



Programa Interdisciplinar de Pós-Graduação em  
**Computação Aplicada**  
Mestrado Acadêmico

Luiz Rodrigo Jardim da Silva

MSSearch: Busca Semântica de Objetos de Aprendizagem  
OBAA com Suporte a Alinhamento Automático de Ontologias

São Leopoldo, 2013

Luiz Rodrigo Jardim da Silva

**MSSearch: BUSCA SEMÂNTICA DE OBJETOS DE APRENDIZAGEM OBAA COM  
SUPORTE A ALINHAMENTO AUTOMÁTICO DE ONTOLOGIAS**

Dissertação apresentada como requisito parcial  
para a obtenção do título de Mestre pelo  
Programa Interdisciplinar de Pós-  
Graduação em Computação Aplicada da  
Universidade do Vale do Rio dos Sinos –  
UNISINOS.

Área de Atuação: Ciências Exatas e da Terra

Orientador: Prof. Dr. João Carlos Gluz

São Leopoldo

2013

S586m Silva, Luiz Rodrigo Jardim  
MSSearch: busca semântica de objetos de aprendizagem  
OBAA com suporte a alinhamento automático de ontologias /  
por Luiz Rodrigo Jardim da Silva. – 2013.  
91 f. :il. color. ; 30cm.

Dissertação (mestrado) -- Universidade do Vale do Rio dos  
Sinos, Programa de Pós-Graduação em Computação Aplicada,  
2013.

Orientador: Prof. Dr. João Carlos Gluz.

1. Web Semântica. 2. Busca semântica. 3. Ontologia. 4.  
Alinhamento de ontologia. 5. Objetos de aprendizagem. 6.  
Sistema multiagente. 7. Repositório semântico. I. Título. II.  
Gluz, João Carlos.

CDU 004.738.5:004.8

Catálogo na Publicação:  
Bibliotecária Eliete Mari Doncato Brasil - CRB 10/1184:

Luiz Rodrigo Jardim da Silva

**MSSearch: BUSCA SEMÂNTICA DE OBJETOS DE APRENDIZAGEM OBAA COM  
SUPORTE A ALINHAMENTO AUTOMÁTICO DE ONTOLOGIAS**

Dissertação apresentada como requisito parcial  
para a obtenção do título de Mestre pelo  
Programa Interdisciplinar de Pós-  
Graduação em Computação Aplicada da  
Universidade do Vale do Rio dos Sinos –  
UNISINOS.

Aprovado em: \_\_\_/\_\_\_/\_\_\_.

**BANCA EXAMINADORA**

---

Prof. Dr. João Carlos Gluz – UNISINOS (orientador)

---

Prof.Dr. Sandro José Rigo - UNISINOS

---

Prof.Dra. Rosa Maria Vicari - UFRGS

## **AGRADECIMENTOS**

Agradeço inicialmente à empresa Conforto Artefatos de Couro por ter me dado todo apoio para realização deste trabalho.

Ao meu orientador Prof. Dr. João Carlos Gluz, agradeço por ter acreditado na minha capacidade em desenvolver este trabalho e por ter proposto inúmeros desafios que contribuíram para o meu crescimento, durante esses dois anos de estudos.

Agradeço à CAPES pelo apoio financeiro.

Finalmente, gostaria de agradecer à minha família pela compreensão e pelo apoio durante esta jornada.

## RESUMO

Problemas relacionados à heterogeneidade semântica vêm se mostrando atualmente como um importante campo de pesquisa. Dentro do contexto educacional, pesquisadores têm se dedicado ao desenvolvimento de novas tecnologias que visam melhorar os processos de localização, recuperação, catalogação, e reutilização de objetos de aprendizagem. Baseado neste cenário, destaca-se o uso de técnicas de alinhamento de ontologias para prover integração entre ontologias distintas. Assim, o objetivo deste trabalho é desenvolver uma ferramenta que forneça mecanismos de busca semântica de objetos de aprendizagem com suporte a alinhamento automático de ontologias.

**Palavras-chave:** Web Semântica, Busca Semântica, Ontologias, Alinhamento de ontologias, Objetos de aprendizagem, Sistemas multiagentes, Repositórios semânticos.

## **ABSTRACT**

Semantics heterogeneity problems are becoming an important field of research. Within the educational context, researchers have focused on developing new technologies to improve the processes of localization, retrieval, cataloging, and reuse of learning objects. This scenario highlights the use of ontology alignment techniques to provide integration between different ontologies. Therefore, the goal of the present work is to develop a tool that provides mechanisms for semantic search of learning objects, with support for automatic aligning ontologies.

**Keywords:** Semantic Web, Semantic Search, Ontology Alignment, Learning Objects, Multiagent Systems, Semantic Repositories.

## LISTA DE FIGURAS

Figura 1: Tripla RDF .....	20
Figura 2: Consulta SPARQL ( <i>Query</i> ). .....	20
Figura 3: Atualização de metadados SPARQL ( <i>Update</i> ).....	21
Figura 4: Heterogeneidade semântica.....	25
Figura 5: Processo de alinhamento .....	27
Figura 6: Duas ontologias alinhadas.....	28
Figura 7: Arquitetura MILOS.....	35
Figura 8: Representação esquemática da ferramenta JAW .....	39
Figura 9: Arquitetura do sistema de busca Ochs .....	40
Figura 10: Arquitetura BROAD-PL .....	42
Figura 11: Arquitetura do serviço de ontologias .....	43
Figura 12: Tecnologias envolvidas no projeto .....	48
Figura 13: Arquitetura do sistema <i>MSSearch</i> .....	49
Figura 14: Diagrama de casos de uso do sistema .....	50
Figura 15: Alinhamento resultante e mapeamento da ontologia OBAA.....	53
Figura 16: Exemplo de rotulação de ontologias de domínios educacionais.....	54
Figura 17: Adaptação realizada do algoritmo da <i>AlignApi</i> .....	55
Figura 18: Processo de Alinhamento das Ontologias de Domínios Educacionais .....	56
Figura 19: Exemplo de alinhamento.....	58
Figura 20: Trecho OWL do Arquivo resultante do alinhamento.....	59
Figura 21: Funcionamento do sistema de busca semântica.....	60
Figura 22: Exemplo de consulta Sparql/Terp .....	60
Figura 23: Consulta Sparql para recuperação de OAs.....	62
Figura 24: Consulta TERP/SPARQL de OAs de um determinado catálogo.....	62
Figura 25: Consulta Terp/Sparql que retorna OA suportados pelo UNIX .....	63
Figura 26: Consulta de OA que funcionam tanto em UNIIX quanto SBTVD.....	63
Figura 27: Consulta que mostra a localização do conteúdo base e equivalente para SBTVD .	64
Figura 28: Disposição dos Agentes na arquitetura do sistema <i>MSSearch</i> .....	66
Figura 29: Diagrama de sequência da busca semântica .....	67
Figura 30: Diagrama de sequência da atualização de metadados.....	67
Figura 31: Diagrama de sequência da consulta Terp/Sparql .....	68
Figura 32: Diagrama de sequência do alinhamento.....	69
Figura 33: Menu principal do sistema .....	70

Figura 34: Busca semântica.....	71
Figura 35: Tela de login .....	71
Figura 36: Tela de Consulta para os formatos Terp/Sparql.....	72
Figura 37: Tela de atualização de metadados.....	73
Figura 38: Tela alinhamento.....	74
Figura 39: Consulta Sparql.....	76
Figura 40: Alinhamento das ontologias de matemática e lógica.....	78
Figura 41: Experimento teste de consulta com alinhamento.....	78

## LISTA DE TABELAS

Tabela 1: Incompatibilidade na Integração de Ontologias .....	26
Tabela 2: Ferramentas para alinhamento de ontologias. ....	32
Tabela 3: Análise comparativa dos trabalhos relacionados.....	45
Tabela 4: Detalhamento do caso de uso .....	50
Tabela 5: OAs retornados da consulta Sparql .....	62
Tabela 6: OA suportados por UNIX.....	63
Tabela 7: OA suportados em UNIX e SBTVD .....	64
Tabela 8: Localização do conteúdo base e equivalente para SBTVD.....	64
Tabela 9: Dados sobre carga de OA .....	76
Tabela 10: Dados sobre a consulta de OA.....	77
Tabela 11: Resultados da avaliação comparativa entre os sistemas <i>MSSearch</i> e <i>BIOE</i> .....	79

## LISTA DE SIGLAS

API	Application Programming Interface
CPU	Central Processing Unit
HTML	HiperText Markup Language
LARQ	Free Text Searches to SPARQL
MILOS	Multiagent Infrastructure for Learning Object Support
OA	Objetos de Aprendizagem
OAI-PMH	Open Archives Initiative Protocol for Metadata Harvesting
OBAA	Objetos de Aprendizagem Suportados por Agentes
OMV	Ontology Metadata Vocabulary
OWL	Ontology Web Language
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
SBIE	Simpósio Brasileiro de Informática na Educação
SDB	Persistent Triple Stores Using Relational Databases
SGBD	Sistema de Gerenciamento de Banco de Dados
TDB	Component of Jena for RDF Storage Query
URI	Universal Resource Identifier
WEB	World Wide Web
XML	Extensible Markup Language
YAGO	Yet Another Great Ontology

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>12</b>
<b>2 QUESTÃO DE PESQUISA .....</b>	<b>14</b>
<b>2.1 Caracterização do Problema e Motivações .....</b>	<b>14</b>
<b>2.2 Objetivos.....</b>	<b>15</b>
<b>2.3 Metodologia.....</b>	<b>16</b>
<b>3 FUNDAMENTAÇÃO DA PESQUISA.....</b>	<b>18</b>
<b>3.1 Tecnologias para Web Semântica .....</b>	<b>18</b>
3.1.1 Framework Jena.....	18
3.1.2 OWL API.....	19
3.1.3 Protégé.....	19
3.1.4 Linguagem SPARQL.....	20
3.1.5 Sintaxe Terp.....	21
<b>3.2 Repositórios semânticos para OA .....</b>	<b>22</b>
<b>3.3 Ontologias.....</b>	<b>23</b>
<b>3.4 Heterogeneidade Semântica.....</b>	<b>24</b>
<b>3.5 Alinhamento de ontologias.....</b>	<b>26</b>
3.5.1 Métricas de similaridades .....	29
3.5.2 Desempenho e eficiência .....	30
3.5.3 Ferramentas para alinhamento de ontologias .....	31
<b>3.6 Objetos de aprendizagem.....</b>	<b>33</b>
<b>3.7 Infraestrutura OBAA-MILOS .....</b>	<b>34</b>
<b>3.8 Agentes e sistemas multiagentes.....</b>	<b>36</b>
<b>4 TRABALHOS CORRELATOS .....</b>	<b>38</b>
<b>4.1 JAW .....</b>	<b>38</b>
<b>4.2 D-OSWS System: Google Sabe quem é Famoso hoje?.....</b>	<b>40</b>
<b>4.3 . BROAD-PL: Rede de Ontologias: apoio semântico a linha de produtos .....</b>	<b>41</b>
<b>de objetos de aprendizagem.....</b>	<b>41</b>
<b>4.4 Um Servidor de Ontologias para Sistemas de Biodiversidade .....</b>	<b>42</b>
<b>4.5 Análise dos trabalhos correlatos .....</b>	<b>43</b>
<b>5 <i>MSSearch</i> - BUSCA SEMÂNTICA DE OBJETOS DE APRENDIZAGEM.....</b>	<b>47</b>
<b>5.1 Visão geral do sistema .....</b>	<b>47</b>
<b>5.2 Arquitetura geral do sistema .....</b>	<b>48</b>
<b>5.3 Casos de uso do sistema <i>MSSearch</i> .....</b>	<b>49</b>
<b>5.4 Sistema de alinhamento.....</b>	<b>51</b>
5.4.1 Anotações para alinhamento e mapeamento .....	52
5.4.2 Adaptação da API de alinhamento .....	54
5.4.3 Processo de alinhamento .....	55
<b>5.5 Mecanismo de busca semântica.....</b>	<b>59</b>
<b>5.6 Repositório semântico .....</b>	<b>61</b>
<b>5.7 Agentes do sistema <i>MSSearch</i> .....</b>	<b>65</b>
<b>5.8 Protótipo do sistema .....</b>	<b>69</b>
5.8.1 Interface de interação.....	69
<b>6 EXPERIMENTOS E VALIDAÇÕES .....</b>	<b>75</b>
<b>6.1 Avaliação do desempenho do repositório semântico .....</b>	<b>75</b>
<b>6.2 Avaliação da funcionalidade do sistema de alinhamento .....</b>	<b>77</b>

<b>6.3 Validação do sistema <i>MSSearch</i> na percepção dos usuários .....</b>	<b>79</b>
<b>7 CONCLUSÕES.....</b>	<b>81</b>
<b>REFERÊNCIAS .....</b>	<b>83</b>
<b>ANEXO I - ARTIGOS PUBLICADOS .....</b>	<b>87</b>
<b>ANEXO II - FORMULÁRIO DE AVALIAÇÃO DO SISTEMA <i>MSSearch</i> .....</b>	<b>88</b>
<b>ANEXO III - RESULTADOS OBTIDOS DO QUESTIONÁRIO DE AVALIAÇÃO DO SISTEMA <i>MSSearch</i> .....</b>	<b>89</b>

## 1 INTRODUÇÃO

O Brasil atualmente vem adotando estratégias para ampliar a disseminação do conhecimento entre as escolas, através da disponibilização de ferramentas que possam auxiliar alunos e professores no processo de aprendizagem. Dentre as ferramentas disponíveis pode-se destacar o Banco Internacional de Objetos de Aprendizagem (BIOE), onde neles estão armazenados Objetos de Aprendizagem (OA) em diversos formatos (áudio, vídeos, imagens, etc). Para permitir a padronização e a interoperabilidade foram utilizados metadados para descrever seus principais atributos para catalogação (BIOE, 2012).

A catalogação de OA por meio de padrões de metadados contrasta com a grande parte das informações encontradas na Web (*World Wide Web*), pois estas estão armazenadas sem nenhum critério ou padrão de organização, tendo como objetivo principal a leitura e interpretação feitas por humanos. Devido ao armazenamento desestruturado dessas informações, a tarefa de busca e recuperação por meio de mecanismos inteligentes automatizados torna-se ineficiente em muitos casos.

Com intuito de dar mais significado às informações, Berners-Lee, Hendler e Lassila (2001) propuseram uma extensão da Web, denominada Web Semântica, onde a informação deve possuir um significado claro e bem definido, possibilitando a interação entre pessoas e máquinas.

Desde sua proposta inicial, vários aspectos da Web Semântica tem apresentado uma grande evolução. Em particular, a especificação dos significados (a semântica) dos elementos que compõe os documentos da Web tem se centrado fortemente no uso de ontologias, com a definição e adoção da linguagem OWL pelo W3C para a especificar as ontologias da Web Semântica. Porém, um dos principais desafios atuais da Web Semântica é prover mecanismos que permitam integrar ontologias heterogêneas, não só de domínios de conhecimentos distintos, mas até de mesmo domínio. Ontologias pertencentes ao mesmo domínio podem, muitas vezes, ser escritas através de vocabulários distintos, dificultando a interoperabilidade entre elas, bem como a implementação de ferramentas de busca. O *alinhamento de ontologias* (OAEI, 2012), vem sendo discutido amplamente como uma alternativa possível para o tratamento do problema de integração de ontologias heterogêneas. Essa é, entretanto, uma importante questão de pesquisa em aberto, porque a complexidade das tarefas de alinhamento exige que as técnicas, algoritmos e ferramentas empregadas neste processo, estejam preparadas para lidar com ontologias provenientes de diversas fontes.

Com base no problema descrito, pretende-se com este trabalho desenvolver um sistema que seja capaz de realizar alinhamento automático de ontologias provenientes de domínios educacionais, combinando o uso de tecnologias de agentes inteligentes, para realização de busca semântica de objetos de aprendizagem.

Este trabalho está organizado da seguinte forma: o capítulo 2 destina-se a caracterização do problema, os objetivos gerais e específicos, bem como a motivação para o desenvolvimento do trabalho e a metodologia utilizada. No capítulo 3 será apresentada a fundamentação teórica do trabalho destacando as tecnologias e as técnicas empregadas no seu desenvolvimento. Os trabalhos correlatos serão apresentados no capítulo 4. A arquitetura do sistema, além de sua modelagem e funcionamento é mostrada na Seção 5. Na Seção 6 poderão ser vistos os experimentos realizados para a validação do sistema, bem como os resultados obtidos. Finalmente, na Seção 7 serão apresentadas as conclusões obtidas com o desenvolvimento deste trabalho.

## 2 QUESTÃO DE PESQUISA

Este capítulo dedica-se a apresentação da caracterização do problema de pesquisa, a motivação para o seu desenvolvimento, os objetivos que se pretende alcançar, bem como a metodologia empregada na sua construção.

### 2.1 Caracterização do Problema e Motivações

Ferramentas para auxílio na Educação a Distância (EAD) vêm sendo utilizadas como meio de levar educação a todo território brasileiro, e seu uso extensivo está fazendo com que haja uma rápida evolução das tecnologias envolvidas nessa área. A utilização de objetos de aprendizagem possui um papel essencial no processo de ensino, desde que sua construção esteja compatível com a definição proposta, ou seja, um OA é caracterizado como qualquer entidade digital que pode ser utilizado, reutilizado e referenciado com apoio de computadores (IEEE-LTSC, 2012).

O uso de ontologias para descrição de propriedades e relações de domínios de conhecimento está se constituindo em peça chave dentro dos ambientes computacionais onde estão inseridos os objetos de aprendizagem, pois estas fornecem meios de representação do conhecimento contido no próprio OA, conhecimento sobre como está armazenado e como está publicado, facilitando o processo de busca e compartilhamento dos mesmos (GLUZ e VICCARI, 2010).

Entretanto, o crescente uso de ontologias para representação do conhecimento expõe algumas limitações, como a existência de ontologias de um mesmo domínio de conhecimento, mas criadas por fontes distintas e, muitas vezes, baseadas em vocabulários distintos. Com isso, tarefas como a busca semântica <sup>1</sup> de objetos de aprendizagem pode se tornar limitada ou até mesmo inviável, dificultando sua disseminação e seu compartilhamento.

Assim o problema de pesquisa do presente trabalho é verificar como mecanismos de alinhamento de ontologias, especializados para domínios educacionais podem tornar o processo de busca semântica mais abrangente nesses domínios, mas ao mesmo tempo possibilitando a apresentação de resultados mais significativos.

Portanto, o objetivo computacional da presente dissertação é desenvolver um sistema que possa contribuir para resolução do problema descrito, utilizando conceitos usados

---

<sup>1</sup> É uma técnica de pesquisa de dados que visa não só recuperar informações relacionadas com palavras-chave submetidas em um mecanismo de busca, mas sim leva-se em conta a intenção do usuário e o significado contextual acerca destas palavras.

atualmente, através do uso de tecnologias de alinhamento de ontologias, busca semântica, sistemas multiagente e armazenamento semântico de metadados.

Este trabalho está inserido dentro do contexto do projeto OBAA-MILOS que tem como objetivo criar uma infraestrutura baseada em agentes capaz de suportar o ciclo de vida completo de um objeto de aprendizagem compatível com a proposta de metadados OBAA (GLUZ e VICCARI, 2012). A infraestrutura resultante desse projeto é constituída por vários sistemas multiagente capazes de auxiliarem nas atividades de autoria, busca, uso e gerência de OA (GLUZ e VICCARI, 2012). A especificação de metadados OBAA (VICARI et al., 2010) é uma extensão dos metadados IEEE-LOM, que tem como objetivo suportar requisitos de acessibilidade e conteúdos multimídia, além de permitir a interoperabilidade de OA nas plataformas Web, TV-Digital e dispositivos móveis.

## 2.2 Objetivos

Em linhas gerais, este trabalho tem como objetivo principal desenvolver um sistema, baseado em agentes, que seja capaz de prover mecanismos para que ontologias distintas possam ser alinhadas, e com isso auxiliar na busca de objetos de aprendizagem estruturados conforme padrão de metadados OBAA(VICARI et al, 2010), armazenados em um repositório semântico nativo. Os objetivos específicos pretendidos com este trabalho são descritos a seguir:

- Implementar um sistema de busca semântica que seja capaz de lidar com bases de dados ontológicas heterogêneas;
- Definir uma arquitetura de sistemas multiagentes que implemente todas as funcionalidades necessárias para realização da busca semântica de OA;
- Modelar e desenvolver um mecanismo de integração de ontologias de domínios de ensino distintos, utilizando técnicas de alinhamento de ontologias especializadas para contextos educacionais;
- Realização do estudo do estado da arte dos temas envolvidos na pesquisa, como: busca semântica, alinhamento de ontologias, repositórios semânticos, sistemas multiagentes;
- Definir e implementar uma interface de serviços (Web Services);

- Especificar e implementar uma interface para que o usuário possa realizar buscas semânticas e atualizações de metadados de OA;
- Apresentar uma interface para que o usuário possa executar atualizações de metadados;
- Modelar e implementar um protótipo de repositório semântico que armazenará os OA;
- Desenvolver um conversor de OA armazenados em outros repositórios para popular um banco de dados semântico, visando a realização dos testes;
- Validar o sistema através de experimentos práticos;

Os objetivos listados acima também estão alinhados com objetivos importantes do projeto OBAA-MILOS. Os protótipos de sistemas e aplicações desenvolvidos nessa dissertação serão partes importantes da infraestrutura de agentes resultante desse projeto: o repositório semântico se constitui no núcleo do sistema de gerência de OA da infraestrutura (GLUZ e VICARI, 2010), enquanto que o sistema de busca semântica completa o sistema de busca de OA da infraestrutura (BARCELOS e GLUZ, 2011).

### **2.3 Metodologia**

Para que fosse possível concretizar os objetivos destacados na seção 2.2, uma metodologia de trabalho foi estabelecida, visando organizar o desenvolvimento deste projeto de pesquisa. Segue abaixo a metodologia adotada:

- Revisão da literatura dos temas inerentes ao projeto de pesquisa. Temas como: busca semântica, alinhamento de ontologias, repositórios semânticos e sistemas multiagentes;
- Identificação do problema de pesquisa e possíveis soluções já existentes, através do estudo de trabalhos relacionados;
- Testes e avaliações das tecnologias, identificando o seu potencial de contribuição para o presente trabalho;
- Projeto e implementação do sistema proposto, utilizando todo ferramental tecnológico previamente definido;

- Planejamento e modelagem de cenários para realização de testes e para validação do sistema proposto;
- Identificação de repositórios de objetos de aprendizagem educacionais para obtenção de metadados, possibilitando assim a realização de testes controlados em laboratório;
- Registro e documentação dos resultados obtidos através dos testes de validações;

### 3 FUNDAMENTAÇÃO DA PESQUISA

O presente trabalho caracteriza-se por utilizar uma gama de tecnologias e conceitos diretamente relacionados à Web Semântica. Muitas destas encontram-se bem consolidadas atualmente, sendo adotadas como padrão por diversos projetos, e são facilmente encontradas na literatura. Porém, algumas tecnologias são emergentes e carecem de uma revisão crítica mais aprofundada. Também se faz necessário este tipo de revisão para incorporar conhecimento necessário ao autor para então habilitá-lo a manipular todas estas tecnologias.

Destaca-se o uso de sistemas baseados em agentes, o uso de ontologias, a manipulação de metadados e técnicas de alinhamento de ontologias.

Sendo assim, dedica-se este capítulo a apresentar uma revisão teórica sobre os temas que estão diretamente relacionados ao trabalho desenvolvido. Tecnologias voltadas a manipulação e criação de sistemas para Web Semântica são descritos na Seção 3.1, na Seção 3.2 são apresentados alguns repositórios destinados ao armazenamento de triplas RDF. Na Seção 3.3 é realizada uma breve contextualização sobre ontologias. Nas Seções 3.4 e 3.5, temas referente a heterogeneidade semântica e alinhamento de ontologias são abordados. Já nas Seções 3.6 e 3.7 conceitos ligados a Objetos de Aprendizagem, apresentação da infraestrutura MILOS (na qual este projeto está inserido), são demonstrados respectivamente. Um resumo sobre tecnologias baseadas em agentes pode ser vista na Seção 3.8.

#### 3.1 Tecnologias para Web Semântica

##### 3.1.1 Framework Jena

O framework Jena é formado por um conjunto de APIs (*Application Programming Interfaces*) escritas em Java que auxiliam no desenvolvimento de sistemas dentro do contexto da Web Semântica. Este framework possui APIs que permitem realizar leitura e escrita RDF em formato XML, N-triples<sup>2</sup> e Turtle<sup>3</sup>. Também fornece bibliotecas para manipulação de ontologias, sendo possível realizar consultas e inferências sobre base de dados semântica (JENA, 2012).

Pode-se destacar dentro do projeto Jena, as APIs para armazenamento de triplas RDF em disco. Estes componentes permitem armazenar, de forma eficiente, uma grande

---

<sup>2</sup> Uma tripla RDF é formada pela sequência (Sujeito, Predicado e Objeto). N-triples especifica um padrão de serialização de um conjunto de triplas em arquivos de texto simples, sendo que cada tripla RDF deve estar organizada em uma linha somente.

<sup>3</sup> É uma extensão do padrão N-triples e permite armazenar triplas RDF em um formato simplificado.

quantidade de dados em formato RDF. Para armazenamento em banco de dados relacionais é disponibilizada a API SDB, e para armazenamento nativo de triplas RDF o componente denominado TDB (JENA, 2012).

Atualmente este conjunto de bibliotecas que formam o framework Jena, pode ser carregado diretamente do seu site oficial, sendo distribuído de forma livre para desenvolvimento e pesquisa.

### 3.1.2 OWL API

OWL API é constituído por uma série de bibliotecas destinadas à manipulação de ontologias escritas em OWL. Conforme OWL-API (2012), sua última versão fornece suporte a escrita e leitura de ontologias nos formatos RDF/XML, OWL/XML e Turtle, além de ser possível integrá-lo com os principais motores de inferência disponíveis, como: Pellet, FaCT++, HermiT e RacerPro.

Atualmente este framework é mantido e atualizado por uma comunidade de desenvolvedores, liderados pela Universidade de Manchester. O seu *download* pode ser realizado diretamente do site oficial, de forma gratuita.

### 3.1.3 Protégé

É um ambiente gráfico escrito em Java, direcionado a modelagem de ontologias, utilizando metodologias de Frames e OWL. Possibilita a exportação dos seus modelos para formatos RDF(S), OWL e XML (PROTÈGÈ-API, 2012). Abaixo algumas funcionalidades da ferramenta são destacadas:

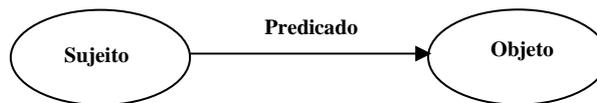
- a) Adição de *Plug-ins* escritos em Java, tornando a ferramenta mais flexível e extensível;
- b) Execução de *Reasoners* como classificadores de lógica de descrição;
- c) Edição de indivíduos para marcação de Web Semântica;
- d) Carregamento e criação de novas ontologias OWL e RDF;
- e) Extensão *OWLviz* para visualização em forma de gráfico de ontologias;

A ferramenta Protégé é distribuída de forma livre, sendo amplamente utilizada por desenvolvedores e pesquisadores de diversas áreas de conhecimento.

### 3.1.4 Linguagem SPARQL

Dados estruturados semanticamente são baseados na especificação RDF, formatados em triplas (sujeito, predicado, objeto) em arquivos do formato XML. Um grafo é formado a partir de um conjunto de triplas RDF, compondo assim uma base de dados semântica. Uma representação de uma tripla RDF é ilustrada na Figura 1.

**Figura 1: Tripla RDF**



Fonte: W3C (2012).

Devido ao seu modelo de estruturação e de como esses dados estão representados, torna-se impossível realizar buscas, atualizações e inserções de novos conjuntos de dados manualmente. Como em qualquer banco de dados relacional, o uso de linguagens específicas para manipulação de dados se torna imprescindível, como é o caso da linguagem SQL, muito utilizada em bancos de dados transacionais.

SPARQL é uma linguagem de consultas (*Query*) destinada à recuperação de informação sobre dados anotados semanticamente. Mais recentemente foi introduzida uma especificação para realização de atualizações (*Updates*).

Apesar de possuir uma sintaxe que auxilia na construção de sentenças para execução de consultas, é necessário que se tenha conhecimento da ontologia que esses dados estão representados, o que muitas vezes pode ser difícil para usuários que não conhecem estas estruturas.

**Figura 2: Consulta SPARQL (*Query*).**

```

1. PREFIX dc: <http://purl.org/dc/elements/1.1/>
2. PREFIX ns: <http://example.org/ns#>
3. SELECT ?titulo ?preco
4. WHERE { ?x ns:price ?preco .
5.         FILTER (?preco < 30.5)
6.         ?x dc:title ?titulo .
}

```

Fonte: W3C (2012).

**Figura 3: Atualização de metadados SPARQL (*Update*)**

```

1. PREFIX dc: <http://purl.org/dc/elements/1.1/>
2.
3. DELETE DATA FROM <http://example/bookStore>
4. {<http://example/book3> dc:title "Fundamentals of Compiler Desing 2" }
5.
6. INSERT DATA INTO <http://example/bookStore>
7. {<http://example/book3> dc:title "Fundamentals of Compiler Design 3" }

```

Fonte: W3C (2012).

Um exemplo de consulta SPARQL (*Query*) é apresentado na Figura 2, onde nas linhas 1 e 2 são definidos os prefixos para as URIs<sup>4</sup>, tornando mais fácil a construção das sentenças SPARQL. Já na linha 3, constam as variáveis que serão apresentadas com o resultado da busca. Na linha 4, a cláusula “WHERE”, especifica as restrições para recuperação das informações. Por fim, na Figura 3 é demonstrada a execução de um conjunto de instruções, que tem por finalidade realizar atualizações dos metadados armazenados em repositórios semânticos. Todo processo de atualização ocorre em dois passos distintos (Figura 3), sendo que primeiramente é executado uma instrução “DELETE”, e logo após “INSERT”. Como demonstrado no exemplo de consultas, na linha 1 é definido um prefixo para representar a URI onde se encontra o recurso, e na linha 3, através da instrução “DELETE DATA FROM” é especificado de qual fonte (*Graph Store*) a tripla será excluída. Por último, nas linhas 6 e 7, é realizado a inserção do novo dado, pelo comando “INSERT DATA INTO”.

Para facilitar o uso desses comandos, existem alguns frameworks que implementam as especificações destacadas, fornecendo uma camada de abstração e com isso auxiliando os desenvolvedores criarem sistemas Web Semânticos.

### 3.1.5 Sintaxe Terp

A evolução das tecnologias vinculadas à Web Semântica, possibilita que ocorra melhoramentos das técnicas e das ferramentas existentes. Destaca-se nesse contexto a linguagem de consultas TERP, sendo ela uma extensão da sintaxe SPARQL citada anteriormente. Basicamente ela permite que sujeitos ou objetos de uma tripla RDF, em uma consulta SPARQL, sejam especificados diretamente em OWL, através da sintaxe

<sup>4</sup> Identificador Uniforme de Recursos é utilizado para denominar ou identificar um recurso na internet.

Manchester<sup>5</sup>. Isto possibilita, por exemplo, que tipos de dados de variáveis SPARQL sejam definidos através de expressões *Manchester* OWL, aumentando muito a legibilidade da consulta, bem como sua simplificação (SIRIN; BULKA; SMITH, 2010).

### 3.2 Repositórios semânticos para OA

Existem vários repositórios destinados ao armazenamento de objetos de aprendizagem, dentre eles pode-se destacar: Ariadne<sup>6</sup>, Edna<sup>7</sup>, Merlot<sup>8</sup> e Cesta<sup>9</sup>. Porém, a maioria dos repositórios destacados utilizam bancos de dados relacionais para armazenamento das suas ontologias. Segundo Harrison e Chan (2005), a principal dificuldade de se armazenar ontologias em bancos de dados utilizando um SGDB é a exigência de se ter uma estrutura padronizada de armazenamento, o que muitas vezes a torna inviável. Processos de inferência e recuperação de informações podem acarretar em um demanda excessiva de computação, tornando o sistema imprevisível. Abaixo são destacadas algumas tecnologias para armazenamento de triplas RDF (PENA; PENYA, 2011):

- Sesame: Fornece armazenamento nativo OWL, porém não possui suporte nativo a inferência. Armazena os dados em formato binário, possibilitando uma compressão de dados eficiente e recuperação rápida de informações. Pode ser usado como uma biblioteca por outras tecnologias ou como um servidor dedicado (com Apache Tomcat);
- OWLIM: Também fornece armazenamento nativo semântico, e possui 3 versões: versão *Lite* que pode ser usada livremente, a versão *Standard* e *Enterprise* que possui algumas restrições de uso quanto a licença. Esta tecnologia pode ser usada em combinação com outras já existentes (Sesame e Jena API);
- AllegroGraph: Foi especialmente desenvolvido para sistemas 64 bits e utiliza Prolog Reasoning para inferência. Permite busca de texto livre e usa TWINQL (uma extensão da linguagem SPARQL) como linguagem de consulta;

---

<sup>5</sup> É uma sintaxe amigável e compacta designada para consultas semânticas, sendo suportada pela linguagem OWL 2.

<sup>6</sup> *Foudation for the European Knowledge Pool* (<http://www.ariadne-eu.org>).

<sup>7</sup> *Education Network Australia* ([http://apps-new.edna.edu.au/edna\\_retired/edna/go.html](http://apps-new.edna.edu.au/edna_retired/edna/go.html)).

<sup>8</sup> *Multimedia Educational Resource for Learning and Online Teaching* (<http://www.merlot.org>).

<sup>9</sup> Coletânea de Entidades de Suporte ao uso de Tecnologias de Aprendizagem (<http://cesta2.cinted.ufrgs.br/xmlui>).

- Oracle 11g: É uma plataforma para RDF apoiado pelo RDBMS (*Relational Database Management System*) da Oracle. Possui uma interface para interação com Protégé, Jena e Sesame;
- Jena TDB: Possibilita o armazenamento nativo de ontologias, possui suporte a inferências combinada com uso da biblioteca Pellet Reasoner<sup>10</sup>. Versões mais recentes já vêm com suporte a transações, sendo possível ainda realizar buscas por texto livre, utilizando a biblioteca LARQ (extensão do processador de consultas Jena ARQ). Possui versões para Windows e Unix, sendo distribuída de forma livre para uso;
- Jena SDB: API disponibilizada pelo framework Jena, fornece mecanismos para escrita e leitura de triplas RDF em banco de dados relacionais;
- Stardog: Banco de dados transacional semântico comercial. Fornece armazenamento nativo de triplas RDF, disponível para plataformas Windows e Linux;

Tecnologias para armazenamento de ontologias tem um papel importante dentro do contexto da web semântica, pois as mesmas devem lidar com uma quantidade muito grande de dados, prover mecanismos de inferência, recuperação e atualização de dados em um tempo aceitável. Como visto anteriormente, hoje em dia existem inúmeras ferramentas que são destinadas ao armazenamento de metadados estruturados no padrão OWL, podendo ser combinadas com tecnologias de bancos de dados tradicionais (SGBD), ou podem armazenar nativamente seus dados, sendo que estas ferramentas podem ser fornecidas de forma livre ou comercial.

Para auxiliar na escolha do repositório ideal, alguns critérios devem ser definidos e então avaliados, como: tempo de processamento de consultas, capacidade de armazenamento, escalabilidade e expressividade semântica (THAKKER et al., 2010).

### 3.3 Ontologias

Embora o conceito de ontologias tenha sua origem na filosofia, recentemente tem sido utilizada dentro da área da ciência da computação, como um mecanismo para representação de conhecimento. Dentro do contexto da Web Semântica, as ontologias possuem papel

---

<sup>10</sup> Motor de inferência escrito em Java que pode ser integrado aos frameworks Jena, OWL API e pela ferramenta de modelagem de ontologias Protégé.

fundamental para sua disseminação e para a evolução da Web atual, conforme sugerido por (BERNERS-LEE; HENDLER; LASSILA, 2001). Esta importância se dá principalmente pelo fato das ontologias fornecerem um modelo para organização das informações de um determinado domínio, através da definição de uma taxonomia de conceitos (classes) e das suas relações. E com o uso de regras de inferências, é possível extrair implicitamente informações armazenadas na WEB.

Para Gruber (1993), uma ontologia é uma especificação formal e explícita de uma conceitualização compartilhada. O conhecimento é explicitado através das definições de conceitos, das relações e axiomas, para que seja possível ser interpretados por computadores. A conceitualização refere-se ao modelo abstrato de uma determinada área de conhecimento.

Uma ontologia pode ser comparada como um nível de abstração do modelo de dados de sistemas de bancos de dados tradicionais, análogo aos modelos hierárquicos e relacionais. Porém, diferencia-se pelo fato de que a modelagem de ontologias se dá sobre o conhecimento da área ao qual ela representa, sendo que sua modelagem é feita sobre os seus indivíduos, seus atributos e suas relações com outros indivíduos (GRUBER, 2008).

Ontologias são descritas em linguagens OWL que possuem características de expressividade do formalismo lógico. Este tipo de especificação possibilita que ferramentas possam extrair informações de forma inteligente, usando motores de inferências, combinado com uso de agentes inteligentes automatizados.

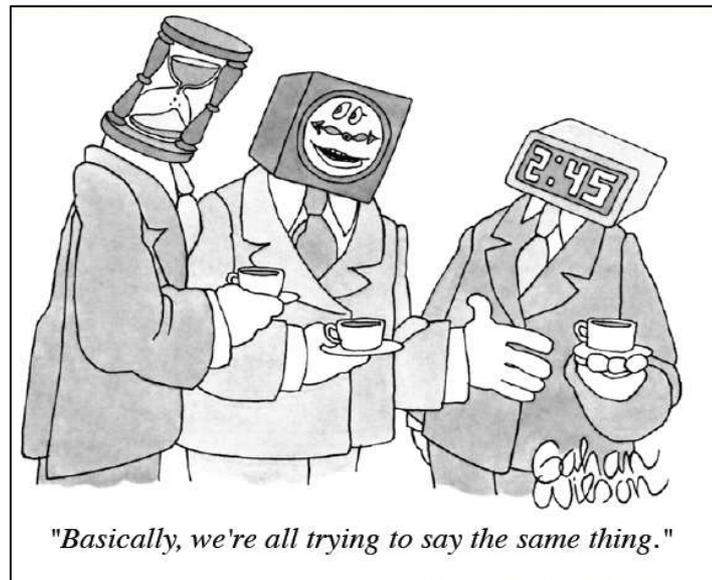
Levando-se em consideração que uma das premissas básicas das ontologias é que elas devem prover um entendimento comum e compartilhado sobre um determinado domínio, e que devem ser descritas de tal forma que computadores e humanos possam se comunicar, a necessidade de se integrar ontologias de fontes distintas torna-se peça chave dentro do papel da Web Semântica, permitindo que haja uma maior interoperabilidade entre ontologias e consequentemente uma ampliação da cobertura do conhecimento.

### **3.4 Heterogeneidade Semântica**

Segundo Ehrig (2007), com a expansão das bases de dados de conhecimento, os sistemas computacionais precisam estar cada vez mais preparados para lidar com informações que se encontram armazenadas de forma heterogêneas e distribuídas. Sendo assim, um dos grandes desafios da Web Semântica é encontrar mecanismos para que essas fontes sejam integradas (ou alinhadas), não só sintaticamente, mas também semanticamente. A Figura 4 faz uma analogia ao problema relacionado à heterogeneidade semântica.

Heterogeneidade Semântica refere-se ao fato de que os dados presentes em diferentes sistemas podem ser sujeitos a diferentes interpretações, mesmo quando os esquemas de banco de dados correspondentes são idênticos (GOH, 1997, p.21).

**Figura 4: Heterogeneidade semântica**



Fonte: Noy (2012)

Ainda segundo Goh (1997), o problema relacionado ao conflito semântico pode ser classificado da seguinte maneira:

- 1) *Conflito de nomes*: ocorre quando o valor de um atributo difere significativamente, e pode ser caracterizado pela presença de sinônimos e homônimos. O uso de padronizações e tabelas de mapeamentos podem ajudar a atenuar este problema;
- 2) *Conflitos de escalas e unidades*: geralmente é ocasionado por adoção de escalas de medidas distintas. Sistemas que medem valores precisam tratar desse problema constantemente;
- 3) *Conflitos de confusão*: por motivos temporais, mesmo que a informação tenha o mesmo significado, pode ser interpretada de forma errada;

O alinhamento de ontologias tem sido proposto como uma alternativa para prover interoperabilidade semântica, através da integração de ontologias escritas em vocabulários

distintos. Existe um grande esforço de comunidades de pesquisadores no desenvolvimento de novas técnicas de alinhamento, porém é um campo que ainda precisa ser muito explorado.

### 3.5 Alinhamento de ontologias

O uso de ontologias possibilita que o conhecimento seja formalizado de tal maneira a permitir que máquinas interajam com humanos, por meio de sistemas inteligentes.

Uma ontologia fornece um vocabulário que descreve um domínio de uma determinada área do conhecimento. Porém, estes vocabulários podem ser especificados utilizando uma linguagem ou um padrão distinto, e com isso pode-se gerar problemas de interoperabilidade, devido à impossibilidade de troca de informações entre essas ontologias.

Quando ontologias distintas precisam interoperar, alguns problemas de compatibilidade semântica podem gerar resultados imprevisíveis, ambíguos ou incompletos, contrapondo-se ao propósito da Web Semântica. Na Tabela 1 são apontadas algumas incompatibilidades provenientes da integração de ontologias (NOY, 2012).

**Tabela 1: Incompatibilidade na Integração de Ontologias**

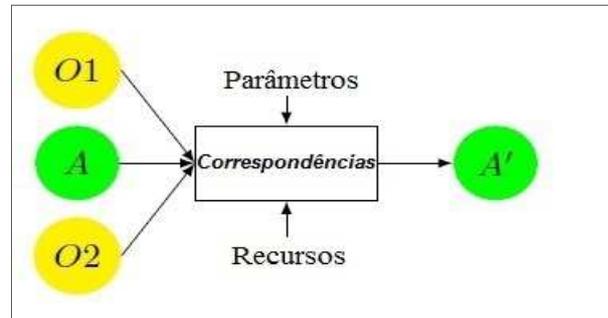
<b>Tipo de Incompatibilidade</b>	<b>Ontologia 1</b>	<b>Ontologia 2</b>
Diferentes nomes para os mesmos conceitos	<b>Código Postal</b>	<b>CEP</b>
Mesmos termos para diferentes contextos	<b>Projeto:</b> Somente projetos atuais	<b>Projeto:</b> Projetos e propostas antigas
Granularidade	<b>Professor Acadêmico</b>	<b>Adjunto, afiliado, associado, principal,</b> etc.
Modelagens distintas	<b>Jornal</b> é uma classe	<b>Jornal</b> é uma propriedade
Escopo	<b>Inclui</b> periódicos, publicações compostas	<b>Inclui</b> alunos, participantes, palestrantes
Foco diferente na sua definição	<b>Publicação:</b> Restrições em citações, referências.	<b>Publicação:</b> Restrições em resumos, editor.
Diferentes modelagens em nível detalhado	<b>endereço:</b> é uma <i>Property</i>	<b>endereço:</b> é uma <i>Data Property</i>

Fonte: NOY (2012).

Para Euzenat (2007), o termo *Alinhamento* é definido como o resultado do processo que tem a finalidade de encontrar correspondências entre entidades semanticamente

relacionadas, criando um conjunto de relações entre ontologias distintas. Este processo é denominado “*Ontology Matching*” (Correspondências de Ontologias), conforme ilustrado na Figura 5.

**Figura 5: Processo de alinhamento**



Fonte: Shvaiko e Euzenat (2011).

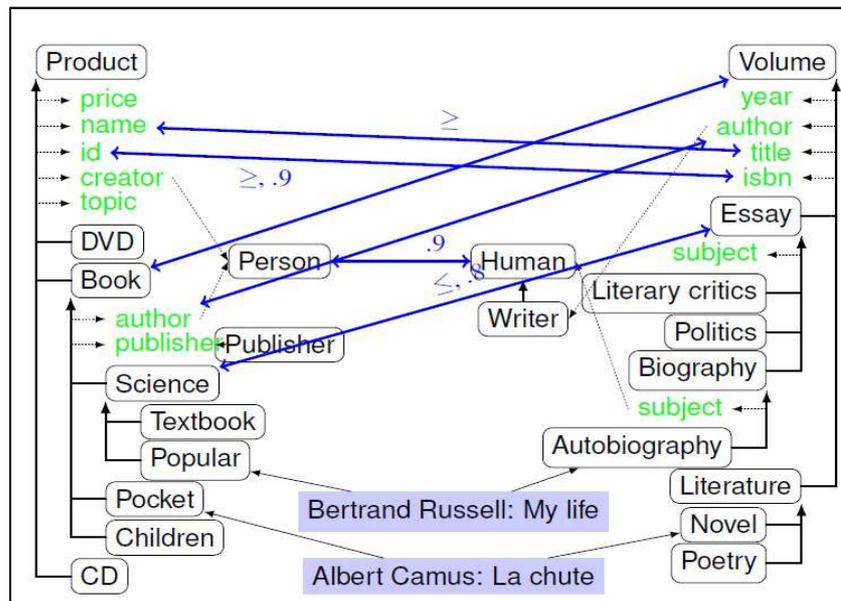
Uma função de alinhamento de ontologias *align*, é uma função parcial que parte do conjunto  $E$  de todas as possíveis entidades pertencentes as ontologias que se pretende alinhar e de um par de ontologias  $O_1, O_2$  resultando nos elementos de  $E$  que estão devidamente alinhados às ontologias  $O_1$  e  $O_2$  (EHRIG, 2007). Assumindo  $O$  como o conjunto das ontologias que se pretende alinhar, o tipo de *align* é:

$$align: E \times O \times O \rightarrow E$$

Sendo assim, para um dado elemento  $e$  do conjunto  $E$  e para duas ontologias  $O_1$  e  $O_2$ , temos  $align(e, O_1, O_2)$  com o elemento de  $E$  correspondente a  $e$  que está alinhado entre  $O_1$  e  $O_2$ . Note que para esse alinhamento ser possível o elemento  $e$  deve pertencer a ontologia  $O_1$ , enquanto que o elemento resultante  $align(e, O_1, O_2)$  pertence a ontologia  $O_2$ . Caso  $O_1$  e  $O_2$  estejam subentendidas, pode-se usar simplesmente  $align(e)$  para denotar esse elemento. Assim, caso exista um alinhamento parcial entre duas ontologias  $O_1$  e  $O_2$ , dizemos que uma entidade  $e$  está alinhada com a entidade  $f$  quando  $align(e)=f$ .

Todo processo de alinhamento parte de pares de entidades  $(e, f)$ , denominado de alinhamentos candidatos, sendo posteriormente necessário a realização de processos de verificação para consolidação do processo de alinhamento. Um exemplo de alinhamento entre duas ontologias é apresentado na Figura 6.

Figura 6: Duas ontologias alinhadas



Fonte: Euzenat (2007).

Segundo Ehrig (2007), o processo de alinhamento de ontologias pode ser realizado de três maneiras distintas:

- Manualmente: neste caso o próprio usuário realiza o alinhamento entre as ontologias, geralmente com apoio de ferramentas gráficas;
- Semi-Automático: o usuário participa em determinadas etapas do processo de alinhamento, porém conta com auxílio de algoritmos que podem realizar pré-alinhamentos, intervindo somente em algumas etapas no processo de tomada de decisão;
- Automático: todo o processo é realizado sem intervenção humana, apoiado por algoritmos inteligentes e técnicas de alinhamento sofisticadas;

Dependendo da quantidade de ontologias que se deseja alinhar, torna-se humanamente impossível realizar o procedimento manualmente (EHRIG, 2007). Dentre os processos destacados, o alinhamento automático vem recebendo uma atenção de destaque por parte dos pesquisadores, através do desenvolvimento e avaliações de algoritmos e de novas técnicas (OAEI, 2012).

### 3.5.1 Métricas de similaridades

É possível encontrar na literatura atual, diversas técnicas e metodologias para identificação de similaridades entre ontologias. Para Ehrig (2007), a similaridade de ontologias refere-se à comparação de um conjunto de ontologias e seus subelementos, sendo que esta comparação é realizada com base na utilização de funções que retornam um valor numérico indicando o grau de semelhança entre seus elementos. Esta função é expressa como:

$$sim: \mathcal{B}(E) \times \mathcal{B}(E) \times O \times O \rightarrow [0, 1]$$

Onde  $\mathcal{B}(E)$  representa o conjunto de entidades que se quer verificar a similaridade e  $O$  o conjunto de ontologias correspondentes.

Várias propriedades importantes podem ser definidas para a função de similaridade *sim*:

- $\forall e, f \in \mathcal{B}(E), O1, O2 \in O, sim(e, f, O1, O2) \geq 0$ : positividade da similaridade;
- $\forall e, f, g \in \mathcal{B}(E), O1, O2 \in O, sim(e, e, O1, O2) \geq sim(f, g, O1, O2)$ : maximalidade da similaridade;
- $\forall e, f \in \mathcal{B}(E), O1, O2 \in O, sim(e, f, O1, O2) = sim(f, e, O1, O2)$  : simetria da similaridade;
- $\forall e, f \in \mathcal{B}(E), O1, O2 \in O, sim(e, f, O1, O2) = 1 \Leftrightarrow e = f$ : identidade de entidades;
- $\forall e, f \in \mathcal{B}(E), O1, O2 \in O, 0 < sim(e, f, O1, O2) < 1$ : entidades similares (ou diferentes) até certo ponto;
- $\forall e, f \in \mathcal{B}(E), O1, O2 \in O, sim(e, f, O1, O2) = 0 \Leftrightarrow e \neq f$ : entidades completamente diferentes (não tem nenhuma característica em comum);

Euzenat (2007) define algumas técnicas básicas similaridade dentro do processo de alinhamento de ontologias, sendo elas:

- *Técnicas Terminológicas*: essas técnicas se baseiam nos textos encontrados dentro das ontologias para identificar suas entidades. Geralmente são originárias de técnicas de processamento de linguagem natural e recuperação de informação;

- *Técnicas Estruturais*: técnicas que verificam as relações entre as entidades das ontologias a serem alinhadas. Sendo que estas relações podem ser entre suas entidades e seus atributos, incluindo restrições de valores;
- *Técnicas Extensionais*: técnicas que comparam a extensão de entidades, com base nas suas instâncias. Dependendo das ontologias, suas respectivas classes são diferentemente rotuladas, porém o uso dessas técnicas pode deduzir uma relação a partir de seus indivíduos;
- *Técnicas Semânticas*: nessas técnicas leva-se em consideração a interpretação semântica, são utilizados métodos dedutivos para encontrar alinhamento ou detectar conflitos;

Muitas técnicas podem ser utilizadas de forma conjunta, incremental, de modo a encontrar os melhores resultados de alinhamentos. As técnicas terminológicas são as mais utilizadas para provimento de alinhamentos, isto se deve ao fato de ser mais facilmente implementadas, porém o ideal é que sejam utilizadas de forma conjunta com as outras técnicas, de modo a encontrar os melhores resultados de alinhamentos. Essas técnicas geralmente utilizam métodos de comparações de *strings*, distância de *strings* e medidas estatísticas com base em frequência e ocorrência de caracteres.

Para Cohen (2003), funções de distância de *strings* mapeiam um par de *strings*  $s$  e  $t$  para um número real  $r$ , onde o menor valor de  $r$  indica o grau de similaridade entre  $s$  e  $t$ . Funções de distância de edição correspondem aos algoritmos que identificam o grau de similaridades entre duas *strings*, através do número mínimos de operações necessárias para transformar uma cadeia de caracteres em outra. Essas operações podem ser de inserções, substituições e deleções de caracteres.

Dentre os algoritmos existentes, destacam-se, distância de *Levenstein*, de *Monger-Elkan* e *Jaro-Winkler* (COHEN, 2003). Ainda segundo Cohen (2003), o algoritmo de *Monger-Elkan* foi o que apresentou os melhores resultados, conforme dados obtidos através dos seus experimentos.

### 3.5.2 Desempenho e eficiência

A complexidade que envolve as tarefas de alinhamento exige que seus algoritmos e ferramentas estejam preparados para lidar com a integração de ontologias provenientes de

diversas fontes, que mesmo em casos em que pertencem a mesmos domínios, muitas vezes diferem na sua estrutura e na escrita dos seus vocabulários.

Com a adoção cada vez mais crescente no uso de ontologias, as técnicas e ferramentas de alinhamento existentes atualmente se deparam cada vez mais com a diversidade de domínios de conhecimento. Tendo em vista este cenário, a complexidade envolvida neste processo por vezes pode exigir muita capacidade computacional. O tempo de execução determina a eficiência do processo de alinhamento, e para que seja possível atingir uma boa resposta de tempo para este quesito, muitas vezes é necessário aumentar a capacidade de processamento, através do aumento da memória principal, largura de banda e CPU (SHVAIKO; EUZENAT, 2011).

Dentro do processo de alinhamento, algumas medidas podem ser tomadas para otimizar o sistema e garantir sua eficiência. O uso de heurísticas pode ser considerado com uma alternativa para se chegar a resultados próximos do ideal. Também, a definição de requisitos básicos de alinhamento, fornecendo subsídios para que algoritmos localizem “atalhos” dentro do processo, para que então seja possível se chegar a um resultado qualificado, em um tempo aceitável.

### 3.5.3 Ferramentas para alinhamento de ontologias

A comunidade OAEI (*Ontology Alignment Evaluation Initiative*), é uma organização que tem como objetivo avaliar métodos voltados a alinhamento e correspondências (*matching*) de ontologias. Esta iniciativa estabeleceu alguns objetivos a serem atingidos, conforme destacado abaixo:

- Avaliar a força e a fraqueza dos sistemas de alinhamentos atuais;
- Comparar o desempenho das técnicas existentes;
- Aumentar a comunicação entre a comunidade de desenvolvedores;
- Melhorar as técnicas de avaliação;
- Acima de tudo, promover a evolução das técnicas de alinhamento/casamento de ontologias;

Um levantamento das principais ferramentas e técnicas de alinhamento de ontologias existentes, conforme demonstrado na Tabela 2.

Tabela 2: Ferramentas para alinhamento de ontologias.

Ferramentas	Disponível para download	Código Aberto	Linguagem	Documentação	Última Atualização
ASMOV	X <sup>11</sup>		Java	Não	2012
DSSim	X		Java	Sim	2009
OntoDNA	X		Não Informado	Não	2007
Falcon	X	X	Java	Não	2008
FOAM		X	Java	Não	2006
SAMBO			Não Informado	Não	2008
RIMOM			Java	Não	2009
Lily	X		Java	Não	2009
Aroma	X	X	Java	Sim	2009
AlignAPI	X	X	Java	Sim	2012

Fonte: Elaborado pelo Autor.

Apesar de constarem na lista de ferramentas analisadas pela OAEI, muitas destas APIs atualmente encontram-se indisponíveis para *download*, ou não são disponibilizadas de forma gratuita. Outro fator que precisa ser destacado é o fato de que muitos desses sistemas estão descontinuados, impossibilitando ou comprometendo seu uso.

Com base no levantamento demonstrado anteriormente, realizou-se uma pesquisa, tendo como objetivo principal encontrar ferramentas que pudessem de fato implementar as técnicas de alinhamento de ontologias. Alguns critérios foram adotados a fim de selecionar o melhor sistema, como: os códigos fontes deveriam ser distribuídos no formato livre; seu desenvolvimento esteja plenamente ativo dentro de uma comunidade (OAEI); escrito em Java, devido ao ambiente de programação escolhido para este trabalho, bem como por fornecer compatibilidade com as APIs destacadas na seção 3.1.

Sendo assim, a ferramenta proposta por David et al. (2012), denominada *Alignment API*, mostrou-se como a API para alinhamento de ontologias com maior cobertura dos critérios de escolha apresentado. *Alignment API* oferece um conjunto de interfaces que auxiliam na realização de alinhamentos, sendo elas:

- *OntologyNetwork*: é um *container* para um conjunto de ontologias e alinhamentos, tornando sua recuperação e manipulação mais fácil ;
- *Alignment*: é definida como a classe principal da API, ou seja, essa classe possui toda as especificações do alinhamento;

<sup>11</sup> *Download* mediante solicitação encaminhada por email.

- *Cell*: representa a correspondência, pela união entre duas entidades com uma relação (*Relation*). Uma entidade pode ser identificada como qualquer elemento de uma ontologia, suportando qualquer tipo de metadados adicionais;
- *Relation*: representa a relação entre duas entidades;

A ferramenta descrita propõe-se a oferecer serviços de armazenamento e localização de alinhamentos, contribuir para a evolução dos algoritmos e das técnicas existentes. Mecanismos para manipulação de alinhamentos (fusão, inversão) e geração de testes também são serviços oferecidos pela API (DAVID et al., 2012).

### 3.6 Objetos de aprendizagem

O uso de objetos de aprendizagem dentro do contexto educacional promove a disseminação da educação, tornando assim um excelente meio de propagação de conhecimento.

Caracteriza-se como um objeto de aprendizagem (OA) qualquer entidade digital, ou não, que pode ser utilizada, reutilizada ou referenciada em um processo de aprendizagem apoiado por meios tecnológicos. Atualmente muito tem se discutido quanto à possibilidade de reutilização de objetos de aprendizagem, devido ao surgimento de diversas tecnologias e plataformas distintas, obrigando uma maior adaptabilidade destes objetos (IEEE-LTS, 2012).

Um objeto de aprendizagem, segundo MEC (2012), deve essencialmente ser reutilizável, podendo estar disposto na forma de animações, *slides* de apresentação ou ainda como um objeto para simulação, que neste caso possui um nível maior de complexidade. Alguns fatores corroboram para que OA sejam utilizados no processo de aprendizagem, sendo eles (MEC, 2007):

- *Flexibilidade*: possibilita ser reutilizado, sem nenhum custo adicional de manutenção;
- *Facilidade para Atualização*: estas entidades são atualizadas com muita frequência, devido à natureza a qual as mesmas se propõem. Geralmente são armazenados em um banco de dados, o que facilita este processo;
- *Customizável*: O mesmo objeto pode ser utilizado em vários cursos, podendo ser adaptado em diferentes contextos de aprendizagens;

- *Interoperável*: OA podem ser adaptados e utilizados em diferentes plataformas tecnológicas;

Além da reusabilidade, existe um grande esforço por parte dos pesquisadores quanto ao desenvolvimento de padrões de empacotamento e descrição de conteúdo de OA, tendo como objetivo promover acessibilidade e interoperabilidade. Estes padrões podem ser classificados em dois níveis: *nível de metadados*, onde são descritos as características do conteúdo de forma não relacionada a formatos de conteúdo ou sistemas de gerenciamento de conteúdo; *nível de conteúdo* é o próprio material de aprendizagem contido no OA (GLUZ et al., 2011).

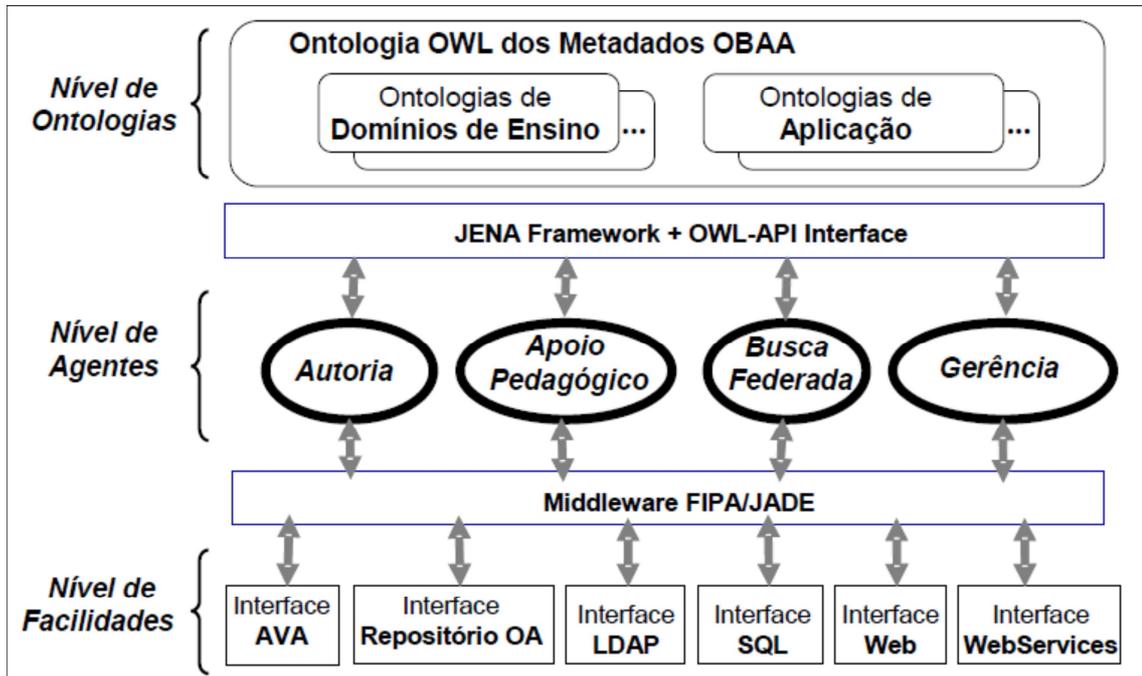
### 3.7 Infraestrutura OBAA-MILOS

O padrão OBAA proposto por Vicari et al. (2010), tem como objetivo fornecer mecanismos de interoperabilidade de objetos de aprendizagem em plataformas heterogêneas, como: TV Digital, dispositivos móveis e Web. Estes objetivos serão atingidos através de pontos de convergências entre a tecnologia de agentes, sistemas multiagente, objetos de aprendizagem e computação ubíqua, que permitem a especificação de padrões para OA, possibilitam a autoria, armazenamento e recuperação de OA.

De acordo com Gluz e Vicari (2010), uma das tarefas críticas em um projeto de qualquer sistema educacional é a definição do padrão de metadados para os OA. O padrão de metadados OBAA foi baseado no padrão IEEE-LOM, porém com um conjunto mais amplo de requisitos, tais como: adaptabilidade, interoperabilidade, compatibilidade com padrões internacionais, acessibilidade, independência e flexibilidade tecnológica.

A infraestrutura MILOS (*Multiagent Infrastructure for Learning Object Support*) é constituída por um conjunto de tecnologias, que combinadas oferecem suporte aos requisitos do padrão OBAA. A arquitetura MILOS é formada por três grandes níveis de abstração, conforme ilustrado na Figura 7.

Figura 7: Arquitetura MILOS



Fonte: Gluz e Vicari (2010).

Os componentes que compõem a arquitetura MILOS desempenham papéis específicos dentro da infraestrutura, que combinados habilitam seu funcionamento, sendo eles:

- *Nível Ontologias:* Representam a especificação dos conhecimentos que serão compartilhados entre os agentes;
- *Nível de Agentes:* Esta camada é constituída por agentes inteligentes, que darão suporte aos requisitos de adaptabilidade, interoperabilidade e acessibilidade, conforme a especificação do padrão OBAA;
- *Nível de Facilidades de Interface:* Este nível fornece mecanismos para que os agentes possam se comunicar com outros sistemas, permitindo que a infraestrutura possa interoperar;

Os agentes da infraestrutura MILOS possuem um papel importante dentro do sistema como um todo, pois eles devem incorporar os conhecimentos, com base nas ontologias, a fim de executar um conjunto de tarefas sem exigir conhecimento técnico por parte do usuário, sendo que este usuário pode ser humano ou outro agente de software proveniente de outras plataformas.

Quando implementadas as funcionalidades especificadas na proposta do projeto OBAA, é esperado que a infraestrutura MILOS ofereça os seguintes serviços:

- Catalogação de OA, auxiliando na especificação dos seus metadados;
- Adaptação e distribuição de objetos de aprendizagem para plataformas tecnológicas distintas, conforme especificado pelo padrão OBAA;
- Armazenamento, gerenciamento e publicação de OA para ambientes multiplataformas;
- Busca e localização semântica de OA, levando-se em consideração o contexto;

Atualmente inúmeros trabalhos estão sendo realizados dentro do contexto da proposta OBAA-MILOS, desenvolvidos por comunidades de pesquisadores e apoiados por universidades, tendo como objetivo principal evoluir as especificações do projeto proposto.

### **3.8 Agentes e sistemas multiagentes**

Com a expansão da internet e a necessidade cada vez maior de troca de informações entre sistemas de domínios diferentes, sendo que os mesmos muitas vezes estão dispostos de forma distribuída, tem levado a adoção novos paradigmas computacionais. Dentro desse contexto, o uso de tecnologias de agentes mostra-se como alternativa para essa nova era da computação.

Em termos gerais, tomando como base a definição proposta por Wooldridge e Jennings (1995), um agente basicamente caracteriza-se em termos computacionais como um processo de software executado em uma determinada plataforma tecnológica agindo de forma autônoma sobre um dado ambiente, podendo interagir com outros agentes para satisfazer objetivos definidos.

Os agentes individuais de um dado sistema multiagente podem ser projetados e desenvolvidos de várias formas distintas. Uma arquitetura de agentes individuais bastante disseminada é baseada em um modelo de estados mentais que idealiza os agentes de software como agentes racionais capazes das atitudes mentais de *crença*, *desejo* e *intenção* (em inglês, *Belief-Desire-Intention* – BDI). O modelo BDI se originou do modelo de raciocínio prático humano (BRATMAN et al. , 1988). Uma arquitetura BDI representa seus processos internos através dos estados mentais acima citados, definindo mecanismos de controle que selecionam de maneira racional o curso das ações (RAO; GEORGEFF, 1991).

A linguagem de programação AgentSpeak(L) foi projetada para a programação de agentes BDI na forma de sistemas de planejamento reativos (RAO, 1996). Um agente

AgentSpeak(L) é especificado por um conjunto inicial de crenças e um conjunto de planos (BORDINI et al., 2004). A ferramenta JASON (BORDINI et al., 2007) oferece um ambiente de desenvolvimento de sistemas multiagentes baseada em um interpretador para uma versão estendida da linguagem AgentSpeak(L) e também oferece uma série de extensões que são necessárias para o desenvolvimento de tais sistemas.

Um Sistema MultiAgente (SMA) é basicamente um sistema composto por múltiplos agentes de software que trabalham em conjunto para atingir um determinado objetivo (WOOLDRIDGE, 2002). A comunicação é fundamental para permitir que haja colaboração, negociação, cooperação entre os agentes. Nos SMA a comunicação deve ser até certo ponto padronizada para seja possível, necessitando assim uma linguagem que possa ser compreendida pelos agentes presentes no ambiente. Assim, a principal forma de comunicação entre agentes ocorre por meio de uma Linguagem de Comunicação entre Agentes (Agent Communication Language - ACL) (LABROU et al., 1999; CHAIB-DRAA e DIGNUM, 2002). A linguagem de comunicação de agentes mais utilizada atualmente é a linguagem FIPA-ACL (FIPA, 2002), disponível em diversas implementações, denominadas Plataformas de Agente. A plataforma de agentes aberta (*open-source*) mais disseminada atualmente é a plataforma JADE (BELLIFEMINE et al., 2007), que foi implementada inteiramente em JAVA, suportando agentes desenvolvidos em tal linguagem. A plataforma JADE disponibiliza um conjunto de classes que devem ser especializadas para criar os agentes. Por exemplo, o comportamento de um agente em particular é criado através de uma especialização da classe pré-definida Behaviour. A plataforma JADE é distribuída sob a licença pública GNU.

## 4 TRABALHOS CORRELATOS

A grande procura por recursos digitais educacionais na internet evidencia um cenário de transformação que eventualmente levará a uma mudança nas relações tradicionais de ensino entre aluno, professor e escola. Estes recursos quando encapsulados na forma de objetos de aprendizagem, favorecem a sua difusão, o seu compartilhamento, bem como sua reutilização. No entanto, para que isso seja possível, torna-se necessário o uso de ferramentas tecnológicas capazes de fornecer suporte aos personagens envolvidos. Frente a este cenário, nota-se um grande esforço por parte de pesquisadores em contribuir para a solução deste problema, sendo comprovado através da existência de vários trabalhos publicados sobre esses temas.

Com intuito de explorar e contextualizar as possibilidades tecnológicas relacionadas a pesquisa desenvolvida nessa dissertação, foram selecionados quatro trabalhos, cujo objetivos estão relacionados ao problema de pesquisa definido para a dissertação.

Este capítulo apresenta na Seção 4.1 um artigo que descreve uma ferramenta para recuperação semântica de objetos de aprendizagem baseada em tesauros. Já na Seção 4.2 é apresentado um sistema que implementa busca semântica com suporte a sugestões de pesquisas dinâmicas. A Seção 4.3 traz uma proposta de desenvolvimento de uma sistema baseado em redes de ontologias, para criação de objetos de aprendizagem. Finalmente, na Seção 4.4 é apresentada uma ferramenta que tem na sua proposta principal, oferecer um serviço Web para integrar ontologias da área da biologia.

### 4.1 JAW

O trabalho proposto por Souza et al. (2008), tem por finalidade recuperar objetos de aprendizagens em repositórios semânticos, utilizando tesauros genéricos da língua inglesa .

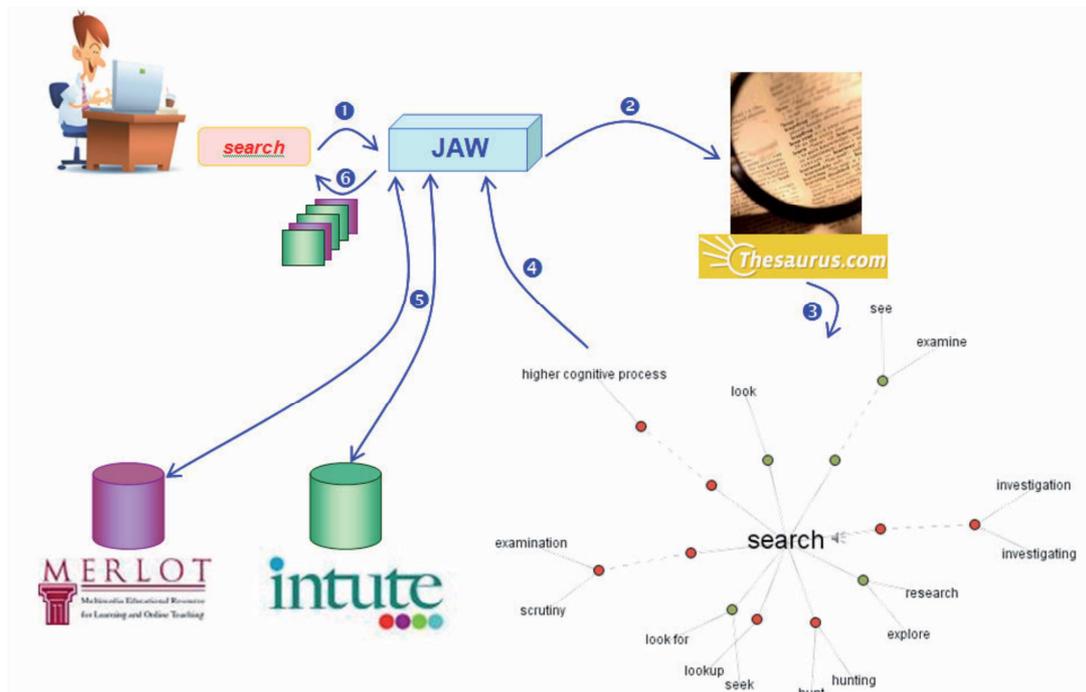
Este projeto resultou em uma ferramenta denominada JAW, e baseia-se no uso de tesauros no formato de texto estático (*thesaurus.com*) navegáveis, para extração sinônimos provenientes da língua inglesa. A semântica, conforme destacado pelos autores, emerge dos relacionamentos entre esses termos, proporcionando um aumento dos critérios de busca, bem como na precisão dos resultados obtidos. A seguir uma descrição mais detalhada sobre o funcionamento do sistema JAW será apresentada, conforme esquematização vista na Figura 8:

- Interface de busca (*Search*): para que o usuário consiga interagir com o sistema, uma interface WEB é disponibilizada. Nesta tela, o usuário informa a palavra chave que

deseja consultar e então envia uma requisição para que o sistema consulte no banco de dados de tesouros ;

- Extração dos termos: logo após acionada a pesquisa, a ferramenta se conecta ao banco de dados de tesouros, extrai os sinônimos com base na palavra chave fornecida;
- Processamento da Consulta: após a consulta no banco de dados de tesouros, é montada uma lista de termos semelhantes à palavra fornecida, sendo que, sobre essa lista uma estratégia de *ranking* é aplicada, visando retornar resultados com maior relevância para o usuário.
- Apresentação dos Resultados: com a lista de termos ordenadas de acordo com a estratégia de *ranking* aplicada, o sistema de busca é requisitado novamente, porém desta vez o mesmo envia essa lista para os repositórios, retornando os objetos de aprendizagens armazenados;

Figura 8: Representação esquemática da ferramenta JAW



Fonte: Silva et al. (2008)

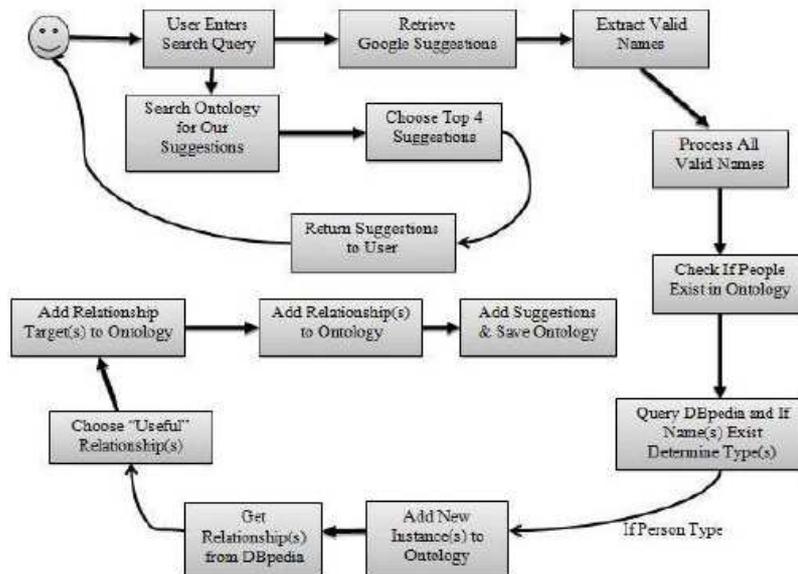
Ainda segundo os autores, este sistema pode ser facilmente portado para utilizar tesouros genéricos de qualquer língua, inclusive a portuguesa.

## 4.2 D-OSWS System: Google Sabe quem é Famoso hoje?

Mecanismos atuais de busca precisam lidar com uma grande quantidade de dados, que geralmente estão armazenados em bancos de dados distribuídos e distintamente estruturados. Para que estas ferramentas sejam bem sucedidas, muitas vezes é necessário que as mesmas interajam com o usuário. Desta forma, no trabalho realizado por Ochs et al. (2011), foi apresentado o sistema de busca D-OSWS (*Dynamic Ontology-Supported Web Search*), que constrói dinamicamente uma ontologia de “pessoas famosas”, combinada com dados provenientes do DBpedia<sup>12</sup>.

O objetivo principal deste sistema é apresentar ao usuário sugestões de termos de pesquisas dinamicamente, com desambiguação de homônimos. Este processo se dá através de sugestões de pesquisas extraídas a partir do *Google* e do mapeamento das ontologias YAGO (*Yet Another Great Ontology*) e DBpedia. A arquitetura deste sistema pode ser vista na Figura 9.

Figura 9: Arquitetura do sistema de busca Ochs



Fonte: Ochs et al (2011).

Para tratamento das desambiguações uma *Tag* é adicionada junto aos dados obtidos. Por exemplo, conforme o autor, existe um grande número de páginas que se refere a *Michael Jordan* no Wikipédia, ou seja, existe a página “Michael\_Jordan” que é um famoso jogador de basquete, também “Michael\_Jordan” jogador de futebol, bem como o mesmo nome para um

<sup>12</sup> Projeto que visa extrair informações estruturadas do Wikipédia.

político irlandês. Esta *Tag* foi inserida no final do nome da página extraída, entre parênteses, conforme segue: “Michael\_Jordan\_(footballer)” e “Michael\_Jordan\_(Irish politician)”.

### **4.3 . BROAD-PL: Rede de Ontologias: apoio semântico a linha de produtos de objetos de aprendizagem**

O artigo em questão define uma infraestrutura baseada na abordagem de linha de produtos de OA, associada a uma rede de ontologias para oferecer apoio semântico à busca, recuperação e ao reuso de objetos de aprendizagem. Com isso, este sistema apoiará os usuários na construção de OA voltados para a composição de processos de ensino e aprendizagem (CAMPOS et al., 2012).

Este trabalho é uma versão estendida do projeto BROAD<sup>13</sup>, que tem como objetivo fornecer uma infraestrutura completa para gerenciamento de objetos de aprendizagem, onde cada versão representa uma melhoria ou um complemento do projeto anterior.

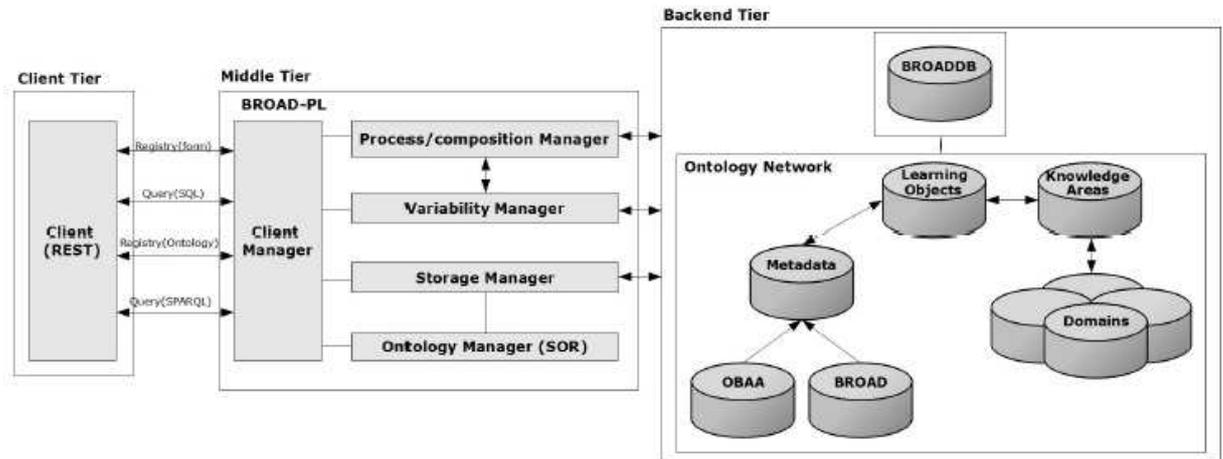
Para garantir o reuso e compartilhamento do conhecimento, este projeto utiliza uma rede de ontologias denominada BROADNet (Figura 10). Os autores destacam que esta rede é formada por:

- Ontologias de topo: representam áreas de conhecimento (*Knowledge Areas*);
- Ontologias gerais: representam as ontologias de metadados, para catalogação de objetos de aprendizagem (*Learning Objects*). Por exemplo a ontologia de metadados OBAA;
- Ontologias de domínio (*Domains*): representam um domínio específico de uma área de conhecimento. Por exemplo: uma ontologia de doenças humanas (*Human Disease*) pertence à área de conhecimento de ciências biológicas;

---

<sup>13</sup> O projeto BROAD busca a adoção de tecnologias como ontologias, serviços web semânticos, agentes e *workflow* para a construção de uma arquitetura para composição e sequenciamento de OA (CAMPOS et al., 2012).

Figura 10: Arquitetura BROAD-PL



Fonte: Campos et al. (2012).

Como destacado anteriormente, este projeto tem como finalidade conceber OA a partir de outros objetos já pré-existent, possibilitando assim sua derivação. Este processo é apoiado por ontologias que estão semanticamente relacionadas através de técnicas de alinhamento/mapeamento de ontologias.

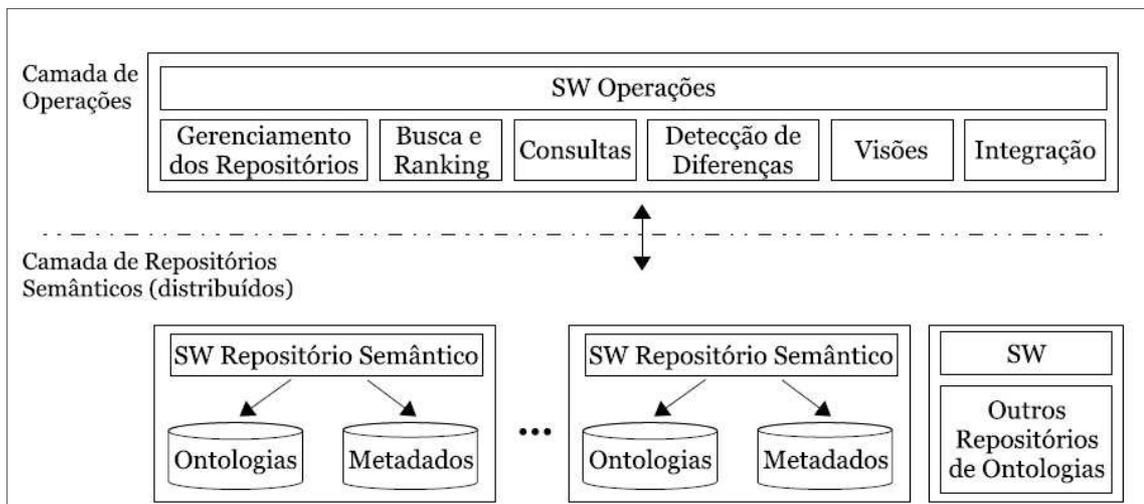
#### 4.4 Um Servidor de Ontologias para Sistemas de Biodiversidade

O uso de ontologias para correlacionar dados sobre seres vivos e seus habitats são frequentemente utilizados por pesquisadores da área de biologia. Isso deve principalmente pelo fato de que as ontologias possibilitam a classificação desses seres e suas relações. Porém, devido a grande quantidade de informações, onde muitas vezes são armazenadas e estruturadas de forma heterogênea, impossibilita a interoperabilidade dos sistemas dessa área. Dentro deste contexto, o trabalho apresentado por Daltio e Medeiros (2005), se propõe a contribuir para resolução desse problema, através do desenvolvimento de um serviço web que possa integrar diferentes fontes de informações da área de biologia.

O sistema proposto é composto por duas camadas, conforme demonstrado na Figura 11. A camada referente ao repositório semântico é responsável por armazenar os metadados, e estão estruturados no formato OMV (*Ontology Metadata Vocabulary*). A camada de operações é responsável por gerir e disponibilizar os serviços oferecidos pela ferramenta. Dentre os módulos constantes neste trabalho, pode-se destacar os seguintes:

- Gerenciamento dos Repositórios: funciona como uma interface de acesso e manipulação dos metadados armazenados. Aqui são executadas funções de inclusão, alteração e exclusão;
- Busca e *Ranking*: no momento que é submetido uma busca, o sistema retorna um conjunto de 1 até  $N$  ontologias que "casam" integralmente o parcialmente com o termo fornecido. A estratégia de *Ranking* utilizada é baseada conforme uma análise das estruturas internas (suas taxonomias) das ontologias retornadas, ou seja, quanto mais similar for a ontologia melhor será sua posição no *ranking*;
- Integração de Ontologias: este módulo fornece mecanismos de alinhamento de ontologias. Este alinhamento é realizado com auxílio de algoritmos de comparação de similaridades de *strings*.

**Figura 11: Arquitetura do serviço de ontologias**



Fonte: Daltio e Medeiros(2005).

#### 4.5 Análise dos trabalhos correlatos

A análise comparativa dos trabalhos correlatos apresentados tem por objetivo identificar as possíveis lacunas existentes dentro do contexto de pesquisa no qual o presente trabalho está inserido. Tendo em vista este objetivo, foram definidos alguns critérios de comparação a fim de propor possíveis contribuições. Abaixo a descrição desses critérios são destacados:

- Interoperabilidade: entende-se como interoperabilidade a capacidade do sistema integrar ontologias provendo assim uma maior cobertura do conhecimento;
- Independência de Domínio: capacidade do sistema integrar ontologias de domínios distintos;
- Uso de Inferência: uso de mecanismos de inferência para extrair informações armazenadas implicitamente;
- Repositório Semântico: informações estruturadas conforme modelos ontológicos, geralmente são organizadas em forma de metadados. É entendido que repositórios para armazenamento de metadados, precisam ser projetados para lidar com uma grande quantidade de informação e suportar mecanismos de inferências (HARRISON e CHAN, 2005) ;
- Uso de Agentes: uso de agentes de software que sejam capazes de incorporar todo o conhecimento necessário para resolução de tarefas pertinentes ao sistema;
- Interface de Serviços: capacidade do sistema oferecer uma interface de serviços padronizada para que outros sistemas interajam, possibilitando o reuso da tecnologias independente da plataforma;
- Interface de Web: capacidade do sistema oferecer uma interface, com um *Layout* humanamente operável para que o usuário possa interagir com o sistema, independente do local. Este critério, apesar de parecer irrelevante, tem um certo grau de importância na visão do pesquisador deste projeto, pois muitos softwares são desenvolvidos no formato *Desktop* e não estão disponíveis na WEB, com isso dificulta a validação de ferramentas que estão formalizadas em formato de artigos ou em outros trabalhos científicos;

É possível verificar na Tabela 3 a análise dos trabalhos relacionados, de acordo com os critérios estabelecidos anteriormente. Para um melhor entendimento, os quadros assinalados como (Sim) indicam que o critério foi atendido, já os quadros marcados como (Não) indicam que não foram atendidos. Poderá haver casos em que o critério tenha sido atingido parcialmente, sendo então assinalada como (Parcial).

**Tabela 3: Análise comparativa dos trabalhos relacionados.**

<b>Critério</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Interoperabilidade	Não	Parcial	Parcial	Sim	Sim
Independência de Domínio	Não	Não	Não	Parcial	Parcial
Uso de Inferência	Não	Parcial	Parcial	Não	Sim
Repositório Semântico	Sim	Não	Parcial	Sim	Sim
Uso de Agentes	Não	Não	Não	Não	Sim
Interface de Serviços	Não	Não	Parcial	Não	Sim
Interface Web	Sim	Sim	Não	Não	Sim
<b>LEGENDA:</b> 1 - JAW: Recuperação Semântica de Objetos de Aprendizagem: Uma Abordagem Baseada em Tesouros de Propósito Genérico (SOUZA et al., 2008). 2- D-OSWS System: Google Sabe quem é Famoso hoje? (OCHS et al., 2011). 3- BROAD-PL: Rede de Ontologias: apoio semântico a linha de produtos de objetos de aprendizagem (CAMPOS et al., 2012); 4- Um Servidor de Ontologias para Sistemas de Biodiversidade (DALTIO e MEDEIROS, 2005). 5- <i>MSSearch</i> : Busca Semântica de Objetos de Aprendizagem OBAA com Suporte a Alinhamento Automático de Ontologias.					

Fonte: Elaborado pelo Autor

Observou-se durante o desenvolvimento desta pesquisa, que muitos trabalhos abordam de forma pouco concisa questões referente à interoperabilidade semântica para o compartilhamento de conhecimento, sendo essa uma das premissas da Web Semântica. Outro dado importante observado é referente o pouco uso de tecnologias baseadas em agentes para realização de buscas semânticas, bem como o provimento da interoperabilidade semântica.

O trabalho que mais apresentou semelhanças com a proposta, foi o projeto desenvolvido por Daltio e Medeiros (2005), isto porque o mesmo trata do problema de interoperabilidade usando técnicas de alinhamento de ontologias. Porém, apesar desta semelhança, o artigo não especifica quais as técnicas que foram utilizadas para realização da integração das ontologias utilizadas no seu projeto.

Importante salientar a parcialidade destacada no critério que trata da independência de domínios, pois apesar deste sistema dar enfoque no alinhamento de ontologias de domínios educacionais, possivelmente a técnica utilizada poderá ser facilmente adotada para alinhamento de ontologias de outros domínios.

Com base nos resultados obtidos da avaliação, é possível concluir que o sistema proposto contribui para a cobertura das carências identificadas, através da construção de um sistema cujo sua arquitetura engloba várias tecnologias que de maneira geral são utilizadas separadamente nos trabalhos correlatos pesquisados.

Portanto, conclui-se que o principal diferencial deste projeto de pesquisa frente aos demais, está no uso efetivo de técnicas de alinhamento automáticas de ontologias, para construção de um sistema de busca capaz de recuperar semanticamente objetos de aprendizagem .

## 5 *MSSearch* - BUSCA SEMÂNTICA DE OBJETOS DE APRENDIZAGEM

O armazenamento estruturado de objetos de aprendizagem, organizado através das informações contidas nos seus metadados, pode fornecer subsídios para que sistemas computacionais os recuperem semanticamente, facilitando o seu uso e a sua disseminação em ambientes educacionais. Porém, existem grandes desafios, como tratar as informações estruturadas em modelos ontológicos de representação de conhecimento e como alinhar essas informações, quando originalmente estruturadas em ontologias distintas.

Neste capítulo será especificado o sistema *MSSearch* (*MILOS Store and Search*), que é proposto como uma evolução dos sistemas e aplicações vistos no Capítulo 4, capaz de auxiliar na resolução da questão de pesquisa levantada no Capítulo 2. Assim, este sistema foi projetado para tratar do problema de representar e alinhar ontologicamente as informações de metadados dos Objetos de Aprendizagem, com fins de suporte à busca semântica desses objetos. Para tanto, o sistema utiliza os conceitos e tecnologias apresentados no Capítulo 3 da fundamentação teórica.

Na Seção 5.1, uma visão geral do sistema com a pilha de tecnologias utilizadas e sua relação com a infraestrutura MILOS. A arquitetura geral do sistema *MSSearch* é mostrado na Seção 5.2, bem como o seu diagrama de caso de uso na Seção 5.3. Uma descrição mais aprofundada do processo de alinhamento pode ser visto na Seção 5.4. O detalhamento do mecanismo de busca implementado é mostrado na Seção 5.5. A Seção 5.6 destaca as características do repositório semântico implementado no *MSSearch*. Já a Seção 5.7 apresenta a especificação dos agentes utilizados no sistema. Por fim, na Seção 5.8 é descrito o protótipo do sistema e o seu funcionamento.

### 5.1 Visão geral do sistema

O sistema *MSSearch* tem por finalidade auxiliar na busca e recuperação semântica de objetos de aprendizagem que estejam armazenados em um repositório semântico com suporte OWL nativo, e que sejam estruturados pelo padrão de metadados OBAA e pela sua correspondente ontologia OWL (VICARI e GLUZ, 2012). O processo de busca semântica implementado pelo *MSSearch* leva em consideração ontologias de domínios educacionais previamente alinhados. Este alinhamento possibilita uma maior cobertura do conhecimento sobre o domínio inserido, e ao mesmo tempo restringe o universo da busca através da apresentação de resultados com maior relevância. A execução de inferências sobre esses modelos habilita o sistema a extrair informações armazenadas implicitamente.

Para viabilização do sistema foi necessário o uso combinado de diversas ferramentas e tecnologias. Na Figura 12, estão dispostas as principais tecnologias utilizadas para concepção do sistema proposto.

**Figura 12: Tecnologias envolvidas no projeto**



Fonte: Elaborado pelo Autor

Como observado na Figura 12, a ferramenta desenvolvida é um projeto de pesquisa resultante da proposta OBAA-MILOS. É possível ainda visualizar dentro do contexto do presente trabalho a pilha de tecnologias utilizadas, como: JSF<sup>14</sup> usada como *framework* para desenvolvimento da interface Web, RESTFul<sup>15</sup> para camada de interface de serviços (Web Service), Jade/FIPA como plataforma de agentes, as linguagens de consultas OWL SPARQL/TERP, bem como a biblioteca de inferências Pellet. A *AlignAPI* incorpora todas as funcionalidades necessárias para realização de alinhamentos, e o repositório semântico utilizado para o armazenamento dos metadados estruturados conforme padrão OBAA.

## 5.2 Arquitetura geral do sistema

A arquitetura geral do sistema desenvolvido segue as diretrizes da Infraestrutura MILOS (ver Seção 3.7), sendo organizada em três camadas. A Figura 13 mostra esta organização, além da disposição dos componentes de cada camada. Sua estrutura se divide basicamente em três camadas:

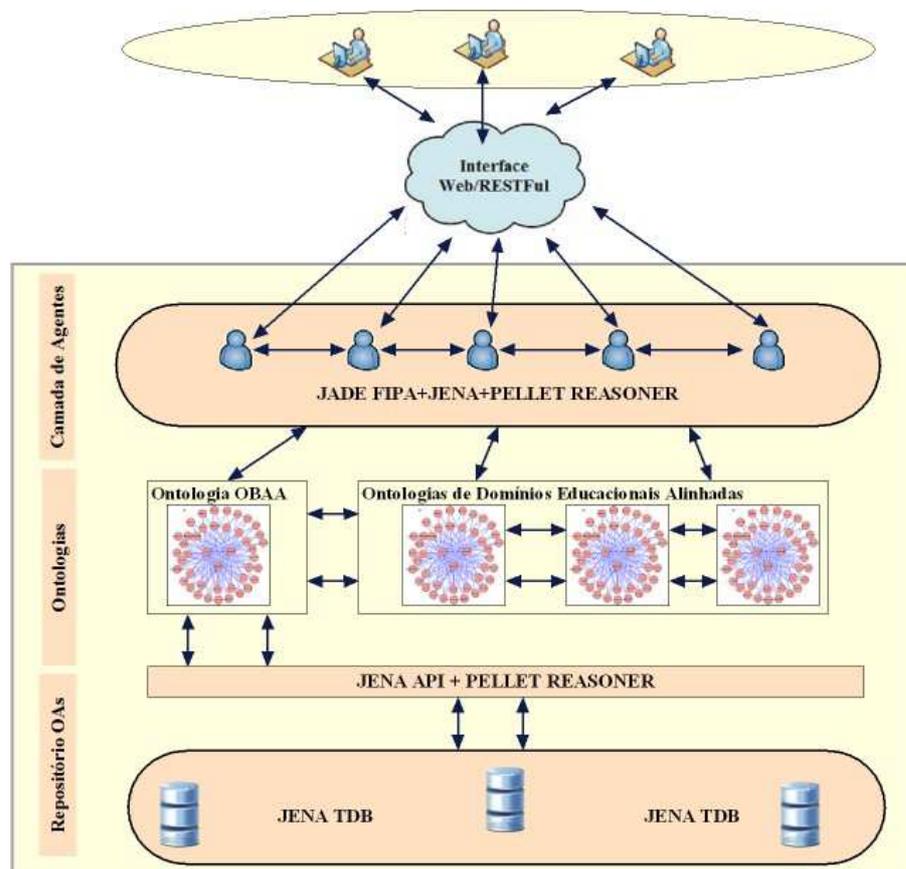
- *Camada de Acesso*: camada responsável pela interação do sistema com o usuário via página Web, ou com outros sistemas, via Web Services;

<sup>14</sup> *Java Server Faces* - Framework Web baseado em Java.

<sup>15</sup> Web Service caracterizado por não usar trocas de mensagens e sim o uso de recursos nomeados em forma de URIs (*Uniform Resource Identifier*), URLs (*Uniform Resource Locator*) e URN (*Uniform Resource Name*).

- *Camada de Agentes*: composta por agentes de software, inseridos dentro de uma plataforma (*Container*) Jade, que se comunicam por intermédio do protocolo FIPA, visando atingir objetivos previamente estabelecidos de acordo com sua função;
- *Camada de Armazenamento*: nesta camada é implementado um repositório semântico, com suporte OWL nativo, onde os metadados padrão OBAA estarão armazenados. A organização, acesso e atualização dessas informações é baseada na ontologia de metadados OBAA, eventualmente mapeada com outras ontologias sobre domínios educacionais;

Figura 13: Arquitetura do sistema *MSSearch*

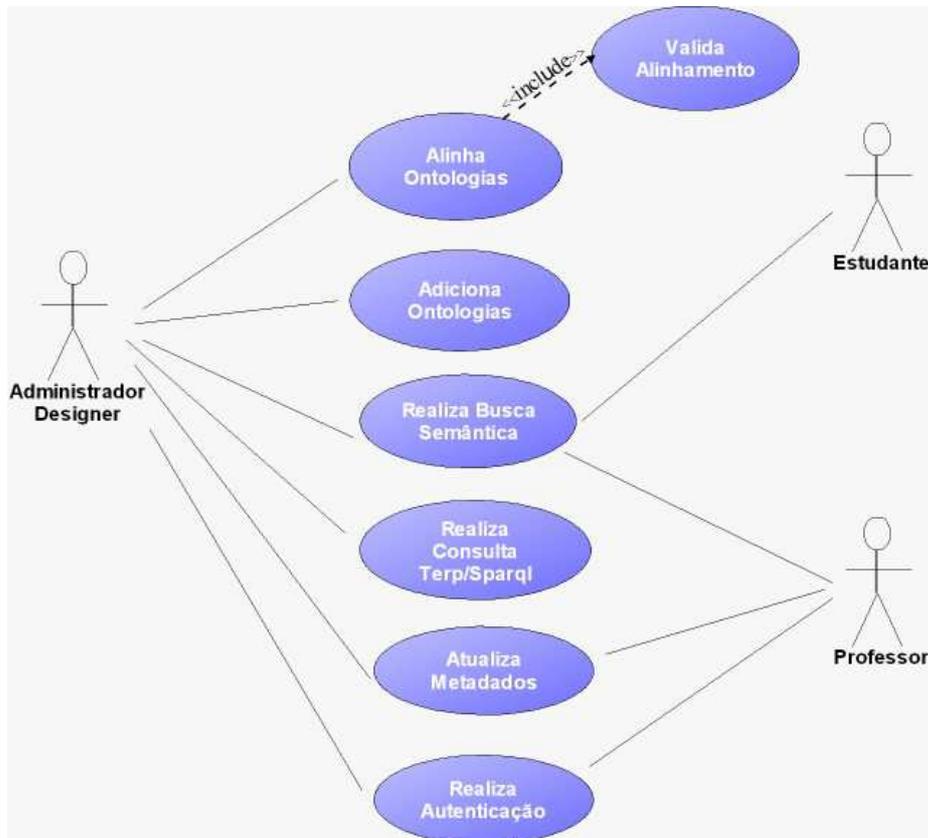


Fonte: Elaborado pelo autor.

### 5.3 Casos de uso do sistema *MSSearch*

O diagrama de caso de uso apresentado na Figura 14 ilustra as ações que o usuário pode executar no sistema. Todas as interações dos usuários ocorrem via interface Web.

**Figura 14: Diagrama de casos de uso do sistema**



Fonte: Elaborado pelo autor

O detalhamento dos casos de uso do sistema mostrado na Figura 14 é apresentado na Tabela 4.

**Tabela 4: Detalhamento do caso de uso**

Caso de Uso	Descrição
Realiza Autenticação	O usuário informa os dados para autenticação e o sistema retorna uma mensagem de sucesso ou insucesso.
Realiza Busca Semântica	O usuário solicita uma consulta informando uma palavra chave, e o sistema retorna uma lista de resultados, caso exista.
Realiza Busca Terp/Sparql	O usuário solicita uma consulta avançada informando um conjunto de instruções em formato Terp/Sparql e o sistema retorna uma lista de resultados, caso exista.
Adiciona Ontologias	O usuário adiciona 1 até $N$ ontologias e envia para o sistema, o mesmo grava estas ontologias em um sistema de arquivos.
Alinha Ontologias	O usuário solicita a execução do alinhamento, o sistema executa o alinhamento e posteriormente encaminha para validação.
Valida Alinhamento	O sistema executa a validação das ontologias alinhadas e envia uma

	mensagem de sucesso ou insucesso.
Atualiza Metadados	O usuário indica o metadado que deseja alterar com seu respectivo valor e o sistema grava no repositório semântico.

Fonte: Elaborado pelo Autor.

Os personagens que irão interagir com o sistema poderão ter os seguintes perfis, com suas respectivas atribuições:

- *Administrador/Designer de Conteúdos*: este tipo de usuário avalia e prepara as ontologias para serem alinhadas. Ficará a cargo do *Designer* a adição de novas ontologias no sistema. O mesmo terá condições de realizar consultas no formato *Terp/Sparql* diretamente no repositório de OA, bem com atualizar seus metadados;
- *Professor*: usuário que além de realizar busca semântica de OA, também pode executar atualizações dos seus metadados, mediante autenticação;
- *Estudante*: este perfil não demanda autenticação, pois este tipo de usuário somente poderá realizar buscas semânticas de OA;

#### 5.4 Sistema de alinhamento

O objetivo geral do sistema *MSSearch* é oferecer um repositório de metadados de objetos de aprendizagem com capacidade de busca semântica, em relação aos contextos educacionais. Para garantir que o sistema consiga realizar buscas semânticas com maior cobertura de conhecimento é necessário que o mesmo esteja preparado para tratar ontologias distintas. Propõe-se então, conforme destacado anteriormente, o uso de técnicas de alinhamento de ontologias.

O processo de alinhamento de ontologias pode exigir uma grande capacidade computacional para sua realização. Na verdade o caso geral pode se aproximar de uma complexidade  $O(n^2)$  em relação ao número de elementos das ontologias envolvidas (EHRIG, 2007).

Outra questão relacionada ao processo de alinhamento é o emprego de vocabulários léxicos gerais nas métricas de similaridade terminológicas, que são as técnicas mais disseminadas de comparação dos elementos das ontologias. O serviço aberto e público *WordNet* ([wordnet.princeton.edu](http://wordnet.princeton.edu)) possui uma ampla cobertura dos nomes, verbos, adjetivos e

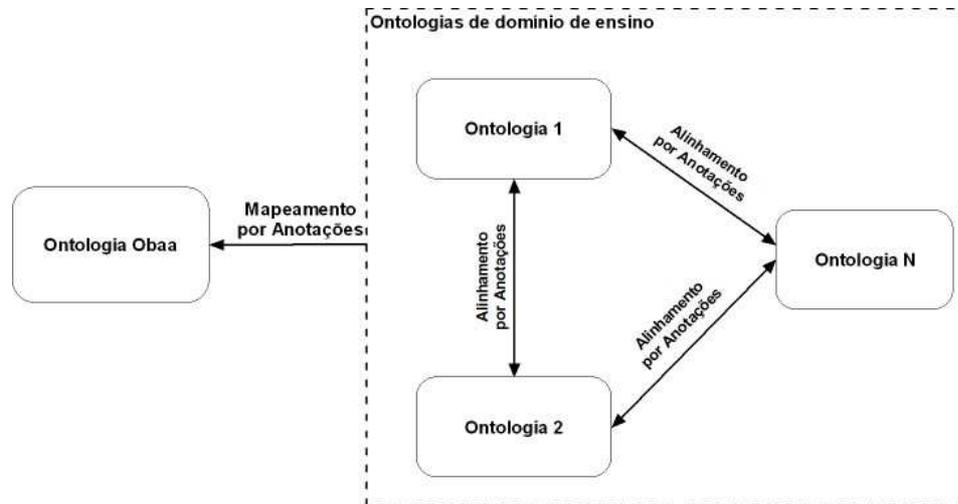
advérbios existentes na Língua Inglesa, trazendo bons resultados quando utilizado na função de similaridade. Porém, em termos da Língua Portuguesa não há um serviço aberto e público similar.

É importante salientar que o processo de alinhamento, apesar de necessário, não é suficiente para implementar o mecanismo de busca semântica. Para que este mecanismo fique completo também é necessário mapear os termos e conceitos das ontologias educacionais previamente alinhadas nos metadados OBAA, tais como os representados na ontologia de metadados OBAA. Esse processo denominado de *mapeamento*, não pode ser caracterizado como um alinhamento entre ontologias, porque existe uma importante diferença conceitual entre as ontologias educacionais e a ontologia de metadados OBAA: ontologias educacionais tipicamente identificam os conhecimentos e conceitos sobre conteúdos educacionais, métodos de ensino e processos de aprendizagem, enquanto que uma ontologia de metadados tipicamente representa os tipos de dados e valores possíveis para esses metadados. Pode-se supor que os *valores* dos metadados educacionais, em uma ontologia de metadados, possam ser associados às *classes e relacionamentos* que representam conceitos educacionais, em uma ontologia de domínio educacional, porém tais informações estão em níveis conceituais distintos, impossibilitando o uso de técnicas de alinhamento. O processo de mapeamento será descrito com mais detalhes na seção que trata do funcionamento do mecanismo de busca (Seção 5.5). Porém nessa seção são descritas as anotações empregadas tanto para o alinhamento quanto para o mapeamento.

#### 5.4.1 Anotações para alinhamento e mapeamento

Um dos principais componentes do *MSSearch* é o agente que realiza o alinhamento de ontologias de domínios educacionais. Conforme comentado anteriormente, é importante destacar que o alinhamento somente ocorrerá entre ontologias de domínios educacionais, e a ontologia resultante desse alinhamento será mapeada com a ontologia de metadados OBAA. A Figura 15 mostra as características desse processo.

Figura 15: Alinhamento resultante e mapeamento da ontologia OBAA



Fonte: Elaborado pelo Autor

O principal requisito para possibilitar o alinhamento e o mapeamento de uma ontologia de domínios educacionais com a ontologia de metadados OBAA é a necessidade das entidades dessa ontologia, sejam elas classes ou relações, terem seus nomes e sinônimos explicitamente definidos através de anotações da ontologia. Esse nome será definido através de uma anotação de sinônimos (*alias*). O mecanismo de anotação de ontologias OWL permite que qualquer elemento de uma ontologia possa ter anotações cuja língua natural tenha sido explicitamente indicada.

Um exemplo de rotulação de uma ontologia de domínios educacionais é demonstrada na Figura 16, onde é possível observar que as anotações são definidas pela entidade OWL *AnnotationProperties* denominada *Alias*. Foi definido que todas as *AnnotationsValues* iniciadas pelo termo *Align* (ex: *Align.Conteudo*) deverão ser entendido pelo algoritmo de alinhamento automático como uma tradução para a classe *Conteudo\_Matemática\_Ensino\_Médio*. Já os valores iniciados pelo termo *Obaa* (Ex: *Obaa.General.itsKeywords*) é entendido pelo algoritmo de busca como uma mapeamento entre as ontologias alinhadas e a ontologia de metadados OBAA.

É importante salientar que o processo de anotação de ontologias de domínios de ensino, mesmo quando se feito manualmente, não é um processo tedioso: somente os níveis mais altos das hierarquias de classes (no caso do exemplo, somente a classe *Conteudo\_Matematica\_Ensino\_Medio*) precisam ser anotados. O processo de alinhamento do *MSSearch*, implementa um mecanismo de herança de anotações que garante que os demais

níveis das hierarquias de classes sejam automaticamente anotados com os termos “*Align.*” e “*Obaa.*” das anotações *Alias* das classes superiores.

Apesar deste trabalho descrever o uso de anotações semânticas para ontologias de domínios de ensino, o mesmo método pode ser utilizado para outros domínios do conhecimento.

**Figura 16: Exemplo de rotulação de ontologias de domínios educacionais**



Fonte: Elaborado pelo Autor

#### 5.4.2 Adaptação da API de alinhamento

Na implementação do *MSSearch* optou-se pelo uso da ferramenta denominada *AlignApi*. Esta API é usada ativamente pela comunidade OAEI, auxiliando na avaliação de outros algoritmos de alinhamento submetidos para a própria comunidade. A possibilidade de se estender esta API foi levada em consideração na sua escolha. Esta extensão se fez necessária pois em alguns casos a complexidade envolvida no processo de alinhamento tende a aumentar, dependendo do número de elementos existentes entre as ontologias que se deseja alinhar. Devido a esta possibilidade, foi realizado uma adaptação do algoritmo de similaridade, tendo como o objetivo principal evitar a comparação com todas as entidades, habilitando somente a comparação com os elementos previamente anotados. Na Figura 17 o algoritmo adaptado é demonstrado.

Figura 17: Adaptação realizada do algoritmo da *AlignApi*

```

FunçãoAlinhamentoAdaptada (O1.listEntAnotadas, O2.listEnt)
1. para cada Entidade E1 de O1.listEntAnotadas faça
2.     nomeEntidade <- E1.nome
3.     maxSimilaridade <- 0.0
4.     anotaçãoEscolhida <- nulo
5.     para cada Anotação A1 de E1.listaAnotações faça
6.         para cada Entidade E2 de O2.listEnt faça
7.             similaridade <- CalculaSim(A1.nome, E2.nome)
8.             se similaridade > maxSimilaridade então
9.                 maxSimilaridade <- similaridade
10.                entidadeEscolhida <- E2.nome
11.                anotaçãoEscolhida <- A1.nome
12.            fim se
13.        fim para
14.    fim para
15. fim para
16. Retorne entidadeEscolhida, anotaçãoEscolhida, maxSimilaridade

```

Fonte: Elaborado pelo Autor

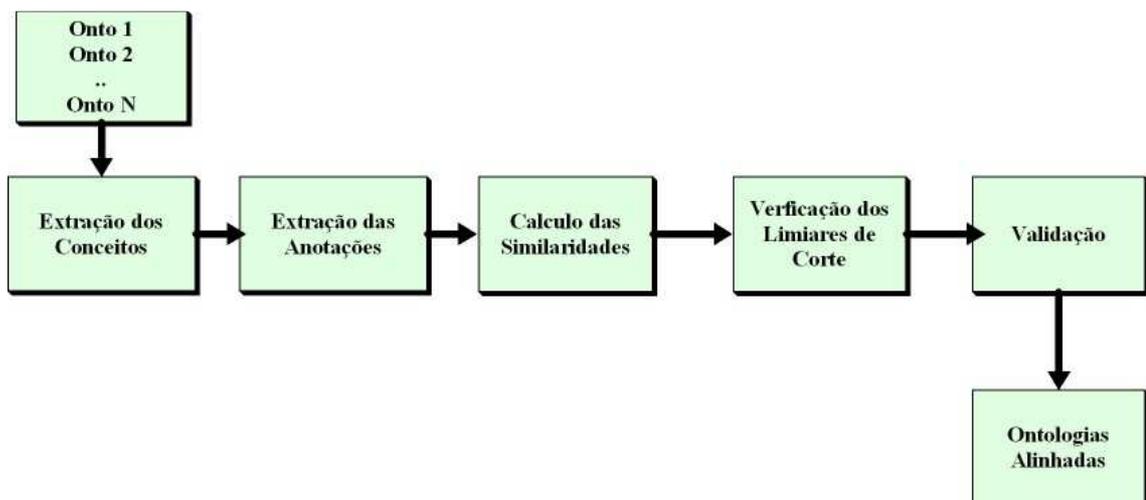
#### 5.4.3 Processo de alinhamento

A Figura 18 especifica as etapas do processo de alinhamento implementado pelo agente de alinhamento. Cada etapa é descrita a seguir:

- Entrada: o processo de alinhamento inicia-se com a passagem das ontologias como parâmetro para as funções de alinhamento da *AlignApi*;
- Extração dos Conceitos (Classes): nesta etapa os nomes (*labels*) são extraídos e após é montada uma lista destes nomes;

- Extração das Anotações: como visto no algoritmo demonstrado na Figura 17, de cada classe é extraída uma lista das suas respectivas anotações (rotuladas com o nome de *AnnotationProperty:Alias*);
- Calculo da Similaridade: nesta etapa são utilizados algoritmos de comparação de *strings* para calcular a similaridade entre as entidades da Ontologia  $O_1$  e  $O_2$ . Para este trabalho foi utilizado o algoritmo de *Monger-Elkan* (ver Seção 3.5.1);
- Verificação dos Limiares de Corte: é possível definir limiares de aceitação do alinhamento. O grau de semelhança de um alinhamento se dará na faixa de  $[0,1]$ . Valores resultantes fora da faixa pré-configurada serão automaticamente desconsiderados;
- Validação: a ontologia resultante deverá ser validada, sendo que sua validação se dará na sua estrutura, na sua taxonomia e nos seus relacionamentos. Esta tarefa ficará a cargo dos *reasoners* (Pellet para este trabalho);
- Ontologias Alinhadas: após a validação da nova ontologia, um arquivo é gerado no formato OWL, indicando as suas respectivas correspondências;

**Figura 18: Processo de Alinhamento das Ontologias de Domínios Educacionais**



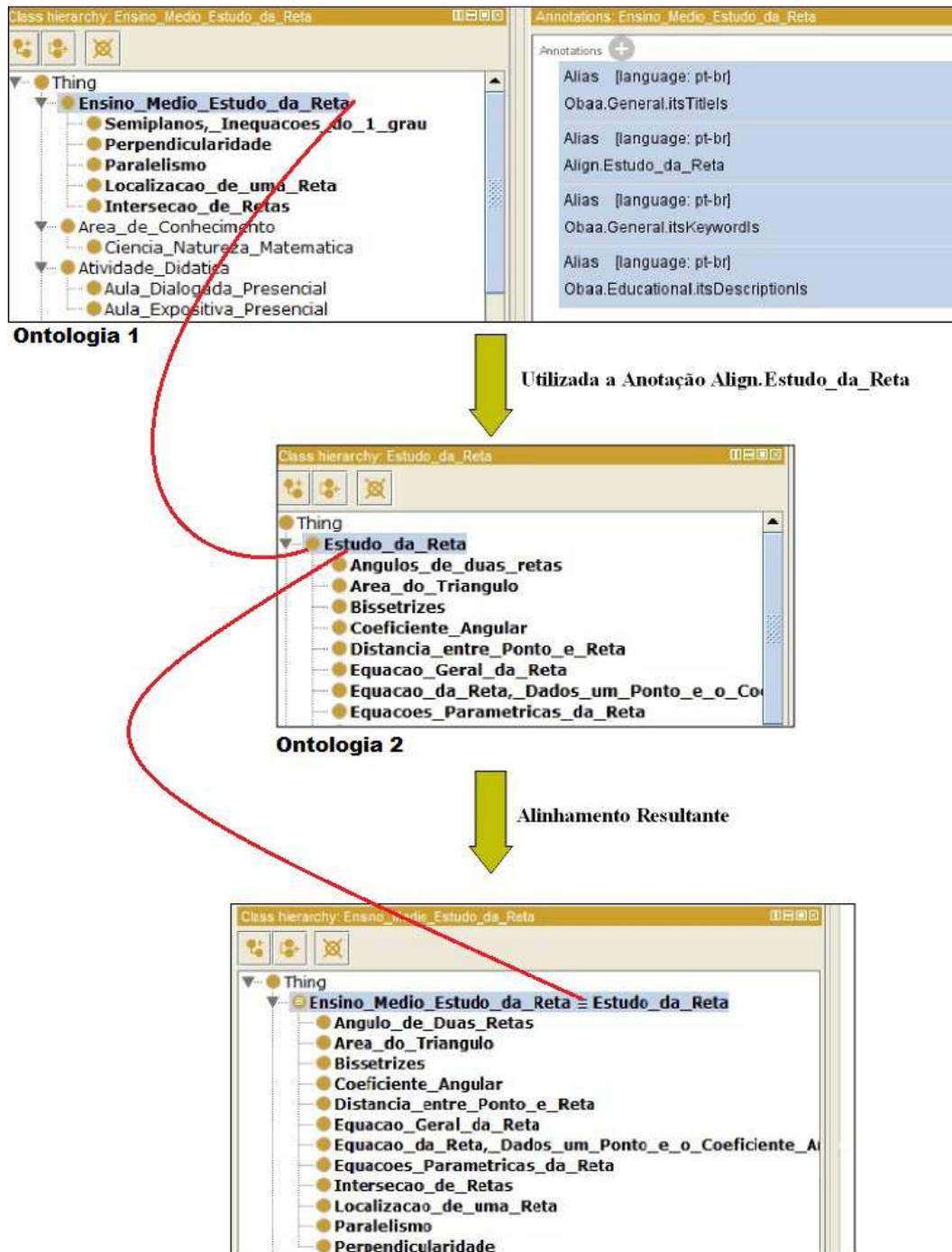
Fonte: Elaborado pelo autor.

Para um melhor entendimento, foi configurado duas ontologias para realização de um alinhamento. Uma ontologia(*Ontologia 1*) que representa o estudo da reta da disciplina de matemática do ensino médio(*Ensino\_Medio\_Estudo\_da\_Reta*) e outra ontologia(*Ontologia*

2) que também trata do estudo da reta (*Estudo\_da\_Reta*). Ambas são provenientes de duas fontes distintas, onde os nomes das suas classes (*Labels*) foram especificados de forma diferente. Tomando conhecimento dessas ontologias, o *Designer de Conteúdos* irá prepará-las, informando a anotação que facilitará no processo de alinhamento. Neste exemplo o *Designer* anotou a classe "*Ensino\_Medio\_Estudo\_da\_Reta*" com a anotação "*Align.Estudo\_da\_Reta*".

Um exemplo de alinhamento pode ser visto na Figura 19. É possível observar que a classe "*Ensino\_Medio\_Estudo\_da\_Reta*" da *Ontologia 1* é correspondente com a classe "*Estudo\_da\_Reta*" da *Ontologia 2*, ou seja, através desta correspondência, as duas ontologias mencionadas foram alinhadas. Na Figura 20, um trecho do arquivo resultante é mostrado, observa-se que as classes foram mapeadas como *EquivalentClass* em OWL. Ainda no mesmo arquivo, a Tag XML *Measure* indica que o grau de similaridade entre as classes correspondentes foi de 0.89 (89%).

Figura 19: Exemplo de alinhamento



Fonte: Elaborado pelo autor

**Figura 20: Trecho OWL do Arquivo resultante do alinhamento**

```
<owl:Class rdf:about="http://obaa.unisinos.br/Ontologia1#Ensino_Medio_Estudo_da_Reta">
  <owl:equivalentClass rdf:resource="http://obaa.unisinos.br/Ontologia2#Estudo_da_Reta"/>
  <measure rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.89</measure>
</owl:Class>
```

Fonte: Elaborado pelo autor

## 5.5 Mecanismo de busca semântica

O sistema de busca semântica do *MSSearch* se propõe a recuperar objetos de aprendizagem estruturados pelo padrão OBAA, localizado em um repositório semântico com suporte OWL nativo, mediante informação de palavras-chaves de livre escolha para realização da pesquisa.

Este sistema apoia-se no uso de técnicas de alinhamento de ontologias, agentes de *software* e mecanismos de inferências para recuperação da informação semanticamente anotada. A seguir será descrito o funcionamento do mecanismo de busca semântica implementado, conforme a sequência numerada mostrada na Figura 21:

- Passo 1: quando informada a palavra chave, é submetida ao sistema de busca, porém antes é aplicado um algoritmo para remoção de sufixos e de *Stop Words*<sup>16</sup> (utilizado neste trabalho o algoritmo de Orengo<sup>17</sup>);
- Passo 2: após a normalização da palavra chave, é enviado uma consulta a base de ontologias alinhadas, procurando através de inferência extrair os termos semanticamente relacionados. Também por inferência, é extraído as anotações do tipo *Alias*, que inicie por "Obaa." para que então o sistema mapeie com a ontologia OBAA e execute a consulta nos metadados armazenados. Importante salientar que se nenhuma anotação para mapeamento for encontrado, o sistema por *Default* utilizará a anotação "*Obaa.General.itsKeywordIs*", ou seja, se por

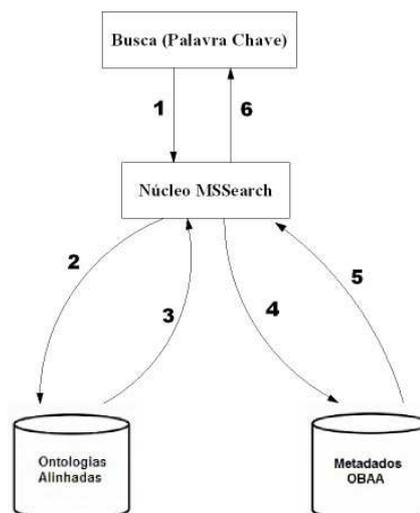
<sup>16</sup> *Stop Words* são palavras consideradas irrelevantes, tais como artigos, pronomes, interjeições, advérbios, preposições, etc.

<sup>17</sup> Proposto por Orengo e Huyck (2001), este algoritmo faz a redução do plural, dos advérbios, do diminutivo/aumentativo, nominal e verbal de palavras.

acaso o *Designer de Conteúdos* esquecer de mapear a ontologia OBAA, o sistema sempre vai procurar informações neste metadado;

- Passo 3: com base na lista de termos extraídos das ontologias alinhadas, uma consulta semântica no formato Terp/Sparql é construída, passando como parâmetro os termos inferidos e as anotações que indicarão em qual metadado consultar. Um exemplo de consulta Terp/Sparql simulando este processo pode ser visto na Figura 22;
- Passo 4: neste momento o sistema envia a consulta(construída no passo anterior) e extrai o conjunto de triplas RDF do banco de dados semântico;
- Passo 5: nesta etapa, o sistema organiza as informações para apresentar ao requisitante. Primeiramente essas informações serão ordenadas de acordo com a sua relevância<sup>18</sup>, de forma decrescente e logo após será aplicado um filtro, eliminando os resultados menos relevantes. Este filtro o próprio usuário indica no momento da pesquisa;
- Passo 6: formatação e apresentação do conjunto de resultados. Nesse momento o sistema habilita para o usuário uma lista de termos inferidos, extraída da base de ontologias alinhadas, de modo a sugerir uma nova pesquisa, bem como a apresentação dos resultados em uma página HTML devidamente formatada;

**Figura 21: Funcionamento do sistema de busca semântica**



Fonte: Elaborado pelo autor

**Figura 22: Exemplo de consulta Sparql/Terp**

<sup>18</sup> A relevância utilizada no sistema leva em conta o número de ocorrências encontradas no texto dos metadados consultados, ou seja, quanto maior a ocorrência maior será relevância.

```

SELECT distinct ?lobj ?key ?desc ?loc ?title ?format
WHERE {
  ?lobj a obaa:LearningObject .
  ?lobj obaa:hasMetadata ?mdata.
  ?mdata obaa:itsKeywordIs ?key.
  ?lobj obaa:hasMetadata ?mdtit. ?mdtit obaa:itsTitleIs ?title .
  ?lobj obaa:hasMetadata ?mddesc. ?mddesc obaa:itsDescriptionIs ?desc .
  ?lobj obaa:hasMetadata ?mdloc. ?mdloc obaa:itsLocationIs ?loc .
  ?lobj obaa:hasMetadata ?mdformat. ?mdformat obaa:itsFormatIs ?format.

  FILTER (regex (?key,'Bissetrizes','i') || regex (?desc,'Bissetrizes','i') )
}

```

Fonte: Elaborado pelo autor

## 5.6 Repositório semântico

No sistema *MSSearch* também é implementado um *repositório semântico* que será utilizado para armazenar localmente os metadados dos OAs. Esse repositório também será uma contribuição importante para a infraestrutura MILOS, se constituindo no núcleo do Sistema de Gerência dessa infraestrutura.

Um repositório semântico deve fornecer todos os serviços de armazenamento e recuperação de metadados OBAA, necessário aos demais agentes do sistema *MSSearch*. O aspecto semântico desse repositório é devido ao suporte nativo ao formato OWL, tanto para fins de consulta, através da linguagem de consulta SPARQL, quanto para fins de armazenamento em triplas RDF. No caso do *MSSearch*, também se está supondo que o repositório semântico também suporte mecanismos de inferência OWL.

O repositório semântico do sistema *MSSearch* poderá ser acessado através de três tipos de interface: interface Web, para usuários finais, e interfaces de serviços RESTful e mensagens FIPA, para interoperação com outras aplicações e agentes.

Em todas essas interfaces a linguagem a ser utilizada para consultas e atualizações é SPARQL, podendo ser opcionalmente estendida com construções da sintaxe Manchester tal como definido no formato TERP/SPARQL (SIRIN et al., 2010). Todos os metadados disponíveis no repositório semântico devem estar representados de acordo com a ontologia de metadados OBAA (GLUZ e VICARI, 2012).

Uma exemplo de busca inicial, muito simples, que permite descobrir quais objetos estão contidos no repositório poderia ser implementada pela seguinte consulta SPARQL(Figura 23):

**Figura 23: Consulta Sparql para recuperação de OAs**

```
SELECT ?lobj
WHERE {?lobj a LearningObject}
```

Fonte: Elaborado pelo Autor.

Esta consulta irá identificar todos os objetos pertencentes a classe *LearningObject* contidos atualmente na base de testes. A Tabela 5, mostra a resposta para tal consulta.

**Tabela 5: OAs retornados da consulta Sparql**

lobj
lo001
lo002
lo003
lo004

Fonte: Elaborado pelo Autor.

Neste caso se está assumindo que somente objetos de teste serão contidos nesta base. Porém, caso se queira especificamente localizar os objetos pertencentes a um determinado catálogo, como por exemplo “OBAA Test Objects”, então deve-se buscar por OA cuja informação de catálogo, no metadado de identificação, seja “OBAA Test Objects”. Esta consulta(Figura 24) pode ser implementada através da seguinte instrução TERP/SPARQL:

**Figura 24: Consulta TERP/SPARQL de OAs de um determinado catálogo**

```
SELECT ?lobj
WHERE {?lobj a
      (LearningObject and
      (hasMetadata some (hasIdentifier some
      (itsCatalogIs value "OBAA Test Objects")))).
}
```

Fonte: Elaborado pelo Autor.

Nesse caso a variável ?lobj deve ser substituída apenas por indivíduos da classe *LearningObject*, mas que tenham algum metadado (indicados por *hasMetadata some*) de identificação (indicado por *hasRequirement some*), cuja informação de catálogo (indicado pelo atributo *itsCatalogIs*) seja “OBAA Test Objects”. Atualmente a resposta desta consulta será a mesma da consulta anterior, mas caso outros catálogos de objetos venham a ser armazenados na base, então os resultados poderão diferir no futuro.

Outras informações também podem ser obtidas. A seguinte consulta obtém os títulos e localização de todos os OA que são suportados pela sistema operacional UNIX.

**Figura 25: Consulta Terp/Sparql que retorna OA suportados pelo UNIX**

```
SELECT ?title ?loc
WHERE {?lobj a
  (hasMetadata some
    (hasRequirement some
      (hasOrComposite some
        (itsTechNameIs value "unix")))).
  ?lobj hasMetadata ?mdtit, ?mdloc .
  ?mdtit itsTitleIs ?title .
  ?mdloc itsLocationIs ?loc.
}
```

Fonte: Elaborado pelo Autor.

A Tabela 6 a seguir mostra um possível resultado para esta consulta:

**Tabela 6: OA suportados por UNIX**

Title	loc
"Objeto de Aprendizagem de Teste da Ontologia OBAA. Objeto 1: teste de metadados gerais e técnicos"@pt-br	http://obaa.unisinos.br/obaa-test-objects/obj0001.html

Fonte: Elaborado pelo Autor.

A seguinte consulta (Figura 26) apresenta um caso um pouco mais geral, onde se busca obter os títulos e localização dos OA que funcionam tanto em UNIX quanto no SBTVD:

**Figura 26: Consulta de OA que funcionam tanto em UNIX quanto SBTVD**

```
SELECT ?title ?loc
WHERE {
  ?lobj a
    (hasMetadata some (hasRequirement some
      (hasOrComposite some (itsTechNameIs value "unix"))))
    or
    (hasMetadata some
      (itsSupportedPlatformsIs value "sbtvd")).
  ?lobj hasMetadata ?mdtit, ?mdloc .
  ?mdtit itsTitleIs ?title .
  ?mdloc itsLocationIs ?loc.
}
```

Fonte: Elaborado pelo Autor.

Possíveis resultados dessa consulta são apresentados na Tabela 7.

**Tabela 7: OA suportados em UNIX e SBTVD**

Title	loc
"Objeto de Aprendizagem de Teste da Ontologia OBAA. Objeto 1: teste de metadados gerais e técnicos"@pt-br	http://obaa.unisinos.br/obaa-test-objects/obj0001.html
"Objeto de Aprendizagem de Teste da Ontologia OBAA. Objeto 2: teste de metadados para plataforma de TV Digital"@pt-br	http://obaa.unisinos.br/obaa-test-objects/obj0002.pdf

Fonte: Elaborado pelo Autor.

Sabendo da existência de OA com suporte para TV Digital, pode-se verificar se existem conteúdos equivalentes ao conteúdo base, definido pelo metadado *Location*, para a plataforma de TV Digital. A seguinte consulta (Figura 27) obtém os endereços do conteúdo base e do conteúdo equivalente para TV Digital, caso estiver disponível em algum OA.

**Figura 27: Consulta que mostra a localização do conteúdo base e equivalente para SBTVD**

```
SELECT ?locbase ?locsbtd
WHERE {
  ?lobj a (hasMetadata some
    (itsSupportedPlatformsIs value "sbtvd")) .
  ?psbtd a (PlatformSpecificFeatures
    and itsPlatformTypeIs value "sbtvd") .
  ?lobj hasMetadata ?mdbaseloc , ?mdplat .
  ?mdplat hasPlatformSpecificFeatures ?psbtd .
  ?mdbaseloc itsLocationIs ?locbase .
  ?psbtd itsLocationIs ?locsbtd
}
```

Fonte: Elaborado pelo Autor.

Os resultados dessa consulta são apresentados na Tabela 8, mostrando que o OA cujo conteúdo é formado pelo arquivo *obj0002.pdf* em formato PDF, que não pode ser visualizado por uma *set-top box* compatível com o SBTVD, tem um conteúdo equivalente no arquivo *obj0002.jpg*, no formato de imagem JPEG, compatível com o SBTVD.

**Tabela 8: Localização do conteúdo base e equivalente para SBTVD**

Locbase	locsbtd
http://obaa.unisinos.br/obaa-test-objects/obj0002.pdf	http://obaa.unisinos.br/obaa-test-objects/obj0002.jpg

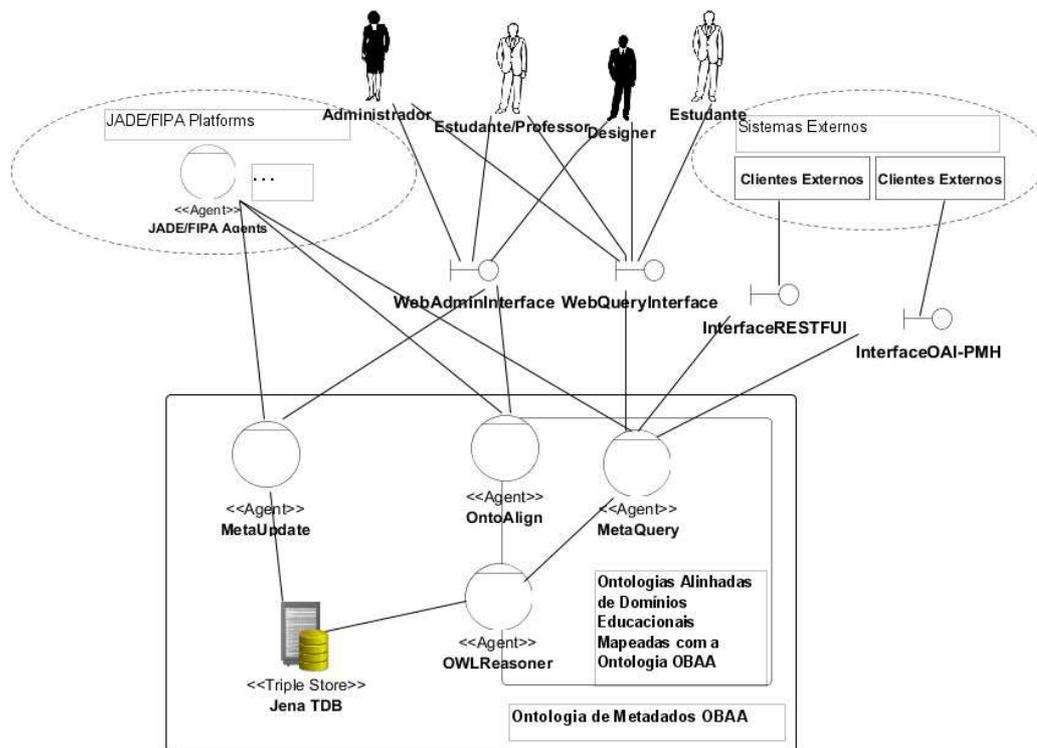
Fonte: Elaborado pelo Autor.

## 5.7 Agentes do sistema *MSSearch*

Os agentes que compõe o sistema *MSSearch* seguem o padrão proposto pela plataforma Jade/FIPA. O sistema é constituído por quatro agentes que têm papéis distintos dentro da arquitetura. Um visão geral da disposição dos agentes e sua relação com o sistema, é apresentada na Figura 28. Cada agente é nomeado conforme sua atribuição dentro sistema, sendo eles:

- *MetaQuery*: agente de consultas à base de metadados, este agente recebe requisições provenientes da interface de serviços (*InterfaceRESTFul*) e da interface de Web (*WebQueryInterface* ou *WebAdminInterface*) para realização de consultas, comunica-se com o agente *OWLReasoner*;
- *MetaUpdate*: agente de atualização da base de metadados, que recebe requisições da interface *WebAdminInterface*;
- *OntoAlign*: agente de alinhamento de ontologias, que também recebe e processa requisições da interface *WebAdminInterface*, comunica-se com o agente *OWLReasoner*;
- *OWLReasoner*: agente que encapsula toda a capacidade de raciocínio para realização de inferências nas ontologias alinhadas de domínios educacionais e na ontologia OBAA. No caso do *MSSearch* se optou, por questões de desempenho e flexibilidade, pela utilização do reasoner *Pellet* para implementar os processos de inferência neste agente;

**Figura 28: Disposição dos Agentes na arquitetura do sistema *MSSearch***



Fonte: Elaborado pelo autor

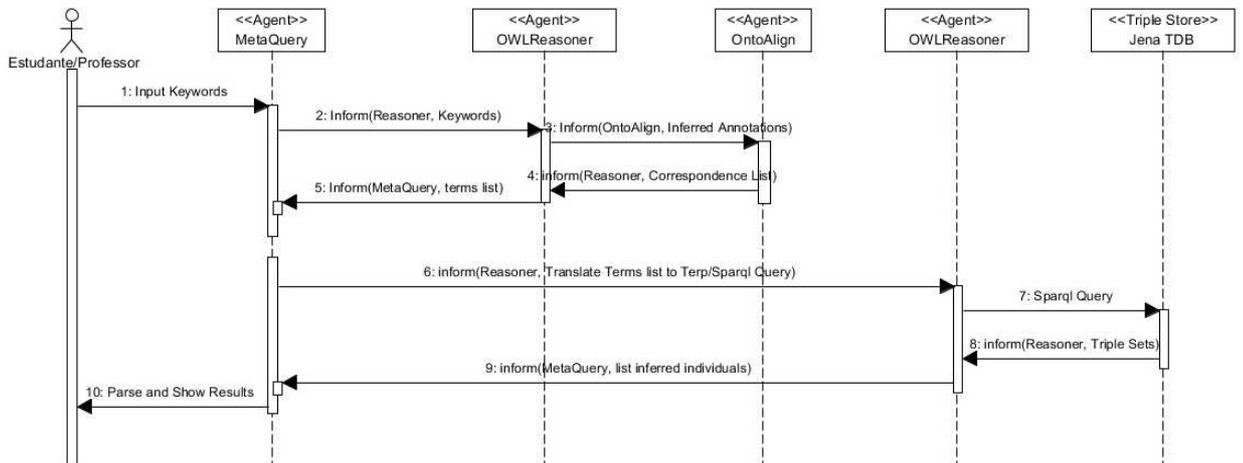
O mecanismo de armazenamento de triplas RDF fornecido pela biblioteca Jena (a camada de persistência TDB da Jena) é utilizado pelos agentes *MetaUpdate* e *OWLReasoner* para armazenar e recuperar os metadados OBAA. O subsistema formado por estes agentes e pela Jena TDB implementa o repositório semântico do *MSSearch*.

Para demonstrar as interações entre os agentes do sistema *MSSearch*, serão apresentados diagramas de sequência para especificar a comunicação.

A Figura 29 traz o diagrama de sequência para realização da busca semântica, onde pode ser visto o usuário, aqui representando por um Estudante ou Professor, interagindo com o sistema, informando palavras chaves (*Keywords*) para execução de uma pesquisa. No momento que a consulta é acionada o agente *MetaQuery* percebe que houve uma requisição e inicia o processamento. Este agente faz a normalização da palavra chave, e encaminha para o agente *OWLReasoner* executar uma consulta na base de ontologias alinhadas. Com base nesta lista de termos inferidos o mesmo envia para o agente *OntoAlign* uma outra lista com as anotações desses termos para que sejam encontrados as correspondências, e com isso decidir qual a ontologia será utilizada para realização da pesquisa. Com a ontologia escolhida, o agente *OWLReasoner* encaminha para o agente *MetaQuery* a lista de termos para realização da pesquisa, sendo que este por sua vez construirá a consulta no formato *Terp/Sparql* para que

seja consultado no repositório semântico onde estão armazenados os metadados dos objetos de aprendizagem.

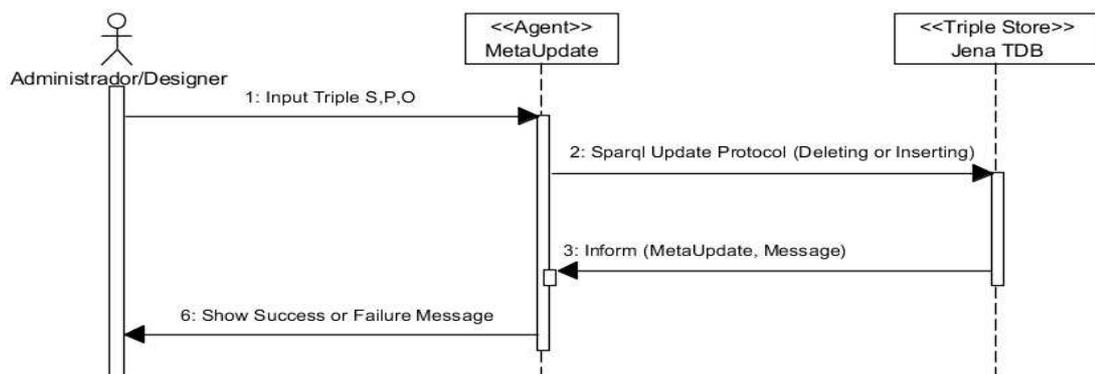
**Figura 29: Diagrama de sequência da busca semântica**



Fonte: Elaborado pelo autor

Já na Figura 30 pode ser visto o diagrama de sequência que representa a execução da atualização dos metadados dos OA. Usuários com perfis de Administrador ou de Designer de Conteúdos poderão realizar este tipo de procedimento. Para que seja possível executar esta atualização, o usuário precisa informar a tripla RDF(Sujeito, Predicado, Objeto) que deseja alterar. O Agente *MetaUpdate*, percebendo que houve este tipo de solicitação, formata um conjunto de instruções para a realização da atualização (*Sparql Update*) no banco de dados semântico. Uma mensagem de sucesso ou insucesso é encaminhado para o agente *MetaUpdate*, onde este por sua vez apresenta ao usuário.

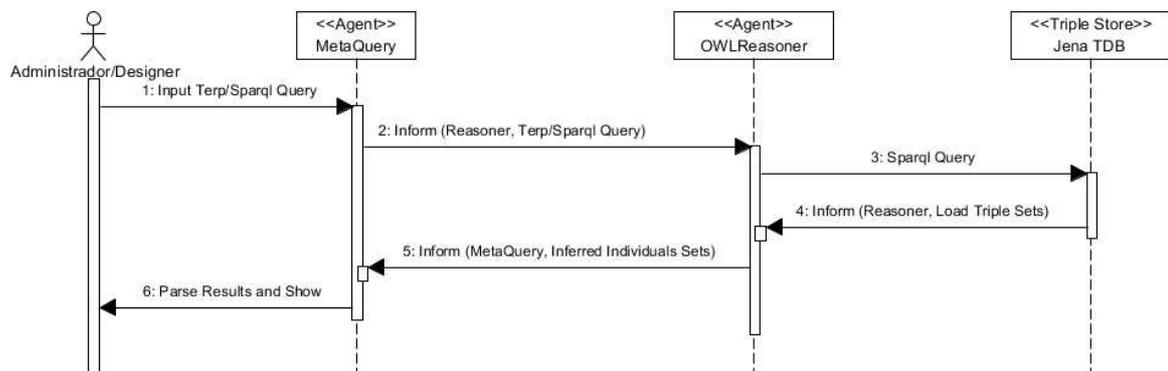
**Figura 30: Diagrama de sequência da atualização de metadados**



Fonte: Elaborado pelo autor

Como visto anteriormente, o sistema habilita o usuário, desde que este seja pertencente ao perfil de Administrador ou Designer, a executar consultas no formato Terp/Sparql diretamente no repositório semântico. Este procedimento também é apoiado por agentes de software, e está representado no diagrama de sequência da Figura 31. Como pode ser visto, o usuário formata a consulta no formato Terp/Sparql e submete ao sistema. Nesse momento o agente *MetaQuery* percebe que houve uma solicitação do usuário e envia estas instruções para o agente *OWLReasoner* executar no repositório semântico, utilizando mecanismos de inferências. Este agente também faz um *parser* da consulta que está no formato Terp para Sparql (aqui se faz o uso da API Pellet). Com a informação recuperada, o agente *OWLReasoner* envia para o agente *MetaQuery*, e este por sua vez formatará e apresentará ao usuário requisitante.

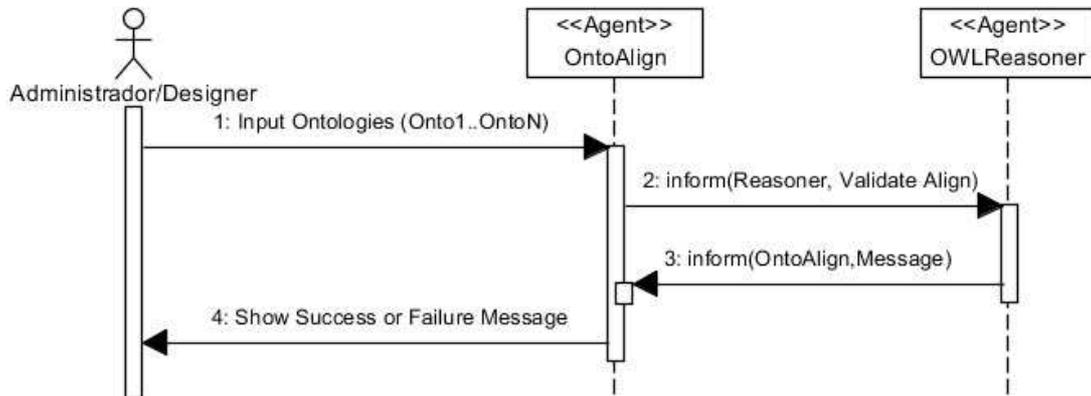
**Figura 31: Diagrama de sequência da consulta Terp/Sparql**



Fonte: Elaborado pelo autor

A Figura 32 apresenta o diagrama de sequência do processo de alinhamento das ontologias. Para iniciar este processo, primeiramente é necessário adicionar as ontologias que se deseja alinhar, tarefa esta executada por um Designer ou Administrador. Quando o usuário executa o processo de alinhamento, entra em ação o agente *OntoAlign*. A API de alinhamento (*AlignAPI*) está encapsulada dentro deste agente e fornece todos mecanismos necessários para encontrar as correspondências entre as ontologias informadas. Logo após este processo, o agente *OntoAlign* encaminha para o agente *OWLReasoner* o alinhamento resultante para que o mesmo faça uma validação deste alinhamento.

**Figura 32: Diagrama de sequência do alinhamento**



Fonte: Elaborado pelo autor

## 5.8 Protótipo do sistema

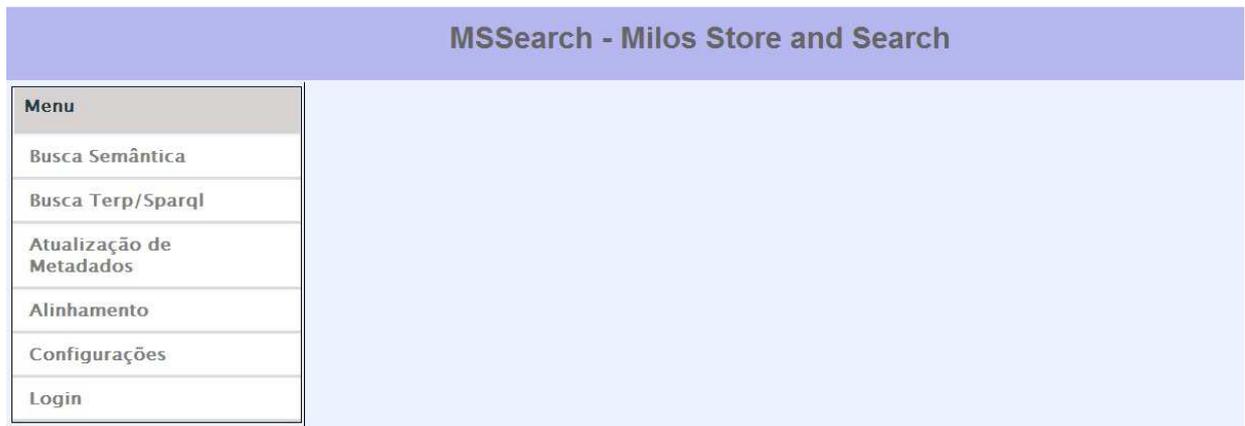
O sistema desenvolvido caracteriza-se por apresentar quatro camadas bem definidas (camada de interface, de agentes, manipulação de ontologias e de armazenamento), conforme arquitetura demonstrada na Seção 5.1. As subseções a seguir apresentam os detalhes de funcionamento de cada camada do protótipo do sistema *MSSearch*.

### 5.8.1 Interface de interação

A interface Web possibilita que o usuário interaja com o sistema independentemente do sistema operacional ou do local que ele se encontra, ou seja, o programa pode ser aberto em qualquer navegador Web disponível, com suporte ao padrão HTML.

No seu desenvolvimento, optou-se pela utilização do framework JSF, juntamente com a biblioteca de componentes (*RichFaces*). Esta escolha se deve principalmente pelo fato de todo ambiente de programação deste trabalho ser em Linguagem Java e por possuir uma vasta documentação, auxiliando no seu desenvolvimento. Conforme demonstrado na Figura 33, no momento que o usuário faz seu primeiro acesso, é apresentado uma tela com seis opções organizadas em um formato de menu.

Figura 33: Menu principal do sistema



Fonte: Elaborado pelo autor

A primeira opção do menu denominada "Busca Semântica", dentro do menu principal, habilita o usuário a realizar buscas de objetos de aprendizagem com suporte a inferência (Figura 34). A sua utilização não exige que o usuário tenha conhecimentos avançados, é preciso somente que o mesmo informe a palavra chave para pesquisa e acione o botão "Pesquisar". O resultado da pesquisa retorna uma lista de OA, com informações sobre seus metadados, estruturados pelo padrão OBAA. O sistema possui ainda opções de selecionar os resultados mais relevantes e também de limitar a quantidade de resultados a serem apresentados. A ferramenta ainda habilita uma lista de termos inferidos, logo após a submissão da primeira consulta. Esses termos serão extraídos com auxílio do mecanismo de inferência (*Reasoner*), configurado no sistema.

Figura 34: Busca semântica

The screenshot shows a web interface for semantic search. At the top, there is a search bar with the text 'cones' entered. To the right of the search bar, there is a dropdown menu set to 'Até 20 Resultados' and a 'Pesquisar' button. Below the search bar, there is a 'Termos Inferidos' dropdown menu. The main content area is titled 'Resultados' and displays two search results. Each result includes an OA (Open Access) link, a title, an objective, a description, and keywords. The first result is titled 'Volume, cones e cilindros - Parte I' and the second is 'Volumé, cones e cilindros - Parte II'. Both results are in Portuguese and have an audio format.

Fonte: Elaborado pelo autor

Dependendo do tipo de ação que o usuário deseja executar, é necessário que o mesmo se autentique no sistema (Figura 35), informando seu usuário e senha. O sistema irá habilitar seu acesso de acordo com o seu perfil. Atualmente esse cadastro é realizado manualmente, na forma de um pré-cadastro.

Figura 35: Tela de login

The screenshot shows the login screen of the 'MSSearch - Milos Store and Search' system. On the left, there is a 'Menu' sidebar with options: 'Busca Semântica', 'Busca Terp/Sparql', 'Atualização de Metadados', 'Alinhamento', 'Configurações', and 'Login'. The main area is titled 'Login - Versão: 4.6' and contains a login form with fields for 'Usuário' and 'Senha', and an 'Entrar' button.

Fonte: Elaborado pelo autor.

Selecionando a opção "Busca no Formato Terp/Sparql", o usuário poderá executar consultas no formato TERP ou SPARQL diretamente sobre os metadados armazenados no repositório semântico. O mecanismo de inferência já está habilitado neste sistema. Importante salientar que neste módulo é necessário que o usuário tenha conhecimento sobre a estrutura da

ontologia OBAA (seu vocabulário), e também consiga formular sentenças para realização da consulta. Este módulo é apresentado na Figura 36.

**Figura 36: Tela de Consulta para os formatos Terp/Sparql**

**MSSearch - Milos Store and Search**

Busca Semântica - Formato Terp/Sparql

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX obaa: <http://obaa.unisinos.br/obaa22.owl#>

SELECT distinct ?lobj ?title ?loc
WHERE {
  ?lobj a (obaa:hasMetadata some
    (obaa:hasRequirement some
      (obaa:hasOrComposite some
        ( obaa:hasTechName some
          (obaa:itsValueIs value "any")))).
  ?lobj obaa:hasMetadata ?mdtit,?mdloc,?mddesc,?tech,?mdd .
  ?mdtit obaa:itsTitles ?title . ?mdloc obaa:itsLocations ?loc.
  ?mddesc obaa:itsKeywords ?key. ?mdd obaa:itsDescriptions ?description .
  ?tech obaa:hasRequirement ?tn. ?tn obaa:hasOrComposite ?oc.
  ?oc obaa:hasTechName ?tnn. ?tnn obaa:itsValueIs ?value
}

```

Resultados

lobj	title	loc
lo84	Curso UCA - Módulo 1 (Introdução à Informática)	http://repositorio.portalobaa.org/handle/obaa/84
lo86	Curso UCA - Módulo 3 (Aplicativos KOffice)	http://repositorio.portalobaa.org/handle/obaa/86
lo80	Vida de Bicho	http://repositorio.portalobaa.org/handle/obaa/80
lo74	Curso UCA	http://repositorio.portalobaa.org/handle/obaa/74

Fonte: Elaborado pelo autor.

Já na Figura 37, quando o usuário seleciona a opção “Atualização de Metadados”, uma tela é apresentada para que sejam informados os dados para realização das atualizações.

**Figura 37: Tela de atualização de metadados**

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>  
 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>  
 PREFIX owl: <http://www.w3.org/2002/07/owl#>  
 PREFIX obaa: <http://obaa.unisinos.br/obaa22.owl#>

Informe os Parâmetros para atualização

S

P

O (Antigo)

O (Novo)

Consultar Atualizar

Consultar

Fonte: Elaborado pelo autor

Os parâmetros para atualização são informados no formato de triplas RDF (**S**ujeito, **P**redicado, **O**bjeto). É possível visualizar neste exemplo que será realizada uma atualização do título do (*DataProperty itsTitleIs*) “md001”, com o nome “Novo Título Metadado 1” .

A opção "Alinhamento" do menu, apresenta uma tela (Figura 38) ao usuário para que o mesmo consiga adicionar ontologias e executar o alinhamento. Inicialmente o operador do sistema deverá clicar no botão "Adicionar Ontologia" e após acionar o botão "Enviar". Nesse momento a nova ontologia é enviada para o servidor e entra na lista de ontologias que serão alinhadas. Havendo mais de uma ontologia nessa lista, o botão "Alinhar" é habilitado ao usuário, bastando a ele executar. Com isso, o processo de alinhamento será disparado automaticamente. Nessa mesma tela, logo abaixo, pode ser vista o resultado da validação da ontologia alinhada, executada pelo *Reasoner Pellet*. Caso haja falha na validação, não será gerado o arquivo resultante "AlignedOntology" e uma mensagem é enviada ao usuário.

Figura 38: Tela alinhamento

Adicionar nova Ontologia de Conteudos Educacionais

+ Adicionar Ontologia

Arquivo	Opção
<a href="#">MilosEduLogica-v10.owl</a>	<a href="#">Excluir</a>
<a href="#">MilosEduMat-v10.owl</a>	<a href="#">Excluir</a>

[AlignedOntology.owl](#)

Validação da Ontologia Alinhada - Pellet Reasoner

Validação Ok..  
AlignedOntology.owl criado com sucesso

Fonte: Elaborado pelo autor

## 6 EXPERIMENTOS E VALIDAÇÕES

Pretende-se neste capítulo descrever os experimentos realizados, bem como apresentar os resultados obtidos com a validação do sistema *MSSearch*. A Seção 6.1 apresenta os resultados dos testes de desempenho executados no repositório semântico modelado para este sistema. Na Seção 6.2, um experimento com uso de alinhamento de ontologias é demonstrado. Por fim, os resultados obtidos com a validação do sistema por especialistas pode ser vista na Seção 6.3.

### 6.1 Avaliação do desempenho do repositório semântico

O repositório semântico utilizado neste projeto, serve como base de dados para o armazenamento dos metadados dos OA padrão OBAA. Como visto na Seção 5.6, este repositório constitui-se como um dos principais componentes da infraestrutura MILOS. Para tanto, se fez necessário avaliar o desempenho do repositório utilizado, e esta avaliação se deu na forma de inserções de triplas RDF(*Load*) e de consultas (*Query*).

O seguinte ambiente computacional foi configurado para realização dos testes:

- Sistema operacional Linux Ubuntu versão 12.4;
- Processador Intel Core i7 2GHz;
- 2 gigabytes de memória;
- Disco rígido SATA de 350 gigabytes;
- Jena TDB versão 0.8.10;
- Jena API versão 2.6.4;

Para que fosse possível realizar os testes com um número significativo de objetos de aprendizagem, foi necessário selecionar uma fonte externa de dados e então extrair as informações nela contida. Essas informações estavam armazenadas na forma de metadados do padrão *Dublin Core*<sup>19</sup>. A fonte escolhida foi a do BIOE (Banco Internacional de Objetos de Aprendizagem), onde consta na sua base aproximadamente 17.600 objetos de aprendizagem.

Devido ao formato que se encontram os metadados armazenados, foi necessário criar um conversor para o padrão de metadados OBAA. A coleta das informações se deu por

---

<sup>19</sup> Padrão de metadados que visa descrever objetos digitas.

intermédio do protocolo OAI-PMH <sup>20</sup>, oferecida pelo próprio BIOE. Na Tabela 9 é possível verificar a quantidade de objetos de aprendizagem carregados (*Load*) juntamente com seus respectivos tempos de carga no repositório;

**Tabela 9: Dados sobre carga de OA**

OA	Tempo(Seg)	Triplas RDF	TPS <sup>21</sup>
99	13	2354	181,08
198	14	5107	364,79
412	17	9836	578,59
897	20	17928	896,40
1888	24	39883	1661,79
4196	33	72446	2195,33
11088	66	192785	2920,98

Fonte: Elaborado pelo autor

Com base nos resultados obtidos, é possível observar que o tempo de carga se manteve linearmente proporcional ao número de objetos, indicando uma possível complexidade máxima de ordem  $O(n)$  para esse processo. Inclusive, o processo de carga no repositório se mostrou com uma melhor performance com cargas acima de 2000 OA.

Em outro teste, foi executada uma consulta (Figura 39) no formato Sparql, onde a mesma tem por objetivo recuperar todos os objetos de aprendizagem armazenados, ordenados por título.

**Figura 39: Consulta Sparql**

```
SELECT ?lobj ?key ?desc ?loc ?plat ?title
WHERE {
  ?lobj a obaa:LearningObject .
  ?lobj obaa:hasMetadata ?mdata.
  ?mdata obaa:itsKeywordIs ?key.
  ?lobj obaa:hasMetadata ?mdtit. ?mdtit obaa:itsTitleIs ?title .
  ?lobj obaa:hasMetadata ?mddesc. ?mddesc obaa:itsDescriptionIs ?desc .
  ?lobj obaa:hasMetadata ?mdloc. ?mdloc obaa:itsLocationIs ?loc .
}
order by ?title
```

Fonte: Elaborado pelo autor

O tempo despendido para execução da consulta pode ser visto na Tabela 10.

<sup>20</sup> *Open Archives Initiative Protocol for Metadata Harvesting* - Mecanismo para interoperabilidade entre repositórios de metadados.

<sup>21</sup> Triplas por segundo

**Tabela 10: Dados sobre a consulta de OA**

OA	Tempo(Seg)	OA por Segundo
99	1,78	55,6
198	2,11	93,8
412	2,81	146,6
897	3,27	274,3
1888	5,07	372,3
4196	6,30	635,7
11088	14,28	776,4

Fonte: Elaborado pelo autor

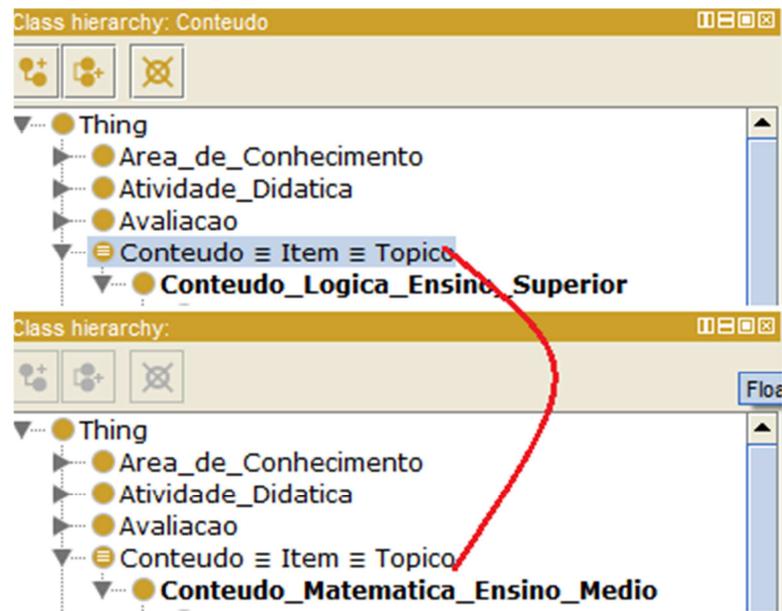
De acordo com os dados apresentados, o desempenho da busca mostra-se logaritmicamente proporcional ao número de 99 OA até os 4200 OA, passando a um desempenho mais linear após esta quantidade de OA. Apesar da necessidade de mais testes, esses dados são indicativos de um possível ótimo desempenho de ordem  $O(\log(n))$  para a busca, com um eventual máximo de ordem  $O(n)$ , ambos resultados muito bons para consultas.

Os resultados dos testes de desempenho de carga e consulta permitem concluir preliminarmente que esta tecnologia de repositório semântico poderá suportar uma relativa quantidade de OA, sem que a performance seja comprometida.

## 6.2 Avaliação da funcionalidade do sistema de alinhamento

Este experimento tem por finalidade verificar de forma funcional, se a técnica de alinhamento desenvolvida agregou mais semântica ao sistema *MSSearch*. Para isto, este experimento conta com o alinhamento de duas ontologias de domínios educacionais, sendo a ontologia de lógica *MilosEduLogica-v10.owl* e a ontologia de matemática *MilosEduMat-v10.owl*, ambas ontologias fazem parte do conjunto de ontologias desenvolvidas dentro do projeto OBAA-MILOS. Estas duas ontologias foram alinhadas automaticamente pelo sistema *MSSearch*, e este alinhamento se deu pela correspondência da classe *Conteudo*, conforme pode ser visto na Figura 40.

Figura 40: Alinhamento das ontologias de matemática e lógica



Fonte: Elaborado pelo autor

Após a preparação e a execução do alinhamento, o sistema *MSSearch* está apto a localizar objetos de aprendizagem com conteúdos de lógica e matemática. Na Figura 41, uma busca teste é realizada, com o objetivo de localizar objetos de aprendizagem no repositório semântico, cujo seu conteúdo seja referente a "tautologias".

Figura 41: Experimento teste de consulta com alinhamento

Busca Semântica

tautologias

Até 20 Resultados

Pesquisar

Termos Inferidos

Resultados

OA:	lo405	Localização:	<a href="http://obaa.unisinos.br">http://obaa.unisinos.br</a>	Relevância	64%
Título:	Objeto de Aprendizagem que fala sobre <b>equivalencias tautologicas</b>				Formato:
Descrição:	Considere que P e Q sejam duas formulas logicas quaisquer e que $P \rightarrow Q$ seja uma tautologia. Então pela propria definição do conetivo $\rightarrow$ , sempre que P for V numa dada linha da tabela verdade de $P \rightarrow Q$ , a fórmula Q também deverá ser V nesta linha. O mesmo acontece para quando P tem valor F. Neste caso se diz que P e Q são fórmulas equivalentes. Esta propriedade e denotada pelo operador $\leftrightarrow$ de <b>equivalencia tautologica</b> entre as formulas P e Q, simbolicamente fica $P \leftrightarrow Q$				
Palavras Chaves:	Equivalencias Tautologicas, Logica	Plataforma:	Plataforma Esp:	mobile	Requerimento: any/browser
Contexto:	school	Estratégia Didática:	Dificuldade:	easy	Iteração:
OA:	lo406	Localização:	<a href="http://obaa.unisinos.br">http://obaa.unisinos.br</a>	Relevância	36%
Título:	OA sobre <b>Regras de Dedução Natural</b>				Formato:
Descrição:	Dedução natural é um dos sistemas <b>dedutivos</b> utilizados para construir demonstrações formais na Lógica. Nos anos 30, foram introduzidos pela primeira vez, por Gentzen e Ja?kowski, os sistemas de <b>Dedução Natural</b> para a Lógica Clássica. As demonstrações realizadas no sistema de dedução natural seguem uma via sintática e utilizam árvores de derivação				
Palavras Chaves:	Deducão Natural, Regras Deducão, Logica	Plataforma:	Plataforma Esp:		Requerimento: pc-dos/
Contexto:		Estratégia Didática:	Dificuldade:	very easy	Iteração:

Fonte: Elaborado pelo autor

Como pode ser visto, o sistema além de recuperar um OA sobre tautologias, que posicionou em primeiro lugar, ainda recuperou outro OA sobre regras de dedução natural, sendo que este resultado foi inferido, conforme a ontologia de lógica alinhada. Nota-se também que na tela, o resultados inferido estão grifados em amarelo.

### 6.3 Validação do sistema *MSSearch* na percepção dos usuários

O experimento em questão, tem por objetivo fazer uma análise comparativa entre o sistema *MSSearch* com o sistema BIOE. Inicialmente, foi necessário preparar um repositório e popular com OA extraídos do BIOE (ver Seção 6.1). Todos os objetos de aprendizagem de ensino médio de matemática que constavam no BIOE também se encontravam no repositório semântico do sistema *MSSearch*. Também foi disponibilizado um acesso via internet ao sistema (<http://obaa.unisinos.br/MSSearch>), possibilitando assim o seu uso de qualquer lugar.

Para realização dos testes de validação, foram convidados seis professores, residentes no Rio Grande do Sul e no Rio de Janeiro, todos graduados e pós-graduados em matemática, com mais de 10 anos de experiência de atuação na área. No entanto, somente quatro professores realizaram os testes.

Nos testes os professores puderam criar termos para a busca (por exemplo: "*Polinômios*"), de livre escolha, e então submeteram ao sistema de busca *MSSearch* e ao BIOE. Com base nos resultados retornados, os professores responderam um questionário de avaliação. A avaliação (ver anexo II) requer que cada professor faça dois testes de busca distintos, com duas palavras chaves diferentes, mas de sua livre escolha. A Tabela 11 sumariza os resultados obtidos das avaliações realizadas pelos professores.

**Tabela 11: Resultados da avaliação comparativa entre os sistemas *MSSearch* e BIOE**

<b>O sistema apresentou algum resultado?</b>	<b>Sim</b>	<b>Não</b>	<b>Parcial</b>
<i>MSSearch</i>	75,00%	00,00%	25,00%
BIOE	75,00%	25,00%	00,00%
<b>O sistema apresentou resultados conforme esperado?</b>	<b>Sim</b>	<b>Não</b>	<b>Parcial</b>
<i>MSSearch</i>	62,50%	25,00%	12,50%
BIOE	37,50%	50,00%	12,50%
<b>A relevância dos resultados apresentados foram satisfatórios?</b>	<b>Sim</b>	<b>Não</b>	<b>Parcial</b>
<i>MSSearch</i>	62,50%	00,00%	37,50%
BIOE	12,50%	50,00%	37,50%

<b>O sistema apresentou resultados dentro do contexto da pesquisa ?</b>	<b>Sim</b>	<b>Não</b>	<b>Parcial</b>
<i>MSSearch</i>	50,00%	25,00%	25,00%
BIOE	37,50%	37,50%	25,00%
<b>O sistema foi capaz de ordenar por relevância?</b>	<b>Sim</b>	<b>Não</b>	<b>Parcial</b>
<i>MSSearch</i>	62,50%	12,50%	25,00%
BIOE	12,50%	75,00%	12,50%
<b>O sistema limitou o número de resultados conforme selecionado?</b>	<b>Sim</b>	<b>Não</b>	<b>Parcial</b>
<i>MSSearch</i>	75,00%	12,50%	12,50%
BIOE	25,00%	25,00%	50,00%
<b>A quantidade de informações apresentadas na tela de consulta foi satisfatória?</b>	<b>Sim</b>	<b>Não</b>	<b>Parcial</b>
<i>MSSearch</i>	50,00%	25,00%	25,00%
BIOE	37,50%	50,00%	12,50%
<b>O tempo de resposta do sistema de busca foi dentro do esperado?</b>	<b>Sim</b>	<b>Não</b>	<b>Parcial</b>
<i>MSSearch</i>	50,00%	50,00%	00,00%
BIOE	50,00%	12,50%	37,50%

Fonte: Elaborado pelo autor

A avaliação comparativa realizada entre os sistemas destacados apresentou na maioria das questões resultados favoráveis. Destaque para a ordenação por relevância, e para a relevância dos resultados apresentados pelo sistema *MSSearch*. Esta boa avaliação se deve principalmente pelo uso dos termos inferidos na ordenação por relevância e também na localização dos OA semanticamente relacionados.

## 7 CONCLUSÕES

Este trabalho apresentou o projeto e o desenvolvimento de um sistema denominado *MSSearch*, para recuperação semântica de objetos de aprendizagem, estruturado conforme o padrão de metadados OBAA, armazenados em um repositório OWL nativo.

O uso de técnicas de alinhamento de ontologias possibilitou que ontologias provenientes de fontes distintas fossem integradas, resultando em uma maior cobertura do conhecimento e compartilhamento de informações. Porém, este tipo de técnica por vezes, dependendo do cenário, não apresenta resultados satisfatórios, pois muitas vezes as ontologias apresentam grandes diferenças na sua estrutura e também nos nomes dos seus vocabulários. Sendo assim, foi proposto neste trabalho o uso de anotações OWL para auxiliar a localização de correspondências entre ontologias de domínios educacionais. O uso de anotações OWL também possibilitou o mapeamento entre ontologias de domínios educacionais alinhadas com a ontologia de metadados OBAA, habilitando assim o sistema a localizar semanticamente objetos de aprendizagem. Apesar de mencionar no texto que esta técnica seria usada em ontologias de domínios educacionais, a mesma pode ser utilizada em ontologias de outros domínios, porém este assunto não foi abordado por estar fora do escopo deste projeto.

No decorrer do desenvolvimento deste trabalho de dissertação, notou-se uma preocupação dos pesquisadores em desenvolver mecanismos para tratar do problema relacionado a heterogeneidade semântica. Apesar da ferramenta ter sido projetada e executada com sucesso, sendo comprovada através dos resultados obtidos nos experimentos, muito ainda pode ser explorado. O uso de anotações OWL para auxílio no alinhamento de ontologias se mostrou eficiente, porém este processo é realizado manualmente, necessitando conhecimento prévio das ontologias que se deseja anotar e conseqüentemente alinhar. É possível desenvolver estudos em cima de técnicas que possibilitem que estas anotações sejam feitas com a mínima interferência dos usuários, e que preferencialmente sejam automáticas. Foi utilizada basicamente técnicas terminológicas para a realização dos alinhamento, porém entende-se como necessário o uso combinado das outras técnicas apresentadas na Seção 3.5.1.

Uma outra contribuição resultante deste trabalho, foi o conversor de metadados provenientes de outros repositórios, formatados em outros padrões, para o padrão OBAA. Pretende-se evoluir com esta ferramenta, criando um mecanismo automático de sincronização desses metadados diretamente no repositório OWL nativo.

Este trabalho também teve por finalidade explorar novas tecnologias de armazenamento e recuperação de dados anotados semanticamente. Testes de desempenho demonstraram que bancos de dados nativos OWL podem servir como plataforma de gerenciamento de dados semântico, mostrando-se capaz de recuperar e armazenar eficientemente uma quantidade razoável de metadados de OA em um tempo satisfatório.

O desempenho dos mecanismos de inferências mostrou-se um tanto ineficiente em alguns casos, e muitas das APIs disponíveis atualmente encontram-se desatualizadas, gerando incompatibilidades de integração com outras APIs. Portanto, se faz necessário desenvolver novos estudos em cima destes mecanismos.

Durante o desenvolvimento deste projeto de pesquisa, foi produzido o artigo intitulado "*MSSearch*: Busca Semântica de Objetos de Aprendizagem OBAA com Suporte a Alinhamento Automático de Ontologias", publicado nos anais do SBIE 2012.

Os resultados obtidos com as validações, demonstraram que o sistema *MSSearch* obteve avaliações positivas pelos usuário que o utilizaram. Dentre estas avaliações, destaca-se sua performance na busca e apresentação dos resultados e também a qualidade da informação apresentada. Isto se deve principalmente pelo uso de mecanismos de inferência e a utilização de técnicas de alinhamentos de ontologias, combinada com o uso de repositórios nativos OWL. Conclui-se então que a ferramenta atingiu seus objetivos, e acredita-se que a mesma poderá dar uma contribuição relevante para busca e localização de objetos de aprendizagem, favorecendo assim a disseminação do conhecimento.

## REFERÊNCIAS

BARCELOS, C. F. ; GLUZ, J. C. ; VICARI, R. M.. An Agent-based Federated Learning Object Search Service. **Interdisciplinary Journal of E-Learning and Learning Objects (IJELLO)**, v. 7, p. 37-54, 2011.

BELLIFEMINE, F. L.; CAIRE, G.; GREENWOOD, D. **Developing Multi-Agent Systems with JADE**. John Wiley & Sons, 2007.

BERNERS-LEE, T; HENDLER, J; LASSILA, O. **The SemanticWeb. Issue of Scientific American**, Maio 2001.

BIOE. **Banco Internacional de Objetos Educacionais (BIOE) Normas para a definição dos metadados**. Disponível em: <<http://objetoseducacionais2.mec.gov.br/retrievefile/normas>>. Acesso em: 30 Abril 2012.

BORDINI, R. H.; FISHER, M.; VISSER, W.; WOOLDRIDGE, M.; **Verifiable multi-agent programs. Programming Multi-Agent Systems**. Capítulo de livro, Berlin / Heidelberg, Springer, v. 3067, p. 72-89, 2004.

BORDINI, R.; HÜBNER, J. F.; WOOLDRIDGE, Michael. **Programming Multi-Agent Systems in AgentSpeak using Jason**, 2007.

BRATMAN, M. E.; ISRAEL, D. J.; POLLACK. Plans and Resource-Bounded Practical Reasoning. **Computational Intelligence**, v.4, n.3, p.349-355, 1988.

CAMPOS, F.; et al. Rede de Ontologias: apoio semântico a linha de produtos de objetos de aprendizagem. **Anais do XXIII Simpósio Brasileiro de Informática na Educação (SBIE 2012)**, ISSN 2316-6533, 2012.

CHAIB-DRAA, B.; DIGNUN, F. Trends in Agent Communication Language. **Computational Intelligence**, Cambridge, v. 2, n. 5, 2002.

COHEN, W., et al. A comparison of string distance metrics for name-matching tasks. Proceedings of IJCAI-03 Workshop on Information Integration on the Web, 2003.

DALTIO, J.; MEDEIROS, F. Um Servidor de Ontologias para Sistemas de Biodiversidade. **Anais do XXVII Congresso da SBC - XXXIV Seminário Integrado de Software e Hardware**, 2002.

DAVID, J.; EUZENAT, J.; SCHARFFE, F.; SANTOS, C.; T. **The Alignment API 4.0**. Disponível em: <<http://www.semantic-web-journal.net/content/new-submission-alignment-api-40>>. Acesso em: 20 Maio 2012.

EHRIG, M. **Ontology alignment: bridging the semantic gap**. Springer, 2007.

EUZENAT, J. **Introduction to ontology matching and alignment (first M2R lecture notes)**. France, 2007.

EUZENAT, J. **Semantic Interoperability**. Disponível em: <<http://www.dit.unitn.it/~p2p/RelatedWork/Matching/SIatISWC08.pdf>>. Acesso em: 05 Abril 2012.

FIPA . **SC00037J: FIPA Communicative Act Library Specification**, 2002.

GLUZ, J.; C.; VICARI, R.; M. MILOS: Infraestrutura de Agentes para Suporte a Objetos de Aprendizagem OBAA. **Anais do XXI Simpósio Brasileiro de Informática na Educação (SBIE 2010)**, 2010.

GLUZ, J.; C.; VICARI, R.; M. Uma Ontologia OWL para Metadados IEEE-LOM, Dubli-Core e OBAA . **Anais do XXII Simpósio Brasileiro de Informática na Educação (SBIE 2011)**, 2011.

GLUZ, J.; C.; VICARI, R.; M.; RIBEIRO, A.; M.; et al. **Especificação Detalhada da Arquitetura da Infraestrutura MILOS Versão 1.0**. UFRGS-UNISINOS, Outubro 2011.

GLUZ, J.; C.; VICARI, R.; M. An OWL Ontology for IEEE-LOM and OBAA. **Proceedings of 11th International Conference on Intelligent Tutoring Systems (ITS 2012)**, Chania, Crete, 2012.

GRUBER, T. R. A translation approach to portable ontology specifications. **Knowledge Acquisition**, 5(2):199-220, 1993.

GRUBER, T. R. “**Ontology**” in Encyclopedia of Database Systems, Ling Liu and M. Tamer Ozsü (Eds.), Springer-Verlag, 2008.

GOH, C.; H. **Representing and Reasoning about Semantic Conflicts in Heterogeneous Information Sources**. 1997. Phd Thesis (Philosophy Course), MIT, 1997.

IEEE-LTSC. **Learning Technology Standards Committee**. Disponível em: <<http://ltsc.ieee.org/wg12/>>. Acesso em: 25 Abril 2012.

JENA. **Framework for Building Semantic Web Applications**. Disponível em: <<http://jena.apache.org/>>. Acesso em: 10 Maio 2012.

LABROU, Y.; FININ T.; PENG Y. Agent communication languages: the current landscape. **IEEE Intelligent Systems**. p.45-52, Março-abril de 1999.

MEC. **Objetos de aprendizagem: uma proposta de recurso pedagógico**. Organização: PRATA, C.,NASCIMENTO, A. C. – Brasília: MEC, SEED, 2007. 154 p.

MILLER, A.; G. WordNet: A Lexical Database for English. **Communications of the ACM**, v.38, n. 11, p. 39-41. Princeton University. 1995.

NOY, N. **Ontology Mapping and Alignment**. Disponível em: <<http://disi.unitn.it/~accord/RelatedWork/Matching/Noy-MappingAlignment-SSSW-05.pdf>>. Acesso em: 12 Maio 2012.

OAEI. **Ontology Alignment Evaluation Initiative**. Disponível em: <<http://oaei.ontologymatching.org/>>. Acesso em: 2 Maio 2012.

OCHS, C.; et al. Google Knows Who Is Famous Today - Building an Ontology From Search Engine Knowledge and DBpedia. **Proceedings of Fifth IEEE International Conference on Semantic Computing**, 2011.

ORENGO, V.; M.; HUYCK, C.; A. Stemming Algorithm for Portuguese Language. **Proceedings of the Symposium on String Processing and Information Retrieval**, 2001.

OWL-API. **API for Manipulation and Serialization OWL Ontologies**. Disponível em: <<http://owlapi.sourceforge.net/>>. Acesso em: 20 Maio 2012.

PADGHAN, L., WINIKOFF, M. A Methodology for Developing Intelligent Agents. **Proceedings of AOSE**, 2002.

PENA, A., PENYA, Y. K. Distributed semantic repositories in smart grids. **Proceedings of 9th IEEE International Conference on Industrial Informatics (INDIN 2011)**, pp.721-726, 26-29, 2011.

PROTÈGÈ-API. **Open Source Ontology Editor and Knowledge-base Framework**, Disponível em: <<http://protege.stanford.edu/>>. Acesso em: 20 Maio 2012.

RAO, A. S.; GEORGEFF, M. Modeling rational agents within a BDI-architecture. **Proceedings of International Conference on Principles of Knowledge Representation and Reasoning**, Cambridge, Massachusetts: Morgan Kaufmann, p. 473-484, 1991.

RAO, A. S. AgentSpeak(L): BDI agents speak out in a logical computable language. **Proceedings of the 7th European workshop on Modelling autonomous agents in a multi-agent world : agents breaking away (MAAMAW '96)**. Springer-Verlag, New York, 1996.

SHVAIKO, P.; EUZENAT, J. Ontology matching: state of the art and future challenges. **IEEE Transactions on Knowledge and Data Engineering**, vol. PP, no. 99, pp. 1, 0, 2011.

SILVA, VIVIAN. **Uma abordagem para alinhamento de ontologias biomédicas para apoiar a anotação genômica**. 2010. Dissertação (Mestrado em Informática) – Núcleo de Computação Eletrônica. Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ, 2010.

SIRIN, E., BULKA, B., SMITH, M. Terp: Syntax for OWL-friendly SPARQL Queries. **Proceedings of 7th OWL Experiences and Directions Workshop**. San Francisco, 2010.

SOUZA, A.; et al. Recuperação Semântica de Objetos de Aprendizagem: Uma Abordagem Baseada em Tesouros de Propósito Genérico. **Anais XIX Simpósio Brasileiro de Informática na Educação (SBIE 2008)**, 2008.

THAKKER, D.; OSMAN, T.; GOHIL, S.; LAKIN, P. A Pragmatic Approach to Semantic Repositories Benchmarking. **Proceedings of the 7th international conference on The Semantic Web: research and Applications - Volume Part I (ESWC'10)**, 2010.

WORDNET. **A lexical database for English**. Disponível em: <<http://wordnet.princeton.edu/>>. Acesso em: 25 Abril 2012.

WOOLDRIDGE, M. **An Introduction to Multiagent Systems**. John Wiley & Sons, 2002.

WOOLDRIDGE, M.; JENNINGS, R.; M. Intelligent Agents: theory and practice. **The Knowledge Engineering Review**, Vol. 10:2,1995, 115-152.

VICCARI, R.; GLUZ, J.; PASSERINO, L.; et al. The OBAA Proposal for Learning Objects Supported by Agents. **Proceedings of MASEIE Workshop – AAMAS 2010**, Toronto, Canada, 2010.

**ANEXO I - ARTIGOS PUBLICADOS**

Anais do XXIII SBIE - XVIII CBIE Rio de Janeiro, 26 a 30 de novembro de 2012.  
*MSSearch*: Busca Semântica de Objetos de Aprendizagem OBAA com Suporte a Alinhamento Automático de Ontologias.

## ANEXO II - FORMULÁRIO DE AVALIAÇÃO DO SISTEMA *MSSearch*

### Avaliação Comparativa do Sistema *MSSearch* X *BIOE* - Teste Nº 1

Palavra Chave Pesquisada \*

Avaliação do Sistema de Busca *MSSearch* \*

	Sim	Não	Parcial
O sistema apresentou algum resultado?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O sistema apresentou resultados conforme esperado?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A relevância dos resultados apresentados foram satisfatórios?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O sistema apresentou resultados dentro do contexto da pesquisa ?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O sistema foi capaz de ordenar por relevância?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O sistema limitou o número de resultados conforme selecionado?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A quantidade de informações apresentadas na tela de consulta foi satisfatória?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O tempo de resposta do sistema de busca foi dentro do esperado?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

### ANEXO III - RESULTADOS OBTIDOS DO QUESTIONÁRIO DE AVALIAÇÃO DO SISTEMA *MSSearch*

