

# Localizando Pessoas em um Ambiente Ubíquo para Colaboração\*

Cássia Pereira Nino<sup>1</sup>, Débora Nice Ferrari Barbosa<sup>2</sup>, Jorge Luís Victória Barbosa<sup>1</sup>

<sup>1</sup> Universidade do Vale do Rio dos Sinos, São Leopoldo, RS, Brasil, 93022-000s  
cnino@turing.unisinos.br, jbarbosa@unisinos.br

<sup>2</sup> Centro Universitário La Salle, Canoas, RS, Brasil, 92200-000  
nice@unilasalle.edu.br

**Abstract** Collaborative applications have been explored for decades by several research areas. Problems such as file and workspace sharing have been covered extensively in the literature and solutions for them have been proposed. However, the advent of new technologies regarding mobile devices and context-aware systems, foster the raising of paradigms such as Ubiquitous Computing, bringing together with them new styles of interaction and, therefore, novel opportunities for collaboration that have not been completely investigated so far. Thus, novel solutions that leverage these new elements towards tools that enable more flexible and efficient patterns of collaboration in these scenarios are highly desirable. This article presents MaPS, a framework for collaborative applications that builds on context-awareness to improve the primary steps of collaboration: the search for people and the selection of the best suited communication channels through which the collaboration participants shall interact.

**Keywords:** Framework, Collaborative Applications, Computer Supported Collaboration, Ubiquitous Computing.

**Resumo** Aplicações colaborativas são exploradas há várias décadas por diversas áreas de pesquisa. Problemas como compartilhamento de arquivos e espaços de trabalho foram tratados amplamente na literatura, apresentando-se soluções e alternativas para resolvê-los. No entanto, o advento das novas tecnologias relativas a dispositivos móveis e sistemas cientes do contexto promovem o surgimento de paradigmas como a Computação Ubíqua, trazendo consigo novos estilos de interação e, por conseguinte, novas oportunidades de colaboração que não foram completamente investigadas até agora. Com isso, novas soluções que tirem proveito destes novos elementos no cenário colaborativo são necessárias. Este artigo apresenta o MaPS, um framework para aplicações colaborativas que explora a ciência do contexto de maneira a incrementar os passos iniciais de uma colaboração: a busca por pessoas e canais de comunicação que possam ser utilizados na interação entre os participantes.

**Palavras chave:** Framework, Aplicações Colaborativas, Colaboração Suportada por Computador, Computação Ubíqua.

---

\* Este trabalho recebeu apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq - Brasil (Edital 15/2007 - Universal)

## 1 Introdução

Na década de 70 os microcomputadores surgiram com a promessa de prover um suporte às mais variadas atividades, incluindo-se àquelas destinadas a grupos e organizações, e então, quase uma década após, aparecem as aplicações de groupware. Este tipo de aplicação pode ser definido como "sistemas baseados em computador que suportam um grupo de pessoas envolvidas em uma tarefa, ou objetivo, comum e que provêem uma interface compartilhada para seus participantes" [1].

Neste tipo de aplicação, a comunicação é um elemento chave, pois a interação entre as pessoas é permanente. Soma-se isto ao fato de a computação prover cada vez mais tecnologias para interação entre pessoas e tem-se que hoje, o computador desempenha um papel importante na comunicação entre indivíduos, não sendo exclusivo a esta categoria de sistemas. Contudo, como observado por Stahl [2], o potencial do computador como ferramenta de suporte tende a ser mais proveitoso para grupos do que para indivíduos, uma vez que a comunicação entre os participantes sempre sofreu diversas restrições, mas que agora podem ser flexibilizadas com este tipo de suporte.

O suporte computacional para a colaboração é descrito pelo modelo 3C proposto por Fuks et al. [3], onde são apresentados os três aspectos para o suporte: comunicação, coordenação e cooperação. Seguindo este modelo, em uma colaboração, os participantes devem dialogar (comunicação), organizar-se (coordenação) e trabalhar em conjunto em um espaço compartilhado (cooperação) [4].

E estar ciente da presença de outras pessoas, e encontrá-los, é o primeiro passo para a colaboração. Depois na descoberta, o usuário pode interagir com as pessoas que ele encontra e então formar grupos. Este processo de busca e seleção de usuários é crucial na colaboração [5]. Outro ponto crítico nestes primeiros passos de uma colaboração é o tipo de mídia de comunicação utilizada [3]. Neste cenário, a computação ubíqua, em sua visão de mobilidade livre de usuário e invisibilidade da computação [6], surge como um elemento potencializador, pois suas tecnologias associadas, como dispositivos móveis e adaptação ao contexto [7], permitem incrementar a qualidade do processo de busca por pessoas. Destes aspectos, a mobilidade facilita o acesso do usuário a recursos dos quais ele necessita. Assim, utilizando-se desta tecnologia, criam-se novas oportunidades de interação e colaboração entre pessoas que, sem esta característica, não se encontrariam, por não pertencerem a um contexto social comum [8]. Já a ciência do contexto [9] permite que possam ser utilizados elementos do ambiente no qual os participantes estão, com o objetivo de otimizar os processos de busca. Esta otimização pode ser obtida, por exemplo, em função de informações como localização dos usuários e dispositivos que eles portam. Com isso, o sistema pode configurar-se a estas informações dinamicamente de modo a adaptar-se ao usuário [10], exibindo em um resultado de busca, por exemplo, as pessoas mais indicadas para cada situação. Por sua vez, a invisibilidade garante a utilização de computação pelos usuários de modo, praticamente, imperceptível [11]. Com esta característica, o uso da computação nas atividades rotineiras torna-se natural e intuitivo.

No entanto, quando e como cada um dos participantes toma conhecimento de outras pessoas no ambiente para então iniciar a interação e colaboração é ainda um tópico de pesquisa em aberto [5].

### 1.1 Colaboração e Computação Ubíqua

A noção de colaboração ubíqua ganhou destaque ao longo dos últimos anos. Por sua característica de heterogeneidade de dispositivos e redes sem fio, o ambiente ubíquo apresenta novas oportunidades para o trabalho colaborativo [12]. O aparecimento da computação móvel, o amadurecimento das redes sem fio e a rápida disseminação de dispositivos de computação heterogêneos têm ajudado a formar esse modelo de colaboração. A maior motivação da colaboração ubíqua é permitir uma continuidade do trabalho colaborativo através de um acesso pervasivo à informações compartilhadas, facilidades de comunicação e serviços de groupware.

## 2 Framework MaPS

Atualmente, existem soluções voltadas para o suporte ao desenvolvimento de aplicações colaborativas. Estas soluções vão desde frameworks conceituais até plataformas de serviços que têm como intuito otimizar a criação de aplicações deste tipo. Entretanto, os processos envolvidos nos primeiros passos de uma colaboração, seleção de pares e canais de comunicação, não estão detalhados o suficiente em trabalhos já apresentados [5,13,14]. Este artigo apresenta o framework MaPS (*Matching People to Share*), um framework de suporte no desenvolvimento de aplicações colaborativas em ambientes ubíquos.

MaPS tem como escopo os primeiros passos de uma colaboração (representados na figura 1): a busca por pessoas e por canais de comunicação que possam ser utilizados na interação entre os participantes. Assim, é apresentado a seguir um modelo e uma proposta de arquitetura que visa considerar critérios de pesquisa e informações de contexto para essas buscas por pessoas em um ambiente ubíquo. O framework auxilia no desenvolvimento de aplicações colaborativas, incrementando e definindo estes processos de pesquisa.

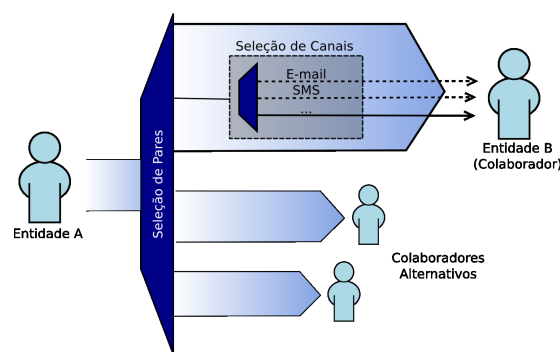


Figura 1. Passos para uma colaboração

Uma das motivações do MaPS é a definição de uma solução que viabilize a sua utilização em diferentes cenários de colaboração em ambientes ubíquos. Contudo, exis-

tem requisitos básicos que devem ser atendidos pelo ambiente no qual o sistema que utiliza o framework será executado. Estes requisitos compreendem funcionalidades não cobertas pelo framework, mas que são necessárias a sua operação, como: (i) um serviço de gerenciamento de perfis de pessoas; (ii) um serviço de contexto; e (iii) um serviço de repositório de ontologia de termos.

## 2.1 Visão Funcional

Os conceitos utilizados para construir as funcionalidades do MaPS (escolha de pessoas e de canais de comunicação) estão exibidos no diagrama da figura 2. Este é uma simplificação do diagrama funcional do framework.

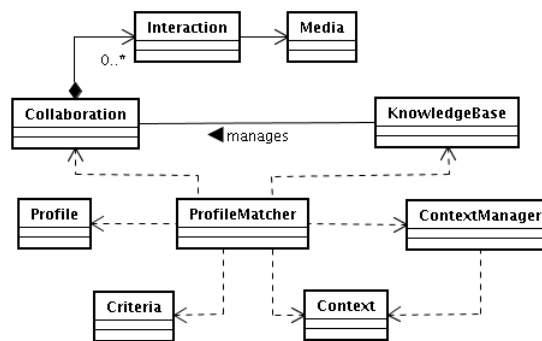


Figura 2. Framework MaPS - Visão Funcional

A construção da seleção de pares é feita através dos conceitos de (i) perfis que contêm informações de usuários (*Profile*), (ii) *matcher* que é um elemento que faz o casamento de perfis (*ProfileMatcher*), e (iii) critérios de *matching* que usam operadores predefinidos para especificar a busca (*Criteria*).

*ProfileMatcher* é o elemento que faz o casamento de *Profiles*. Ele utiliza-se de informações provenientes da *KnowledgeBase*, que representa uma abstração de repositório persistente que centraliza as informações pertinentes aos processos de colaboração. Estas informações estão definidas na fase de identificação dos requisitos funcionais do ambiente colaborativo, como as informações de usuários e de recursos e ontologia de termos. Cada instância concreta de *ProfileMatcher* é responsável por implementar uma heurística distinta para a funcionalidade, sendo a utilização de uma heurística específica determinada por uma configuração do framework ou pelo estado do contexto. Por exemplo, se houver uma informação de localização de usuários, o algoritmo pode levar este dado em consideração na hora de selecionar as pessoas.

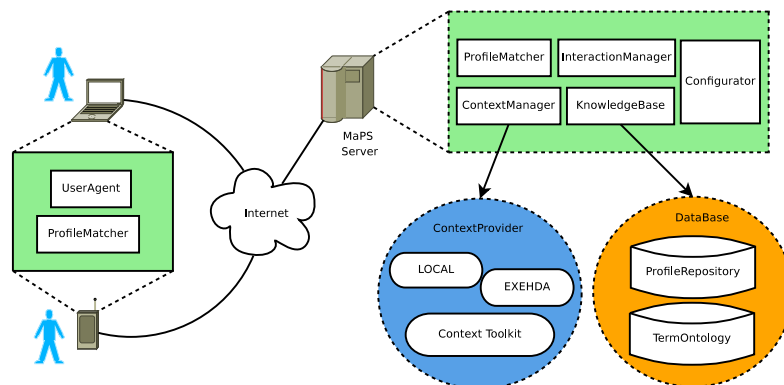
A busca realizada é sempre feita em função de critérios estabelecidos pelo usuário e representados pelo elemento *Criteria*. Neste elemento também estão representados os operadores utilizados pelo framework nas pesquisas. Estes operadores atuam sobre

os valores especificados em cada critério, sendo que os valores dizem respeito a uma informação do perfil do usuário. MaPS define quatro tipo de operadores diferentes:

- CONTAINS: indica que as pessoas que retornarem na consulta possuem o valor especificado no critério.
- LIKE: mostra no resultado da consulta pessoas que possuem ou o valor especificado no critério ou um valor aproximado. Este valor aproximado é disponibilizado por um serviço externo: a ontologia de termos.
- GREATER\_THAN: indica que as pessoas que retornarem na consulta possuem um valor maior do que o informado no critério.
- LESS\_THAN: significa que as pessoas que retornarem na consulta possuem um valor menor do que o especificado no critério.

## 2.2 Arquitetura

A organização apresentada na figura 3 contempla uma seleção multi-nível de *Profiles*. Nela, os clientes são os usuários portando diferentes tipos de dispositivos (desktop, laptop, PDA), nos quais o framework está presente através de dois componentes: *UserAgent* e a instância local de *ProfileMacther*.



**Figura 3.** Visão Arquitetural do MaPS

O *UserAgent* representa a lógica específica da aplicação colaborativa construída, sendo responsável pelas interações com o usuário. A instância local de *ProfileMacther*, por sua vez, implementa o primeiro nível do mecanismo de seleção multi-nível de *Profiles*, sendo acionado pelo *UserAgent* em função das ações realizadas pelo usuário.

Este exemplo de arquitetura baseia-se em uma solução centralizada, onde o servidor possui instâncias dos seguintes elementos: *ProfileMacther*, *InteractionManager*, *Configurator*, *ContextManager* e *KnowledgeBase*. Informações de configuração do framework, como por exemplo: que tipos de algoritmo ele irá utilizar para efetuar as buscas, são registrados no elemento *Configurator*.

Como representado no diagrama, *ContextManager* e *KnowledgeBase* são adaptadores que acessam serviços externos ao framework. Estes serviços são determinados pelo ambiente onde o framework está sendo utilizado. Na organização proposta na figura, *ContextManager* abstrai o acesso aos *middleware* de contexto, exemplificados pelos *middleware* LOCAL [15], EXEHDA [16] e ContextToolkit [17]. Enquanto, *KnowledgeBase* abstrai o acesso a dois tipos de serviços: o repositório de perfis e a ontologia de termos. Como exemplo de uma ontologia de termos, pode-se citar o projeto WordNet [18].

### 3 Protótipo

A utilização do MaPS no desenvolvimento de uma ferramenta de suporte à colaboração, prototipada em Java, é caracterizada no cenário discutido a seguir:

*"Pedro faz parte de uma equipe de desenvolvimento de software em uma empresa com filiais em diversos países. Os participantes desta equipe, por pertencerem a diferentes filiais da empresa, trabalham de forma geograficamente distribuída.*

*Como nova tarefa para esta semana, Pedro deve desenvolver um módulo para um sistema já existente e a comunicação entre estes componentes será feita através de web services. Entretanto, Pedro não domina a utilização de web services, por isso, decide entrar em contato com sua equipe e verificar quem pode lhe auxiliar nesta tarefa."*

Neste cenário, pode-se levantar a seguinte questão: como Pedro pode realizar esta tarefa de pesquisa por pessoas de forma otimizada? Já que sua equipe é numerosa e esparsa. Seria improdutivo entrar em contato com todos, questionando sobre um determinado assunto. Neste caso, uma aplicação colaborativa, que implemente os conceitos de busca por colaboradores auxiliaria Pedro a executar suas tarefas.

A aplicação Looking4 endereça esta perspectiva de CSCW (*Computer Supported Cooperative Work*) pela instanciação do framework MaPS. Especificamente, a aplicação está integrada à uma IDE (*Integrated Development Environment*), o Eclipse<sup>3</sup>, na forma de um *plug-in*. Assim, tem seu acesso facilitado para pessoas que trabalham neste tipo de ambiente de desenvolvimento. O objetivo da aplicação é ajudar seus usuários a encontrarem as pessoas mais indicadas para colaborar com eles. Esta funcionalidade é construída com o auxílio das abstrações propostas no framework MaPS e através de sua API (*Application Programming Interface*).

A figura 4 captura a visão do usuário correspondente à funcionalidade provida na aplicação Looking4 pela instanciação do framework MaPS.

Na tela principal da aplicação é disponibilizado um indicador de status, o qual permite que o usuário reparametrize a operação do framework, controlando seu nível de visibilidade no sistema pela escolha de um *perfil de visibilidade*. A configuração do *perfil de visibilidade* permite restringir que canais de comunicação estão habilitados pelo usuário em uma determinada situação. Nesse sentido, três perfis de visibilidade são pré-definidos no sistema:

- *Disponível*: usuário pode aparecer como resultado em buscas feitas por outras pessoas e também é visível para sua lista de contatos. Neste perfil, tanto as interações assíncronas quanto síncronas estão habilitadas;

<sup>3</sup> <http://www.eclipse.org>



Figura 4. Tela principal do Looking4

- *Privado*: usuário não aparece como resultado em buscas feitas por outras pessoas. Neste estado ele é visível somente para usuários de sua lista de contatos, estando disponível para estes contatos na modalidade assíncrona;
- *Invisível*: usuário não é visível para nenhuma pessoa do ambiente, contudo ele pode visualizar o estado de outros usuários (que estejam configurados para tal).

Adicionalmente, o usuário pode personalizar os perfis existentes ou definir novos perfis de visibilidade. Através deste mecanismo, o usuário pode obter um controle mais preciso sobre quais canais de comunicação estão habilitados para um dado perfil de visibilidade e em que momento outras pessoas poderão comunicar-se com ele no ambiente. Por exemplo, pode restringir as interações, quando no modo *Privado* somente para e-mail e para qualquer pessoa.

A interface da aplicação permite ao usuário gerenciar uma lista de contatos, a qual é construída a partir da funcionalidade de busca provida pelo framework, conforme descrito a seguir. Para cada um dos contatos, o usuário pode executar ações como visualizar o seu perfil e comunicar-se utilizando as indicações de meios de comunicação síncronos (por exemplo, *chat* ou vídeo conferência) e assíncronos (por exemplo, e-mail ou fóruns) que a aplicação sugere. Além destas ações, existem operações básicas de gerenciamento destes contatos como exclusão e inclusão de novas pessoas na lista.

Todos os usuários que acessam o ambiente possuem também um perfil, onde estão cadastrados seus dados de identificação como nome e local de trabalho (filial e setor), os canais de comunicação habilitados (por exemplo: e-mail, telefone e mensagem instantânea), e ainda um conjunto de informações sobre suas preferências e conhecimentos. Este perfil possui ainda dados dinâmicos que não são informados explicitamente pelo usuário, mas que são obtidos em função do contexto em que ele se encontra, como por exemplo localização e dispositivo que está portando.

Voltando ao cenário apresentado no início da seção, ao usar a aplicação Looking4, Pedro define como critério de busca as pessoas que possuem conhecimentos em *web*

*services*. Ao submeter a consulta, o sistema pesquisa nos perfis dos usuários e localiza aqueles que possuem este conhecimento cadastrado e que, potencialmente, podem colaborar com Pedro. Para esta busca, o sistema utiliza, primeiramente, o operador LIKE (como descrito na seção 2.1), com o objetivo de expandir o conjunto de resultados possíveis.

Considerando que o número de resultados que satisfazem a consulta pode ser elevado, refinamentos subsequentes são desejáveis de forma a melhorar a qualidade da resposta e assim, reduzir o ônus de seleção imposto ao usuário da aplicação. Portanto, com o intuito de otimizar a busca feita por Pedro, um outro passo é executado na sequência de filtros montados pela aplicação. O objetivo da busca é a colaboração, interação. Para tanto, só são levados em consideração aqueles usuários que estão disponíveis no ambientes (informação obtida através do *status* de cada participante). E ainda, somente aqueles que possuem canais de comunicação em comum com o usuário que realizou a pesquisa. Com isso, o resultado inicial da consulta é refinado e o usuário tem como resposta um conjunto de resultados adaptados para si.

Por ter ciência de informações de contexto dos usuários, como localização física, a aplicação também considera como resultados mais relevantes aqueles que estão mais próximos fisicamente de Pedro. Dessa forma, acrescenta-se a possibilidade de uma interação face-a-face.

Continuando o cenário apresentado, Pedro pode então encontrar pessoas que o auxiliam em sua tarefa fazendo uso do Looking4.

*"Para localizar pessoas que possam ajudá-lo, Pedro utiliza a sua aplicação Looking4. Através da busca oferecida pelo software, ele informa o valor 'web services' para o filtro 'Tópico'. Com esta informação, o sistema realiza uma pesquisa, verificando nos perfis cadastrados de todos os usuários aqueles que possuem conhecimentos em web services e que, potencialmente, podem colaborar com Pedro.*

*Para esta busca, o sistema encontra inicialmente 238 resultados, o que não considera-se como uma resposta satisfatória, já que o usuário deseja resultados mais consolidados e significativos para si. Portanto, a aplicação dá início a seus processos de filtros e seleção, escolhendo pessoas, dentre as retornadas pela consulta, em função de sua visibilidade no ambiente, canais de comunicação que têm disponível e localização física.*

*Como resultado destes processos, a lista de resultados é filtrada para que atenda às expectativas do usuário, de modo que Pedro recebe como resposta uma relação de 6 pessoas. Todas estas pessoas, como Pedro observa, estão disponíveis no sistema. O primeiro contato a ser exibido na lista é Luciano, que Pedro já conhece mas não sabia que tinha conhecimentos em web services. Por ser uma pessoa que Pedro já teve contato e ainda, encontra-se no mesmo prédio que ele, decide comunicar-se com Luciano através de mensagem instantânea, que a aplicação indicou que eles possuem em comum."*

A funcionalidade de busca de colaboradores oferecida na aplicação Looking4 é construída pelo reuso e personalização das abstrações definidas no MaPS. Nesse sentido, a utilização do MaPS permite reduzir substancialmente o custo de desenvolvimento desse estilo de aplicações. Adicionalmente aos filtros de pesquisa e da proposta de arquitetura já caracterizados, o modelo de perfil de usuário definido no framework é

maleável, permitindo sua personalização para acomodar as informações específicas da aplicação Looking4.

#### 4 Trabalhos Relacionados

Hoje, existem soluções voltadas para o suporte ao desenvolvimento de aplicações colaborativas. Estas soluções vão desde frameworks conceituais até plataformas de serviços que têm como intuito otimizar a criação de aplicações deste tipo.

Projetos como UbiCollab [8] e Collaborator [14] oferecem um suporte aos desenvolvedores, em forma de API ou ambiente de desenvolvimento, como o MaPS. Contudo nenhum dos dois oferece um apoio explícito à funcionalidade de seleção de pessoas, tampouco tratam da escolha de canais de comunicação.

Já Chen e Yang [19] aplicam o conceito de redes peer-to-peer e a tecnologia de mensagens instantâneas para a distribuição de conhecimento em espaços virtuais de colaboração. Segundo os autores, duas grandes barreiras que existem no compartilhamento de informações é a determinação das informações mais relevantes e também dos pares mais relevantes em uma colaboração [20]. No trabalho desenvolvido por eles, estes dois problemas são tratados, já que são considerados como um importante suporte a aplicações colaborativas. Porém, o aspecto de seleção de canais de comunicação não é considerado.

#### 5 Considerações Finais

O framework MaPS, apresentado neste artigo, tem como objetivo auxiliar o desenvolvimento de aplicações colaborativas em ambientes ubíquos. Mais especificamente, a proposta visa incrementar o processo de busca por pessoas e a forma como pode se dar a interação entre os participantes nestes ambientes. E para isso, faz uso de informações de contexto e perfis de usuários, a fim de tornar este processo mais eficiente.

O diferencial do MaPS em relação a outros projetos que tratam do suporte à funcionalidade de seleção de pessoas para uma colaboração está em considerar informações diferenciais, que melhoram a qualidade da busca, retornando potenciais colaboradores que são pertinentes, considerando o contexto em que a busca é realizada. Dados como tipos de canais de comunicação disponíveis, perfis de usuários e localização dos mesmos não são consideradas em sua totalidade nos trabalhos relacionados estudados, tratando-se de um aspecto que distingue MaPS dos demais trabalhos.

Os próximos passos a serem seguidos são o uso do protótipo em uma escala maior, verificando o seu uso e eficiência na utilização frequente pelos usuários.

#### Referências

1. Ellis, C.A., Gibbs, S.J., Rein, G.: Groupware: some issues and experiences. *Communications of the ACM* **34** (1991) 39–58
2. Stahl, G.: Groupware goes to school. In: 8th International Workshop on Groupware: Design, Implementation, and Use, CRIWG 2002. Volume 2440 of Lecture Notes in Computer Science., La Serena, Chile, Berlin: Springer (2002) 7–24

3. Fuks, H., Raposo, A., Gerosa, M.A., Pimental, M., Lucena, C.J.P. In: *The 3C Collaboration Model*. Ned kock edn. Information Science Reference, Texas A&M International University, USA (December 2007) 750
4. Gerosa, M.A., Fuks, H., Lucena, C.J.P.: Suporte à percepção em ambientes digitais de aprendizagem. *Revista Brasileira de Informática na Educação* **11** (November 2003)
5. Zhang, G., Jin, Q.: Research on collaborative service solution in ubiquitous learning environment. In: *Parallel and Distributed Computing, Applications and Technologies*, 2005. PDCAT 2005. Sixth International Conference on. (2005) 804–806
6. Weiser, M.: The computer for the 21st century. *Scientific American* **265**(9) (September 1991)
7. Satyanarayanan, M.: Pervasive computing: vision and challenges. *Personal Communications, IEEE* [see also *IEEE Wireless Communications*] **8** (2001) 10–17
8. Divitini, M., Farshchian, B.A., Samset, H.: Ubicollab: collaboration support for mobile users, Nicosia, Cyprus, *ACM* (2004) 1191–1195
9. Dey, A.K., Abowd, G.D.: Towards a better understanding of context and context-awareness. In: *Conference on Human Factors in Computing Systems*, The Hague, The Netherlands (2000)
10. Barbosa, D.N.F.: *Um Modelo de Educação Ubíqua Orientado à Consciência do Contexto do Aprendiz*. Doutorado em ciência da computação, Instituto de Informática, PPGC/UFRGS, Porto Alegre (Março 2007)
11. Weiser, M.: Some computer science issues in ubiquitous computing. *Communications of the ACM* **36** (1993) 75–84
12. Izadi, S., Coutinho, P., Rodden, T., Smith, G.: The fuse platform: Supporting ubiquitous collaboration within diverse mobile environments. *Automated Software Engineering* **9** (April 2002) 167–186
13. Farshchian, B., Divitini, M.: Ubicollab: improving collaboration with location services. In: *Pervasive Services*, 2005. ICPS '05. Proceedings. International Conference on. (2005) 417–420
14. Bergenti, F., Costico, S., Poggi, A.: A portal for ubiquitous collaboration. In: *Conference on Advanced Information Systems Engineering*, Klagenfurt, Austria (2003)
15. Barbosa, J.V., Hahn, R., Rabello, S., Barbosa, D.N.F.: Local: Um modelo para suporte a aprendizagem consciente de contexto. In: *Simpósio Brasileiro de Informática na Educação*, Brasília, Porto Alegre: SBC (2006) 437–446
16. Yamin, A., Augustin, I., Silva, L.d., Real, R., Geyer, C.: Exehda middleware: aspects to manage the isam pervasive environment. In: *Proceedings of the 25th International Conference of the Chilean Computer Science Society (SCCC 2005)*, Valdivia, Chile, Washington: IEEE Computer Society (2005) 9 pp.
17. Salber, D., Dey, A.K., Abowd, G.D.: The context toolkit: aiding the development of context-enabled applications, Pittsburgh, Pennsylvania, United States, *ACM* (1999) 434–441
18. Miller, G.A.: Wordnet: a lexical database for english. *Communications of the ACM* **38** (1995) 39–41
19. Chen, I., Yang, S.: Peer-to-peer knowledge sharing in collaboration supported virtual learning communities. In: *Advanced Learning Technologies*, 2006. Sixth International Conference on. (2006) 807–809
20. Yang, S.J., Chen, I.Y.: A social network-based system for supporting interactive collaboration in knowledge sharing over peer-to-peer network. *International Journal of Human-Computer Studies* **66** (2008) 36–50