

DESCOBERTA SEMÂNTICA DE RECURSOS NA UBICOMP: UM ESTUDO DE CASO APLICADO A CASAS DE VEGETAÇÃO

NELSI WARKEN¹
RENATO M. DILLI²
SERGIO L. RODRIGUES³
LUTHIANO R. VENECIAN⁴
JOÃO L. B. LOPES⁵
ADENAUER C. YAMIN⁶
CLAUDIO F. R. GEYER⁷

RESUMO: Na Computação Ubíqua (UbiComp) os recursos que constituem os ambientes devem estar compartilhados para serem acessados de qualquer lugar e a qualquer momento. Este artigo apresenta o EXEHDA-SD (*Semantic Discovery*), um mecanismo para descoberta de recursos, que agrega em sua arquitetura tecnologias para o processamento semântico de requisições por recursos. Com isso, busca-se aumentar a expressividade na representação e consulta de recursos. Para avaliar as funcionalidades do mecanismo é apresentado um estudo de caso desenvolvido na Embrapa Clima Temperado.

PALAVRAS-CHAVE: Computação Ubíqua, Ontologias, Casas de Vegetação.

SEMANTIC DISCOVERY OF RESOURCES IN UBICOMP: A CASE STUDY APPLIED TO GREENHOUSES

ABSTRACT: In Ubiquitous Computing (UbiComp) the resources that constitute the environment must be shared to be accessed from anywhere and anytime. This article presents the EXEHDA-SD (EXEHDA-Semantic Discovery), a mechanism for resource discovery, which combines in its architecture technologies for the semantic processing of resources requests. With this, we intend to increase the expressiveness in representation and query for resources. To evaluate the features of the mechanism is presented a case study developed at Embrapa Temperate Climate.

KEYWORDS: Ubiquitous Computing, Ontologies, Greenhouses.

1. INTRODUÇÃO

Um mecanismo para descoberta de recursos deve gerenciar a relação entre os consumidores e provedores de recursos, permitindo a localização dos recursos disponíveis, com pouca ou nenhuma intervenção do usuário (ROBINSON; INDULSKA, 2007).

O emprego de ontologias e tecnologias semânticas eleva a expressividade na representação e consulta dos recursos, evitando ambiguidade de conceitos e possibilitando a realização de inferência sobre o modelo ontológico (SOLDATOS et al., 2007).

Considerando este cenário, propomos o EXEHDA-SD, um mecanismo para descoberta de recursos com suporte semântico que visa qualificar o processo de consulta por recursos no ambiente ubíquo gerenciado pelo *middleware* EXEHDA (LOPES et al., 2007).

¹ Mestre em Ciência da Computação – Embrapa Clima Temperado – nelsi.warken@cpact.embrapa.br

² Mestre em Ciência da Computação – Instituto Federal Sul-rio-grandense – dilli@ifsul.edu.br

³ Mestre em Ciência da Computação – Instituto Federal Sul-rio-grandense – sergio@ifsul.edu.br

⁴ Mestre em Ciência da Computação – Universidade Católica de Pelotas – venecian@ucpel.tche.br

⁵ Mestre em Ciência da Computação – Universidade Federal do Rio Grande do Sul – jlblopes@inf.ufrgs.br

⁶ Doutor em Computação – Universidade Federal de Pelotas – adenauer@ufpel.edu.br

⁷ Doutor em Informática – Universidade Federal do Rio Grande do Sul – geyer@inf.ufrgs.br

O modelo busca suprir demandas inerentes aos ambientes ubíquos, tais como: escalabilidade, preferências do usuário, notificação ao usuário da disponibilidade do recurso desejado, controle de acesso a recursos de acordo com o perfil do usuário e controle da disponibilidade de recursos através de intervalos de tempo (COSTA; YAMIN; GEYER, 2008).

O artigo está organizado da seguinte forma: a seção 2 apresenta a arquitetura de software do EXHEDA-SD; a seção 3 destaca um estudo de caso; a seção 4 sistematiza e compara os trabalhos relacionados; e a seção 5 apresenta as considerações finais.

2. ARQUITETURA DE SOFTWARE

A arquitetura de software do EXHEDA-SD está organizada em três componentes distintos.

O **Componente Cliente** (CC) é responsável pela seleção dos recursos desejados, através de critérios, e um perfil de descoberta que define características pertinentes no processamento da consulta, e ainda, dependendo da natureza da consulta são utilizadas as preferências do usuário. O CC é composto dos módulos Construtor de Consultas e Administrador de Reservas. O Construtor de Consultas é responsável por interpretar os recursos necessários para atender uma aplicação e gerar o arquivo XML com a pesquisa desejada. O arquivo XML é enviado ao **Componente Diretório** (CD) para geração da consulta em formato SPARQL (PRUD'HOMMEAUX; SEABORNE, 2008). O Administrador de Reservas é um módulo responsável por notificar o cliente quando o diretório verificar que o recurso desejado tornou-se disponível.

A notificação do estado atual do recurso é responsabilidade do **Componente Recurso** (CR). Isto é feito por troca de mensagens com o CD, dentro de um intervalo de tempo (*lease*). Outra função do CR é anunciar seus recursos em células que estão sendo visitadas. Quando o usuário ingressa em uma nova célula o CR envia a descrição dos seus recursos para o CD da célula que está sendo visitada. O CD adiciona as instâncias referentes à descrição do recurso em sua ontologia.

Por sua vez, o (CD) é formado por quatro módulos: Tratador de Recursos, Processador Semântico, Comunicador P2P e Seletor. Quando o CC solicita que seja notificado sobre a disponibilidade de algum recurso é realizada uma consulta sobre todos os recursos "Ativos" no ambiente ubíquo. Caso a consulta não retorne resultados que atendam à requisição, essa será refeita com uso dos recursos "Inativos", mas existentes na ontologia. Retornando recursos que satisfaçam a requisição, o **Tratador de Recursos** notifica o CC quando o CR do recurso desejado renovar o *lease*, alterando o estado do recurso para "Ativo". Neste estado o recurso estará disponível para ser alocado pelo *Resource Broker* do *middleware* EXHEDA. Esse módulo também pode controlar o agendamento da disponibilidade do recurso.

O **Processador Semântico** emprega tecnologias semânticas, como a API Jena (DICKINSON 2009) e a linguagem de consulta SPARQL. Esse módulo é formado pelos seguintes componentes: (i) Modelo Ontológico (vide Figura 1), constituído por duas ontologias representadas em OWL (MCGUINNESS; VAN HARMELEN, 2009): (a) OntUbi: ontologia do ambiente ubíquo; e (b) OntSD: desenvolvida para atender exclusivamente o EXHEDA-SD; (ii) Raciocinador, responsável por aplicar regras e processar as consultas realizadas em SPARQL. O raciocinador procura por recursos idênticos e semelhantes ao solicitado; e (iii) Persistência de Dados, responsável pelo armazenamento das triplas da ontologia no banco de dados PostgreSQL. O Processador Semântico é responsável pela instanciação de novos recursos e processamento de regras de inferência, possibilitando ao mecanismo de descoberta o reconhecimento de conceitos implícitos na ontologia. Para processar as regras é utilizada a API Jena que adiciona os conceitos deduzidos pelas regras na ontologia e a linguagem SPARQL que realiza a consulta, considerando o novo conceito.

Através do suporte à persistência de dados da API Jena, as triplas da ontologia são armazenadas, tornando possível executar consultas SPARQL diretamente no banco de dados.

Para processar a solicitação do pesquisador, o Componente Cliente gera um arquivo XML com as especificações de recursos necessárias para realizar a consulta e envia ao Tratador de Recursos do Componente Diretório da célula. Por sua vez, o Processador Semântico traduz e processa a consulta de XML para SPARQL, envia para o Seletor aplicar as preferências do usuário. Por fim, o Tratador de Recursos envia no formato XML o resultado ao CC.

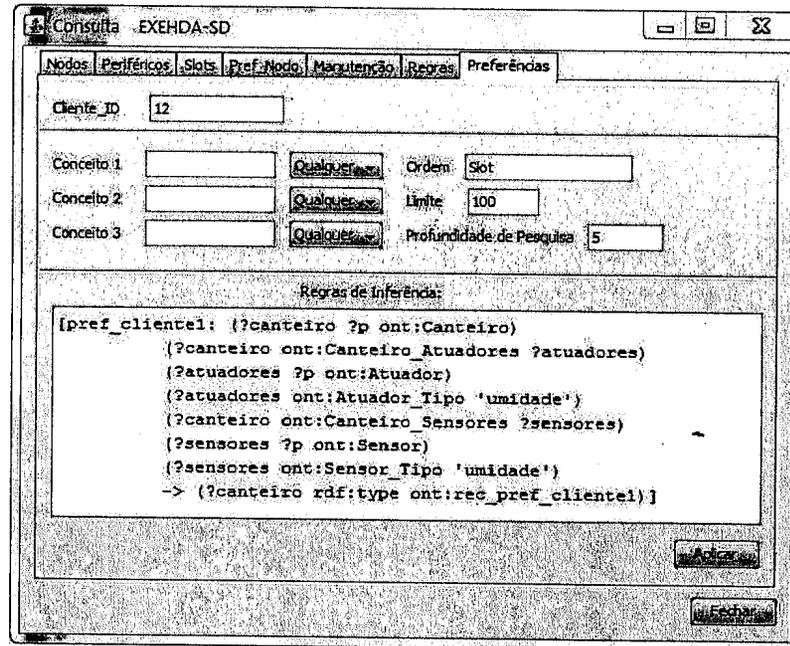


Figura 2: Preferências do Cliente

A Figura 3 mostra o resultado obtido pelo cliente. Os *slots* 3, 6 e 45 possuem sensores e atuadores de umidade. Os dois primeiros estão localizados na mesma célula (célula_10) em que o cliente estava no momento da solicitação da consulta. O *slot* 45, localizado na "célula_6" (Sede2), é exibido em função da profundidade de pesquisa definida pelo cliente (vide Figura 2), a qual altera o escopo da consulta para utilizar células vizinhas estáticas e dinâmicas. Como está definido o valor "5" na profundidade de pesquisa, os Comunicadores P2P replicam a consulta aos Comunicadores P2P de células vizinhas, até o quinto nível.

slot	canteiro	célula	célulaID	atuadores	atuadorTipo	sensores	sensorTipo	perfil
Slot_6	Canteiro_8	Célula_10	EMBRAPA_SE...	Atuador_7	umidade	Sensor_4	umidade	Pesquisador
Slot_3	Canteiro_42	Célula_10	EMBRAPA_SE...	Atuador_8	umidade	Sensor_2	umidade	Pesquisador
Slot_45	Canteiro_35	Célula_6	EMBRAPA_SE...	Atuador_1	umidade	Sensor_21	umidade	Pesquisador

Figura 3: Resultado da Consulta

4. TRABALHOS RELACIONADOS

Allemand (2006) utiliza ontologias descritas em OWL para representação dos recursos e a linguagem RDQL para consultar recursos instanciados no modelo ontológico. O projeto foi concebido para uso em ambientes de grade e não considera as preferências do usuário, ao contrário do EXEHDA-SD. Ainda, o EXEHDA-SD se diferencia pelo uso da linguagem de consulta SPARQL, considerada uma evolução da RDQL, sendo recomendada pelo W3C como padrão para realização de consultas em ontologias descritas com OWL.

PerDiS (SCHAEFFER, 2005), assim como o EXEHDA-SD, foi modelado para descoberta de recursos em ambientes ubíquos e utiliza redes *super peer* para localização de recursos entre células. No EXEHDA-SD a representação e consulta por recursos é feita através de ontologia, descrita em OWL, e SPARQL. Já no PerDiS os recursos são representados em XML e a consulta é realizada pela comparação de atributos e valores, utilizando XML. Assim, ao contrário do EXEHDA-SD, o PerDiS permite apenas *matching* sintático e não possui raciocinadores para inferência de recursos.

OMM (TANGMUNARUNKIT et al., 2003) é um serviço de *matchmaking* em ambiente de grade. As consultas são realizadas através da linguagem TRIPLE sobre ontologias descritas em RDF *Schema*. O EXEHDA-SD utiliza SPARQL e OWL, ambas são recomendadas pela W3C. OWL se traduz em uma evolução de outras linguagens como RDF *Schema*.

5. CONSIDERAÇÕES FINAIS

O principal diferencial do EXEHDA-SD diz respeito à maior expressividade na representação e consulta de recursos. Isso é obtido através da concepção de um modelo arquitetural baseado em ontologias, que provê processamento semântico das requisições por recursos.

O estudo de caso apresentado demonstrou a aplicabilidade do mecanismo de descoberta de recursos no atendimento de uma das atividades fins da Embrapa Clima Temperado, indicando positivamente a viabilidade da proposta.

REFERÊNCIAS

- ALLEMAND, J. N. C. **Serviço Baseado em Semântica para Descoberta de Recursos em Grade Computacional**, 2006. (Mestrado em Ciência da Computação), UnB, Brasília, DF.
- COSTA, C. A.; YAMIN, A. C.; GEYER, C. F. R. **Toward a General Software Infrastructure for Ubiquitous Computing**. *IEEE Pervasive Computing*, Los Alamitos, CA, USA, 2008.
- DICKINSON, I. **Jena Ontology API**. Disponível em: <<http://jena.sourceforge.net/ontology/>> Acesso em maio de 2011.
- DILLI, R. M. **Uma Proposta para Descoberta de Recursos na Computação Ubíqua com Suporte Semântico**, 2010. Mestrado em Ciência da Computação, UCPEL, Pelotas, RS.
- LOPES, J. L. B.; PILLA, M. L.; YAMIN, A. C. EXEHDA: a middleware for complex, heterogeneous and distributed applications. **Iberian-American Conference on Technology Innovation and Strategic Areas**. 2007.
- MCGUINNESS, D. L.; HARMELEN, F. van. **OWL Web Ontology Language Overview**. Disponível em: <<http://www.w3.org/TR/owl-features/>> Acesso em maio de 2011.
- PRUD'HOMMEAUX, E.; SEABORNE, A. **SPARQL - A Query Language for RDF**. Disponível em: <<http://www.w3.org/TR/rdf-sparql-query/>> Acesso em maio de 2011.
- ROBINSON, R.; INDULSKA, J. **Resource Discovery in Pervasive Computing Environments**. *Handbook on Mobile Ad Hoc and Pervasive Communications*. American Scientific Publishers, 2007.
- SCHAEFFER, A. E. **PerDiS: um Serviço para Descoberta de Recursos no ISAM Pervasive Environment**, 2005. Mestrado em Computação, UFRGS, Porto Alegre, RS.
- SOLDATOS, J., STAMATIS, K., AZODOLMOLKY, S., PANDIS, I., and POLYMENAKOS, L. Semantic web technologies for ubiquitous computing resource management in smart spaces. *Int. J. Web Eng. Technol.*, 3(4):353–373, 2007.
- TANGMUNARUNKIT, H.; DECKER, S.; KESSELMAN, C. **Ontology-based resource matching in the grid - the grid meets the semantic web**. In Fensel, D., Sycara, K. P., and Mylopoulos, J., editors, **International Semantic Web Conference**, volume 2870 of *Lecture Notes in Computer Science*, pages 706–721. Springer, 2003.