EXEHDA-TS: um modelo de espaço de tuplas para a Computação Pervasiva

Rodrigo Santos de Souza[1], **Adenauer Corrêa Yamin**[1]

[1]PPGINF – Escola de Informática – Universidade Católica de Pelotas (UCPEL)
Rua Félix da Cunha, 412, Prédio C Sala 431– 96010-000 – Pelotas – RS – Brasil

{rsouza,adenauer}@ucpel.tche.br

1. Introdução

Este trabalho, tem como objetivo central propor um modelo para a coordenação de aplicações na Computação Pervasiva, estando prevista sua implementação como um serviço para o middleware EXEHDA.

O middleware EXEHDA, em desenvolvimento no G3PD/UCPEL, consiste em uma arquitetura para um ambiente de grade computacional direcionado às aplicações distribuídas, móveis e conscientes do contexto da Computação Pervasiva. Maiores detalhes sobre sua arquitetura e funcionalidades podem ser obtidas no documento [Yamin 2004].

As motivações de pesquisa centrais deste trabalho envolvem os diversos desafios de coordenação gerados pela mobilidade lógica e física suportadas pelo EXEHDA.

2. Visão geral do EXEHDA-TS

A coordenação, em computação distribuída, consiste de um sistema que manipula a comunicação e cooperação entre os diferentes processos envolvidos. Modelos baseados em coordenação caracterizam-se pela separação entre computação e coordenação [Tanenbaum and Steen 2007].

Um dos modelos de coordenação mais difundidos é o de Espaço de Tuplas (TS) que foi introduzido por Linda [Carriero and Gelernter 1989]. Este modelo consiste de uma memória associativa independente, compartilhada entre todos os nodos do sistema [Yamin 2004]. Esta memória pode ser considerada como uma espécie de repositório para as estruturas denominadas tuplas. A comunicação entre processos se dá através da inserção e leitura destas tuplas neste espaço compartilhado.

Os modelos de coordenação tradicionais apresentam limitações de utilização considerando as demandas da Computação Pervasiva [Mamei and Zambonelli 2006]. Por isso, quando da concepção do EXEHDA, foi introduzida a perspectiva de uso de um TS distribuído com atuação proativa, isto é, com a possibilidade de notificação aos componentes de software associados as diferentes tuplas [Yamin 2004][Augustin 2004]. O EXEHDA-TS tem por foco tornar substantiva esta premissa, e se propõem a modelar os diversos aspectos inerentes a sua operação de modo a atender as demandas da Computação Pervasiva. Nos estudos realizados durante a concepção do EXEHDA-TS foram identificados alguns trabalhos relacionados, como Lime [Murphy et al. 2006] e Tota [Mamei et al. 2003], entre outros.

No EXEHDA-TS, os TS são associados aos Objetos EXEHDA (OX). Conforme o OX se estabelece em um EXEHDAnodo específico, seu TS pode ser estendido de forma

a ser acessível às demais aplicações, formando um TS único. Esta estratégia proporciona um tratamento às desconexões.

Por questões de desempenho, o TS não pode prescindir de uma área de abrangência limitada [Mamei and Zambonelli 2006]. Reforçando a indicação para a pesquisa em andamento que o EXEHDA-TS tenha abrangência em nível celular, considerando o ambiente pervasivo previsto pelo EXEHDA.

Para promover uma melhor escalabilidade, o EXEHDA-TS prevê o suporte a tratamento de eventos, que vem também a contribuir na redução dos esforços computacionais, porque libera as aplicações do ônus de monitorarem o TS por conta própria.

Como estratégia para coordenar aplicações com componentes de software localizados em células distintas, o EXEHDA-TS prevê a incorporação sob demanda, de espaços de tuplas de OX externos a célula.

O EXEHDA-TS prevê a possibilidade de organizar tuplas com características semelhantes em espaços de tuplas particulares. Esta organização é determinada por regras associadas aos espaços de tuplas de cada OX. Estas regras podem ser utilizadas, por exemplo, para separar tuplas com informações de contexto das demais.

3. Considerações Finais

O EXEHDA-TS é um modelo de Espaço de Tuplas focado na coordenação de aplicações da Computação Pervasiva. O trabalho está em fase de modelagem e pretende ser implementado como um serviço para o EXEHDA. Está previsto o desenvolvimento de um estudo de caso contemplando aplicações em medicina pervasiva.

Referências

Augustin, I. (2004). *Abstrações para uma Linguagem de Programação Visando Aplicações Móveis em um Ambiente de Pervasive Computing*. PhD thesis, UFRGS.

Carriero and Gelernter (1989). How to write parallel programs: A guide to the perplexed. *CSURV: Computing Surveys*, 21.

Mamei, M. and Zambonelli, F. (2006). *Field-Based Coordination for Pervasive Multiagent Systems*. Springer.

Mamei, M., Zambonelli, F., and Leonardi, L. (2003). Tuples on the air: A middleware for context-aware computing in dynamic networks. In *ICDCS Workshops*, pages 342–347. IEEE Computer Society.

Murphy, A. L., Picco, G. P., and Roman, G.-C. (2006). Lime: A coordination model and middleware supporting mobility of hosts and agents. *ACM Transactions on Software Engineering and Methodology*, 15(3):279–328.

Tanenbaum, A. S. and Steen, M. V. (2007). *Distributed Systems: Principles and Paradigms*. Prentice Hall, 2 edition.

Yamin, A. C. (2004). *Arquitetura para um Ambiente de Grade Computacional Direcionado às Aplicações Distribuídas, Móveis e Conscientes do Contexto da Computação Pervasiva*. PhD thesis, UFRGS.