

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

Antonio Phillipi Maciel Silva

*Uma Abordagem para a Junção de Ontologias e sua utilização no
Desenvolvimento de Ontologias de Aplicação*

São Luís - MA

2014

Antonio Phillipi Maciel Silva

*Uma Abordagem para a Junção de Ontologias e sua utilização no
Desenvolvimento de Ontologias de Aplicação*

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão como requisito parcial para a obtenção do grau de MESTRE em Engenharia de Eletricidade.

Orientador: Rosario Girardi

Doutora em Ciência da Computação – UFMA

São Luís - MA

2014

Silva, Antonio Fhillipi Maciel

Uma Abordagem para a Junção de Ontologias e sua utilização no Desenvolvimento de Ontologias de Aplicação / Antonio Fhillipi Maciel Silva. – São Luís - MA, 2014.

114 f.

Orientador: Rosario Girardi.

Impresso por computador (fotocópia).

Dissertação (Mestrado) – Universidade Federal do Maranhão, Programa de Pós-Graduação em Engenharia de Eletricidade. São Luís - MA, 2014.

1. Reúso de Ontologias. 2. Junção de Ontologias. 3. Análise de Similaridades I. Girardi, Rosario.

CDU 004.056.xx

Antonio Phillipi Maciel Silva

*Uma Abordagem para a Junção de Ontologias e sua utilização no
Desenvolvimento de Ontologias de Aplicação*

Este exemplar corresponde à redação final da dissertação devidamente corrigida e defendida por Antonio Phillipi Maciel Silva e aprovada pela comissão examinadora.

Aprovada em 07 de Novembro de 2014

BANCA EXAMINADORA

Rosario Girardi (orientador)

Doutora em Ciência da Computação – UFMA

Francisco José da Silva e Silva

Doutor em Ciência da Computação – UFMA

Lucelene Lopes

Doutora em Ciência da Computação – PUCRS

*À minha mãe e a meu
pai, que me ajudaram nessa
caminhada.*

RESUMO

O reúso de ontologias é um processo em que o conhecimento ontológico existente é usado como entrada para gerar novas ontologias, visando a redução de custos e o aumento da qualidade do produto final. No entanto, as técnicas para construção de ontologias não abordam de maneira satisfatória o reúso, mesmo este sendo uma fase indispensável para a engenharia de ontologias. O presente trabalho apresenta o OntoJoin, um processo para junção de ontologias, que emprega a análise de similaridade lexical, estrutural e relacional como mecanismos de mapeamento. Esses mecanismos são responsáveis por identificar correspondências entre elementos de duas ontologias dadas como entrada. Essa correspondência é utilizada para combinar os elementos similares, resultando assim, em uma nova ontologia gerada a partir do reúso. O uso conjunto dos mecanismos “Comparação Lexical”, que realiza uma comparação entre os rótulos dos termos dos elementos; “Comparação Estrutural”, que realiza uma análise dos conceitos e sua respectiva estrutura hierárquica; “Comparação Relacional”, que realiza uma análise dos conceitos, suas propriedades e relações não taxonômicas; e “Indexação dos Termos”, que altera a hierarquia dos conceitos para representar melhor semanticamente os termos a serem combinados, é a característica principal que confere ao OntoJoin o potencial de obter maior efetividade na junção de ontologias em relação a técnicas até então propostas. A avaliação experimental do processo foi realizada conforme dois procedimentos baseados no princípio de comparação da ontologia combinada com a de referência. Esse experimento consistiu em mensurar com as medidas de avaliação cobertura e precisão, a efetividade do processo em combinar duas ontologias nos domínios do turismo e vendas. Os resultados obtidos demonstram preliminarmente a viabilidade do processo proposto na junção de ontologias.

Palavras-chaves: Reúso de Ontologias, Junção de Ontologias, Análise de Similaridades.

ABSTRACT

The reuse of ontologies is a process in which the existing ontological knowledge is used as input to generate new ontologies, in order to reduce costs and increase the quality of the final product. However, techniques for building ontologies do not address reuse satisfactorily, even though this is an indispensable phase in ontology engineering. This work presents OntoJoin, a process for joining ontologies which employs lexical, structural and relational similarity analysis as mapping mechanisms. These mechanisms are responsible for identifying correspondences between elements of two ontologies given as input. These mechanisms are used to match similar elements, thus resulting in a new ontology generated from reuse. The set of mechanisms use "Lexical Comparison", which performs a comparison between the labels of the terms of the ontology elements; "Structural Comparison", which performs an analysis of the concepts and their respective hierarchical structure; "Relational Comparison", which performs an analysis of concepts, their properties and non-taxonomic relationships; and "Index of Terms", which alter the concepts for their better representation, is the main feature that gives OntoJoin the potential to achieve greater effectiveness at the junction of ontologies compared with previously proposed techniques. An experimental evaluation has been performed according two procedures based on the principle of comparing the joint ontology against a reference one. This experiment consisted in measuring with recall and precision the effectiveness of the process for combining two ontologies in the tourism and sales domains. The preliminary results demonstrate the feasibility of the proposed process in joining ontologies.

Keywords: Ontologies Reuse, Ontologies Join, Similarity analysis.

Agradecimentos

As primeiras palavras de agradecimento vão a Deus, por manifestar Sua glória fornecendo-me condições para suportar os desafios encontrados durante minha trajetória de vida e agora me permitindo alcançar mais este sonho.

A minha família que sempre me deu suporte e confiança.

A minha namorada Yanna, pelo carinho, atenção, paciência e incentivo durante esta fase.

A minha orientadora Prof.^a Dr.^a Rosario Girardi, pela oportunidade concedida.

Aos queridos colegas do Grupo de pesquisa em Engenharia de Software e Engenharia do Conhecimento - GESEC/ UFMA, pelo essencial apoio a esta pesquisa e pelas alegrias partilhadas, em especial à Suzane Carvalho, Ivo Serra e Paulo Cardoso. Obrigado, Adriana, Willian, Messias e Rayane, pelo apoio recebido durante a realização deste trabalho. E ainda aos colegas de mestrado, Ariel, Arikleyton, Berto, Dhully, Haroldo, Luís Aurélio, Marcus Vinicius, Raquel, Thiago Pinheiro, pela parceria amistosa durante este curso.

Ao Programa de Pós-Graduação em Engenharia de Eletricidade, pela oportunidade da realização do curso; e a Alcides e Neto, pelo bom atendimento às solicitações acadêmicas.

A todos meus professores do CEUMA do curso de graduação em Sistemas de Informação. E também, ao Prof. Dr. Nilson Costa, Prof. Dr. Rômulo Martins e Prof. Dr. Ademir Martins que forneceram minhas cartas de recomendação exigidas por este Programa de Pós-Graduação. A estes, com carinho, exprimo meu respeito, admiração e gratidão.

*"E disse Jesus: Se tu podes crer, tudo é possível
ao que crê."*

Marcos 9.23

Lista de Figuras

2.1	Demonstração gráfica de parte de uma ontologia	21
2.2	Hierarquia de ontologias [17]	22
2.3	Ciclo de vida da técnica 101 [28]	25
2.4	Ciclo de vida da técnica de Uschold e King [46]	28
2.5	Ciclo de vida da técnica de Methontology [12]	31
2.6	Ciclo de vida da técnica de GAODT [39]	33
2.7	Exemplo do alinhamento de ontologias, adaptado a partir de Noy [32] .	38
2.8	Exemplo da integração de ontologias, adaptado a partir de Noy [32] . .	39
2.9	Exemplo da junção de ontologias, adaptado a partir de Noy [32]	41
2.10	Ciclo de vida do processo FCA-MERGE [44]	46
3.1	Visão geral do processo OntoJoin	52
3.2	Visão geral da fase Identificação e Extração dos Elementos das Ontologias	54
3.3	Demonstração gráfica de parte da ontologia O_1	55
3.4	Demonstração gráfica de parte da ontologia O_2	56
3.5	Visão geral da fase de Mapeamento	57
3.6	Visão geral da atividade Comparação Lexical	58
3.7	Regra 1 para indexação de termos	67
3.8	Regra 2 para indexação de termos	68
3.9	Regra 3 (a) para indexação de termos	69
3.10	Visão geral da fase Junção	72
3.11	Exemplo da combinação de elementos do conjunto P	74
3.12	Ilustração de parte de uma ontologia reusada	75

3.13	Visão geral da Ferramenta OntoJoinTool	76
3.14	Interface para selecionar arquivos OWL e botão para configuração dos pesos das medidas de similaridade	77
3.15	Interface que demonstra os termos extraídos e seus respectivos conjuntos (a)	78
3.16	Interface que demonstra os termos extraídos e seus respectivos conjuntos (b)	78
3.17	Interface para inserção do valor de corte	79
3.18	Valores de similaridade obtidos após o mapeamento	79
3.19	Redefinição dos termos correspondentes	79
3.20	Barra de progresso da interface da etapa Join	80
3.21	Termino da execução da ferramenta	81
4.1	Classificação das ontologias (adaptadas a partir de [17])	84
4.2	Ontologia OntoTur - Parte 1	84
4.3	Ontologia OntoTur - Parte 2	85
4.4	Ontologia OntoTur - Parte 3	85
4.5	Ontologia OntoTur - Parte 4	85
4.6	Ontologia Sales	88
4.7	Interface para selecionar arquivos OWL e botão para configuração dos pesos das medidas de similaridade	91
4.8	Elementos do conjunto C_C e P extraídos a partir das ontologias dadas como entrada	91
4.9	Elementos do conjunto R e H extraídos a partir das ontologias dadas como entrada	91
4.10	Valores do similaridade obtidos após o mapeamento	92
4.11	Redefinição dos termos correspondentes	92
4.12	Termino da execução da ferramenta	93

A.1	Ontologia OntoTurSales - Parte 1	108
A.2	Ontologia OntoTurSales - Parte 2	108
A.3	Ontologia OntoTurSales - Parte 3	109
A.4	Ontologia OntoTurSales - Parte 4	109
B.1	Ontologia OntoTurSales2 - Parte 1	110
B.2	Ontologia OntoTurSales2 - Parte 2	110
B.3	Ontologia OntoTurSales2 - Parte 3	111
B.4	Ontologia OntoTurSales2 - Parte 4	111
B.5	Ontologia OntoTurSales2 - Parte 5	112

Lista de Tabelas

2.1	Comparativo entre as Técnicas para Construção de Ontologias	35
2.2	Comparativo entre as abordagens para o reúso de ontologias	42
2.3	Comparativo entre as Técnicas e Ferramentas para o Reúso de Ontologias	50
3.1	Exemplo de lista de termos similares lexicalmente	59
3.2	Exemplo de lista de termos similares estruturalmente	62
3.3	Exemplo de lista de termos com similaridade aceitável	66
3.4	Exemplo de lista de termos órfãos	66
3.5	Lista de termos Redefinidos	71
3.6	Lista de termos combinados	74
3.7	Lista de termos combinados e analisados	75
3.8	Comparativo entre as técnicas apresentadas no estado da arte em relação ao processo OntoJoin	82
4.1	Quantidade de termos presentes nas ontologias resultantes	93
4.2	Percentual de cobertura e precisão	96

Lista de Siglas

SaaS Software as a Service.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xii
Lista de Siglas	xiii
1 Introdução	17
1.1 Relevância e Motivação	17
1.2 Objetivos	18
1.2.1 Objetivo Geral	18
1.2.2 Objetivos Específicos	18
1.3 Estrutura da Dissertação	18
2 Construção e Reúso de Ontologias	19
2.1 Ontologias	19
2.1.1 Tipos de Ontologias	22
2.2 Construção de Ontologias	24
2.2.1 A Técnica 101	24
2.2.2 A Técnica de Uschold e King	28
2.2.3 A Técnica Methontology	31
2.2.4 A Técnica GAODT	33
2.2.5 Considerações sobre as Técnicas para Construção de Ontologias	35
2.3 Reúso de Ontologias	36
2.3.1 Alinhamento	37
2.3.2 Integração	38

2.3.3	Junção	40
2.3.4	Análise Comparativa entre as Abordagens	41
2.3.5	Mapeamento	42
2.3.6	Técnicas e Ferramentas que suportam o Reúso de Ontologias	44
2.3.7	Considerações sobre as Técnicas para o reúso de Ontologias	48
2.4	Considerações Finais	50
3	O Processo OntoJoin	51
3.1	Identificação e Extração dos Elementos das Ontologias	53
3.2	Mapeamento	56
3.2.1	Comparação Lexical	58
3.2.2	Comparação Estrutural	59
3.2.3	Comparação Relacional	62
3.2.4	Indexação de Termos	66
3.2.5	Considerações sobre a fase do Mapeamento	70
3.3	Análise de Integridade dos Termos	70
3.4	Junção	72
3.4.1	Combinação de Elementos do Conjunto C_C	72
3.4.2	Combinação de Elementos dos Conjuntos P, R, H e A	73
3.5	Análise e Representação da nova Ontologia	75
3.6	OntoJoinTool – Uma Ferramenta de Suporte ao processo OntoJoin	76
3.6.1	Choosing OWL Files	76
3.6.2	Identification and Extraction of the Elements of Ontologies	77
3.6.3	Mapping	78
3.6.4	Integrity Analysis of Terms	79
3.6.5	Join	80
3.6.6	Control and Generation of the New Ontology	81

3.7	Considerações Finais	81
4	Avaliação	83
4.1	Os Domínios Abordados	83
4.1.1	Ontologia de Domínio: OntoTur	84
4.1.2	Ontologia de Tarefas: Sales	88
4.2	Procedimentos Realizados	90
4.2.1	Junção das Ontologias OntoTur e Sales	90
4.2.2	Métricas Utilizadas	93
4.3	Discussão dos Resultados Obtidos	95
4.4	Considerações Finais	98
5	Conclusão	99
5.1	Contribuições e Resultados da Pesquisa	100
5.1.1	Publicações	100
5.2	Trabalhos Futuros	101
	Referências Bibliográficas	102
A	Anexo: Ontologia de Aplicação - OntoTurSales	108
B	Anexo: Ontologia de Aplicação (Referência) - OntoTurSales2	110

1 Introdução

1.1 Relevância e Motivação

Diversas técnicas têm sido desenvolvidas com o intuito de suportar a construção de ontologias, como por exemplo: Uschold e King [46], Methontology [12], 101 [28] e a GAODT [38]. Algumas destas também provêm o reúso no processo de construção [46] [12] [28] [38]. Entretanto, a maioria delas não fornecem detalhes de como o reúso deve ser realizado.

A reusabilidade é um fator crucial na engenharia de ontologias [29]. Portanto, a elaboração de técnicas que contemplem satisfatoriamente esse processo é de fundamental importância para o aumento da produtividade durante as fases de desenvolvimento e manutenção de ontologias. Neste sentido, umas das vantagens de se trabalhar com o reúso é o aumento da interoperabilidade entre aplicações, pois estas poderão reutilizar o conhecimento compartilhado pelas correspondentes ontologias. Além do mais, verificar a existência de outras ontologias que possam ser reutilizadas em um novo empreendimento, pode ser considerada uma boa prática de desenvolvimento de projetos [28].

Conseqüentemente, é significativo o uso da junção como mecanismo para a realização do reúso de ontologias [29]. A junção prover como produto final uma única ontologia composta por elementos herdados a partir de ontologias dadas como entrada, além desta ser independente das ontologias de origem. Tais características são fundamentais para o desenvolvimento de ontologias de aplicação.

Por esses motivos, este trabalho propõe um processo que auxilia o engenheiro do conhecimento durante a fase do reúso. Esta abordagem realiza a junção entre duas ontologias dadas como entrada, baseando-se na análise de similaridade lexical, estrutural, relacional e indexação de termos como mecanismos de mapeamento. Esses mecanismos são responsáveis por identificar correspondências entre elementos de duas ontologias dadas como entrada. Essa correspondência é

utilizada para combinar os elementos similares, resultando assim, em uma nova ontologia gerada a partir do reúso.

1.2 Objetivos

1.2.1 Objetivo Geral

Estabelecer um processo para o reúso e junção de ontologias, contribuindo para a redução de custos e esforços na construção de ontologias de aplicação.

1.2.2 Objetivos Específicos

- Realizar uma revisão crítica do estado da arte do reúso e junção de ontologias e sua aplicação na construção de ontologias de aplicação;
- Especificar um processo para junção de ontologias e sua aplicação na construção de ontologias de aplicação baseada no reúso;
- Avaliar o processo proposto através do desenvolvimento de um estudo de caso.

1.3 Estrutura da Dissertação

Este trabalho, incluindo a introdução, está estruturado em cinco capítulos. O segundo capítulo apresenta as mais relevantes técnicas abordadas na fase de reúso e junção de ontologias e sua aplicação na construção de ontologias, sendo ilustradas por meio de exemplos práticos. Por fim, é realizada uma análise comparativa das técnicas apresentadas. O terceiro capítulo apresenta o OntoJoin, um processo baseado no mapeamento semântico para junção de ontologias. O quarto capítulo apresenta um estudo de caso, que tem como objetivo avaliar preliminarmente a efetividade do processo proposto. O quinto capítulo apresenta as conclusões deste trabalho.

2 Construção e Reúso de Ontologias

Nesse capítulo são apresentadas as áreas de conhecimento que fornecem a fundamentação teórica sobre a construção e reúso de ontologias. A seção 2.1 discorre sobre a definição de ontologia adotada no contexto desse trabalho. A seção 2.2 apresenta as principais abordagens destinadas à construção de ontologias. A seção 2.3 apresenta os principais conceitos e abordagens relacionadas ao reúso de ontologias. Por fim, a seção 2.4 apresenta as considerações finais do capítulo.

2.1 Ontologias

Para contextualizar o reúso de ontologias, mais especificamente a Junção, primeiro deve ser definida o que realmente é uma ontologia, além dos fatores que definem a sua criação. A seguir, são descritos alguns dos conceitos adotados neste trabalho relacionados à ontologia.

As ontologias constituem uma abordagem para a representação do conhecimento capaz de expressar um conjunto de entidades, seus relacionamentos, restrições e regras sobre um determinado domínio [17]. Esses relacionamentos podem ser taxonômicos (definem uma hierarquia de conceitos) e não taxonômicos (definem os relacionamento entre conceitos). Além desses relacionamentos, existem instâncias relacionadas a um determinado conceito.

De acordo com Russel e Norvig [37] o termo “ontologia” diz respeito a uma teoria sobre a existência. Esse termo foi inicialmente usado na filosofia e em seguida incorporado por cientistas da computação e da ciência da informação para expressar uma especificação de uma conceituação.

Uma ontologia é uma especificação formal e explícita de uma conceituação compartilhada de um domínio de interesse [16]. Conceituação refere-se a um modelo abstrato de algum fenômeno do mundo. A especificação é explícita porque os conceitos utilizados e as limitações do seu uso são explicitamente definidos; formal porque

é processável computacionalmente; compartilhada porque captura o conhecimento consensual, isto é, não é privada de algum indivíduo, mas aceita por um grupo.

Formalmente uma ontologia pode ser definida como uma 6-tupla [14]:

$$O = (C, H, I, R, P, A)$$

onde,

$C = C_C \cup C_I$ é o conjunto de entidades do domínio sendo modelado. O conjunto C_C é formado por classes, ou seja, conceitos que representam entidades que descrevem um conjunto de objetos (por exemplo, "Mãe" $\in C_C$) enquanto que o conjunto C_I é formado por instâncias, ou seja, entidades únicas no domínio (por exemplo, "Anne Smith" $\in C_I$).

$H = \{\text{tipo_de}(c_1, c_2) \mid c_1 \in C_C \wedge c_2 \in C_C\}$ é o conjunto de relações taxonômicas que definem a hierarquia de classes da ontologia e são denotadas por "{tipo_de(c_1 , c_2)}" indicando que c_1 é uma subclasse de c_2 . Um exemplo desse relacionamento é "{tipo_de(Mãe, Pessoa)}".

$I = \{\text{é_um}(c_1, c_2) \mid c_1 \in C_I \wedge c_2 \in C_C\} \cup \{\text{prop}_I(c_k, \text{valor}) \mid c_k \in C_I\} \cup \{\text{rel}_k(c_1, c_2, \dots, c_n) \mid \forall i, c_i \in C_I\}$ é o conjunto de relacionamentos entre os elementos da ontologia e suas instâncias, por exemplo "é_um("Anne Smith", Mãe)", "data_de_nascimento ("Anne Smith", "12/02/1980")" e "mãe_de("Anne Smith", "Clara Smith")" são relacionamentos entre classes, relacionamentos, propriedades e suas instâncias.

$R = \{\text{rel}_k(c_1, c_2, \dots, c_n) \mid \forall i, c_i \in C_C\}$ é o conjunto de relacionamentos não taxonômicos de uma ontologia. Por exemplo, "mãe_de(Mãe, Filha)".

$P = \{\text{prop}_C(c_k, \text{tipo}) \mid c_k \in C_C\}$ é o conjunto de propriedades das classes de uma ontologia e seu tipo de dados básico. Por exemplo, "data_de_nascimento(Mãe, dd/mm/aaaa)".

$A = \{\text{condition}_X \implies \text{conclusion}_Y(c_1, c_2, \dots, c_n) \mid \forall j, C_j \in C_C\}$ é um conjunto de axiomas, regras que permitem checar a consistência da ontologia e deduzir novos conhecimentos através de algum mecanismo de inferência. O termo $\{\text{condition}_X$ é dado por: $\{\text{condition}_X = \{(\text{cond}_1, \text{cond}_2, \dots, \text{cond}_n) \mid \forall z, \text{cond}_z \in H \cup \mid \cup R\}$. Por exemplo, "Mãe, Filha1, Filha2, mãe_de(Mãe, Filha1), mãe_de(Mãe, Filha2) \implies irmã_de(Filha1,

Filha2)” é uma regra que indica que, se duas filhas têm a mesma mãe, então as filhas são irmãs.

Para ilustrar, uma ontologia simples que descreve parte do domínio jurídico do direito da família é apresentada na Figura 2.1.

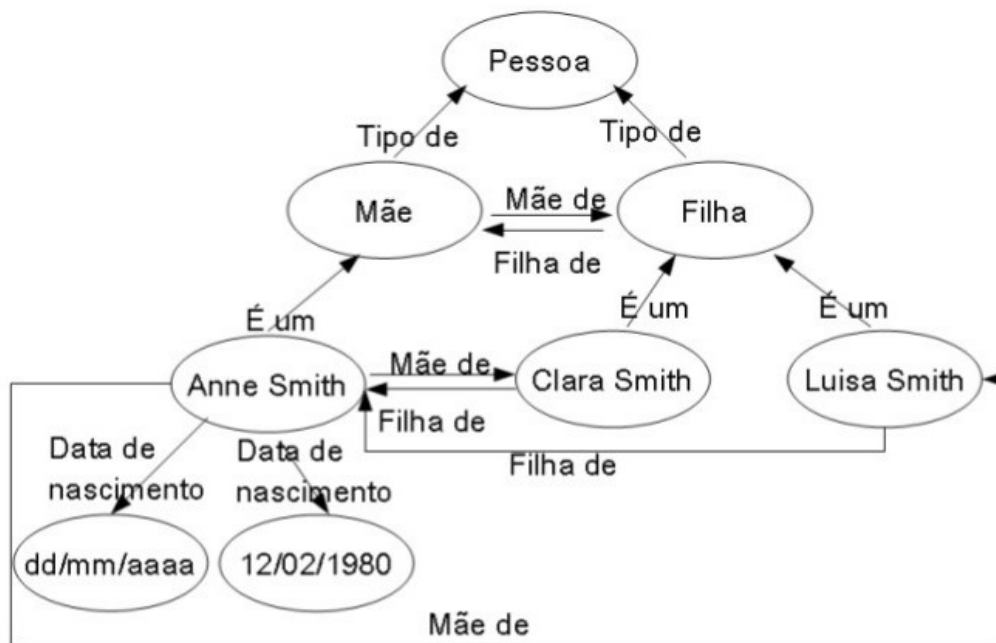


Figura 2.1: Demonstração gráfica de parte de uma ontologia

Considerando a definição e a ontologia da Figura 2.1, os seguintes conjuntos podem ser identificados:

C_C = mãe, filha, pessoa

C_I = Anne Smith, Luisa Smith, Clara Smith

H = tipo_de(mãe, pessoa), tipo_de(filha, pessoa)

I = é_um(Anne Smith, mãe), é_um(Clara Smith, filha), é_um(Luisa Smith, filha), mãe_de(Anne Smith, Clara Smith), mãe_de(Anne Smith, Luisa Smith), filha_de(Clara Smith, Anne Smith), filha_de(Luisa Smith, Anne Smith), data_de_nascimento (Anne, 12/02/1980)

R = mãe_de(mãe, filha), filha_de(filha, mãe)

P = data_de_nascimento (mãe, dd/mm/aaaa)

A = \forall Mãe, Filha1, Filha2, mãe_de(Mãe, Filha1), mãe_de(Mãe, Filha2) \implies irmã_de(Filha1, Filha2)

2.1.1 Tipos de Ontologias

Tais elementos presentes na definição formal podem ser encontrados nos diferentes tipos de ontologias. Guarino [17] define que as ontologias podem ser classificadas como: ontologias genéricas, ontologias de domínio, ontologias de tarefas e ontologias de aplicação. A Figura 2.2 demonstra o modelo hierárquico proposto por Guarino [17]. A seguir são descritas cada uma delas.

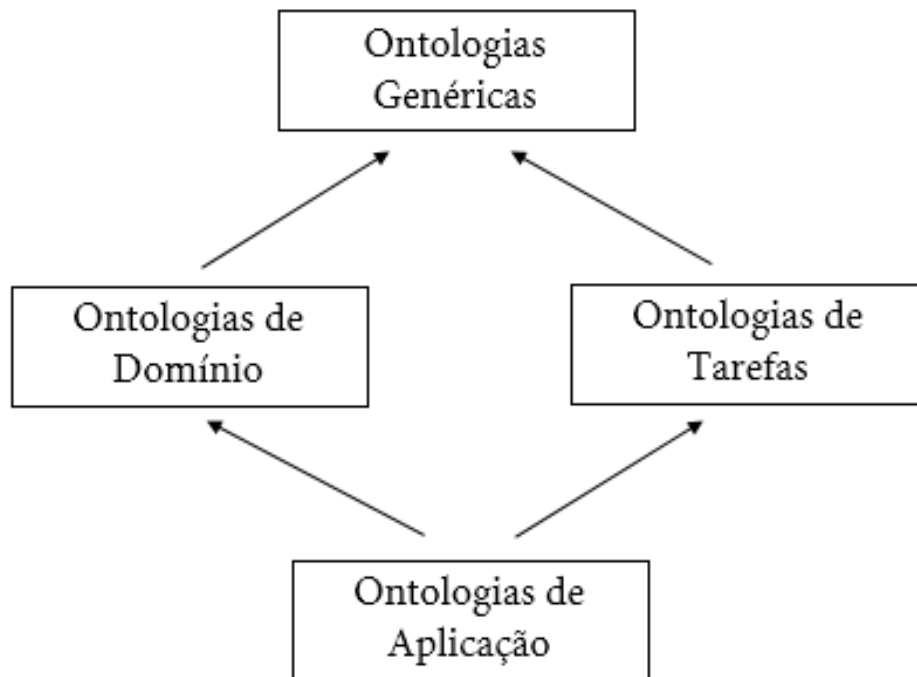


Figura 2.2: Hierarquia de ontologias [17]

Ontologia genéricas

Segundo Guarino [17] ontologias genéricas descrevem conceitos bastante gerais, tais como, espaço, tempo, matéria, objeto, evento, ação, etc., que são independentes de um problema ou domínio particular. De acordo com Guizzardi [18], ontologias genéricas procuram construir teorias básicas do mundo, de caráter bastante abstrato, aplicáveis a qualquer domínio (conhecimento de senso comum). Elas estão relacionadas principalmente ao sentido filosófico de categorização e linguística.

Ontologia de domínio

Segundo Noy et al. [28] uma ontologia de domínio define um vocabulário comum para compartilhar informações de um domínio específico. É o tipo de ontologia mais comum e geralmente é construída para representar um “micromundo” [18]. Estas ontologias descrevem conceitos e vocabulários relacionados a domínios particulares, tais como medicina ou computação.

Martins [24] ressalta que os principais objetivos de se desenvolver uma ontologia de domínio são: compartilhar informação, reusar elementos do domínio, tornar suposições do domínio explícitas, separar conhecimentos do domínio de conhecimento operacional e analisar o conhecimento do domínio.

Ontologia de tarefa

Segundo Martins [24] uma ontologia de tarefa provê um vocabulário de termos usados para resolver problemas associados com uma tarefa, que pode ou não ser realizada em um mesmo domínio. É capaz de capturar o conhecimento a respeito de uma solução de um problema independentemente do domínio na qual está inserida, além de especificar o contexto onde os conceitos do domínio são usados [19].

Martins [24] ressalta que diferentemente das ontologias de domínio, as ontologias de tarefas são extremamente restritas no que diz respeito à metodologias para a sua construção. Neste sentido, a falta de padronização durante o seu desenvolvimento acaba tornando as ontologias de tarefas heterogêneas, reduzindo assim, o reúso deste tipo de ontologias.

Ontologia de aplicação

Gennari et al. [13] definem que ontologias de aplicação são criadas com o propósito de reduzir a distância entre as ontologias de domínio e de tarefas, assimilam o conhecimento requerido por um método de solução de problema que usa a terminologia de uma ontologia de domínio.

Segundo Van Heijst et al. [47] ontologias de aplicação contêm as definições que são necessárias para modelar o conhecimento requerido por uma aplicação

particular. Elas são uma mistura de conceitos derivados de ontologias de domínio, genéricas e de tarefa.

Guarino [17] ressalta que ontologias de aplicação descrevem, no contexto de uma aplicação, conceitos dependentes de um domínio e de tarefas particulares, os quais são, frequentemente, especializações das ontologias relacionadas.

2.2 Construção de Ontologias

O aumento significativo do uso de ontologias como forma adequada para representação do conhecimento, fez com que surgissem inúmeras técnicas destinadas à sua construção, como por exemplo: Técnica 101 [28], Técnica de Uschold e King [46], Técnica Methontology [12] e Técnica GAODT [38] [39].

Segundo Santos [38] a escolha de qual técnica a ser utilizada está relacionada aos objetivos finais que se deseja alcançar com o desenvolvimento da ontologia, como o nível de detalhes dos processos e sua manutenção, assim como o tipo de ontologia que se deseja construir para o domínio considerado.

A seguir, as seções 2.2.1, 2.2.2, 2.2.3 e 2.2.4 apresentam uma sucinta descrição de cada uma dessas técnicas, abordando as suas principais características e peculiaridades. A seção 2.2.5 apresenta uma comparação entre estas técnicas, assim como uma breve consideração sobre elas.

2.2.1 A Técnica 101

A técnica 101 foi desenvolvida a partir da experiência das autoras Natalya F. Noy e a Deborah L. McGuinness na construção de ontologias de domínio [28]. A técnica formaliza algumas regras que são consideradas fundamentais para o processo de construção de ontologias:

1. Não há uma única forma correta de modelar um domínio. A melhor forma sempre depende do propósito que se tem em mente.
2. O desenvolvimento de ontologias é necessariamente um processo iterativo. Ou seja, pode ser repetido em busca do resultado ideal.

3. Conceitos da ontologia devem ser próximos a objetos (físicos ou lógicos) e relacionamentos em um domínio de interesse. Estes conceitos são mais prováveis de serem substantivos (objetos) ou verbos (relacionamentos) em sentenças que descrevem o domínio.

A seguir, a Figura 2.3 apresenta as sete etapas que compõem a técnica 101: “Determinar o domínio e o escopo”, “Considerar o reuso”, “Enumerar termos importantes”, “Definir classes e hierarquia de classes”, “Definir as propriedades”, “Definir as restrições das propriedades” e “Criar as instâncias”.

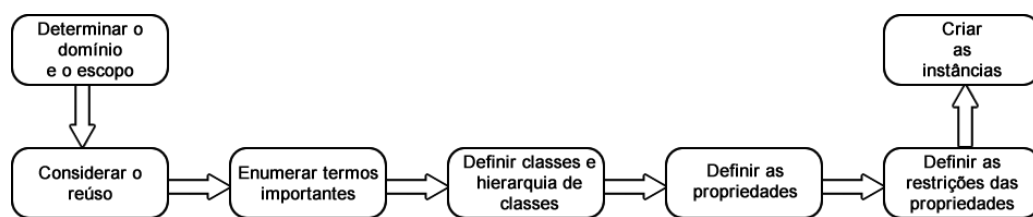


Figura 2.3: Ciclo de vida da técnica 101 [28]

Ao desenvolver uma ontologia seguindo as definições formalizadas pela técnica 101, o primeiro passo é “Determinar o domínio e o escopo da ontologia”. Ou seja, o engenheiro do conhecimento deve responder os seguintes questionamentos:

1. Quais características de vinho eu devo considerar quando escolher um vinho?
2. Bordeaux é um vinho tinto ou branco?
3. Cabernet Sauvignon vai bem com frutos do mar?
4. Qual a melhor escolha de vinho para carne grelhada?
5. Quais as características de um vinho que afetam sua relação com um prato (comida)?
6. O sabor ou corpo de um vinho específico muda com o ano da safra?
7. Qual foi a melhor safra do Napa Zinfandel?

A partir dessas questões conclui-se que a ontologia ira cobrir o domínio de “vinhos e comidas”; a ontologia será utilizada para as aplicações que desejem “sugerir boas combinações de vinhos e alimentos”; as questões que a ontologia deve responder

serão as questões de competência destacadas anteriormente, por fim, será utilizada por clientes de um restaurante e será mantida pelos proprietários desse restaurante.

A etapa “Considerar o reuso de ontologias” consiste em averiguar se alguém já desenvolveu ontologias a cerca do domínio que se deseja trabalhar [28], a fim de não se “reinventar a roda”. Existem diversas bibliotecas de ontologias reutilizáveis na Web e um local adequado para encontrar ontologias para se reutilizar é o Swoogle [34], um motor de busca para acesso a ontologias desenvolvido pela Google, e conta com cerca de 10.000 ontologias armazenadas em seu repositório [38].

Nesta etapa, pode-se observar que embora a técnica proponha a realização do reuso, a mesma não define um formalismo a ser seguido pelo engenheiro do conhecimento, ocasionando a criação de ontologias cada vez mais heterogêneas, pois o conhecimento já formalizado não é aproveitado pelas ontologias em construção, aumentando assim os custos com desenvolvimento e manutenção.

Em seguida, a etapa “Enumerar termos importantes” especifica que o desenvolvedor deve criar uma lista de todos os termos possíveis presentes no domínio [28]. Santos [38] ressalta que no desenvolvimento dessa lista, o desenvolvedor não deve se preocupar com possíveis redundâncias entre os conceitos que eles possam representar, ou que tipo de relações existem entre os termos, ou que características eles possuem ou se estes termos são classes ou propriedades.

De acordo com o domínio obtido na primeira etapa, podem ser encontrados os seguintes termos: uva, vinícola, localização, cor do vinho, corpo do vinho, sabor do vinho, quantidade de açúcar, vinho branco, vinho tinto, vinho Bordeaux, frutos do mar, peixe, carne, vegetais, queijo, fabricante, região de origem, coloração e tipo de comidas.

Posteriormente, é realizada a etapa “Definir classes e hierarquia de classes” que consiste na definição das classes e subclasses da ontologia [28]. Para isso, dos termos obtidos na etapa anterior são selecionados somente aqueles que descrevem um conjunto de objetos, estes termos são definidos como classes. Em seguida, para a obtenção das hierarquias é analisado se possíveis instâncias de uma classe são necessariamente instâncias de outra classe.

O exemplo a seguir utiliza como base os termos obtidos a partir da etapa anterior: Vinho Branco, Vinho Rosé e Vinho são definidos como classes, pois são termos

que definem um conjunto de objetos. A hierarquia dessas classes pode ser encontrada da seguinte forma: uma instância de Vinho Branco é uma instância de Vinho Rosé? Não, então Vinho Branco não é subclasse de Vinho Rosé; Uma instância de Vinho Branco é uma instância de Vinho? Sim, então Vinho Branco é subclasse de Vinho. Essa pergunta deve ser realizada a todas as classes da ontologia.

Em seguida, na etapa “Definir as propriedades das classes” são definidos os atributos e relacionamentos da ontologia [28]. Para isso, são utilizados os termos obtidos na etapa “Enumerar termos importantes”. Por exemplo, os atributos da classe vinho seriam: cor, quantidade de açúcar, preço e idade. Já as relações seriam entre vinho e região, através da relação “feito em”.

Santos [38] ressalta que é mais apropriado colocar as propriedades identificadas na classe mais genérica que possui essa propriedade, ou seja, utilizando-se da característica de herança. Por exemplo, as classes Vinho Tinto e Vinho Rosé possuem a propriedade Idade, sendo mais adequado colocar a propriedade Idade na classe Vinho, pois ambas a herdam.

A etapa “Definir as restrições das propriedades” consiste em definir as restrições de cada propriedade [28]. Uma típica restrição para as propriedades é a cardinalidade, que diz respeito à quantidade de um determinado valor uma propriedade pode receber. Por exemplo, um Vinho pode ter no máximo certa quantidade de álcool. Outra restrição utilizada, é o tipo do valor que uma propriedade pode receber. Por exemplo, o nome do vinho é do tipo string. Os principais tipos são: string, integer, float, boolean.

Por fim, é realizada a etapa “Criar instâncias” que consiste em definir os indivíduos (instâncias) das classes [28]. Para desenvolver essa etapa são necessários três passos: escolher a classe, criar uma instância desta classe e Preencher os valores das propriedades. Por exemplo, criar uma instância da classe Chateau-Morgon-Beaujolais para representar um tipo específico de vinho Beaujolais. Esta instância tem os valores dos slots definidos a seguir: Cor: vermelho, Sabor: delicado, Composto de: Uva Gamay, Feito em: Região Beaujolais.

A técnica é muito intuitiva e de fácil entendimento, formaliza os principais passos para a construção de ontologias de domínio. Define de forma linguística como encontrar conceitos através dos substantivos e os relacionamentos através de

verbos. Contudo, não define formalmente como realizar o reuso de ontologias já existentes. Além de não definir formalmente como identificar as instâncias do domínio em questão.

2.2.2 A Técnica de Uschold e King

Proposta em 1995 foi uma das primeiras técnicas para a construção de ontologias e foi desenvolvida a partir da experiência dos autores no projeto Enterprise Ontology, por meio do Instituto de Aplicações em Inteligência Artificial da Universidade de Edinburgh e parceiros como IBM, Unilever e outros [46]. A seguir, a Figura 2.4 apresenta as cinco etapas que compõem a técnica de Uschold e King: “Identificar o propósito”, “Captura”, “Codifica”, “Integra” e “Avalia”.

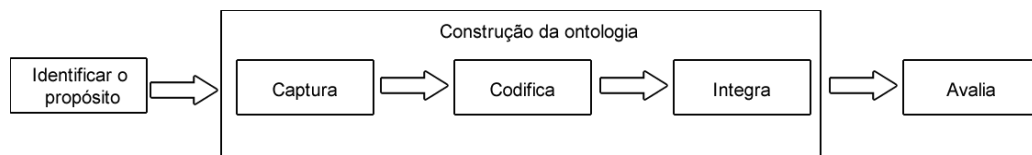


Figura 2.4: Ciclo de vida da técnica de Uschold e King [46]

A primeira etapa “Identificar o propósito” estabelece os principais aspectos sobre a construção da ontologia [46]:

- Definir o domínio da ontologia, como por exemplo: domínio de Viagens;
- Definição do porque construir a ontologia: prover um modelo de conhecimento consensual entre agentes de viagens;
- Quais os seus usos previstos: o compartilhamento de conhecimento;
- Fazer um levantamento dos termos mais relevantes do domínio: locais, tipo de locais, hospedagem, tipo de hospedagens, hotel, pousada, trens, ônibus, ônibus local, meios de transporte.

A partir dos termos identificados na atividade anterior, a etapa “Captura da ontologia” define os termos que estão associados a conceitos e a relacionamentos e ainda realiza uma definição textual, precisa e não ambígua dos conceitos e relacionamentos [46]. Santos [38] ressalta que estas definições textuais não seguem

o estilo clássico de dicionários, sendo construídas criando uma referência a outros termos (termos destacados em negrito), incluindo noções de classes e relações. A seguir, é apresentado um pequeno exemplo:

- Meios de Transporte: É uma classe. Todo meio de transporte tem um **ponto de partida**.
- Ônibus: É uma classe. É um **tipo de meio de transporte**.
- Ônibus local: É uma classe. É um **tipo de ônibus**, que tem **local de partida** e **local de chegada** e as **paradas** na mesma **localização**.

Uschold e King [46] definem três abordagens para o desenvolvimento das hierarquias de classes: “Top-down”, “Bottom-up” e “Middle-out”.

- **Top-down:** o processo se inicia pela definição dos conceitos mais genérico do domínio, os quais vão sendo especializados. Por exemplo, defini-se algo bem genérico como “Objeto”. Então se especializa para “Objeto Concreto” e “Objeto Abstrato”, e se especializa mais ainda de “Objeto Concreto” para “Taxi”, “Ônibus” e “Trem”, já “Objeto Abstrato” se especializa para “Transporte por Trem”, “Transporte por Taxi” e “Transporte por Ônibus”.
- **Bottom-up:** o processo de desenvolvimento começa com a definição dos conceitos mais específicos e depois generaliza-se. Por exemplo, define-se algo bem específico como “Metrô de Londres”, “Ônibus Local de Londres”, “Taxi de Londres”, “Metrô de Madri”, “Ônibus local de Madri” e “Taxi de Madri”, sendo então generalizados para “Metrô”, “Ônibus Local” e “Taxi”, por fim, generaliza-se para “Meios de Transporte”.
- **Middle-out:** é uma combinação das abordagens Top-down e Bottom-up. Definem-se os conceitos mais importantes primeiro e depois estes vão sendo generalizados e especializados. Por exemplo, descobrem-se inicialmente os conceitos “Metrô”, “Ônibus” e “Taxi”, em seguida, generaliza-se para “Meios de Transporte”, por fim, especializa-se “Ônibus” para “Ônibus Local” e “Ônibus de Passeio”. Essa abordagem é considerada a melhor, pois permite um maior controle do nível de detalhes da hierarquia.

Em relação a etapa “Codificação” não há uma definição formal de como ser realizada. Contudo, são apresentados alguns critérios a serem seguidos para uma boa escolha da linguagem de representação de ontologias a ser adotada:

- Distância conceitual: é o esforço necessário para passar de um modelo para outro;
- A linguagem está de acordo com algum padrão, por exemplo: a linguagem OWL segue os padrões do W3C;
- Portabilidade: ao desenvolver em uma linguagem é fácil migrar para outra, por exemplo: representações definidas em RDF podem ser transformadas em OWL, já o inverso pode não acontecer em alguns casos;
- Existe alguma técnica ou algum guia para a utilização dessa linguagem, por exemplo: Linguagens como RDF e OWL disponibilizam guias e manuais para desenvolvedores;
- A base de usuários que usam essa linguagem deve ter um tamanho razoável;

A etapa “Integração” consiste em integrar ontologias existentes [46], reutilizando o conhecimento já formalizado. Entretanto, assim como na técnica 101 [28], não há uma definição formal de como ser realizado o reúso. É apenas descrito a sua possibilidade.

Por fim, a etapa de “Avaliação” é realizada com intuito de verificar o quão válida é a ontologia resultante [46]. Essa verificação pode ser realizada através de uma comparação com uma ontologia de referência, calculando-se o percentual de similaridade entre os seus elementos.

A técnica não define como identificar propriedades, axiomas e instâncias. Além de não definir formalmente como realizar o reúso de ontologias já existentes. De forma geral, não descreve de uma maneira precisa a execução das diferentes etapas do processo de desenvolvimento, apresentando somente princípios gerais muito vagos.

2.2.3 A Técnica Methontology

Baseada no padrão IEEE para o desenvolvimento de software, a técnica Methontology foi proposta por Mariano Fernández, Asunción Gómez Pérez e Natalia Juristo no Laboratório de Inteligência Artificial da Universidade Politécnica de Madrid na Espanha em 1997 [12] [15].

A abordagem possui um ciclo de vida baseado na evolução de protótipos para o desenvolvimento de ontologias, permitindo adicionar, mudar ou remover termos em cada nova versão. A seguir a Figura 2.5 apresenta as atividades que compõe a abordagem Methontology.

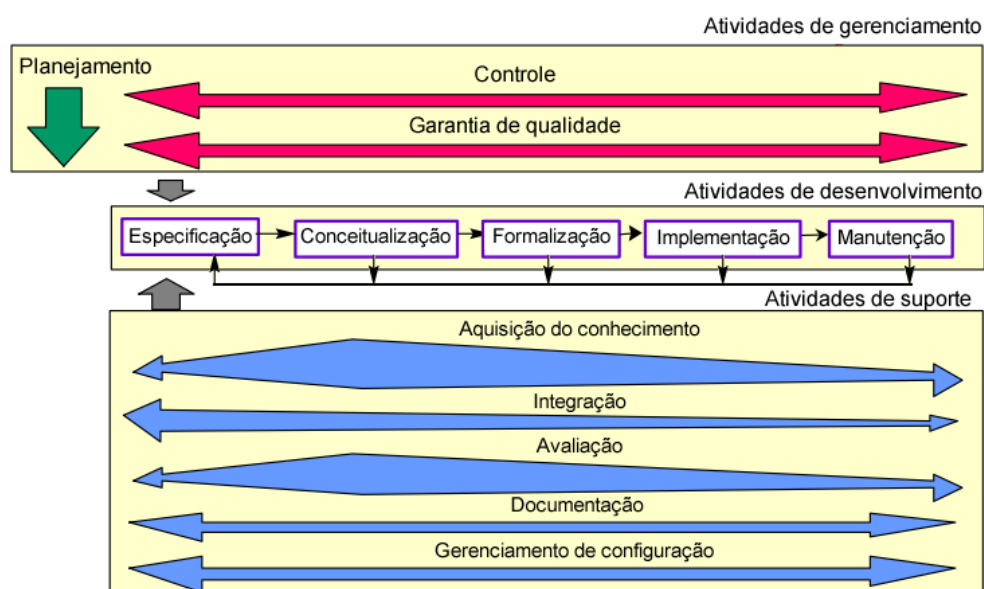


Figura 2.5: Ciclo de vida da técnica de Methontology [12]

A técnica está dividida em três atividades: “Gerenciamento”, “Desenvolvimento” e “Suporte”. A seguir será apresentado uma breve descrição de cada uma delas, assim como, de suas subatividades:

- A atividade “Gerenciamento” possui três subatividades: “Planejamento”, “Controle” e “Garantia de qualidade”. A seguir é descrito cada uma delas:
 - A subatividade “Planejamento” define as principais tarefas a serem realizadas, o cronograma a ser seguido, os recursos necessários (pessoas, software e hardware) e como essas tarefas serão organizadas;
 - A subatividade “Controle” garante a conclusão das tarefas conforme o pretendido, procurando sempre seguir o cronograma;

- A subatividade “Garantia de qualidade” verifica a qualidade de cada produto desenvolvido, por exemplo, a ontologia e a documentação.
- A atividade “Desenvolvimento” possui cinco subatividades: “Especificação”, “Conceitualização”, “Formalização”, “Implementação” e “Manutenção”. A seguir é descrito cada uma delas:
 - A subatividade “Especificação” é responsável por elaborar um documento descrevendo o propósito e o escopo da ontologia;
 - A subatividade “Conceitualização” é considerada a atividade mais relevante e consiste na estruturação do conhecimento do domínio em um modelo conceitual, nesta etapa são descobertos os elementos (classes, relações, axiomas, etc) que compõe a ontologia e são estruturados através de gráficos e tabelas;
 - A subatividade “Formalização” utiliza o modelo conceitual desenvolvido anteriormente e o representa em uma linguagem formal, por exemplo, em Lógica de Primeira Ordem ou Lógica de Descrição;
 - A subatividade “Implementação” utiliza o modelo desenvolvido anteriormente e realiza sua tradução para alguma linguagem de ontologias, por exemplo, OWL (Web Ontology Language) [25];
 - A subatividade “Manutenção” consiste em alterações quando necessárias, para possíveis melhorias ou correções.
- A atividade “Suporte” possui cinco subatividades: “Aquisição de conhecimento”, “Integração”, “Avaliação”, “Documentação” e “Gerenciamento”. A seguir é descrito cada uma delas:
 - A subatividade “Aquisição de conhecimento” é responsável por adquirir o conhecimento sobre o domínio da ontologia, essa aquisição pode ser realizada através de entrevistas com o especialista do domínio;
 - A subatividade “Integração” procura integrar o máximo possível de ontologias existentes à nova ontologia, evitando o trabalho repetitivo.
 - A subatividade “Avaliação” busca realizar revisões no que está sendo desenvolvido, além de efetuar alguns testes, como por exemplo, integridade semântica e léxica;

- A subatividade “Documentação” registra oficialmente os detalhes de cada etapa concluída e o produto gerado;
- A subatividade “Gerenciamento” arquiva as versões da ontologia e da documentação para controlar as mudanças, realiza um controle de versão e de distribuição da ontologia.

A técnica apresenta um bom formalismo, definindo especificamente cada uma de suas fases. Entretanto, embora exista uma subatividade destinada a integração com outras ontologias, nesta etapa não há uma definição formal de como esta operação deve ser realizada.

2.2.4 A Técnica GAODT

GAODT (Goal-Oriented Application Ontology Development Technique) é uma técnica orientada por objetivos para a construção de ontologias de aplicação que traduz objetivos e fatos em linguagem natural e que expressam os requisitos de um SBC, para regras e fatos em Lógica de Primeira Ordem – LPO. Estas regras são mapeadas para uma ontologia de aplicação [39].

A Figura 2.6 ilustra a visão geral da técnica GAODT juntamente com suas quatro atividades: “Seleção dos Objetivos e Fatos”, “Representação dos Predicados em LPO”, “Especificação dos Axiomas em LPO” e “Especificação/Extensão da Ontologia de Aplicação”.

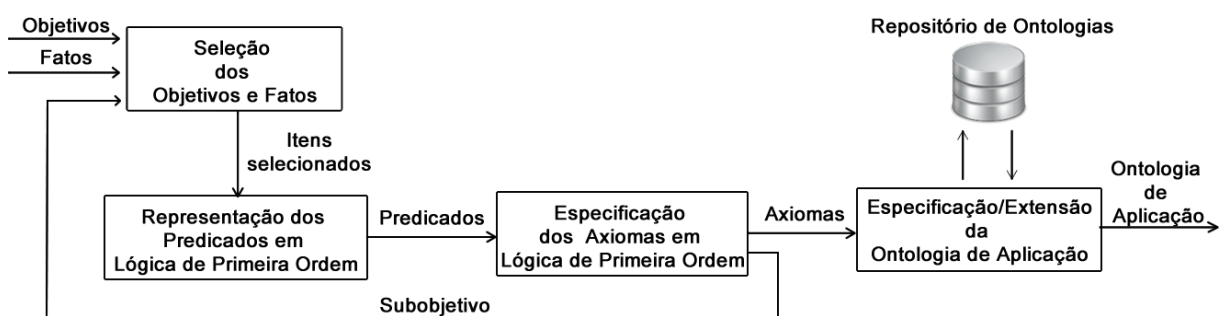


Figura 2.6: Ciclo de vida da técnica de GAODT [39]

O desenvolvedor da ontologia de aplicação e o especialista do domínio participam da execução destas atividades. O desenvolvedor é o engenheiro do conhecimento responsável pela construção da ontologia de aplicação. O especialista do

domínio é alguém que detém informações específicas daquela área de conhecimento [39].

O processo inicia a partir da entrada de um conjunto de objetivos e fatos sobre um determinado domínio de aplicação, que são selecionados pelo engenheiro do conhecimento e pelo especialista do domínio [39]. Por exemplo, o objetivo “Determinar o genitor de uma pessoa” para ser alcançado necessita dos fatos “Uma pessoa tem pai” e “Uma pessoa tem mãe”.

Na atividade “Seleção de Objetivos e Fatos” o especialista do domínio juntamente com o engenheiro do conhecimento selecionam a partir de um determinado domínio de aplicação os objetivos e fatos escritos em linguagem natural a serem utilizados como entrada na próxima fase.

A atividade “Representação dos Predicados em Lógica de Primeira Ordem”, que consiste em transformar os objetivos e fatos inseridos em linguagem natural para linguagem de máquina. Nesta fase é realizada uma interpretação dos predicados para LPO [39].

A atividade “Especificação dos Axiomas em Lógica de Primeira Ordem” recebe como entrada os predicados especificados na atividade anterior e especifica em LPO as regras necessárias para alcançar os objetivos do sistema. Esta atividade é iterativa, isto é, um predicado objetivo pode necessitar de outros subobjetivos ou fatos para satisfazê-lo [39]. Por exemplo, para satisfazer o objetivo “Determinar os ascendentes de uma pessoa” o subobjetivo “Determinar o genitor de uma pessoa” deve ser alcançado. Todo o processo é executado iterativamente até que todos os objetivos tenham sido satisfeitos.

A atividade “Especificação/Extensão da Ontologia de Aplicação” utiliza os axiomas gerados na atividade anterior e extrai a partir deles os elementos necessários para compor a ontologia de aplicação. A ontologia de aplicação criada pode ser ainda estendida através da realização de uma busca semântica em um repositório de ontologias de aplicação. É válido ressaltar, que embora Santos [39] descreva que nesta fase a ontologia de aplicação possa ser estendida, não há definição formal de como este procedimento deve ser realizado, sendo abstraído pelo desenvolvedor.

A técnica apresenta um bom formalismo, definindo especificamente cada uma de suas fases. Entretanto, embora exista uma atividade destinada a extensão da

ontologia a ser gerada, nesta etapa não há uma definição formal de como esta operação deve ser realizada.

2.2.5 Considerações sobre as Técnicas para Construção de Ontologias

A Tabela 2.1 apresenta o resultado de uma análise comparativa entre as mais representativas técnicas para a construção de ontologias de acordo com os seguintes critérios: o tipo da ontologia desenvolvida, a possibilidade do reuso de ontologias existentes e a utilização de conhecimento linguístico para encontrar classes e relacionamentos.

Tabela 2.1: Comparativo entre as Técnicas para Construção de Ontologias

	Técnica 101	Técnica de Uschold e King	Técnica Methontology	GAODT
Tipo de Ontologias Desenvolvida	Ontologia de Domínio	Ontologia de Domínio	Ontologia de Domínio	Ontologia de Aplicação
Utiliza o Reúso	sim	sim	sim	sim
Utiliza o conhecimento Linguístico	sim	não	não	sim

As ontologias estão classificadas em quatro tipos [17]: genéricas, domínio, tarefa e aplicação. A técnica escolhida deve ser apropriada pra a construção de um tipo particular de ontologia. Entre as técnicas apresentadas neste trabalho somente GAODT apóia o desenvolvimento de ontologias de aplicação. As demais técnicas apresentadas descrevem procedimentos para a criação de ontologias de domínio.

Somente as técnicas 101 e GAODT definem regras para encontrar classes e relacionamentos através de substantivos e verbos respectivamente. Esta é uma vantagem por permitir a fácil identificação dos elementos da ontologia através de estratégias linguísticas. Outra vantagem do uso de conhecimento linguístico é a possível automação da extração destes elementos usando técnicas de Processamento da Linguagem Natural - PLN [38].

Boas técnicas devem considerar o reúso de ontologias existentes. Das técnicas analisadas, todas abordam o reúso, alguma destas ainda informam boas fontes de ontologias reusáveis. Porém, nenhuma destas técnicas definem formalmente como o reúso deve ser realizado de fato, tornando o trabalho do engenheiro do conhecimento inconcebível, pois não definem os procedimentos essenciais para a realização do reúso.

2.3 Reúso de Ontologias

O reúso de ontologias é um processo em que o conhecimento ontológico existente é usado como entrada para gerar novas ontologias [42]. O compartilhamento e reutilização de ontologias aumenta tanto a qualidade das aplicações que as utiliza como o nível de interoperabilidade destas aplicações, além do mais, aumenta-se o grau de satisfação da ontologia resultante perante a comunidade pertencente ao domínio na qual faz parte. Noy [28] ressalta ainda, que é aconselhável verificar a existência de ontologias que podem ser reutilizadas em um novo projeto, a fim de não se “reinventar a roda”

A reutilização reduz os custos relacionados ao desenvolvimento de ontologias, pois evita a implementação de componentes ontológicos já disponíveis em algum repositório. O reúso potencialmente aumenta a qualidade das ontologias reutilizadas, pois continuamente essas ontologias são sempre revisadas e avaliadas ao fim do processo de reúso realizado pelo engenheiro do conhecimento [42].

Em contra partida, a maioria das abordagens para construção de ontologias mesmo que ressaltam alguma atividade ou fase destinada ao reúso, não disponibilizam formalmente, detalhes de como o reúso deve ser realizado de fato. Ou seja, mesmo que tais abordagens destaquem que o reúso deve ser realizado e que o mesmo é de extrema importância para um bom projeto de construção de ontologias, tornam o trabalho do engenheiro do conhecimento inconcebível, pois não definem os procedimentos essenciais para a realização do reúso.

Consequentemente, abordagens que visam prover o reúso de ontologias surgem com o intuito de sanar essa carência encontrada entre as técnicas para construção de ontologias. Na literatura [22] [29] [34] [35] [41] pode-se observar diversas definições para tais abordagens, que ora resultam em uma nova ontologia

independente das ontologias reusadas, ora preservam a dependência com estas ontologias, e em algumas vezes, geram somente vínculos semânticos entre as ontologias dadas como entrada. Essas abordagens que visam o reúso de ontologias recebem diferentes denominações e interpretações, as mais comuns são: alinhamento, integração e junção.

Para este trabalho foram adotadas as definições encontradas em [22] [29] [34] [35] [41]. As Seções 2.3.1, 2.3.2 e 2.3.3 apresentam as principais características e peculiaridades de cada uma delas. A Seção 2.3.4 apresenta uma análise comparativa entre essas abordagens. A seção 2.3.5 apresenta a abordagem “Mapeamento” utilizada pelas demais técnicas com o intuito de identificar elementos correspondentes. A seção 2.3.6 apresenta as ferramentas que dão suporte ao reúso de ontologias. Por fim, a seção 2.3.7 traz as considerações sobre as técnicas para o reúso.

2.3.1 Alinhamento

O alinhamento é um processo que visa gerar um “vínculo físico” entre duas ontologias. Para que isso ocorra são dadas como entrada duas ontologias a serem alinhadas. Em seguida, seus termos são mapeados na busca por elementos correspondentes. Após a identificação de similaridade entre os elementos das duas ontologias, são gerados vínculos que representam essa correspondência, definindo qual elemento de uma ontologia é similar ao elemento da outra [29].

De acordo com Fellicissimo [10] alinhamento de ontologias é o processo pelo qual diversas aplicações, que utilizam diferentes ontologias terão que se submeter para garantir uma representação intermediária da informação, que poderá ser compartilhada entre elas. O resultado deste processo é um modelo persistente que estabelece a ligação entre as ontologias alinhadas, permitindo o compartilhamento e a reutilização de suas informações.

Segundo Silva [41] o alinhamento de ontologias tem se mostrado uma solução razoável para problemas presentes tanto em aplicações tradicionais, como integração de esquemas de bases de dados, como nos emergentes sistemas de compartilhamento de informação peer-to-peer e composição de serviços Web. Áreas como engenharia de ontologias, integração de informações, navegação e consulta

na Web Semântica, entre outras, cada vez mais tiram proveito de operações de alinhamento semântico que visam conciliar modelos estruturalmente heterogêneos.

A Figura 2.7 representa duas ontologias distintas O_1 e O_2 que são dadas como entrada ao processo de alinhamento. Após ser realizado esse processo, são criados vínculos físicos entre os elementos similares de O_1 e O_2 . Desta forma, o elemento “carro” pertencente à O_1 e o elemento “veículo” pertencente à O_2 , por serem similares, possuem um vínculo entre si, possibilitando que consultas realizadas ao elemento “carro” reuses o conhecimento explícito de “veículo”.

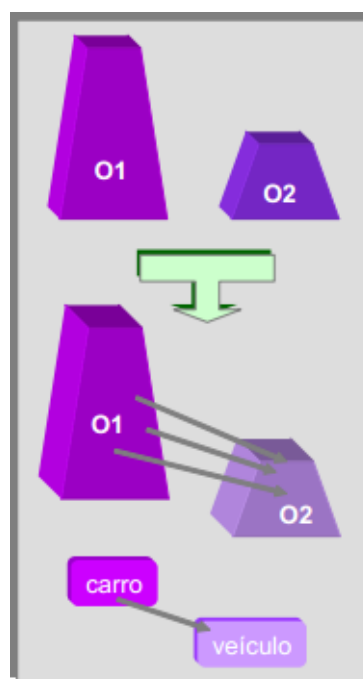


Figura 2.7: Exemplo do alinhamento de ontologias, adaptado a partir de Noy [32]

Ao fim do processo de alinhamento, o resultado obtido são as mesmas ontologias dadas como entrada. Porém, estas ontologias agora possuem uma pequena modificação em seu código, um vínculo físico propriamente dito é criado, possibilitando que o conhecimento explícito nestas ontologias seja reusado.

2.3.2 Integração

Segundo Sowa [43] integração de ontologias é definida como um processo para busca de pontos em comum entre duas diferentes ontologias O_1 e O_2 e deriva-las em uma nova ontologia O_3 . Esse processo facilita a interoperabilidade entre sistemas

de software, que ao utilizarem a ontologia O_3 , possibilita a reutilização de todo o conhecimento ontológico expresso em O_1 e O_2 .

A integração de ontologias não deve ser vista como uma única atividade ou metodologia a ser utilizada em algum momento do ciclo de vida da construção de ontologias, mas, deve ser compreendida como um processo que ocorre ao longo de todo o processo de construção, podendo ocorrer mais do que uma vez. O processo de integração envolve duas ou mais ontologias, denominadas de ontologias de origem. Ao serem integradas derivam uma nova ontologia, denominada de ontologia resultante. A ontologia resultante contém informações que permitem identificar os componentes integrados, assim como suas respectivas ontologias de origem [34].

Esta abordagem possibilita a identificação das ontologias de origem e seus respectivos elementos integrados. O resultado deste processo será uma única ontologia contendo os elementos integrados e seus respectivos “links” que representará uma ponte entre o elemento integrado na ontologia resultante e a ontologia de origem na qual esse elemento fazia parte. Além do mais, outros elementos poderão ser adicionados na ontologia resultante, fazendo com que a mesma seja estendida.

A seguir na Figura 2.8, ilustra um exemplo baseado na ontologia O_1 que representa parte do domínio de aluguel de carros, e a ontologia O_2 que representa parte do domínio de aluguel de veículos. Neste exemplo o resultado é uma ontologia integrada O_3 , que possui alguns dos elementos contidos em O_1 e em O_2 .

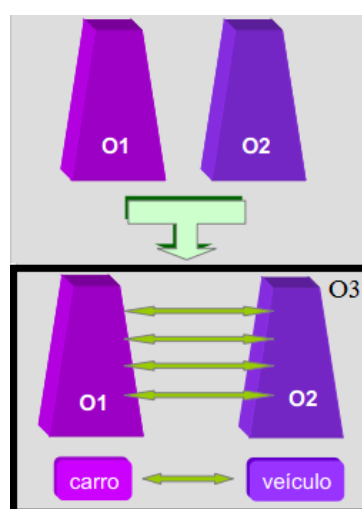


Figura 2.8: Exemplo da integração de ontologias, adaptado a partir de Noy [32]

Ao fim do processo de integração, o resultado obtido é uma única ontologia resultante, capaz de expressar o conhecimento ontológico de ambas as ontologias

dadas como entrada. Além disso, esta ontologia resultante pode ser estendida, adquirindo novos elementos que atenda às necessidades do especialista do domínio.

2.3.3 Junção

A Junção é o processo pela qual duas ou mais ontologias são utilizadas e após a realização deste, o resultado final é uma única ontologia. A ontologia resultante não possibilita a identificação da origem de seus respectivos elementos [3].

Esta abordagem muitas vezes pode ser denominada de Combinação [22] [3], e embora resulte em uma única ontologia, se diferencia do processo de integração por não possuir vínculo algum com as ontologias de origem. Dessa maneira, enquanto no alinhamento é criado um vínculo físico e externo às ontologias de origem, e na integração é criada uma ontologia intermediária, a Junção por sua vez, gera uma nova ontologia simplesmente para atender a novas necessidades específicas do engenheiro do conhecimento e do especialista de domínio.

Conseqüentemente, uma ontologia combinada se torna completamente independente de suas origens, ou seja, durante o processo de Junção, a única preocupação dos seus desenvolvedores é apenas em reusar o que já existe e atender aos requisitos iniciais do processo de criação, indo muitas vezes em sentido contrário ao que foi definido por Gruber [16], uma “conceituação compartilhada”, podendo não ser adequada aos demais interesses da comunidade e limitando assim o reúso dessa ontologia.

Porém, para objetivos específicos, essa liberdade pode ser um ponto forte, como por exemplo, no desenvolvimento de aplicações que utilizam ontologias como forma de representação do conhecimento ou em ambientes proprietários, onde o uso conseqüentemente é restrito e não compartilhado ou seja específico para determinado domínio, a representação deve ser expressiva, mas limitada ao local para a qual foi projetada.

A seguir na Figura 2.8, ilustra um exemplo baseado na ontologia O_1 que representa parte do domínio de aluguel de carros, e a ontologia O_2 que representa parte do domínio de aluguel de veículos. Neste exemplo, o resultado é uma ontologia combinada O_3 , que possui alguns dos elementos contidos em O_1 e em O_2 .

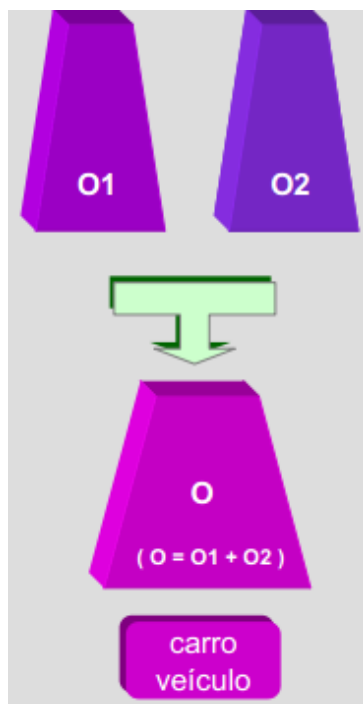


Figura 2.9: Exemplo da junção de ontologias, adaptado a partir de Noy [32]

Ao fim do processo de junção, o resultado obtido é uma única ontologia resultante, capaz de expressar o conhecimento ontológico de ambas as ontologias dadas como entrada. Esta ontologia não possui vínculos com as ontologias iniciais, tornando-a completamente independente. Além do mais, esta ontologia resultante pode ser estendida, adquirindo novos elementos que atenda às novas necessidades do especialista do domínio.

2.3.4 Análise Comparativa entre as Abordagens

A Tabela 2.2 apresenta o resultado de uma análise comparativa entre as diferentes abordagens que visam realizar o reúso de ontologias. A análise leva em consideração os seguintes critérios: o tipo de entrada, o tipo de saída e se possui vínculo entre entrada e saída.

Na abordagem alinhamento duas ou mais ontologias são utilizadas como entrada o produto do alinhamento não resulta em uma nova ontologia, mas, em um artefato “fisicamente separado”, capaz de expressar um conjunto de elementos correspondentes, vinculando-os. Esse artefato é utilizado para centralizar o conhecimento expresso em várias ontologias, reduzindo assim o nível de heterogeneidade das mesmas.

Na abordagem integração duas ou mais ontologias são utilizadas como entrada, após a realização deste procedimento o resultado final é uma nova ontologia. Nesta ontologia resultante é possível identificar os elementos oriundos das ontologias dadas como entrada.

Na abordagem junção duas ou mais ontologias são utilizadas como entrada, após a realização deste processo o resultado final é uma nova ontologia. Nesta ontologia resultante não há identificação da origem de seus elementos, tornando-a independente das ontologias de origem.

Tabela 2.2: Comparativo entre as abordagens para o reúso de ontologias

	Alinhamento	Integração	Junção
Entrada	Duas ou mais Ontologias	Duas ou mais Ontologias	Duas ou mais Ontologias
Saída	As mesmas Ontologias	Uma nova e única Ontologia	Uma nova e única Ontologia
Vínculos com as Ontologias de origem	Sim	Sim	Não

2.3.5 Mapeamento

Segundo Campos et al. [2] o mapeamento é um conjunto de regras ou procedimentos que estabelecem de que maneira os elementos de uma ontologia devem corresponder-se aos elementos de uma outra, fazendo uso de alguma medida de similaridade. Desta forma, o mapeamento é um procedimento fundamental a ser utilizado pelas abordagens que visam o reúso de ontologias, pois somente através deste procedimento é possível identificar os elementos a serem reusados de forma automática.

De acordo com Lachtim et al. [22] a busca pela similaridade entre elementos é essencial em abordagens que visam o estabelecimento de uma relação formal entre entidades de duas ontologias, tais como alinhamento, junção e integração. Quanto mais aprimorado for este mecanismo, melhores são os resultados obtidos pelas técnicas que provêm o reúso, propiciando maior grau de automatização do processo. Esta seção apresenta um conjunto de características encontradas nessas técnicas, que segundo

Shvaiko e Euzenat [9] são divididas em técnicas de similaridade no nível de elementos e técnicas de similaridade no nível de estruturas.

Técnicas de Similaridade no Nível de Elementos

A nível de elemento as técnicas de similaridade realizam um mapeamento analisando as entidades isoladamente, descartando quaisquer que seja seus relacionamentos com outras entidades. A seguir são apresentadas alguma dessas técnicas:

- **Técnicas baseadas em strings:** Esta abordagem é utilizada geralmente para casar nomenclaturas de entidades, considerando que quanto mais similares são as cadeias de caracteres, maior é a similaridade entre essas entidades.
- **Técnicas baseadas em linguagens:** Esta abordagem utiliza o PLN, explorando as propriedades morfológicas dos termos, como por exemplo: ao analisar os nomes das entidades, são eliminados os espaços em branco, preposições, entre outros, resultando somente uma substring principal que então será comparada.
- **Recursos linguísticos:** Esta abordagem utiliza dicionários de domínio na tentativa de combinar palavras baseando-se nos relacionamentos linguísticos entre elas (sinônimos, hipônimos ou hiperônimos). O Wordnet [27] é um exemplo de dicionário lexical que pode ser utilizado por estas técnicas.
- **Técnicas baseadas em restrições:** Esta abordagem realiza uma comparação analisando certas restrições aplicadas às definições das entidades, como seus tipos de dados, cardinalidades e instâncias.

Técnicas de Similaridade no Nível de Estrutura

A nível de estrutura as técnicas de similaridade realizam um mapeamento analisando as entidades de acordo com suas relações, a seguir são apresentadas algumas dessas técnicas:

- **Técnicas baseadas em grafo:** Esta abordagem utiliza algoritmos que consideram as ontologias como grafos rotulados. Cada grafo é dividido em sub-grafos

menores e comparados com os sub-grafos da ontologia oposta. Caso um nó seja identificado como similar a outro, este também será comparado a seus vizinhos.

- Técnicas baseadas em taxonomia: Esta abordagem também consiste em algoritmos baseados em grafo, que levam em consideração apenas a relação de especialização, ou seja, ligações do tipo `kind_of` (`tipo_de`) que conectam a termos já considerados similares, tornando possíveis que seus nós adjacentes também sejam similares.

Como pode ser observado anteriormente, de acordo com a classificação de Shvaiko e Euzenat [9] estas abordagens podem realizar a análise de similaridade entre as ontologias a serem reutilizadas através de uma simples comparação lexical entre os conceitos, até uma análise mais detalhada, examinando a estrutura hierárquica desses conceitos, além de outros elementos que compõem uma ontologia, como propriedades e relações não taxonômicas.

2.3.6 Técnicas e Ferramentas que suportam o Reúso de Ontologias

A busca por automatizar o estabelecimento de correspondências entre elementos diversos de ontologias distintas, ocasionou o surgimento de técnicas e ferramentas que empregam técnicas de similaridade a fim de proverem o suporte à alguma abordagem que visa o reúso de ontologias, seja ela, o alinhamento, a integração ou a junção. Estas ferramentas podem utilizar mais de uma técnica de similaridade, além de darem suporte a mais de uma abordagem que visa o reúso. A seguir são apresentadas algumas delas.

CMS

CMS (CROSI Mapping System) [21] [22] [10] é um sistema que visa realizar o mapeamento entre duas ou mais ontologias, realiza uma captura semântica dos construtores OWL [25], possibilitando com sua arquitetura modular, consultar recursos linguísticos externos para apoiar no processo do cálculo de similaridades, a exemplo do Wordnet [27]. Os procedimentos a serem realizados pelo sistema estão divididos em quatro fases apresentadas a seguir.

- Geração de características: diferentes características de entrada são utilizadas de acordo com os parâmetros de entrada de cada comparação. As características geradas são: URI das classes, classes equivalentes, URI de propriedades, URI das instâncias, super-classes, sub-classes, etc;
- Processamento: a partir da obtenção dos parâmetros de entrada são selecionados mecanismos que visam o cálculo do valor de similaridade entre esses elementos. Para isso, utiliza mecanismos que abordam técnicas de similaridade no nível de seus elementos e de suas estruturas;
- Agregação: a partir do cálculo de similaridade realizado pelos comparadores escolhidos, pois o sistema possui vários, são realizadas combinações entre todas as medidas de similaridade encontradas, determinando um valor global através destas médias;
- Avaliação: a similaridade geral é avaliada para iniciar iterações de backtracking para estágios diferentes, dessa forma, de maneira iterativa múltiplos loops são realizados, retornando para diferentes fases do processo em busca de um valor de similaridade mais conciso.

A ferramenta possui algumas características funcionais que devem ser observadas, como por exemplo:

- Tipo de entrada: utiliza com entrada ontologias expressas na linguagem OWL;
- Automação: a ferramenta realiza o processo de mapeamento automaticamente (usuário realiza somente a seleção das ontologias a serem utilizadas).
- Disponibilidade: a ferramenta está disponibilizada em forma de uma API/Java.

FCA MERGE

Esta abordagem tem como objetivo final a combinação de duas ontologias [44] [22]; possui como entrada duas ontologias e um corpus. Este corpus deve ser relevante para as ontologias envolvidas no processo, visto que é descrito a partir dos conceitos dessas ontologias.

Do corpus apresentado são extraídas instâncias para ambas as ontologias através de técnicas de PLN. Esta aquisição automática de conhecimento gera, para cada ontologia, um contexto formal que indica quais conceitos das ontologias aparecem em quais documentos. Esse processo é dividido em três fases [44]:

- Extração das instâncias de um corpus D relacionado a um domínio específico aplicando técnicas de PLN e geração de dois contextos formais K_1 e K_2 . O objetivo deste primeiro passo é gerar, para cada ontologia, um contexto formal, $K=(G,M,I)$, onde G é um conjunto de objetos; M um conjunto de atributos, e I uma relação binária entre G e M . Assim, a expressão $(g,m) \in I$, indica que o objeto g possui o atributo m .
- Em seguida, a entrada é representada por dois contextos formais gerando como estrutura resultante um grafo de conceitos (pruned concept lattice). Este é um grafo orientado, indicando a interconexão dos conceitos dos dois contextos formais que devem ser unidos em um novo contexto formal.
- O resultado produzido é explorado e transformado em uma ontologia combinada por um engenheiro do conhecimento. Esta ontologia combinada é baseada nos conceitos do grafo: nas fases anteriores o processo é completamente automático, enquanto nesta última fase é necessária a intervenção humana para a derivação da ontologia final.

Em relação às técnicas de similaridades aplicadas, utiliza uma base de dados léxica para associar palavras únicas ou expressões compostas com conceitos das ontologias, além de utilizar PLN para a identificação de instâncias. Por fim utiliza o algoritmo TITANIC [22] [44] para relacionar os conceitos e suas hierarquias.

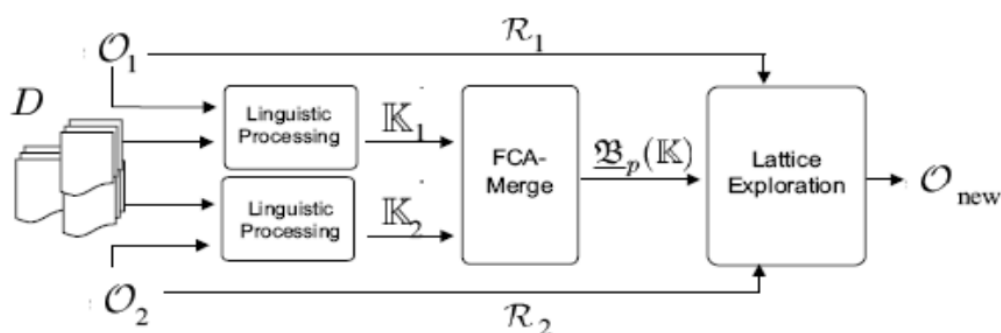


Figura 2.10: Ciclo de vida do processo FCA-MERGE [44]

CATO

CATO (Componente para Alinhamento Taxonômico de Ontologias) [10] [11] [22] é uma abordagem desenvolvida na Pontifícia Universidade Católica do Rio de Janeiro, com o objetivo de alinhar automaticamente as taxonomias das ontologias de entrada a serem comparadas. O alinhamento realizado é dividido em três etapas:

- A primeira etapa compara lexicamente os conceitos das ontologias de entrada e usa um mecanismo de comparação de conceitos baseado em grafo;
- A segunda etapa compara estruturalmente as hierarquias das ontologias, identificando as similaridades entre suas sub-árvores comuns;
- A terceira etapa refina os resultados da etapa anterior, classificando os conceitos identificados como: similares, bem similares ou pouco similares, de acordo com um percentual de similaridade pré-definido.

Ao fim da realização destas etapas, as informações de equivalência dos conceitos similares são adicionadas às ontologias resultantes da primeira etapa. Após esta adição, as ontologias alinhadas são unidas em uma única ontologia. Esta ontologia é o resultado final da abordagem.

É importante ressaltar que, conforme descrito na seção 2.3, quando uma abordagem para o reúso resulta em uma única ontologia é característica da integração e/ou combinação de ontologias. Porém, esta abordagem é definida por seus autores [10] [11] [22], como uma técnica voltada para o alinhamento de ontologias, pois seu produto final possui um conjunto de vínculos que identificam sua ontologias de origem. Contudo, nas considerações finais sobre as técnicas e ferramentas que suportam o reúso esta técnica será definida como técnica para a integração de ontologias, pois seu conjunto de vínculos pode estar presente em uma ontologia integrada.

Para dar suporte à técnica existe uma ferramenta web, que recebe como entrada duas ontologias em OWL, onde realiza a análise de similaridade através de uma comparação lexical e faz uso de algoritmos como: TreeDiff [49] [4], TreeToTree [45] [4], TreeMatche [49] [4] para comparação hierárquica dos conceitos. Resulta como

saída uma única ontologia no formato OWL. A ferramenta possibilita ainda que o alinhamento seja realizado de forma manual, semi-automática ou automática.

iPROMPT

O iPROMPT é uma abordagem semi-automática para o mapeamento e combinação de ontologias [32] [31] [4]. A ferramenta recebe como entrada duas ontologias a serem combinadas e resulta em uma única ontologia reúsada.

Após a seleção das ontologias a serem dadas como entrada, a ferramenta fornece ao engenheiro do conhecimento uma lista de sugestões de combinações a serem realizadas, obtidas através do cálculo de similaridade definido pelas medidas de similaridade suportadas pela técnica, que podem ser desde similaridade linguística entre os nomes das classes, até uma análise da estrutura hierárquica destas classes. A ferramenta possibilita ainda que seja utilizado um dicionário lexical como o Wornet [27] para encontrar sinônimos e auxiliar na comparação dos termos.

A lista de sugestões resume-se a termos que podem ser combinados, além de problemas ou inconsistências encontrados e que devem ser resolvidos. Os termos não comuns são sugeridos para serem copiados para a ontologia resultante.

A ferramenta iPROMPT, inicia as combinações por similaridade lingüística e ao decorrer do algoritmo a ferramenta busca encontrar informações baseadas na semântica da ontologia e nas ações do usuário. O resultado é uma nova ontologia com partes das ontologias fontes, das informações obtidas pelo usuário e das interações resultantes das respostas do usuário.

A ferramenta é semi-automática, utilizada como plugin do software Protégé [36]. Faz parte de um pacote de outros softwares voltados para o reúso de ontologia como: Anchor_PROMPT [30] que realiza o alinhamento de ontologias; PROMPTFactor que separa e divide a ontologia em outras ontologias semanticamente independentes e PROMPTDiff que encontra a diferença entre versões da mesma ontologia.

2.3.7 Considerações sobre as Técnicas para o reúso de Ontologias

A Tabela 2.3 apresenta uma comparação entre as técnicas analisadas. A seguir é apresentada uma breve consideração sobre cada uma delas.

As abordagens analisadas suportam em sua maioria o padrão OWL como linguagem para representação de ontologias, porém abordagens como CATO e iPROMPT suportam outras linguagens como o XML e o RDF respectivamente.

Todas as abordagens recebem como entrada duas ontologias a serem reusadas e retornam como resultado uma única ontologia combinada. Contudo, abordagens como a CMS e a CATO resultam em ontologias vinculadas ou seja, possuem vínculos físicos com as ontologias de origem não sendo assim independentes, não possuindo a característica da junção de ontologia. A abordagem CMS se enquadra no alinhamento, enquanto a CATO se enquadra à integração. As demais realizam de fato a junção entre ontologias.

Tanto a CMS como a FCA MERGE retornam ontologias compostas por elementos dos conjuntos C_C , H, I, P. Enquanto CATO e a iPROMPT retornam ontologias compostas por elementos dos conjuntos C_C , P, H, R.

Sobre as técnicas de similaridade adotadas: a abordagem CMS realiza a análise lexical, linguística, estrutural e entre propriedade; a abordagem FCA MERGE realiza a análise lexical e estrutural. A abordagem CATO realiza a análise lexical, linguística e estrutural; a abordagem iPROMPT realiza a análise lexical, linguística e estrutural.

Em relação ao nível de automação, a abordagem CMS é a única que realiza todas as suas fases automaticamente, as demais necessitam ao menos de uma intervenção do usuário, tornando tais técnicas semi-automáticas.

Após esta análise, foi constatada a necessidade de uma abordagem que ao realizar a junção de duas ontologias gere uma única ontologia composta pelos elementos do conjunto C_C , P, H, R e A, pois estes elementos são fundamentais para a criação de ontologias de aplicação. Há também a necessidade de um número maior de técnicas de similaridade a serem adotadas, a fim de estabelecer valores de similaridade mais expressivos, ou seja, quanto mais amplo o conjunto de técnicas adotadas, mais relevante será a precisão da similaridade encontrada.

Tabela 2.3: Comparativo entre as Técnicas e Ferramentas para o Reúso de Ontologias

	CMS	FCA MERGE	CATO	iPROMPT
Conteúdo da Entrada	Duas Ontologias	Duas Ontologias e um Corpus	Duas Ontologias	Duas Ontologias
Conteúdo da Saída	Uma Ontologia Vínculada	Uma Ontologia Independente	Uma Ontologia Vínculada	Uma Ontologia Independente
Elementos Resultantes	C _C , H, I, P	C _C , H, I, P	C _C , H, P, R	C _C , P, H, R
Linguagens Suportadas	OWL	OWL	OWL e XML	OWL e RDF
Técnicas de Similaridade	Lexical, Linguística, Estrutural e Análise de Propriedades	Lexical e Estrutural	Lexical, Linguística e Estrutural	Lexical, Linguística, Estrutural e Análise de Propriedades
Automação	Automática	Semi-automática	Semi-automática	Semi-automática

2.4 Considerações Finais

Neste capítulo foram apresentadas as principais técnicas para a construção e reúso de ontologias, tendo como finalidade a obtenção de embasamento teórico e referencial bibliográfico para a elaboração do processo proposto neste trabalho.

Inicialmente explicou-se os conceitos básicos de ontologia, assim como suas principais técnicas de construção. Ressaltou ainda a importância do reúso perante a engenharia de ontologias, apresentando as características de cada uma de suas principais abordagens.

Em seguida, tendo realizado uma análise, constatou-se a necessidade de um processo capaz de combinar duas ou mais ontologias, que resultasse em uma ontologia independente das ontologias de origem, além de ser completa ou possuir a maior quantidade de elementos que compõem a sua definição formal. Esta constatação será de grande utilidade para o delineamento do processo proposto no capítulo 3.

3 O Processo OntoJoin

O processo OntoJoin tem por objetivo primordial, auxiliar o engenheiro do conhecimento durante a fase do reúso de ontologias. Esta fase pode ser observada em inúmeras técnicas destinadas à construção de ontologias, como por exemplo: as técnicas, 101 [28], Uschold e King [46], Methontology [12] e GAODT [38]. Entretanto, nenhuma dessas técnicas descrevem satisfatoriamente como a fase do reúso deve ser realizada de fato, tornando o processo para a construção e manutenção de ontologias uma tarefa custosa e repetitiva.

O OntoJoin é um processo para a junção de ontologias, que emprega a análise de similaridade lexical, estrutural, relacional e indexação de termos como mecanismo de mapeamento. Esse mecanismo é responsável por identificar correspondências entre elementos de duas ontologias dadas como entrada. Essa correspondência é utilizada para combinar os elementos similares, resultando assim, em uma nova ontologia gerada a partir do reúso.

O processo recebe como entrada duas ontologias a serem combinadas e resulta em uma nova ontologia de aplicação. A ontologia resultante possui elementos contidos nas duas ontologias dadas como entrada. Desta maneira, a primeira ontologia é definida como Ontologia de Origem (O_1), enquanto a segunda é definida como ontologia complementar (O_2). Essas nomenclaturas são bastante utilizadas na literatura e para fins de padronização também foram utilizadas neste trabalho.

A Figura 3.1 ilustra a visão geral do processo OntoJoin, assim como suas cinco fases: “Identificação e Extração dos Elementos das Ontologias”, “Mapeamento”, “Análise de Integridade dos Termos”, “Junção” e “Análise e Representação da nova Ontologia”.

A seção 3.1 refere-se à fase “Identificação e Extração dos Elementos das Ontologias”, na qual é responsável por identificar e extrair os termos que representam cada elemento que compõem as duas ontologias a serem combinadas. A fase resulta em duas listas de termos: “lista de termos de O_1 ” e “lista de termos de O_2 ” de acordo com a respectiva ontologia dada como entrada;

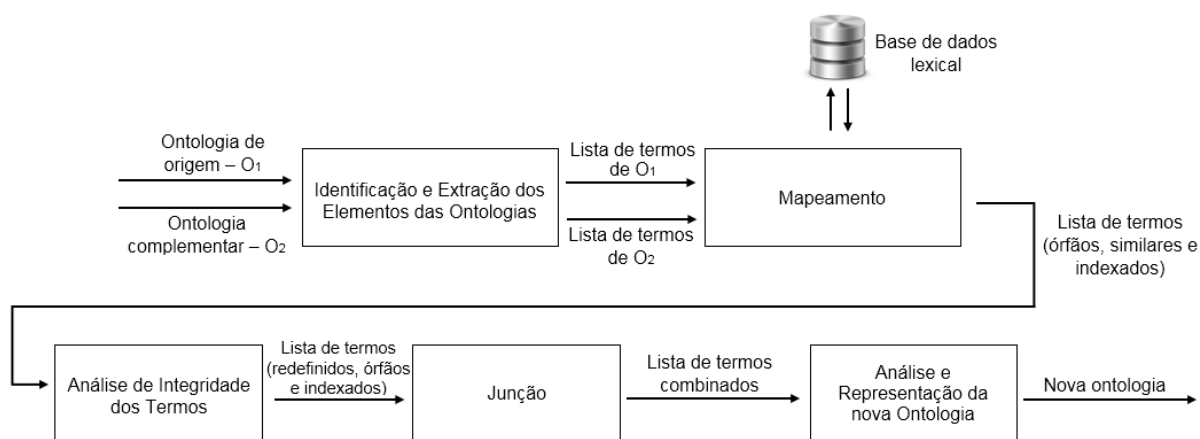


Figura 3.1: Visão geral do processo OntoJoin

A seção 3.2 refere-se à fase “Mapeamento”, na qual é responsável por identificar as correspondências entre os elementos das ontologias a serem reusadas, utiliza uma análise de similaridade entre os termos que correspondem a cada elemento presente nas listas de termos obtidas na fase anterior. A fase resulta em três listas: “lista de termos órfãos”, “lista de termos similares” e “lista de termos indexados”, a primeira destaca uma relação de termos que não obtiveram correspondentes, enquanto a segunda destaca uma relação de termos que obtiveram similaridade, já a última traz os termos do conjunto C_C que tiveram sua estrutura hierárquica modificada.

A seção 3.3 refere-se à fase “Análise de Integridade dos Termos”, na qual possibilita ao engenheiro do conhecimento em conjunto com o especialista do domínio intervir no processo, e escolher ou alterar a nomenclatura dos termos que obtiveram um valor de similaridade aceitável. Além desta intervenção é realizada uma verificação de integridade dos termos a fim de garantir que os mesmos não possuam inconsistências de integridade.

A seção 3.4 refere-se à fase “Junção”, responsável por realizar a combinação entre os termos que compõem as ontologias O_1 e O_2 . Essa fase recebe como entrada três listas de termos: “lista de termos com similaridade aceitável”, “lista de termos indexados” e “lista de termos órfãos”. Como resultado gera uma única lista contendo todos os elementos das ontologias dadas como entrada.

A seção 3.5 refere-se à atividade “Análise e Representação da nova Ontologia”, onde são analisadas as integridades dos elementos que foram combinados. Essa verificação garante que as regras e os demais elementos sejam gerados

corretamente. Tem como resultado uma ontologia gerada em alguma linguagem de representação de ontologias i.e. OWL [25]. Esta ontologia reúne todos os elementos contidos nas ontologias dadas como entrada.

A seção 3.6 refere-se à ferramenta “OntoJoinTool”, uma ferramenta de software semi-automatizada, desenvolvida no âmbito do Gesec¹ com a finalidade de dar suporte ao processo OntoJoin. A ferramenta foi utilizada durante a avaliação deste processo como meio de viabilizar os valores utilizados pelas métricas definidas na seção 4.2.

A seção 3.7 contempla as “Considerações Finais”, concluindo de maneira objetiva as descrições abordadas no presente capítulo.

3.1 Identificação e Extração dos Elementos das Ontologias

Esta fase recebe como entrada duas ontologias a serem combinadas, a primeira delas é definida como ontologia de origem O_1 , a segunda é definida como ontologia complementar O_2 . Em seguida, os termos representativos dos elementos constantes nestas ontologias são extraídos e agrupados de acordo com seu respectivo conjunto (C_C , P , R , H e A) da definição formal de ontologias apresentados na seção 2.1

A Figura 3.2, ilustra a decomposição da fase “Identificação e Extração dos Elementos das Ontologias” em cinco atividades: “Identificação e Extração das Classes”, “Identificação e Extração das Propriedades”, “Identificação e Extração dos Elementos não Taxonômicos”, “Identificação e Extração dos Relacionamentos Taxonômicos” e “Identificação e Extração dos Axiomas”. Estas etapas abordam os diferentes conjuntos de elementos que compõem as ontologias O_1 e O_2 .

A Atividade “Identificação e Extração das Classes” possui como objetivo identificar e extrair os elementos pertencentes ao conjunto C_C , ou seja, as classes pertencentes às ontologias dadas como entrada. Por exemplo: as classes “Pessoa” e “Instituicao”. Além do mais, as classes são elementos chave para a identificação dos

¹Grupo de pesquisa em Engenharia de Software e Engenharia do Conhecimento

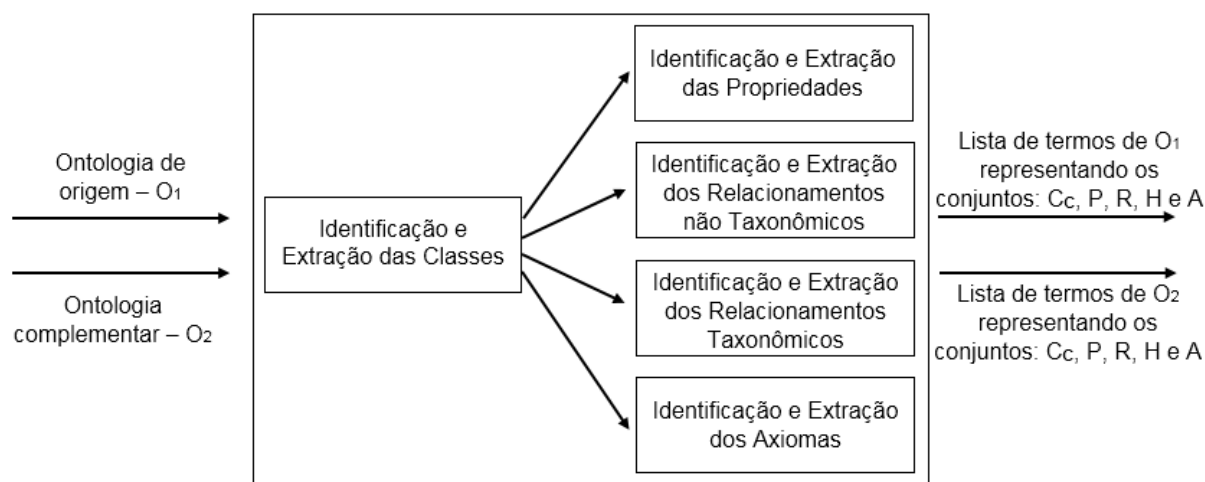


Figura 3.2: Visão geral da fase Identificação e Extração dos Elementos das Ontologias

demais elementos, pois qualquer elemento que esteja contido nos conjuntos P, R, H e A possui alguma relação direta com algum elemento contido no conjunto C_C .

A Atividade “Identificação e Extração das Propriedades” possui como objetivo identificar e extrair os elementos pertencentes ao conjunto P, ou seja, as propriedades pertencentes às ontologias e relacionadas a uma determinada classe C_C . Por exemplo: a classe “Pessoa” possui algumas propriedades a serem identificadas: “nome_completo” e “data_de_nascimento”.

A Atividade “Identificação e Extração dos Relacionamentos não Taxonômicos” possui como objetivo identificar e extrair os elementos pertencentes ao conjunto R, ou seja, os relacionamentos não taxonômicos pertencentes às ontologias dadas como entrada, este relacionamento por sua vez, faz relação a dois elementos do conjunto C_C . Por exemplo: o relacionamento “Ministrar_aula_de” pode ser considerado um relacionamento não taxonômico entre as classes “Professor” e “Disciplina”.

A Atividade “Identificação e Extração dos Relacionamentos Taxonômicos” possui como objetivo identificar e extrair os elementos pertencentes ao conjunto H, ou seja, os relacionamentos taxonômicos pertencentes às ontologias. Por exemplo: a classe “Professor_Adjunto” possui uma superclasse “Professor” e esta por sua vez, possui a superclasse “Pessoa”.

A Atividade “Identificação e Extração dos Axiomas” possui como objetivo identificar e extrair os elementos pertencentes ao conjunto A, ou seja, as

regras pertencentes às ontologias dadas como entrada. Por exemplo: a regra “grandfather_of(X, Z) \Rightarrow fathe_of(X, Y) \wedge father_of(Y, Z)” que define o avô de uma pessoa a partir de duas condições.

As Figuras 3.3 e 3.4 ilustram duas ontologias O_1 e O_2 a serem utilizadas como exemplo nesta atividade. Os itens 1 e 2 demonstram formalmente parte das listas de termos extraídos a partir destas ontologias e agrupados de acordo com seus respectivos conjuntos.

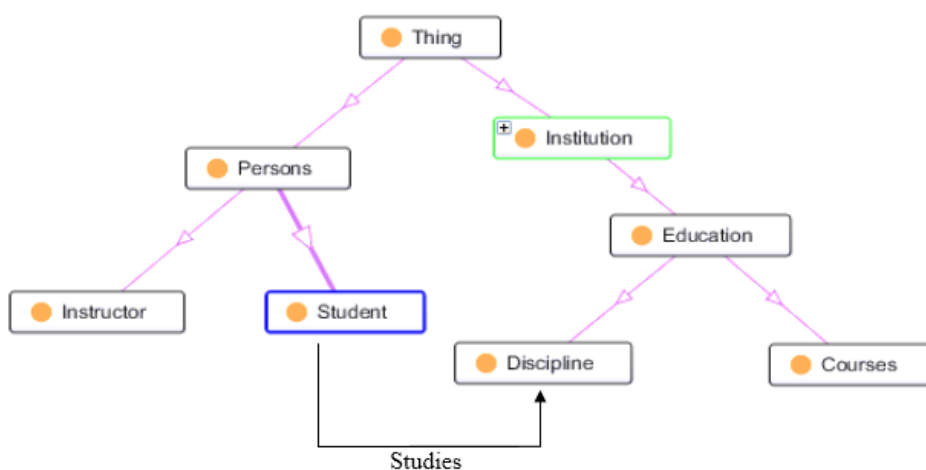


Figura 3.3: Demonstração gráfica de parte da ontologia O_1

1. Ítems extraídos a partir da O_1 :

- (a) Conjunto $C_C = \{\text{Thing, Person, Instructor, Student, Institution, Education, Discipline, Course}\}$;
- (b) Conjunto $P = \{\text{Name(Person, String), Registration(Instructor, String), Registration(Student, String), Name(Institution, String), Name(Discipline, String), Length(Course, String)}\}$;
- (c) Conjunto $R = \{\text{Studies(Student, Discipline), Teaching(Instructor, Discipline)}\}$;
- (d) Conjunto $H = \{\text{kind_of(Person, Thing), kind_of(Institution, Thing), kind_of(Instructor, Person), kind_of(Student, Person), kind_of(Education, Institution), kind_of(Discipline, Education), kind_of(Discipline, Course)}\}$.

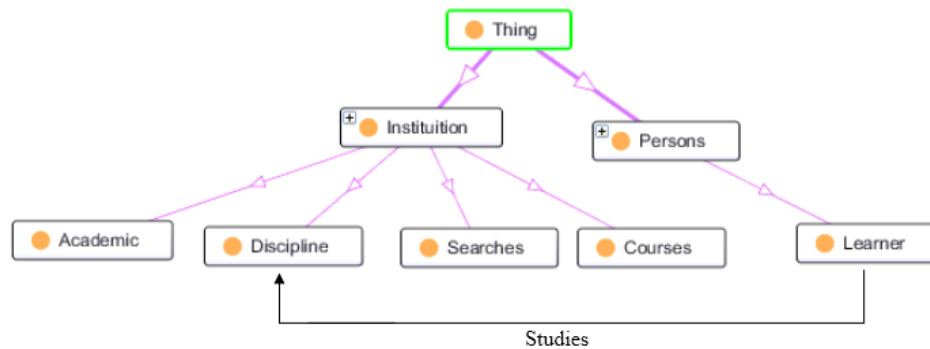


Figura 3.4: Demonstração gráfica de parte da ontologia O_2

2. Ítems extraídos a partir da O_2 :

- (a) Conjunto $C_C = \{\text{Thing, Person, Institution, Academic, Discipline, Seachers, Courses, Learner}\}$;
- (b) Conjunto $P = \{\text{Name(Person, String), Date_of_birth(Person, Date), Bout(Courses,String)}\}$;
- (c) Conjunto $R = \{\text{Studies(Learner, Discipline)}\}$;
- (d) Conjunto $H = \{\text{kind_of(Institution,Thing), kind_of(Person,Thing), kind_of(Academic, Institution), kind_of(Discipline, Institution), kind_of(Seachers, Institution), kind_of(Courses, Institution), kind_of(Learner, Person)}\}$.

Após a identificação e extração de todos os elementos presentes nas ontologias dadas como entrada, os termos que representam esses elementos são agrupados em duas listas, de acordo com a ontologia a que pertenciam. Estas listas servirão como entrada para a atividade mapeamento descrita na seção 3.2.

3.2 Mapeamento

A fase “Mapeamento” tem o objetivo de identificar as correspondências entre os elementos das ontologias dadas como entrada. Utiliza a análise de similaridade entre os termos que correspondem a cada elemento presente nas listas obtidas a partir da atividade “Identificação e Extração dos Elementos das Ontologias”. A fase é composta por quatro atividades: “Comparação Lexical”, “Comparação Estrutural”, “Comparação Relacional” e “Indexação de Termos Órfãos”.

Como entrada recebe duas listas de termos: “lista de termos de O_1 ” e “lista de termos de O_2 ” conforme os itens 1 e 2 da seção 3.1. Como resultado são geradas três listas: “lista de termos órfãos”, “lista de termos com similaridade aceitável” e “lista de termos indexados”, a primeira destaca uma relação de termos que não obtiveram correspondentes, enquanto a segunda destaca uma relação de termos que obtiveram similaridade, já a última traz os termos do conjunto C_C que tiveram sua estrutura hierárquica modificada. A seguir, a Figura 3.5 ilustra uma visão geral da atividade Mapeamento.

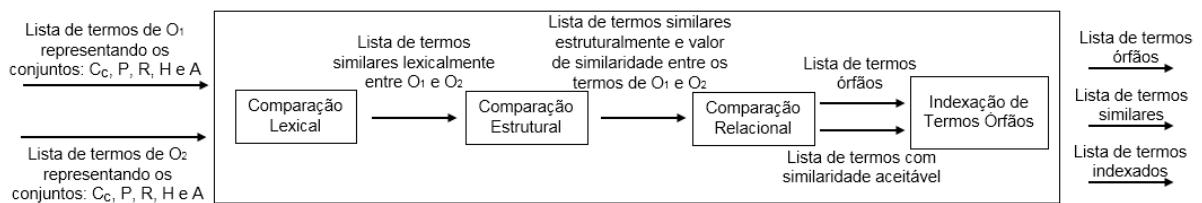


Figura 3.5: Visão geral da fase de Mapeamento

A seção 3.2.1 descreve a atividade “Comparação Lexical”, onde duas listas de termos “lista de termos de O_1 ” e “lista de termos de O_2 ” são dadas como entrada, cada termo é enriquecido através da sinonímia, intermediado por um dicionário lexical. Posteriormente, cada termo e seus sinônimos são comparados com os termos da lista oposta, respeitando os conjuntos dos componentes da ontologia ao qual pertencem. A atividade tem como resultado uma “lista de termos similares lexicalmente”. A Tabela 3.1 apresenta um exemplo dessa lista.

A seção 3.2.2 descreve a atividade “Comparação Estrutural”, onde duas listas de termos “lista de termos de O_1 ” e “lista de termos de O_2 ” são dadas como entrada. A atividade visa identificar uma medida de similaridade capaz de expressar o quão semelhantes são os termos de O_1 e O_2 , levando em consideração as estruturas hierárquicas em que cada termo do conjunto C_C está inserido. A atividade tem como resultado uma “lista de termos similares estruturalmente”, nessa lista contém o par de termos correspondentes e seu respectivo valor de similaridade. A Tabela 3.2 apresenta um exemplo dessa lista.

A seção 3.2.3 descreve a atividade “Comparação Relacional”, onde duas listas de termos “lista de termos de O_1 ” e “lista de termos de O_2 ” são dadas como entrada. A atividade tem como objetivo realizar uma análise de similaridade, levando em consideração os conjuntos de propriedades e de relações não taxonômicas

pertencentes a cada termo do conjunto C_C , a fim de atribuir pesos à medida de similaridade obtida na atividade “Comparação Estrutural”. A atividade tem como resultado duas listas: “lista de termos similares” e “lista de termos órfãos”.

A seção 3.2.4 descreve a atividade “Indexação de Termos”, onde é realizada uma varredura top_down das estruturas hierárquicas dos termos representativos dos elementos do conjunto C_C , a atividade busca encontrar correspondências implícitas nas hierarquias desses termos, além de realizar alterações nos termos representativos dos elementos do conjunto H, preservando a semântica dos elementos do conjunto C_C .

A seguir, são apresentados detalhes das atividades que compõem a fase “Mapeamento”. Assim como, suas entradas, procedimentos e o produto resultante de cada uma delas.

3.2.1 Comparação Lexical

A atividade “Comparação Lexical” tem o objetivo de realizar uma comparação lexical entre os termos representativos dos elementos obtidos a partir da ontologia O_1 e os termos representativos dos elementos obtidos a partir da ontologia O_2 . A Figura 3.6 ilustra a atividade “Comparação Lexical”.

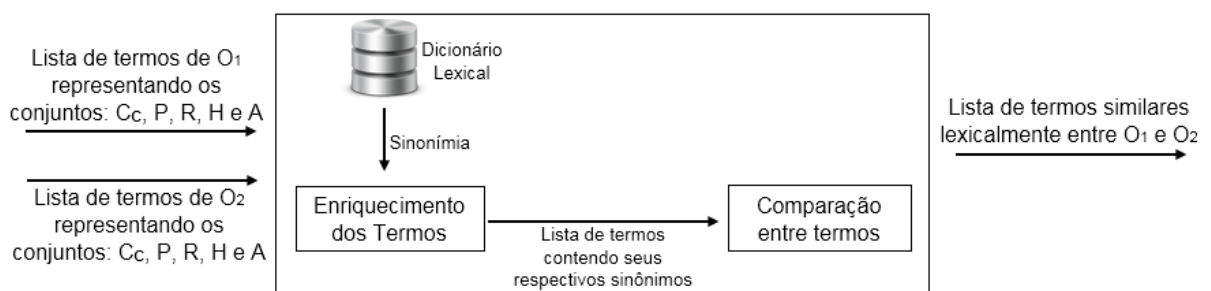


Figura 3.6: Visão geral da atividade Comparação Lexical

A atividade recebe como entrada duas listas de termos, “lista de termos de O_1 ” e “lista de termos de O_2 ” compostas pelos elementos do conjunto C_C , P, R, H e A, estas listas são os produtos resultantes da atividade descrita na seção 3.1. Em seguida, cada termo é enriquecido através da sinonímia, por intermédio de um dicionário lexical, por exemplo, o termo “professor” é enriquecido pela sinonímia adquirindo assim seus respectivos sinônimos como, por exemplo, “Instructor” e “Docente”. Para este trabalho, foi utilizado como dicionário lexical o *Wordnet* [27].

Posteriormente, cada termo e seus sinônimos são comparados com os termos da lista oposta, respeitando o conjunto do elemento da ontologia ao qual pertencem. Desta forma, os termos do conjunto C_C somente são comparados com termos da lista oposta, que pertençam a este mesmo conjunto, pois não se pode comparar uma classe com uma propriedade.

A seguir são definidos os procedimentos a serem realizados para identificar as correspondências de cada conjunto de termos. A Tabela 3.1 apresenta um exemplo de lista de termos resultante desta atividade

- Classes: a comparação utiliza somente o rótulo (*label*) da classe;
- Propriedades: a comparação utiliza o rótulo da propriedade e seu tipo de dado, i.e., String, int, date, etc;
- Relacionamentos não taxonômicos: a comparação utiliza o rótulo da relação, o rótulo da classe que representa o domínio (*domain*) e o rótulo da classe que representa o valor (*range*);
- Hierarquias: a comparação utiliza o rótulo da classe e o rótulo de sua superclasse;
- Axiomas: a comparação utiliza o rótulo da conclusão do axioma (axiomas são compostos por duas partes: condições e conclusão).

Tabela 3.1: Exemplo de lista de termos similares lexicalmente

Termos de O_1	Termos de O_2
Persons	Persons
Student	Learner
Discipline	Discipline
Courses	Courses

3.2.2 Comparação Estrutural

A atividade “Comparação Estrutural” tem o objetivo de realizar uma comparação entre os termos dos conjuntos C_C e H das duas ontologias dadas como entrada. Esta atividade recebe como entrada duas listas de termos, “lista de termos O_1 ”

e “*lista de termos O_2* ” obtidas a partir da fase “Identificação e Extração dos Elementos das Ontologias” descrita na seção 3.1. Além destas listas, a atividade recebe uma outra, “*lista de termos similares lexicalmente*”, obtida a partir da atividade “Comparação Lexical” descrita na seção 3.2.1, na qual contém a similaridade lexical entre os termos de O_1 e O_2 .

Essa atividade visa identificar um valor de similaridade capaz de expressar o quão semelhante são os termos pertencentes ao conjunto C_C presentes nas ontologias O_1 e O_2 . Esse valor de similaridade é calculado através da Fórmula 3.1, que foi definida a partir do trabalho de Mendes [26] e adaptada ² de acordo com as necessidades deste processo.

$$\text{Sim}_1(c_i, c_j) = \frac{2 * |C_{iH} \cap C_{jH}|}{|C_{iH}| + |C_{jH}|} \quad (3.1)$$

onde,

c_i : é uma classe da Ontologia O_1 ;

c_j : é uma classe da Ontologia O_2 ;

C_{iH} : é uma lista de superclasses da classe c_i na hierarquia H da ontologia O_1 ;

C_{jH} : é uma lista de superclasses da classe c_j na hierarquia H da ontologia O_2 .

O cálculo do valor de similaridade é realizado utilizando os elementos da matriz $m \times n$, onde m representa a “*lista de termos O_1* ” e n representa a “*lista de termos O_2* ”. Sendo assim, os termos constantes na “*lista de termos O_1* ” são comparados com os termos presentes na “*lista de termos O_2* ”. Além dessas listas, o cálculo do valor de similaridade utiliza uma lista contendo as similaridades léxicas dos termos, a fim de identificar as interseções entre os elementos contidos na “*lista de termos O_1* ” e na “*lista de termos O_2* ”. Essa comparação resulta em uma nova lista contendo os termos comparados e seus respectivos valores de similaridade, que podem variar de 0 a 1 de acordo com seu grau de similaridade.

²Foram alteradas as variáveis da fórmula a fim de utilizar os elementos da definição formal de ontologia, apresentada na seção 2.1

A seguir, a fim de exemplificar a Fórmula 3.1 descrita nesta seção, são demonstrados os procedimentos a serem realizados para calcular o valor de similaridade entre termos pertencentes à ontologia O_1 e à O_2 . A Figura 3.3 e 3.4 ilustram respectivamente as ontologias O_1 e O_2 utilizadas como exemplo. A Fórmula 3.2 representa o cálculo do valor de similaridade entre o termo “Student” pertencente à O_1 e o termo “Seachers” pertencente à O_2 .

$$\text{Sim}_1(\text{Student}, \text{Seachers}) = \frac{2 * |C_{iH} \cap C_{jH}|}{|C_{iH}| + |C_{jH}|} = \frac{2 * |1|}{6} = 0,333 \quad (3.2)$$

Em um primeiro momento, são analisadas as hierarquias dos termos a partir de “Student” e “Seachers”, sendo identificados respectivamente os seguintes termos: [Student | Persons | Thing] e [Seachers | Institution | Thing], destes termos, somente o termo “Thing” está presente em ambas às listas e por sua vez representa a interseção entre elas. Esta interseção é identificada a partir do resultado obtido na atividade “Comparação Lexical” que consistiu em encontrar uma correspondência lexical entre o termo “Thing” de O_1 e o termo “Thing” de O_2 .

Outro exemplo do cálculo do valor de similaridade é demonstrado a seguir, utilizando os termos “Student” e “Learner” das respectivas listas de termos de O_1 e O_2 . Vale ressaltar, que embora o termo “Student” seja encontrado somente em uma das listas, o seu conjunto de sinônimos contém o termo “Learner”, desta forma, também é contado para o número de interseções entre as listas, pois ocorreu similaridade lexical entre os termos. A seguir, a Fórmula 3.3 demonstra o cálculo do valor de similaridade deste exemplo.

$$\text{Sim}_1(\text{Student}, \text{Learner}) = \frac{2 * |C_{iH} \cap C_{jH}|}{|C_{iH}| + |C_{jH}|} = \frac{2 * |3|}{6} = 1 \quad (3.3)$$

A atividade “Comparação Estrutural” torna possível que a comparação analise não apenas a nomenclatura dos termos, mas também a hierarquia em que esses termos estão inseridos, levando em consideração sua semântica. A Tabela 3.2 apresenta um exemplo da lista de termos resultante desta atividade.

Tabela 3.2: Exemplo de lista de termos similares estruturalmente

Termos de O_1	Termos de O_2	Valor de Similaridade
Student	Searchers	0.333
Student	Leaner	1

3.2.3 Comparação Relacional

A atividade “Comparação Relacional” tem o objetivo de realizar uma comparação entre os termos dos conjuntos P e R que pertençam a algum elemento do conjunto C_C . Essa comparação resulta em outros dois valores de similaridade que serão agregados ao valor de similaridade obtido na fase anterior, a fim de obter um valor de similaridade que englobe uma maior quantidade de elementos.

Esta atividade recebe como entrada duas listas de termos, “*lista de termos O_1* ” e “*lista de termos O_2* ” obtidas a partir da fase “Identificação e Extração dos Elementos das Ontologias” descrita na seção 3.1. Além destas listas, a atividade recebe uma outra, “*lista de termos similares lexicalmente*”, obtida a partir da atividade “Comparação Lexical” descrita na seção 3.2.1. A atividade está dividida em três subatividades, “Comparação de Propriedades”, “Comparação de Relacionamentos não Taxonômicos” e “Agregação dos Valores de Similaridade”. A seguir são apresentadas cada uma destas subatividades.

Comparação de Propriedades

Essa subatividade visa identificar um valor de similaridade capaz de expressar o quão semelhante são os termos pertencentes ao conjunto C_C presentes nas ontologias O_1 e O_2 , analisando o conjunto de propriedades relacionados a cada elemento do conjunto C_C . Esse valor de similaridade é calculado através da Fórmula 3.6, que foi definida a partir do trabalho de Mendes [26] e adaptada de acordo com as necessidades deste processo.

$$\text{Sim}_2(P_{C_i}, P_{C_j}) = \frac{2 * |P_{C_i} \cap P_{C_j}|}{|P_{C_i}| + |P_{C_j}|} \quad (3.4)$$

onde,

P_{c_i} : são todos os termos que compõem o conjunto P da classe c_i , pertencente a uma ontologia O_1 ;

P_{c_j} : são todos os termos que compõem o conjunto P da classe c_j , pertencente a uma ontologia O_2 .

O cálculo do valor de similaridade é novamente realizado entre todas as classes da ontologia O_1 em relação às classes da ontologia O_2 . Desta vez, a fórmula busca identificar correspondências entre o conjunto de propriedades de cada classe analisada.

O que se propõe é que classes que possuem um conjunto de propriedades semelhantes tendem a ter um maior valor de similaridade. A Fórmula 3.7 apresenta um exemplo do valor de similaridade entre a classe "Person" de O_1 e "Person" O_2 .

$$\text{Sim}_2(P_{1_{\text{person}}}, P_{2_{\text{person}}}) = \frac{2 * |P_{1_{\text{person}}} \cap P_{2_{\text{person}}}|}{|P_{1_{\text{person}}}| + |P_{2_{\text{person}}}|} = \frac{2}{3} = 0,666 \quad (3.5)$$

$P_{1_{\text{person}}}$: Representa o conjunto de propriedades da classe "Person" de O_1 ,
i.e. $P_{1_{\text{person}}} = \{(\text{Name}, \text{String})\}$;

$P_{2_{\text{person}}}$: Representa o conjunto de propriedades da classe "Person" de O_2 ,
i.e. $P_{2_{\text{person}}} = \{(\text{Name}, \text{String}), (\text{Date_of_birth}, \text{Date})\}$;

Cada elemento do conjunto P é composto por seu rótulo e por seu tipo de dado. Dessa forma, ao analisar o conjunto de propriedades da classe "Person" de O_1 e "Person" O_2 , obteve-se um elemento correspondente entre eles, a propriedade "Name" está presente em ambas as classes. Contudo, a propriedade "Date_of_birth" não obteve correspondente, fazendo com que o valor de similaridade seja reduzido a 0,666.

Comparação de Relacionamentos não Taxonômicos

Essa subatividade visa identificar um valor de similaridade capaz de expressar o quão semelhante são os termos pertencentes ao conjunto C_C presentes nas ontologias O_1 e O_2 , analisando o conjunto de relações não taxonômicas relacionados a cada elemento do conjunto C_C . Esse valor de similaridade é calculado através da

Fórmula 3.6, que foi definida a partir do trabalho de Mendes [26] e adaptada de acordo com as necessidades deste processo.

$$\text{Sim}_3(R_{c_i}, R_{c_j}) = \frac{2 * |R_{c_i} \cap R_{c_j}|}{|R_{c_i}| + |R_{c_j}|} \quad (3.6)$$

onde,

R_{c_i} : são todos os termos que compõem o conjunto R da classe c_i , pertencente a uma ontologia O_1 ;

R_{c_j} : são todos os termos que compõem o conjunto R da classe c_j , pertencente a uma ontologia O_2 .

O cálculo do valor de similaridade é novamente realizado entre todas as classes da ontologia O_1 em relação às classes da ontologia O_2 . Desta vez, a fórmula busca identificar correspondências entre o conjunto de relações não taxonômicas de cada classe analisada. O que se propõe é que classes que possuem um conjunto de relações não taxonômicas semelhantes tendem a ter um maior valor de similaridade. A Fórmula 3.7 apresenta um exemplo do valor de similaridade entre a classe “Student” de O_1 e “Learner” O_2 .

$$\text{Sim}_3(R_{1\text{student}}, R_{2\text{learner}}) = \frac{2 * |R_{1\text{student}} \cap R_{2\text{learner}}|}{|R_{1\text{learner}}| + |R_{2\text{student}}|} = \frac{2}{2} = 1 \quad (3.7)$$

$R_{1\text{student}}$: Representa o conjunto de relações não taxonômicas da classe “Student” de O_2 , i.e. $R_{2\text{student}} = \{(\text{Studies}, \text{Discipline})\}$;

$R_{2\text{learner}}$: Representa o conjunto de relações não taxonômicas da classe “Learner” de O_1 , $R_{2\text{learner}} = \{(\text{Studies}, \text{Discipline})\}$.

Cada elemento do conjunto R é composto por seu rótulo e por seu valor. Dessa forma, ao analisar o conjunto de relações da classe “Student” de O_1 e Learner O_2 , obteve-se um elemento correspondente entre eles, a relação não taxonômica “Studies” está presente em ambas as classes, fazendo com que o valor de similaridade obtido seja 1.

Agregação dos Valores de Similaridade

Esta subatividade tem como objetivo agregar os valores de similaridades obtidos nas atividades de “Comparação Estrutural” e “Comparação Relacional”, esta última por sua vez possui dois tipos de valores (obtidos a partir Sim_2 e Sim_3). Nesta subatividade são ponderados os valores obtidos a fim de resultar em um único valor que melhor represente a similaridade entre os elementos do conjunto C_C . A Fórmula 3.8 define como o cálculo do valor de similaridade final é obtido.

$$Sim_{final} = Sim_1 * p_1 + Sim_2 * p_2 + Sim_3 * p_3 \quad (3.8)$$

onde,

Sim_1 : é o valor de similaridade entre as classes c_i e c_j obtidos na atividade “Comparação Estrutural” descrita na seção 3.2.2.

Sim_2 : é o valor de similaridade entre as classes c_i e c_j obtidos na subatividade “Comparação de Propriedades” da atividade “Comparação Relacional” descrita na seção 3.2.3.

Sim_3 : é o valor de similaridade entre as classes c_i e c_j obtidos na subatividade “Comparação de Relacionamento não Taxonômicos” da atividade “Comparação Relacional” descrita na seção 3.2.3.

p_1 , p_2 e p_3 : são pesos predefinidos pelo engenheiro do conhecimento em conjunto com o especialista do domínio de acordo com a prioridade da medida de similaridade. Desta forma, foi definido que p_1 deve ser maior que os demais pesos (p_2 e p_3), pois está relacionado ao valor de similaridade obtido pela fórmula Sim_1 , esta fórmula utiliza classes e hierarquias como entrada, elementos prioritários para a criação de uma ontologia. Os demais pesos apresentam prioridades iguais, ficando a critério do desenvolvedor a escolha de seus valores. É importante ressaltar que os valores dos pesos (p_1 , p_2 e p_3) ficam a critério do engenheiro do conhecimento em conjunto com o especialista, podendo estes escolherem quaisquer valores entre 0 e 1.

Ao fim desta subatividade são geradas duas listas: “*lista de termos com similaridade aceitável*” e “*lista de termos órfãos*”. A lista de termos com similaridade

aceitável contém todos os termos que atingiram um valor similaridade preestabelecido pelo o engenheiro do conhecimento em conjunto com o especialista do domínio. Para este trabalho foi definido que o valor de similaridade aceitável (que define se um termo é similar ao outro) é 0.6. Sendo assim, caso exista um termo da “*lista de termos O_1* ” e este possua um valor de similaridade igual ou superior a 0.6 a um termo “*lista de termos O_2* ”, ambos serão adicionados na “*lista de termos com similaridade aceitável*”. Por outro lado, a “*lista de termos órfãos*” contém todos os termos que não atingiram o valor de similaridade aceitável. As Tabelas 3.3 e 3.4 apresentam respectivamente cada uma delas.

Tabela 3.3: Exemplo de lista de termos com similaridade aceitável

Termos de O_1	Termos de O_2	Valor de similaridade
Persons	Persons	0,875
Student	Learner	0,805
Discipline	Discipline	0,605
Courses	Courses	0,750

Tabela 3.4: Exemplo de lista de termos órfãos

Termos de O_1	Termos de O_2
Education	
Instructor	
	Academic
	Searches

3.2.4 Indexação de Termos

Após a atividade “*Comparação Relacional*” descrita na seção anterior, foram geradas duas listas de termos: “*lista de termos com similaridade aceitável*” e “*lista de termos órfãos*”. A primeira contém todos os termos que atingiram um valor similaridade preestabelecido pelo o engenheiro do conhecimento em conjunto com o especialista do domínio, conforme estabelecido na seção 3.2.3. A segunda contém todos os termos que não obtiveram correspondentes.

A fim de identificar elementos correspondentes para termos que compõem a “lista de termos órfãos”, além de alterar a estrutura hierárquica dos conceitos a fim de encontrar uma harmonia entre as estruturas de ambas as ontologias dadas como entrada é utilizado mais um mecanismo de mapeamento, esse por sua vez leva em consideração os hiperônimos dos termos. Esse critério é utilizado, pois todo termo que representa um elemento do conjunto C_C possui um termo que represente seu hiperônimo, termo este adquirido através de um dicionário lexical, conforme descrito na seção 3.1.

A atividade “Indexação dos Termos” tem o objetivo de realizar uma varredura top_down das estruturas hierárquicas dos termos representativos dos elementos do conjunto C_C . Essa varredura através do uso da hiperonímia busca encontrar correspondências implícitas nas hierarquias desses termos. Tal varredura se torna importante para encontrar a melhor estrutura hierárquica dos termos, preservando da melhor maneira a sua semântica.

Em um primeiro momento é realizada uma varredura de cima para baixo, partindo do pressuposto que todos os termos descendem do termo “Thing”. A fim de prover essa varredura, alguns critérios de indexação foram elaborados. A seguir são descritos cada um deles:

Subclasses diretas da classe “Thing”

1. Quando x_1 for subclasse de “Thing” em O_1 e similar a x_2 subclasse de “Thing” em O_2 , então seus hiperônimos serão descartados, pois não há necessidade de indexá-la como subclasse de outra classe que não seja a “Thing”.

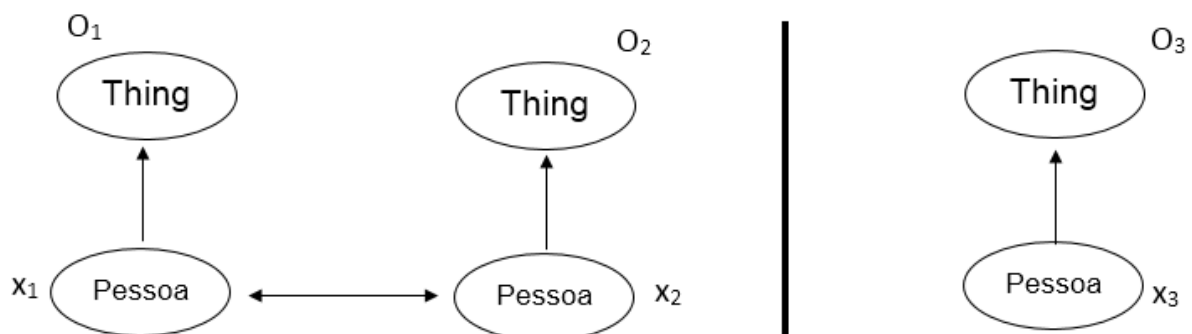


Figura 3.7: Regra 1 para indexação de termos

A Figura 3.7 ilustra um exemplo desta regra, tendo com entrada duas ontologias O_1 e O_2 . O_3 representa a ontologia combinada seguindo as definições desta regra. A classe x_3 adquire agora todos os elementos dos conjuntos P, R, H e A que pertenciam a x_1 e x_2 . A classe x_3 recebe uma nomenclatura na fase “Análise de Integridade dos Termos” descrita na seção 3.3, para então os elementos serem combinados na etapa “Junção” descrita na seção 3.4.

- Quando x_1 for subclasse de “Thing” em O_1 e similar a y_2 que é subclasse de x_2 e este por sua vez é subclasse de “Thing” em O_2 , então x_1 será indexada a y_2 , deixando de ser subclasse de “Thing” em O_1 .

A Figura 3.8 ilustra um exemplo desta regra, tendo com entrada duas ontologias O_1 e O_2 . A ontologia O_3 representa a combinação destas, seguindo as definições desta regra. A classe y_3 adquire agora todos os elementos dos conjuntos P, R, H e A que pertenciam a x_1 e y_2 . A classe y_3 recebe uma nomenclatura na fase “Análise de Integridade dos Termos” descrita na seção 3.3, para então os elementos serem combinados na etapa “Junção” descrita na seção 3.4.

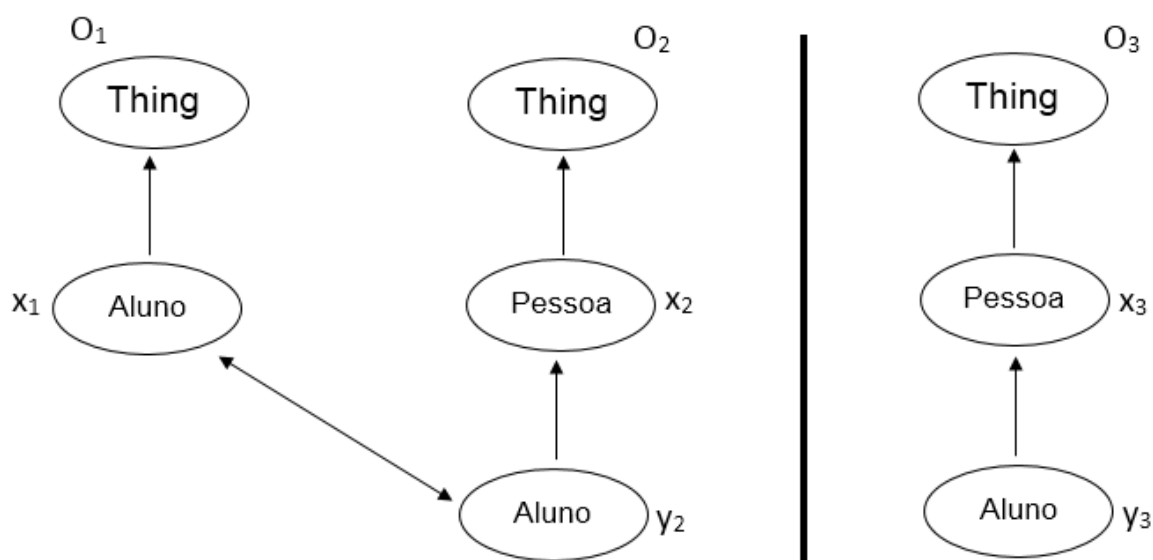


Figura 3.8: Regra 2 para indexação de termos

- Quando x_1 for subclasse de “Thing” em O_1 e não obtiver similaridade a nenhuma classe pertencente a O_2 , então será analisado o conjunto de termos que represente a sua hiperonímia, comparando cada elemento deste conjunto com as classes de O_2 , essa comparação resultará em três possibilidades:

- (a) Um termo do conjunto de hiperônimos de x_1 é equivalente a uma classe z_2 de O_2 , então x_1 será indexada como subclasse de x_2 , deixando de ser subclasse de “Thing” em O_1 ;

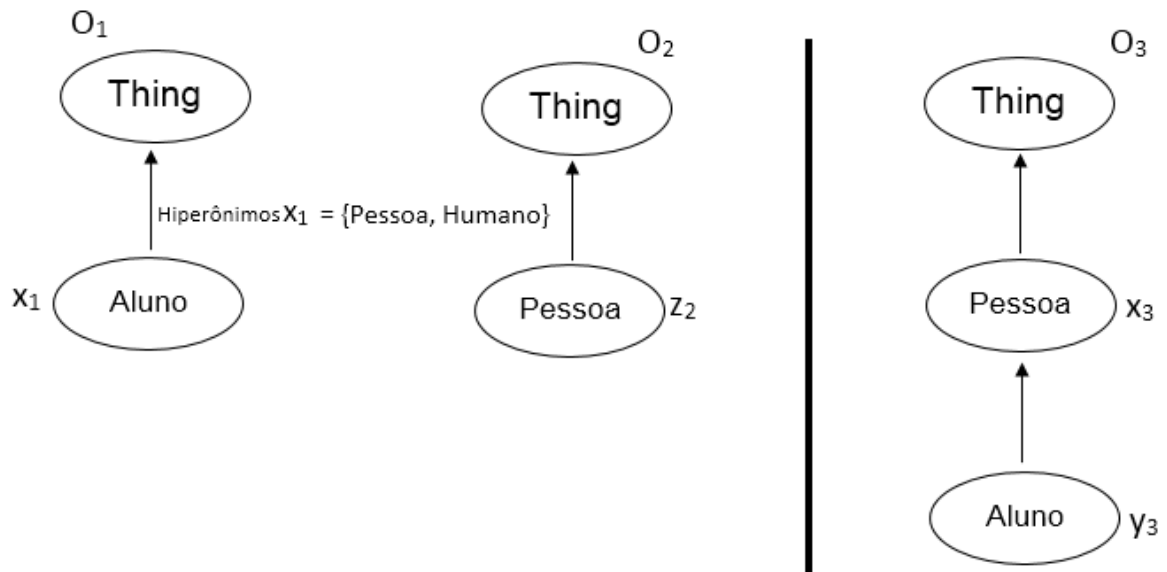


Figura 3.9: Regra 3 (a) para indexação de termos

- (b) O hiperônimo de x_1 é equivalente a mais de uma classe (z_1, \dots, z_n), então x_1 será indexada como subclasse de z que estiver mais distante hierarquicamente da classe “thing” de O_2 ;
- (c) O hiperônimo de x_1 não possui equivalências, então será indexado diretamente como subclasse de “thing” em O_1 .

Classes descendentes de subclasses de “Thing”

1. Quando y_1 for subclasse de x_1 e este por sua vez for subclasse de “Thing” em O_1 e y_1 for similar a z_2 que é subclasse y_2 que é subclasse de x_2 e este por sua vez é subclasse de “Thing” em O_2 , então y_1 será indexada a z_2 , deixando de ser subclasse de x_2 e passando à fase de “Análise de Integridade dos Termos” juntamente com seu termo similar z_2 ;
2. Quando y_1 for subclasse de x_1 e este por sua vez for subclasse de “Thing” em O_1 e y_1 não obtiver similaridade a nenhuma classe pertencente a O_2 e x_1 for similar a x_2 em O_2 , então y_1 será indexada como subclasse de x_2 .

3. Quando y_1 for subclasse de x_1 e este por sua vez for subclasse de “Thing” em O_1 e y_1 não obtiver similaridade a nenhuma classe pertencente a O_2 e x_1 for similar a x_2 em O_2 , mas x_1 possuir uma subclasse y_3 diferente de y_1 que tenha similaridade a alguma classe z_2 descendente direta ou indireta de x_2 em O_2 , então y_1 será indexada como subclasse da super classe de z_2 .

3.2.5 Considerações sobre a fase do Mapeamento

A fase “Mapeamento” estabelece formalismos capazes de identificar a correspondência entre os termos das ontologias dadas como entrada. Estes formalismos podem realizar uma análise de similaridade entre os elementos através de uma simples comparação lexical, assim como uma análise mais detalhada, examinando sua estrutura hierárquica e demais elementos como propriedades e relações não taxonômicas. Quando esta correspondência não existe explicitamente é utilizado o recurso da hiperonímia, a fim de que sejam identificadas correspondências aproximadas.

A fase de mapeamento resulta em três lista de termos: “lista de termos órfãos”, “lista de termos similares” e “lista de termos indexados”, a primeira destaca uma relação de termos que não obtiveram correspondentes, conforme descrito na seção 3.2.3, enquanto a segunda destaca uma relação de termos que obtiveram similaridade, conforme descrito na seção 3.2.3, já a última traz os termos do conjunto C_C que tiveram sua estrutura hierárquica modificada, conforme descrito na seção 3.2.4.

3.3 Análise de Integridade dos Termos

A finalidade desta atividade é permitir ao engenheiro do conhecimento em conjunto com o especialista do domínio a escolha e/ou alteração das denominações dos termos que tiveram um nível de similaridade aceitável, conforme definido na seção 3.2.3. Ou seja, quando dois termos possuem correspondências, mas com nomenclaturas diferentes, o engenheiro do conhecimento com auxílio do especialista do domínio tem o papel de escolher e/ou alterar uma nomenclatura que melhor expresse o significado de ambos os termos, podendo ser escolhida a nomenclatura

de um dos termos existentes ou ser inserida uma nova que nomenclatura que melhor represente ambos.

Um exemplo simples da necessidade desta fase se dá quando dois termos como é o caso de “Student” e “Learner” são definidos como similares, cabendo ao engenheiro do conhecimento em conjunto com o especialista do domínio a escolha e/ou alteração da nomenclatura dos termos que melhor expresse o significado destes como por exemplo: o termo “schoolboy” pode representar os termos “Student” e “Learner”;

Contudo, essa liberdade dada ao engenheiro do conhecimento torna o processo semiautomático, pois nesta atividade há a necessidade de intervenção do usuário. Esta intervenção deve ser resguardada de uma verificação de integridade dos termos, haja visto que o usuário não poderá utilizar nomenclaturas iguais para representar os termos dos conjuntos C_C , P , R e A .

Um outro exemplo, diz respeito aos termos do conjunto A , pois ao serem analisadas as conclusões dos axiomas, podem ser identificadas condições diferentes para a mesma conclusão; Caso haja duplicidade, deve analisar se as duas regras se equivalem. Caso a regra seja a mesma, uma delas é utilizada. Caso sejam regras diferentes, ambas são indexadas e utilizadas posteriormente na fase “Junção” descrita na seção 3.4.

A Tabela 3.5 demonstra parte dos termos similares entre as ontologias O_1 e O_2 , assim como os termos utilizados pelo engenheiro do conhecimento em conjunto com o especialista do domínio para redefini-los.

Tabela 3.5: Lista de termos Redefinidos

Termos de O_1	Termos de O_2	Redefinição
Person	Person	Person
Lecturer	Instructor	Lecturer
Student	Learner	Student
Course	Course	Course
Name	Name	Name
Registration	Registration	Registration
Studies	Studies	Studies

3.4 Junção

A fase “Junção” tem o objetivo de realizar a combinação (propriamente dita) entre os termos que compõem as ontologias O_1 e O_2 . Essa fase recebe como entrada três listas de termos: “*lista de termos com similaridade aceitável*”, “*lista de termos indexados*” e “*lista de termos órfãos*”. Como resultado é gerado uma única lista contendo todos os elementos das ontologias dadas como entrada conforme apresentada na Tabela 3.5. A fase possui duas atividades: “Combinação de Elementos do Conjunto C_C ” e “Combinação de Elementos dos Conjunto P, R, H e A”. A Figura 3.10 apresenta uma visão geral da fase Junção.

A seção 3.4.1 descreve a atividade “Combinação de Elementos do Conjunto C_C ”, onde os elementos do conjunto C_C contidos nas “*lista de termos com similaridade aceitável*”, “*lista de termos indexados*” e “*lista de termos órfãos*” são combinados em uma única lista de termos.

A seção 3.4.2 descreve a atividade “Combinação de Elementos dos Conjuntos P, R, H e A”, onde os elementos dos conjuntos P, R, H e A contidos nas “*lista de termos com similaridade aceitável*”, “*lista de termos indexados*” e “*lista de termos órfãos*” são relacionados aos termos do conjunto C_C combinados pela seção 3.4.1.

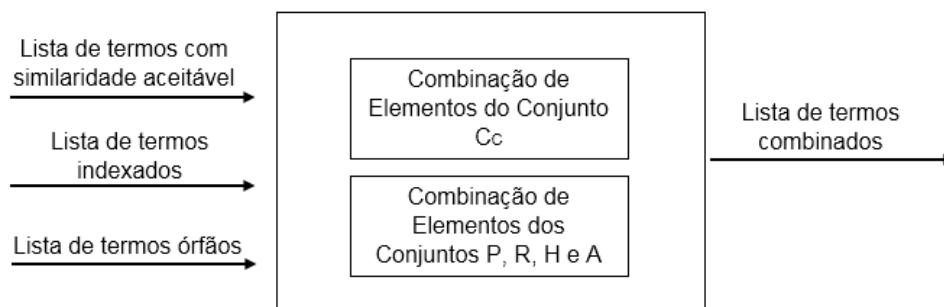


Figura 3.10: Visão geral da fase Junção

3.4.1 Combinação de Elementos do Conjunto C_C

A atividade “Combinação de Elementos do Conjunto C_C ” tem o objetivo de realizar uma combinação entre os elementos do conjunto C_C contidos nas “*lista de termos com similaridade aceitável*”, “*lista de termos indexados*” e “*lista de termos órfãos*”, adicionando-os em uma única listagem, definida como “*lista de termos combinados*”.

É importante ressaltar que os termos deste conjunto ao obterem termos correspondentes na fase de mapeamento descrita na seção 3.2 são redefinidos conforme descreve a fase de Análise de Integridade dos Termos descrita na seção 3.3. Sendo assim, não há duplicidade de termos do conjunto C_C , mesmo que exista termos similares, pois nesta fase quando houver termos com similaridade aceitável são adicionados somente os termos que os represente, redefinidos pelo próprio engenheiro do conhecimento em conjunto com o especialista do domínio.

Um exemplo é dado quando dois termos “Student” e “Learner” respectivamente pertencentes ao conjunto C_C das ontologias O_1 e O_2 são definidos como similares, estes por sua vez são redefinidos pelo engenheiro do conhecimento em conjunto com o especialista do domínio, onde o termo “Student” é atribuído para melhor representar os termos anteriores. Sendo assim, o termo “Student” passa a representar o termo “Student” de O_1 e “Learner” de O_2 e conseqüentemente é utilizado nesta atividade, sendo adicionado à “*lista de termos combinados*”.

Além dos termos redefinidos, outros termos são adicionados à lista de termos combinados, como é o caso dos termos do conjunto C_C contidos na “*lista de termos indexados*” e “*lista de termos órfãos*”, estes por não possuírem correspondentes são simplesmente adicionados à “*lista de termos combinados*”.

Ao fim desta atividade são reunidos em uma única listagem todos os elementos do conjunto C_C obtidos a partir das “*lista de termos com similaridade aceitável*”, “*lista de termos indexados*” e “*lista de termos órfãos*”.

A atividade “Combinação de Elementos do Conjunto C_C ” é fundamental para a fase da Junção, pois é a partir desta fase que os demais elementos dos conjuntos P, R, H e A são indexados à “*lista de termos combinados*”.

3.4.2 Combinação de Elementos dos Conjuntos P, R, H e A

A atividade “Combinação de Elementos dos Conjuntos P, R, H e A” tem o objetivo de realizar uma indexação dos elementos dos conjuntos P, R, H e A contidos nas “*lista de termos com similaridade aceitável*”, “*lista de termos indexados*” e “*lista de termos órfãos*” aos termos do conjunto C_C já adicionados na “*lista de termos combinados*”.

Essa indexação utiliza as relações existentes destes elementos com os termos do conjunto C_C , a fim de atribuir aos elementos de C_C já combinados os termos dos conjuntos P , R , H e A das duas ontologias dadas como entrada, pois todo termo pertencente aos conjuntos P , R , H ou A possui alguma relação com um termo do conjunto C_C . Essa relação pode ser observada na definição formal de ontologia descrita na seção 2.1 e nos exemplos definidos na seção 3.1.

Sendo assim, os elementos dos conjuntos P , R , H ou A são adicionados a “lista de termos combinados” a sua relação com o termo do conjunto C_C já existe e este por sua vez já está adicionado na lista conforme descrito na seção anterior. Desta forma, dada uma classe c_i são indexadas a ela todos elementos dos conjuntos P , R , H ou A que possuem relação formal com a classe c_i . Dessa forma, conforme ilustrado na Figura 3.11, dado o termo “Person” pertencente ao conjunto C_C adicionado à “lista de termos combinados”, os termos dos conjuntos P , R , H ou A que faziam relação a este nas ontologias dadas como entrada, agora são indexados diretamente ao termo “Person”. A Tabela 3.6 apresenta a “lista de termos combinados” após a junção dos elementos.

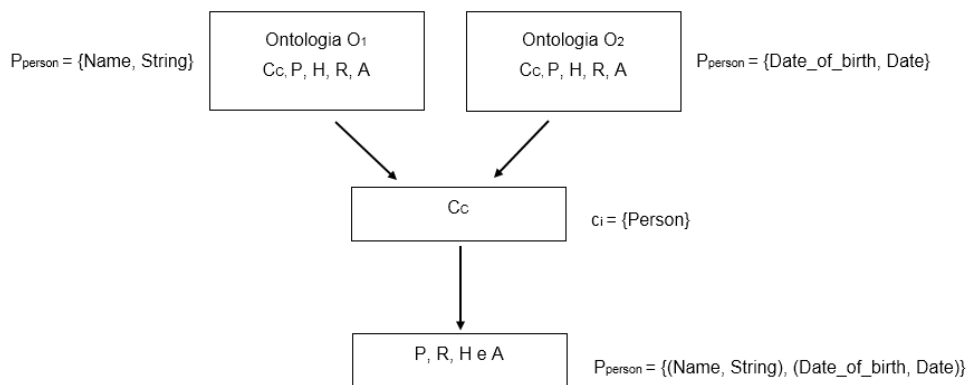


Figura 3.11: Exemplo da combinação de elementos do conjunto P

Tabela 3.6: Lista de termos combinados

Classes	Propriedades	Relações não Taxonômicas	Relações Taxonômicas
Person	Name, Date_of_birth		Thing
Lecturer	Wages, Registration	Teaching {Discipline}	Person
Student	Registration	Studies {Discipline}	Person
Course	Name, Bout, Length		Thing

3.5 Análise e Representação da nova Ontologia

A fase “Análise e Representação da nova Ontologia” tem como objetivo realizar uma última análise de integridade nos termos obtidos a partir da fase “Junção”. Essa análise visa encontrar termos que não deveriam estar presentes nesta lista, como por exemplo: termos com a mesma nomenclatura estarem presentes no mesmo conjunto ou possuir algum termo com a mesma nomenclatura de uma palavra reservada da linguagem na qual se deseja representar formalmente a nova ontologia, isso ocasionaria inconsistências e inviabilizaria seu uso futuro.

Realizada essa verificação, os termos resultantes fazem partes da composição formal da ontologia, pois possuem elementos dos conjuntos C_C , P , R e H . Entretanto, não estão expressos formalmente em uma linguagem de representação de ontologias. Por esse motivo, após a verificação de integridade os termos são representados em uma linguagem. Para este trabalho foi utilizado a linguagem OWL [25], pois esta é uma linguagem padronizada pelo W3C. A Tabela demonstra a “lista de termos combinados” e analisados nesta atividade. A Figura 3.12 demonstra uma ilustração de parte da ontologia resultante.

Tabela 3.7: Lista de termos combinados e analisados

Classes	Propriedades	Relações não Taxonômicas	Relações Taxonômicas
Person	Name, Age		Thing
Lecturer	Wages, Registration	Teaching {Discipline}	Person
Student	Registration	Studies {Discipline}	Person
Course	Name, Bout, Length		Thing

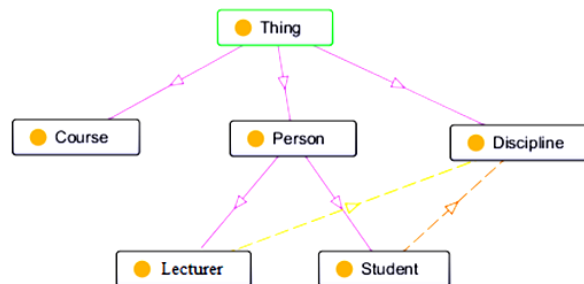


Figura 3.12: Ilustração de parte de uma ontologia reusada

3.6 OntoJoinTool – Uma Ferramenta de Suporte ao processo OntoJoin

Para efeitos de avaliação e auxiliar a junção de ontologias seguindo o escopo definido pelo processo OntoJoin, desenvolveu-se a OntoJoinTool, uma ferramenta de software semi-automatizada, implementada utilizando as linguagens de programação PHP [33] e Java. Para manipulação das ontologias na forma de objetos utilizou-se a API Jena [5] e a API Protege [36]. Além destas linguagens, foram utilizadas XHTML [50], CSS [6] e JavaScript [20] para desenvolvimento e manipulação das interfaces.

A Figura 3.13 apresenta a tela inicial da ferramenta OntoJoinTool contendo as suas seis etapas: “Choosing OWL Files”, “Identification and Extraction of the Elements of Ontologies”, “Mapping”, “Integrity Analysis of Terms”, “Join” e “Control and Generation of the New Ontology”. Cada etapa corresponde a cada uma das fases do processo OntoJoin. Para cada uma delas há uma tela específica na qual serão apresentadas a seguir, de acordo com a ordem em que foram definidas pelo processo.



Figura 3.13: Visão geral da Ferramenta OntoJoinTool

3.6.1 Choosing OWL Files

A Figura 3.14 representa a etapa “Choosing OWL Files” na qual corresponde a interface que permite ao engenheiro do conhecimento informar os dois arquivos OWL correspondentes as duas ontologias a serem utilizadas como entrada. Esta linguagem para representação de ontologias foi escolhida como padrão a ser

adotado pela ferramenta devido ao seu grande potencial de expressividade conforme descreve McGuinness, além da mesma ser padronizada pelo W3C [25].

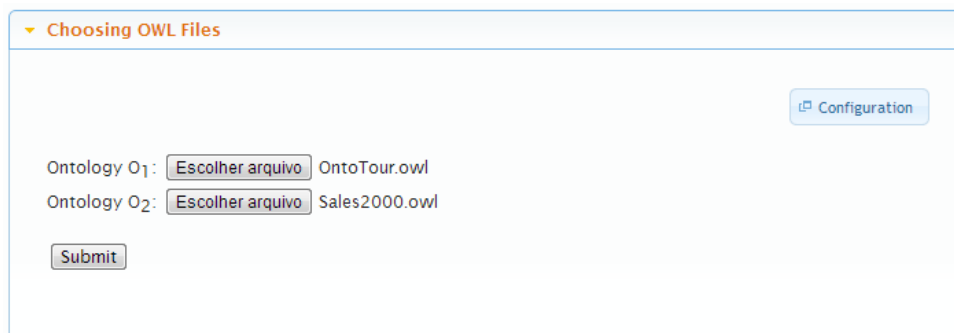


Figura 3.14: Interface para selecionar arquivos OWL e botão para configuração dos pesos das medidas de similaridade

Nesta etapa da ferramenta o engenheiro do conhecimento já dispõe das ontologias que deseja combinar. Fases anteriores como busca e seleção destas ontologias são abstraídas e não fazem parte dos requisitos da ferramenta.

Como exemplo são inseridas como entrada na ferramenta duas ontologias: “OntoTour.owl” e “Sales2000.owl”, estas ontologias utilizadas foram desenvolvidas pelo grupo GESEC, utilizando a ferramenta Protégé. As ontologias foram geradas já no formato OWL, e por estarem relacionadas a trabalhos da área, facilitou-se a sua seleção.

3.6.2 Identification and Extraction of the Elements of Ontologies

As Figuras 3.15 e 3.16 representam a etapa “Identification and Extraction of the Elements of Ontologies” na qual corresponde a interface que permite ao engenheiro do conhecimento visualizar os elementos extraídos a partir das ontologias dadas como entrada. A ferramenta disponibiliza em sua interface os elementos de acordo com os seus respectivos conjuntos, facilitando assim, a sua visualização e sua análise por parte do engenheiro do conhecimento.

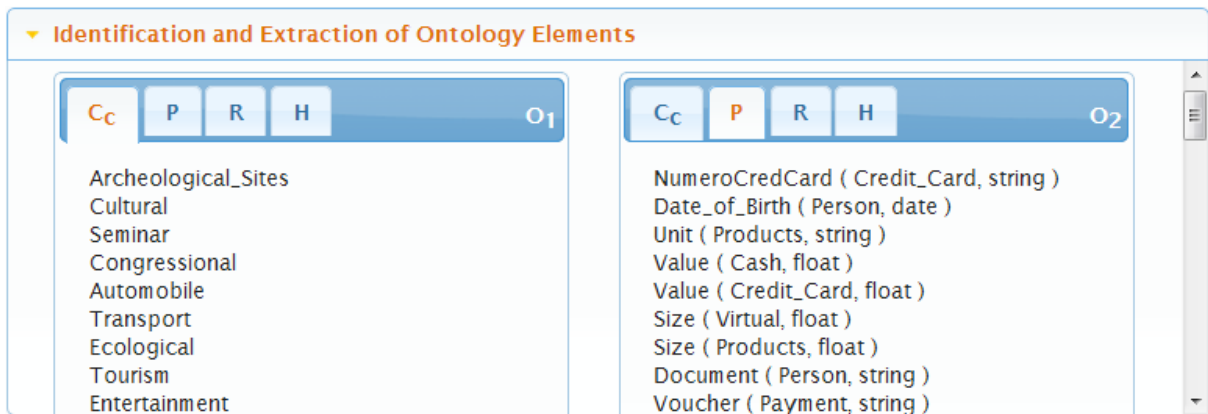


Figura 3.15: Interface que demonstra os termos extraídos e seus respectivos conjuntos (a)

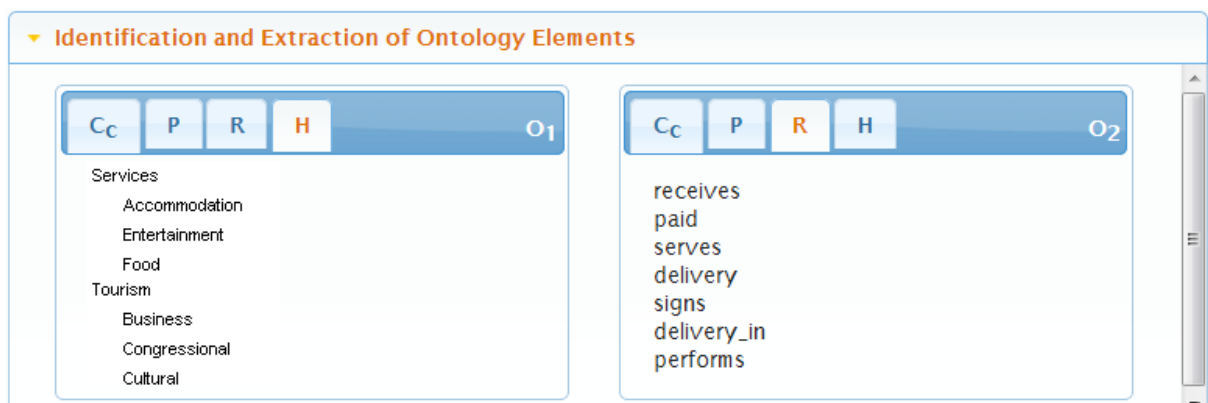


Figura 3.16: Interface que demonstra os termos extraídos e seus respectivos conjuntos (b)

3.6.3 Mapping

A Figura 3.18 representa a etapa “Mapping” na qual corresponde a interface que permite ao engenheiro do conhecimento visualizar o valor de similaridade entre os termos extraídos de acordo com os procedimentos definidos na seção 3.2.

Os termos similares são disponibilizados em ordem decrescente de acordo com o valor de corte inserido pelo engenheiro do conhecimento, este valor de corte representa o “valor de similaridade aceitável” definido na seção 3.2.3. A Figura 3.17 ilustra a interface que disponibiliza a inserção deste valor.

Além da visualização reduzida dos dos termos similares por causa do valor de corte, a ferramenta disponibiliza ao engenheiro do conhecimento a possibilidade de visualizar os valores de similaridade entre todos os elementos extraídos.

Os termos que obtiverem similaridade igual ou superior ao valor de corte (valor de similaridade aceitável) servem como entrada para a próxima etapa da ferramenta: “Integrity Analysis of Terms”.

▼ Mapping

Start similarity analysis

Similarity analysis

Please give a cut off similarity value:

Ok

Figura 3.17: Interface para inserção do valor de corte

▼ Mapping

Classes de O1	Classes de O2	=	Valor de Similaridade
Transport	Transport	=	0.685714285671428571429
Food	Food	=	0.685714285671428571429
Services	Services	=	0.63999999992000000000
Tourism	Tourism	=	0.63333333335333333333

[See All Values Similarity.](#)

Figura 3.18: Valores de similaridade obtidos após o mapeamento

3.6.4 Integrity Analysis of Terms

A Figura 3.19 representa a etapa “Redefinition and Verification” na qual corresponde a interface que permite ao engenheiro do conhecimento informar quais as novas nomenclaturas a serem utilizadas pelos termos selecionados na etapa anterior (seção 3.6.3) para serem combinados.

▼ Integrity Analysis of Terms

Classes de O1	Classes de O2	=	Terms Redefined
Transport	Transport	=	<input type="text" value="Transport"/>
Food	Food	=	<input type="text" value="Food"/>
Services	Services	=	<input type="text" value="Services"/>
Tourism	Tourism	=	<input type="text" value="Tourism"/>

Reset Elements

Figura 3.19: Redefinição dos termos correspondentes

Nesta etapa o engenheiro possui a liberdade de escolher a nomenclatura a ser adotada por dois termos similares. O engenheiro pode escolher como nova nomenclatura qualquer uma das duas nomenclaturas anteriores, ou se preferir pode definir uma nova nomenclatura que melhor atenda as suas necessidades e que represente com maior expressividade os termos anteriores.

Essa liberdade dada ao engenheiro do conhecimento é resguardada por uma verificação de integridade, impedindo que o usuário insira termos não válidos, como por exemplo: termos completamente formados por caracteres especiais, termos idênticos a palavras reservadas da linguagem owl, entre outros casos definidos no escopo do processo OntoJoin na seção 3.3.

3.6.5 Join

Nesta etapa a junção entre os termos das ontologias dadas como entrada é realizada de fato. É a fase crucial para a ferramenta, pois nela são realizadas as manipulações necessárias para que ocorra a junção entre os termos.

A ferramenta executa em segundo plano processos obrigatórios para a realização da junção, como por exemplo: a combinação de dois elementos em um só, a sua associação com as relações anteriores e a realocação desse termos de acordo com a regras de indexação definidas na seção 3.2.4.

Por esta etapa da ferramenta ser dedicada exclusivamente à execução de técnicas definidas pelo processo OntoJoin, é disponibilizado ao usuário somente uma barra de progresso informando o decorrer da sua execução em segundo plano. A Figura 3.20 ilustra a interface desta etapa.

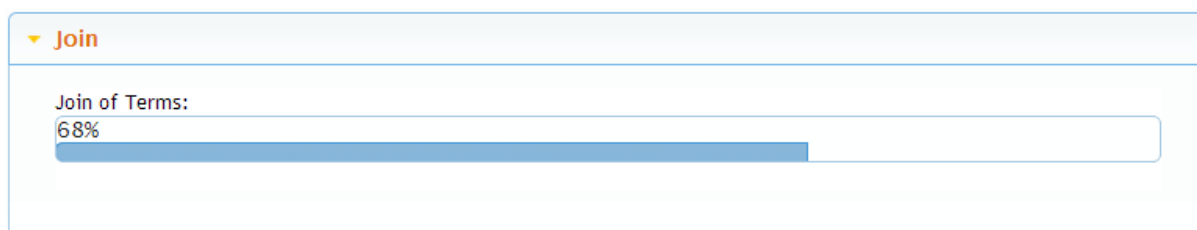


Figura 3.20: Barra de progresso da interface da etapa Join

3.6.6 Control and Generation of the New Ontology

Nesta etapa da ferramenta, o arquivo da ontologia reusada é gerado no formato OWL. A interface permite ao engenheiro do conhecimento visualizar (*download*) o arquivo OWL da ontologia gerada. A Figura 3.21 demonstra a interface da ferramenta responsável por esta etapa.

Contudo, antes de ser gerado este arquivo owl é realizado em segundo plano uma última análise de integridade dos termos, visando garantir que tal ontologia não apresente erros ao fim do processo.



Figura 3.21: Termina da execução da ferramenta

3.7 Considerações Finais

Neste capítulo foi apresentado o OntoJoin, um processo para a junção de ontologias que emprega a análise de similaridade lexical, estrutural, relacional e indexação de termos como mecanismo de mapeamento. Esse mecanismo é responsável por identificar correspondências entre elementos de duas ontologias dadas como entrada. Essa correspondência é utilizada para combinar os elementos similares, resultando assim, em uma nova ontologia gerada a partir do reuso.

Foi realizado ainda, a construção de uma ferramenta capaz de dá suporte às definições apresentadas pelo processo OntoJoin, descritas neste capítulo. A ferramenta auxilia o engenheiro do conhecimento na junção de ontologias, além de prover subsídios para a elaboração da avaliação do processo proposto.

A Tabela 3.8 descreve um comparativo entre o processo OntoJoin e as demais técnicas de reuso de ontologias discutidas no capítulo 2. É possível observar que o processo proposto possui algumas vantagens, como é o caso do uso em conjunto dos mecanismos “Comparação Lexical”, que realiza uma comparação entre os rótulos dos termos dos elementos; “Comparação Estrutural”, que realiza uma

análise dos conceitos e sua respectiva estrutura hierárquica; “Comparação Relacional”, que realiza uma análise dos conceitos, suas propriedades e relações não taxonômicas; e “Indexação dos Termos”, que altera a hierarquia dos conceitos para representar melhor semanticamente os termos a serem combinados. Essa característica confere ao OntoJoin o potencial de obter maior efetividade na junção de ontologias em relação as técnicas até então propostas.

Uma outra vantagem observada é a capacidade do processo gerar ontologias compostas por todos os elementos da definição formal de ontologia, excetuando-se os elementos do conjunto I. Essa característica possibilita que o produto final gerado seja uma ontologia mais completa e por sua vez mais expressiva.

Tabela 3.8: Comparativo entre as técnicas apresentadas no estado da arte em relação ao processo OntoJoin

	CMS	FCA MERGE	CATO	iPROMPT	OntoJoin
Conteúdo da Entrada	Duas Ontologias	Duas Ontologias e um Corpus	Duas Ontologias	Duas Ontologias	Duas Ontologias
Conteúdo da Saída	Uma Ontologia Vinculada	Uma Ontologia Independente	Uma Ontologia Vinculada	Uma Ontologia Independente	Uma Ontologia Independente
Elementos Resultantes	C _C , H, I, P	C _C , H, I, P	C _C , H, P, R	C _C , P, H, R	C _C , P, H, R, A
Linguagens Suportadas	OWL	OWL	OWL e XML	OWL e RDF	OWL
Técnicas de Similaridade	Lexical, Linguística, Estrutural e Análise de Propriedades	Lexical e Estrutural	Lexical, Linguística e Estrutural	Lexical, Linguística, Estrutural e Análise de Propriedades	Lexical, Linguística, Estrutural, Análise de Propriedades, Análise de Rel. não Taxonômicas e Indexação de Termos (Grafos)
Automação	Automática	Semi-automática	Semi-automática	Semi-automática	Semi-automática

4 Avaliação

O objetivo deste capítulo é apresentar um estudo de caso, visando avaliar a efetividade do processo OntoJoin através dos valores obtidos pelas métricas de cobertura (*recall*) e precisão (*precision*) [8]. Estes valores são calculados a partir dos elementos resultantes deste processo comparando-os com valores resultantes de uma junção realizada manualmente por um especialista do domínio.

O capítulo está organizado na seguinte forma: A seção 4.1 apresenta as ontologias utilizadas e o domínio abordado. A seção 4.2 apresenta as técnicas e os procedimentos utilizados para avaliação do processo OntoJoin. A seção 4.3 demonstra os resultados obtidos, assim como uma sucinta discussão sobre estes resultados. A seção 4.4 traz as considerações finais sobre o capítulo.

4.1 Os Domínios Abordados

Guarino [17] afirma que as ontologias de aplicação herdam características das ontologias de domínio e das ontologias de tarefas. Neste sentido, a avaliação do processo OntoJoin foi realizada em um primeiro momento, através da combinação de duas ontologias: a “OntoTur”, ontologia de domínio do setor turístico, e a “Sales”, ontologia do setor de vendas. Como resultado foi obtido uma ontologia de aplicação reusada, intitulada de “OntoTurSales” ilustrada nas Figuras A1, A2, A3 e A4 (Anexo A). Também foi desenvolvida pelo especialista do domínio uma ontologia de referência, intitulada de “OntoTurSales2” ilustrada nas Figuras B1, B2, B3, B4 e B5 (Anexo B). A Figura 4.1 ilustra o modelo proposto por Guarino [17], no qual baseou-se este estudo de caso. As seções 4.1.1 e 4.1.2 apresentam respectivamente cada uma dessas ontologias.

É importante ressaltar, que para alcançar resultados mais expressivos foi estabelecido que as ontologias utilizadas não fossem de domínios similares, pois ocasionariam alguns “falsos-positivos”. Neste sentido, entende-se como “falso-positivo”, a ideia de que domínios muito similares apresentariam uma ocorrência

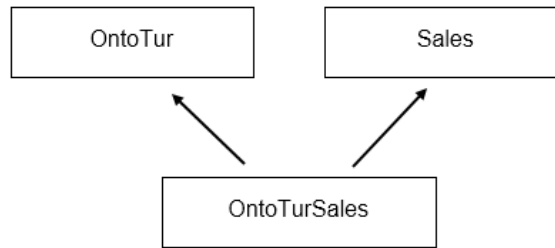


Figura 4.1: Classificação das ontologias (adaptadas a partir de [17])

mais elevada de termos similares. Desta forma, a fim de evitar a ocorrência desta problemática, foram utilizadas ontologias de domínios distintos. Contudo, embora o setor turístico esteja relacionado ao setor de vendas, ambos possuem características próprias, limitando assim, a ocorrência de similaridades explícitas pelo domínio.

4.1.1 Ontologia de Domínio: OntoTur

A ontologia OntoTur foi desenvolvida pelo grupo GESEC com o uso da ferramenta Protégé. A OntoTur descreve conhecimento acerca do domínio Turístico e consiste de duas grandes classes “Tourism” e “Services”. Da classe “Tourism” ilustrada nas Figuras 4.2 e 4.3 derivam nove grandes classes “Business”, “Congressional”, “Cultural”, “Ecological”, “Ecotourism”, “Exchange”, “Religious”, “Scientific” e “Vacation”. Estas classes descrevem os principais tipos de turismo. Da classe “Services” ilustrada nas Figuras 4.4 e 4.5 derivam quatro grandes classes “Transport”, “Food”, “Accommodation” e “Entertainment”. Estas classes descrevem os principais serviços oferecidos aos turistas. Turista é todo indivíduo que permanece por mais de 24 horas em uma localidade visitada e não realiza nenhuma atividade remunerada.

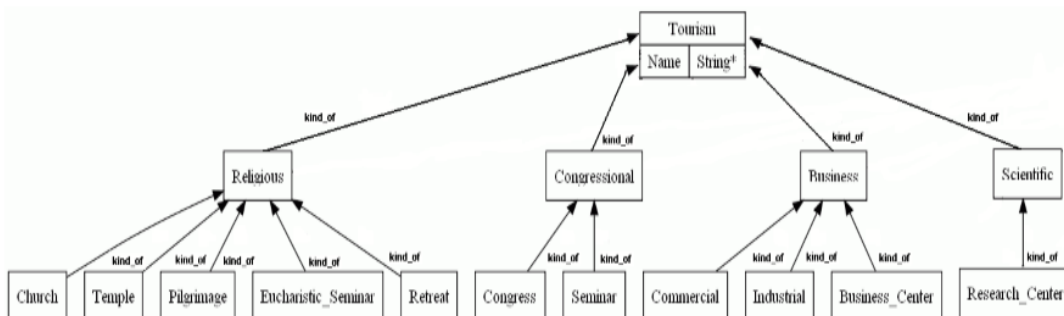


Figura 4.2: Ontologia OntoTur - Parte 1

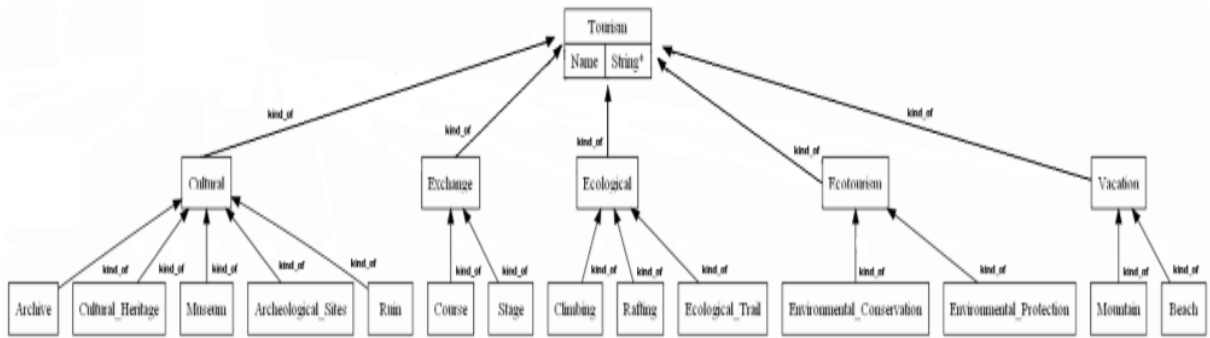


Figura 4.3: Ontologia OntoTur - Parte 2

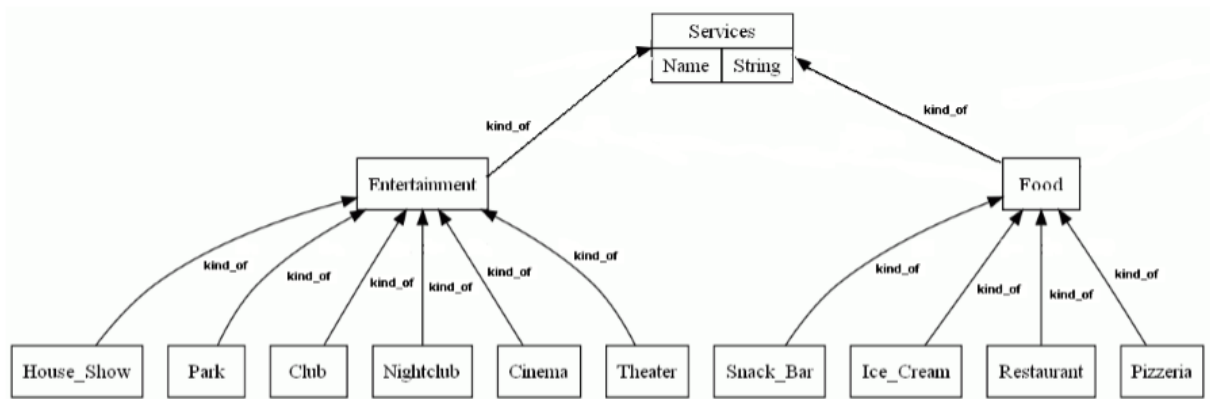


Figura 4.4: Ontologia OntoTur - Parte 3

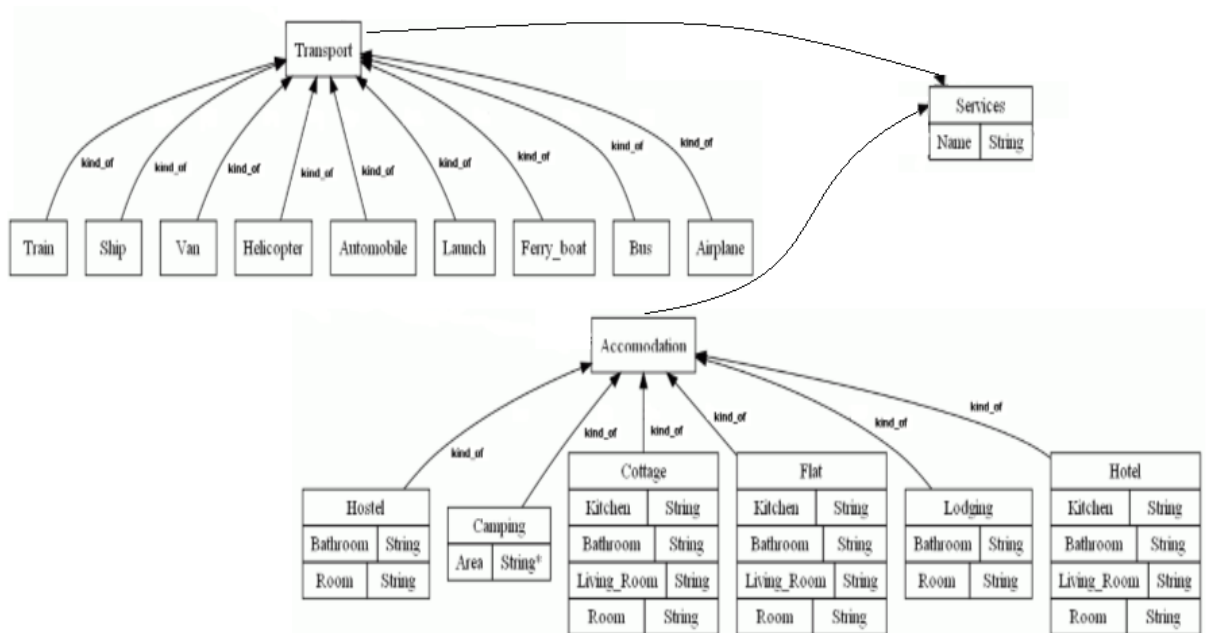


Figura 4.5: Ontologia OntoTur - Parte 4

Há diferentes tipos de turismo, conforme apresenta a hierarquia de classes da ontologia ilustrada nas Figuras 4.2 e 4.3. O Turismo de Negócios, representado pela classe "Business", é caracterizado pelo deslocamento de executivos, que afluem aos grandes centros empresariais e cosmopolitas a fim de efetuar transações e atividades profissionais, comerciais e industriais. Da classe "Business" derivam três subclasses "Comercial", "Industrial" e "Business Center".

O Turismo de Congresso, representado pela classe "Congressional", é caracterizado pelo deslocamento de turistas que se destinam a núcleos receptores eleitos para a realização de congressos e seminários de distintos assuntos e especialidades. Da classe "Congressional" derivam duas subclasses "Congress" e "Seminar".

O Turismo Cultural, representado pela classe "Cultural", é caracterizado pelo deslocamento de turistas a núcleos receptores que oferecem como produto essencial o legado histórico do homem em distintas épocas, representado a partir do patrimônio e do acervo cultural. Da classe "Cultural" derivam cinco subclasses "Archeological Sites", "Archive", "Museum", "Cultural Heritage" e "Ruin".

O Turismo Ecológico, representado pela classe "Ecological", é caracterizado pelo deslocamento de turistas para espaços naturais, com ou sem receptivos, motivados pelo desejo de estar em contato com a natureza. Da classe "Ecological" derivam três subclasses "Climbing", "Rafting" e "Ecological Trail".

O Ecoturismo, representado pela classe "Ecotourism", é caracterizado pelo deslocamento de turistas para espaços naturais delimitados e protegidos pelo estado ou controlados em parceria com associações. Da classe "Ecotourism" derivam duas subclasses "Environmental Conservation" e "Environmental Protection".

O Turismo de Intercâmbio, representado pela classe "Exchange", é caracterizado pelo deslocamento de turistas para outros países a fim de estudar ou estagiar. Da classe "Exchange" derivam duas subclasses "Stage" e "Course".

O Turismo Religioso, representado pela classe "Religious", é caracterizado pelo deslocamento de turistas que se destinam a centros religiosos, motivados pela fé em distintas crenças. Da classe "Religious" derivam cinco subclasses: "Church", "Eucharistic Seminar", "Temple", "Pilgrimage" e "Retreat".

O Turismo Científico, representado pela classe “Scientific”, é caracterizado pelo deslocamento de turistas potenciais que se dirigem a grandes centros universitários com atuação no setor de pesquisa e desenvolvimento. Da classe “Scientific” deriva uma subclasse “Research Center”.

O Turismo de Férias, representado pela classe “Vacation”, é caracterizado pelo deslocamento de turistas potenciais para grandes centros urbanos, litoral ou campo para poderem descansar e praticar lazer em família. Da classe “Vacation” derivam duas subclasses “Beach” e “Mountain”.

O Serviço Turístico é caracterizado pela prestação de serviços com o propósito de satisfazer os desejos e as expectativas do turista. Há quatro tipos de “Services”, conforme mostra a hierarquia de classes da ontologia ilustrada nas Figuras 4.4 e 4.5: “Accommodation”, “Food”, “Transport” e “Entertainment”.

O Serviço Turístico de Hospedagem, representado pela classe “Accommodation”, é caracterizado pela prestação do serviço de hospedagem ao turista. O turista pode se hospedar em hotel, albergue, pousada, chalé, camping e flat. Da classe “Accommodation” derivam seis subclasses “Hotel”, “Camping”, “Hostel”, “Cottage”, “Lodging” e “Flat”. As propriedades que caracterizam as classes “Hotel”, “Cottage” e “Flat” são: “room”, “bathroom”, “kitchen” e “living room”. A “área” é a propriedade que caracteriza a classe “Camping”. As propriedades que caracterizam as classes “Hostel” e “Lodging” são: “room” e “bathroom”.

O Serviço Turístico de Alimentação, representado pela classe “Food”, é caracterizado pela prestação do serviço de alimentação ao turista. O turista pode se alimentar em restaurante, pizzaria, sorveteria e lanchonete. Da classe “Food” derivam quatro subclasses: “Restaurant”, “Pizzeria”, “Snack Bar” e “Ice Cream”.

O Serviço Turístico de Entretenimento, representado pela classe “Entertainment”, é caracterizado pela prestação do serviço de entretenimento ao turista. O turista pode se divertir em parque de diversão, parque temático, cinema, teatro, clube, boate e casa de espetáculo. Da classe “Entertainment” derivam seis subclasses: “Park”, “Cinema”, “Club”, “Nightclub” e “House Show”.

O Serviço Turístico de Transporte, representado pela classe “Transport”, é caracterizado pela prestação do serviço de transporte ao turista. O turista pode se transportar por via aérea, marítima, ferroviária e rodoviária. Da classe “Transport”

derivam quatro subclasses: “Air Transportation”, “Ocean Transportation”, “Ground Transportation” e “Rail Transportation”. Da classe “Air Transportation” derivam duas subclasses: “Airplane” e “Helicopter”. Da classe “Ocean Transportation” derivam três subclasses: “Ship”, “Launch” e “Ferry Boat”. Da classe “Ground Transportation” derivam três subclasses: “Automobile”, “Bus” e “Van”. Da classe “Rail Transportation” deriva uma subclasse: “Train”.

4.1.2 Ontologia de Tarefas: Sales

A ontologia Sales foi desenvolvida pelo grupo GESEC com o uso da ferramenta Protégé, baseando-se na metodologia para o desenvolvimento de ontologias de tarefas definida por Martins [24]. A Sales descreve o conhecimento acerca da tarefa de Vendas e consiste de quatro grandes classes “Availability”, “Payment”, “Person” e “Resource”. A Figura 4.6 ilustra parte desta ontologia.

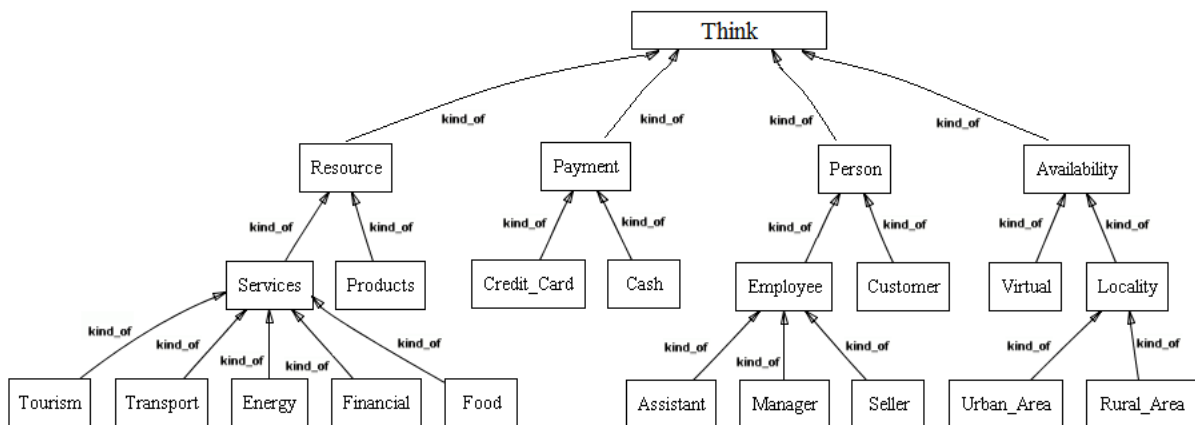


Figura 4.6: Ontologia Sales

A classe “Availability” representa o tipo de disponibilidade que algum recurso pode ser oferecido. Desta classe derivam a classe “Virtual”, que define que a disponibilidade de um recurso seja virtual; e a classe “Locality” que define que um recurso esteja localizado em algum lugar definido pelas suas subclasses: “Urban_Area” e “Rural_Area”.

A classe “Payment” representa o tipo de pagamento a ser realizado em uma venda. Desta classe derivam a classe “Cash”, que define que o pagamento será à vista; a classe “Credit_Card”, que define que o pagamento será por meio de um cartão de

crédito; e a classe "Signature", que define que o pagamento é uma assinatura, podendo esta ser semanal, mensal, semestral ou anual.

A classe "Person" representa os tipos de pessoas envolvidas na transação. Desta classe derivam a classe "Customer", que define o cliente que deseja realizar uma compra; a classe "Employee", que define os funcionários de um estabelecimento envolvidos em uma compra, como atendente, vendedor e gerente.

A classe "Resource" representa o tipo de recurso a ser oferecido por determinado estabelecimento. Desta classe derivam a classe "Products", que define produtos materiais a serem vendidos; e a classe "Services", que define os recursos intangíveis oferecidos por estabelecimento, podendo ser serviços do setor energético, financeiro, alimentício, turístico e de transportes.

4.2 Procedimentos Realizados

A avaliação experimental do processo foi realizada conforme dois procedimentos baseados no princípio de comparação da ontologia combinada com a de referência. Esse experimento consistiu em mensurar com as medidas de avaliação cobertura (*recall*) e precisão (*precision*), a efetividade do processo em combinar duas ontologias nos domínio do turismo e vendas.

Para isso, as ontologias OntoTur e Sales foram combinadas seguindo os passos definidos pelo processo OntoJoin, auxiliado pela ferramenta OntoJoinTool. A seção 4.2.2 descreve passo-a-passo os procedimentos realizados para a junção destas ontologias. A seção 4.2.2 apresenta as métricas utilizadas nesta avaliação, assim como suas características e a motivação do seu uso para este trabalho.

4.2.1 Junção das Ontologias OntoTur e Sales

A fim de avaliar a efetividade do processo OntoJoin, foi realizada a junção entre as duas ontologias definidas anteriormente. Essa combinação fez uso da ferramenta OntoJoinTool apresentada na seção 3.6, na qual auxiliou o engenheiro do conhecimento em conjunto com o especialista do domínio, a realizarem a junção seguindo as definições estabelecidas pelo processo. A seguir é apresentado um passo-a-passo dos procedimentos realizados na aplicação para a obtenção do produto final do processo.

A Figura 4.7 ilustra a seleção das ontologias, nesta etapa selecionou-se as ontologias “OntoTour” e “Sales” para serem combinadas. Também nesta etapa adicionou-se os valores dos pesos para cada medida de similaridade, conforme descrito na seção 3.2.3. Os valores dos pesos foram estabelecidos pelo engenheiro do conhecimento em conjunto com o especialista do domínio, a fim de atingirem um valor de similaridade final mais efetivo entre os elementos comparados. Os valores 0.8, 0.15 e 0.05 foram atribuídos respectivamente às medidas de similaridade “Comparação Estrutural”, “Comparação de Propriedades” e “Comparação de Relacionamentos não Taxonômicos”. A definição dos valores dos pesos foi estabelecida de acordo com a precedência das medidas de similaridade. O engenheiro do conhecimento estipulou que a medida similaridade *Sim1* possui precedência maior à *Sim2* e esta por sua vez

tem precedência maior à *Sim3*, pois para o cálculo de similaridade de *Sim1* é envolvido diretamente as classes e suas hierarquias, elementos fundamentais para ontologia.

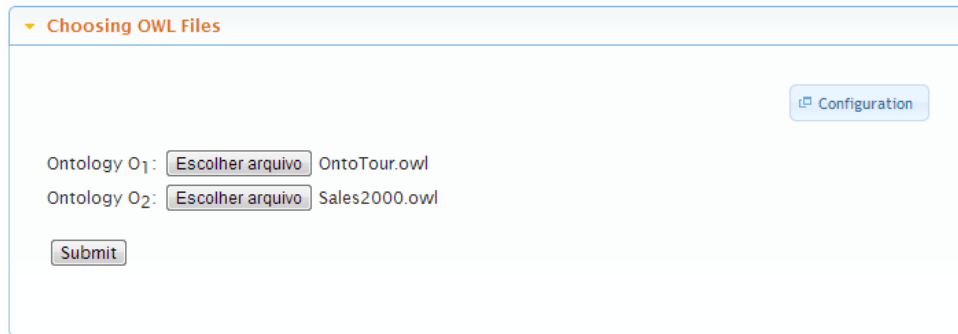


Figura 4.7: Interface para selecionar arquivos OWL e botão para configuração dos pesos das medidas de similaridade

Posteriormente, a Figura 4.8 ilustra os elementos do conjunto C_C e P extraídos a partir das ontologias dadas como entrada, a Figura 4.9 ilustra os elementos do conjunto R e H .

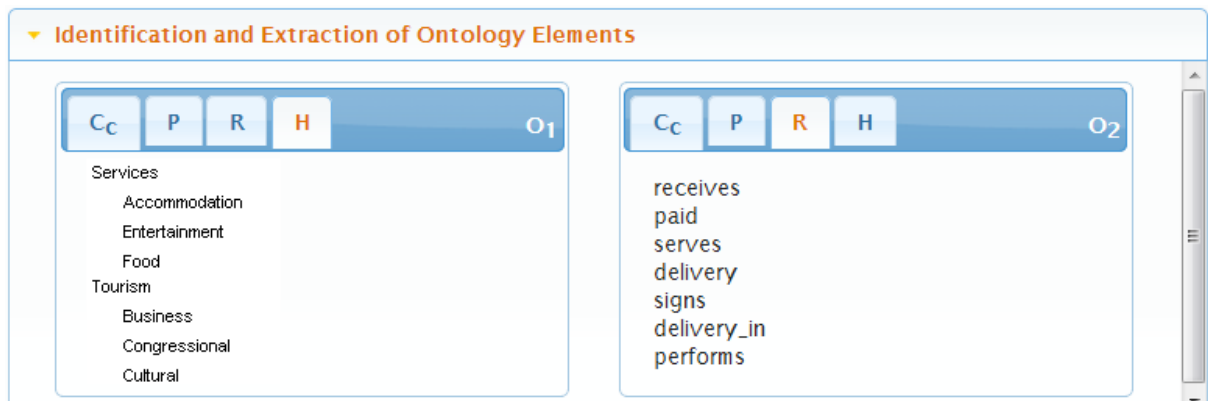


Figura 4.8: Elementos do conjunto C_C e P extraídos a partir das ontologias dadas como entrada

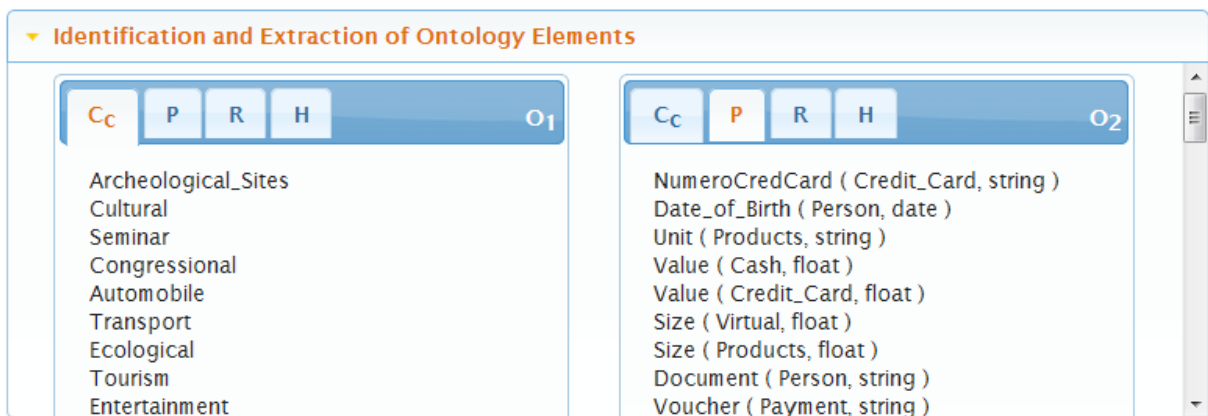


Figura 4.9: Elementos do conjunto R e H extraídos a partir das ontologias dadas como entrada

Em seguida, a ferramenta disponibiliza o valor de similaridade final entre os elementos, calculados seguindo as fórmulas e pesos definidos na seção 3.2. Nesta etapa é disponibilizado a opção de seleção para que o usuário escolha quais termos são correspondentes e posteriormente utilizados na etapa de redefinição. A Figura 4.10 ilustra os valores do similaridade obtidos após o mapeamento.

▼ Mapping			
Classes de O1	Classes de O2	=	Valor de Similaridade
Transport	Transport	=	0.685714285671428571429
Food	Food	=	0.685714285671428571429
Services	Services	=	0.63999999992000000000
Tourism	Tourism	=	0.63333333335333333333

[See All Values Similarity.](#)

Figura 4.10: Valores do similaridade obtidos após o mapeamento

Após o usuário selecionar quais termos são correspondentes, a ferramenta disponibiliza a opção de redefinir tais termos, essa redefinição pode ser por uma nomenclatura já existente ou por uma nova, que melhor expresse o significado destes termos. A Figura 4.11 ilustra está etapa.

▼ Integrity Analysis of Terms			
Classes de O1	Classes de O2	=	Terms Redefined
Transport	Transport	=	<input type="text" value="Transport"/>
Food	Food	=	<input type="text" value="Food"/>
Services	Services	=	<input type="text" value="Services"/>
Tourism	Tourism	=	<input type="text" value="Tourism"/>

Figura 4.11: Redefinição dos termos correspondentes

Em seguida, a ferramenta combina os elementos similares em um só e os indexa com os demais elementos seguindo as regras de indexação apresentadas na seção 3.2.4. Esta etapa é realizada automaticamente pela ferramenta, ao seu termino é disponibilizado a opção de download do arquivo da ontologia reusada na extensão OWL. A Figura 4.12 ilustra o termino da execução desta etapa.



Figura 4.12: Termino da execução da ferramenta

Ao fim da execução da ferramenta uma ontologia no formato OWL foi gerada, contendo todos os elementos contidos nas ontologias dadas como entrada. Em seguida, o especialista do domínio realizou novamente a junção entre as ontologias “OntoTour” e “Sales”, desta vez, não utilizou as definições abordadas pelo processo OntoJoin, mas sim uma combinação manual. Ele estabeleceu ainda, que esta ontologia gerada manualmente é uma ontologia que melhor atende as suas necessidade e desta forma foi defina como ontologia de referência. A Tabela 4.1 apresenta os valores obtidos a partir da junção manual e da junção realizada pela ferramenta OntoJoinTool.

Tabela 4.1: Quantidade de termos presentes nas ontologias resultantes

	C_C	P	R	H
Termos resultantes a partir da ontologia gerada pelo processo OntoJoin	83	48	8	21
Termos resultantes a partir da ontologia gerada pelo especialista do domínio	84	54	8	28

4.2.2 Métricas Utilizadas

A avaliação de ontologias é indispensável para que boas práticas de desenvolvimento possam ser utilizadas pelo engenheiro do conhecimento. Contudo, avaliar ontologias se torna um trabalho complexo, pois não existem abordagens bem definidas de como deve ser realizado tal procedimento [7]. Brewster et al. [1] afirmam que “Há problemas inerentes em tentar avaliar uma ontologia, uma vez que não está claro o que exatamente se está tentando avaliar”. Freitas [7] relata ainda que “boas ontologias são aquelas que servem aos seus propósitos”. Sendo assim, pode ser observado na literatura [8] [23] [48] [7] [1] que algumas técnicas de diferentes áreas são adaptadas com intuito de abordarem a esta necessidade.

Diante disso, a efetividade dos resultados foram avaliados por intermédio dos valores obtidos pelas medidas de cobertura (*recall*) e precisão (*precision*) adaptadas a partir da área de Recuperação de Informação (RI) [8]. Foram utilizadas como entrada a quantidade de termos resultantes em um conjunto, seja ele (C_C , P, R, H ou A) resultante do processo ontojoin, em relação a quantidade de termos presentes neste mesmo conjunto obtido a partir da junção realizada manualmente pelo especialista do domínio.

De acordo com Silva [40] cobertura (*recall*) é a fração de documentos relevantes recuperados, e avalia a capacidade do sistema em recuperar a informação desejada. Precisão (*precision*) é a fração dos documentos recuperados que são relevantes, e avalia se o sistema é capaz de ignorar material irrelevante. Segundo Dellschaft e Staab [8], as medidas de cobertura (*recall*) e precisão (*precision*) são utilizadas para comparar uma recuperação de referência com uma recuperação computadorizada. Neste sentido, esta adaptação das medidas de cobertura e precisão não utilizam *termos recuperados* como se estivessem calculando a efetividade de uma técnica de RI, mas recebem como entrada termos que fazem parte da composição das ontologias resultantes de um processo de junção.

Desta forma, como citado por Dellschaft e Staab [8], as medidas necessitam de um valor de referência e para este fim foi solicitado a um especialista do domínio que o mesmo realizasse uma junção manual, combinando as ontologias “OntoTur” e “Sales”, de tal forma que o resultado obtido fosse definido como retorno ideal, ou simplesmente a ontologia que melhor atenda às suas necessidades. Em contrapartida, as medidas de cobertura (*recall*) e precisão (*precision*) necessitam de mais uma entrada, esta advém de uma ontologia resultante de uma junção realizada computacionalmente, seguindo as definições do processo OntoJoin descrito na seção 3 e auxiliada pela ferramenta OntoJoinTool descrita na seção 3.6, esta ontologia deve ser idêntica ou estar o mais próximo do resultado obtido manualmente pelo o especialista do domínio.

As Fórmulas 4.1 e 4.2 apresentadas a seguir foram adaptadas a partir dos trabalhos de Dellschaft e Staab [8] e utilizadas neste contexto para calcular os valores de cobertura (*recall*) e precisão (*precision*) necessários para esta avaliação.

$$\text{Recall}(S_i, S_j) = \frac{|S_i \cap S_j|}{|S_j|} \quad (4.1)$$

$$\text{Precision}(S_i, S_j) = \frac{|S_i \cap S_j|}{|S_i|} \quad (4.2)$$

onde,

S_i : É a quantidade de termos presente em um conjunto (C_C , P, R ou H) da ontologia gerada a partir das definições do processo OntoJoin;

S_j : É a quantidade de termos presente em um conjunto (C_C , P, R ou H) da ontologia de referência gerada manualmente pelo especialista do domínio;

A fim de evitar que valores excessivamente altos ou baixos interfiram nos valores de cobertura e de precisão da ontologia como um todo. Foi definido que o cálculo dessas medidas fosse realizado separadamente para cada um dos conjuntos de termos presentes na definição formal de ontologias (C_C , P, R ou H). Isso permite que a verificação da efetividade do processo seja realizada individualmente para cada um destes conjuntos. Caso o cálculo de cobertura e de precisão fosse realizado para a ontologia completa a variação dos valores obtidos pelas métricas citadas não estariam totalmente explícitas. A próxima seção apresenta os valores obtidos, assim como uma discussão concisa sobre esses resultados.

4.3 Discussão dos Resultados Obtidos

Definido os procedimentos a serem realizados foram utilizados como entrada para calcular os valores de cobertura (*recall*) e precisão (*precision*), os termos contidos em cada conjunto da definição formal de ontologias resultantes do processo de junção, seja ela manual ou computacional. Esses valores de entrada podem ser observados na Tabela 4.1 apresentada na seção 4.2.1. A Tabela 4.2 apresenta os valores de cobertura (*recall*) e precisão (*precision*) calculados a partir de cada um dos conjuntos de termos.

A partir dos resultados apresentados na Tabela 4.2 observou-se que a ontologia gerada pelo processo OntoJoin obteve 95.2% de cobertura para o conjunto

Tabela 4.2: Percentual de cobertura e precisão

	C_C	P	R	H
Cobertura	95.2%	85.1%	62.5%	64.2%
Precisão	96.3%	95.8%	62.5%	85.7%

C_C , já para o conjunto P obteve 85.1%, para o conjunto R obteve 62.5% e para o conjunto H obteve 64.2%.

O conjunto C_C obteve o maior valor de cobertura, pois o processo busca incorporar à ontologia resultante todos os conceitos das ontologias dadas como entrada. Neste sentido, todos os elementos do conjunto C_C das ontologias iniciais estão presentes na ontologia resultante, divergindo com a ontologia comparada apenas nos casos em que a seleção da similaridade e a redefinição da nomenclatura dos termos são alteradas.

É observado também um valor de cobertura muito baixo para o conjunto R. Isto se deve ao fato de que a análise de similaridade para os termos deste conjunto leva em consideração, o rótulo do relacionamento não taxonômico, o rótulo da classe que representa o seu domínio e o rótulo da classe que representa o seu valor. Diante disso, alterações nos conceitos implicam duplamente na análise de similaridade dos termos do conjunto R, proporcionando a este conjunto um valor de cobertura menos contundente.

Além disso, o conjunto H também obteve um valor de cobertura muito baixo, pois estes termos estão ligados às regras de indexação predefinidas no processo, onde são realizadas alterações nestes termos de tal forma que encontre a melhor estrutura hierárquica para qualquer combinação de ontologias. Essa generalização acaba criando disparidades entre a taxonômia da ontologia gerada pelo o processo e a taxonômia da ontologia gerada manualmente.

Em relação ao conjunto P, este obteve o valor de cobertura 85.1%. Os termos deste conjunto estão relacionados diretamente com o conjunto C_C , como este por sua vez obteve um alto valor de cobertura, não houve variação em relação aos valores do conjunto P, pois os termos contidos neste conjunto são basicamente os mesmos contidos na ontologia de referência, divergindo somente quando o termo do conjunto C_C não existe ou quando os termos do conjunto P são renomeados.

Se tratando da precisão, observou-se que a ontologia gerada pelo processo OntoJoin obteve 96.3% de precisão para os termos do conjunto C_C , já para os termos do conjunto P obteve 95.8%, para os termos do conjunto R obteve 62.5% e para os termos do conjunto H obteve 85.7%.

Estes valores apresentam um fato já esperado, onde o valor de precisão é maior para os termos do conjunto C_C , pois ao ser analisada a similaridade dos termos deste conjunto leva-se em consideração termos do próprio conjunto C_C , além dos termos dos conjuntos P, R e H, tornando esta análise mais pertinente, assemelhando-se à análise manual.

Contudo, a Tabela 4.2 apresenta um valor de precisão muito baixo para os termos do conjunto R, pois os critérios da análise de similaridade implicam diretamente na quantidade de termos gerados, estes por sua vez apresentam divergências aos termos da ontologia comparada já que os elementos destes conjuntos sofrem duplamente alterações tanto no termo do seu domínio quanto no termo do seu valor.

Diferentemente do valor de cobertura dos termos do conjunto H, a precisão deste conjunto teve um valor relativamente alto, pois embora os critérios da análise de similaridade reduza a quantidade de termos do conjunto H presente na ontologia resultante e esta por sua vez seja reduzida ainda mais quando comparada com a ontologia de referência, torna os termos resultantes mesmo que em menor quantidade semelhantes aos termos contidos na ontologia de referência.

Em relação aos termos do conjunto P, este obteve o valor de precisão de 95.8%. Este valor está relacionado diretamente com os termos do conjunto C_C , como este obteve um valor de precisão relativamente alto, a variação aos termos do conjunto P quase não foi observada, exceto os casos em que houve disparidade entre nomenclaturas e quando houve divergências entre conceitos.

De acordo com os valores de cobertura e precisão apresentados até aqui foi possível estabelecer que, em relação as classes o processo atingiu o seu mais alto nível de efetividade, o que já é esperado de técnicas que trabalhem com ontologias, pois este é o seu elemento fundamental. Em relação ao termos do conjunto H embora tenha um baixo valor de cobertura, houve uma precisão elevada, dessa forma mesmo que não tenha a mesma estrutura hierarquica esperada pelo especialista do domínio é plausível

dizer que o produto gerado pelo processo não esteja errado, pois existem semelhanças entre os elementos deste conjunto. O conjunto P atingiu valores altos em ambas as métricas o que o torna efetivo. O conjunto R foi o único que atingiu valores baixos em ambas as medidas, um fato prejudicial ao se tratar do elemento isoladamente. Contudo, os demais valores demonstram a alta capacidade do processo em gerar ontologias de aplicação através do reúso e que mesmo os valores baixo apresentados pelo conjunto R não são suficientes para reduzir o alto nível de efetividade do processo.

4.4 Considerações Finais

Neste capítulo foi descrito um estudo de caso elaborado para avaliar a junção de ontologias através da aplicação do processo OntoJoin. Esta avaliação experimental foi desenvolvida conforme dois procedimentos baseados no princípio de comparação da ontologia combinada com uma de referência.

O experimento consistiu em mensurar com as medidas de avaliação cobertura e precisão, a efetividade do processo em combinar duas ontologias nos domínios do turismo e vendas. Foi ainda apresentada uma discussão dos resultados obtidos e a partir desta foi possível demonstrar preliminarmente a viabilidade do processo proposto na junção de ontologias, além de demonstrar a alta capacidade do processo em gerar ontologias de aplicação através do reúso.

5 Conclusão

Devido à carência de abordagens que visam o reúso de ontologias, principalmente no que diz respeito à técnica da junção, surgiu a necessidade de desenvolver um processo que auxilie o engenheiro do conhecimento durante esta fase do processo de criação e manutenção de ontologias.

A fim de atender a esta necessidade, este trabalho apresentou o OntoJoin, um processo que combina ontologias baseando-se na análise de similaridade entre os diferentes termos que as compõem. Esta análise utiliza termos presentes nos conjuntos C_C , P , R , H e A , ocasionando em uma comparação semântica mais eficiente entre as ontologias. O produto final desse processo consiste em uma ontologia composta por elementos reusados a partir das ontologias dadas como entrada.

Para suportar o processo, foi desenvolvida a OntoJoinTool, uma ferramenta de software semi-automática de interface intuitiva e de fácil navegação, que além de prover suporte as fases do processo, automatiza a criação de uma nova ontologia no formato OWL, tendo como base, elementos reusados a partir das ontologias dadas como entrada. A ferramenta permite que o engenheiro do conhecimento altere e/ou insira novos elementos à nova ontologia a ser gerada, além de realizar uma análise de integridade, visando garantir que a ontologia não apresente inconsistências em sua codificação.

Para avaliar e demonstrar a efetividade do processo perante a junção de ontologias foi apresentado um estudo de caso, que consistiu em duas integrações entre ontologias. A primeira integração foi realizada de forma manual pelo especialista do domínio. A segunda combinação foi realizada seguindo as fases descritas formalmente pelo processo OntoJoin auxiliadas pela ferramenta de software OntoJoinTool. Em seguida, calculou-se a efetividade do processo a partir das métricas cobertura "*recall*" e precisão "*precision*", utilizando como valores a quantidade de elementos das ontologias geradas a partir das respectivas integrações. Os resultados obtidos a partir desta avaliação demonstraram preliminarmente a viabilidade do processo proposto na junção de ontologias.

5.1 Contribuições e Resultados da Pesquisa

As principais contribuições desta pesquisa foram:

- Análise do estado da arte de algumas das mais bem documentadas técnicas para o desenvolvimento e reuso de ontologias.
- Uma análise comparativa das técnicas estudadas, ressaltando suas principais características bem como suas carências.
- Desenvolvimento do processo OntoJoin para o reuso de ontologias. O processo proporciona ao desenvolvedor uma forma bem definida para reusar ontologias, utilizando o mapeamento sintático e semântico entre os termos, a fim de realizar a junção entre duas ontologias.
- Desenvolvimento da ferramenta de software OntoJoinTool, permitindo que a execução do processo seja executada por um especialista do domínio em conjunto com o engenheiro do conhecimento de modo semiautomático.
- Aplicação da OntoJoinTool no domínio turístico com o objetivo de construir uma ontologia de aplicação por meio da junção de uma ontologia de domínio e uma de tarefa.

5.1.1 Publicações

A publicação "**An Approach to Join Ontologies and their Reuse in the Construction of Application Ontologies**". Phillipi Silva e Rosario Girardi. In: 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014, Varsovia. Proceedings of the 2014 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2014). Los Alamitos: IEEE, p. 424-431. 2014 ", consistiu na primeira publicação sobre o processo OntoJoin, na qual descreveu as fases que o compõem, assim como, uma demonstração de viabilidade de sua utilização durante a construção de ontologias a partir do reuso.

5.2 Trabalhos Futuros

Vários trabalhos podem ser abordados a partir dos resultados obtidos neste:

- A integração com a técnica GAODT [38] [39] contribuirá para o desenvolvimento e reuso de ontologias de aplicação, pois com auxílio desta técnica será gerada uma ontologia com um conjunto maior de axiomas. Além do mais, a técnica GAODT aborda todo o processo de criação de ontologias, podendo incorporar o processo OntoJoin, pois formaliza em uma de suas fases a possibilidade do reuso;
- A integração com outros dicionários lexicais ampliará o conjunto de sinônimos e hiperônimos usados pelo processo OntoJoin, influenciando diretamente os resultados das atividades “Análise Lexical” e “Indexação de Termos”. É válido ressaltar, que o dicionário lexical usado neste trabalho é o *Wordnet*, tornando o processo limitado ao uso de ontologias que estejam no idioma inglês. O uso de dicionários lexicais em outros idiomas permitirá ao processo OntoJoin que reúse ontologias de outros idiomas, ampliando assim o seu uso;
- A utilização de elementos que pertençam ao conjunto I da definição formal de ontologias, durante o cálculo da medida de similaridade, realizará uma comparação semântica mais concisa entre todos os elementos das ontologias, pois quanto mais conjuntos de elementos usados para o cálculo da medida de similaridade, mais preciso será o valor de similaridade obtido;
- Desenvolvimento de uma nova versão da ferramenta OntoJoinTool, capaz de fazer uso de elementos do conjunto de axiomas das ontologias. A ferramenta até então não permite a junção de elementos deste conjunto, embora seu uso fosse bem definido no processo OntoJoin;
- A avaliação apresentada neste trabalho não considerou os elementos dos conjuntos de instâncias e axiomas. Desta forma, há a necessidade de realizar uma nova avaliação, considerando estes elementos. Esta nova avaliação apresentaria valores mais conclusivos sobre a viabilidade do processo, pois contemplaria todos os elementos da definição formal de ontologia, permitindo definir com mais veemência se o processo OntoJoin é viável ou não para a junção de todos os elementos que compõe uma ontologia.

Referências Bibliográficas

- [1] BREWSTER, C., ALANI, H., DASMAHAPATRA, S., AND WILKS, Y. Data driven ontology evaluation. *International Conference on Language Resources and Evaluation - LREC* (May 2004), 24–30. Lisbon, Portugal.
- [2] CAMPOS, M. L. D. A., AND CAMPOS, M. L. M. Integração de ontologias em domínio interdisciplinar: experiência no campo da biomedicina. In *Congress International Society for Knowledge Organization - ISKO/SPAIN* (Valencia, Spain, March 2009), pp. 170–192.
- [3] CAMPOS, M. L. D. A., MARCONDES, C. H., LIRA, L., DA COSTA, L. C., CAMPOS, L. M., AND MALHEIROS, L. R. Ontologias: representando a pesquisa na área através de mapa conceitual ontologies: representing the research in the area through conceptual maps. In *Encontro Nacional de Pesquisa em Ciência da Informação* (Salvador, Outubro 2007).
- [4] CAPOBIANGO, I. P., AND SOUZA, J. F. D. Ferramentas para a compatibilização de ontologias. *Relatórios Técnicos do DCC/UFJF 001/* (2011). Juiz de Fora.
- [5] CARROLL, J. J., DICKINSON, I., DOLLIN, C., REYNOLDS, D., SEABORNE, A., AND WILKINSON, K. Jena: implementing the semantic web recommendations. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters* (2004), ACM, pp. 74–83.
- [6] CSS. <http://www.w3schools.com/css/default.asp>. [Online: accessed 16-Junho-2014].
- [7] DE FREITAS, M. C. *Elaboração automática de ontologias de domínio: discussão e resultados*. PhD thesis, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2007.
- [8] DELLSCHAFT, K., AND STAAB, S. On how to perform a gold standard based evaluation of ontology learning. In *The Semantic Web Congress - ISWC* (Athens, GA, USA, November 2006), Springer, LNCS.

- [9] EUZENAT, J., AND SHVAIKO, P. *Ontology matching*, 2nd ed. Springer-Verlag, Heidelberg (DE), 2013.
- [10] FELICÍSSIMO, C. H. Interoperabilidade semântica na web: Uma estratégia para o alinhamento taxonômico de ontologias. Master's thesis, Dissertação de Mestrado, PUC-Rio, 2004.
- [11] FELICÍSSIMO, C. H., AND BREITMAN, K. K. Uma estratégia para o alinhamento taxonômico de ontologias. In *Proceedings of the 1st Brazilian Workshop on Semantic Web (WWS'2004), Brasilia* (2004).
- [12] FERNÁNDEZ-LÓPEZ, M., GÓMEZ-PÉREZ, A., AND JURISTO, N. Methontology: from ontological art towards ontological engineering. In *Association for the Advancement of Artificial Intelligence - AAAI / Technical Report SS-97-06* (Montegancedo - Madrid, 1997), Stanford University, pp. 33–40.
- [13] GENNARI, J. H., TU, S. W., ROTHENFLUH, T. E., AND MUSEN, M. A. Mapping domains to methods in support of reuse. *International Journal of Human-Computer Studies* 41, 3 (1994), 399–424.
- [14] GIRARDI, R. Guiding ontology learning and population by knowledge system goals. In *2nd International Conference on Knowledge Engineering and Ontology Development* (2010), INSTIIC, pp. 480–484.
- [15] GÓMEZ-PÉREZ, A., CORCHO, O., AND FERNANDEZ-LOPEZ, M. *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*, first ed. Springer Publishing Company, Incorporated, 2010.
- [16] GRUBER, T. R. A translation approach to portable ontology specifications. *Knowledge Acquisition* 5, 2 (1993), 199 – 220.
- [17] GUARINO, N. Formal ontology in information systems: Proceedings of the first international conference. vol. 46. IOS press, Trento, Italy, June 6-8 1998, pp. 3–15.
- [18] GUIZZARDI, G. Desenvolvimento para e com reuso: Um estudo de caso no domínio de vídeo sob demanda. Master's thesis, Universidade Federal do Espírito Santo, 2000.

- [19] IKEDA, M., VANWELKENHUYSEN, J., AND MIZOGUCHI, R. Task ontology: Ontology for building conceptual problem solving models. *Workshop on Applications of ontologies and problem-solving models* (1998), 126–133. IOS press Amsterdam.
- [20] JAVASCRIPT. <http://www.w3schools.com/js/default.asp>. [Online: accessed 16-Junho-2014].
- [21] KALFOGLOU, Y., AND HU, B. Cms: Crosi mapping system. *results of the 2005 ontology alignment contest: Integrating Ontologies Workshop Proceedings* (2005), 77–85.
- [22] LACHTIM, F., DE SOUZA JR, H., MOURA, A. D. C., AND CAVALCANTI, M. C. Interoperabilidade de ontologias. *Monografias em Sistemas e Computação / Instituto Militar de engenharia*, 03 (Julho 2008).
- [23] MAEDCHE, A., AND STAAB, S. Measuring similarity between ontologies. In *Knowledge engineering and knowledge management: Ontologies and the semantic web*, S. B. Heidelberg, Ed. Springer, 2002, pp. 251–263.
- [24] MARTINS, A. F. Construção de ontologias de tarefa e sua reutilização na engenharia de requisitos. Master's thesis, Dissertação de Mestrado, Espírito Santo: UFES, 2009.
- [25] MCGUINNESS, D. L., VAN HARMELEN, F., ET AL. Owl web ontology language overview. *W3C recommendation 10*, 03 (2004), 10.
- [26] MENDES, W., GIRARDI, R., AND LEITE, A. Arquitetura baseada em ontologias de um agente rbc ontology-based architecture of a cbr agent. In *Information Systems and Technologies (CISTI), 2013 8th Iberian Conference on* (2013), IEEE, pp. 1–6.
- [27] MILLER, G. A., BECKWITH, R., FELLBAUM, C., GROSS, D., AND MILLER, K. J. Introduction to wordnet: an on-line lexical database. *International Journal of Lexicography* 3, 4 (1990), 235–244.
- [28] NOY, N. F., MCGUINNESS, D. L., ET AL. *Ontology development 101: A guide to creating your first ontology*, vol. 15. Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880, 2001.

- [29] NOY, N. F., AND MUSEN, M. A. Smart: Automated support for ontology merging and alignment. *Proc. of the 12th Workshop on Knowledge Acquisition, Modelling, and Management (KAW'99), Banf, Canada* (1999).
- [30] NOY, N. F., AND MUSEN, M. A. Anchor-prompt: Using non-local context for semantic matching. In *Proceedings of the workshop on ontologies and information sharing at the international joint conference on artificial intelligence (IJCAI)* (2001), pp. 63–70.
- [31] NOY, N. F., AND MUSEN, M. A. Promptdiff: A fixed-point algorithm for comparing ontology versions. In *Eighteenth National Conference on Artificial Intelligence* (Menlo Park, CA, USA, 2002), American Association for Artificial Intelligence, pp. 744–750.
- [32] NOY, N. F., AND MUSEN, M. A. The prompt suite: interactive tools for ontology merging and mapping. *International Journal of Human-Computer Studies* 59, 6 (2003), 983–1024.
- [33] PHP. <http://www.php.net>. [Online: accessed 16-Junho-2014].
- [34] PINTO, H. S., AND MARTINS, J. Reusing ontologies. *Association for the Advancement of Artificial Intelligence - AAI / Spring Symposium on Bringing Knowledge to Business Processes 2* (2000).
- [35] PINTO, H. S., AND MARTINS, J. P. A methodology for ontology integration. In *Proceedings of the 1st international conference on Knowledge capture* (2001), ACM, pp. 131–138.
- [36] PROTÉGÉ. <http://protege.stanford.edu/support.php>. [Online: accessed 16-Junho-2014].
- [37] RUSSELL, S. J., AND NORVIG, P. *Artificial intelligence: a modern approach (3rd edition)*. Prentice Hall, 2009.
- [38] SANTOS, L. E. Uma tecnica e ferramenta orientadas por objetivos para a construcao de ontologias de aplicação. Master's thesis, Dissertação de Mestrado em Engenharia de Eletricidade - Universidade Federal do Maranhão, 2012.

- [39] SANTOS, L. E., GIRARDI, R., AND NOVAIS, P. A case study on the construction of application ontologies. In *Information Technology: New Generations (ITNG), 2013 Tenth International Conference on (2013)*, IEEE, pp. 619–624.
- [40] SILVA, F. Um modelo de recuperação de informação para a web semântica. Master's thesis, Dissertação de Mestrado em Engenharia de Eletricidade - Universidade Federal do Maranhão, 2009.
- [41] SILVA, V. Uma abordagem para alinhamento de ontologias biomédicas para apoiar a anotação genômica. Master's thesis, PPGI, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil, 2010.
- [42] SIMPERL, E. Reusing ontologies on the semantic web: A feasibility study. *Data Knowl. Eng.* 68, 10 (outubro 2009), 905–925. Elsevier.
- [43] SOWA, J. F. *Knowledge representation: logical, philosophical, and computational foundations*, vol. 511. Brooks/Cole, 1994.
- [44] STUMME, G., AND MAEDCHE, A. Fca-merge: Bottom-up merging of ontologies. In *International Joint Conferences on Artificial Intelligence - IJCAI (2001)*, vol. 1, pp. 225–230.
- [45] TAI, K.-C. The tree-to-tree correction problem. *Journal of the ACM (JACM)* 26, 3 (1979), 422–433. New York, USA.
- [46] USCHOLD, M., AND KING, M. Towards a methodology for building ontologies. In *In Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-International Joint Conference on AI-95 (1995)*, University of Edinburgh, pp. 15–30.
- [47] VAN HEIJST, G., SCHREIBER, A. T., AND WIELINGA, B. J. Using explicit ontologies in kbs development. *International journal of human-computer studies* 46, 2 (1997), 183–292. Elsevier.
- [48] VELARDI, P., NAVIGLI, R., CUCCHIARELLI, A., AND NERI, F. Evaluation of ontolearn, a methodology for automatic learning of domain ontologies. *Ontology Learning and Population (2005)*.

- [49] WANG, J. T.-L., SHAPIRO, B. A., SHASHA, D., ZHANG, K., AND CURREY, K. M. An algorithm for finding the largest approximately common substructures of two trees. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 20, 8 (1998), 889–895.
- [50] XHTML. http://www.w3schools.com/html/html_xhtml.asp. [Online: accessed 16-Junho-2014].

A Anexo: Ontologia de Aplicação - OntoTurSales

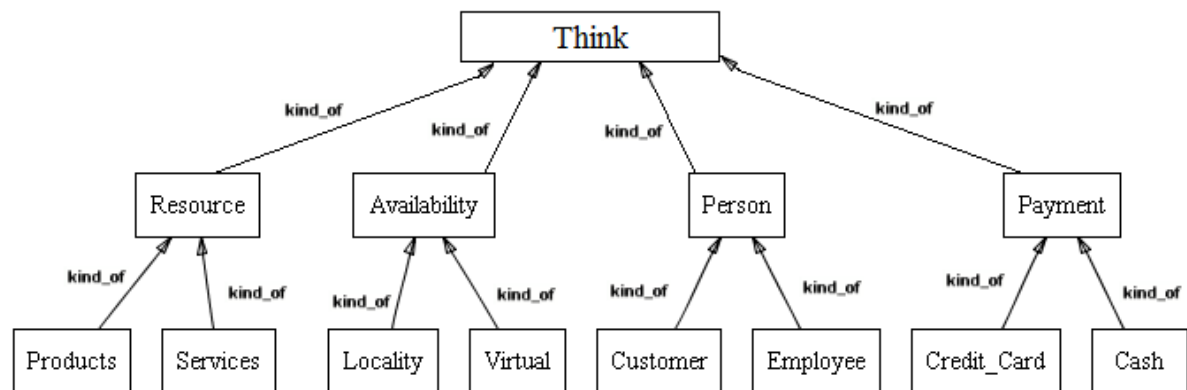


Figura A.1: Ontologia OntoTurSales - Parte 1

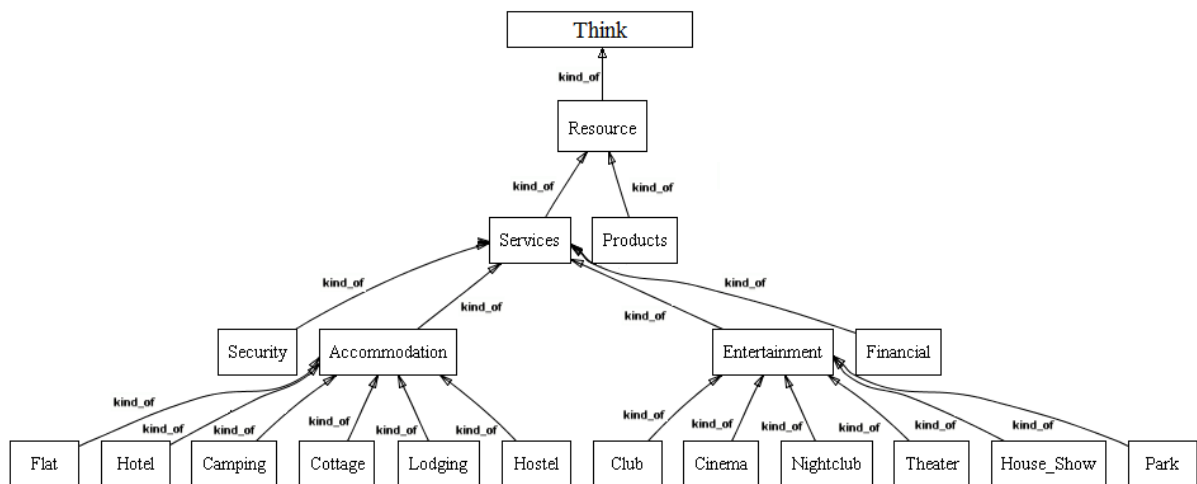


Figura A.2: Ontologia OntoTurSales - Parte 2

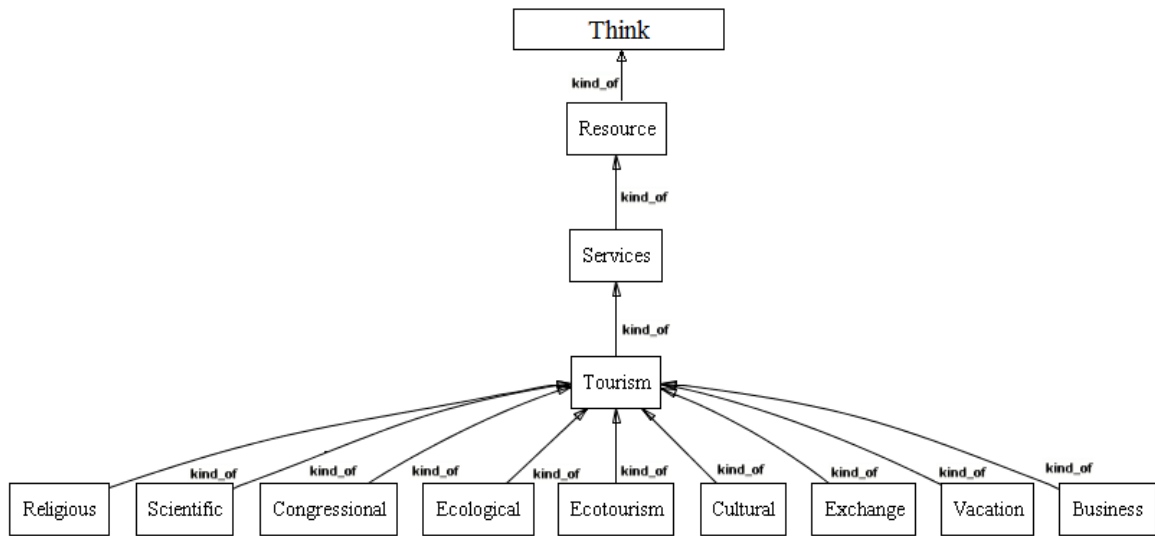


Figura A.3: Ontologia OntoTurSales - Parte 3

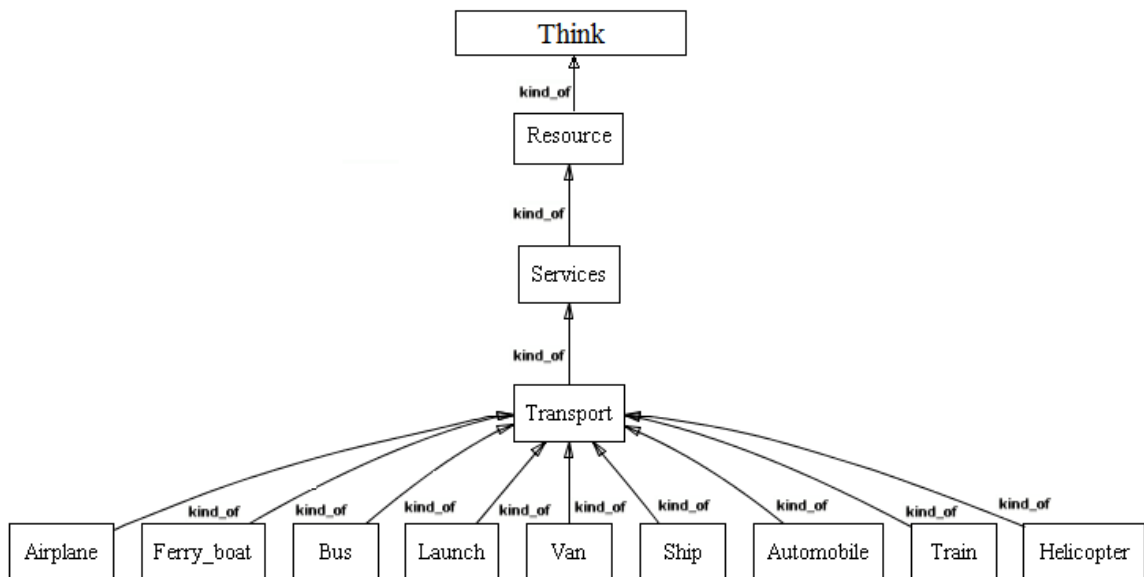


Figura A.4: Ontologia OntoTurSales - Parte 4

B Anexo: Ontologia de Aplicação (Referência) - OntoTurSales2

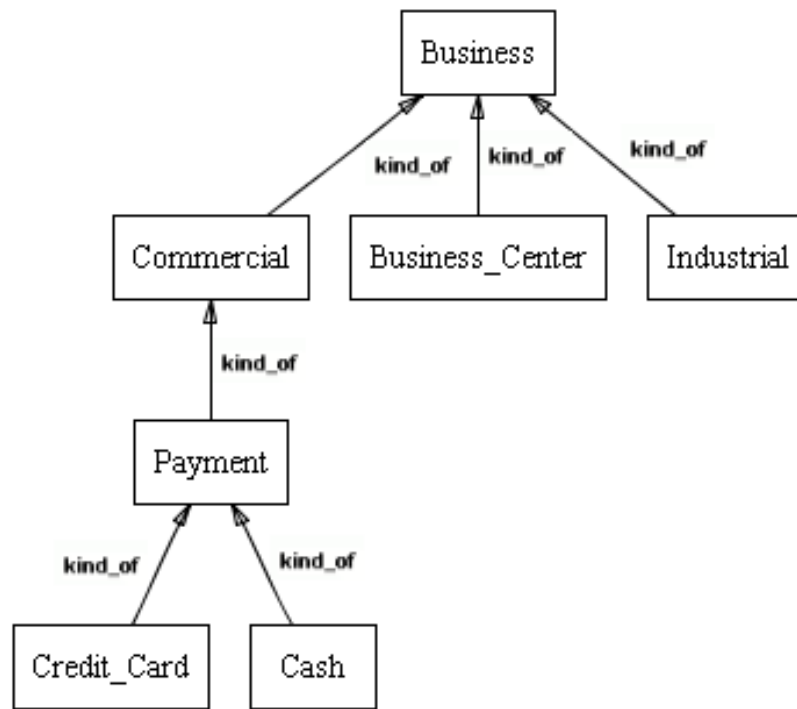


Figura B.1: Ontologia OntoTurSales2 - Parte 1

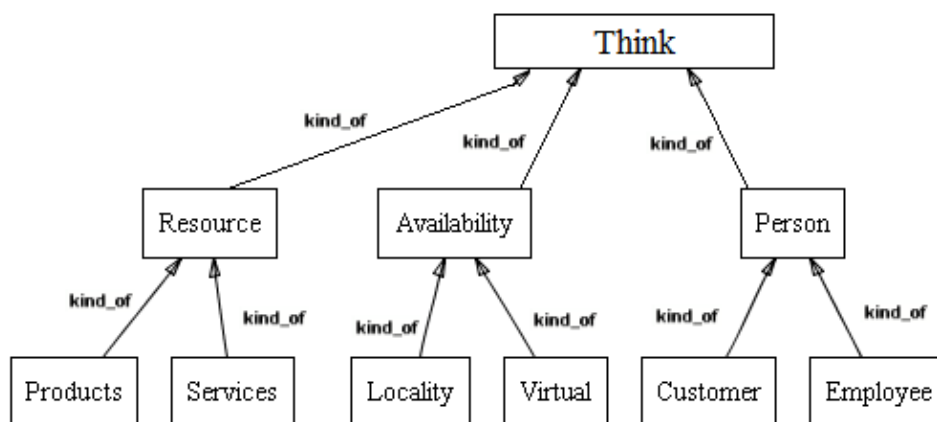


Figura B.2: Ontologia OntoTurSales2 - Parte 2

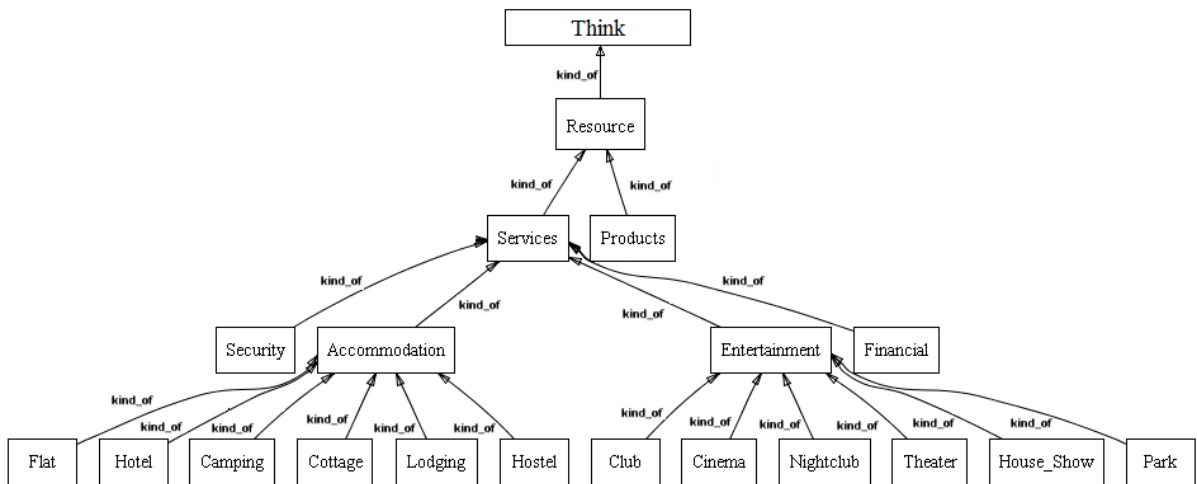


Figura B.3: Ontologia OntoTurSales2 - Parte 3

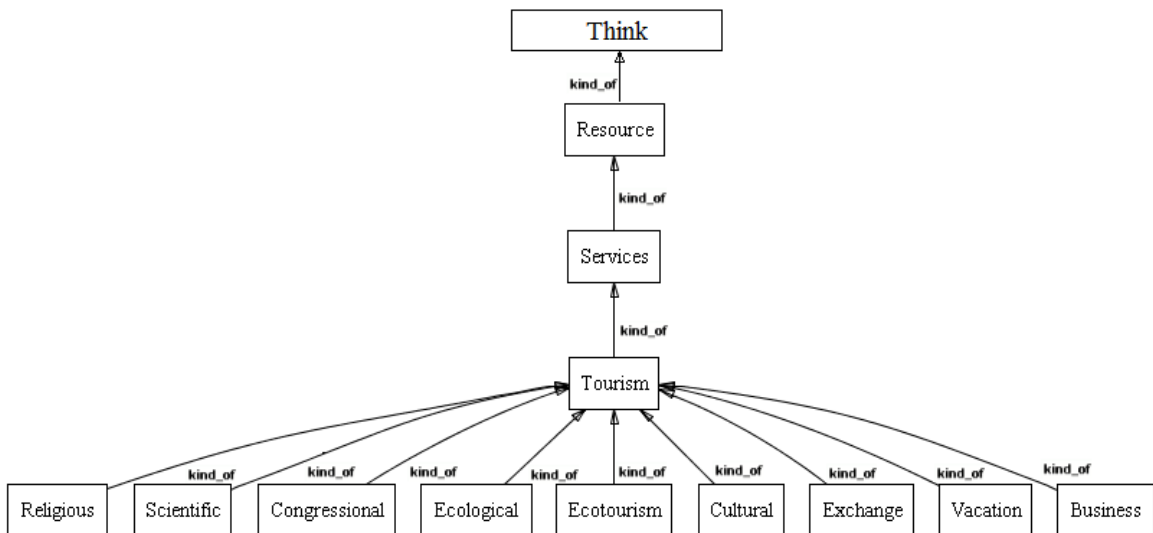


Figura B.4: Ontologia OntoTurSales2 - Parte 4

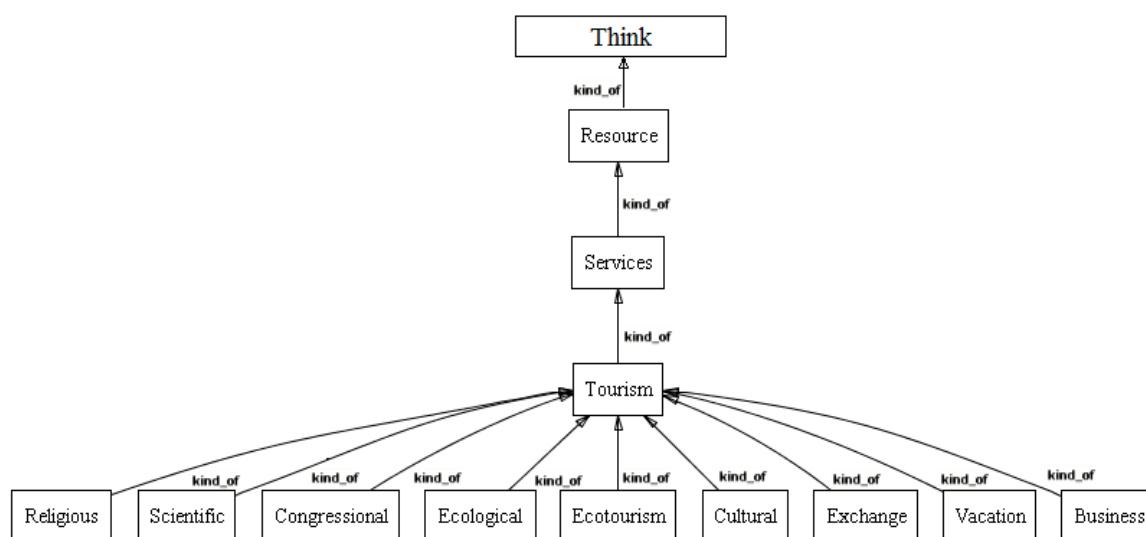


Figura B.5: Ontologia OntoTurSales2 - Parte 5