

uMED – Uma Proposta Baseada em Processamento Semântico para a Medicina Ubíqua

Sérgio Luis Rodrigues¹, Luthiano Rodrigues Venecian²

^{1,2} PPGINF – Centro Politécnico – Universidade Católica de Pelotas (UCPEL)

Resumo - Este artigo apresenta o uMED, um *framework* concebido para as demandas da Medicina Ubíqua, utilizando tecnologias de processamento semântico, nas tarefas de aquisição, processamento e distribuições das informações contextuais. Para avaliação do uMED, foi desenvolvido um estudo de caso direcionado à área médica, no qual, foram monitorados dados vitais de pacientes e empregados níveis de alertas aos principais atores (médicos, paramédicos, entre outros), produzidos por regras de dedução semântica definidas para a aplicação.

Palavras-chave: Medicina Ubíqua, Processamento Semântico, Ontologia

Abstract - This article presents the uMED, a framework designed for the demands of Medicine Ubiquitous using semantic processing technology, the tasks of acquisition, processing and distribution of contextual information. To assess uMED, we developed a case study directed to the medical area, which were monitored vital signs of patients and staff levels of alerts to main actors (doctors, paramedics, among others), produced by deduction rules for defined semantics application.

Keywords: Ubiquitous Medicine, Semantic Processing, Ontology

Introdução

Aplicações ubíquas executam em ambientes instrumentados com sensores, geralmente dotados de interfaces de redes sem fio, nos quais dispositivos, agentes de software e serviços são integrados e cooperam para atender os objetivos os seus usuários. Essa categoria de aplicações caracteriza-se por constantes mudanças em seu contexto de execução, geradas pelos ambientes altamente dinâmicos em que tipicamente executam.

Nesta perspectiva introduzida pela computação ubíqua (3), o processamento está distribuído no ambiente através de vários dispositivos, que executam tarefas bem definidas dependendo de sua natureza, interligados de forma que a infraestrutura de comunicações torne-se o mais imperceptível possível para o usuário.

Deste modo, sensibilidade ao contexto é um paradigma computacional que se propõe a permitir que as aplicações tenham acesso e tirem proveito de informações contextuais que digam respeito às computações que realizam.

Este trabalho contempla esforços de pesquisa na área de Medicina Ubíqua. Esta área tem sido potencializada pelos avanços das tecnologias móveis e sem fio, os quais

somados a popularização dos dispositivos portáteis, dentre eles *Smartphones*, PDAs e dispositivos médicos, como *holters*, potencializam a tarefa de monitoramento de pacientes (9).

A infraestrutura de software pretendida visa o gerenciamento das informações contextuais, coletando informações de diferentes naturezas do ambiente, analisando essas informações como variáveis independentes, ou combinadas. Por sua vez, aplicações ubíquas são caracterizadas por apresentarem contextos altamente dinâmicos e heterogêneos, com um grande grau de mobilidade dos seus principais atores (médicos, pacientes, paramédicos, etc.). Este trabalho tem como premissa principal propor a integração de tecnologias de processamento semântico nas tarefas de aquisição, processamento e distribuição das informações contextuais.

Neste sentido, está sendo concebido o *framework* uMED, direcionado as demandas da Medicina Ubíqua, que deverá ser integrado ao mecanismo de sensibilidade ao contexto denominado EXEHDA-SS (11). O processamento semântico previsto irá empregar uma ontologia denominada OntContext.

1

Métodos

O *framework* uMED está inserido nos esforços de pesquisa do G3PD (Grupo de Pesquisa em Processamento Paralelo e Distribuído) do Centro Politécnico da Universidade Católica de Pelotas e do Centro de Desenvolvimento Tecnológico da Universidade Federal de Pelotas, e conta com a colaboração de equipes médicas ligadas a essas universidades.

O EXEHDA-SS, que irá compor a infraestrutura do *framework* uMED, emprega o ambiente ubíquo o qual é definido no projeto ISAM (Infraestrutura de Suporte às Aplicações Móveis Distribuídas) (6) e provido pelo *middleware* EXEHDA (*Execution Environment for Highly Distributed Applications*) (13).

O uMED irá contemplar o uso de tecnologias de processamento semântico nas funcionalidades de: (i) aquisição, (ii) processamento e (iii) distribuição das informações contextuais direcionadas as aplicações ubíquas.

A arquitetura de software prevista contempla gerentes autônomos e cooperantes para a realização de tarefas de manipulação e dedução sobre o contexto.

Para a modelagem das informações contextuais processadas pelo mecanismo de sensibilidade ao contexto concebido, foi selecionado o uso de ontologias.

Na perspectiva da arquitetura, ontologias são usadas para representar o ambientes ubíquo, descrevendo, usualmente, entidades envolvidas e suas respectivas propriedades. Neste sentido, elas definem principalmente os diferentes tipos de aplicações, serviços e dispositivos. Além disso, ontologias também podem ser utilizadas para descrições de padrões para informações médicas por exemplo: temperatura corporal, pressão arterial, dentre outras (12).

As ontologias são empregadas no uMED sob duas perspectivas: (i) representação semântica dos dados contextuais e (ii) o estabelecimento de relações entre os mesmos, possibilitando a realização de inferências. Através do uso de inferências espera-se contribuir para qualificação das informações contextuais adquiridas e notificadas pelo uMED.

A linguagem OWL-DL foi adotada para a construção do modelo ontológico da OntContext (11), a qual fornece suporte a metadados RDF, abstrações de classes, generalização, agregação, entre outros (12).

O núcleo do modelo semântico para representação de contexto é a OntContext, vide figura 1, ontologia definida para ser responsável pela representação dos contextos coletados, notificados e das instâncias dos Contextos de Interesse do *framework* uMED.

A OntContext contempla a seguinte estrutura de classes e sub-classes:

- Classe 'Sensor', a qual contém as instâncias dos sensores que participam do sensoramento do ambiente ubíquo;
- Classe 'Contexto', armazena os dados coletados pelos sensores no RIC – Repositório de Informações Contextuais, localizado na arquitetura de software do *framework* uMED, vide figura 2;
- Classe 'Contexto_Notificado', persiste os contextos notificados, processados e deduzidos pelo Motor de Inferência da arquitetura de software;
- Sub-classes 'ContextoNotificado_Sensor' e 'ContextoNotificado_Deduzido', no qual registram seus dados no RCN – Repositório de Contextos Notificados;
- Classes 'Contexto_Interesse', 'Sensor_Public' e 'Contexto_Deduzido', ficam instanciados as aplicações, os parâmetros operacionais de publicação dos sensores; e as regras de dedução definidas pelo desenvolvedor para processamento no Motor de Inferência.

A arquitetura de software proposta para o uMED é alimentada por contextos de interesses, os quais são responsáveis por caracterizar os aspectos que devem ser considerados nos procedimentos de monitoração do ambiente ubíquo, de interpretação dos dados capturados e das respectivas notificações.

dinamicamente por mecanismo de sensoramento de sinais vitais (pressão alta e frequência cardíaca); (ii) emitir, de forma automatizada, diferentes níveis de alertas, em função dos dados sensoreados, para os agentes de saúde (médicos, enfermeiros); e (iii) permitir acesso ubíquo ao histórico dos dados sensoreados dos pacientes por agentes de saúde.

O desenvolvedor ao configurar a aplicação no modelo ontológico da “OntContext”, necessita habilitar seus contextos de interesse, especificando (i) os parâmetros operacionais e as regras de tradução dos sensores de monitoramento de pacientes (pressão alta e frequência cardíaca), e (ii) as regras de dedução, para a funcionalidade de Envio Automático de Mensagens (SMS, email, entre outros), o qual, os atores interessados, devem se registrar, para que possam receber notificações produzidas pelo processamento desta regra.

Sendo assim, no estudo de caso realizado foi definido dois parâmetros operacionais para publicação dos sensores de monitoramento de pacientes. O primeiro relacionado ao intervalo de medição: pressão alta a cada duas horas e frequência cardíaca a cada cinco minutos. E o segundo a faixa de flutuação mínima: pressão alta um percentual e frequência cardíaca cinco percentuais.

As regras de tradução e dedução empregadas utilizam uma sintaxe de construção adotada pelo subsistema de inferência da API JENA (7), baseado no raciocinador *Generic Rule Reasoner*, o qual permite a criação de regras definidas pelo desenvolvedor da aplicação.

Deste modo, foram definidas regras de tradução para caracterizar que a informação sensorada apresenta sinais normais ou não. Portanto, pressão alta superior a 180090 (180 por 90 mmHg) gera um valor traduzido para 1, caso inferior para 0. Frequência cardíaca superior a 100 (100 bpm), é traduzido para 1 ou inferior, para 0. Vide figura 3 e 4, respectivamente.

Para o Envio Automático de Mensagens definiu-se uma regra de dedução para risco de infarto, onde foram considerados os seguintes parâmetros: sensor de frequência cardíaca com batimentos superiores a 180 (180 bpm), e pressão arterial superior a 240100 (240 por 100 mmHg), conforme apresentado na figura 5.

```
[PA_ForaFaixa: (?c rdf:type ont:Contexto)
(?c ont:Contexto_Sensor 102)
(?c ont:Contexto_Valor ?cv)
(greater (?cv,180090))
-> (?n rdf:type ont 1)]
[PA_DentroFaixa: (?c rdf:type ont: Contexto)
(?c ont:Contexto_Sensor 102)
(?c ont:Contexto_Valor ?cv)
(lessThan(?cv,180090))
-> (?n rdf:type ont0)]
```

Figura 3 – Regra de Tradução do Sensor de Pressão Alta

```
[FC_ForaFaixa: (?c rdf:type ont: Contexto)
(?c ont:Contexto_Sensor 100)
(?c ont:Contexto_Valor ?cv)
(greaterTham (?cv,100))
-> (?n rdf:type ont 1)]
[FC_DentroFaixa: (?crdf:type ont: Contexto)
(?c ont:Contexto_Sensor 100)
(?c ont:Contexto_Valor ?cv)
(lessThan (?cv,101))
-> (?n rdf:type ont 0)]
```

Figura 4 – Regra de Tradução do Sensor de Frequência Cardíaca

```
[FC: (?c rdf:type ont: Contexto)
(?c ont:Contexto_Sensor 100)
(?c ont:Contexto_Valor ?c)
(greaterThan (?cv,180))
-> (?c rdf:type ont: FCsup)]
[PA: (?c rdf:type ont: Contexto)
(?c ont:Contexto_Sensor 102)
(?c ont:Contexto_valor ?cv)
(greaterThan (?cv,240100))
-> (?c rdf:type ont PASup)]
[RI: (?c rdf:type ont:FCsup)
(?c rdf:type ont:PASup)
-> (?c rdf:type ont Risco_infarto)]
```

Figura 5 – Regra de Dedução para Risco de Infarto

Resultados

Nesta seção são apresentados os resultados da AUP. Os sensores de monitoramento de pacientes definidos para a AUP ao serem disparados, processam os parâmetros operacionais para publicação das informações coletadas.

Esta publicação ocorre através da conexão periódica dos sensores ao Gerente de Aquisição, permitindo a instanciação das informações contextuais, produzidas pelos

4

mesmos, no RIC. Desta forma, ao armazenar esses valores coletados, o Gerente de Aquisição também processa as regras de tradução.

O Motor de Inferência ao ser acionado realiza uma leitura dos dados dos demais sensores especificados para o monitoramento de pacientes da AUP e os instancia no RCN.

A associação entre os dados coletados e os níveis de alertas em função do sensoramento são resumidos a seguir:

- Nível de Alerta Verde: sinais normais;
- Nível de Alerta Laranja: início de problema, frequência cardíaca ou pressão arterial fora do normal;
- Nível de Alerta Vermelho: alerta máximo, frequência cardíaca e pressão arterial fora do normal.

A figura 6, apresenta o nível de alerta vermelho notificado pelo *framework* uMED a aplicação AUP.

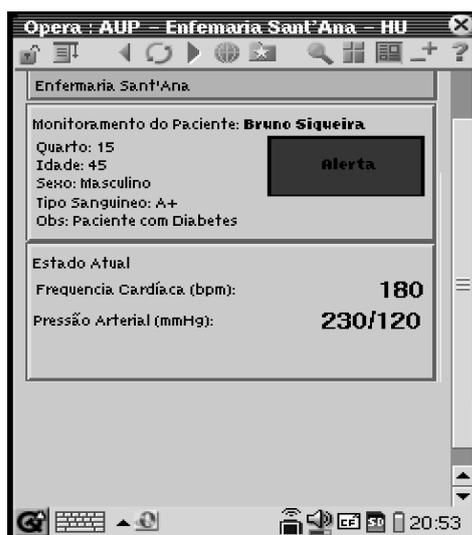


Figura 6 – Nível de Alerta Vermelho Notificado pelo *Framework* uMED

Discussões

Foram avaliadas no total onze trabalhos relacionados, dos quais seis são os mais relevantes, sendo estes: (A) Service-Oriented Context-Aware Middleware (4); (B) Context Broker Architecture (2); (C) Mobile Collaboration Architecture (10); (D) *Framework*

de Contexto (5); (E) Semantic Context Kernel (1); e (F) Infraware (8).

Em linhas gerais, este estudo além de prover uma aproximação com as tecnologias empregadas na concepção do uMED, conforme apresentado na seção Métodos, possibilitou uma sistematização das principais características a serem consideradas quanto da concepção dos componentes da arquitetura. A sistematização teve por base os aspectos considerados na tabela 1.

Tabela 1 – Comparação entre o uMED e os Trabalhos Relacionados

	A	B	C	D	E	F	uMED
Aquisição	X	X	X	X	X	X	X
Interpretação			X		X	X	X
Mecanismo de Dedução	X	X			X		X
Notificação	X		X	X		X	X
Ontologias	X	X	X	X	X	X	X

O emprego de ontologias permitiu ao uMED, realizar o processamento de regras de dedução dos dados contextuais, sendo que três dos trabalhos avaliados na literatura: (A), (B) e (E) possuem um mecanismo de dedução em suas arquiteturas, porém nenhum contempla a definição de regras externas à aplicação pelo desenvolvedor como é realizado pelo *framework* uMED, portanto, a arquitetura proposta, se mostra diferencial as demais, pois em tempo de execução, é possível que o desenvolvedor possa definir novas regras de dedução a serem processadas pelo seu Motor de Inferência.

Outro aspecto a ser considerado é a seleção da API *Jena*, a qual é decorrente de sua significativa presença na literatura especializada, sendo também empregada por outros trabalhos: (A), (B), (D) e (E), entretanto, nestes trabalhos a abordagem utiliza a máquina de inferência *OWLReasoner*, enquanto que no *framework* uMED, além de empregar esta máquina, usa também o raciocinador *Generic Rule Reasoner*, o qual não foi encontrado em nenhum dos trabalhos avaliados.

Conclusão

Na computação ubíqua um aspecto fundamental relaciona-se ao monitoramento e a manipulação das informações contextuais.

A utilização de tecnologias para processamento semântico no tratamento de informações de contexto traz como características: (a) a descrição formal, padrão e estruturada de cada dimensão semântica de informação de contexto; (b) o suporte à interoperabilidade sintática, estrutural e, principalmente, semântica entre aplicações sensíveis ao contexto; e (c) a capacidade de interpretar e inferir inter-relacionamentos com base nos conteúdos e descrições semânticas das entidades envolvidas.

Considerando estas premissas, o *framework* uMED, empregou um modelo de representação ontológico das informações de contexto, contemplando em sua arquitetura, o tratamento de eventos produzidos por contextos, tendo a mesma sido modelada para ser expansível, tanto no que diz respeito a captura de dados do ambiente ubíquo, bem como, quanto aos possíveis consumidores de contextos de interesse. Os consumidores podem ser as aplicações como a AUP, avaliada neste artigo, nas suas tarefas de produção de alertas e disparo automático de mensagens através do processamento de regras de dedução.

Referências

1. Bulcao Neto, R.F., Pimentel, M.G.C.: Toward a Domain-Independent Semantic Model for Context-Aware Computing (2005). In: Proceedings of the 3rdIW3C2 Latin American Web Congress, IEEE Computer Society.
2. Chen, H., Finin, T., Joshi, A., Peng, Y.: Umbc eBiquity Project: Context Broker Architecture, disponível em <http://ebiquity.umbc.edu/project>. Acesso novembro de 2009.
3. da Costa, C.A., Yamin, A.C., Geyer, C.F.R.: Toward a General Software Infrastructure for Ubiquitous Computing (2008). IEEE Pervasive Computing 7(1), 64{73}.
4. Gu, T., Wang, X.H., Pung, H.K., Zhang, D.Q.: An Ontology-based Context Model in Intelligent Environments (2005). In: Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference.
5. Henricksen, K., Indulska, J.: Developing Context-Aware Pervasive Computing Applications: Models and Approach (2007). In: Pervasive and Mobile Computing Journal.
6. ISAM: Infra-estrutura de Suporte as Aplicações Móveis (2007), disponível em: <https://saloon.inf.ufrgs.br/twiki/view/Projetos/ISAM>. Acesso maio de 2010.
7. JENA: Jena a Semantic Web Framework for Java (2009), disponível em: <http://jena.sourceforge.net/>. Acesso maio de 2010.
8. Pereira Filho, J.G., Pessoa, R.M., Calvi, C.Z., Oliveira, N.Q., Carmo, R.R.M., Barbosa, A.C.P., Farias, C.R.G., Leite, M.M.: Infraware: um Middleware de Suporte a Aplicações Móveis Sensíveis ao Contexto (2006). In: SBRC.
9. PERTMED: Sistema de Telemedicina Móvel (2009), disponível em: <http://gmobserver02.inf.ufsm.br:8082/features/projetos.html>. Acesso novembro de 2009.
10. Sacramento, V., Endler, M., Rubinsztein, H.K., Lima, L.S., Goncalves, K., Nascimento, F.N., Bueno, G.A.: Moca: A Middleware for Developing Collaborative Applications for Mobile Users (2004). IEEE Distributed Systems Online 5(10).
11. Venecian, L.R.: EXEHDA-SS: Um Mecanismo para Sensibilidade ao Contexto com Suporte Semântico (2010), disponível em <http://olaria.ucpel.tche.br/luthiano>. Acesso abril de 2010.
12. W3C: OWL Web Ontology Language (2010), disponível em <http://www.w3.org/TR/owl-guide/>. Acesso fevereiro de 2010.
13. Yamin, A.: Arquitetura para um Ambiente de Grade Computacional Direcionado as Aplicações Distribuídas, Móveis e Conscientes do Contexto da Computação Pervasiva (2004), UFRGS, Porto Alegre, RS.