

1 Introdução

A Computação Móvel desempenha um papel crescente nos sistemas em função dos avanços na computação pessoal e nas tecnologias de comunicação (Varshney & Vetter, 2000). Com a Computação Móvel, cresce também a popularidade de tecnologias sensíveis ao contexto, as quais possibilitam aos sistemas monitorar e utilizar informações relevantes que provêm do ambiente ou do usuário. Tais tecnologias são importantes para o desenvolvimento de aplicações que se adaptam a mudanças de contexto (Coutaz et al., 2005; Mitchell, 2002a). Através da característica de sensibilidade ao contexto, torna-se possível que os usuários móveis explorem os recursos da computação e as oportunidades de colaboração enquanto se movimentam através de regiões físicas distintas.

Segundo a definição de Dey e Abowd (1999), contexto é qualquer informação que pode ser utilizada para caracterizar a situação de uma pessoa, um lugar ou um objeto relevante para a interação entre um usuário e uma aplicação. As aplicações móveis são executadas em contextos extremamente dinâmicos. Enquanto um dispositivo se movimenta, sua localização muda constantemente e, em consequência, mudam os serviços e os outros dispositivos acessíveis a ele. As aplicações móveis também precisam tratar de variações frequentes em seus contextos de execução, tais como, mudanças de temperatura, redução da bateria disponível no dispositivo móvel, mudanças na localização ou capacidades do dispositivo, grau de proximidade a outros usuários, etc. Portanto, as aplicações móveis devem ser sensíveis ao contexto a fim de fornecer uma qualidade de serviços razoável para seus usuários (Capra, et al. 2001).

Por outro lado, o tratamento de exceções é essencial para incorporação e estruturação das atividades de tolerância a falhas em qualquer sistema de software confiável. Particularmente, o tratamento de exceções em aplicações móveis sensíveis ao contexto é uma tarefa não-trivial em consequência de características

intrínsecas destes sistemas, como mobilidade, abertura¹, comunicação assíncrona, aumento de imprevisibilidade e falta de mecanismos modulares apropriados para propagação de erros (Iliasov & Romanovsky, 2005a; Tripathi et al., 2005). Portanto, para o desenvolvimento de aplicações móveis confiáveis, é fundamental que o tratamento de exceções nestes sistemas possa satisfazer requisitos específicos associados com sensibilidade ao contexto.

1.1. Problema

Tratamento de erros é uma tarefa muito mais desafiadora em aplicações móveis. O que se observa na prática é que a maior parte das aplicações utilizam apenas o mecanismo de tratamento de exceções fornecido pelas linguagens de programação subjacentes. Evidentemente, o tratamento de exceções em aplicações móveis deve ser integrado ao mecanismo de tratamento de exceções existente em linguagens de programação. Entretanto, além desta integração, faz-se necessário o desenvolvimento de mecanismos de tratamento de exceções que considerem as características próprias de aplicações móveis, como sensibilidade ao contexto, comunicação assíncrona e aumento de imprevisibilidade.

No caso específico de sistemas sensíveis ao contexto, os mecanismos convencionais de tratamento de exceções não são satisfatórios por vários motivos. Primeiro, a propagação de erros deve considerar as mudanças contextuais que constantemente ocorrem nestes sistemas. Segundo, as atividades de recuperação de erros e a estratégia de tratamento de exceções frequentemente precisam ser selecionadas de acordo com as informações de contexto. Por exemplo, pode ser necessário ativar diferentes tratadores para uma mesma exceção dependendo das informações contextuais. Terceiro, a própria caracterização de uma exceção pode depender do contexto dos dispositivos envolvidos. Como exemplo, temos que um estado do sistema pode ser considerado uma condição errônea em uma dada localização onde o dispositivo se encontra, mas não em outra localização.

Portanto, em aplicações móveis é impossível o uso direto dos mecanismos convencionais de tratamento de exceções (Garcia et al., 2001; Goodenough, 1975). Os modelos tradicionais de tratamento de exceções negligenciam os

¹ Do inglês, *openness*.

problemas derivados da sensibilidade ao contexto e não fornecem soluções apropriadas para os mesmos (Garcia, 2005; Tripathi & Miller, 2000).

1.2. Limitações dos Trabalhos Relacionados

Não foram encontrados na literatura trabalhos que considerem os requisitos de sensibilidade ao contexto *diretamente* no tratamento de exceções de aplicações móveis. Por outro lado, existem alguns trabalhos que consideram questões gerais relacionadas ao tratamento de exceções em aplicações móveis baseadas em agentes. Entretanto, os mecanismos propostos em tais abordagens restringem-se a características de tratamento de exceções específicas de propriedades como mobilidade e autonomia de agentes, sem levar em consideração os requisitos de sensibilidade ao contexto. A partir destas restrições, no decorrer deste texto aplicações sensíveis ao contexto são tratadas como sinônimo de aplicações móveis baseadas em agentes. A seguir, introduzimos algumas abordagens para tratamento de exceções em aplicações baseadas em agentes, as quais são discutidas em detalhes no Capítulo 3 (Seção 3.3).

Na abordagem de Tripathi e Miller (2000), as exceções são classificadas em internas e externas ao agente. As exceções internas são tratadas localmente pelo agente e as exceções externas são propagadas para agentes especiais denominados “guardiões”, os quais implementam reações gerais para as exceções. A principal limitação, portanto, diz respeito à impossibilidade de existirem tratadores no contexto das entidades que chamam o serviço. Esta é uma solução extremamente restritiva tendo em vista que a maior parte do código de tratamento de exceções em sistemas reais é específico de aplicação (Garcia et al, 2001). Além disso, cria-se um gargalo para o sistema, pela centralização do tratamento de exceções em uma entidade especializada.

Na abordagem descrita em Souchon et al. (2004), tratadores de exceções estão associados com as entidades que são responsáveis pela execução das atividades dos agentes, isto é, serviços, agentes e papéis. Esta abordagem preocupa-se quase que exclusivamente com a definição dos tipos de regiões protegidas que podem ser associadas a um tratador, esquecendo-se de muitos outros aspectos fundamentais em um mecanismo de tratamento de exceções. Além

disso, a abordagem de Souchon et al. (2004) não possibilita a associação² dinâmica entre tratadores e ocorrências de exceção, fundamental para o tratamento de exceções no domínio de sistemas com mobilidade e sensibilidade a contexto.

Em Iliasov e Romanovsky (2005a), Context-Aware Mobile Agents (CAMA) é um *framework* para o desenvolvimento de aplicações móveis que fornece um conjunto de abstrações, juntamente com um *middleware* e uma camada de adaptação que permitem aos desenvolvedores tratar características de aplicações móveis como abertura, tolerância a falhas, comunicação assíncrona, como também mobilidade de dispositivo e de código. CAMA suporta o conceito de escopos aninhados, que agrupam erros e os tratadores de exceção aos quais estão associados. Contudo, CAMA não suporta um tratamento de exceções colaborativo, sensível ao contexto. Por exemplo, CAMA não trata a definição de contextos excepcionais e a busca de tratadores sensível ao contexto.

Além das três abordagens acima, também podem ser vistos como trabalhos relacionados os diversos *middlewares* que oferecem suporte à construção de aplicações móveis sensíveis ao contexto. Tais sistemas são discutidos em detalhes no Capítulo 2. Suas arquiteturas subjacentes utilizam diferentes técnicas de coordenação, tais como espaços de tuplas (Iliasov & Romanovsky, 2005a), *publish-subscribe* (Sacramento, et al., 2004) e reflexão computacional (Capra, et al. 2003). Entretanto, assim como nas abordagens anteriores, em tais sistemas não existe um suporte explícito para *tratamento de exceções sensível ao contexto*.

1.3. Solução Proposta

Este trabalho realiza uma análise das soluções existentes para o tratamento de exceções, considerando os requisitos de sensibilidade ao contexto. A análise deste problema foi conduzida através da identificação de questões relacionadas ao tratamento de erros que não são satisfeitas por (i) mecanismos de exceções de linguagens de programação, (ii) mecanismos de exceções existentes para aplicações baseadas em agentes e (iii) uso dos mecanismos convencionais disponíveis nos *middlewares* sensíveis ao contexto.

² Do inglês, *binding*.

Para o tratamento de exceções sensível ao contexto, propomos: (1) um modelo de tratamento de exceções sensível ao contexto, (2) uma arquitetura de software que define componentes e interfaces necessários para a implementação do modelo e (3) um mecanismo que implementa o modelo e a arquitetura a partir de MoCA (Mobile Collaboration Architecture). MoCA é um *middleware publish-subscribe* que possibilita o desenvolvimento de aplicações móveis colaborativas pela incorporação de serviços que garantem aos agentes de software a característica de sensibilidade ao contexto (Sacramento, et al., 2004). Além do modelo, da arquitetura e do mecanismo para tratamento de exceções sensível ao contexto, relatamos nossa experiência de implementação de estratégias de tratamento de exceções sensível ao contexto em alguns protótipos de aplicações colaborativas desenvolvidas a partir de MoCA.

1.4. Objetivos

Este trabalho investiga a relação entre tratamento de exceções e sensibilidade ao contexto. São objetivos específicos:

1. Identificar os problemas inerentes aos mecanismos existentes para tratamento de exceções em aplicações móveis bem como analisar suas limitações para o desenvolvimento de aplicações robustas sensíveis a contexto.
2. Descrever e analisar um conjunto representativo de *middlewares* para desenvolvimento de aplicações móveis sensíveis ao contexto: MoCA (Sacramento et al., 2004), CAMA (Iliasov & Romanovsky, 2005a) e CARISMA (Capra et al., 2003).
3. Propor um modelo e uma arquitetura para tratamento de exceções sensível ao contexto.
4. Implementar um mecanismo de tratamento de exceções, de acordo com o modelo e a arquitetura referidos no item 3, a partir da utilização do *middleware* MoCA.
5. Utilizar e avaliar o mecanismo de tratamento de exceções em estudos de caso de aplicações sensíveis ao contexto já existentes. Atualmente o uso

do mecanismo abrange as seguintes aplicações: (1) *Virtual Lines* (VL) e (2) *Health Care* (HC).

1.5. Organização do Texto

Este trabalho está organizado como a seguir. No Capítulo 2, são apresentados os principais conceitos associados com aplicações móveis e sensibilidade ao contexto, além de alguns *middlewares* existentes na literatura para o desenvolvimento das aplicações móveis.

No capítulo 3, são apresentados inicialmente alguns conceitos gerais relacionados ao tratamento de exceções e, posteriormente, são discutidos alguns requisitos do tratamento de exceções sensível ao contexto.

Tendo como base as questões levantadas no capítulo 3, um modelo geral para tratamento de exceções em aplicações móveis sensíveis ao contexto é apresentado no capítulo 4. Posteriormente, no capítulo 5, são apresentados os principais componentes de uma arquitetura para tratamento de exceções sensível ao contexto, assim como as funções e interfaces destes componentes.

O capítulo 6 discute a implementação de nosso mecanismo de tratamento de exceções sensível ao contexto a partir do middleware MoCA. O capítulo 7 apresenta os estudos de caso e avalia o mecanismo anterior. Finalmente, o capítulo 8 apresenta as conclusões deste trabalho e os trabalhos futuros.