



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE CIÊNCIAS
DEPARTAMENTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
MESTRADO ACADÊMICO EM CIÊNCIA DA COMPUTAÇÃO

FRANCISCA JAMIRES DA COSTA

**ABORDAGEM BASEADA NA ARQUITETURA DIKW PARA CONSTRUÇÃO DE
GRAFOS DE CONHECIMENTO SEMÂNTICO QUE INTEGRAM DADOS IOT COM
FONTES DE DADOS HETEROGÊNEAS**

FORTALEZA

2023

FRANCISCA JAMIRES DA COSTA

ABORDAGEM BASEADA NA ARQUITETURA DIKW PARA CONSTRUÇÃO DE GRAFOS
DE CONHECIMENTO SEMÂNTICO QUE INTEGRAM DADOS IOT COM FONTES DE
DADOS HETEROGÊNEAS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Orientadora: Prof^ª. Dr^ª. Vânia Maria Ponte Vidal

FORTALEZA

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

C872a Costa, Francisca Jamires da.

Abordagem baseada na arquitetura DIKW para construção de Grafos de conhecimento semântico que integram Dados IoT com fontes de dados Heterogêneas / Francisca Jamires da Costa. – 2023.
89 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação, Fortaleza, 2023.

Orientação: Profa. Dra. Vânia Maria Ponte Vidal.

1. Grafo de Conhecimento. 2. Internet das Coisas - IoT. 3. Ontologia. 4. Hierarquia DIKW. 5. Metadados. I. Título.

CDD 005

FRANCISCA JAMIRES DA COSTA

ABORDAGEM BASEADA NA ARQUITETURA DIKW PARA CONSTRUÇÃO DE GRAFOS
DE CONHECIMENTO SEMÂNTICO QUE INTEGRAM DADOS IOT COM FONTES DE
DADOS HETEROGÊNEAS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Aprovada em:

BANCA EXAMINADORA

Prof^a. Dr^a. Vânia Maria Ponte Vidal (Orientadora)
Universidade Federal do Ceará (UFC)

Dr^a. Chiara Renso
Institute of National Research Council of Italy (CNR)

Prof^a. Dr^a. Rossana Maria de Castro Andrade
Universidade Federal do Ceará (UFC)

Dr. Fabio André Machado Porto
Laboratório Nacional de Computação Científica
(LNCC)

À minha família, por todo apoio e encorajamento durante este processo.

AGRADECIMENTOS

Inicialmente, expresso minha profunda gratidão a Deus por ter me concedido a força e coragem necessárias para enfrentar os desafios que encontrei ao longo desta jornada.

À Professora Doutora Vânia Maria Ponte Vidal, minha orientadora, dedico meus mais sinceros agradecimentos. Sou profundamente grata por sua orientação ao longo de todo o mestrado. Cada ideia compartilhada, cada feedback valioso e cada ensinamento transmitido foram fundamentais para o meu crescimento acadêmico. Obrigada por me ensinar não apenas os fundamentos da pesquisa, mas também como desenvolvê-la com excelência.

Às amigadas que conquistei durante o programa de mestrado e doutorado, em especial a Caio Viktor da Silva Ávila, Túlio Vidal Rolim, Amanda Drielly Pires Venceslau, Danne Makleyston Pereira Gomes e Pedro Luis Saraiva Barbosa, expresso minha sincera gratidão. O apoio, suporte e compartilhamento de materiais de estudo foram inestimáveis para o sucesso desta pesquisa.

Aos renomados professores do MDCC, agradeço por compartilharem seu profundo conhecimento de maneira admirável. Suas aulas foram essenciais para minha formação e instrumentalização na área de pesquisa.

À minha família, namorado e amigos, que compreenderam e apoiaram minha ausência durante os estudos, expresso minha profunda gratidão. Suas palavras de encorajamento sempre me lembraram da importância da dedicação no presente para construir um futuro promissor.

Ao Instituto Federal de Educação, Ciência e Tecnologia do Ceará - IFCE, campus Cedro, agradeço por conceder-me a oportunidade de afastamento para capacitação profissional, o que viabilizou minha dedicação ao mestrado. Agradeço por acreditar no meu potencial e por apoiar meu desenvolvimento acadêmico.

“Sabemos que Deus age em todas as coisas para o bem daqueles que o amam, dos que foram chamados de acordo com o seu propósito.”

(Romanos 8.28)

RESUMO

Nos últimos anos, os Grafos de Conhecimento Semântico (GCS) têm conquistado um espaço sólido no mercado como uma solução dinâmica e escalável para atender às demandas de integração de dados em larga escala. Nesse contexto, um GCS funciona como um centro de informações, abrangendo dados, metadados e conteúdo, proporcionando uma visão unificada, consistente e livre de ambiguidades dos dados provenientes de diversas fontes e sistemas. Entretanto, a construção de um grafo de conhecimento semântico é uma tarefa complexa que exige uma abordagem meticulosa para garantir sua eficácia, escalabilidade e capacidade de atender aos objetivos das aplicações. Assim sendo, este trabalho propõe uma abordagem fundamentada na arquitetura *DIKW* para a construção e especificação de Grafos de Conhecimento Semântico que integram dados *IoT* com fontes de dados heterogêneas. Neste contexto, investigamos três questões de pesquisa fundamentais que resultaram em três contribuições significativas: Questão 1: Como construir um GCS de dados *IoT* que seja compatível com os diferentes níveis da arquitetura *DIKW*? Questão 2: Como construir um GCS que integre os dados de *IoT* com fontes de dados heterogêneas? Questão 3: Como descrever um grafo de conhecimento de dados *IoT*, tornando-o acessível e compreensível para os usuários? Em resposta, à primeira questão da pesquisa, este trabalho apresenta a abordagem *DIKW4IoT*, que fornece a estrutura necessária para representar informações *IoT* em diferentes granularidades, permitindo que os dados evoluam na hierarquia *DIKW*. Isso os capacita a se transformarem em informações valiosas e, em última instância, a desencadear ações e decisões informadas. Em relação à segunda questão propõe-se o *SIGIoT-S*, um *framework* conceitual que orienta quais dados podem ser utilizados no processo de integração e identifica os passos necessários para alcançar uma integração harmoniosa. Por fim, no que diz respeito à terceira questão, elaborou-se a ontologia *MetaGraphIoT*. Esta ontologia tem como propósito a anotação de metadados de um grafo de conhecimento de dados *IoT*. Essa anotação torna o grafo de conhecimento mais interpretável e utilizável pelos usuários. Essas contribuições respondem às questões de pesquisa identificadas e representam avanços substanciais na gestão de dados *IoT*, promovendo uma melhor utilização e compreensão desses dados em diversas aplicações.

Palavras-chave: ontologia; grafo de conhecimento; hierarquia *DIKW*; internet das coisas - *IoT*; integração de dados; metadados

ABSTRACT

In recent years, Semantic Knowledge Graphs (SKGs) have gained a solid foothold in the market as a dynamic and scalable solution to meet the demands of large-scale data integration. In this context, an SKG functions as an information hub, encompassing data, metadata, and content, providing a unified, consistent, and unambiguous view of data from various sources and systems. However, building a semantic knowledge graph is a complex task that requires a meticulous approach to ensure its effectiveness, scalability, and ability to meet application objectives. Therefore, this work proposes an approach based on the DIKW architecture for the construction and specification of Semantic Knowledge Graphs that integrate IoT data with heterogeneous data sources. In this context, we investigate three fundamental research questions that result in three significant contributions: Question 1: How to build an IoT data SKG that is compatible with the different levels of the DIKW architecture? Question 2: How to build an SKG that integrates IoT data with heterogeneous data sources? Question 3: How to describe an IoT data knowledge graph, making it accessible and understandable to users? In response to the first research question, this work introduces the “DIKW4IoT” approach, which provides the necessary framework to represent IoT information at different granularities, allowing data to evolve within the DIKW hierarchy. This empowers them to transform into valuable insights and ultimately trigger informed actions and decisions. Regarding the second question, we propose “SIGIoT-S”, a conceptual framework that guides which data can be used in the integration process and identifies the necessary steps to achieve seamless integration. Finally, concerning the third question, we have developed the “MetaGraphIoT” ontology. This ontology aims to annotate metadata in an IoT data knowledge graph, making the knowledge graph more interpretable and usable for users. These contributions address the identified research questions and represent significant advancements in IoT data management, promoting better utilization and understanding of this data in various applications.

Keywords: ontology; knowledge graph; DIKW hierarchy; internet of things - IoT; data integration; metadata.

LISTA DE FIGURAS

Figura 1 – Uma das possíveis arquitetura da Web Semântica	22
Figura 2 – Modularização vertical e horizontal da ontologia SSN	24
Figura 3 – Dados de sensores baseado na hierarquia <i>DIKW</i>	33
Figura 4 – Arquitetura em 4 Camadas do Grafo de Conhecimento de <i>IoT</i>	34
Figura 5 – Ilustração de um Grafo de Conhecimento de dados <i>IoT</i> com base na abordagem <i>DIKW4IoT</i>	35
Figura 6 – Ontologia <i>IoT</i> - Lite	36
Figura 7 – Ontologia <i>SOSA</i> - Perspectiva observação	37
Figura 8 – Ontologia <i>IoT-Stream</i> para anotação de fluxos agregados	38
Figura 9 – Ontologia <i>IoT-Stream</i> para anotação de eventos	39
Figura 10 – Anotação semântica referente a aplicação de gerenciamento de tráfego de veículos	40
Figura 11 – Ontologia para especificação de sinais de veículo	41
Figura 12 – Anotação semântica de fluxos de dados utilizados em um processo de análise de um classificador de aprendizagem de máquina	41
Figura 13 – Módulos do <i>BuildingIoTGraph</i>	44
Figura 14 – Anotação semântica dos dados de sensores	45
Figura 15 – Anotação semântica das observações de sensores	46
Figura 16 – Anotação semântica de dados e observações de sensores	46
Figura 17 – Anotação semântica de resultados de agregação de dados de sensores	47
Figura 18 – <i>AppMonitoraLisboa</i>	48
Figura 19 – Anotação semântica de eventos	49
Figura 20 – Integração de dados <i>IoT</i> com fontes de dados heterogêneas	52
Figura 21 – Construção de um Grafo de Conhecimento Semântico	54
Figura 22 – Bases dados para construção do GCS	59
Figura 23 – Ontologia Turismo	60
Figura 24 – Integração de subgrafos	61
Figura 25 – Amostra de nós de subgrafos integrados	63
Figura 26 – Metadados de um Grafo de Conhecimento de dados <i>IoT</i>	65
Figura 27 – Etapas da metodologia 101	66
Figura 28 – Classes para descrição das fontes de dados	70

Figura 29 – Classes para descrição da tarefa de anotação semântica dos dados	70
Figura 30 – Classes para descrição dos consumidores	71
Figura 31 – Classes para descrição do serviço de armazenamento dos dados	71
Figura 32 – Ontologia <i>MetaGrafoIoT</i>	73
Figura 33 – Operações realizadas pelo <i>buildingMetagraph</i>	74
Figura 34 – Exemplo de anotação semântica utilizando a ontologia <i>MetaGraphIoT</i> . . .	75
Figura 35 – Exemplo de um subgrafo do <i>Metagraph</i>	76
Figura 36 – Parte do resultado obtido pela consulta 2	77
Figura 37 – Parte do resultado obtido pela consulta 3	79
Figura 38 – Resultados obtidos das consultas 4, 5 e 6	80
Figura 39 – Resultados obtidos na consulta 7	82
Figura 40 – Resultados obtidos para responder a consulta 8	83
Figura 41 – Resultados obtidos para responder a consulta 9	83

LISTA DE QUADROS

Quadro 1 – Níveis de dados presentes no grafo de conhecimento	28
Quadro 2 – Ontologias utilizadas na construção da base de conhecimento <i>IoT</i>	28
Quadro 3 – Principais conceitos abordados pelos vocabulários explanados nos trabalhos relacionados	29
Quadro 4 – Vocabulários utilizados na construção da ontologia <i>MetaGraphIoT</i>	68

LISTA DE CÓDIGOS-FONTE

Código-fonte 1	– Consulta <i>Sparql</i> em subgrafos integrados	61
Código-fonte 2	– Consulta <i>Sparql</i> - fontes de dados	76
Código-fonte 3	– Consulta <i>Sparql</i> - Ontologias	77
Código-fonte 4	– Consulta <i>Sparql</i> - Classes	79
Código-fonte 5	– Consulta <i>Sparql</i> - objectProperty	79
Código-fonte 6	– Consulta <i>Sparql</i> - dataProperty	80
Código-fonte 7	– Consulta <i>Sparql</i> - armazenamento dos dados	81
Código-fonte 8	– Consulta <i>Sparql</i> - Aplicações	81
Código-fonte 9	– Consulta <i>Sparql</i> - fluxos de dados	82

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Objetivos	17
1.1.1	<i>Objetivo Geral</i>	18
1.1.2	<i>Objetivos Específicos</i>	18
1.2	Questões de pesquisa	18
1.3	Contribuições da pesquisa	18
1.4	Organização do Trabalho	19
1.5	Trabalho Aprovado em Conferência	19
2	FUNDAMENTAÇÃO TEÓRICA	21
2.1	Tecnologias da <i>Web Semântica</i> para integração de dados <i>IoT</i>	21
2.2	Ontologias utilizadas na anotação de dados de sensores	23
3	TRABALHOS RELACIONADOS	26
3.1	Integração de dados de <i>IoT</i>	26
3.2	Vocabulário para anotação semântica de metadados	27
3.3	Integração de dados de <i>IoT</i> com fontes de dados heterogêneas e externa	30
4	DIKW4IOT: CONSTRUINDO UM GRAFO DE CONHECIMENTO DE DADOS <i>IOT</i> BASEADO NA HIERARQUIA <i>DIKW</i>	32
4.1	Representação do conhecimento <i>IoT</i> baseado na hierarquia <i>DIKW</i>	32
4.2	Abordagem <i>DIKW4IoT</i> para construção de grafos de conhecimentos de dados <i>IoT</i>	33
4.2.1	<i>Anotação Semântica de Sensores</i>	35
4.2.2	<i>Anotação Semântica de Observações de Sensores</i>	37
4.2.3	<i>Anotação Semântica de Agregações de Dados de Sensores</i>	38
4.2.4	<i>Anotação Semântica do Conhecimento Indutivo e Dedutivo</i>	39
4.3	Estudo de Caso: Construindo um grafo de conhecimento a partir de parâmetros ambientais de Lisboa utilizando a abordagem <i>DIKW4IoT</i>	43
4.3.1	<i>Descrição do conjunto de dados</i>	43
4.3.2	<i>Construindo o grafo de conhecimento de dados <i>IoT</i></i>	44
4.3.3	<i>AppMonitoraLisboa: Uma aplicação mobile para consulta de informações relacionadas ao estado do tempo e qualidade do ar na cidade de Lisboa.</i>	47

5	SIGIOT-S: UM FRAMEWORK CONCEITUAL PARA INTEGRAÇÃO DE DADOS DE SENSORES COM FONTES DE DADOS HETEROGÊNEAS E EXTERNAS	51
5.1	Uma visão geral do <i>framework SIGIoT-S</i>	51
5.2	Primeira etapa: Construção do Grafo de conhecimento de dados <i>IoT</i>	52
5.3	Segunda etapa: Integração com fontes heterogêneas e externas	53
5.3.1	<i>Aquisição do Conhecimento</i>	53
5.3.2	<i>Modelagem da ontologia de domínio</i>	54
5.3.3	<i>Construção dos Grafos de conhecimentos Locais</i>	55
5.3.4	<i>Criação de visões de ligação Semântica</i>	56
5.3.5	<i>Inferência e Limpeza</i>	57
5.4	Estudo de Caso: Integração de parâmetros ambientais aos dados turísticos da cidade de Lisboa	58
6	METAGRAFO-IOT: UMA ONTOLOGIA PARA ANOTAÇÃO SEMÂNTICA DE METADADOS DE UM GRAFO DE CONHECIMENTO DE DADOS <i>IOT</i>	64
6.1	Metadados relacionados ao Grafo de Conhecimento de dados <i>IoT</i>	64
6.2	Metodologia utilizada na criação da ontologia <i>MetaGrafoIoT</i>	65
6.2.1	<i>Escopo da ontologia MetaGrafoIoT e questões de competência</i>	66
6.2.2	<i>Vocabulários reusados</i>	67
6.2.3	<i>Termos importantes na ontologia</i>	69
6.2.4	<i>Classes, propriedades, restrições e instâncias</i>	69
6.3	Metagraph: Construindo um grafo de conhecimento a partir de metadados - um estudo de caso	74
7	CONCLUSÕES E TRABALHOS FUTUROS	85
	REFERÊNCIAS	88

1 INTRODUÇÃO

Nos últimos anos, tem-se observado um aumento significativo na conectividade de dispositivos à internet, o que tem resultado em transformações significativas no cenário digital. Esses dispositivos, que incluem desde *smartphones* e *smartwatches* até sensores industriais e eletrodomésticos conectados, têm gerado volumes substanciais de dados continuamente.

A chave para esse fenômeno reside em avanços tecnológicos críticos, como a proliferação de redes de comunicação sem fio de alta velocidade e a evolução da computação de borda. A comunicação sem fio, incluindo tecnologias como 4G e 5G, tornou possível conectar dispositivos em praticamente qualquer lugar, eliminando a necessidade de cabos e facilitando a disseminação da conectividade (HAJLAOUI *et al.*, 2020).

Esses e outros avanços tecnológicos permitiu a criação de diversos ambientes de aplicação, abrangendo desde casas inteligentes e cidades inteligentes até fábricas e campos agrícolas (CHAE, 2019). Em uma casa inteligente, por exemplo, termostatos, câmeras de segurança, eletrodomésticos e assistentes virtuais estão todos interconectados, compartilhando informações para melhorar o conforto, a segurança e a eficiência energética. Em uma cidade inteligente, sensores podem monitorar o tráfego, a qualidade do ar, o uso de energia e até mesmo o nível da água em rios e córregos, ajudando a administrar recursos e serviços urbanos de forma mais eficaz.

À medida que esses ambientes de aplicação se multiplicam, a produção de dados para a Internet das Coisas (*IoT*) está crescendo exponencialmente. A combinação da vasta quantidade de dispositivos conectados com a capacidade de coletar, processar e analisar dados em tempo real está resultando em uma transformação fundamental no paradigma da *Web* (GAO *et al.*, 2021).

Essa nova realidade está impactando não apenas a interação com a tecnologia, mas também os processos operacionais das empresas, a prestação de serviços públicos pelos governos e a adaptação da sociedade à revolução digital. Conforme a Internet das Coisas continua a expandir-se, a capacidade de explorar e aproveitar as informações derivadas desses volumes significativos de dados desempenhará um papel crucial na formação do futuro, promovendo a inovação, a eficiência e a qualidade de vida em âmbito global.

A integração de dados de *IoT* com outras fontes de informações podem trazer diversas vantagens para aplicações em diferentes áreas, como saúde, agricultura, indústria, transporte, entre outras. O processo de enriquecimento dos dados de sensores por meio da integração

com fontes de dados externas permite a agregação de informações adicionais e estabelece relações significativas entre os dados coletados. Isso possibilita uma interpretação mais precisa e sofisticada dos dados de sensores, permitindo análises avançadas, detecção de padrões, tomadas de decisão mais informadas e a obtenção de *insights* valiosos. Esse enriquecimento dos dados amplia o potencial de aplicação dos dados *IoT*, promovendo uma visão mais abrangente e inteligente do ambiente em que os dispositivos estão inseridos.

Grafo de conhecimento Semântico (GCS) é uma solução dinâmica e escalável, já comprovada, para lidar com os requisitos de Integração de Dados em Larga Escala. Para tanto, um GCS atua como um *hub* para dados, metadados e conteúdo, oferecendo uma visão unificada, consistente e sem ambiguidades dos dados espalhados por diferentes fontes de dados e sistemas.

Ao criar uma descrição semântica comum, um grafo de conhecimento cria um nível mais elevado de abstração que não depende da infraestrutura física ou formato dos dados. Em vez disso, ele cria uma maneira unificada, de acessar de uma maneira contínua os dados de uma organização, bem como dados de terceiros ou globais. Os dados são representados de maneira uniforme, amigável e significativa para os seres humanos. Grafos de conhecimento usam metadados semânticos para fornecer uma visão uniforme e consistente sobre diversas fontes de dados, interligando o conhecimento que foi disperso em diferentes sistemas.

O coração do GCS construído com o padrão W3C, é um modelo de conhecimento (ou ontologia), no qual consiste em uma coleção de descrições interligadas de conceitos, entidades, relacionamentos e eventos onde:

- As descrições possuem semântica formal que permite tanto às pessoas quanto aos computadores processá-las de maneira eficiente e inequívoca;
- As descrições contribuem umas para as outras, formando uma rede, em que cada entidade representa parte da descrição das entidades relacionadas a ela;
- Dados diversos são conectados e descritos por metadados semânticos de acordo com o modelo de conhecimento.

As ontologias são adotadas como o esquema de dados do grafo, elas funcionam como um contrato entre os desenvolvedores do grafo de conhecimento e seus usuários em relação ao significado dos dados nele contidos. Um usuário pode ser outro ser humano ou um aplicativo de software que deseja interpretar os dados de maneira confiável e precisa. Ontologias garantem um entendimento compartilhado dos dados e de seus significados.

No contexto da Internet das Coisas (*IoT*), o Grafo de Conhecimento Semântico

e construído pela integração de vários sensores e atuadores em um ambiente, o que amplia significativamente a capacidade de adquirir informações complexas. O uso de ontologia em dados de sensores possibilita a criação de uma base de conhecimento. Possibilita a redução de esforço do usuário em conhecer os diferentes esquemas de dados, assim como, a redução de esforços na elaboração de diferentes estruturas para transformação dos dados (STEINDL *et al.*, 2019).

A integração contínua de dados em um grafo de conhecimento representa um passo crucial na busca pelo conhecimento desejado. Esse conhecimento emerge da interconexão entre diversas informações. No contexto do grafo de conhecimento, as informações são adquiridas por meio das complexas teias de relações e conceitos aplicados aos dados. Essa hierarquia, que segue a estrutura *DIKW* (*Data - Information - Knowledge - Wisdom*), também conhecida como a Pirâmide do Conhecimento, é fundamental para compreender a evolução do dado até o nível mais alto de sabedoria.

O processo de integração de dados desempenha um papel importante na busca por essas diferentes camadas da Pirâmide do Conhecimento. Ele capacita a transformação dos dados em informações de valor e, subsequentemente, em conhecimento. Esse conhecimento, por sua vez, é o combustível essencial para a tomada de decisões informadas, que pode ser definida como a verdadeira manifestação da sabedoria.

Essas diversas camadas de abstração de dados presentes no grafo de conhecimento de dados *IoT* ampliam significativamente suas possibilidades de aplicação. Elas tornam o grafo uma ferramenta versátil, adequada tanto para aplicações altamente especializadas quanto para aquelas que demandam o uso dos dados em sua menor unidade.

Entretanto, a construção de um Grafo de Conhecimento Semântico que integra dados de *IoT* com fontes de dados heterogêneas é uma tarefa complexa que exige uma abordagem metódica para garantir sua eficácia, escalabilidade e capacidade de atender aos objetivos das aplicações. Assim sendo, este trabalho propõe uma abordagem fundamentada na arquitetura *DIKW* para a construção e especificação de Grafos de Conhecimento Semântico que integram dados *IoT* com fontes de dados heterogêneas.

1.1 Objetivos

Diante do problema apresentado, essa pesquisa possui os seguintes objetivos:

1.1.1 *Objetivo Geral*

Desenvolver técnicas que viabilizem a construção de um Grafo de Conhecimento Semântico para dados *IoT*, abrangendo diferentes níveis de dados. Essas técnicas permitirão uma integração eficiente do grafo com fontes externas e heterogêneas de dados. Além disso, busca-se elaborar uma ontologia que descreva os metadados associados a esse grafo, com o propósito de torná-lo acessível e de fácil compreensão para os usuários finais.

1.1.2 *Objetivos Específicos*

Como objetivos específicos têm se:

1. Elaborar uma abordagem que viabilize a construção de um Grafo de Conhecimento para dados *IoT*, com a capacidade de estruturar os dados em diversos níveis de abstração.
2. Identificar as principais ontologias no contexto *IoT* que possam ser utilizadas na anotação semântica dos dados.
3. Desenvolver estratégia eficaz para a integração de dados da *IoT* com fontes heterogêneas e externas.
4. Propor ontologia para descrever os metadados de um grafo de conhecimento de dados *IoT*.

1.2 **Questões de pesquisa**

Com base nos objetivos estabelecidos, as seguintes questões de pesquisa orientarão este estudo:

1. **Questão 1:** Como construir um GCS de dados *IoT* que seja compatível com os diferentes níveis da arquitetura *DIKW*?
2. **Questão 2:** Quais estratégias podem ser empregadas para integrar eficazmente um grafo de conhecimento de dados *IoT* com fontes de dados heterogêneas e externas?
3. **Questão 3:** De que forma podemos descrever um grafo de conhecimento de dados *IoT*, tornando-o acessível e compreensível para os usuários?

1.3 **Contribuições da pesquisa**

Com base nas questões de pesquisa definidas e nos objetivos estabelecidos, esta pesquisa oferece as seguintes contribuições:

1. *DIKW4IoT*: uma abordagem para construir um grafo de conhecimento de dados *IoT* com base na hierarquia *DIKW*.
2. *SIGIoT-S*: um *framework* conceitual para a integração de dados de sensores com fontes de dados heterogêneas e externas.
3. *MetaGraphIoT*: uma ontologia para anotação de metadados de um grafo de conhecimento de dados *IoT*.

1.4 Organização do Trabalho

Esta dissertação é composta por sete capítulos, incluindo esta introdução que estabelece o contexto do trabalho, delinea os objetivos, formula as questões de pesquisa e destaca as contribuições desta pesquisa.

O Capítulo 2 apresenta o referencial teórico, abordando as principais tecnologias aplicadas na Web Semântica e as ontologias fundamentais no contexto da Internet das Coisas (*IoT*).

No Capítulo 3, é realizada uma análise de trabalhos relacionados à temática, com o objetivo de expor e comparar esses estudos com a nossa proposta, identificando semelhanças e diferenças.

O Capítulo 4 apresenta a abordagem *DIKW4IoT* para a construção de grafos de conhecimento de dados *IoT*, contemplando diferentes níveis de dados, com base na hierarquia *DIKW*.

O Capítulo 5 introduz o *framework* conceitual *SIGIoT-S*, elaborado para a integração eficiente de um grafo de conhecimento de dados *IoT* com fontes de dados heterogêneas e externas.

No Capítulo 6, detalha-se a criação da ontologia *MetaGraphIoT*, destinada à anotação de metadados em um grafo de conhecimento de dados *IoT*.

Finalmente no Capítulo 7, apresenta-se as conclusões deste trabalho, bem como as direções para pesquisas futuras.

1.5 Trabalho Aprovado em Conferência

Trabalho aprovado no Simpósio Brasileiro de Banco de Dados - SBBD 2022:

- *DIKW4IoT*: Uma abordagem baseada na hierarquia *DIKW* para a construção de grafos de

conhecimento para integração de dados de *IoT*. DOI: <https://doi.org/10.5753/sbbd.2022.224648>

2 FUNDAMENTAÇÃO TEÓRICA

Com a proliferação das tecnologias de sensores e Redes de Sensores Sem Fio (RSSF) muitos dados têm sido gerados. Devido a heterogeneidade das tecnologias utilizadas, os dados de sensores apresentam problemas relacionados a heterogeneidade semântica e sintática. A falta de uma semântica comum entre os dados inviabiliza a interoperabilidade entre as aplicações e reforça um problema existente que é a produção de muitos dados, todavia a geração de pouco conhecimento (DJEZZAR *et al.*, 2018).

Este capítulo tem como proposta apresentar na seção 2.1 as principais tecnologias aplicadas na *web* semântica para integração de dados, assim como, na seção 2.2 mostrar um breve resumo sobre as principais ontologias utilizadas no contexto de *IoT* para anotação semântica dos dados.

2.1 Tecnologias da *Web Semântica* para integração de dados *IoT*

Com a evolução da Internet das Coisas (*IoT*) e o aumento de cenários e aplicações que utilizam dados de sensores, atuadores e dispositivos inteligentes, um novo ecossistema digital tem se formado. Esse novo ecossistema é baseado em uma ampla variedade de sistemas e tecnologias heterogêneas que não possuem um modelo de informação padronizado para o processamento dos dados.

Algumas pesquisas (ALVAREZ-COELLO *et al.*, 2021) (KELLER, 2019) têm se utilizado de tecnologias da *web* semântica a fim de alcançar a interoperabilidade semântica. O objetivo da *web* semântica é permitir que os dados sejam processáveis e significativos tanto para máquinas quanto para humanos, e dada a perspectiva de *IoT*, que possamos ter “Coisas conectadas a *web*” (SZILAGYI; WIRA, 2016).

Conforme (LIMA; CARVALHO, 2005) a arquitetura da *web* semântica, de forma geral, pode ser dividida em três camadas: camada de estrutura, camada de esquema e camada lógica, como é apresentado na figura 1.

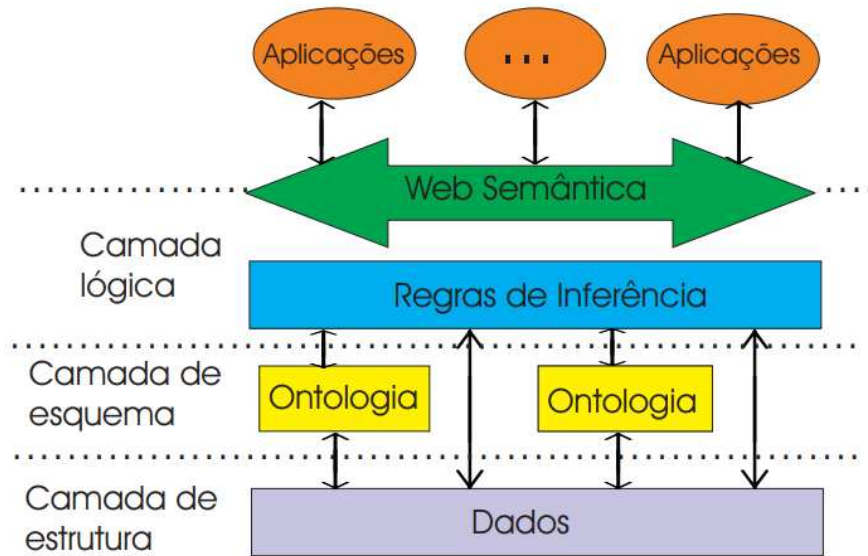
A camada de estrutura é responsável por estruturar e atribuir significado aos dados. Para isso utilizam as tecnologias *XML* (*Extensible Markup Language*)¹, *RDF* (*Resource Description Framework*)² e *RDF Schema*³ para sua elaboração.

¹ <https://www.w3.org/XML/>

² <https://www.w3.org/RDF/>

³ <https://www.w3.org/TR/rdf-schema/>

Figura 1 – Uma das possíveis arquitetura da Web Semântica



Fonte: adaptado de (LIMA; CARVALHO, 2005)

O *Resource Description Framework (RDF)* é um *framework* para representar informações na *web*, o qual foi estabelecido pelo *W3C (World Wide Web Consortium)*. O *RDF* possui um modelo de dados simples que é baseado em uma estrutura de grafo. Tem como resultado uma tripla que consiste em sujeito, predicado e objeto.

Cada parte da tripla pode ser representada por meio de um identificador de recurso uniforme (*URI*). O sujeito implica em informações relacionadas ao recurso apresentado, o predicado refere-se a uma característica ou propriedade do sujeito e o objeto consiste no valor desta descrição, podendo apresentar valor de um dado literal ou uma *URI*, a qual aponta para outro sujeito (KIM *et al.*, 2021).

No contexto de *IoT* tem-se o modelo de dados *RDF stream*. A comunidade *RSP W3C*⁴ define que um *RDF stream* é uma sequência infinita de eventos com informações de tempo vinculada. Estas informações não são decrescentes e podem ser representadas como uma sequência ordenada, a descrição de um instante de tempo ou a descrição de um intervalo de tempo.

Sendo assim um *RDF stream* pode ser representado como:

$$((s, p, o)_{t_1}) \text{ ou } ((s, p, o), t_i, t_f),$$

em que s, p, o significam respectivamente sujeito, predicado e objeto, t_1 representa o instante em que um evento ocorreu e t_i e t_f representam o intervalo de tempo em que um evento

⁴ <https://www.w3.org/community/rsp/>

ocorreu.

No quesito formato dos dados, diferentes formatos podem ser adotados para que haja o intercâmbio dos dados, como exemplo pode-se citar os formatos *Turtle*, *RDF/XML*, *N3*, *N-Triples*.

Na camada de esquema é utilizado ontologias para projetar e representar o conhecimento sobre algo, assim como definir um conjunto de regras de inferência. As ontologias são definidas por meio da linguagem *OWL (Web Ontology Language)*⁵.

A linguagem *OWL* consiste em uma linguagem de marcação semântica projetada para representar o conhecimento rico e complexo sobre coisas, grupo de coisas e suas relações⁶. Por meio da linguagem *OWL* é possível definir classes, propriedades, indivíduos e afirmações sobre estes. Assim como, pode-se definir axiomas lógicos que permitem a identificação de novos fatos sobre estes dados quando realizado o processo de inferência sobre os dados (LIMA; CARVALHO, 2005).

Uma ontologia refere-se a um conjunto de declarações descritivas e precisas sobre alguma coisa ou domínio de interesse definindo formalmente um vocabulário com um conjunto de termos centrais com seus significados e relações⁷. Foi desenvolvida como uma extensão do vocabulário *RDF* permitindo mais expressividade em suas relações e definições. Sendo assim uma ontologia *OWL* é um grafo *RDF* que consiste em um conjunto de triplas *RDF*.

A camada lógica é responsável pelos mecanismos que verificam e validam as regras de inferência que foram definidas sobre os dados (LIMA; CARVALHO, 2005). Acima da camada lógica têm-se as aplicações que fazem uso dos dados.

2.2 Ontologias utilizadas na anotação de dados de sensores

Ontologias podem ser utilizadas como modelos formais para atribuição de semântica aos dados. São modeladas por meio de classes, propriedades e relações bem definidas e permitem uma interpretação precisa dos dados.

A ontologia *SSN (Semantic Sensor Network)* é considerada o *backbone* das ontologias que descrevem sensores. Sua terminologia possui conceitos para descrever sensores e observações, não possuindo informações que descrevam o tempo, locais ou domínios.

⁵ <https://www.w3.org/OWL/>

⁶ <https://www.w3.org/OWL/>

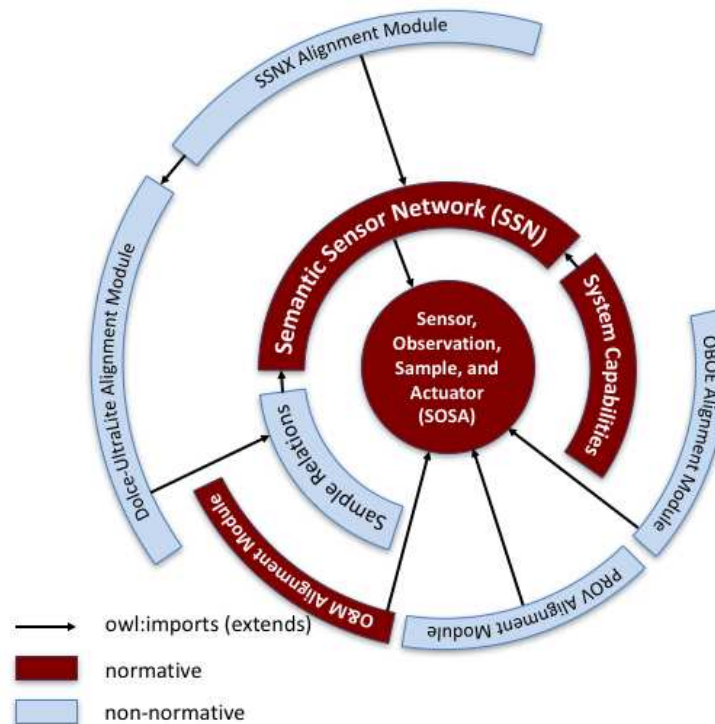
⁷ <https://www.w3.org/TR/owl2-primer/>

A ontologia *SSN* foi desenvolvida de forma modularizada. A modularização de ontologias é um método utilizado para segmentar uma ontologia em partes menores, fazendo com que cada subconjunto da ontologia tenha um escopo menor e bem definido.

No caso da ontologia *SSN* adotou-se a modularização vertical para a maioria dos módulos ontológicos. A modularização vertical realiza a importação de outras ontologias por meio da declaração *owl:imports*. Os módulos são construídos uns sobre os outros, em que os módulos de nível inferior são independentes dos módulos de nível superior. Neste caso, a ontologia importadora assume o significado dos termos importados, entretanto a ontologia importada não herda nenhuma semântica de quem a importou (HALLER *et al.*, 2019).

A modularização horizontal, é aplicada apenas nos módulos *Sample Relations* e o *System Capabilities*. Na modularização horizontal existe uma independência entre os módulos (HALLER *et al.*, 2019). A figura 2 apresenta a modularização da ontologia *SSN*.

Figura 2 – Modularização vertical e horizontal da ontologia *SSN*



Fonte: Adaptado de (HALLER *et al.*, 2019)

O módulo central é composto pela ontologia *SOSA* (*Sensor, Observation, Sample, and Actuator*). A ontologia *SOSA* adota uma perspectiva centrada em eventos e fornece uma especificação de propósito geral abordando conceitos relacionados a observação, atuação, amostragem e procedimentos (JANOWICZ *et al.*, 2019). Com o intuito de ser mais leve sua axiomatização é mais limitada. A anotação semântica por meio de conceitos da ontologia *SOSA*

pode ser realizada a partir de 03 (três) perspectivas: observação, amostragem e atuação.

Visando reduzir o tempo excessivo de processamento ao consultar a ontologia *SSN*, a ontologia *IoT-Lite*⁸ foi proposta. Esta constitui uma instanciação leve da ontologia *SSN*. Tem como foco descrever os recursos de sensoriamento. Desta forma, aborda os principais conceitos de *IoT* distribuídos em três classes: objetos, sistemas ou recursos e serviços. A proposta da ontologia *IoT-Lite* é ser uma ontologia leve e que permita realizar o tempo de processamento da anotação semântica mais rápido (BERMUDEZ-EDO *et al.*, 2017).

A ontologia *IoT-Lite* pode ser combinada com outras ontologias, como a *SAO*⁹ que é utilizada para representar fluxos de dados *IoT*. A ontologia *SAO* aborda conceitos temporais como *streamData*, *streamAnalyses*, *Segment* entre outros.

Entretanto, algumas aplicações que fazem uso de dados de sensores precisam realizar o processamento dos dados em um tempo próximo ao tempo real que os dados foram coletados. As ontologias que apresentam muitos conceitos e descrevem em detalhes os recursos presentes culminam em algum atraso nesse tempo de processamento. Devido a essa questão foi proposta a ontologia *IoT-Stream*¹⁰.

A ontologia *IoT-Stream* tem como objetivo realizar a anotação do fluxo de dados *IoT*. Propõe um modelo de informação simples em que separa as informações do fluxo de dados dos metadados dos sensores. A ontologia pode ser utilizada para anotação de dados brutos ou dados processados. A anotação torna-se leve por remeter apenas ao valor de uma observação e uma anotação temporal na forma de um *timestamp*, para dados brutos, ou um intervalo de uma janela de tempo, para dados processados (ELSALEH *et al.*, 2020).

Em relação a ontologias voltadas para um domínio específico, o catálogo *Lov4IoT* (*Linked Open Vocabularies for Internet of Things*)¹¹ reúne várias ontologias que seguem boas práticas da *web* semântica e que são aplicadas a diferentes domínios abordados na internet das coisas. Neste, as ontologias são classificadas e organizadas de acordo com o seu domínio, por exemplo: Saúde, Meio ambiente, Casa inteligente, etc.

⁸ <https://www.w3.org/Submission/2015/SUBM-iot-lite-20151126/>

⁹ <http://iot.ee.surrey.ac.uk/citypulse/ontologies/sao/sao#>

¹⁰ <http://iot.ee.surrey.ac.uk/iot-crawler/ontology/iot-stream/>

¹¹ <https://lov4iot.appspot.com/>

3 TRABALHOS RELACIONADOS

Neste capítulo, apresentam-se os trabalhos relacionados à pesquisa desenvolvida. A Seção 3.1 aborda as pesquisas relacionadas à integração de dados de IoT. Na Seção 3.2, discute-se os trabalhos relacionados à anotação semântica de metadados. Já na Seção 3.3, serão discutidos os trabalhos que tratam da integração de dados IoT com fontes de dados heterogêneas e externas.

3.1 Integração de dados de IoT

A pesquisa em (LE-PHUOC *et al.*, 2016) apresenta uma arquitetura para construção de um grafo de conhecimento a partir de dados de sensores. A arquitetura proposta é formada por quatro camadas: *aquisição dos dados*, *grafo das coisas*, *acesso aos dados e aplicação*. O processo de anotação semântica consiste em atribuir termos e relacionamentos específicos da ontologia *SSN* aos dados de sensores e suas observações.

O trabalho de (SOARES *et al.*, 2019) aborda a integração de dados relacionados ao domínio marítimo. Propõe um *framework* denominado como *Crisis*. O *framework* é composto por quatro módulos: mapeamento dos dados, espaço de informação, motores de consulta e canais de dados. A anotação semântica dos dados utiliza a ontologia *SSN* e uma ontologia voltada para o domínio marítimo.

O trabalho de (DING *et al.*, 2020) apresenta um *framework* baseado em ontologia para integração e análise de dados. O *framework* é composto por dois módulos: *OBDI* e *GeoVa*. Na anotação semântica dos dados é utilizado a ontologia *SSN* para descrever os sensores e observações.

O trabalho de (ALVAREZ-COELLO; GÓMEZ, 2021) realizam a integração de dados de sensores baseada em ontologia no domínio automotivo. Diferente dos demais trabalhos, tem como proposta a anotação semântica dos resultados de uma análise efetuada por uma aplicação. Estes resultados de análise compreendem o valor de uma média, o desvio padrão e soma dos valores das observações de um sensor dentro de uma janela de tempo instituída, como também a anotação de eventos relacionados a mudanças que ocorrem em um fluxo categórico. Para isso, utilizam a ontologia *IoT-Stream*¹ e a ontologia *VSSo*² para anotação semântica dos dados.

Os trabalhos foram analisados buscando entender o processo de construção do grafo

¹ <http://iot.ee.surrey.ac.uk/iot-crawler/ontology/iot-stream/>

² <https://www.w3.org/TR/vsso/>

de conhecimento de dados *IoT*. Nessa análise avaliou-se a construção do grafo de conhecimento permitia ou fazia menção a presença de diferentes níveis de dados. Adotou-se a hierarquia *DIKW* como padrão para definição dos seguintes níveis de dados: dados, informação, conhecimento e sabedoria. Em relação ao nível “Dados”, considerou-se como dados as descrições dos sensores nos grafos e não os dados brutos em si. No que se refere ao nível “Sabedoria”, entende-se que a partir da análise do conhecimento se faz possível efetuar uma tomada de decisão ou tornar uma ação automatizada de algum sistema.

O Quadro 1 apresenta os níveis de dados identificados em cada trabalho. Os trabalhos de (LE-PHUOC *et al.*, 2016), (SOARES *et al.*, 2019), (DING *et al.*, 2020) contemplam os níveis de dados referentes a “Dados” e “Informação”. Os grafos construídos por estes trabalhos realizaram a anotação semântica das descrições dos sensores e de suas observações. Não foram realizadas abstrações em um nível de granularidade maior nestes dados. Assim como também não foram anotados semanticamente resultados advindos de análise de dados ou eventos. Sendo assim, não contemplam o nível de dados referente ao conhecimento.

O trabalho de (ALVAREZ-COELLO; GÓMEZ, 2021), por sua vez, realizou uma abordagem contrária as anteriores. O processo de anotação semântica é realizado sobre os resultados de análise e agregação de dados. Todavia, o grafo não possui a anotação semântica dos dados de sensores e suas respectivas observações. Seguindo a análise inicial, o trabalho de (ALVAREZ-COELLO; GÓMEZ, 2021) contemplaria o nível de dados relacionado ao “Conhecimento” quando efetua a anotação semântica dos resultados de análises de dados, e possui o nível de dados relacionado a “Informação”. Entretanto, o grafo não possui descrições dos sensores. A abordagem *DIKW4IoT* objetiva construir o grafo de conhecimento com os níveis de dados: Dados, Informação e Conhecimento.

O Quadro 2 refere-se ao levantamento dos vocabulários utilizados na anotação semântica dos dados. As ontologias predominantes nos trabalhos são: *SSN* e *IoT-Stream*. A abordagem proposta *DIKW4IoT* propõe uma ontologia específica para cada nível de dados, de maneira que estas ontologias estejam relacionadas entre si.

3.2 Vocabulário para anotação semântica de metadados

Nesta seção serão apresentados alguns trabalhos que realizam a anotação semântica de metadados relacionados a repositório de dados, proveniência, serviços de *streaming*, etc.

A pesquisa de (TOMMASINI *et al.*, 2018) apresenta um vocabulário para descrever

Quadro 1 – Níveis de dados presentes no grafo de conhecimento

Trabalhos	Dados	Informação	Conhecimento
(LE-PHUOC <i>et al.</i> , 2016)	✓	✓	-
(DING <i>et al.</i> , 2020)	✓	✓	-
(SOARES <i>et al.</i> , 2019)	✓	✓	-
(ALVAREZ-COELLO; GÓMEZ, 2021)	-	✓	✓
DIKW4IoT	✓	✓	✓

Fonte: elaborado pelo autor.

Quadro 2 – Ontologias utilizadas na construção da base de conhecimento *IoT*

Trabalhos	Dados	Informação	Conhecimento
(LE-PHUOC <i>et al.</i> , 2016)	SSN	SSN	-
(DING <i>et al.</i> , 2020)	SSN e geoSparql	SSN e geoSparql	-
(SOARES <i>et al.</i> , 2019)	SSN e ontologia de domínio	SSN e ontologia de domínio	-
(ALVAREZ-COELLO; GÓMEZ, 2021)	-	-	IoT-Stream e VSSo
DIKW4IoT	IoT-Lite	Sosa e IoT-Stream	IoT-Stream e ontologia da aplicação

Fonte: elaborado pelo autor.

fluxos de dados, serviços de *streaming* e rastrear a proveniência do processamento de fluxo. O vocabulário é denominado como *Vocals (Vocabulary Catalog of Linked Streams)* e tem como pretensão atuar na formação de um catálogo de *Linked Streams* representando o conhecimento relacionado à publicação e consumo de fluxos de dados na *Web*. Desta forma, os consumidores podem descobrir e selecionar fluxos de dados que são relevantes para suas tarefas.

O trabalho abordado por (DIBOWSKI; SCHMID, 2021) propõe um vocabulário denominado como *DCPAC (Catálogo de Dados, Proveniência e Controle de Acesso)* que fornece a descrição semântica do conteúdo, proveniência e permissões de controle de acesso dos recursos em um *data lake*. A proposta é utilizar o vocabulário como uma camada semântica em um *data lake*.

O trabalho de (ARRUDA *et al.*, 2020) apresenta um vocabulário para especificar uma visão semântica. Este vocabulário é dividido em três categorias: metadados gerais, metadados de especificação e metadados de qualidade. A categoria metadados gerais possui termos para descrever um conjunto de dados *RDF*. A categoria metadados de especificação possui conceitos para especificar uma visão semântica. A categoria de metadados de qualidade possui termos para especificar a qualidade de uma visão semântica.

O uso de uma ontologia para anotação de metadados é essencial para estruturar e

enriquecer a descrição de informações específicas dentro de um domínio, isso permite que os dados se tornem mais acessíveis e úteis.

Para analisar os trabalhos referente aos vocabulários para anotação semântica dos metadados definiu-se alguns critérios com base no que seria necessário para descrição de um grafo de conhecimento *IoT* e sua manutenção. Os critérios estabelecidos foram: fluxos de dados, proveniência, mapeamento, ontologia, entidade de armazenamento de dados e consumidores.

Os trabalhos aqui citados (TOMMASINI *et al.*, 2018), (DIBOWSKI; SCHMID, 2021) e (ARRUDA *et al.*, 2020) possuem finalidades específicas como descrever fluxos de dados na *web*, descrever um *data lake* e descrever uma visão semântica, respectivamente. Estas finalidades se relacionam com o trabalho proposto, mas não o contemplam em sua totalidade.

O vocabulário proposto por (TOMMASINI *et al.*, 2018) possui conceitos que podem ser usados na descrição dos fluxos de dados que constituem o grafo de conhecimento, todavia, não possui conceitos para descrição de mapeamentos, ontologias e etc.

O vocabulário proposto por (DIBOWSKI; SCHMID, 2021) possui termos e relações para descrever repositório de dados, o que podem ser utilizados para descrição das entidades de armazenamento do grafo de conhecimento. Entretanto não possui conceitos para descrever ontologias, mapeamentos, regras de integração de dados, e etc, justamente por ser conceitos que não estão presentes em seu escopo.

Quadro 3 – Principais conceitos abordados pelos vocabulários explanados nos trabalhos relacionados

Trabalhos	Fluxos de dados	Proveniência	Mapeamento	Ontologia	Entidade de armazenamento de dados	Consumidores
(TOMMASINI <i>et al.</i> , 2018)	✓	✓	-	-	-	-
(DIBOWSKI; SCHMID, 2021)	✓ (descreve um <i>dataset</i>)	✓	-	-	✓	-
(ARRUDA <i>et al.</i> , 2020)	✓ (descreve uma fonte de dados)	-	✓	✓	-	-
MetaGraphIoT	✓	✓	✓	✓	✓	✓

Fonte: elaborado pelo autor.

O vocabulário proposto por (ARRUDA *et al.*, 2020) possui escopo voltado para grafos de conhecimentos. Possui conceitos que permitem descrever *mashup* de dados, ontologias, mapeamentos, regras de *linkage*, entre outros. Contudo, não possui conceitos para descrição de entidades de armazenamento, ferramentas utilizadas para aquisição e publicação dos dados,

informações relacionadas a proveniência dos dados, assim como informações relacionadas aos consumidores. Mesmo tendo o escopo voltado para grafos de conhecimentos não se refere a grafos de conhecimento construídos a partir de dados de *IoT*.

Entende-se e reconhece-se a completude dos vocabulários citados para o escopo para o qual foram criados, entretanto não contemplam em totalidade o escopo de um grafo de conhecimento de dados *IoT*. Até o momento não foram encontrados trabalhos relacionados ao gerenciamento de metadados de um grafo de conhecimento de dados *IoT*. O quadro 3 apresenta uma comparação dos principais conceitos abordados pelos vocabulários e pela ontologia proposta nesta pesquisa.

3.3 Integração de dados de *IoT* com fontes de dados heterogêneas e externa

O trabalho de (LE-PHUOC *et al.*, 2016) apresenta uma arquitetura em quatro camadas para a construção do grafo de conhecimento de dados *IoT*. A camada definida como *Graph of Things - GOT* representa a camada responsável pela anotação semântica dos dados, pela vinculação deste dados a *linked data Cloud* e pelo serviço de armazenamento dos dados. O trabalho relata a importância do enriquecimento dos dados de sensores, propriedades e recursos estarem ligados a recursos externos, como o *DBPedia*, todavia não demonstra como essa ligação poderá ser efetuadas e quais dados devem ser considerados neste processo.

O trabalho de (SOARES *et al.*, 2019) apresenta um *framework* dividido em cinco passos para a construção e consumo do grafo de conhecimento de dados *IoT*. As bases de dados externas não estruturadas ou semi-estruturadas são mapeadas para *RDF*, assim como os dados dos sensores. Nestes são usados a ontologia *SSN* e a ontologia de domínio. O *Framework* apresenta também um passo dois denominado como espaço de informação. Neste passo deve ser realizado a vinculação dos dados a *Linked Open data*, ou seja, deve-se criar *links* para grafos ou bases de dados externas como *DBPedia*, *Geonames*, etc. Os passos apresentado no *framework* de (SOARES *et al.*, 2019) apresenta o processo de forma geral, entretanto não especifica quais dados e de que forma deverão ser vinculados as fontes de dados externas e heterogêneas.

Em resumo, as novas contribuições do presente trabalho em comparação com as abordagens do estado da arte são:

1. A definição de uma abordagem para criação de um grafo de conhecimento de dados *IoT* com diferentes níveis de dados (dados, informação e conhecimento).
2. A criação de estratégias para integração do grafo de conhecimento de dados *IoT* com

fontes de dados externas e heterogêneas.

3. A definição de um vocabulário semântico para descrição de uma grafo de conhecimento de dados *IoT*.

4 DIKW4IOT: CONSTRUINDO UM GRAFO DE CONHECIMENTO DE DADOS IOT BASEADO NA HIERARQUIA DIKW

A abordagem *DIKW4IoT* propõe a construção de um grafo de conhecimento de dados *IoT* que leve em consideração as camadas de dados presentes na hierarquia *DIKW*. Deste modo, a Seção 4.1 apresenta como os dados de sensores podem ser representados segundo a hierarquia *DIKW*. A Seção 4.2 apresenta como o grafo de conhecimento de dados *IoT* pode ser construído mantendo os níveis de dados estabelecidos na hierarquia *DIKW*. A Seção 4.3 apresenta um estudo de caso sobre a aplicação da abordagem *DIKW4IoT* na construção de um grafo de conhecimento de parâmetros ambientais da cidade de Lisboa.

4.1 Representação do conhecimento *IoT* baseado na hierarquia *DIKW*

A Pirâmide do Conhecimento também conhecida como hierarquia *DIKW* (*Data-Information-Knowledge-Wisdom*) classifica os dados em 4 (quatro) camadas, são elas: dados, informação, conhecimento e sabedoria (METER, 2020). Esta classificação é realizada a partir das características dos dados e como estes estão estruturados.

A camada de **dados** da hierarquia *DIKW* estabelece que os dados são símbolos gravados e leitura de sinais, em que os símbolos são representados por palavras, números, diagramas e imagens e os sinais incluem sensores e leituras sensoriais de luz, som, etc (LIEW, 2013). Sendo apresentados como símbolos observáveis do mundo e que sem contexto não apresentam nenhum significado.

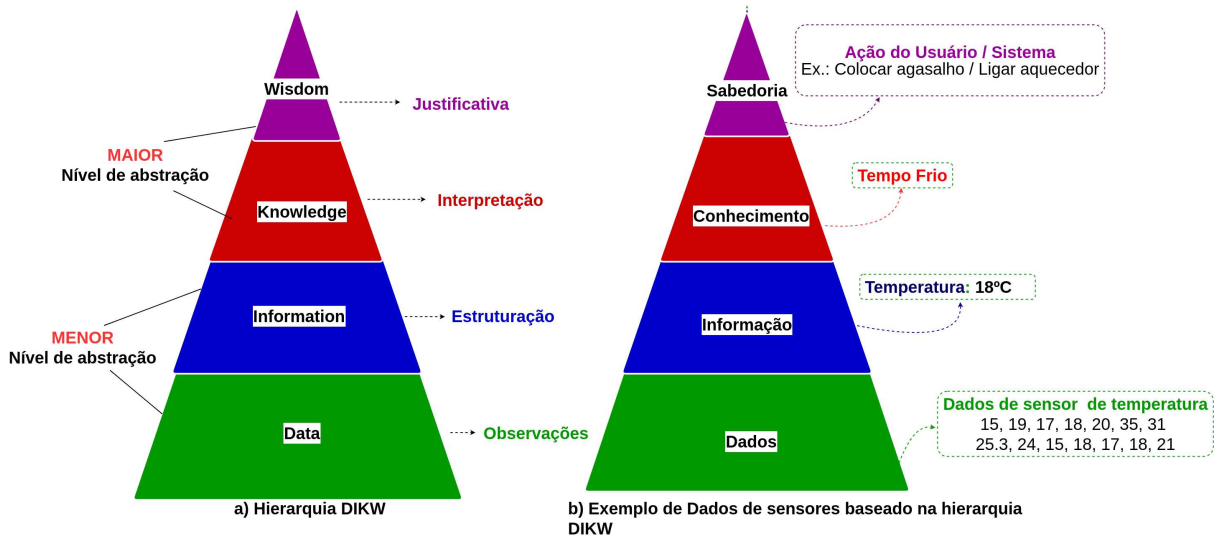
A **informação**, por sua vez, refere-se a uma mensagem com significado mais relevante e que pode ser utilizada como entrada para decisão e/ou ação, ou seja, a informação são dados que por meio de conexões relacionais e contextuais receberam mais significado (DUAN *et al.*, 2017).

O **conhecimento** refere-se a capacidade de agir e a compreensão em relação a algo, isto é, a partir da informação recebida, processada e validada cognitivamente o conhecimento é construído (DUAN *et al.*, 2017).

A **sabedoria** está relacionada a possibilidade de avaliar um cenário e discernir entre o certo e o errado, sendo capaz de projetar as consequências de uma ação (REMOR *et al.*, 2017).

A Figura 3.a. apresenta um exemplo da abstração dos dados segundo a hierarquia *DIKW*. A Figura 3.b apresenta um exemplo da classificação de dados de sensores segundo essa hierarquia. Supondo que os dados de sensores podem ser representados e organizados com base

Figura 3 – Dados de sensores baseado na hierarquia *DIKW*



Fonte: a) Hierarquia *DIKW* - adaptado de (ROBERTO, 2021) b) Dados de sensores baseado na hierarquia *DIKW* - elaborado pelo autor

na hierarquia *DIKW*, uma observação de um sensor representa o nível mais básico da hierarquia - a camada de dados. Por si só, as leituras dos sensores não possuem significado ou contexto. Para descrevê-los, os sensores possuem metadados que incluem informações como sua unidade de medida, localização e tipo de sensor. Quando essas descrições são associadas às observações, elas evoluem para uma informação que pode ser interpretada. Por exemplo, podemos saber que a temperatura é de 18 °C e que a velocidade do vento é de 35 km/h.

A próxima camada na hierarquia *DIKW* é a camada de conhecimento, onde a informação é analisada ou processada para deduzir algo a partir dela. Por exemplo, a partir da análise dos dados de temperatura e velocidade do vento em um período de tempo estabelecido, podemos concluir que o tempo está frio ou estará frio em determinado período, e assim obter o evento “Tempo Frio”.

O conhecimento obtido a partir das informações pode levar à sabedoria. Por exemplo, se alguém sabe que o clima em uma cidade para a qual irá viajar está frio, pode decidir levar roupas mais quentes na bagagem. Da mesma forma, um sistema de casa inteligente que possua informações sobre as condições climáticas externas pode ligar automaticamente o aquecedor da casa se detectar que o clima está frio.

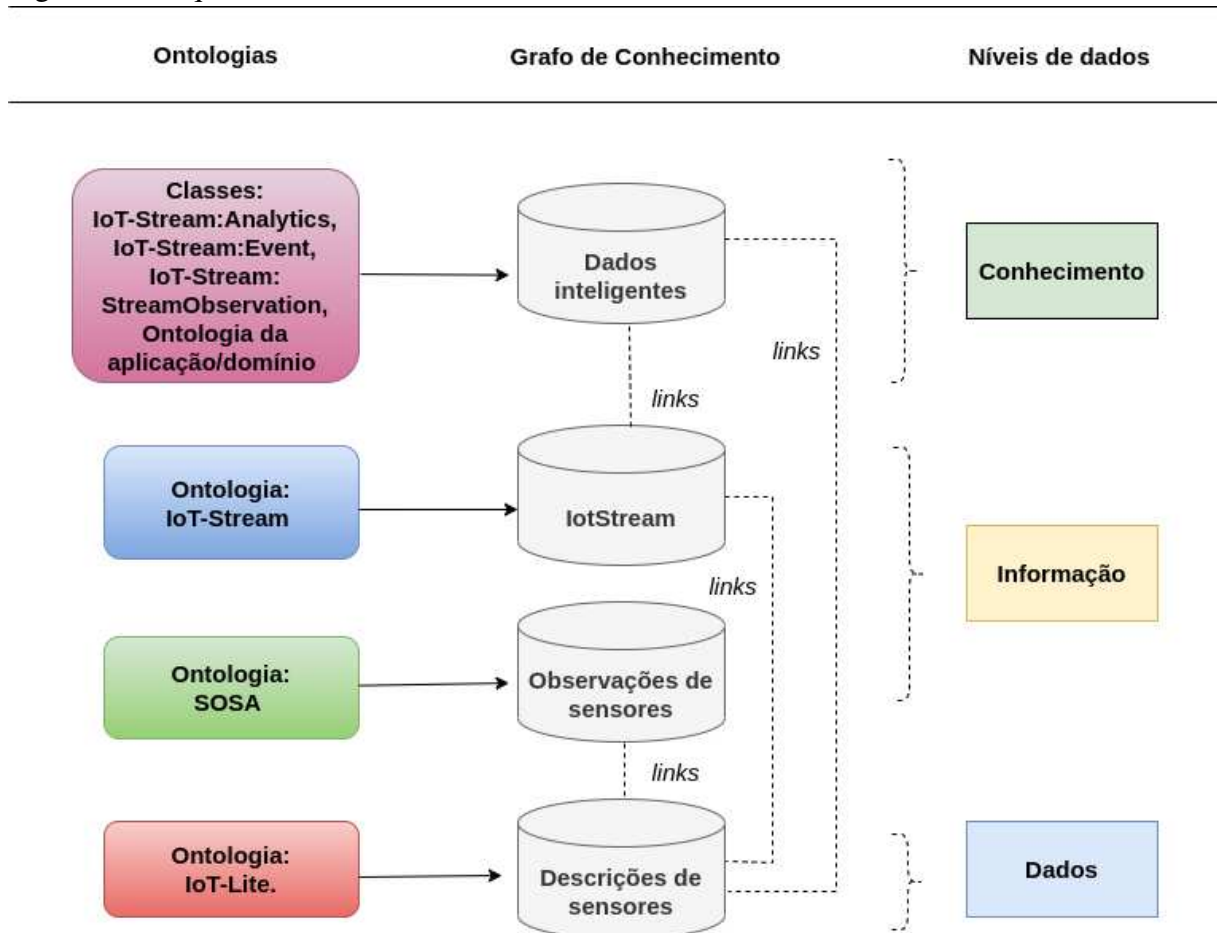
4.2 Abordagem *DIKW4IoT* para construção de grafos de conhecimentos de dados IoT

A abordagem *DIKW4IoT* tem como proposta a construção de um grafo de conhecimento de dados *IoT* baseado na hierarquia *DIKW* para representação e classificação dos

dados.

A Figura 4 ilustra as camadas de dados da abordagem *DIKW4IoT*. Essa abordagem busca classificar os dados coletados em diferentes níveis de abstração, desde dados brutos até o conhecimento. Esses níveis incluem: Descrição de Sensores, Observações de Sensores, Agregações de Dados de Sensores e Dados Inteligentes provenientes do Conhecimento Indutivo e Dedutivo.

Figura 4 – Arquitetura em 4 Camadas do Grafo de Conhecimento de *IoT*



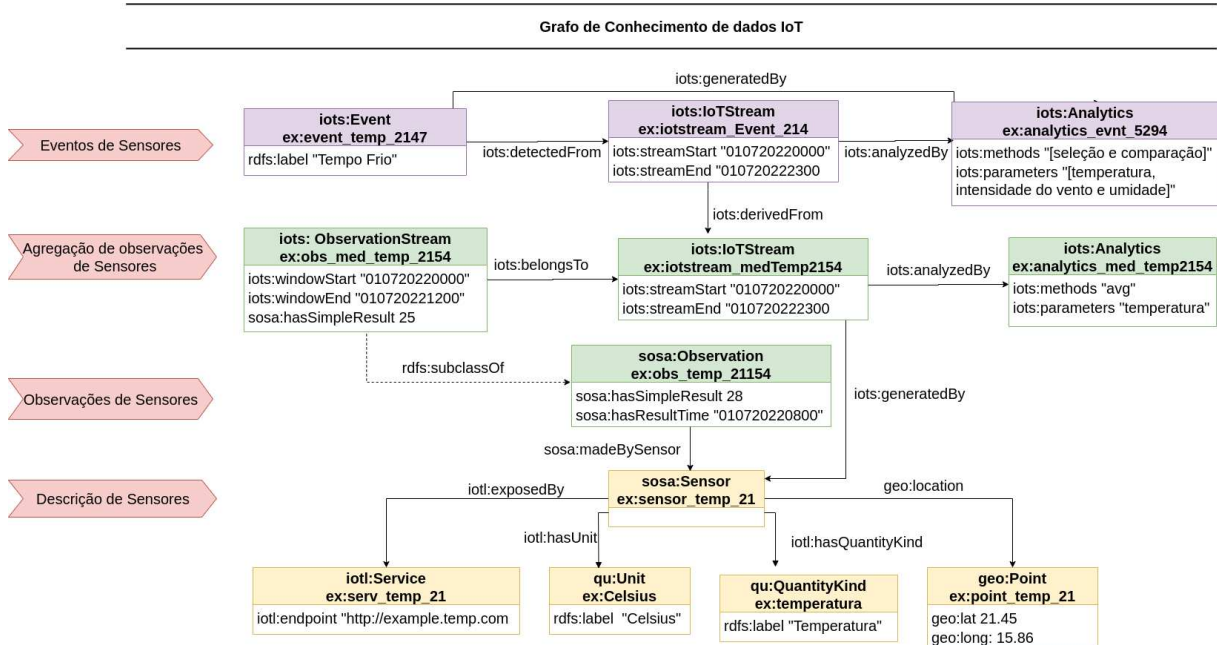
Fonte: elaborado pelo autor

A abordagem *DIKW4IoT* utiliza ontologias específicas para realizar a anotação semântica dos dados coletados pelos sensores em diferentes níveis. No primeiro nível, é utilizada a ontologia *IoT-lite* para anotar semanticamente os dados de sensores.

Já no segundo nível, a anotação semântica é realizada para as observações de sensores utilizando a ontologia *Sosa*. Além disso, nesse mesmo nível, também ocorre a anotação semântica de valores agregados, em que se utiliza a ontologia *Iot-Stream*. Nesse caso, as informações são abstraídas com base em uma janela temporal previamente especificada.

Por fim, no terceiro nível, é realizada a anotação semântica do conhecimento produzido por meio de regras de inferência ou aplicações de dados. Cada nível da abordagem é detalhado em seguida, utilizando como estudo de caso o fragmento do grafo de conhecimento apresentado na Figura 5.

Figura 5 – Ilustração de um Grafo de Conhecimento de dados *IoT* com base na abordagem *DIKW4IoT*



Fonte: elaborado pelo autor

4.2.1 Anotação Semântica de Sensores

A anotação semântica dos sensores é uma etapa importante na integração de dados em um grafo de conhecimento. Essa anotação consiste em identificar os sensores que serão integrados e atribuir-lhes informações semânticas, que descrevem as propriedades e características dos sensores. Geralmente, essa anotação é realizada apenas uma vez para cada sensor.

Para realizar essa anotação semântica é utilizada a ontologia *IoT-Lite*¹. A ontologia *IoT-Lite* foi desenvolvida para descrever de forma mais leve os sensores e dispositivos de Internet das Coisas (*IoT*). A ontologia *IoT-Lite* estende a ontologia *SSN* e fornece um conjunto de classes e relações que são distribuídos nas classes *Object*, *System* e *Lite-Service*.

A classe *Device* é uma subclasse da classe *System* e é usada para representar dispositivos *IoT*. Essa classe possui três subclasses: *ActuatingService* para dispositivos de

¹ <https://www.w3.org/Submission/2015/SUBM-iot-lite-20151126/>

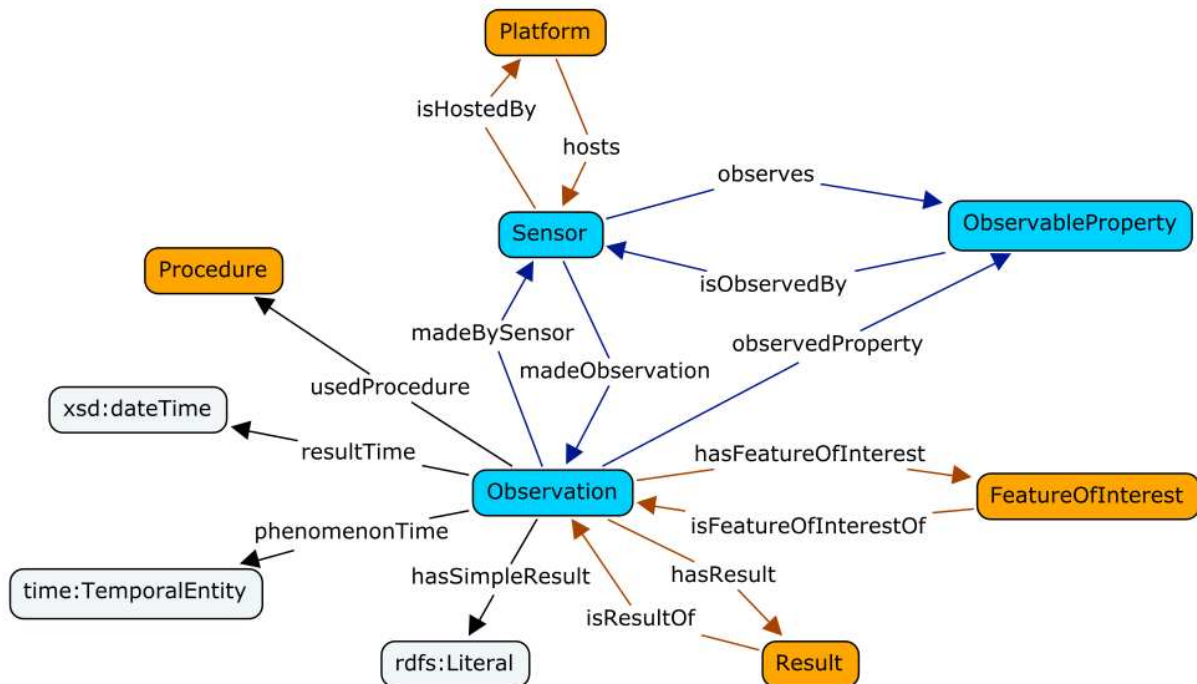
4.2.2 Anotação Semântica de Observações de Sensores

Os sensores cadastrados em um sistema de *IoT* produzem uma grande quantidade de dados que são chamados de observações. Essas observações correspondem aos dados que são utilizados no nível de anotação semântica de observações de sensores.

Para realizar essa anotação semântica, é utilizada a ontologia *Sosa*², que compreende o núcleo central da ontologia *SSN*, mas atua de forma independente. Através dessa ontologia, os dados de uma observação são transformados em uma tripla, que inclui um resultado numérico representado pela propriedade *hasSimpleResult* e uma anotação temporal representada pela propriedade *hasResultTime*.

A Figura 7 apresenta algumas das classes e relações da ontologia *Sosa* na perspectiva de uma observação.

Figura 7 – Ontologia SOSA - Perspectiva observação



Fonte: Adaptado de (JANOWICZ *et al.*, 2019)

A abordagem *DIKW4IoT* adota o uso de uma classe e algumas propriedades específicas para a anotação semântica das observações de sensores. Apenas a classe *Observation* e as propriedades *hasSimpleResult*, *resultTime* e *madeBySensor* são utilizadas, o que torna a anotação mais leve e simples.

² <https://www.w3.org/TR/vocab-ssn/>

É importante destacar que cada observação deve manter o link com o sensor de origem por meio da propriedade `madeBySensor`. Dessa forma, é possível recuperar as informações relacionadas ao sensor responsável pela observação.

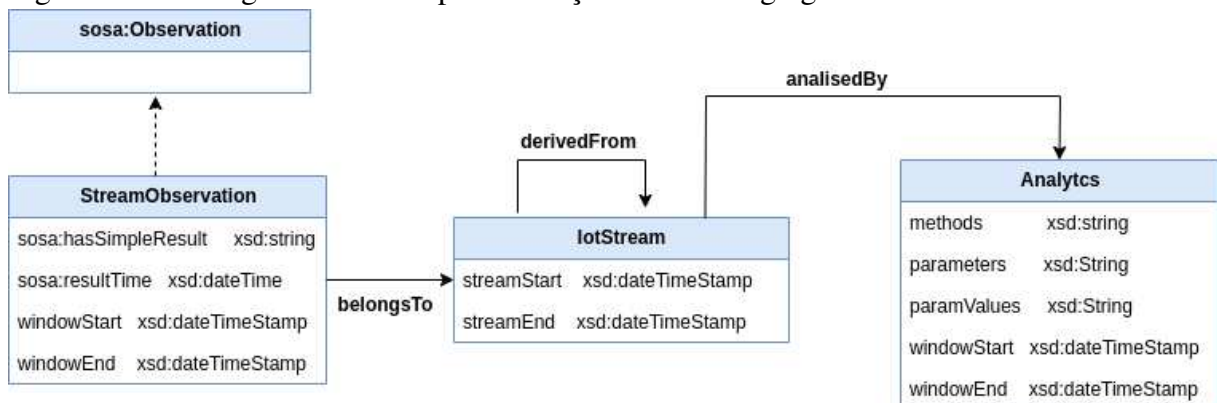
A Figura 5 apresenta um exemplo de como as observações de um sensor podem ser anotadas semanticamente por meio da ontologia *Sosa*. É possível perceber que, ao aplicarmos conceitos e propriedades da ontologia sobre os dados, eles se tornam informações significativas. Por meio das propriedades, é possível identificar sobre o que aquele dado se refere, tornando a interpretação mais clara e precisa.

4.2.3 Anotação Semântica de Agregações de Dados de Sensores

A anotação semântica de agregações de dados de sensores é feita por meio da anotação de resultados provenientes de métodos de agregação, como média, contagem, soma, máximo, mínimo, etc. Esses métodos fornecem informações sobre um conjunto de observações e requerem a definição de uma janela temporal sobre os dados para determinar quais observações serão incluídas na agregação.

A ontologia *IoT-Stream* é utilizada na anotação semântica de fluxos de dados processados e possui a classe `Analytics`, que identifica o tipo de processamento realizado sobre os dados. A propriedade `method` é usada para identificar o método utilizado sobre as observações, enquanto as propriedades `windowStart` e `windowEnd` identificam o intervalo temporal aplicado sobre os dados. A Figura 8 apresenta as principais classes e propriedades da ontologia *IoT-Stream*.

Figura 8 – Ontologia *IoT-Stream* para anotação de fluxos agregados



Fonte: Adaptado de (ELSALEH *et al.*, 2020)

Na abordagem *DIRK4IoT*, a classe `IoTStream` é usada para representar os dados

processados, como é o caso da agregação de dados. Armazenar os dados de forma sintetizada pode ser útil para muitas aplicações, especialmente quando a frequência de produção e disponibilização dos dados é alta.

A classe `StreamObservation` é uma subclasse da classe `Observation` da ontologia *Sosa* e é usada para armazenar o valor da observação do `IoTStream`. O valor para a classe `StreamObservation` pode ser um valor atômico ou um valor que resume um intervalo de dados.

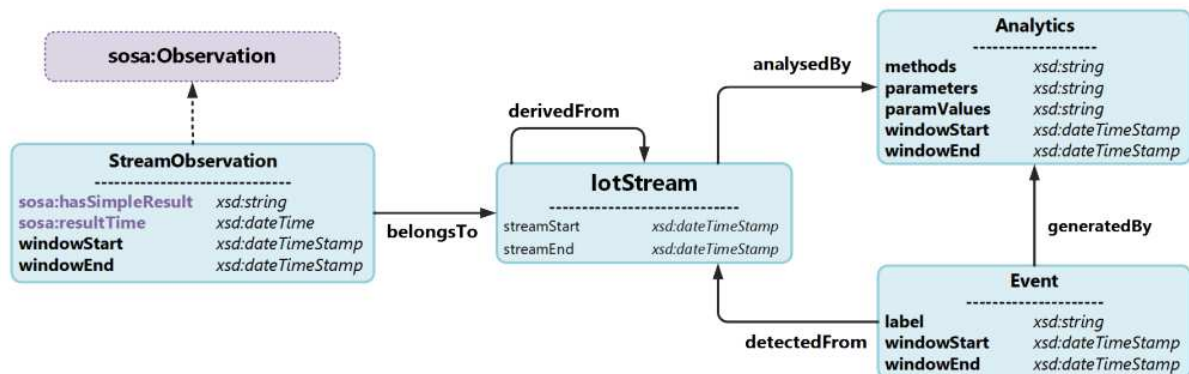
A Figura 5 apresenta um exemplo de anotação semântica de uma agregação de dados usando a ontologia *IoT-Stream*, e demonstra como as propriedades da ontologia se relacionam com outras ontologias utilizadas até o momento. O exemplo mostra a anotação semântica da média dos valores de observações em um intervalo de 12 horas, e como essa informação pode ser integrada ao grafo de conhecimento, fornecendo tanto informações atômicas como informações resumidas.

4.2.4 Anotação Semântica do Conhecimento Indutivo e Dedutivo

A anotação semântica de conhecimento indutivo é utilizada para anotar os resultados obtidos a partir de serviços de análise de dados e aplicações de detecção de eventos. Para esse fim, a ontologia *IoT-Stream* pode ser adotada, já que ela propõe uma descrição dos eventos e análises realizadas sobre os dados.

Na Figura 9, é apresentada a ontologia *IoT-Stream* com a classe relacionada à anotação de eventos. A classe `Event` refere-se ao evento detectado a partir de um fluxo de dados *IoT* e gerado por um processo analítico. A anotação semântica permite identificar o fluxo de dados no qual o evento foi gerado e qual processo analítico foi responsável por sua detecção.

Figura 9 – Ontologia *IoT-Stream* para anotação de eventos



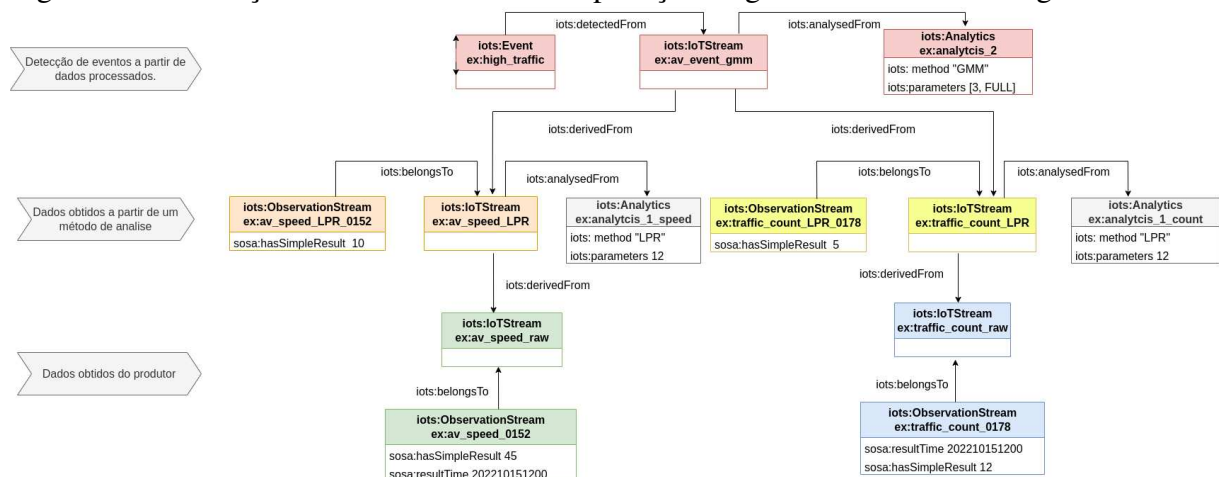
Fonte: Adaptado de (ELSALEH *et al.*, 2020)

Na Figura 5, o quarto nível apresenta informações sobre eventos detectados por uma aplicação. Por meio da ontologia *IoT-Stream*, é possível anotar o método utilizado e o evento identificado, bem como os fluxos de dados analisados para a detecção do evento. O evento “Tempo Frio” foi detectado e adicionado ao grafo, conforme demonstrado na Figura 5.

Para exemplificar ainda mais, a Figura 10 apresenta uma anotação semântica de fluxos de dados em uma aplicação de gerenciamento de tráfego de veículos. A detecção do tráfego “baixo” ou “alto” é baseada em dois fluxos de dados: a quantidade de veículos por hora e a média de velocidade dos veículos no local. Esses fluxos de dados passaram por dois processos de análise.

O primeiro processo de análise utilizou o método *Representação do Padrão Lagrangeano (LPR)* em um conjunto de dados para gerar um novo *IoTStream* com valores associados que são definidos como padrões de horários de dados (ELSALEH *et al.*, 2020).

Figura 10 – Anotação semântica referente a aplicação de gerenciamento de tráfego de veículos



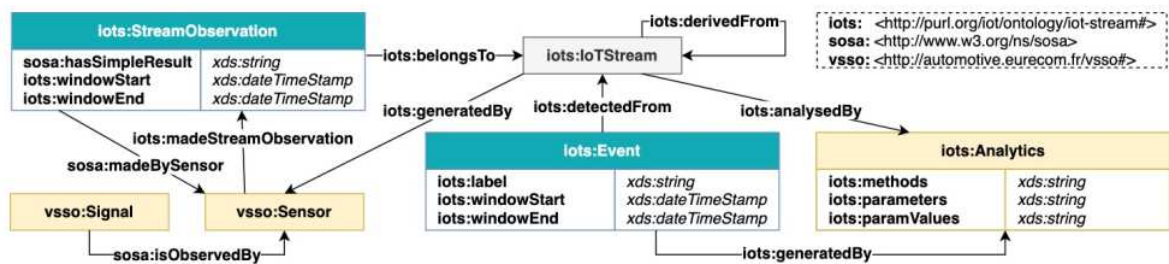
Fonte: Adaptado de (ELSALEH *et al.*, 2020)

O segundo serviço de análise de dados utiliza os dados processados e aplica o método *Gaussian Mixture Models (GMM)* para agrupar os padrões em *clusters* definidos, conforme descrito em (ELSALEH *et al.*, 2020). Em seguida, os eventos são avaliados para determinar se o tráfego de veículos está lento ou não. No exemplo, a análise identificou o evento *High Traffic*, indicando alto tráfego de veículos na região. Esse evento é então anotado semanticamente e adicionado ao grafo de conhecimento.

Os dados compartilhados no grafo de conhecimento podem ser utilizados por outros serviços de dados e/ou aplicações, permitindo a geração de conhecimento mais avançado. Além disso, eles podem ser utilizados para estender conceitos relacionados à ontologia da aplicação. A

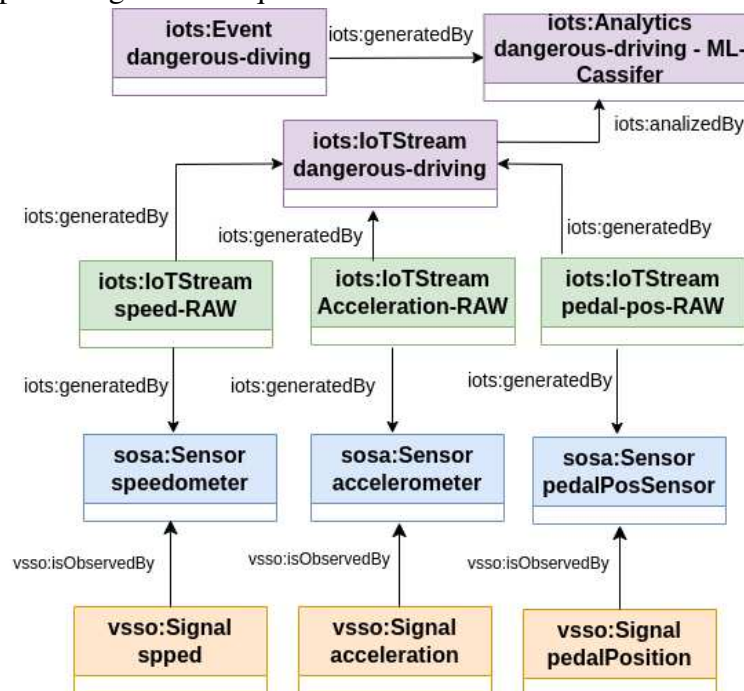
Figura 11 exemplifica o uso da ontologia *IoT-Stream* em conjunto com a extensão *VSSo* (*Vehicle Signal Specification Ontology*)³, que inclui conceitos relacionados à especificação de sinais de veículos.

Figura 11 – Ontologia para especificação de sinais de veículo



Fonte: Adaptado de (ALVAREZ-COELLO; GÓMEZ, 2021)

Figura 12 – Anotação semântica de fluxos de dados utilizados em um processo de análise de um classificador de aprendizagem de máquina



Fonte: Adaptado de (ALVAREZ-COELLO; GÓMEZ, 2021)

Um exemplo de aplicação que ilustra esse conceito é apresentado em (ALVAREZ-COELLO *et al.*, 2019), que utiliza modelos de aprendizado de máquina para classificar fluxos de dados contínuos e categóricos. Os fluxos contínuos, tais como aceleração, velocidade e posição do pedal do acelerador, são processados por meio de funções estatísticas, como média, mediana e desvio padrão. Já os fluxos categóricos, como eficiência da aceleração e estado do controle

³ <https://www.w3.org/TR/vssso/>

dinâmico de estabilidade do freio, são avaliados por meio da mediana. Os modelos utilizados são *Random Forest* e *Recurrent Neural Network* (ALVAREZ-COELLO *et al.*, 2019).

A figura 12 ilustra a anotação semântica da aplicação mencionada anteriormente. Nesta figura, é possível observar que os sinais de veículos, tais como *Speed-RAW*, *Acceleration-RAW* e *Pedal-Pos-RAW*, foram identificados e vinculados a um *IoTStream*. Os dados de interesse derivados desses fluxos de dados geram um novo *IoTStream* chamado *IoTStream Dangerous-driving*, o qual é utilizado pelos classificadores para a classificação das manobras realizadas por um motorista, dentre outras informações.

Esses exemplos ilustram como o conhecimento indutivo gerado pode ser compartilhado no grafo de conhecimento. A anotação semântica permite não apenas a anotação dos dados obtidos, mas também de todo o processo de análise aplicado e seus resultados. À medida que os dados são processados, diferentes camadas de informações são obtidas. Adicionalmente, é possível enriquecer o contexto incluindo classes e propriedades relacionadas ao domínio da aplicação.

No contexto do conhecimento dedutivo, a anotação semântica corresponde as triplas que são geradas a partir do processo de inferência realizado sobre os dados. De acordo com a ontologia de domínio são definidas classes e axiomas que estabelecem regras pertinentes ao domínio da aplicação e limitações para sua interpretação, ou seja, os axiomas consistem em regras de lógica de primeira ordem que são especificadas a partir de termos de uma ontologia de domínio e que definem uma verdade sobre o conjunto de dados.

Por exemplo, suponha que em um prédio existem dois tipos de sensores: um sensor para detectar temperatura e um sensor para detectar fumaça. Foi estabelecido previamente que na presença de fumaça no ambiente e temperatura igual ou maior que 37°C indicaria a ocorrência de um “Incêndio”. Na ontologia de domínio pode-se definir o seguinte axioma: $Fire \equiv (Temperature \sqcup (hasTemperature \geq 37)) \sqcup (Smoke \sqcup (hasSmokePresence true))$.

Em um ambiente monitorado, a medida que o processo de inferência identificar nos fluxos de dados temperatura com valores maiores que 37°C e presença de fumaça como *true*, um novo dado será inferido, apontando a ocorrência de um incêndio.

Sendo assim é possível por meio de axiomas estabelecer padrões sobre os dados. O processo de inferência consiste em inferir novas afirmações sobre os dados quando estes padrões são identificados. Estas novas afirmações obtidas são definidas como conhecimento dedutivo.

Seguindo o exemplo apresentado na Seção 4.1, um exemplo de axioma nesse contexto

poderia ser estabelecido da seguinte forma: $Cold \equiv (Temperature \sqcup (hasTemperature \leq 18)) \sqcup (Wind \sqcup (hasWindSpeed \geq 35))$, isto é, quando a temperatura for menor ou igual a 18°C e a intensidade do vento for maior ou igual a 35 km/h o tempo será considerado frio.

A definição das regras podem envolver mais de um fluxo de dados, caso julgue necessário, e leva em consideração os conceitos abordados na ontologia de domínio da aplicação.

4.3 Estudo de Caso: Construindo um grafo de conhecimento a partir de parâmetros ambientais de Lisboa utilizando a abordagem *DIKW4IoT*

Neste seção, apresenta-se um estudo de caso sobre a construção de um grafo de conhecimento de dados *IoT* por meio da abordagem *DIKW4IoT*. Para isso, utilizou-se parâmetros ambientais de Lisboa - Portugal.

4.3.1 Descrição do conjunto de dados

Lisboa Aberta⁴ é um portal de dados abertos da cidade de Lisboa que disponibiliza diferentes conjuntos de dados sobre ambiente, educação, mobilidade, turismo, etc.

O conjunto de dados utilizado na pesquisa refere-se ao monitoramento de parâmetros ambientais da cidade de Lisboa. Os dados obtidos pertencem a uma rede que engloba 82 (oitenta e duas) estações, compostas por 754 (setecentos e cinquenta e quatro) sensores. As medições abrangem dados relativos à qualidade do ar, ruído, clima e tráfego. Os parâmetros registrados apresentam informações sobre:

ESTADO DO TEMPO:

1. Pressão atmosférica – média horária, mbar
2. Umidade relativa – média horária,
3. Temperatura – média horária, °C
4. Vento intensidade – média horária, km/h
5. Vento direção – média horária, °
6. Precipitação - média horária, mm
7. Radiação global – média mensal, W/m²
8. Radiação Ultravioleta – índice UV, unidade

QUALIDADE DO AR:

⁴ <https://lisboaaberta.cm-lisboa.pt/index.php/pt/>

1. Dióxido de Azoto (NO₂) - média horária, µg/m³
2. Ozono (O₃) - média horária, µg/m³
3. Partículas (PM₂₅) – média horária, µg/m³
4. Partículas (PM₁₀) – média horária, µg/m³
5. Dióxido de enxofre (SO₂) – média horária, µg/m³
6. Benzeno (C₂H₆) – média horária µg/m³
7. Monóxido de Carbono (CO) – média horária, mg/m³
8. Nobélio (NO) – média horária, mg/m³

RUÍDO:

1. 1. Nível sonoro contínuo equivalente (LAEQ) – média horária, dB(A)

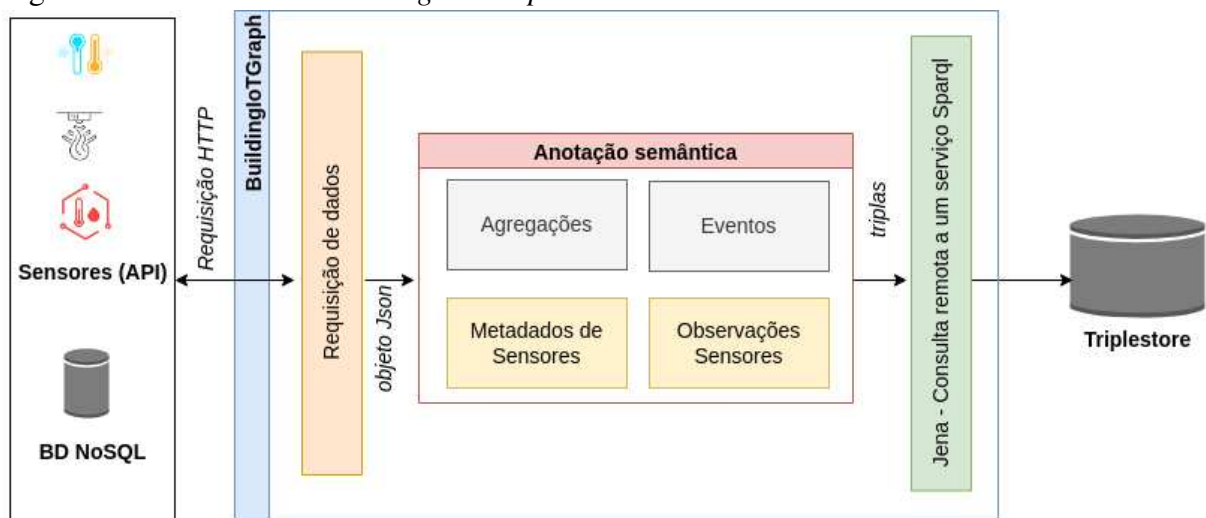
CONTADORES DE TRÁFEGO:

1. Volume Tráfego Horário (VTH) - número de veículos por sentido, por hora.

Esses dados são exibidos em tempo real no painel de monitoramento⁵. Eles podem ser obtidos por tipo de sensor ou por localização (estação). Os dados são fornecidos no formato *JSON* e estão sob a licença de uso "*Creative Commons CCZero*", tornando-os de domínio público. As informações publicadas representam os valores médios de cada parâmetro ao longo de uma hora.

4.3.2 Construindo o grafo de conhecimento de dados IoT

Figura 13 – Módulos do *BuildingIoTGraph*



Fonte: elaborado pelo autor

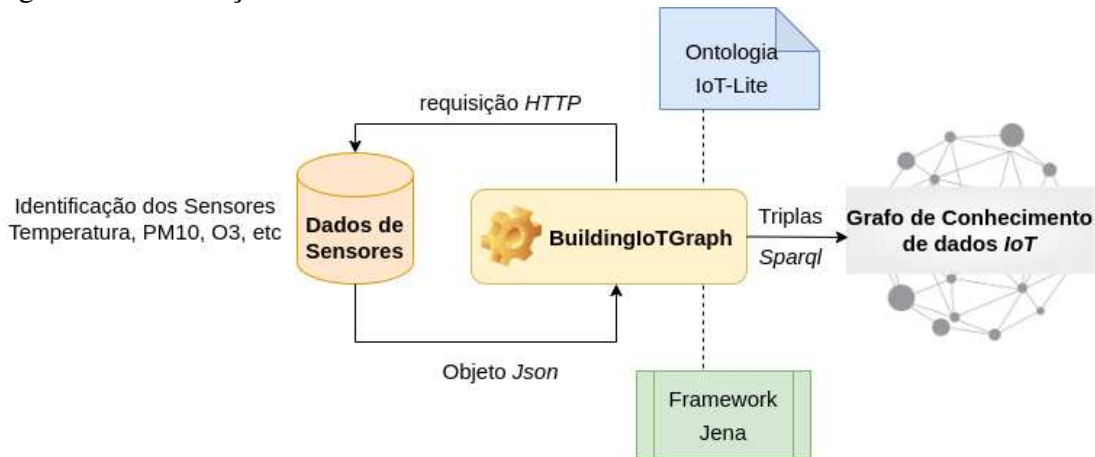
⁵ <https://dados.cm-lisboa.pt/dataset/monitorizacao-de-parametros-ambientais-da-cidade-de-lisboa>

Para a construção do grafo de conhecimento, desenvolveu-se um código em linguagem *Java* com o objetivo de efetuar as anotações semânticas nos dados. Esse código foi nomeado como *BuildingIoTGraph*⁶. Ele compreende métodos que permitem a aquisição dos dados, a realização de diferentes tipos de anotações semânticas (incluindo informações de sensores, observações, resultados de agregações e eventos) e o envio de triplas para um repositório de triplas. A estrutura modular que constitui o *BuildingIoTGraph* é apresentada na Figura 13.

Dentro do *BuildingIoTGraph*, encontra-se uma classe denominada *OntologyInstances*, a qual disponibiliza métodos para criar instâncias baseadas no mapeamento dos dados definido.

Para realizar a anotação semântica dos dados dos sensores, utilizou-se a ontologia *IoT-Lite*. Além disso, outros conceitos e relações foram incorporados a fim de abranger todo o domínio. No início do processo, foram solicitados os dados que descrevem os sensores. Conforme pode ser observado na Figura 14, esses dados foram adquiridos através de uma requisição *HTTP*. Uma vez obtidos, os dados foram convertidos em triplas conforme o mapeamento predefinido e, posteriormente, encaminhados para o repositório de triplas *GraphDB*⁷.

Figura 14 – Anotação semântica dos dados de sensores



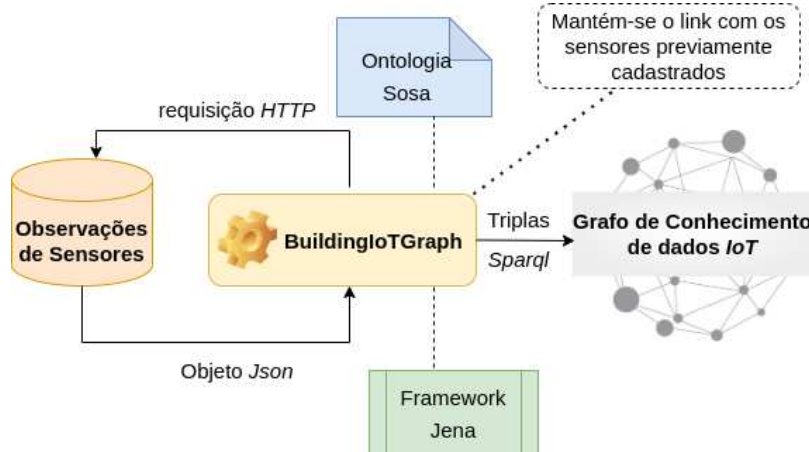
Fonte: elaborado pelo autor

Após o cadastro dos dados dos sensores no grafo de conhecimento, procedeu-se à obtenção das observações. Para a anotação semântica das observações, utilizou-se a classe *Observation* e as propriedades *hasSimpleResult*, *hasResultTime* e *madeBySensor* da ontologia *SOSA*. A propriedade *madeBySensor* estabelece a ligação entre a observação e o sensor correspondente. A Figura 15 ilustra a maneira pela qual as observações foram

⁶ <https://github.com/searchmdcc/BuildingIoTGraph>

⁷ <https://graphdb.ontotext.com/>

Figura 15 – Anotação semântica das observações de sensores



Fonte: elaborado pelo autor

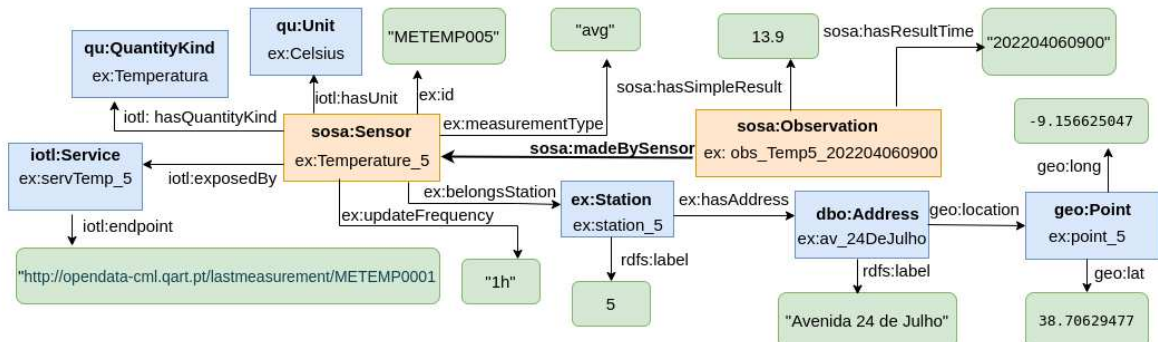
incorporadas ao grafo de conhecimento.

Por meio de uma requisição *HTTP*, os dados são obtidos do *endpoint*. Eles chegam no formato *JSON* e, em seguida, são transformados conforme o modelo de dados predefinido. Esses dados são enviados ao módulo de anotação semântica, onde conceitos semânticos são aplicados e transformados em triplas. Finalmente, essas triplas são enviadas ao repositório de dados.

A Figura 16 traz um exemplo de anotação semântica de um sensor de temperatura e uma observação. Nessa figura, é possível identificar informações como a geolocalização do sensor, o tipo de medida e unidade, bem como o serviço que disponibiliza os dados.

Figura 16 – Anotação semântica de dados e observações de sensores

Prefixos:
 iotl: <http://purl.oclc.org/NET/UNIS/firmware/iot-lite#>
 qu: <http://purl.org/NET/snx/qu/qu#>
 sosa: <http://www.w3.org/ns/sosa/>
 ex: <http://www.example.com/>
 rdfs: <http://www.w3.org/2000/01/rdf-schema#>
 geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>

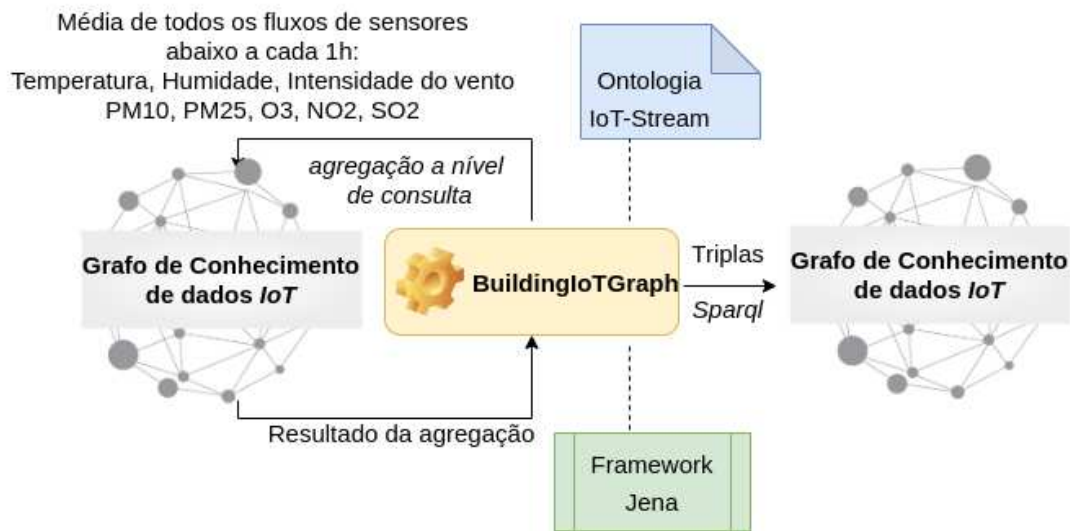


Fonte: elaborado pelo autor

No contexto das agregações de dados, foram criadas agregações considerando a temática comum dos dados. A definição da operação de agregação de dados envolveu um

intervalo temporal de 1 hora. Por exemplo, dentre os 754 (setecentos e cinquenta e quatro) sensores integrados, 80 (oitenta) deles correspondem a medições de temperatura. Para calcular a média geral da temperatura na cidade de Lisboa, é necessário determinar a média a partir das medições dos 80 sensores de temperatura. A Figura 17 proporciona uma visão geral de como esse processo foi realizado.

Figura 17 – Anotação semântica de resultados de agregação de dados de sensores



Fonte: elaborado pelo autor

A agregação foi conduzida em nível de consulta. Assim, uma consulta *SPARQL* foi empregada para realizar as agregações de dados no grafo de conhecimento. O resultado dessa operação, obtido através da consulta, foi direcionado ao método responsável pela anotação semântica especificada e, posteriormente, pela inclusão no grafo de conhecimento.

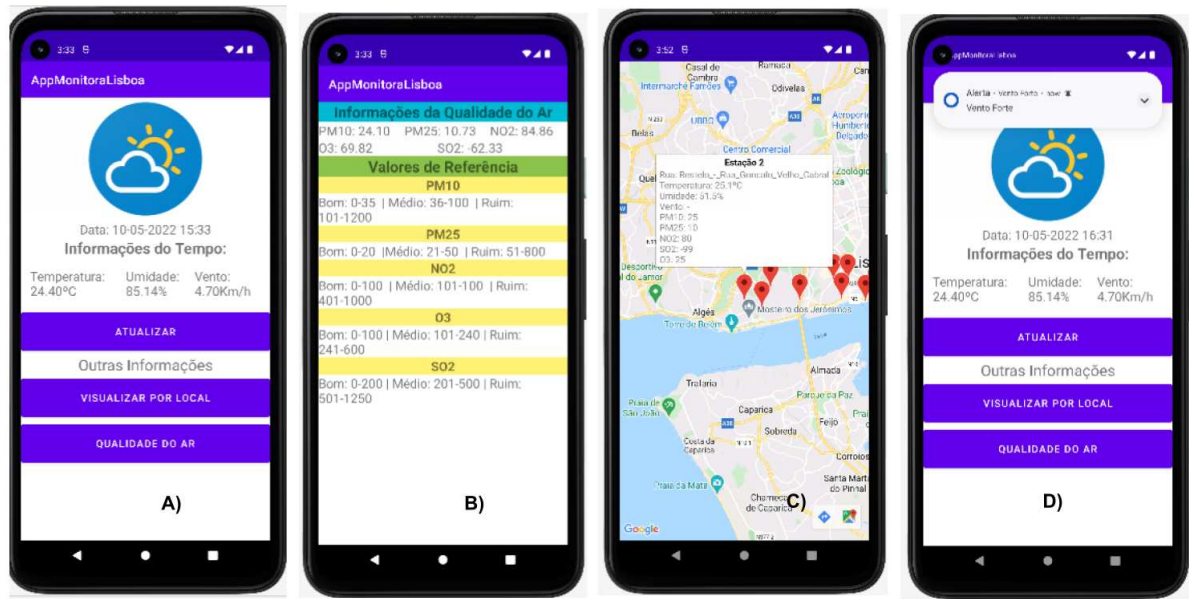
Esse procedimento foi aplicado nos seguintes fluxos de dados: temperatura, umidade do vento, intensidade do vento, PM10, PM25, NO2, O3 e SO2. Isso possibilita o acesso direto aos parâmetros gerais que monitoram as condições climáticas em Lisboa.

Até este ponto, temos um grafo de conhecimento composto por descrições de sensores, observações e agregações de dados. A Seção 4.3.3 detalhará a utilização desse grafo e o compartilhamento dos dados que constituem a camada de conhecimento.

4.3.3 *AppMonitoraLisboa: Uma aplicação mobile para consulta de informações relacionadas ao estado do tempo e qualidade do ar na cidade de Lisboa.*

Para consumir o grafo de conhecimento de dados *IoT* e gerar informações em nível de conhecimento, foi desenvolvida uma aplicação *mobile* chamada *AppMonitoraLisboa*.

Figura 18 – AppMonitoraLisboa



Fonte: elaborado pelo autor

O objetivo da aplicação *AppMonitoraLisboa*⁸ é disponibilizar informações relacionadas ao estado do tempo e à qualidade do ar na cidade de Lisboa. A aplicação monitora eventos simples e, quando detectados, os disponibiliza para compartilhamento no grafo de conhecimento.

A construção da aplicação *AppMonitoraLisboa* foi realizada utilizando a plataforma *Android*⁹. Ela possui funções básicas, como fornecer informações sobre temperatura, umidade e intensidade do vento em Lisboa, além de apresentar parâmetros de qualidade do ar, como PM10, PM25, O3, NO2 e SO2.

Para acessar o grafo de conhecimento, foi utilizado o *framework Jena* para *Android*¹⁰. O *framework Jena* oferece mecanismos para consultar um grafo de conhecimento através de consultas utilizando a linguagem *SPARQL*¹¹.

A Figura 18 apresenta a interface da aplicação *AppMonitoraLisboa*. A parte 18.A exibe informações relacionadas à temperatura, umidade e intensidade do vento em Lisboa. Para isso, são realizadas consultas *SPARQL* no grafo de conhecimento, buscando essas informações nos fluxos de dados agregados. Assim, os valores apresentados oferecem uma visão geral desses parâmetros em Lisboa.

Na parte 18.B apresenta os valores de PM10, PM25, SO2, NO2 e O3, que são índices utilizados para avaliar a qualidade do ar. A aplicação disponibiliza alguns valores de referência

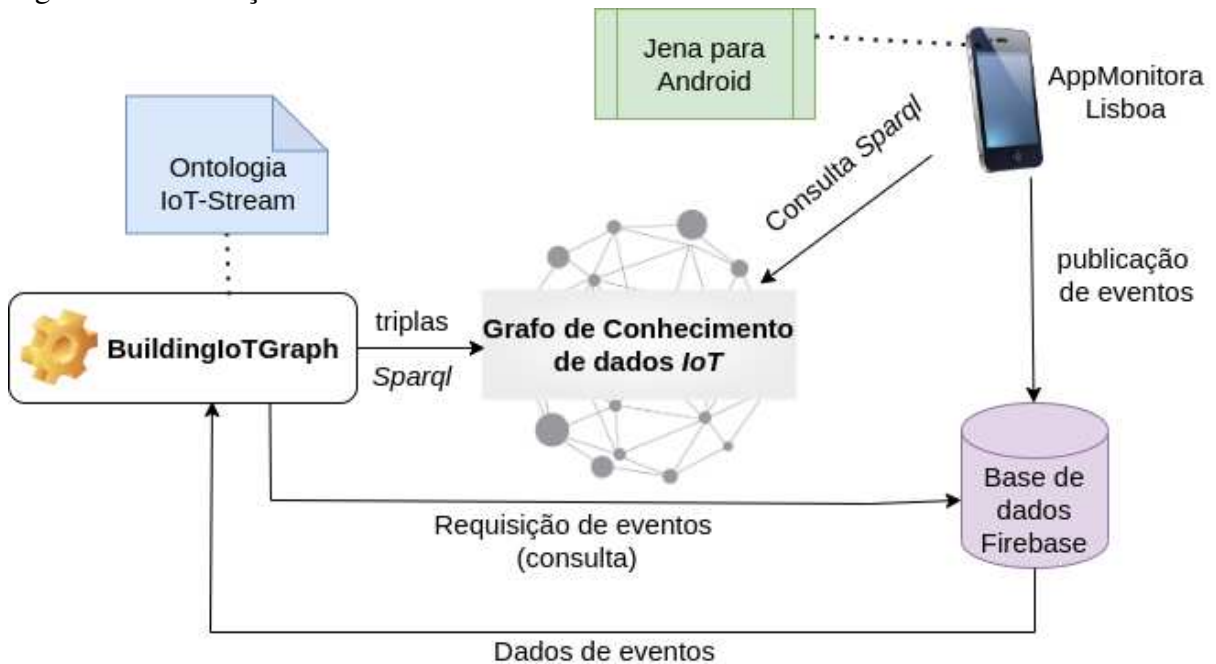
⁸ <https://github.com/searchmdcc/appMonitoraLisboa/tree/master>

⁹ <https://developer.android.com/docs?hl=pt-br>

¹⁰ <https://github.com/maddymz/Topic-Recommendation-System/tree/master/app/app/libs>

¹¹ <https://www.w3.org/TR/sparql11-query/>

Figura 19 – Anotação semântica de eventos



Fonte: elaborado pelo autor

que auxiliam os usuários a identificar e classificar a qualidade do ar no momento.

Na parte 18.C, os usuários podem visualizar as estações de sensores no mapa. Ao clicar em cada marcação, são exibidos os resultados dos sensores presentes naquela estação. Isso permite que os usuários obtenham informações mais específicas sobre uma determinada área de Lisboa.

A aplicação também inclui uma funcionalidade de monitoramento de eventos. Alguns eventos simples foram configurados, como "Vento Forte" para intensidades de vento acima de 70 km/h, "Baixa Umidade" para valores de umidade abaixo de 30%, entre outros. Quando um desses eventos é detectado, uma notificação é enviada ao usuário, como pode ser observado na parte 18.D. Este evento é enviado para uma base de dados no *Firestore*¹².

Para estabelecer o nível de dados relacionados ao conhecimento no grafo de conhecimento de dados *IoT*, os dados associados aos eventos foram adquiridos e integrados ao grafo. A Figura 19 apresenta de forma ilustrativa o processo pelo qual esses dados são adquiridos da base de dados *Firestore*, submetidos a um processo de transformação de acordo com o mapeamento pré-definido e, em seguida, disponibilizados no próprio grafo de conhecimento.

Nesse fluxo, os dados são inicialmente obtidos da base de dados *Firestore*. Essa etapa envolve a requisição dos dados a partir da fonte original, que pode ser um conjunto variado de eventos registrados em diferentes momentos e contextos.

¹² <https://firebase.google.com/>

Uma vez adquiridos, os dados passam por um processo de transformação. Essa transformação é orientada pelo mapeamento que foi previamente definido, o qual estabelece a correspondência entre os atributos e conceitos dos dados originais e as estruturas semânticas utilizadas no grafo de conhecimento. Essa etapa é fundamental para garantir a integração coerente dos dados ao contexto do grafo.

Após a transformação, os dados resultantes são incorporados ao grafo de conhecimento. Nesse ponto, os dados adquiridos e processados são publicados no grafo, enriquecendo-o com informações relevantes sobre eventos e atividades. Isso, por sua vez, amplia o valor do grafo como um todo, permitindo a realização de consultas complexas, análises e inferências que transcendem os dados brutos iniciais.

A Figura 19 oferece uma visão concisa desse processo, destacando a sequência de etapas que levam desde a aquisição dos dados até a sua integração no grafo de conhecimento. Esse processo é crucial para garantir a precisão, a consistência e a utilidade dos dados no contexto do conhecimento estabelecido pelo grafo.

5 **SIGIoT-S: UM FRAMEWORK CONCEITUAL PARA INTEGRAÇÃO DE DADOS DE Sensores COM Fontes DE DADOS Heterogêneas E Externas**

A integração de dados de *IoT* com outras fontes de informações podem trazer diversas vantagens para aplicações em diferentes áreas, como saúde, agricultura, indústria, transporte, entre outras. Por exemplo, na área de saúde, a integração de dados de sensores *wearable* com dados de saúde eletrônica pode permitir o monitoramento contínuo da saúde de pacientes, facilitando o diagnóstico precoce de doenças e o tratamento personalizado. Na agricultura, a integração de dados de sensores meteorológicos com dados de solos e plantas pode permitir o monitoramento do crescimento das plantas e do clima, otimizando o uso de recursos, como água e fertilizantes

O processo de enriquecimento dos dados de sensores por meio da integração com fontes de dados externas permite a agregação de informações adicionais e estabelece relações significativas entre os dados coletados. Isso possibilita uma interpretação mais precisa e sofisticada dos dados de sensores, permitindo análises avançadas, detecção de padrões, tomadas de decisão mais informadas e a obtenção de *insights* valiosos. Esse enriquecimento dos dados amplia o potencial de aplicação dos dados *IoT*, promovendo uma visão mais abrangente e inteligente do ambiente em que os dispositivos estão inseridos.

Neste Capítulo é apresentado o *SIGIoT-S*: um *framework* conceitual para integração de dados de sensores com fontes de dados heterogêneas e externas. Na Seção ?? é apresentado as etapas que compõe o *SIGIoT-S*. A Seção 5.4 apresenta um estudo de caso relacionado a integração de um grafo de conhecimento de dados *IoT* com fontes de dados externas.

5.1 Uma visão geral do *framework SIGIoT-S*

O *SIGIoT-S* tem como proposta facilitar o processo de integração de dados provenientes de sensores com fontes externas e heterogêneas, orientando esse processo em duas etapas distintas. O *framework SIGIoT-S* é ilustrado na Figura 20.

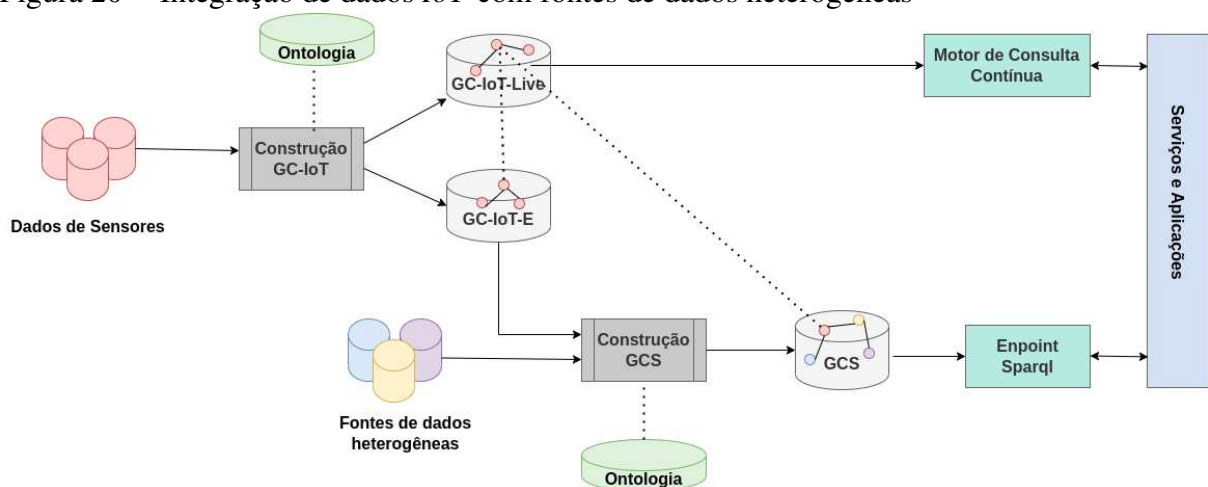
A primeira etapa envolve a construção do Grafo de Conhecimento de Dados *IoT* (*GC-IoT*) utilizando o método baseado na abordagem *DIKW4IoT*, o qual tem por finalidade a integração e enriquecimento dos dados *IoT*, conforme previamente discutido no Capítulo 4. A segunda etapa, por sua vez, concentra-se na construção do Grafo de Conhecimento Semântico (*GCS*), que integra os dados estáveis provenientes do *GC-IoT* com fontes de dados heterogêneas.

Esses dados podem ser acessados por meio de aplicações utilizando motores de

consulta contínua e *SPARQL*. O motor de consulta contínua realiza consultas na porção do grafo que contém dados dinâmicos, ou seja, dados coletados em tempo real ou próximos a ele. O motor de consulta *SPARQL*, por sua vez, efetua consultas no GCS, que abriga informações de domínio provenientes das diferentes fontes de dados integradas. Além disso, o GCS mantém informações relacionadas ao *GC-IoT* por meio dos dados de sensores integrados. Isso permite a correlação dos dados dos sensores com observações, tanto históricas quanto as mais recentes.

Nas próximas seções, detalha-se mais minuciosamente as etapas que compõem o *SIGIoT-S*.

Figura 20 – Integração de dados *IoT* com fontes de dados heterogêneas



Fonte: elaborado pelo autor

5.2 Primeira etapa: Construção do Grafo de conhecimento de dados *IoT*

A primeira etapa consiste na construção do Grafo de Conhecimento de dados *IoT* - *GC-IoT*. Essa construção pode ser realizada aplicando-se a abordagem *DIKW4IoT*, apresentada no Capítulo 4.

O *GC-IoT* é composto por dois sub-grafos: o grafo de conhecimento *IoT* de dados estáveis (*GC-IoT-E*) e o grafo de conhecimento de dados dinâmicos (*GC-IoT-Live*). O grafo de conhecimento de dados dinâmicos (*GC-IoT-Live*) representa uma visão dos dados *GC-IoT* em tempo real (ou próxima ao tempo real). O *GC-IoT-Live* captura e modela os dados em constante atualização, permitindo uma visão em tempo real do ecossistema *IoT* e facilitando análises e tomadas de decisão em tempo real. Para acessar os dados utiliza-se um motor de consulta contínua.

A consulta contínua estabelece uma janela de tempo para o consumo dos dados no

GC-IoT. Essa janela pode ser física, indicando um número exato de triplas a serem consumidas, ou lógica, abrangendo triplas que ocorrem em determinado intervalo de tempo. A escolha entre essas opções depende do motor de consulta utilizado. Motores de consulta como *C-Sparql*¹ ou *CQELS*² podem ser utilizados para esse propósito.

O grafo de conhecimento de dados estáveis (*GC-IoT-E*) é uma visão dos dados *GC-IoT* contendo informações estáticas ou imutáveis que descrevem os sensores *GC-IoT*. Essas informações estão relacionadas aos metadados e propriedades do sensor como, ID do sensor, tipo, fabricante, modelo, tipo de medida e unidade, e informações relacionadas à rede dos sensores como, endereços IP, protocolos de roteamento e topologia de rede, entre outros.

As observações de sensores com o tempo podem ser definidas como observações históricas de sensor. Estas observações se referem às observações que foram registradas anteriormente, por não possuírem mudanças no futuro, pode-se classificar esse tipo de observação como estável. Nesse contexto, estável significa que os dados observados permanecem fixos e não serão atualizados com o passar do tempo.

5.3 Segunda etapa: Integração com fontes heterogêneas e externas

A segunda etapa consiste na integração do *GC-IoT* com fontes de dados heterogêneas e externas. Para essa integração utiliza-se a parte do Grafo de Conhecimento de dados *IoT* estáveis, o *GC-IoT-E*. A integração pode ser efetivada durante o processo de construção do Grafo de Conhecimento Semântico - *GCS*. O trabalho de (ROLIM *et al.*, 2021) estabelece um processo para a construção de um *GCS*.

Esse processo é realizado de forma incremental e possui cinco passos distintos, sendo eles: aquisição do conhecimento, modelagem da ontologia de domínio, construção dos *GCs* locais, criação de visões de Ligação e inferência e limpeza. A Figura 21 apresenta as etapas para construção do *GCS*. Nas subseções a seguir será explanado o passo a passo para a construção de um *GCS* a partir de fontes de dados heterogêneas e a integração com o *GC-IoT*.

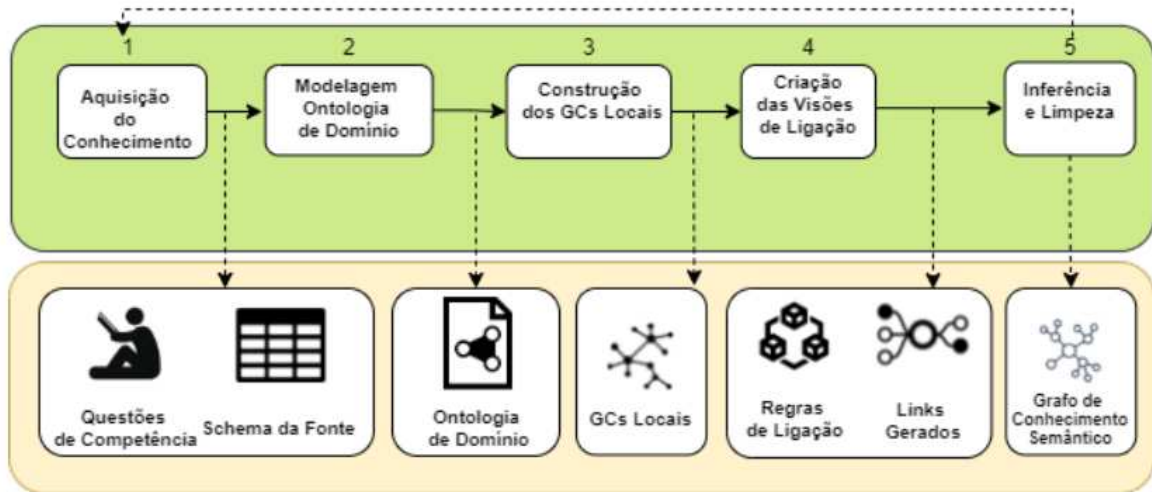
5.3.1 Aquisição do Conhecimento

O primeiro passo da construção de um *GCS* é a aquisição do conhecimento. É essencial estabelecer o objetivo da coleta e integração dos dados, bem como as questões que

¹ <https://github.com/streamreasoning/CSPARQL-engine>

² <https://github.com/cqels>

Figura 21 – Construção de um Grafo de Conhecimento Semântico



Fonte: adaptado de (ROLIM *et al.*, 2021)

o usuário deseja responder. Nesse processo, é necessário identificar as fontes de dados mais relevantes em termos de domínio, escopo, proveniência e manutenção.

A seleção das fontes de dados apropriadas envolve a análise cuidadosa das necessidades do usuário e dos requisitos do sistema. Isso inclui considerar a confiabilidade dos dados, a qualidade da informação, a consistência e a atualização dos dados, bem como a capacidade de integrar diferentes fontes para obter uma visão abrangente.

Além disso, é importante levar em conta o domínio de conhecimento necessário para atingir os objetivos estabelecidos. Isso implica identificar especialistas, fontes de informações confiáveis e ontologias relevantes que possam ser utilizadas para estruturar e enriquecer os dados coletados.

Com base nessa seleção criteriosa das fontes de dados, é possível iniciar o processo de aquisição, que envolve a extração, transformação e carga dos dados em um formato adequado para a construção do grafo de conhecimento semântico. Esse processo pode incluir a normalização dos dados, a resolução de inconsistências e a padronização das representações, para garantir a integração efetiva e coerente das informações. No caso do *GC-IoT*, os dados já possuem um vocabulário comum e adotam o formato *RDF*.

5.3.2 Modelagem da ontologia de domínio

O segundo passo consiste na modelagem de uma ontologia de domínio, que desempenha um papel fundamental na construção de uma camada semântica para o *GCS*. A ontologia de domínio é uma representação formal e estruturada do conhecimento em um domínio específico.

Ela descreve os conceitos, as entidades, as propriedades e as relações relevantes dentro desse domínio, capturando também as restrições e as regras que governam esses elementos.

Para criar uma ontologia de domínio sólida, é necessário realizar uma análise detalhada dos esquemas das fontes de dados selecionadas anteriormente. Essa análise permite identificar as estruturas de dados existentes, compreender as relações entre as entidades e as propriedades, e mapear essas informações para um modelo comum.

Durante a modelagem da ontologia, é importante lidar com a heterogeneidade dos dados provenientes de diferentes fontes. Isso envolve reconciliar diferenças nos esquemas, na nomenclatura, nos formatos e nas representações dos dados, para garantir a integração coerente e consistente no modelo comum. É necessário definir termos e conceitos de forma clara e precisa, estabelecer as relações semânticas adequadas entre os elementos e considerar a extensibilidade da ontologia para futuras evoluções e incorporação de novos conhecimentos.

Nesse passo, é necessário realizar também a análise do vocabulário utilizado no *GC-IoT*. É importante compreender suas estruturas, conceitos e relações para identificar as correspondências possíveis entre os dados do *GC-IoT-E* e as demais fontes de dados. Essas correspondências deverão integrar a ontologia de domínio de *GCS*. Esse processo também pode ser definido como alinhamento de ontologias, já que tem como propósito encontrar correspondências entre os conceitos da ontologia utilizada no *GC-IoT* e na ontologia utilizadas nas fontes de dados heterogêneas. Permitindo assim a interoperabilidade e integração entre os grafos.

5.3.3 Construção dos Grafos de conhecimentos Locais

No terceiro passo, realiza-se a construção de um Grafo de Conhecimento Local (*GCL*) para cada fonte de dados. O *GCL* é uma representação de conhecimento que se limita a um contexto ou domínio específico, focando em capturar as relações e informações relevantes dentro desse escopo. O *GC-IoT* atua como um *GCL* em que o seu domínio é relacionado ao contexto *IoT* em que foi adquirido.

Nessa passo, utiliza-se uma abordagem *top-down*, que consiste em três tarefas principais. A primeira é o mapeamento de cada fonte de dados para a ontologia de domínio, estabelecendo a correspondência entre os elementos da fonte de dados e os conceitos da ontologia. Em seguida, ocorre a especificação da ontologia do *GCL* com base nos mapeamentos gerados, definindo as relações e as propriedades específicas do domínio.

A terceira tarefa é a implementação do *GCL*, que pode ser feita de duas maneiras:

abordagem materializada ou abordagem virtual. Na abordagem materializada, o conhecimento é pré-processado e armazenado explicitamente no grafo, sendo extraído, transformado e carregado antes de ser consultado ou utilizado. Já na abordagem virtual, o conhecimento não é armazenado explicitamente no grafo, sendo inferido ou calculado sob demanda a partir das fontes de dados subjacentes.

A escolha entre a abordagem materializada e a abordagem virtual depende das características do domínio, dos dados envolvidos, dos requisitos de desempenho e das necessidades de atualização e consulta no contexto do *GCS*.

5.3.4 Criação de visões de ligação Semântica

A correspondência e vinculação de dados é um passo crucial no processo de integração de dados. Envolve a identificação de instâncias de dados correspondentes ou relacionadas em diferentes fontes de dados e o estabelecimento de *links* ou conexões entre eles. Várias abordagens de correspondência de dados podem ser empregadas para identificar instâncias de dados correspondentes.

Uma das abordagens é a correspondência exata de atributos, ou seja, encontrar instâncias de dados que tenham valores idênticos em determinados atributos ou campos. Por exemplo, se ambos os conjuntos de dados tiverem um campo "ID do cliente", a correspondência exata poderá ser executada para identificar instâncias com o mesmo ID.

A abordagem baseada em regras usa regras ou padrões predefinidos para identificar instâncias correspondentes. Essas regras podem ser baseadas em condições específicas, expressões regulares ou operações de transformação de dados. A correspondência baseada em regras é útil quando há padrões ou relacionamentos conhecidos entre as fontes de dados.

A abordagem baseada em correspondência estatística se utiliza de modelos probabilísticos e algoritmos estatísticos para estimar a probabilidade de duas instâncias de dados corresponderem. Essas técnicas levam em consideração vários atributos e seus valores para calcular a similaridade ou as pontuações de correspondência.

Outra abordagem é à correspondência baseada em aprendizado de máquina em que se utiliza algoritmos de aprendizado de máquina para aprender padrões e fazer previsões sobre o status de correspondência de instâncias de dados. Esses algoritmos podem ser treinados usando dados rotulados ou utilizar técnicas de aprendizado não supervisionado para descobrir padrões por conta própria.

Essas abordagens são aplicadas com base nos *links* semânticos identificados, alinhando as instâncias de dados com base em atributos ou relacionamentos compartilhados. No caso de vincular os dados de um sensor com outras fontes de dados, vários tipos de *links* podem ser definidos, dependendo da natureza dos dados e da finalidade da integração, como por exemplo, *links* espaciais, *links* de entidade e *links* contextuais.

Os *links* espaciais são estabelecidos quando os dados do sensor e outras fontes de dados são conectados com base em seus atributos geográficos ou espaciais. Por exemplo, se os dados do sensor incluírem coordenadas de GPS, pode-se vinculá-los aos dados meteorológicos com base nas informações de localização para analisar como as condições climáticas afetam as leituras do sensor.

Os *links* de entidade, por sua vez, são formados conectando os dados do sensor e outras fontes de dados com base em entidades compartilhadas ou atributos comuns. Isso envolve a identificação de campos-chave ou atributos que representam a mesma entidade em ambos os conjuntos de dados. Por exemplo, pode-se vincular os dados do sensor aos dados do cliente com base em uma ID de cliente exclusiva para analisar como as leituras do sensor se relacionam com o comportamento do cliente.

No que se refere aos *links* contextuais, estes são criados ao vincular os dados do sensor a outras fontes de dados que fornecem informações contextuais. Isso pode incluir a vinculação de dados do sensor com dados ambientais, dados de mídia social ou dados de notícias para obter uma compreensão mais ampla dos fatores que influenciam as leituras do sensor. Por exemplo, pode-se vincular os dados do sensor aos dados de mídia social para analisar como o sentimento do público afeta determinadas leituras do sensor.

A definição do tipo de *link* apropriado se deve com base nos objetivos específicos da análise desejada e nas características das fontes de dados envolvidas. Diferentes tipos de *links* podem ser combinados ou usados em conjunto para criar uma compreensão abrangente das relações entre os dados do sensor e outras fontes de dados.

5.3.5 Inferência e Limpeza

Na quinto passo, é realizada a limpeza dos dados para aprimorar a qualidade do GCS gerado. Essa etapa aborda problemas relacionados a conflitos e redundâncias, especialmente no contexto de recursos conectados por meio de *links owl : sameAs*.

A limpeza dos dados visa identificar e resolver conflitos entre diferentes representa-

ções de um mesmo recurso. Isso pode ocorrer quando existem múltiplas instâncias referindo-se à mesma entidade, causando ambiguidade e dificuldades na interpretação dos dados. Ao detectar essas situações de conflito, é necessário realizar a reconciliação, ou seja, escolher a representação mais adequada e corrigir as referências para garantir a consistência no *GCS*.

Além disso, a limpeza dos dados também trata a redundância de informações, que ocorre quando existem múltiplas ocorrências do mesmo dado em diferentes partes do *GCS*. Isso pode resultar em uma sobrecarga desnecessária de dados e dificultar a análise e a interpretação correta das informações. Nessa etapa, é importante identificar e eliminar essas redundâncias, mantendo apenas a informação relevante e necessária para o *GCS*.

No contexto dos recursos conectados por links `owl:sameAs`, é preciso ter cuidado para evitar conflitos ou inconsistências nos dados. Esses *links* são usados para estabelecer conexões entre entidades equivalentes ou representando o mesmo conceito em diferentes partes do *GCS*. Durante a limpeza dos dados, é fundamental verificar se os *links owl:sameAs* estão corretamente associados aos recursos relevantes, garantindo que não haja duplicações ou associações errôneas que possam comprometer a integridade e a qualidade do *GCS*.

No caso dos dados de sensores, deve-se identificar as situações em que um sensor tem ligações semânticas com várias entidades semanticamente relacionadas e de fontes diferentes. Nesse caso, deve-se analisar se existe problema de inconsistência e fazer a fusão das entidades semanticamente equivalentes por meio de *links sameAs*.

5.4 Estudo de Caso: Integração de parâmetros ambientais aos dados turísticos da cidade de Lisboa

Aplicando-se as etapas do *framework SIGIoT-S*, foi construído um *GCS* direcionado ao contexto de Turismo em Lisboa. Esse *GCS* foi construído com o intuito de reunir e organizar informações pertinentes ao segmento de parques na cidade de Lisboa e juntamente com parâmetros ambientais relacionados ao tempo nas regiões conectadas.

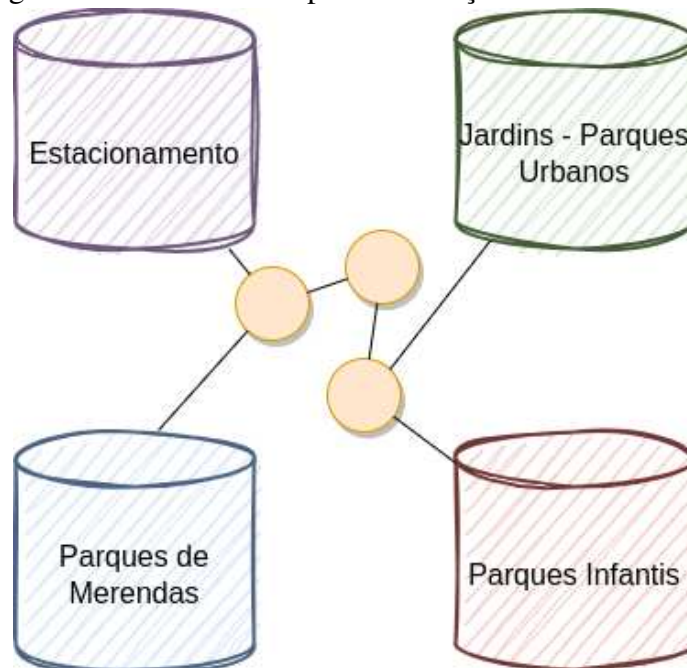
As informações que compõem o grafo foram adquiridas pelo o portal de dados abertos de Lisboa, o qual disponibiliza uma série de conjuntos de dados relevantes.³

A Figura 22 ilustra as bases sobre as quais o *GCS* foi construído, representando a fundação essencial para a sua criação e posterior aplicação do *framework SIGIoT-S*.

A base de dados intitulada “Estacionamento” engloba informações geográficas

³ Fonte: <https://lisboaaberta.cm-figlisboa.pt/index.php/pt/dados/conjuntos-de-dados>

Figura 22 – Bases dados para construção do GCS



Fonte: elaborado pelo autor

juntamente com características detalhadas acerca dos pontos de estacionamento específicos destinados aos veículos “*Tuk Tuk*”. Por meio desta base, é possível compreender de forma precisa a localização exata desses pontos de estacionamento, além das peculiaridades que os cercam.

Já a base de dados denominada “Parques de Merendas” contém uma compilação de informações relativas a locais que oferecem oportunidades para atividades ao ar livre. Nesta base, é possível encontrar detalhes sobre as áreas destinadas a piqueniques, fornecendo uma visão completa das opções disponíveis para aqueles que buscam desfrutar de momentos ao ar livre em Lisboa.

No que tange à base de dados intitulada “Jardins - Parques Urbanos”, esta reúne um conjunto abrangente de dados relacionados aos jardins e parques urbanos presentes na cidade de Lisboa. Por meio desses dados, é possível explorar informações relevantes acerca desses espaços verdes, como suas localizações e descrições, contribuindo para a compreensão da riqueza de opções naturais presentes na área urbana.

Por fim, a base de dados “Parques Infantis” concentra-se em disponibilizar informações relativas aos parques infantis existentes na cidade de Lisboa. Ao explorar essa base, é possível obter um panorama completo sobre esses espaços voltados para o entretenimento e lazer das crianças.

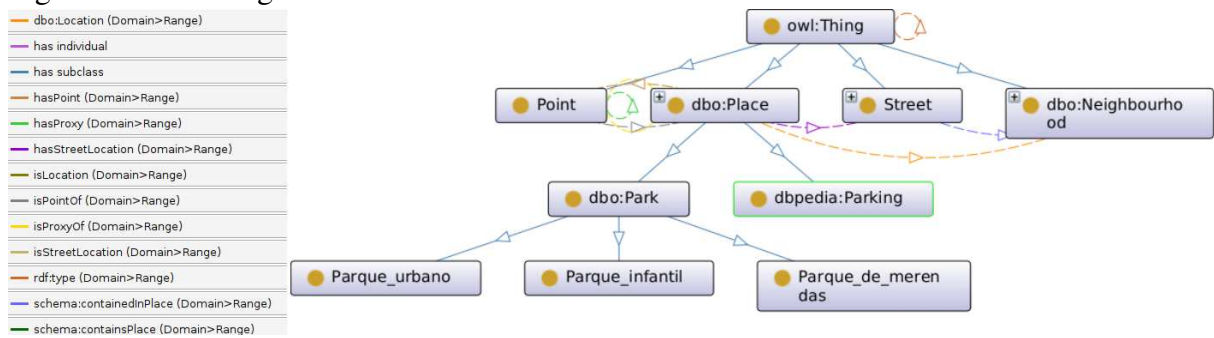
Cada uma dessas bases de dados contribui de maneira única para o GCS voltado ao turismo em Lisboa, fornecendo uma gama diversificada de informações que enriquecem a expe-

riência dos visitantes e residentes ao explorar a cidade e suas ofertas de lazer e entretenimento.

Com o objetivo de viabilizar a integração dessas diferentes bases de dados, foi desenvolvida uma ontologia para a atribuição de termos e relações semânticas aos dados. Essa ontologia desempenha um papel crucial ao fornecer uma estrutura comum de entendimento, permitindo que os dados provenientes das diferentes fontes sejam interpretados de maneira coesa e interconectada.

A Figura 23 ilustra as classes fundamentais presentes na ontologia Turismo, que foi concebida para enriquecer os dados com um contexto semântico enriquecido. Cada classe presente na ontologia desempenha um papel específico na definição e categorização dos elementos abordados pelas bases de dados mencionadas anteriormente. Isso cria um arcabouço sólido para a atribuição de significado e a compreensão interligada dos dados, impulsionando a capacidade de extrair informações relevantes e inter-relacionadas.

Figura 23 – Ontologia Turismo



Fonte: elaborado pelo autor

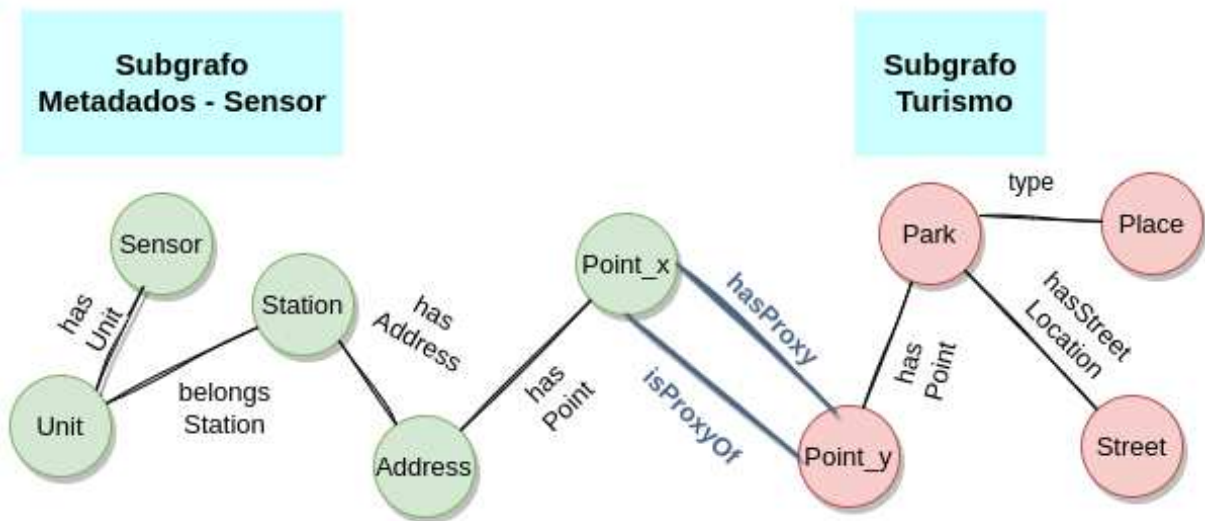
No *framework SIGIoT-S*, os dados estáveis do grafo de conhecimento de dados *IoT* são incorporados ao *GCS*. O objetivo central é estabelecer conexões entre os subgrafos, permitindo assim a integração harmoniosa das informações presentes em ambos os grafos de conhecimento. No subgrafo de dados *IoT*, estão contidas informações detalhadas sobre as características dos sensores. Nesse contexto, é possível identificar as localizações das estações de sensores, os tipos de sensores utilizados, as unidades de medida aplicadas, entre outras informações relevantes.

No subgrafo voltado às informações turísticas da cidade de Lisboa, encontram-se dados específicos sobre os parques e estacionamentos, incluindo suas coordenadas geográficas. Com o propósito de exemplificar o processo de integração entre esses dois subgrafos, foram introduzidas as propriedades *hasProxy* e *isProxyOf*. Essas propriedades desempenham um papel fundamental na construção das ligações entre os diferentes conjuntos de dados. Para tal, uma re-

gra foi estabelecida, permitindo a associação das propriedades mencionadas e, conseqüentemente, viabilizando a integração das informações relacionadas às localizações geográficas.

Para concretizar essa integração, um procedimento foi implementado: ao fornecer as coordenadas de latitude e longitude de uma estação de sensores, verifica-se dentro de um raio de 5 quilômetros existe algum dos pontos de interesse turístico, como um parque ou estacionamento, dentro dessa área especificada. Caso fosse identificada a presença de um parque ou estacionamento dentro desse limite, as propriedades *hasProxy* e *isProxyOf* eram automaticamente atribuídas entre os pontos de dados relevantes. A figura 24 apresenta um exemplo dessa integração.

Figura 24 – Integração de subgrafos



Fonte: elaborado pelo autor

Com essa estratégia, alcança-se a conexão efetiva entre os dois subgrafos, o que facilita a navegação conjunta por meio das informações integradas. Essa abordagem não apenas estabelece uma ligação entre os conjuntos de dados, mas também promove uma experiência de exploração mais rica e contextualizada dos subgrafos. A proposta é que o usuário possa ter acesso a informações relacionadas ao tempo e o ambiente dos pontos turísticos desejados.

Para ilustrar a integração dos subgrafos, foi desenvolvida uma consulta *SPARQL* que é aplicada aos repositórios *Turismo* e *Metadata_sensor*. A Consulta 1 foi concebida com o propósito de extrair informações dos subgrafos, identificando tanto as estações de sensores como os locais de interesse turístico próximos a essas estações.

Código-fonte 1 – Consulta *Sparql* em subgrafos integrados

```

1 PREFIX sosa: <http://www.w3.org/ns/sosa/>
2 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
3 PREFIX iotl: <http://purl.oclc.org/NET/UNIS/fiware/iot-lite/>
4 PREFIX owl: <http://www.w3.org/2002/07/owl#>
5 prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
6 PREFIX ex: <http://example.com/>
7 prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos/>
8 Prefix dbo: <http://dbpedia.org/ontology/>
9 Prefix tur: <http://www.semanticweb.org/example/ontologies/2023/6/tourism/>
10 SELECT distinct ?station ?local WHERE{
11     SERVICE<repository:Metadata_sensor>{
12         ?sensTemp rdf:type sosa:Sensor;
13             iotl:hasUnit ?unit;
14             iotl:hasQuantityKind ?kind;
15             ex:belongsStation ?station .
16         ?station ex:hasAddress ?end .
17         ?end geo:location ?point1 .
18     }
19     SERVICE<repository:Turismo>{
20         ?local a dbo:Park;
21         tur:hasPoint ?point2 .
22         ?point2 tur:isProxyOf ?point1 .
23     }
24 }

```

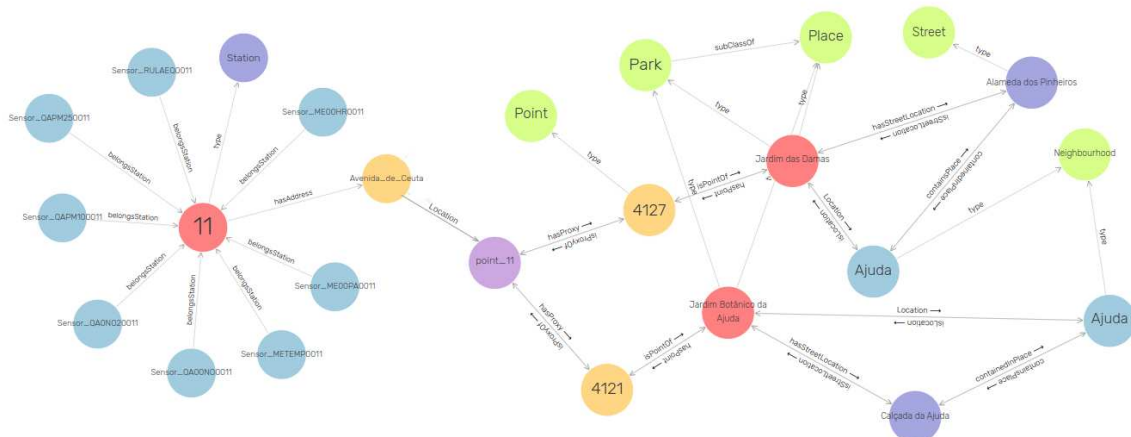
Nessa consulta, são usadas diversas abreviações de prefixo para simplificar a leitura do código *SPARQL*. Através das cláusulas `SERVICE<repository:Metadata_sensor>` e `SERVICE<repository:Turismo>`, a consulta percorre os repositórios *Metadata_sensor* e *Turismo*, respectivamente. Através da propriedade `tur:isProxyOf`, a consulta identifica quais os pontos turísticos ficam próximos as estações de sensores.

Essa consulta exemplifica a funcionalidade da integração dos subgrafos, permitindo recuperar informações valiosas sobre as interações entre as estações de sensores e os locais de interesse turístico próximos. Essa abordagem proporciona *insights* significativos para a tomada de decisões informadas e análises aprofundadas nas áreas de monitoramento de sensores e turismo.

Um exemplo da integração bem-sucedida dos subgrafos é apresentado na Figura 25. Nessa ilustração, é possível visualizar de maneira concreta um ponto turístico em proximidade com uma estação de sensor. Essa representação gráfica traz à vida a conexão entre informações aparentemente distintas, mostrando como elas se entrelaçam no contexto da cidade.

Imagine que o ponto turístico em questão seja um parque famoso de Lisboa. Com

Figura 25 – Amostra de nós de subgrafos integrados



Fonte: elaborado pelo autor

a integração dos subgrafos, quando você seleciona esse ponto no mapa, os dados relacionados às estações de sensores nas proximidades também são acessados. Isso significa que você pode obter informações valiosas sobre os parâmetros ambientais daquela área específica do parque.

Por exemplo, os sensores podem fornecer dados sobre a qualidade do ar, temperatura, umidade e outros fatores ambientais. Essas informações detalhadas podem enriquecer significativamente a experiência dos visitantes e dos próprios gestores do parque. Imagine um turista que queira fazer um piquenique: ele pode verificar a qualidade do ar e as condições climáticas antes de decidir o melhor momento para sua visita. Ou ainda, os administradores do parque podem utilizar esses dados para tomar decisões informadas sobre a gestão ambiental e o conforto dos visitantes.

Portanto, a figura em questão não apenas ilustra a conexão visual entre o ponto turístico e a estação de sensor, mas também enfatiza como essa integração concreta traz benefícios tangíveis aos usuários e à gestão urbana como um todo.

6 METAGRAFO-IOT: UMA ONTOLOGIA PARA ANOTAÇÃO SEMÂNTICA DE METADADOS DE UM GRAFO DE CONHECIMENTO DE DADOS *IoT*

O vocabulário de metadados desempenha um papel fundamental na definição de um modelo de anotação de alto nível, responsável por registrar informações descritivas sobre um recurso de informação. Esse vocabulário é essencial para facilitar o processo de compreensão semântica dos dados (SATIJA *et al.*, 2020b). Ele pode ser elaborado de maneira geral, capturando a semântica de recursos de forma abrangente, ou pode ser desenvolvido para um domínio específico, a fim de capturar informações mais especializadas.

A construção de um grafo de conhecimento de dados *IoT* é uma tarefa complexa que resulta na geração de diferentes metadados em cada etapa do processo. Esses metadados gerados permitem a descrição do grafo de conhecimento e tornam-o acessível para serviços e aplicações de dados. Portanto, é necessário desenvolver um vocabulário para a anotação semântica desses metadados.

Neste capítulo é apresentado na Seção 6.1 os metadados de um grafo de conhecimento de dados *IoT*. A Seção 6.2 apresenta a metodologia utilizada para o desenvolvimento da ontologia *MetaGraph-IoT*. A seção 6.3 apresenta um estudo de caso sobre a construção de um grafo de conhecimento a partir de metadados e que utiliza a ontologia *MetaGraph-IoT* na anotação semântica dos dados.

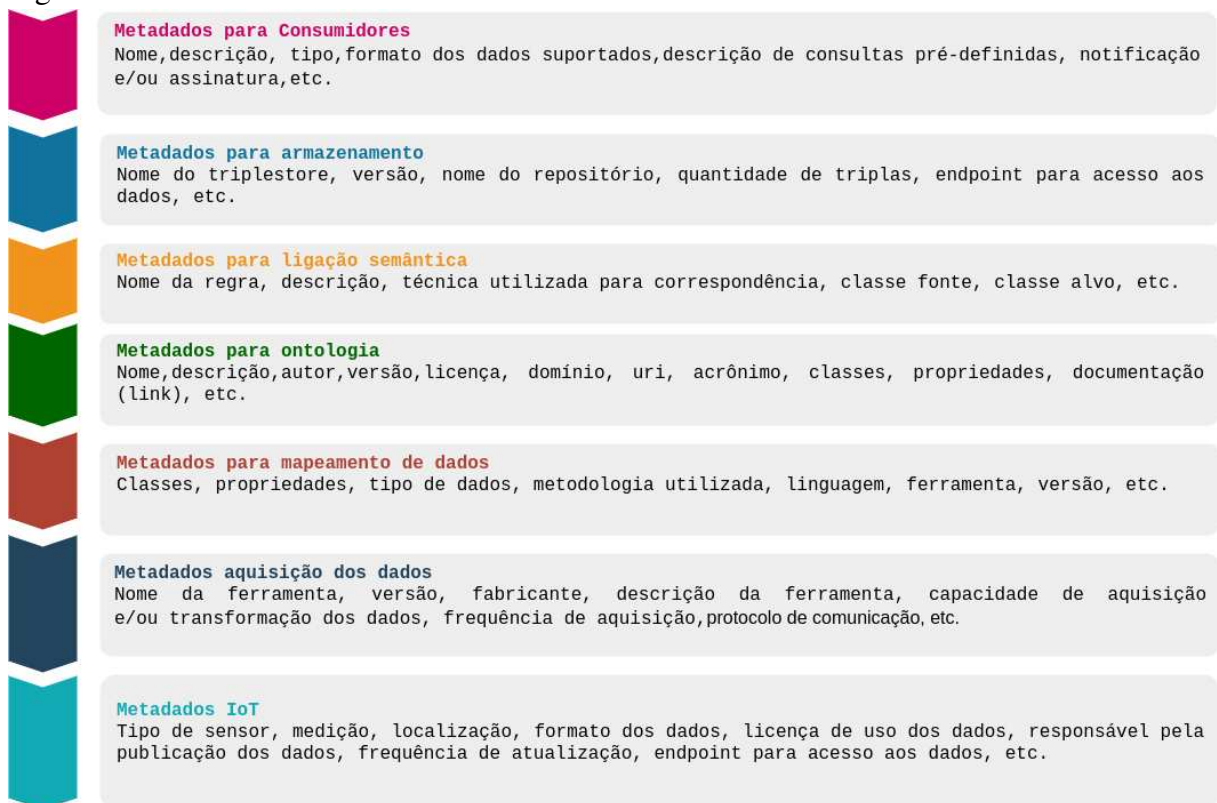
6.1 Metadados relacionados ao Grafo de Conhecimento de dados *IoT*

De acordo com (SATIJA *et al.*, 2020a), os metadados podem ser classificados em diferentes tipos, tais como descritivos, estruturais, técnicos, de proveniência, entre outros. Os metadados descritivos são responsáveis por descrever um recurso de informação, permitindo assim um maior entendimento sobre o conteúdo abordado. Já os metadados estruturais referem-se à organização da informação, enquanto os metadados técnicos descrevem informações sobre o funcionamento de uma entidade de informação. Por fim, os metadados de proveniência abordam informações relacionadas ao ciclo de vida de um recurso.

Na Figura 26 é apresentado um exemplo dos metadados de um grafo de conhecimento de dados *IoT*. Esses metadados consistem na descrição das fontes de dados que compõem o grafo, na descrição da ferramenta responsável pela aquisição dos dados (chamada de produtor), no vocabulário semântico utilizado, nas regras utilizadas para integração dos dados, nas entidades responsáveis pelo armazenamento dos dados e nos consumidores desses dados.

Através dos metadados, torna-se possível descrever não apenas o conteúdo do grafo de conhecimento, mas também os processos pelos quais os dados passaram até chegar ao seu estado atual. Esse tipo de anotação é de extrema importância, pois permite que o grafo de conhecimento possa ser reutilizado em outros domínios e aplicações. Além disso, a anotação de metadados auxilia os usuários a conhecer e entender melhor o grafo de conhecimento em questão.

Figura 26 – Metadados de um Grafo de Conhecimento de dados *IoT*

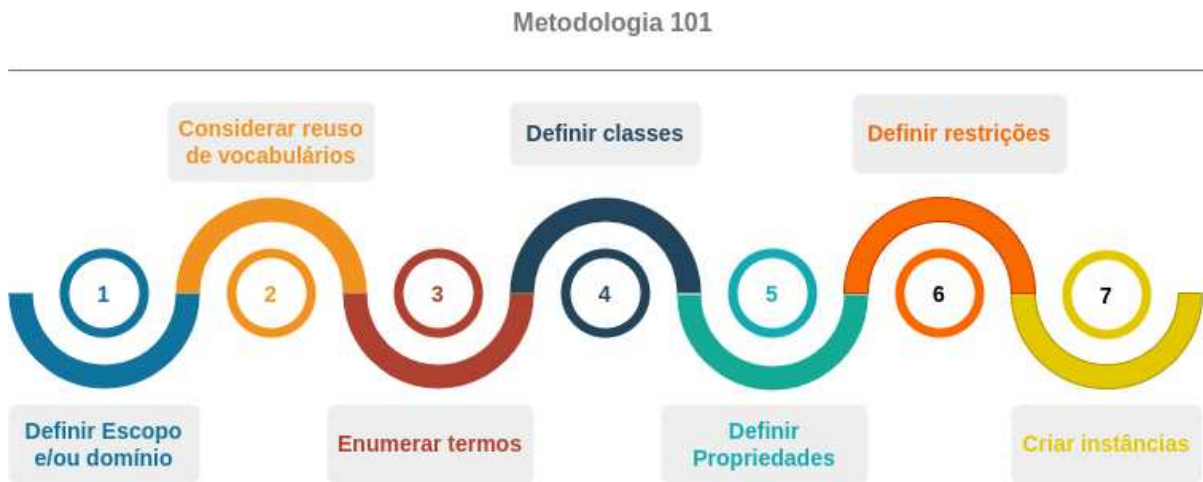


Fonte: elaborado pelo autor

6.2 Metodologia utilizada na criação da ontologia *MetaGrafoIoT*

Para criar a ontologia *MetaGraphIoT*, utilizou-se a metodologia 101 (também conhecida como *Ontology Development 101*) proposta por Noy e McGuinness (NOY *et al.*, 2001). Essa metodologia é simples e consiste em sete etapas: determinar o domínio e o escopo da ontologia, considerar a reutilização de ontologias existentes, listar os termos importantes da ontologia, definir as classes e a hierarquia de classes, definir as propriedades das classes, definir as restrições e criar instâncias. A Figura 27 ilustra o ciclo de vida para criação de uma ontologia com base na metodologia 101.

Figura 27 – Etapas da metodologia 101



Fonte: adaptado de (NOY *et al.*, 2001)

6.2.1 Escopo da ontologia *MetaGrafoIoT* e questões de competência

A primeira etapa da metodologia consiste em definir o domínio e o escopo da ontologia, o qual pode ser estabelecido por meio de questões de competência. Essas questões são utilizadas como requisitos funcionais da ontologia (REN *et al.*, 2014), e são expressas como perguntas que se espera responder com a ajuda da ontologia. Dessa forma, as questões de competência são úteis na validação da ontologia.

A ontologia *MetaGrafoIoT* foi desenvolvida com o objetivo de fornecer um modelo semântico para anotação de metadados que surgem durante a construção, manutenção e consumo de um grafo de conhecimento de dados *IoT*. Cada etapa da construção do grafo gera metadados que descrevem os processos e ferramentas envolvidas, como fontes de dados, ontologias utilizadas, mapeamentos, regras de integração de dados e ferramentas empregadas em cada etapa.

Os metadados do grafo de conhecimento de dados *IoT* permitem ao usuário conhecer melhor o tipo de dados presentes no grafo, quais aplicações consomem o grafo, qual modelo de informação foi aplicado sobre os dados, entre outros aspectos importantes.

Foram definidas algumas questões de competência para alinhar o escopo da ontologia *MetaGrafoIoT*, incluindo:

- Quais tipos de dados de sensores estão presentes no grafo de conhecimento de dados *IoT*?
- Qual é a licença de uso dos dados de sensores presentes no grafo de conhecimento de dados *IoT*?
- Quem são os agentes responsáveis pela disponibilização dos dados de sensores no grafo de conhecimento de dados *IoT*?

- Quais vocabulários foram utilizados como modelo de informação no grafo de conhecimento de dados *IoT*?
- Qual é o *endpoint* utilizado para acessar o grafo de conhecimento de dados *IoT*?
- Quais classes foram utilizadas para a anotação semântica dos dados no grafo de conhecimento de dados *IoT*?
- Quais serviços e/ou aplicações consomem o grafo de conhecimento de dados *IoT*?
- No caso de integração com ontologia de domínio ou fonte de dados externa, quais são as classes utilizadas no processo de *linkage* de dados no grafo de conhecimento de dados *IoT*?

Essas questões de competência foram utilizadas para guiar o processo de criação da ontologia *MetaGrafoIoT* e garantir que todas as informações relevantes para o uso e compreensão do grafo de conhecimento de dados *IoT* estejam devidamente anotadas.

6.2.2 Vocabulários reusados

A segunda etapa consiste na reutilização de ontologias existentes, o que é fundamental para assegurar a interoperabilidade semântica com outras aplicações que utilizam diferentes vocabulários (NOY *et al.*, 2001). É importante destacar que essa etapa não implica em reutilizar toda uma ontologia, mas sim conceitos específicos de outras ontologias que sejam similares aos conceitos que a ontologia em construção visa abordar (ISOTANI; BITTENCOURT, 2015).

Com base nesse entendimento, foram examinadas ontologias voltadas para a anotação de metadados, bem como ontologias no contexto de *IoT*, possibilitando o reuso de diversos termos. O Quadro 4 apresenta os vocabulários reutilizados na construção da ontologia *MetaGrafoIoT*.

A **Ontologia de Proveniência - PROV-O**¹ é utilizada para descrever informações de proveniência geradas em diferentes sistemas e contextos. No contexto da ontologia *MetaGrafoIoT* tem como objetivo descrever o responsável pela disponibilização dos dados que serão utilizados.

O **Vocabulário De Catálogo De Fluxos Vinculados (VoCaLS)** (TOMMASINI *et al.*, 2018) é utilizado para descrever *streams* e os serviços de *streaming* na *web*. Na ontologia *MetaGrafoIoT* tem como objetivo descrever o *stream* que está sendo capturado, o *endpoint* que dá acesso a este e a ferramenta utilizada para publicação dos dados.

O **Vocabulário de Catálogo de Dados (DCAT)**² foi projetado para facilitar a

¹ <https://www.w3.org/TR/2013/REC-prov-o-20130430/>

² <https://www.w3.org/TR/vocab-dcat-2/>

Quadro 4 – Vocabulários utilizados na construção da ontologia *MetaGraphIoT*

VOCABULÁRIO	PREFIXO	NAMESPACES IRI
DUBLIN CORE TERMS	dct	https://dublincore.org/specifications/dublin-core/dcmi-terms/
PROV-O	prov	https://www.w3.org/TR/prov-o/
OMV (ONTOLOGY METADATA VOCABULARY)	omv	http://omv.ontoware.org/ontology#
DUBLIN CORE ELEMENTS	dce	http://purl.org/dc/elements/1.1/
DESCRIÇÃO DE SERVIÇO SPARQL 1.1	sd	https://www.w3.org/TR/sparql11-service-description/
FOAF (FRIEND OF A FRIEND VOCABULARY)	foaf	http://xmlns.com/foaf/0.1/
VOCALS (VOCABULARY CATALOG OF LINKED STREAMS)	vocals	http://w3id.org/rsp/vocals/
DATA CATALOG VOCABULARY	dcat	http://www.w3.org/ns/dcat/
ML SCHEMA	ml	http://www.w3.org/ns/mls#
IOT-STREAM	iots	http://purl.org/iot/ontology/iot-stream#

Fonte: Elaborado pelo autor

interoperabilidade entre catálogos de dados publicados na *Web*. Na ontologia *MetaGrafoIoT* busca descrever o serviço de dados e o acesso a este.

O **Vocabulário de Metadados de Ontologia (OMV)**³ é específico para descrição de recursos de uma ontologia. Neste contexto é aplicado para descrever a ontologia utilizada para anotação semântica dos fluxos de dados.

O **Serviço de Descrição Sparql (SD)**⁴ tem como proposta descrever serviços *Sparql*. Sendo assim, na ontologia *MetaGrafoIoT* é utilizado para descrever o *endpoint sparql*, a linguagem e o formato dos dados.

Os **vocabulários Dublin Core Terms (dct) e Dublin Core Elements (dce)**⁵ referem-se a termos para anotações de metadados de serviços físicos e da *web*. Na ontologia *MetaGrafoIoT* são usados termos para atribuir título e descrição das entidades envolvidas, como também a frequência em que um fluxo de dados é obtido.

O **vocabulário FOAF**⁶, por sua vez, é utilizado na descrição do agente responsável

³ <http://omv2.sourceforge.net/>

⁴ <https://www.w3.org/TR/sparql11-service-description/vocab>

⁵ <https://www.dublincore.org/specifications/dublin-core/dcmi-terms/>

⁶ <http://xmlns.com/foaf/spec/>

pela publicação dos dados e o **ML-Schema**⁷ possui classes e propriedades para representar algoritmos de aprendizagem de máquina, os quais podem ser utilizados em serviços de análise. O **IoT-stream**⁸ é utilizado para representar as características dos fluxos derivados.

6.2.3 Termos importantes na ontologia

A terceira etapa da metodologia 101 consiste em definir os termos que a ontologia deverá abranger. Essa etapa não exige que os termos sejam definidos como classes ou propriedades, mas sim que sejam identificados quais termos são importantes para a declaração e quais relações existem entre eles (NOY *et al.*, 2001).

Os termos mais relevantes levantados incluem: sensores, agente responsável pela publicação dos dados, ontologia, mapeamento, repositório de dados, linguagem de consulta, formato dos dados, licença dos dados, produtor e consumidor dos dados, regras de *linkage*, serviço de publicação dos dados, eventos, entre outros.

6.2.4 Classes, propriedades, restrições e instâncias

As próximas etapas da metodologia correspondem à criação da ontologia, que envolve a definição de classes, propriedades, restrições e instâncias. Com base nos termos previamente definidos e nos conceitos a serem reutilizados, deu-se início à criação da ontologia *MetaGrafoIoT*.

Essa ontologia é composta por classes que descrevem os metadados dos seguintes segmentos: fonte de dados, anotação semântica, integração de dados, serviço de armazenamento de dados e consumidores.

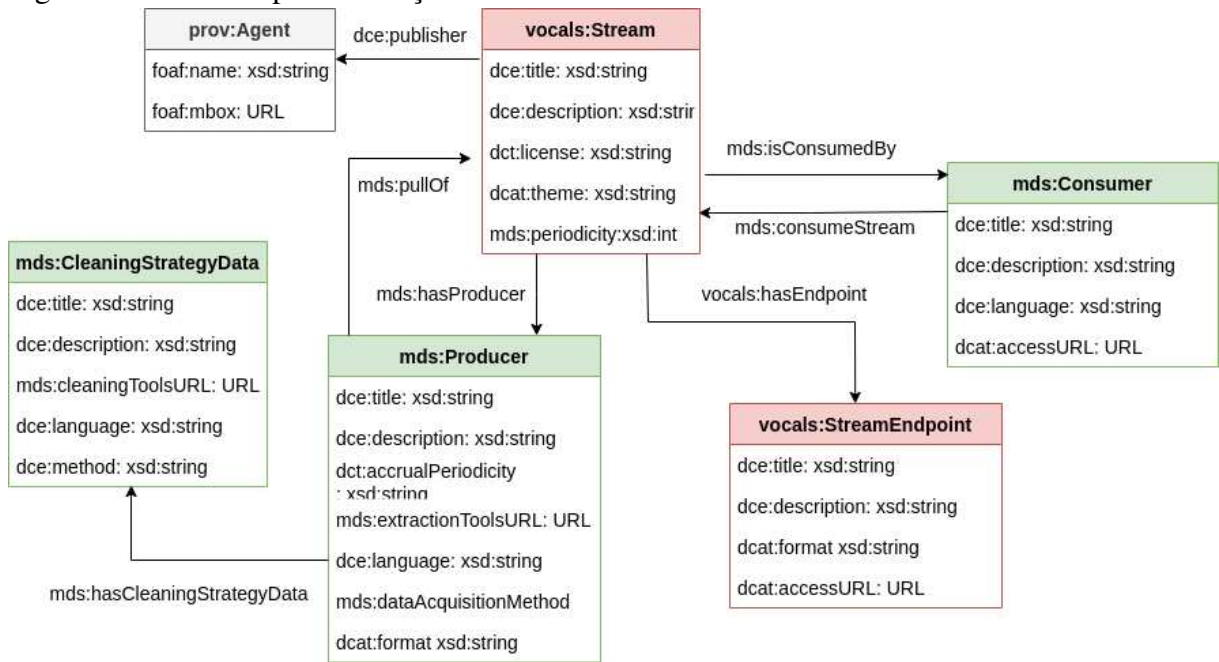
A Figura 28 apresenta as classes que descrevem as fontes de dados presentes no grafo de conhecimento da ontologia *MetaGrafoIoT*. As classes *Stream* e *Agent* incluem metadados relacionados aos fluxos de dados e ao agente responsável pela publicação dos dados. Através dessas classes é possível identificar informações importantes como o tipo de dados que o fluxo se refere, a frequência de atualização e a licença de uso. A classe *StreamEndpoint* fornece informações sobre o endpoint de acesso aos dados e o formato dos dados.

Com base nas informações sobre o acesso e formato dos dados, o produtor pode adquirir os dados. A classe *Producer* fornece informações sobre a ferramenta ou linguagem

⁷ <http://ml-schema.github.io/documentation/ML%20Schema.html>

⁸ <http://iot.ee.surrey.ac.uk/iot-crawler/ontology/iot-stream/>

Figura 28 – Classes para descrição das fontes de dados

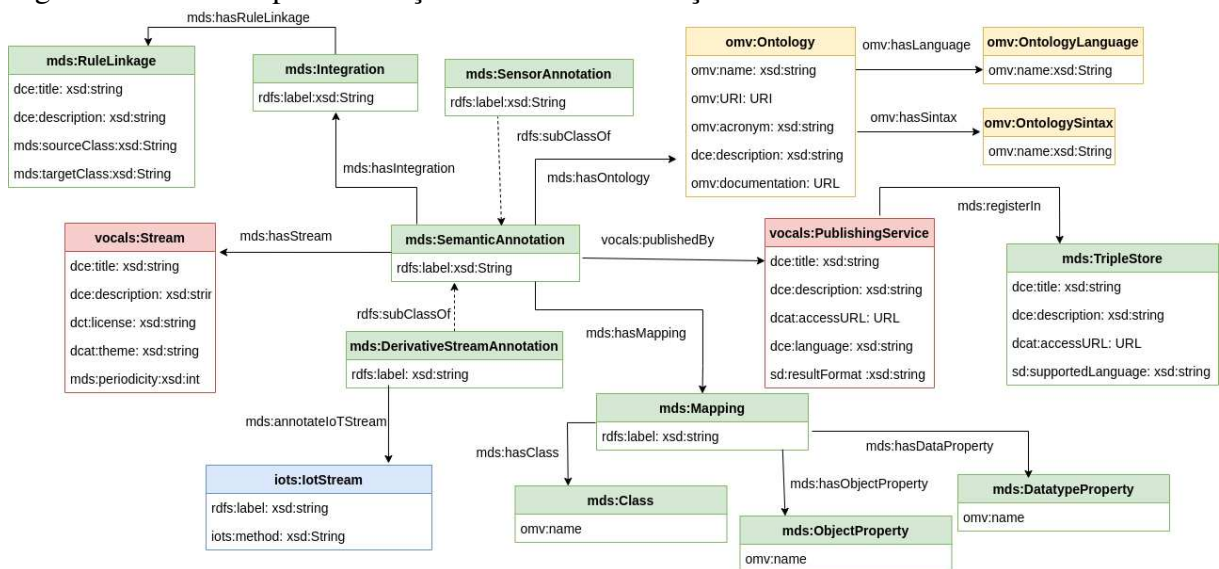


Fonte: elaborado pelo autor

utilizada para a aquisição dos dados e a frequência com que essa tarefa é realizada.

A classe CleaningStrategyData aborda a estratégia de limpeza de dados aplicada. Quando se trabalha com dados de sensores, a limpeza de dados é importante para eliminar dados inválidos decorrentes de falhas na comunicação entre os dispositivos ou exclusão de dados irrelevantes para o domínio do grafo de conhecimento.

Figura 29 – Classes para descrição da tarefa de anotação semântica dos dados



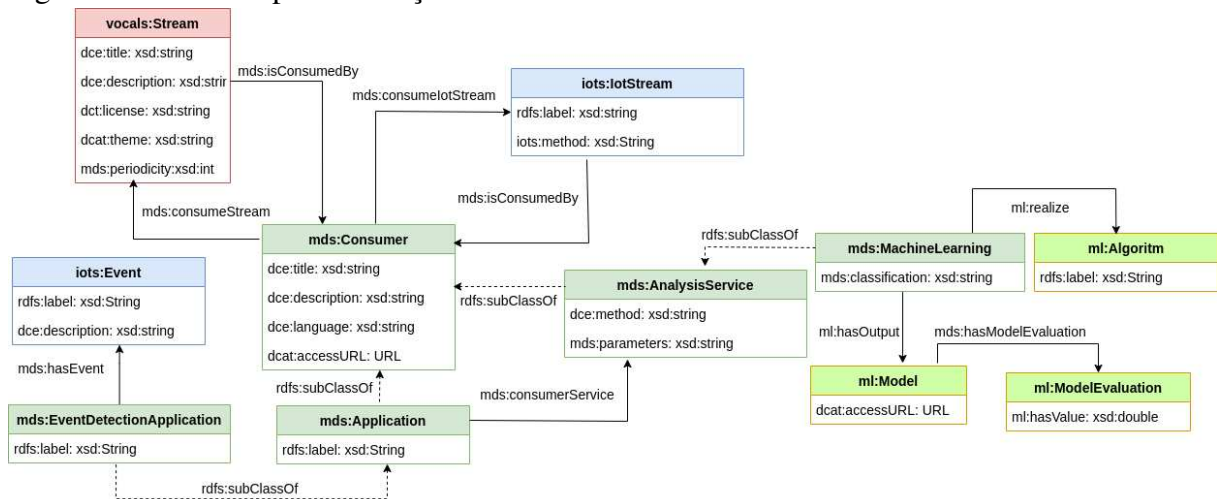
Fonte: elaborado pelo autor

A Figura 29 apresenta as classes utilizadas na descrição do processo de anotação

semântica e integração de dados no grafo de conhecimento *IoT*. A anotação semântica é realizada por meio das classes *SensorAnnotation* e *DerivativeStreamAnnotation* para anotação dos sensores e fluxos derivados, respectivamente. Como existem diferentes tipos de anotações semânticas, cada uma é vinculada a uma ontologia específica. Portanto, é importante identificar o tipo de anotação semântica e a ontologia utilizada.

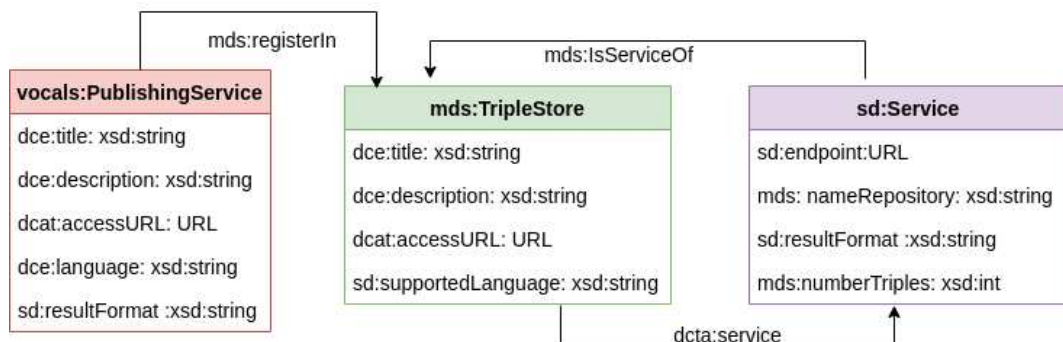
A classe *SemanticAnnotation* é conectada às classes *PublishingService* e *Ontology*, que descrevem o serviço de publicação e a ontologia utilizada. As classes *Integration* e *RuleLinkage* são utilizadas para descrever os metadados relacionados à integração com outros conjuntos de dados. Por fim, as classes *Mapping*, *Class*, *DatatypeProperty* e *ObjectProperty* descrevem o mapeamento entre os dados e a ontologia.

Figura 30 – Classes para descrição dos consumidores



Fonte: elaborado pelo autor

Figura 31 – Classes para descrição do serviço de armazenamento dos dados



Fonte: elaborado pelo autor

A Figura 30 ilustra as classes envolvidas na descrição dos consumidores do grafo de conhecimento de dados *IoT*. A relação da classe *Consumer* com as classes *Stream* e *IoTStream*

permite identificar quais fluxos de dados são consumidos pelas aplicações e/ou serviços de dados. Portanto, a classe `Consumer` tem como objetivo descrever os consumidores do grafo de conhecimento, representados pela classe ***Application*** e pela classe `AnalysisService`. A classe `EventDetectionApplication` descreve as aplicações orientadas a eventos, enquanto a classe `MachineLearning` descreve os serviços de análise classificados como métodos de aprendizagem de máquina.

Por sua vez, a Figura 31 apresenta as classes relacionadas aos serviços de armazenamento de dados. As classes `TripleStore` e `Service` são responsáveis pela anotação dos metadados relacionados à entidade de armazenamento e à descrição do repositório de dados. É importante destacar que uma instância de um `triplestore` pode possuir mais de um repositório.

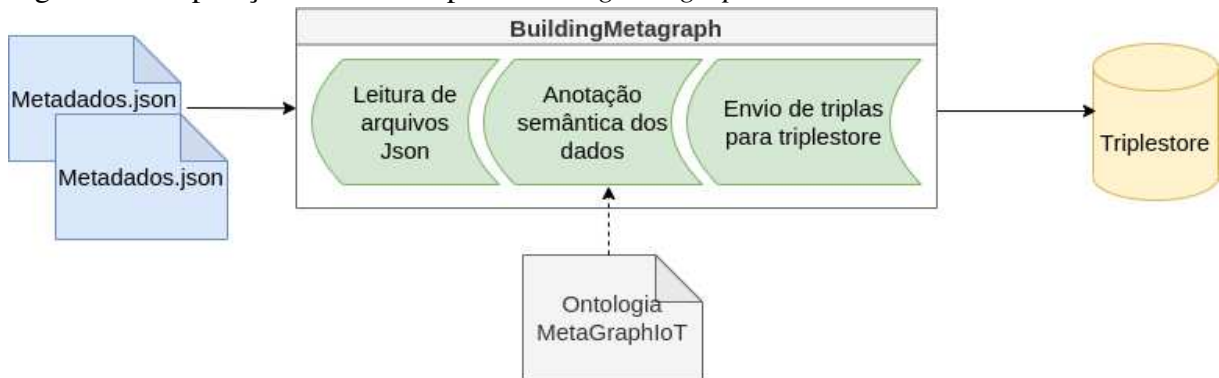
Por fim, o sétimo passo do processo de criação do grafo de conhecimento refere-se à criação de instâncias, as quais são exemplificadas no estudo de caso apresentado no capítulo ???. A Figura 32 apresenta a ontologia *MetaGrafoIoT* em sua totalidade, a qual está disponível para consulta por meio deste *link*.

6.3 Metagraph: Construindo um grafo de conhecimento a partir de metadados - um estudo de caso

No decorrer da construção de um grafo de conhecimento, diversos metadados são gerados. Esses metadados permitem a descrição de todo o processo de construção do grafo, assim como a própria representação do grafo de conhecimento resultante. Nesta seção, exploraremos a criação de um grafo de metadados, intitulado *Metagraph*.

O *Metagraph* foi desenvolvido a partir dos metadados do grafo de conhecimento de dados *IoT* referente à cidade de Lisboa. Para a construção do *Metagraph*, foi desenvolvido um código para realizar a anotação semântica dos metadados, fazendo uso dos conceitos e propriedades da ontologia *MetaGrafoIoT*. Essa estrutura de código foi nomeada de *buildingMetagraph*⁹. O *buildingMetagraph* foi implementado utilizando a linguagem de programação Java¹⁰, com o objetivo de ler arquivos *JSON* contendo os metadados gerados durante o processo de construção do grafo de conhecimento. A Figura 33 ilustra o fluxo de operações realizado pelo *buildingMetagraph*.

Figura 33 – Operações realizadas pelo *buildingMetagraph*



Fonte: elaborado pelo autor

Os metadados são disponibilizados por meio de arquivos *JSON*. O código *buildingMetagraph* realiza a leitura desses arquivos e os transforma em objetos de dados, seguindo o modelo de dados definido. Esse modelo segue o esquema de dados e relações estabelecidos na ontologia *MetaGrafoIoT*. Após a leitura dos dados, eles são enviados para o método encarregado da anotação semântica. Este método recebe os objetos de dados como entrada, mapeia esses dados para os conceitos e relações da ontologia *MetaGrafoIoT* e, por fim, gera as triplas resultantes desse mapeamento. As triplas geradas são então enviadas para o *triplestore* definido

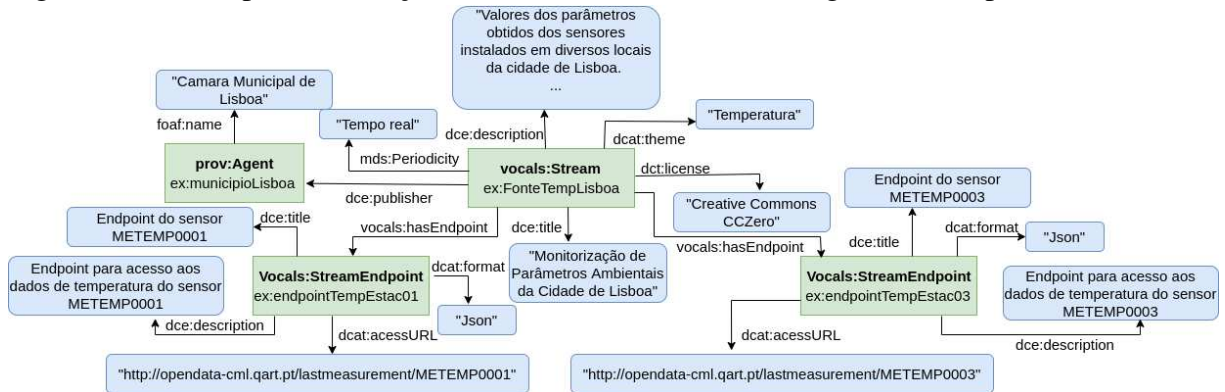
⁹ <https://github.com/searchmdcc/buildingMetagraph/tree/master>

¹⁰ [https://pt.wikipedia.org/wiki/Java_\(linguagem_de_programacao\)](https://pt.wikipedia.org/wiki/Java_(linguagem_de_programacao))

por meio da execução de uma consulta *SPARQL*.

A Figura 34 exemplifica um sensor que faz parte do grafo de conhecimento de dados *IoT*. É possível observar que os metadados permitem identificar os tipos de sensores presentes no grafo de conhecimento, além de fornecer informações relevantes como a licença de uso dos dados, o agente responsável pela publicação dos dados e os *endpoints* para acesso aos dados.

Figura 34 – Exemplo de anotação semântica utilizando a ontologia *MetaGraphIoT*



Fonte: elaborado pelo autor

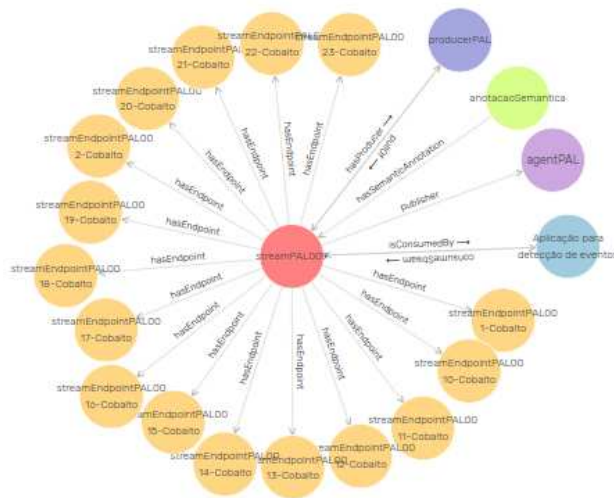
Optou-se pelo uso do *triplestore GraphDB* para armazenamento dos dados, devido à sua capacidade de armazenar os dados gratuitamente e também pelas ferramentas que oferece para a exploração visual dos dados.

Dentro do *GraphDB*, foi criado um repositório de dados denominado *metagraph*, designado para o armazenamento do grafo de conhecimento. A Figura 35 ilustra um exemplo de um subgrafo relacionado ao sensor que realiza medições de Cobalto. Nessa figura, é possível identificar os *endpoints* que disponibilizam os dados de Cobalto, bem como o agente responsável pela divulgação dos dados, o produtor que adquiriu os dados, as ontologias empregadas na anotação semântica dos dados e, por fim, a aplicação que consome esse tipo de dado.

Dessa forma, por meio das informações presentes no *Metagraph*, os usuários poderão compreender os dados e as relações existentes no grafo de conhecimento de dados *IoT*. Os metadados disponibilizados permitem avaliar a proveniência e o rastreamento dos dados. Além disso, o *Metagraph* também auxilia na manutenção contínua do grafo de conhecimento, uma vez que facilita a identificação dos consumidores e dos fluxos de dados que esses consumidores utilizam.

Para apresentar as informações presentes no grafo de conhecimento de metadados, o *Metagraph*, foram elaboradas consultas *SPARQL*. A Consulta 2 tem como proposta apresentar informações relacionadas aos sensores que compõem o grafo de conhecimento de dados *IoT*.

Figura 35 – Exemplo de um subgrafo do Metagraph



streamPAL009

streamPAL009

Types:

<http://w3id.org/rsp/vocals/Stream>

RDF rank:

0

Pressão Atmosférica (mbar), Humidade Relativa, (%) Intensidade do Vento (km/h), Direção do Vento (Grau), Precipitação (mm), Radiação Global (um) e Radiação UV (Índice UV).

<http://purl.org/dc/elements/1.1/title>

Monitorização de Parâmetros Ambientais da Cidade de Lisboa

<http://www.w3.org/ns/dcat/theme>

Cobalto

<https://www.ufc.mdcc.metragrafiot/periodicity>

Tempo real

Fonte: elaborado pelo autor

Código-fonte 2 – Consulta *Sparql* - fontes de dados

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX vocal:<http://w3id.org/rsp/vocals/>
3 PREFIX dct:<http://dublincore.org/specifications/dublin-core/dcmi-terms/>
4 PREFIX dce: <http://purl.org/dc/elements/1.1/>
5 PREFIX dcat:<http://www.w3.org/ns/dcat/>
6 PREFIX foaf: <http://xmlns.com/foaf/0.1/>
7
8 SELECT ?stream ?theme ?licence ?nameAgent WHERE{
9   ?stream rdf:type vocal:Stream;
10     dcat:theme ?theme;
11     dct:license ?licence;
12     dce:publisher ?agent.
13     ?agent foaf:name ?nameAgent.
14 }

```

Nesta consulta é retornado informações relevantes sobre os sensores incorporados ao grafo de conhecimento. Esses detalhes compreendem o nome do sensor, a categoria da medição que o classifica como um tema específico, a licença que rege o uso dos dados e o nome da entidade responsável pela sua publicação. Essa consulta direciona a atenção para o entendimento fundamental dos sensores presentes no grafo.

Adicionalmente, a consulta disponibiliza informações cruciais relacionadas à aquisição dos dados e à forma de acesso aos mesmos. Esses detalhes abrangem a maneira pela qual os

dados foram coletados e processados, além de fornecer a *URL* do ponto de acesso (*endpoint*) que possibilita a interação com esses dados. Essa parte da consulta garante uma compreensão abrangente sobre a proveniência e a acessibilidade dos dados.

Na Figura 36, é apresentado um recorte do resultado obtido por meio dessa consulta. Esse recorte da saída demonstra de forma tangível como as informações são organizadas e retornadas. A combinação de dados essenciais sobre os sensores, sua temática, licenciamento e acesso facilita a compreensão das capacidades e possibilidades oferecidas pelo grafo de conhecimento. Isso contribui para uma compreensão mais clara e informada dos aspectos essenciais relacionados aos dados e aos sensores incorporados ao grafo de conhecimento.

Figura 36 – Parte do resultado obtido pela consulta 2

stream	theme	licence	nameAgent	methodAcquicition	url 
http://example.com/streamPAL002	Contagem_de_veiculos	Creative commons CCZero	Câmara Municipal de Lisboa	Request HTTP	URL: http://opendata-cml.qart.pt/lastmeasurement/CT0VTH0034
http://example.com/streamPAL002	Contagem_de_veiculos	Creative commons CCZero	Câmara Municipal de Lisboa	Request HTTP	URL: http://opendata-cml.qart.pt/lastmeasurement/CT0VTH0069
http://example.com/streamPAL002	Contagem_de_veiculos	Creative commons CCZero	Câmara Municipal de Lisboa	Request HTTP	URL: http://opendata-cml.qart.pt/lastmeasurement/CT0VTH0076
http://example.com/streamPAL002	Contagem_de_veiculos	Creative commons CCZero	Câmara Municipal de Lisboa	Request HTTP	URL: http://opendata-cml.qart.pt/lastmeasurement/CT0VTH1027
http://example.com/streamPAL002	Contagem_de_veiculos	Creative commons CCZero	Câmara Municipal de Lisboa	Request HTTP	URL: http://opendata-cml.qart.pt/lastmeasurement/CT0VTH1079
http://example.com/streamPAL009	Umidade_relativa	Creative commons CCZero	Câmara Municipal de Lisboa	Request HTTP	URL: http://opendata-cml.qart.pt/lastmeasurement/ME00HR0001
http://example.com/streamPAL009	Umidade_relativa	Creative commons CCZero	Câmara Municipal de Lisboa	Request HTTP	URL: http://opendata-cml.qart.pt/lastmeasurement/ME00HR0002

Fonte: elaborado pelo autor

Esse tipo de informação possibilita aos usuários determinar se os dados podem ser reutilizados por outras aplicações ou serviços, além de permitir avaliar a qualidade dos dados por meio das informações sobre a proveniência. Outros detalhes relacionados à aquisição e ao tratamento dos dados também podem ser obtidos, como a estratégia de limpeza aplicada aos dados. Portanto, identificar o tipo de informação disponível no grafo de conhecimento permite avaliar o potencial uso do grafo em diferentes domínios.

A segunda consulta visa identificar as ontologias que foram empregadas na anotação semântica dos dados e observações dos sensores. A consulta 3 obtém essa informação.

Código-fonte 3 – Consulta *Sparql* - Ontologias

```
1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX mds: <https://www.ufc.mdcc.metragrafoiot/>
```

```

3 PREFIX dce: <http://purl.org/dc/elements/1.1/>
4 PREFIX omv: <http://omv.ontoware.org/ontology#>
5
6 SELECT distinct ?nameOntology ?doc ?uri ?acronym ?description WHERE{
7     ?annotation rdf:type mds:SensorAnnotation;
8         mds:hasOntology ?ontology.
9     ?ontology omv:name ?nameOntology;
10        omv:documentation ?doc;
11        omv:URI ?uri;
12        omv:acronym ?acronym;
13        dce:description ?description.
14 }

```

Nessa consulta, a abreviação de prefixos (PREFIX) facilita a compreensão do código *SPARQL*. O uso da cláusula `mds:hasOntology` possibilita a extração das ontologias empregadas na anotação semântica dos sensores. A resposta a essa consulta fornece informações cruciais sobre as estruturas semânticas que enriquecem os dados, permitindo uma compreensão mais profunda das relações e dos significados presentes no grafo de conhecimento.

A Figura 37 apresenta o resultado gerado pela consulta. Nesse conjunto de saída, é possível observar as principais ontologias empregadas durante o processo de anotação semântica dos dados dos sensores e das observações. Duas ontologias cruciais se destacam nesse contexto: a ontologia *IoT-Lite* e a ontologia *SOSA*.

Além das próprias ontologias, outros detalhes relevantes também são disponibilizados para visualização. Isso inclui informações como o endereço da documentação da ontologia, a *URI* (Identificador Uniforme de Recurso) e o acrônimo associado a cada ontologia mencionada. Esses detalhes adicionais têm um papel significativo em permitir que os usuários conheçam e compreendam plenamente o vocabulário semântico aplicado aos dados.

Através desses detalhes, os usuários podem explorar as ontologias utilizadas para enriquecer os dados, entendendo os conceitos, termos e relações que são empregados para representar os diferentes aspectos dos sensores e das observações. Isso proporciona uma visão mais aprofundada e esclarecedora sobre a estrutura semântica subjacente aos dados presentes no grafo de conhecimento.

Compreender o vocabulário semântico aplicado aos dados é uma tarefa fundamental para otimizar sua utilização e aproveitamento. No trabalho de (DIBOWSKI; SCHMID, 2021), é enfatizada a importância de atribuir e compreender a semântica aplicada aos dados, possibilitando que novas aplicações possam se beneficiar plenamente dessas informações.

Figura 37 – Parte do resultado obtido pela consulta 3

	nameOntology	doc	uri	acronym	description
1	"SOSA"	"https://www.w3.org/TR/vocab-ssn/"	"http://www.w3.org/ns/sosa/"	"sosa"	"Ontologia para anotação de dados de sensores, atuadores e observações"
2	"IoT-Lite"	"https://www.w3.org/Submission/2015/SUBM-iot-lite-20151126/"	"http://purl.oclc.org/NET/UNIS-fiware/iot-lite#"	"iot-lite"	"A ontologia IoT-Lite é uma ontologia leve para representar recursos, entidades e serviços da Internet das Coisas (IoT)"

Fonte: elaborado pelo autor

As próximas consultas buscam identificar as classes e propriedades aplicadas sobre os dados do grafo de conhecimento. Essas consultas formam uma abordagem abrangente para identificar as classes e as relações aplicadas aos dados, permitindo uma representação adequada e precisa dos elementos no grafo de conhecimento. As consultas 4, 5 e 6 fornecem uma visão completa das estruturas semânticas presentes nos dados.

Código-fonte 4 – Consulta *Sparql* - Classes

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX mds: <https://www.ufc.mdcc.metragrafoiot/>
3 PREFIX omv: <http://omv.ontoware.org/ontology#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?nameMap ?nameClass WHERE{
7     ?annotation rdf:type mds:SensorAnnotation;
8         mds:hasMapping ?mapping.
9     ?mapping mds:hasClass ?class;
10        rdfs:label ?nameMap.
11        ?class omv:name ?nameClass.
12 }
```

Código-fonte 5 – Consulta *Sparql* - objectProperty

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX mds: <https://www.ufc.mdcc.metragrafoiot/>
3 PREFIX omv: <http://omv.ontoware.org/ontology#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?nameMap ?nameobjectProperty WHERE{
7     ?annotation rdf:type mds:SensorAnnotation;
8         mds:hasMapping ?mapping.
9     ?mapping mds:hasObjectProperty ?objectProperty;
10        rdfs:label ?nameMap.
11        ?objectProperty omv:name ?nameobjectProperty.
12 }
```


Código-fonte 6 – Consulta *Sparql* - *dataProperty*

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX mds: <https://www.ufc.mdcc.metragrafoiot/>
3 PREFIX omv: <http://omv.ontoware.org/ontology#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?nameMap ?namedataProperty WHERE{
7     ?annotation rdf:type mds:SensorAnnotation ;
8         mds:hasMapping ?mapping .
9     ?mapping mds:hasDataProperty ?dataProperty ;
10        rdfs:label ?nameMap .
11        ?dataProperty omv:name ?namedataProperty .
12 }

```

A Figura 38 ilustra os resultados obtidos. As consultas que descrevem as classes e propriedades empregadas no mapeamento dos dados são valiosas, uma vez que auxiliam os usuários a restringirem suas buscas por esses termos nas ontologias mencionadas na consulta 3. Ao possuir essas informações, os usuários podem compreender a maneira como as relações foram estabelecidas entre os dados.

Figura 38 – Resultados obtidos das consultas 4, 5 e 6

I) Classes

nameMap	nameClass
Mapeamento semântico de dados de sensores	Sensor
Mapeamento semântico de dados de sensores	Unit
Mapeamento semântico de dados de sensores	QuantityKind
Mapeamento semântico de dados de sensores	Service
Mapeamento semântico de dados de sensores	Station
Mapeamento semântico de dados de sensores	Address
Mapeamento semântico de dados de sensores	Point
Mapeamento semântico de Observações de sensores	Sensor
Mapeamento semântico de Observações de sensores	Observation

III) DataProperty

nameMap	namedataProperty
Mapeamento semântico de dados de sensores	id
Mapeamento semântico de dados de sensores	measurementType
Mapeamento semântico de dados de sensores	updateFrequency
Mapeamento semântico de dados de sensores	endpoint
Mapeamento semântico de dados de sensores	label
Mapeamento semântico de dados de sensores	lat
Mapeamento semântico de dados de sensores	long
Mapeamento semântico de Observações de sensores	hasSimpleResult
Mapeamento semântico de Observações de sensores	hasResultTime

II) ObjectProperty

nameMap	nameobjectProperty
Mapeamento semântico de dados de sensores	hasUnit
Mapeamento semântico de dados de sensores	hasQuantityKind
Mapeamento semântico de dados de sensores	exposedBy
Mapeamento semântico de dados de sensores	belongsStation
Mapeamento semântico de dados de sensores	hasAddress
Mapeamento semântico de dados de sensores	location
Mapeamento semântico de Observações de sensores	madeBySensor

I - Resultado da consulta referente à 4ª QC para apresentação das Classes utilizadas no mapeamento dos dados.

II - Resultado da consulta referente à 4ª QC para apresentação das relações do tipo ObjectProperty utilizadas no mapeamento dos dados.

III - Resultado da consulta referente à 4ª QC para apresentação das relações do tipo DataProperty utilizadas no mapeamento dos dados.

Fonte: elaborado pelo autor

Compreender as classes e relações aplicadas aos dados desempenha um papel fundamental ao auxiliar os usuários na formulação de consultas a serem executadas no grafo de

conhecimento. Essa compreensão permite não apenas a exploração mais eficaz dos dados, mas também a identificação de como o contexto das informações foi descrito por meio de conceitos semânticos.

A Consulta 7 tem como objetivo fornecer informações relacionadas às entidades de armazenamento dos dados. Os resultados provenientes dessa consulta oferecem detalhes valiosos sobre o *triplestore* adotado e informações específicas acerca dos repositórios de dados. Na Figura 39, é possível observar informações como o nome do *triplestore* e a *URL* que direciona para sua localização de acesso. Além disso, os nomes dos repositórios de dados são apresentados junto aos seus *endpoints*, proporcionando aos usuários uma clara indicação de como acessar os repositórios que abrigam os dados do grafo de conhecimento.

Essa representação gráfica ilustrada na Figura 39 fornece um instantâneo prático e tangível das estruturas de armazenamento associadas ao grafo de conhecimento. Essas informações são cruciais para facilitar o acesso e a utilização eficaz dos dados, permitindo que os usuários explorem e consultem o grafo com maior confiança e compreensão.

Código-fonte 7 – Consulta *Sparql* - armazenamento dos dados

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX mds: <https://www.ufc.mdcc.metragrafoiot/>
3 PREFIX sd: <https://www.w3.org/TR/sparql11-service-description/>
4 PREFIX dce: <http://purl.org/dc/elements/1.1/>
5 PREFIX dcat:<http://www.w3.org/ns/dcat/>
6
7 SELECT ?title ?access ?language ?repository ?endpoint WHERE{
8     ?tripleStore rdf:type mds:TripleStore;
9         dce:title ?title;
10        dcat:accessURL ?access;
11        sd:supportedLanguage ?language;
12        dcat:service ?service.
13    ?service mds:nameRepository ?repository;
14        sd:endpoint ?endpoint.
15 }

```

A Consulta 8 busca reunir informações relacionadas às aplicações e serviços que fazem uso do grafo de conhecimento. Esta consulta provê informações detalhadas sobre uma aplicação específica, como por exemplo a *appMonitoraLisboa*, que foi apresentada na seção 4.3.3. Além de identificar a aplicação, a consulta também lista os nomes dos eventos que essa aplicação definiu e utiliza.

Figura 39 – Resultados obtidos na consulta 7

	title	access	language	repository	endpoint
1	"GraphDB Free"	"https://graphdb.ontotext.com/"	"RDF, TSV, CSV, Excel (.xls and .xlsx), JSON, XML, or Google Sheet"	"Metadata_sensor"	"http://172.18.0.1:7200/repositories/Metadata_sensor"
2	"GraphDB Free"	"https://graphdb.ontotext.com/"	"RDF, TSV, CSV, Excel (.xls and .xlsx), JSON, XML, or Google Sheet"	"Observation_sensor"	"http://172.18.0.1:7200/repositories/Observation_sensor"
3	"GraphDB Free"	"https://graphdb.ontotext.com/"	"RDF, TSV, CSV, Excel (.xls and .xlsx), JSON, XML, or Google Sheet"	"IoTStream"	"http://172.18.0.1:7200/repositories/IoTStream"

Fonte: elaborado pelo autor

Código-fonte 8 – Consulta *Sparql* - Aplicações

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX mds: <https://www.ufc.mdcc.metragrafoiot/>
3 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4
5 SELECT ?nameApp ?nameEvent WHERE{
6     ?a rdf:type mds:EventDetectionApplication;
7     rdfs:label ?nameApp;
8     mds:hasEvent ?event.
9     ?event rdfs:label ?nameEvent.
10 }

```

Por outro lado, a Consulta 9 oferece uma perspectiva ampliada ao apresentar os tipos de fluxos de dados que são consumidos pela aplicação. Essa consulta proporciona uma visão mais abrangente sobre como os dados são utilizados na aplicação, permitindo uma compreensão mais profunda de sua finalidade e funcionalidade.

Código-fonte 9 – Consulta *Sparql* - fluxos de dados

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX mds: <https://www.ufc.mdcc.metragrafoiot/>
3 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4 PREFIX iots: <http://purl.org/iot/ontology/iot-stream#>
5 SELECT ?nameApp ?nameIoTStream ?method WHERE{
6     ?a rdf:type mds:EventDetectionApplication;
7     rdfs:label ?nameApp;
8     mds:consumeIoTStream ?iotStream.
9     ?iotStream rdfs:label ?nameIoTStream;
10     iots:method ?method.
11 }

```

Essas duas consultas, em conjunto, proporcionam informações cruciais sobre os consumidores do grafo de conhecimento. Elas revelam as aplicações específicas que fazem uso dos dados e como esses dados são incorporados em fluxos de informações para atender a diferentes

propósitos. Essa compreensão contribui para uma visão mais abrangente e esclarecedora das interações entre o grafo de conhecimento e as aplicações que o utilizam.

Figura 40 – Resultados obtidos para responder a consulta 8

	nameApp	nameEvent
1	"Aplicação para detecção de eventos"	"Chuva forte"
2	"Aplicação para detecção de eventos"	"Umidade baixa"
3	"Aplicação para detecção de eventos"	"Vento forte"
4	"Aplicação para detecção de eventos"	"Qualidade do ar prejudicada por nível elevado de partículas PM25"
5	"Aplicação para detecção de eventos"	"Qualidade do ar prejudicada por nível elevado de partículas PM10"
6	"Aplicação para detecção de eventos"	"Qualidade do ar prejudicada por nível elevado de dióxido de azoto(NO2)"
7	"Aplicação para detecção de eventos"	"Qualidade do ar prejudicada por nível elevado de ozonio (O3)"
8	"Aplicação para detecção de eventos"	"Qualidade do ar prejudicada por nível elevado de dióxido de enxofre (SO2)"
9	"Aplicação para detecção de eventos"	"Temperatura Elevada"

Fonte: elaborado pelo autor

Figura 41 – Resultados obtidos para responder a consulta 9

	nameApp	nameIotStream	method
1	"Aplicação para detecção de eventos"	"Temperatura"	"AVG"
2	"Aplicação para detecção de eventos"	"Umidade_relativa"	"AVG"
3	"Aplicação para detecção de eventos"	"Intensidade_do_vento"	"AVG"
4	"Aplicação para detecção de eventos"	"PM10"	"AVG"
5	"Aplicação para detecção de eventos"	"PM25"	"AVG"
6	"Aplicação para detecção de eventos"	"Enxofre"	"AVG"
7	"Aplicação para detecção de eventos"	"Dioxido_de_azoto"	"AVG"
8	"Aplicação para detecção de eventos"	"Ozonio"	"AVG"

Fonte: elaborado pelo autor

A Figura 40 exibe os resultados obtidos da consulta 8. Por meio dessa consulta, é possível visualizar o nome de uma aplicação voltada à detecção de eventos, além dos nomes dos eventos que essa aplicação registra. Essa representação visual revela as interações entre a aplicação e os tipos de eventos que ela está preparada para identificar.

Por outro lado, a Figura 41 apresenta os resultados provenientes da consulta 9. Nessa visualização, são fornecidos os nomes dos fluxos de dados que são consumidos pela aplicação de detecção de eventos, juntamente com o método empregado para a geração desses fluxos, conhecido como *iotStream*.

As informações relativas aos consumidores, eventos e fluxos de dados assumem uma

importância significativa, contribuindo para a avaliação do escopo de conhecimento abrangido pelo grafo. Além disso, esses detalhes são fundamentais para a manutenção contínua do próprio grafo de conhecimento.

Compreender os fluxos de dados consumidos pelas aplicações capacita os usuários a identificarem quais aplicações seriam impactadas no caso de uma interrupção na obtenção de dados de sensores específicos. Essa compreensão é crucial tanto para a administração do grafo de conhecimento quanto para a verificação e validação do funcionamento das aplicações e serviços. Se uma aplicação depende de um “fluxo de dados X” e esse fluxo torna-se indisponível, a aplicação pode ser afetada em sua execução. Isso destaca a importância de entender as relações entre as aplicações e os fluxos de dados, garantindo o bom desempenho do sistema em sua totalidade.

7 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho, abordou-se uma série de desafios relacionados à criação de Grafos de Conhecimento Semântico (GCS) para aplicações que demandam a integração de dados provenientes de dispositivos *IoT* e fontes heterogêneas de informações.

A fim de nortear o desenvolvimento da pesquisa definiu-se três questões de pesquisa, as quais buscou-se responder no decorrer do estudo. A primeira questão de pesquisa indagou como é possível desenvolver um grafo de conhecimento de dados *IoT* que englobe diversos níveis de dados. Para responder a essa questão, realizou-se uma busca na literatura para identificar as diferentes ontologias que vinham sendo utilizadas na anotação semântica de dados de sensores e como estas poderiam ser relacionadas. Definiu-se a abordagem *DIKW4IoT* para a construção do grafo de conhecimento de dados *IoT*. Esta abordagem utiliza como base a hierarquia *DIKW* para estabelecer e representar os diferentes níveis de dados presentes no grafo de conhecimento. Para cada nível de dados a abordagem *DIKW4IoT* estabelece o uso de uma ontologia específica para anotação dos dados.

A segunda questão de pesquisa buscou investigar que estratégias poderiam ser empregadas para integrar um grafo de conhecimento de dados *IoT* com fontes de dados heterogêneas e externas. A partir da análise dos trabalhos relacionados e a partir da análise da natureza dos dados de sensores, definiu-se o *SIGIoT-S*, um *framework* conceitual que orienta o processo de integração do grafo de conhecimento de dados *IoT* com outras fontes de dados. O *SIGIoT-S* estabelece duas etapas principais, a construção do grafo de conhecimento de dados *IoT* e o processo de integração propriamente dito. Definiu-se que o grafo de conhecimento de dados *IoT* é composto pela união de dois subgrafos: O *GC-IoT-Live* (dados de sensores dinâmicos) e o *GC-IoT-E* (dados de sensores estáveis). O processo de integração com outras fontes de dados heterogêneas e externas se dá por meio da criação de *links* entre o *GC-IoT-E* e as outras fontes de dados.

A terceira questão de pesquisa buscou analisar formas para descrever um grafo de conhecimento de dados *IoT* de forma que este se torne acessível e compreensível aos usuários. Para responder esta questão de pesquisa, viu-se a importância dos metadados e que estes são capazes de descrever tanto os dados como os passos e ferramentas utilizadas na construção do grafo de conhecimento. Propôs-se então a ontologia *MetaGraphIoT*. Esta ontologia tem como objetivo descrever o grafo de conhecimento de dados *IoT* por meio dos metadados.

Cada resultado proposto foi exemplificado por meio de estudos de caso, demons-

trando a aplicação eficaz das soluções propostas. Esses estudos de caso validaram a replicabilidade e eficácia das soluções, fornecendo uma base sólida para futuras pesquisas e aplicações nesta área em constante evolução.

Uma questão de grande relevância que não foi abordada na dissertação é a manutenção contínua do GCS. Isso se torna particularmente desafiador devido à natureza dinâmica dos dados envolvidos. As fontes de dados utilizadas para construir o Grafo de Conhecimento Semântico estão em constante evolução: novas entidades podem surgir, entidades existentes podem ser excluídas e informações sobre essas entidades podem ser atualizadas em frequências variadas. Portanto, a manutenção eficaz do GCS é uma tarefa complexa que exige a implementação de estratégias robustas para acompanhar e atualizar continuamente o grafo à medida que novos dados são disponibilizados e as condições do domínio evoluem.

Outro ponto importante é o desenvolvimento e manutenção de aplicações que fazem uso do GCS. Em geral, a maior parcela de esforço para desenvolvimento de aplicações especializadas é direcionada à construção do GCS, também conhecido como “mashup de dados”, cujo propósito principal é capturar e organizar o conhecimento especializado relacionado ao domínio da aplicação.

Muitas empresas e organizações estão investindo na construção de um Grafo de Conhecimento Global (GC Global) com o objetivo de integrar de forma contínua todos os dados relevantes para o domínio da organização. O GC Global desempenha o papel central como um *hub* que abrange dados, metadados e conteúdo, proporcionando uma visão unificada, constantemente atualizada, consistente e livre de ambiguidades dos dados. Essa visão é compartilhada com todas as aplicações da organização.

Através do reuso do conhecimento armazenado no GC Global, a construção e manutenção de *mashups* especializados para aplicações de descoberta de conhecimento torna-se significativamente simplificado. Além disso, essa abordagem garante que os dados utilizados sejam contextualmente relevantes, coesos e perfeitamente alinhados com o domínio da aplicação. Como resultado, a eficácia das análises e operações executadas pela aplicação é aprimorada, permitindo que ela descubra conhecimentos valiosos e tome decisões informadas com base em dados confiáveis e substanciais.

Como perspectivas para o futuro, tem-se a intenção de investir em pesquisas e desenvolvimento de métodos, técnicas e ferramentas destinados a aprimorar a construção e a manutenção de GC Globais que lidam com integração de dados em larga escala. Adicional-

mente, pretende-se elaborar mecanismos que simplifiquem o desenvolvimento e manutenção de aplicações inteligentes que consomem dados de um GC Global.

Essas aplicações podem gerar novos dados e metadados, os quais devem ser cuidadosamente incorporados ao GC Global. Essa integração pode incluir a criação de novas entidades, a adição de relações ou a expansão das ontologias existentes para acomodar os dados recém-gerados. Portanto, é fundamental oferecer suporte a todas as etapas do ciclo de evolução de um GC global em conjunto com aplicações de descoberta de conhecimento.

REFERÊNCIAS

- ALVAREZ-COELLO, D.; GÓMEZ, J. M. Ontology-based integration of vehicle-related data. In: IEEE. **2021 IEEE 15th International Conference on Semantic Computing (ICSC)**. [S. l.], 2021. p. 437–442.
- ALVAREZ-COELLO, D.; KLOTZ, B.; WILMS, D.; FEJJI, S.; GÓMEZ, J. M.; TRONCY, R. Modeling dangerous driving events based on in-vehicle data using random forest and recurrent neural network. In: IEEE. **2019 IEEE Intelligent Vehicles Symposium (IV)**. [S. l.], 2019. p. 165–170.
- ALVAREZ-COELLO, D.; WILMS, D.; BEKAN, A.; GÓMEZ, J. M. Generic semantization of vehicle data streams. In: IEEE. **2021 IEEE 15th International Conference on Semantic Computing (ICSC)**. [S. l.], 2021. p. 112–117.
- ARRUDA, N.; VENCESLAU, A.; CRUZ, M. M. Lima da; VIDAL, V. M.; PEQUENO, V. M. Publishing and consuming semantic views for construction of knowledge graphs. In: SCITEPRESS DIGITAL LIBRARY. **22nd International Conference on Enterprise Information Systems (ICEIS)**. [S. l.], 2020. p. 197–204.
- BERMUDEZ-EDO, M.; ELSALEH, T.; BARNAGHI, P.; TAYLOR, K. Iot-lite: a lightweight semantic model for the internet of things and its use with dynamic semantics. **Personal and Ubiquitous Computing**, Springer, v. 21, n. 3, p. 475–487, 2017.
- CHAE, B. K. The evolution of the internet of things (iot): A computational text analysis. **Telecommunications Policy**, v. 43, n. 10, p. 101848, 2019. ISSN 0308-5961. ITS Seoul 2018.
- DIBOWSKI, H.; SCHMID, S. Using knowledge graphs to manage a data lake. **INFORMATIK 2020**, Gesellschaft für Informatik, Bonn, 2021.
- DING, L.; XIAO, G.; CALVANESE, D.; MENG, L. A framework uniting ontology-based geodata integration and geovisual analytics. **ISPRS International Journal of Geo-Information**, Multidisciplinary Digital Publishing Institute, v. 9, n. 8, p. 474, 2020.
- DJEZZAR, M.; HEMAM, M.; MAIMOUR, M.; AMARA, F. Z.; FALEK, K.; SEGHIR, Z. A. An approach for semantic enrichment of sensor data. In: IEEE. **2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS)**. [S. l.], 2018. p. 1–7.
- DUAN, Y.; SHAO, L.; HU, G.; ZHOU, Z.; ZOU, Q.; LIN, Z. Specifying architecture of knowledge graph with data graph, information graph, knowledge graph and wisdom graph. In: IEEE. **2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)**. [S. l.], 2017. p. 327–332.
- ELSALEH, T.; ENSHAEIFAR, S.; REZVANI, R.; ACTON, S. T.; JANEIKO, V.; BERMUDEZ-EDO, M. Iot-stream: A lightweight ontology for internet of things data streams and its use with data analytics and event detection services. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 20, n. 4, p. 953, 2020.
- GAO, H.; DUAN, Y.; SHAO, L.; SUN, X. Transformation-based processing of typed resources for multimedia sources in the iot environment. **Wireless Networks**, Springer, v. 27, n. 5, p. 3377–3393, 2021.

- HAJLAOUI, E.; ZAIER, A.; KHLIFI, A.; GHODHBANE, J.; HAMED, M. B.; SBITA, L. 4g and 5g technologies: A comparative study. In: **2020 5th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)**. [S. l.: s. n.], 2020. p. 1–6.
- HALLER, A.; JANOWICZ, K.; COX, S. J.; LEFRANÇOIS, M.; TAYLOR, K.; PHUOC, D. L.; LIEBERMAN, J.; GARCÍA-CASTRO, R.; ATKINSON, R.; STADLER, C. The modular ssn ontology: A joint w3c and ogc standard specifying the semantics of sensors, observations, sampling, and actuation. **Semantic Web**, IOS Press, v. 10, n. 1, p. 9–32, 2019.
- ISOTANI, S.; BITTENCOURT, I. I. **Dados abertos conectados: em busca da web do conhecimento**. [S. l.]: Novatec Editora, 2015.
- JANOWICZ, K.; HALLER, A.; COX, S. J.; PHUOC, D. L.; LEFRANÇOIS, M. Sosa: A lightweight ontology for sensors, observations, samples, and actuators. **Journal of Web Semantics**, Elsevier, v. 56, p. 1–10, 2019.
- KELLER, R. M. Building a knowledge graph for the air traffic management community. In: **Companion Proceedings of The 2019 World Wide Web Conference**. [S. l.: s. n.], 2019. p. 700–704.
- KIM, S.; JEON, T. H.; RHIU, I.; AHN, J.; IM, D.-H. Semantic scene graph generation using rdf model and deep learning. **Applied Sciences**, Multidisciplinary Digital Publishing Institute, v. 11, n. 2, p. 826, 2021.
- LE-PHUOC, D.; QUOC, H. N. M.; QUOC, H. N.; NHAT, T. T.; HAUSWIRTH, M. The graph of things: A step towards the live knowledge graph of connected things. **Journal of Web Semantics**, Elsevier, v. 37, p. 25–35, 2016.
- LIEW, A. Dikiw: Data, information, knowledge, intelligence, wisdom and their interrelationships. **Business Management Dynamics**, Society of Business and Management Dynamics (SBMD), v. 2, n. 10, p. 49, 2013.
- LIMA, J. C. de; CARVALHO, C. L. de. Ontologias-owl (web ontology language). **Relatório técnico, Universidade Federal de Goiás**, 2005.
- METER, H. J. V. Revising the dikw pyramid and the real relationship between data, information, knowledge, and wisdom. **Law, Technology and Humans**, v. 2, n. 2, p. 69–80, 2020.
- NOY, N.; MCGUINNESS, D. L. *et al.* Ontology development 101. **Knowledge Systems Laboratory, Stanford University**, v. 2001, 2001.
- REMOR, C. A.; FIALHO, F. A.; QUEIROZ, M. P. Analisando a hierarquia dikw. In: **Anais do Congresso Internacional de Conhecimento e Inovação–ciki**. [S. l.: s. n.], 2017. v. 1, n. 1.
- REN, Y.; PARVIZI, A.; MELLISH, C.; PAN, J. Z.; DEEMTER, K. v.; STEVENS, R. Towards competency question-driven ontology authoring. In: SPRINGER. **European Semantic Web Conference**. [S. l.], 2014. p. 752–767.
- ROBERTO, L. **A pirâmide do conhecimento**. 2021. Artigo disponível em <https://professorluizroberto.com/a-piramide-do-conhecimento/>. Acesso dia: 07/04/2022.
- ROLIM, T. V.; AVILA, C. V. S.; MARIANO, R. G.; CALIXTO, T.; IVO, P. Uso das tecnologias da web semântica na construção de grafos de conhecimento semântico baseado no enfoque híbrido. 2021.

SATIJA, M.; BAGCHI, M.; MARTÍNEZ-ÁVILA, D. Metadata management and application. **Library Herald**, v. 58, p. 84–107, 2020.

SATIJA, M.; BAGCHI, M.; MARTÍNEZ-ÁVILA, D. Metadata management and application. **Library Herald**, v. 58, p. 84–107, 12 2020.

SOARES, A.; DIVIDINO, R.; ABREU, F.; BROUSSEAU, M.; ISEÑOR, A. W.; WEBB, S.; MATWIN, S. Crisis: Integrating ais and ocean data streams using semantic web standards for event detection. In: IEEE. **2019 International conference on military communications and information systems (ICMCIS)**. [S. l.], 2019. p. 1–7.

STEINDL, G.; HEINZL, B.; KASTNER, W.; MAURI, J. A novel ontology-based smart service architecture for data-driven model development. In: **eKNOW 2019-The Eleventh International Conference on Information, Process, and Knowledge Management**. [S. l.: s. n.], 2019. p. 26–27.

SZILAGYI, I.; WIRA, P. Ontologies and semantic web for the internet of things-a survey. In: IEEE. **IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society**. [S. l.], 2016. p. 6949–6954.

TOMMASINI, R.; SEDIRA, Y. A.; DELL'AGLIO, D.; BALDUINI, M.; ALI, M. I.; PHUOC, D. L.; VALLE, E. D.; CALBIMONTE, J.-P. Vocals: Vocabulary and catalog of linked streams. In: SPRINGER. **International Semantic Web Conference**. [S. l.], 2018. p. 256–272.