

# Colaboração em Sistemas Multiagentes Modelados por Ontologias

Verlani Timm Hinz<sup>1</sup>, Luiz Antônio Moro Palazzo<sup>1</sup>

<sup>1</sup>Programa de Pós-graduação em Informática – Universidade Católica  
de Pelotas (UCPel)  
{vertimm,lpalazzo}@ucpel.tche.br

## *Abstract*

Ontology-based collaboration is an interesting theme, once it allows modeling the knowledge of agents supporting interaction and information exchange. This paper aims to show the application of interoperability mechanisms of existing currently, applied to collaborative agents modeling through ontologies. The application of the mechanisms on the ontologies environment and agents may produce a dynamic environment propitious to collaboration. It approaches the main mechanisms of interoperability: combination, integration, alignment and mapping, and their application in collaborative systems.

Key-words: Collaborative agents, ontologies, interoperability mechanisms.

## *Resumo*

A colaboração baseada em ontologias é um tema interessante, pois permite modelar o conhecimento dos agentes permitindo a interação e troca de informações. Este artigo tem por objetivo mostrar a aplicação dos mecanismos de interoperabilidade existentes atualmente, aplicados na modelagem dos agentes colaborativos através de ontologias. A aplicação dos mecanismos sobre a ontologia do ambiente e as dos agentes pode produzir um ambiente dinâmico propício à colaboração. São abordados os principais mecanismos de interoperabilidade: combinação, integração, alinhamento e mapeamento e sua aplicação em sistemas colaborativos.

*Palavras-chave* – Agentes colaborativos, ontologias, mecanismos de interoperabilidade.

## 1 Introdução

O crescimento da Internet tem popularizado de forma significativa o conceito de Agentes Inteligentes, ou seja, aplicações capazes de executar tarefas de forma inteligente e autônoma, tais como agentes responsáveis em efetuar transações de negócios ou tutores virtuais [4].

Conceitualmente [15][18], entende-se por agente autônomo como um sistema fundamentado em uma base de conhecimento, que percebe um ambiente, seja este um mundo físico, representação gráfica, uma coleção de outros agentes ou ambientes complexos; raciocina para interpretar estas percepções; desenvolve inferências; resolve problemas e determina ações, agindo sobre o ambiente para realizar um conjunto de metas ou tarefas para as quais foi projetado. A aplicação que possui mais de um agente é chamada de um sistema multi-agente (SMA). Além disso, quando estes trocam experiências chamamos de agentes colaborativos.

Agentes colaborativos são capazes de cooperar com outros agentes a fim de trocar conhecimento.

Atualmente existe um grande número de aplicações baseadas em agentes colaborativos, como sistemas web, comunidades virtuais, etc.

Neste trabalho a proposta é modelar o conhecimento dos agentes através de ontologias, para que possam interagir de forma integrada trocando informações e contribuindo para gerar novos conhecimentos.

## 2 Ontologias

### 2.1 Definição

A palavra ontologia tem a sua origem na Filosofia, onde significa uma explicação sistemática da existência. A disciplina de computação resgatou este termo, inserindo-o num novo contexto, apesar de próximo do significado original. Neste sentido, apesar dos múltiplos significados do ser, ele faz referência a um único princípio, que une todos estes múltiplos significados.

No entanto, é impossível representar o mundo real, ou mesmo alguma parte dele, com todos os detalhes. Para representar algum fenômeno ou parte do mundo, que nós chamamos domínio, é necessário focar em um limitado número de conceitos que são suficientes e relevantes para criar uma abstração do fenômeno à disposição. Assim, um aspecto central, de qualquer atividade de modelagem consiste de desenvolver uma conceitualização: um conjunto de regras informais que confine a estrutura de uma parte da realidade, o qual um agente utiliza para isolar e organizar objetos e relações relevantes. [8]

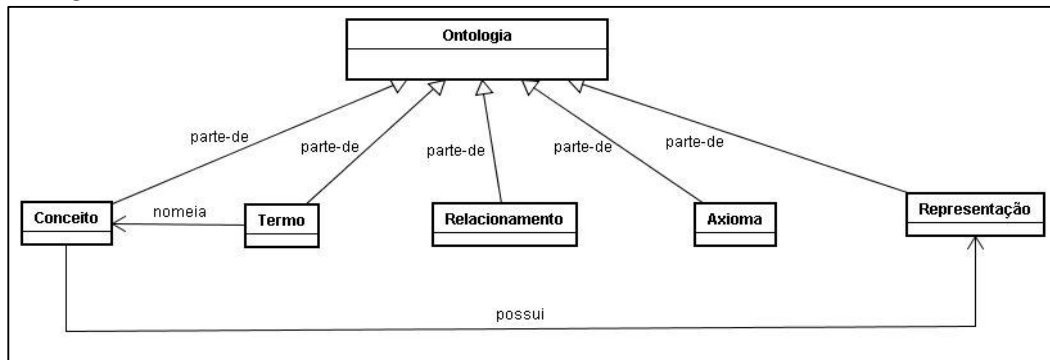
Muitos trabalhos sobre ontologias são encontrados na literatura e são apresentadas diversas definições sobre o termo, no entanto, [6] apresenta uma definição simples e completa: "Uma ontologia é uma especificação formal e explícita de uma conceitualização compartilhada". Nessa definição, "formal" significa legível para computadores; "especificação explícita" diz respeito a conceitos, propriedades, relações, funções, restrições, axiomas, explicitamente definidos; "compartilhado" quer dizer conhecimento consensual; e

"conceitualização" diz respeito a um modelo abstrato de algum fenômeno do mundo real. A conceitualização é formada por um vocabulário controlado que é arranjado hierarquicamente e através de relações entre conceitos, como nas taxonomias e tesouros. Uma conceitualização é uma visão abstrata e simplificada do mundo que se deseja representar.

## 2.2 Modelagem

As informações contidas em uma ontologia são modeladas através de classes, que são os conceitos do domínio, sendo organizadas em hierarquias, definindo relacionamentos de generalização (subclasses → superclasse) e especialização (superclasse → subclasse).

A figura 1, ilustra um diagrama contendo a descrição básica de uma ontologia e seus elementos básicos.

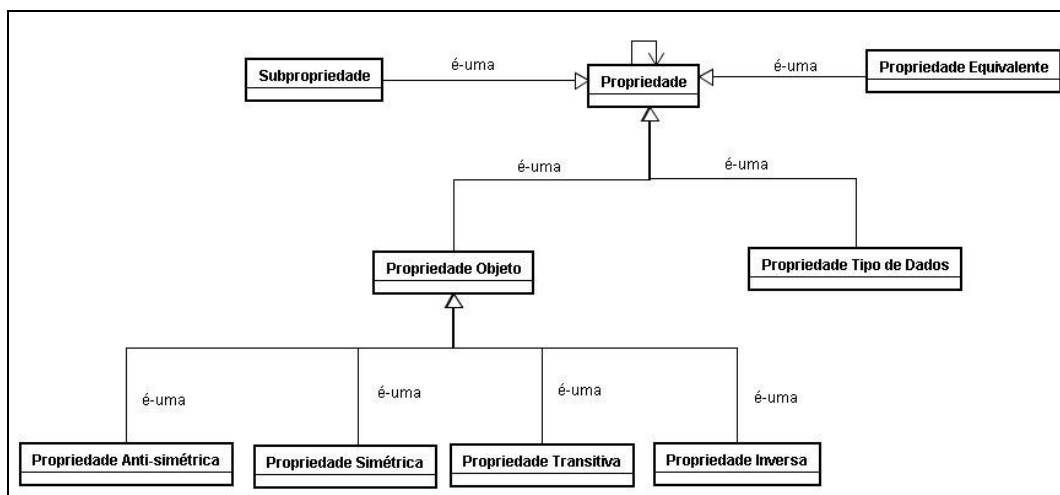


**Figura 1: Representação hierárquica de uma ontologia**

No entanto, somente as classes não fornecerão informação suficiente para responder às perguntas da competência do escopo. Uma vez que definidas as classes, deve-se descrever a estrutura interna dos conceitos. Para isso devem ser definidas também as propriedades (atributos e relacionamentos), sendo que cada classe é caracterizada por um conjunto de atributos e relacionamentos.

As propriedades de uma classe são herdadas por suas subclasses, portanto as propriedades somente precisam ser definidas em um dos níveis da hierarquia.

A figura 2 mostra a hierarquia de propriedades, onde, as propriedades são especializadas em Propriedade Objeto e Propriedade de Tipo de Dados.



**Figura 2: Hierarquia de propriedades**

Uma propriedade é declarada como Propriedade Objeto quando tem o papel de relacionar uma classe à outra classe.

A Propriedade de Tipo de Dados se diferencia da Propriedade Objeto por utilizar uma variável para representar qualquer coisa no domínio abordado. Neste tipo de propriedade, também é necessário definir o domínio a qual ela pertence e o seu valor que, não mais será uma classe, mas um elemento do tipo string, boolean, int, entre outros.

Cada propriedade é modelada por um conjunto de restrições que definem os valores que podem ser assumidos, os quais descrevem ou limitam o conjunto de valores possíveis para uma propriedade. As restrições mais comuns são a cardinalidade do valor da propriedade, seus limites, strings, números, elemento de um conjunto, etc.

As instâncias são entidades do domínio que atendem às especificações de uma classe. A atividade de instanciação corresponde a criar os registros de uma base de dados a partir do seu esquema (descrição das classes).

### 3 Mecanismos de Interoperabilidade de ontologias

De forma geral, as ontologias tem como objetivo promover um entendimento comum e compartilhado sobre um domínio, que pode ser comunicado entre pessoas e sistemas de aplicação [1]. No entanto para que trabalhem em conjunto, trocando automaticamente as informações representadas em ontologias, são necessários mecanismos que garantam a sua interoperabilidade.

Abaixo são destacados alguns mecanismos que podem ser usados para propiciar a compatibilidade de ontologias:

**Combinação de ontologias:** tem-se como resultado a versão das ontologias originais combinadas em uma ontologia única com todos seus

termos juntos, sem a definição clara de suas origens. Normalmente as ontologias originais descrevem domínios similares ou de sobreposição [11].

A figura 3, ilustra um exemplo de combinação de ontologias, onde os dois conceitos compatíveis, por exemplo **carro** da ontologia **O1** e **veículo** da ontologia **O2**, poderiam ser combinados, i.e., unidos, gerando uma ontologia única **O3**.

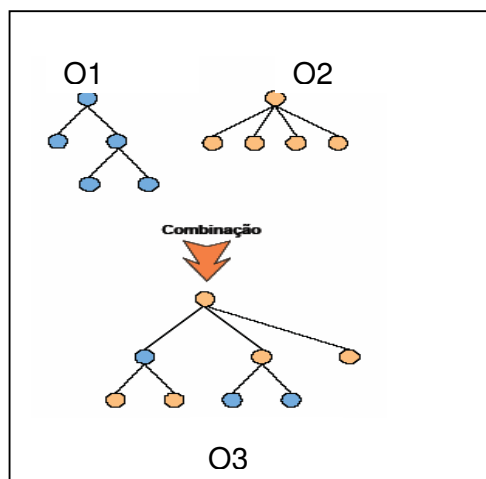


Figura 3: Exemplo do mecanismo de combinação de ontologias

**Integração de ontologias:** tem-se como resultado uma ontologia única criada pela montagem, extensão, especialização ou adaptação de outras ontologias que tratam não necessariamente do mesmo assunto. Na integração de ontologias é possível identificar as regiões que foram criadas a partir das ontologias originais [13].

A figura 4 ilustra um exemplo de integração de ontologias, onde os conceitos **carro** de **O1**, **veículo** de **O2** e **casa** de **O3** são integrados, i.e., unidos, na ontologia única **O**.

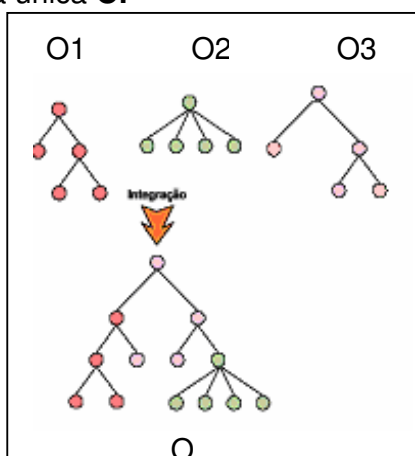


Figura 4: Exemplo do mecanismo de integração de ontologias

Apesar do resultado final, tanto da combinação quanto da integração de ontologias ser uma ontologia única constituída pela união dos termos das ontologias originais, a principal diferença entre estes dois mecanismos é que no primeiro as ontologias tratam do mesmo assunto, o que não acontece necessariamente no segundo.

**Alinhamento de ontologias:** tem-se como resultado as duas ontologias originais separadas, mas com as ligações estabelecidas entre elas, permitindo que as ontologias alinhadas reusem as informações uma das outras. O alinhamento normalmente é realizado quando as ontologias são de domínios complementares [11].

A figura 5 ilustra um exemplo de alinhamento de ontologias, onde a ontologia **O1** se refere a **carros**, a ontologia **O2** se refere a **veículos** e a **O3** a **meios de transporte**.

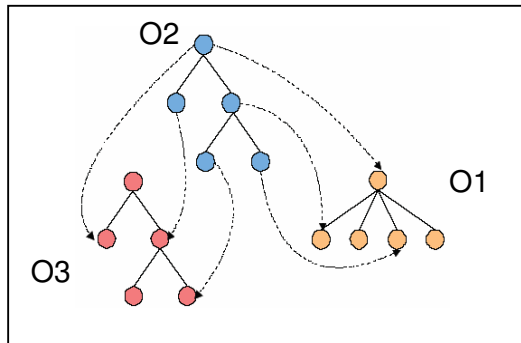


Figura 5: Exemplo do mecanismo de alinhamento de ontologias

**Mapeamento de ontologias:** tem-se como resultado uma estrutura formal que contém expressões que fazem a ligação de conceitos de um modelo em conceitos de um segundo modelo. Este mapeamento pode ser usado para transferir instâncias de dados, esquemas de integração, esquemas de combinação e tarefas similares [12].

A figura 6 ilustra um exemplo de mapeamento de ontologias, onde os conceitos **carro** de **O1** e **veículo** de **O2** são mapeados em expressões formais.

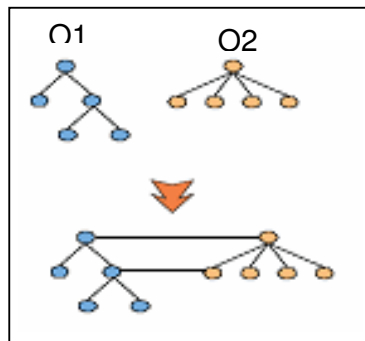


Figura 6: Exemplo do mecanismo de mapeamento de ontologias

A fim de minimizarem possíveis problemas nos mecanismos de interoperabilidades citados acima, devem ser utilizados bons algoritmos de identificação, principalmente no que diz respeito a problemas de identificação dos termos a serem alinhados em ontologias comparadas, como: presença de inconsistências encontradas nas comparações dos termos de ontologias; falta de completude no alinhamento de todos seus possíveis termos e risco de inviabilidade da estratégia frente às inconsistências existentes.

O processo de alinhamento de ontologias pode ser realizado de forma automática, semi-automática, onde há a necessidade de intervenção humana em algumas etapas para a tomada de decisão, ou até mesmo manual.

Atualmente, apesar dos trabalhos realizados para garantir mecanismos que suportem a interoperabilidade de ontologias, não foi encontrada solução alguma para o alinhamento de ontologias, que esteja totalmente de acordo com os requisitos da Web Semântica. No entanto, boas idéias surgiram para elaborar uma estratégia que esteja em conformidade com tais requisitos.

Dentre elas, podemos citar o conjunto de ferramentas PROMPT apresentado em [12] Dentre as ferramentas deste conjunto, citamos: *iPROMPT*, uma ferramenta interativa para combinação de ontologias; *AnchorPROMPT*, uma ferramenta automática baseada em grafos para alinhamento de ontologias; *PROMPTFactor*, uma ferramenta para extração de partes de ontologias e *PROMPTDiff*, uma ferramenta para identificação de diferenças entre duas versões da mesma ontologia. Além disso, o Protegé, possui um *plugin* chamado *PromptTab* onde se é possível aplicar mecanismos de interoperabilidade, conforme demonstra a figura 7.

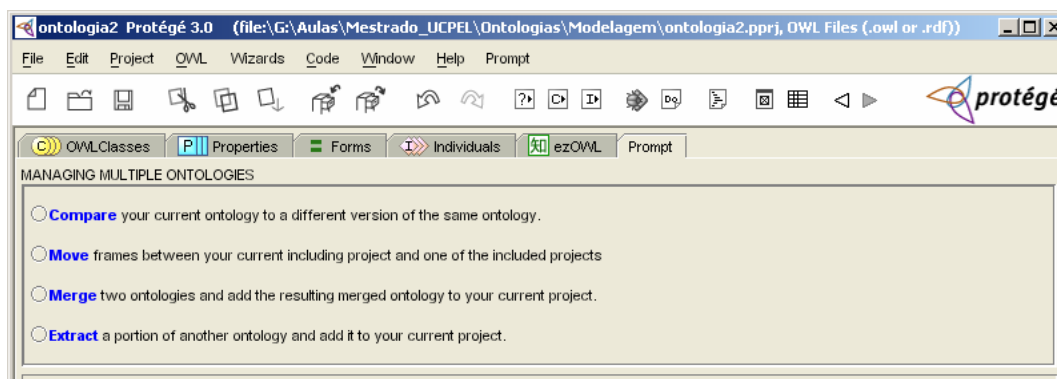


Figura 7: *Plugin PromptTab* disponível no ambiente *Protege*

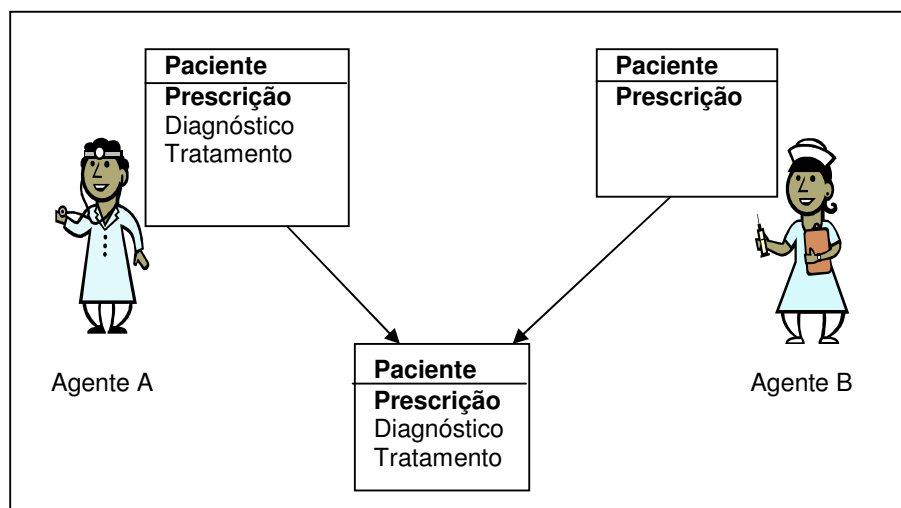
## 4 A aplicação dos mecanismos

Os mecanismos de interoperabilidade citados acima podem ser aplicados em um ambiente de agentes orientado por ontologias.

Considerando dois agentes colaborativos A e B modelados através de uma ontologia A e B, que possuem suas classes, propriedades, restrições e instâncias. Cada classe possui uma definição associada, assim como os

termos sinônimos, sendo os mesmos definidos em um glossário de termos do domínio em questão.

No mecanismo de combinação de ontologias para que seja possível efetuar a troca de conhecimentos, dois agentes com domínios similares<sup>1</sup> ou de sobreposição<sup>2</sup>, interagem formando uma única ontologia. No caso, um agente colaborador A com conhecimentos de enfermagem e um agente colaborador B com conhecimentos de Medicina, cada um tem na sua ontologia os conhecimentos específicos de sua área, no momento que eles interagirem para trocarem informações será formada uma única ontologia com as informações que as suas área possuem em comum, demonstrado na figura 8.



**Figura 8: Exemplo de combinação de ontologias**

No exemplo, a classe paciente possui a propriedade prescrição, que é definida como propriedade objeto, justamente por ser comum as áreas de medicina e enfermagem, podendo se relacionar e formar uma única ontologia. As demais propriedades são definidas como propriedade de tipo de dados.

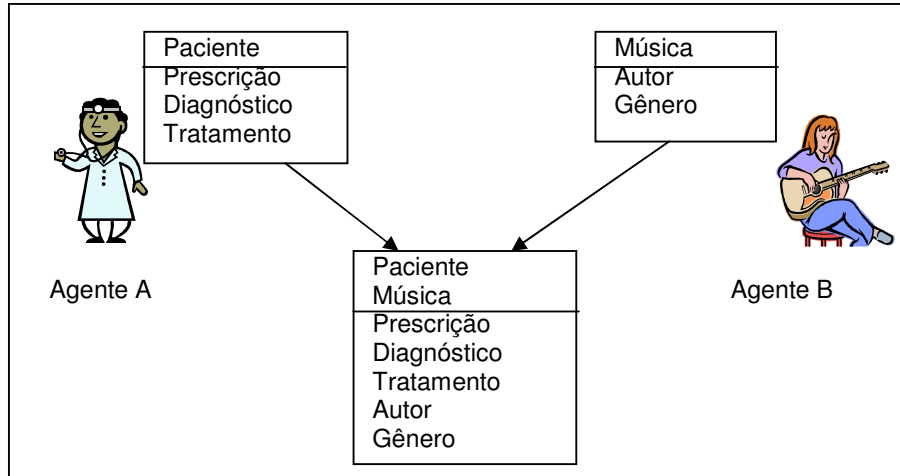
No mecanismo de integração de ontologias para que seja possível efetuar a troca de conhecimentos, dois agentes de domínios diferentes são integrados e formam uma única ontologia. Neste caso um agente colaborador A com conhecimentos de Medicina e outro agente colaborador B com conhecimentos de Música, por exemplo, quando interagirem irão formar uma

<sup>1</sup> Neste trabalho, entende-se por domínios similares aqueles domínios que tratam do mesmo assunto.

<sup>2</sup> Neste trabalho, entende-se que um domínio A qualquer é de sobreposição a um domínio B qualquer, quando A além de possuir as informações contidas em B também possui informações adicionais sobre o mesmo assunto de B.

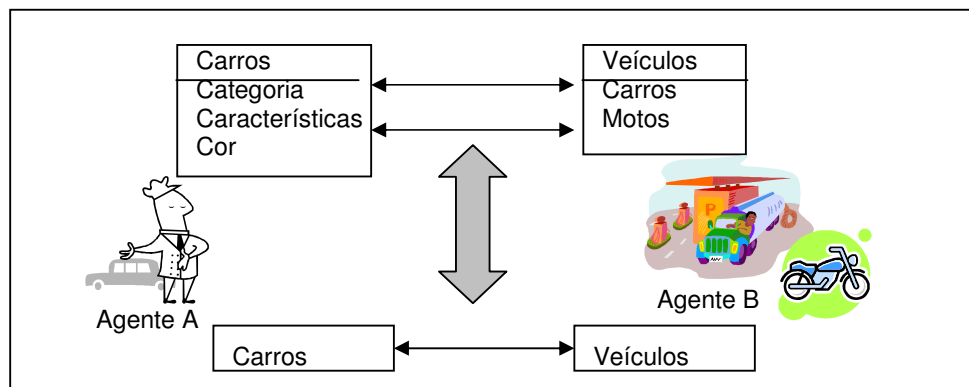


única ontologia contendo todos os dados das 2 ontologias, ou seja, será feita uma união das ontologias, demonstrado na figura 9.



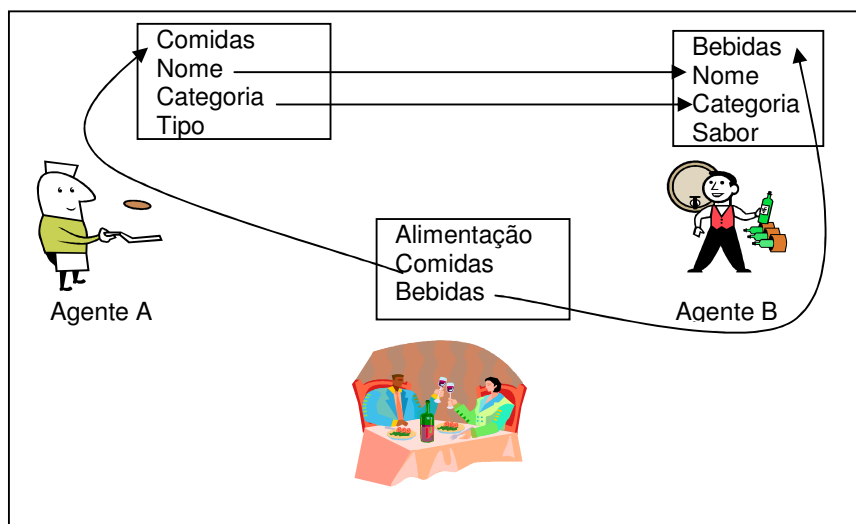
**Figura 9: Exemplo de integração de ontologias**

Para o mecanismo de mapeamento de ontologias, devem ser mapeadas as relações, instâncias de dados, esquemas de integração e tarefas similares, tendo como resultado uma estrutura formal com expressões que ligam os termos de uma ontologia nos termos de outra ontologia. Como demonstração de exemplo poderíamos ter um agente colaborador A com conhecimentos de carro e um agente colaborador B com conhecimentos de veículos, conforme figura 10.



**Figura 10: Exemplo de mapeamento de ontologias**

O mecanismo de alinhamento de ontologias, acontece quando os domínios das ontologias são complementares<sup>3</sup>, sendo que são geradas 2 ontologias separadas mas com as ligações estabelecidas entre elas. Seria o caso de um agente colaborador A com conhecimentos específicos de comidas e outro agente colaborador B com conhecimentos de bebidas, ambos pertencem ao ramo da alimentação e quando feito o alinhamento haverão propriedades que serão complementares, conforme figura 11.



**Figura 11: Exemplo de alinhamento de ontologias**

No exemplo o agente colaborador A possui conhecimentos específicos de comidas em geral, enquanto o agente colaborador B é especializado em vinhos, no entanto, ambos pertencem ao ramo da alimentação, quando alinhados suas ontologias se complementarão.

#### 4.1 Representação da ontologia de agentes colaborativos

A fase final no desenvolvimento de uma ontologia é realizada por meio de uma representação específica que permite o processamento e a abrangência do conhecimento pela máquina.

Isso é possível através de uma linguagem específica para a criação de ontologias e de uma ferramenta que permita sistematizar e integrar as especificações definidas à linguagem utilizada.

<sup>3</sup> Neste trabalho, entende-se que um domínio A qualquer é complementar a um domínio B qualquer, quando A adiciona informações a B. Normalmente, domínios complementares tratam de assuntos diferentes, mas suas partes comuns tratam de um mesmo assunto.

No desenvolvimento das ontologias de agentes, foi utilizada a linguagem OWL e a na implementação utilizou-se o editor Protégé [14]. O Protégé possui uma interface de fácil utilização para a criação de instâncias e uma quantidade considerável de *plugins* que aumentam o número de funcionalidades do Protégé.

A OWL é baseada na Lógica de Descrições, havendo uma correspondência entre as linguagens XML para expressar ontologias e uma lógica de descrição. Foram definidas três sub-linguagens OWL cada uma correspondendo a uma lógica de descrição (*OWL Lite, DL e Full*).

As lógicas de descrições, são formalismos utilizados para representar o conhecimento e raciocinar sobre ele. Isso significa que a sintaxe deste formalismo foi definida para facilitar o raciocínio, tornando-o computacionalmente menos custoso. [2]

A sintaxe das lógicas de descrições é formada por símbolos representando conceitos e papéis, construtores e quantificadores. Os conceitos são representações de classes, conjunto de indivíduos que apresentam as mesmas características gerais. [2]

## 5 Trabalhos futuros

Para que várias ontologias trabalhem em conjunto, trocando automaticamente as informações, são necessários mecanismos que garantam a interoperabilidade de ontologias.

Para [6], interoperabilidade é a capacidade que um sistema possui de compartilhar e trocar informações em aplicações.

Conforme descrito no capítulo 3, atualmente existem algumas aproximações para o processo de compatibilidade de ontologias, entre elas: combinação de ontologias, alinhamento de ontologias, mapeamento de ontologias, integração de ontologias, entre outras.

No entanto, nestes mecanismos alguns fatores ainda não foram explorados, como a quantificação dos relacionamentos existentes, ou seja, todas as relações possuem a mesma intensidade não havendo diferença entre alguns atributos que podem ser considerados mais relevantes que outros. Em uma ontologia do domínio de pessoas, todas as pessoas possuem um perfil, com características e preferências próprias, sendo algumas mais destacadas que outras, como gostar de assistir televisão, por exemplo.

Além disso, a partir deste trabalho inúmeros trabalhos futuros poderão surgir, pois os sistemas utilizando agentes estão se tornando cada vez mais populares atraindo, inclusive, pesquisadores de outras áreas.

## 6 Considerações Finais

Neste trabalho demonstrou-se como os mecanismos de interoperabilidade aplicados às ontologias de agentes colaborativos, contribui para o aprimoramento de conhecimentos, assim como prometem melhor o

desempenho de agentes individuais, através da partilha de conhecimento decorrente de experiências de outros agentes.

Outro aspecto importante é a necessidade de atrelar uma aplicação de SMA à um modelo organizacional, de modo que se possa tirar proveito das informações que podem ser adquiridas. Portanto, compartilhamento de conhecimento, assim como nas sociedades humanas, é um aspecto cada vez mais relevante para geração de conhecimento.

Espera-se contribuir com este trabalho para um maior entendimento do assunto agentes colaborativos e ontologias, servindo, também como ponte para outras pesquisas, além de fornecer um conjunto de referências a serem exploradas com mais detalhes.

## 7 Referências Bibliográficas

- [1] **ANTONIOU, G.; VAN HARMELEN, F.** *A Semantic Web Primer* Massachusetts Institute of Technology, 2004.
- [2] **BAADER, F. , CALVANESE, D., D. L. MCGUINNESS, D.L., NARDI, D., PATEL-SCHNEIDER, P.F** apud : *The Description Logic Handbook: Theory, Implementation, Applications*. Cambridge University Press, Cambridge, UK, 2003.
- [3] **BERNERS-LEE, T.; HENDLER, J.; LASSILA, O.** *The Semantic Web., Scientific American*. May, 2001.
- [4] **BRENNER, W.; ZARNEKOV, R.; WITTING, H.** *Intelligent Software Agents: Foundations and Applications*. Berlim: Springer, 1998. 326p.
- [5] **BISHR, Y.** *Semantic Aspect of Interoperable GIS*. Ph.D. Thesis, Wageningen Agricultural University, The Netherlands, 1997.
- [6] **BORST, W. N.** *Construction of engineering ontologies*. 1997. Tese (Doutorado). Disponível em: <<http://www.ub.utwente.nl/webdocs/inf/1/t0000004.pdf>>. Acesso em: 21 maio 2006.
- [7] **CORAZZON, R.** *What is ontology?* [S. l. : s. n.], 2002. Disponível em: <[http://www.formalontology.it/section\\_4.htm](http://www.formalontology.it/section_4.htm)>. Acesso em: 20 jul. 2006.
- [8] **GUARINO, N. (1998)**. *Formal Ontology and Information Systems*. Disponível em: <http://www.loa-cnr.it/Papers/FOIS98.pdf>. Último acesso em 21 de setembro de 2006.
- [9] **JENNINGS, N.; WOOLDRIGDE.** *Applications of intelligent agents. In Agent Technology: Foundations, Applications and Markets*. Springer Verlag, 1998.
- [10] **NOY, F. N.; GUINNESS, D. L.** *Ontology development 101: a guide to create your first ontology*. Stanford University, USA, 2002. Disponível em: <<http://ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.doc>>. Acesso em: 12 setembro 2006.

- [11] **NOY, N.; MUSEN, M. SMART.** *Automated Support for Ontology Merging and Alignment*. Banff Workshop on Knowledge Acquisition, Modeling, and Management, Banff, Alberta, Canada, 1999.
- [12] **NOY, N.; MUSEN, M.** *The PROMPT Suite: Interactive Tools For Ontology Merging And Mapping*. International Journal of Human-Computer Studies, 2003.
- [13] **PINTO, S.; GOMEZ-PEREZ, A.; MARTINS, J.** *Some Issues on Ontology Integration*. Workshop on Ontologies and Problem Solving Methods: Lessons Learned and Future Trends, 1999.
- [14] **[PROTEGÉ, 2005]**. Disponível em: <http://protege.stanford.edu>. Acesso em: 28 março de 2007.
- [15] **RUSSEL, S. ; NORVIG, P.** *Artificial Intelligence: A Modern Approach*. New Jersey: Prentice-Hall, 1995. 932 p.
- [16] **SHOHAM, Y.** *Agent-oriented programming*. Artificial Intelligence, 60(1), 51-92.
- [17] **SOWA, J. F.** *Knowledge Representation: Logical, Philosophical, and Computational Foundation*. Brooks Cole Publishing Co., Pacific Grove, CA, 2000. Actual publication date, 16 August 1999.
- [18] **TECUCI, G.** *Building Intelligent Agents*. Academic Press :San Diego, 1998. 320p.