

UNIVERSIDADE FEDERAL DE PELOTAS
Centro de Desenvolvimento Tecnológico
Programa de Pós-Graduação em Computação



Dissertação

Pensamento Computacional como uma Aplicação em Dados Abertos
Conectados

Ândrea Völz Garcez

Pelotas, 2022

Ândrea Völz Garcez

**Pensamento Computacional como uma Aplicação em Dados Abertos
Conectados**

Dissertação apresentada ao Programa de Pós-Graduação em Computação do Centro de Desenvolvimento Tecnológico da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Mestre em Ciência da Computação.

Orientadora: Profa. Dra. Ana Marilza Pernas Fleischmann

Pelotas, 2022

Universidade Federal de Pelotas / Sistema de Bibliotecas
Catalogação na Publicação

G215p Garcez, Ândrea Völz

Pensamento computacional como uma aplicação em dados abertos conectados / Ândrea Völz Garcez ; Ana Marilza Pernas Fleischmann, orientadora. — Pelotas, 2022.

52 f. : il.

Dissertação (Mestrado) — Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, 2022.

1. Ontologia. 2. Pensamento computacional. 3. Dados abertos conectados. 4. Padronização. I. Fleischmann, Ana Marilza Pernas, orient. II. Título.

CDD : 005

Ândrea Völz Garcez

**Pensamento Computacional como uma Aplicação em Dados Abertos
Conectados**

Dissertação aprovada, como requisito parcial, para obtenção do grau de Mestre em Ciência da Computação, Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas.

Data da Defesa: 07 de abril de 2022

Banca Examinadora:

Profa. Dra. Ana Marilza Pernas Fleischmann (orientadora)
Doutora em Computação pela Universidade Federal do Rio Grande do Sul.

Profa. Dra. Larissa Astrogildo De Freitas
Doutora em Computação pela Pontifícia Universidade Católica do Rio Grande do Sul.

Prof. Dr. Marilton Sanhotene De Aguiar
Doutor em Computação pela Universidade Federal do Rio Grande do Sul.

Profa. Dra. Renata Hax Sander Reiser
Doutora em Computação pela Universidade Federal do Rio Grande do Sul.

RESUMO

GARCEZ, Ândrea Völz. **Pensamento Computacional como uma Aplicação em Dados Abertos Conectados**. Orientadora: Ana Marilza Pernas Fleischmann. 2022. 52 f. Dissertação (Mestrado em Ciência da Computação) – Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2022.

Dados estão cada vez mais presentes em nosso dia a dia e podem ser representados por diversas formas e fontes. Utilizando tecnologias abertas é possível que sistemas sejam desenvolvidos mais rapidamente com a colaboração de qualquer pessoa que detenha o conhecimento necessário.

Dados abertos podem ser reutilizados e distribuídos por qualquer pessoa, mas se avaliados de forma isolada podem não levar a derivação de informações relevantes. Visto isso, torna-se importante o uso de dados abertos conectados, pois permitem cruzar dados de diversas fontes, processando esses dados para geração de novas informações e conhecimentos. Essas características fazem com que sejam uma solução para problemas derivados da falta de padronização e interligação de dados.

O Pensamento Computacional (PC) é visto como um conhecimento introduzido a partir da relação entre conceitos de computação, sendo aplicado na resolução de problemas. O PC é trabalhado em diversos níveis de ensino por profissionais de diferentes áreas, que buscam trazer conceitos e habilidades relevantes ao desenvolvimento do PC à educação desde os primeiros anos de ensino. Entretanto, seu uso resulta em divergências quanto a padronização de tarefas, conceitos e habilidades do PC.

Desta forma, neste trabalho é proposta a criação e a disponibilização de um modelo capaz de capturar e integrar múltiplas perspectivas de autores sobre PC, afim de prover uma forma de padronizar as diversas definições existentes. Neste esforço, propõe-se uma ontologia de domínio, expansível para comportar a definição de diversos autores, e ontologias de topo, responsáveis por representar as visões de autores com perspectivas distintas. Com isso, busca-se demonstrar a expressividade e o alcance de operação da ontologia de domínio desenvolvida.

O modelo desenvolvido foi testado em três cenários típicos do PC, mostrando-se capaz de modelar e integrar pontos de vista de diferentes autores e de vincular recursos de diferentes naturezas de acordo com suas finalidades e assuntos.

Com o uso de dados abertos conectados foi possível disponibilizar o modelo criado em um *endpoint*, para que outros usuários e aplicações possam estender as definições propostas com definições de outros autores ou pesquisar recursos alternativos que possam ser utilizados para o ensino de determinada habilidade.

Palavras-chave: Ontologia. Pensamento computacional. Dados abertos conectados. Padronização.

ABSTRACT

GARCEZ, Ândrea Völz. **Study on Computational Thinking as an Application in Open Linked Data**. Advisor: Ana Marilza Pernas Fleischmann. 2022. 52 f. Dissertation (Masters in Computer Science) – Technology Development Center, Federal University of Pelotas, Pelotas, 2022.

Data are increasingly present in our daily lives and can be represented in different ways and sources. Using open technologies it is possible for systems to be developed more quickly with the collaboration of anyone who has the necessary knowledge.

Open data can be reused and distributed by anyone, but if evaluated in isolation it may not lead to the derivation of relevant information. In view of this, the use of connected open data becomes important, as they allow the crossing of data from different sources, processing this data to generate new information and knowledge. These characteristics make them a solution to problems arising from the lack of standardization and data interconnection.

Computational Thinking (CP) is seen as knowledge introduced from the relationship between computing concepts, being applied in problem solving. The PC is worked at different levels of education by professionals from different areas, who seek to bring concepts and skills relevant to the development of the PC to education from the early years of teaching. However, its use results in divergences regarding the standardization of PC tasks, concepts and skills.

Thus, this work proposes the creation and availability of a model capable of capturing and integrating multiple perspectives of authors on CP, in order to provide a way to standardize the various existing definitions. In this effort, we propose a domain ontology, expandable to accommodate the definition of several authors, and top ontologies, responsible for representing the views of authors with different perspectives. With this, we seek to demonstrate the expressiveness and scope of operation of the domain ontology developed.

The developed model was tested in three typical PC scenarios, proving to be capable of modeling and integrating different authors' points of view and linking resources of different natures according to their purposes and subjects.

With the use of connected open data, it was possible to make the created model available on an endpoint, so that other users and applications can extend the proposed definitions with definitions from other authors or search for alternative resources that can be used to teach a certain skill.

Keywords: Ontology. Computational thinking. Linked open data. Standardization.

LISTA DE FIGURAS

Figura 1	Pirâmide de dados	11
Figura 2	Camadas da Web Semântica	17
Figura 3	Ontologias e Web Semântica	18
Figura 4	Representação gráfica do uso de RDF com N-Triplas	18
Figura 5	Formatos de apresentação do modelo RDF	19
Figura 6	Dados conectados segundo as camadas da Web Semântica	21
Figura 7	Esquema de 5 estrelas para dados abertos	21
Figura 8	Arquitetura de Provisionamento de dados do DBpedia	22
Figura 9	Pilares do pensamento computacional	24
Figura 10	Princípios do pensamento computacional	24
Figura 11	Ontologia de topo - Jeannette Wing	33
Figura 12	Ontologia de topo - Karen Brennan	34
Figura 13	Ontologia de topo - Silva Junior	34
Figura 14	Ontologia de domínio	35
Figura 15	Propriedades da ontologia de domínio	36
Figura 16	Camada alvo de aprendizagem	37
Figura 17	Relação extrínseca	38
Figura 18	Relacionamento entre autores	38
Figura 19	Cenários para validação da ontologia de domínio	40
Figura 20	Publicação da ontologia no endpoint	44
Figura 21	Exemplo de consulta SPARQL submetida a ontologia desenvolvida	45
Figura 22	Primeiros registros da consulta SPARQL	46

LISTA DE TABELAS

Tabela 1	Formatos de apresentação RDF e suas características	19
Tabela 2	Relacionamentos (propriedades) da ontologia de domínio	36

LISTA DE ABREVIATURAS E SIGLAS

APIs	<i>Application Programming Interfaces</i>
APP	Aplicativo
AVAs	Ambientes Virtuais de Aprendizagem
CSU	<i>Unplugged Computer Science</i>
CTt	<i>Computational Thinking Test</i>
DMZ	<i>Demilitarized Zone</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IRI	<i>International Resource Identifier</i>
JSON	<i>JavaScript Object Notation</i>
LOD	<i>Linked Open Data</i>
LOD Cloud	<i>Linked Open Data Cloud</i>
LD	<i>Linked Data</i>
NAT	<i>Network Address Translation</i>
PC	Pensamento Computacional
PDF	<i>Portable Document Format</i>
OF	<i>Open Format</i>
OKBR	<i>Open Knowledge Brasil</i>
OKF	<i>Open Knowledge Foundation</i>
OL	<i>Open License</i>
OLW	<i>Ontology Web Language</i>
RDF	<i>Resource Description Framework</i>
RE	<i>REused</i>
URI	<i>Uniform Resource Identifier</i>
W3C	<i>World Wide Web Consortium</i>
WEB	<i>World Wide Web</i>

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Motivação	14
1.2	Objetivo	14
2	BASE CONCEITUAL PARA DADOS ABERTOS CONECTADOS	15
2.1	Ontologias	15
2.2	Web Semântica	17
2.3	Operacionalização da Web Semântica como Dados Abertos Conectados	20
3	PENSAMENTO COMPUTACIONAL	23
3.1	Conceitos em Pensamento Computacional	25
3.1.1	Conceitualização de acordo com Wing	25
3.1.2	Conceitualização de acordo com Brennan e Resnick	26
3.1.3	Conceitualização de acordo com Silva Junior	27
4	TRABALHOS RELACIONADOS	28
4.1	Ontologias para Pensamento Computacional	28
4.2	Dados Abertos Conectados Voltados à Educação	30
5	CONCEPÇÃO E MODELAGEM	31
5.1	Definição do Escopo e Modelo da Ontologia	32
5.1.1	Ontologias de Topo	32
5.1.2	Ontologia de Domínio	35
5.1.3	Integração entre as Ontologias	37
6	DISPONIBILIZAÇÃO DA ONTOLOGIA DE DOMÍNIO	40
6.1	Cenários de Avaliação	40
6.1.1	Teste de pensamento computacional	41
6.1.2	Ciência da computação desplugada	42
6.1.3	Scratch	43
6.2	Disponibilização da Ontologia e Teste dos Cenários	44
7	CONCLUSÃO	47
	REFERÊNCIAS	49

1 INTRODUÇÃO

Dados são a alma da tomada de decisões e a matéria-prima para a prestação de contas. É quase impossível a concepção, o acompanhamento e a avaliação de políticas eficazes sem dados de alta qualidade que fornecem as informações corretas sobre as coisas certas no momento certo (REVOLUTION, 2014).

Dados estão presentes no nosso cotidiano. Cada vez que alguém usa uma rede social, utiliza o telefone celular ou simplesmente vai ao supermercado comprar um produto que possua código de barras, dados estão sendo gerados e processados.

Eles não fazem sentido a princípio, mas se os dados forem tratados, ou seja, organizados e compreendidos dentro de um contexto, então estão sendo geradas informações. O conhecimento, por sua vez, é a informação tratada e processada de acordo com determinadas experiências e gerando novas informações que não estavam explícitas anteriormente. A Figura 1 mostra a pirâmide de dados mostrando os conceitos discutidos nesse parágrafo.



Figura 1 – Pirâmide de dados

Fonte: (LÍTERIS, 2015)

Os dados podem estar representados de diversas formas e fontes, podendo estas serem: sites web; documentos formato PDF; planilhas eletrônicas; bancos de dados; redes sociais, e é vital tornar possível seu acesso e interoperação para que informação e conhecimento possam ser gerados de forma facilitada. Com essa visão, a adoção de tecnologias abertas permite que sistemas sejam desenvolvidos mais rapidamente, pois qualquer pessoa com conhecimento necessário pode modificar, otimizar ou adicionar novas funções. Essa rapidez é essencial hoje em dia, pois os avanços tecnológicos ocorrem de forma acelerada.

De acordo com James (2013), dados abertos são dados que podem ser usados, reutilizados e redistribuídos por qualquer pessoa. Um exemplo de aplicação de dados abertos é para dados governamentais, os quais consistem de dados gerados pelo governo para promover uma transparência dos gastos públicos.

Uma aplicação desenvolvida com base em dados governamentais é o "Pra onde foi o meu dinheiro?"¹. Segundo Dados abertos (2018), ele demonstra de forma interativa e gráfica as despesas diretas do executivo federal do estado de São Paulo e na esfera federal. E também permite observar a distribuição dos investimentos do governo em áreas como saúde, educação e outros.

A Open Definition², criada pela Open Knowledge, anteriormente chamada de Open Knowledge Foundation³ (OKF) foi quem formalizou o conceito de dados abertos. Segundo informações do site, a OKF é uma rede internacional, com grupos em mais de 40 países, dentre eles, o Brasil. No Brasil, o grupo é chamado de Open Knowledge Brasil⁴ (OKBR), e é uma organização sem fins lucrativos que promove conhecimento livre.

A OKBR possui diversos projetos, dentre eles salienta-se o transparência da Covid-19⁵, o Open Data Index⁶ e a Operação Serenata de Amor⁷. O projeto de transparência da Covid-19 avalia a qualidade dos dados publicados pela União e pelos estados brasileiros relativo ao coronavírus com o objetivo de padronizar a divulgação dos dados. Segundo Brasil (2018), Open Data Index consiste de indicadores que ajudam a avaliar políticas, identificar gargalos e orientar municípios com o intuito de melhorar suas políticas de dados abertos. O projeto Serenata de Amor é uma iniciativa que surgiu para auditar contas públicas.

De acordo com James (2013) a definição de dados abertos tem 3 pilares: disponibilidade de acesso; participação universal e reutilização e redistribuição. A disponibilidade de acesso define que os dados devem estar completos e disponíveis para que

¹<http://www.paraondefoiomeudinheiro.com.br/>

²<https://opendefinition.org>

³<https://okfn.org>

⁴<https://www.ok.org.br>

⁵<https://www.ok.org.br/projetos/indice-de-transparencia-da-covid-19>

⁶<https://www.ok.org.br/projetos/open-data-index>

⁷<https://www.ok.org.br/projetos/serenata-de-amor>

qualquer pessoa possa acessá-los; a participação universal serve para que qualquer pessoa possa usar os dados independentemente do seu campo de atuação; e a reutilização e redistribuição implica que os dados devem ser distribuídos com uma licença que permita que sejam reutilizados e mesclados com outros dados.

Sistemas abertos e dados abertos podem ser grandes aliados na solução de problemas dos cidadãos, mas simplesmente coletar os dados abertos e avaliá-los de forma isolada pode não gerar informações relevantes e é exatamente isso que torna os dados abertos conectados tão importantes, pois ao cruzar dados de diversas fontes e processá-los é possível gerar novas informações e conhecimentos. Eles podem ser usados para criar aplicativos que melhorem a mobilidade urbana, o serviço de transporte público, a gestão de cidades e estados, sistemas de recomendação, marketing, entre inúmeros outros benefícios.

Dois exemplos de APPs que consomem dados abertos conectados para facilitar a vida dos cidadãos são os aplicativos Menor Preço⁸ e o aplicativo Sinesp Cidadão⁹.

O Menor preço é um APP que consome dados abertos conectados para informar aos usuários qual estabelecimento comercial está oferecendo o menor preço para o produto pesquisado. Quando uma pessoa vai ao supermercado ou a algum estabelecimento comercial e opta por colocar o CPF na nota, esse valor vai para a base de dados vinculando o horário, local, nome do produto, código de barras e valor, e então quando outro cidadão entra no APP para buscar por um código de barras, combustível ou nome de produto, então o Menor Preço irá informar os valores pagos possibilitando que a busca seja ordenada por menor valor.

Já o aplicativo Sinesp Cidadão permite que os usuários consultem placas de veículos para ver se estão em situação ilegal, busquem por desaparecidos ou também consultem mandados e procurados.

Tanto nas pesquisas apontadas quanto nas demais pesquisas em dados abertos conectados o ponto chave é definir a forma com que os dados serão modelados e disponibilizados, permitindo seu amplo acesso e integração entre aplicações. Para isso, é importante fazer uso de padrões para modelagem, representação, intercâmbio e toda gerência dos dados. Nesse aspecto as ontologias e a Web Semântica aparecem como área de pesquisa que visa fornecer toda a base para tratamento e apresentação dos dados, com o objetivo de torná-los elementos úteis, compreensíveis e legíveis por máquina, estruturando as informações através de relações semânticas.

Dentro da área de Web Semântica, destaca-se o trabalho Silva et al. (2018), que propõe uma abordagem que tem como objetivo anotar automaticamente entidades nos Currículo Lattes por meio de bases de dados abertos, que armazenam o significado de termos e expressões, criando um sistema Lattes web semântico.

⁸<https://www.rs.gov.br/carta-de-servicos/servicos?servico=1052>

⁹<https://www.justica.gov.br/sua-seguranca/seguranca-publica/sinesp-1/sinesp-Cidadao>

1.1 Motivação

Nos últimos anos, diversas iniciativas têm visado introduzir conceitos relacionados a computação e robótica na educação básica (BLIKSTEIN, 2018). O Pensamento Computacional (PC) é um dos conhecimentos introduzidos através dessas iniciativas, ele usa conceitos de computação para resolução de problemas. Para Denning; Tedre (2019), PC é visto como conjuntos de habilidades que auxiliam no processo de resolução de problemas.

O PC é trabalhado em diversos níveis de ensino por profissionais de diversas áreas, que buscam trazer os conceitos e habilidades identificados como relevantes ao desenvolvimento do PC à educação desde os primeiros anos de ensino. Entretanto, por esta ser uma área abrangente, resulta em uma série de divergências quanto a padronização de tarefas, conceitos e habilidades do PC.

Muitos autores definem de forma diferente os mesmos tópicos, por esse motivo, observou-se no PC uma área oportuna para disponibilização dos dados de forma aberta e conectada. Havendo uma camada semântica de padronização de dados, que pudesse ser acessada pelas diferentes pesquisas em PC, seria possível compartilhamento e reuso de informações.

1.2 Objetivo

Tendo em vista a dificuldade em definir conceitos e padronizar termos e habilidades em PC, uma vez que diferentes autores definem as mesmas habilidades de formas diferentes, e também a dificuldade de reutilização e interoperabilidade de dados, o objetivo deste trabalho é criar uma camada semântica que possa ser usada como ferramenta para padronização dos conceitos de PC, sendo disponibilizada de forma aberta na web para que possa ser amplamente difundida e utilizada.

Essa camada semântica tem como objetivo definir classificações genéricas que organizam recursos e ao mesmo tempo reúne todos os alvos de aprendizagem de PC. Ela permite incorporar a definição dos conceitos de acordo com múltiplos autores ao invés de definir arbitrariamente o que é PC.

Ontologias mostram-se como uma forma de solução para representar conhecimento de forma compartilhada e permite integração e interoperação. Sendo assim, é uma forma de normalizar as visões de diferentes autores permitindo a integração de dados interpretáveis por máquina.

A fim de ilustrar a capacidade de integração da ontologia, são descritos na seção 3.1 as perspectivas de três autores de PC e na seção 6.1 três recursos com o objetivo de mostrar a abrangência da ontologia.

2 BASE CONCEITUAL PARA DADOS ABERTOS CONECTADOS

Neste capítulo serão abordados os principais conceitos relacionados a ontologias e Web Semântica, assuntos tratados como base para representação de dados abertos conectados.

2.1 Ontologias

Segundo o ponto de vista de Guarino (1998), a palavra ontologia escrita com a letra "o" minúscula possui significado diferente do que Ontologia escrita em maiúsculo. De acordo com o autor, se escrita em maiúsculo, ela se refere a uma disciplina filosófica, porém se for escrita em minúsculo, o termo se refere ao sentido usado dentro da comunidade de inteligência artificial.

O termo ontologia, de acordo com Guarino; Oberle; Staab (2009), no contexto de conhecimento, significa a descrição de conceitos e relacionamentos, tendo como propósito o compartilhamento e a reutilização do conhecimento.

Do ponto de vista de Gruber (1993), o termo ontologia representa classes, funções, relações e outros objetos que especificam um vocabulário que representa um domínio compartilhado.

Como ontologias são modelos de dados usados para representar qualquer coisa existente. Dentro da Web Semântica, as ontologias podem interligar conteúdos de diferentes páginas e conectar termos.

De acordo com Isotani; Bittencourt (2015), as ontologias são compostas por um conjunto termos, ou seja, um vocabulário que descreve adequadamente seu conjunto de dados e que representa a expressividade dentro do contexto em que é utilizado, com intuito de reduzir a ambiguidade de sua interpretação através da explicitação das relações entre conceitos que representam o conjunto de dados.

Segundo Kiryakov; Ognyanov; Manov (2005), uma ontologia pode ser definida como um relacionamento representado pela quadrupla $O = \{C, R, I, A\}$, onde:

- C é o conjunto de classes que representam os conceitos em um domínio;

- R é o conjunto de relações ou associações entre os conceitos do domínio;
- I é o conjunto de instâncias derivadas das classes;
- A é o conjunto de axiomas do domínio.

Conforme Isotani; Bittencourt (2015); Guarino (1998), elas podem ser classificadas: (i) de acordo com a definição de seus conceitos e propriedades, como leves ou pesadas, e (ii) de acordo com seu propósito, como genéricas, de domínio, de tarefas ou de aplicação:

Leves: são aquelas em que cada conceito não é definido em detalhes. Ela só apresenta a relação hierárquica entre conceitos. Geralmente elas são definidas como vocabulários.

Pesadas: necessitam de um alto nível de formalismo se comparado as ontologias leves. Necessita que cada conceito seja definido e organizado baseando-se em princípios bem definidos. Geralmente é usada para criação de bases que serão reutilizadas e compartilhadas.

Genéricas: são usadas para descrever conceitos gerais como objetos, tempo e eventos.

Domínio: descrevem vocabulários relacionados a determinados tópicos.

Tarefa: aplica o conhecimento para resolver problemas em inúmeras circunstâncias sem se importar em qual domínio está.

Aplicação: são usadas para aplicar conceitos de domínio e tarefas, ou seja, são funções que as entidades realizam quando desempenham determinada tarefa.

As ontologias são representadas por meio de linguagens para serem processadas por máquina. De forma ampla, é possível classificar essas linguagens de duas formas: as linguagens voltadas à lógica e as linguagens baseadas em marcação (GAVA; MENEZES, 2003). Este trabalho foca nas linguagens baseadas em marcação, pois foram desenvolvidas para especificação de ontologias na Web Semântica.

No caso da Web Semântica, as ontologias desenvolvidas são classificadas como de peso leve, pois os vocabulários desenvolvidos tendem a não necessariamente representar todo o formalismo habitual das ontologias pesadas. Como linguagem, uma das mais usadas atualmente é a OWL (Web Ontology Language), sendo também bastante utilizada a RDF (Resource Description Framework) como framework para padronização e formatação de dados na web.

Com a OWL é possível representar o conhecimento sobre os dados e a relação entre eles. Segundo Sikos (2015), a OWL descreve ontologias com o uso de alguns

componentes como: atributos, indivíduos, classes, relações, restrições, termos funcionais, regras, axiomas e eventos. Já o RDF é apenas um modelo de representação para descrição de recursos, ele não é uma linguagem como o OWL.

OWL é bem mais complexo que RDF, enquanto o RDF é usado para definir a estrutura dos dados, o OWL descreve relacionamentos semânticos. De acordo com Semantics (2017), o RDF tem um formato flexível e fornece um vocabulário que pode ser usado para descrever recursos de forma simples, enquanto que o OWL tem um formato bem rígido e mais completo, permitindo maior expressividade do modelo de dados. Com ela é possível, por exemplo, restringir opções de modelagem de dados, determinando restrições lógicas de disjunção, inverso de, entre outras, e também especificar consultas de forma mais formal e eficiente.

Na próxima seção, serão dados maiores detalhes com relação às linguagens de marcação voltadas ao desenvolvimento de ontologias na Web Semântica.

2.2 Web Semântica

A Web Semântica permite que humanos e máquinas trabalhem em conjunto através do uso de padrões de formatação de dados, ela faz com que a web seja vista como um enorme banco de dados. A Web Semântica pode ser vista como uma implementação prática das ontologias.

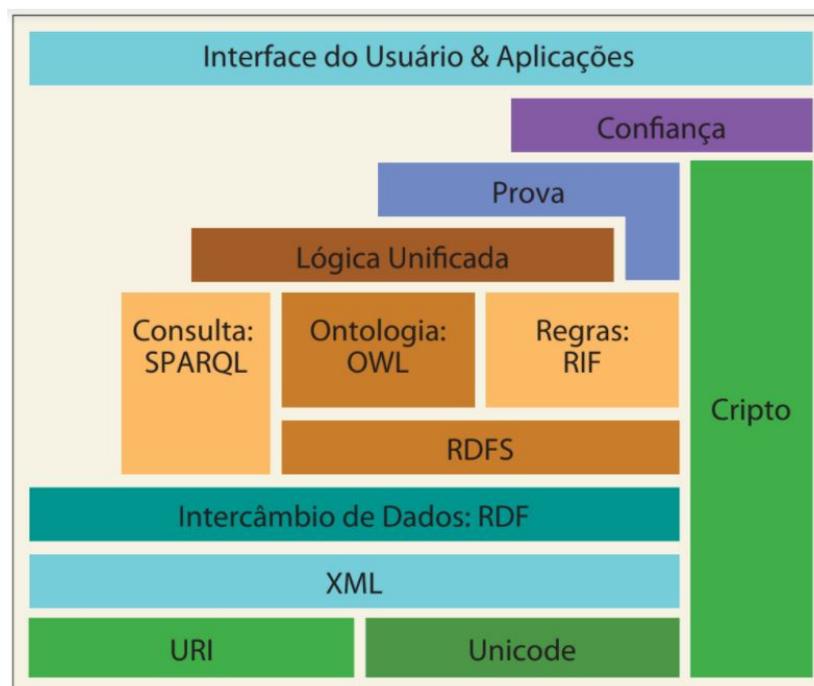


Figura 2 – Camadas da Web Semântica
Fonte: (ISOTANI; BITTENCOURT, 2015)

A pilha de tecnologias que compõe a Web Semântica é apresentada na Figura 2, e no decorrer deste Capítulo as principais tecnologias serão definidas.

Na Figura 3, pode-se observar que a pilha de protocolos definida pela W3C (World Wide Web Consortium) para provimento de Web Semântica agrega vocabulários ontológicos em sua organização.

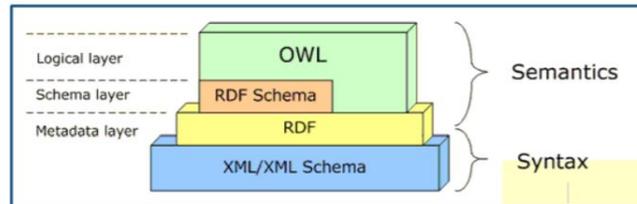


Figura 3 – Ontologias e Web Semântica

Fonte: (DJURIĆ et al., 2004)

Para descrever ontologias dentro da Web Semântica são usadas linguagens baseadas em marcação. Atualmente as linguagens de marcação mais populares são RDF e OWL. A linguagem OWL, criada pela W3C em 2004, tem como objetivo a representação de conhecimento rico e complexo sobre coisas, classes e relações (GROUP, 2012).

Enquanto OWL é rígido, o RDF tem um formato que permite flexibilizações. O RDF é um framework para padronização de formatação de dados. Ele é formado por 3 componentes: recursos, propriedades e triplas. Os recursos são cadeia de caracteres que identificam um recurso na internet, as propriedades são características para descrever um recurso, e as triplas são a união de um recurso, uma propriedade e um valor.

O RDF tem uma extensão chamada RDF-S ou RDF Schema que nada mais é do que uma camada de descrição semântica. RDF-S permite que classes e suas relações sejam descritas usando triplas. Na Figura 4 apresenta-se uma relação entre recursos, propriedades e valores descrita em RDF.

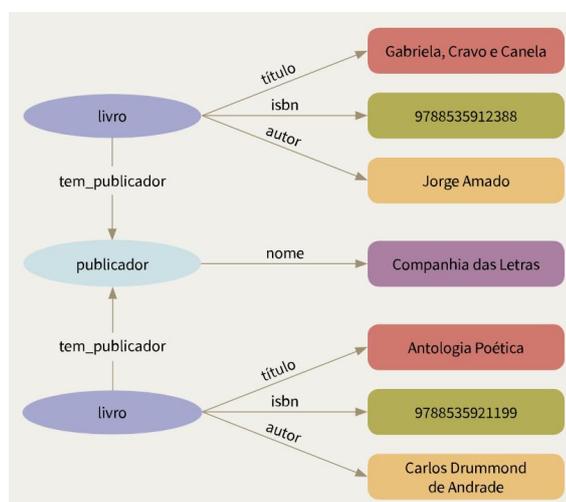


Figura 4 – Representação gráfica do uso de RDF com N-Triplas

Fonte: (LAUFER, 2015)

De acordo com a Figura 5, o formato RDF pode ser apresentado em diversos formatos, sendo estes apresentados na Tabela 1 em maior detalhe. Na Tabela citada cada formato RDF é listado seguido dos tipos de código que cada formato possui e informações de quando usar cada um deles.

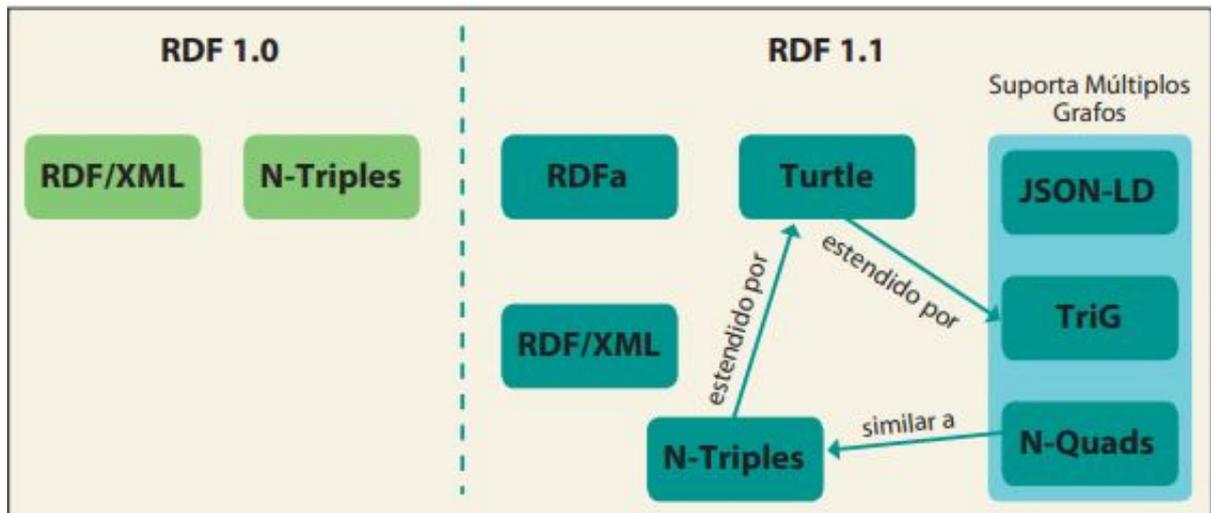


Figura 5 – Formatos de apresentação do modelo RDF

Fonte: (ISOTANI; BITTENCOURT, 2015)

Tabela 1 – Formatos de apresentação RDF e suas características

Formato	Tipo de código	Quando usar
RDFa	Código RDF embutido em HTML	<i>SEO (Search Engine Optimization)</i>
RDF/XML	Código RDF com estrutura em XML	<i>Aplicações que usam estruturas em XML</i>
JSON-LD	Código RDF com estrutura JSON	<i>Aplicações que usam JSON</i>
N-Triples	Código RDF com estrutura de triplas	<i>Processamento e intercâmbio de Big Data em RDF</i>
Turtle	Código RDF para facilitar a leitura humana	<i>Processamento e intercâmbio de Big Data em RDF</i>
TriG	Código com estrutura Turtle	<i>Representação de múltiplos grafos</i>
N-Quads	Código RDF com estrutura de triplas	<i>Processamento e intercâmbio de grandes catálogos de dados</i>

Fonte: (ISOTANI; BITTENCOURT, 2015)

Para se fazer consultas, tanto em OWL quanto no RDF, utiliza-se uma linguagem de consulta chamada SPARQL¹, ela é orientada a dados e só consulta informações presentes nos modelos.

¹<https://www.w3.org/TR/rdf-sparql-query>

Segundo Prud'hommeaux; Seaborne (2008), grupo de trabalho SPARQL que produziu a recomendação W3C para a nova versão do SPARQL 1.1, a linguagem SPARQL possui IRIs (Internationalized Resource Identifiers), que são uma classificação de URI (Uniform Resource Identifiers) que estende o número de caracteres, permitindo que os caracteres chineses, japoneses, cirílicos e coreanos possam ser utilizados. Cada IRI identifica de maneira única um recurso na web.

No SPARQL variáveis são identificadas com os símbolos ? e/ou \$. Já os literais são delimitados por aspas simples ou dupla. Uma consulta SPARQL possui algumas cláusulas iguais ao SQL: SELECT, FROM, WHERE, ORDER BY, além de outras cláusulas específicas da SPARQL, entre elas:

BASE: define a URI base de um recurso;

FILTER: aplica um filtro sobre a consulta;

LIMIT: limita a quantidade de linhas da resposta da consulta;

OFFSET: aplica um deslocamento ao resultado da consulta;

OPTIONAL: permite recuperar uma linha mesmo que ela não possua o valor de uma propriedade do RDF;

PREFIX: cria um apelido para a URI.

2.3 Operacionalização da Web Semântica como Dados Abertos Conectados

Dados conectados são dados ligados entre si, ou seja, representam e descrevem as conexões entre recursos presentes na web. De acordo com Berners-lee (2006), visto que a web mantém hiperlinks para interligar documentos da web, os dados conectados possuem 4 princípios: usar URIs para nomear coisas, usar URIs HTTP (Hypertext Transfer Protocol) para permitir que as pessoas procurem o que desejam, também as URIs devem fornecer informações úteis usando padrões RDF e SPARQL e é preciso incluir links para outros URIs tornando possível explorar mais dados.

Ao observar a pilha de tecnologias que compõe a Web Semântica na Figura 2, é possível identificar que os dados conectados estão presentes dentro da Web Semântica, e isso é o que está sendo apresentado na Figura 6.

O Linked Open Data (LOD), de acordo com Project (2007), é um movimento de dados abertos que tem como objetivo tornar o acesso aos dados gratuito para todos. Alguns exemplos de conjuntos de dados disponíveis são Wikipedia², Geonames³, Cre-

²<https://www.wikipedia.org/>

³<http://www.geonames.org/>

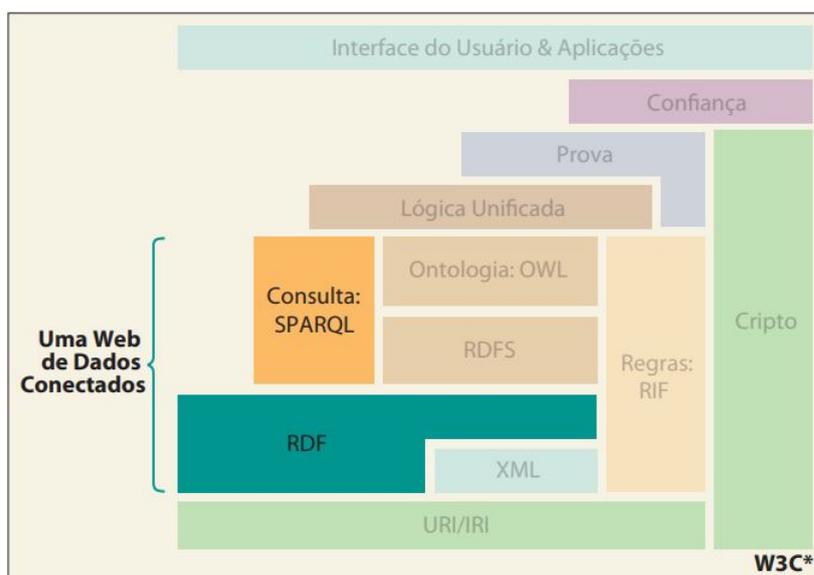


Figura 6 – Dados conectados segundo as camadas da Web Semântica

Fonte: (ISOTANI; BITTENCOURT, 2015)

ative Commons⁴, Linked Brain Data⁵. Esse projeto da comunidade W3C SWEO Linking Open Data tem o intuito de criar vários conjuntos de dados abertos e conectá-los através do padrão RDF.

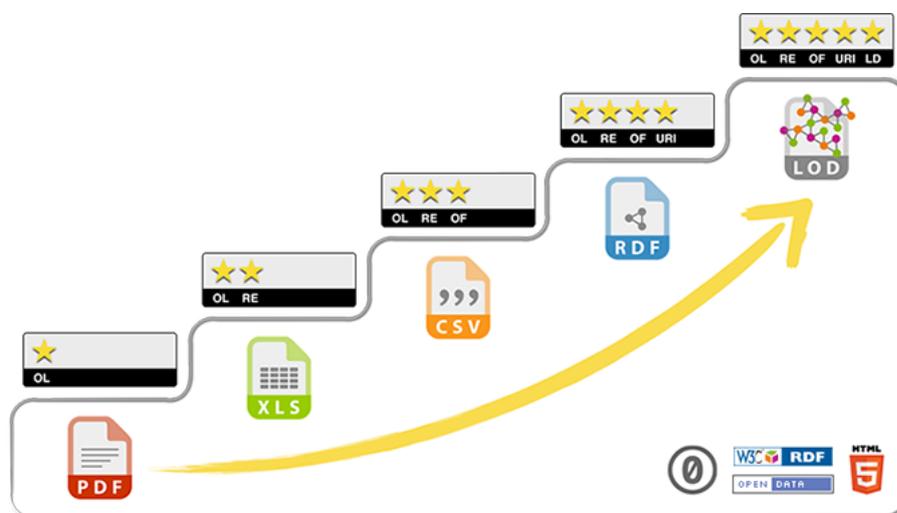


Figura 7 – Esquema de 5 estrelas para dados abertos

Fonte: (HAUSENBLAS, 2012)

Berners-lee (2006) implementou um esquema de 5 estrelas para classificar dados abertos. Na Figura 7 está descrito esse esquema, onde *open license* (OL) quer dizer que os dados devem estar disponíveis na web sob uma licença aberta, *reused* (RE) que os dados devem estar estruturados, *open format* (OF) que os dados precisam estar disponíveis em formato não-proprietário, *uniform resource identifier* (URI) que

⁴<https://creativecommons.org/>

⁵<http://www.linked-brain-data.org/>

os dados precisam ter URIs para que seja possível que outras pessoas possam criar links para eles e *linked data* (LD) que seus dados estejam com link para outros dados.

De acordo com Isotani; Bittencourt (2015), os dados conectados possuem estrutura baseada em RDF e podem ser considerados como um subconjunto da Web Semântica.

Dados são extraídos de inúmeras fontes e devido a isso acabam surgindo problemas de acesso a esses dados. Como solução surgiu a Linked Open Data (LOD) Cloud, que segundo Uyi idehen (2019) é um gráfico de conhecimento capaz de manifestar Linked Data como Web Semântica.

LOD Cloud são criadas utilizando padrões abertos de acesso a dados, sistemas de gerenciamento de banco de dados que armazenam, indexam e realizam consultas a dados estruturados e *endpoints* que servem para acessar dados que são publicados no formato de LOD.

Como exemplos de padrões abertos de acesso a dados temos URI, RDF e Sparql. Um exemplo de SGBD seria o OpenLink Virtuoso⁶. Como exemplo de *endpoint* e também de LOD Cloud temos o DBpedia⁷ que segundo Mccrae et al. (2019) é o maior LOD Cloud disponível atualmente.

Do ponto de vista de *endpoint*, o RDF do DBpedia é hospedado e publicado usando o OpenLink Virtuoso e a infraestrutura do Virtuoso permite a consulta via Sparql através do *endpoint* Sparql publicado utilizando o Virtuoso. A Figura 8 mostra a arquitetura de provisionamento de dados da DBpedia.

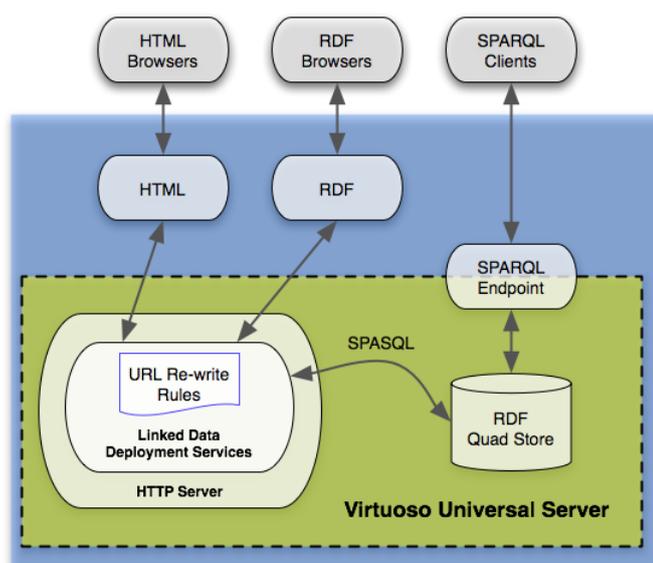


Figura 8 – Arquitetura de Provisionamento de dados do DBpedia
Fonte: (DBPEDIA, 2020)

⁶<https://virtuoso.openlinksw.com/>

⁷<https://www.dbpedia.org/>

3 PENSAMENTO COMPUTACIONAL

Entende-se o Pensamento Computacional (PC) como sendo uma competência atualmente necessária para qualquer pessoa. Segundo Wing (2006), o PC traduz um comportamento e um conjunto de habilidades que auxiliam no processo de resolução de problemas. Ainda segundo a autora, o PC influencia positivamente outras disciplinas, exemplos disso são o aprendizado de máquina transformando a estatística, análise de dados para detecção de fraudes de cartão de crédito ou spam (WING, 2011).

O PC traz diversos benefícios, entre eles estão (WING, 2011; CUNY; SNYDER; WING, 2010):

- Aplicar ou adaptar uma ferramenta ou técnica computacional para um novo uso;
- Aplicar estratégias para dividir e conquistar em qualquer domínio;
- Reformular problemas para serem passíveis de estratégias computacionais;
- Aplicar novos métodos computacionais aos seus problemas.

De acordo com Azevedo et al. (2019) e com a Figura 9, os seguintes princípios, representados como pilares do PC, o tornam decisivo para auxílio na resolução de problemas:

Decomposição: dividir o problema em problemas menores para que seja resolvido diretamente;

Reconhecimento de padrões: encontrar semelhanças entre problemas;

Abstração: ignorar detalhes para que uma solução possa ser usada para mais de um problema;

Algoritmos: enumerar os passos e ações necessárias para resolver o problema.



Figura 9 – Pilares do pensamento computacional
 Fonte: (SÁ, 2018)



Figura 10 – Princípios do pensamento computacional
 Fonte: (PROFLAB, 2017)

Em sala de aula, o PC auxilia os alunos a transformar problemas complexos em algo mais fácil de ser compreendido e solucionado, utilizando os princípios mencionados na Figura 9 e na Figura 10. Além disso, desenvolve habilidades importantes como o raciocínio lógico, capacidade de aprendizado, planejamento, resolução de problemas, entre outras.

3.1 Conceitos em Pensamento Computacional

Conceituar e definir a área de PC é um desafio, visto que é um campo enorme, pois trabalham nela profissionais de várias áreas, além de seus conceitos serem aplicados em todos os níveis de ensino e os autores não possuem uma convergência e padronização para definir os conjuntos de habilidades e tópicos que formam o PC.

A diversidade de definições de PC é um dos principais problema tratado neste trabalho, com vistas disso, foram buscados conceitos com base no estudo desenvolvido por Silva Júnior (2020), o qual identifica como principais fontes na área de PC as autoras e autores Jeannet Wing, Karen Brennan e Mitchel Resnick, cujas publicações possuem propósito e natureza distintas. Como Silva Júnior (2020) propõem uma divisão própria a respeito dos conceitos em PC, explora-se também a proposta de (SILVA JÚNIOR, 2020) neste trabalho. As próximas subseções apresentam as conceitualizações dos autores citados.

Jeannet Wing tem influentes pontos de vista; Karen Brennan e Michel Resnick apresentam conceitos sólidos, tendo desenvolvido um textitframework para Scratch¹; Braz Araujo da Silva Junior, que serve como base para este trabalho, desenvolveu um modelo, baseado em uma revisão sistemática entre diversos autores, no qual propõe uma parceria para o avanço e disseminação do PC (SILVA JÚNIOR, 2020).

3.1.1 Conceitualização de acordo com Wing

Jeannette Wing foi uma das principais responsáveis pela disseminação e popularização do termo PC, em Wing (2006) ela resume PC como um conjunto de habilidades utilizadas para resolução de problemas.

De acordo com Wing (2006) e Wing (2011), problemas complexos são reformulados utilizando decomposição e abstração. Métodos e modelos computacionais possibilitam a resolução de problemas e a complexidade explica métodos e modelos computacionais. A resolução de problemas leva à complexidade e a eficiência, sendo que a avaliação também leva a eficiência. Modelar sistemas, resolver problemas e compreender o comportamento humano se baseiam em conceitos de PC.

A ontologia de topo desenvolvida com base na conceitualização de PC de acordo

¹Scratch é uma linguagem de programação usada para criar jogos, histórias interativas e animações, mais informações em <https://scratch.mit.edu/>

com Jeannette Wing está presente na seção 5.1.1.

3.1.2 Conceitualização de acordo com Brennan e Resnick

Karen Brennan e Mitchel Resnick são autores de artigos sobre PC amplamente conhecidos. Segundo Brennan; Resnick (2012), as competências de PC são desenvolvidas a partir de três vertentes: conceitos computacionais, práticas computacionais e perspectivas computacionais.

Segundo Brennan; Resnick (2012) uma atividade ou tarefa é expressada por uma série de etapas, e uma sequência de instruções especifica uma ação que deverá ser realizada. O entendimento dos conceitos computacionais são necessários para que as pessoas saibam como realizar essas instruções, visando a reprodução de um determinado comportamento. Sequências são usadas para orquestrar as instruções. Paralelismo permite que mais de uma instrução seja realizada ao mesmo tempo. Para se executar uma sequência diversas vezes, são usados os loops. Já os eventos possibilitam que diferentes ações sejam tomadas de acordo com cada situação. Os condicionais permitem que sejam tomadas decisões. Operadores possibilitam manipulações numéricas e de texto. Por fim, os dados que nada mais são do que variáveis, são utilizadas para armazenar, recuperar e atualizar valores.

Práticas computacionais desenvolvem competências para representar os conceitos computacionais, sendo elas divididas em: incremental e iterativo, teste e depuração, reutilização e remixagem, abstração e modularização. Observando Brennan; Resnick (2012), o desenvolvimento de um projeto é um processo adaptativo e incremental, ou seja, ele pode mudar de acordo com a abordagem utilizada para resolução. Testes e depuração são necessários para validar o que pensamos, pois muitas vezes nosso pensamento não prevê todas as possibilidades e por vezes esquecemos algo importante que faz com que todo o desenvolvimento esteja errado. A reutilização e remixagem são necesserárias para evitar que tenhamos que pensar sobre o mesmo problema diversas vezes e assim permitindo que seja utilizado o que foi desenvolvido e validado anteriormente. Já a modularização e a abstração possibilitam que a reutilização e remixagem sejam mais simples, pois permite que pessoas leigas abstraíam o conhecimento que foi usado para resolver algum problema e usem dentro do seu projeto sem precisar entender em como foi desenvolvido.

As perspectivas computacionais desenvolvem como se expressar, como se conectar e como questionar. De acordo com Brennan; Resnick (2012), o desenvolvimento da competência de expressão modifica como as pessoas pensam, passando a ver o PC como algo que possa ser consumido e usado para a resolução de problemas. A conexão com outras pessoas e com outros projetos desenvolve a criatividade, através da conexão surgem novas ideias. E o questionamento é utilizado para ir além das limitações do que é dado como certo e utilizado para melhorar o que já existe.

Na seção 5.1.1 temos a ontologia de topo definida com base nos conceitos de Karen Brennan.

3.1.3 Conceitualização de acordo com Silva Junior

O autor Bras Araujo da Silva Junior desenvolve estudos de revisão sistemática com a intenção de propor um modelo que comporte a maior parte do que estudo de PC abordam, tentando organizar diversos conceitos em um modelo.

Ele divide PC nas seguintes linhas: algoritmo, dados, abstração, decomposição, avaliação e automação. Cada linha é um assunto, e alguns se cruzam.

A linha de **avaliação**, trata os conceitos de otimização, teste e depuração.

O assunto **algoritmo**, abrange paralelismo, fluxo de controle e funções ou procedimentos.

Enquanto **abstração** trata os conceitos de generalização, reconhecimento de padrões e modelagem. E ainda realiza um cruzamento com **decomposição**, que por sua vez abrange recorrência, reuso e emergência.

E por último, a linha **dados** trata de coleta de dados, visualização de dados e estrutura de dados e ainda permeia com **automação**, que por sua vez, trata de desenvolvimento de software, simulação e hardware.

Com base na conceitualização de PC de acordo com Braz Araújo da Silva Júnior está presente na seção 5.1.1.

4 TRABALHOS RELACIONADOS

Durante a fase de referencial teórico presente no capítulo 2, foram buscados trabalhos com propostas de ontologias para PC e dados abertos conectados voltados à educação.

Infelizmente, encontrou-se apenas 1 trabalho voltado a ontologias para PC, o qual está descrito na seção 4.1. Existem muitos trabalhos que visam desenvolver ontologias para ambientes educacionais e educação. Vocabulários como OBBA Vicari et al. (2010), IEEE-LOM Gluz; Vicari (2012) foram desenvolvidos buscando padronização de recursos educacionais, práticas, objetivos de aprendizagem, entre outros. Entretanto, demais esforços voltados especificamente ao PC não foram encontrados.

Com relação à disponibilização de dados educacionais como dados abertos existem já esforços diversos, pois neste caso observa-se criação de bibliotecas digitais de uso aberto, repositórios de aprendizagem ou até mesmo Ambientes Virtuais de Aprendizagem (AVAs) com catálogos e acesso aberto. Pensando na relação mais estrita com a proposta desta Dissertação, buscou-se por iniciativas voltadas a disponibilização de dados educacionais como dados abertos conectados, os quais são apresentados na seção 4.2.

4.1 Ontologias para Pensamento Computacional

Na busca por trabalhos relacionados a padronização de dados e ontologias para PC, foi encontrada uma ontologia disponibilizada por meio de uma plataforma chamada "Micas, a Web Platform to Support Teachers of Computing at School" ¹ Azevedo et al. (2019) proveniente de uma dissertação de mestrado da Universidade do Minho.

O Micas é uma plataforma web de apoio aos professores no ensino de computação através da coleta e classificação de recursos de aprendizagem a serem usados pelos professores nas aulas de informática.

A plataforma web do Micas permite que sejam inseridos quaisquer tipos de recur-

¹<https://micas.epl.di.uminho.pt/ontology>

tos de aprendizagem que possam ser aplicados por professores no ensino de informática. Ela provê uma ontologia que é utilizada para categorizar e pesquisar recursos de aprendizagem por área e por ano de ensino. A ideia dos autores é auxiliar os professores na busca por recursos a serem aplicados em sala de aula para ensino de PC de acordo com o nível de ensino.

Recursos de aprendizagem são utilizados pelos educadores para que o ensino seja mais eficaz e estimulante. Através da plataforma web do Micas é possível pesquisar quais recursos de aprendizagem são adequados no ensino de cada conceito ou tópico específico de acordo com o ano escolar especificado.

Sendo assim, pode-se concluir que o Micas tem um propósito diferente do que o trabalho proposto nessa dissertação. O Micas possui uma ontologia que é disponibilizada através de uma plataforma web em que educadores podem pesquisar recursos de aprendizagem de acordo com o tema escolhido e o ano escolar em que seus alunos estão ou os educadores podem inserir novos recursos de aprendizagem. A ontologia que está por trás da plataforma categoriza esses recursos para que outros educadores possam utilizá-los.

Enquanto o trabalho proposto na dissertação tem o intuito de ser usado como ferramenta para padronização dos conceitos de PC e disponibilizado de forma aberta na web por meio de um *endpoint* para que possa ser amplamente difundido e utilizado.

O mesmo grupo de pesquisa que desenvolveu o Micas, também desenvolveu dois outros trabalhos na área de PC e ontologias. Um dos trabalhos é o Araújo et al. (2020), que é um artigo onde é discutido quais os benefícios de desenvolver uma ontologia e aplicar a diferentes níveis de ensino. De acordo com o artigo, devido a falta de motivação e a ausência de habilidades para resolução de problemas ocorre a falha no ensino de PC.

Sendo assim, surge a necessidade de criar uma ontologia para que ocorra uma adaptação de ensino de cada conceito de PC de acordo com o nível de escolaridade e a profundidade que o conceito seja abordado em cada nível educacional.

O outro trabalho desenvolvido pelo grupo é o Araújo; Lima; Henriques (2019) que investiga o impacto do uso de realidade aumentada para ensinar recursos através de aulas de informática no ensino fundamental e médio para desenvolvimento de lógica e raciocínio abstrato para resolução de problemas, afim de tornar o ensino de programação de computadores uma tarefa menos complexa.

Todos os três trabalhos mencionados acima estão preocupados em como ensinar PC de acordo com os níveis de ensino e definindo a profundidade com que cada conceito de PC deve ser ensinado em cada ano de ensino. Enquanto esses trabalhos tem o intuito de criar uma espécie de base comum curricular para ensino de PC, essa dissertação tem o objetivo de padronizar os conceitos de PC, visto que cada autor descreve um conceito de maneira diferente.

4.2 Dados Abertos Conectados Voltados à Educação

Na busca por trabalhos relacionados que envolvem os temas de dados abertos conectados e educação, foram encontrados dois artigos. O primeiro deles, intitulado "Dados abertos conectados para educação" Bandeira et al. (2015), discute o potencial de dados abertos conectados para aperfeiçoar a educação através da produção, do compartilhamento e do reuso de dados.

Segundo Bandeira et al. (2015), existem diversas iniciativas, entre elas a *LinkedEducation.org*², que revelam que o uso de dados abertos conectados podem fomentar atividades relacionadas à educação e viabilizar a criação de bases de conhecimento que irão possibilitar uma melhor gestão e desenvolvimento educacional.

Em Bandeira et al. (2015) os autores identificam as iniciativas existentes ligadas ao desenvolvimento de dados abertos conectados e educação, com objetivo de evidenciar os benefícios trazidos por essas iniciativas e mostrar que existem ainda diversas investigações que podem ser desenvolvidas visando a aplicação de dados abertos conectados em maior escala no contexto educacional. Este objetivo é diferente do apresentado nesta dissertação, na qual busca-se fornecer um modelo de dados para que outras iniciativas tenham uma camada voltada ao conceito geral da raiz de PC.

Outro trabalho relacionado encontrado voltado ao tema de dados abertos conectados e educação é intitulado "Desafios no uso de Dados Abertos Conectados na Educação Brasileira" Alcantara et al. (2015). Segundo os autores, nos últimos anos a quantidade de dados publicados na web cresceu muito e existem diversas iniciativas que visam a transparência de dados, inclusive na educação. Porém, muitas bases apresentam limitações que dificultam o enriquecimento dos dados e conexão com outras bases de dados.

O objetivo de Alcantara et al. (2015) é elencar os principais desafios quanto ao consumo e publicação de dados abertos conectados na educação brasileira. Ele busca estabelecer discussões que proponham soluções para esses desafios. Já a presente dissertação tem o objetivo de desenvolver uma ontologia para padronização dos conceitos de PC, visando publicá-la em formato aberto para que qualquer pessoa possa reutilizá-la.

²<http://www.linkededucation.org/>

5 CONCEPÇÃO E MODELAGEM

PC é um termo que está em alta nas publicações Ulas; Haseski; Tugtekin (2018), contudo as visões dos autores são um tanto divergentes, tornando PC um termo genérico. Dado que muitos pesquisadores entusiastas estão aprendendo sobre esse termo, é um grande problema não ter uma definição consensual para guiar as pesquisas de quem está iniciando nessa área.

Como discutido na seção 1.1, a motivação deste trabalho é justamente criar um modelo que capture e seja capaz de integrar as múltiplas perspectivas dos autores sobre o tema PC. E como esse trabalho visa auxiliar a comunidade que pesquisa sobre PC, é preciso que o mesmo seja disponibilizado de forma aberta possibilitando reuso para que qualquer pessoa possa utilizá-lo.

A primeira decisão de projeto foi buscar na literatura padrões para definição e representação de dados. Nesse ponto, optou-se por explorar definições em dados abertos conectados.

Os dados abertos conectados fornecem padrões para aplicação nos diferentes níveis de criação da plataforma, desde criação de vocabulário através de ontologias, representação de dados com os formatos OWL e RDF, integração e compartilhamento de dados por meio de arquivos JSON e XML.

O uso de ontologias foi necessário para a criação de um vocabulário semântico, que represente o conhecimento do domínio e seja interpretável por máquina e também por sua capacidade de representação conceitual compartilhada.

Para a representar o conhecimento sobre os dados e a relação entre eles foi utilizado o formato OWL. Ele possui flexibilidade para definir propriedades das relações, o que não é possível com o uso do RDF e também permite a criação de vocabulários mais ricos. E permite criar relações de mais alto nível, por exemplo, fazer raciocínios usando a lógica descritiva.

Afim de realizar consultas em dados representados no formato OWL optou-se por uma linguagem de consulta chamada SPARQL, ela é orientada a dados e realiza consultas em informações presentes no modelo.

Como editor de ontologias foi utilizado o *software* Protégé¹, pois ele é gratuito, tem código aberto, possui uma comunidade ativa na WEB e tem *plugins* para todos os padrões de *linked data* voltados à web.

5.1 Definição do Escopo e Modelo da Ontologia

Como explicado na seção 2.1, as ontologias podem ser classificadas quanto ao nível de abstração. Nesse trabalho são exploradas as ontologias de topo e as ontologias de domínio.

Ontologias de topo são aquelas que descrevem conceitos partilhados entre domínios, ou seja, elas resolvem problemas abstraindo o conceito do domínio. Já as ontologias de domínio descrevem conceitos próprios de um domínio. Partindo das ontologias de topo, se tem uma ontologia de domínio dentro do cenário da aplicação.

Sendo objetivo deste trabalho integrar perspectivas e visões de diferentes autores, estas foram representadas na forma de ontologias de topo, independentes, mas que podem ser importadas para interoperação em uma ontologia de domínio, a qual fará efetivamente a instanciação dos termos tratados nos cenários de aplicação.

Foram definidas três ontologias de topo, com visões de três autores do PC. Entretanto, entende-se como necessário que outras visões possam ser incorporadas futuramente. Desta forma, criou-se a ontologia de domínio importando os conceitos de alto nível das ontologias de topo definidas.

Para criação do vocabulário da ontologia de domínio partiu-se da premissa de definição de conceitos do domínio educacional. Nas subseções seguintes são apresentadas as ontologias de topo, a ontologia de domínio e a forma adotada para integração entre as ontologias desenvolvidas neste trabalho.

5.1.1 Ontologias de Topo

Para possibilitar que visões de diferentes autores do PC fossem representadas em uma ontologia de domínio, foram modeladas as ontologias de topo descritas abaixo. A ideia foi permitir que a ontologia de domínio pudesse importar tantos conceitos de topo quanto fossem requeridos para representar o conhecimento.

Levando em consideração os conceitos de PC explicados no capítulo 3, seção 3.1 foram desenvolvidas três ontologias de topo, uma para cada autor identificado como relevante conforme explicação também realizada na mesma seção.

Com relação as Figuras 11, 12 e 13, os conceitos estão representados em retângulos e as arestas representam os relacionamentos. Arestas representadas em linhas são relações de subclasse, já as arestas hachuradas, que estão presentes apenas na Figura 11, representam propriedades.

¹<https://protege.stanford.edu/>

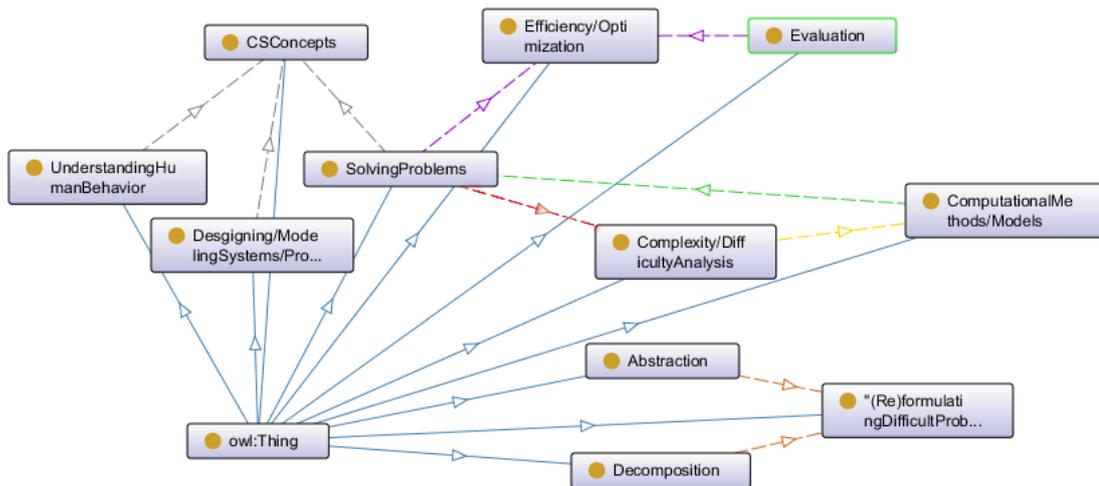


Figura 11 – Ontologia de topo - Jeannette Wing

Fonte: Própria

Conforme conceitualização de Wing na subseção 3.1.1 foi desenvolvida a ontologia de topo da Figura 11. Na Figura 11 é importante salientar suas propriedades, que são as arestas tracejadas que definem a relação entre os conceitos. As arestas na cor cinza representam a relação de envolvimento, sendo assim, com base em **conceitos de ciência da computação** (*CS concepts*) PC envolve **compreender o comportamento humano** (*understanding human behavior*), **resolver problemas** (*solving problems*) e **projetar sistemas** (*modeling systems*). Quanto às arestas de cor laranja, dizem respeito a como atacar problemas difíceis, ou seja, PC utiliza **abstração** (*abstraction*) e **decomposição** (*decomposition*) para **reformular problemas difíceis** (*reformulating difficult problems*). Arestas roxas representam a propriedade de levar a eficiência, com isso, a **avaliação** (*evaluation*) e a **resolução de problemas** levam a **eficiência** (*efficiency*). E por fim, **métodos computacionais** (*computational methods*) que de acordo com a aresta verde, permitem (**resolver problemas**) e através da aresta na cor amarela **análise de dificuldade** (*complexity/difficulty analysis*) explica **métodos computacionais**. Já a **resolução de problemas** leva a **análise de dificuldade** que é representada pela aresta vermelha.

A conceitualização de acordo com Brennan, desenvolvida na subseção 3.1.2, é refletida na Figura 12. Importante salientar que tanto a ontologia de topo de Brennan quanto a de Silva Junior são expressadas por relação de classes e subclasses, ou seja, essas duas ontologias de topo não possuem propriedades com ranges e domínios, sendo assim, não possuem as arestas hachuradas como a ontologia de domínio e a ontologia de topo da Wing.

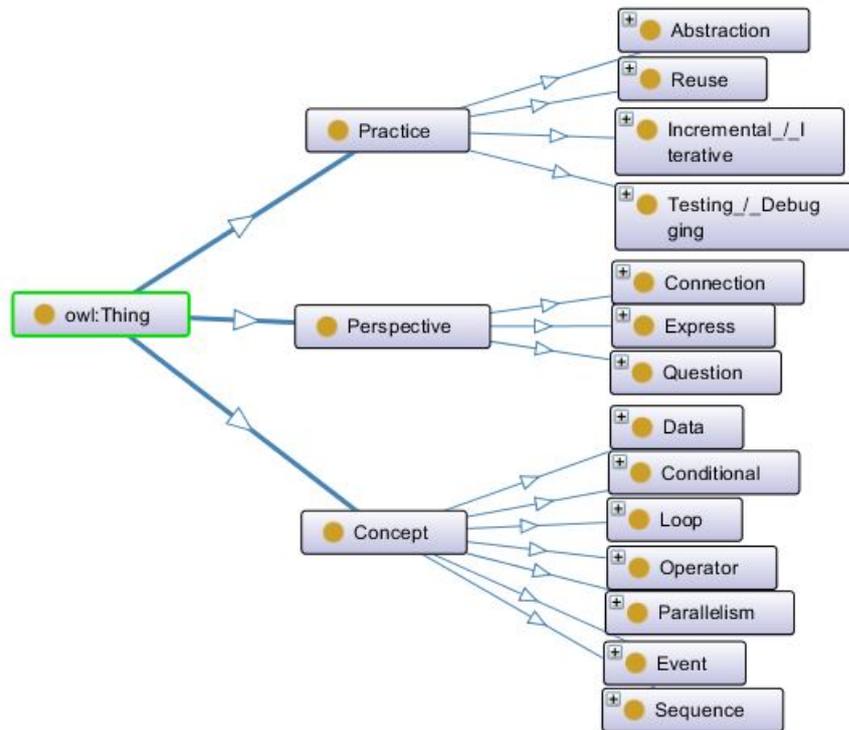


Figura 12 – Ontologia de topo - Karen Brennan
 Fonte: Própria

Na Figura 13 é apresentada a ontologia de topo desenvolvida com base nos conceitos de Silva Junior, descritos na subseção 3.1.3.

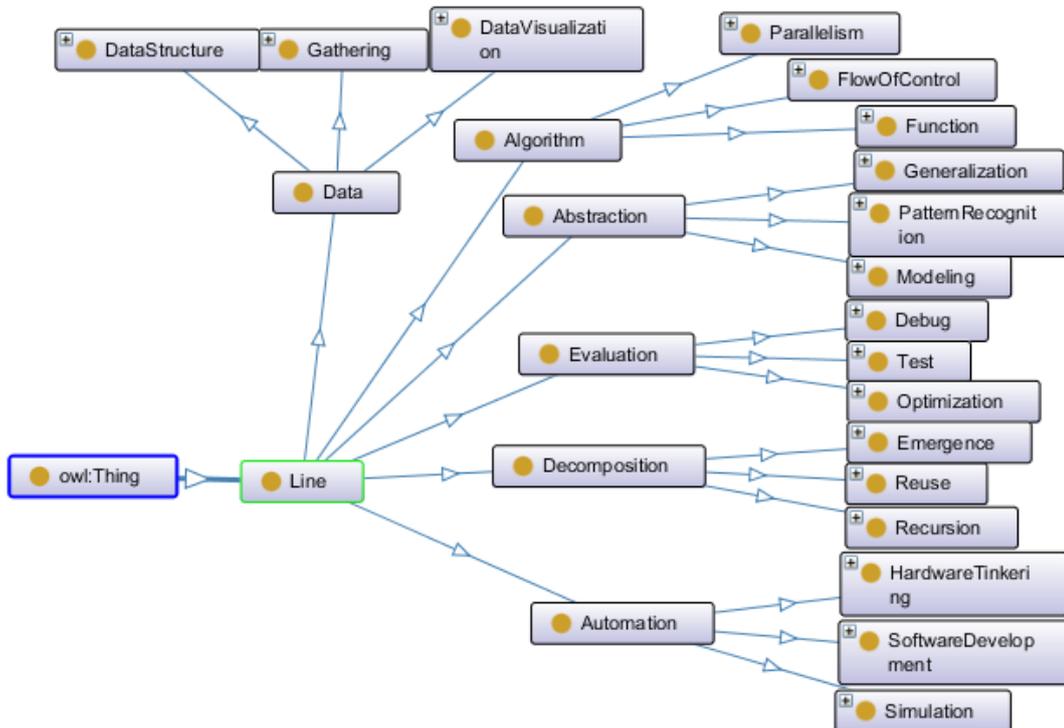


Figura 13 – Ontologia de topo - Silva Junior
 Fonte: Própria

5.1.2 Ontologia de Domínio

Conforme conceituado nas seções 2.1 e 5.1, ontologias de domínio caracterizam e descrevem determinado domínio. Neste trabalho, a ontologia de domínio estabelece conceitos básicos para que seja possível classificar, organizar e relacionar recursos ou tarefas de PC, conforme modelagem apresentada na Figura 14.

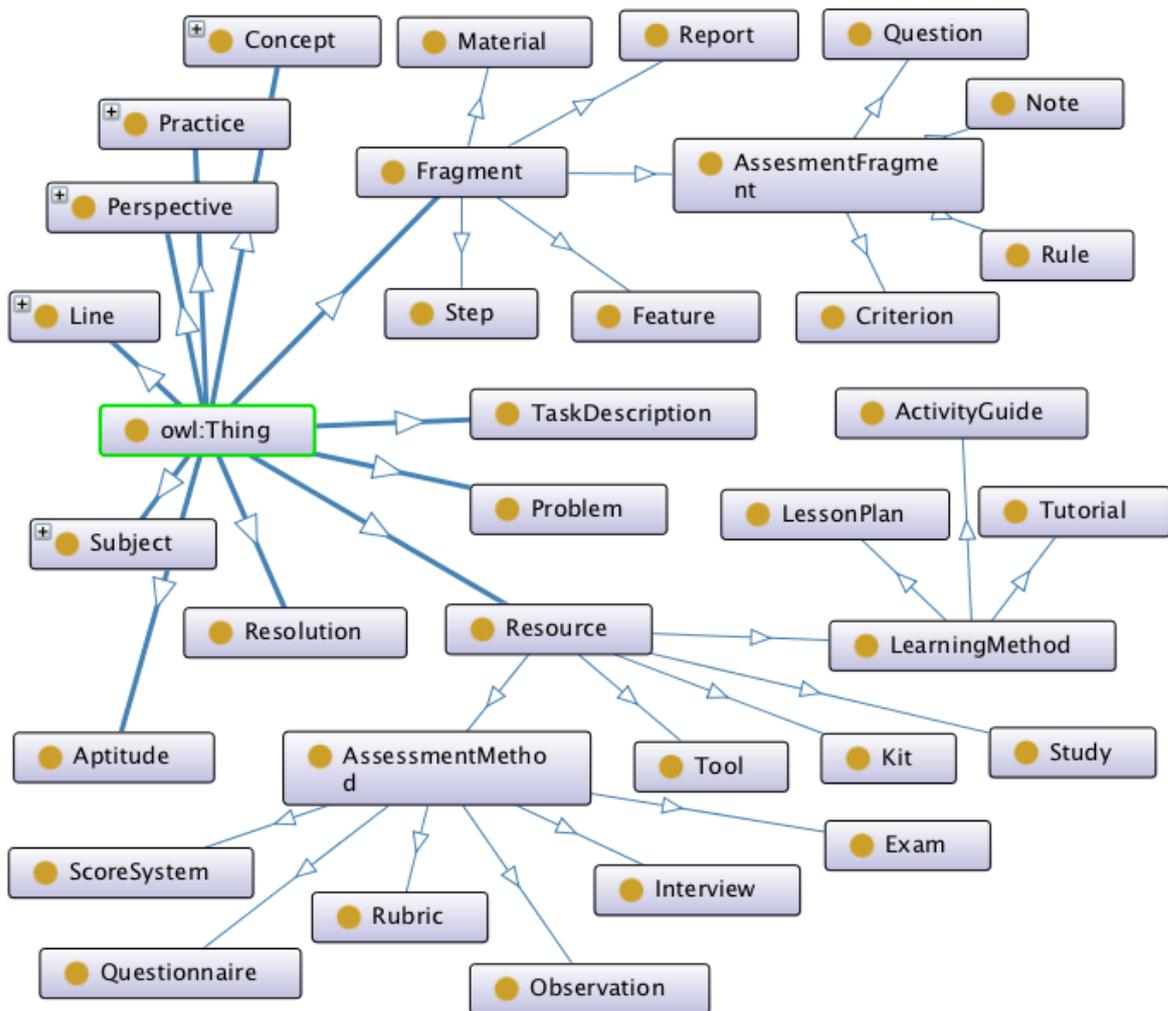


Figura 14 – Ontologia de domínio

Fonte: Própria

Como explicado na seção 3.1, para o desenvolvimento da ontologia de domínio usou-se como base a pesquisa desenvolvida em Silva Júnior (2020), além de diversas reuniões com o autor para compreensão dos conceitos e formulação das classes e dos relacionamentos.

Para explicar a ontologia de domínio da Figura 14, explora-se a Figura 15 e a Tabela 2, onde cada classe (descrita em **negrito** no texto) e sua relação (em *itálico*, seguida de um número correspondente a sua localização na Figura 15) pode ser observada. Além disso, na Tabela 2 a coluna número da propriedade diz respeito a numeração das relações da Figura 15.

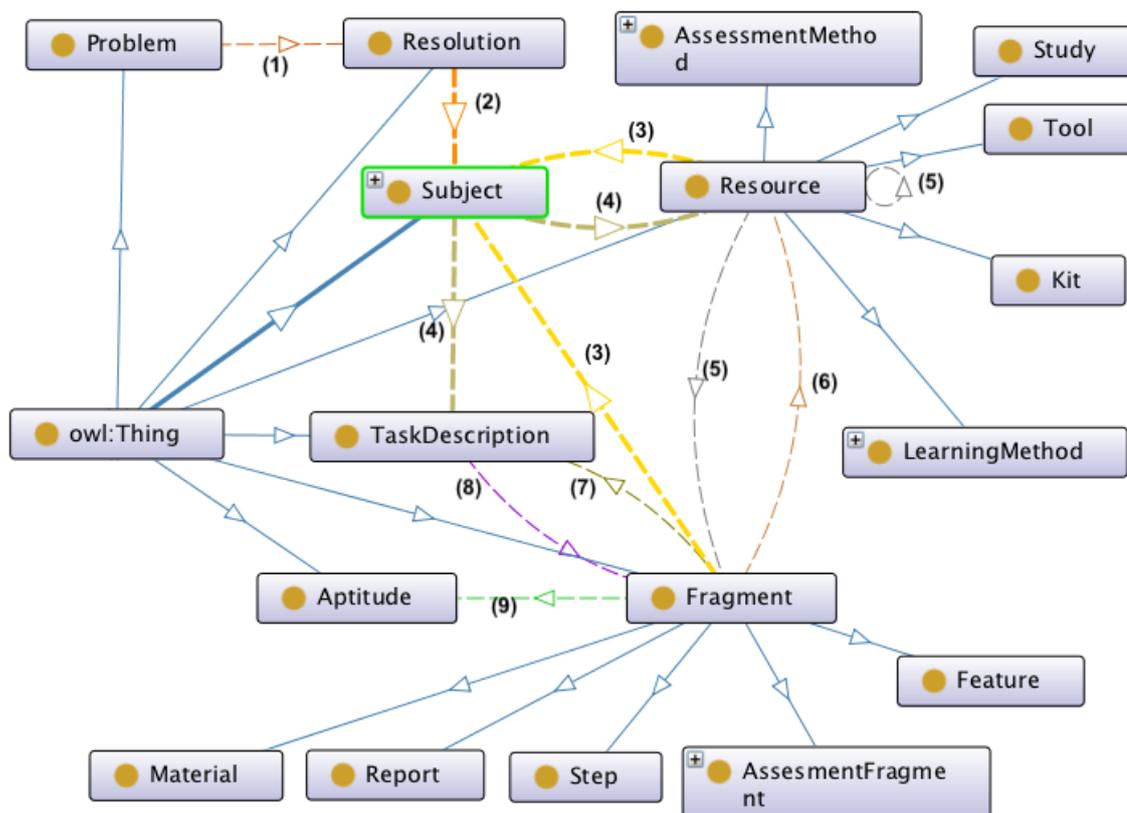


Figura 15 – Propriedades da ontologia de domínio

Fonte: Própria

Tabela 2 – Relacionamentos (propriedades) da ontologia de domínio

Propriedade	N.º Propriedade	Domínio	Range
achievedByResolution	2	resolution	subject
approachedBy	4	subject	resource, taskDescription
approachesSubject	3	resource, fragment	subject
composedBy	5	resource, fragment	fragment, resource
demandsTaskDescription	7	fragment	taskDescription
makeByResource	6	fragment	resource
materializedByFragment	8	taskDescription	fragment
requireAptitude	9	fragment	aptitude
requireResolution	1	problem	resolution

Um **problema (problem)** *requer* (1) uma **solução (resolution)**, a resolução de problemas é comum para as conceitualizações discutidas na seção 3.1. A **solução** pode ser alcançada *através* (2) de **alvos de aprendizagem**, que podem ser *abordados por* (4) um **recurso**, e por sua vez, um **recurso aproxima** (3) um **alvo de aprendizagem**.

Recurso é uma classe utilizada com o intuito de generalizar qualquer conteúdo que possa ser usado para desenvolver uma aptidão.

Com intuito de explorar quais **alvos de aprendizagem** os **recursos** abordam, determinou-se que **recursos** são *compostos por* (5) outros **recursos** ou **fragmen-**

tos, e fragmentos podem compor (6) um recurso. Já fragmentos materializam (7) uma tarefa/descrição, que pode ser materializada por (8) diversos fragmentos. Por fim, fragmentos requerem (9) uma aptidão que é uma estimativa de sua dificuldade.

Os fragmentos (*fragments*) presentes na ontologia de domínio na Figura 14, mapeiam pontos de interesse e permitem recuperar recursos de acordo com **alvos de aprendizagem** (*subjects*) específicos, os alvos de aprendizagem podem ser visualizados na Figura 16. Sendo assim, se determinada tarefa não for suficiente para o desenvolvimento de determinado alvo de aprendizagem, é possível encontrar tarefas alternativas para o desenvolvimento desse alvo de aprendizagem.



Figura 16 – Camada alvo de aprendizagem
Fonte: Própria

5.1.3 Integração entre as Ontologias

A conexão entre as ontologias de topo e a ontologia de domínio é feita a partir do conceito **alvo de aprendizagem**, o qual acaba por se comportar como um conceito de mais alto nível, como uma camada interna ao modelo. A classe alvo de aprendizagem, apresentada na Figura 16, possui diversas subclasses necessárias para a classificação no PC. A comunicação entre as ontologias é realizada a partir da importação das ontologias de topo de cada autor.

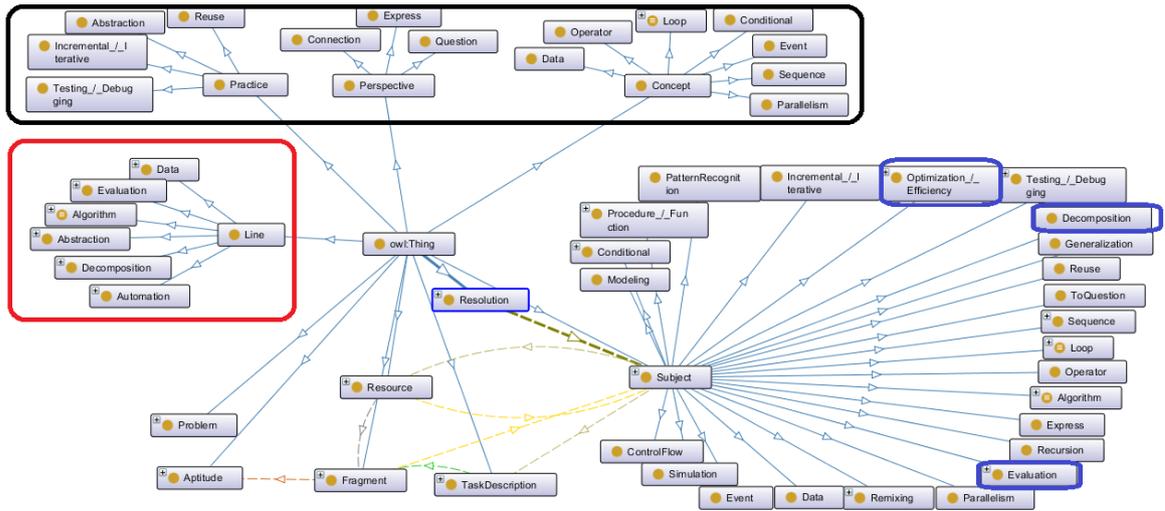


Figura 17 – Relação extrínseca
 Fonte: Própria

Na Figura 17 podemos verificar de perto a ontologia de domínio apenas com as partes que se relacionam com as ontologias de topo. As classes **conceito**, **perspectiva**, **prática** que estão dentro do retângulo preto são as classes oriundas da importação da ontologia de topo de Brennan, a classe **linha** e suas subclasses que estão com contorno em vermelho aparecem devido a importação da ontologia de topo de Silva Junior.

A ontologia de topo da autora Wing, por não fornecer uma hierarquia organizada em alvos de aprendizagem, é representada no domínio diretamente por subclasses dentro da classe alvos de aprendizagem. Na Figura 17 essas subclasses são as que possuem um retângulo azul em volta.

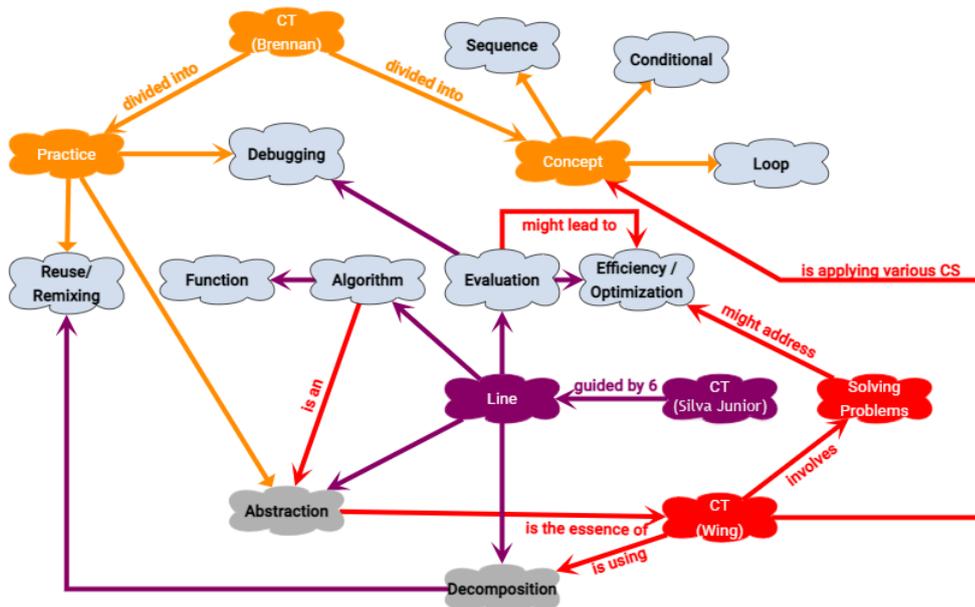


Figura 18 – Relacionamento entre autores

Partindo da Figura 18, é possível perceber que vários **alvos de aprendizagem** se relacionam. E esses alvos de aprendizagem podem ser organizados de acordo com a visão de cada um dos três autores cujos conceitos de PC foram descritos nesse trabalho e podem ser vistos nas subseções 3.1.1, 3.1.2 e 3.1.3.

Na Figura 18 são apresentados alguns dos **alvos de aprendizagem** presentes nos cenários da Figura 19, e como esses **alvos de aprendizagem** são visto por cada um dos três autores.

O relacionamento entre autores, presente na Figura 18, que também está presente no artigo Silva et al. (2021), deixa claro que a **abstração** é um **alvo de aprendizagem** auxiliar. Para o entendimento de Wing um **algoritmo** (*algorithm*) é uma **abstração**, assim como para Silva Junior a **decomposição** alcança o **reuso/remixagem** (*reuse/remixing*).

A **abstração** é uma interseção entre os três autores, já a **decomposição** é uma interseção entre Wing e Silva Junior. Essas interseções não são discordâncias, pois ao dizer que **abstração** é uma essência para Wing não anula o fato de ser uma linha para Silva Junior e uma prática para Brennan.

6 DISPONIBILIZAÇÃO DA ONTOLOGIA DE DOMÍNIO

Neste capítulo são apresentados três cenários concebidos para avaliar a ontologia de domínio criada. Na sequência, é descrito o processo de disponibilização da ontologia como dados abertos conectados, que ocorreu por meio da definição de um *endpoint*. Por fim, são apresentadas consultas SPARQL realizadas no *endpoint* também para teste da ontologia e dos cenários apresentados.

6.1 Cenários de Avaliação

Para avaliar a ontologia de domínio descrita na subseção 5.1.2, foram criados três cenários, os quais também foram publicados no artigo Silva et al. (2021), com recursos de diferentes naturezas para que a ontologia de domínio modele-os e também exponha as diferenças e semelhanças entre os recursos dos cenários.

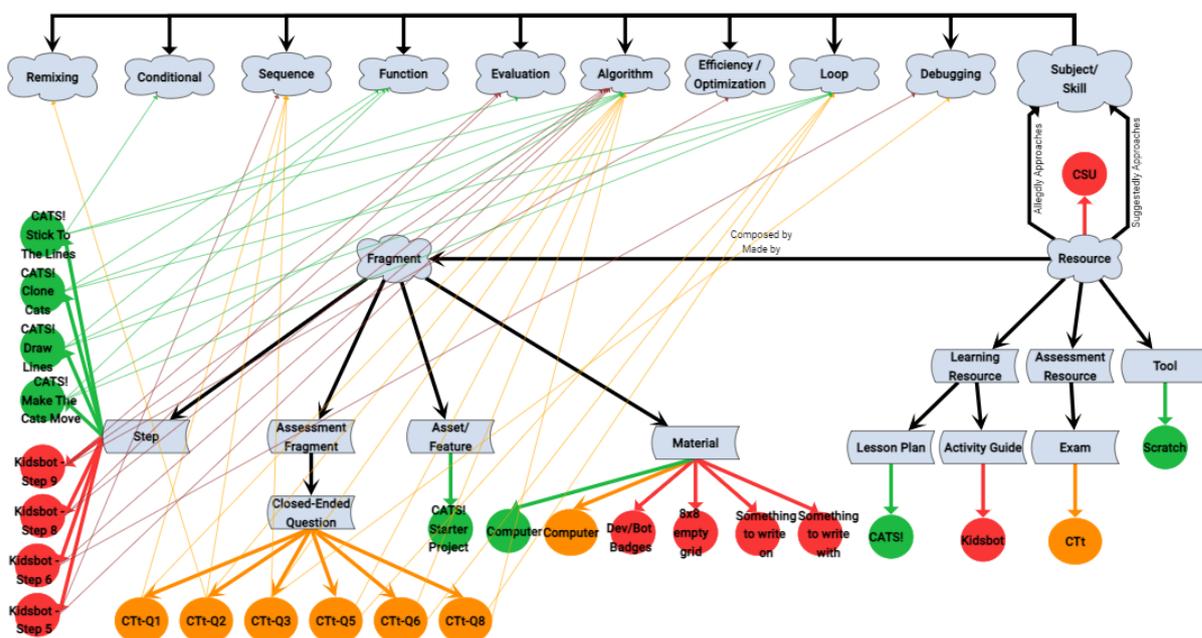


Figura 19 – Cenários para validação da ontologia de domínio

Na Figura 19, são apresentados os três cenários escolhidos com recursos instanciados na ontologia de domínio desenvolvida. Em seguida são descritos os cenários

escolhidos: teste de pensamento computacional (*computational thinking test - CTt*)¹, ciência da computação desplugada (*computer science unplugged - CSU*)² e Scratch³.

Os três cenários são apresentados na Figura 19 como instâncias da ontologia de domínio. Para melhor entendimento da explicação sobre os cenários na Figura 19, as classes serão colocadas em **negrito**, as relações em *itálico* e as instâncias dentro de **[colchetes]**.

6.1.1 Teste de pensamento computacional

O Teste de Pensamento Computacional é um instrumento de avaliação de habilidades de PC que fornece evidências de confiabilidade e validade (ROMÁN-GONZÁLEZ; PÉREZ-GONZÁLEZ; JIMÉNEZ-FERNÁNDEZ, 2017).

Os testes consistem em questões de múltipla escolha com 4 alternativas cada questão e apenas uma das respostas está correta. As questões abordam sequências básicas (4 itens), repetições *loops* (4 itens), repetições até *loops repeat until* (4 itens), condicionais simples (4 itens), condicionais complexas (4 itens), condicionais enquanto (4 itens), funções simples (4 itens), sequências onde é preciso colocar de forma ordenada um conjunto de comandos (14 itens), preencher um conjunto incompleto de comandos (9 itens), depurar um conjunto de comandos incorretos (5 itens). As questões são aplicadas a alunos do 5º ao 10º ano, oferecendo um novo instrumento para medição de PC e reforçam a definição de PC como uma habilidade de resolver problemas.

CTt está sendo usado como um caso de estudo dentro da ontologia desenvolvida pois oferece uma maneira de avaliar que tipo de questão desenvolve cada habilidade.

Para o cenário de CTt, foram criadas seis questões para avaliar habilidades de PC de acordo com informações extraídas em (ROMÁN-GONZÁLEZ; PÉREZ-GONZÁLEZ; JIMÉNEZ-FERNÁNDEZ, 2017).

Na Figura 19 **[CTt]**, que está representado na cor laranja, é uma instância de **recurso** *resource* do tipo **exame** *exam* da classe de **recurso de avaliação** *assessment resource* e está representado na cor laranja. Ele *aborda* **alvos de aprendizagem** e é *composto por* um **material** chamado **[computador]** *[computer]* pois se trata de uma avaliação virtual. Também é *composto por* seis **questões fechadas** *closed-ended question* que são **fragmentos de avaliação** *assessment fragment*. Cada uma das questões *requer* uma **aptidão** e *materializa* uma **tarefa**.

As questões desenvolvidas para esse cenário, são as seguintes:

¹Teste de pensamento computacional, encontrado em: https://docs.google.com/forms/d/e/1FAIpQLSdPdSj_ZVUhIhG4S3bCH6zXSHZoHHbv60smCF9drmbDpfBy_Q/viewform?fbzx=1710660382635002705

²Desplugada é um termo usado para se referir a uma abordagem de ciência da computação que não requer um computador Bell et al. (2009)

³Scratch é uma linguagem de programação usada para criar jogos, histórias interativas e animações, mais informações em <https://scratch.mit.edu/>

- **[CTt-Q1]** materializa uma **definição de tarefa (tarefa)** (*task definition*) que trata sobre sequenciamento de direções e abordam os alvos de aprendizagem **sequência** (*sequence*) e **algoritmo**;
- **[CTt-Q2]** materializa uma **tarefa** que completa um caminho por sequenciamento de direções e abordam os alvos de aprendizagem **sequência**, **reuso** (*remixing*) e **algoritmo**;
- **[CTt-Q3]** materializa uma **tarefa** que consiste em corrigir caminhos e aborda **sequência**, **depuração** (*debugging*) e **algoritmo**;
- **[CTt-Q5]**, **[CTt-Q6]** e **[CTt-Q8]** materializam uma **tarefa** que realiza um caminho que repete direções, abordando **repetição** (*loop*) e **algoritmo**.

Das questões utilizadas nesse cenário para avaliar habilidades de PC, todas avaliam conhecimentos em **algoritmos**, três delas envolvem habilidades em **sequência**, uma avalia **reuso**, uma requer **depuração** e três avaliam o conhecimento em **loop**. A Figura 19 mostra o cenário do CTt instanciado na ontologia, reforçando a capacidade de ser reutilizada por outras pessoas, e a Figura 22 mostra a possibilidade de avaliar as instâncias dos cenários.

6.1.2 Ciência da computação desplugada

A Ciência da Computação Desplugada é um projeto que fornece material didático gratuito e que tem o objetivo de ensinar ciência da computação usando jogos (UN-PLUGGED, 2021).

No cenário de CSU, representado na Figura 19 na cor vermelha, foi modelado um material didático, o *Activity Guide for KidBots*⁴, que é uma atividade em que um aluno descreve uma sequência de instruções para controlar outro aluno, para a modelagem foram destacadas 4 de suas etapas (passos de 5 a 9) que são descritas nos parágrafos seguintes.

[KidBots] é um **recurso** do tipo **guia de atividade** (*activity guide*) da classe **recurso de aprendizagem** (*learning resource*). O **[KidBots]** é *composto por oito fragmentos*, sendo quatro dos **fragmentos** do tipo de **materiais** (*materials*): **[Dev/Bot Badges]**, **[grade vazia 8x8]** (*8x8 empty grade*) e **[algo para escrever]** (*something to write on*), **[algo com o que escrever]** (*something to write with*). E quatro **fragmentos** do tipo **passos** (*steps*) que exploram **depuração**, **sequência**, **eficiência/otimização** (*efficiency/optimization*), **avaliação** e **algoritmo**.

Os quatro passos são os seguintes:

- **[Kidsbot - Step 5]** materializa uma **tarefa** que consiste em percorrer um caminho sequenciando direções, onde são abordados **sequência** e **algoritmo**;

⁴<https://www.csunplugged.org/en/at-home/kidbots/>

- **[Kidsbot - Step 6]** materializa uma **tarefa** para corrigir caminho, e ela desenvolve **depuração** e **algoritmo**;
- **[Kidsbot - Step 8]** materializa uma **tarefa** onde é necessário encontrar caminhos alternativos e desenvolvem **avaliação** e **algoritmo**;
- **[Kidsbot - Step 9]** materializa uma **tarefa** que reconhece caminhos ineficientes, desenvolvendo **eficiência/otimização**, **avaliação** e **algoritmo**.

6.1.3 Scratch

Segundo Resnick et al. (2009), Scratch é uma linguagem de programação em blocos, possuindo atualmente a maior comunidade online para compartilhamento de projetos educacionais.

De acordo com ABOUT (2016), Scratch é muito utilizado no ensino de programação para crianças pois possui uma interface visual, é simples e intuitivo, sendo assim facilita o aprendizado sobre programação e possibilita a criação de jogos, histórias e animações.

Pensando no cenário do Scratch, foi modelado o CATS!⁵, onde um aluno cria um aplicativo que coloca inúmeros gatos na tela.

No cenário do Scratch, representado pela cor verde na Figura 19, **[Scratch]** é um **recurso** do tipo **ferramenta** que *está relacionado* com outros **recursos**, em nosso exemplo, *está relacionado* ao **[CATS!]**, que é um **recurso de ensino** do tipo **plano de aula**.

O **[CATS!]** *requer* um **fragmento** do tipo **material** chamado **[computador]**. **[CATS!]** é *composto por* um **fragmento** chamado **[CATS! projeto inicial]** e também é *composto por* mais quatro **fragmentos** do tipo **passos** que desenvolvem **condicionais**, **funções**, **avaliação**, **algoritmo** e **repetição**.

Os quatro **passos** mencionados acima, são:

- **[CATS! - draw lines]** materializa uma **tarefa** que deverá programar um evento que será disparado por uma entrada, desenvolvendo os **alvos de aprendizagem** **repetição**, **função** e **algoritmo**;
- **[CATS! - clone cats]** materializa uma **tarefa** responsável por programar um evento que é acionado periodicamente, abordando **repetição**, **função** e **algoritmo**;
- **[CATS! - make the cats move]** materializa uma **tarefa** em que o gato realiza um caminho repetindo direções, desenvolvendo **repetição**, **função** e **algoritmo**;

⁵<https://projects.raspberrypi.org/en/projects/cats>

- **[CATS! - stick to the lines]** materializa uma **tarefa** onde deverá ser programado um evento que será disparado de acordo com uma condição e corrigir caminhos, desenvolvendo **condicional**, **avaliação** e **algoritmo**.

6.2 Disponibilização da Ontologia e Teste dos Cenários

Seguindo os moldes da arquitetura do DBpedia, mencionado na seção 2.3, está sendo disponibilizado um *endpoint* da ontologia desenvolvida ao decorrer da presente dissertação de mestrado. Esse *endpoint* é acessível a qualquer pessoa, de forma aberta, e permite reuso.

A ontologia foi desenvolvida usando o software Protégé e exportada no modelo RDF. O código correspondente, em formato RDF, foi importado para o OpenLink Virtuoso⁶ via *quad store upload*⁷, conforme a Figura 20. O endereço do *endpoint* publicado localiza-se no campo *named graph*⁸ IRI da Figura 20.

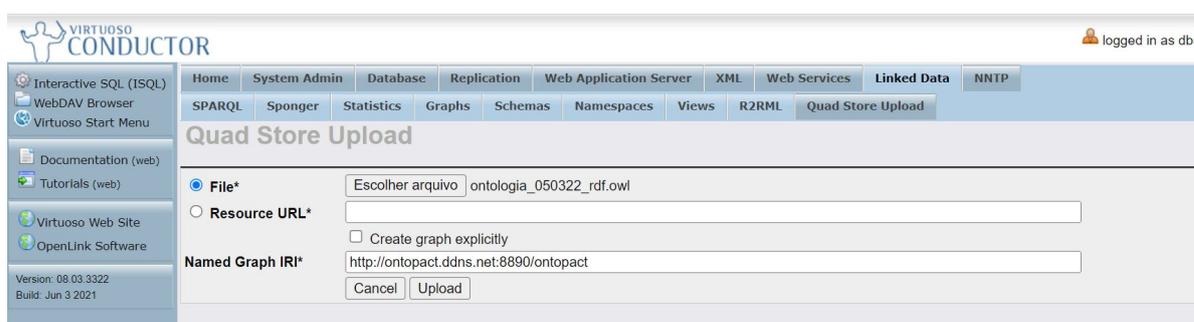


Figura 20 – Publicação da ontologia no endpoint

Inicialmente, no momento de instalação do Virtuoso, este começa a funcionar a partir do *localhost*⁹ da máquina em que foi instalado, usando a porta padrão do próprio Virtuoso, cujo número é 8890. Porém, como espera-se que o *endpoint* esteja acessível de qualquer local, foi necessário configurar o endereço IP da máquina na qual o Virtuoso está instalado como IP fixo, mapear a porta 8890 para seguir o *network address translation (NAT)*¹⁰ e habilitar o *demilitarized zone (DMZ)*¹¹ do IP interno da máquina na qual o Virtuoso foi instalado. Sendo assim, o Virtuoso, que anteriormente só poderia ser acessado na máquina local em que foi instalado através do endereço `http://localhost:8890`, agora é acessível de qualquer local da web utilizando o endereço `http://200.203.24.12:8890` ou simplesmente utilizando o domínio registrado `http://ontopact.ddns.net:8890`.

⁶<https://virtuoso.openlinksw.com/>

⁷Função do OpenLink Virtuoso que permite carregar arquivos OWL e RDF para serem realizadas consultas SPARQL

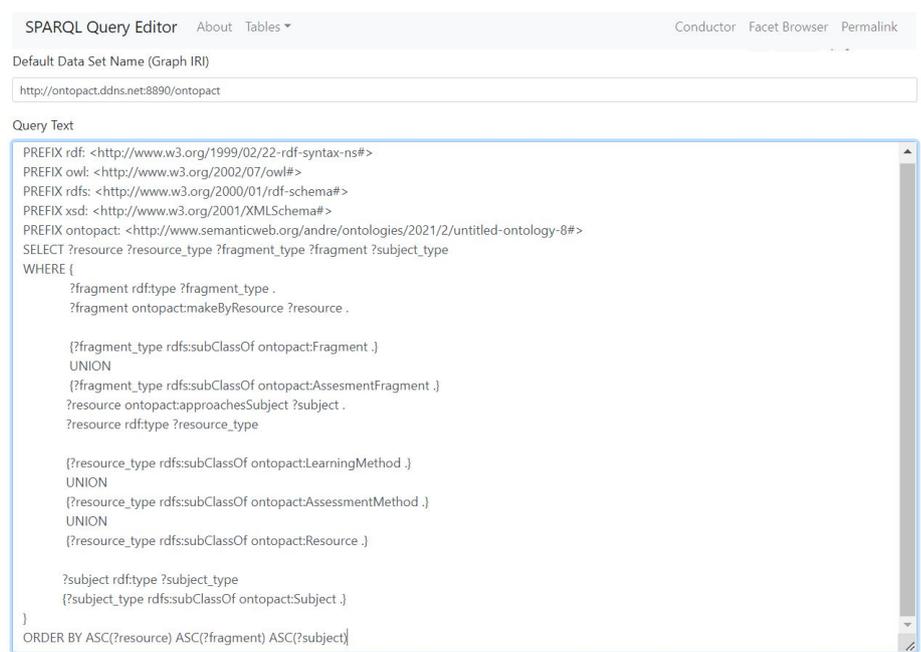
⁸Usado para colocar o *endpoint* que foi carregado para o Virtuoso

⁹Nome de host do dispositivo atual usado para acessá-lo

¹⁰Habilita redes privadas que usam endereços IP não registrados para se conectar à Internet, ou seja, todos os IPs de uma rede interna aparecem com o mesmo IP na rede externa

¹¹Usado para expôr serviços da rede interna para a internet

Para realizar consultas no *endpoint* público, basta acessar o IP:PORTA/sparql que, no caso deste projeto, ficou como <http://ontopact.ddns.net:8890/sparql>. Neste momento, é apresentada uma página conforme Figura 21, que permite fazer consultas SPARQL, então no campo *default data set name (graph IRI)* é necessário informar o *endpoint* público, no caso, <http://ontopact.ddns.net:8890/ontopact>, o qual foi definido no campo *named graph IRI* na Figura 20.



The screenshot shows the SPARQL Query Editor interface. At the top, there are navigation links: "SPARQL Query Editor", "About", "Tables", "Conductor", "Facet Browser", and "Permalink". Below this, there is a field for "Default Data Set Name (Graph IRI)" containing the URL "http://ontopact.ddns.net:8890/ontopact". The main area is titled "Query Text" and contains the following SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX ontopact: <http://www.semanticweb.org/andre/ontologies/2021/2/untitled-ontology-8#>
SELECT ?resource ?resource_type ?fragment_type ?fragment ?subject_type
WHERE {
  ?fragment rdf:type ?fragment_type .
  ?fragment ontopact:makeByResource ?resource .

  {?fragment_type rdfs:subClassOf ontopact:Fragment .}
  UNION
  {?fragment_type rdfs:subClassOf ontopact:AssesmentFragment .}
  ?resource ontopact:approachesSubject ?subject .
  ?resource rdf:type ?resource_type

  {?resource_type rdfs:subClassOf ontopact:LearningMethod .}
  UNION
  {?resource_type rdfs:subClassOf ontopact:AssesmentMethod .}
  UNION
  {?resource_type rdfs:subClassOf ontopact:Resource .}

  ?subject rdf:type ?subject_type
  {?subject_type rdfs:subClassOf ontopact:Subject .}
}
ORDER BY ASC(?resource) ASC(?fragment) ASC(?subject)
```

Figura 21 – Exemplo de consulta SPARQL submetida a ontologia desenvolvida

Para teste da ontologia de domínio foi realizada a consulta que está descrita na Figura 21, na qual são testados os três cenários presentes na seção 6.1. Com a consulta SPARQL é possível identificar que os recursos (CATS!, Scratch, CTt, CSU, Kidsbot) explicados na seção 6.1 estão associados a um tipo de recurso e são capazes de desenvolver uma série de alvos de aprendizagem. Esses recursos são compostos por uma série de fragmentos e um tipo de fragmento pode abordar um ou mais fragmentos. Cada fragmento é capaz de mapear quais alvos de aprendizagem eles desenvolvem.

O parte do resultado dessa consulta é possível ser visualizado na Figura 19 em forma de grafo e na Figura 22 em forma de texto.

resource	resource_type	fragment_type	fragment	subject_type
CATS!	LessonPlan	Step	CATS!-_clone_cats	Algorithm
CATS!	LessonPlan	Step	CATS!-_clone_cats	Conditional
CATS!	LessonPlan	Step	CATS!-_clone_cats	Evaluation
CATS!	LessonPlan	Step	CATS!-_clone_cats	Procedure_/Function
CATS!	LessonPlan	Step	CATS!-_clone_cats	Loop
CATS!	LessonPlan	Step	CATS!-_draw_lines	Algorithm
CATS!	LessonPlan	Step	CATS!-_draw_lines	Conditional
CATS!	LessonPlan	Step	CATS!-_draw_lines	Evaluation
CATS!	LessonPlan	Step	CATS!-_draw_lines	Procedure_/Function
CATS!	LessonPlan	Step	CATS!-_draw_lines	Loop
CATS!	LessonPlan	Step	CATS!-_make_the_cats_move	Algorithm
CATS!	LessonPlan	Step	CATS!-_make_the_cats_move	Conditional
CATS!	LessonPlan	Step	CATS!-_make_the_cats_move	Evaluation
CATS!	LessonPlan	Step	CATS!-_make_the_cats_move	Procedure_/Function
CATS!	LessonPlan	Step	CATS!-_make_the_cats_move	Loop
CATS!	LessonPlan	Step	CATS!-_stick_to_the_lines	Algorithm
CATS!	LessonPlan	Step	CATS!-_stick_to_the_lines	Conditional
CATS!	LessonPlan	Step	CATS!-_stick_to_the_lines	Evaluation
CATS!	LessonPlan	Step	CATS!-_stick_to_the_lines	Procedure_/Function
CATS!	LessonPlan	Step	CATS!-_stick_to_the_lines	Loop
CATS!	LessonPlan	Feature	CATS!_starter_project	Algorithm
CATS!	LessonPlan	Feature	CATS!_starter_project	Conditional
CATS!	LessonPlan	Feature	CATS!_starter_project	Evaluation
CATS!	LessonPlan	Feature	CATS!_starter_project	Procedure_/Function
CATS!	LessonPlan	Feature	CATS!_starter_project	Loop
CATS!	LessonPlan	Material	Computer	Algorithm
CATS!	LessonPlan	Material	Computer	Conditional
CATS!	LessonPlan	Material	Computer	Evaluation
CATS!	LessonPlan	Material	Computer	Procedure_/Function
CATS!	LessonPlan	Material	Computer	Loop

Figura 22 – Primeiros registros da consulta SPARQL

A Figura 22 mostra parte do resultado de uma consulta SPARQL realizada no *end-point* da ontologia, o qual está descrito nessa seção. A partir da Figura 22 é possível validar o cenário descrito na subseção 6.1.3. Observando as cinco primeiras linhas, as quais estão contornadas com um retângulo vermelho, é possível notar quais são todos cinco alvos de aprendizagem que o fragmento **CATS! clone cats** desenvolve e ele é um fragmento do tipo **passo** (*step*), também nota-se que o fragmento **CATS! clone cats** pertence ao recurso **CATS!** que é um recurso do tipo **plano de ensino** (*lesson plan*).

O teste dos cenários mostra que a ontologia de domínio é capaz de mapear recursos que devem ser usados para o desenvolvimento de alvos de aprendizagem e também sua relação inversa. Permite, ainda, mapear quais alvos de aprendizagem um determinado fragmento pode desenvolver.

7 CONCLUSÃO

Este trabalho apresentou um estudo sobre PC como uma aplicação em dados abertos conectados. Para seu entendimento foi necessário estudar os principais conceitos relacionados a ontologias e web semântica, assuntos tratados como base para representação de dados abertos conectados e também sobre PC.

As ontologias são modelos de dados capazes de representar qualquer coisa existente e, dentro da computação, são representadas por meio de linguagens processadas por máquina. Como linguagem, uma das mais utilizadas atualmente é a OWL, sendo bastante utilizado também o RDF como framework para padronização de dados na web. Também foi estudada a linguagem de consulta SPARQL, usada tanto no OWL quanto em RDF. A SPARQL é orientada a dados e só consulta informações presentes nos modelos.

A web semântica é uma implementação prática das ontologias, e os dados conectados estão presentes dentro da web semântica.

Sabendo do problema atual enfrentado pela comunidade de PC, relativo às dificuldades para definição consensual quanto aos significado dos termos e conjuntos de habilidades, identificou-se como esforço de pesquisa trazer o PC como uma aplicação em dados abertos conectados, a fim de padronizar os conjuntos de habilidades de PC para que pesquisadores, professores e entusiastas da área possam trabalhar de forma coerente e convergente. E disponibilizar essa padronização de forma aberta na web para que possa ser amplamente utilizada e estendível.

Para isso, neste trabalho foi proposto um modelo, na forma de ontologias, capaz de representar e integrar múltiplas perspectivas sobre PC a fim de padronizar conceitos e habilidades sobre o tema.

A ontologia de domínio desenvolvida também é capaz de vincular recursos de diferentes naturezas de acordo com suas finalidades e assuntos abordados. Ela está disponibilizada em forma de *endpoint* SPARQL.

Como forma de teste do modelo desenvolvido, foram propostos três cenários com visões de diferentes autores mostrando a expressividade, o alcance e a operação da ontologia de domínio. Com esses cenários o modelo revelou-se capaz de modelar

diferentes visões de autores, bem como vincular recursos de diferentes naturezas de acordo com a finalidade e assunto abordado. Também é possível adicionar mais ontologias de topo com visões de outros autores e instanciar outros cenários.

Como trabalhos futuros a serem desenvolvidos a partir deste, identificam-se os seguintes:

- Aumentar a descrição dos conceitos existentes a fim de utilizar a ontologia como forma de avaliar habilidades de PC, pois cada fragmento possui uma aptidão que é uma estimativa de sua dificuldade. Também classificar qual a melhor idade para aplicação de cada fragmento ou desenvolvimento de cada habilidade.
- Criar uma plataforma na qual seja possível inserir mais recursos ou até novos cenários na ontologia de domínio para que outros pesquisadores, professores e entusiastas possam compartilhar diferentes recursos para o desenvolvimento de determinada competência. Essa plataforma deverá mostrar a ontologia em forma gráfica na tela, em alguma linguagem de programação que permita ao usuário instanciar novos recursos e fazer consultas sem a necessidade de entender sobre SPARQL. A plataforma ainda seria responsável por atualizar a ontologia e mostrar os novos recursos inseridos pelos usuários.

REFERÊNCIAS

ABOUT. Disponível em: <<https://scratch.mit.edu/about>>. Acesso em: 2022-02-25.

ALCANTARA, W. et al. Desafios no uso de dados abertos conectados na educação brasileira. In: IV WORKSHOP DE DESAFIOS DA COMPUTAÇÃO APLICADA À EDUCAÇÃO, 2015. **Anais...** SBC, 2015. p.11–20.

ARAÚJO, C.; LIMA, L. V.; HENRIQUES, P. R. An Ontology based approach to teach Computational Thinking. In: INTERNATIONAL SYMPOSIUM ON COMPUTERS IN EDUCATION (SIIE), 2019., 2019. **Anais...** IEEE, 2019. p.1–6.

ARAÚJO, C.; TEIXEIRA, S.; BARBOSA, D.; HENRIQUES, P. R. MOVE: Measuring ontologies in value-seeking environments: Using Ontologies to plan Computer Programming courses for different levels. In: CONFERENCE COMPANION PUBLICATION OF THE 2020 ON COMPUTER SUPPORTED COOPERATIVE WORK AND SOCIAL COMPUTING, 2020. **Anais...** Association for Computing Machinery, 2020. p.475–482.

AZEVEDO, A. et al. Micas, a web platform to support teachers of computing at school. **Challenges**, Basel, Switzerland, p.625–641, 2019.

BANDEIRA, J. et al. Dados abertos conectados para a Educação. **Jornada de Atualização em Informática na Educação**, Maceió, v.4, n.1, p.47–69, 2015.

BELL, T.; ALEXANDER, J.; FREEMAN, I.; GRIMLEY, M. Computer science unplugged: School students doing real computing without computers. **The New Zealand Journal of Applied Computing and Information Technology**, New Zealand, v.13, n.1, p.20–29, 2009.

BERNERS-LEE, T. **Linked Data**. Disponível em: <<https://www.w3.org/DesignIssues/LinkedData>>. Acesso em: 2020-11-28.

BLIKSTEIN, P. Maker movement in education: History and prospects. **Handbook of Technology Education**, New York City, p.419–437, 2018.

BRASIL, O. K. **Open Data Index**. Disponível em: <<https://www.ok.org.br/projetos/open-data-index/>>. Acesso em: 2020-11-25.

BRENNAN, K.; RESNICK, M. New frameworks for studying and assessing the development of computational thinking. In: AMERICAN EDUCATIONAL RESEARCH ASSOCIATION, VANCOUVER, CANADA, 2012., 2012. **Proceedings...** American Educational Research Association, 2012. v.1, p.25.

CUNY, J.; SNYDER, L.; WING, J. M. Demystifying computational thinking for non-computer scientists. **Unpublished manuscript in progress, referenced in <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>**, New England, 2010.

DADOS ABERTOS, P. B. de. **Aplicativos e serviços que utilizam dados abertos**. Disponível em: <<https://dados.gov.br/pagina/aplicativos#tabs-5>>. Acesso em: 2020-12-14.

DBPEDIA. **DBpedia Architecture Diagram**. Disponível em: <<http://wikidata.dbpedia.org/openlink-software-%E2%80%94-virtuoso-universal-server>>. Acesso em: 2022-02-07.

DENNING, P. J.; TEDRE, M. **Computational thinking**. Cambridge, Massachusetts, United States: MIT Press, 2019.

DJURIĆ, D.; GAŠEVIĆ, D.; DEVEDŽIĆ, V.; DAMJANOVIĆ, V. A UML profile for OWL ontologies. In: **Model driven architecture**. New York City: Springer, 2004. p.204–219.

GAVA, T. B. S.; MENEZES, C. S. de. Uma ontologia de domínio para a aprendizagem cooperativa. In: BRAZILIAN SYMPOSIUM ON COMPUTERS IN EDUCATION (SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO-SBIE), 2003. **Anais...** SBIE, 2003. v.1, n.1, p.336–345.

GLUZ, J. C.; VICARI, R. M. An OWL ontology for IEEE-LOM and OBAA metadata. In: INTERNATIONAL CONFERENCE ON INTELLIGENT TUTORING SYSTEMS, 2012, New York City. **Anais...** Springer, 2012. p.691–693.

GROUP, O. W. **Web Ontology Language (OWL)**. Disponível em: <<https://www.w3.org/OWL/>>. Acesso em: 2020-12-14.

GRUBER, T. R. A translation approach to portable ontology specifications. **Knowledge acquisition**, Amsterdam, v.5, n.2, p.199–220, 1993.

GUARINO, N. **Formal ontology in information systems**: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy. Amsterdam: IOS press, 1998. v.46.

GUARINO, N.; OBERLE, D.; STAAB, S. What is an ontology? In: **Handbook on ontologies**. New York City: Springer, 2009. p.1–17.

HAUSENBLAS, M. **5 Stars dos Dados Abertos**. Disponível em: <<https://5stardata.info/pt-BR/>>. Acesso em: 2020-11-28.

ISOTANI, S.; BITTENCOURT, I. I. **Dados Abertos Conectados**: Em busca da Web do Conhecimento. São Paulo: Novatec Editora, 2015.

JAMES, L. **Defining Open Data**. Disponível em: <<https://blog.okfn.org/2013/10/03/defining-open-data/>>. Acesso em: 2020-11-25.

KIRYAKOV, A.; OGNJANOV, D.; MANOV, D. OWLIM—a pragmatic semantic repository for OWL. In: INTERNATIONAL CONFERENCE ON WEB INFORMATION SYSTEMS ENGINEERING, 2005, New York City. **Anais...** Springer, 2005. p.182–192.

LAUFER, C. **Guia de Web Semântica**. Disponível em: <<https://ceweb.br/guias/web-semantica/capitulo-4/>>. Acesso em: 2020-11-16.

LÍTERIS. **Descomplicando a Gestão do Conhecimento**. Disponível em: <<https://literis.com.br/blog/descomplicando-a-gestao-do-conhecimento/>>. Acesso em: 2020-11-16.

MCCRAE, J. P. et al. The linked open data cloud. **Lod-cloud.net**, Online, 2019.

PROFLAB. **Infográficos**. Disponível em: <<https://www.souproflab.com.br/recursos/infograficos/>>. Acesso em: 2020-12-07.

PROJECT, W. S. C. **Linking Open Data**. Disponível em: <<https://www.w3.org/wiki/SweolG/TaskForces/CommunityProjects/LinkingOpenData/literis.com.br/a-gestao-do-conhecimento/>>. Acesso em: 2020-11-26.

PRUD'HOMMEAUX, E.; SEABORNE, A. **SPARQL Query Language for RDF**. Disponível em: <<https://www.w3.org/TR/rdf-sparql-query/>>. Acesso em: 2020-11-28.

RESNICK, M. et al. Scratch: programming for all. **Communications of the ACM**, New York, v.52, n.11, p.60–67, 2009.

REVOLUTION, U. D. **A world that counts**: Mobilising the data revolution for sustainable development by ECLAC in Latin America and the Caribbean. Santiago, Chile: ECLAC, 2014.

ROMÁN-GONZÁLEZ, M.; PÉREZ-GONZÁLEZ, J.-C.; JIMÉNEZ-FERNÁNDEZ, C. Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. **Computers in human behavior**, Amsterdam, v.72, p.678–691, 2017.

SÁ, F. E. de. **Disciplina:** Introdução a Programação. Disponível em: <<http://estacio.webaula.com.br/cursos/go0067/aula1.html>>. Acesso em: 2020-12-07.

SEMANTICS, C. **RDFS vs. Owl.** Disponível em: <<https://www.cambridgesemantics.com/blog/semantic-university/learn-owl-rdfs/rdfs-vs-owl/>>. Acesso em: 2020-11-03.

SIKOS, L. **Mastering structured data on the Semantic Web:** From HTML5 microdata to linked open data. New York City: Apress, 2015.

SILVA, B. A. da et al. PACT: A Citizen Science Project for Computing Education. In: IEEE FRONTIERS IN EDUCATION CONFERENCE (FIE), 2021., 2021. **Anais...** IEEE, 2021. p.1–9.

SILVA JÚNIOR, B. A. d. **GGasCT:** bringing formal methods to the computational thinking. 2020. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal de Pelotas.

SILVA, W. D. d.; PARREIRAS, F. S.; MAIA, L. C. G.; BRANDÃO, W. C. Anotação semântica automática do currículo Lattes utilizando Linked Open Data. **Perspectivas em Ciência da Informação**, São Paulo, v.23, n.4, p.53–72, 2018.

ULAS, I.; HASESKI, H. İ.; TUGTEKIN, U. Publication trends over 10 years of computational thinking research. **Contemporary Educational Technology**, Turkey, v.9, n.2, p.131–153, 2018.

UNPLUGGED, C. S. **Computer Science without a computer.** Disponível em: <<https://csunplugged.org/en/>>. Acesso em: 2021-05-17.

UYI IDEHEN, K. **What is the Linked Open Data Cloud, and why is it important?** Disponível em: <<https://medium.com/virtuoso-blog/what-is-the-linked-open-data-cloud-and-why-is-it-important-1901a7cb7b1f>>. Acesso em: 2021-11-01.

VICARI, R. M. et al. Proposta brasileira de metadados para objetos de aprendizagem baseados em agentes (obaa). **RENOTE: revista novas tecnologias na educacao [recurso eletronico]**. Porto Alegre, RS, Porto Alegre, 2010.

WING, J. Research notebook: Computational thinking — What and why. **The link magazine**, Greenville, v.6, p.20–23, 2011.

WING, J. M. Computational thinking. **Communications of the ACM**, New York, v.49, n.3, p.33–35, 2006.