

Position Paper: Ontologias no Controle da Adaptação ao Contexto em Espaço Ubíquo

Nelsi Warcken¹, Luiz A. M. Palazzo², Adenauer C. Yamin³

¹²³Escola de Informática – Universidade Católica de Pelotas (UCPel)
Rua Félix da Cunha, 412 - 96010-000 - Pelotas - RS - Brasil

{nelsi.warcken@gmail.com, lpalazzo@atlas.ucpel.tche.br,
adenauer@ucpel.tche.br}

***Resumo.** O objetivo central deste trabalho é o estudo de ontologias como mecanismo de controle da adaptação ao contexto na computação ubíqua, considerando suas principais características e desafios de pesquisa. Será criada uma ontologia para o ambiente computacional provido pelo middleware EXEHDA, com o propósito de tomar decisões de adaptação para este ambiente, com base em informações monitoradas, informações semânticas e inferências a partir das mesmas.*

1. Introdução

Computação ubíqua trata do acesso ao ambiente computacional do usuário, isto é, ao espaço ubíquo do usuário, em qualquer lugar, todo o tempo com qualquer dispositivo, constituindo um ambiente altamente distribuído, heterogêneo, dinâmico, móvel, mutável e com forte interação entre homem e máquina (AUGUSTIN et al., 2004). Ainda existem muitas limitações para o desenvolvimento de tais softwares, pois poucas linguagens e ferramentas estão disponíveis para a programação (WANT e PERING, 2005). As aplicações precisam “entender” e se adaptar ao ambiente, compreendendo o contexto em que estão inseridas (MACIEL, 2004). Essa nova classe de sistemas computacionais, sensíveis ao contexto, abre perspectivas para o desenvolvimento de aplicações muito mais ricas e elaboradas, que exploram a natureza dinâmica e a mobilidade do usuário. Entretanto, o desenvolvimento de aplicações que se adaptem continuamente ao ambiente e permaneçam funcionando mesmo quando o indivíduo se movimentar ou trocar de dispositivo (GRIMM et al., 2004), continua um desafio de pesquisa em aberto.

Contexto é qualquer informação que pode ser usada para caracterizar a situação de uma entidade. Uma entidade é qualquer pessoa, lugar ou objeto que é considerado relevante para a interação entre usuário e a aplicação, incluindo o próprio usuário e a própria aplicação (DEY et al., 2001).

O EXEHDA é um middleware adaptativo ao contexto, baseado em serviços, que tem por objetivo criar e gerenciar um ambiente ubíquo. As aplicações deste ambiente são distribuídas, móveis e adaptativas ao contexto.

Se tornará cada vez mais frequente, a utilização combinada das áreas de ontologias e computação ubíqua, pelo avanço rápido tanto da Web Semântica como dos ambientes ubíquos, além de facilitar a análise e a reutilização destas novas aplicações e a integração entre estas duas áreas, que estão tornando-se uma tendência natural em pouco tempo (HILERA, RUIZ, 2006).

A motivação para o desenvolvimento deste trabalho advém da pouca disponibilidade de material tratando da interseção das duas áreas: o controle da adaptação na computação

ubíqua e ontologias. Ontologia, além de ser um elemento importante da Web Semântica, evolução da Web atual (Sintática), é um bom instrumento para especificar os complexos conceitos da computação ubíqua, seu contexto (entidades ou elementos de contexto como usuários, dispositivos, serviços, localização, códigos por dispositivos e serviços, entre outros), atuando como um padrão para descrições, características, configurações e perfis, que serão compartilhados pelas aplicações.

O objetivo central do trabalho é o estudo de ontologias como mecanismo de controle de adaptação ao contexto na computação ubíqua, considerando suas principais características e desafios de pesquisa. Será criada uma ontologia para o ambiente computacional provido pelo middleware EXEHDA, e a proposta é tomar decisões de adaptação para este ambiente, com base em informações monitoradas, informações semânticas e inferências a partir das mesmas.

2. Ontologias: Principais Conceitos

Atualmente, a definição mais amplamente aceita e citada pelos autores da área de Computação é a que define ontologia como uma “especificação formal e explícita de uma conceituação compartilhada” onde: “conceituação” se refere ao modelo abstrato do mundo real; “explícita” significa que os conceitos e seus requisitos são definidos explicitamente; “formal” indica que a ontologia é processável por máquina, permite raciocínio automático e possui semântica lógica formal; compartilhada” significa que uma ontologia captura o conhecimento apresentado não apenas por um único indivíduo, mas por um grupo (GRUBER, 1993 e FENZEL, 2000).

As ontologias constituem o núcleo da Web Semântica. Aplicações Semânticas buscam interpretar o significado de textos e outros formatos de dados, permitindo estabelecer relacionamentos e inferências entre os dados, a reutilização e a conectividade são características fundamentais para tais aplicações.

Categorias genéricas de ontologias: Ontologias de Domínio que descrevem conhecimento sobre um domínio ou sub-domínio; Ontologias como Artefatos de Software: usadas como artefatos de diversos tipos no processo de desenvolvimento de aplicações ou durante a execução de uma aplicação (HILERA e RUIZ, 2006).

Atualmente a linguagem mais completa para representar ontologias é OWL (Web Ontology Language), é uma extensão do RDFS (Resource Description Framework Schema) e permite obter um maior nível de expressividade.

3. SOAM: An Environment Adaptation Model for the ubiquitous Semantic Web

Dentre as várias alternativas de trabalhos relacionados, destacamos o SOAM em função de proximidade com as motivações do trabalho em andamento. Em SOAM, o modelo de adaptação ao ambiente é baseado em preferências, onde parte do contexto do usuário, está construído pelas preferências semânticas sobre as condições ambientais (temperatura e iluminação) (VASQUEZ, 2006).

Um *smobject* (*smart object*) é um agente, na forma de uma peça de software, representando um dispositivo inteligente, vários dispositivos ou parte do evento de um dispositivo. *Smobjects* capturam um subconjunto de condições ambientais, que fornecem informação de contexto percebida em requisição solicitada, e age sobre o mesmo ou outro subconjunto de condições ambientais, a fim de modificá-lo e adaptá-lo quando necessário. *Smobjects* necessitam de acesso interno de sensores, portas de comunicação, capacidade de armazenamento, se disponível no dispositivo, e assim por

diante. Os *smartobjects* recebem ou percebem informações de dispositivos, usuários e procuram ou inferem perfis e configurações destes dispositivos e usuários, modelados numa ontologia. Todas estas informações necessárias para efetuar as adaptações, são enviadas pelos *smartobjects* para o “orquestrador” que deverá efetivá-las. Não foi explicado como o “orquestrador” funciona e nem como continua o processo de adaptação.

4. Proposta de Trabalho

A proposta do trabalho é criar um modelo de controle da adaptação dinâmica de aplicações em ambiente ubíquo. A idéia é controlar estas adaptações quando da tomada de decisões considerando o contexto, com base em informações monitoradas, informações semânticas e inferências a partir destas mesmas informações.

Etapas da adaptação:

- detecção de alterações, que engloba monitoração, interpretação e notificação, visa observar o ambiente para detectar e notificar alterações significativas;
- escolha da ação, engloba a seleção da ação adaptativa entre alternativas pré-definidas pelo programador;
- ativação da ação, refere-se à execução da ação adaptativa selecionada.

Nesta perspectiva, a adaptação se refere à alteração no comportamento, na estrutura ou na interface da aplicação, em resposta a trocas arbitrárias no estado do elemento de contexto.

As características deste modelo devem ter variadas noções de adaptação, permitindo identificar os requisitos de propósito geral e específico para expressar adaptabilidade ao contexto, os quais são:

- identificação dos elementos de contexto (entidades) que compõem o ambiente e modelam o contexto de interesse (classes no OWL): usuários, dispositivos, serviços, localização, coleções de códigos, de interfaces para dispositivos e serviços;
- descrição do comportamento adaptativo: coleções de ações e o respectivo conjunto de restrições onde estas podem ocorrer;
- descrição de preferências de usuários em determinadas situações;
- descrição de políticas: regras de propósito geral ou particular que orientam as decisões da adaptação realizadas pelo gerenciador de adaptação;
- inferências a partir das informações da ontologia para subsidiar as tomadas de decisões do gerenciador de adaptação;
- estabelecimento de um padrão para descrições, características, e configurações dos elementos do contexto.

Este modelo pode ter como vantagens:

- facilidade de manutenção e reutilização das diferentes informações necessárias ao modelo (vide acima);
- facilidade e maior agilidade no desenvolvimento de novas aplicações;
- descrição e definição dos estados de cada elemento ativo e das ações a serem tomadas pelo gerenciador de adaptação do ambiente ubíquo;
- possibilidade de inferências para a tomada de decisões de adaptação e descrição das diferentes políticas de adaptação necessárias para a adaptação dinâmica.

Assim sendo, o objetivo é o de adaptar os serviços da computação ubíqua ao ambiente sem, ou mínima, intervenção explícita do usuário, utilizando a “inteligência” de dispositivos e de softwares.

Para a criação e manutenção da ontologia sugerida será utilizada a linguagem OWL, o editor de ontologias Protégé e a API JENA.

Definição do modelo de ontologia proposto:

- Ontologia de Domínio: Computação Ubíqua;
- Ontologia de Subdomínio: Controle da Adaptação ao Contexto;
- Ontologia como Artefato de Software:
 - Em tempo de desenvolvimento de serviços/aplicações: modelagem, criação, manutenção e integração de serviços.
 - Em tempo de execução
 - ◆ Aplicações orientadas a ontologia: atualização do modelo ontológico.
 - ◆ Aplicações cientes da ontologia: descoberta de recursos, sensibilidade ao contexto, controle da adaptação ao contexto.

5. Referências Bibliográficas

- AUGUSTIN, I. et al. ISAM, Joining Context-Awareness and Mobility to Building Pervasive Applications. In: ILYAS, M.; MAHGOUB, I. (Eds.) **Mobile Computing Handbook**. Boca Raton: CRC Press, 2004. p. 73-94.
- DEY, A. ; ABOWD, G.; SALBER,D. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. **Human-Computer Interaction**, [S.l.], v.16, n.2,3&4, p.97-166, 2001.
- FENSEL, D. Ontologies: Silver Bullet for Knowledge Management and Eletronic Commerce. Springer - Verlag, Berlin, 2000.
- GRIMM, R. et al. System support for pervasive applications. **ACM Transactions on Computer Systems**, New York, v.22, n.4, p. 421-486, Nov. 2004.
- GRUBER, T. A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition, [S.l.], p.199–220, 1993.
- HILERA, J.R.; RUIZ, F. **Ontologies in Ubiquitous Computing**. <ftp.informatik.rwth-aachen.de/Publications/CEUR-S/Vol-208/paper15.pdf>, 2006
- MACIEL, R. S. P.; ASSIS, S. R. de. **Middleware: Uma solução para o desenvolvimento de aplicações distribuídas**. In: Científico, Ano IV. [S.l.: s.n.], 2004.
- VAZQUEZ, J.I., LÓPEZ de IPIÑA, D. and SEDANO, I. SOAM: An Environment Adaptation Model for the Pervasive Semantic Web. **UWSI 2006: The Second Ubiquitous Web Systems and Intelligence Workshop**. May 2006.
- WANT, R.; PERING, T. System Challenges for Ubiquitous & Pervasive Computing. In: INTERNATIONAL CONFERENCE OF SOTWARE ENGINEERING, ICSE, 27., 2005, St. Louis. **Proceedings...** New York: ACM Press. 2005. p. 9-14.
- YAMIN, A. C. **Arquitetura para um Ambiente de Grade Computacional Direcionada às Aplicações Distribuídas, Móveis e Conscientes de Contexto da Computação Pervasiva**. 2004. 204f. Tese (Doutorado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.