Chapter 9
Community Computing:
Multi-Agent Based Computing Paradigm for Cooperative Pervasive System

Youna Jung
University of Pittsburgh, USA

Minsoo Kim
University of Pittsburgh, USA

ABSTRACT

Many huge and complex pervasive services can be provided by employing cooperation among smart objects like agents. To offer such services efficiently, community computing was proposed as a new computing paradigm, in which pervasive services are provided through cooperation among agents (Jung, 2006). To design a community computing system, we proposed two abstraction models distinguished by intelligence level of community; the simple model (Jung, 2006) and the static community situation based model. In addition, we introduced a development process based upon the Model-Driven Architecture (MDA) approach for fast and convenient implementation of community computing systems. In this chapter, our contribution is to organize previous work related to cooperation and then clearly present the position of community computing in comparison. In addition, we refine the proposed two models including all their intermediate models in the development process, such as CCM (Community Computing Model), CIM-PI (Platform Independent Community Computing Implementation Model), and CIM-PS (Platform Specific Community Computing Implementation Model). To help automatic transformation from CCM to source codes, we improve a development toolkit called CDTK (Community Computing Development Toolkit). By using CDTK, a community computing system can be implemented semi-systematically. Finally, to verify the feasibility of community computing, we present implementation results of two scenarios by using proposed two models and CDTK. Through the simulated results, we examine the possibility of community computing.

DOI: 10.4018/978-1-60960-735-7.ch009

Copyright © 2012, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

INTRODUCTION

Since ubiquitous computing was articulated by Mark Weiser in 1991 (Weiser, 1991), many researchers have attempted to realize the potential for diverse pervasive services. As we surveyed existing research, we were able to find the unique characteristics of pervasive computing. First of all, a pervasive computing system is composed of highly heterogeneous computing elements (Weiser, 1991; Kindberg, 2002). As computing elements have mobility and their status changes frequently, the environment of a pervasive system is dynamically changing. In such a dynamic environment, predictable or unpredictable pervasive services are dynamically requested. Among all characteristics, what we concentrate on the most is that many pervasive services can be provided by cooperation among heterogeneous smart objects rather than by the ability of a single smart object. In order to design and develop a pervasive system having all such characteristics, we surveyed existing approaches, but they were not perfectly adequate to do it.

In the PICO Project, community computing was introduced as a framework for cooperation among agents in a pervasive environment (Kumar, 2003). However, in this approach, some room was left to develop the pervasive systems providing services by using cooperation among intelligent devices. Most of all, there is no well-defined formal model. The PICO Project and Active Space Project have introduced a similar view of pervasive services to our community computing, but they provide no concrete model to represent such pervasive systems. Accordingly, to design cooperation among agents, PICO needs an abstraction model to describe cooperation. In order to find an appropriate model, we surveyed quite a few models, especially focusing on multi-agent models. It should be noted, however, that there are some differences between multi-agent approaches and our approach for cooperative systems. Furthermore, detailed and strong concern about cooperation is lacking. As the required pervasive services are getting larger and more complex, cooperation among pervasive objects becomes increasingly important. Accordingly, it is necessary to raise concern about cooperation in a pervasive environment. For example, it can be the cooperation-based service providing scheme, the configuration of cooperative organizations, cooperative behaviors between pervasive objects, etc.

The ultimate objective of our work is to design and develop pervasive computing systems efficiently which provide services by using cooperation among objects that already exist within a given environment. Existing approaches, including multi-agent approaches and other pervasive system development approaches, are not adequate for our purpose. Therefore, in this chapter, we introduce community computing, a new approach to design and develop such a cooperative pervasive computing system. In order to meet our objective, we are employing the concept of community. In community computing, community is a high-level abstract concept for organizing, managing, operating, and repairing groups of computing elements in a pervasive environment. Using the community concept, we are able to meet all requirements of pervasive computing. First of all, community computing can adapt to changes in runtime by dynamic creation of a goal-driven community and dynamic binding of the roles in the community to actual objects. Secondly, community computing supports dynamic cooperation among objects through dynamic decisions about cooperative behavior and dynamic injection, a cooperation process into member objects. Also, we can guarantee proper separation of concerns in community computing. In the community computing models, group concerns are discriminated from concerns about individual objects. Finally, community computing supports the scalability of services by merging of communities or the scalability of systems by merging of societies. (See section 3 for the definition of society).

21 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/chapter/community-computing-multi-agent-based/55441

Related Content

Software Engineering at Full Scale: A Unique Curriculum
Jochen Ludewig (2009). *Software Engineering: Effective Teaching and Learning Approaches and Practices (pp. 265-277).*
www.irma-international.org/chapter/software-engineering-full-scale/29603

Mutation Testing for Evaluating PHP Web Applications
Ahmad A. Saifanand Mahmoud Bani Ata (2019). *International Journal of Software Innovation (pp. 25-50).*
www.irma-international.org/article/mutation-testing-for-evaluating-php-web-applications/236205

An Empirical Study on Novice Programmer’s Behaviors with Analysis of Keystrokes
Dapeng Liu, Shaochun Xuand Huafu Liu (2013). *International Journal of Software Innovation (pp. 68-87).*
www.irma-international.org/article/empirical-study-novice-programmer-behaviors/77619

Activity Theory to Guide Online Collaborative Learning Instructional Design
Siti Rosni Mohamad Yusoffand Nor Azan Mat Zin (2012). *International Journal of Systems and Service-Oriented Engineering (pp. 15-29).*
www.irma-international.org/article/activity-theory-to-guide-online-collaborative-learning-instructional-design/78916

Access Control in Service Compositions: Challenges and Solution Architecture
Aurélien Faravelonand Stéphanie Chollet (2013). *Service-Driven Approaches to Architecture and Enterprise Integration (pp. 165-187).*
www.irma-international.org/chapter/access-control-service-compositions/77949