

# EXEHDA-IoT: Uma Abordagem Consciente de Contexto Direcionada à Internet das Coisas

Patrícia Davet<sup>1</sup>, Huberto Kaiser Filho<sup>1</sup>, Leonardo João<sup>1</sup>, Lucas Xavier<sup>1</sup>, Tainã Carvalho<sup>1</sup>,  
Rodrigo Souza<sup>2</sup>, João Lopes<sup>2</sup>, Adenauer Yamin<sup>1</sup>

<sup>1</sup>Centro de Desenvolvimento Tecnológico (CDTec) – Universidade Federal de Pelotas (UFPel)  
Caixa Postal 354 – 96010-900 – Pelotas – RS – Brazil

<sup>2</sup>Instituto de Informática  
Universidade Federal do Rio Grande do Sul (UFRGS) – Porto Alegre, RS – Brazil

{ptdavet,hkaiser,lrsjoao,lmdsxavier,trcarvalho,adenauer}@inf.ufpel.edu.br,  
{rssouza,jlblopes}@inf.ufrgs.br

**Abstract.** *One of the major research challenges of Ubiquitous Computing is related to the need of the applications being aware of their context of interest, and when appropriate, respond to it. This paper presents an architecture for context awareness, called EXEHDA-IoT. We consider that the main contribution of this work is an IoT based architecture that supports the managing of the acquisition, storage, and processing of context data, in a distributed way, independently of the application, in an autonomic perspective and rule-based. To assess the functionality of the EXEHDA-IoT, we present a case study, highlighting the prototyping and tests performed.*

**Resumo.** *Um dos principais desafios de pesquisa da Computação Ubíqua está relacionado à necessidade das aplicações terem consciência do seu contexto de interesse, e quando for o caso, reagir ao mesmo. Este artigo apresenta uma arquitetura para consciência do contexto, denominada EXEHDA-IoT. Considera-se como principal contribuição deste trabalho a concepção de uma arquitetura direcionada à IoT para suporte ao gerenciamento da aquisição, armazenamento e processamento dos dados de contexto, de forma distribuída, independente das aplicações, em uma perspectiva autônoma baseada em regras. Para avaliar as funcionalidades do EXEHDA-IoT é apresentado um estudo de caso, destacando a prototipação e os testes realizados.*

## 1. Introdução

A Internet das Coisas, do inglês *Internet of Things* (IoT) vem se consolidando como o novo paradigma de evolução da Internet, na qual se preconiza a ideia do “tudo conectado”, ou seja, qualquer “coisa” (pessoa, animal ou objeto) pode se comunicar através da Internet, possuindo uma identificação única e sem a obrigatoriedade de uma intervenção humana. Esta visão da IoT promove a integração do mundo físico ao digital, de forma a criar uma rede de objetos inteligentes incorporados ao ambiente de forma ubíqua [Perera et al. 2013].

Esses objetos inteligentes possuem dispositivos embarcados com capacidade para armazenar e processar os dados sensorados (captados do mundo físico por sensores), que

interconectam-se com outros dispositivos e recursos (físicos ou virtuais), o que permite o surgimento de uma miríade de aplicações.

Aplicações de IoT, enquanto ubíquas e assim dotadas de comportamento autônomo, e com uma organização distribuída, necessitam ter consciência dos dados contextuais que lhe interessam e, quando for o caso, se adaptar aos mesmos. Esta classe de sistemas computacionais, reativos ao contexto, abre perspectivas ao desenvolvimento de aplicações mais ricas, elaboradas e complexas, e que exploram a natureza dinâmica das modernas infraestruturas computacionais, bem como a mobilidade do usuário [Caceres and Friday 2011].

No entanto, para a implementação de aplicações conscientes de contexto em cenários introduzidos pela IoT, uma série de desafios devem ser superados para a concepção de infraestruturas que atendam à demandas destes novos cenários. Um dos principais desafios está relacionado com a alta heterogeneidade decorrente da grande diversidade de tecnologias de *hardware* e *software* presentes neste ambiente, o que faz com que haja uma busca de soluções que permitam a interoperabilidade e integração destes diferentes componentes. Uma alternativa de solução promissora para tais desafios, como o da heterogeneidade, está na utilização de plataformas de *middleware* [Maia et al. 2015].

Este trabalho, denominado EXEHDA-IoT tem como objetivo principal contribuir com o Subsistema de Reconhecimento de Contexto e Adaptação do *middleware* EXEHDA (*Execution Environment for Highly Distributed Applications*) [Lopes et al. 2014] com uma abordagem que o capacite para atendimento das demandas IoT. A expectativa é obter uma arquitetura de *software* distribuída e extensível, comprometida com a premissa de fornecer mecanismos para o desenvolvimento de aplicações conscientes de contexto na IoT.

O artigo está organizado da seguinte forma: a Seção 2 caracteriza o *middleware* EXEHDA. A Seção 3 descreve a modelagem da arquitetura de software proposta. A Seção 4 apresenta um estudo de caso. A Seção 5 discute os trabalhos relacionados. Por fim, a Seção 6 apresenta as considerações finais.

## 2. Middleware EXEHDA

O EXEHDA é um *middleware* baseado em serviços que visa criar e gerenciar um ambiente ubíquo, bem como promover a execução de aplicações sobre esse ambiente. No EXEHDA, as condições de contexto são pró-ativamente monitoradas e o suporte à execução deve permitir que tanto a aplicação como ele próprio utilizem essas informações na gerência da adaptação de seus aspectos funcionais e não-funcionais. O mecanismo de adaptação proposto para o EXEHDA emprega uma estratégia colaborativa entre aplicação e ambiente de execução, através da qual é facultado ao programador individualizar políticas de adaptação para reger o comportamento de cada um dos componentes que constituem o software da aplicação [Lopes et al. 2014].

A estrutura de software do EXEHDA contempla um núcleo mínimo e serviços carregados sob demanda, os quais estão organizados nos seguintes subsistemas: (i) Reconhecimento de Contexto e Adaptação, (ii) Acesso Ubíquo, (iii) Execução Distribuída e (iv) Comunicação.

O EXEHDA-IoT contribui especificamente com o Subsistema de Reconheci-

mento de Contexto e Adaptação, ampliando as funcionalidades dos serviços que tratam a extração da informação direta dos sensores, passando pela identificação em alto nível dos elementos de contexto, até o disparo das ações de adaptação em reação a modificações no estado de tais elementos de contexto.

### 3. EXEHDA-IoT: Visão e Modelagem da Arquitetura Direcionada à IoT

Dentre os desafios inerentes a cenários IoT para aplicações conscientes de contexto destacam-se: (i) a aquisição do contexto a partir de um grande número de dispositivos heterogêneos e distribuídos; (ii) o processamento das informações de contexto adquiridas e a respectiva atuação sobre o meio físico; e (iii) a disponibilização dos dados contextuais processados aos usuários de forma distribuída e no momento oportuno.

No EXEHDA-IoT os dispositivos computacionais do ambiente ubíquo são distribuídos em células, sendo cada célula 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 EXEHDA nodo que corresponde aos dispositivos computacionais responsáveis pela execução das aplicações; (iii) o EXEHDA nodo móvel, um subcaso do anterior, que corresponde aos dispositivos tipicamente móveis que podem se deslocar entre as células do ambiente ubíquo, como *notebooks*, *tablets* ou *smartphones*; e (iv) o EXEHDAborda que consiste no elemento de borda do ambiente ubíquo, responsável por fazer a interoperação entre os serviços do *middleware* e as redes de sensores e atuadores.

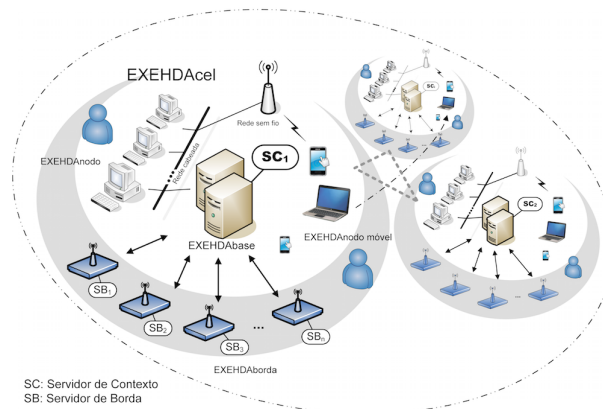


Figura 1. Organização Celular do Ambiente Ubíquo Gerenciado pelo EXEHDA-IoT

Para provimento de consciência de contexto no ambiente ubíquo, o EXEHDA-IoT (vide Figura 1) se vale de dois tipos principais de servidores: o Servidor de Borda, responsável pela interação com o meio através de sensores, atuadores ou *gateways*, e o Servidor de Contexto que atua no armazenamento e processamento das informações contextuais. A arquitetura proposta provê comunicação (i) entre os Servidores de Borda e o Servidor de Contexto; (ii) entre os Servidores de Contexto localizados em diferentes células do ambiente ubíquo; e (iii) com outros serviços do *middleware* ou aplicações. Os principais módulos dos Servidores de Contexto e Borda estão descritos a seguir.

### 3.1. Servidor de Contexto

Uma visão do Servidor de Contexto e sua relação com o Servidor de Borda pode ser vista na Figura 2. O Servidor de Contexto está organizado em seis módulos autônomos descritos a seguir, os quais interoperam no provimento das funcionalidades necessárias ao Serviço de Consciência de Contexto do EXEHDA.

**Coletor:** provê suporte à captura das informações contextuais, coletadas pelos Servidores de Borda, considerando sensores lógicos (software) e/ou hardware.

**Atuador:** controla os atuadores, após ser notificado pelos outros módulos do Servidor de Contexto. Dispara no ambiente ubíquo ações que mudam o estado do meio, viabilizando o uso de serviços de consciência de contexto em aplicações de controle e automação.

**Interpretador:** realiza tarefas de manipulação e dedução das informações contextuais, utilizando para isto informações especificadas nos **Contextos de Interesse** das aplicações. Este módulo mantém um **Repositório de Contexto**, onde são armazenadas as informações contextuais obtidas pelo Coletor, provendo a possibilidade de registro histórico dos contextos, o que permite a construção de regras que explorem aspectos temporais. Esses dados são utilizados pelas regras do componente **Tratador de Regras**, o qual dispara as ações pertinentes em função do estado contextual. A natureza das regras - tratamento lógico, numérico ou temporal - é uma decorrência do tipo de domínio da aplicação atendida pelo Serviço de Consciência de Contexto do EXEHDA.

**Notificador:** notifica o resultado do processamento contextual realizado pelo Interpretador. Recebe subscrições de todos os serviços e/ou aplicações que desejem notificações a respeito dos estados contextuais, interoperando através do módulo Gerenciador de Comunicação. Passa pelo Notificador todas as decisões de atuação provenientes do tratamento autônomo de regras de processamento contextual.

**Gerenciador de Comunicação:** empregado por Servidores de Contexto remotos e/ou aplicações quando da solicitação de dados contextuais e/ou o disparo de atuadores. Esse módulo provê a disseminação de informações para outros serviços do *middleware*, bem como o envio de mensagens aos usuários.

**Gerenciador de Configuração:** permite ao usuário um gerenciamento confortável das configurações do Servidor de Contexto. O mesmo provê facilidades para que sejam especificados os diferentes aspectos dos sensores e atuadores, bem como informações dos equipamentos cujo contexto está sendo adquirido.

**Gerenciador de Acesso Móvel:** provê acesso móvel ao EXEHDA-IoT, possibilitando a exibição de informações contextuais e a disponibilização de alertas proativos. Particularmente, a disponibilização de alertas proativos em uma plataforma de hardware que possa acompanhar o usuário enquanto este desempenha suas atividades nos mais diferentes lugares, se mostrou um procedimento que potencializa a ubiquidade da solução de consciência de contexto disponibilizada.

### 3.2. Servidor de Borda

A arquitetura do Servidor de Borda contempla três módulos principais (vide Figura 2), os quais são destinados a: (i) tratar a rede de sensores; (ii) efetuar as publicações; e (iii) tratar a rede de atuadores. A seguir é realizada uma discussão sobre estes módulos.

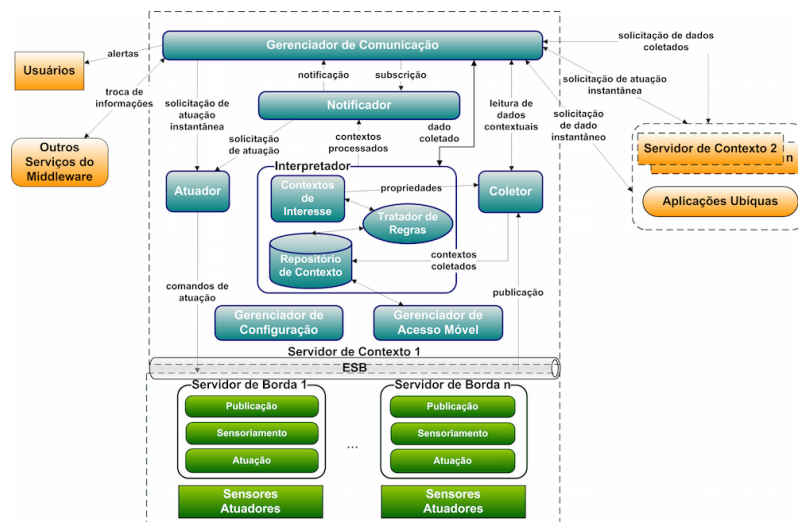


Figura 2. Arquitetura EXEHDA-IoT

O **Módulo de Sensoriamento** provê o tratamento de uma rede de sensores permitindo uma individualização de processamento por sensor. Este tratamento é responsável por aspectos desde gerência física (interfaces, frequência de leitura) até normalização computacional (validação, tradução) dos valores aquisitados. Também, este módulo provê funcionalidades para publicação das informações coletadas da rede de sensores no Servidor de Contexto.

O **Módulo de Publicação** é responsável por coordenar o principal fluxo de dados entre os Servidores de Borda e o Servidor de Contexto, promovendo a publicação de todos os dados coletados e garantindo uma **Persistência Local** dos mesmos nos períodos que a publicação ficar inviabilizada.

O **Módulo de Atuação** é responsável pelo gerenciamento dos atuadores. Pode receber comandos de atuação originários tanto do Servidor de Contexto como consequência da execução de uma regra, como de uma aplicação controlada pelo usuário.

#### 4. Estudo de Caso

Nesta seção estão resumidos os principais aspectos do estudo de caso relacionado ao projeto AMPLUS<sup>1</sup>, empregado na avaliação das funcionalidades do EXEHDA-IoT. O estudo de caso contemplou tarefas referentes ao sensoriamento e a coleta de dados contextuais, bem como o processamento e notificação das informações de contexto. Neste estudo de caso foi desenvolvida uma aplicação para interface Web, e outra para dispositivos móveis.

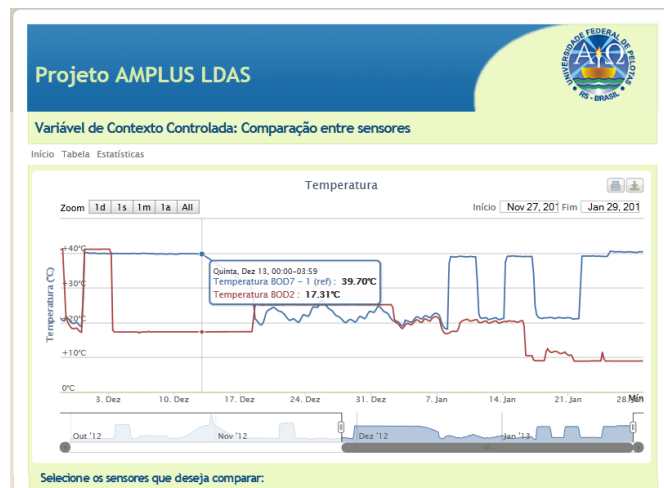
O Projeto AMPLUS foi concebido para prover serviços de consciência do contexto, permitindo um registro dos estados contextuais em que se encontram os equipamentos do Laboratório Didático de Análise de Sementes - LDAS<sup>2</sup>, ao longo de todo o tempo de realização dos vários testes, e uma atuação proativa quando necessário.

<sup>1</sup>AMPLUS: <http://amplus.ufpel.edu.br/>

<sup>2</sup>LDAS: <http://amplus.ufpel.edu.br/ldas>

Para prototipação dos mecanismos de coleta, armazenamento e processamento do contexto, bem como atuação junto ao meio foram priorizadas tecnologias escaláveis e robustas. Nesse sentido, o código dos servidores de contexto e de borda está escrito na linguagem Python, sendo empregado XML-RPC<sup>3</sup> (*Extensible Markup Language - Remote Procedure Call*). O repositório de contextos emprega o gerenciador PostgreSQL para implementação das bases de dados. Os sensores e atuadores são acessados pelo protocolo 1-Wire<sup>4</sup>.

A interface Web possibilita a seleção do contexto de interesse a ser exibido, disponibilizando relatórios textuais e gráficos. O relatório gráfico (vide Figura 3) oferecido pelo sistema permite a visualização simultânea das informações de vários sensores. A seleção dos sensores é feita a partir de um menu com suporte a múltipla seleção. Também é disponibilizado um recurso de inspeção que permite a comparação dos valores em um determinado instante do tempo.



**Figura 3. Projeto AMPLUS - Relatório gráfico**

Ainda, com o intuito de promover a proatividade do Projeto AMPLUS junto a comunidade usuária, foram desenvolvidas interfaces para dois serviços públicos de comunicação: e-mail e SMS para rede celular. Estas mensagens são produzidas a partir do processamento das regras contextuais de forma autônoma pelo Servidor de Contexto.

Para atender o fato da rotina de trabalho dos laboratoristas do LDAS implicar em uma mobilidade nos diversos recintos do laboratório, foi desenvolvida uma interface de alerta visual, a qual é ativada sempre que um dispositivo estiver em um estado contextual que exija atenção. A partir deste alerta, detalhes podem ser inferidos através da interface computacional do Projeto AMPLUS.

A interface para acesso móvel foi concebida para a plataforma Android. Os relatórios gráficos e textuais oferecem a opção do usuário especificar intervalo de

<sup>3</sup>XML-RPC: <http://www.xmlrpc.com>

<sup>4</sup>1-Wire: <http://ubiq.inf.ufpel.edu.br/1-wire/>

visualização (hora, dia, semana), sendo o ajuste do eixo vertical feito de forma automática, minimizando o emprego de rolagem de tela. Para exibição dos alertas foi explorada a interface disponibilizada pela plataforma Android para esta finalidade, este aspecto potencializa a integração do mecanismo de alertas às funcionalidades do *smartphone* do usuário. As interfaces correspondentes a estas funcionalidades são exibidas na Figura 4.



Figura 4. Interfaces da aplicação móvel

## 5. Trabalhos Relacionados

O estudo dos trabalhos relacionados CARE [Agostini et al. 2009], CoCA [Ejigu et al. 2008], HiCon [Cho et al. 2008], Solar [Chen et al. 2008], WComp [Tigli et al. 2009] foi desenvolvido considerando as premissas de concepção do EXEHDA-IoT: (1) arquitetura (distribuída ou centralizada); (2) sensoriamento (suporte a redes de sensores); (3) aquisição dos dados de contexto; (4) suporte ao tratamento de regras e; (5) suporte à atuação sobre o meio.

As arquiteturas estudadas não mantêm um caráter descentralizado para todas as etapas de tratamento dos dados de contexto, o que não é oportuno para o requisito de distribuição em larga escala dos ambientes ubíquos. Por sua vez, o modelo arquitetural do EXEHDA-IoT diferencia-se dos trabalhos relacionados por estar estruturado de forma distribuída, em todas as etapas de tratamento das informações de contexto, desde a aquisição até os procedimentos de atuação sobre o meio.

O EXEHDA-IoT pode gerenciar redes de sensores e atuadores, tal característica é encontrada em parte nos projetos CoCA e HiCon, que têm suporte a redes de sensores. O projeto WComp, por sua vez, permite atuação sobre o meio, entretanto, não suporta o gerenciamento de redes de atuadores.

Com exceção dos projetos CARE e Solar, os demais prevêm o emprego de mecanismos específicos para aquisição do contexto, que adotam uma estratégia de separação entre a obtenção e o uso do contexto. Além de contemplar esse aspecto, o EXEHDA-IoT apresenta um diferencial em relação aos projetos relacionados, que é o emprego de um caráter autônomo na aquisição dos dados de contexto, visto que estes continuam a ser obtidos pelo mecanismo, mesmo que as aplicações interessadas em seu uso estejam inoperantes.

A maioria dos projetos estudados possui suporte ao tratamento de regras, porém esta funcionalidade usualmente está restrita a algumas etapas do processamento do con-

texto. O EXEHDA-IoT, diferencia-se destes trabalhos, por sua arquitetura de software dar suporte ao tratamento distribuído de regras, operando tanto nos Servidores de Borda, como nos Servidores de Contexto.

## 6. Considerações Finais

O principal diferencial do EXEHDA-IoT em relação aos trabalhos relacionados diz respeito a possibilidade de gerenciar a aquisição, armazenamento e processamento dos dados de contexto, de forma distribuída, independente das aplicações, em uma perspectiva autônoma baseada em regras. No que tange à interoperabilidade entre os servidores que compõem o Serviço de Consciência de Contexto, o EXEHDA-IoT contempla uma abordagem compatível com a expectativa de operação, utilizando protocolos padrões da Internet para as comunicações.

Como trabalhos futuros, os seguintes aspectos foram priorizados: (i) explorar estudos de caso em que as regras de processamento contextual utilizem outros mecanismos de inferência de mais alto nível; e (ii) utilizar a arquitetura do EXEHDA-IoT para provimento de consciência de situação.

## Referências

- Agostini, A., Bettini, C., and Riboni, D. (2009). Hybrid reasoning in the CARE middleware for context awareness. *International Journal of Web Engineering and Technology*, 5(1):3.
- Caceres, R. and Friday, A. (2011). Ubicomp systems at 20: Progress, opportunities, and challenges. *IEEE Pervasive Computing*, 11(1):14–21.
- Chen, G., Li, M., and Kotz, D. (2008). Data-centric middleware for context-aware pervasive computing. *Pervasive and Mobile Computing*, 4(2):216–253.
- Cho, K., Hwang, I., Kang, S., Kim, B., and Lee, J. (2008). HiCon: a hierarchical context monitoring and composition framework for next-generation context-aware services. *Network, ...*, 22(4):34–42.
- Ejigu, D., Scuturici, M., and Brunie, L. (2008). Hybrid Approach to Collaborative Context-Aware Service Platform for Pervasive Computing. *Journal of Computers*, 3(1):40–50.
- 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. *JUCS*, 20(9):1327–1351.
- Maia, P., Baffa, A., Cavalcante, E., Delicato, F. C., Batista, T., and Pires, P. F. (2015). Uma plataforma de middleware para integração de dispositivos e desenvolvimento de aplicações em e-health. *Anais do XXXIII SBRC*, pages 361–374.
- Perera, C., Zaslavsky, A., Christen, P., and Georgakopoulos, D. (2013). Context aware computing for the internet of things: A survey. *Communications Surveys & Tutorials, IEEE*, 16(1):414–454.
- Tigli, J.-Y., Lavirotte, S., Rey, G., Hourdin, V., Cheung-Foo-Wo, D., Callegari, E., and Riveill, M. (2009). WComp middleware for ubiquitous computing: Aspects and composite event-based Web services. *annals of telecommunications - annales des télécommunications*, 64(3-4):197–214.