

Descoberta de Recursos no Middleware EXEHDA para o Cenário da IoT

Maurício M. de Azevedo¹, Renato M. Dilli², Patrícia T. Davet², Adenauer C. Yamin^{1,2}

¹Universidade Católica de Pelotas (UCPel)

Rua Gonçalves Chaves, 373 - Centro - CEP 96015-560 – Pelotas - RS - Brasil

²Universidade Federal de Pelotas (UFPel)

Rua Gomes Carneiro, 1 - Centro - CEP 96010-610 – Pelotas – RS – Brasil

madrugadeazevedo@gmail.com, {renato.dilli, ptdavet, adenauer}@inf.ufpel.edu.br

Abstract. *The high number of devices connected to the Internet, provided by the infrastructure of the Internet of Things, brought new research challenges to be addressed. One of these challenges is how to make the discovery of the increasing new resources offered by this scenario, making access as easy and transparent as possible. Thus, this article presents a proposal for architectural model integrated with middleware EXEHDA, a new Resource Discovery Service directed the IoT. The research is ongoing, and the studies made so far are promising.*

Resumo. *O elevado número de dispositivos conectados à Internet, propiciado pela infraestrutura da Internet das Coisas, trouxe novos desafios de pesquisa a serem endereçados. Um destes desafios está em como realizar a descoberta dos crescentes recursos disponibilizados por este cenário, tornando o seu acesso o mais facilitado e tão transparente quanto possível. Desta forma, este artigo apresenta uma proposta de modelo arquitetural integrado ao middleware EXEHDA, de um novo Serviço de Descoberta de Recursos direcionado à IoT. A pesquisa está em andamento, e os estudos realizados até o momento se mostram promissores.*

1. Introdução

Há vários desafios a serem superados para o desenvolvimento da IoT (*Internet of Things*), entre eles, questões de identificação dos inúmeros dispositivos levando a uma necessidade de transição para IPv6, eficiência energética e a descoberta dos dispositivos e seus recursos [Gubbi et al. 2013].

A descoberta de recursos é um aspecto importante na IoT devido ao elevado número de dispositivos distribuídos dinamicamente e com características heterogêneas, o que constitui um desafio na identificação e disponibilização de seus recursos. O sistema precisa ter conhecimento dos recursos que possui e quais atendem as necessidades de uma determinada aplicação, para que possam ser alocados e descartados com a mínima interferência do usuário.

Tendo em vista este cenário, um mecanismo de descoberta é proposto visando contribuir com os esforços de pesquisa em descoberta de recursos e processamento semântico do LUPS (*Laboratory of Ubiquitous and Parallel Systems*) da Universidade Federal de

Pelotas. Este trabalho visa remodelar a arquitetura de software existente no serviço de descoberta do middleware EXEHDA para dar suporte à IoT.

O artigo está organizado da seguinte forma: na Seção 2 são apresentados conceitos e definições sobre Descoberta de Recursos, na Seção 3 conceitos sobre IoT, na Seção 4 é apresentado o Escopo do Ambiente Ubíquo, na Seção 5 o Modelo Proposto, a Seção 6 discute os Trabalhos Relacionados e na Seção 7 estão as Considerações Finais.

2. Descoberta de Recursos

Dois níveis diferentes de descoberta são encontrados nos sistemas e mecanismos de descoberta utilizados na Internet das Coisas e na Computação Ubíqua [Edwards 2006]. O primeiro nível é a descoberta de recursos, ou seja, a descoberta de dispositivos na rede; e o segundo nível é a descoberta de serviços, métodos e funções oferecidos por um recurso específico. Normalmente, a Internet é considerada como um recurso a partir de um ponto de vista geral, onde os serviços são parte desses recursos. No entanto, quando a Internet não está limitada a apenas arquivos, aplicações e serviços, também é necessária a identificação e localização física das informações.

O mecanismo do serviço de descoberta permite a detecção automática de serviços oferecidos por qualquer nó na rede [Marin-Perianu et al. 2005]. Em outras palavras, a descoberta de serviços é a ação de encontrar um fornecedor de serviços para um serviço solicitado. Quando a localização do serviço exigido é recuperada, o cliente terá acesso ao recurso.

Para [Klusck 2014] o serviço de descoberta é o processo de localização de serviços existentes que são relevantes para uma determinada requisição baseado na descrição de sua semântica.

3. Internet das Coisas

A Internet das Coisas (IoT), vem ganhando destaque como o novo paradigma de evolução da Internet [Charith et al. 2014]. Esta evolução deve-se ao fato que a IoT preconiza a ideia do tudo conectado, ou seja qualquer “coisa”, tais como utensílios, equipamentos pessoais, redes de energia, como também animais e o nosso corpo podem possuir sensores capazes de gerar informações que são compartilhadas tendo como principal meio de interoperação, a Internet. O compartilhamento destas informações disponibilizadas pelo cenário da IoT ocorre automaticamente com outros sistemas computacionais externos, o que introduz o conceito de objetos inteligentes.

Os principais desafios que devem ser vencidos no desenvolvimento de sistemas de suporte para Internet das Coisas são os seguintes [Miorandi et al. 2012]: (i) Heterogeneidade de dispositivos; (ii) Escalabilidade; (iii) Eficiência energética; (iv) Auto-organização; (v) Interoperabilidade semântica e gerenciamento de dados.

4. Escopo do Ambiente Ubíquo

O ambiente é instanciado e gerenciado pelo EXEHDA [Davet 2016] [Lopes et al. 2014]. Este ambiente computacional é constituído por células de execução, nas quais os dispositivos computacionais são distribuídos (vide Figura 1). Cada célula é constituída dos seguintes componentes: (i) EXEHDAbase, o elemento central da célula, sendo responsável por

todos serviços básicos e constituindo referência para os demais elementos; (ii) o EXEH-DAnodo que corresponde aos dispositivos computacionais responsáveis pela execução das aplicações; (iii) o EXEHDAAnodo móvel, um subcaso do anterior, que corresponde aos dispositivos tipicamente móveis que podem se deslocar entre as células do ambiente ubíquo, (iv) o EXEHDAborda, responsável por fazer a interoperação entre os serviços do middleware e os diversos tipos de *gateways*; e (v) o EXEHDAgateway, que consiste no elemento responsável por setorizar pontos de coleta e/ou atuação distribuídos, disponíveis no meio físico, realizando a interação destes com os outros componentes do middleware.

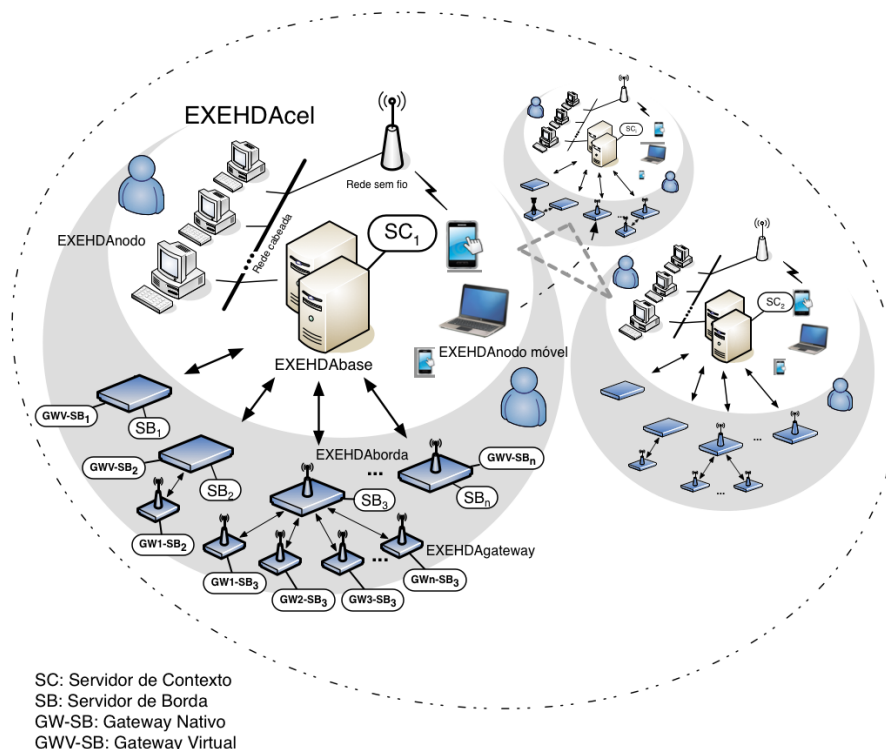


Figura 1. Organização Celular do Ambiente Ubíquo Gerenciado pelo EXEHDA

O middleware EXEHDA provê comunicação: (i) entre os *gateways* e o Servidor de Borda; (ii) entre os Servidores de Borda e o Servidor de Contexto; (iii) entre os Servidores de Contexto localizados em diferentes células do ambiente ubíquo; e (iv) com outros serviços do *middleware* ou aplicações. Além disso, é previsto que algum dos tipos de *gateways* podem não estar presentes, a qual possibilita utilizar somente *gateways* físicos (Nativos ou Proprietários), somente Gateway Virtual, ou ambos.

5. Modelo Proposto

O modelo proposto utiliza tecnologias de Web Semântica para a descrição e consulta de recursos, com intuito de potencializar a descoberta de recursos no ambiente. O pro-

cessamento das requisições por recursos são inferidas na ontologia através da API Jena, banco de dados para a persistência do modelo ontológico e linguagem de programação SPARQL. Também estão previstas as seguintes características:

Presença de Dispositivos confinados: dispositivos com poder computacional limitado, muitas vezes precisam estar conectados à gateways com interface de rede. O gateway deverá ter uma camada de software para gerenciar o estado dos recursos conectados. Deverá ser definido o meio de comunicação até o componente Diretório do mecanismo de descoberta localizado no EXEHDABase.

Otimização do Processamento Semântico: a arquitetura do mecanismo de descoberta será remodelada, limitando a ontologia para a descrição e consulta de recursos.

Aprimoramento na seleção dos recursos: após a realização do *matching* um resultado poderá conter uma quantidade muito grande de recursos disponíveis no ambiente, cabe ao mecanismo de descoberta selecionar os recursos mais adequados à solicitação do cliente utilizando critérios que qualificam os mais adequados.

A arquitetura proposta na Figura 2 está organizada em três componentes distintos: (CD) Componente Diretório, (CR) Componente Recurso e (CC) Componente cliente.

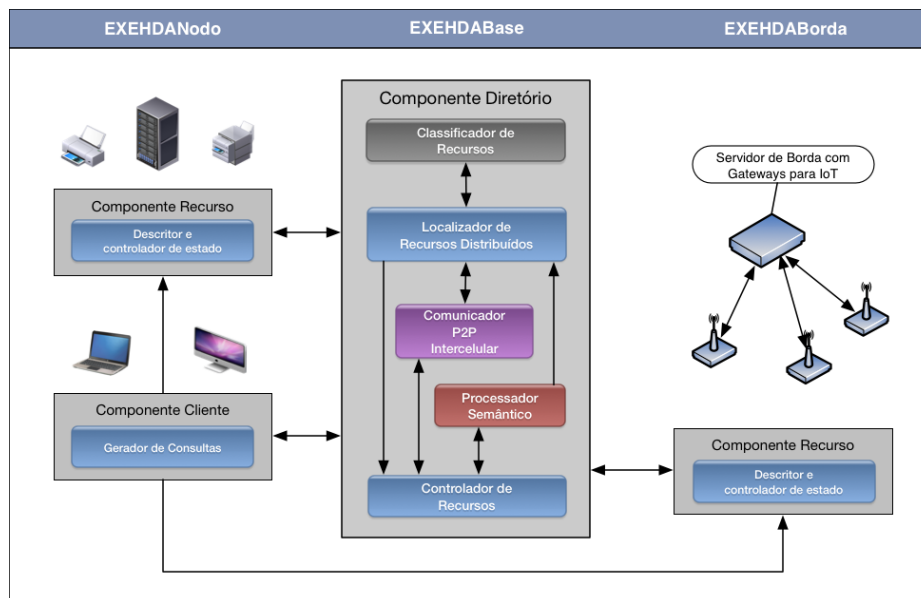


Figura 2. Arquitetura do Serviço de Descoberta

5.1. CC - Componente Cliente

O CC é responsável pela especificação dos recursos desejados, através de critérios que definem características pertinentes no processamento da consulta. O cliente é formado por componentes de software de uma aplicação instanciada no modelo ontológico do ambiente ubíquo. Na condição de clientes, estes componentes são os responsáveis pelo disparo da consulta por recursos. O CC é composto do módulo Gerador de Consultas.

5.1.1. CR - Componente Recurso

O Componente Recurso (CR) é responsável por notificar o estado atual recurso. Isto é feito por troca de mensagens com o CD, dentro de um intervalo de tempo. Quando o recurso ingressa pela primeira vez em uma célula, o CR envia a descrição dos seus recursos para o CD da célula onde está localizado o recurso. O CD gera uma instância do recurso na ontologia local. O CR é composto pelo módulo Descritor e Controlador de Estado.

O CR é ativado em nodos do ambiente ubíquo que possuem recursos a ser compartilhados. Este componente também é ativado no EXEHDABorda, cuja finalidade é anunciar a presença dos dispositivos confinados que estão conectados aos gateways da célula.

5.2. CD - Componente Diretório

O Componente Diretório é formado por cinco módulos: Controlador de Recursos, Processador Semântico, Comunicador P2P Intercelular, Localizador de Recursos Distribuídos e Classificador de Recursos.

5.2.1. Controlador de Recursos

O Controlador de Recursos tem as seguintes funcionalidades: (i) realizar a manutenção dos recursos no diretório; (ii) disponibilizar uma interface de consulta aos usuários para que possam verificar se os recursos desejados estão disponíveis no momento; (iii) realizar a manutenção de estado dos recursos; (iv) manter o Controle de Recursos Ativos; (v) manter o Controle de Notificação de Recursos Disponíveis; (vi) receber a consulta dos Componente Cliente; (vii) receber e registrar recursos anunciados pelos Componentes Recursos no repositório de recursos ativos.

O *lease* é um intervalo de tempo que é gerenciado pelo Controlador de Recursos. O Componente Recurso precisa renovar o *lease* do recurso localizado no CD periodicamente, caso contrário será removido do Controle de Recursos Ativos (CRA).

5.2.2. Processador Semântico

Este módulo é responsável pelo processamento da consulta através tecnologias de Web Semântica. Os recursos serão previamente instanciados na ontologia através de uma interface de administração do ambiente. Através do motor de inferência é possível definir regras para aumentar a expressividade das consultas.

5.2.3. Comunicador P2P Intercelular

Através deste módulo o mecanismo de descoberta processa a consulta de clientes em outras células do ambiente utilizando protocolo P2P.

5.2.4. Localizador de Recursos Distribuídos

O Localizador de Recursos Distribuídos é responsável por receber e organizar as respostas das consultas realizadas em outras células, recebidas pelo módulo Comunicador P2P Inter celular.

5.3. Classificador de Recursos

Este módulo tem por finalidade realizar a classificação dos recursos mais adequados à solicitação do cliente através de critérios que estão sendo definidos.

6. Trabalhos Relacionados

No levantamento do estado da arte, foram considerados projetos atuais na área de descoberta de recursos. Dois trabalhos foram selecionados com objetivo comum que é a descoberta automática de recursos sem a intervenção humana, no contexto da IoT. As principais características são abordadas e posteriormente comparadas.

6.1. A Scalable and Self-Configuring Architecture for Service Discovery in the Internet of Things

Este trabalho propõe uma arquitetura baseada em P2P, auto-configurável, escalável para mecanismos de descoberta de recursos e serviços, sem intervenção humana [Cirani et al. 2014].

A arquitetura proposta é formada por gateways IoT que gerenciam nodos confinados e não confinados. O gateway realiza o processo de descoberta de recursos e serviços, proxy e controla o acesso às funcionalidades.

O protocolo Zeroconf foi utilizado para a descoberta de recursos dentro de redes locais. Ao ingressar em um ambiente inteligente dispositivos se anunciam através do zeroconf e o gateway percebe sua presença. Logo a seguir através do protocolo CoAP - Constrained Application Protocol (CoAP), é feita a consulta da descrição do dispositivo.

IoT Gateways são organizados como pares de uma sobreposição P2P estruturada, o que proporciona uma resolução de nomes eficiente para serviços CoAP. A arquitetura para descoberta de serviços em grande escala apresentada neste trabalho baseia-se em duas sobreposições P2P: (i) o serviço de localização distribuída (DLS) e (ii) a tabela geográfica distribuída (DGT). O DLS oferece um serviço de resolução de nomes para recuperar todas as informações necessárias para acessar um recurso identificado por um URI - Uniform Resource Identifier (URI). A DGT constrói um conhecimento geográfico distribuído, com base na localização dos nós, que pode ser usado para recuperar uma lista de recursos correspondentes critérios geográficos.

6.2. Web of Things: Automatic Publishing and Configuration of Devices

Propõe uma solução baseada em modelos para configuração e publicação automáticas na Web das Coisas. A solução proposta utiliza o protocolo Zeroconf para a descoberta e configuração de dispositivos que se anunciam ao se conectarem em uma rede local. No momento em que o dispositivo se conecta na rede, o mesmo é identificado, e uma aplicação é gerada e publicada em um barramento de serviços, de forma dinâmica e automatizada. Essa aplicação permitirá o acesso às funcionalidades do dispositivo na Web via protocolo HTTP [de Andrade Junior et al. 2014].

Foram implementados dois serviços: i) o serviço Detector que, através do Zero-conf, percebe a entrada de novos dispositivos na rede e coleta informações; ii) o serviço Generator, que gera a aplicação que disponibiliza as funcionalidades oferecidas pelo dispositivo, implantando-a no barramento de serviços.

7. Comparativo entre os Trabalhos Relacionados

Nesta seção é realizada uma comparação entre os trabalhos de descoberta de recursos apresentados. Foram selecionados os seguintes critérios para análise:

- Linguagem. Refere-se a linguagem utilizada na descrição e consulta por recursos;
- Expressividade. Se a proposta apresenta alguma tecnologia para inferir conhecimento;
- Escalabilidade. Se a proposta trata o grande aumento de recursos presentes no ambiente;
- Arquitetura. Refere-se como são organizadas as informações referentes aos recursos presentes no ambiente. Pode ser centralizada, descentralizada ou híbrida;
- Descoberta. Protocolos utilizados para a descoberta local de recursos;
- Invocação. Refere-se a forma como é acessado o recurso descoberto.

Tabela 1. Comparativo entre os Serviços de Descoberta

Critério	TR1*	TR2**	Proposto
Linguagem	Core Link	WADL	OWL
Expressividade	Não	Sim	Sim
Escalabilidade	P2P DHT	Não	P2P
Arquitetura	Híbrida	Centralizada	Híbrida
Descoberta	DNS-SD	DNS-SD	UPnP
Invocação	CoAP	HTTP	HTTP

*TR1 - A Scalable and Self-Configuring Architecture for Service Discovery in the Internet of Things

**TR2 - Web of Things: Automatic Publishing and Configuration of Devices

Observa-se que apenas um trabalho prevê a expressividade em sua arquitetura e o protocolo DNS-SD encontra-se presente nos dois trabalhos para realizar a descoberta sem intervenção humana na IoT. A escalabilidade é implementada através de tecnologias P2P, prevendo o crescimento do número de recursos e qualificando o processo de descoberta de recursos dispersos geograficamente.

8. Considerações Finais

O principal diferencial deste serviço quando confrontado aos modelos existentes é a expressividade na representação e consulta de recursos e sua integração ao EXEHDA, expandindo a capacidade do serviço de descoberta de recursos localizar dispositivos confinados e limitados.

O serviço de descoberta proposto possui uma linguagem de descrição e consulta com alta expressividade através da linguagem OWL, prevê a escalabilidade entre células do middleware EXEHDA através do protocolo P2P, possui uma arquitetura híbrida, sendo o processo de descoberta centralizado em cada célula, a descoberta em redes confinadas é

realizada através do protocolo UPnP configurado nos gateways do ambiente e a invocação do recurso é feito através do protocolo HTTP.

O trabalho encontra-se em fase inicial e em sua continuidade será revisto o modelo ontológico utilizado, restringindo seu domínio para ser utilizado apenas pelo módulo processador semântico. Os demais módulos da arquitetura utilizarão banco de dados tradicional, diminuindo o tempo necessário para o processamento das regras de inferência.

A validação do serviço se dará através de um ambiente de teste, com sensores e atuadores reais e simulados. Neste ambiente será possível comprovar o aumento da expressividade com a utilização de tecnologias de Web Semântica, bem como a localização de dispositivos confinados através de gateways. O refinamento da resposta será realizado através de critérios e algoritmos para seleção dos recursos mais adequados entre os presentes no resultado do *matching*.

Referências

- Charith, P., Arkady, Z., Peter, C., and Dimitrios, G. (2014). Context aware computing for the internet of things: A survey.
- Cirani, S., Davoli, L., Ferrari, G., Leone, R., Medagliani, P., Picone, M., and Veltri, L. (2014). A Scalable and Self-Configuring Architecture for Service Discovery in the Internet of Things. *IEEE Internet of Things Journal*, 1(5):508–521.
- Davet, P. T. (2016). EXEHDA-IoT: Uma Contribuição à Arquitetura do Middleware EXEHDA Direcionada à Internet das Coisas. Dissertação de mestrado, Programa de Pós-Graduação em Computação - Universidade Federal de Pelotas.
- de Andrade Junior, N. V., Bastos, D. B., and Prazeres, C. V. S. (2014). Web of things: Automatic publishing and configuration of devices. In *Proceedings of the 20th Brazilian Symposium on Multimedia and the Web, WebMedia '14*, pages 67–74, New York, NY, USA. ACM.
- Edwards, W. K. (2006). Discovery Systems in Ubiquitous Computing Ubiquitous. *Pervasive Computing*, 5(2):70–77.
- Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7):1645–1660.
- Klusch, M. (2014). Service discovery. In *Encyclopedia of Social Network Analysis and Mining*, pages 1707–1717.
- Lopes, J., Souza, R., Geyer, C., Costa, C., Barbosa, J., Pernas, A., and Yamin, A. (2014). A Middleware Architecture for Dynamic Adaptation in Ubiquitous Computing. *J-Jucs*, 20(9):1327–1351.
- Marin-Perianu, R., Hartel, P., and Scholten, H. (2005). A classification of service discovery protocols. *Centre for Telematics and Information Technology, Univ. of Twente, The Netherlands*, (June):05–25.
- Miorandi, D., Sicari, S., De Pellegrini, F., and Chlamtac, I. (2012). Internet of things: Vision, applications and research challenges.