

Uma Abordagem Baseada em Web Semântica para o Processamento das Informações de Contexto na Computação Pervasiva

João Lopes, Luiz Palazzo, Adenauer Yamin

Universidade Católica de Pelotas, Programa de Pós-Graduação em Informática,
Pelotas, Brasil
{joaolopes, lpalazzo, adenauer}@ucpel.tche.br

Iara Augustin

Universidade Federal de Santa Maria, Programa de Pós-Graduação em Informática,
Santa Maria, Brasil
august@inf.ufsm.br

Claudio Geyer

Universidade Federal do Rio Grande do Sul, Programa de Pós-Graduação em Computação,
Porto Alegre, Brasil
geyer@inf.ufrgs.br

Resumo

Uma das características mais marcantes das aplicações pervasivas é a adaptação ao contexto de seu interesse. Para tratar alguns dos problemas envolvidos nesse processo, propõem-se o uso de ontologias. Pela possibilidade de empregar uma semântica de maior expressividade que a usualmente praticada na coleta e no tratamento dos dados sensorados, se pretende atingir melhores níveis de descrição nas informações que caracterizam o contexto do ambiente computacional. Na proposta apresentada neste artigo, chamada EXEHDA-ON, o uso de ontologias permite pesquisa e inferência em uma linguagem de alto nível. Entende-se como principais contribuições deste trabalho: (i) a definição de um modelo ontológico que caracteriza um ambiente pervasivo de computação e (ii) a integração deste modelo, através de um mecanismo de sensibilidade ao contexto, à arquitetura de software do middleware EXEHDA.

Palavras chave: Computação Pervasiva, Computação Sensível ao Contexto, Arquitetura de Software, Web Semântica.

1 Introdução

Mark Weiser [22] idealizou ambientes físicos com dispositivos computacionais integrados que auxiliariam indivíduos na realização de suas tarefas cotidianas ao fornecer-lhes informações e serviços de forma contínua e transparente, iniciando uma área de pesquisa, denominada Computação Ubíqua, que estuda a integração da tecnologia às atividades humanas de forma transparente, quando e onde for necessário. Essa proposta ainda está distante de uma ampla aplicação prática baseada em produtos de mercado. Porém, alguns autores [12] [5] têm apontado a gradativa concretização da proposta de Weiser, através de vários avanços tecnológicos na micro-eletrônica, nas tecnologias relativas a sensores, redes sem-fio e redes de alta velocidade, aumento contínuo do poder de processamento computacional e disponibilização de dispositivos móveis.

Com uma premissa mais próxima das atuais tecnologias de hardware e software, a Computação Pervasiva representa uma etapa em direção à consecução das propostas de Weiser. A Computação Pervasiva é um paradigma que permite ao usuário o acesso a seu ambiente computacional independente de localização, tempo e equipamento. Em um ambiente de Computação Pervasiva, os dispositivos, serviços e componentes de software devem ser conscientes de seus contextos e automaticamente adaptar-se às suas mudanças, caracterizando assim sensibilidade ao contexto [23] [17] [24] [2].

Um mecanismo para sensibilidade ao contexto deve prover suporte para diferentes tarefas, dentre as quais: (i) obtenção do contexto de diversas fontes (sensores físicos e sensores lógicos); (ii) interpretação dos dados sensorados, gerando dados contextualizados e (iii) disseminação das informações para as partes interessadas, de modo distribuído e personalizado.

Uma questão relevante na sensibilidade ao contexto é o grau de expressividade que se pode obter na descrição dos possíveis estados do mesmo. Quanto maior a expressividade do modelo de informação do contexto, maior é a capacidade de representar a estrutura e a semântica dos conceitos. Também, quanto mais formal o modelo para descrição do contexto, maior é a capacidade de realização de pesquisa e inferência sobre ele. Neste sentido, considera-se que o uso de ontologias contribui para qualificar os mecanismos de sensibilidade ao contexto, em função da elevada expressividade que o uso destas pode propiciar.

Desta forma, este trabalho apresenta um mecanismo de sensibilidade ao contexto para coletar, processar e disseminar informações de contexto na perspectiva da Computação Pervasiva, considerando o ambiente pervasivo definido no projeto ISAM (Infra-estrutura de Suporte às Aplicações Móveis Distribuídas) e provido pelo *middleware* EXEHDA (*Execution Environment for Highly Distributed Applications*) [1] [23].

O EXEHDA-ON utiliza uma abordagem baseada em ontologias para a modelagem do contexto do ambiente pervasivo, bem como para realizar pesquisas no correspondente modelo ontológico. Este modelo ontológico tem abrangência celular, sendo alimentado por um serviço de monitoramento. Deste modo, o modelo descreve semanticamente o estado atual do ambiente, utilizando um vocabulário comum e interpretável pelos servidores de contexto existentes nas células de execução do EXEHDA. Além disso, o fato de ser utilizado um modelo ontológico também torna possível a realização de inferências sobre o estado do ambiente pervasivo.

O artigo está organizado nas seguintes seções: a seção 2 apresenta uma descrição do *middleware* EXEHDA, o qual constitui o ambiente foco da proposta; a seção 3 apresenta as tecnologias utilizadas na construção do modelo ontológico do EXEHDA-ON; a seção 4 descreve a concepção da modelagem do EXEHDA-ON; a seção 5 descreve a prototipação do EXEHDA-ON; a seção 6 apresenta os trabalhos relacionados; e a seção 7 apresenta as considerações finais.

2 *Middleware* EXEHDA: Principais Características

O EXEHDA é um *middleware* adaptativo ao contexto e baseado em serviços que visa criar e gerenciar um ambiente pervasivo, bem como promover a execução das aplicações direcionadas à Computação Pervasiva. Estas aplicações são distribuídas, móveis e adaptativas ao contexto em que seu processamento ocorre, devendo estar disponíveis, a partir de qualquer lugar, todo o tempo.

Para atender a elevada flutuação na disponibilidade dos recursos, inerente à Computação Pervasiva, o EXEHDA é estruturado em um núcleo mínimo e em serviços carregados sob demanda. Os principais serviços fornecidos estão organizados em subsistemas que gerenciam: (i) a execução distribuída; (ii) a comunicação; (iii) o reconhecimento do contexto; (iv) a adaptação; (v) o acesso pervasivo aos recursos e serviços; (vi) a descoberta e; (vii) o gerenciamento de recursos.

O suporte à adaptação no EXEHDA está associado à operação do subsistema de reconhecimento de contexto e adaptação. Este subsistema inclui serviços que tratam desde a extração das “informações brutas” sobre as características dinâmicas e estáticas dos recursos que compõem o ambiente pervasivo, 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.

O serviço “Collector” é responsável pela extração das informações diretamente dos recursos envolvidos, as quais, posteriormente refinadas, darão origem aos elementos de contexto. Um componente “Monitor”, localizado no recurso, gerencia um conjunto de sensores parametrizáveis. Por outro lado, entre os consumidores da informação extraída pela monitoração estão os serviços “ContextManager” e “Scheduler” [23].

No *middleware*, 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. Atualmente, o processo de tradução dos dados sensorados para contextualizados é feito por algoritmo e estrutura de dados personalizados por tipo de aplicação [23]. A contribuição central da pesquisa EXEHDA-ON é minimizar a gerência desta personalização por parte do desenvolvedor.

3 EXEHDA-ON: Tecnologias de Web Semântica Utilizadas

A construção e o processamento do modelo ontológico do EXEHDA-ON são baseados em tecnologias da Web Semântica. Nesta seção são caracterizadas as principais tecnologias empregadas na sua implementação.

Ontologias: uma ontologia corresponde a uma especificação formal e explícita de uma conceituação compartilhada, onde: (i) conceituação, se refere ao modelo abstrato do mundo real; (ii) explícita, significa que os conceitos e seus requisitos são

definidos explicitamente; (iii) formal, indica que a ontologia é processável por máquina, permite raciocínio automático e possui semântica lógica formal; (iv) compartilhada, significa que uma ontologia captura o conhecimento apresentado não apenas por um único indivíduo, mas por um grupo [8]. Como um dos elementos básicos que constitui a Web Semântica, as ontologias proporcionam a interoperabilidade semântica entre aplicações. Isso significa, que é possível explorar aspectos além da interoperabilidade sintática e estrutural fornecida pelos padrões XML (*eXtensible Markup Language*), Esquema XML, RDF (*Resource Description Framework*) e Esquema RDF [9]. Para isso, é necessário utilizar uma linguagem de ontologia: (i) compatível com padrões como XML, Esquema XML, RDF e Esquema RDF; (ii) com sintaxe rica para representar conhecimento e regras e permitir a inferência de novos dados; (iii) de fácil compreensão e extensão; e (iv) que tenha suporte eficiente de ferramentas para a manipulação de informações de contexto segundo o modelo subjacente às ontologias criadas nessa linguagem.

OWL: a linguagem escolhida para construção do modelo ontológico do EXEHDA-ON foi a OWL (*Web Ontology Language*) [3], padrão para a Web Semântica. Como sub-linguagem foi adotada a OWL-DL, esta sub-linguagem corresponde à lógica de descrição e provê um maior grau de expressividade onde todas as conclusões são computáveis e todas as computações terminam em tempo finito. A OWL fornece suporte a metadados RDF, abstrações de classes, generalização, agregação, relações de transitividade, simetria e detecção de inconsistências.

SPARQL: *SPARQL Protocol And RDF Query Language* [20] é uma linguagem recomendada pelo W3C (*World Wide Web Consortium*) para realização de consultas em ontologias descritas em RDF e linguagens derivadas. É uma linguagem “orientada a dados”, ou seja, permite apenas extrair dados de documentos RDF disponíveis em rede ou armazenados em um meio físico qualquer, não possuindo mecanismos de inferência.

API Jena: no EXEHDA-ON é utilizada a API Java do *toolkit* Jena [15], por oferecer: (i) mecanismos para manipulação de modelos RDF em memória, bases de dados relacionais e arquivos; (ii) suporte à linguagem de consulta de dados RDF SPARQL; (iii) um conjunto de APIs para manipulação de ontologias codificadas em OWL; (iv) máquinas de inferência baseadas em ontologias e regras, bem como um mecanismo que permite integrar máquinas de inferência de terceiros. O módulo *Reasoner* permite fazer inferências sobre modelos OWL. O uso das inferências sobre modelos semânticos permite obter informação adicional (inferida) sobre as ontologias. Os *reasoners* de OWL da API Jena funcionam aplicando regras tipo *if-then-else* sobre instâncias OWL. O *Reasoner* Genérico Jena permite inferência sobre grafos RDF através de regras que podem ser definidas externamente. Estas regras são formadas basicamente por uma lista de termos de corpo (premissas), uma lista de termos de cabeça (conclusões) e um nome opcional que identifica a regra. Cada termo pode ser uma tripla padrão, uma tripla estendida padrão ou uma chamada a uma primitiva embutida [18]

4 Aspectos de Modelagem do EXEHDA-ON

A modelagem do EXEHDA-ON contemplou dois principais esforços de concepção: (i) modelagem ontológica do ambiente pervasivo e do contexto de interesse das aplicações. Fundamentalmente, esta modelagem prevê o uso de ontologias implementadas em OWL, sobre as quais o EXEHDA-ON realiza a representação e o processamento das informações de contexto; (ii) modelagem da arquitetura de software com a especificação dos diferentes serviços que a integram.

4.1 Modelagem do Ambiente Pervasivo

O modelo ontológico para uso no EXEHDA-ON foi definido considerando aspectos que modelassem o domínio do ambiente pervasivo provido pelo EXEHDA. A perspectiva é que este modelo represente o estado atual do ambiente de execução pervasivo provido pelo EXEHDA, gerando deste modo um conhecimento sobre o mesmo, possibilitando assim sua manipulação pelo servidor de contexto, o qual responde às demandas introduzidas pelas aplicações dos usuários.

A natureza da informação de contexto, nas células de execução, pode ser estática ou dinâmica. Um exemplo de informação estática são os atributos relativos ao tipo do dispositivo, os quais não se alteram com o passar do tempo. Informações dinâmicas traduzem aspectos do contexto que oscilam em uma determinada frequência. Como exemplo de informações dinâmicas têm-se: a temperatura do processador, a ocupação de memória e a ocupação de disco. Estas informações são obtidas por meio de monitoramento periódico ou coletadas por procedimentos disparados por eventos [23].

A metodologia utilizada para a construção da ontologia descritiva do ambiente pervasivo é baseada na proposta de [10]. Os passos utilizados na definição da ontologia projetada para uso no EXEHDA-ON foram os seguintes: (i) definição do domínio e captura do conhecimento; (ii) conceituação do conhecimento capturado em um conjunto de representações; (iii) desenvolvimento do modelo conceitual em uma linguagem formal.

Entre os resultados da etapa de “conceituação do conhecimento capturado em um conjunto de representações”, destaca-se o desenvolvimento de um glossário de termos, contendo os conceitos pesquisados, sua descrição e atributos relacionados.

Ainda nesta etapa foi construída uma árvore de classificação de conceitos, que permite visualizar as classes definidas para a ontologia do ambiente pervasivo tratado pelo EXEHDA-ON, mostrando sua hierarquia (vide Figura 1).

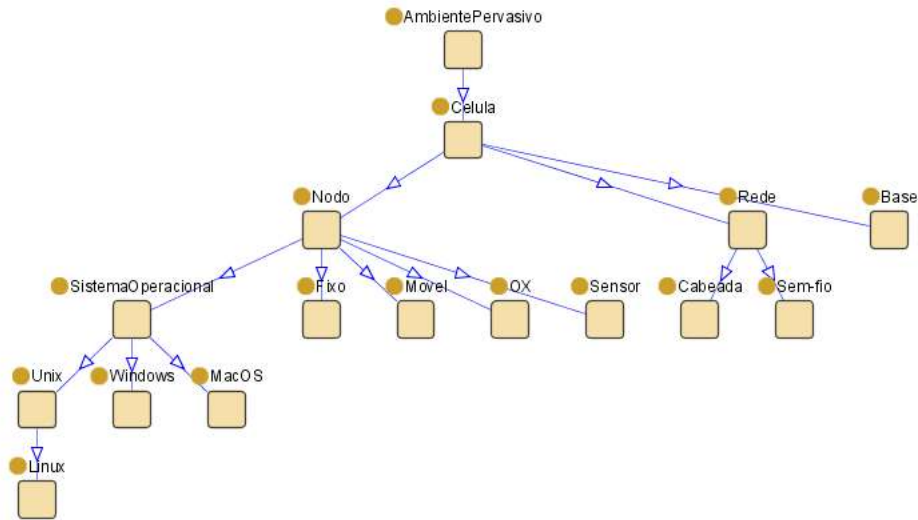


Figura 1. Árvore de conceitos da ontologia do ambiente pervasivo

4.2 Modelagem do Contexto de Interesse das Aplicações

O modelo ontológico para uso no EXEHDA-ON também prevê a representação do contexto de interesse das aplicações através de uma ontologia construída com a mesma metodologia descrita na seção anterior (seção 4.1). O contexto de interesse da aplicação é um subconjunto do contexto geral do ambiente pervasivo, quando o contexto de interesse ocorre a aplicação é notificada. Na Figura 2 é possível visualizar as classes definidas para a ontologia do contexto de interesse das aplicações, mostrando sua hierarquia.

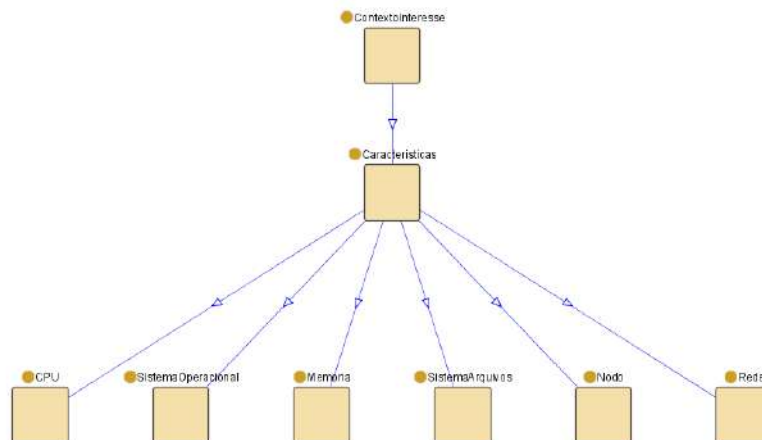


Figura 2. Árvore de conceitos da ontologia do contexto de interesse das aplicações

4.3 Modelagem da Arquitetura de Software

Nesta seção é descrita a arquitetura de software do EXEHDA-ON e a forma de adequação da mesma ao Subsistema de Reconhecimento de Contexto e Adaptação do EXEHDA, identificando seus pontos de integração e seus serviços.

De modo geral, uma arquitetura para sistemas sensíveis ao contexto envolveria uma série de sensores, de software e/ou de hardware, que monitoram os aspectos de interesse do ambiente computacional (*middleware* e aplicações). As informações colhidas por esses sensores são passadas a um conjunto de serviços de contexto, onde são processadas e/ou modificadas para que possam ser entregues aos consumidores das informações contextualizadas [13].

A arquitetura do EXEHDA-ON prevê a inclusão de serviços e componentes nos “EXEHDANodos” e no “EXEHDABase”, os quais são abstrações do ambiente pervasivo provido pelo EXEHDA. A Figura 3 mostra as funcionalidades do EXEHDA-ON integradas ao Subsistema de Reconhecimento de Contexto e Adaptação do EXEHDA.

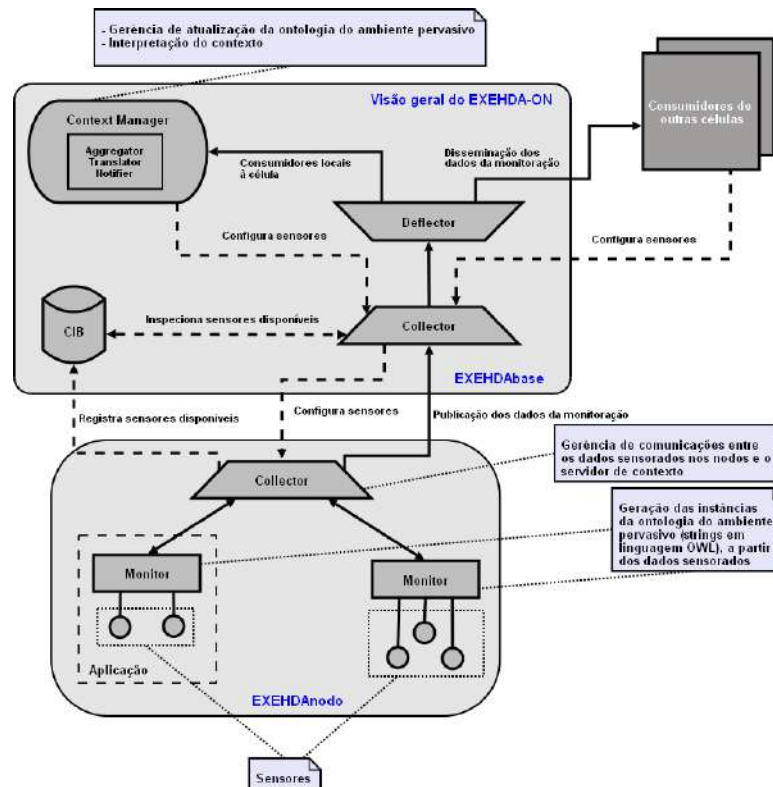


Figura 3. Integração das funcionalidades do EXEHDA-ON

Em cada EXEHDANodo, junto ao componente “Monitor”, o serviço do EXEHDA-ON “Gerenciador de monitoramento do contexto” fica com a responsabilidade de receber os dados sensorados e gerar as instâncias da ontologia do ambiente pervasivo, representadas por *strings* em linguagem OWL. Por sua vez, o serviço “Collector” do EXEHDA é responsável pela gerência das comunicações entre os dados sensorados nos nodos e o servidor de contexto que aglutina os dados da célula como um todo.

A localização dos serviços do EXEHDA-ON dentro do EXEHDABase é junto ao “ContextManager”, que é serviço chave na construção de informações globais de contexto, mais especificamente dentro das estruturas de cadeias de detecção de contexto do “ContextManager” (*aggregator e translator*), que são empregadas, respectivamente, para composição dos dados de um ou mais sensores e para abstração da informação de contexto.

Assim, no EXEHDABase atua o serviço responsável pela aglutinação das instâncias na ontologia do ambiente pervasivo, denominado “Gerenciador de atualização da base ontológica”, e o serviço responsável pelas consultas e inferências sobre a base ontológica do EXEHDA-ON, denominado “Interpretador de contexto”.

4.4 Descrição dos serviços do EXEHDA-ON

No EXEHDA-ON estão previstos diversos serviços que gerenciam a obtenção, processamento e disseminação das informações de contexto. A Figura 4 apresenta uma visão geral destes serviços, os quais são descritos a seguir.

Gerenciador de subscrição: as subscrições correspondem ao meio pelo qual os consumidores têm a possibilidade de registrar suas requisições aos serviços de contexto providos pelo EXEHDA-ON. Assim, este serviço interage com os consumidores das informações de contexto possibilitando o registro de interesse em determinadas condições de contexto. Também,

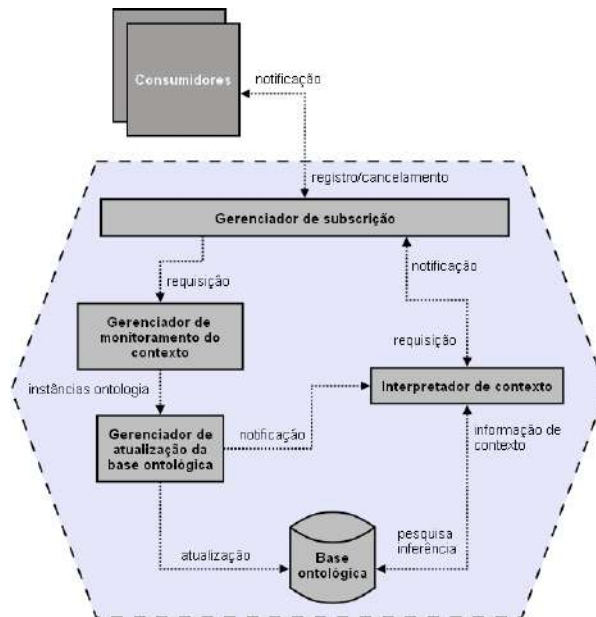


Figura 4. Visão geral dos serviços do EXEHDA-ON

quando o consumidor não tiver mais interesse em alguma informação de contexto este serviço permite o cancelamento da subscrição. Cabe ao “Gerenciador de subscrição” interpretar e gerenciar as solicitações enviadas pelos consumidores e distribuí-las aos demais serviços que formam a arquitetura do EXEHDA-ON, com a intenção de identificar a existência das condições de contexto de interesse do consumidor e notificá-lo de sua existência.

Interpretador de contexto: o “Gerenciador de subscrição” envia para o “Interpretador de contexto” as requisições dos consumidores registrados com o intuito de identificar a existência das condições de contexto de interesse do consumidor. Para executar essa tarefa, o serviço “Interpretador de contexto” pode realizar consultas e inferências sobre a base ontológica.

Gerenciador de monitoramento do contexto: caso as informações de contexto existentes na base ontológica não atendam ao interesse de algum consumidor registrado, o “Gerenciador de subscrições” encaminha o pedido ao serviço “Gerenciador de monitoramento do contexto”, o qual através de arquitetura de monitoração do EXEHDA (i) ativa ou desativa sensores em decorrência do interesse dos consumidores; (ii) recebe os dados sensorados e (iii) gera instâncias da ontologia no formato da linguagem OWL. Estas tarefas possibilitam a atualização da base ontológica.

Gerenciador de atualização da base ontológica: este serviço recebe as instâncias da ontologia do ambiente pervasivo expressas em linguagem OWL geradas pelo serviço “Gerenciador de monitoramento do contexto” e, através da API Jena, manipula a base ontológica realizando a atualização desta ontologia. Também, notifica o “Interpretador de contexto” quando ocorre uma atualização nesta ontologia.

4.5 Serviço de Interpretação de Contexto

Dentre os serviços que constituem o EXEHDA-ON, o serviço “Interpretador de contexto” é central para consecução dos objetivos propostos para o EXEHDA-ON.

O “Interpretador de contexto” realiza o processamento das informações de contexto com o intuito de identificar na base ontológica a existência das condições de contexto de interesse dos consumidores registrados. Este serviço recebe as solicitações dos consumidores a partir do “Gerenciador de subscrições” e realiza consultas e inferências sobre a base ontológica, utilizando o componente “Gerenciador de consultas e inferências”.

O serviço de interpretação do contexto possui um componente que realiza consultas à base ontológica. O componente “Gerenciador de consultas e inferências” constrói consultas com a linguagem SPARQL, as quais são submetidas à base ontológica. Também, realiza serviços de inferências sobre a ontologia, identificando, por exemplo, eventuais inconsistências que possam existir nas consultas, bem como extraindo informações que não estejam explicitamente escritas na estrutura do modelo ontológico (vide Figura 5).

Quando um consumidor registra-se no *middleware* EXEHDA, este deve informar seu contexto de interesse. As informações registradas são processadas pelo componente “Gerenciador de consultas e inferências” com base na ontolo-

gia de contexto de interesse da aplicação. Este componente, então, (i) analisa os requisitos especificados pelo consumidor, aplicando regras de inferência que permitem verificar a consistência das consultas e estendê-las semanticamente; (ii) gerencia a construção das consultas; (iii) executa as consultas sobre a ontologia; e (iv) notifica as condições de contexto existentes, em função do interesse dos consumidores registrados.

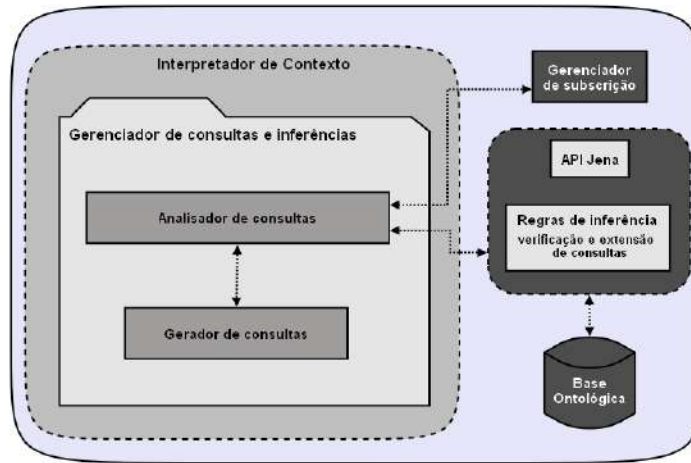


Figura 5. Componente Gerenciador de consultas e inferências

5 Prototipação do EXEHDA-ON

Nesta seção estão descritos os aspectos de prototipação do servidor de contexto do EXEHDA-ON e de uma aplicação desenvolvida para inspeção do estado do contexto celular.

5.1 Servidor de contexto

Um protótipo do servidor de contexto para o ambiente pervasivo foi modelado e desenvolvido com o uso da API Jena e a linguagem de programação Java. Para testar o protótipo foram criados sensores para dados estáticos (arquitetura, número de núcleos, tipo de processador, número de interfaces de rede, memória física, disco instalado) e dinâmicos (processador disponível, memória disponível, disco disponível, tráfego de entrada na rede, tráfego de saída da rede).

Os dados sensorados são compostos em instâncias da ontologia do ambiente pervasivo no nodo (processamento de *strings*), as quais são enviadas ao servidor de contexto, localizado no EXEHDABase. Estas instâncias contêm as informações de identificação do nodo e os dados sensorados. No servidor de contexto é feita a aglutinação destas instâncias na ontologia do ambiente pervasivo na célula com a utilização da API Jena.

Com o intuito de abstrair aspectos de baixo nível relativos ao tratamento das comunicações em rede, na implementação do servidor de contexto foram utilizados os serviços de comunicação do EXEHDA. A classe principal do servidor de contexto executa uma *thread* responsável por utilizar esses serviços para aguardar as atualizações de contexto dos nodos.

O servidor de contexto mantém uma base de dados ontológica com informações sobre o estado do ambiente pervasivo. Todos os procedimentos relativos à manipulação da ontologia são realizados com o uso da API Jena. Quando a *thread* responsável pela comunicação recebe a ontologia do ambiente pervasivo atualizada de algum nodo ela a repassa para a classe principal “*ContextServer*”, a qual possui o método “*mergeOntologies*” responsável por verificar a ontologia recebida e realizar as atualizações necessárias na base de dados ontológica. O servidor é executado na base da célula, gravando as alterações ocorridas na ontologia dentro do arquivo *log* do EXEHDA.

Também, um serviço de consultas foi desenvolvido junto ao servidor de contexto. Este serviço possibilita a realização de pesquisas na ontologia do ambiente pervasivo utilizando a linguagem SPARQL. A linguagem de consulta SPARQL é utilizada através da biblioteca ARQ da API Jena. Os métodos da classe *QueryConnection* implementam as consultas em SPARQL que pesquisam propriedades dos nodos, tais como: temperatura do processador, memória total, tamanho do disco e número de processadores.

5.2 Inspetor do estado do contexto celular

Para avaliação da integração das tecnologias no atendimento às demandas de prototipação do modelo foi desenvolvida uma aplicação que se vale do EXEHDA-ON para inspecionar o estado atual do contexto celular. Esta aplicação foi desenvolvida utilizando a linguagem Java e a API Jena.

O Inspetor disponibiliza um resumo do estado celular, indicando o número de nodos móveis e fixos que estão ativos. Também, permite ver informações detalhadas a respeito de um nodo selecionado. Estas informações abrangem: (i) dados estáticos: arquitetura, disco instalado, memória física, número de interfaces de rede, número de núcleos; e (ii) dados dinâmicos: ocupação de CPU, ocupação de memória, disco disponível, temperatura processador.

Este aplicativo utiliza as consultas SPARQL previamente definidas no componente de software “Gerenciador de consultas e inferências” do “Servidor de contexto do EXEHDA-ON”, sendo possível visualizar o estado geral da célula (vide figura 6) e o estado específico de cada nodo (vide figura 7).

O fato de utilizar a linguagem SPARQL permite que as informações disponibilizadas pelo Inspetor possam ser ampliadas pelo acréscimo de novas consultas ao componente “Gerenciador de consultas e inferências”, extraindo outras informações da base ontológica.

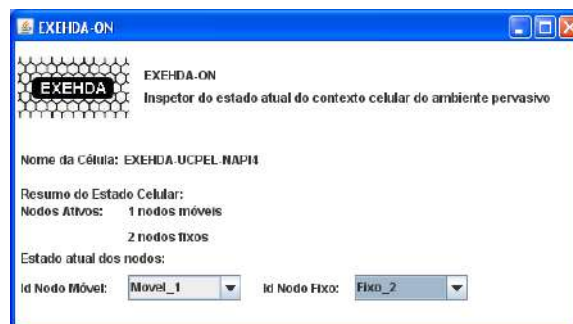


Figura 6. Interface principal da aplicação Inspetor

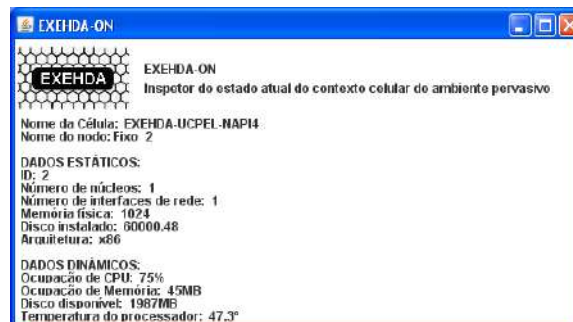


Figura 7. Detalhes de um nodo na aplicação Inspetor

6 Trabalhos Relacionados

Nesta seção é feita uma comparação entre os principais projetos em Computação Sensível ao Contexto utilizados como referência para a definição deste trabalho, e a solução adotada no EXEHDA-ON. Esta comparação, apresentada nos itens a seguir, está organizada em três grandes categorias: arquitetura, modelagem de contexto e processamento de contexto.

- **Arquitetura:** CASS [7] tem uma arquitetura baseada em *middleware* centralizado. CoBra [4] possui arquitetura baseada em agentes. A arquitetura da plataforma *Infraware* [16] é constituída por um *middleware* baseado em *Web-Services*. Considerando que o EXEHDA-ON foi concebido como extensão do *middleware* EXEHDA, ele mantém os aspectos arquiteturais do mesmo. Assim, sua arquitetura é baseada em serviços, cuja integração visa fornecer a infra-estrutura necessária para suporte à sensibilidade ao contexto em um ambiente pervasivo.

- **Modelagem de contexto:** *Context Toolkit* [6] manipula o contexto através de tuplas com atributos e valores que são codificados usando XML. *Hydrogen* [14] usa uma abordagem orientada a objetos para modelagem do contexto. A estrutura e o vocabulário da ontologia aplicada no *Context Managing Toolkit* são descritos em RDF. No Gaia [19] o contexto é representado através de predicados escritos em DAML+OIL. Abordagens baseadas em ontologias escritas em OWL para modelagem de contexto são encontradas nos projetos *SOCAM* [11], *CoBrA* e *Infraware*. A solução adotada no EXEHDA-ON para modelagem do contexto consiste no uso de ontologias desenvolvidas em OWL. O modelo baseado em ontologias é o mais promissor para a modelagem de contexto em ambientes pervasivos [21]. As ontologias do EXEHDA-ON são próprias e descrevem o contexto de interesse das aplicações e o ambiente pervasivo. A linguagem OWL, escolhida para especificação das ontologias do EXEHDA-ON, é recomendada como um padrão de linguagem pela W3C, visto que, revisa e incorpora melhoramentos às características das demais linguagens, tais como: RDF e *RDF Schema* e DAML-OIL.
- **Processamento de contexto:** o processamento do contexto nos projetos *CASS*, *CoBrA*, *Context Toolkit*, *CORTEX*, *SOCAM* e *Infraware* é baseado em motores de inferência que realizam a interpretação do contexto. Modelos de contexto não baseados em ontologias apresentam menor expressividade o que reduz a possibilidade de realização de inferências, sendo esta uma das limitações dos projetos *CASS*, *Context Toolkit* e *CORTEX*. No EXEHDA-ON o processamento do contexto é realizado através de serviços e componentes de software que manipulam a base ontológica. As tarefas de interpretação de contexto incluem pesquisa e inferência sobre o modelo ontológico, utilizando regras de inferência e a linguagem de consulta SPARQL. O “Interpretador de contexto” é implementado com o uso da API Jena.

7 Considerações Finais

Uma questão relevante na sensibilidade ao contexto é o grau de expressividade que se pode obter na descrição dos possíveis estados do mesmo. Quanto maior a expressividade do modelo de informação do contexto, maior é a capacidade de representar a estrutura e a semântica dos conceitos. A análise dos métodos e tecnologias de Web Semântica caracteriza que quanto mais formal o modelo para descrição do contexto, maior é a capacidade de realização de pesquisa e inferência sobre o mesmo. Neste sentido, o uso de ontologias contribui para qualificar os mecanismos de sensibilidade ao contexto, em função da elevada expressividade que o uso destas pode propiciar.

A utilização de tecnologias de Web Semântica, em especial de ontologias, permite criar modelos formais, semânticos e extensíveis que descrevam o domínio das aplicações a partir da especificação de conceitos, relações e axiomas. O compartilhamento desses modelos promove a interoperabilidade semântica entre as aplicações e o EXEHDA-ON. A extensibilidade proporcionada pelo uso de ontologias permite ainda que mudanças no modelo acarretem alterações mínimas nas aplicações desenvolvidas. A especificação semântica das informações contextuais e do domínio das aplicações possibilita ainda a interpretação e inferência de novos fatos, baseados nas informações semânticas descritas nos modelos.

O processo de tradução dos dados sensorados para contextualizados no EXEHDA vinha sendo feito por algoritmos e estruturas de dados particulares para cada tipo de aplicação. Neste sentido, a contribuição central da pesquisa EXEHDA-ON é minimizar a gerência desta personalização por parte do programador, devido à construção de um modelo ontológico que descreve semanticamente o estado atual do ambiente, utilizando um vocabulário comum e interpretável pelo servidor de contexto. Além disso, o fato de ser utilizado um modelo ontológico também torna possível a realização de pesquisas e inferências sobre o estado do ambiente pervasivo, com a utilização de uma linguagem de alto nível.

Na perspectiva de continuidade da pesquisa desenvolvida durante a concepção do EXEHDA-ON os seguintes aspectos poderão ser explorados em trabalhos futuros: (i) analisar o desempenho do EXEHDA-ON considerando as diferentes possibilidades de modelagem ontológica do contexto; (ii) construir diferentes modelos ontológicos em função do domínio a ser tratado; (iii) expandir o mecanismo para construção de contextos que englobem várias células; (iv) ampliar a abrangência das regras de inferência utilizadas pelo serviço de interpretação de contexto.

Referências

- [1] I. Augustin. *Abstrações para uma Linguagem de Programação Visando Aplicações Móveis Conscientes do Contexto em um Ambiente de Pervasive Computing*. Tese (doutorado em ciência da computação), Instituto de Informática, UFRGS, Porto Alegre, RS, 2004.
- [2] I. Augustin, A. C. Yamin, L. C. da Silva, R. A. Real, G. Frainer, and C. F. R. Geyer. Isamadapt: abstractions and tools for designing general-purpose pervasive applications: Experiences with auto-adaptive and reconfigurable systems. *Softw. Pract. Exper.*, 36(11-12):1231–1256, 2006.
- [3] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein. Owl web ontology language reference, 2004. Disponível em: <<http://www.w3.org/TR/owl-ref/>>. Acesso em maio de 2008.

- [4] H. Chen. *An Intelligent Broker Architecture for Pervasive Context-Aware Systems*. Dissertation (doctor of philosophy), University of Maryland, Baltimore, 2004.
- [5] C. A. da Costa, A. C. Yamin, and C. F. R. Geyer. Toward a general software infrastructure for ubiquitous computing. *IEEE Pervasive Computing*, 7(1):64–73, 2008.
- [6] A. Dey, D. Salber, and G. Abowd. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction*, 16, 2001.
- [7] P. Fahy and S. Clarke. Cass - middleware for mobile context-aware applications. *MobiSys - International Conference on Mobile Systems, Applications, and Services*, 2004.
- [8] D. Fensel. Ontologies: Silver bullet for knowledge management and electronic commerce. *Springer - Verlag*, 2000.
- [9] D. Fensel, W. Wahlster, H. Lieberman, and J. Hendler. Spinning the semantic web: Bringing the world wide web to its full potential. *The MIT Press*, 2005.
- [10] M. Fernandez, A. Gomes-Perez, and N. Juristo. Methontology: From ontological art towards ontological engineering. In *National Conference on Artificial Intelligence (AAAI) Spring Symposium on Ontological Engineering, 14.*, Papers from the AAAI Spring Symposium, pages 33–40, Stanford, EUA., 1997. AAAI press.
- [11] T. Gu, H. Pung, and D. Zhang. A middleware for building context-aware mobile services. In *Proceedings of IEEE Vehicular Technology Conference*, Milão, Itália, Maio 2004. IEEE Press.
- [12] S. Helal. Programming pervasive spaces. *IEEE Pervasive Computing*, 4(1):84–87, 2005.
- [13] K. Henriksen and J. Indulska. Developing context-aware pervasive computing applications: Models and approach. *Pervasive and Mobile Computing*, 2(2):37–64, 2006.
- [14] T. Hofer, W. Schwinger, M. Pichler, G. Leonhartsberger, and J. Altmann. Context-awareness on mobile devices - the hydrogen approach. 2002.
- [15] B. McBride. Jena api - a semantic web framework for java, 2007. Disponível em: <<http://jena.sourceforge.net/ontology/>>. Acesso em julho de 2008.
- [16] J. G. Pereira Filho, R. M. Pessoa, C. Z. Calvi, N. Q. Oliveira, R. R. M. Carmo, A. C. P. Barbosa, C. R. G. Farias, and M. M. Leite. Infraware: um middleware de suporte a aplicações móveis sensíveis ao contexto. In *SBRC - Simpósio Brasileiro de Redes de Computadores*, Curitiba, 2006.
- [17] M. Raento, A. Oulasvirta, R. Petit, and H. Toivonen. Contextphone: A prototyping platform for context-aware mobile applications. *IEEE Pervasive Computing*, 4(2):51–59, April-June 2005.
- [18] D. Reynolds. Jena 2 inference support, 2007. Disponível em: <<http://jena.sourceforge.net/inference/index.html>>. Acesso em julho de 2008.
- [19] M. Roman, C. Hess, R. Cerqueira, A. Ranganat, R. Campbell, and K. Nahrstedt. Gaia: A middleware infrastructure to enable active spaces. *IEEE Pervasive Computing*, Outubro 2002.
- [20] A. Seaborne. Sparql - a query language for rdf, 2007. Disponível em: <<http://www.w3.org/TR/rdf-sparql-query/>>. Acesso em julho de 2008.
- [21] T. Strang and C. Linnhoff-Popien. A context modeling survey. In *Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - The Sixth International Conference on Ubiquitous Computing*, Nottingham, England, 2004. UbiComp.
- [22] M. Weiser. The computer for the 21st century. *Scientific American*, 3(265):94–104, Setembro 1991.
- [23] A. Yamin. *Arquitetura para um Ambiente de Grade Computacional Direcionado às Aplicações Distribuídas, Móveis e Conscientes do Contexto da Computação Pervasiva*. Tese (doutorado em ciência da computação), Instituto de Informática, UFRGS, Porto Alegre, RS, 2004.
- [24] A. C. Yamin, I. Augustin, J. Barbosa, L. C. da Silva, R. A. Real, A. S. Filho, and C. F. R. Geyer. Exehda: Adaptive middleware for building a pervasive grid environment. *Frontiers in Artificial Intelligence and Applications - Self-Organization and Autonomic Informatics*, 135:203–219, 2005.