
Um Modelo Multi-agente Descentralizado para Ambientes de Educação Ubíqua¹

Jezer Machado de Oliveira, Solon Rabello, Jorge Luis Victória Barbosa

Programa Interdisciplinar de Pós-Graduação em Computação Aplicada (PIPCA)
Universidade do Vale do Rio dos Sinos (UNISINOS) São Leopoldo – RS – Brasil
jezeroliveira@gmail.com, solonr@unisinos.br, jbarbosa@unisinos.br

Abstract. *This paper presents Global, an ubiquitous, decentralized education model, based on multi-agent systems. Global provides software agents that execute tasks common to ubiquitous learning processes. Besides, global presents itself as a framework - it can be extended either by adding new agents or extending the existing ones. The paper also presents a prototype application, and its use in an educational environment.*

Resumo. *Este artigo propõe o Global, um modelo de educação ubíqua, descentralizado, baseado em sistemas multi-agentes. O Global disponibiliza agentes de software que executam tarefas comuns ao processo de aprendizagem ubíqua. Além de ser um modelo de educação ubíqua, o Global também é uma infra-estrutura similar a um framework, sendo que sua customização é feita através da extensão de seus agentes ou pela adição de novos. O texto também apresenta um protótipo e sua aplicação em um ambiente educacional.*

1. Introdução

A computação ubíqua [Weiser 1999], também conhecida como computação pervasiva [Satyanarayanan 2001], é considerada uma extensão da computação móvel, herdando características como mobilidade e localização. Neste contexto, a educação vem sendo considerada uma importante área de aplicação. A capacidade de um aluno equipado com um dispositivo móvel, de aprender enquanto se desloca por um ambiente, utilizando as informações desse deslocamento para contextualizar e agregar conhecimento traz inúmeros benefícios ao processo de aprendizagem. Atualmente, existem diversas propostas de ambientes de educação ubíqua, como SmartClassroom [Yau 2003], LOCAL [Barbosa et al. 2008a], JAPELAS [Ogata et al. 2005] e GlobalEdu [Barbosa et

¹ Este trabalho recebeu apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq - Brasil, através dos Editais MCT/CNPq 15/2007 (Universal) e MCT/CNPq 27/2007 (Bolsas de Mestrado).

al. 2008b], assim como os trabalhos propostos por [Chen et al. 2007], [Yang 2006] e [Zhang et al. 2005].

As propostas de ambientes de educação ubíqua, em sua maioria, utilizam estruturas centralizadas [Yang 2006]. Contudo, tal abordagem não se mostra adequada a definição de computação ubíqua apresentada por [Satyanarayanan 2001], que entre suas características requer redes móveis, alta disponibilidade, escalabilidade localizada e acesso móvel à informação. Em contrapartida modelos descentralizados suportam mais facilmente tais características.

Em modelos centralizados tradicionais, a comunicação é feita em uma infraestrutura pré-configurada através de um intermediário (servidor) que é responsável pela entrega da mensagem, o que dificulta a criação de redes móveis. Já em modelos descentralizados a comunicação é feita de forma direta entre usuários. Integrando tal característica ao uso de algoritmos de roteamento descentralizados (redes *Ad Hoc* [Royer et al. 1999]) é possível à criação de redes móveis. Além disso, caso um servidor fique inoperante ou inacessível, os elementos gerenciados por ele também ficam ou pelo menos perdem parte de suas funcionalidades. Quando um elemento em um modelo descentralizado fica inoperante ou inacessível somente funcionalidades que dependem dele ficam comprometidas, garantindo uma maior disponibilidade.

Escalabilidade localizada é a capacidade de priorização de interação com usuários mais próximos. Em ambientes descentralizados baseados em redes *Ad Hoc* a própria topologia da rede favorece a escalabilidade localizada, pois ela pode ser controlada através do número de saltos entre os nodos. Em ambientes centralizados, para fazer essa gerência é necessária a participação do servidor o que reduz a escalabilidade. Em modelos descentralizados, devido à ausência de um servidor, as informações são armazenadas no próprio dispositivo, garantindo o acesso móvel à informação.

Além disso, no desenvolvimento de ambientes de educação ubíqua, outros três fatores são relevantes: 1) mesmo com o avanço nas tecnologias de dispositivos móveis, as ferramentas e APIs para desenvolvimento ainda são bastante limitadas em comparação ao desenvolvimento de aplicações *desktop*; 2) o desenvolvimento de aplicações ubíquas é um processo oneroso devido à variedade de tecnologias necessárias para sua realização, tais como comunicação *wireless* e sistemas de localização; 3) softwares de educação ubíqua apresentam funcionalidades comuns, destacando-se a gerência de perfis, a comunicação entre usuários e a tutoria.

Em vista disso, este trabalho propõe um modelo de educação ubíqua, descentralizado, baseado em sistemas multi-agentes, denominado Global. O Global disponibiliza agentes de *software* [Shoham 1990] que executam tarefas comuns ao processo de aprendizagem ubíqua. Sendo que sua customização para criação de novos ambientes de educação ubíqua é feita através da extensão desses agentes ou adição de novos. Além de ser um modelo de educação ubíqua, o Global também é uma infraestrutura, similar a um *framework*. A diferença em relação a um *framework* é que o Global é um modelo completo, não sendo necessária nenhuma especialização ou customização para sua utilização, como é o caso dos *frameworks*. O texto está organizado em cinco seções. A seção 2 apresenta o modelo proposto. A seção 3 aborda detalhes sobre a implementação do protótipo e sua aplicação na avaliação da proposta. A seção 4 discute os trabalhos relacionados. Por fim, a seção 5 apresenta conclusões e trabalhos futuros.

2. Modelo Global

O Global é formado por doze componentes (Figura 1). Interface com Usuário, Persistência, Restritores e Proxys são componentes de suporte representados por API's. Os demais componentes são agentes de *software*. Esses agentes são executados de forma isolada, sendo que cada um disponibiliza um grupo específico de funcionalidades para o apoio da aprendizagem. Cada instância do Global, isto é, cada cópia sendo executada em um dispositivo, tem uma instância desses agentes em execução. O modelo não exige a execução de todos os agentes, sendo que a ausência de um, compromete apenas as funcionalidades que dependem dele.

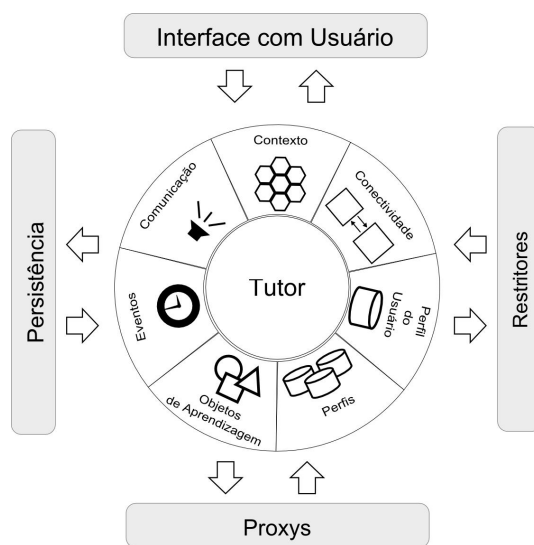


Figura 1 - Arquitetura do Global

2.1 Agente de Conectividade e Proxys

A comunicação entre os agentes é feita pelo agente de Conectividade, através de troca de mensagens, sendo baseadas no padrão FIPA-ACL [FIPA 2009]. Cabe ao agente de Conectividade encontrar a melhor forma para entregar a mensagem, sendo de forma direta entre agentes de uma mesma instância do Global, ou pela escolha de um Proxy que tenha acesso ao dispositivo do agente destinatário, caso sejam agentes de instâncias diferentes.

Um Proxy representa um meio de comunicação possível entre dispositivos, por exemplo, um Proxy pode ser implementado para realizar a troca de mensagens utilizando conexão *bluetooth* e outro pode ser implementado para realizar troca de mensagens através de uma rede Wi-Fi ou 3G. Cada Proxy é responsável por identificar novas conexões, perda de conexões, manter a lista de usuários disponíveis e gerenciar tanto o envio de mensagens como o seu recebimento. O modelo de entrega de mensagens é o de melhor esforço (*best-effort*) [Comer e Yavatkar 1989], no qual não é garantida a entrega, visto que a disponibilidade do usuário pode ser intermitente.

Cada usuário no modelo é representado por um identificador único, formado por um identificador do usuário e um domínio, similar ao *e-mail*. Por exemplo, “carlos@entidade.br”, significa que o usuário se chama “carlos” e o seu domínio é “entidade.br”. A identificação dos agentes dentro do dispositivo funciona de forma similar, onde o

nome do agente é concatenado à identificação, por exemplo, caso uma mensagem seja destinada a “eventos:carlos@entidade.br”, o destinatário é o agente de Eventos do usuário “carlos” do domínio “entidade.br”. Os domínios devem disponibilizar um *webservice* com um serviço de autenticação com acesso na *web*, caso um dispositivo queira comprovar a autenticidade de outro. Essa autenticação é baseada em chave pública e privada [Ferguson and Schneier 2003], na qual a chave privada fica registrada no dispositivo e o serviço presente no domínio disponibiliza a chave pública. Essa autenticação é um processo opcional, pois o dispositivo não tem sempre disponibilidade de acesso à *web*, contudo por estar presente no mesmo ambiente do outro dispositivo, pode garantir sua autenticidade.

2.2 Agente de Contexto e Restritores

Contextos são fragmentos de informação que caracterizam a situação de um participante em uma interação [Dey 2001]. O Global define dois grupos distintos de informações de contexto: 1) Informações de ambiente, tais como a posição no espaço e no tempo; 2) Informações educacionais, que definem elementos agrupadores no modelo educacional, como instituições, cursos, turmas e áreas de interesse.

Devido às inúmeras tecnologias de percepção de ambientes (por exemplo, GPS, câmeras digitais, RFID e sensores de calor), o Global propõe uma estrutura genérica para gerência das características de ambiente baseada em restrições e Restritores. Nessa proposta, um contexto possui uma lista de restrições. Por sua vez, cada restrição é associada a um Restritor que armazena a descrição da restrição. Os Restritores, por sua vez, têm a função de monitorar o ambiente e identificar mudanças de estado nas restrições (a restrição é satisfeita ou não) e notificar seus ouvintes (*listeners*). Cabe ao agente de Contexto associar as restrições do mesmo aos seus respectivos Restritores e monitorar suas mudanças de estado. Um contexto satisfaz sua característica de ambiente, somente se todas as suas restrições são satisfeitas.

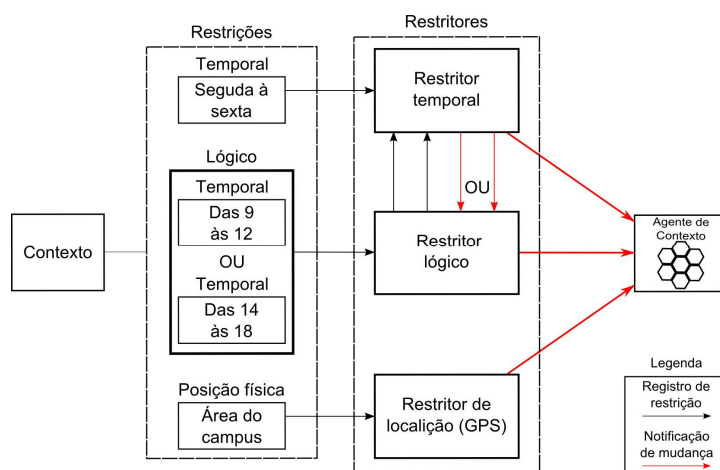


Figura 2 - Exemplo de restrições de ambiente

A Figura 2 mostra um exemplo das restrições de ambiente, onde um contexto possui uma restrição temporal para representar os dias da semana, uma restrição lógica de operação “ou” formada por duas restrições temporais para representar os horários de atuação e uma restrição física que representa a área da atuação do contexto. A vantagem da gerência de contextos por Restritores é que a adição de novos Restritores é simples,

bastando criar a restrição e o Restritor, fazendo seu registro no agente de Contexto, não sendo necessária nenhuma outra alteração no Global.

O contexto educacional é definido pelas informações de aprendizagem (lista de palavras chaves que caracterizam o contexto) e pelos participantes. Além disso, um contexto pode ser: 1) derivado de outro, assim herdando as características de seu contexto parente e acrescentando as suas; 2) centralizado em um usuário, no qual somente o usuário controla a gerência do contexto; 3) descentralizado, quando qualquer usuário do contexto pode gerenciá-lo.

O agente de Contexto também serve como uma ponte para os demais agentes obterem informação sobre os contextos. Através da estrutura de subscrição (*subscription*), do modelo FIPA-ACL, outros agentes podem pedir notificação de mudança de estado dos contextos. Este agente também mantém o registro de todas as mudanças de contexto, inclusive com o horário de entrada e saída. Tendo como base essas informações, é criado um *tracking* (histórico do deslocamento do usuário pelos contextos) que pode ser utilizado posteriormente por outros agentes no processo de ensino e de aprendizagem.

2.3 Agente de Perfil do Usuário e Agente de Perfis

O agente de Perfil do Usuário é responsável pela gerência de informações do usuário e pela sua disponibilização para aos demais agentes do Global. Os perfis seguem o padrão PAPI (*Public and Private Information for Learners*) [PAPI 2009]. Esta escolha foi feita com base na extensibilidade proporcionada pelos elementos *bucket*, na modularidade (divisão lógica das informações em categorias) e na facilidade de integração do PAPI com outros sistemas. A Tabela 1 mostra a organização da estrutura do perfil. *Bucket* no Global são ternas formadas pelo nome do agente, chave e valor. Outros agentes podem utilizar o *Bucket* para guardar informações de seu interesse no perfil do aprendiz.

Tabela 1 - Metadados do Perfil

Categoria	Elementos
<i>Personal Information</i>	<i>ID, Domain, Name, Address, E-mail, Home-page, Phone, Role, Bucket</i>
<i>Security</i>	<i>Password, Public-key, Private-Key, Bucket</i>
<i>Relation</i>	<i>Other-ID, Relation, Context, Bucket</i>
<i>Preferences</i>	<i>Bucket</i>
<i>Performance</i>	<i>General-Interest, Special-interest, Context, Description, Certifier, Metric, Value, Bucket</i>
<i>Portfolio</i>	<i>Object-ID, Bucket</i>
<i>Interests</i>	<i>General-Interest, Special-interest, Description, Goal, Context, Magnitude, Bucket</i>

O perfil é dividido em sete categorias de informações: 1) Pessoais (*Personal Information*): armazena informações básicas do usuário; 2) Segurança (*Security*): armazena as credenciais do aprendiz, como senha, chave pública e privada; 3) Relações (*Relation*): define uma lista com relações entre os usuários do Global, sendo composto pelo ID do usuário da relação e tipo da relação, podendo ser associada a um contexto ou ser uma relação global; 4) Preferências (*Preferences*): descreve as preferências do aluno, com o objetivo de facilitar a sua interação com o sistema; 5) Desempenho (*Performance*): relaciona a história do aluno, listando seu desempenho e avaliações, indicando a área de interesse geral e específica, métrica de avaliação, valor recebido pelo usuário e quem certificou a avaliação, podendo ser associada a um contexto ou ser um desempenho global; 6) Portfólio (*Portfolio*): armazena um conjunto de referências para os trabalhos feitos pelo aluno; 7) Interesses (*Interests*): o grupo de interesses, não é parte do padrão PAPI e foi integrando do padrão LIP [IMS 2009]. A categoria interesses

armazena informações sobre tópicos relacionados as áreas de interesse do aprendiz, conforme as categorias definidas da seguinte forma: uma área de interesse geral (por exemplo, “Matemática”) e uma área de interesse específica, dentro do escopo definido pela área geral (por exemplo, “Teoria dos Grupos”). Os interesses são classificados de acordo com as metas do aprendiz (por exemplo, “aprender” ou “ensinar”). Essas informações são utilizadas pelo agente Tutor para estimular a interação entre aprendizes.

Como o Global é um modelo descentralizado, não existe um local que centralize todas as informações dos usuários. Surge então a necessidade de gerenciar perfis de outros usuários no próprio dispositivo do usuário. Essa é a função do agente de Perfis, que gerencia o acesso ao repositório de perfis de outros usuários. É sua responsabilidade manter os dados desses usuários e requisitar informações ao agente de Perfil do Usuário do outro usuário caso não possua a informação requisitada. Esse agente mantém uma estrutura similar ao *cache* de dados dos navegadores *web*. Quando uma informação sobre um usuário é requisitada e essa informação está disponível, o agente de Perfis contata o agente de Perfil do Usuário do outro usuário, perguntado se existe atualização na informação, caso não exista ou ele não consiga contatar o agente, ele utiliza a informação já existente.

2.4 Agente de Comunicação e Agente de Eventos

O agente de Comunicação gerencia o recebimento e o envio de mensagens. O agente pode ser utilizado tanto pelo usuário, através da interface, quanto por outros agentes do Global. As mensagens são notificações textuais que ficam armazenadas em uma “caixa de entrada” para consulta posterior. O agente de Comunicação suporta tanto o envio de mensagens para um usuário específico, quanto o envio de mensagens para um contexto específico. Caso o destinatário da mensagem não esteja disponível, o agente armazena a mensagem até que ele esteja.

O agente de Eventos tem uma função similar a um calendário no Global, a de suportar o gerenciamento de eventos. Um evento é caracterizado por uma série de propriedades, incluindo palavras-chave, data inicial e contexto. Eventos podem ser definidos para datas futuras (com notificação pela Interface com Usuário na data marcada).

2.5 Persistência e Interface com Usuário

O componente Persistência mantém a persistência das informações quando a aplicação é finalizada. Esse componente disponibiliza uma API para gravação e recuperação de informação, abstraindo as operações de I/O presentes no resto da aplicação. O componente ainda disponibiliza uma estrutura de exportação e importação de dados, com a finalidade de realizar *backup* ou migrações, caso o usuário queira trocar de dispositivo, mantendo as mesmas configurações e informações já existentes no dispositivo anterior. O componente Interface com Usuário é o responsável por disponibilizar os meios para acesso do usuário às funcionalidades do sistema.

2.6 Tutor e Objetos de Aprendizagem

O agente Tutor usa informações dos agentes de Perfil do Usuário, Perfis e Contexto para a inferência de oportunidades pedagógicas. Existem dois tipos de atuação: indicação de objetos de aprendizagem e estímulo à interação entre aprendizes.

O agente Tutor suporta duas formas de indicação de objetos de aprendizagem:

- 1) Indicação para contexto: quando um aprendiz entra em um contexto o agente Tutor analisa os seus objetos de aprendizagem e sugere objetos relevantes para o contexto, cabendo ao aprendiz a escolha de adicionar esses objetos ao contexto ou não;
- 2) Indicação de objeto do contexto: quando um aprendiz entra em um contexto ou recebe notificação de adição de objeto ao contexto, o agente Tutor analisa esses objetos, comparando com as preferências e interesses do aprendiz e sugere a aquisição dos objetos que sejam de seu interesse. Além disso, através das informações disponíveis nos perfis, o agente Tutor propõe interações entre aprendizes. O Global propõe dois tipos de interação: 1) Interesses similares: o agente Tutor encontra aprendizes com interesses similares no mesmo contexto e estimula sua interação; 2) Interesses complementares: o agente Tutor encontra aprendizes que possuem interesses complementares.

O agente de Objetos de Aprendizagem tem a finalidade de armazenar, adquirir e distribuir objetos de aprendizagem. A especificação dos metadados desses objetos no Global segue a norma IEEE/LTSC/LOM [Ieee/ltsc/lom 2009]. A ampla aceitação desse padrão pode ser conferida em [Rigaux and Spyrtos 2007]. A aquisição dos objetos pode ser feita de três maneiras: 1) pelo próprio aprendiz que adiciona um objeto de aprendizagem que considera relevante, tanto para uso pessoal como para compartilhar com outros aprendizes; 2) por indicação de outro aprendiz quando um aprendiz envia um material que considere relevante a outro aprendiz ou grupo de aprendizes (participantes de um contexto); 3) pelo agente Tutor, conforme descrito no parágrafo anterior.

3. Aspectos de implementação e Avaliação

3.1 Implementação

O protótipo do Global foi desenvolvido em J2ME (Java 2 - *Micro Edition*), MIDP 2.0 (*The Mobile Information Device Profile*) e CLDC 1.1 (*Connected Limited Device Configuration*), podendo ser usado em qualquer *smartphone* que suporte essa configuração. Testes foram realizados nos seguintes equipamentos: Nokia N95, HTC Touch HD e HTC TyTN II.

Os agentes de Conectividade, Contexto e Tutor foram implementados seguindo todas as especificações do modelo e os demais agentes possuem implementações de subconjuntos de suas especificações.

Para facilitar o desenvolvimento das APIs de suporte, algumas bibliotecas de terceiros foram utilizadas. O componente de Persistência foi desenvolvido com o auxílio da biblioteca Floggy que é um *framework* de persistência para dispositivos móveis. A Interface com Usuário foi implementada com o LWUIT (*Lightweight UI Toolkit*), um *framework* que oferece componentes visuais para aplicativos J2ME. Além disso, ele disponibiliza uma API padrão L10n, que facilita o suporte a múltiplas línguas.

Dois Proxys foram implementados, um utilizando tecnologia *bluetooth* e ou outro baseado em *WebServices* para Wi-Fi/3G. Quatro Restritores foram desenvolvidos: 1) Temporal: mapeia intervalos de tempo como, horas, dias e semanas; 2) GPS: permite criar restrições por posição física, utilizando API para GPS do J2ME; 3) Lógica: permite operações lógicas, como “e”, “ou” e negação, com as demais restrições; 4) QR Code: utilizado para localização física em ambientes fechados, onde o GPS possui precisão

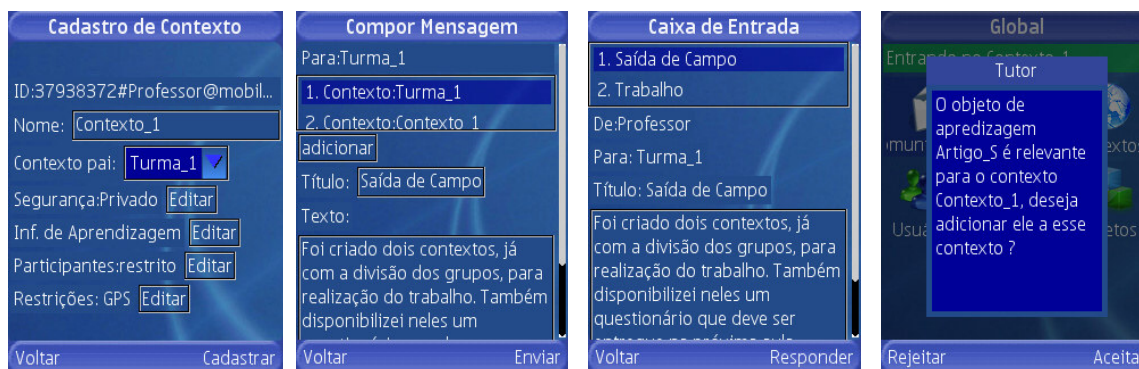
limitada ou em dispositivos sem GPS. Imagens codificadas em QR Code são utilizadas para identificar os ambientes e, através da câmera do dispositivo as trocas de ambiente são mapeadas. Esse Restritor foi criado para avaliar a capacidade de adição de novos Restritores, sem necessidade de alterações dos demais componentes.

3.2 Avaliação

Os cenários vêm sendo utilizados pela comunidade científica para avaliação de ambientes cientes de contexto (conforme abordagem de [Dey 2001]) e ambientes ubíquos (conforme [Satyanarayanan 2001]). Seguindo essa estratégia, foi criado um cenário para teste e avaliação do Global, envolvendo uma saída de campo de uma turma de graduação do Curso de Geologia. O cenário foi implementado com o emulador da suíte *Sun Java Wireless Toolkit*, onde sete instâncias do emulador executando o Global, representaram os sete participantes do cenário. A movimentação dos aprendizes foi simulada através de um *script* XML, representando os deslocamentos no espaço. As coordenadas foram obtidas pelo mapeamento de uma área real. As limitações de comunicação entre usuários foram simuladas via software, pela alteração do Proxy *bluetooth*. A Tabela 2 descreve a dinâmica do cenário e a Figura 3 mostra as telas do protótipo durante a avaliação.

Tabela 2 – Cenário simulado

	Ação
1	Com a ajuda de um software de mapas presente no dispositivo, o professor mapeia e divide a área da saída de campo, cadastrando dois contextos no Global (Contexto_1 e Contexto_2), adicionando características geológicas do terreno de cada área e distribuindo os alunos neles: 1) Contexto_1: Aluno_A, Aluno_B e Aluno_C; 2) Contexto_2: Aluno_D, Aluno_E e Aluno_F. A Figura 3a mostra o cadastro do contexto.
2	O professor envia uma comunicação para todos os alunos com as informações da saída de campo e um questionário com entrega para a aula seguinte. Como nenhum aluno está acessível no momento, essas comunicações ficam armazenadas no dispositivo do professor. A Figura 3b mostra a composição da mensagem.
3	No ônibus, enquanto os alunos se deslocam, as várias instâncias do Global sincronizam suas informações e os alunos recebem a comunicação enviada pelo professor. A Figura 3c mostra o recebimento.
4	Os alunos chegam ao local e se deslocam para as áreas que representam seus respectivos contextos. Quando os alunos entram nessas áreas, seus Tutores são notificados pelos agentes de Contexto. Através da análise das informações adicionadas pelo professor no contexto e as informações cadastradas nos objetos de aprendizagem, o agente Tutor do Aluno_A no Contexto_1 sugere um Artigo_S, que ele já possuía no dispositivo de um trabalho anterior, com informações a respeito do solo em questão. O Aluno_A aceita a sugestão e adiciona o artigo ao contexto. A Figura 3d mostra a notificação.
5	Os agentes de Contexto dos demais alunos do Contexto_1, recebem a notificação de um novo objeto no contexto e notificam seus agentes Tutores. Após a análise os agentes Tutores sugerem a aquisição do Artigo_S (notificação similar a mostrada na Figura 3d). Os alunos aceitam a sugestão e iniciam o processo de aquisição do objeto de aprendizagem.
6	O agente Tutor do Aluno_E, após não ter encontrado nenhum objeto de aprendizagem relevante sobre o Contexto_2, procura por algum perfil que tenha algum conhecimento sobre o tipo de terreno. Mesmo sem acesso aos integrantes do Contexto_1 pela distância, através das informações armazenadas em <i>cache</i> pelo agente de Perfis, o agente Tutor identifica o Aluno_C como conhecedor daquele tipo de solo (notificação similar a da Figura 3d). Como o Aluno_E não tem acesso pelo agente de Comunicação, ele opta por fazer uma chamada telefônica com o <i>smartphone</i> para perguntar ao Aluno_C sobre o PH daquele tipo de solo (uma das questões do questionário).



a) Cadastro de contexto b) Envio de mensagem c) Leitura de mensagem d) Indicação de objeto

Figura 3 - Telas do cenário

4. Trabalhos Relacionados

Atualmente, poucas propostas de modelos de educação ubíqua são descentralizadas ou apresentam alguma característica que possa ser considerada descentralizada. Japelas [Ogata et al. 2005] é um sistema que possibilita a aprendizagem de expressões de tratamento na língua Japonesa. Os estudantes, portando dispositivos móveis, são assistidos na tarefa de identificar as expressões adequadas para cada contexto. Japelas utiliza tecnologias de localização e perfis de usuário. No entanto, diferentemente do Global que foi modelado visando extensibilidade, a abordagem empregada pelo Japelas é voltada para uma aplicação específica, não apresentando, por exemplo, uma estrutura que facilite adição de novos idiomas.

[Zhang et al. 2005] propôs um *framework* para suporte à interação social entre aprendizes em ambientes de computação móvel. O artigo propõe ainda um modelo para a organização das interações, envolvendo três funções: encontro, comunicação e colaboração. A proposta não se enquadra na definição de ubiquidade proposta por [Satyanarayanan 2001], pois não suporta mobilidade ou sensibilidade ao contexto, características viabilizadas pelo Global através da restrição física (GPS) e do agente de Contexto.

[Yang 2006] apresenta um ambiente para a aprendizagem colaborativa ponto a ponto (P2P). O modelo suporta à aprendizagem através da criação de comunidades virtuais onde professores e alunos descobrem, acessam e compartilham recursos. O ambiente não é totalmente descentralizado, necessitando da intervenção de um servidor para a determinação do contexto do aprendiz.

Tendo como base a revisão bibliográfica desse artigo, pode-se afirmar que o Global é a proposta mais adequada a definição de ubiquidade estabelecida por [Satyanarayanan 2001], sendo o primeiro modelo que disponibiliza uma infra-estrutura completamente descentralizada e extensível para o desenvolvimento de novos ambientes de educação ubíqua.

5. Considerações Finais

Este trabalho apresentou o Global, um modelo de educação ubíqua descentralizado, baseado em sistemas multi-agentes. As principais conclusões foram: 1) o protótipo e o experimento comprovaram a viabilidade da proposta; 2) os modelos descentralizados são mais adequados aos conceitos da computação ubíqua definidos por [Satyanarayanan 2001]; 3) agentes de *software* facilitam a criação de modelos extensíveis, proporcionando componentes com fraco acoplamento; 4) a estrutura de restritores facilita a adição de novas formas de percepção de contextos.

Global constitui uma proposta inicial que necessita melhorias. As inferências pedagógicas do agente Tutor devem ser aperfeiçoadas, pois atualmente suportam apenas a interação entre aprendizes e a recomendação de objetos de aprendizagem. Além disso, o protótipo deverá ser finalizado através da implementação dos agentes que suportam apenas parte das funcionalidades do modelo. Novos testes deverão ser realizados para avaliação da proposta. Nesse sentido, destacam-se duas abordagens: 1) avaliação com voluntários, focando aspectos de usabilidade e funcionalidade; 2) avaliação do Global como infra-estrutura extensível, mensurando de forma qualitativa a adição, remoção e adaptação de agentes para a execução de novas funcionalidades.

Referências Bibliográficas

- Barbosa, J.; Hahn, R.; Rabello, S.; Barbosa, D. LOCAL: a Model Geared Towards Ubiquitous Learning. Proceedings of the ACM SIGCSE 2008a.
- Barbosa, J.; Hahn, R.; Rabello, S.; Barbosa, D.; Geyer, C. Learning in Small and Large Ubiquitous Computing Environments. Proceedings of the EUC, 2008b.
- Chen, C.; Li, Y.; Chen, M. Personalized Context-Aware Ubiquitous Learning System for Supporting Effectively English Vocabulary Learning. Computing, 2007.
- Comer, D. E.; Yavatkar, R. FLOWS: Performance guarantees in best effort delivery systems, in Proc. INFOCOM, Ottawa. Canada, 1989.
- Dey, A. K. Understanding and Using Context. Personal and Ubiquitous Computing, Volume 5, Number 1 / February, 2001.
- Ferguson, N.; Schneier, B. Practical cryptography, Wiley. ISBN 0471223573, 2003.
- FIPA-ACL. Foundation of Intelligent Physical Agents-Agent Communication Language. Disponível em: <http://www.fipa.org/specs/fipa00061>. Acesso em: junho de 2009.
- Ieee/ltsclom learning technology standards committee. draft standard for learning object metadata. http://ltscl.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf. Acesso em: Agosto de 2009.
- IMS. Global Learning Consortium, Inc. IMS Learner Information Package - Information Model. v. 1.0. 2001a. Disponível em: <http://www.imsglobal.org/profiles/lipinfo01.html>. Acesso em: Agosto 2009.
- Ogata, H.; Yin, C.; Yano Y.; Oishi Y. JAPELAS: Supporting Japanese Polite Expressions Using PDA(s) Towards Ubiquitous Learning, 2005.
- PAPI draft standard for learning technology. public and private informationfor learners. <http://www.cen-ltso.net/Users/main.aspx?put=407>. Acesso em: Agosto de 2009.
- Rigaux, P.; Spyrtos, N. Selene report:Metadata management and learning object composition in a self elearning network. <http://dcs.bbk.ac.uk/selene/reports>, 2007.
- Royer, E.; Toh, C.-K. A Review of Current Routing Protocols for Ad Hoc Wireless Networks, IEEE Pers. Commun, 1999.
- Satyanarayanan, M. Pervasive computing: vision and challenges. Personal Communications, IEEE, v. 8, n. 4, p. 10–17, 2001.
- Shoham, Y. Agent-oriented programming. Technical Report STAN-CS-1335-90, Computer Science Department, Stanford University, Stanford, CA 94305, 1990.
- Weiser, M. The computer for the 21st century. SIGMOBILE Mob. Comput. Commun. Rev., ACM Press, New York, NY, USA, v. 3, n. 3, p. 3–11, July, 1999.
- Yang, S. J. Context Aware Ubiquitous Learning Environments for Peer-to-Peer Collaborative Learning. Educational Technology & Society, 9, 188-201, 2006.
- Yau, S. SmartClassroom: Enhancing Collaborative Learning Using Pervasive Computing Technology. In II American Society of Engineering Education (ASEE), 2003.
- Zhang, G.; Jin, Q.; Lin, M. A Framework of Social Interaction Support for Ubiquitous Learning. Proceedings of AINA - Volume 2, 2005.