

1 Introdução

A pesquisa em redes móveis e computação pervasiva têm demonstrado que adaptações dinâmicas e sensibilidade ao contexto são requisitos básicos para aplicações nestes ambientes. Em tais sistemas, assim que as aplicações detectam alguma variação significativa em seu contexto, elas devem se adaptar para otimizar o seu desempenho no novo contexto.

As adaptações podem envolver diferentes componentes da aplicação em questão e podem estar relacionadas a diversos tipos de informações de contexto. Por exemplo, os dispositivos móveis podem adaptar o algoritmo de codificação/compressão usado em uma camada de comunicação para que, dependendo da taxa de transmissão da rede sem fio, utilizem um algoritmo mais eficiente do que outro, e assim reduzam a quantidade de pacotes transmitidos pelo enlace sem fio.

Muitos dos trabalhos sobre adaptação dinâmica sensível ao contexto encontrados na literatura se baseiam nas informações de contexto do dispositivo para executar uma determinada adaptação apenas no próprio dispositivo. Entretanto, a maneira da adaptação diz respeito à comunicação entre dispositivos, e para que esta seja efetiva, em muitos casos ela requer mudanças coordenadas em pelo menos dois dispositivos. Esta dissertação trata a adaptação para aplicações colaborativas, que frequentemente envolvem mais do que dois dispositivos. Para uma adaptação coordenada em um grupo, seria inadequado considerar apenas o contexto de um único dispositivo para servir como base para cada dispositivo executar a adaptação. Portanto, este trabalho utiliza um contexto acordado entre os membros do grupo, que é usado como uma referência para que todos os membros executem a mesma adaptação.

Muitos dos sistemas sensíveis ao contexto possuem mecanismos específicos para detectar o contexto no qual eles estão inseridos e executar adaptações locais para se adequar melhor ao ambiente. No exemplo citado acima, as adaptações não podem ser executadas isoladamente com base no contexto de cada dispositivo,

pois todos os dispositivos têm que usar o mesmo algoritmo de compressão para se comunicar corretamente, ou seja, o pacote codificado/comprimido por um dispositivo será decodificado/descomprimido corretamente por todos os dispositivos envolvidos na comunicação. Nesta situação, a adaptação de uma componente do sistema deve ser coordenada em todos os dispositivos.

Existem trabalhos propondo soluções para o problema da adaptação dinâmica em grupos de dispositivos. Porém, cada um desses trabalhos apresenta limitações no processo de adaptação coordenada. Por exemplo, alguns não permitem a adaptação suave e transparente de componentes do sistema (a comunicação entre os dispositivos é interrompida para que a adaptação seja executada) como no Ensemble (Hayden, 1998), ou não tratam as desconexões dos dispositivos envolvidos como no NeCoMan (Janssens, 2004) e no Protocolo de Adaptação Graciosa (Chen, 2001), ou então se limitam a uma adaptação coordenada entre pares de dispositivos, como no NeCoMan (Janssens, 2004).

1.1. Objetivos

Este trabalho propõe o protocolo Moratus e o serviço de *middleware* que implementa o protocolo, denominado Serviço de Adaptação Coordenada de Serviços (SACS) com o intuito de oferecer uma solução sem as limitações dos outros trabalhos citados na Seção 1. Esse serviço possibilita uma adaptação transparente e coordenada de componentes de comunicação, o tratamento de desconexões de dispositivos durante o processo adaptativo e a adaptação coordenada em grupos. O *middleware* SACS foi projetado para grupos de dispositivos móveis interconectados por meio de redes sem fio infra-estruturadas, como por exemplo redes locais sem fio (*Wireless LAN*)

O protocolo Moratus determina o *contexto global* do grupo, que é um contexto acordado entre diversos dispositivos (membros do grupo) para a escolha de uma única adaptação adequada a todos os dispositivos. A determinação do contexto global se baseia nos contextos de cada um dos dispositivos, e de posse do contexto global, os dispositivos executarão a mesma adaptação, transparentemente, ou seja, de uma maneira que a execução da aplicação distribuída não tenha que ser interrompida.

O SACS é o serviço de *middleware* responsável por oferecer serviços para otimizar a comunicação de aplicações distribuídas e coordenar a adaptação de serviços relacionados ao processamento de mensagens em diversos dispositivos. Através dele aplicações distribuídas podem chavear entre diferentes *serviços de transformação de mensagens* (como, por exemplo, compressão de mensagens), que serão selecionados, instalados e ativados automaticamente pelo SACS.

O alvo do SACS/Moratus são aplicações móveis colaborativas, usadas para a troca de mensagens ou compartilhamento de dados, de forma síncrona ou assíncrona entre usuários. Pode-se citar como exemplo sistemas de troca de mensagens instantânea, video conferência, compartilhamento de agendas, troca de arquivos e co-edição de documentos. Estas aplicações tem como característica comum a adoção do paradigma de comunicação de grupo, utilizado para manter a sincronização entre os estados dos clientes da aplicação. Nestas aplicações, os grupos geralmente são fixos e pequenos (até 10-20 usuários). Exemplos dos tipos de usuários que pertencem a esses grupos são: técnicos, vendedores ou engenheiros em trabalho de campo, grupos de turistas e grupos de amigos.

1.2. Organização da Dissertação

Esta dissertação está organizada da seguinte forma: o capítulo 2 apresenta os principais conceitos nos quais o Moratus se baseia. O capítulo 3 apresenta o protocolo Moratus, o serviço de *middleware* SACS (Serviço de Adaptação Coordenada de Serviços) e sua arquitetura. O capítulo 4 apresenta a descrição da implementação do protocolo Moratus e do serviço de *middleware* SACS. O capítulo 5 apresenta uma avaliação do SACS, como o resultado dos testes de desempenho e dos testes de desconexão. O capítulo 6 apresenta os principais trabalhos relacionados com o Moratus e faz um estudo comparativo entre os trabalhos. Por fim, o capítulo 7 apresenta as conclusões e as propostas para trabalhos futuros.