com isso obter as informações necessárias.

5.3 Análise dos *Frameworks*

Os dois *frameworks* analisados tiveram resultados positivos, uma vez que foi possível realizar todos os passos descritos em cada um, e ao final foi possível a criação de *queries* em SPARQL e com isso extrair informações da integração. Porém cada um tem sua particularidade, a maior delas é a segunda camada de baixo para cima, onde o Ontop necessita de um banco de dados relacional, enquanto a LDM utiliza arquivos de triplas nos formatos *"turtle"*, isso é destacado na Figura 10.

Figura 10 – Diferença de Camadas entre *Frameworks*



Fonte: Elaborada pelos Autores.

Essa diferença da segunda camada é o que torna os *frameworks* únicos, cada um com as suas vantagens e desvantagens. O Ontop gera uma separação a partir da segunda camada, onde os dados são mapeados do banco de dados para a ontologia. Essa traz grandes benefícios por padronizar o mapeamento, ou seja, caso os dados presentes no banco sejam alterados não é necessário refazer o processo de mapeamento. E no caso de serem feitas alterações no banco

[6] https://github.com/ontop/ontop/wiki/OntopCLI

em si, só será necessário alterar a *query* em SQL. O LDM realiza o mapeamento diretamente no arquivo CSV, e caso esse seja alterado o processo deverá ser feito do começo, o que é uma grande desvantagem devido ao retrabalho.

A maior diferença entre os *frameworks* se dá pela segunda camada, porém essa não é a única, na Tabela 1 é possível verificar todas as diferenças entre os *frameworks*.

Tabela 1 – Particularidade entre *Frameworks*

Particularidades	**Ontop**	**LDM**
Forma dos predicados resultantes nas triplas	Mais expressivos e específicos	Mais genéricos
Padronização e limpeza dos dados	Mais complexa por causa do banco de dados relacional	Mais simples devido a ferramenta LOD-Refine
Quantidade de ferramentas necessárias	Apenas 1	3

A primeira particularidade pode ser vista através dos predicados de tipo (*rdf:Type*), onde no Ontop a estrutura dos dados é específico. Como por exemplo uma classe é definida pelo predicado *owl:Class* enquanto que no LDM seria definido como *rdf:Property*, um termo mais genérico.

Para a segunda particularidade, a complexidade atribuída ao Ontop se deve à formatação dos dados do banco relacional, onde existem padrões, como o *8-bit Unicode Transformation Format* (UTF-8), que exigem uma limpeza e organização maior dos dados. Enquanto que no LDM não existem padrões de formatação de dados, e as próprias ferramentas disponibilizam recursos para organizar e limpar os dados.

Levando em consideração as vantagens e desvantagens de cada *framework*, o Ontop foi o escolhido para realizar a integração das bases deste trabalho. Os fatores de escolha foram: a maior especificidade apresentada nas triplas resultantes da materialização, e por gerar a padronização do mapeamento sem a necessidade de o refazer. O primeiro fator foi importante para evitar a criação de possíveis inferências erradas, e para facilitar a reutilização da ontologia em outras bases de dados. Já o segundo fator foi importante devido a todas alterações dos dados que foram realizadas durante o percurso deste trabalho, assim evitando o retrabalho do mapeamento.

6 CRIAÇÃO DO GRAFO DE CONHECIMENTO SEMÂNTICO

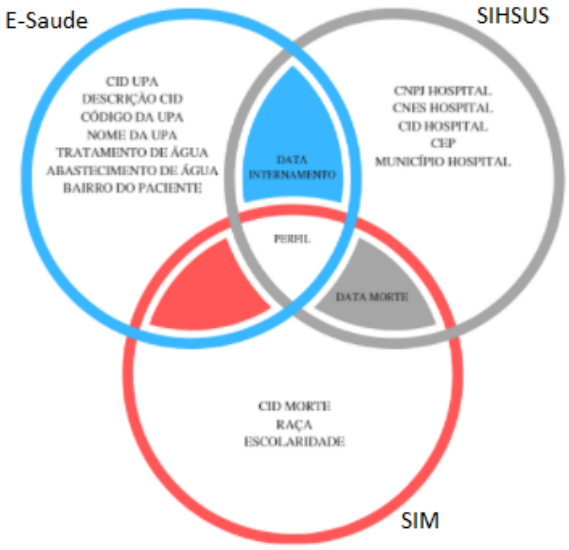
O grafo de conhecimento semântico foi criado em etapas, começando pela limpeza e normalização das bases, removendo colunas consideradas não relevantes, linhas com dados obrigatórios em branco, padronização de dados, entre outras etapas. Em seguida, foi realizada a criação da nova coluna chamada Perfil, que é utilizada de base para integração das bases, para então finalmente ser criada a ontologia que irá integrá-las.

A normalização e limpeza dos dados das bases foi necessário para padronizar as informações e as formas de representação contidas nas mesmas, como identificar sexo com as letras M para masculino e F para feminino por exemplo, ou a remoção de linhas com dados em brancos em colunas consideradas obrigatórias para o projeto, colunas utilizadas na construção do perfil por exemplo. Em seguida, com os dados normalizados e limpos, e a coluna do perfil criada, foi possível realizar a criação da ontologia sem grandes dificuldades, utilizando a nova coluna de base para integrar as bases.

6.1 Descrição das bases de dados utilizadas

A Figura 11 utiliza conjuntos para demonstrar quais dados foram selecionados para serem mantidos, além das informações utilizadas no perfil, em azul a base “E-Saude”, em cinza “SIHSUS” e em vermelho “SIM”. Lembrando que a construção do perfil, utilizando dados em comum, foi a forma encontrada para integrar as bases, e o mesmo será melhor detalhado na subseção 6.3.

Figura 11 – Conjuntos das Bases com seus Dados



Fonte: Elaborada pelos Autores.

Os arquivos disponibilizados pelo SUS estão no formato DBC, mas como foram selecionadas ferramentas que melhor trabalham com arquivos CSV, foi utilizada a ferramenta TabWin415[1], disponibilizada pelo próprio SUS, para converter esses arquivos de DBC para CSV.

O período das bases escolhidas foi 2017 devido a base do “SIM” estar disponível somente até esse ano, no momento em que este projeto foi desenvolvido. Além disso, como a base do “SIM” disponibiliza seus dados anualmente, sendo a base com maior granularidade, foi decidido que todas as bases deveriam conter dados anuais. Para as bases do “SIHSUS” e do “E-Saude” foi necessário criar um arquivo de cada concatenando seus registros, devido a essas bases serem disponibilizadas em períodos menores que um ano, mês a mês e trimestral, respectivamente.

Para a construção da base do “E-Saude” foram utilizados quatro CSV’s, todas contendo dados referentes aos últimos 3 meses antes de serem liberados. Já para a construção da base do “SIHSUS” foram utilizados doze CVS’s, cada um referente a um único mês. Para base do “SIM” não foi necessário nenhum tipo de junção, pois a mesma já é disponibilizada em um único arquivo com todos os dados do ano.

Após as bases serem construídas, elas passaram por uma série de limpezas e cortes para padronizar e normalizar seus dados. A Tabela 2 exibe a quantidade total de registros antes e após esses tratamentos.

Tabela 2 – Quantidade de Registros das Bases

Bases	Antes dos Tratamentos	Após os Tratamentos
E-Saude	3.278.505	24.938
SIHSUS	846.947	17.620
SIM	71.574	41.318

6.2 Limpeza e Normalização

Nessa seção será descrito todas as etapas realizadas para limpeza e normalização dos dados realizadas, os detalhes do código podem ser vistos no Anexo A. Todo o código do trabalho foi realizado utilizando a linguagem de programação Python.

O tratamento das bases começou pela remoção de colunas indesejadas ou irrelevantes para este projeto, ou seja, colunas que não agregam nenhum valor semântico relevante para a análise proposta neste projeto. A seleção de colunas para essa remoção usou como critério colunas em branco ou que continham um único valor se repetindo por todas as suas linhas.

Após a limpeza, foi realizado uma seleção manual das colunas que restaram, de forma a manter somente as colunas consideradas mais relevantes para o projeto. Dentre as colunas mantidas em cada base, três são obrigatórias para a criação da coluna Perfil e foram mantidas em todas, apresentadas na tabela 3

[1] http://www2.datasus.gov.br/

Tabela 3 – Colunas Utilizadas no Perfil

Coluna	Descrição
Sexo	Sexo do Paciente (Masculino ou Feminino)
Data de Nascimento	Data de Nascimento do Paciente
Cidade	Cidade de Nascimento do Paciente

A base do “E-Saude” continha 37 colunas, feita a remoção sobraram 36, e dentre essas foram selecionadas 11 colunas, sendo 3 pertencentes ao perfil, sobram as 8 apresentadas na tabela 4.

Tabela 4 – Colunas E-Saude

Coluna	Descrição
Código do CID	Código da Classificação Internacional de Doença (CID)
Descrição do CID	Descrição do Diagnóstico
Código da Unidade	Código da Unidade de Atendimento
Descrição da Unidade	Nome da Unidade de Atendimento
Tratamento no Domicílio	Tipo de Tratamento de Água no domicílio
Abastecimento	Tipo de Abastecimento de Água no domicílio
Bairro	Bairro da moradia do paciente
Data do Internamento	Data do Internamento do paciente

A base do “SIHSUS” continha 113 colunas, feita a remoção sobraram 67, e dentre essas foram selecionadas 10 colunas, sendo 3 pertencentes ao perfil, sobram as 7 apresetandas na tabela 5.

Tabela 5 – Colunas SIHSUS

Coluna	Descrição
CGC_HOSP	CNPJ do Hospital
CNES	Código Cadastro Nacional de Estabelecimentos de Saúde (CNES) do hospital
DIAG_PRINC	Código do diagnóstico
CEP	CEP do paciente
MUNIC_MOV	Município do Hospital
DT_INTER	Data de internação do paciente
DT_SAIDA	Data de saída do paciente

A base do “SIM” continha 83 colunas, feita a primeira remoção sobraram 82, dentre essas foram selecionadas 7 colunas, sendo 3 pertencentes ao perfil, sobram as 4 apresentadas na tabela 6.

Tabela 6 – Colunas SIM

Coluna	Descrição
CAUSABAS	Causa básica da morte, conforme Classificação Internacional de Doença (CID)
RACACOR	Raça/Cor do paciente (Branca, Preta, Amarela, Parda, Indígena)
ESC	Escolaridade, calculada em quantidade anos de estudo (1 a 3 anos, 4 a 7 anos, etc.)
DTOBITO	Data do óbito do paciente

Após ser efetuado a limpeza das colunas, também foi necessário uma remoção de linhas das bases. O primeiro critério foi remover quaisquer linhas que não continham informação

preenchida em alguma das colunas obrigatórias para a criação do Perfil, ou seja, linhas sem informação nas colunas ”Sexo”, ”Data de Nascimento” e/ou ”Cidade”. O segundo critério foi a remoção de registros que não teriam como existir nas 3 bases. Na base do ”E-Saude” foram removidos registros nos quais não ocorreu internamento, e portanto não existem na base do ”SIHSUS”, na base do ”SIHSUS” foram removidos registros que não houve óbito, e portanto não existem na base do ”SIM”, e por último foram removidos registros das bases do ”SIHSUS” e do ”SIM” que fossem de cidades diferentes das que existem na base do ”E-Saude”.

Com a limpeza das colunas e das linhas finalizadas, foi necessário efetuar um tratamento nos dados, uma remoção de acentos e caracteres especiais de todos. As colunas utlizadas para a criação do ”Perfil” também passaram por uma padronização, para garantir que o perfil fosse o mesmo em todas as bases.

Na coluna “Sexo” foi necessário que todas as bases utilizassem o mesmo padrão de identificação. Para este trabalho foi optado por utilizar os valores ‘M’ para masculino e ‘F’ para feminino, padrão já utilizado pelo “E-Saude”. Foi utilizado o auxílio dicionário de dados de cada base para realização dessa etapa. A base do ”SIHSUS” necessitou de uma análise um pouco mais profunda, uma vez que seu dicionário está incompleto. Foi realizado uma busca de doenças exclusivamente masculinas e femininas para extrair a informação de quais valores indica cada sexo.

Os dados da coluna “Data de Nascimento” foram modificados para o formato ‘yyyymmdd’, onde ‘yyyy’ é o ano, ‘mm’ é o mês e o ‘dd’ é o dia. Esse é o formato utilizado pela base do “SIHSUS”, escolhido para ser utilizado como padrão.

Para a coluna “Cidade” foi optado por utilizar o código disponibilizado pelo Instituto Brasileiro de Geografia e Estatistica (IBGE) de cada cidade como padrão. Para alterar a formatação da coluna foi utilizado um CSV auxiliar adquirido pela (CENSEC, 2018), com as informações do código de cada cidade. Essa formatação foi necessária somente na base do “E-Saude”, pois é a única que utiliza o nome das cidades por extenso ao invés do código propriamente dito.

6.3 Criação do perfil

Finalizada a limpeza e padronização das colunas “Sexo”, “Data de Nascimento” e “Cidade”, a construção do perfil se tornou relativamente simples. Nas 3 bases foi criada uma coluna chamada UID, a qual foi preenchida utilizando a concatenação das informações contidas nessas 3 colunas, criando um perfil por paciente.

Esse UID gerado utilizando essa técnica se torna eficaz uma vez que permite localizar fluxos de pacientes, mas sem identificar o paciente em si, assim não prejudicando a anonimização existente nos dados. Além disso, ela permite extrair informações interessantes, como os fluxos mais comuns para certos perfis de pacientes ou até mesmo doenças mais comuns entre eles, por exemplo.

Como foram utilizadas somente 3 colunas para a geração dos UIDs, a variação dos mesmos foi prejudicada, a unicidade dos dados criados poderia ser maior caso fossem utilizadas mais colunas para a construção dos dados, em contrapartida, isso permitiu acumular mais dados por perfil abrangendo mais informações contidas em cada um. A construção dos UIDs foi realizada utilizando o código a seguir, no qual é realizado a concatenação dos dados contidos nas colunas selecionadas.

```
1 data['UID'] = data['Sexo'] + data['Data de Nascimento'].astype(
    str) + data['Cidade'].astype(str)
```

A função “astype” foi utilizada para converter os dados das colunas “Data de Nascimento” e “Cidade” de numéricas para texto, assim permitindo que suas informações fossem concatenadas com a coluna “Sexo”, assim formando os dados da coluna UID, como os exemplos exibidos na figura 12.

Figura 12 – Exemplos de UIDs (Perfis) Gerados

```
ESaude:  M20070620410690
SIHSUS:  F19570726411990
SIM:  M19460808410620
```

Fonte: Elaborada pelos Autores.

6.4 Construção da Ontologia

A ontologia criada neste trabalho foi intitulada de “PHO”, um acrônimo para *Public Healthcare Ontology*[2], essa contém 3 classes e 2 sub-classes, 3 propriedades de objetos e 17 propriedades de dados. Todas essas propriedades foram criadas com o intuito de realizar a integração e possibilitar responder às questões de competência apresentadas no começo deste trabalho. Cada propriedade foi mapeada de uma base, e se relaciona com as outras propriedades presentes na ontologia.

Mapeamento de Classes:

- **Person:** Classe que especifica uma pessoa na ontologia, é instanciada por meio do UID e é mapeada utilizando qualquer uma das bases nas colunas “UID”.
- **Diagnosis:** Classe que especifica um diagnóstico, é instanciada por meio de uma Classificação Internacional de Doenças (CID), é mapeada nas três bases, na coluna

[2] https://github.com/YagoGarcia/Public-Healthcare-Ontology

“DIAG_PRINC” na base ”SIHSUS”, na coluna “CAUSABAS” em “SIM”, e “Codigo do CID” em “E-saude”.

- **Place:** Classe genérica para ser utilizada nas sub-classes **Upa** e **Hospital**.

Mapeamento de Sub-Classes:

- **Upa:** Sub-Classe de **Place**, especifica uma UPA, uma UMS ou uma unidade do SIACE, é instanciada por meio do nome da unidade e é mapeada da base “E-Saude” na coluna “Descrição da unidade”
- **Hospital:** Sub-Classe de **Place**, especifica um hospital, é instanciada por meio do CNES do hospital e é mapeada da base “SIHSUS” na coluna “CNES”

Mapeamento de Propriedades de Objetos:

- **hasDeathDiagnosis:** Cria um relacionamento entre a classe **Person** e a classe **Diagnosis**, é mapeada da base ”SIM” na coluna "CAUSABAS"
- **hasHospitalDiagnosis:** Cria um relacionamento entre a classe **Person** e a classe **Diagnosis**, é mapeada da base ”SIHSUS” na coluna "DIAG_PRINC"
- **hasPublicHealthcareDiagnosis:** Cria um relacionamento entre a classe **Person** e a classe **Diagnosis**, é mapeada da base ”E-Saude” na coluna "Codigo do CID"

Mapeamento de Propriedades de Dados:

- **hasBirthDate:** Cria um relacionamento entre a classe **Person** com o literal contendo a data de nascimento do paciente, é mapeada de qualquer uma das bases por ser comum entre as três, porém foi feito utilizando somente a base “E-Saude” na coluna “Data de Nascimento”
- **hasCep:** Cria um relacionamento entre a classe **Person** com o literal contendo o cep, é mapeada da base “SIHSUS” na coluna “CEP”
- **hasCID:** Cria um relacionamento entre a classe Diagnosis com o literal contendo o código do diagnóstico, é mapeada nas três bases, na base “SIHSUS” na coluna “DIAG_PRINC”, na base “SIM” em “CAUSABAS” e “E-Saude” em “Codigo do CID”
- **hasCIDDescription:** Cria um relacionamento entre a classe Diagnosis com o literal contendo a descrição do diagnóstico, é mapeada da base “E-Saude” na coluna “Descricao do CID”

- **hasCNES:** Cria um relacionamento entre a classe Hospital com o literal contendo o CNES do hospital, é mapeada da base “SIHSUS” na coluna “CNES”
- **hasCNPJ:** Cria um relacionamento entre a classe Hospital com o literal contendo o CNPJ do hospital, é mapeada da base “SIHSUS” na coluna “CGC_HOSP”
- **hasCounty:** Cria um relacionamento entre a classe **Person** e a classe Hospital com o literal contendo o município do paciente e do hospital, é mapeada da base “E-Saude” na coluna “Municicio”, e na base “SIHSUS” na coluna “MUNIC_RES”
- **hasDeathDate:** Cria um relacionamento entre a classe **Person** com o literal contendo a data de morte do paciente, é mapeada da base “SIM” na coluna “DTOBITO”
- **hasGenre:** Cria um relacionamento entre a classe **Person** com o literal contendo o gênero do paciente, é mapeada de qualquer uma das bases por ser comum entre as três, porém foi feito utilizando somente a base “E-Saude” na coluna “Sexo”
- **hasInternmentDate:** Cria um relacionamento entre a classe **Person** com o literal contendo a data de internamento do paciente, é mapeada da base “SIHSUS” na coluna “DT_INTER”
- **hasNeighborhood:** Cria um relacionamento entre a classe **Person** com o literal contendo o bairro do paciente, é mapeada da base “E-Saude” na coluna “Bairro”
- **hasRace:** Cria um relacionamento entre a classe **Person** com o literal contendo a raça do paciente, é mapeada da base “SIM” na coluna “RACACOR”
- **hasScholarship:** Cria um relacionamento entre a classe **Person** com o literal contendo a escolaridade da pessoa, é mapeada da base “SIM” na coluna “ESC”
- **hasUpaCode:** Cria um relacionamento entre a classe **Upa** com o literal contendo o código da **Upa**, é mapeada da base “E-Saude“ na coluna “Codigo da Unidade”
- **hasUpaName:** Cria um relacionamento entre a classe **Upa** com o literal contendo o nome da **Upa**, é mapeada da base “E-Saude” na coluna “Descricao da Unidade”
- **hasWaterSupply:** Cria um relacionamento entre a classe **Person** com o literal contendo o nível de abastecimento da água na casa da pessoa, é mapeada da base “E-Saude” na coluna “Abastecimento”
- **hasWaterTreatment:** Cria um relacionamento entre a classe **Person** com o literal contendo o tipo de tratamento de água na casa da pessoa, é mapeada da base “E-Saude” na coluna "Tratamento no Domicilio"

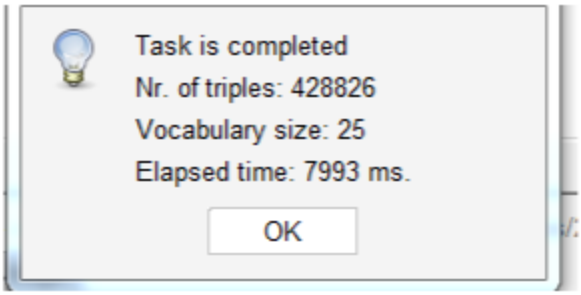
6.5 Mapeamento dos dados

Utilizando a ontologia, que foi criada com o intuito de possibilitar a integração através de um grafo semântico, foram realizados mapeamentos direto dos dados, utilizando um banco de dados relacional. Assim como nos testes feitos para comparar os frameworks, foi utilizado o banco de dados relacional PostgreSQL, com a ferramenta PgAdmin 4, porém com todos os dados já normalizados, padronizados, formatados em UTF-8. Na ferramenta Protégé, juntamente do plugin Ontop, foram criados 7 mapeamentos, 1 para os dados referentes à classe **Person**, 3 para os dados da classe **Diagnosis**, 1 para a classe **Hospital** e 1 para a classe **Upa**.

O mapeamento é feito com base em 2 campos, o Alvo das triplas e a Query. O primeiro é feito através da linguagem do próprio Ontop, onde o primeiro atributo é a Classe, enquanto que as seguintes são propriedades de dados ou de objeto que estarão vinculados a esta Classe. Já a Query é realizada em linguagem SQL, e armazena os valores de retorno nas variáveis, que podem ser vistas entre chaves na seção de Alvos das triplas. Os detalhes dos mapeamentos realizados podem ser vistos no Anexo B.

Após o mapeamento, foi necessário materializar as triplas resultantes deste utilizando o próprio plugin Ontop, para a inserção na *Triple Store* GraphDB. Ao final foram obtidas 428826 triplas inseridas em um arquivo no formato *"turtle"*, na Figura 13 é possível observar os resultados da materialização e o tempo necessário.

Figura 13 – Resultado da Materialização das Triplas



Fonte: Elaborada pelos Autores.

Com o mapeamento finalizado e a materialização de triplas completas foi possível a inserção das triplas no GraphDB, e com isso realizar queries que possibilitaram a criação de extração de informação sobre esses dados.

7 AVALIAÇÃO E RESULTADOS

Com a materialização completa e a inserção feita, foram criadas múltiplas queries em SPARQL, com o intuito de avaliar a integração realizada através das respostas às questões de competência apresentadas no início deste trabalho. Todas as informações extraídas nesta seção foram empíricas, sem a possibilidade de afirmar categoricamente qualquer relação apresentada por essas. Para isso seria necessária uma consulta a especialistas da área que seriam os responsáveis por contextualizar as informações extraídas, e assim ter conclusões mais respaldadas.

A primeira questão norteadora do trabalho foi: “é possível analisar a trajetória de pacientes que passam por uma unidade pública de saúde, são internadas em hospitais e vêm a óbito?”. Para responder essa questão foi elaborada a query apresentada abaixo, com a qual foram obtidas 3006 triplas contendo o perfil e os diagnósticos feitos na unidade de atendimento, hospitais e de óbito.

```
1  PREFIX pho: <http://www.semanticweb.org/auceli/ontologies
       /2019/4/pho#>
2  select DISTINCT ?person ?cidUpa ?cidHospital ?cidMorte where
3  {
4      ?person a pho:Person.
5      ?person pho:hasPublicHealthcareDiagnosis ?cid1.
6      ?person pho:hasHospitalDiagnosis ?cid2.
7      ?person pho:hasDeathDiagnosis ?cid3.
8      ?cid1 pho:hasCID ?cidUpa.
9      ?cid2 pho:hasCID ?cidHospital.
10     ?cid3 pho:hasCID ?cidMorte.
11 }
```

Como pode ser observado na tabela 7, o perfil instanciado como uma *"person"* na linha 3 sob o código "F19340714410690"contém 3 fluxos distintos, passando pela UPA com CID I64 (Acidente vascular cerebral, não especificado como hemorrágico ou isquêmico), L023 (Abscesso cutâneo, furúnculo e antraz da nádega) e I64 novamente, em seguida passando pelo hospital com CID A09 (Diarréia e gastroenterite de origem infecciosa presumível) em todos os fluxos, e finalmente vindo a óbito com CID I64, também em todos os fluxos encontrados.

Com esses resultados é possível supor que utilizando o método de integração exercido neste trabalho foi possível analisar os possíveis ciclos de diagnóstico que um perfil pode conter. Além disso, a tabela 7 também demonstra que não é possível ser realizado o reconhecimento

Tabela 7 – Ciclo de diagnósticos de um perfil

Person	cidUpa	cidHospital	cidMorte
F19270527410690	R100	A09	G309
F19270527410690	R100	A09	I219
F19340714410690	I64	A09	I64
F19340714410690	L023	A09	I64
F19340714410690	I64	A09	I639

de uma pessoa dentro do perfil, assim não ferindo a anonimização dos dados. Isso pode ser observado com os perfis presentes na tabela, por exemplo, o perfil 'F19270527410690' contém 2 fluxos distintos, embora parecidos uma vez que somente o CID de morte é diferente.

O fato dos perfis contemplarem mais de um paciente pode ser visto como uma limitação do trabalho em si, uma vez que o paciente não pode ser identificado, seja pela falta de dados disponibilizados pelo SUS ou pela obrigatoriedade da anonimização dos dados. Essa limitação impede estudos mais aprofundados, como identificar todas as passagens daquele paciente pelo sistema do SUS, mas permite outros tipos de análises, como identificar principais doenças que afentam um perfil específico, por exemplo.

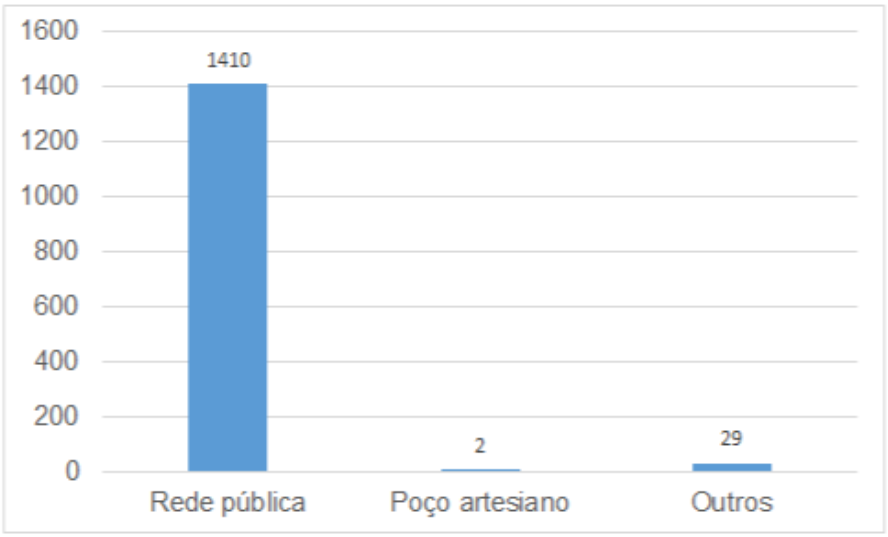
A segunda questão de competência apresentada era: "é possível encontrar uma relação entre pacientes que são encaminhados para hospitais e acabam falecendo, com o nível de abastecimento e tratamento de água de suas residências?". Para responder essa questão foram necessárias várias extrações para buscar uma relação entre os dados.

Primeiro foi necessário recuperar qual o tipo de abastecimento predominante nas casas de cada perfil (Person), informação extráida através da *query* apresentada a seguir, com a qual foi possível projetar o gráfico apresentado na Figura 14, onde fica claro o tipo de abastecimento predominante.

```
1 PREFIX pho: <http://www.semanticweb.org/auceli/ontologies
     /2019/4/pho#>
2 select ?abastecimento (count(?abastecimento) as ?count) where
3 {
4     ?person a pho:Person.
5     ?person pho:hasWaterTreatment ?tratamento.
6     ?person pho:hasWaterSupply ?abastecimento.
7 } groupby(?abastecimento)
```

Através desses resultados foi feita a escolha de utilizar somente os registros com abastecimento da rede pública, já que a vasta maioria dos casos se encontram neste tipo. Com isso foi necessário analisar o tipo de tratamento da água presente nos perfis, isso foi possível através da *query* apresentada a seguir, que permitiu gerar o gráfico da Figrua 15.

Figura 14 – Tipos de abastecimento nos perfis



Fonte: Elaborada pelos Autores.

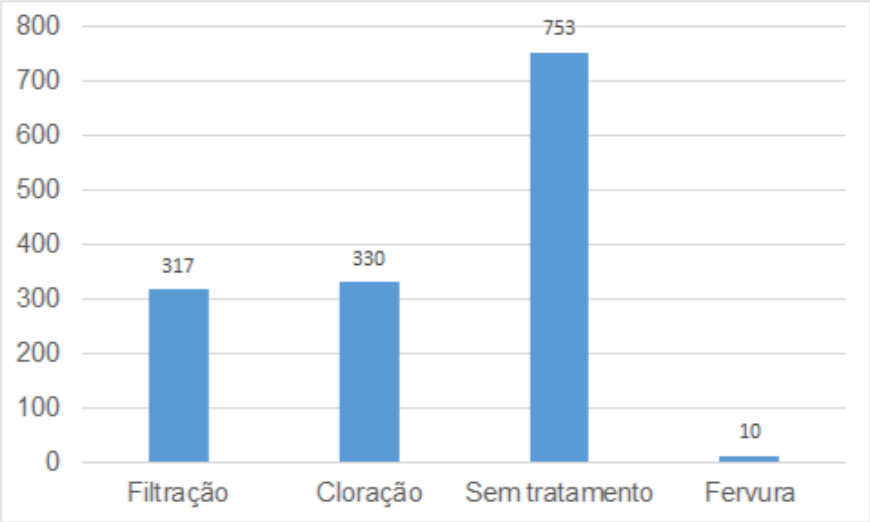
```
PREFIX pho: <http://www.semanticweb.org/auceli/ontologies
   /2019/4/pho#>
select ?tratamento(count(?tratamento) as ?count) where
{
    ?person a pho:Person.
    ?person pho:hasWaterTreatment ?tratamento.
    ?person pho:hasWaterSupply ?abastecimento.
    VALUES ?abastecimento {"REDE PUBLICA"}
} groupby(?tratamento)
```

No gráfico da Figura 15 é possível análisar que ocorre uma discrepância maior entre os tipos de tratamento de água do que os tipos de abastecimento apresentados no gráfico da Figura 14.

Com esses resultados em mãos, foi possível verificar que a maioria dos perfis não tratam a sua água proveniente da rede pública. Esse foi o caso selecionado para observar a informação que relaciona o tratamento de água com as doenças apresentadas por esses perfis. Com todas as restrições já analisadas, foi criada a *query* apresentada abaixo, feita para analisar os diagnósticos para este caso específico.

```
PREFIX pho: <http://www.semanticweb.org/auceli/ontologies
   /2019/4/pho#>
select ?cid (count(?cid) as ?count) where {
```

Figura 15 – Tipos de tratamento de água feito pelos perfis



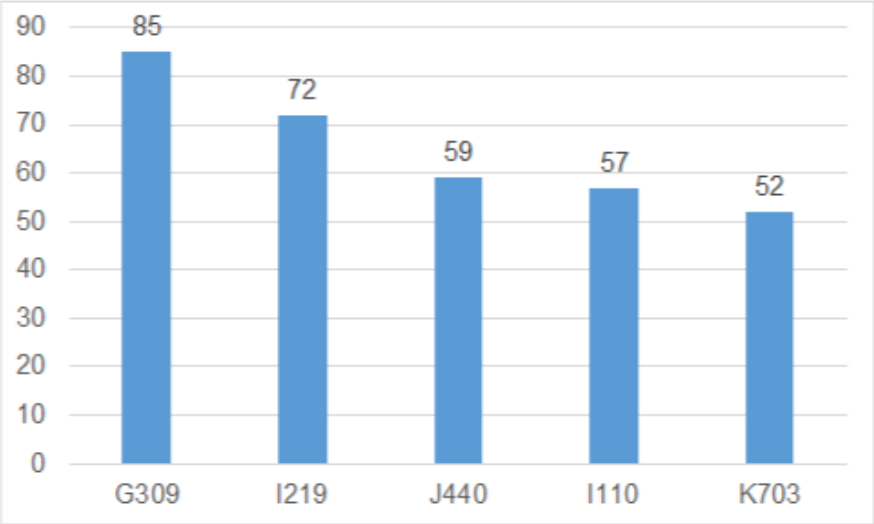
Fonte: Elaborada pelos Autores.

```
3       ?person a pho:Person.
4       ?person pho:hasWaterTreatment ?tratamento.
5       VALUES ?tratamento {"SEM TRATAMENTO"}.
6       ?person pho:hasWaterSupply ?abastecimento.
7       VALUES ?abastecimento {"REDE PUBLICA"}.
8       ?person pho:hasDeathDiagnosis ?diag.
9       ?diag pho:hasCID ?cid.
10      ?person pho:hasScholarship ?escolaridade.
11      ?person pho:hasCep ?cep.
12      ?person pho:hasNeighborhood ?bairro.
13      ?person pho:hasRace ?raca.
14      ?person pho:hasGenre ?genero.
15  } group by (?cid)
16  order by desc (?count)
```

Com o retorno recebido da *query*, foram disponibilizados vários diagnósticos que foram considerados a causa de morte dos perfis, com isso foi possível agrupar os diagnósticos e visualizar através do gráfico da Figura 16 as 5 doenças mais comuns em perfis que não tratam sua água proveniente da rede pública.

Assim podendo verificar que os diagnósticos mais comuns são: “Doença de Alzheimer não especificada” com o CID “G309”; “Infarto agudo do miocárdio não especificado” com o

Figura 16 – 5 Doenças mais comuns em perfis que não tratam sua água proveniente de rede pública



Fonte: Elaborada pelos Autores.

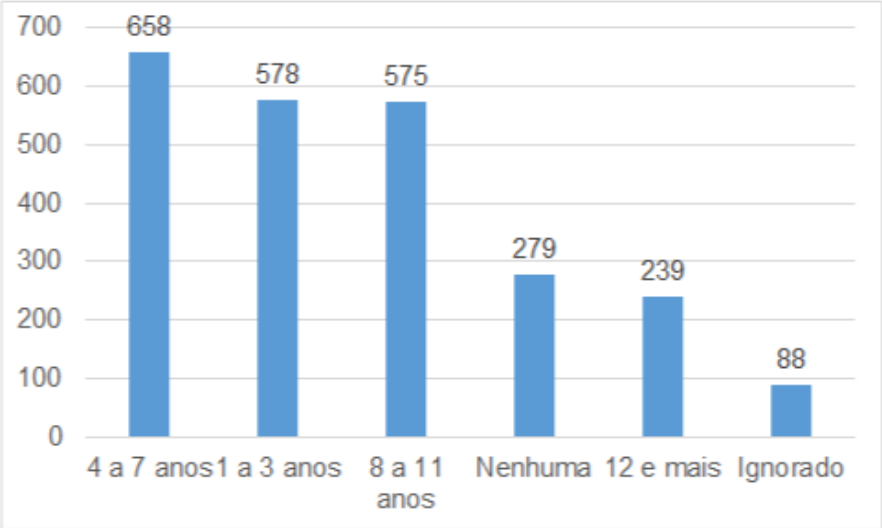
CID “I219”; “Doença pulmonar obstrutiva crônica com infecção respiratória aguda do trato respiratório inferior” com o CID “J440”; “Doença cardíaca hipertensiva com insuficiência cardíaca (congestiva)” com o CID “I110”; e “Cirrose hepática alcoólica” com o CID “K703”. Como não é o foco deste trabalho realizar análises sem especialistas do domínio, não é possível evidenciar uma relação entre os diagnósticos com o tratamento de água. Porém foi demonstrando que é possível recuperar essa informação que reúne dados de diferentes bases recuperando uma informação integrada de dois domínios completamente diferentes como saneamento e saúde.

Por terem sido mapeados múltiplos dados dos datasets foi possível realizar outras análises que não fizeram parte do conjunto de perguntas que nortearam a criação do grafo de conhecimento, expandindo os tipos de informações que são possíveis de se obter com a integração. Uma relação interessante que pode ser obtida, utilizando a segunda questão de competência como base, foi sobre o "nível de escolaridade dos perfis que não tratam a água proveniente da rede pública". Para responder a essa questão foi elaborada a *query* apresentada a seguir, com a qual foi gerada o gráfico apresentado na Figura 17.

```
1 PREFIX pho: <http://www.semanticweb.org/auceli/ontologies
     /2019/4/pho#>
2 select ?escolaridade (count(?escolaridade) as ?count) where {
3     ?person a pho:Person.
4     ?person pho:hasWaterTreatment ?tratamento.
5     VALUES ?tratamento {"SEM TRATAMENTO"}.
6     ?person pho:hasWaterSupply ?wsupply.
7     VALUES ?wsupply {"REDE PUBLICA"}.
8     ?person pho:hasDeathDiagnosis ?diag.
```

```
9       ?diag pho:hasCID ?cid.
10      ?person pho:hasBirthDate ?data.
11      ?person pho:hasScholarship ?escolaridade.
12      ?person pho:hasCep ?cep.
13      ?person pho:hasNeighborhood ?bairro.
14      ?person pho:hasRace ?raca.
15      ?person pho:hasGenre ?genero.
16  } group by (?escolaridade)
17  order by desc (?count)
```

Figura 17 – Nível de escolaridade de perfis com registro de óbito que não tratam a água proveniente da rede pública



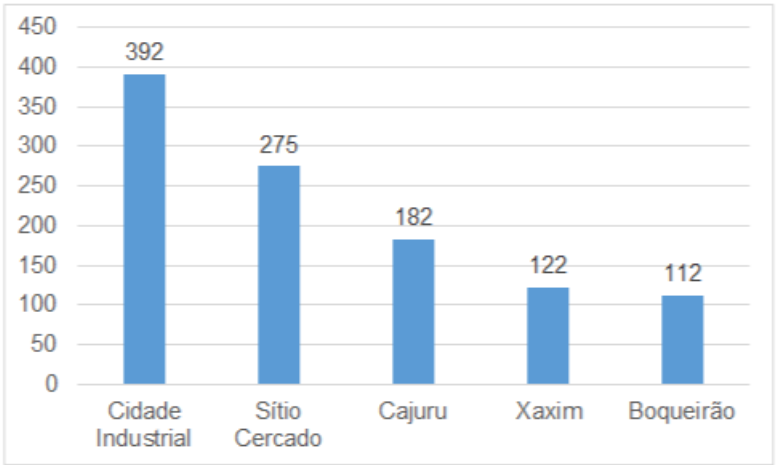
Fonte: Elaborada pelos Autores.

Observando o gráfico da Figura 17, é possível averiguar uma possível relação entre o nível de escolaridade e o fato dos pacientes tratarem ou não a água proveniente de rede pública de suas residências, ocasionando em possíveis doenças que podem ser fatais. Outra relação interessante foi a verificação dos bairros com maior ocorrência de óbitos dos perfis que não tratam a água de suas residências. Para extrair essa informação, foi criada a *query* a seguir, com a qual foi possível gerar o gráfico da Figura 18. Algumas triplas presentes na *query*, como por exemplo “?person pho:hasNeighborhood ?bairro”, aparecem para filtrar registros que obrigatoriamente contém esses dados.

```
1  PREFIX pho: <http://www.semanticweb.org/auceli/ontologies
       /2019/4/pho#>
2  select ?bairro (count(?bairro) as ?count) where {
```

```
3       ?person a pho:Person.
4       ?person pho:hasWaterTreatment ?tratamento.
5       VALUES ?tratamento {"SEM TRATAMENTO"}.
6       ?person pho:hasWaterSupply ?abastecimento.
7       VALUES ?abastecimento {"REDE PUBLICA"}.
8       ?person pho:hasDeathDiagnosis ?diag.
9       ?diag pho:hasCID ?cid.
10      ?person pho:hasScholarship ?escolaridade.
11      ?person pho:hasCep ?cep.
12      ?person pho:hasNeighborhood ?bairro.
13      ?person pho:hasRace ?raca.
14      ?person pho:hasGenre ?genero.
15  } group by (?bairro)
16  order by desc (?count)
```

Figura 18 – Bairros com maior taxa de mortes em perfis que não tratam a água proveniente da rede pública



Fonte: Elaborada pelos Autores.

Através deste gráfico é possível verificar que existe uma diferença grande entre os 3 bairros com a maior quantidade de registros, para com os outros 2. Essa informação permite considerar que talvez exista uma relação entre o fato dessas regiões conterem as maiores taxas de morte e não terem um tratamento adequado da água, o que é uma informação valiosa que a Prefeitura de Curitiba poderia utilizar para a manutenção da rede pública.

Embora a falta de um especialista para afirmar as informações extraídas, as análises demonstram a riqueza de novas informações que podem ser geradas com os dados conectados, uma vez que foi possivel vincular e relacionar dados que estavam contidos em bases diferentes.

8 EVOLUÇÃO DO GRAFO DE CONHECIMENTO SEMÂNTICO

Um ponto importante a se ressaltar sobre os dados desse trabalho é sua granularidade, o período a que se refere os dados utilizados para análise, limpeza e construção do grafo de conhecimento semântico. Como já dito na seção 6.1, as bases são disponibilizadas em seus portais com granularidades diferentes, as bases do E-Saude contém dados trimestrais, as bases do SIHSUS contém dados mensais e as bases do SIM contém dados anuais.

Como a base do SIM é disponibilizada anualmente, contendo a maior granularidade dos dados, ela foi escolhida para reger o período e a granularidade dos dados utilizados, portanto todas bases deveriam conter dados anuais, assim sendo necessário efetuar a etapa de junção dos arquivos obtidos das bases do E-Saude e do SIHSUS, citada na seção 6.1, para gerar um único arquivo de cada com seus dados do ano.

No começo desse trabalho foi escolhido o ano de 2017 como período para obtenção dos dados, isso devido aos dados da base SIM estarem disponíveis somente até esse ano. Sabendo o tempo que se passou e que os portais já disponibilizaram bases com dados mais atuais, foi analisado a relevância e necessidade de atualização dos dados utilizados nesse trabalho.

No momento em que esse trabalho está sendo apresentado já existem dados de 2022 disponíveis nas bases do E-Saude e do SIHSUS, mas a base do SIM contém dados diponíveis somente até 2020, portanto esse foi o ano escolhido para ser analisado quanto a necessidade de atualização dos dados.

8.1 Análise das Bases 2020

Ao analisar as bases e seus dicionários de dados de 2020, comparando-os com os de 2017, foi possível averiguar que não houve mudanças significativas na estrutura e nos dados em si, ou seja, informações como nomes das colunas, quantidade de colunas, padrões de nomenclatura e de preenchimento continuam os mesmos. Embora algumas alterações tenham sido feitas nas estruturas das bases, como adição de colunas e alterações de preenchimento, as colunas utilizadas para esse trabalho não sofreram modificações.

A mudança mais significativa que pode ser observada foi a dos próprios portais pelos quais é possível se obter as bases de dados, tanto suas URLs quanto suas interfaces sofreram modificações, afim de facilitar o acesso aos dados. As URLs atualizadas estão disponíveis nas referências desse trabalho, elas substituiram as antigas que estavam sem acesso as páginas dos portais.

8.2 Atualização das Bases

Confirmando que as colunas utilizadas nesse trabalho continuam as mesmas, efetuar uma atualização dos dados desse trabalho se torna algo simples e desnecessário, uma vez que não agregaria valor ao projeto em si, pois embora dados mais atuais, eles continuam os mesmos de 2017.

Para adicionar os dados de 2020 ao trabalho bastaria seguir os mesmos passos descritos na seção 4, na qual é descrito a metodologia utilizada no projeto. A maior diferença seria que essas bases seguiriam somente até a etapa de criação dos perfis, pois não é necessário recriar o grafo de conhecimento semântico, uma vez que ele já existe e contempla as bases de 2017.

Após essas bases passarem pelos mesmos tratamentos utilizados nas bases de 2017, elas estariam exatamente na mesma estrutura final, com a mesma quantidade de colunas, nomenclaturas e formato dos dados. Tendo essas bases tratadas, seria necessário somente inserir esses dados no banco que o grafo de conhecimento semântico está utilizando para ler os dados, dessa forma ele passará a ter dados de 2017 e 2020 para realizar as análises.

9 CONSIDERAÇÕES FINAIS

Neste trabalho foi apresentado o problema de integração semântica de dados provenientes de bases de dados abertas de mesmo domínio. Além disso, foram apresentados trabalhos correlatos nas áreas de Ontologias, dados abertos, integração de dados e dados médicos. Que fundamentaram a metodologia do presente trabalho, e as ferramentas e *frameworks* utilizados no experimento.

Foi apresentado também um método para ser possível a integração das bases anonimizadas, a criação de um perfil em comum nas três bases. Esse foi inserido em uma nova coluna chamada UID, onde foi utilizada para a comparação e igualdade de registros nas 3 bases. Assim possibilitando a análise dos dados pertinentes a cada perfil ao longo dos 3 datasets, permitindo localizar informações pertinentes aquele perfil, como fluxos diagnósticos concedidos ou CIDs mais comuns para o perfil em questão, por exmeplo.

Embora fosse interessante a possibilidade de identificar um paciente nas bases, de forma a localizar fluxos únicos de CIDs recebidos em cada situação, com a técnica de criação de perfis desenvolvida não foi possível alcançar esse nível de unicidade. Uma vez que cada perfil contém vários pacientes, devido as caracteristicas utilizadas em sua construção serem comuns entre os pacientes. Essa identificação de pacientes na base também é algo a ser discutido, uma vez que isso irá ferir a anonimização dos dados disponibilizados.

Foi criada uma ontologia intitulada *Public Healthcare Ontology* (PHO), onde os dados presentes nos datasets foram mapeados para classes, propriedades de objeto e de dados disponibilizados na ontologia. O mapeamento foi feito através do *framework* Ontop, onde foi utilizado um banco de dados relacional PostgreSQL.

Após o mapeamento, as triplas foram materializadas em um arquivo, que foi inserido em uma *Triple Store* chamada GraphDB. Ele possibilitou a criação de queries em SPARQL, para a análise dos dados. Essas informações deveriam responder perguntas norteadoras do trabalho, que tinham como objetivo averiguar se a integração foi bem sucedida ou não.

As observações sobre os dados integrados realizadas foram todas empíricas, devido a ausência de um especialista do domínio que deveria contextualizá-las. Porém todas as perguntas foram respondidas ou foi apresentado a possibilidade de respondê-las com a ajuda de um especialista, o que explicita a completude da integração. Além disso, também foram feitas outras observações a partir de consultas e foram geradas novas informações com os dados integrados. Isso demonstra a importância da integração de bases de dados, devido à riqueza de informações que podem ser obtidas através desta.

Algumas restrições do trabalho foram consideradas fora do escopo, como a escalabilidade da integração, que considerou dados somente do ano de 2017, sendo possível afirmar que a mesma iria funcionar com um período maior. A atualização da granularidade dos dados pertencentes ao ano de 2017 foi considerada, uma vez que já foram disponibilizados dados mais recentes, até 2020. Porém, ao analisar os novos dados disponíveis, foi averiguado que não

houve uma mudança significativa na estrutura dos dados ou uma adição relevante de novas informações, para este projeto.

Apesar das bases de 2020 terem sofrido alterações, a estrutura das colunas utilizadas nesse projeto continuam as mesmas. Embora isso facilite a adição desses dados a ontologia, isso foi considerado desnecessário, uma vez que esse fato também garante que os resultados obtidos com os dados de 2017 ainda são considerados relevantes, pois o foco desse trabalho era a construção de um grafo de conhecimento semântico, o qual foi obtido e avaliado.

Como trabalhos futuros, os autores indicam testar a escalabilidade da integração para dados de outros anos, e possivelmente ampliar ela caso ela seja insuficiente. Além disso, ampliar a ontologia para que essa possa descrever novos relacionamentos entre dados, assim gerando mais informações. Ainda sobre a ontologia, tentar conectar ela com outras ontologias já definidas para que ela possa ser reutilizada, também seria um trabalho pertinente. Por fim, agregar um especialista do domínio para que este possa contextualizar as análises apresentadas, e para que ele possa opinar sobre possíveis mudanças que seriam interessantes para a área médica.

REFERÊNCIAS

ALMEIDA, M. B. Uma abordagem integrada sobre ontologias: Ciência da informação, ciência da computação e filosofia. **Perspectivas em Ciência da Informação**, SciELO Brasil, v. 19, n. 3, p. 242–258, 2014.

BERNERS-LEE, T.; HENDLER, J.; LASSILA, O. The semantic web. **Scientific american**, JSTOR, v. 284, n. 5, p. 34–43, 2001.

BRICKLEY, D.; MILLER, L. Foaf vocabulary. Accessed: 2021-10-25. 2000. Disponível em: http://www.foaf-project.org/.

CALVANESE, D. *et al.* Obda with the ontop framework. In: CITESEER. **SEBD**. [*S.l.*], 2015. p. 296–303.

CENSEC, C. N. d. S. E. C. Tabela de municípios ibge - cesdi. Accessed: 2021-11-07. 2018. Disponível em: ftp://geoftp.ibge.gov.br/organizacao_do_territorio/estrutura_territorial/divisao_territorial/2018/DTB_2018.zip.

CIVIL, C. Lei de acesso a informação. Accessed: 2021-10-25. 2018. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/l12527.htm.

CURITIBA, P. de. Sistema e-saude - perfil de atendimento de enfermagem nas unidades municipais de saúde de curitiba. Accessed: 2021-10-26. 2020. Disponível em: http://dadosabertos.c3sl.ufpr.br/curitiba/SESPAEnfermagem/.

DING, L. *et al.* Sameas networks and beyond: analyzing deployment status and implications of owl: sameas in linked data. In: SPRINGER. **International Semantic Web Conference**. [*S.l.*], 2010. p. 145–160.

FOUNDATION, O. K. Open data handbook. Accessed: 2021-11-04. 2010. Disponível em: http://opendatahandbook.org/guide/en/.

GUARINO, N.; OBERLE, D.; STAAB, S. **What Is an Ontology?** [*S.l.*]: Handbook on Ontologies, International Handbooks on Information Systems, 2009.

ILIADIS, A. *et al.* Covid-19 knowledge graphs in health communication and information. **COVID-19 Research**, 2021.

ISOTANI, S.; BITTENCOURT, I. **Dados Abertos e Conectados**. [*S.l.*]: ceweb br, 2015.

JAFFRI, A.; GLASER, H.; MILLARD, I. Uri identity management for semantic web data integration and linkage. In: SPRINGER. **OTM Confederated International Conferences"On the Move to Meaningful Internet Systems"**. [*S.l.*], 2007. p. 1125–1134.

JUNIOR, F. *et al.* Avaliação da prontidão para abertura de dados das instituições públicas brasileiras: caso de uma instituição financeira pública brasileira. **Brazilian Journal of Information Studies: Research Trends**, 2018.

KEET, M. C. **An Introduction to Ontology Engineering**. [*S.l.*]: a Creative Commons Attribution 4.0 International License (CC BY 4.0), 2020.

KLÍMEK, J. *et al.* Publication and usage of official czech pension statistics linked open data. **Journal of Web Semantics**, Elsevier, v. 48, p. 1–21, 2018.

LOPES, G.; VIDAL, V.; OLIVEIRA, M. Construção de linked data mashup para integração de dados da saúde pública. SBBD, 2016.

MENDES, P. N.; MÜHLEISEN, H.; BIZER, C. Sieve: linked data quality assessment and fusion. In: **Proceedings of the 2012 Joint EDBT/ICDT Workshops**. [*S.l.*: *s.n.*], 2012. p. 116–123.

MUSEN, M. The protégé project: A look back and a look forward. Accessed: 2021-10-26. 2015. Disponível em: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4883684/.

NUNES, G. M. O.; BERARDI, R. C. G. Ontological model for decision support about bariatric surgery. In: **ONTOBRAS**. [*S.l.*: *s.n.*], 2020. p. 280–285.

OLIVEIRA, H.; LOSCIO, B. **Web semântica e matching de ontologias: Uma visão geral**. [*S.l.*]: InfoBrasil, 2008.

PEREIRA, D.; WASSERMANN, R.; SALVADOR, L. Integração semântica das bases de dados do município de são paulo: Um estudo de caso com anomalias congênitas. CNPq, 2017.

QUEIROZ, M.; LINO, N.; MOTTA, G. Uma ontologia de domínio para preservação de privacidade em dados publicados pelo governo brasileiro. **XII Brazilian Symposium on Information Systems, Florianópolis, SC**, 2016.

ROLIM, T. *et al.* Um enfoque incremental para construção do grafo de conhecimento do sus. In: SBC. **Anais do XX Simposio Brasileiro de Computação Aplicada à Saúde**. [*S.l.*], 2020. p. 72–83.

SANTAREM, J.; CONEGLIAN, C. **Web semântica e ontologias: Um estudo sobre construção de axiomas e uso de inferências**. [*S.l.*]: CNPq, 2016.

SAúDE, M. da. Datasus. Accessed: 2021-10-25. 2017. Disponível em: https://datasus.saude.gov.br/transferencia-de-arquivos/.

SCHULTZ, A. *et al.* **LDIF - A Framework for Large-Scale Linked Data Integration**. [*S.l.*]: WWW2012 Developer Track, 2012.

TEAM, C. Pandas documentation. Accessed: 2021-10-25. 2021. Disponível em: https://pandas.pydata.org/pandas-docs/stable/.

TOMA, I. *et al.* **Enabling Scalable Multi-Channel Communication through Semantic Technologies**. [*S.l.*]: IEEE, 2013.

VIDAL, V. *et al.* Specification and incremental maintenance of linked data mashup views. **International Conference on Advanced Information Systems Engineering**, 2015.

VOLZ, J. *et al.* Silk-a link discovery framework for the web of data. In: **Ldow**. [*S.l.*: *s.n.*], 2009.

W3C. Resource description framework (rdf). Accessed: 2021-11-07. 2010.

W3C. Web ontology language (owl). Accessed: 2021-10-28. 2012. Disponível em: https://www.w3.org/OWL/.

W3C. Linked data. Accessed: 2021-10-28. 2015. Disponível em: https://www.w3.org/standards/semanticweb/data.

W3C. Rdf schema 1.1. Accessed: 2021-10-28. 2015. Disponível em: https://www.w3.org/TR/rdf-schema/.

W3C. Data on the web best practices. Accessed: 2021-10-26. 2017. Disponível em: https://www.w3.org/TR/dwbp/.

WU, J. Construct a knowledge graph for china coronavirus (covid-19) patient information tracking. **Risk Management and Healthcare Policy**, Dove Press, v. 14, p. 4321, 2021.

ANEXO A – Código Python Utilizado para Limpeza e Normalização das Bases

```
"""
Imports utilizados para realizacao dos tratamentos
"""
import pandas as pd
import numpy as np
from datetime import datetime

"""
Etapa de remocao de colunas com menos de 2 valores distintos
"""
for coluna in data.columns:
  regs = data.groupby(coluna)
  if (len(regs) < 2):
    data.drop(columns=coluna, inplace=True)

"""
Etapa de remocao de colunas em branco utilizando a funcao
   dropna, propria da lib Pandas
"""
data = data.dropna(subset=[coluna])

"""
Etapa utilizada para filtrar registos das bases SIHSUS e SIM
   que estejam em cidades existentes na base do E-Saude,
   utilizando a funcao loc, propria da lib Pandas
"""
data = data.loc[condicao]

"""
Etapa de normalizacao dos dados das bases, remocao de acentos e
    caracteres especiais
"""
```

```
29 data[colunas] = data[colunas].apply(lambda x: x.str.normalize('
      NFKD').str.encode('ascii', errors='ignore').str.decode('utf
      -8'))
30
31 """
32 Etapa de padronizacao da coluna Sexo, deixando todas as bases
      com os padroes M para Masculino e F para Feminino
33 """
34 data.replace({'Sexo': {idMasculino: 'M', idFeminino: 'F'}},
      inplace = True)
35
36 """
37 Etapa de padronizacao da data de nascimento, deixando todas as
      bases com o padrao 'yyyymmdd'
38 """
39 data['Data de Nascimento'] = data['Data de Nascimento'].apply(
      lambda x: x.strftime('%Y%m%d'))
40
41
42 """
43 Etapa de padronizacao das cidades utilizando uma base auxiliar
      do IBGE para transformar as cidades escritas por extenso em
      codigos
44 """
45 data['Cidade'] = data['Cidade'].apply(lambda x: ibge_citys.loc[
      ibge_citys['Cidade'] == x]['Codigo IBGE'])
```

ANEXO B – Mapeamentos de Classes

Mapeamento Classe Person

```
-- Alvo das Triplas:
:{e.uid} a: Person;
    :hasBirthDate {e.dataNascimento};
    :hasWaterTreatment {e.tratamentoAgua};
    :hasWaterSupply {e.abastecimento};
    :hasCep {cep};
    :hasScholarship {escolaridade};
    :hasGenre {e.sexo};
    :hasNeighborhood {bairro};
    :hasCounty {municipiores};
    :hasRace {racacor};
    :hasInternmentDate {h.datainternamento};
    :hasDeathDiagnosis :{cidbasico};
    :hasHospitalDiagnosis :{cid};
    :hasPublicHealthcareDiagnosis :{codigocid};

-- Query:
SELECT
    e.uid, e.dataNascimento, e.tratamentoAgua,
    e.abastecimento, codigocid, cid, cidbasico,
    cep, escolaridade, e.sexo, h.datainternamento,
    bairro, municipiores, racacor, h.datainternamento
FROM
    esaude e, sihsus h, sim d
WHERE
    (e.uid=h.uid and h.uid=d.uid)
```

Mapeamento Classe Diagnosis (E-Saude)

```
-- Alvo das Triplas:
```

```
:{codigocid} a :Diagnosis;
    :hasCID {codigoCid};
    :hasCIDDescription {descricaocid};

-- Query:
SELECT codigocid, descricaocid FROM esaude
```

Mapeamento Classe Diagnosis (SIHSUS)

```
-- Alvo das Triplas:
:{codigocid} a: Diagnosis;
    :hasCID {codigoCid};
    :hasCIDDescription {descricaocid};

-- Query:
SELECT codigocid, descricaocid FROM esaude
```

Mapeamento Classe Diagnosis (SIM)

```
-- Alvo das Triplas:
:{cidbasico} a: Diagnosis;
    :hasCID {cidbasico};

-- Query:
SELECT cidbasico FROM sim
```

Mapeamento Classe Hospital

```
-- Alvo das Triplas:
:{cnes} a :Hospital;
    :hasCNPJ {cnpj};
    :hasCNES {cnes};
```

```
    :hasCounty {municipioestabelecimento};

-- Query:
SELECT cnpj, cnes, municipioestabelecimento FROM sihsus
```

Mapeamento Classe Hospital

```
-- Alvo das Triplas:
:{nomeunidade} a: Upa;
    :hasUpaCode {codigounidade};
    :hasUpaName {nomeunidade};

-- Query:
SELECT codigounidade, nomeunidade FROM esaude
```