

Uso de Ontologias na Garantia da Qualidade e Melhoria de Processos

Érica Ferreira de Souza¹, Ricardo A. Falbo², Nandamudi L. Vijaykumar³

¹Programa de Doutorado em Computação Aplicada – CAP
Instituto Nacional de Pesquisas Espaciais – INPE

²Departamento de Informática – Universidade Federal do Espírito Santo – UFES
Vitória – Espírito Santo – ES – Brazil

³Laboratório Associado de Computação e Matemática Aplicada – LAC
Instituto Nacional de Pesquisas Espaciais – INPE

{erica.souza, vijay}@lac.inpe.br, falbo@inf.ufes.br

Abstract. *Quality Assurance and Process Improvement are essential in the evaluation of the software process. Software processes are strategic elements to conduct projects and product quality. Therefore, software development organizations have shown a growing interest in mechanisms of improving of these processes, for example, the use of ontologies. This paper presents an overview of the study of the art of ontologies and its use in quality assurance and process improvement software.*

Resumo. *Garantia da Qualidade e Melhoria de Processos são essenciais na avaliação de processo de software. Os processos de software são elementos estratégicos para a condução de projetos e qualidade do produto. Diante disso, organizações de desenvolvimento de software têm mostrado um crescente interesse por mecanismos de melhoria desses processos, por exemplo, o uso de ontologias. Este trabalho apresenta uma visão geral do estudo da arte das ontologias e seu uso na garantia da qualidade e na melhoria de processos de software.*

Palavras-chave: *Processos de software, melhoria de processos, reuso de conhecimento e ontologia.*

1. Introdução

Durante as últimas décadas, as organizações de desenvolvimento de software tem priorizado a utilização de processos de software mais maduros, capazes de atender às demandas de qualidade e produtividade em um mercado tão exigente. Dessa forma, ocorre um aumento do interesse pelas organizações na melhoria de seus processos. A garantia de qualidade do processo de software influencia significativamente na qualidade do software gerado [Fuggetta 2000].

Para as organizações (empresas ou instituições), os processos de desenvolvimento devem ajudar a gerar produtos que atendam aos requisitos estabelecidos. É neste contexto que a melhoria de processo de software se tornou um desafio para as organizações, pois bons processos ajudam a produzir softwares melhores, mais rapidamente e com custo menor [Koomen and Pol 1999]. Visando melhorar os processos de desenvolvimento de software

aplicados na organização, diversos modelos e métodos de melhoria do processo são desenvolvidos, como: *Capability Maturity Model* (CMM) [Paulk 1995], *Capability Maturity Model Integration* (CMMI) [SEI 2006], Melhoria de Processo do Software Brasileiro (MPS.BR) [Softex 2009], dentre outros.

Para aplicação de melhoria de processos muitas organizações tem como base a coleta de dados da execuções dos projetos e comparação destes com os valores planejados [Barcellos 2009]. Apesar dessa ser uma abordagem adequada nos níveis iniciais, ela não é suficiente para as organizações que buscam maturidade em seus processos.

A comunidade acadêmica tem trabalhado em dados¹ coletados com o intuito de encontrar informações importantes para a realização e ações corretivas. O papel da reutilização oferece uma importante oportunidade para alcançar melhorias no processo de software [Martins 2009]. Para desenvolver o conhecimento para a reutilização, necessita-se de modelos que capturem o conhecimento do domínio e ontologias podem ser utilizadas para esse fim.

Para lidar com os temas acima abordados, este trabalho apresenta uma revisão da literatura e estado da arte focando nos seguintes assuntos: processo de software, modelos para melhoria de processos, reutilização de conhecimento e ontologias.

2. Processo de Software e Reutilização do Conhecimento

Um processo de software pode ser definido como um conjunto coerente de políticas, estruturas organizacionais, tecnologias, procedimentos e artefatos necessários para conceber, desenvolver e implantar um produto de software [Fuggetta 2000]. Um processo de software bem definido e com qualidade está diretamente relacionado com a garantia da qualidade do produto final e esse é um dos principais motivos para que organizações de software adotem uma visão de melhoria contínua de seus processos.

A maioria das abordagens e experiências encontradas na literatura, para a melhoria de processo de software são baseadas em modelos de maturidade e são utilizados com sucesso nas instituições acadêmicas e na própria indústria. Os modelos de maturidade fornecem aos gestores um poderoso instrumento para avaliarem os processos utilizados na organização.

Os modelos descrevem a maturidade da empresa baseado nos projetos que ela está desenvolvendo. A maturidade implica num potencial de crescimento da capacidade e indica tanto a riqueza do processo na organização, quanto a consistência na qual o processo é aplicado nos projetos. Entre os modelos que mais se destacam tem-se o CMM e o CMMI, e no Brasil foi criado o modelo MPS.BR.

Atualmente, as organizações buscam a alta maturidade dos seus processos investindo na automatização das tarefas de aquisição, processamento, análise e principalmente no reuso de informações. O reuso de software pode ocorrer em diversos níveis, desde o código até o conhecimento. Em um ambiente de engenharia de software uma quantidade enorme de informação é produzida ao longo do processo. Estas informações precisam ser compartilhadas para auxiliarem em processos de aprendizagem, para tornar mais ágil a resolução

¹ Alguns dados que podem ser coletados são: tempo gasto para um processo em particular ser concluído; recursos requeridos, como, por exemplo, esforço total por pessoa e gastos com equipamentos; e número de ocorrências decorrente de um evento, por exemplo, mudanças nos requisitos.

de problemas e para desenvolver novos produtos.

Para o desenvolvimento com reuso, é necessário capturar o conhecimento do domínio e ontologias podem ser utilizadas para esse propósito. Uma ontologia define um vocabulário para descrever uma certa realidade [Guarino 1998] e pode ser utilizada para organizar uma porção do conhecimento capturado, com o objetivo de facilitar o acesso, o entendimento e permitir o reuso.

2.1. Ontologias

Ontologia é uma especificação de uma conceituação compartilhada [Gruber 1993]. A ontologia envolve a descrição de conceitos em um determinado domínio de conhecimento, com suas propriedades e restrições. Ontologias fornecem uma descrição exata do conhecimento em uma linguagem para facilitar a comunicação, integração, busca, armazenamento, compartilhamento, reutilização e representação do conhecimento [O'Leary 1998].

O grande interesse em ontologias, hoje, dá-se pela necessidade de haver, cada vez mais, uma maior interoperabilidade e reutilização de informações entre os sistemas e as pessoas dentro de uma organização [Rios 2005].

Classificação

São várias as classificações de ontologia encontradas na literatura. Guarino (1998) define a seguinte classificação:

- **Ontologias Genéricas:** Também chamada ontologia de fundamentação ou ontologia de nível superior, é uma conceituação abstrata sobre elementos genéricos para fazer parte de um domínio específico, como espaço, tempo, matéria, objeto, evento e ação.
- **Ontologias de Domínio:** Descrevem o vocabulário relacionado a conceituações de domínios particulares, como medicina ou direito.
- **Ontologias de Tarefa:** Similar a uma ontologia de domínio, porém, ao invés de mapear os conceitos de um domínio particular, mapeia conceitos de uma tarefa ou atividade específica, como diagnóstico, venda e matrícula.
- **Ontologias de Aplicação:** Descrevem conceitos que são dependentes de um domínio e de uma tarefa particular e, assim, combina especializações de conceitos presentes nas ontologias de domínio e de tarefa.

A Figura 1 apresenta os quatro tipos de ontologias e seus relacionamentos. No topo estão as ontologias genéricas que definem conceitos gerais que embasam todas as outras conceituações e no nível mais baixo encontram-se as ontologias de aplicação, que especializam conceituações para uma determinada classe de aplicações.

Ferramentas

Existem diversas ferramentas de apoio à construção de ontologias, por exemplo, a Protégé 2000 [Noy and Musen 2000], a *OntoEdit* [Staab and Maedche 2000] e a *Text-to-onto* [Maedche and Volz 2001]. Em geral, as ferramentas utilizam linguagem de representação para a construção das ontologias.

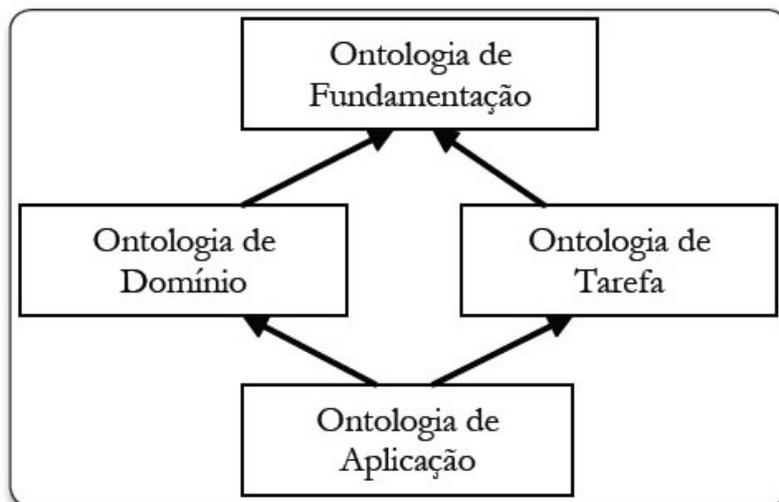


Figura 1. Relacionamentos entre os tipos de ontologias.
Adaptado de Guarino (1998)

Linguagens de Representação

Para representar uma conceitualização compartilhada, é necessária uma linguagem de representação. Algumas das linguagens existentes estão detalhadas na Tabela 1.

Tabela 1. Linguagens de representação de Ontologias

Linguagem	Descrição
RDF/RDF Schema	A <i>Resource Description Framework</i> (RDF)/ <i>RDF Schema</i> foi criada com o objetivo de representar o conhecimento utilizando redes semânticas. São linguagens que permitem a representação de conceitos, taxonomias de conceitos e relações binárias.
OWL	A <i>Web Ontology Language</i> (OWL) é uma linguagem de marcação semântica utilizada para representar explicitamente o conjunto de termos de um vocabulário e os relacionamentos que existem entre eles. É especialmente indicada quando há necessidade de descrever ontologias e para ser usada em aplicações Web que necessitam processar o conteúdo de informações disponíveis.

Linguagens Gráficas	O uso de uma linguagem gráfica age como um elemento facilitador da comunicação entre os agentes envolvidas no processo. Existem várias linguagens gráficas passíveis de utilização para este propósito, por exemplo, diagramas de classes e modelos de entidades e relacionamentos. Um outro exemplo é a linguagem Lingo, proposta por [Falbo 1998], a fim de garantir independência de semântica numa linguagem gráfica.
---------------------	---

Métodos de construção de ontologias

Os métodos de construção são desenvolvidos no intuito de sistematizar a construção e a manipulação de ontologias. Diversos métodos para construção de uma ontologia foram propostos, por exemplo, *Sensus* [Swartout et al. 1996] e *On-to-knowledge* [Fensel et al. 2000]. Com a finalidade de reunir em uma única proposta, as melhores características dos métodos existentes, [Falbo 2004] apresentou um método para a construção de ontologias chamado *Systematic Approach for Building Ontologies* (SABiO). As principais fases do método são:

- **Propósito e Especificação de Requisitos:** identificar a competência da ontologia e analisar aquilo que é relevante. Inclui a definição de questões de competência que indicam as questões que a ontologia deve ser capaz de responder;
- **Captura da Ontologia:** consiste em capturar a conceituação do domínio, com base na competência da ontologia. Conceitos, relações, propriedades e axiomas relevantes devem ser identificados e organizados;
- **Formalização da Ontologia:** representar a conceituação capturada pela ontologia em uma linguagem conceitual;
- **Integração de Ontologias Existentes:** reutilizar conceituações de outras já existentes, de modo a reutilizar conceituações previamente estabelecidas, quando necessário;
- **Avaliação da Ontologia:** uma ontologia deve ser avaliada, com o propósito de verificar se os requisitos estabelecidos na especificação são satisfeitos; e
- **Documentação da Ontologia:** o desenvolvimento da ontologia deve ser documentado com propósitos, requisitos, descrições textuais da conceituação, ontologia conceitual e critérios de projeto.

O método SABiO inclui, ainda, o uso de uma linguagem de modelagem para facilitar a comunicação dos modelos da ontologia, sugerindo um perfil UML como linguagem gráfica para representação das ontologias.

2.2. Ontologias na Melhoria de Processos

Pesquisas voltadas para a melhoria do processo de software podem ser consideradas hoje uma das maiores prioridades para as organizações que trabalham com software. Diversos trabalhos surgiram ao longo dos últimos anos utilizando a criação de ontologias com o propósito de entender e avaliar os processos na engenharia de software, e somente então, tomar ações que apoiem a melhoria desses processos. Alguns destes trabalhos foram selecionados para ilustrar as diferentes maneiras de lidar com tais temas e estão apresentados a seguir.

Em [Barcellos 2009], é apresentada uma estratégia para auxiliar as organizações de software na obtenção e manutenção de bases de medidas adequadas ao controle estatístico de processos de software, o que conduz à melhoria contínua dos processos exigida na alta maturidade. A estratégia é composta por uma Ontologia de Medição de Software, um Instrumento para Avaliação de Bases de Medidas para o Controle Estatístico de Processos e um Conjunto de Recomendações para Medição de Software.

A reutilização do conhecimento capturado por ontologias de domínio e de tarefa no processo de Engenharia de Requisito (ER) é apresentada no trabalho de [Martins 2009]. O processo ER envolve as atividades relativas ao tratamento dos requisitos de um sistema, incluindo: levantamento, modelagem, negociação, documentação, validação e gerência de requisitos. A modelagem aparece como núcleo do processo ER, sendo uma atividade cara, pois muitos esforços são despendidos com a elaboração desses modelos. Uma forma de minimizar esses esforços é inserir abordagens de reuso no processo da ER. Para capturar o conhecimento de domínio e de tarefa, é necessário capturar e formalizar esses conhecimentos, o que por ser feito por meio de ontologias.

O trabalho de Martins (2009), propõe um conjunto diretrizes para apoiar a reutilização de conhecimento na ER. É apresentada uma abordagem para o reuso de conhecimento no processo ER, onde a partir da integração de ontologias de domínio e de tarefa, pode-se obter um modelo estrutural de uma ontologia de classes de aplicações e a partir das diretrizes desenvolvidas é possível derivar diagramas de classes, de atividades, de estados e de casos de uso, incluindo a elaboração de descrições de casos de uso.

Em [Bringunte 2011], é apresentada uma abordagem de integração semântica baseada em ontologia, para integrar na camada de dados ferramentas que apoiam o planejamento, controle e acompanhamento de projeto de software. Foi utilizada uma ontologia de processo de software, a SPO (*Software Process Ontology*) para adicionar semântica aos conceitos das ferramentas envolvidas nesse processo.

Recentemente os autores desse artigo também desenvolveram um trabalho o qual propõe uma estrutura de um modelo dimensional de Data Warehouse (DW) para armazenamento do conteúdo de uma ontologia [Souza et al. 2012]. Este trabalho apresenta a elaboração de uma das etapas da tese de doutorado do autor desse artigo, que é a criação de uma base de conhecimento a partir de uma ontologia de teste de software. Uma vez construída a base de conhecimento, podem ser realizados diversos experimentos, promovendo um entendimento fácil aos especialistas do domínio sobre o processo, podendo tirar melhor proveito das interpretações conceituais da ontologia criada.

A manipulação de ontologias em bancos de dados pode representar ganhos na recuperação de informações estratégicas e melhoria de processos. A ontologia utilizada também está no contexto da tese de doutoramento do autor deste artigo e visa a gestão de conhecimento no ambiente de teste de software. Para converter o conteúdo da ontologia escrita em RDF/OWL em uma arquitetura DW, segue-se o processo apresentado na Figura 2.

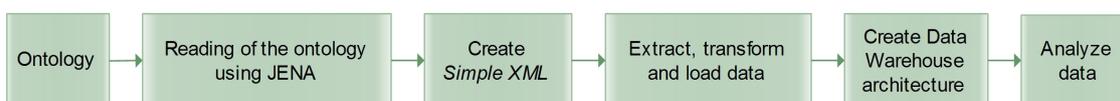


Figura 2. Processo proposto para converter uma ontologia em um modelo dimensional

Conforme apresentado na Figura 1, para permitir a leitura da ontologia foi necessário transformar o RDF/OWL em um *XML simples* e assim poder exportar os dados para o modelo DW criado utilizando a ferramenta Pentaho. Para isso, foi utilizado o framework Jena [Jena 2012] que converte os dados em um XML puro, ou seja, usando apenas as tags nativas do XML sem as tags específicas do RDF/OWL. A base de conhecimento criada em DW possibilitará o armazenamento e o processamento dos dados para obtenção de informações estratégicas que podem facilitar a tomada de decisão.

3. Conclusões

Atualmente, diversas organizações de desenvolvimento de software buscam utilizar a reutilização do conhecimento para desenvolverem competências específicas e capacidade inovadora, que se traduzem em serviços com menor custo e tempo, e produtos com maior índice de qualidade. O presente trabalho apresentou uma visão geral de como as ontologias são usadas na garantia da qualidade e na melhoria de processos de software.

De uma forma geral, na engenharia de software, uma ontologia pode ser vista como um modelo que representa conceitos e relações, bem como suas restrições, e pode ser definida como um artefato de engenharia. Ontologias são apontadas, hoje, como sendo cruciais na melhoria dos processos. Atualmente, a utilização do termo ontologia é considerado forte na área de engenharia de software.

Referências

- Barcellos, M. P. (2009). *Uma estratégia para medição de software e avaliação de bases de medidas para controle estatístico de processos de software em organizações de alta maturidade*. PhD thesis, Tese de Doutorado do Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- Bringunte, A. C. O. (2011). *Reengenharia de uma ontologia de processo de software e seu uso para a integração de ferramentas de apoio ao planejamento de projetos*. Master's thesis, Universidade Federal do Espírito Santo (UFES), Programa de Pós-Graduação em Informática, Vitória, UFES.
- Falbo, R. A. (1998). *Integração de conhecimento em um ambiente de desenvolvimento de software*. PhD thesis, Tese de Doutorado do programa de pós-graduação de engenharia da Universidade Federal do Rio de Janeiro (UFRJ), no curso de Ciências em Engenharia de Sistemas e Computação., Rio de Janeiro, RJ - Brasil.
- Falbo, R. A. (2004). Experiences in using a method for building domain ontologies. *In: International Workshop on Ontology in Action. Proceedings of the 16th International Conference on Software Engineering and Knowledge Engineering*, pages 474–477.
- Fensel, D., Harmelen, F. v., Klein, M., and Akkermans, H. (2000). On-to-knowledge: Ontology-based tools for knowledge management. *In Proceedings of the eBusiness and eWork 2000 (EMMSEC 2000) Conference*.
- Fuggetta, A. (2000). Software process: A roadmap. *In Proc. of The Future of Software Engineering, ICSE'2000*, pages 25–34.
- Gruber, T. R. (1993). Toward principles for the design of ontologies used for knowledge sharing. *In Formal Ontology in Conceptual Analysis and Knowledge Representation*.

- Guarino, N. (1998). Formal ontology and information systems. *In Proceedings of International Conference in Formal Ontology and Information Systems - FOIS'98*, pages 3 – 15.
- Jena (2012). Apache Software Foundation - Jena. <http://jena.apache.org/documentation>. Acesso em agosto de 2012.
- Koomen, T. and Pol, M. (1999). Test process improvement: A practical step-by-step guide to structured testing. *ACM Press, London, England*.
- Maedche, A. and Volz, R. (2001). The text-to-onto ontology extraction and maintenance environment to appear. *In: Proceedings of the ICDM Workshop on integratin data mining and knowledge management*.
- Martins, A. F. (2009). Construção de ontologias de tarefa e sua reutilização na engenharia de requisitos. Master's thesis, Universidade Federal do Espírito Santo (UFES), Programa de Pós-Graduação em Informática, Vitória, UFES.
- Noy, N. F. and Musen, M. A. (2000). Prompt: algorithm and tool for automated ontology merging and alignment. *In: National Conference on Artificial Intelligence*, pages 450–455.
- O'Leary, D. (1998). Using ai in knowledge management: knowledge bases and ontologies. *IEEE Intelligent Systems*, 13, n. 3:34–39.
- Paulk, M. (1995). Evolution of the sefs capability maturity model for software. *Software Process: Improvement and Practice, Pilot Issue, Spring*, 1:3–15.
- Rios, J. A. (2005). Ontologias: alternativa para a representação do conhecimento explícito organizacional. *In Proceedings CINFORM - Encontro Nacional de Ciência da Informação VI*.
- SEI (2006). CMMI Capability Maturity Model Integration for Development (CMMI-DEV). Technical report, Software Engineering Institute (SEI), Carnegie Mellon University. Pittsburgh, PA. <http://www.sei.cmu.edu/reports/06tr008.pdf>. Acesso em agosto 2012.
- Softex (2009). MPS.BR - Melhoria de Processo do Software Brasileiro - Guia Geral. Technical report. http://www.softex.br/mpsbr/_guias/guias/MPS.BR_Guia_Geral_2009.pdf. Acesso em agosto de 2012.
- Souza, E. F., OLIVEIRA, L. E., FALBO, R. A., and VIJAYKUMAR, N. L. (2012). Using ontologies to build a database to obtain strategic information in decision making. *In: ONTOBRAS - MOST. 5º Seminário de Pesquisas em Ontologias do Brasil, 7º International Workshop on Metamodels, Ontologies, Semantic Technologies*.
- Staab, S. and Maedche, A. (2000). Ontology engineering beyond the modeling of concepts and relations. *14th European Conference on Artificial Intelligence (ECAI). Workshop on Applications of Ontologies and Problem-Solving Methods*.
- Swartout, B., Patil, R., Knight, K., and Russ, T. (1996). Toward distributed use of large-scale ontologies. *In Proceedings of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop*.