

Desenvolvimento de uma ontologia sobre componentes de ontologias

Gisele Vasconcelos Dziekaniak

Professora do Instituto de Ciências Humanas e da Informação da FURG. Doutoranda no Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento da UFSC.

Apresenta os principais passos no desenvolvimento da ontologia de domínio sobre Componentes de Ontologias. Demonstra como se chegou metodologicamente à ontologia, desenvolvida no Ontokem e exportada para o Protégé 3.3.1, onde foram desenvolvidas as inferências via Pellet 1.5.2 para fazê-la inferir sobre perguntas. As metodologias utilizadas foram Methontology, 101 e On-to-knowledge. Como resultado apresenta breve revisão de literatura sobre ontologias e um passo-a-passo do seu desenvolvimento para auxiliar quem deseja aventurar-se no caminho da construção de ontologias com o objetivo de classificação do conhecimento.

Palavras-chave: *Desenvolvimento de ontologias; Projeto de ontologia; Ontokem; Protégé; Organização do conhecimento; Ontologia.*

An ontology development about Ontologies Component

This abstract presents the main steps for the development of a domain ontology about Ontologies Components. Its goal is to demonstrate how we arrived methodologically to this ontology, developed in the OntoKEM editor and, subsequently exported to the Protégé 3.3.1, where inferences were developed and applied the plugging Pellet 1.5.2. to make the ontology "think" and infer about the questions asked. The methodologies used were the methodology, 101 and On-to-knowledge. As a result it presents brief literature review about ontologies and a step-by-step of its development to assist whom wishes to venture to the way of the construction of ontologies aiming the knowledge classification.

Keywords: *Ontologies development; Ontology project; OntoKEM; Protégé.*

1 Introdução

O desenvolvimento de ontologias prevê a elaboração de um projeto bem definido, o que significa dizer que é preciso conhecer os objetivos que se deseja atingir e o tipo de conhecimento que ela irá representar. Outra faceta importante na sua construção é o exercício de compreensão de que uma ontologia objetiva a representação do conhecimento, sendo assim, ela é um exercício político e, *a priori* limitado, como toda representação.

Logo, a busca pela consensualidade deve ser uma constante, uma vez que, de um modo geral, não se desenvolve uma ontologia para uso privado; o que irá requerer consenso acerca da visão de mundo adotada no projeto entre os desenvolvedores e os pretensos usuários desta ontologia. Segundo Gruber (1992) uma ontologia deve ser desenvolvida com o objetivo de compartilhar conhecimento entre pessoas e também entre agentes computacionais.

A ontologia apresentada se constitui em uma obra que não pretende ser a visão única da representação e classificação dos conhecimentos do domínio de *Componentes de Ontologias* e não está acabada, por se compreender que, sendo uma atividade realizada em sala de aula, com cronograma definido, exigiu maior esforço na absorção da interface do uso do sistema utilizado e dos passos a serem seguidos para a construção da ontologia, do que especificamente, a preocupação com o esgotamento da classificação de todos os conhecimentos do domínio eleito para representar.

Para tanto foi decidido trabalhar com um limite hierárquico de três níveis de classes, para facilitar a manipulação da ontologia e a implementação das relações entre os conceitos. Sendo o objetivo maior, o exercício mental na construção de hierarquias entre as classes, a elaboração de relações, restrições, propriedades e desenvolvimento da aplicação através de construções lógicas.

Partindo destas premissas foi que se deu o desenvolvimento do projeto desta ontologia piloto. O mesmo serviu para fornecer uma experiência prática no desenvolvimento de ontologias após um período de cerca de seis meses de estudo sobre a teoria da engenharia de ontologias.

2 Etapas metodológicas envolvendo a natureza e o domínio da ontologia

A seguir apresentam-se as etapas metodológicas para o desenvolvimento da ontologia proposta, a qual conta com contribuições advindas das seguintes metodologias: *101, Methontology* e *On-To-Knowledge*, devido ao *Ontokem*, editor utilizado no projeto e desenvolvido pelo Laboratório de Engenharia do Conhecimento da Universidade Federal

de Santa Catarina (LEC/UFSC), ter sido construído baseado na fusão destas três metodologias.

Esta ontologia piloto foi desenvolvida no âmbito da disciplina *Desenvolvimento de Ontologias para Engenharia do Conhecimento* no Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento da Universidade Federal de Santa Catarina.

2.1 Escolha do domínio e seu escopo

O desenvolvimento de uma ontologia demanda decisões políticas, epistemológicas e conceituais (CAMPOS *et al.*, 2007). No desenvolvimento desta ontologia não foi diferente. A primeira questão a demandar decisão, após a escolha do domínio a ser abordado pela ontologia, esteve relacionada com quais aspectos sobre *Ontologias* se iria eleger. Esta decisão esteve diretamente relacionada com os objetivos que se desejou atingir ao desenvolver a ontologia; quais necessidades iria atender, a quem interessaria e qual seria sua razão de ser.

Foi eleito o domínio dos *Componentes de Ontologias*, pela proximidade da autora com o domínio e por conta desta ontologia ser desenvolvida no âmbito de uma disciplina, o que por questões temporais tornaria impossível anexar no projeto, especialistas de um domínio diferente daquele de conhecimento da autora, embora recomendado.

Uma vez pensada estas questões optou-se pela classificação dos componentes presentes no âmbito das ontologias e de seu universo, entendendo por componentes os projetos de ontologias existentes, as linguagens de programação utilizadas para escrever ontologias, as metodologias existentes para se desenvolver ontologias, os editores de ontologias e os autores de ontologias.

Estes conceitos foram considerados como classes principais da ontologia e entendidos não como os únicos componentes, o que possibilita ampliação da ontologia, porém como aqueles indispensáveis para representar o domínio dos *Componentes de Ontologias*. A única classe a sofrer subdivisão foi a classe autores de ontologias, sendo subdividida nas seguintes subclasses: autores de projetos de ontologias, autores de metodologias de ontologias, autores de linguagem de ontologias e autores de editores de ontologias.

2.2 Aquisição de conceitos

Para captura dos principais conceitos pertinentes ao domínio eleito, foram realizadas pesquisas em fontes de informação como glossários *online*, *Wikipédia*, textos científicos da área e principalmente a obra da Dra. Karin Breitman *Web semântica* (2005). As fontes adotadas encontram-se referenciadas no final deste artigo.

2.3 Natureza da aplicação

Quanto à natureza da ontologia, esta é uma ontologia de domínio, porque explicita conceitos e relações provenientes de uma área específica do conhecimento (domínio), neste caso *Componentes de Ontologias*, buscando descrever e classificar o conhecimento relacionado a este domínio do conhecimento.

Segundo Guarino (2002) existem quatro tipos de ontologias, a saber: *ontologias de alto-nível* (também conhecidas como ontologias de topo) que descrevem conceitos genéricos como espaço, tempo, dentre outros conceitos abstratos que independem de um domínio do conhecimento específico ou de um problema do mundo real; *ontologias de domínio*, que representam um domínio do conhecimento, com especialização dos conceitos; *ontologias de tarefa*, as quais descrevem o contexto do desenvolvimento de uma tarefa genérica, especializando os conceitos e as *ontologias de aplicação*, as quais são específicas por serem usadas dentro de uma aplicação particular, descrevendo os conceitos relacionados a esta aplicação. Esse tipo de ontologia mistura os objetivos tanto das ontologias de tarefas como das ontologias de domínio.

3 Desenvolvimento da ontologia

Esta seção apresenta os passos utilizados no desenvolvimento e construção da ontologia, após a decisão pelo domínio a ser representado e pela tipologia de ontologia a ser desenvolvida, conforme já mencionado.

Uma das ferramentas utilizadas para o desenvolvimento deste projeto de ontologia foi o *Ontokem*. Este editor de ontologias é um sistema desenvolvido pelo Laboratório de Engenharia do Conhecimento da Universidade Federal de Santa Catarina (LEC/UFSC) Sua estrutura contempla 6 abas, que representam as etapas básicas para construção de uma ontologia.

A primeira aba *Projeto* traz os dados sobre descrição dos projetos e do controle de versões de um mesmo projeto. Essa possibilidade de controle de etapas é muito importante para que o desenvolvedor possa documentar e controlar os processos que ficaram pendentes dentro de um projeto ou permitir o controle das diferentes versões de um mesmo projeto até se chegar à ontologia final.

Na aba *Perguntas de Competência*, define-se na prática, o que se deseja que a ontologia responda. O *Ontokem* ainda contempla a aba para criação do *Vocabulário*, onde se definirão as classes e instâncias e os termos originados na fase da criação das perguntas de competência; ele oferece ainda a aba para construção da *Hierarquia*, espaço onde se desenvolve na ontologia, as noções de hierarquia dentro do projeto, (podendo ser realizada de duas formas: através da lógica de subordinação, quando se elegem as classes que serão subclasses da classe que se está

hierarquizando ou pode-se optar pela lógica da superordenação, quando se trabalha com a atribuição de um Pai para toda classe).

Há ainda a aba *Dicionário de Classes* espaço onde se irão associar as relações, propriedades, restrições e instâncias para cada classe. É esta aba que possibilitará os relacionamentos dentro da ontologia. Quanto às inferências, o *Ontokem* não possui um motor de inferência. Porém convém lembrar que editores de ontologias, como o *Protégé* ou o *OilEd* permitem inserir *pluggins* com raciocinadores, para que a ontologia adquira a habilidade de "raciocinar" em cima de inferências e axiomas. Devido a esta limitação do *Ontokem*, exportou-se a ontologia para o *Protégé 3.3.1* assim que chegou o momento de desenvolver inferências na ontologia.

Sobre a aba *Perguntas de Competência* pode se mencionar que, segundo Gruninger e Fox (1995 apud Breitman, 2005, p. 71): "[...] especificam os requisitos de uma ontologia e, como tais, são os mecanismos de definição do espaço de projeto da própria ontologia". Estas perguntas servem de base para consulta à ontologia, ou seja, respondem aquilo que o projeto de ontologia se propõe em responder a partir da base de conhecimento que será gerada. As perguntas de competência desta ontologia foram geradas no *Ontokem*, considerando-as como as perguntas que a ontologia se propunha a responder.

Além de espaço para preenchimento da pergunta de competência existe no *Ontokem* a presença de abas para preenchimento dos *Termos* relacionados com a pergunta e uma aba para inserir as possíveis *Relações* existentes nesta pergunta.

Neste projeto as perguntas de competência foram as seguintes: "Quais os editores de ontologia utilizam a linguagem ?Y?", "Quais projetos de ontologias são escritos na linguagem ?Y?", "Qual autor desenvolveu o projeto de ontologia ?Y?", "Qual autor desenvolveu a metodologia ?Y?", "Qual autor desenvolveu a linguagem de ontologia ?Y?", "Qual metodologia é utilizada no projeto de ontologia?Y".

Quanto aos termos relacionados (possíveis metadados que descrevem o teor da pergunta), exemplifica-os através do exemplo usado em uma das questões criadas para esta ontologia, como por exemplo, na pergunta *Quais projetos de ontologias são descritos na linguagem?Y? os termos relacionados a esta questão de competência são Linguagem_de_Ontologia e Projeto_de_Ontologia.*

A relação entendida como presente na pergunta referenciada acima vem a ser do tipo *ehEscritoEm*, ou seja, projetos de ontologias *são escritos em* uma determinada linguagem, importando saber qual vem a ser esta linguagem para o projeto desta ontologia. Entendeu-se como pertinente elaborar para toda relação existente, sua relação inversa, para que todas as perguntas de competência possam ser invertidas. Neste caso, a relação inversa designada para *ehEscritoEm* foi *Escreve*. Com isso se consegue saber ao mesmo tempo quais projetos são escritos em determinada

linguagem e também qual linguagem é usada para escrever determinado projeto de ontologia.

Dando seguimento, após estabelecer os termos e as relações, procedeu-se à declaração das *definições dos termos, as classes de termos, as instâncias, as relações e as regras*. A seguir identificam-se estes conceitos para melhor compreensão ao que se propõem.

As *definições* são utilizadas para descrever a semântica entendida pela ontologia e o termo que a representa. Ou seja, qual conceito o termo define e representa. A lógica da univocidade (relação de adoção de apenas um termo para representar um conceito) é extremamente necessária e deve ser seguida em toda ontologia, observando sempre o fator da consensualidade no domínio. A seguir apresenta-se na FIG. 1 uma tela do *Ontokem* onde procedeu-se a definição dos termos, demonstrando quais termos na estrutura hierárquica foram considerados classes no projeto.

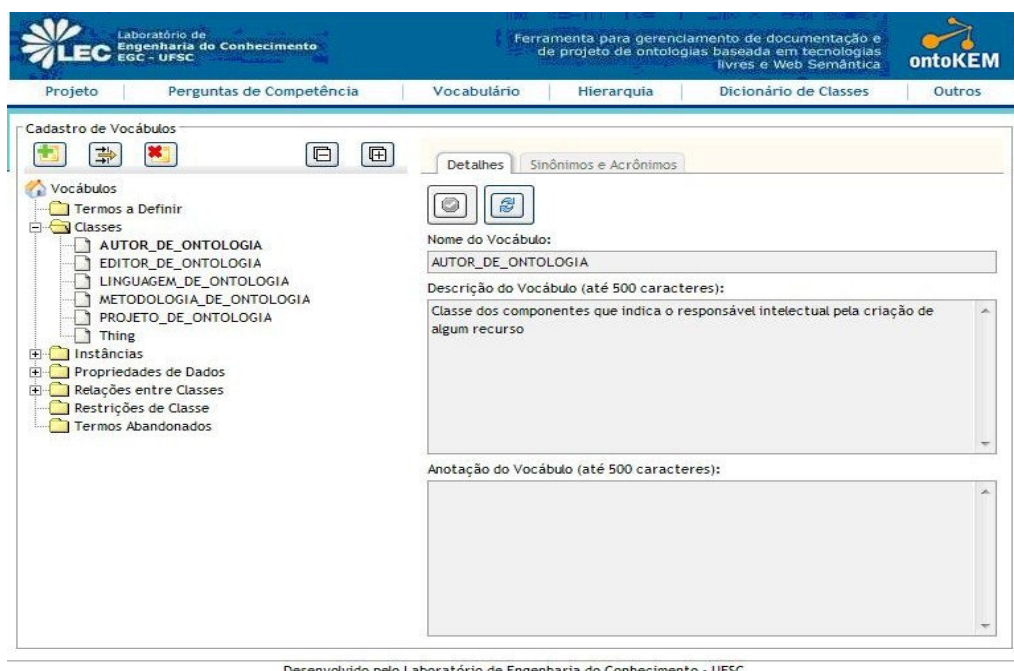


FIGURA 1 - Cadastro de vocábulo

Fonte: Dados da pesquisa.

Após exportação dos termos criados, abre-se a aba *A Definir*, a partir de onde se irá definir se o conceito é uma classe, uma subclasse ou uma instância de classe.

É fundamental compreender que uma das lógicas utilizadas na construção de ontologias, típica na linguagem OWL (*Web Ontology Language*), utilizada tanto pelo *Ontokem*, como pelos principais editores de ontologias como o *Protégé*, é o relacionamento transitivo, para o qual, por exemplo, se todo avião (subclasse) é um meio de transporte (classe), logo, um avião de hélice (subclasse de avião) tem que ser um meio de transporte. O conceito de herança, derivado da orientação a objetos é

seguido na construção de uma ontologia, onde objetos mais específicos na hierarquia das classes herdam todos os atributos dos objetos mais genéricos na hierarquia.

Conceitos como *classe* e *instância* em uma ontologia são indispensáveis para o desenvolvimento de um projeto. Para Breitman (2005) as classes servem para descrever conceitos básicos de dado domínio, ou seja: “uma classe representa um conjunto ou coleção de indivíduos (objetos, pessoas, coisas) que compartilham de um grupo de características que os distinguem dos demais.” (BREITMAN, 2005, p. 61). Elas representam os conceitos mais amplos em uma hierarquia.

Porém ao se trabalhar apenas com classes, deve-se entender que, formalmente, elas apenas dizem o que existe na ontologia. Sendo assim, o rótulo usado para definir livro poderia ser qualquer um que não fosse a palavra livro que, para a ontologia, seria a mesma coisa. O que realmente define uma classe são as características que se atribuem a esta classe.

Para exemplificar esta definição pode-se pensar de maneira concreta que a classe *avião* apresenta características que dizem o que um avião é. Por exemplo, *todo avião voa, todo avião tem uma rota e não há avião que não precise de um piloto*. Essas características que definem a classe avião são chamadas de *restrições*. Sendo, neste caso, axiomas validados pela visão de mundo da ontologia e devem ser respeitados. Estes axiomas acima querem dizer que, para um objeto ser considerado um avião no contexto da ontologia, ele obrigatoriamente precisa possuir todas as características assumidas pela ontologia, caso contrário não pode ser considerado um avião.

Por exemplo, nesta ontologia, ao vocábulo AUTOR_DE_ONTOLOGIA, além da descrição de seu conceito, foi atribuído que este termo representaria uma *classe* dentro da estrutura hierárquica da ontologia. Assim procedeu-se com todos os demais vocábulos criados no projeto.

Logo em seguida procedeu-se à atribuição das *instâncias* dentro do *Ontokem*, as quais foram pensadas desde o começo do desenvolvimento da ontologia, e propostas em um mapa conceitual desenvolvido no aplicativo *SmartDraw* 6.0, anteriormente ao desenvolvimento da ontologia.

A atribuição das instâncias é feita no *Ontokem* em *Cadastro de Vocábulos*, no momento em que após a criação dos termos e atribuição de conceitos a eles, se estabeleceu quais conceitos dentro da hierarquia, seriam *classes*, *subclasses* e *instâncias*.

Por *instâncias* se compreende o que o *Protégé* entende por *Individuals*. Ou seja, uma instância nada mais é do que a concretização de uma classe. Uma instância representa um indivíduo único em uma hierarquia. Por exemplo, a classe *Ser humano* possui como subclasses, as classes *Masculino* e *Feminino*. A subclasse *Feminino* possui a instância *Maria_de_Souza* (que é o metadado que define este objeto e deve ser único neste projeto de ontologia). Porque só deve existir uma Maria de Souza na ontologia, tornando-a única; uma vez que ela possui propriedades e características que a distinguem de todos os demais

objetos. E todos os atributos da *Maria de Souza* precisam existir naquilo que se considera *Feminino*.

Como aplicação deste tipo de raciocínio deve-se pensar como Breitman (2005, p. 77) demonstra: "Se uma classe A é superclasse de uma classe B, então toda instância de B também é instância de A", ou seja, se está se lidando com a classificação de *componentes de ontologias*, como é o caso desta ontologia, então as subclasses e as instâncias geradas na hierarquia devem ser obrigatoriamente *componentes de ontologias* também.

Em uma ontologia, além de ser necessário criar classes e instâncias das classes é necessário gerar a *hierarquia de classes*. Este projeto seguiu a estratégia *bottom up* (de baixo para cima), na qual se definiu primeiramente o conjunto de termos específicos para depois identificar os termos mais genéricos (aos quais os termos específicos estão subordinados). Sendo assim a cada classe se atribuiu um *pai* dentro da hierarquia (conceito derivado da orientação a objetos).

Após esta etapa procedeu-se à definição das *instâncias* demonstrada na FIG.2. Cada instância constitui-se na materialização da classe em indivíduos únicos, que representam, teoricamente, a última subdivisão de uma classe. Uma classe abriga diversas instâncias, assim como uma instância pode derivar de mais de uma classe – caso que não ocorreu neste projeto, mas que é passível de ocorrer.

Na FIG. 2 tem-se a classe *metodologia de ontologia* em destaque, e a ela se atribuiu algumas instâncias que representam indivíduos considerados metodologias para desenvolvimento de ontologias, como é o caso da *OnToKnowledge*, *101*, *Methontology*, dentre outras citadas no projeto.

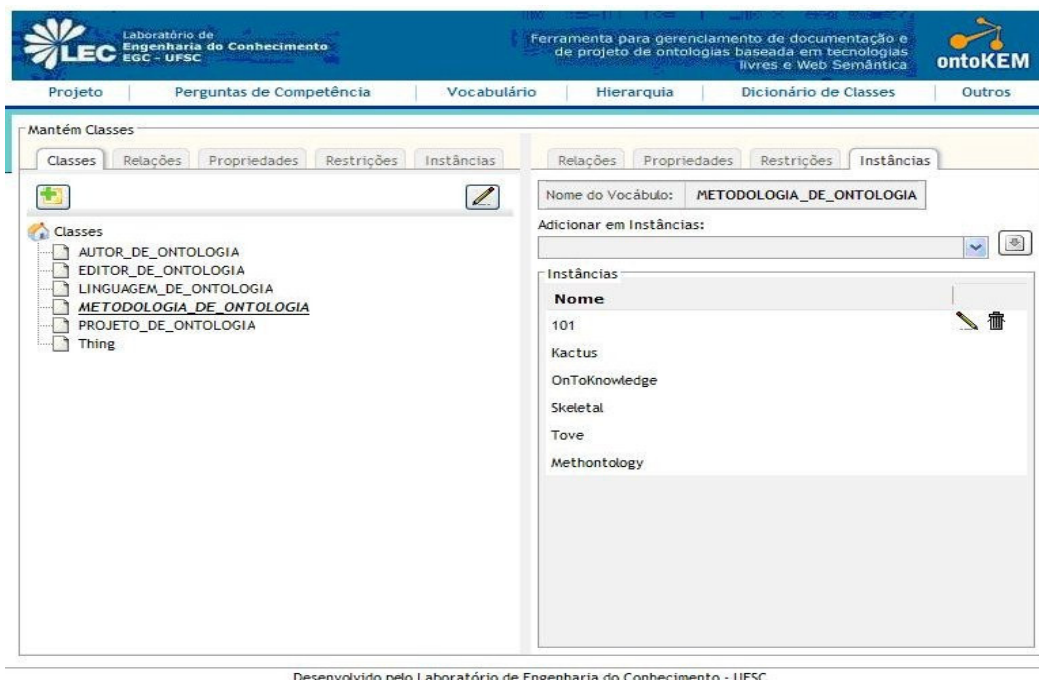


FIGURA 2 - Atribuição das instâncias às classes

Fonte: Dados da pesquisa.

Quanto às *Relações*, na engenharia de ontologias elas são consideradas uma espécie de classificação, significando o envolvimento que dado conceito possui com outro(s). Adotaram-se apenas seis tipos de relações para este projeto de ontologia, sendo elas: *escreve*, *ehEscritoEm*, *ehCriadorDe*, *ehCriadoPor*, *Usa*, *ehUsadoPor*. Atenta-se para o fato de que se trabalhou com relações inversas no *Protégé*, para possibilitar que perguntas inversas possam ser feitas à base de conhecimento da ontologia, o que se considera útil e prático, conforme já mencionado.

A seguir exportou-se o projeto da ontologia sobre *Componentes de Ontologias* para o editor *Protégé 3.3.1*, onde foram trabalhados os domínios e os ranges e também criadas as restrições no projeto. Apresenta-se a tela inicial do *Protégé* em sua versão 3.3.1.

Quanto às *Restrições* estas servem para limitar e auxiliar no processo de inferência em uma pergunta à base ontológica. Neste projeto de ontologia utilizou-se apenas a restrição *some* para “forçar” que todo projeto de ontologia citado nesta ontologia, toda metodologia de ontologia, toda linguagem de ontologia, todo editor de ontologia tenha pelo menos um autor. Uma vez que para o projeto desenvolvido, a questão autoral foi considerada essencial.

Outra etapa importante no desenvolvimento de uma ontologia diz respeito à atribuição dos *domínios* e *ranges*, conceitos herdados da linguagem RDF (*Resource Description Framework*). Esta linguagem é adotada nos projetos de ontologias devido à possibilidade de representação entre um relacionamento e os objetos que se relacionam através da tripla sujeito-propriedade-objeto, as quais contém afirmações sobre dada informação. Esta representação em RDF é reconhecida pela linguagem XML, o que possibilita que a ontologia seja implementada em ambiente web.

Em uma ontologia é preciso definir quem na hierarquia, são *domínios* e quem são *ranges* (sendo que essa definição é relativa, podendo uma classe, ora ser domínio e ora ser range em arranjos diferentes). De modo simples, se pode entender como sendo um domínio, o que pode ser entendido como o sujeito da pergunta (aquele ou aquilo que executa a ação) e o range como sendo o objeto (aquele que é executado).

A FIG. 3 demonstra, em sua aba *Property Editor*, quem é o domínio (*Domain*) e o range para a relação *ehCriadoPor* que tem por propriedade inversa *ehCriadorDe*. Neste caso foram atribuídos vários domínios para o mesmo range porque tanto o domínio LINGUAGEM_DE_ONTOLOGIA, como METODOLOGIA_DE_ONTOLOGIA, PROJETO_DE_ONTOLOGIA e EDITOR_DE_ONTOLOGIA possuem o mesmo range AUTOR_DE_ONTOLOGIA. Uma vez que todos os domínios indicados têm um autor de ontologia, algum responsável intelectual que os desenvolveu.

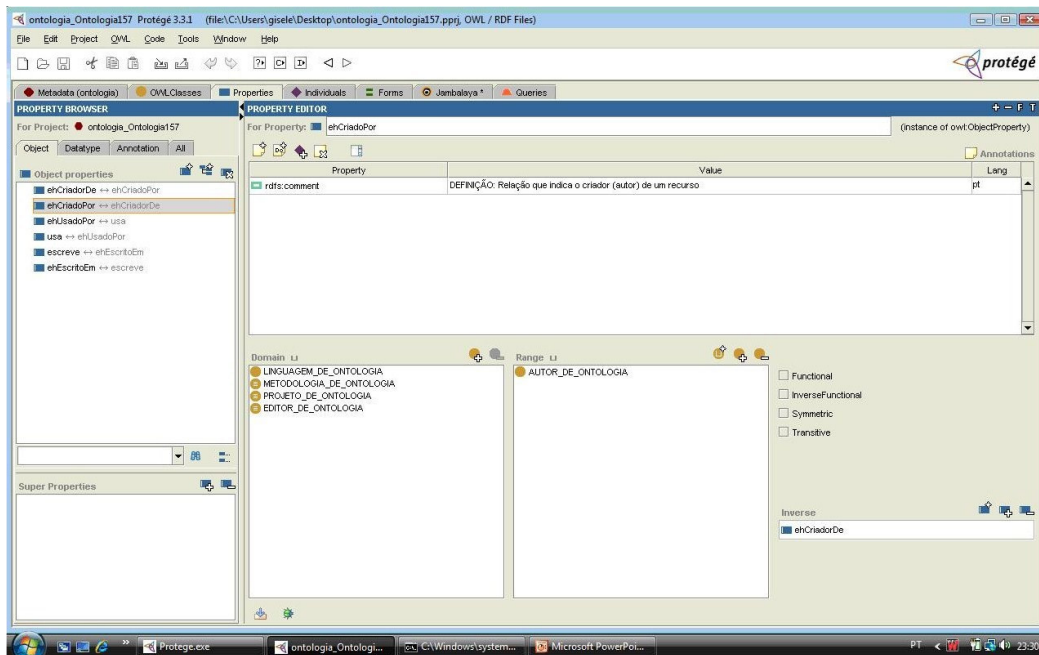


FIGURA 3 - Definição de domínios e ranges

Fonte: Dados da pesquisa.

O *Protégé* possui o aplicativo *Jambalaya*, este é um visualizador da hierarquia desenvolvida na ontologia. Ele é útil por possibilitar a leitura visual do projeto, porém trabalha apenas com as classes. As instâncias de classes não aparecem no visualizador, até mesmo porque a hierarquia é realizada no âmbito das classes. Quanto às instâncias, a hierarquia delas é limitada à classe da qual se está instanciando.

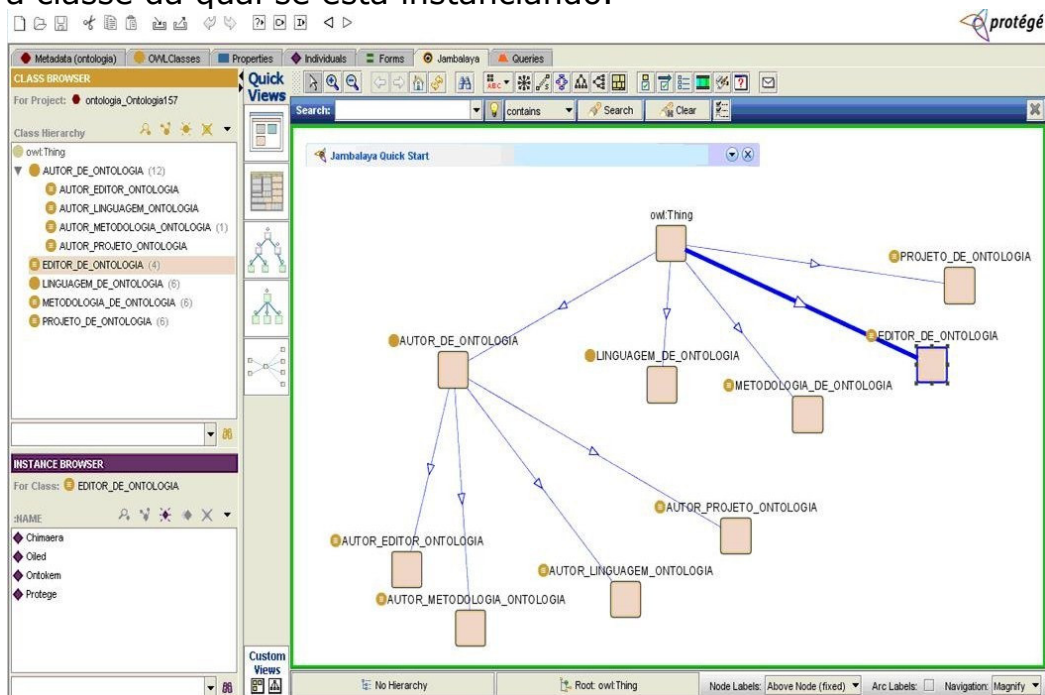


FIGURA 4: Hierarquia da ontologia no Jambalaya Views do Protégé

Fonte: Dados da pesquisa.

Logo em seguida procedeu-se ao estabelecimento de algumas restrições à ontologia, em busca de que a mesma tenha um desempenho satisfatório na realização de inferências e raciocínio.

Um exemplo de restrição criado no *Protégé* para este projeto de ontologia dizia que: "é necessário e suficiente que *ehCriadoPor some EDITOR_DE_ONTOLOGIA*", e também criou-se outra restrição que diz que "é necessário *AUTOR_DE_ONTOLOGIA*". Estas sentenças querem dizer à ontologia que, todo editor de ontologia tem que ter algum responsável intelectual por sua criação e, que para ser considerado um criador (que é uma instância) é preciso necessariamente ser um autor de ontologia (uma das classes principais na hierarquia desta ontologia).

A seguir apresenta-se a utilização do raciocinador *Pellet 5.2.1*, um *plugging* que dialoga com o *Protégé* para realizar inferências na base de conhecimento desenvolvida no projeto.

Procedeu-se a etapa tão esperada, após alimentar a ontologia com as perguntas de competência que a mesma deveria responder, após estabelecer os termos, as relações entre os termos, preencher com as instâncias, desenvolver restrições, foi realizado o primeiro teste com este projeto de ontologia que foi oferecer perguntas à ontologia (dentro das perguntas de competência criadas) e comprovar que a mesma respondeu corretamente à pergunta.

A função do raciocinador, também conhecido por motor de inferência, é a de realizar inferências dentro da base de conhecimentos da ontologia. Isso significa que se pode apenas dizer a ela que, para ser autor de metodologia de ontologia, este criador precisa ser um autor de ontologia e, que determinado autor é o criador de uma metodologia de ontologia. Com isso o raciocinador cruzará os dados e retornará quem são os autores de ontologia que são autores apenas de metodologia de ontologias.

Perguntou-se a ontologia, através de caixas Box oferecidas em formato *default* pelo *Protégé 3.3.1*, quem são os autores de ontologias que são autores da categoria, metodologias de ontologias.

É interessante perceber como o raciocinador agiu. Sua resposta foi mais que simplesmente ir até o repositório de dados criados nesta ontologia para retornar uma resposta simples como uma consulta a um banco de dados. Ele precisou cruzar dados de naturezas distintas. Sua trajetória possivelmente foi ir até a classe *metodologia de ontologias*, perceber que não havia nenhuma instância "pendurada" nela, retornar à classe autor de ontologia e tentar validar a existência de algum autor de ontologia (instância) que seja autor de alguma metodologia de ontologia. Foi o que ele fez, retornando corretamente as instâncias *McGuinees*, *Noy*, *Fox*, *Gruninger* e *Gomes-Perez* como autores de metodologias de ontologias, respondendo a uma das questões de competência presentes neste projeto.

Para identificar a robustez e grau de confiabilidade da ontologia, é preciso que o raciocinador responda a todas as perguntas de competência presentes no projeto. Assim um dos objetivos da ontologia se cumpre.

Todavia há outros objetivos subjetivos que se devem compreender quando se elabora uma ontologia. De nada adianta que a mesma raciocine e responda corretamente às perguntas a ela realizadas. É preciso que a mesma seja acolhida por alguma comunidade ou projeto existente para que seja utilizada, gerenciada e continuamente repensada.

Uma vez o conhecimento não sendo estático, toda ontologia precisa ter em seu projeto, espaço para evolução, adaptação e continuidade e, os atores que a utilizarão e manterão, precisam perceber que, sendo a ontologia, uma tentativa de representar dada visão de mundo e de algum domínio de conhecimento, a mesma é limitada, como todo modelo de representação e por que não dizer como toda linguagem documentária (apesar de ela não ser somente isto).

Não obstante pode-se considerar por hora, como sendo uma ontologia talvez o melhor instrumento para classificação e representação do conhecimento até então pensado e desenvolvido, fornecendo ampla possibilidade de uso inclusive por atores que não dominam linguagem de máquina, ou seja, códigos de programação.

Referências

BREITMAN, K. *Web semântica: a internet do futuro*. Rio de Janeiro: LTC, 2005.

CAMPOS, M. L. A. *et al.* Ontologias: representando a pesquisa na área através de mapa conceitual. In: Encontro Nacional de Pesquisa em Ciência da Informação, 8. , Salvador, 2007. *Anais...* Salvador: UFBA, 2007. [CDRom].

CLARK, K.; PARSIA, B. *Pellet: the open source OWL DL reasoner*. Disponível em: <<http://clarkparsia.com/pellet/>>. Acesso em: 20 out. 2008.

GRUBER, T. *What is an Ontology?* 1996. Disponível em: <<http://www-ksl.stanford.edu/kst/what-is-an-ontology.html>>. Acesso em: 14 set. 2008.

GRUNINGER, M.; FOX, M.S. *Methodology for the design and evaluation of Ontologies*. 1995. Disponível em: <http://citeseer.ist.psu.edu/gruninger95methodology.html> apud BREITMAN, K. *Web semântica: a internet do futuro*. Rio de Janeiro: LTC, 2005.

GUARINO, N.; GIARETTA, P. *Ontologies and KBs, towards a terminological clarification*. In: MARS, N. (Ed.) *Towards very large knowledge bases*. 1995. Disponível em: <<http://www.ladseb.pd.cnr.it/infor/Ontology/Papers/KBKS95.pdf>>. Acesso em: 13 jul. 2008.