

OntoDiSEN: uma ontologia para apoiar o desenvolvimento distribuído de software

Ana Paula Chaves¹, Igor Steinmacher¹, Gislaine Camila L. Leal², Elisa H. M. Huzita³

¹Coordenação de Informática – Universidade Tecnológica Federal do Paraná
Campus Campo Mourão – Campo Mourão – PR – Brasil

²Departamento de Engenharia de Produção – Universidade Estadual de Maringá
Avenida Colombo, 1648 – Maringá – PR – Brasil

³Departamento de Informática – Universidade Estadual de Maringá
Avenida Colombo, 1648 – Maringá – PR – Brasil

chavesana@gmail.com, igorfs@utfpr.edu.br
gclleal2@uem.br, emhuzita@din.uem.br

Abstract. *Global Software Development brought competitive advantages to organizations, but also imposed some drawbacks due to the physical distribution. A critical aspect of this approach is related to communication. In order to provide the same semantic understanding about information exchanged on the environment to all team members, a way to represent these information is needed to minimize the ambiguity. This paper presents OntoDiSEN, an application ontology for a distributed software development environment. The goal of this ontology is support communication among team members, allowing users to have the same understanding about a given piece of information.*

Resumo. *O Desenvolvimento Distribuído de Software (DDS) trouxe vantagens competitivas, mas também trouxe novos desafios. Um dos fatores críticos desse tipo de desenvolvimento é a comunicação. Para que todos os indivíduos, independente da sua localização geográfica, possam a mesma compreensão semântica sobre as informações trocadas pelo ambiente, é necessária uma forma de representar essas informações, aumentando a clareza e diminuindo as ambiguidades. Este artigo apresenta a ontologia OntoDiSEN, uma ontologia de aplicação para um ambiente de DDS. Seu objetivo é auxiliar a comunicação entre os indivíduos, possibilitando que as informações trocadas no ambiente sejam compreendidas por todos os participantes do trabalho colaborativo.*

1. Introdução

O Desenvolvimento Distribuído de Software (DDS) é uma forma de desenvolvimento em que equipes atuantes em um mesmo projeto, trabalhando colaborativamente, encontram-se distribuídas geograficamente. Entre suas vantagens estão a proximidade aos mercados, a redução de custos, a distribuição de recursos em âmbito global e o desenvolvimento contínuo [Prikladnicki and Audy 2004]. Entretanto, em contraposição às vantagens apresentadas, o DDS traz novos desafios, relativos à coordenação, cooperação e comunicação, entre outros fatores como diferenças culturais e temporais.

A comunicação é um fator crítico em projetos que envolvem equipes dispersas. Os participantes estão muito menos propensos a comunicação informal com participantes

de outras localidades devido à ausência de contato face-a-face e conversas de corredor [Mockus and Herbsleb 2001]. Os indivíduos nem sempre se conhecem pessoalmente, podendo haver falta de confiança em relação aos demais ou ausência de predisposição em ajudar outros membros da equipe, como pode ser observado em [Moe and Šmite 2008].

O DiSEN (*Distributed Software Engineering Environment*) é um ambiente de DDS que tem como objetivo fornecer ferramentas de apoio à comunicação, à persistência e à cooperação entre equipes dispersas. Diversos trabalhos foram desenvolvidos para este ambiente, como pode ser visto em [Huzita et al. 2007]. Dentre esses trabalhos, [Chaves 2009] propôs um modelo baseado em contexto para disseminação de informações, com o objetivo de facilitar o compartilhamento automático de informações sobre as interações que ocorrem no ambiente. Neste trabalho, o contexto de uma interação entre um agente e uma aplicação, com o objetivo de realizar alguma tarefa, é um conjunto de elementos instanciados que são necessários para apoiar a execução daquela tarefa, naquele momento [Vieira 2008].

Durante o desenvolvimento deste modelo, notou-se que, como as mensagens são enviadas automaticamente pelo ambiente, havia a necessidade de diminuir as possíveis ambiguidades e encontrar maneiras de possibilitar que todos os indivíduos que recebem a mensagem tenham a mesma compreensão semântica sobre o que foi informado. Neste artigo, portanto, é apresentado o projeto da OntoDiSEN, uma ontologia de aplicação, voltada para o ambiente DiSEN, cujo objetivo é descrever os conceitos e os elementos contextuais representados e compartilhados pelo modelo de disseminação de informações.

Este artigo possui, além desta, quatro seções, organizadas da seguinte forma: a Seção 2 apresenta os conceitos relacionados à ontologia; a Seção 3 descreve o ambiente DiSEN e o modelo baseado em contexto de disseminação de informações, para o qual a ontologia foi desenvolvida; a Seção 4 apresenta a ontologia OntoDiSEN; e a Seção 5 discute um cenário de aplicação e descreve os próximos passos dessa pesquisa.

2. Ontologia

O termo ontologia surgiu na Filosofia, para descrever as coisas do mundo real. Nos últimos anos, conquistou espaço no domínio da computação em áreas como Inteligência Artificial e Linguística Computacional. Nessas áreas, a definição mais aceita para o termo foi cunhada por [Gruber 1993], que afirma que ontologia é uma especificação formal e explícita de uma conceitualização. Para o autor, uma conceitualização corresponde a uma visão abstrata e simplificada do mundo que se deseja representar. Esta é a definição utilizada neste artigo, já que o objetivo da OntoDiSEN consiste em representar formal e explicitamente os conceitos existentes no ambiente DiSEN, oferecendo semântica a esses conceitos e, conseqüentemente, diminuindo ambiguidades e incompreensões.

Uma ontologia é um conjunto de entidades, também denominadas conceitos ou classes, que representam os conceitos do domínio [Noy and McGuinness 2001] e podem ser organizadas hierarquicamente. As entidades possuem propriedades, que correspondem às características e atributos que as identificam. Além disso, essas propriedades podem ter restrições, para aumentar a precisão da especificação. Cada entidade possui uma população de indivíduos, que representam as instâncias dos conceitos.

Para [Nunes et al. 2007], a motivação para usar ontologias está em: compartilhar um entendimento comum sobre a estrutura da informação, entre pessoas ou agentes

computacionais; analisar o conhecimento do domínio; e permitir o uso de mecanismos de inferência para raciocinar sobre vários contextos. Outra vantagem do uso de ontologias, destacada em [Neto 2006], é descrever um domínio modelando o conhecimento sem qualquer compromisso com a implementação de um sistema de software.

3. Ambiente DiSEN

O DiSEN é um ambiente que visa disponibilizar ferramentas de apoio à persistência e manipulação de informações e à comunicação e cooperação entre indivíduos de equipes de desenvolvimento geograficamente dispersas [Huzita et al. 2007]. No contexto deste ambiente, um modelo baseado em contexto para disseminação de informações foi proposto por [Chaves 2009]. Este modelo, denominado DiSEN-CSE (*DiSEN-Context Sensitive Environment*), tem como objetivo aumentar a percepção dos membros das equipes sobre ações que ocorrem no ambiente e, de acordo com o contexto que envolve essas ações e os indivíduos influenciados por ela, compartilhá-las automaticamente.

As informações sobre a construção dos objetos de colaboração produzidas por equipes distantes, podem apresentar ambiguidades ou falta de clareza, o que pode provocar falhas ou incertezas no desenvolvimento do software. A solução proposta por [Chaves 2009] foi incluir no DiSEN-CSE um elemento responsável pela representação das informações, que baseia-se em um modelo ontológico para promover a disseminação do contexto de forma uniforme e padrão. O objetivo é que, mesmo com entidades receptoras diferentes, a compreensão semântica seja a mesma.

4. OntoDiSEN

A OntoDiSEN é uma ontologia de aplicação, por ser desenvolvida especificamente para o ambiente DiSEN, no domínio do DDS. Existem diversas metodologias que tem como objetivo conduzir e auxiliar a construção de ontologias [Simperl and Tempich 2006]. A metodologia utilizada para a construção da OntoDiSEN corresponde a uma intersecção, baseada em [Martimiano 2006], entre os passos seguidos em trabalhos encontrados na literatura. Essa intersecção utiliza as abordagens propostas em [Fernández et al. 1997] e [Noy and McGuinness 2001] de forma complementar. A primeira, define as fases da metodologia e a segunda auxilia o processo de conceituação. Mais detalhes sobre a metodologia podem ser encontrados em [Martimiano 2006, Chaves 2009]. O desenvolvimento desta ontologia baseou-se no seguinte planejamento:

Definição do domínio: ambiente de desenvolvimento distribuído de software;

Definição da aplicação: ambiente DiSEN;

Definição do objetivo principal: representar de forma não ambígua as informações relacionadas ao contexto das ações de indivíduos, locais e ferramentas de um ambiente de DDS, mais especificamente, o DiSEN;

Definição dos usuários: os usuários da ontologia são os usuários do ambiente DiSEN e as ferramentas disponíveis no próprio ambiente;

Definido o planejamento, o próximo passo foi a aquisição de conhecimento. Para isso, foi necessário pesquisar nas diversas fontes de conhecimento, informações relevantes para compor o contexto de ações, sobre o domínio de DDS, e sobre as características do DiSEN. As técnicas utilizadas para essa aquisição foram: *brainstorming* com integrantes

do projeto DiSEN, que possuem familiaridade com a regra de negócio do ambiente; entrevistas com especialistas (coordenadora e 2 professores responsáveis pelo DiSEN); análise de textos formais contendo informações sobre o ambiente DiSEN (dissertações, trabalhos de conclusão de curso, relatórios técnicos e artigos publicados em anais de eventos) como os listados em [Huzita et al. 2007].

Com base no conhecimento adquirido por essas fontes, definiu-se questões de competência, que representam questões que a ontologia precisa responder, facilitando a fase de definição de conceitos. Para definir essas questões, o estudo concentrou-se nos elementos que auxiliam a definição do contexto apresentada na Seção 1 (localidade, identidade, atividade, tempo e presença) e nos elementos de percepção (quem, o que, quando, onde, porque e como). Percepção, neste artigo, refere-se à compreensão das atividades dos outros, que oferece um contexto para sua própria atividade [Dourish and Bellotti 1992]. Além disso, as questões estão centradas nas três categorias de recursos do ambiente: usuários, locais e ferramentas. Exemplos dessas questões são:

Usuário: Em que local o usuário está logado? (localidade); Que papéis o usuário desempenha? (identidade); Quais tarefas o usuário realiza? (atividade); Qual o estado do projeto do usuário? (atividade); Quando iniciou/concluiu um artefato? (tempo); Quais usuários do mesmo projeto estão logados? (presença)

Ferramenta: Em quais *workspaces* está instalada? (localidade); Que tipo de artefatos gera? (identidade); Qual projeto inclui a utilização dessa ferramenta? (atividade); Quem está utilizando a ferramenta agora? (presença)

Locais: Quem está conectado? (identidade); Há quanto tempo o local está disponível? (tempo); Qual dos locais permanece mais tempo disponível? (tempo)

Além das questões de competência, foi elaborado um mapa conceitual, apresentado na Figura 1, com o objetivo de facilitar a compreensão sobre os relacionamentos existentes entre os conceitos identificados durante a etapa de aquisição de conhecimento.

Com base na semântica descrita pela aquisição de conhecimento, os conceitos e relacionamentos foram transcritos na forma de entidades e propriedades, utilizando a linguagem OWL-DL¹ e a ferramenta Protégé². Em OWL, as propriedades são divididas em propriedades de objetos e de dados. Nas propriedades de objeto, estabelece-se uma relação entre duas entidades, sendo uma imagem (entidade que possui a propriedade) e um domínio (faixa de valores possíveis para a propriedade). Nas propriedades de dados, o domínio não é uma entidade, mas um tipo de dado, como *string*, *char*, *integer*, entre outros. As Figuras 2, 3 e 4 apresentam, respectivamente, o conjunto de entidades, propriedades de dados e de objetos da OntoDiSEN. A seguir, são descritas algumas das principais entidades e suas propriedades. As demais estão descritas em [Chaves 2009]:

Usuário: especialização de *Recurso*, são indivíduos que possuem acesso ao ambiente, podendo ser alocados a projetos, nesse caso, especializados em *Participantes*. Possui nome, senha, login, custo por hora, email. Um usuário pertence a uma localidade, pode ser responsável pelo funcionamento de um local, baixar e instalar ferramentas no seu *workspace* e acessa o ambiente utilizando um *workspace*.

¹<http://www.w3.org/TR/owl-features>

²<http://protege.stanford.edu>

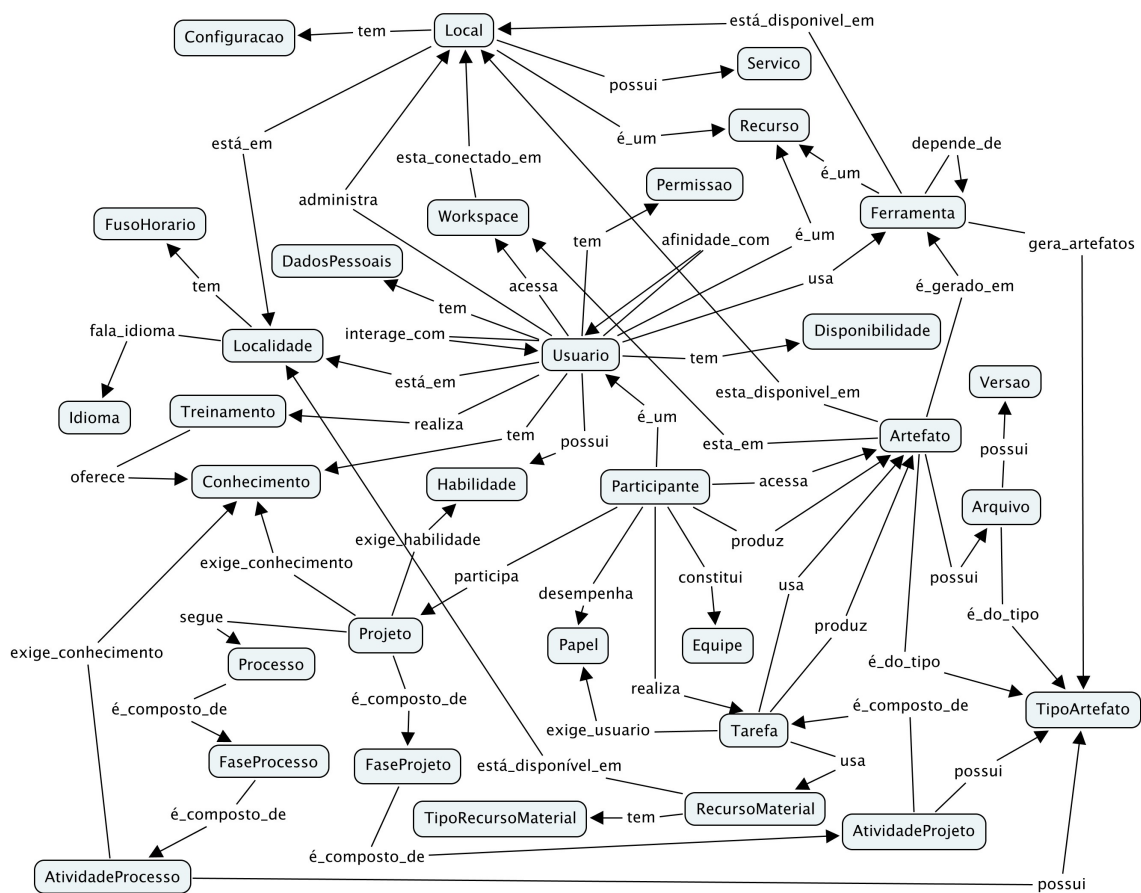


Figura 1. Principais conceitos e relacionamentos da OntoDiSEN.

Ferramenta: especialização de *Recurso*, consiste nas ferramentas de software que podem ser instaladas e utilizadas para a geração dos artefatos. Possui um conceito, que indica seu objetivo, uma data de atualização, fornecedor, nome, plataforma, tamanho e versão. Uma ferramenta é capaz de gerar artefatos de um tipo específico, pode depender de outra para seu funcionamento correto, está disponível em algum repositório de ferramentas, pode estar instalada em *workspaces* e utilizada por usuários.

Local: especialização de *Recurso*, são máquinas ou equipamentos de hardware que oferecem serviços ao ambiente, por exemplo, para funcionar como repositório de dados, artefatos ou ferramentas, para conectar *workspaces*, para interligar servidores. Cada local possui uma *configuracao*, um usuário responsável por garantir sua disponibilidade e está disponível em um lugar físico (*localidade*). Um local pode constituir um repositório de artefatos ou de ferramentas, além de poder possuir recursos materiais utilizados por um projeto (por exemplo, um servidor de impressão). Usuários acessam o ambiente conectando seu *workspace* em um local.

Workspace: espaço de trabalho que contém artefatos e ferramentas que os usuários utilizam para realizar seu trabalho. Um *workspace* está ativo quando está conectado a um local e pertence ao usuário que está conectado nele. Um *workspace* pode possuir cópias dos artefatos, funcionando como um repositório local, para que os usuários possam modificar os artefatos. Além disso, pode possuir ferramentas instaladas.



Figura 2. Entidades da OntoDiSEN



Figura 3. Propriedades de dados da OntoDiSEN

Processo: conjunto de ações predeterminadas que devem ser seguidas para o desenvolvimento de um produto de software. Possui nome, descrição e versão. Os processos são utilizados para guiar os projetos e é composto de fases.

Projeto: instância de um processo que inclui detalhes específicos da execução do processo, como os objetivos a serem atingidos, prazos e metas a serem cumpridos. Possui nome, descrição, datas de início e fim previstas, datas de início e fim efetivas. Um projeto possui participantes alocados para ele, é composto de fases e possui um estado, que indica se está em planejamento, andamento, concluído, cancelado ou suspenso. O projeto deve ser conduzido por um processo, respeitar um fuso horário e, para sua concretização, serão necessários participantes com habilidades e conhecimentos específicos. A diferença entre habilidades e conhecimentos foi definida, para o ambiente DiSEN, por [Lima 2004].

Em seguida, cada entidade foi formalmente definida, utilizando a lógica descritiva para inserir restrições. Como exemplo, definiu-se que acessar um *workspace* e possuir um login e uma senha são condições necessárias e suficientes para uma entidade ser considerada um usuário. Além disso, definiu-se que um usuário é um recurso que administra um local, está em uma localidade, usa uma ferramenta e que possui apenas essas carac-



Figura 4. Propriedades de objetos da OntoDiSEN

terísticas. Essa definição é apresentada, em lógica descritiva, na forma

$$\begin{aligned}
 & Usuario \equiv =1 \text{acessaWorkspace.Workspace} \sqcap =1 \text{login.String} \sqcap =1 \text{senha.String} \\
 & Usuario \sqsubseteq Recurso \sqcap \exists \text{administraLocal.Local} \sqcap =1 \text{estaEmLocalidade.Localidade} \sqcap \\
 & \exists \text{usaFerramenta.Ferramenta} \sqcap \forall \text{acessaWorkspace.Workspace} \sqcap \forall \text{administraLocal.Local} \sqcap \\
 & \exists \text{estaEmLocalidade.Localidade} \sqcap \forall \text{usaFerramenta.Ferramenta}
 \end{aligned}$$

Para verificar a consistência das entidades, propriedades e restrições geradas, a OntoDiSEN foi submetida ao raciocinador Fact++, integrado à interface da ferramenta Protégé, que não encontrou quaisquer inconsistências.

5. Considerações Finais

Este artigo apresentou a OntoDiSEN, uma ontologia de aplicação voltada para o ambiente DiSEN. A motivação para seu desenvolvimento foi a dificuldade de comunicação em um ambiente de DDS, a qual é influenciada pelas diferenças de idioma, cultura e fuso-horário. O desenvolvimento da ontologia objetiva incorporar ao modelo de disseminação de informações, DiSEN-CSE, uma forma de representação que diminua as ambiguidades nas mensagens enviadas pelo ambiente DiSEN aos participantes dos projetos, possibilitando que todos os indivíduos tenham a mesma compreensão semântica sobre o que está sendo informado. Após a formalização do conhecimento do domínio, através da construção da ontologia, as pesquisas que atualmente estão em andamento se concentram em aplicar as estratégias para a utilização da mesma no compartilhamento de informações sobre o contexto, conforme o modelo DiSEN-CSE. Uma limitação deste trabalho é que a OntoDiSEN foi desenvolvida especificamente para o ambiente DiSEN, o que limita seu aproveitamento em outros ambientes. Dessa forma, trabalhos futuros envolvem a generalização da ontologia para envolver conhecimentos relacionados à ambientes de DDS como um todo.

Conforme definido na Seção 2, um dos critérios para a escolha dessa forma de representação de informações é a capacidade de inferência. Assim, a ontologia está sendo utilizada para representar o contexto atual do ambiente DiSEN. Conforme definido pelo modelo DiSEN-CSE, quando uma ação ocorre, esta deve ser processada pela ontologia. Assim, agentes de software capazes de capturar essas ações e representar o novo estado

do sistema na ontologia estão sendo desenvolvidos. As mudanças no contexto serão então processadas, com base em regras de inferência, transformando as informações de contexto em ações de auxílio aos usuários. Os usuários do ambiente DiSEN recebem, então, mensagens automáticas sobre as ações que influenciam seu trabalho, oferecendo percepção sobre as ações que ocorrem e facilitando a tomada de decisões. Os resultados dessa pesquisa serão publicados em trabalhos futuros.

Referências

- Chaves, A. P. (2009). DiSEN-CSE: Um modelo baseado em context-awareness para disseminação de informações em um ambiente de desenvolvimento distribuído de software. Master's thesis, DIN, Universidade Estadual de Maringá.
- Dourish, P. and Belloti, V. (1992). Awareness and coordination in shared workspaces. In *ACM Conference on Computer-Supported Cooperative Work*, pages 107–114. ACM.
- Fernández, M., Gómez-Pérez, A., and Juristo, N. (1997). Methontology: from ontological art towards ontological engineering. In *Proceedings of the AAAI97 Spring Symposium*, pages 33–40, Stanford, USA.
- Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:199–220.
- Huzita, E. H. M., Tait, T. F. C., Colanzi, T. E., and Quinaia, M. A. (2007). Um ambiente de desenvolvimento distribuído de software – DiSEN. In *I Workshop de Desenvolvimento Distribuído de Software*, pages 31–38, João Pessoa - PB.
- Lima, F. (2004). Mecanismo de apoio ao gerenciamento de recursos humanos no contexto de um ambiente distribuído de software. Master's thesis, DIN – UEM.
- Martimiano, L. A. F. (2006). *Sobre a estruturação de informação em sistemas de segurança computacional: o uso de ontologia*. PhD thesis, ICMC – USP.
- Mockus, A. and Herbsleb, J. (2001). Challenges of global software development. In *Proceedings of the 7th International Symposium on Software Metrics*. IEEE CS.
- Moe, N. B. and Šmite, D. (2008). Understanding a lack of trust in global software teams: a multiple-case study. *Softw. Process*, 13(3):217–231.
- Neto, R. F. B. (2006). *Um processo de software e um modelo ontológico para apoio ao desenvolvimento de aplicações sensíveis a contexto*. PhD thesis, ICMC – USP.
- Noy, F. N. and McGuinness, D. L. (2001). Ontology development 101: a guide to creating your first ontology. Tech. Report KSL-01-05, Stanford Knowledge Systems Lab.
- Nunes, V. T., Santoro, F. M., and Borges, M. R. S. (2007). Um modelo para gestão de conhecimento baseado em contexto. In *XXVII CSBC - SBSC*, pages 1841–1854.
- Prikladnicki, R. and Audy, J. (2004). Munddos: Um modelo de referência para desenvolvimento distribuído de software. In *XVIII SBES*.
- Simperl, E. P. B. and Tempich, C. (2006). Ontology engineering: A reality check. In *OTM Conferences (1)*, volume 4275 of LNCS, pages 836–854. Springer.
- Vieira, V. (2008). *CEManTIKA: A Domain-Independent Framework for Designing Context-Sensitive Systems*. PhD thesis, CIn – Universidade Federal de Pernambuco.