

***Visitor: um sistema de informação dependente da localização**

Helder Pinto

Universidade do Minho, Departamento de Sistemas de Informação, Guimarães, Portugal

helder@dsi.uminho.pt

Rui José

Universidade do Minho, Departamento de Sistemas de Informação, Guimarães, Portugal

rui@dsi.uminho.pt

Resumo

Os sistemas de informação com carácter de mobilidade oferecem novas possibilidades aos utilizadores, nomeadamente em relação aos serviços de que estes podem usufruir. Este tipo de modelo de sistemas de informação possui características que resultam em questões que não eram comuns nos sistemas de informação tradicionais. As fontes de informação são heterogéneas, dispersas, desconhecidas e associadas a localizações físicas. O próprio carácter de mobilidade acarreta consigo as questões que lhe são inerentes, como a localização e inconstância da conectividade. Este artigo descreve um sistema de informação dependente da localização, apresentando as soluções desenvolvidas para lidar com as questões próprias a este tipo de ambiente.

Palavras-chave: sistemas de informação baseados na localização, computação móvel.

1. Introdução

A evolução tecnológica em áreas como as redes sem fios, os dispositivos móveis e os sistemas de localização, proporcionou condições para o surgimento dos sistemas de informação dependentes da localização. Estes sistemas de informação são caracterizados pela mobilidade dos utilizadores e pela existência de fontes de informação heterogéneas, desconhecidas, dispersas e associadas a localizações físicas.

* 2ª Conferência da Associação Portuguesa de Sistemas de Informação, 21-23 Novembro 2001, Évora

Estas características resultam em novos problemas para o desenvolvimento de sistemas de informação, para os quais se têm estudado soluções. O factor mobilidade contribui para a instabilidade de um sistema de informação, pelo facto da ligação à rede por parte dos dispositivos móveis não ser constante, pelas limitações de largura de banda e pela dependência da informação em relação à localização [Pitoura e Samaras 1998]. Além disso, as características das fontes de informação exigem novos modelos para a sua disponibilização e acesso.

Este artigo apresenta um sistema de informação dependente da localização – o sistema Visitor – e evidencia as soluções propostas para os problemas característicos deste tipo de sistemas de informação. O sistema Visitor tem por objectivo ajudar os visitantes do *campus* de Azurém da Universidade do Minho, provendo-lhes informação tão diversa como instruções de orientação, dados sobre os lugares livres nos parques de estacionamento, contactos das entidades do *campus*, eventos e previsões meteorológicas para a região. Esta informação é disponibilizada através de serviços, associados a diferentes zonas geográficas do *campus*, que a aplicação vai descobrindo.

O artigo está organizado da seguinte forma: a secção seguinte descreve o sistema dependente da localização AROUND, do qual faz parte o sub-sistema Visitor; a apresentação do sistema Visitor é feita na secção 3; a secção 4 é dedicada a trabalho relacionado com o tema deste artigo; e termina-se com as referências a trabalho a desenvolver no futuro e às conclusões obtidas.

2. Sistema AROUND

Esta secção descreve a arquitectura do sistema AROUND [José 2001], no qual o sistema Visitor se baseia para descobrir os serviços de que necessita para assistir um visitante do *campus* de Azurém. A arquitectura AROUND é uma proposta de mecanismo de suporte a serviços baseados na localização (*Location Based Services* ou LBS) implementada no projecto homónimo [GET 2001], desenvolvido pelo Grupo de Engenharia da Telecomputação do Departamento de Sistemas de Informação da Universidade do Minho.

2.1. Serviços baseados na localização

Os serviços LBS são uma abordagem para suportar uma ligação sistemática entre a *web* e o mundo físico. No contexto deste trabalho, um serviço LBS é um serviço de rede cuja utilização está explicitamente associada a uma determinada zona geográfica, a qual constitui o âmbito do serviço. Esta associação entre um serviço de rede e uma área no espaço físico resulta

tipicamente de uma de duas situações: o serviço permite alguma interação com o ambiente físico, como por exemplo controlar a temperatura ou a iluminação de uma sala; ou o serviço proporciona o acesso a informação de âmbito local, como por exemplo informações de trânsito ou mapas.

O modelo de serviço LBS pode ser aplicado num conjunto bastante vasto de cenários em que as aplicações necessitem de obter informação relativa à actual localização do utilizador. A grande vantagem do conceito de serviço LBS é a possibilidade de suportar a dependência da localização com base num processo de descoberta de serviços. Este processo fornece o nível de abstracção necessário para permitir o desenvolvimento de sistemas dependentes da localização sem necessidade de lidar com a complexidade e potencial heterogeneidade do ambiente de suporte.

2.2. Arquitectura do sistema

O sistema AROUND possibilita o registo e localização de serviços LBS de forma a permitir a descoberta na Internet de serviços que estejam associados a uma dada localização. Na base da arquitectura do sistema AROUND, pela forma como suporta as restantes componentes, está o modelo que permite representar o âmbito dos serviços LBS, ou seja a área física à qual este se encontra associado. Essa área é definida através de um conjunto de representações simbólicas dos espaços físicos designadas por contextos de localização. Os contextos de localização estão directamente associados a espaços físicos conhecidos e são definidos e estruturados de forma a que a sua percepção seja o mais natural possível, aproximando-se da percepção humana de espaço. Como exemplo de contextos de localização pode-se ter “GuimarãesCidade”, “CentroCidade”, “EdifícioA” ou “SalaB”. Um serviço LBS está associado pelo menos a um contexto de localização e cada contexto de localização pode ter vários serviços a ele associados.

A arquitectura AROUND permite a clientes sensíveis à localização o acesso a serviços cuja cobertura inclua o contexto de localização em que o dispositivo móvel se encontra. A arquitectura é constituída por servidores AROUND, cada um responsável pela manutenção de um determinado número de contextos de localização. Os contextos podem ser ligados entre si através de relações de inclusão ou de adjacência. Estas ligações fornecem uma via para a execução de pesquisas de serviços que atravessem vários contextos e vários servidores AROUND. A arquitectura suporta também vários mecanismos de replicação que permitem aumentar a escalabilidade, tolerância a faltas e desempenho do sistema.

Cada componente da arquitectura ocupa-se de cada uma das fases que vão desde a localização do cliente até à obtenção da informação dependente do contexto de localização determinado. A

figura 1 representa a composição do sistema: a aplicação sensível à localização; o módulo de contextualização, que determina o actual contexto de localização com base na informação prestada por uma ou mais fontes de localização; o serviço de nomes, que indica a referência a um ou mais servidores AROUND para o contexto determinado; o serviço AROUND, que permite o acesso aos serviços registados para cada contexto de localização; e os serviços LBS. No contexto da arquitectura AROUND, um cliente sensível à localização interage com cada uma das componentes anteriormente enunciadas. Para fazer uso dos serviços LBS, a aplicação determina o seu contexto espacial através do módulo de contextualização (cujas fontes podem ser locais – GPS – ou externas – células de redes sem fios, endereço IP, *beacons* de rede, etc.), questiona o serviço de nomes para que lhe seja indicado um servidor AROUND que abranja o contexto determinado e, finalmente, requisita a este último os serviços LBS, indicando o tipo de serviço e os atributos pretendidos. O cliente interage então com os serviços a partir dos quais extrai a informação de que precisa.

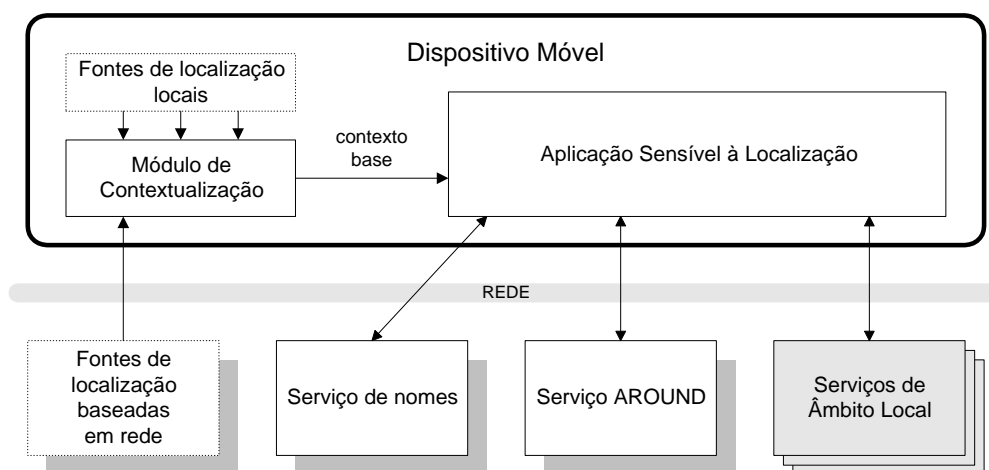


Figura 1 – Modelo funcional do sistema AROUND

Na secção seguinte, é descrito o sistema Visitor, um dos clientes móveis de serviços LBS desenvolvidos no âmbito da implementação da arquitectura AROUND.

3. Sistema Visitor

Na secção anterior foi apresentada a arquitectura AROUND para suporte a serviços LBS. Esta secção descreve em mais detalhe o sistema Visitor, uma das aplicações sensíveis à localização desenvolvidas na esfera do projecto AROUND. Começa-se por apresentar uma visão geral da aplicação e do ambiente no qual se enquadra, prossegue-se com a descrição dos serviços de que

faz uso e dos contextos de localização a que estão associados, continua-se com o retrato do *interface* com o utilizador e termina-se com a descrição do seu comportamento em face das alterações no seu ambiente computacional.

3.1. Visão geral

A aplicação Visitor é um sistema de informação que, partindo de um conjunto de serviços LBS disponibilizados no *campus* de Azurém, fornece informação sobre parques de estacionamento, programa de eventos, orientação no campus, serviço de directoria e previsão meteorológica.

A aplicação, destinada ao sistema operativo Windows CE 2.11, foi implementada em Java, usando o ambiente de desenvolvimento PersonalJava [Sun Microsystems 2000] e foi optimizada para o *palmtop* HP Jornada 680. Este dispositivo móvel inclui um amplo visor com 640x240 *pixels* e 256 cores, um teclado e uma *slot* PCMCIA (tipo II) utilizada para ligar placas IEEE 802.11. Estas permitem a comunicação sem fios com as restantes componentes da arquitectura AROUND, usando uma infraestrutura de rede sem fios WaveLAN, instalada em algumas zonas do *campus* de Azurém. Esta infraestrutura de rede é também utilizada como fonte de localização para a aplicação Visitor. O módulo de contextualização anexo à aplicação consulta periodicamente um conjunto de servidores de localização existentes na rede local de modo a determinar qual o contexto de localização em que a aplicação se encontra.

3.2. Contextos de localização e serviços LBS utilizados

Cada um dos serviços utilizados pela aplicação está associado a um determinado contexto de localização de entre os definidos para o *campus* de Azurém. A hierarquia destes contextos, incluída na estrutura desenvolvida para o projecto AROUND, é representada pela seguinte figura:

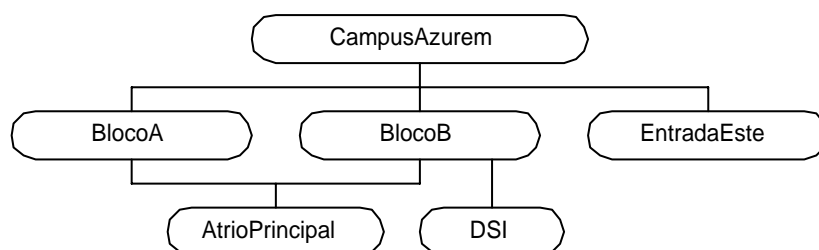


Figura 2 – Contextos de localização do *campus* de Azurém

O contexto *CampusAzurem* representa o nível mais alto da hierarquia; dentro deste, no segundo nível, definiram-se os contextos *BlocoA*, *BlocoB* (estes dois associados aos dois principais edifícios do *campus*) e *EntradaEste* (uma das entradas do *campus*). Os contextos *AtrioPrincipal* e *DSI* estão situados no nível inferior da hierarquia. Cada serviço, ao ser registado, é associado a um destes contextos, definindo assim o âmbito geográfico no qual vai estar disponível.

Associados a cada um destes contextos de localização definidos para o *campus* de Azurém estão diversos serviços LBS, de conteúdo e objectivos variados, cuja informação a aplicação Visitor utiliza afim de prestar assistência ao visitante do *campus* de Azurém. Mais concretamente, foram disponibilizados os seguintes serviços:

- *Directory* (com serviços disponíveis nos contextos *CampusAzurem* e *DSI*), que fornece um serviço de directoria, apresentando os contactos das pessoas e entidades localizadas em cada um dos contextos aos quais está associado, mediante a indicação de uma palavra-chave de procura;
- *Events* (serviço disponibilizado apenas para o contexto *CampusAzurem*), que tem como finalidade prover informação sobre eventos a decorrer no *campus*;
- *Guidance* (com serviços associados a todos os contextos de localização), que oferece informação de localização de diversas entidades nos edifícios do *campus*, orientando inclusive o visitante nos passos a dar até à referida entidade;
- *LocalDescription* (com serviços associados a todos os contextos de localização), que presta a descrição do local onde o visitante se encontra, podendo ir de uma simples descrição textual a um mapa do local;
- *Parking* (disponível no contexto de localização *CampusAzurem*), que informa o utilizador dos parques de estacionamento existentes no *campus*, da sua localização e dos lugares livres existentes em cada um;
- *Weather* (acessível apenas no contexto *CampusAzurem*), que fornece informação sobre as condições meteorológicas actuais e previsões para os dias seguintes.

O *interface* de cada um destes serviços foi desenvolvido com o objectivo de manter um carácter geral, de modo a poder ser também utilizado noutros serviços do mesmo tipo, associados a outros contextos de localização e cuja implementação possa variar, mas que provejam o mesmo género de informação.

3.3. Interface com o utilizador

O *interface* da aplicação cliente com o utilizador é constituído por três áreas: *menus* temáticos, área de visualização e barra de estados (ver figura 3).

O *menu* principal (*Visitor*) possibilita ao utilizador controlar a ligação da aplicação à rede, a actualização da informação apresentada e a gestão da *cache*. Os restantes *menus* (*WhereAmI*, *Parking*, *Directory*, *Events* e *Weather*) são de carácter temático, podendo estar associados a mais que um tipo de serviço, e destinam-se a despoletar a visualização da informação provida pelos diversos serviços LBS.

A área de visualização é dedicada à apresentação da informação prestada pelos serviços. Para o efeito, é utilizada a componente ICEBrowser [IceSoft 1999], a qual permite visualizar informação em formato HTML. A aplicação obtém a informação dos serviços, formata-a em HTML e apresenta-a posteriormente usando a referida componente.

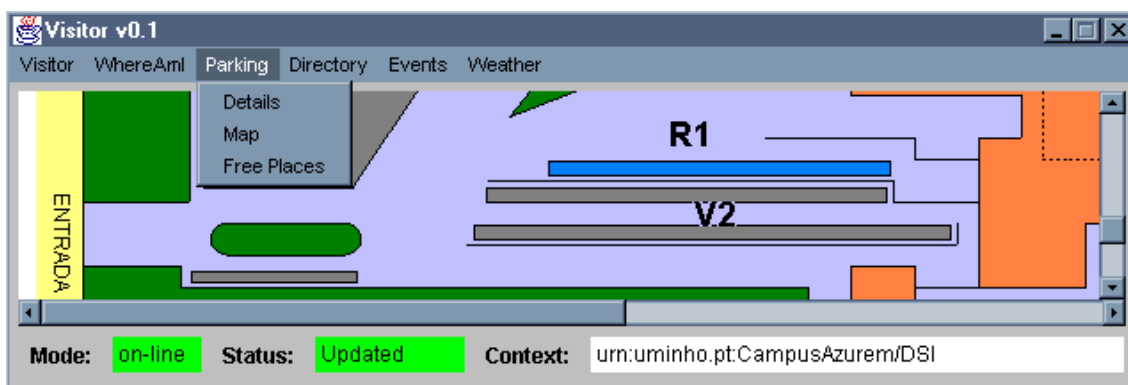


Figura 3 – Aspecto da aplicação Visitor

A barra de estados, na zona inferior, é utilizada para notificar o utilizador do estado da aplicação. Esse estado pode variar em função da conectividade (ligada ou desligada), do comportamento da aplicação e do contexto de localização no qual se encontra.

3.4. Sensibilidade ao contexto

No desenvolvimento de uma aplicação sensível ao contexto (neste caso à localização), há questões específicas a ter em consideração. Tomando como exemplo um utilizador que vai consultando o mapa da zona pela qual se desloca, surgem as seguintes interrogações [Acharya et al. 1995]: deve o cliente actualizar imediatamente o mapa à medida que o contexto de localização se altera ou deve ser o utilizador a dar essa ordem? Como deve o utilizador tomar

conhecimento de que o mapa que está a consultar já não é válido para o contexto onde se encontra? Como deve o cliente reagir no caso de deixar de existir ligação ou de não conseguir determinar o contexto de localização?

No caso da aplicação *Visitor*, estas questões também foram consideradas. O seu comportamento depende da existência de ligação, da localização e da existência de serviços LBS para o contexto determinado. A cada uma destas variáveis do contexto da aplicação correspondem estados pelos quais a mesma passa à medida que determinados eventos ocorrem. O diagrama de estados da figura 4 complementa as sub-secções seguintes onde é descrito em mais detalhe o comportamento da aplicação à luz de cada uma das variáveis de contexto.

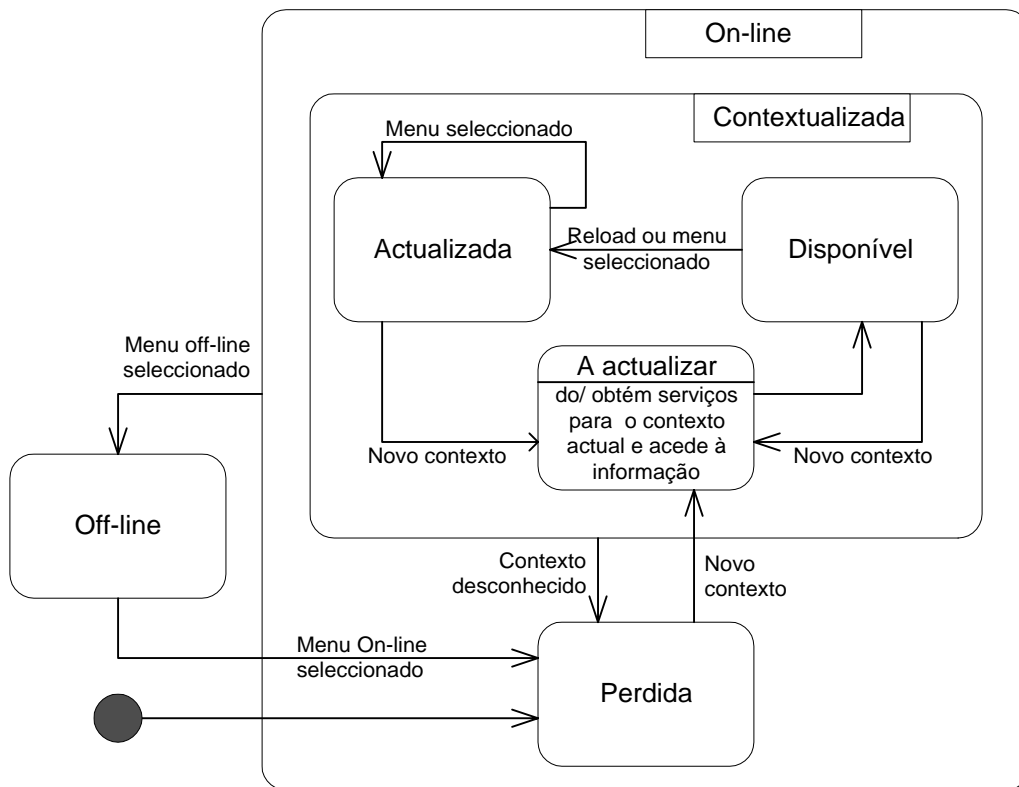


Figura 4 – Diagrama de estados UML da aplicação *Visitor*

Sensibilidade à existência de ligação

A aplicação *Visitor* foi desenvolvida com o objectivo de utilizar serviços LBS *on-line*. No entanto, caso a rede de comunicações apresente problemas na ligação aos serviços, deve ser possível ao utilizador visualizar a informação anteriormente obtida sem a aplicação ter de estar ligada à rede. De modo a permitir uma maior flexibilidade, a aplicação suporta os modos de

funcionamento *on-line* e *off-line*. O seu funcionamento é iniciado, por defeito, em modo *on-line*. No entanto, o utilizador poderá seleccionar o modo desejado através do item *Mode*, do *menu Visitor*.

Caso o utilizador deseje ou só possa visualizar a informação em modo *off-line*, passa a estar dependente da *cache* da aplicação, que é alimentada pelos ficheiros HTML gerados aquando da utilização dos serviços em modo *on-line*. Estes ficheiros são mantidos durante a execução da aplicação, o que permite que, em modo *off-line*, a informação anteriormente obtida possa ser novamente visualizada. Os ficheiros só são eliminados se o utilizador o ordenar, através do item *Clear Cache* do *menu Visitor*. Sempre que o modo *off-line* é seleccionado, a aplicação verifica, para cada serviço, os ficheiros que existem em *cache*. Caso não tenha sido encontrado qualquer ficheiro, a *cache* encontra-se vazia, impedindo à aplicação o acesso a qualquer informação. Se existir algum ficheiro em *cache*, o *menu* do respectivo serviço é activado e a aplicação permite a consulta da informação anteriormente obtida.

Sensibilidade à localização

Quando no estado *on-line*, a aplicação está constantemente a tentar obter informação sobre a localização actual. O estado inicial da aplicação é *Perdida*, pois não tem conhecimento do contexto de localização em que se encontra. Neste estado, os *menus* dos serviços estão desactivados. Tendo sido recebido o primeiro contexto por parte do módulo de contextualização, a aplicação passa ao estado *Contextualizada*, sendo despoletado o processo de obtenção de serviços, constituído pelo contacto com o serviço de nomes, onde é obtida a referência ao servidor AROUND e pela requisição a este último dos serviços desejados.

A qualquer momento, quando no estado *Contextualizada*, a aplicação pode deixar de receber informação sobre o contexto de localização, tanto por ter perdido ligação como por ter entrado numa rede que não oferece mecanismos de contextualização. Neste caso, a aplicação entra no estado *Perdida* e não podem ser feitas nenhuma asserções sobre a informação disponível no momento.

Sensibilidade à existência de serviços

Quando no estado *Contextualizada*, a aplicação está em condições de proceder à obtenção de serviços. Este estado tem três sub-estados: *A actualizar*, *Disponível* e *Actualizada*. A aplicação entra no estado *A actualizar* quando ocorre uma mudança no contexto corrente, o que obriga a que se proceda à descoberta e utilização dos serviços existentes para o contexto determinado. A aplicação passa então a ter informação disponível (estado *Disponível*), ou seja a informação

disponibilizada por cada serviço foi descarregada para a aplicação mas ainda não foi apresentada. Os *menus* que fazem referência aos serviços obtidos passam a estar activos. Nenhum dos eventos e transições de estado anteriores afecta a informação que está a ser apresentada ao utilizador. Isto impede mudanças não solicitadas em informação na qual o utilizador possa estar ainda interessado. No entanto, há uma notificação de que a informação apresentada naquele momento já não é válida. Se o utilizador actualizar a informação que estava a visualizar ou consultar a informação sobre outro serviço, através do *menu* respectivo (*reload*), a aplicação passa ao estado *Actualizada*, o que significa que a informação que está a ser visualizada é a mais recente. Pode acontecer que, enquanto o utilizador visualiza determinada informação, seja recebido um novo contexto de localização. Este evento faz com que a aplicação passe novamente ao estado *A actualizar*, sendo reiniciada a obtenção dos serviços para o contexto determinado.

Nesta secção foi apresentado a aplicação Visitor, focando os seus objectivos, o ambiente em que se insere, os serviços de que faz uso, o interface com o utilizador e a forma como se comporta face às alterações do ambiente. Foi possível observar algumas das especificidades das aplicações sensíveis à localização e de ligação instável e a forma como estas foram tratadas no caso em estudo.

4. Trabalho relacionado

Têm sido feitos diversos trabalhos no âmbito dos sistemas de informação destinados a ambientes móveis cujo comportamento dependa da localização.

No projecto GUIDE [Cheverst et al. 2000], foi implementado um sistema que fornece informação turística a visitantes da cidade de Lancaster. Um visitante com equipamento adequado obtém informação turística relativa ao local em que se encontra. Para além de considerar a variável localização, este sistema também condiciona o seu comportamento face a outros factores tais como a hora do dia ou as condições meteorológicas.

Outro trabalho é o desenvolvido no projecto Cooltown [HP Labs 2001], que consiste numa plataforma para aplicações baseadas na localização com o objectivo de suportar uma correlação sistemática entre entidades do mundo físico e as respectivas páginas *web*. O cenário da utilização desta plataforma pode ser aquele em que pessoas se movem com dispositivos móveis munidos de sensores, e.g. infra-vermelhos ou leitores de código de barras, e recolhem URLs de serviços, outras pessoas e objectos.

A abordagem utilizada nestes sistemas para associar as fontes de informação à localização é baseada em fortes pressupostos sobre aspectos-chave do ambiente computacional, e.g. rede

disponível ou tecnologias de posicionamento. Estes pressupostos simplificam o domínio do problema, mas restringem a aplicabilidade deste tipo de abordagem a outro tipo de cenários. O que distingue principalmente a nossa abordagem é o uso de mecanismos genéricos de descoberta de serviços que suportem a associação entre a localização física e os recursos computacionais.

Outro trabalho [Imielinski e Viswanathan 1994] no qual são explorados vários aspectos dos serviços dependentes da localização foi desenvolvido na Universidade de Rutgers, pelo grupo DATAMAN. O ambiente é baseado na WWW, com serviços de informação constituídos por páginas presentes em servidores associados a determinadas áreas. Também este trabalho difere do descrito no presente artigo pois sendo o cliente um *browser*, esta abordagem tem limitações quanto à possibilidade do cliente responder a eventos despoletados por entidades externas que não o utilizador, e.g. a mudança de contexto de localização não pode ser notificada ao cliente em tempo útil, não sendo possível informar o utilizador da validade da informação em visualização.

5. Trabalho futuro e conclusões

5.1. Trabalho futuro

Uma limitação que se observa no sistema de informação apresentado neste artigo prende-se com o facto da sua utilização estar restrita aos tipos de serviços para os quais está preparado. Caso aparecessem novos tipos de serviços (por exemplo, um serviço do restaurante universitário), a aplicação não poderia fazer uso dos mesmos.

Qualquer aplicação que faça uso de serviços de informação baseados na localização deve possuir atributos que lhe permitam lidar com a grande heterogeneidade que se verifica neste tipo de ambientes. Deve ser possível à aplicação fazer uso de todos os tipos de serviços de informação existentes na área pela qual se vai movendo e não se restringir apenas a um conjunto previamente determinado.

O trabalho a realizar futuramente consiste em desenvolver uma infraestrutura que atribua um elevado nível de flexibilidade e adaptabilidade às aplicações sensíveis à localização. Ou seja, permitir que as aplicações tenham acesso a qualquer serviço LBS, não necessitando de ter conhecimento prévio deste. Por exemplo, no caso da aplicação apresentada neste artigo, deixaria de haver uma relação directa entre os *menus* temáticos e os serviços LBS e passariam a existir *menus* dinâmicos, cujo conteúdo variaria à medida que fossem descobertos novos serviços e que outros deixassem de estar acessíveis. Desta forma, a aplicação que faça uso de

serviços LBS não fica limitada a um determinado conjunto de serviços, mas torna-se flexível ao ponto de descobrir e disponibilizar qualquer tipo de serviço que esteja associado à área onde se encontre a aplicação.

5.2. Conclusões

Os sistemas de informação dependentes da localização possuem características que resultam em questões que não são comuns nos sistemas de informação tradicionais. A heterogeneidade e dispersão das fontes de informação e a associação destas a localizações físicas, assim como a inerente condição de mobilidade dos utilizadores, colocam desafios para os quais se têm estudado soluções.

O trabalho apresentado neste artigo focou os problemas resultantes das especificidades dos sistemas de informação sensíveis ao contexto. As variáveis de contexto consideradas neste trabalho são: a localização (a aplicação reage a mudanças na localização); a existência de serviços LBS (a informação disponibilizada depende dos serviços LBS existentes em determinada zona geográfica); e a existência de ligação. As soluções desenvolvidas para lidar com cada uma destas variáveis constituíram os resultados deste trabalho. Identificaram-se limitações quanto à flexibilidade do sistema desenvolvido, nomeadamente em relação à capacidade de lidar com serviços LBS cujo tipo seja desconhecido, sendo este tema objecto de trabalho futuro.

Este trabalho foi realizado no âmbito do projecto AROUND suportado financeiramente pela FCT (PRAXIS/P/EEI/14267/1998).

6. Referências

Acharya, A., B. R. Badrinath, T. Imielinski e J. C. Navas, *A WWW-based Location-Dependent Information Service for Mobile Clients*, Rutgers University, 1995.

Cheverst, K., N. Davies, K. Mitchell e A. Friday, "Experiences of Developing and Deploying a Context-Aware Tourist Guide: The GUIDE Project", *6th Annual International Conference on Mobile Computing and Networking*, Boston, 2000.

GET, *AROUND – Suporte a Serviços Internet de Âmbito Local*, <http://www.dsi.uminho.pt/get/around>, Departamento de Sistemas de Informação, Universidade do Minho, 2001.

HP Labs, CoolTown Web Site, <http://www.cooltown.hpl.hp.com>, 2001.

IceSoft, *ICE Browser*, <http://www.icesoft.no/icebrowser5/index.html>, IceSoft, 1999.

Imielinski, T. e S. Viswanathan, “Adaptive Wireless Information Systems”, *SIG in Data Base Systems Conference*, Tokyo, 1994.

José, R., A. Moreira, F. Meneses e G. Coulson, “An Open Architecture for Developing Mobile Location-Based Applications over the Internet”, *6th IEEE Symposium on Computers and Communications*, Hammamet, Tunisia, 2001.

Pitoura, E. e G. Samaras, *Data Management for Mobile Computing*, Kluwer Academic Publishers, 1998.

Sun Microsystems, *PersonalJava Application Environment*,
<http://java.sun.com/products/personaljava>, Sun Microsystems, 2000.