

An Automated Distributed Meeting Scheduler for FCVW Plug-in

Hsiang-Hwa Koo and Elhadi Shakshuki

Jodrey School of Computer Science

Acadia University

Wolfville, Nova Scotia, Canada B4P 2R6

032139k@acadiau.ca, elhadi.shakshuki@acadiau.ca

Abstract

People have faced conflicts for shifting scheduled meetings to other time slots in order to fit incoming meetings or to find a time slot to book a meeting. The goal of this research is to develop a personal distributed meeting scheduler in FCVW (Federated Collaborative Virtual Workspace) that assists users to deal with these situations. FCVW is an extension of CVW [6] developed by MITRE. In our approach to meeting scheduling, is to provide each user with meeting scheduler agent. Each agent is able to manage, negotiate and schedule tasks, meetings, events, appointments for its user. This paper describes the objective of our research to develop an automated distributed meeting scheduler.

1 Introduction

A scheduler is one of the most important personal tools in daily routines. Since computing technologies have improved rapidly, software designers have developed many useful personal applications to fit users' needs and replace the traditional tools. Scheduler is one of those significant developments, and Figure 1 shows the traditional scheduler, and Figure 2 is an instance of using scheduling software to book meetings. In the software realm there are many scheduling applications such as Microsoft Outlook [4] and Microsoft Exchange Server [5].

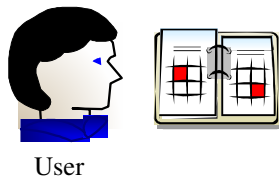


Figure 1. Traditional scheduler

These applications can be combined together through a network. The combination of these programs allows one user to book meetings on another user's scheduler. Figure 3 shows an example booking meetings via the network. PDA

is another alternative to help users' book and arrange their meetings and appointments.

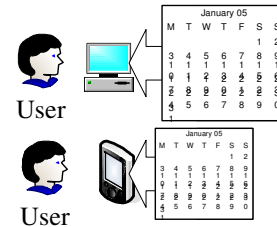


Figure 2. Computer scheduling applications

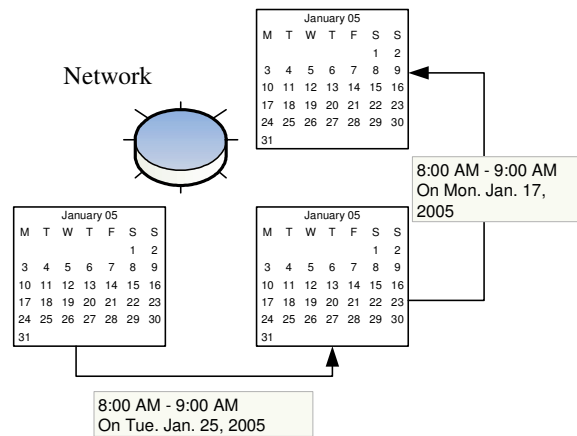


Figure 3. Book meetings through the network

Although the aforementioned schedulers are useful tools, they cannot work on behalf of their users automatically and autonomously. The main goal of our work is to develop an automated distributed meeting scheduler which consists of several agents. Each user can have an agent that acts on behalf of him or her. This agent utilizes the calendar and allows its user to inputs a meeting request and then it negotiates with agents corresponding to other users to schedule the meeting. The agents will communicate with each other using the email server. Figure 4 shows our view how a user might book meetings through the network with the personal assisting agents.

The agent can help the user to manage multiple tasks, events, and appointments. Utilizing different negotiation strategies [3] the agent will be able to make its decisions

which are based on the users' constraints and preferences that allow it to alter meetings in different time slots. We will be also considering incorporating a learning technique to the agent. This makes the agent to monitor the user behaviour as well as other agents' behaviours.

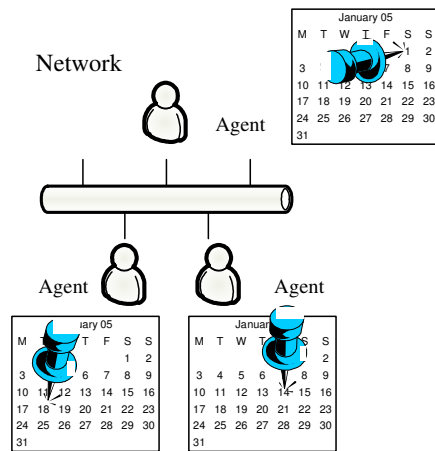


Figure 4. Book meetings using agents

2 Design Concepts

In this work, we will utilize the approaches presented in Pragnesh[1] and Sandip Sen [2]. The design of the meeting schedulers has two main parts, the server and the client. The components of the client include calendar displayer, negotiation manager, learning processor, email generator, and schedule finder. The calendar displayer displays the visual contact with the user. The negotiation manager manages tasks or appointments with other users or agents. The learning processor experiences from real world, and provides feedbacks to increase decision-making. The email generator generates the latest notification, and sends it to attendees. The schedule finder provides available time slots to the negotiation manager, and arrange meetings internally. The server consists of three components, including knowledge base, message formatter, backup database and communicator. The knowledge base stores all previous negotiation cases that will be used by the learning processor. The message formatter responsible for parsing incoming messages and formatting the outgoing messages to the required message formats. The backup database assists users to backup data of their local hard drive, and update meetings in their scheduler when they are offline. The communicator builds a link between users, and transfers messages between users. This forms the basic architecture of our proposed meeting scheduler system.

3 Implementation

Currently, we started implementing the proposed meeting scheduler in FCVW. FCVW consists of Server, client and a middleware layers. The server program contains abilities to identify users' authorization, and to modify users'

messages into the machine language and transfer to the destinations. The client application has abilities to communicate with server, manage the negotiation with other agents, book or find the desired time slots, and display the calendar. The client is a part of FCVW plug-in in the Eclipse platform. FCVW is an extension of Collaborative Virtual Workspace (CVW) [6]. CVW provides the "virtual building" which users can communicate, collaborate, and share information without geographic problem. The Eclipse [7] is an open source platform with an IDE which allows users to edit programming languages such as JAVA and C/C++. The Eclipse consists of extensive plug-ins, and users can use the plug-in development environment to develop their own plug-ins, and then integrate them into the Eclipse workbench. The meeting scheduler will be used as a plug-in.

4 Conclusion

Automated distributed meeting schedulers are the next generation of the scheduling applications. It is used the personal assisting agent to let schedulers work intelligently. The learning processor and negotiation strategies are the two most difficult concepts to materialize. Although researchers have found some solutions to solve these two problems, it requires a lot of improvements and study in order to achieve optimal result, and stability.

References

- [1] Pragnesh Jay Modi, Manuela Veloso, Stephen F. Smith, and Jean Oh, 2004, *CMRadar: A Personal Assist Agent for Calendar Management*, At The Agent Oriented Information Systems.
- [2] Sandip Sen, 1997, *An automated distributed meeting scheduler*, IEEE Expert: Intelligent Systems and Their Applications, Volume 12 Issue 4.
- [3] Roman Berres, and Eugenio Oliveira, 1999, *Automated Distributed Meeting Scheduler*, Faculdade de Engenharia, Universidade do Porto.
- [4] Microsoft Outlook <http://office.microsoft.com/en-us/FX010857931033.aspx>
- [5] Microsoft Exchange Server <http://www.microsoft.com/exchange/default.msp>
- [6] Collaborative Virtual Workspace (CVW) <http://cvw.sourceforge.net/>
- [7] Eclipse Foundation <http://www.eclipse.org/>