

Controle da Adaptação na UbiComp (Position Paper)

Nelsi Warken¹, Adenauer Corrêa Yamin¹, Iara Augustin², Cláudio F. R. Geyer³

¹PPGINF – Centro Politécnico – Universidade Católica de Pelotas

²PPGI – Centro de Tecnologia – Universidade Federal Santa Maria

³PPGC – Instituto de Informática – Universidade Federal Rio Grande do Sul

Abstract. *This article presents review of the state of the art in terms of control of adaptation in UbiComp. We selected twenty projects that met, even partially, twelve functional characteristics perceived as important for the mechanism being proposed for the middleware EXEHDA. The projects were compared considering these characteristics. This revision is a necessary step in proposing the mechanism desired for adaptation control. After analysing these projects, we highlighted the reasons for the use of ontologies in this mechanism and the challenges to overcome in the design of the mechanism to control the adaptation in ubiquitous computing.*

Resumo. *Este artigo contribui com uma revisão sistematizada do estado da arte em termos de controle da adaptação na UbiComp. Foram selecionados vinte projetos que atendessem, mesmo que parcialmente, doze características funcionais entendidas como importantes para o mecanismo a ser proposto para o middleware EXEHDA. Os projetos foram comparados considerando estas características. Foram feitas algumas considerações a respeito destes projetos. Esta revisão é etapa indispensável na proposição do mecanismo de controle da adaptação pretendido. Após esta revisão, foram destacadas as razões para o uso de ontologias neste mecanismo e os desafios a serem vencidos para a concepção do mecanismo para controle da adaptação na Computação Ubíqua.*

1. Introdução

Computação Ubíqua, ou UbiComp, trata do acesso ao ambiente computacional do usuário, isto é, ao espaço ubíquo do usuário, em qualquer lugar, todo o tempo, com qualquer dispositivo, constituindo um ambiente altamente distribuído, heterogêneo, dinâmico, móvel, mutável e com forte interação entre homem e máquina [Augustin et al. 2006].

As aplicações ubíquas precisam adaptar-se ao ambiente, compreendendo o contexto em que estão inseridas [Maciel and Assis 2004]. Essa nova classe de sistemas computacionais, adaptativos ao contexto, abre perspectivas para o desenvolvimento de aplicações mais ricas, elaboradas e complexas, que exploram a natureza dinâmica e a mobilidade do usuário. Entretanto, o desenvolvimento de aplicações que se adaptem continuamente ao ambiente e permaneçam funcionando mesmo quando o indivíduo se movimentar ou trocar de dispositivo [Costa et al. 2008], continua um desafio de pesquisa em aberto.

Assim, motivado por estas idéias, o trabalho como um todo propõe-se a utilizar Ontologias, Web Services e Sistemas Autônomos para Controle da Adaptação ao Contexto na Computação Ubíqua, com o objetivo de facilitar o desenvolvimento de aplicações ubíquas. Outra motivação é a possibilidade de sugerir um mecanismo de controle de adaptação, funcional e não funcional, para o *middleware* EXEHDA, foco de desenvolvimento do G3PD [Yamin et al. 2005]. Será criado um modelo ontológico para o ambiente computacional ubíquo provido pelo *middleware* EXEHDA. A proposta é tomar decisões automáticas de adaptação para este ambiente, com base em informações monitoradas, informações semânticas e inferências lógicas a partir das mesmas.

Ontologia é um elemento central da Web Semântica e vem se mostrando um instrumento promissor para especificar as descrições de contexto na computação ubíqua, atuando como um padrão para descrições, caracterizações, configurações e definições de perfis de usuários, que poderão ser compartilhados pelas aplicações e/ou sistema de execução [Hilera and Ruiz 2006].

Por sua vez, a Computação Autônoma, também denominada Autonômica, apresenta uma visão onde os sistemas de software possuem a capacidade de auto-gerência, baseados em informações contextuais e guiados por políticas descritas em alto nível por seus administradores. Assim, as complexidades de baixo-nível seriam abstraídas dos usuários, permitindo que estes possam se concentrar na especificação de políticas descritas em alto nível [Menasce and Kephart 2007].

Este artigo contribui com uma revisão sistematizada do estado da arte em termos de controle da adaptação na UbiComp. Esta revisão é etapa indispensável na proposição do mecanismo de controle da adaptação pretendido. O texto tem a seguinte organização: na seção 2, são relacionados as características de comparação, os projetos selecionados, a tabela comparativa e algumas considerações a respeito destes projetos; na seção 3 são mostradas as razões para o uso de ontologias no mecanismo proposto e na seção 4 são feitas as considerações finais deste artigo.

2. Uma revisão do Controle da Adaptação na UbiComp

Esta seção resume a revisão feita na área de controle da adaptação na UbiComp. Para esta revisão foram selecionados projetos que atendessem as características funcionais entendidas como importantes para o mecanismo a ser proposto para o *middleware* EXEHDA. As características consideradas foram:

1. Monitoração em tempo real: para perceber alterações no contexto, as informações obtidas dos sensores e dispositivos devem ser as mais atuais possíveis;
2. Monitoração de vários tipos de contexto: contexto do usuário, contexto do ambiente, contexto das aplicações, contexto computacional;
3. Localização: essa característica permite que o usuário visualize a sua localização, a localização de seus serviços e seus dispositivos;
4. Análise do perfil do usuário: informações sobre as suas preferências para determinado serviço;
5. Anúncio de produtos ou serviços: esta característica faz parte do contexto do ambiente e permite com que o usuário receba anúncios de produtos e/ou serviços que estejam próximos de sua localização;

6. Orientado a serviço: os serviços são executados sob demanda e permite que novos serviços heterogêneos sejam adicionados a arquitetura de forma padronizada, acrescentando mais funcionalidades ao sistema;
7. Descoberta de recursos ou serviços: mostra se a solução provê os mecanismos necessário para a descoberta de recursos ou serviços;
8. Modelagem de Domínio: indica se a solução permite que a informação referente ao domínio da aplicação seja modelada;
9. Suporte a Raciocínio: indica se a solução provê algum mecanismo que permita que seja realizado algum raciocínio sobre a informação adquirida;
10. Compartilhamento: indica se a solução permite que a informação contextual seja compartilhada pelas aplicações;
11. Histórico: se existe alguma forma de registro das situações de contexto e/ou adaptações realizadas pelo modelo;
12. Ontologias: se o modelo usou semântica para modelagem de contexto, ou especificação de regras ou condições para adaptação, descrição ou seleção de ações de adaptação.

Tendo por base estas características foram selecionados 20 projetos, quais sejam: OMNIPRESENT, SOCAM, CybreMinder, Context Toolkit, AROUND, CoBrA, Online Aalborg Guide, Flame2008, Nexus, ICAMS, AMS, SOAM, CAMidO, JCAF, CORTEX, LOTUS, ADAPT!, SECAS, PROTEUS, CAPPUCINO. Estes trabalhos estão discutidos e referenciados na publicação [Warken 2008].

A Tabela 1 caracteriza quais modelos possuem as características enumeradas acima. Os campos marcados com "+" indicam que o modelo possui a característica. Os campos sem marcação indicam que a ferramenta não possui a mesma funcionalidade.

Em relação à monitoração do contexto, o Omnipresent possibilita a construção de regras que podem ser adicionadas ou removidas a qualquer momento. Em outras aplicações, como o SOCAM, as regras devem ser pré-carregadas no sistema para que passem a monitorar o contexto. No Flame2008, o usuário escolhe situações pré-definidas no sistema, no qual, os anúncios de produtos devem ser apresentados ao cliente através do seu dispositivo móvel. No Omnipresent, o usuário pode criar regras para encontrar pontos de interesse, quando certas situações acontecem, por exemplo, quando estiver com fome, para listar as lanchonetes num raio de 1 Km. Por sua vez o Flame2008 analisa o histórico do usuário para deduzir novas informações sobre o seu perfil.

Muitos projetos que contemplam sensibilidade ao contexto consideram apenas a localização, *Location Based Systems*. Mudando a localização do usuário, o sistema propõe diferentes informações, serviços ou produtos. Um exemplo típico seriam sistemas voltados ao turismo. Muitos sistemas Web são adaptativos quanto ao conteúdo, em função de perfil do usuário. Outros ainda, são adaptativos ao dispositivo utilizado, onde a interface do usuário muda de acordo com o dispositivo. Também temos alguns sistemas que se adaptam na área médica, onde alguns softwares adaptam o conteúdo a ser disponibilizado em função do usuário (administrador, médico, enfermeiro, atendente, residente, entre outros) e do dispositivo utilizado, adaptando o formato, interface do usuário (resolução da imagem, cores, sintetização de voz, tamanho de tela, imagens, texto). Nestes sistemas podem ser manipulados exames médicos, como eletrocardiogramas, ultra-sons, análises laboratoriais. Outros tipos de adaptações de conteúdo também podem ser encontrados,

Tabela 1. Comparativo: Características (1 a 12) e Modelos *Context-aware*

MODELOS	01	02	03	04	05	06	07	08	09	10	11	12
1. OMNIPRESENT	+	+	+	+	+	+		+	+		+	+
2. SOCAM	+	+	+		+	+			+	+		
3. CYBREMINDER	+	+	+							+		
4. Context Toolkit	+	+	+	+			+	+				
5. AROUND	+		+							+		
6. CoBrA	+							+	+	+		+
7. Online Aalborg Guide	+	+	+	+								
8. FLAME2008	+	+	+	+	+	+		+			+	+
9. NEXUS	+	+	+			+	+	+		+		
10. ICAMS	+	+	+	+						+		
11. AMS	+	+	+	+								
12. SOAM	+	+	+					+	+	+		+
13. CAMidO	+	+						+	+			+
14. JCAF		+				+		+		+		
15. CORTEX	+							+				
16. LOTUS	+	+					+	+	+	+		+
17. ADAPT!		+					+	+	+			
18. SECAS		+	+	+				+	+			+
19. PROTEUS		+		+				+	+			+
20. CAPPUCINO		+	+				+					

como adaptação de idiomas, compressão, decomposição, agregação de dados. Temos um número bem expressivo de trabalhos em adaptação de conteúdo para dispositivos móveis (telefone celular, PDAs, entre outros).

Os trabalhos relacionados na Tabela 1 focam alguns aspectos e deixam lacunas em outros. Por exemplo, a maioria dos *middlewares* focam a etapa de aquisição da informação contextual, levando em conta questões como descoberta de nós e serviços em redes heterogêneas, porém não abordam a forma como a informação é representada e como é feita a inferência sobre a mesma. Já as abordagens que usam ontologias, lidam bem com a parte de representação do contexto, porém não fazem foco na forma de obtenção do mesmo, muitas vezes não provendo nenhum suporte para isso. Além disso, as soluções, de um modo geral, são acopladas ou a um determinado domínio de aplicação, principalmente as soluções baseadas em ontologias, ou à determinada tecnologia de rede, mais comum nos *middlewares*.

O controle da adaptação, via de regra, é específico a determinado contexto ou a determinada aplicação, o que caracteriza a existência de inúmeras oportunidades de estudo e pesquisa na área.

3. Uma Visão Semântica para o Controle da Adaptação na UBICOMP

A idéia central do mecanismo que está sendo proposto é controlar as adaptações quando da tomada de decisões considerando o contexto, tendo por base: (i) informações monitoradas, (ii) informações semânticas e (iii) inferências lógicas a partir destas mesmas

informações. Serão considerados dois tipos básicos de adaptação: (a) adaptação não-funcional: atua sobre a gerência da execução distribuída, operações de mapeamento, re-escalamento, instanciação remota, migração; (b) adaptação funcional: adaptação do código da aplicação, atua sobre a seleção de componentes de software.

O mecanismo de adaptação proposto será utilizado pelas aplicações em tempo de execução. Neste mecanismo teremos, no repositório de adaptação, todas as regras, políticas e ações globais e ações específicas de cada aplicação. Estas informações serão utilizadas pelo mecanismo de adaptação, para a partir das mesmas e das mudanças do contexto inferir de que forma e onde a aplicação deverá ser executada. Para que a aplicação possa ser descrita em alto nível, sem onerar seu desempenho, pode-se pensar em permitir inserção, por parte do desenvolvedor da aplicação, de políticas de adaptação (regras, ações e restrições), utilizando a programação e meta-programação. No modelo de controle de adaptação ao contexto proposto, pretende-se possibilitar uma evolução incremental das especificações de políticas, regras e ações de adaptação. Permitindo a reutilização e a customização destas para o desenvolvimento de novas aplicações *context-aware*. Além disso, deverá possuir mecanismos para administrar situações de conflito quando da tomada de decisões do servidor de adaptação. Assim sendo, o objetivo é o de adaptar os serviços da computação ubíqua ao ambiente sem, ou com mínima, intervenção explícita do usuário.

Um desafio que será perseguido está em determinar onde o controle de adaptação deverá ser efetuado, nas aplicações ou no *middleware*. A proposição prevê um serviço de controle de adaptação que seja utilizado tanto pelas aplicações, quanto pelo *middleware*, de forma Multi-nível Colaborativa [Augustin 2003, Yamin 2004].

Este mecanismo será modelado e prototipado como um serviço de adaptação funcional e não funcional para o *middleware* EXEHDA. Para avaliação do mesmo pretende-se desenvolver softwares na área de Telemedicina Ubíqua e Agropecuária de Precisão.

4. Considerações Finais

Considerando o esforço de pesquisa realizado até o momento, destacaríamos as seguintes desafios para a área de Controle da Adaptação na UbiComp, os quais entendemos inerentes ao esforço de estudo e pesquisa necessário à concepção do mecanismo pretendido para o EXEHDA:

- Como tomar decisões automáticas para adaptar a aplicação, considerando preferências e perfil do usuário, contexto, tipos de dispositivos e recursos em geral?
- Como realizar a definição do que é relevante para a aplicação (elementos de contextos + estados)?
- Como obter e entregar a informação de contexto em tempo de execução?
- Qual o nível oportuno de colaboração entre a aplicação e o *middleware* e como acontece esta colaboração no controle da adaptação?
- Como realocar os recursos de hardware e software para manter a aplicação operacional?
- Como resolver, com mínima intervenção do usuário, situações de conflito na tomada de decisões da adaptação?
- Como diminuir a complexidade e o tempo de desenvolvimento das aplicações adaptáveis ao contexto?

- Como deve ser o mecanismo de controle da adaptação para que possa ser reusável e customizado para diferentes demandas de aplicações.
- De que forma viabilizar todos estes desafios, perseguindo a eficiência tanto das aplicações quanto do *middleware*?

O modelo semântico proposto, na fase atual do trabalho, apresenta a seguinte composição:

- **OntUbi** - Ontologia Básica para a UbiComp. Entidades, atributos e relacionamentos possíveis entre os elementos do contexto ubíquo.
- **OntContext** - Ontologia da Situação de contexto dos seus elementos: estado / situação, identificação do dispositivo, identificação da aplicação / serviço / componente, identificação usuário / login, localização, time(data e horário).
- **OntAdapt** - Ontologia de Adaptação: regras, perfis e preferências, restrições e ações de adaptação para as aplicações; regras / políticas globais e específicas.
- **OntHistAdapt** - histórico das decisões de adaptação realizadas.

Este trabalho contempla a utilização de ontologias, pelo fato destas, enquanto artefatos de software, apresentarem uma maior expressividade na definição e descrição de contexto e de adaptações, por oferecerem a possibilidade de realização de inferências lógicas a partir das informações semânticas e pela facilidade de reutilização e extensibilidade para novas aplicações ou novas situações de contexto.

Referências

- Augustin, I. (2003). Abstrações para uma linguagem de programação visando aplicações móveis conscientes do contexto em um ambiente de pervasive computing. Tese de doutorado em ciência da computação, Instituto de Informática/UFRGS, Porto Alegre, RS.
- Augustin, I., Yamin, A. C., da Silva, L. C., Real, R. A., Frainer, G., and Geyer, C. F. R. (2006). Isadapt: abstractions and tools for designing general-purpose pervasive applications: Experiences with auto-adaptive and reconfigurable systems.
- Costa, C. A., Yamin, A. C., and Geyer, C. F. R. (2008). Toward a general software infrastructure for ubiquitous computing. *IEEE Pervasive Computing*, 7(1):64–73.
- Hilera, J. R. and Ruiz, F. (2006). Ontologies in ubiquitous computing.
- Maciel, R. S. P. and Assis, S. R. (2004). Middleware: Uma solução para o desenvolvimento de aplicações distribuídas.
- Menasce, D. A. and Kephart, J. O. (2007). *Guest Editors' Introduction: Autonomic Computing*, volume 11.
- Warken, N. (2008). Uma contribuição ao controle da adaptação na computação ubíqua. Trabalho individual mestrado ciência computação, UCPEL, Pelotas, RS. Disponível: <http://olaria.ucpel.tche.br/nelsiw/>.
- Yamin, A. C. (2004). Arquitetura para um ambiente de grade computacional direcionada às aplicações distribuídas, móveis e conscientes de contexto da computação pervasiva. Tese de doutorado em ciência da computação, Instituto de Informática/UFRGS, Porto Alegre-RS.
- Yamin, A. C., Augustin, I., Barbosa, J., and Geyer, C. (2005). Exehda-adaptative middleware for building a pervasive grid environment.