

5 Conclusões e Trabalhos Futuros

Através da experiência obtida com o desenvolvimento do aplicativo *Nita* e do estudo o domínio das ACBL, obtivemos os conceitos-chaves e os componentes básicos presentes nas arquiteturas de várias aplicações descritas na literatura. Como resultado, desenvolvemos o *FLoCS*.

FLoCS oferece uma infra-estrutura básica para o desenvolvimento de ACBL, com vários pontos de flexibilização que podem ser customizados de acordo com as necessidades da aplicação. Além disso, todos esses pontos possuem uma ou mais implementações-padrões que podem ser reutilizadas. A reutilização dessas funcionalidades previamente desenvolvidas resulta na redução do tempo de desenvolvimento, possibilitando que novas aplicações sejam facilmente criadas a partir deste framework.

O framework propõe uma generalização do conceito de “localidade (região) simbólica”, tornando-a uma entidade ativa que pode gerar eventos de notificação para as entidades clientes, bem como gerenciar as mensagens que lhe foram associadas. Isso não é feito pela maioria dos demais trabalhos, que adotam apenas uma versão passiva do conceito “localidade”.

No estudo de casos, procuramos demonstrar a flexibilidade do framework e a sua generalidade para o desenvolvimento de novos serviços baseados em localização. Verificamos que o uso do *FLoCS* permite um alto grau de reutilização de código, além do reuso de projeto. Vale ressaltar que, além do grau de reutilização de código, outro fator que deve ser levado em consideração para a avaliação de um framework é a complexidade das classes que tiveram de ser criadas para estender as interfaces/classes abstratas. Infelizmente, as métricas para medir tal complexidade extrapolam o escopo deste trabalho, mas acreditamos que o esforço para criar essas classes é menor ou, no máximo, igual ao esforço necessário sem o uso do *FLoCS*.

Se as aplicações instanciadas como estudo de casos fossem criadas a partir dos frameworks apresentados nos trabalhos relacionados (*Nimbus* e *LSM*), obter-

se-ia apenas a flexibilização da tecnologia de localização a ser utilizada, sem contemplar outros aspectos, tais como comunicação, persistência de dados, associação de mensagens a localidades, etc. Em compensação, haveria incorporação de outras funcionalidades úteis, tais como troca de tecnologia de localização em tempo de execução ou o apoio para *handoff*, que não são oferecidas pelo *FLoCS*.

Os pontos adaptáveis do framework foram testados quanto à sua flexibilidade, criando implementações para diferentes tecnologias. No estudo de caso do *BuddySpace*, por exemplo, onde a mudança da tecnologia de localização utilizada foi feita sem grandes dificuldades, ficou provado que diferentes implementações desses pontos podem ser facilmente adicionadas ao framework sem nenhum efeito colateral para o restante do sistema. Contribui também para isso o fato de a arquitetura do *FLoCS* ser particionada em serviços, de modo que alterações nas classes de um serviço não afetam outras partes do sistema. Além disso, esse particionando permite que o programador se concentre no desenvolvimento de cada um dos serviços em separado, sem se preocupar de como estes vão se relacionar. O framework se incumbem disso.

Apesar do foco do framework ser a instanciação do servidor da aplicação, verificamos no estudo de casos que o provimento de dois serviços importantes (comunicação e persistência de dados) facilitou também o desenvolvimento do cliente da aplicação.

Naturalmente, poder-se-ia usar também qualquer outro framework para sistemas colaborativos ou Sistemas de Informação Geográfica (GIS), para desenvolver uma aplicação de comunicação baseada em localização, uma vez que, essencialmente, uma ACBL é uma combinação de tais sistemas. No entanto, por tratar os aspectos de comunicação e de localização de forma integrada, acreditamos que o framework *FLoCS* fornece um apoio mais adequado ao desenvolvimento de ACBLs.

O desenvolvimento do *FLoCS*, no entanto, é apenas o primeiro passo no sentido de apoiar a implementação de serviços/aplicativos baseados em localização, podendo ainda ser estendido e aprimorado em vários aspectos. Seria importante instanciar um número maior de aplicações para que se possa refinar as APIs dos pontos adaptáveis. Devem ser desenvolvidas também novas implementações desses pontos adaptáveis para que os desenvolvedores tenham

mais funcionalidades à sua disposição, como por exemplo, interface para outros protocolos de comunicação. Além disso, os serviços de comunicação e persistência de dados deveriam também ter uma implementação em J2ME⁷, de modo a estimular o desenvolvimento de clientes para dispositivos móveis mais limitados, como celulares, *handhelds* e etc. Outra importante contribuição para o framework seria a criação de um serviço que provesse um *proxy* para cada cliente móvel, cuja função seria lidar com os problemas inerentes a um ambiente sem fio, como baixa largura de banda, desconexão, *handover*, etc.

A experiência adquirida com o *FLoCS* servirá como base para o desenvolvimento de novos frameworks e *middlewares*. Ao invés de se ter um framework que após ser instanciado vai dar origem a apenas uma única aplicação, o próximo passo seria a criação de uma plataforma cujos serviços, dados dos usuários, informação de localização e etc, seriam compartilhados por vários tipos de aplicações. Seria a evolução do *FLoCS* para *PLoCS (Platform for Location-based Communication Services)*.

⁷ *Java 2 Micro Edition* é uma linguagem de programação da *Sun Microsystems* para dispositivos móveis. Maiores informações podem ser obtidas no site <http://java.sun.com/j2me/>.