

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

ANDRÉ ROLIM BEHR

**Uma Abordagem para Promover Reuso e
Processamento de Inferências em
Ontologias de Metadados Educacionais**

Dissertação apresentada como requisito parcial
para a obtenção do grau de Mestre em Ciência da
Computação

Orientador: Prof^a. Dr^a. Rosa Vicari

Porto Alegre
2016

CIP — CATALOGAÇÃO NA PUBLICAÇÃO

Behr, André Rolim

Uma Abordagem para Promover Reuso e Processamento de Inferências em Ontologias de Metadados Educacionais / André Rolim Behr. – Porto Alegre: PPGC da UFRGS, 2016.

89 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2016. Orientador: Rosa Vicari.

1. Web Semântica, ontologias, padrões de metadados educacionais. I. Vicari, Rosa. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Rui Vicente Oppermann

Vice-Reitora: Prof^a. Jane Fraga Tutikian

Pró-Reitor de Pós-Graduação: Prof. Celso Giannetti Loureiro Chaves

Diretora do Instituto de Informática: Prof^a. Carla Maria Dal Sasso Freitas

Coordenador do PPGC: Prof. João Luiz Dihl Comba

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

*“Somos o que fazemos repetidamente.
Por isso o mérito não está na ação e sim no hábito”*

— ARISTÓTELES

AGRADECIMENTOS

Agradeço à minha mãe por estar presente em todos os momentos e apoio incondicional, ao meu padrasto que já vem sendo um pai por muitos anos, à minha vó Gessy por sempre me proporcionar uma vida melhor e ao meu vô José (in memoriam) pelo exemplo.

À minha namorada, Lana, que vivenciou todos os altos e baixos deste trabalho, pelo companheirismo e amor. Também à família dela pelo acolhimento e carinho.

Ao Tiago Primo pelas dicas e co-autoria em artigos ao longo do caminho.

Aos colegas e chefias do CPD pela compreensão e apoio para o desenvolvimento desta dissertação.

À Prof^a Rosa pelo incentivo em continuar os estudos no mestrado e orientação.

A todos que direta ou indiretamente contribuíram para que este trabalho fosse realizado.

RESUMO

Metadados vêm sendo utilizados amplamente para descrever objetos de aprendizagem na Web. Contudo, mesmo que a adoção de um único padrão de metadados pudesse assegurar a reusabilidade de recursos e interoperabilidade entre aplicações, ainda não existe um esquema de metadados que seja apropriado para preencher todos os requisitos de cada aplicação. Com isso, a criação de novos padrões de metadados e perfis de aplicação torna-se constante com o passar dos anos. Atualmente, a Web está sendo estendida pela Web Semântica de forma sistemática. A integração dos seus dados vem sendo obtida em grande parte pela adoção de ontologias. A presente dissertação propõe uma abordagem de representação de conhecimento em três camadas baseada em ontologias para metadados educacionais. Esta abordagem é composta de ontologias modulares com intuito de aumentar o reuso e otimizar o processamento de inferências dos metadados. Além disto, é proposto um método de interoperabilidade entre metadados descritos em XML e OWL para a proposta de ontologias modulares. Os resultados apresentaram ganhos quanto ao uso de ontologias modulares e verificações de cardinalidades em mundo fechado. As ontologias propostas apresentam uma representação de conhecimento de forma unificada e são compatíveis com as tecnologias atuais da Web Semântica.

Palavras-chave: Web Semântica, ontologias, padrões de metadados educacionais.

An Approach to Improve Reuse and Inference Processing in Educational Metadata Ontologies

ABSTRACT

Metadata has been broadly employed to describe learning objects on the Web. However, even with the adoption of a unique metadata standard could secure reusability of resources and interoperability among applications, there is no metadata schema that would be enough to comply with all requirements of each application yet. Therewith, the creation of new metadata standards and profile applications are regularly over the years. Nowadays, the Semantic Web is extending the Web systematically. The integration of data has been activated mostly by the adoption of ontologies. This dissertation proposes an approach to knowledge representation in three layers based on ontologies for educational metadata. This approach is composed of modular ontologies that aim improve reuse and optimize the inference processing of metadata. Beyond that, an interoperability method is proposed to metadata described in XML and OWL to the modularized ontologies. The results show some optimization in using modular ontologies and cardinality verification in a closed world. The proposed ontologies unify the knowledge representation and they are compatible with the current Semantic Web technologies.

Keywords: Semantic Web, ontology, educational metadata standards.

LISTA DE ABREVIATURAS E SIGLAS

ABox	Assertional Box
API	Application Programming Interface
BIOE	Banco Internacional de Objetos Educacionais
BLD	Basic Logic Dialect
CPU	Central Processing Unit
DB	DataBase
DC	Dublin Core
DCMI	Dublin Core Metadata Initiative
DL	Description Logic
DTD	Document Type Definition
DTV	Digital TeleVision
EL	Existential Language
EMP	Educational Metadata Profile
ES	Engenharia de Software
EXCO	Efficient XML Converter to OWL
FEB	Federação de Repositórios Educa Brasil
GB	GigaBytes
GHZ	GigaHertz
HTTP	Hypertext Transfer Protocol
IA	Inteligência Artificial
IEEE	Institute of Electrical and Electronics Engineers
IHC	Interação Humano-Computador
IRI	Internationalized Resource Identifiers
JUNG	Java Universal Network/Graph framework

LOM	Learning Object Metadata
LD	Learning Design
MS	Milissegundos
OA	Objeto de Aprendizagem
OAI	Open Archives Initiative
OBAA	OBjetos de Aprendizagem baseados em Agentes
OOLO	Ontology for Organizational Learning Objects
OWL	Web Ontology Language
PHM	Protocol for Metadata Harvesting
PRD	Production Rule Dialect
QL	Query Language
QM	Quad-core Mobile
RAM	Random Access Memory
RBox	Reasoning Box
RDF	Resource Description Framework
RDF-S	RDF-Schema
REST	REpresentational State Transfer
RIF	Rule Interchange Format
RL	Rule Language
RNP	Rede Nacional de Pesquisa
SKOS	Simple Knowledge Organization System
SQL	Structured Query Language
SQWRL	Semantic Query-Enhanced Web Rule Language
SWRL	Semantic Web Rule Language
TBox	Terminological Box
UNA	Unique Name Assumption

URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
W3C	World Wide Web Consortium
WS	Web Semântica
XML	eXtensible Markup Language
XSD	XML Schema Definition
XSLT	eXtensible Stylesheet Language Transformations
XSOM	XML Schema Object Model

LISTA DE FIGURAS

Figura 1.1	Etapas Práticas de Pesquisa	17
Figura 3.1	Classificação de ontologias baseado no nível de abstração	27
Figura 3.2	Um módulo importando n módulos	29
Figura 3.3	N módulos importando um módulo	30
Figura 3.4	N módulos importando n-1 módulos	30
Figura 3.5	Combinação dos padrões 1, 2 e 3	31
Figura 3.6	Visão geral do processo de alinhamento de ontologias	32
Figura 4.1	Linguagens para a Web Semântica	34
Figura 4.2	Estado atual da Web Semântica	35
Figura 4.3	Perfis e expressividades em OWL 1 e OWL 2.....	37
Figura 6.1	Camadas para representação de conhecimento.....	48
Figura 6.2	Hierarquia de classes das ontologias General e Identifier.	51
Figura 6.3	Controle de valores condicionais para o metadado Name	52
Figura 6.4	Relacionamento entre indivíduos do OA Brasília.	53
Figura 6.5	Relacionamento entre indivíduos do OA Klimadiagramm von Athen.....	54
Figura 6.6	Relacionamento entre indivíduos do OA Geometria Descritiva em Contos.	55
Figura 6.7	Relacionamento entre indivíduos do OA Infraestrutura do Serviço Videoaula@RNP: Módulo 2.....	56
Figura 6.8	Classe de equivalência para OBAA-LITE	57
Figura 6.9	Conjunto de regras para OBAA-LITE	57
Figura 6.10	Classe de equivalência para OBAA-VideoAula	58
Figura 6.11	Conjunto de regras para OBAA-VideoAula	59
Figura 6.12	Regra para verificação de cardinalidade do metadado Keyword da categoria General.....	59
Figura 6.13	Composição de ontologia de metadados a partir de diferentes fontes na Web Semântica.	60
Figura 6.14	Reuso do indivíduo Meta-Metadado Contribute de ontologia externa.	61
Figura 6.15	Classificação por consulta SPARQL.....	62
Figura 7.1	Abordagem de Extração-Transformação-Carga para prover interoperabilidade de metadados na Web Semântica	64
Figura 7.2	Consulta OAI-PMH para recuperar uma lista de metadados em XML.....	64
Figura 7.3	Consulta SPARQL para recuperar uma lista de URIs de ontologias	64
Figura 7.4	XML do Alkaline Fuel Cell LifeCycle (SILVA, 2011) transposto para triplas.....	66
Figura 7.5	Equivalências das propriedades de dados para os metadados version e status.....	66
Figura 7.6	Exemplo de equivalências de triplas entre ontologias OBAA.....	67
Figura 7.7	Relacionamento entre indivíduos do OA Brasília.	67
Figura 7.8	Indexação de triplas dos metadados do OA Brasília.	68
Figura 8.1	Relação entre o tempo de reasoning e número de propriedades general.Keyword.....	72

LISTA DE TABELAS

Tabela 3.1 Mundo aberto x mundo fechado: associações hipotéticas entre pessoas e países por onde viajaram	25
Tabela 5.1 Sumário dos Trabalhos Relacionados.....	46
Tabela 6.1 Grupo General do IEEE-LOM	50
Tabela 8.1 Avaliação de desempenho em ms de ontologias do nível um e dois	71
Tabela 8.2 Avaliação de desempenho em ms de ontologias no nível três	72
Tabela A.1 Mapeamento de metadados entre Dublin Core e IEEE-LOM	83
Tabela B.1 Metadados do perfil de aplicação OBAA-LITE	84
Tabela B.2 Metadados do perfil de aplicação OBAA-VideoAula – Parte 1.....	85
Tabela B.3 Metadados do perfil de aplicação OBAA-VideoAula – Parte 2.....	86

SUMÁRIO

1 INTRODUÇÃO	14
1.1 Motivação.....	15
1.2 Objetivos e Contribuições	16
1.3 Organização do Texto	17
2 PADRÕES DE METADADOS EDUCACIONAIS	18
2.1 Dublin Core	18
2.2 IEEE-LOM	19
2.3 OBAA	20
2.4 VideoAula@RNP.....	21
2.5 Perfis de Aplicação	21
2.6 Repositórios	22
2.7 Considerações do Capítulo.....	23
3 ONTOLOGIA	24
3.1 Suposição de Mundo	25
3.2 Tipos	26
3.3 Metodologias para Ontologia.....	27
3.4 Modularização.....	28
3.5 Alinhamento	31
3.6 Considerações do Capítulo.....	32
4 WEB SEMÂNTICA	33
4.1 Tecnologias.....	34
4.2 Web Ontology Language	36
4.3 Semantic Query-enhanced Web Rule Language.....	37
4.4 Considerações do Capítulo.....	38
5 TRABALHOS RELACIONADOS	39
5.1 Estado da Arte.....	39
5.2 Sumário dos Trabalhos.....	46
5.3 Considerações do Capítulo.....	46
6 REPRESENTAÇÃO DE CONHECIMENTO	48
6.1 Camada de Descrição	48
6.2 Camada de Instanciação.....	52
6.2.1 Brasília	53
6.2.2 Klimadiagramm von Athen.....	54
6.2.3 Geometria Descritiva em Contos	54
6.2.4 Infraestrutura do Serviço Videoaula@RNP: Módulo 2	55
6.3 Camada de Raciocínio	55
6.3.1 OBAA-LITE	56
6.3.2 OBAA-VideoAula.....	57
6.3.3 Cardinalidades.....	59
6.4 Considerações do Capítulo.....	60
7 MÉTODO PARA INTEROPERABILIDADE	63
7.1 Recuperação de Metadados	64
7.2 Processamento de Metadados	65
7.2.1 XML para OWL.....	65
7.2.2 OWL para OWL.....	65
7.2.3 Dublin Core para OBAA.....	66
7.3 Armazenamento de Metadados	68
7.4 Considerações do Capítulo.....	69

8 RESULTADOS	70
8.1 Verificação de Consistência	70
8.2 Classificação.....	71
8.3 Interoperabilidade	72
8.4 Considerações do Capítulo.....	73
9 CONCLUSÃO	74
REFERÊNCIAS	77
APÊNDICE A — MAPEAMENTO IEEE-LOM PARA CONJUNTO NÃO QUALIFICADO DE METADADOS DUBLIN CORE	83
APÊNDICE B — PERFIS DE APLICAÇÃO	84
APÊNDICE C — LISTA DE ONTOLOGIAS DESENVOLVIDAS	87

1 INTRODUÇÃO

Atualmente, dados são criados e consumidos das mais diversas maneiras. Conteúdos em redes sociais, sites de notícias, ambientes de aprendizagem, dentre outros, são disponibilizados na Internet¹. Contudo, estes dados estão na maioria das vezes em forma bruta e não categorizada, havendo uma sobrecarga de dados e uma subutilização de informação relevante.

Geralmente, o conteúdo das páginas Web é percebido pelas máquinas apenas sintaticamente. A linguagem das páginas somente é percebida pelos computadores de forma semântica. A interpretação da informação, muitas vezes, acaba sendo realizada pelas pessoas que visualizam as páginas.

O gerenciamento de grandes volumes de dados, oriundos de localidades e formatos heterogêneos, é uma das qualidades mais desejadas da última década (PATIL; DAVENPORT, 2012). Quando estes dados estão em forma estruturada e conectada, eles são passíveis de processamento por computador, de forma que diferentes tarefas podem ser automatizadas (BARROS et al., 2011).

Desta forma, é importante que dados sejam providos em alta qualidade, de modo que tanto humanos quanto máquinas possam ler informações, reutilizá-las e combiná-las de forma a gerar novos conhecimentos. Atualmente, dados são descritos na maioria das vezes por metadados na Web em formato XML, almejando facilitar buscas e reuso sobre eles.

A Web Semântica vem se consolidando, através da adoção de tecnologias que estão fundamentando conceitos discutidos pela comunidade, para que os dados sejam tratados de maneira mais produtiva. Portanto, ambientes e ferramentas compatíveis com esta nova abordagem se estabelecem para gerenciar dados mais ricos e complexos.

Neste cenário, as informações disponíveis são segmentadas por nós, passíveis de serem acessados individualmente. Portanto, granularidades mais finas de dados podem ser recuperadas e agrupadas de diferentes maneiras.

No caso da Educação, recursos podem ser descritos de forma mais rica, através de ontologias. Deste modo, agentes e ambientes inteligentes conseguem adotar a melhor ação possível diante de uma situação, tal como recomendar um material personalizado para estudo de determinado usuário de acordo com suas características.

¹O website Qmee (<<http://blog.qmee.com/qmee-online-in-60-seconds/>>) rastreou a Web durante o ano de 2013 e apontou que em 60 segundos ocorrem 2 milhões de buscas, 571 criações de websites, 72 horas de vídeos carregados no Youtube, 204 milhões de e-mails enviados, dentre outros.

Além disto, uma larga variedade de sistemas inteligentes pode se beneficiar de representações ontológicas. Pode-se citar um framework de integração de recursos educacionais baseado em ontologias que objetiva aumentar a utilização de recursos (XIONG; LIU; LIU, 2014); abordagem web semântica para recomendar objetos de aprendizagem (PRIMO; BEHR; VICARI, 2013); sistema de recomendação de objetos de aprendizagem multiagente que utiliza um modelo ontológico para entregar conteúdos educacionais adaptados (RODRÍGUEZ et al., 2014); sistema tutor inteligente (LALINGKAR; RAMNATHAN; RAMANI, 2014); wiki semântico (ARAPOV; BUFFA; OTHMANE, 2014); dentre muitos outros.

1.1 Motivação

Com a extensão da Web pela Web Semântica, padrões de metadados educacionais vêm sendo migrados de formatos baseados em XML para formatos semânticos por meio de ontologias (AL-KHALIFA; DAVIS, 2006). Breitman (2005) aponta que entre as maiores razões para se construir uma ontologia estão o compartilhamento de informação e a possibilidade de reuso do conhecimento sobre domínios específicos.

Além disso, a aplicação de raciocínio automático sobre ontologias é um dos pilares para sistemas mais complexos. Contudo, motores de inferência quando aplicados a um grande volume de indivíduos, tendem a perder desempenho. A realização de experimentos em ambientes repletos de indivíduos, relacionamentos e axiomas de inferência se caracteriza como um desafio futuro (PRIMO, 2013). Por enquanto, a escalabilidade é inversamente proporcional a funcionalidade. A investigação destes objetivos conflituosos, de modo que se obtenha escalabilidade e alta funcionalidade levará a novas direções de pesquisa (MIZOGUCHI, 2012).

Isotani e Costa (2009) citam que afora alguns sistemas educacionais utilizem padronizações de linguagens para formatar cursos e OAs, o compartilhamento e interoperabilidade de informações são comprometidos, de certa forma, pela arquitetura destes sistemas. Informações detalhadas sobre o uso dos cursos e/ou dos OAs não podem ser compartilhados quando, por exemplo, um curso é representado em IMS-LD e os OAs estejam em IEEE-LOM. As ontologias são determinantes para garantir a interoperabilidade e o uso das informações contidas em sistemas educacionais. Contudo, a criação de ontologias é um processo complexo e demorado. Deste modo, a Web Semântica tem encontrado dificuldades em ser adotada em cenários fora da academia.

Por outro lado, é importante que uma descrição semântica seja realizada ao longo do ciclo de vida do OA, ou seja, um modelo ontológico para acompanhar o OA desde a sua criação, uso e manutenção. Ontologias de metadados de OAs são parte relevante na estrutura de projetos (GLUZ; VICARI, 2014).

Diferentes trabalhos tendem a criar suas próprias representações ontológicas de padrões de metadados educacionais pertinentes ao seu contexto. É recorrente a criação de perfis de aplicação de metadados para uso em aplicações (GASCUEÑA; FERNANDEZ-CABALLERO; GONZALEZ, 2006), (MACHADO; OLIVEIRA, 2014), (BEHR; PRIMO; VICARI, 2014), (MILUTINOVIĆ et al., 2016), (SILVA, 2016).

O compartilhamento de uma mesma ontologia pelas aplicações é praticamente impossível. Deste modo, é importante o estudo de como diferentes descrições ontológicas de metadados possam ser reutilizadas e integradas no contexto da Web Semântica.

1.2 Objetivos e Contribuições

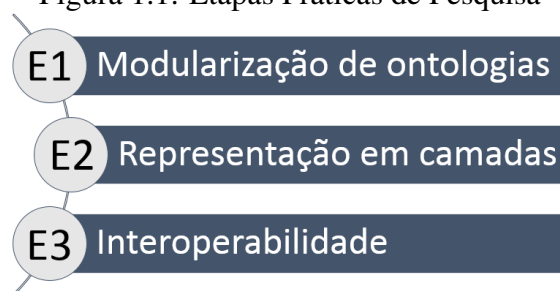
A presente dissertação tem como objetivo analisar descrições e representações de metadados de OAs descritos como ontologias e como motores de inferência se comportam para tarefas como verificação de consistência e classificação. Para auxiliar no desenvolvimento da pesquisa, as seguintes questões foram levantadas:

Questão de Pesquisa 1 (Q1): Fragmentar uma ontologia em módulos pode aumentar o reuso e otimizar o desempenho do motor de inferências?

Questão de Pesquisa 2 (Q2): Como diferentes descrições de metadados educacionais podem interoperar na Web Semântica?

A partir destas questões de pesquisa, organizou-se as etapas de trabalho prático conforme a Figura 1.1, onde se pretende estabelecer as contribuições do trabalho. Após uma análise do estado da arte de ontologias que representam metadados educacionais, pretende-se desenvolver ontologias modulares para os metadados (E1), com o intuito de aumentar o reuso e diminuir a quantidade de axiomas para a realização de inferências. Em seguida, reutilizar-se destas descrições para representar em camadas, com aumento de expressividade, descrições de metadados e analisar como se comportam em classificações a partir do motor de inferências (E2). Por fim, estabelecer um método simples para estabelecer interoperabilidade com diferentes ontologias educacionais (E3).

Figura 1.1: Etapas Práticas de Pesquisa



Fonte: Elaborado pelo autor

1.3 Organização do Texto

A disposição dos capítulos segue apresentando temas fundamentais para o entendimento da dissertação, como Metadados Educacionais, Ontologias e Web Semântica, nos Capítulos 2, 3 e 4, respectivamente. Trabalhos relacionados a presente dissertação são discutidos no Capítulo 5. Uma abordagem ontológica de representação de conhecimento é apresentada no Capítulo 6. Em seguida, um método para interoperar ontologias de metadados educacionais é sintetizada no Capítulo 7. Por fim, os resultados são apresentados no Capítulo 8. As conclusões e trabalhos futuros são discutidos no Capítulo 9. Complementarmente, o mapeamento entre IEEE-LOM e Dublin Core é ilustrado no Apêndice A. Os metadados dos perfis de aplicação utilizados são descritos no Apêndice B. Uma lista das ontologias desenvolvidas para padrões de metadados é apresentada no Apêndice C.

2 PADRÕES DE METADADOS EDUCACIONAIS

Objetos de aprendizagem (OAs) são considerados um dos principais meios para a obtenção de um ensino de qualidade. A escolha de um OA é preponderante para a disseminação correta de conhecimento (TAROUÇO et al., 2014).

Metadados são utilizados para descrever dados na Web, com o objetivo de promover e apoiar compartilhamento e reuso (TANI; CANDELA; CASTELLI, 2013). O desenvolvimento de padrões de metadados para recursos educacionais tem sido um esforço para aprimorar a descrição destes tipos de conteúdo na Web. Dublin Core e IEEE-LOM são iniciativas amplamente conhecidas para o contexto educacional, mais precisamente para descrever OAs (PALAVITSINIS; MANOUSELIS; SANCHEZ-ALONSO, 2014).

Um objeto de aprendizagem é definido como uma entidade, sendo digital ou física, que possa ser utilizada para ensino, educação ou treinamento (LTSC, 2002). A representação e o tratamento de OAs necessitam ser considerados de forma epistêmica (GLUZ; VICARI, 2014).

A seguir, os padrões de metadados Dublin Core e IEEE-LOM serão descritos. Além destes, os padrões de metadados OBAA e VideoAula@RNP também serão apresentados por se tratarem do contexto educacional brasileiro.

2.1 Dublin Core

A Dublin Core Metadata Initiative (DCMI) é responsável pelo gerenciamento do padrão de metadados Dublin Core (DC). Este padrão é considerado genérico para a descrição de recursos, mas também é utilizado no contexto educacional. A sua simplicidade contribuiu para que seja amplamente adotado por comunidades educacionais digitais. É composto por um conjunto de quinze metadados dispostos de forma plana (DCMI, 2008):

- Contributor: Uma entidade responsável por realizar contribuições para o recurso;
- Coverage: Tópico espacial ou temporal do recurso. A aplicabilidade espacial do recurso ou a jurisdição sob a qual o recurso é relevante;
- Creator: Uma entidade essencialmente responsável por confeccionar o recurso;
- Date: Um ponto ou período de tempo associado com um evento no ciclo de vida do recurso;
- Description: Um relato do recurso;

- **Format:** O formato do arquivo, meio físico ou dimensões do recurso;
- **Identifier:** Uma referência que não seja ambígua para o recurso dentro de um contexto;
- **Language:** Uma língua do recurso;
- **Publisher:** Uma entidade responsável por confeccionar o recurso disponível;
- **Relation:** Um recurso relacionado;
- **Rights:** Informação sobre direitos detidos e sobre o recurso;
- **Source:** Um recurso relacionado a partir do qual o recurso descrito é derivado;
- **Subject:** O tópico do recurso;
- **Title:** Um nome dado ao recurso;
- **Type:** A natureza ou gênero do recurso.

2.2 IEEE-LOM

O Institute of Electrical and Electronics Engineers (IEEE) gerencia o padrão de metadados Learning Object Metadata (LOM). O padrão é específico para a descrição de objetos de aprendizagem. Diferentemente do DC, o IEEE-LOM é disposto de forma hierárquica. No último nível, elementos simples de dados, tais como *Catalog*, *Entry*, *Title*, *Language*, *Description*, dentre outros. Um nível acima, os metadados agregados: *Identifier*, *Contribute*, *Requirement*, *OrComposite*, *Resource*, *TaxonPath* e *Taxon*. Por fim, em um primeiro nível, as categorias principais (LTSC, 2002):

- **General:** Esta categoria agrupa a informação geral que descreve o OA como um todo;
- **LifeCycle:** Categoria que retrata o histórico e estado corrente deste OA e daquelas entidades que afetaram o OA durante a sua evolução;
- **Meta-Metadata:** Descreve o próprio registro de metadados. Por exemplo, como os metadados instanciados podem ser identificados, quem criou a instância de metadados, etc;
- **Technical:** Categoria responsável por elencar os requisitos técnicos e características do OA;
- **Educational:** A categoria que representa as características educacionais ou pedagógicas-chaves do OA;

- Rights: Esta categoria descreve os direitos de propriedade intelectual e as condições de uso para o OA;
- Relation: Define a relação entre o OA e outro OA. Em caso de múltiplas relações, devem existir múltiplas instâncias desta categoria;
- Annotation: Categoria que provê comentários do uso educacional do OA e informação de quando e por quem os comentários foram criados;
- Classification: Descreve onde o OA se encaixa em um sistema de classificação. Para múltiplas classificações, haverá múltiplas instâncias.

2.3 OBAA

O padrão brasileiro OBAA de metadados (VICARI et al., 2010) foi especificado para atender às necessidades do país. Ele realiza uma extensão dos metadados do IEEE-LOM com o objetivo de manter interoperabilidade com Web, dispositivos móveis e TV digital. Este amplo escopo de cobertura permite descrever diversos tipos de objetos de aprendizagem. Por exemplo, vídeos segmentados e objetos projetados para pessoas com algum tipo de inaptidão podem ser descritos pelo padrão.

As categorias Technical e Educational foram estendidas. Além disso, metadados de acessibilidade e segmentação foram adicionados. Os metadados em questão são:

- Technical:
 - SupportedPlatforms: Plataformas suportadas pelo OA. Atualmente, elas podem ser: Web, DTV e Mobile;
 - PlatformSpecificFeatures: Conjunto de características técnicas das mídias específicas desenvolvidas para cada plataforma na qual o OA foi previsto;
 - Service: Container para a especificação de serviços, ontologias, linguagem e protocolos de interação relacionados ao OA;
- Educational:
 - LearningContentType: Especificação educacional do tipo de conteúdo do OA;
 - Interaction: Especifica a interação educacional proposta pelo OA e usuário(s);
 - DidacticStrategy: Conjunto de ações planejadas e conduzidas pelo professor a fim de promover o envolvimento e comprometimento dos alunos com um conjunto maior de atividades;
- Accessibility: Habilidade do OA de se adaptar às necessidades do usuário. É defi-

nido pelo padrão IMS AccessForAll (IMS, 2004).

- Segmentation: Conjunto de informações para segmentação de OAs e grupos de segmentos de OAs.

2.4 VideoAula@RNP

A visualização de vídeos também se mostra como um importante meio para aprendizagem. Por exemplo, estudantes da Geração Z (nascidos a partir de 1995) empregam vídeos do Youtube¹ como forma preferida para aprender novas informações. A natureza multimídia dos vídeos permite aos estudantes ir além do que simplesmente ler sobre um conceito, mas testemunhá-lo e até experimentá-lo virtualmente. Além disso, são utilizados como um suplemento de aulas acadêmicas (SEEMILLER; GRACE, 2016).

A proposta de metadados para videoaulas realiza uma extensão sobre o padrão de metadados OBAA. Estas videoaulas são compostas por múltiplas mídias sincronizadas (PRIMO, 2013). Metadados foram adicionados as categorias Technical e Educational:

- Technical:
 - RelatedMedia: mídias relacionados com a videoaula;
 - Thumbnail - miniatura da videoaula;
- Educational:
 - Course: dados do curso associado à videoaula;
 - Program: dados do programa associado à videoaula.

2.5 Perfis de Aplicação

Diferentes atividades têm direcionado comunidades educacionais a adaptar esquemas de metadados existentes para a adequação de suas próprias necessidades (FRIESEN; MASON; WARD, 2002). Aumentar a interoperabilidade semântica das instâncias resultantes de metadados dentro de uma comunidade de prática é o principal objetivo de perfis de aplicação, indo além do consenso universal de um simples padrão, sem comprometer a básica interoperabilidade que o padrão habilita entre as fronteiras destas comunidades (DUVAL et al., 2002).

Castro-García e López-Morteo (2013) analisaram 44 trabalhos relacionados ao

¹<http://www.youtube.com>

padrão IEEE-LOM. Como resultado, foram encontrados quinze tipos de extensões implementadas. A partir disto, cinco categorias de perfis de aplicação foram definidas: (i) adição de novos elementos, (ii) exclusão de elementos, (iii) modificação do esquema padrão, (iv) referência para sistemas adicionais de classificação e (v) variados.

Perfis de aplicação também podem refinar definições vigentes ou restringir valores permitidos dentro de domínios de elementos existentes (HEERY; PATEL, 2000). Deste modo, perfis de aplicação podem ser utilizados para classificar OAs de acordo com algum contexto específico.

2.6 Repositórios

Metadados vêm sendo amplamente utilizados em repositórios para descrever objetos de aprendizagem (RAJU; AHMED, 2012). Essa descrição é utilizada para representar seus conteúdos e auxiliar na busca e reuso. A adoção de padrões de metadados objetivam aumentar a interoperabilidade entre comunidades na Web (GREENBERG; SWAUGER; FEINSTEIN, 2013).

Contudo, a heterogeneidade de metadados acarreta em repositórios locais e diminui a interoperabilidade entre dados. A utilização de perfis de aplicação também influencia diretamente na interoperabilidade entre repositórios.

Repositórios indexam seus OAs através de metadados. Contudo, o processo de verificação de metadados é custoso e normalmente feito por pessoas. Além disto, uma descrição incorreta pode resultar em buscas e recuperações problemáticas.

Carvalho et al. (2015) propuseram uma análise quantitativa sobre repositórios distribuídos pelo mundo. Quanto ao tipo de recurso, destacou-se uma maior recuperação de documentos textuais, com formato *pdf*, *xls*, *doc* e extensões similares. Quanto aos metadados, a média de entradas de metadados de OAs pode ser considerada pequena, levando a dificuldades na recuperação e reuso. Repositórios usando Dublin Core dificilmente têm os campos de título, assunto e descrição preenchidos juntos; repositórios IEEE-LOM comumente apresentam título, descrição e formato preenchidos. Os autores também constataram que repositórios pequenos normalmente têm alta taxa de preenchimento de metadados. Na medida em que o tamanho dos repositórios aumenta, a proporção de preenchimento de metadados relacionados à educação é altamente reduzida.

2.7 Considerações do Capítulo

Através de perfis de aplicação, muitas organizações têm atingido seus requisitos, mantendo interoperabilidade com sistemas de organizações que implementam um esquema básico do padrão. Além do fato do padrão IEEE-LOM ser o mais referenciado na literatura, trabalhos recentes foram desenvolvidos para estender o padrão sob a forma de perfis, mostrando a evidência de que o padrão não cobre todos os requisitos dos seus usuários (CASTRO-GARCÍA; LÓPEZ-MORTEO, 2013).

O Dublin Core possui grande aceitação pelo fato de ser um conjunto mais reduzido de metadados. Por sua vez, o IEEE-LOM é definido especificamente para descrever OAs e possui um conjunto maior de metadados. A agregação de novos metadados visa enriquecer e adaptar tecnologias e contextos às descrições dos OAs. É o caso do padrão brasileiro de metadados OBAA que, embora adaptado ao contexto educacional brasileiro, já possui propostas de extensões de metadados para ensino de profissionais da saúde (SCHWARZ et al., 2012), videoaulas (BEHR; PRIMO; VICARI, 2014) e jogos sérios (CARLI et al., 2014).

Um aspecto crítico quanto ao potencial reuso de OAs é o preenchimento dos seus metadados. Quanto maior o número e mais completo são as descrições disponíveis para os campos dos metadados, mais útil o OA pode ser (CARVALHO et al., 2015).

No próximo capítulo, o conceito de ontologia será apresentado. De modo que no decorrer desta dissertação, ele venha a ser utilizado de forma que os metadados sejam legíveis tanto para seres humanos quanto para máquinas no contexto da Web Semântica.

3 ONTOLOGIA

Uma ontologia é definida como uma especificação formal e explícita de uma conceitualização compartilhada (GUARINO; OBERLE; STAAB, 2009). Em outras palavras, as ontologias são descrições consensuais de conhecimento (significado de conceitos e relações) previamente definidos que têm por objetivo ser sem ambiguidades e ser processável por máquinas. Ontologias podem descrever uma hierarquia de conceitos e propriedades no domínio. Seus axiomas expressam relações entre esses conceitos. Os axiomas fazem ontologias mais expressivas, permitindo o uso de mecanismos de inferências.

Uma ontologia O pode ser definida como um conjunto de axiomas (fatos) que descrevem uma situação particular no mundo. Formalmente, uma ontologia consiste de três conjuntos: um conjunto de classes (conceitos), denotado N_C ; um conjunto de propriedades (regras), denotado N_P ; e um conjunto de instâncias (indivíduos), denotado N_I . De forma conceitual, é geralmente dividido em duas partes: TBox que contem axiomas sobre as classes (conhecimento de domínio), denominado T ; e ABox que contém axiomas sobre as instâncias (dados), denominado A . Tal como a equação (CHAHDI et al., 2016):

$$O = \langle T, A \rangle = \langle N_C, N_P, N_I \rangle \quad (3.1)$$

Shadbolt, Berners-Lee e Hall (2006) defendem que as ontologias fornecerão a semântica para que a Web Semântica seja desenvolvida, gerenciada e endorsada por comunidades de prática. Estas comunidades realizarão mudanças em ontologias de modo que a preocupação no custo de desenvolvimento e manutenção de ontologias é levado em consideração. Ontologias são definidas cuidadosamente através de um processo explícito que pretende remover ambiguidades. Elas são meios para fazer um acordo explícito para o compartilhamento de significado entre uma comunidade interessada, mas qualquer um pode usar essas ontologias para descrever seus próprios dados. Similarmente, qualquer um pode estender ou reusar elementos de uma ontologia que se queira. A Web Semântica idealizada faz reuso substancial de ontologias existentes e dados.

Contudo, é esperado que mais de uma ontologia venha a descrever o mesmo domínio. Cada organização produzirá sua própria ontologia. Como resultado, a Web Semântica será composta por uma grande variedade de pequenas ontologias, altamente contextualizadas, desenvolvidas localmente por engenheiros de software e não por especialistas em ontologias (HENDLER, 2001).

Nas próximas seções será abordado como ontologias descrevem domínios, quais

são os tipos de ontologias existentes, as metodologias que auxiliam na sua elaboração. Além disso, conceitos de modularização e alinhamento, pois serão empregados no estudo de caso.

3.1 Suposição de Mundo

Na literatura, duas formas principais de se lidar com descrições das coisas no mundo são encontradas. As suposições de mundo fechado e aberto são empregadas de diferentes modos.

A suposição de mundo fechado pressupõe que um fato não conhecido como verdadeiro, é assumido como falso. Deste modo, a ausência de informação é interpretada como uma informação negativa. Neste contexto, é assumido que a informação completa sobre o estado das coisas é provida (KEET, 2013a).

De outro modo, a suposição de mundo aberto assume que o fato que não é conhecido como verdadeiro ou falso pode ser verdadeiro, pois a ausência de informação é interpretada como informação desconhecida, não como uma informação negativa. É assumido como informação incompleta para um dado estado das coisas, ou seja, pode existir mais informação relevante do que é fornecido por hora (KEET, 2013b).

A Tabela 3.1 descreve uma associação hipotética entre pessoas e países para os quais viajaram. Sobre este cenário, se perguntássemos: “Joaquim viajou para Holanda?”, em um mundo fechado a resposta seria *não*, mas no mundo aberto seria *desconhecida*.

Tabela 3.1: Mundo aberto x mundo fechado: associações hipotéticas entre pessoas e países por onde viajaram

Pessoa	País
João	Estados Unidos, Canadá
Maria	Itália, Bélgica, Alemanha
Pedro	Japão, China
Joaquim	França

Fonte: Elaborado pelo autor

A definição de mundo fechado é útil para restringir informações e validar dados em aplicações com bases de dados relacionais. Contudo, ontologias descrevem conhecimento de maneira extensível, ou seja, quando um sistema trabalha com informação incompleta. Deste modo, ontologias pressupõem mundo aberto.

3.2 Tipos

Isotani e Bittencourt (2015) destacam que existem duas formas mais comuns de diferenciar as ontologias. A primeira delas distingue ontologias leves de pesadas. A segunda discerne ontologias de domínio das de tarefa.

Shadbolt, Berners-Lee e Hall (2006) citam que ontologias pesadas são geralmente aquelas encontradas em ciência e engenharia, onde esforços consideráveis são empregados na construção e desenvolvimento da conceitualização. Isto torna-se aparente quando se usa uma ontologia para classificar conjuntos complexos de propriedades tais como constituindo de certas ordenações de objeto. Ontologias leves compreendem relativamente pequenos termos que não mudam e que organizam quantidades de dados muito grandes.

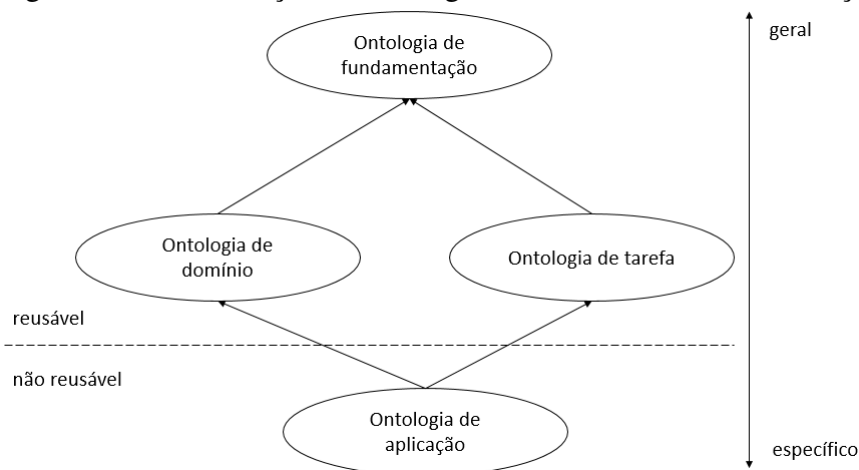
As ontologias pesadas (ou de fundamentação) descrevem conceitos gerais comuns a diversos domínios, onde atributos relevantes são formalizados de forma a dar identidade a estes conceitos. De outra forma, as ontologias leves (ou de aplicação) não se preocupam com a expressão formal de todas as características dos conceitos, mas sim dos seus significados principais.

Quanto à sua função, ontologias podem ser diferenciadas como de domínio e de tarefa. Ontologias de domínio são reutilizáveis no domínio empregado, fornecem vocabulários sobre conceitos, seus relacionamentos, atividades e regras que os governam. Ontologias de tarefa fornecem um vocabulário sistematizado de termos, especificando tarefas que podem ou não estar no mesmo domínio (MIZOGUCHI; VANWELKENHUYSEN; IKEDA, 1995).

A Figura 3.1 apresenta os tipos de ontologias de acordo com seu nível de generalidade e também com seu nível de dependência em uma tarefa particular ou ponto de vista. Setas orientadas representam relacionamentos de especialização.

Além disso, ontologias podem ser classificadas como monolíticas e modulares. Uma ontologia modular pode ter múltiplas partes ou módulos caracterizados por alta coesão entre classes dentro de um módulo. Uma ontologia modular com uma qualidade maior pode ser facilmente escalável, gerenciável e reusável (BALIYAN; KUMAR, 2016).

Figura 3.1: Classificação de ontologias baseado no nível de abstração



Fonte: Adaptado de (PAULHEIM, 2011)

3.3 Metodologias para Ontologia

Uma metodologia é definida como compreensivas séries integradas de técnicas ou métodos que criam uma teoria geral de sistemas de como uma classe de trabalho intelectual intensiva deveria ser executada (RADATZ; GERACI; KATKI, 1990). A composição de métodos, técnicas, processos e atividades também pode definir uma metodologia (GOMEZ-PEREZ; FERNÁNDEZ-LÓPEZ; CORCHO, 2006).

Metodologias para ontologias são geralmente adaptadas de softwares existentes e metodologias de engenharia de sistemas de conhecimento. Elas podem ser classificadas em três categorias (CASELLAS, 2011):

- Metodologias inspiradas em IA: desenvolvida a partir de processos e técnicas de aquisição de conhecimento. Exemplos: TOVE (GRÜNINGER; FOX, 1995), Enterprise Model Approach (USCHOLD, 1996) e IDEF5 (KBSI, 1994).
- Metodologias inspiradas em ES: processos tradicionais e metodologias ágeis inspiraram metodologias para o desenvolvimento de ontologias, tais como METHONTOLOGY (FERNÁNDEZ-LÓPEZ; GÓMEZ-PÉREZ; JURISTO, 1997), Ontology Development 101 (NOY; MCGUINNESS et al., 2001) e RapidOWL (AUER; HERRE, 2006).
- Metodologias inspiradas em IHC: provêm uma maior participação (colaboração e negociação) para construir ontologias. DILIGENT (DAVIES et al., 2005) e HCOME (KOTIS; VOUIROS, 2006) são exemplos.

O ciclo de vida de uma ontologia pode variar de acordo com a metodologia utili-

zada. Por isso, esta seção não visa entrar em detalhes específicos das metodologias para o desenvolvimento de ontologias.

3.4 Modularização

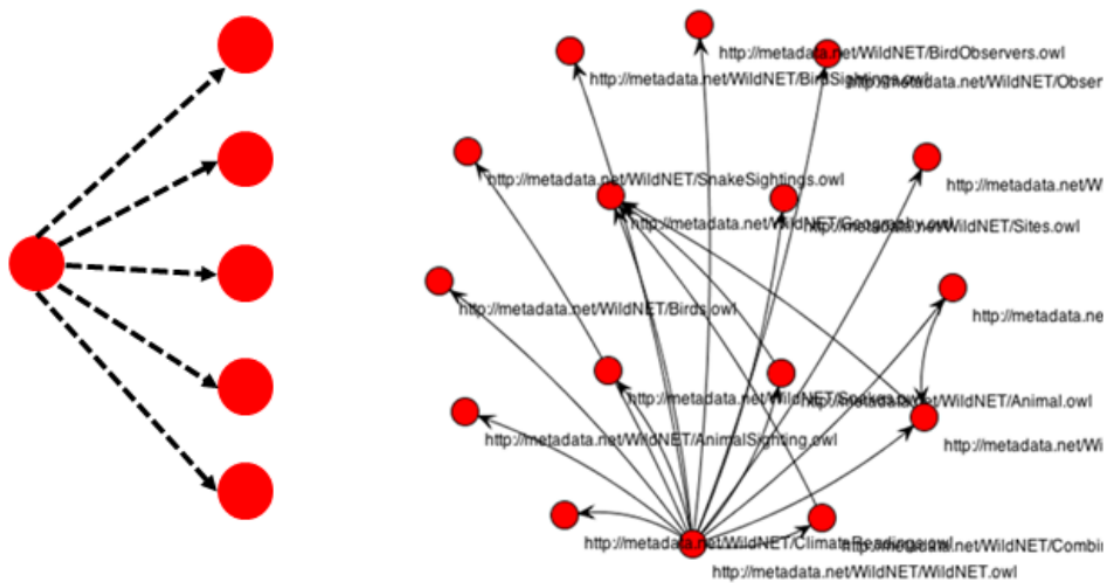
A modularização de ontologias ainda está em fase inicial. Então, ainda se carecem de conceitos e definições maduros, bem-definidos, bem-entendidos e consensualizados como os definidos em modularização na área de engenharia de software (SULLIVAN et al., 2001).

Em geral, a modularização de ontologias objetiva prover a usuários de ontologias o conhecimento requerido, reduzindo o escopo o quanto for possível para aquilo que é estritamente necessário. Em particular, módulos de ontologias (i) facilitam o reuso de conhecimento através de várias aplicações, (ii) são mais fáceis de construir, manter e substituir, (iii) habilitar engenharia de ontologias em módulos distribuídos em diferentes localizações e diferentes áreas de especialidades, e (iv) habilitar gerenciamento efetivo e busca de módulos (JARRAR, 2005).

Quatro padrões existentes que caracterizam ontologias modularmente organizadas são propostos a partir de métricas, tais como tamanho, coesão e acoplamento (ABBES et al., 2012):

1. Um módulo importa n módulos: O padrão proposto está em conformidade com o conceito de agregação. Este padrão estabelece uma relação entre um único módulo e um conjunto de módulos na mesma ontologia. Esta conexão se estabelece de forma unidirecional. A Figura 3.2 exemplifica o módulo *WildNet.owl* que importa várias outras como *Animal.owl*, *AnimalSighting.owl*, *Birds.owl*, etc. Este padrão possui uma alta coesão comparada com a métrica de acoplamento. A métrica de coesão pode ser considerada como um indicador do grau para o qual os elementos no módulo pertencem juntos. A ideia é que conceitos agrupados em uma ontologia deveriam ser conceitualmente relacionados em um domínio particular a fim de atingir objetivos comuns.
2. N módulos importam um módulo: Este padrão pretende corresponder ao conceito de herança. Ele estabelece uma correspondência entre um conjunto de módulos e um único módulo na mesma ontologia. Esta correspondência também é unidirecional. Na Figura 3.3, por exemplo, existem três módulos independentes importando

Figura 3.2: Um módulo importando n módulos

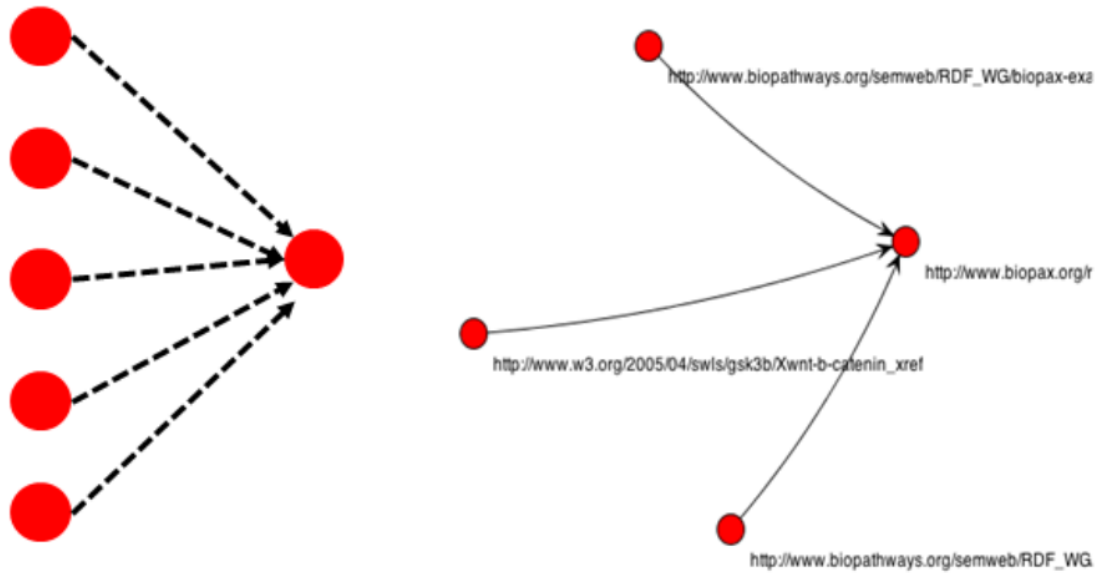


Fonte: (ABBES et al., 2012)

um módulo, o qual contém o conhecimento geral (*biopax-level1.owl*). Este padrão é caracterizado por interconexões entre os módulos. O grau de acoplamento depende de quão complicado as conexões são e no tipo de conexões entre os módulos. O englobamento de todos os outros módulos, os quais estão fortemente relacionados, leva a uma alta coesão.

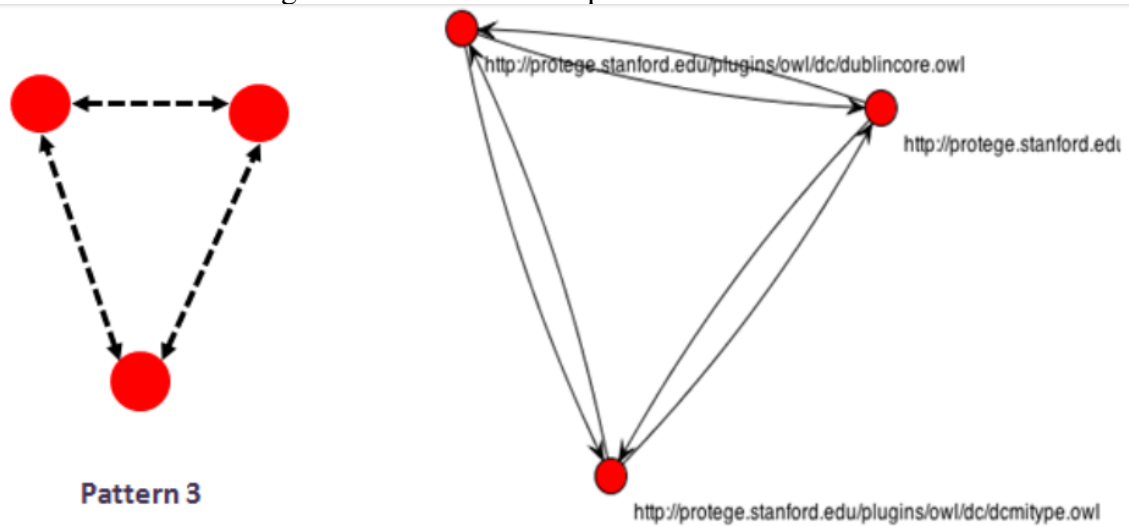
3. N módulos importam n-1 módulos: A característica que distingue este padrão é que cada um dos n módulos importam todos os outros. As correspondências entre módulos é bidirecional. Por exemplo, na Figura 3.4 tem-se três módulos dependentes: *dublincore.owl*, *terms.owl* e *dcmitype.owl*.
4. Misto: Este padrão combina todos os padrões anteriormente citados, é uma proposta de padrão misto. As correspondências podem ser unidirecionais ou bidirecionais. Na Figura 3.5 podem ser encontrados os dois primeiros padrões citados. A maior característica deste padrão é o alto acoplamento no que diz respeito a coesão. A modularização de ontologias de domínio tende a depender de padrões do tipo 1 e 2, anteriormente citados. Em contraste, ontologias de fundamentação aparentam ter uma estrutura menos direta e uma organização particular quando comparadas a ontologias de domínio.

Figura 3.3: N módulos importando um módulo



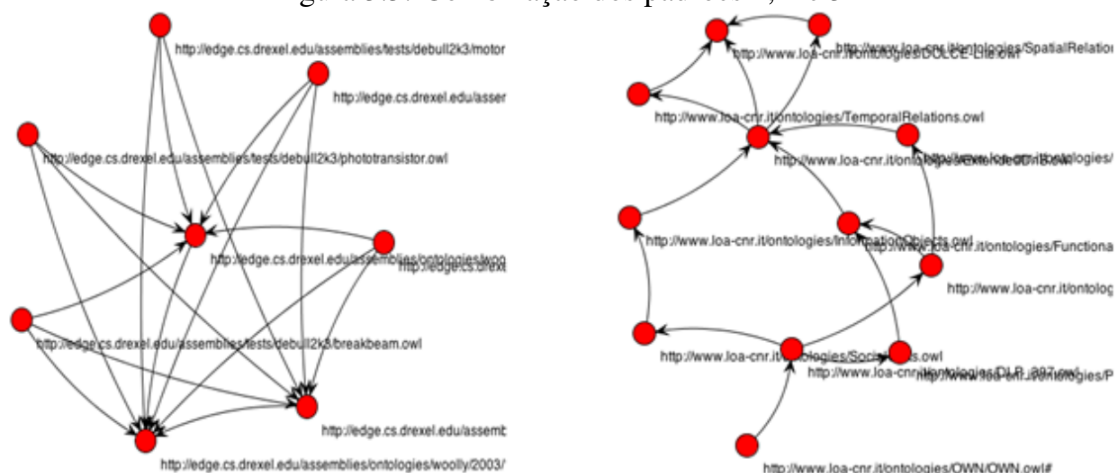
Fonte: (ABBES et al., 2012)

Figura 3.4: N módulos importando n-1 módulos



Fonte: (ABBES et al., 2012)

Figura 3.5: Combinação dos padrões 1, 2 e 3



Fonte: (ABBES et al., 2012)

3.5 Alinhamento

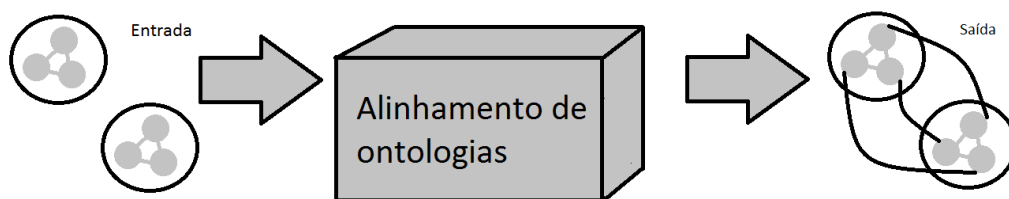
Ehrig (2006) propõe que, geralmente, existem duas abordagens para integrar ontologias. Na primeira, engenheiros de ontologias discutem, negociam e votam para diretamente encontrar uma ontologia compartilhada, o qual normalmente é um processo longo e tedioso. Na segunda, o alinhamento é realizado mais tarde, alinhando ontologias e esquemas já existentes. O campo de pesquisa considerado de alinhamento de ontologia e integração caracteriza um grande número de termos tais quais alinhamento, mapeamento, mediação, fusão, etc. Infelizmente, as definições de diferentes autores são confusas, parcialmente inconsistentes e algumas vezes até contraditórias. O autor define uma função de alinhamento tal que:

$$\text{align}(e) : E \times O \times O \rightarrow E \quad (3.2)$$

Uma função de alinhamento ontológico parcial align , baseada no conjunto E de todas entidades $e \in E$ e no conjunto de possíveis ontologias O . Quando um alinhamento (parcial) align entre duas ontologias O_1 e O_2 é estabelecido, diz-se que uma entidade e é alinhada com uma entidade f quando $\text{align}(e) = f$. Um par de entidades (e, f) ainda não está em align quando para cada critério de alinhamento apropriado ainda precisa ser testado, então é chamado de um candidato a alinhamento.

Dadas duas ontologias, alinhar uma ontologia com outra significa que para cada entidade (conceito, relação ou instância) na primeira ontologia é tentado encontrar uma entidade correspondente que tenha o mesmo significado intencionado em uma segunda

Figura 3.6: Visão geral do processo de alinhamento de ontologias



Fonte: Adaptado de (EHRIG, 2006)

ontologia. Portanto, um alinhamento é uma relação de igualdade um para um, como ilustrado pela Figura 3.6. Obviamente, para algumas entidades nenhuma entidade correspondente pode existir.

3.6 Considerações do Capítulo

A evolução de representação de metadados se encaminha a uma forma semântica (AL-KHALIFA; DAVIS, 2006). Deste modo, os metadados podem ser processáveis por máquinas, sendo flexíveis para extensão, fonte para motores de inferências e interoperáveis. As ontologias fornecem estas características, sendo grandes responsáveis por interligar recursos Web à uma série de classes e propriedades definidas.

O presente trabalho emprega a modularização de ontologias de domínio para a representação de metadados como um catalizador para reuso e verificações através de motores de inferências. Além disso, apresenta um método baseado em ontologias como provedor de semântica para que metadados legados em formato XML sejam transpostos para descrições ontológicas em OWL. O alinhamento de ontologias também é utilizado como uma descrição enriquecida para prover equivalências de descrições OWL. Por fim, será estudado como diferentes tecnologias da Web Semântica operam sobre visões de mundo aberto e mundo fechado e qual a relação com os motores de inferências.

4 WEB SEMÂNTICA

A Web Semântica (WS) vem estendendo a atual World Wide Web. A informação contida nela possui estrutura e significado para ser entendível pelas máquinas. Além disso, a informação é interconectada, construindo relações entre os conceitos. Deste modo, as máquinas estão habilitadas a conduzir raciocínio automático (BERNERS-LEE et al., 2001).

A Web Semântica é uma Web de informação acionável, cuja informação é derivada de dados através de uma teoria semântica para a interpretação de símbolos. A teoria semântica provê um significado no qual as conexões lógicas dos termos estabelecem interoperabilidade entre sistemas. A integração de diversos e heterogêneos conjuntos de dados vem sendo obtida, em grande parte, pela adoção de conceitualizações comuns, referidas como ontologias (SHADBOLT; BERNERS-LEE; HALL, 2006).

A conexão entre informações é feita através de triplas. As triplas podem ligar qualquer coisa na Web. Cada tripla é composta de três partes: sujeito, predicado e objeto. O sujeito é um Uniform Resource Identifier (URI) que descreve o recurso com um único conceito. O predicado correlaciona o sujeito e o objeto. Por fim, o objeto pode ser uma URI ou um valor literal. A adoção deste e outros princípios direciona a iniciativa chamada Dados Conectados¹ (HEATH; BIZER, 2011), uma subárea da Web Semântica.

A integração de dados é apontada como uma das maiores diretivas da Web Semântica. Para cooperar com isto, linguagens têm sido usadas para codificar semântica, fornecendo suporte à integração e interoperabilidade de dados. A eXtensible Stylesheet Language (XSLT) é considerada uma linguagem facilitadora para geração de conteúdo para a Web Semântica, especialmente XML provenientes de banco de dados (SHADBOLT; BERNERS-LEE; HALL, 2006). Atualmente, tanto companhias quanto a academia necessitam de bases de dados conectadas, altamente compartilháveis, que permitem interoperabilidade e podem lidar com acumulação de conhecimento disponíveis na Web (ISOTANI; BITTENCOURT, 2015).

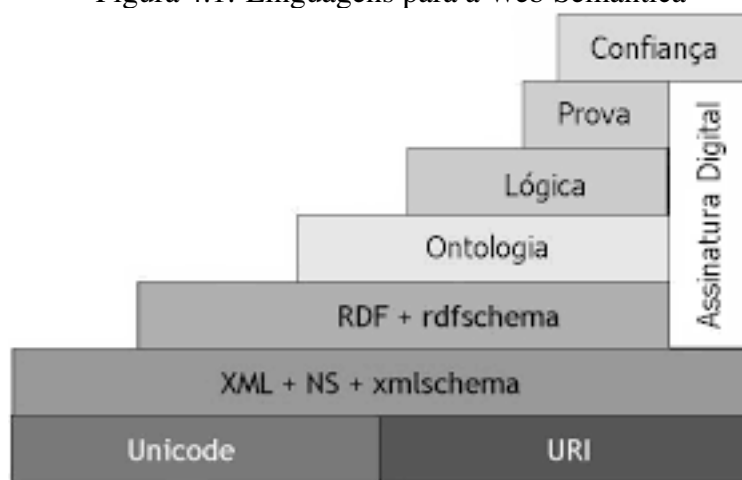
A certeza de como a Web Semântica será construída é motivo de debates. Contudo, existem temas que estão presentes em quase a maioria deles, tais como: ferramentas, agentes, serviços web, construção de modelos semânticos, metadados, ontologias e linguagens da Web Semântica (BREITMAN, 2005). Além disso, linguagens da WS estão sendo usadas para representar, codificar e trocar dados semânticos em muitos contextos

¹Em inglês: Linked Data

além da Web (DING; FININ, 2006). Como exemplos, os autores citam: banco de dados, sistemas multiagentes, computação móvel e ambientes de rede *ad hoc*.

A arquitetura da Web Semântica é definida como um modelo em camadas. O objetivo é ir realizando expansões sobre aquilo que já existe (inclusive sobre a Web atual) e ir obtendo consensualizações em cada mudança (BREITMAN, 2005). A Figura 4.1 ilustra a arquitetura proposta por Tim Berners Lee. Na próxima seção, cada uma destas camadas será explanada em conjunto com as tecnologias empregadas para cada uma delas.

Figura 4.1: Linguagens para a Web Semântica



Fonte: Adaptado de (BERNERS-LEE, 2000)

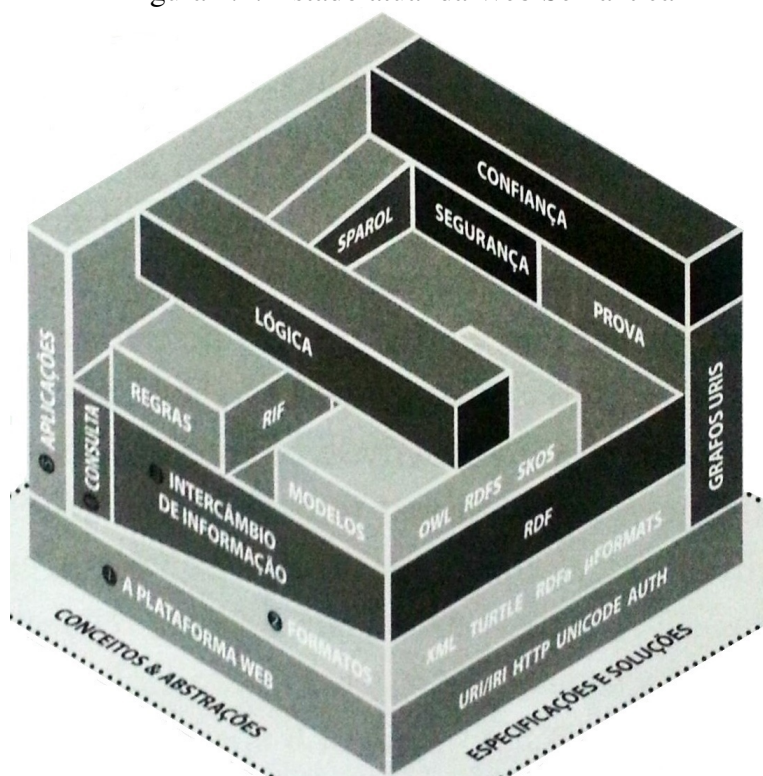
4.1 Tecnologias

Com o passar dos anos, a arquitetura proposta por Tim Berners Lee vem incorporando as tecnologias apropriadas para cada uma das camadas do modelo. A Figura 4.2 correlaciona cada uma das camadas (conceitos e abstrações) com as tecnologias propostas (especificações e soluções).

A compatibilidade com a Web atual é permitida incorporando tecnologias já consolidadas: URI/IRI, HTTP, UNICODE e AUTH. A URI/IRI é utilizada para identificação única de recursos abstratos ou físicos. Ela é composta pela união de URN (para nomes) e URL (para endereços). A recuperação de recursos pode ser realizada por servidores HTTP (protocolo para comunicação) baseados em UNICODE (padrão para representar e manipular textos) em conjunto com restrições de acesso básicos a usuários (AUTH).

Os dados podem ser representados nos mais diversos formatos, objetivando um entendimento mais fácil para máquinas, humanos ou ambos. Pode-se citar XML, Turtle, RDFa, μ formats, dentre outros. Contudo, o fator determinante para que ocorra interope-

Figura 4.2: Estado atual da Web Semântica



Fonte: Adaptado de (ISOTANI; BITTENCOURT, 2015)

tabilidade entre os dados é que os mesmos estejam em forma padronizada. No caso da Web Semântica, os dados devem estar em RDF.

O RDF é um modelo de dados para representar grafos dirigidos e etiquetados, permitindo a descrição de recursos. RDF usa triplas na forma Sujeito - Predicado - Objeto, onde Sujeito e Predicado devem ser URIs e o Objeto pode ser uma URI ou um literal. As declarações em RDF necessitam de URIs que, por sua vez, devem ter um significado bem definido (semântica). O objetivo do RDF é fornecer um conjunto mínimo para a representação de conhecimento na Web Semântica.

A realização de consultas sobre os dados em um grafo RDF é dada por SPARQL. É uma linguagem declarativa para consultas, similar ao SQL. O resultado destas consultas pode ser um conjunto de grafos RDF.

Na Web Semântica, ontologias são utilizadas para definir termos, associar semântica a URIs, descrever um domínio e representar um conhecimento prévio. OWL e RDF-S são linguagens comumente usadas para descrever vocabulários e ontologias.

A linguagem da Web Semântica OWL é projetada para representar um conhecimento completo acerca das coisas, seus grupos e relações. É baseada em lógica computacional de modo que o conhecimento expressado em OWL pode utilizar raciocínio automático por máquinas, de modo que classificação e verificação de consistência podem

ser realizados.

RDF-S possui um conjunto hierárquico de classes com certas propriedades que se utiliza do modelo de dados extensível de representação de conhecimento RDF para agregar semântica a dados. Ele provê elementos básicos para a descrição de ontologias e é almejado que forneça elementos para estruturar os recursos RDF, possibilitando também a realização de inferências que deduzem triplas não declaradas de forma explícita.

O modelo de dados SKOS é comumente empregado para compartilhamento e enlace de dados em sistemas de organizações de conhecimento na Web. Muitos destes sistemas, tais como tesouros e taxonomias, compartilham de uma estrutura similar. A partir disto, SKOS é capaz de capturar muitas destas similaridades e torná-las explícitas, permitindo compartilhar dados e tecnologias entre diversas aplicações.

RIF tem por objetivo definir um padrão para o intercâmbio de regras entre sistemas de regras e, particularmente, entre motores de regras para a web. Junto com este padrão, uma família de linguagens foi definida, chamada dialetos: Core, Basic Logic Dialect (BLD) e Production Rule Dialect (PRD).

Algumas camadas, tais como lógica, prova, segurança e confiança ainda não possuem tecnologias recomendadas. Por fim, muitas aplicações acabam utilizando somente uma parte das tecnologias da pilha.

4.2 Web Ontology Language

A Web Ontology Language (OWL) é esperada que se torne a linguagem ontológica com maior aceitação na Web Semântica. É um padrão de ontologia da World Wide Web Consortium (W3C)². OWL é construída acima do Resource Description Framework (RDF), possuindo um maior poder de expressividade que o RDF-S (ANTONIOU; HARMELLEN, 2004).

Em sua primeira versão (BECHHOFFER, 2009), a OWL foi dividida em três variantes de linguagem: OWL Lite, OWL DL e OWL Full. A OWL DL era a linguagem mais expressiva e decidível, mas carecia de muitos construtos para domínios complexos, tais como restrições de cardinalidades qualificadas. Mesmo decidível, o reasoning em OWL DL tem um pior caso de complexidade computacional alto (NExpTime-complete). OWL Lite era um subconjunto de OWL DL que excluía construtos que são considerados responsáveis pelo aumento na complexidade computacional. OWL Full era a mais

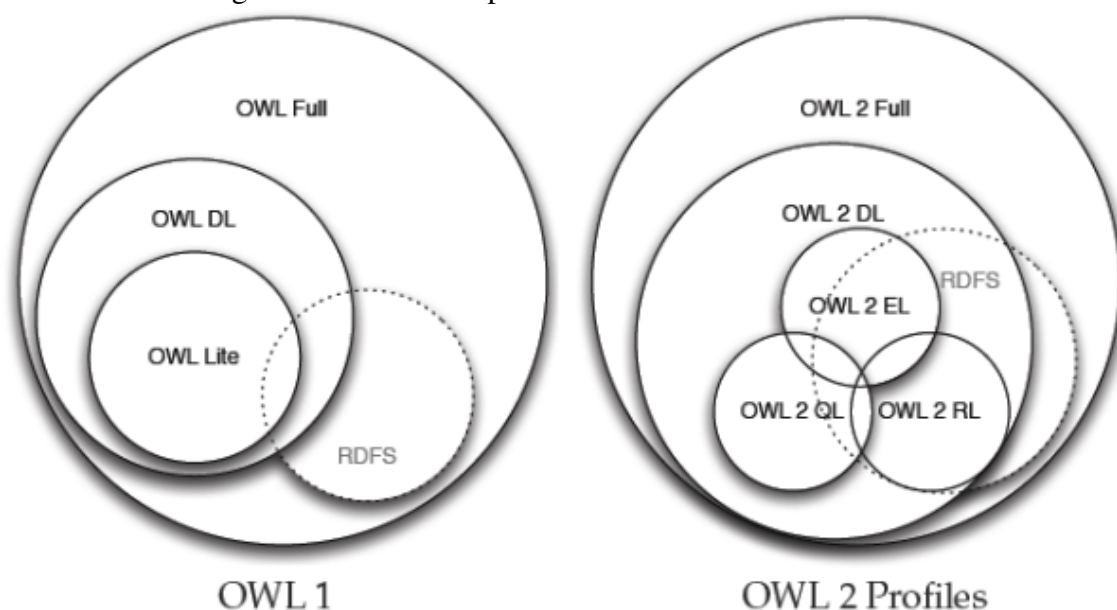
²<<http://www.w3.org>>

expressiva variante de OWL, entretanto indecidível.

A OWL foi rearranjada em sua segunda versão (GRAU et al., 2008), objetivando aumentar seu poder de expressividade. Ela provê três perfis: OWL 2 EL, OWL 2 QL e OWL 2 RL. OWL 2 EL permite um raciocínio eficiente com grandes terminologias, especialmente projetadas para classificação. Este perfil proporciona tempo polinomial de raciocínio. Por sua vez, OWL 2 QL permite um raciocínio eficiente com grandes quantidades de dados estruturados. OWL 2 RL foi projetada para que várias tarefas de raciocínio sejam implementadas como um conjunto de regras em um encadeamento progressivo de sistema de regras. Por esta implementabilidade, OWL RL não é tão expressiva como a OWL 2. A expressividade da OWL 2 acarreta em um tempo de raciocínio $2N\text{ExpTime}$ -complete.

A Figura 4.3 apresenta as relações entre as variantes da OWL e os perfis da OWL 2. Além disso, é possível visualizar a comparação de expressividade com o RDF-S.

Figura 4.3: Perfis e expressividades em OWL 1 e OWL 2



Fonte: <<http://www.mkbergman.com/913/metamodeling-in-domain-ontologies/>>

4.3 Semantic Query-enhanced Web Rule Language

Enquanto que SPARQL é considerado como a linguagem padrão para consulta sobre RDF, ainda não existe uma recomendação para consultas sobre ontologias OWL. Mesmo com a possível serialização de OWL para RDF/XML, ferramentas podem gerar

resultados ligeiramente diferentes, pois não tem entendimento nativo de OWL, o que acarreta em resultados de consultas SPARQL distintos (O'CONNOR; DAS, 2009).

A Semantic Web Rule Language (SWRL) é uma proposta que estende o conjunto de axiomas OWL para incluir regras similares a cláusulas de Horn. Estas regras são na forma de uma implicação entre um antecedente e um conseqüente. Ambos consistem de zero ou mais átomos (HORROCKS et al., 2004).

A extração de informação de ontologias OWL é um requisito básico. SQWRL fornece operadores básicos de busca, tais como *select*, *count*, *orderBy*, que operam sobre o resultado da consulta e não sobre a ontologia em questão. Além disso, a linguagem não viola a suposição de mundo aberto. A utilização de elementos em conjuntos provê um mecanismo de fechamento de mundo. Deste modo, operadores para construir e manipular conjuntos são fornecidos como funcionalidades da linguagem (O'CONNOR; DAS, 2009).

4.4 Considerações do Capítulo

As tecnologias e padrões descritos permitem a incorporação da informação semântica necessária nos sistemas de informação, sendo as anotações semânticas uma via para tal propósito. A anotação semântica se refere ao processo mediante o qual se adicionam metadados semânticos aos recursos web (GARCÍA; FERNÁNDEZ, 2005). Uma anotação semântica combina conceitos de metadados e ontologias, é dizer os campos dos metadados que são associados com termos em uma ontologia, quais são usados para descrever estes campos (MACÁRIO; SOUSA; MEDEIROS, 2010).

A presente dissertação tem por objetivo de relacionar três dos principais temas da Web Semântica citados anteriormente: metadados, ontologias e linguagens da WS de forma que os metadados de OAs descritos por ontologias OWL venham a ser empregados em sistemas e repositórios inteligentes. Desta forma, metadados de recursos educacionais estarão armazenados com anotações semânticas de forma padronizada com a linguagem OWL para que haja interoperação dos dados de maneira mais expressiva. A linguagem de consulta SQWRL será utilizada tanto para a realização de consultas quanto para verificações complementares de consistência em um contexto de suposição de mundo fechado.

5 TRABALHOS RELACIONADOS

Neste capítulo, trabalhos relacionados a abordagem empregada serão apresentados. Em um primeiro momento, pesquisas que retratam ontologias para o padrão de metadados IEEE-LOM e OBAA no formato OWL. Estes padrões foram escolhidos por apresentarem uma hierarquia e por possuírem uma complexidade maior, ou seja, apresentam um nível de expressividade maior na pilha da Web Semântica. Em seguida, estudos relacionados a transformações entre as tecnologias da Web Semântica serão apresentados, relacionados a interoperabilidade.

5.1 Estado da Arte

- (CASALI et al., 2013):

Neste trabalho, uma ontologia para o padrão IEEE-LOM foi projetada e implementada. O propósito da pesquisa é dar semântica aos metadados de objetos de aprendizagem usando uma modelagem ontológica para a estrutura do IEEE-LOM de forma que a ontologia seja utilizada como uma camada intermediária de aplicações.

Na ontologia, as nove categorias principais e elementos agregados de dados foram definidas como classes. Por sua vez, os elementos simples de dados foram representados como propriedades. Por exemplo, *General* e *Identifier* são classes e *hasKeyword* é uma propriedade. Além disso, as cardinalidades foram definidas como superclasses, tais como *hasKeyword max 10 Thing* definida como superclasse de *General*. A disposição de classes foi organizada de forma plana.

Como forma de validação da ontologia, os autores apresentam uma ferramenta que auxilia na catalogação dos metadados. Contudo, esta ferramenta só possibilita o catálogo de um subconjunto de sete metadados. O trabalho também não analisa problemas de desempenho que cardinalidades máximas podem acarretar e alteram a semântica do metadado *Keyword* da categoria *General* para que tenha cardinalidade máxima quatro, ao invés de dez, como descrito no padrão IEEE-LOM.

Este estudo apresenta como uma ontologia de metadados pode ser implementada em uma aplicação, cujo objetivo é auxiliar a catalogação dos metadados de OAs em repositórios. Esta aplicação exemplifica uma alternativa de Carga descrita no Capítulo 7 para as ontologias propostas no Capítulo 6.

- (GARCÍA et al., 2012):

LOM2OWL é um esforço para melhorar busca, reuso e uso de objetos de aprendizagem no portal Organic Edunet. A implementação da capacidade de pesquisa semântica no portal foi realizada através de um mapeamento OWL para o padrão IEEE-LOM, denominado LOM2OWL.

As categorias principais, elementos agregados, objeto de aprendizagem e novos tipos de dados compõem a hierarquia de classes. Os autores definiram classes para novos tipos de dados: para *DateTime*, uma classe *lomDateTime* que possui as propriedades *textDescriptor* e *timeItem* foram definidas; uma classe *singleLangString* para representar o tipo de dado *LangString*; e uma classe *duration* que tem propriedades para duração e descrição. O tipo de dado *Vocabulary* foi representado como uma propriedade de dado com um espaço de valores restrito. As demais classes possuem uma ou mais propriedades.

Os autores em nenhum momento citam se descreveram verificações de cardinalidades na ontologia. Além disto, também não relataram como reutilizaram os metadados legados do repositório e como a ontologia é empregada na busca.

O trabalho dos autores tem como objetivo utilizar ontologias de metadados para melhorar a recuperação de OAs no repositório. Este estudo, além de estar relacionado com as ontologias elaboradas no Capítulo 6, também se relaciona com a fase de Extração do Capítulo 7.

- (CHOE, 2013):

Choe elabora um estudo que analisa como os padrões MPEG-7 e IEEE-LOM podem ser unidos para representar OAs em um sistema *e-learning*. O autor utiliza ontologia para gerar um modelo padronizado para descrever objetos de aprendizagem multimídia.

A interoperabilidade entre ontologias IEEE-LOM e MPEG-7 é realizada através de classes de equivalência. O autor também define classes para categorias e elementos de dados agregados. Existe múltipla herança em algumas classes, como por exemplo, a classe *Description* possui *Annotation*, *Classification*, *Educational*, *General*, *Resource* e *Rights* como superclasses.

O autor exemplifica o alinhamento de classes, mas não menciona se foi realizado o alinhamento de propriedades. Além disto, também não é relatado se o alinhamento foi manual ou automático.

Além de correlato ao Capítulo 6, o trabalho auxilia no entendimento do Capítulo 7. Este capítulo explora como diferentes descrições ontológicas podem interoperar

através de alinhamentos.

- (SOLOMOU; PIERRAKEAS; KAMEAS, 2015):

Neste trabalho, os autores definiram metadados para descrever materiais para educação a distância, denominado Educational Metadata Profile (EMP). EMP utiliza um subconjunto do padrão IEEE-LOM e adiciona alguns conceitos no seu escopo de domínio com o objetivo de descrever efetivamente alguns dos mais importantes aspectos de um recurso educacional (pedagógico, técnico, etc.), mas não ser excessivamente analítico que torne-se difícil de usar.

A ontologia é composta de quatro classes principais: *LearningObject*, *Contributor*, *FileFormat* e *LearningResourceType*. Elementos simples são definidos como propriedades de dados. Contudo, quando elementos possuem um espaço de valores, este é definido como um conjunto de indivíduos. Por exemplo, a classe *LearningResourceType* tem os seguintes indivíduos: *case_study*, *problem_solving*, etc. Os mecanismos de inferência foram explorados através da Semantic Web Rule Language para recuperar objetos de aprendizagem baseados em sua associação com os objetivos de aprendizagem. Vinte e quatro instâncias foram utilizadas para experimentar a ontologia. Além disto, é apresentado um mapeamento do esquema resultante com o padrão de metadados DC.

Os autores não evidenciam como o tratamento das restrições de cardinalidades é efetuado. Também não é avaliada a performance dos mecanismos de inferências explorados.

Esta pesquisa objetiva aumentar o potencial de descoberta e a recuperação de recursos em ambientes inteligentes de *e-learning*. A utilização de um subconjunto do padrão IEEE-LOM caracteriza um potencial uso de ontologias modulares, descritas no Capítulo 6.

- (MENOLLI et al., 2014):

A Ontology for Organizational Learning Objects (OOLO) foi baseada em um subconjunto do padrão IEEE-LOM. O OOLO tem o objetivo de ajudar as organizações na criação de conteúdos, especialmente por organizações de desenvolvimento de *software*.

Em sua hierarquia, as categorias do IEEE-LOM: *Educational*, *General*, *LifeCycle*, *Meta-Metadata*, *Rights*, *Relation* e *Technical* foram modeladas como classes. Além disso, duas classes foram adicionadas: *Context* e *OOLO Model Document*. A ontologia também foi capaz de realizar inferências, tais como inferir a consistência de objeto ou até mesmo o conteúdo da linguagem do objeto.

As propriedades da ontologia não foram exemplificadas pelos autores. Além disto, as inferências realizadas também não foram explanadas.

A proposta dos autores auxilia no entendimento de que ontologias modulares possam vir a ser exploradas para utilização em subconjuntos de padrões de metadados descritos como ontologias. Este estudo é correlacionado ao Capítulo 6.

- (GLUZ; VICARI, 2012):

Gluz e Vicari propuseram uma ontologia para o contexto educacional brasileiro. Este trabalho objetiva prover interoperabilidade entre objetos de aprendizagem na Web, televisão digital e dispositivos móveis. A ontologia se propõe a descrever metadados do padrão IEEE-LOM e OBAA.

A hierarquia de classes é organizada em cinco classes principais: (i) *Aggregate*: classe geral para elementos agregados que podem compor os metadados; (ii) *Content*: classe genérica para possíveis conteúdos de objetos de aprendizagem; (iii) *LearningObject*: os próprios objetos de aprendizagem; (iv) *Metadata*: tem as categorias principais como subclasses; e (v) *Vocabulary*: classes para metadados que tem espaço de valores definidos. Propriedades de dados foram criadas para cada elemento simples de dado. Indivíduos representam elementos das classes *Aggregate* ou *Metadata*. Assim, propriedades de objeto relacionam estes indivíduos com uma propriedade de objeto *hasMetadata* ou uma propriedade de objeto específica, respectivamente.

A ontologia é compatível com a OWL DL. Ela não trabalha restrições de cardinalidades dos metadados. Propriedades de dados e de objetos não possuem domínios e contradomínios definidos.

Esta ontologia auxilia no entendimento de que uma mesma comunidade, com diferentes propósitos, pode produzir ontologias diferentes. A correlação desta pesquisa está no método de interoperabilidade proposto no Capítulo 7.

- (BEHR; PRIMO; VICCARI, 2012):

O estudo apresenta uma proposta de ontologia OBAA que importa ontologias IEEE-LOM e IMS AccessForAll. As ontologias têm o intuito de prover uma capacidade maior de inferências do que a última citada, sendo compatível com a OWL 2.

Diferentemente dos outros trabalhos, toda a hierarquia de metadados (principais categorias, agregados e elementos simples de dados) foi transposta em uma hierarquia de classes, com o objetivo de automatizar a tipificação de indivíduos. Os autores empregam verificações para cardinalidades, valores condicionais e verificação do tamanho de

strings. Somente categorias e metadados agregados com cardinalidade maior que um tem representação com indivíduos almejando, diminuir a granularidade.

Esta pesquisa também não realiza uma mensuração do tempo de *reasoning* das ontologias propostas. Mesmo com a definição de perfis de aplicação, todos os axiomas são importados na representação de conhecimento dos metadados dos OAs.

As ontologias deste estudo foram utilizadas como base para o desenvolvimento das ontologias modulares do Capítulo 6. A aplicação proposta também é um meio relacionado à fase de Extração do Capítulo 7.

- (HACHEROUF; BAHLOUL; CRUZ, 2015):

O trabalho dos autores realiza um estudo sobre o estado da arte de conversões XML para OWL. Os autores definem duas categorias: (i) abordagens em instâncias, baseadas em documentos XML sem qualquer esquema associado; e (ii) abordagens de validação, baseadas no esquema XML e o documento validado pelo esquema associado. A segunda abordagem se beneficia da definição do esquema (XML Schemas ou DTDs) para prover transformações automáticas com restrições lógicas.

A pesquisa identificou oito propriedades para descrever as abordagens: o tipo de entrada de dados, o tipo de saída de dados, o tipo de linguagens OWL utilizadas, a geração da ontologia esquema, se os indivíduos são criados de forma sequencial ou paralela, se é automático ou semiautomático, o uso de restrições de integridade e o reuso de ontologias existentes. Além disso, a pesquisa identificou um conjunto de critérios que deveriam ser considerados. O conjunto é composto de três critérios: a riqueza da transformação dos resultados, a complexidade dos resultados e a perda de informação.

Este estudo auxilia no entendimento do método de interoperabilidade proposto no Capítulo 7. No método proposto na dissertação: o tipo de entrada de dados é a instância XML, a saída são ontologias com indivíduos OWL, a linguagem utilizada é OWL 2, a geração é semiautomática, são usadas restrições de integridade e ontologias existentes. Além disso, o método propõe transformações entre ontologias OWL, o que não foi explorado pelos autores na pesquisa.

- (BEDINI et al., 2011):

O objetivo deste trabalho é identificar padrões práticos para demonstrar como XML Schemas podem ser minerados para extrair automaticamente asserções ontológicas e prover uma abordagem concreta e implementável que otimiza sistemas existentes. Os autores definiram 40 padrões que são capazes de mapear a maior parte dos construtos

do XML Schema pela integração de várias práticas específicas de projetos.

O mapeamento de construtos em padrões objetiva facilitar o processo de criação da ontologia. Uma implementação foi realizada para validar a proposta.

A pesquisa foi realizada somente para XML Schema, ou seja, objetivou-se declarações ontológicas TBox. Não foi realizada integração com instâncias XML, comparadas com ABox.

Este estudo é uma visão complementar do Capítulo 7. Os autores operaram somente no TBox, enquanto que o método proposto opera no ABox, por trabalhar através da semântica de uma ontologia existente.

- (O’CONNOR; DAS, 2011):

Os autores alegam que uma deficiência geral é que os métodos de mapeamento não são centrados em OWL e, portanto, limitam a complexidade das ontologias geradas. Por isso, eles desenvolveram uma linguagem baseada em OWL que pode transformar documentos XML em ontologias OWL arbitrárias. A linguagem é baseada em Manchester OWL Syntax que é estendida com buscas XPath para suportar referências a documentos XML. Esta linguagem é denominada XML Master e suporta documentos XML com ou sem XML Schema.

Como trabalho futuro, os autores esperam implementar um *plugin* para o editor de ontologias Protégé. Além disso, pretendem desenvolver uma ferramenta visual que permita que usuários não especialistas definam mapeamentos entre XML e OWL.

O estudo auxilia no entendimento de que ferramentas e serviços necessitam ser desenvolvidos para a interoperabilidade dos metadados. Por isso, correlaciona-se com a proposta de método do Capítulo 7.

- (YAHIA; MOKHTAR; AHMED, 2012):

Este trabalho propõe um método para criar automaticamente ontologias OWL a partir de múltiplas fontes XML. Ele é baseado em XML Schema para construir a ontologia, mas se o esquema não existir pode ser automaticamente gerado a partir do documento XML.

Este método é dividido pelos autores em quatro passos: (1) Trang API¹ gera o XML Schema a partir do documento XML; (ii) XML Schema Object Model (XSOM)² analisa o XML Schema; (iii) a saída do XSOM é usada como entrada do Java Universal

¹<<https://code.google.com/archive/p/jing-trang/>>

²<<https://xsom.java.net/>>

Network/Graph framework (JUNG)³; e (iv) Jena API⁴ usa o XSG como entrada para gerar entidades OWL.

Uma ferramenta foi desenvolvida pelos autores. Cada fonte de informação é mapeada para a sua própria ontologia local.

A pesquisa ilustra como outras tecnologias podem ser utilizadas na elaboração de um *framework* que gerencia a criação automática de ontologia OWL de fontes de dados XML. Contudo, no Capítulo 7 é utilizado XSLT como tecnologia de transformação por sua simplicidade de uso.

- (LACOSTE; SAWANT; ROY, 2011):

EXCO é uma abordagem completamente automatizada. Ela tem duas partes independentes: (i) o modelo extrator gera o modelo OWL a partir do XSD e (ii) o gerador de instância que produz OWL do XML e XSD. Estes dois passos podem ser paralelizados.

Os autores propuseram um *framework* que suporta referências internas em XML Schemas, tanto para elementos quanto para tipos. O mapeamento é capaz de processar múltiplas instâncias XML.

O *framework* exemplifica como é possível atuar em nível de TBox quanto de ABox independentemente. Na dissertação, no entanto, é trabalhado em nível de ABox, por já se ter ontologias para o TBox.

- (LIANG et al., 2015):

Os autores apresentam uma proposta de conversão semiautomática. A abordagem é dividida em quatro passos e gerencia três conjuntos de dados relacionados a saúde. No primeiro passo são preparadas as descrições. O segundo passo converte os significados semânticos em valores armazenados de acordo com a descrição correspondente. O terceiro é responsável por obter todos os metadados e transpô-los em classes OWL. Por último, o quarto passo transforma os dados de origem em OWL.

Eles optaram por uma hipótese que transformar dados em formato OWL sem modelos pré-definidos permite que diferentes recursos sejam conectados de forma genérica e se comunicam com ontologias existentes sem uma sobrecarga de engenharia. Os autores exploram alinhamentos de conceitos com ontologias existentes, tais como classes e indivíduos.

³<<http://jung.sourceforge.net>>

⁴<<https://jena.apache.org/>>

Diferentemente deste trabalho, a dissertação pressupõe que a predefinição de um modelo ontológico é mais aconselhada. Por outro lado, concorda-se com o fato de que o alinhamento de conceitos entre ontologias existentes deva ser realizado.

5.2 Sumário dos Trabalhos

A Tabela 5.1 apresenta o sumário dos trabalhos relacionados. A coluna **Artigo** apresenta a referência analisada, as demais colunas apresentam “√” quando o trabalho analisado é correlato à área ou “-” quando não foi utilizado. Quando a coluna **Ontologia** apresentar “+” sinaliza que a ontologia do trabalho realiza a verificação de cardinalidade dos metadados. Quando a coluna **Interoperabilidade** apresentar (i) significa que é baseado em instâncias e (ii) significa que é baseado no esquema.

Tabela 5.1: Sumário dos Trabalhos Relacionados

Artigo	Ontologia	Interoperabilidade
(CASALI et al., 2013)	√+	-
(GARCÍA et al., 2012)	√	-
(CHOE, 2013)	√	-
(SOLOMOU; PIERRAKEAS; KAMEAS, 2015)	√	-
(MENOLLI et al., 2014)	√	-
(GLUZ; VICARI, 2012)	√	-
(BEHR; PRIMO; VICCARI, 2012)	√+	-
(HACHEROUF; BAHLOUL; CRUZ, 2015)	√	√
(BEDINI et al., 2011)	√	√(ii)
(O’CONNOR; DAS, 2011)	√	√(i)
(YAHIA; MOKHTAR; AHMED, 2012)	√	√(i)
(LACOSTE; SAWANT; ROY, 2011)	√	√(ii)
(LIANG et al., 2015)	√	√(i)

Fonte: Elaborado pelo autor

5.3 Considerações do Capítulo

O desenvolvimento de ontologias modularizadas para metadados educacionais não foi encontrado na literatura corrente. A descrição dos metadados é realizada em uma única ontologia de representação de conhecimento. Este fato pode ocasionar uma limitação no reuso da ontologia, por exemplo em uma composição de perfil de aplicação, visto que se somente um subconjunto de axiomas é necessário e que os demais não utilizados

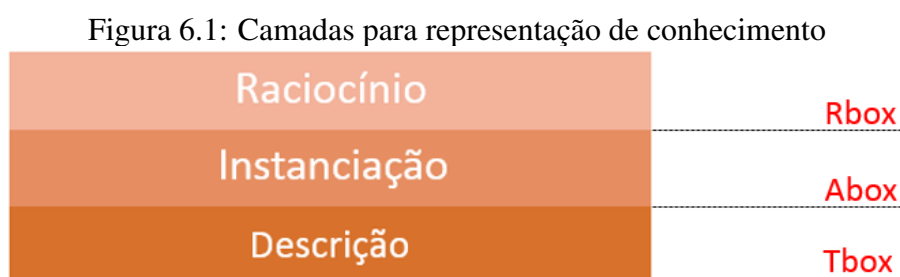
aumentam o tempo de processamento do motor de inferências. Isto pode acarretar no desenvolvimento de nova ontologia.

Por fim, trabalhos relacionados com a interoperabilidade de metadados são predominantemente baseados nas instâncias XML ou nos esquemas XML. Eles não possuem nenhuma ontologia associada a priori e nem trabalham com transformações entre ontologias.

6 REPRESENTAÇÃO DE CONHECIMENTO

O método proposto para a representação de conhecimento para metadados educacionais divide-se em três camadas. A primeira é composta pelo domínio de conhecimento, a segunda pelos objetos que fazem parte deste domínio e a terceira é usada para raciocínio.

A Figura 6.1 ilustra esta composição, onde cada camada se reutiliza do conhecimento representado pela camada inferior. Existe uma clara distinção entre as camadas: Descrição, associada ao conceito de TBox; Instanciação, relacionado a ABox; e Raciocínio, associado a RBox¹.



Fonte: Elaborado pelo autor

Como exemplo prático, é apresentada a descrição de um subconjunto do padrão de metadados IEEE-LOM e suas correspondentes ontologias do nível de descrição, ontologias para cada um dos padrões mencionados no Capítulo 2 para o nível de instanciação e dois perfis de aplicação para execução de raciocínio no último nível. As ontologias foram desenvolvidas com OWL 2 Full.

6.1 Camada de Descrição

A primeira camada é composta por ontologias que descrevem esquemas de metadados. Como por exemplo IEEE-LOM, OBAA, IMS AccessForAll e VideoAula@RNP. Nesta seção será apresentada uma abordagem baseada em múltiplas ontologias para representar estes padrões de metadados hierárquicos. As ontologias resultantes podem ser acessadas através das URIs disponíveis no Apêndice C.

As ontologias do nível um são usadas para descrever classes e propriedades que são usadas para representar indivíduos nas ontologias de nível dois. Este nível contém ontologias com a semântica para esquemas de metadados: suas cardinalidades, proprie-

¹Na literatura o conceito de RBox (Role Box) é comumente relacionado a funções (transitividade, simetria, etc). Neste trabalho, a abordagem do conceito foi expandida para uma relação com motores de inferências, ou seja, uma “Reasoning Box”, visto que estas relações de funções foram realizadas na TBox.

dades, domínios, contradomínios e axiomas que são necessários para descrever qualquer domínio de aplicação.

A hierarquia dos metadados é transposta em uma hierarquia de classes na ontologia. Cada categoria, elementos agregados e simples têm seus metadados como classes e subclasses. Elementos simples também se tornam propriedade de dados; elementos agregados e categorias geram propriedades de objetos de acordo com a semântica de cada um. Esta semântica dos metadados torna-se axiomas de restrição.

Nesta parte, a metodologia de criação de ontologias segue o trabalho anterior realizado em Behr (2014). A modularização da ontologia segue o padrão 1 (um módulo importa n módulos) proposto por Abbas et al. (2012) e foi realizada de modo *bottom-up*². Assim, em um primeiro momento, cada metadado pertencente a um grupo agregado irá compor uma ontologia. Esta ontologia representa um pequeno módulo de toda representação de conhecimento dos metadados. Deste modo, existem três ontologias para *Identifier* (pertencentes a *General*, *Meta-Metadata* e *Resource*), duas para *Contribute* (pertencentes a *LifeCycle* e *Meta-Metadata*), uma *Resource*, uma *Requirement* e uma *OrComposite*. Além disso, cada categoria (dos grupos principais) gera uma ontologia, isto é, para *General*, *LifeCycle*, *Meta-Metadata*, *Technical*, *Educational*, *Rights*, *Relation*, *Annotation* e *Classification*.

A preferência pela criação de mais de uma ontologia com mesmo nome foi tomada para que não haja perda de semântica. Como por exemplo o metadado *Entity* da categoria *LifeCycle* que possui cardinalidade máxima 40, diferentemente do metadado *Entity* da categoria *Meta-Metadata* que possui cardinalidade máxima 10. Deste modo, a atualização da ontologia de determinada categoria ou grupo agregado fica mais fácil. Além disso, esta separação contribui para que o tipo correto seja atribuído a indivíduos na camada de raciocínio.

As ontologias também contêm propriedade de dados³ para cada metadado simples (elementos de dados simples) e uma propriedade de objeto de sua superclasse. Estes axiomas são baseados nas definições dos padrões de metadados IMS AccessForAll, IEEE-LOM, OBAA e VideoAula@RNP.

Com o objetivo de não se perder a hierarquia principal, uma ontologia final é proposta onde contém a representação completa de metadados, importando as demais

²Infelizmente, a reutilização de ontologias compromete a utilização de motores de inferências em certas APIs. Como por exemplo, a ontologia vCard <<https://www.w3.org/TR/vcard-rdf/>> que causa uma exceção de datatype não suportado ao utilizá-la na OWLAPI <<http://owlapi.sourceforge.net/>>.

³Deu-se prioridade ao uso do esquema padrão XSD nos contradomínios para se manter a compatibilidade com XML.

ontologias e realizando relacionamentos *owl:subClassOf*. Dezenove ontologias compõem a representação final do IEEE-LOM, treze para o IMS AccessForAll, 47 para o OBAA e 52 para o VideoAula@RNP.

Como exemplo, a partir da categoria *General*, descrita pela Tabela 6.1, tem-se duas ontologias: a primeira delas é a ontologia *General*, composta pelos metadados *General*, *Title*, *Language*, *Description*, *Keyword*, *Coverage*, *Structure* e *Aggregation Level*; e a segunda é a ontologia *Identifier*, composta pelos metadados *Identifier*, *Catalog* e *Entry*. Optou-se por não importar a ontologia *Identifier* na ontologia *General* para deixar as ontologias mais autocontidas e independentes quanto possível.

Tabela 6.1: Grupo General do IEEE-LOM

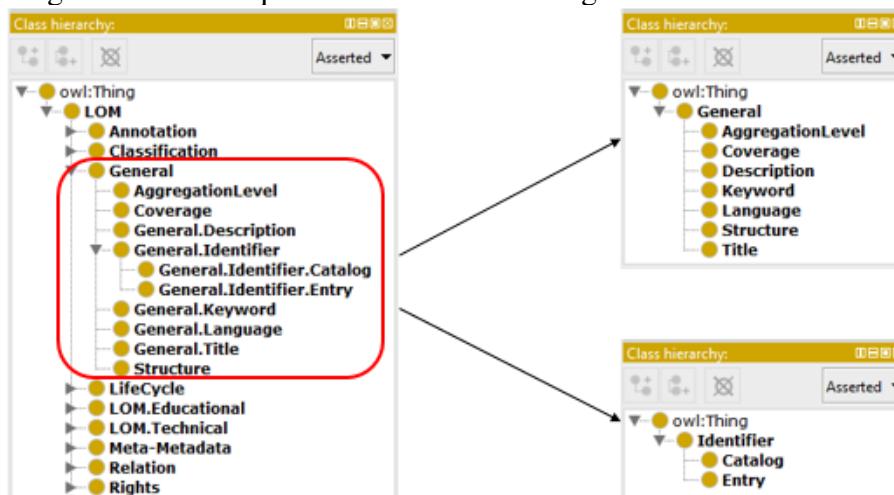
Nr	Nome	Cardinalidade	Espaço de Valores	Tipo de Dado
1	General	1		
1.1	Identifier	10		
1.1.1	Catalog	1		CharacterString (max. 1000 char)
1.1.2	Entry	1		CharacterString (max. 1000 char)
1.2	Title	1		LangString (max. 1000 char)
1.3	Language	10	ISO/IEC 10646-1:2000	CharacterString (max. 1000 char)
1.4	Description	10		LangString (max. 1000 char)
1.5	Keyword	10		LangString (max. 1000 char)
1.6	Coverage	10		LangString (max. 1000 char)
1.7	Structure	1	atomic, collection, networked, hierarchical, linear	Vocabulary
1.8	Aggegation Level	1	1, 2, 3, 4	Vocabulary

Fonte: Adaptado de (LTSC, 2002)

A hierarquia de classes das ontologias e a modularização efetuada são ilustradas pela Figura 6.2. As ontologias possuem uma propriedade de objeto cada para interligar os indivíduos que instanciarão sua representação na camada seguinte, sendo elas *hasGeneral* e *hasIdentifier*. Além disto, elas possuem as propriedades de dados *catalog*, *entry*, *title*, *language*, *description*, *keyword*, *coverage*, *structure* e *aggregationLevel*. As cardinalida-

des são definidas em nível de superclasse⁴. Por exemplo, a classe *Title* é subclasse de *title exactly 1 xsd:string*⁵ e a classe *General* é subclasse de *hasGeneral max 1 owl:Thing*. A propriedade de dado *title* possui *Title* e *xsd:string* como domínio e contradomínio, respectivamente. Já *hasGeneral* possui *General* e *owl:Thing* como domínio e contradomínio, respectivamente. A declaração de um domínio e um contradomínio permite que se faça uma inferência de um possível tipo de um recurso, mesmo que esse tipo não esteja declarado explicitamente por meio de uma tripla. Em caso de espaço de valores definido, como para as propriedades de dados *structure* e *aggregationLevel*, o contradomínio é definido como um conjunto de strings e intervalo de números válidos: {"atomic", "collection", "hierarchical", "linear", "networked"} e *xsd:int*[>= "1"^^xsd:int , <= "4"^^xsd:int], respectivamente.

Figura 6.2: Hierarquia de classes das ontologias General e Identifier.



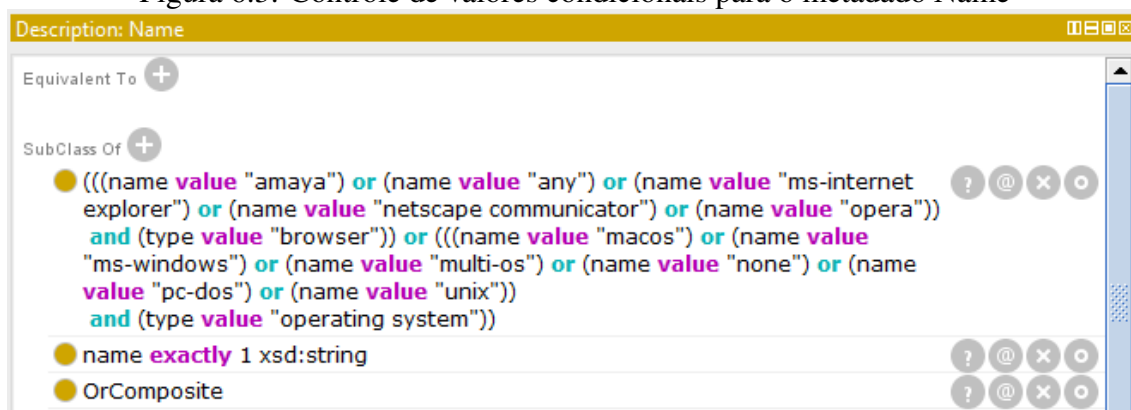
Fonte: Elaborado pelo autor

Metadados com espaço de valores condicionais também são verificados em nível de superclasse. Por exemplo, o metadado *Name* tem conteúdos associados com o metadado *Type*, como ilustrado pela Figura 6.3. Quando *Type* possuir conteúdo *browser*, *Name* deverá ter conteúdo *amaya*, *any*, *ms-internet-explorer* ou *opera*; com conteúdo de *Type* como *operating-system*, *Name* pode possuir *macos*, *ms-windows*, *multi-os*, *none*, *pc-dos* ou *unix*; de acordo com o padrão de metadados IEEE-LOM.

⁴Cardinalidades maiores que um foram definidas com o quantificador existencial *some* devido ao desempenho do reasoner. A verificação de acordo com o padrão é realizada na camada de raciocínio pelo fato de que os motores de inferência ainda não suportarem *built-in atoms* utilizados nas regras SWRL.

⁵Por motivos de desempenho, quando uma cardinalidade máxima é um, ela também é definida como funcional.

Figura 6.3: Controle de valores condicionais para o metadado Name



Fonte: Elaborado pelo autor

6.2 Camada de Instanciação

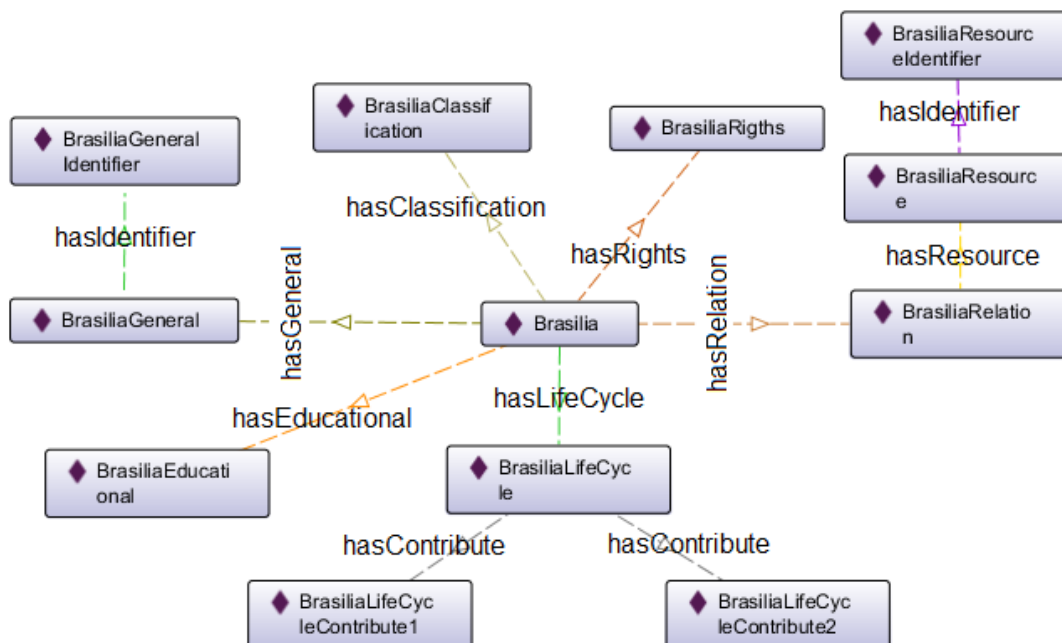
A segunda camada descreve as instanciações de metadados de OAs, ou seja, seus conteúdos e seus relacionamentos hierárquicos através de propriedades das ontologias do primeiro nível. As ontologias de nível dois são caracterizadas por importar n ontologias do nível um e são compostas por n indivíduos com propriedades de dados e propriedades de objetos para descrevê-los.

Geralmente, pode-se aplicar algum mecanismo de raciocínio para verificar a consistência de um indivíduo através de uma ontologia do nível de descrição. Por exemplo, se descrevermos um OA com o padrão OBAA, podemos verificar se o contradomínio e espaço de valores estão propriamente utilizados. Além disto, devido as características de uma ontologia, se alguma descrição está incorreta pode-se aplicar um algoritmo de explicação para ajudar o usuário a corrigir seu erro.

Considerando propósitos de armazenamento, estas ontologias podem ser guardadas em algum repositório formal, como por exemplo um repositório de triplas ou até mesmo simplesmente definindo uma URI de acesso em um servidor. Estas alternativas dão flexibilidade a projetistas de conteúdo que podem simplesmente construir e publicar livremente na Web.

O desenvolvimento de ontologias do segundo nível se dá através da importação de uma ontologia do nível de descrição correspondente à categoria ou ao grupo agregado que se deseja descrever, cria-se um indivíduo que é associado a quantas propriedades de dados sejam necessárias. Por fim, para a representação completa dos metadados do OA importa-se todas as ontologias de nível dois relacionadas, cria-se um indivíduo principal e estabelecem-se relações entre os indivíduos por propriedades de objeto.

Figura 6.4: Relacionamento entre indivíduos do OA Brasília.



Fonte: Elaborado pelo autor

A seguir, serão apresentados alguns exemplos de representações para instanciações de metadados de OAs. Respectivamente para os padrões Dublin Core, IEEE-LOM, OBAA e VideoAula@RNP. Para fins de ilustração as propriedades de dados foram suprimidas, mas podem ser acessadas através das URIs.

6.2.1 Brasília

Os metadados do objeto de aprendizagem Brasília⁶ foram recuperados a partir do Banco Internacional de Objetos Educacionais (BIOE) no formato Dublin Core. A transposição dos conteúdos de metadados para IEEE-LOM foi realizada conforme a Tabela A.1.

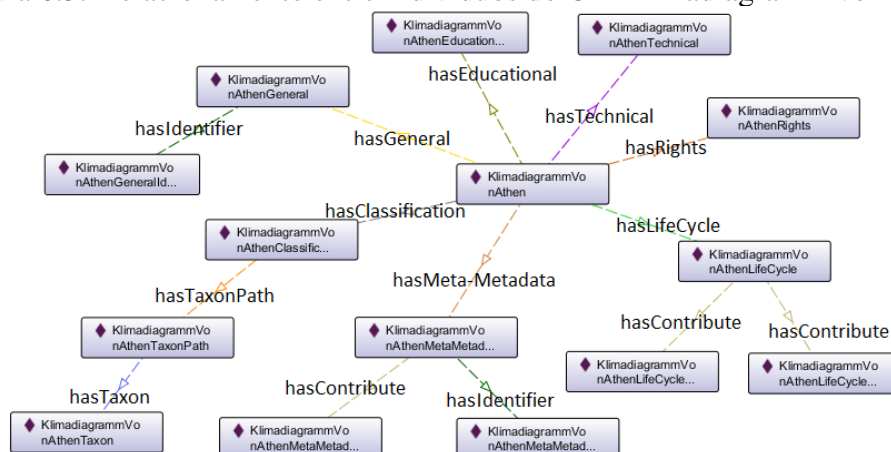
A Figura 6.4 ilustra a visão geral da representação ontológica do OA. Ela é composta por oito ontologias, cada uma representada por um retângulo. A ontologia central (Brasilia) realiza a importação das demais e estabelece a ligação entre elas com propriedades de objeto. Esta ontologia pode ser acessada em: <<http://gia.inf.ufrgs.br/Repository/20160913174652139/Brasilia.owl>>.

⁶<<http://objetoseducacionais2.mec.gov.br/handle/mec/430?show=full>>

6.2.2 Klimadiagramm von Athen

O objeto de aprendizagem Klimadiagramm von Athen possui seus metadados⁷ no padrão IEEE-LOM. A Figura 6.5 apresenta as ontologias que compõem a representação deste OA. Nota-se uma hierarquização maior referente a especificação do padrão IEEE-LOM, com mais propriedades de objetos relacionando os quinze indivíduos das suas respectivas ontologias. A representação completa dos metadados como ontologia pode ser acessado em: <<http://gia.inf.ufrgs.br/Repository/20160912164622189/KlimadiagrammVonAthen.owl>>.

Figura 6.5: Relacionamento entre indivíduos do OA Klimadiagramm von Athen.



Fonte: Elaborado pelo autor

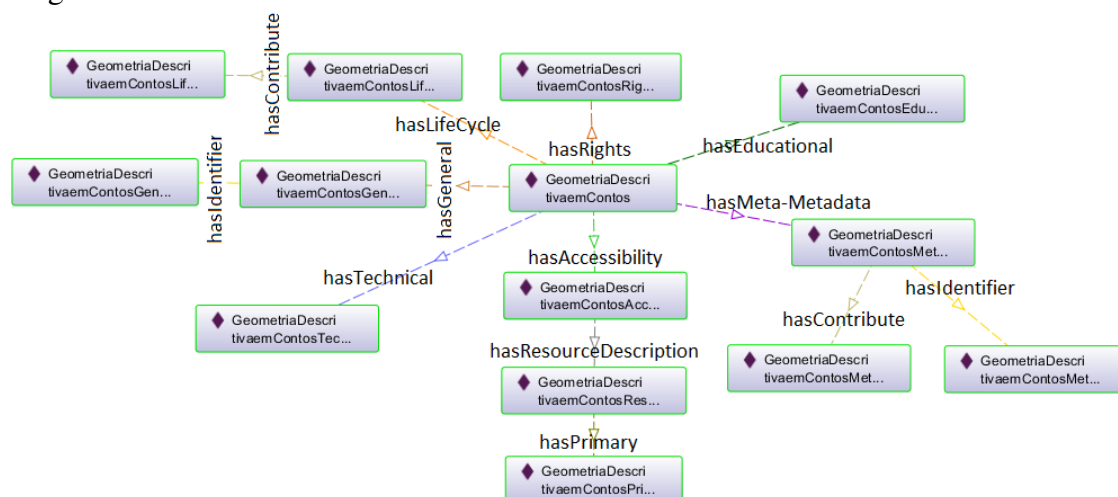
6.2.3 Geometria Descritiva em Contos

Os metadados do objeto de aprendizagem Geometria Descritiva em Contos⁸ estão descritos através do padrão OBAA de metadados. A representação ontológica pode ser acessada em: <<http://gia.inf.ufrgs.br/Repository/20160913195141156/GeometriaDescritivaemContos.owl>>. A Figura 6.6 ilustra tal representação, onde quinze indivíduos estão relacionados.

⁷<http://www.melt.fwu.de/oai2.php?verb=GetRecord&metadataPrefix=oai_lom&identifier=MELT-04601078.1>

⁸<<http://cognix-repo.inf.ufrgs.br/repositorio/oai?verb=ListRecords&metadataPrefix=obaa>>

Figura 6.6: Relacionamento entre indivíduos do OA Geometria Descritiva em Contos.



Fonte: Elaborado pelo autor

6.2.4 Infraestrutura do Serviço Videoaula@RNP: Módulo 2

A descrição do OA Infraestrutura do Serviço Videoaula@RNP: Módulo 2⁹ é realizada através de metadados no padrão Videoaula@RNP. A representação é claramente mais hierarquizada que os demais exemplos, conforme a Figura 6.7. Nesta representação, 28 indivíduos se relacionam para representar os conteúdos dos metadados. A ontologia pode ser acessada em: <<http://gia.inf.ufrgs.br/Repository/20160913192923980/Modulo2.owl>>.

6.3 Camada de Raciocínio

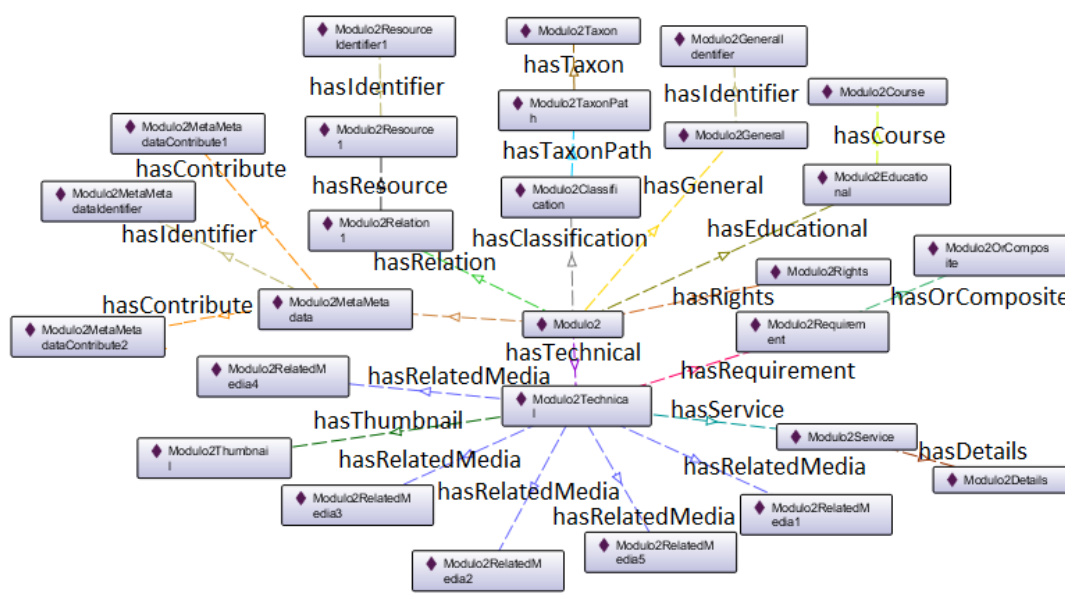
A terceira camada compreende a descrição de perfis de aplicação como declarações lógicas. Deste modo, elas fornecerão mecanismos de raciocínio sobre as ontologias do nível dois.

Esta camada é desenvolvida principalmente para representar perfis de aplicações. Neste trabalho, um perfil de aplicação é uma abstração que é situada em uma prática de derivação de algum conhecimento baseado em axiomas aplicados sobre uma rede ontológica. Este processo de raciocínio pode ser usado para derivar, por exemplo, OAs que tenham algum aspecto pedagógico, OAs destinados para pessoas com necessidades especiais, dentre outros.

Duas alternativas de raciocínio a partir de perfis de aplicações são propostas. A

⁹<http://http://videoaula.rnp.br/riortransfer/rnp/treinamentos/videoaulas/modulo_2/modulo_2.xml>

Figura 6.7: Relacionamento entre indivíduos do OA Infraestrutura do Serviço Videoaula@RNP: Módulo 2.



Fonte: Elaborado pelo autor

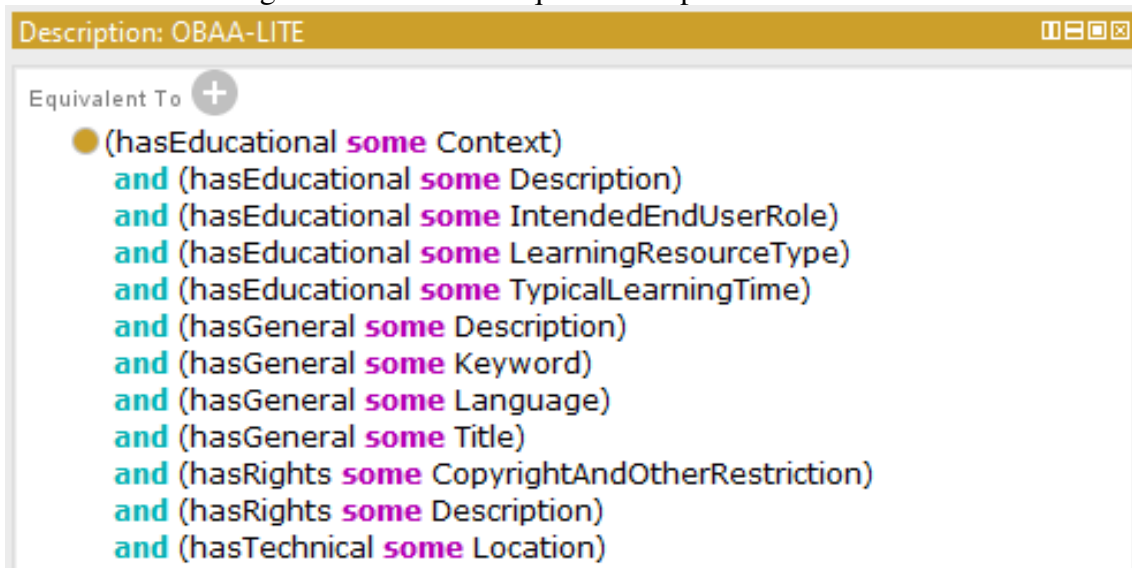
primeira delas é a descrição do perfil de aplicação como uma classe de equivalência, composta por axiomas que possibilitam inferir quais indivíduos são instâncias desta classe. Na segunda, o perfil de aplicação é descrito como um conjunto de regras para a realização de uma consulta SQWRL.

Como exemplo, os perfis de aplicação OBAA-LITE e OBAA-VideoAula serão apresentados como classes de equivalências e conjunto de regras. Estes perfis compõem os metadados mínimos para a representação nos padrões OBAA e VideoAula@RNP, respectivamente.

6.3.1 OBAA-LITE

OBAA-LITE é um modelo simplificado do padrão OBAA de metadados que visa facilitar a descrição dos metadados de OAs pelos seus desenvolvedores e implantá-los em repositórios com elementos de fato utilizados pelos usuários-professores (SILVA, 2011). A Figura 6.8 descreve os metadados do perfil de aplicação OBAA-LITE na forma de uma classe de equivalência. De outro modo, a Figura 6.9 apresenta um conjunto de regras SWRL para este perfil. O conjunto de metadados também pode ser encontrado no Anexo B.

Figura 6.8: Classe de equivalência para OBAA-LITE



Fonte: Elaborado pelo autor

Figura 6.9: Conjunto de regras para OBAA-LITE

```

hasGeneral(?lo, ?g) ∧ General : Description(?g) ∧
General : Keyword(?g) ∧ General : Language(?g) ∧ Title(?g) ∧
hasTechnical(?lo, ?t) ∧ Location(?t) ∧
hasEducational(?lo, ?e) ∧ Context(?e) ∧
Educational : Description(?e) ∧ IntendedEndUserRole(?e) ∧
LearningResourceType(?e) ∧ TypicalLearningTime(?e) ∧
hasRights(?lo, ?r) ∧ CopyrightAndOtherRestriction(?r) ∧
Rights : Description(?r)
→ sqwrl : select(?lo)

```

Fonte: Elaborado pelo autor

6.3.2 OBAA-VideoAula

OBAA-VideoAula é um conjunto de metadados considerados mínimos para representação dos metadados de um OA pelo padrão de metadados VideoAula@RNP (BEHR, 2014). A classe de equivalência que descreve tais metadados é ilustrada pela Figura 6.10. Além desta, a Figura 6.11 representa o seu conjunto de regras SWRL. O conjunto de metadados também pode ser encontrado no Anexo B.

Figura 6.10: Classe de equivalência para OBAA-VideoAula

Description: OBAA-VideoAula

Equivalent To 

- (hasGeneral some Description)
- and (hasGeneral some Keyword)
- and (hasGeneral some Language)
- and (hasGeneral some Title)
- and (hasLifeCycle some Status)
- and (hasMeta-Metadadata some MetadadataSchema)
- and (hasRights some CopyrightAndOtherRestriction)
- and (hasRights some Description)
- and (hasTechnical some Duration)
- and (hasTechnical some Format)
- and (hasTechnical some Location)
- and (hasTechnical some Size)
- and (hasTechnical some SupportedPlatforms)
- and (hasGeneralChain some Catalog)
- and (hasGeneralChain some Entry)
- and (hasLifeCycleChain some Date)
- and (hasLifeCycleChain some Entity)
- and (hasLifeCycleChain some Role)
- and (hasMeta-MetadadataChainContribute some Date)
- and (hasMeta-MetadadataChainContribute some Entity)
- and (hasMeta-MetadadataChainContribute some Role)
- and (hasMeta-MetadadataChainIdentifier some Catalog)
- and (hasMeta-MetadadataChainIdentifier some Entry)
- and (hasTechnicalChainDetails some ServiceLocation)
- and (hasTechnicalChainOrComposite some Name)
- and (hasTechnicalChainOrComposite some Type)
- and (hasTechnicalChainRelatedMedia some AspectRatio)
- and (hasTechnicalChainRelatedMedia some Catalog)
- and (hasTechnicalChainRelatedMedia some Entry)
- and (hasTechnicalChainRelatedMedia some Standard)
- and (hasTechnicalChainService some Essential)
- and (hasTechnicalChainService some Name)
- and (hasTechnicalChainService some Protocol)
- and (hasTechnicalChainService some Provides)
- and (hasTechnicalChainService some Type)
- and (hasTechnicalChainThumbnail some Catalog)
- and (hasTechnicalChainThumbnail some Entry)

Fonte: Elaborado pelo autor

Figura 6.11: Conjunto de regras para OBAA-VideoAula

```

hasGeneral(?lo,?gn)∧General : Description(?gn)∧General : Keyword(?gn)∧
General : Language(?gn) ∧ General : Title(?gn) ∧ General_Identifier :
hasIdentifier(?gn,?gni)∧General_Identifier : Catalog(?gni)∧General_Identifier :
Entry(?gni)∧hasLifeCycle(?lo,?lc)∧Status(?lc)∧LifeCycle_Contribute :
hasContribute(?lc,?lcc)∧LifeCycle_Contribute : Date(?lcc)∧LifeCycle_Contribute :
Entity(?lcc)∧LifeCycle_Contribute : Role(?lcc)∧hasMeta-Metadada(?lo,?mm)∧
MetadataSchema(?mm)∧Meta-Metadada_Contribute : hasContribute(?mm,?mmc)∧
Meta-Metadada_Contribute : Date(?mmc)∧Meta-Metadada_Contribute :
Entity(?mmc) ∧ Meta - Metadada_Contribute : Role(?mmc)∧
Meta - Metadada_Identifier : hasIdentifier(?mm,?mmi) ∧ Meta -
Metadata_Identifier : Catalog(?mmi) ∧ Meta - Metadata_Identifier :
Entry(?mmi)∧hasTechnical(?lo,?tc)∧Technical : Duration(?tc)∧Technical :
Format(?tc)∧Technical : Location(?tc)∧Technical : Size(?tc)∧Technical2 :
SupportedPlatforms(?tc)∧hasRequirement(?tc,?rq)∧hasOrComposite(?rq,?oc)∧
OrComposite : Name(?oc)∧OrComposite : Type(?oc)∧hasRelatedMedia(?tc,?rm)∧
RelatedMedia : Catalog(?rm)∧RelatedMedia : Entry(?rm)∧AspectRatio(?rm)∧
Standard(?rm) ∧ hasThumbnail(?tc,?th) ∧ Thumbnail : Catalog(?th) ∧
Thumbnail : Entry(?th) ∧ hasService(?tc,?sr) ∧ Service : Essential(?sr)∧
Service : Protocol(?sr)∧Service : Provides(?sr)∧Service : Name(?sr)∧
Service : Type(?sr)∧hasDetails(?sr,?dt)∧ServiceLocation(?dt)∧hasRights(?lo,?rg)∧
CopyrightAndOtherRestriction(?rg)∧Rights : Description(?rg) → sqwrl : select(?lo)

```

Fonte: Elaborado pelo autor

6.3.3 Cardinalidades

Além da inferência sobre perfis de aplicação, esta camada também pode ser responsável pela verificação de cardinalidades máximas mais abrangentes¹⁰. Por exemplo, o metadado simples *Keyword* da categoria *General* possui cardinalidade máxima 10. Para que o *reasoner* possa escalar, ele deve operar sobre mundo fechado. Assim, a verificação pode ser determinada por uma regra SWRL tal como ilustrado pela Figura 6.12.

Figura 6.12: Regra para verificação de cardinalidade do metadado *Keyword* da categoria *General*

```

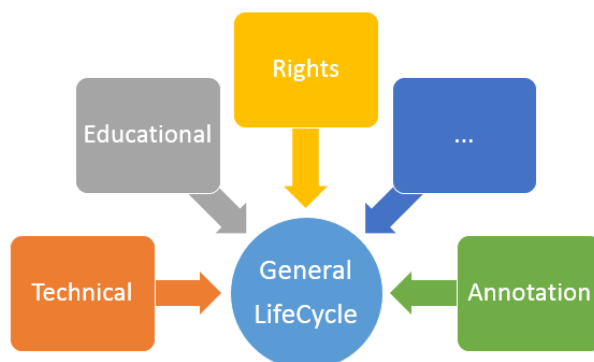
General : keyword(?i,?k) ∧ sqwrl : makeSet(?s,?k) ∧
sqwrl : size(?siz,?s) ∧ swrlb : greaterThan(?siz,10)
→ sqwrl : select(?i)

```

Fonte: Elaborado pelo autor

¹⁰Neste trabalho, uma cardinalidade abrangente é considerada quando for maior que um.

Figura 6.13: Composição de ontologia de metadados a partir de diferentes fontes na Web Semântica.



Fonte: Elaborado pelo autor

6.4 Considerações do Capítulo

As ontologias propostas objetivam facilitar o reuso em qualquer uma das três camadas de representação de conhecimento. Na primeira camada, a modularidade das ontologias facilita a composição e adição de metadados ou até mesmo o uso de módulos em separado. Na segunda, a abordagem com múltiplas ontologias para descrever a completude dos metadados tenta fragmentar e aumentar o reuso das suas partes instanciadas. Deste modo, uma nova ontologia de metadados pode ser composta pela união destas várias partes de ontologias autocontidas.

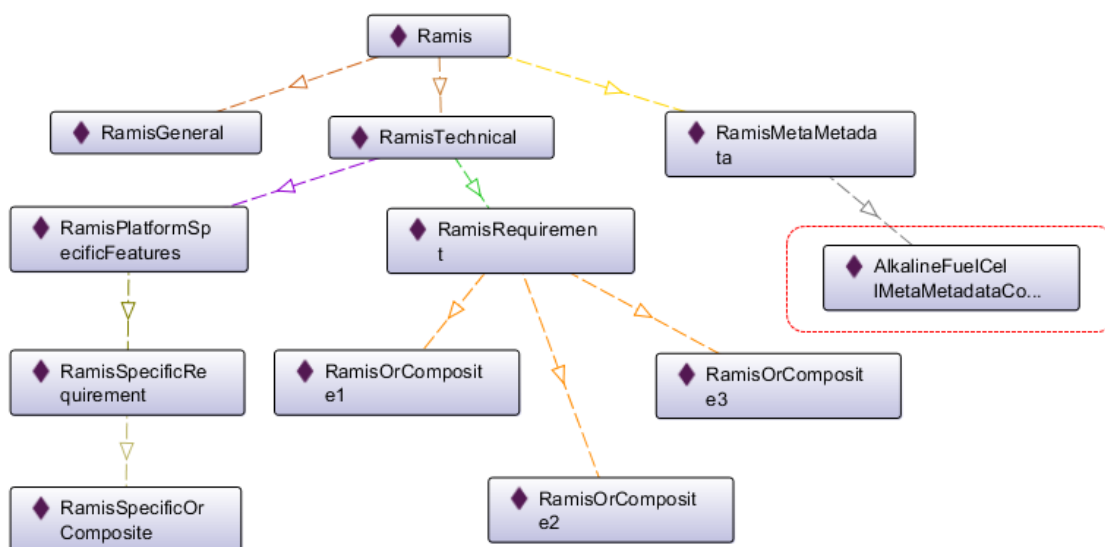
A OWL fornece somente o método denominado *owl:import* para importação de demais ontologias¹¹. A Figura 6.13 exemplifica uma possibilidade de uma ontologia que já possui metadados para General e LifeCycle reusar outras partes de metadados de outros OAs, tais como Technical, Educational, Rights, Annotation, ou qualquer (...) outro novo conjunto de metadados.

Frequentemente, não é desejado reusar todos os metadados de algum padrão para representação de OAs. Isto é uma das desvantagens do estado da arte dos trabalhos relacionados, pelo fato de que em OWL quando uma ontologia é importada, todos os axiomas são adicionados na representação de conhecimento corrente.

A presente proposta em múltiplas ontologias objetiva melhorar o reuso, oferecendo porções de representações dos metadados para importação. Por exemplo, um determinado contribuidor descrito nos metadados pode ser o mesmo em vários OAs. Assim, este metadado pode ser reusado. Isto é ilustrado pela Figura 6.14 onde a ontologia Con-

¹¹Extensões da OWL são encontradas na literatura com o objetivo de trabalhar de outra forma a combinação de ontologias, tais como os formalismos Distributed Description Logics (BORGIDA; SERAFINI, 2003) e E-Connections (KUTZ et al., 2004).

Figura 6.14: Reuso do indivíduo Meta-Metadada Contribute de ontologia externa.



Fonte: Elaborado pelo autor

tribute de Meta-Metadada é importada pela sua URI e relacionada com uma propriedade de objeto da representação de AlkalineFuelCell. Os nomes dos indivíduos são somente para evidenciar o reuso para diferentes OAs, eles podem ter nomes genéricos.

Além disto, no momento da descrição de conhecimento, além de escolher um perfil de aplicação, os perfis podem ser estendidos a partir destes módulos autocontidos de ontologias. Deste modo, deixa-se a cargo do utilizador seleccionar quais os grupos de metadados a serem usados.

A modularidade das ontologias também permite reusar mesmos nomes de metadados em ontologias diferentes pois possuem URIs distintas. Tal fato não é possível em uma mesma ontologia devido a violação do Unique Name Assumption (UNA)¹².

Na camada de raciocínio foi dada preferência pela representação de perfis de aplicação como classes de equivalência ou regras SQWRL por estarem amparadas por reasoner e por serem armazenadas como ontologias OWL, unificando a representação de conhecimento proposta. Contudo, perfis de aplicação também podem ser representadas através de consultas SPARQL, como representado na Figura 6.15, onde o OA Ramis é classificado pelo perfil de aplicação OBAA-LITE.

¹²A utilização do editor de ontologias Protégé apresentou algumas dificuldades em gerenciar nomes iguais de entidades. Seguidamente teve-se que editar manualmente os aliases das ontologias para que a descrição fosse feita de forma consistente

Figura 6.15: Classificação por consulta SPARQL

The image shows a window titled "SPARQL query:" with a purple header bar. The query text is as follows:

```
PREFIX lom: <http://gia.inf.ufrgs.br/ontologies/LOM.owl#>
SELECT DISTINCT ?learning_object WHERE {
  ?learning_object lom:general.Description ?general_description .
  ?learning_object lom:general.Keyword ?general_keyword .
  ?learning_object lom:general.Language ?general_language .
  ?learning_object lom:general.Title ?general_title .
  ?learning_object lom:location ?locattion .
  ?learning_object lom:copyrightAndOtherRestriction ?copyrightAndOtherRestriction .
  ?learning_object lom:rights.Description ?rights_description .
  ?learning_object lom:hasEducational ?educational .
  ?educational lom:context ?context .
  ?educational lom:educational.Description ?educational_description .
  ?educational lom:intendedEndUserRole ?intendedEndUserRole .
  ?educational lom:learningResourceType ?learningResourceType .
  ?educational lom:typicalLearningTime ?typicalLearningTime
}
```

Below the query, there is a table with one column labeled "learning_object" and one row containing the value "Ramis".

learning_object
Ramis

Fonte: Elaborado pelo autor

7 MÉTODO PARA INTEROPERABILIDADE

Nesta abordagem, a ontologia descrita no capítulo anterior é reutilizada como um provedor de semântica para um processador XSLT que é responsável por transpor instâncias de metadados em descrições ontológicas. Uma ontologia para produzir o processador XSLT proporciona uma maior semântica e expressividade aos metadados em XML. Criações automáticas podem dirigir a diferentes ontologias (i.e. com divergentes URIs de conceitos) e um dos princípios da Web Semântica é o reuso. Além disso, a ontologia pode ser alinhada com outras ontologias e a representação de conhecimento pode ser expandida. Através disto, conversões entre formatos OWL também podem ser realizadas.

O método proposto é dividido em três fases: (i) recuperação dos metadados a partir de diferentes repositórios, (ii) transposição dos metadados em uma descrição ontológica e (iii) armazenamento dos metadados em um repositório semântico. Esta ideia geral é representada pela Figura 7.1 e as fases serão abordadas nas seções seguintes.

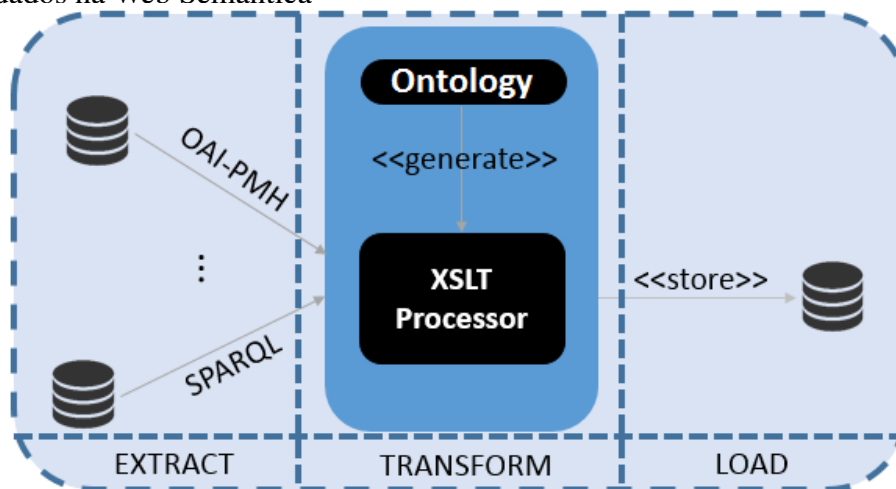
Inicialmente, foram capturados metadados de dois repositórios OBAA¹². No primeiro, dezesseis instâncias de metadados de OAs foram recuperados na forma XML e transpostos para OWL. No segundo, uma conversão realizada anteriormente de mil instâncias de metadados de OAs de XML para OWL (RIBEIRO; PRIMO, 2013) foi utilizada para uma transformação de OWL para OWL. Além disso, metadados em Dublin Core do repositório BIOE³ também foram transpostos para OBAA. Esses dados podem ser armazenados, por exemplo, em banco de dados semânticos (Triple-Store), banco de dados relacionais com um esquema de mapeamento para RDF, etc.

¹<<http://cognix-repo.inf.ufrgs.br/>>

²<<http://repositorio.portalobaa.org/>>

³<<http://objetoseducacionais2.mec.gov.br/>>

Figura 7.1: Abordagem de Extração-Transformação-Carga para prover interoperabilidade de metadados na Web Semântica



Fonte: (BEHR; PRIMO; VICCARI, 2016)

7.1 Recuperação de Metadados

Nesta fase, os metadados educacionais podem ser recuperados de repositórios. Os metadados podem ser descobertos com diferentes tecnologias, tais como OAI-PMH⁴, REST, SPARQL⁵, dentre outros. Para a recuperação de XML em base de dados, a entrada dos metadados pode ser em registro único ou lista de registros (consulta da Figura 7.2). Para repositórios web semânticos, a entrada pode ser como uma ontologia única ou uma lista de URIs de ontologias (obtidos pela saída da consulta SPARQL da Figura 7.3).

Figura 7.2: Consulta OAI-PMH para recuperar uma lista de metadados em XML

```
1 http://feb.ufrgs.br/repositorio/oai?verb=ListRecords&metadataPrefix=obaa
```

Fonte: Elaborado pelo autor

Figura 7.3: Consulta SPARQL para recuperar uma lista de URIs de ontologias

```
1 SELECT *
2 WHERE{
3   ?s ?p ?o .
4   FILTER((STR(?o) = "http://www.w3.org/2002/07/owl#Ontology")).
5 }
```

Fonte: Elaborado pelo autor

⁴<<http://www.openarchives.org/pmh>>

⁵<<http://www.w3.org/TR/rdf-sparql-query/>>

7.2 Processamento de Metadados

Antes do processador XSLT converter os metadados, os dados são normalizados. Por exemplo, o relacionamento do arquivo XSLT correspondente na descrição dos metadados em XML ou OWL e, para conversões OWL, a serialização do arquivo em formato RDF/XML para fácil acesso das triplas. O processamento pode ser entre diferentes formatos da pilha da Web Semântica. Por exemplo, XML para OWL, RDF para OWL e OWL para OWL. É também esperado transpor mapeamentos entre diferentes padrões de metadados, como Dublin Core para IEEE-LOM.

Uma ontologia OBAA⁶ definida proporciona semântica para o XSLT processar os metadados. O processamento pode ser resumido nas seguintes etapas: (i) importação da ontologia de domínio OBAA, (ii) criação de indivíduos com URIs únicas e (iii) atribuição de propriedades de objetos e de dados. Todos estes conceitos são manipulados através de XPath⁷.

7.2.1 XML para OWL

Em XML para OWL, o processamento é feito acessando cada metadado de forma sequencial. A seleção é feita por XPath e então cria-se o indivíduo para a categoria ou metadado agregado, atribui-se a propriedade de dado ou de objeto. A Figura 7.4 exemplifica (a) trecho do código XML dos metadados do OA como entrada para o grupo LifeCycle do IEEE-LOM e (b) a saída de suas triplas. Primeiro, os elementos LifeCycle e Contribute geram indivíduos (à esquerda da tripla) e uma propriedade de objeto (*hasContribute*) relaciona eles. Em seguida, as propriedades de dados são atribuídas.

7.2.2 OWL para OWL

A transformação OWL para OWL é realizada primeiramente através de um alinhamento entre ontologias. Este alinhamento é usualmente feito através de equivalências entre as entidades das ontologias. Por exemplo, a Figura 7.5 ilustra os alinhamentos de propriedades de dados entre ontologias para os metadados *version* e *status*. Através disto,

⁶<<http://gia.inf.ufrgs.br/ontologies/OBAA/OBAA.owl>>

⁷<<https://www.w3.org/TR/xpath/>>

Figura 7.4: XML do Alkaline Fuel Cell LifeCycle (SILVA, 2011) transposto para triplas.

<pre> <obaa> <lifecycle> <version>1.0</version> <status>final</status> <contribute> <role>author</role> <entity>Silva, Vasco Sérgio Correia Freitas </entity> <date>2009-08-31T19:05:41</date> </contribute> </lifecycle> </obaa> </pre>	<table border="1"> <tr><td>AlkalineFuelCell version "1.0"</td></tr> <tr><td>AlkalineFuelCell status "final"</td></tr> <tr><td>AlkalineFuelCell hasContribute AlkalineFuelCellLifeCycleContribute</td></tr> <tr><td>AlkalineFuelCellLifeCycleContribute role "author"</td></tr> <tr><td>AlkalineFuelCellLifeCycleContribute entity "Silva, Vasco Sérgio Correia Freitas"</td></tr> <tr><td>AlkalineFuelCellLifeCycleContribute date "2009-08-31T19:05:41"</td></tr> </table>	AlkalineFuelCell version "1.0"	AlkalineFuelCell status "final"	AlkalineFuelCell hasContribute AlkalineFuelCellLifeCycleContribute	AlkalineFuelCellLifeCycleContribute role "author"	AlkalineFuelCellLifeCycleContribute entity "Silva, Vasco Sérgio Correia Freitas"	AlkalineFuelCellLifeCycleContribute date "2009-08-31T19:05:41"
AlkalineFuelCell version "1.0"							
AlkalineFuelCell status "final"							
AlkalineFuelCell hasContribute AlkalineFuelCellLifeCycleContribute							
AlkalineFuelCellLifeCycleContribute role "author"							
AlkalineFuelCellLifeCycleContribute entity "Silva, Vasco Sérgio Correia Freitas"							
AlkalineFuelCellLifeCycleContribute date "2009-08-31T19:05:41"							
(a)	(b)						

Fonte: Elaborado pelo autor

Figura 7.5: Equivalências das propriedade de dados para os metadados version e status.

```

<EquivalentDataProperties>
  <DataProperty IRI="http://gia.inf.ufrgs.br/ontologies/LOM/LifeCycle.owl#status"/>
  <DataProperty IRI="http://obaa.unisinos.br/obaa22.owl#itsStatusIs"/>
</EquivalentDataProperties>
<EquivalentDataProperties>
  <DataProperty IRI="http://gia.inf.ufrgs.br/ontologies/LOM/LifeCycle.owl#version"/>
  <DataProperty IRI="http://obaa.unisinos.br/obaa22.owl#itsVersionIs"/>
</EquivalentDataProperties>

```

Fonte: Elaborado pelo autor

o XSLT é gerado para a conversão. As triplas são selecionadas também por XPath e cada uma é trocada por sua equivalente.

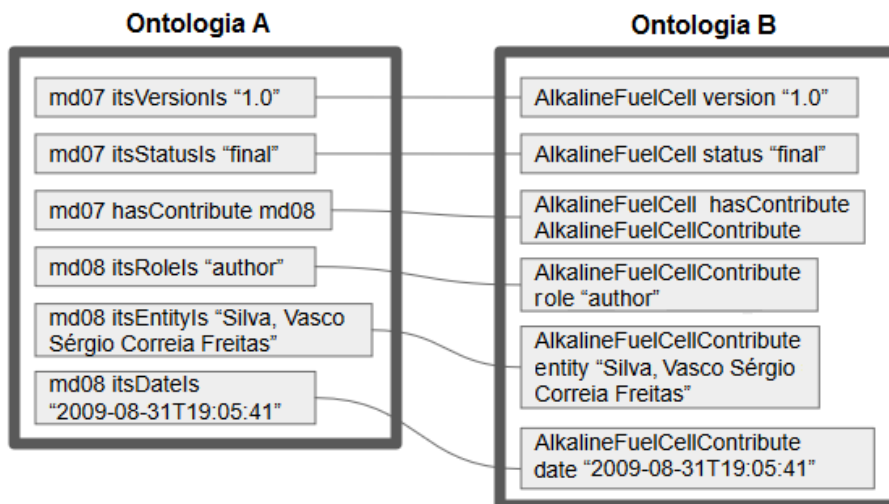
A Figura 7.6 exemplifica parte desta transformação: as triplas da Ontologia A servem como entrada para gerar a saída de triplas na Ontologia B. A transformação troca os nomes de indivíduos, propriedades de dados e de objeto por suas equivalências na outra ontologia.

7.2.3 Dublin Core para OBAA

O padrão de metadados Dublin Core tem um mapeamento direto para o IEEE-LOM, ilustrado pela Tabela A.1. Como o padrão OBAA de metadados estende o IEEE-LOM, os dezesseis elementos do Dublin Core podem ser também transpostos para o OBAA. A Figura 7.7 exemplifica os relacionamentos entre indivíduos gerados por este mapeamento nos metadados do OA Brasília⁸. Algumas das propriedades de objeto e de dados também são ilustradas na Figura 7.8.

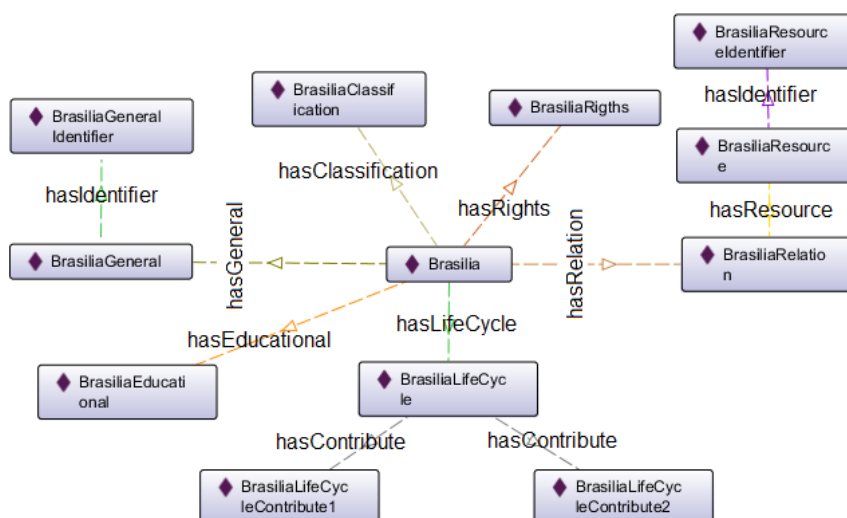
⁸<<http://objetoseducacionais2.mec.gov.br/handle/mec/430?show=full>>

Figura 7.6: Exemplo de equivalências de triplas entre ontologias OBAA.



Fonte: Elaborado pelo autor

Figura 7.7: Relacionamento entre indivíduos do OA Brasília.



Fonte: Elaborado pelo autor

Figura 7.8: Indexação de triplas dos metadados do OA Brasília.

ID	Subject	Predicate	Object
667	.Brasilia	Educational.hasEducational	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaEducational.owl#BrasiliaEducational
668	.Brasilia	General.hasGeneral	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaGeneral.owl#BrasiliaGeneral
669	.Brasilia	LifeCycle.hasLifeCycle	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaLifeCycle.owl#BrasiliaLifeCycle
670	.Brasilia	Relation.hasRelation	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaRelation.owl#BrasiliaRelation
671	.Brasilia	Rights.hasRights	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaRights.owl#BrasiliaRights
672	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaGeneral.owl#BrasiliaGeneral	Identifier.hasIdentifier	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaGeneralIdentifier.owl#BrasiliaGeneralIdentifier
673	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaLifeCycle.owl#BrasiliaLifeCycle	Contribute.hasContribute	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaLifeCycleContribute2.owl#BrasiliaLifeCycleContribute2
674	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaLifeCycle.owl#BrasiliaLifeCycle	Contribute.hasContribute	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaLifeCycleContribute1.owl#BrasiliaLifeCycleContribute1
675	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaRelation.owl#BrasiliaRelation	Resource.hasResource	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaResource.owl#BrasiliaResource
676	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaResource.owl#BrasiliaResource	Identifier2.hasIdentifier	http://gia.inf.ufbrs.br/Repository/20160913174652139/BrasiliaResourceIdentifier.owl#BrasiliaResourceIdentifier

Fonte: Elaborado pelo autor

7.3 Armazenamento de Metadados

Um repositório semântico proporciona um armazenamento de dados na forma de grafos. Isto permite que um conceito possa ter n relacionamentos. Os repositórios semânticos usam ontologias como um esquema semântico. Isto possibilita que inferências automáticas possam ser feitas porque os relacionamentos entre conceitos são construídos dentro da ontologia.

GraphDB⁹ é considerada a solução de base de triplas RDF mais utilizada no mundo. Ele utiliza OWL e suas ontologias permitem ao repositório automaticamente raciocinar sobre os dados (POKORNÝ, 2015). A presente dissertação explorou o GraphDB como uma alternativa para repositório de ontologias OWL. Todas as ontologias foram importadas para o repositório semântico. Este armazenamento pode ser recuperado através de SPARQL ou REST API.

A Figura 7.8 mostra a indexação das triplas dos metadados do OA Brasília. Todas as ontologias podem ser recuperadas por uma consulta SPARQL. Além disso, é possível pesquisar estas ontologias e encontrar OAs para estudantes com necessidades especiais ou classificar OAs de acordo com determinados perfis de aplicação.

⁹<<http://ontotext.com/products/graphdb/>>

7.4 Considerações do Capítulo

Esta proposta tem por objetivo cooperar com a interoperabilidade de metadados educacionais na Web Semântica. O método pode coletar dados de diferentes fontes, processar esta informação para fornecer os resultados a outras aplicações. Os dados são processáveis por máquinas e podem ser acessados por diversos agentes inteligentes de software.

As descrições OWL resultantes também tem por finalidade diminuir problemas com a interpretação de representações de dados. As ontologias também permitem que os dados sejam automaticamente verificados quanto a sua consistência. Mesmo perdendo a ordem dos metadados em XML, a representação em OWL é mais rica em expressividade, pois sua representação em triplas permite um melhor relacionamento entre os conceitos e é possível explorar os seus axiomas relacionados, mesmo em diferentes ontologias. Além disto, equivalências podem ser realizadas para alinhar conceitos nas ontologias de domínio, estabelecendo uma descrição de mundo mais rica.

O método proposto baseado em XSLT provê uma alternativa simplificada e independente de linguagem compilada, mas que é extremamente capaz no sentido de manipular tanto descrições em XML quanto em OWL através de XPath. Contudo, o método ainda lida com dados inconsistentes, incompletos, duplicados e com diferentes padrões. A abordagem baseada com ontologias tenta lidar com estas questões, através de motores de inferências e alinhamentos.

O armazenamento de triplas no repositório GraphDB foi escolhido pela sua facilidade de implantação. Contudo, existem diferentes alternativas que podem ser avaliadas, tais como OWLIM, Virtuoso, Semame, Stardog, etc.

8 RESULTADOS

Neste capítulo será apresentado como o desempenho do motor de inferências se comporta nas camadas da representação de conhecimento proposta. A análise de resultados foi com o uso do framework Protégé 5.1. O motor de inferências empregado foi HerMiT 1.3.8.413 com configuração padrão. O computador utilizado para a coleta foi Intel Core i7-2630QM CPU @ 2.00 GHz com 6 GB de memória RAM.

As seções a seguir analisarão de forma qualitativa as ontologias a partir de cinco perspectivas das instâncias apresentadas na Seção 6.2:

- P1: Ontologia única que representa todo o padrão¹;
- P2: Ontologia que importa todos os módulos do padrão o qual representa;
- P3: A mesma abordagem que P2, porém sem restrições de cardinalidades máximas maiores que um;
- P4: Importação dos módulos de ontologias propriamente utilizados, com restrição de cardinalidades maiores que um;
- P5: Importação dos módulos de ontologias propriamente utilizados, sem restrição de cardinalidades maiores que um.

8.1 Verificação de Consistência

A verificação de consistência pode ser realizada nas duas primeiras camadas. Para a primeira camada (TBox), foi avaliada cada representação completa dos padrões de metadados IEEE-LOM, OBAA e VideoAula. Para a segunda camada (ABox), os metadados de OAs descritos no Capítulo 6 foram analisados. A Tabela 8.1 apresenta o tempo em milissegundos sobre cinco perspectivas².

As perspectivas quatro e cinco foram desconsideradas para avaliação da primeira camada, pois neste momento não se sabe quais módulos serão devidamente utilizados na próxima camada, pois isto varia de acordo com cada instância de metadados. Deste modo, P4 e P5 foram sinalizadas com “-”.

Naturalmente, o tempo de raciocínio é diretamente relacionado com a quantidade de axiomas da ontologia. Deste modo, quanto mais axiomas são adicionados, maior será

¹Nesta perspectiva foram reutilizadas ontologias desenvolvidas em trabalho anterior (BEHR, 2014).

²O símbolo “-” identifica quando uma ontologia não se aplica à perspectiva. De outro modo, o símbolo “**” indica quando o tempo de verificação ultrapassa 8,64e+7 ms (24 horas).

Tabela 8.1: Avaliação de desempenho em ms de ontologias do nível um e dois

Ontologia	P1	P2	P3	P4	P5
IEEE-LOM	2576	2431	441	-	-
OBAA	2718	2728	732	-	-
VideoAula@RNP	3247	2506	572	-	-
Brasília	*	*	426	*	409
Klimadiagramm von Athen	*	*	446	*	413
Geometria Descritiva em Contos	3543	3636	575	2618	438
Infraestr. do Serviço Videoaula@RNP: Mód. 2	24253	23193	630	22195	516

Fonte: Elaborado pelo autor

o tempo demandado pelo motor de inferências. No geral, ontologias modulares (P2) apresentaram uma ligeira melhor performance quanto a ontologias que descrevem os metadados em únicas ontologias (P1). Isto torna-se um pouco mais perceptível quando somente os módulos utilizados pelas instâncias são importados (P4).

Além disto, a restrição de cardinalidades de maneira não simplificada acarreta em uma demora acentuada na verificação de consistência. Motores de inferências quando comparados com verificações máximas de cardinalidades podem levar ao não-determinismo. Isto ocorre nas perspectivas P1, P2 e P4 para os metadados dos OAs Brasília e Klimadiagramm von Athen. Estas foram sinalizadas com “*”, pois foram consideradas intratáveis pelo motor de inferências por demorarem mais de 24 horas para a verificação.

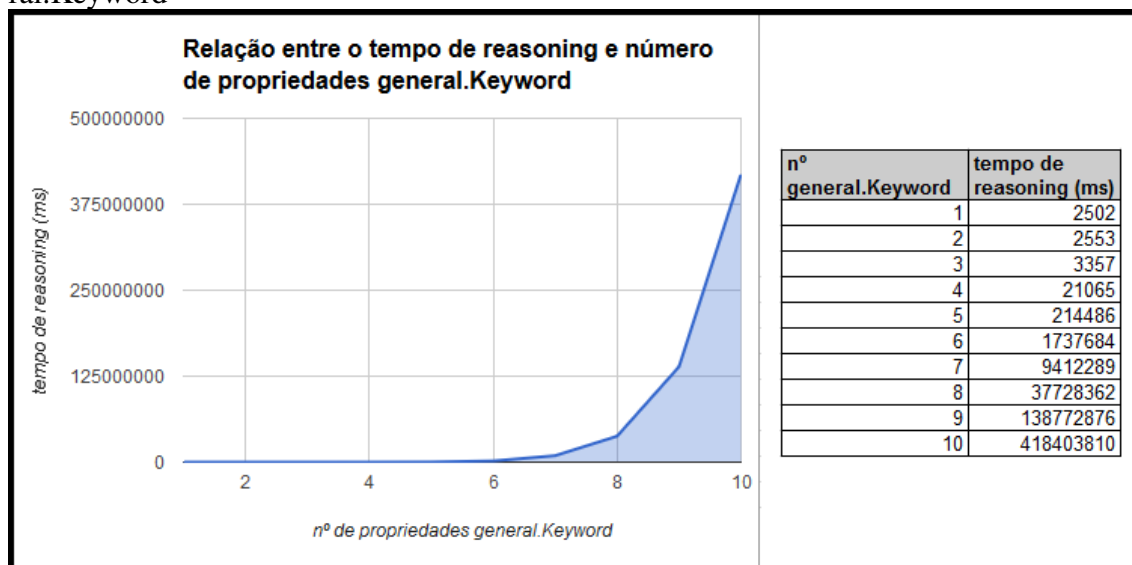
A Figura 8.1 apresenta um comparativo do aumento de asserções da propriedade *general.Keyword* (que possui cardinalidade dez) nos metadados do OA Klimadiagramm von Athen. O tempo total de verificação se comporta de maneira exponencial, de modo que a partir da nona asserção, a verificação termina após aproximadamente 38 horas.

8.2 Classificação

A classificação de OAs foi realizada com os perfis de aplicação OBAA-LITE e OBAA-VideoAula. Dos OAs descritos no Capítulo 6, Klimadiagramm von Athen e Infraestrutura do Serviço Videoaula@RNP: Módulo 2 são classificados nos perfis OBAA-LITE e OBAA-VideoAula, respectivamente. Como comparativo, duas abordagens de RBox para perfis de aplicação foram utilizadas:

- RBox1: Definido como classe de equivalência;
- RBox2: Definido como consulta SQWRL.

Figura 8.1: Relação entre o tempo de reasoning e número de propriedades general.Keyword



Fonte: Elaborado pelo autor

Tabela 8.2: Avaliação de desempenho em ms de ontologias no nível três

Ontologia	RBox1	RBox2
Klimadiagramm von Athen	409	264
Infraestrutura do Serviço Videoaula@RNP: Módulo 2	737	537

Fonte: Elaborado pelo autor

A classificação via RBox1 deu-se sem a verificação de cardinalidades máximas maiores que um, sob a perspectiva P5 da seção anterior. Contudo, como RBox2 opera sobre o mundo fechado, o seu desempenho foi superior ao do RBox1.

8.3 Interoperabilidade

Como avaliação do método proposto no Capítulo 7, um subconjunto de 1100 metadados de OAs no padrão OBAA foram recuperados através da Federação de Repositórios Educa Brasil (FEB)³. Atualmente, o FEB reúne 16 repositórios na federação.

Os metadados foram transpostos para a descrição ontológica proposta a partir da perspectiva P5. As descrições ontológicas resultantes podem ser acessadas em <<http://gia.inf.ufrgs.br/Repository/FEB/>>.

³<<http://feb.ufrgs.br/>>

8.4 Considerações do Capítulo

A comparação com ontologias de trabalhos relacionados foi comprometida pelo fato de ora artigos não fornecerem uma URI de acesso as mesmas, ora por não especificarem como a ontologia deve ser utilizada para a representação dos metadados. Deste modo, um comparativo dentre cinco perspectivas distintas deste trabalho foi aplicado.

A abordagem com múltiplas ontologias permite que agentes inteligentes de software possam então importar URIs e raciocinar a partir delas, com o auxílio da OWLAPI⁴ e SWRLAPI⁵. Um meio para melhorar a verificação de consistência de descrições extensas ou de várias descrições seria dividir a representação para que cada ontologia seja verificada separadamente de forma paralela e/ou distribuída. Por exemplo, uma ontologia que importa muitas outras, tais como General, Educational, Technical e Rights pode ser recuperada a partir de uma base Triple Store⁶ ou até mesmo da URI principal e executar raciocínio de forma individual. Com cada verificação, um resultado de raciocínio pode ser dado.

⁴<https://github.com/owlcs/owlapi>

⁵<https://github.com/protegeproject/swrlapi>

⁶Base de dados web semânticos

9 CONCLUSÃO

A disponibilização de dados na Web vem sendo intensificada cada vez mais. Contudo, os dados encontram-se de forma que na maioria das vezes só são interpretados por seres humanos. Deste modo, a Web Semântica torna-se uma importante alternativa para estender a Web atual de forma que ela seja entendível tanto por humanos quanto máquinas.

Consequentemente, agentes artificiais conseguem operar sobre informações mais ricas de dados. Desta forma, outrora repositórios isolados com milhares de objetos, ambientes sem padrão de metadados ou com metadados pobres, dentre outros, conseguem operar de forma inteligente.

A definição de descrições compartilhadas, denominadas por ontologias, é um dos principais conceitos que sustentam os objetivos da Web Semântica e realizam a integração dos dados. Ontologias são capazes de descrever domínios e conhecimentos prévios. A publicação de ontologias por meio de triplas possibilita que inferências sejam realizadas e que declarações menos extensas de informações sejam feitas. Além disso, é possível realizar operações como classificação, relacionamento, busca e filtragem.

Apesar de RDF ser a especificação W3C para descrever metadados, metadados educacionais descritos como ontologias OWL podem fomentar o desenvolvimento de diversas aplicações para domínios educacionais, tais como autoria de OAs, desenvolvimento de agentes para cursos personalizados a estudantes, sugestão de materiais educacionais, dentre outros. O presente trabalho apresenta uma alternativa para representar metadados de OAs como ontologias para permitir um meio de raciocínio para aplicações educacionais.

As ontologias fazem uso de uma proposta de três camadas para descrever um conhecimento de domínio de forma modularizada. Cada uma delas foi descrita e foram apresentados exemplos, reutilizando-se de camadas anteriores. Embora as ontologias possam ser consideradas complexas, em um processo automático vários benefícios interessantes podem ser obtidos, tais como: fácil atualização, por exemplo um esquema de metadados é atualizado nas ontologias de descrição, elas são automaticamente incorporadas na camada de instanciação; a instância de um novo grupo de metadados é facilmente incorporado na ontologia principal a partir da importação da nova ontologia e da conexão do indivíduo através de uma propriedade de objeto; o reuso de alguns indivíduos por outro OA, por exemplo um indivíduo que represente o grupo Technical que é comum a muitos outros

OAs; a descrição de indivíduos pode ser validada de acordo com a ontologia de nível um; o armazenamento pode ser feito por uma URI; é possível construir um relacionamento entre ontologias de OAs por propriedades e eles são compatíveis com a atual pilha da Web Semântica.

Além do reuso entre as camadas, ele também pode ocorrer com diferentes partes da representação de conhecimento através de instâncias comuns entre ontologias. Além disso, o reuso de instância pode ser potencializado quando novas versões de um mesmo OA surgirem, visto que a quantidade de conteúdos repetidos tende a ser grande.

Entretanto, a modularização em si não garante necessariamente que se tenha um ganho expressivo na verificação dos metadados de OAs. Após o estudo de comportamento das ontologias de forma qualitativa sob diferentes perspectivas, constatou-se que a modularização apresentou apenas um pequeno ganho no desempenho. Contudo, houve um ganho maior quando as cardinalidades máximas (maiores que um) verificadas foram suprimidas da segunda camada e transpostas para a última camada. Quanto a terceira camada, notou-se uma pequena melhoria quando aplicada em mundo fechado.

Na Web Semântica, a interoperabilidade entre descrições de metadados educacionais foi realizada através de um processador XSLT que possui uma semântica dada por uma ontologia, que pode ser alinhada com outra para transpor diferentes descrições ontológicas. Como validação da proposta, 1100 metadados em XML foram transpostos para a abordagem de múltiplas ontologias.

Algumas limitações foram encontradas durante o desenvolvimento da dissertação. Ontologias perdem a ordenação dos metadados por serem baseadas em triplas, como por exemplo uma ordem de autores, devendo ter uma adição de anotação ou chave associada. O armazenamento de ontologias na base Triple-Store do GraphDB deve seguir serialização RDF/XML, caso da ontologia original não a siga, deve-se haver uma mudança de serialização. O gerenciamento de entidades com mesmo nome pelo Protégé por vezes foi realizado de maneira errônea, tendo que ser feita a edição diretamente no código-fonte da ontologia. Além disso, não existe ainda compatibilidade total entre a utilização do motor de inferências com regras SWRL. Deste modo, primeiro se verifica a consistência das duas primeiras camadas para depois se utilizar de SQWRL para a camada de raciocínio no Protégé. O alinhamento entre ontologias por ora foi realizado de maneira manual, a ferramenta Alignment API¹ foi testada e não se encontrou um modo dela alinhar também as ontologias importadas.

¹<<http://alignapi.gforge.inria.fr/>>

Ainda existem trabalhos relacionados que necessitam serem explorados com os objetivos de otimizar interoperabilidade e o uso por motores de inferências. Dentre eles pode-se citar:

- Analisar quantitativamente metadados quanto ao reuso de grupos agregados com conteúdo repetidos;
- Analisar quantitativamente o desempenho de verificação de consistências e classificações de modo a ser possível uma análise estatística;
- Vincular ontologias com trabalhos de dados abertos conectados;
- Explorar performance do processamento paralelo das ontologias importadas, com tecnologias como Hadoop;
- Experimentar com usuários o reuso das descrições ontológicas;
- Pesquisar a aderência das ontologias propostas em ambientes educacionais, tais como DSpace e Moodle;
- Explorar alinhamentos automáticos entre ontologias de metadados educacionais;
- Automatizar modularização de ontologias;
- Concretizar a proposta de interoperabilidade de metadados como um serviço web.

REFERÊNCIAS

- ABBES, S. B. et al. Characterizing modular ontologies. In: **7th International Conference on Formal Ontologies in Information Systems-FOIS 2012**. [S.l.: s.n.], 2012. p. 13–25.
- AL-KHALIFA, H. S.; DAVIS, H. C. The evolution of metadata from standards to semantics in e-learning applications. In: ACM. **Proceedings of the seventeenth conference on Hypertext and hypermedia**. [S.l.], 2006. p. 69–72.
- ANTONIOU, G.; HARMELEN, F. V. Web ontology language: Owl. In: **Handbook on ontologies**. [S.l.]: Springer, 2004. p. 67–92.
- ARAPOV, P.; BUFFA, M.; OTHMANE, A. B. A wiki way of programming for the web of data. In: INTERNATIONAL WORLD WIDE WEB CONFERENCES STEERING COMMITTEE. [S.l.], 2014. p. 231–232.
- AUER, S.; HERRE, H. Rapidowl—an agile knowledge engineering methodology. In: SPRINGER. **International Andrei Ershov Memorial Conference on Perspectives of System Informatics**. [S.l.], 2006. p. 424–430.
- BALIYAN, N.; KUMAR, S. A behavioral metrics suite for modular ontologies. In: ACM. **Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies**. [S.l.], 2016. p. 133.
- BARROS, H. et al. Steps, techniques, and technologies for the development of intelligent applications based on semantic web services: A case study in e-learning systems. **Engineering Applications of Artificial Intelligence**, Elsevier, v. 24, n. 8, p. 1355–1367, 2011.
- BECHHOFER, S. Owl: Web ontology language. In: **Encyclopedia of Database Systems**. [S.l.]: Springer, 2009. p. 2008–2009.
- BEDINI, I. et al. Transforming xml schema to owl using patterns. In: IEEE. **Semantic Computing (ICSC), 2011 Fifth IEEE International Conference on**. [S.l.], 2011. p. 102–109.
- BEHR, A.; PRIMO, T.; VICCARI, R. An ontology for the obaa metadata standard. In: **Anais do Simpósio Brasileiro de Informática na Educação**. [S.l.: s.n.], 2012. v. 23, n. 1.
- BEHR, A.; PRIMO, T.; VICCARI, R. Towards educational metadata interoperability on semantic web. **Simpósio Brasileiro de Informática na Educação**, v. 27, n. 1, p. 1026, 2016.
- BEHR, A.; PRIMO, T. T.; VICARI, R. Obaa-leme: A learning object metadata content editor supported by application profiles and educational metadata ontologies. In: **Anais dos Workshops do Congresso Brasileiro de Informática na Educação**. [S.l.: s.n.], 2014. v. 3, n. 1, p. 455.
- BEHR, A. R. Obaa-leme: um editor de conteúdo de metadados para objetos de aprendizagem a partir de ontologias e perfis de aplicação. 2014.

- BERNERS-LEE, T. Semantic web layer cake. **XML-2000 Conference**, v. 29, n. 05, p. 2005, 2000.
- BERNERS-LEE, T. et al. The semantic web. **Scientific american**, New York, NY, USA:, v. 284, n. 5, p. 28–37, 2001.
- BORGIDA, A.; SERAFINI, L. Distributed description logics: Assimilating information from peer sources. In: **Journal on data semantics I**. [S.l.]: Springer, 2003. p. 153–184.
- BREITMAN, K. K. **Web semântica: a internet do Futuro. Rio de Janeiro: LTC, 2005**. [S.l.], 2005.
- CARLI, V. de et al. Estudo e proposta para utilização dos metadados obaa na descrição de jogos sérios. In: **Anais do Simpósio Brasileiro de Informática na Educação**. [S.l.: s.n.], 2014. v. 25, n. 1, p. 466.
- CARVALHO, A. L. da C. et al. A quantitative analysis of learning objects and their metadata in web repositories. In: SPRINGER. **International Workshop on Social Computing in Digital Education**. [S.l.], 2015. p. 49–64.
- CASALI, A. et al. An assistant for loading learning object metadata: An ontology based approach. Informing Science Institute, 2013.
- CASELLAS, N. Methodologies, tools and languages for ontology design. In: **Legal Ontology Engineering**. [S.l.]: Springer, 2011. p. 57–107.
- CASTRO-GARCÍA, L.; LÓPEZ-MORTEO, G. An international analysis of the extensions to the ieeelomv1.0 metadata standard. **Computer Standards & Interfaces**, Elsevier, v. 35, n. 6, p. 567–581, 2013.
- CHAHDI, H. et al. Towards ontology reasoning for topological cluster labeling. 2016.
- CHOE, H. Interoperability between mpeg-7 and lom using ontology. **ASIAN JOURNAL OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY**, v. 2, n. 11, 2013.
- DAVIES, J. et al. The diligent knowledge processes. **Journal of Knowledge Management**, Emerald Group Publishing Limited, v. 9, n. 5, p. 85–96, 2005.
- DCMI. Dublin core metadata element set, version 1.1. Dublin Core Metadata Initiative, 2008.
- DING, L.; FININ, T. Characterizing the semantic web on the web. In: **The Semantic Web-ISWC 2006**. [S.l.]: Springer, 2006. p. 242–257.
- DUVAL, E. et al. Metadata principles and practicalities. **D-lib Magazine**, v. 8, n. 4, p. 16, 2002.
- EHRIG, M. **Ontology alignment: bridging the semantic gap**. [S.l.]: Springer Science & Business Media, 2006.
- FERNÁNDEZ-LÓPEZ, M.; GÓMEZ-PÉREZ, A.; JURISTO, N. Methontology: from ontological art towards ontological engineering. American Association for Artificial Intelligence, 1997.

FRIESEN, N.; MASON, J.; WARD, N. Building educational metadata application profiles. In: **Proceedings of the International Conference on Dublin Core and Metadata for e-Communities**. [S.l.: s.n.], 2002. p. 63–69.

GARCÍA, A. M. F. et al. Case study: Ontology for metadata in e-learning. In: **KEOD**. [S.l.: s.n.], 2012. p. 317–320.

GARCÍA, N. F.; FERNÁNDEZ, L. S. La web semántica: fundamentos y breve "estado del arte". **Novática: Revista de la Asociación de Técnicos de Informática**, Asociación de Técnicos de Informática, ATI, n. 178, p. 6–11, 2005.

GASCUEÑA, J. M.; FERNANDEZ-CABALLERO, A.; GONZALEZ, P. Domain ontology for personalized e-learning in educational systems. In: **ICALT**. [S.l.: s.n.], 2006. p. 456–458.

GLUZ, J. C.; VICARI, R. M. An owl ontology for iee-lom and obaa metadata. In: **SPRINGER. Intelligent Tutoring Systems**. [S.l.], 2012. p. 691–693.

GLUZ, J. C.; VICARI, R. M. Rumo a uma plataforma semântica de conteúdos educacionais digitais: o modelo ontológico. In: **Anais do Simpósio Brasileiro de Informática na Educação**. [S.l.: s.n.], 2014. v. 25, n. 1, p. 993.

GOMEZ-PEREZ, A.; FERNÁNDEZ-LÓPEZ, M.; CORCHO, O. **Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web**. [S.l.]: Springer Science & Business Media, 2006.

GRAU, B. C. et al. Owl 2: The next step for owl. **Web Semantics: science, services and agents on the World Wide Web**, Elsevier, v. 6, n. 4, p. 309–322, 2008.

GREENBERG, J.; SWAUGER, S.; FEINSTEIN, E. M. Metadata capital in a data repository. In: CITESEER. **Proceedings of the International Conference on Dublin Core and Metadata Applications DC2013**. [S.l.], 2013. p. 140–150.

GRÜNINGER, M.; FOX, M. S. Methodology for the design and evaluation of ontologies. Citeseer, 1995.

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

HACHEROUF, M.; BAHLOUL, S. N.; CRUZ, C. Transforming xml documents to owl ontologies: A survey. **Journal of Information Science**, SAGE Publications, v. 41, n. 2, p. 242–259, 2015.

HEATH, T.; BIZER, C. Linked data: Evolving the web into a global data space. **Synthesis lectures on the semantic web: theory and technology**, Morgan & Claypool Publishers, v. 1, n. 1, p. 1–136, 2011.

HEERY, R.; PATEL, M. Application profiles: mixing and matching metadata schemas. **Ariadne**, n. 25, 2000.

HENDLER, J. Agents and the semantic web. **IEEE Intelligent systems**, v. 16, n. 2, p. 30–37, 2001.

HORROCKS, I. et al. Swrl: A semantic web rule language combining owl and ruleml. **W3C Member submission**, v. 21, p. 79, 2004.

IMS. Ims accessforall meta-data information model. **IMS Global Learning Consortium**, n. Version 1.0 Final Specification, July 2004.

ISOTANI, S.; BITTENCOURT, I. I. **Dados Abertos Conectados**. [S.l.]: Novatec Editora, 2015.

JARRAR, M. Towards methodological principles for ontology engineering. **Mustafa Jarrar: Towards methodological principles for ontology engineering. PhD Thesis. Vrije Universiteit Brussel.(May 2005)**, 2005.

KBSI. **The IDEF5 Ontology Description Capture Method Overview**. [S.l.]: KBSI Report TX, 1994.

KEET, C. M. Closed world assumption. **Encyclopedia of Systems Biology**, Springer, p. 415–415, 2013.

KEET, C. M. Open world assumption. In: **Encyclopedia of Systems Biology**. [S.l.]: Springer, 2013. p. 1567–1567.

KOTIS, K.; VOUIROS, G. A. Human-centered ontology engineering: The hcome methodology. **Knowledge and Information Systems**, Springer, v. 10, n. 1, p. 109–131, 2006.

KUTZ, O. et al. E-connections of abstract description systems. **Artificial intelligence**, Elsevier, v. 156, n. 1, p. 1–73, 2004.

LACOSTE, D.; SAWANT, K. P.; ROY, S. An efficient xml to owl converter. In: **ACM. Proceedings of the 4th India software engineering conference**. [S.l.], 2011. p. 145–154.

LALINGKAR, A.; RAMNATHAN, C.; RAMANI, S. Ontology-based smart learning environment for teaching word problems in mathematics. **Journal of Computers in Education**, Springer, v. 1, n. 4, p. 313–334, 2014.

LIANG, S. et al. Semi automated transformation to owl formatted files as an approach to data integration. **Methods of information in medicine**, Schattauer Publishers, v. 54, n. 1, p. 32–40, 2015.

LTSC. **IEEE Standard for Learning Object Metadata (Draft). IEEE Standard 1484.12.1**. New York: [s.n.], 2002. Available from Internet: <http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf>.

MACÁRIO, C. G. N.; SOUSA, S. R. de; MEDEIROS, C. M. B. Play it again, sam—using scientific workflows to drive the generation of semantic annotations. In: **IEEE. e-Science (e-Science), 2010 IEEE Sixth International Conference on**. [S.l.], 2010. p. 284–291.

MACHADO, G. M.; OLIVEIRA, J. P. M. de. Context-aware adaptive recommendation of resources for mobile users in a university campus. In: **IEEE. 2014 IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)**. [S.l.], 2014. p. 427–433.

- MENOLLI, A. L. A. et al. Unit of organizational learning ontology based on lom standard and ims learning design. In: **FOIS**. [S.l.: s.n.], 2014. p. 396–405.
- MILUTINOVIĆ, M. et al. Ontology-based generated learning objects for mobile language learning. **Computer Science and Information Systems**, n. 00, p. 4–4, 2016.
- MIZOGUCHI, R. On scalability of the semantic web. **Journal of Zhejiang University-Science C**, v. 13, n. 4, p. 245–246, 2012.
- MIZOGUCHI, R.; VANWELKENHUYSEN, J.; IKEDA, M. Task ontology for reuse of problem solving knowledge. **Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing**, IOS press Amsterdam, p. 46–59, 1995.
- NOY, N.; MCGUINNESS, D. L. et al. Ontology development 101. **Knowledge Systems Laboratory, Stanford University**, 2001.
- O'CONNOR, M.; DAS, A. Sqwrl: a query language for owl. In: CEUR-WS. ORG. **Proceedings of the 6th International Conference on OWL: Experiences and Directions-Volume 529**. [S.l.], 2009. p. 208–215.
- O'CONNOR, M. J.; DAS, A. Acquiring owl ontologies from xml documents. In: ACM. **Proceedings of the sixth international conference on Knowledge capture**. [S.l.], 2011. p. 17–24.
- PALAVITSINIS, N.; MANOUSELIS, N.; SANCHEZ-ALONSO, S. Metadata quality in learning object repositories: a case study. **The Electronic Library**, Emerald Group Publishing Limited, v. 32, n. 1, p. 62–82, 2014.
- PATIL, T.; DAVENPORT, D. Data scientist: The sexiest job of the 21st century. **Harvard Business Review**, 2012.
- PAULHEIM, H. Ontology-based system integration. In: **Ontology-based Application Integration**. [S.l.]: Springer, 2011. p. 27–59.
- POKORNÝ, J. Graph databases: Their power and limitations. In: **Computer Information Systems and Industrial Management**. [S.l.]: Springer, 2015. p. 58–69.
- PRIMO, T. T. **Método de representação de conhecimento baseado em ontologias para apoiar sistemas de recomendação educacionais**. Thesis (PhD) — Universidade Federal do Rio Grande do Sul (UFRGS), 2013.
- PRIMO, T. T.; BEHR, A.; VICARI, R. M. A semantic web approach to recommend learning objects. In: **Highlights on Practical Applications of Agents and Multi-Agent Systems**. [S.l.]: Springer, 2013. p. 340–350.
- RADATZ, J.; GERACI, A.; KATKI, F. Ieee standard glossary of software engineering terminology. **IEEE Std**, v. 610121990, n. 121990, p. 3, 1990.
- RAJU, P.; AHMED, V. Enabling technologies for developing next-generation learning object repository for construction. **Automation in Construction**, Elsevier, v. 22, p. 247–257, 2012.
- RIBEIRO, R.; PRIMO, T. T. Proposta para um algoritmo de conversão de metadados em ontologias. In: **WebMedia 2013 WSEDU**. [S.l.: s.n.], 2013.

RODRÍGUEZ, P. et al. Using ontological modeling for multi-agent recommendation of learning objects. In: **Workshop MASLE-Multiagent System Based Learning Environments, Intelligent Tutoring Systems (ITS) Conference, Hawaii, USA**. [S.l.: s.n.], 2014.

SCHWARZ, G. et al. Perfil de metadados obaa utilizado no ensino de profissionais de saúde. **RENOTE**, v. 10, n. 3, 2012.

SEEMILLER, C.; GRACE, M. **Generation Z Goes to College**. [S.l.]: John Wiley & Sons, 2016.

SHADBOLT, N.; BERNERS-LEE, T.; HALL, W. The semantic web revisited. **IEEE Intelligent Systems**, IEEE Educational Activities Department, Piscataway, NJ, USA, v. 21, n. 3, p. 96–101, may 2006. ISSN 1541-1672. Available from Internet: <<http://dx.doi.org/10.1109/MIS.2006.62>>.

SILVA, J. M. C. d. Análise técnica e pedagógica de metadados para objetos de aprendizagem. 2011.

SILVA, J. M. R. da. Usage-driven application profile generation using ontologies. 2016.

SOLOMOU, G.; PIERRAKEAS, C.; KAMEAS, A. Characterization of educational resources in e-learning systems using an educational metadata profile. **Journal of Educational Technology & Society**, JSTOR, v. 18, n. 4, p. 246–260, 2015.

SULLIVAN, K. J. et al. The structure and value of modularity in software design. In: **ACM. ACM SIGSOFT Software Engineering Notes**. [S.l.], 2001. v. 26, n. 5, p. 99–108.

TANI, A.; CANDELA, L.; CASTELLI, D. Dealing with metadata quality: The legacy of digital library efforts. **Information Processing & Management**, Elsevier, v. 49, n. 6, p. 1194–1205, 2013.

TAROUCO, L. M. R. et al. **Objetos de aprendizagem: teoria e prática**. Evangraf, 2014.

USCHOLD, M. Building ontologies: Towards a unified methodology. **TECHNICAL REPORT-UNIVERSITY OF EDINBURGH ARTIFICIAL INTELLIGENCE APPLICATIONS INSTITUTE AIAI TR**, Citeseer, 1996.

VICARI, R. M. et al. Brazilian proposal for agent-based learning objects metadata standard-obao. In: **Metadata and Semantic Research**. [S.l.]: Springer, 2010. p. 300–311.

XIONG, J.; LIU, Y.; LIU, W. Ontology-based integration and sharing of big data educational resources. In: **IEEE. Web Information System and Application Conference (WISA), 2014 11th**. [S.l.], 2014. p. 245–248.

YAHIA, N.; MOKHTAR, S. A.; AHMED, A. Automatic generation of owl ontology from xml data source. **arXiv preprint arXiv:1206.0570**, 2012.

**APÊNDICE A — MAPEAMENTO IEEE-LOM PARA CONJUNTO NÃO
QUALIFICADO DE METADADOS DUBLIN CORE**

Tabela A.1: Mapeamento de metadados entre Dublin Core e IEEE-LOM

DC.Identifier	1.1.2:General.Identifier.Entry
DC.Title	1.2:General.Title
DC.Language	1.3:General.Language
DC.Description	1.4:General.Description
DC.Subject	1.5:General.Keyword or 9:Classification with 9.1:Classification.Purpose equals "Discipline" or "Idea".
DC.Coverage	1.6:General.Coverage
DC.Type	5.2:Educational.LearningResourceType
DC.Date	2.3.3:LifeCycle.Contribute.Date when 2.3.1:LifeCycle.Contribute.Role has a value of "Publisher".
DC.Creator	2.3.2:LifeCycle.Contribute.Entity when 2.3.1:LifeCycle.Contribute.Role has a value of "Author".
DC.OtherContributor	2.3.2:LifeCycle.Contribute.Entity with the type of contribution specified in 2.3.1:LifeCycle.Contribute.Role.
DC.Publisher	2.3.2:LifeCycle.Contribute.Entity when 2.3.1:LifeCycle.Contribute.Role has a value of "Publisher".
DC.Format	4.1:Technical.Format
DC.Rights	6.3:Rights.Description
DC.Relation	7.2.2:Relation.Resource.Description
DC.Source	7.2:Relation.Resource when the value of 7.1:Relation.Kind is "IsBasedOn"

Fonte: Adaptado de (LTSC, 2002)

APÊNDICE B — PERFIS DE APLICAÇÃO

Tabela B.1: Metadados do perfil de aplicação OBAA-LITE

Nr	Nome	Cardinalidade	Espaço de Valores	Tipo de Dado
1	General	1		
1.2	Title	1		LangString (max. 1000 char)
1.3	Language	10	ISO/IEC 10646-1:2000	CharacterString (max. 100 char)
1.4	Description	10		LangString (max. 2000 char)
1.5	Keyword	10		LangString (max. 1000 char)
4	Technical	1		
4.3	Location	10	ISO/IEC 10646-1:2000	CharacterString (max. 1000 char)
5	Educational	100		
5.2	Learning Resource Type	10	exercise, simulation, questionnaire, diagram...	Vocabulary
5.5	Intended End User Role	10	teacher, author, leaner, manager	Vocabulary
5.6	Context	10	school, high education, training, other	Vocabulary
5.9	Typical Learnig Time	1		Duration
5.10	Description	10		LangString (max. 1000 char)
6	Rights	1		
6.2	Copyright and Other Restrictions	1	yes, no	Vocabulary
6.3	Description	10		LangString (max. 1000 char)

Fonte: Adaptado de (SILVA, 2011)

Tabela B.2: Metadados do perfil de aplicação OBAA-VideoAula – Parte 1

Nr	Nome	Cardinalidade	Espaço de Valores	Tipo de Dado
1	General	1		
1.1	Identifier	1		
1.1.1	Catalog	1	ISO/IEC 10646-1:2000	CharacterString (max. 1000 char)
1.1.2	Entry	1	ISO/IEC 10646-1:2000	CharacterString (max. 1000 char)
1.2	Title	1		LangString (max. 1000 char)
1.3	Language	10	ISO/IEC 10646-1:2000	CharacterString (max. 100 char)
1.4	Description	10		LangString (max. 2000 char)
1.5	Keyword	10		LangString (max. 1000 char)
2	LifeCycle	1		
2.2	Status	1	draft, final, revised, unavailable	Vocabulary
2.3	Contribute	30		
2.3.1	Role	1	author, publisher unknown, initiator...	Vocabulary
2.3.2	Entity	40	RFC 2425 RFC 2426	CharacterString (max. 1000 char)
2.3.3	Date	1		DateTime
3	Meta-Metadata	1		
3.1	Identifier	10		
3.1.1	Catalog	1	ISO/IEC 10646-1:2000	CharacterString (max. 1000 char)
3.1.2	Entry	1	ISO/IEC 10646-1:2000	CharacterString (max. 1000 char)
3.2	Contribute	10		
3.2.1	Role	1	creator, validator	Vocabulary
3.2.2	Entity	10	RFC 2425 RFC 2426	CharacterString (max. 1000 char)
3.2.3	Date	1		DateTime
3.3	Metadata Schema	10	ISO/IEC 10646-1:2000	CharacterString (max. 30 char)

Fonte: Adaptado de (BEHR, 2014)

Tabela B.3: Metadados do perfil de aplicação OBAA-VideoAula – Parte 2

Nr	Nome	Cardinalidade	Espaço de Valores	Tipo de Dado
4	Technical	1		
4.1	Format	40	RFC 2048:1996	CharacterString (max. 500 char)
4.2	Size	1	ISO/IEC 646:1991	CharacterString (max. 30 char)
4.4	Requirement	40		
4.4.1	OrComposite	40		
4.4.1.1	Type	1	operating system, browser	Vocabulary
4.4.1.1	Name	1	...if Type="browser" then : any, netscape...	Vocabulary
4.7	Duration	1		Duration
4.8	Supported Platforms	N		String
4.10	Service	N		
4.10.1	Name	1		String
4.10.2	Type	1		String
4.10.3	Provides	1		Boolean
4.10.4	Essential	1		Boolean
4.10.5	Protocol	N		String
4.10.8	Details	N		String
4.10.8.1	Service Location	N		URL, URI
4.13	Related Media	1		
4.13.1	Catalog	1		String
4.13.2	Entry	1		String
4.13.3	Resolution			String
4.13.4	Aspect Ratio	1		String
4.13.5	Bitrate	1		String
4.13.6	Standard			Boolean
6	Rights	1		
6.2	Copyright and Other Restrictions	1	yes, no	Vocabulary
6.3	Description	10		LangString (max. 1000 char)

Fonte: Adaptado de (BEHR, 2014)

APÊNDICE C — LISTA DE ONTOLOGIAS DESENVOLVIDAS

- IEEE-LOM - <http://gia.inf.ufrgs.br/ontologies/LOM/LOM.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/General.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/General/Identifier.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/LifeCycle.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/LifeCycle/Contribute.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Meta-Metadata.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Meta-Metadata/Contribute.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Meta-Metadata/Identifier.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Technical.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Technical/Requirement.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Technical/Requirement/OrContribute.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Educational.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Rights.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Relation.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Relation/Resource.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Relation/Resource/Identifier.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Annotation.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Classification.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Classification/TaxonPath.owl>
 - <http://gia.inf.ufrgs.br/ontologies/LOM/Classification/TaxonPath/Taxon.owl>
- IMS AccessForAll - <http://gia.inf.ufrgs.br/ontologies/IMS/IMS.owl>
 - <http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility.owl>
 - <http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription.owl>
 - <http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalentent.owl>
 - <http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Content.owl>
 - <http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Content/AlternativesToAuditory.owl>
 - <http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Content/AlternativesToAuditory/CaptionType.owl>

<http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Content/AlternativesToAuditory/CaptionType/ReducedSpeed.owl>

<http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Content/AlternativesToText.owl>

<http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Content/AlternativesToVisual.owl>

<http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Content/AlternativesToVisual/AudioDescription.owl>

<http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Content/AlternativesToVisual/ColorAvoidance.owl>

<http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Primary.owl>

<http://gia.inf.ufrgs.br/ontologies/IMS/Accessibility/ResourceDescription/Equivalent/Primary/Earl.owl>

- OBAA - <http://gia.inf.ufrgs.br/ontologies/OBAA/OBAA.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Technical.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Technical/PlatformSpecificFeatures.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Technical/PlatformSpecificFeatures/SpecificRequirement.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Technical/PlatformSpecificFeatures/SpecificRequirement/SpecificOrComposite.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Technical/Service.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Technical/Service/Details.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Technical/Service/Ontology.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Educational.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/Educational/Interaction.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/SegmentInformationTable.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/SegmentInformationTable/SegmentGroupList.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/SegmentInformationTable/SegmentGroupList/SegmentGroupInformation.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/SegmentInformationTable/SegmentGroupList/SegmentGroupInformation/Segments.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/SegmentInformationTable/SegmentList.owl>

<http://gia.inf.ufrgs.br/ontologies/OBAA/SegmentInformationTable/SegmentList/SegmentInformation.owl>

- VideoAula@RNP: <http://gia.inf.ufrgs.br/ontologies/VideoAula/VideoAula.owl>

<http://gia.inf.ufrgs.br/ontologies/VideoAula/Educational.owl>

<http://gia.inf.ufrgs.br/ontologies/VideoAula/Educational/Course.owl>

<http://gia.inf.ufrgs.br/ontologies/VideoAula/Technical.owl>

<http://gia.inf.ufrgs.br/ontologies/VideoAula/Technical/RelatedMedia.owl>

<http://gia.inf.ufrgs.br/ontologies/VideoAula/Technical/Thumbnail.owl>