AGENT BASED BUDDY FINDING METHODOLOGY FOR KNOWLEDGE SHARING

By

XIAOQING LI, B.Eng., M.Eng.

A Thesis

Submitted to the School of Graduate Studies in

Partial Fulfillment of the Requirements

For the Degree

Doctor of Philosophy

McMaster University

AGENT BASED BUDDY FINDING METHODOLOGY FOR KNOWLEDGE SHARING

DOCTOR OF PHILOSOPHY (2003) McMaster University

(Management Science / Systems) Hamilton, Ontario

TITLE: Agent Based Buddy Finding Methodology For Knowledge Sharing

AUTHOR: Xiaoqing Li, B.Eng. (North-western Polytechnical University, Xian, P.R. China), M.Eng. (Shanghai Jiao Tong University, Shanghai, P.R. China)

SUPERVISOR: Professor Ali R. Montazemi

NUMBER OF PAGES: xvii, 180

Abstract

The Internet provides opportunity for knowledge sharing among people with similar interests (i.e., buddies). Common methods available for people to identify buddies for knowledge sharing include emails, mailing lists, chat rooms, electronic bulletin boards, and newsgroups. However, these manual buddy finding methods are time consuming and inefficient.

In this thesis, we propose an agent-based buddy finding methodology based on a combination of case-based reasoning methodology and fuzzy logic technique. We performed two experiments to assess the effectiveness of our proposed methodology. The first experiment was comprised of a stock market portfolio knowledge sharing environment in which a conventional cluster analysis was used as a benchmark to assess the technical goodness of the proposed methodology in identifying the clusters of buddies. Statistical analysis showed that there was no significant ranking difference between conventional cluster analysis and the proposed buddy-finding methodology in identifying buddies. Cluster analysis requires centralized database to form buddies (clusters) with similar properties. The unique advantage of our proposed agent-based buddy finding methodology is that it can identify similar buddies in distributed as well as centralized database environments. A second experiment, in the context of sharing musical-knowledge among human subjects, was used to find out whether selection of the buddies by the proposed methodology is as good as those done by human subjects. The findings from this latter empirical test showed that the buddies found by agents are as

good as the buddies found manually by humans.

Dedication

To my parents: Huaibin Li and Shuliang Chen
My wife: Yan Wang
And my son: Zhiyuan Li

Acknowledgements

I would like to sincerely thank my supervisor Dr. A.R. Montazemi for his guidance during the course of my thesis. I am grateful to him for the engaging discussions that helped shape the ideas in this thesis. Especially, I am very thankful to Dr. Y. Yuan for his significant help in shaping the ideas of experimental design and result analysis. I am thankful to the members of my supervisory committee, Dr. M. Head and Dr. A. Kubursi, for their advice and encouragement.

I am thankful to my professors and colleagues in the Department of MS/IS for their helpful comments and suggestions. Additionally, I would like to acknowledge the scholarship provided by the Department of MS/IS and position of teaching assistant provided by the McMaster University.

I am thankful to my wife Yan Wang for her support, encouragement and patience. Finally, I would also like to thank my son Zhiyuan Li who put up with difficulties when I was doing my program.

Table of Contents

Figures

Tables

Acronyms

ACF	Automatic Collaborative Filtering
AI	Artificial Intelligence
ANOVA	ANalysis Of VAriance
BBS	Bulletin Board System
BDI	Belief-Desire-Intention
BIC	Business Intelligence Center
CA	Central control agent
CAIRO	Collaborative Agent Interaction and synchROnization
CBR	Case-Based Reasoning
CEO	Chief Executive Officer
CIO	Chief Information Officer
DAI	Distributed AI
DEC	Digital Equipment Corp.
DM	Decision Maker
DSS	Decision support systems
EBIN	Executive Business Intelligence Network
ERP	Enterprise Resource Planning
ESs	Expert Systems
FIFO	First-In-First-Out
HTML	Hyper-Text Markup Language

ICM	Intellectual Capital Management
IA	Intelligent Agent
IT	Information Technology
IS	Information Systems
ISO	International Organization for Standardization
IP	Internet Protocol
KBS	Knowledge-Based System
KMS	Knowledge management systems
KQML	Knowledge Query and Manipulation Language
MASs	Multi-agent Systems
MIS	Management Information Systems
NN	Nearest Neighbor
RMS	Root-Mean-Square
WWW	World Wide Web

1. Introduction

1.1 Introduction

Knowledge is a major driving force for organizational change and wealth creation and "knowledge management is an increasingly important source of competitive advantage for organizations" (Gottschalk, 2000). As an effective tool to facilitate sharing personal knowledge (i.e. tacit knowledge) among like-minded people (i.e., buddies), the role of online communities has become increasingly important to the success of organizations (Malhotra, 2002). To organize their communities, conventional methods, like emails, mailing lists, chat rooms and message boards, are widely used by people to find their buddies on the Internet (Catterall and Maclaran, 2002). For example, using the message boards, music lovers find music buddies by posting their favorite music, students find study buddies by posting one particular question (e.g., a test in statistics), computer programmers find programming buddies by posting problems related to the use of particular software, and etc. However, with the increasing number of users, these conventional methods are suffering from information overload and becoming time consuming and not very effective (Geyer, 1996; Gould, 1999; Smith, 2002). To solve this problem, the computerized buddy finding methods can be used to reduce information overload. One successful automatic buddy finding technology is collaborative filtering. Collaborative filtering technology recommends users useful information based on their buddies' interest (Maltz and Ehrlich, 1995). For example, Firefly (www.firefly.com) uses the opinion of buddies to share knowledge about such products as music, books, Web pages and restaurants. However, the collaborative filtering technology needs a centralized knowledge base to save knowledge received from all the users. The question arises as to how to deal with decentralized knowledge-bases (e.g., sharing music through Napster.com) for sharing knowledge among a large number of users.

1.2 Research Objective

The research objective reported in this thesis is to provide a buddy finding methodology in the decentralized knowledge-sharing environment. In the decentralized environment, the software agent is believed to be a promising tool to find buddies for people. The agents are widely recognized as suitable abstractions to deal with complex application environments, especially when the openness and unpredictable dynamics of the environment make traditional approaches less effective (Ricci et al., 2001). Agents are autonomous and intelligent. They can also communicate with each other to exchange information (Newell, 1988). By delegating tasks to agent systems, users not only save time and energy, but also have more opportunities to access valuable information and work on complex and creative jobs. Intelligent agents are expected to embody some of the key capabilities of a human assistant: observing and forming models of the decision environment; inferring the user's intentions based on these observations; and formulating plans and taking actions to help the user achieve these intentions (Hayes-Roth, 1995; Newell, 1988). Therefore, an agent-based methodology is proposed to find buddies autonomously for the users. Our proposed methodology is to deal with the buddy finding issue in the decentralized environment, in which there is no central data repository to store all users' information.

1.3 Research Methodology

This thesis is about finding buddies to facilitate knowledge sharing in the distributed environment. Our surveys fall into two main areas: knowledge management, and Intelligent Agents. With the survey of knowledge management, we clarify the importance of buddies finding technologies and analyze the limitations of existing buddy finding methods, including manual buddy finding methods and automatic buddy finding methods. The major limitations of these existing buddy-finding methods are the incapability of handling information overload. These survey results justified the necessity of developing an agent-based buddy-finding methodology, since an agent is believed to be a powerful tool to ameliorate the information overload in the open and dynamic environment. We surveyed the fundamentals of intelligent agents, and especially the agent coordination strategies, which are relevant to our research issue. We realized that there is no existing agent coordination technology that can fully satisfy our requirements. There are two major types of coordination strategies, centralized and decentralized. In the centralized coordination structure, the multi-agent systems (MASs) need a middle agent to coordinate the communication among peer agents, this structure also suffers from the issue of information overload as the number of agents increases. In the decentralized coordination structure, i.e., acquaintance structure, agents only need to communicate with small set of pre-determined peer agents. The acquaintance structure solves the problem of information overload to some extent. But, unfortunately, the system builders need to build the buddy list when they are implementing the multi-agent systems. That is, the agent buddy list is fixed. When the number of agents is huge and the agents' characteristics are changeable over the time, the fixed buddy list could not adapt to the change of agents' characteristics to reflect the real time relationship among agents.

In our research scenario, agents are expected to dynamically identify a small group of peer-agents with the most probable chance of providing the optimal information requested (i.e., buddy agents) by autonomously communicating with peer-agents. To this end, in this thesis, we propose a methodology to assess the degree of membership of buddy agents. This methodology is based on fuzzy set modeling (Zimmermann, 1987). The objective is to select buddy agents that are expected to meet a set of criteria in responding to a request. Our proposed methodology was empirically tested in two application scenarios. One was the application of stock portfolio selection, and the other was music buddy selection. Conventionally, cluster analysis is widely used to identify like–minded people segments (Gallagher and Mansour, 2000; Gehrt and Shim, 1998; Lin et al., 1999; March, 1997) in a centralized environment (Jain et al., 1999). Cluster analysis is a fundamental technique of unsupervised learning in machine learning and statistics (Duda and Hart, 1973; Hartigan, 1975). Therefore, a conventional cluster analysis is used as a benchmark to assess the technical goodness of the proposed methodology in identifying the clusters of buddies in the first experiment. The second experiment is to evaluate users' satisfaction to agent found buddies through the comparison with the subject found buddies. To apply our proposed methodology to find music buddies, we review the basic music knowledge that is most relevant to the research in this thesis. The most important music attributes are summarized. These attributes are the foundation to apply our proposed methodology. Users' comments are collected to deepen readers' understanding of the test results. These empirical tests were employed to assess the effectiveness of our proposed methodology of buddy agent membership in support of complex decision problems.

1.4 Organization of This Thesis

This thesis is constructed as follows. Chapter 2 describes literature review of knowledge management. We survey the major issues of knowledge management, and the significance of information technologies, such as our proposed buddy-finding methodology, in knowledge sharing. Chapter 3 provides a review of the fundamentals of software agent methodologies and related coordination among multi-agent systems. This review provides the basis to compare and contrast our proposed methodology for coordination among multi-agent systems. Chapter 4 describes our proposed agent-based buddy finding methodology. We postulate two hypotheses to test the goodness of the proposed methodology. Chapter 5 describes an empirical test of stated hypotheses in the scenario of stock-market portfolio selection. Statistical analysis showed that there was no significant ranking difference between conventional cluster analysis and the proposed buddy-finding methodology in identifying buddies. Chapter 6 describes empirical evaluation in the scenario of music buddies finding. Statistical analysis showed that there was no significant preference difference between the agent-found buddies and buddies identified by the subjects. Chapter 7 summarizes the contributions made in the thesis and suggests directions for further research.

2. Knowledge Management

2.1 Introduction

Globalization and digitization are compelling companies to reconsider fundamental business assumptions. These two intertwined strategic forces translate into competition that is increasingly knowledge-based (Lang, 2001). Knowledge is now becoming a major driving force for organizational change and wealth creation. Effective knowledge management is considered an increasingly important source of competitive advantage (Gottschalk, 2000), and a key to the success of contemporary organizations (Irma and Rajiv, 2001). As a result, companies are now showing a tremendous interest in implementing knowledge management processes and supporting technologies.

Knowledge management systems (KMS) refer to a class of information systems developed to support and enhance the organizational processes of knowledge creation, storage/retrieval, transfer, and application (Alavi and Leidner, 2001). Using IT to manage knowledge is not new. Organizational efforts in support of knowledge management through information technologies can be traced back to artificial intelligence methodologies such as expert systems, and case-based reasoning systems. Recent progress in IT, especially Internet and database technology, has transformed the processes applied towards management of organizational knowledge. Various information technologies, like knowledge networks, communities of practice, and virtual communities, are applied to better manage organizational resources, especially the knowledge stored in human minds, so-called tacit knowledge. Tacit knowledge is highly personal and hard to encode. It plays a unique role in building and conserving core competence. The challenge is how to support the sharing of tacit knowledge using information technologies. This chapter provides and discusses the basic issues in support of knowledge management through information technology, especially knowledge networks, communities of practice, and virtual communities. We will discuss the concepts and related technologies. Detailed description of our proposed information technology model in support of knowledge management is presented in chapter 4.

This chapter is constructed as follows. Section 2 describes knowledge management and competence. Several major knowledge management approaches, mainly knowledge networks and virtual communities, are described in section 3. In section 4, we will focus on discussing the current group technologies and their major challenges. In the last section, 5, we will explain the significance of our suggested agent-based buddy matching technology.

2.2 Knowledge Management and Competence

Knowledge management can be defined as a formal, directed process of figuring out what knowledge individuals within a company have that could benefit others in the company, then devising ways of making it easily available (Harvard Management Update, 1999). Civi (2000) defines knowledge management as the acquisition, sharing and use of knowledge within organizations, including learning processes and management information systems. Although there still does not exist a standard definition of knowledge management, knowledge management has already been a common accepted practice within organizations. Today most advanced industries are knowledge-based. In 1996, the Journal of Knowledge Management did a survey sponsored by Ernst & Young/Business Intelligence on senior management's views toward knowledge management (Chase, 1997b). From this survey, some 92% of the respondents reported that they worked in knowledge-intensive organizations. Also from this survey (Chase, 1997b, p40), nearly two-thirds of the respondents reported that costly mistakes were made due to "insufficient knowledge about technology" and "vital knowledge was lost without timely warning." The benefits of knowledge management are visible in knowledge intensive industries such as software, pharmaceuticals, health care, financial services, communications, and consulting.

The business environment is increasingly competitive and the rate of innovation is rising. Companies compete with each other in ways different from before. "Dr. Dorothy Leonard of the Harvard Business School maintains that: It used to be that organizations could compete on the basis of either quality or low cost. Today, almost every organization competes on its ability to continuously innovate – in product, service or concept" (Chase, 1997a, p84). A company's competency can be classified into similar competencies and core competencies. Similar competencies are those common processes that successful companies adopt within an industry. Core competencies, however, are processes unique to individual firms that give rise to their competitive advantage. Whereas generic knowledge is the basis of the competence possessed by all the firms in an industry, specific knowledge is particular to individual firms, resulting in their individual core competencies and potential competitive advantage (Pemberton and Stonehouse, 2000). Firms' "competitive advantage" is based on firm-specific core competencies (Prahalad and Hamel, 1990). To this end, knowledge plays a unique role in building and conserving an organization's core competencies.

Recent literatures on knowledge management classify knowledge within an organization into two categories: tacit knowledge and explicit knowledge. Explicit knowledge can be documented and shared in forms of scientific formula, specifications, manuals, documents, reports, mission statements, etc. Tacit knowledge is knowledge that cannot be articulated because tacit knowledge is highly personal and hard to encode. For example, a physician can use her tacit knowledge to diagnose a rare disease. Nonetheless, she may be unable to easily provide a detailed model of her thought process leading to her final diagnosis. Developments in information technology have transformed the ability of organizations in knowledge management. Present applications of information technology in support of knowledge management mainly deal with explicit organizational knowledge. The integrated solution, known as enterprise resource planning (ERP), promises benefits from increased efficiency to improved quality, productivity, and profitability. ERP systems are software applications that provide transaction management to enable timely execution of decision support systems to plan and manage resources across an enterprise. These systems facilitate well-managed resource planning in the face of rapidly changing constraints such as materials availability, market readiness, plant capacities, personnel certification and business costs per location. Software vendors such as SAP AG, Baan, PeopleSoft and Oracle provide a host of integrated ERP products.

Tacit knowledge is highly personal and hard to encode. Individuals are the primary repositories of tacit knowledge that, due to its transparent characteristics, is difficult to communicate. The main difficulties of sharing tacit knowledge are as follows (Haldin-Herrgard, 2000):

- Perception and language are considered the main difficulties in sharing tacit knowledge.
    - Perceptually the characteristics of unconsciousness entails a problem of people not being aware of the full range of their knowledge.
    - Difficulties with knowledge lie in the fact that tacit knowledge helps in a non-verbal form.
- Time also raises difficulties for sharing tacit knowledge. The internalization of this form of knowledge requires a long time both for individual and organizational forms of knowledge. To not only experience but also to reflect on these experiences is time consuming but a necessity to develop tacitness in one's work.
- Value is another field with difficulties in sharing tacit knowledge as well as explicit knowledge. Many forms of tacit knowledge, like intuition and rule-of-thumb, have not been considered valuable (Zack, 1999).
- Another difficulty is that it is not only valuable and beneficial knowledge that is shared as true organizational or personal tacit knowledge. Bad habits and obsolete

behavior are also diffused. Once shared and internalized, these bad habits tend to be difficult to stop.

- Distance also raises difficulties for sharing tacit knowledge. The need for face-to-face interaction is often perceived as a prerequisite for diffusion of tacit knowledge.

Tacit knowledge plays a unique role in building and conserving core competence, i.e., tacit knowledge is lost to a competitor when a knowledge worker leaves the firm. The business value of tacit knowledge lies at (Horvath, 2000):

- Innovation

  Tacit knowledge is strongly implicated in organizational innovation. People develop and use tacit knowledge before they are able to formalize or codify it. Thus the leading edge of the firm's learning (and a source of its future innovation) is often to be found in tacit knowledge.

- Best practices

  Attention to tacit knowledge can enable firms to identify and transfer best practices more effectively. Excellent business practices cannot be typically transferred unless they are well understood, and effective practices cannot be understood without reference to the tacit knowledge of the people who do the work..

- Imitation

  Tacit knowledge can help firms to resist imitation by competitors. Because it is embodied in people and embedded in the things they create, tacit knowledge tends to be "sticky"--to resist transfer to new groups and settings. Thus, firms that work effectively with tacit knowledge can expect to increase both their ability to innovate and their ability to extract innovation in the marketplace.

- Core competence

  A consideration of tacit knowledge can illuminate the emerging core competencies of the firm. Tacit knowledge represents the unique value added by the people who generate it. It emerges from their particular situations, skills, and experiences and, in the aggregate, reflects the history and circumstances of the firm. Tacit knowledge needs to be considered in the evaluation of the firm's core capabilities--those best-in-the-world capabilities with the potential to distinguish the firm from its competitors.

Knowledge management intends to capture and use organizational knowledge resources as effectively as possible. Today and increasingly in the future, the transfer of knowledge and expertise, and the creation of a "learning" organization, has become a critical factor to company innovation and competitiveness. Developments in technology, and particularly those in information and communications technology, have played a vital role in providing the infrastructures for management of tacit knowledge both within and between collaborating companies. The question addressed here is: how can we use information technology in support of knowledge management, especially tacit knowledge?

2.3 Knowledge Management Approaches

As we said early in this chapter, organizational efforts in support of knowledge management through information technologies can be traced back to artificial intelligence methodologies such as expert systems and case-based reasoning systems. Expert Systems (ESs) use human knowledge in the form of If-Then rules to solve problems that ordinarily require human expertise. ESs imitate the reasoning processes that experts use to solve specific problems. Novices can use ES to improve their problem-solving capabilities. Experts can also use ES as an assistant. Most commercial ESs are rule-based systems; that is, the expert's expertise (tacit knowledge) is stored mainly in the form of production rules. The benefits brought by ESs are apparent. For example, ES can capture scarce expertise and distribute such expertise over a broad geographic area; many tasks require humans to operate in hazardous environments while ES may enable humans to avoid such environments. Expert systems are used by a variety of organizations as a major tool for improving productivity and quality (Turban and Aronson, 1998). For example, Digital Equipment Corp. (DEC) uses an ES called XCON in support of the VAX system configuration. Stanford University developed an ES called DENDRAL to infer the molecular structure of unknown compounds from mass spectral and nuclear magnetic response data. Ford Motor company uses an ES called "Direct Labor Management System" to improve efficiency in all phases of the production process (Awad, 1996).

The case-based reasoning (CBR) paradigm is based on the premise that expertise comprises experience and in solving new decision problems, decision-makers rely on their experience with similar decision problems. For example, a physician – after having examined a patient – gets a reminder about another patient that he treated before. If the reminder was caused by a similarity of important symptoms, the physician uses the diagnosis and treatment of the previous patient as a base and modifies it to incorporate the differences between the new and previous patient. Finally he determines the disease and treatment for the new patient. CBR systems have been adopted successfully in support of complex decision problems within a variety of decision environments (Watson, 1997). For example, a CBR system is used to improve jet engine maintenance and reduce cost in Snecma, a leading French manufacturer of aircraft engines. The project was designed to perform engine troubleshooting using CBR; it performs technical maintenance of the Cfm 56-3 aircraft engine on all Boeing 737s.

The state of the art of IT in support of organizational knowledge management initiatives reveals three common applications: (1) the coding and sharing of best practices, (2) the creation of corporate knowledge directories, and (3) the creation of knowledge networks (Alavi and Leidner, 2001). There are two schools of thought regarding externalization and codification of tacit knowledge. One school believes that tacit knowledge must be made explicit for sharing, and another school regards tacit knowledge as always tacit. For example, now the common form of knowledge management technologies is the electronic knowledge repository (Kankanhalli, et al., 2001). With knowledge repository technologies, organizations capture, organize, store and disseminate knowledge. For example, Ernst & Young has made significant investments in codification of the firm's internal knowledge and development of large knowledge repositories (Horvath, 2000). Andersen Consulting encourages employees to transfer their "tacit" knowledge to "explicit" knowledge in the form of written reports or video presentations (McCampbell et al., 1999). The explicit knowledge is then saved in repositories, such as databases and Intranet Web servers for users to access and use.

Since tacit knowledge is mostly stored in human beings, the chief characteristic of tacit knowledge is the difficulty in coding it so as to be shared. On the other hand, even coding the tacit knowledge successfully does not necessarily lead to improved performance and innovation, because knowledge management that focuses on creating network structures to transfer only explicit forms of knowledge will be severely limited in terms of the contribution to innovation (Swan et al., 1999). Even worse, attempts to codify tacit knowledge may only produce knowledge which is:

- Useless-- if it is too difficult to explain
- Difficult to verify-- if it is too uncertain
- Trivial--if it is too unimportant
- Redundant-if it is subject to continuous change
- Irrelevant to a wider audience--if it is context dependent
- Politically naïve--if it is too political sensitive
- Inaccurate--if it is too valuable and is therefore secreted by the "knower" (Swan et al., 1999)

To avoid problems related to codification of tacit knowledge, many new knowledge management strategies are toward transferring and exchanging tacit knowledge as tacit. In the network model of knowledge management systems, knowledge remains with the individual who develops and possesses it and is transferred mainly through person-to-person contacts. For example, Hoffman-LaRoche, a pharmaceutical company, has developed a knowledge map of its drug approval process (Lynne, 2001). For each step of the process, yellow pages listing relevant people organized according to their knowledge of the key issues are developed. This application of knowledge management is the creation of corporate directories, also referred to as the mapping of internal expertise (Alavi and Leidner, 2001). Because much knowledge in an organization remains uncodified, mapping the internal expertise is a potentially useful application of knowledge management (Ruggles, 1998). With this approach, people with one specified internal expertise can be retrieved from the directory, and then the user can contact this expert for specified knowledge. For example, one novice in Java can find the name and contact information from the yellow pages, and then ask the Java expert for advice on programming questions. But in real life, the issues of knowledge sharing are not all that simple. People discuss many issues with peers but not experts. For example, music fans discuss pop songs with each other. In this case, the yellow page could not help in narrowing the search of wanted peers, since all people in the yellow pages are like-minded.

Besides yellow pages, another important application of IT to organizational knowledge management is the creation of knowledge networks. Knowledge networks bring experts together so that important knowledge is shared and amplified (Alavi and Leidner, 2001). Next, along with discussing the concept of knowledge network and its applications, we will discuss the concepts of the informal knowledge network: community of practice and virtual communities.

2.3.1 Knowledge Network

Firms in technologically intensive fields rely on collaborative relationships among their knowledge workers to access, survey and exploit emerging technological opportunities (Powell, 1998). Network-like relationships within and between such firms are becoming common. For example, in automotive industries, more and more parts and components from stand-alone suppliers are linked into a system of industrial partnerships (Lodge and Walton, 1989). Under such circumstances, traditional knowledge management techniques are not enough to satisfy the increasing demand for knowledge sharing in support of organizational processes such as innovation and competition. To survive, corporations need a knowledge network that captures and stores pertinent knowledge, innovations and new ideas. They also need to distribute the stored knowledge to the decision makers on demand (Hogberg, 1998). To this end, the term "Knowledge networking" is used to signify a number of decision makers, resources and relationships among them, who are assembled in order to accumulate and use knowledge primarily by means of knowledge creation and transfer processes, for creating value. As shown in Figure 2.1, knowledge network makes it possible for valuable knowledge within the organization to be exchanged and advanced at the personal and group level (i.e., knowledge work processes) (Seufert et al., 1999). The structure and culture of the organization (i.e., facilitating conditions) compose the enabling and inhibiting environment for the creation and transfer of knowledge. Knowledge activities as well as information and communication tools is the tool-set (i.e., knowledge network architecture) supporting the social relationship (Von Krogh et al., 1997). A possible framework of knowledge network would include the following components:

1. Actors -- individuals, groups, organizations;
2. Relationships -- relationships between actors, which can be categorized by form, content and intensity; resources -- used by actors within their relationships;
3. Institutional properties -- including structural and cultural dimensions such as control mechanisms, standard operating procedures, norms and rules, communication patterns, etc.

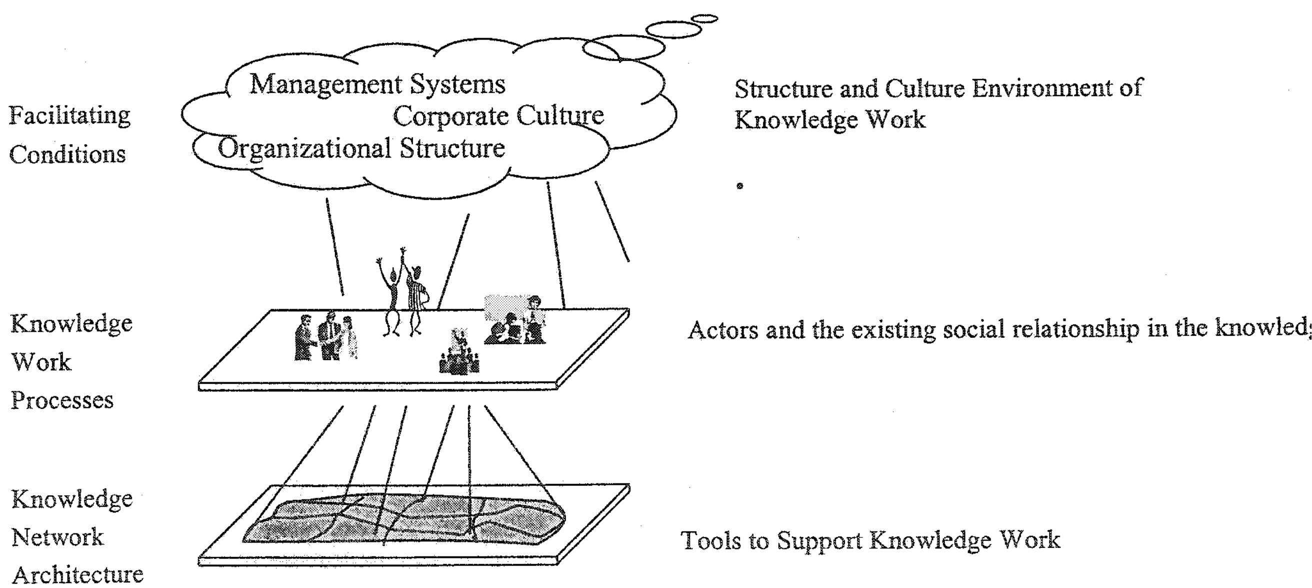


Figure 2.1: Framework of knowledge networks – a micro perspective (Adapted from Seufert et al., 1999)

The knowledge networking could yield great benefits. The openness and richness of networks is expected to foster a fertile environment for the creation of entirely new knowledge, while also accelerating the innovation rate. For example, Ericsson has developed many projects in knowledge networking: Image is used to align web-pages for easy Intranet searches; Knack offers web resources for competence development; Zopps is a general Ericsson knowledge base for off-duty staff; BIC offers business intelligence for middle-top management; Stargate offers web resources for competence and knowledge exchange. More details about these projects are given as follows (Hellström et al., 2000):

The Image initiative originates from Ericsson Radio, and has so far been developed by individuals outside and below the top-management level. The purpose of the Image initiative is to create a structured approach for standardizing and controlling intranet operations at Ericsson. This structure is then going to be applicable as a group-wide tool for intranet processes, rules, policies, etc.

Knack presents a more comprehensive and ambitious version of Competence Exchange. Knack, which is an educational portal on the Ericsson intranet, has a strong emphasis on KM and on providing learning resources for a number of possible users. From the Knack portal, materials, templates, information about internal courses and programs can be found as well as job listings and newsletters. There is also a "coffee shop" on the Knack site where discussion groups and specialist forums are maintained. Apart from these discussion groups, Knack offers competence inventories of experts in different fields.

Zopps is a type of "knowledge web" portal, or an Internet based competence boosting network outside the firewalls of Ericsson. The idea is that families of Ericsson employees, as well as the employee him/herself, can inconspicuously and spontaneously participate on the Zopps web pages, which contain among other things Ericsson information, and form interaction, reflection and knowledge transfer. Zopps provides families and employees with a "playground" for enhancing their computer skills as well as their knowledge of the company.

The BIC at Ericsson is built around an intranet portal, and consists of two core activities: (1) the EBIN network (Executive Business Intelligence Network) for top management, which is a password-protected "executive corner" for strategic information sharing; and (2) the BIC (Business Intelligence Center) for the purchasing, coordination and distribution of external information. Newly developed features include personalized news bulletins. Today the BIC is staffed by four full-time Help Desk personnel who act as "knowledge brokers" and who work intensively with regulars, plus a network of about 200 analysts, who, in addition to their normal roles in the production line, are also trained in business intelligence. They distribute and direct information between groups and individuals who possess or lack valued knowledge.

Stargate is a new initiative growing out of Ericsson Business Consulting, and is an on-line tool for systematizing strategic competence areas of consultants. Stargate originated in the need for re-using experiences and adding value to organizational and structural capital. For several years, "islands" of best practice have emerged at Ericsson, partly as results of the work on quality within the ISO 9000 certification efforts. Initially consultants specialized in SAP and I-Net were targeted and offered "introduction kits" with templates etc, but gradually Stargate expanded and encompassed other competence profiles and project documents as well. Additional content areas became customer and market segment information, products and services, agreements and resource planning.

Several knowledge network software tools are already on the market. For example, KnowNet (Hogberg, 1998) is software that can be used as an enabler of knowledge flow within an organization. The software captures and visualizes knowledge that the employee stores in order to increase the company's total skill availability. It offers an efficient and fast way to spread knowledge among employees. It also is a tool for creating a learning environment within an organization and enables knowledge workers to identify experts and areas of expertise within the organization.

Rather than being an issue of controlling and directing flows of knowledge, the task of managing knowledge networks is one of creating accessibility (Augier and Vendelo, 1999). For example, most of us have experienced the case that our specific skill in one aspect (e.g., selecting a book, a piece of music, or a company's stock from the stock market) can be enhanced by collaboration with like-minded individuals.

In organizations, people also use different kinds of informal communication methods to enhance learning within organizations. Informal networks provide critical channels for collective sensemaking and shared understandings (Lang, 2001). Evidence of such efforts can be seen in Japan, where "talk rooms" are deliberately established in which people meet to converse when they wish (Dougherty, 1999). Within organizations, informal networks of employees play a critical role in managing and transferring organizational knowledge. People share knowledge and work together to solve problems within these kinds of informal groups. We call these groups "communities of practice." In many organizations, people organize communities of practice to share knowledge and skills. These participants are motivated by a desire to use and develop their skills and competencies and to work together on issues of common interest (Regan and O'Connor, 2002). "Community of practice" is a hot term in contemporary knowledge management and refers to a theory that builds on learning as social participation (Wenger, 1998). In knowledge management, a change is occurring in how people think about who in the organization has credible and valuable knowledge that the organization can use to solve its difficult problems. There is evidence that this shift is a movement from the idea that knowledge is found only in a select group of experts or "best" practitioners, and toward the idea that useful knowledge is distributed throughout the whole of an organization (Dixon, 1999). With community of practice, knowledge transfer goes on between "like people" rather than flowing from the "best" to the "less able" (Dougherty, 1999).

Communities of practice have an informal membership that is often fluid and self-organizing in nature. They are formed over time by individuals with a need to associate themselves with others experiencing similar issues and challenges within the organization (Lesser and Prusak, 2000). Communities of practice are important because of the following:

- They provide the opportunity for decision-makers to develop a network of individuals with similar interests. This is particularly valuable as the organization grows "virtual" and individuals find it increasingly difficult to know "who knows what."

- They foster the interpersonal interactions necessary to build a sense of trust and obligation. By being able to bring people together to create and share knowledge, the community creates the condition where individuals can "test" the trustworthiness and commitment of other community members.
- They tend to be organized around common issues or themes to maintain the shared vernacular.

Intranet/Internet are now widely used by organizations to communicate and share knowledge. An intranet, which can be defined as a private network implemented using Internet concepts and technology to disseminate and exchange data, sound, graphics, and other media, is one of the concrete methods that organizations are using to change the way they communicate internally and share information (Stoddart, 2001). Some organizations already display a culture of connection – people regularly meet formally, exchange documents and e-mails, talk, share ideas and meet socially.

For example, IBM Global Services (Mertins et al., 2001) developed an Intranet-based community of practice to enhance knowledge sharing and creation. In order to get in contact with the community and to get the opportunity to identify the requirements of the community the members of the core team organize "ShareNet Meetings" where the participants get the chance to exchange and spread tacit knowledge. The ICM-tool (Intellectual Capital Management) supports the movement of the individual tacit knowledge of each of the numerous members of one network to explicit knowledge that is available to all members of the network. Work Room, Team Room, etc. are Lotus Notes applications that promote the cooperation of real and virtual (i.e., international) teams. In the framework of intellectual capital management (ICM), these applications facilitate the transfer of individual tacit knowledge into explicit public knowledge. A new, even more comfortable application called the "knowledge café" is now available within the ICM framework.

Buckman (Pan and Scarbrough, 1998) identified knowledge as one of the most important resources that contribute to the competitive advantage of an organization. K'Netix, Buckman's global knowledge transfer network, was introduced, supporting seven forums to coordinate Buckman's on-line conversation. By March 1993, every employee was able to access K'Netix, enabling Buckman associates to share knowledge, setting in motion the delivery of enhanced services to customers in over 90 countries worldwide in the form of virtual communities of practice.

2.3.2 Virtual Communities and Support Technologies

There has been considerable growth over the past two decades in the use of electronic discussion groups (Gray and Meister, 2001) -- such as bulletin boards, list-serves, collaborative media, discussion forums, instant messaging, and chat rooms -- as follows:

- **E-mail:** Full-service electronic mail (e-mail) systems send messages or documents from location to location. Usually email systems are considered one-to-one communication, but users also can use email to send messages to multiple recipients at the same time. This function supports people working in one group. Email is used to support the asynchronous (i.e., different time) communication among people.

- **Instant Messaging:** Allows users to see all the online users in a group communication. It is used to support the synchronous (i.e., same time) communication among users. A real-time electronic forum, visitors can meet others via Instant Messaging and share ideas on a particular subject.
- **Newsgroup:** A discussion group on the Internet, which is focused on exploring a particular topic. Discussion takes place by posting messages for everyone to read, having online conversations, and sending email messages to individuals or the group. There are thousands of newsgroups on different topics.
- **BBS:** Bulletin Board System (Forums or Message Boards). Bulletin board systems offers one-to-many asynchronous communication. BBS is a computerized version of the bulletin boards found in stores and other public places, where people can leave messages and advertise things they want to buy or sell. BBS can be open to anyone or restricted to registered users only. Some BBS can be searched and some allow image posting. The main topics in a typical discussion forum are listed along with the date of the last message posted in that topic. Choosing one of the topics either opens a list of subtopics or goes directly to the discussions. The messages themselves may be in chronological order or reverse chronological order (Notess, 1999).
- **Chat Room:**

There are many chat programs available that can be installed on a Web server or used via a chat hosting company. Some are free (ad-based) services and others range widely in price and features. Rather than the one-way interactivity of guest books or message forums, chat promotes live two-way (or more) interactive discussions. One user types a message, and another user can respond even while the first is still typing. Typically, chat sessions are not archived, so they're not searchable unless someone chooses to record the session and post it on the Web (Notess, 1999).

Interaction via these existing electronic discussion group technologies entice millions of people online (Preece, 2002). Communities of practice are moving beyond face-to-face exchanges, to interact in online environments, shared Web spaces, email lists, discussion forums, and synchronous chats (Millen et al., 2002). One of the fast growing, high-tech office trends today is "virtual team" or "virtual community." The conception of virtual communities is often that of a virtual place in which people can meet to socialize, exchange experiences, and enjoy the possibility of establishing relationships without having to expose the physical self (Holmström, 2001; Tung et al., 2001). The team crosses time, space, and cultural boundaries and does so effectively with the use of technology (Johnson et al., 2001). An on-line community is a group of people who use computer networks as their primary mode of interaction. Virtual communities encourage a diversity of participants to share their knowledge as a specific subject. Geography is expected to have no effect on an online community, where people can participate 24 hours a day at their convenience. Such subject-specific virtual meeting places are seeded with content by virtual community "hosts" whose job it is to draw new and existing members into the conversation (Barnatt, 1998). A virtual community of practice offers an excellent opportunity for members in different geographical locations to engage in a focused conversation about the future (Michelle, 2001). For example, by using a virtual community as a means to reach the expertise of experienced gamers, Daydream was able to get valuable input in the product development process. Daydream involved their customers in the development process of the online game Clusterball™ (Holmström, 2001). In online music message boards, music lovers share their interests and knowledge about music with each other (www.mp3.com).

The use of virtual teams is increasingly popular, especially when the members are in different geographic locations. With more users in the virtual communities, finding the right person to contact is becoming challenging. The selection of the right person to contact is a highly personal experience in the virtual communities, and very difficult when there are thousands of members with different knowledge about the specific topic of discussion. Therefore, an effective supporting technique is needed to help users find like-minded people (buddies). A popular research effort in helping find like-minded people in communities is collaborative filtering. Collaborative filtering provides computer-based support for the forwarding of information to others who might be interested in the information (Maltz and Ehrlich, 1995). Collaborative filtering is used in a large number of online companies, such as Amazon (www.amazon.com), Levi's (www.levis.com), and Moviecritic (www.moviecritic.com) (Good et al., 1999). Collaborative filtering works roughly as follows: networked minds provide information concerning their likes or dislikes in the form of ratings. These ratings are aggregated and are then used to compile recommendations for particular items (Maltz and Ehrlich, 1995). For example, a recommender system for movies may recommend movies that received mostly "good' or "very good" ratings and it may not recommend movies that were mostly rated "bad." Social navigation uses data generated by crowds of networked minds and denotes movement from one item to another influenced by the activity of others. Also, a user may choose to read only those messages in a news group that were rated as good. In collaborative filtering, the subjects are requested to evaluate different items. Based on their evaluation of various items, the highest level of overlap indicates that these subjects have similar interests (i.e., they are buddies) (Hayes et al., 2001):

The basic idea of automatic collaborative filtering (ACF) can be shown using Figure 2.2. In this figure three users have all shown an interest in assets A, B & C (for instance they all have rented video A, B & C). This overlap indicates that these users have similar tastes. Further, it seems a safe bet to recommend asset D and E to User 1 because they are 'endorsed' by User 2 and 3 that have similar interest to User 1.

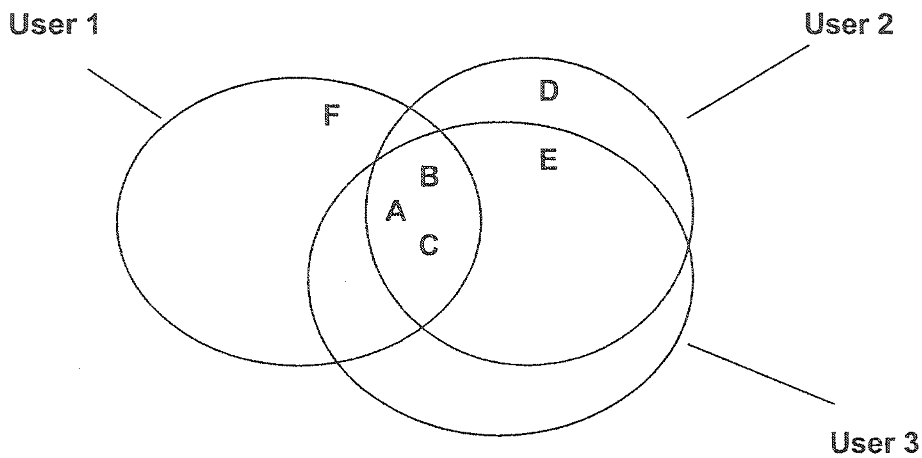


Figure 2.2: A Venn diagram showing interests of three users in assets ABCDEF (adapted from Hayes et al., 2001)

One of the greatest strengths of ACF is that, if enough data is available, good quality recommendations can be produced without needing representations of the assets that are being recommended. There are two distinct approaches to the ACF idea, that are termed invasive and non-invasive. With the invasive approach the user is explicitly asked to rate assets. This approach clearly contains more information (see Table 2.1). Non-invasive data contains less information and can be noisy in the sense that customers may not like some of the items they have used. This can be seen in Table 2.2.

Table 2.1: Data for use in ACF where users have explicitly rated assets (adapted from Hayes et al., 2001)

	A	B	C	D	E	F	G
User 1	0.6	0.6	0.8			0.8	0.5
User 2		0.8	0.8	0.3	0.7		
User 3	0.6	0.6	0.3	0.5		0.7	0.5
User 4					0.7	0.8	0.7
User 5	0.6	0.6	0.8			0.7	
User 6		0.8	0.8	0.7	0.7		
User 7	0.7	0.5			0.7		
User 8					0.7	0.7	0.8

Table 2.2: Data for use in ACF where users have not explicitly rated assets (adapted from Hayes et al., 2001)

	A	B	C	D	E	F	G
User 1	1	1	1			1	1
User 2		1	1	1	1		
User 3	1	1	1	1		1	1
User 4					1	1	1
User 5	1	1	1			1	
User 6		1	1	1	1		
User 7	1	1			1		
User 8					1	1	1

The collaborative filtering technique is a popular research effort used by many online shopping companies, but it is not a panacea for all situations. In the next section we will discuss the major problems and limitations of current electronic discussion group technologies in buddy finding. From there, we will introduce our agent-based buddy finding methodology.

2.4 The Basic Need for Collaborative Filtering

Virtual communities are becoming an increasingly important means for people to share and manage tacit knowledge. For example, in 2000, at least 8 million messages were unevenly distributed over 50,000 or more newsgroups devoted to every topic of possible human interest, from reselling Taiwanese household goods, to debating religion, to trading software (Smith, 2002). However, as Cothrel et al. (1999) found, in the virtual community, based on the study of 15 on-line communities in Europe, the key is to build community. Community members need to find the right person to ask questions and get timely responses. Therefore, "the development of personal relationships between team members is recognized as an important factor in enhancing effective working relationships among members" (Pauleen and Yoong, 2001, p190). For example, experienced Usenet users exploit their knowledge of other members (i.e., personal relationships) to find interesting discussions among those off-topic articles and less interesting discussions (Lueg, 2001). A key finding of collective action studies is that mutual awareness of other participants' histories and relationships is critical to useful cooperation among the members of community (Smith, 2002). In fact, with very low response to the requests in virtual communities, it is difficult for the members to get such kinds of awareness in the virtual community (Gould, 1999; Smith, 2002). Therefore, ignorance is a big burden in keeping virtual communities from optimal collaboration; those who have the knowledge are not aware that others may find it useful. Also those who could benefit from the knowledge of some of the community members are not aware of who has it (O'Dell et al., 1998).

Experiences with the Usenet indicate that people learn about other participants and their habits and interests over time by reading their public statements or by exchanging private e-mails (Lueg, 2001). A major problem with virtual communities is for people to find other members by means of posting, searching, reading and replying. It is time-consuming and frustrating for users to read all the postings in one usual message board. Information overload--including the overload of totally irrelevant information which the Internet provides--discourages many people from joining a virtual community (Geyer, 1996). More seriously, there are many occasions when "team members would send out questions and would never get back a response" (Gould, 1999). For example, Smith (2002, p53) found that "newsgroups are actually remarkably non-responsive, only about 2% of the message in these newsgroups are replies." Dron et al. (1998) found in an experiment using Usenet Newsgroups to accelerate the evolution of a learning resource within a group of students, that some questions remained unanswered and users could not get a solid picture from using the newsgroup. Reasons for such a low response rate are as follows. First, replying to a request requires time for processing and preparing a response. Members must expend effort not only in formulating a response to a request for advice; but also, in a large electronic discussion group, a potential advice-provider may have to read through many requests before finding one to which he/she can respond (Gray and Meister, 2001). Second, some participants post requests/responses that are unnecessarily long; or they lurk rather than contribute to the give-and-take that is an essential feature of any newsgroup; or they post off-topic requests/responses; or violate the local rules of decorum (Smith, 2002). Several researchers have developed some norms to guide users' behavior in electronic discussion groups. For example, Gordon (2000) suggested avoiding posting unsolicited commercial come-ons, avoiding flaming other participants, etc. But "monitoring for compliance with group norms is difficult in an electronic discussion group" (Gray and Meister, 2001). Moreover, information and knowledge are unlike most public goods in that their contribution has potential benefit to everyone except the individual who contributes it. When by definition the content of one's own advice cannot benefit oneself, the incentive to contribute is lower. Self-interested individuals would be motivated to receive others' knowledge but not to share their own (Gray and Meister, 2001).

With little sense of the presence of other people, individuals have a difficult time forming cooperative relationships (Smith, 2002). The existing electronic discussion group tools supply the platform for communication between users. Finding the right person to contact is still a trial and error procedure. To a great extent, the success of using forums depends on how lucky members are, and the process can be time-consuming and frustrating for the members. To this end, Gordon (2000, p12) suggests that "patience is critical to the success of using forums, since you may have to wait hours or even days for another user to stumble across and reply to your message rather than receiving an instantaneous response."

2.5 Issues Affecting Collaborative Filtering

As discussed in the previous section, one popular research effort in helping to build personal relationships in virtual communities is by means of collaborative filtering.

Two distinct approaches to collaborative filtering were discussed earlier: invasive and non-invasive (Hayes et al., 2001). In the invasive approaches, the user must evaluate every item. For example, to select music, after listening to songs, the users need to evaluate those songs. This creates an extra workload for the users. In the non-invasive approach, users do not need to evaluate each item, but, as mentioned earlier, "non-invasive data contains less information and can be noisy in the sense that customers may not like some of the items they have used" (Hayes et al., 2001, p239). Other limitations of collaborative filtering include (1) the relationship among users being based on item selections overlapping; (2) the centralized mechanism usually owned by one organization. In the following we discuss these two limitations. First, in collaborative filtering, only users with the highest level of overlap might be considered as buddies (Hayes et al., 2001). In that case, if users choose similar but not the same items (e.g., similar but not the same songs), since there are no overlaps between their selection, the users will never have opportunities to be identified as buddies with the collaborative filtering. That is, collaborative filtering may keep many potential users with similar interests from being buddies. Secondly, as a centralized mechanism, all customer data are stored in the server. Based on various analyses of the centralized data, the collaborative filtering determines the buddy list of customers and recommends products (Hayes et al., 2001). But the buddy relationships among members are private and users do not want to let others know about and control it. More importantly, for many cases in virtual communities, the data of users are naturally distributed, but not centralized. For example, music lovers use Napster within a distributed environment to share music with each other in a peer-to-peer mode. If a member wants a new music item, he or she types the music title or the author's name. Napster will search all distributed members' sites and return the addresses of music file locations. Then the music requester can download the desired music from a remote site. Napster offers members a platform to share music, but the user needs to search all members when he or she wants a piece of music. Napster cannot help users to organize into like-interest user groups. Within such a like-interest group, the user can recommend new music to their group members directly.

Let us look at another example, that of stock selection by a group of investors. Assume that a group of investors wants to help each other in selecting "favored" stocks. This group can start a news group and members can post the characteristics of their desired stocks so that others can suggest similar stock(s) they know about. For example, an investor posts news that he/she wishes to know about stocks similar to IBM. Those news-group members who care to read the news, search through their portfolio and identify stocks similar to IBM, and then post their view. This is a typical knowledge management issue that requires selecting supporting information technologies, such as a news group support system, to enable members to post their requirements/responses (e.g., see www.Etrade.com). The problem with news group systems is that: (1) members are unable to share their knowledge (i.e., suggesting possible stocks) if they miss the news (i.e., don't read the request for information). This results in a second problem: (2) how can we identify the group of investors who are most likely to provide a good response to the posted request?

It is commonly assumed that learning in firms is enhanced when technically skilled employees are encouraged to collaborate with like-minded individuals. To this end, we propose a combination of agent technology and distributed CBR systems in support of sharing knowledge among like-minded members. To place this in perspective, let us use the example of collaboration among investors to help each other in selecting a stock. In this case, each stock can be considered as a "case" and each investor's portfolio can be considered as a distributed case library. The request of an investor (e.g., for stocks similar to IBM) is communicated to the case-base of other investors to identify the stocks that closely match the characteristics of the requested stocks. The characteristics of matched stocks are sent back to the requesting investor as depicted in Figure 2.3. This process works well as long as there are a small number of investors (members). However, traffic generated by sending a message from one member to every other member is inefficient and can clog the network of an online brokerage firm such as Etrade, which has thousands of members. In this case, we need to identify the subset of members (termed buddies) who are most likely to have stocks similar to the requesting investor (see Figure 2.3). The objective of this research is to develop a methodology to identify buddies in the form of intelligent agents operating in a knowledge network decision environment, such as that which shares information about available stocks or shares music information among music lovers. Our proposed methodology is presented in chapter 4.

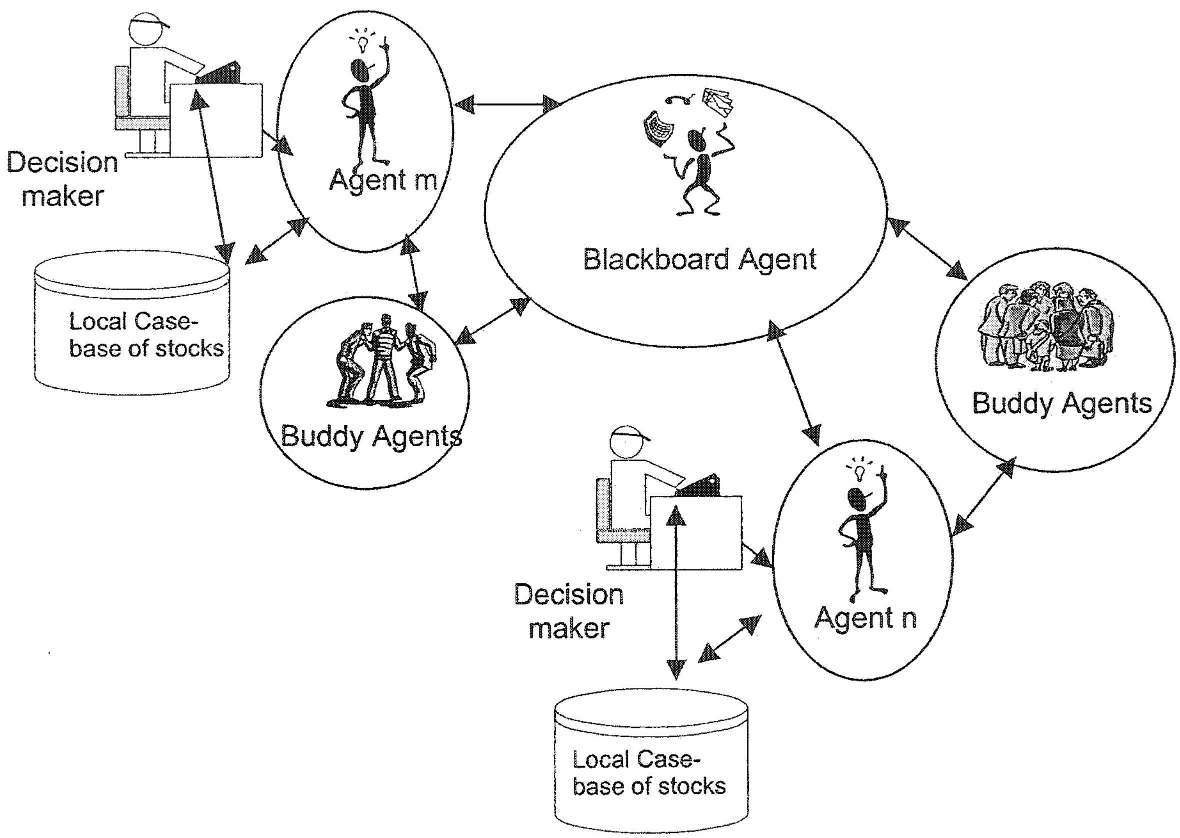


Figure 2.3: An overview of the MAS system in support of stock selection

3. Intelligent Agent

3.1 Introduction

With the development of computer technology, and especially with the advance of the Internet, information systems are becoming the vehicle for an increasing range of everyday activities. Through the computer, people can trade stocks, check email, chat with friends, play games, etc. More and more untrained people have become computer users. Intelligent agent, a technique from the field of artificial intelligence (AI), is expected to assist end users cope with increasing information overload (Maes, 1994). An intelligent agent can reduce the complexity of dialogue by understanding the goals of the user and assisting him/her to interact with the system (Lewis, 1998; Pilkington, 1992). Nowadays, agent-based technologies are considered the most promising means to deploy enterprise-wide and worldwide applications that often must operate across corporations and continents and inter-operate with other heterogeneous systems (Bellifemine et al., 2001). We can identify applications of agent technology in diverse areas such as information retrieval systems to help users to retrieve relevant documents (Shaw et al., 2002); and electronic commerce to help buy and sell (Turowski, 2002). More importantly, with the pricking of the Internet bubble, online retailers are under more pressure than ever to earn their keep, and as a result, many companies are looking at intelligent agents as one of the sophisticated merchandising tools that can recommend products and build customer loyalty and sales (Kwak, 2001).

Generally speaking, an intelligent agent is particularly useful in open and complex systems such as the Internet (Jennings and Wooldridge, 1998). In open systems, the system structure is capable of dynamic change. The availability, type, and reliability of information services are also constantly changing. Information can be ambiguous and possibly erroneous due to the dynamic nature of the information sources, and potential information updating and maintenance problems (Sycara et al., 1996). Therefore, the huge amount of information poses challenges to decision-makers because of the accompanying difficulty in collecting, filtering, evaluating and using it. For example, many institutions (e.g., www.etrade.com) enable investors to purchase common stocks online from their Internet site. However, the onus is on the customer to have perfect knowledge of thousands of common stocks traded in different exchange centers (e.g., The New York Stock Exchange). This renders the online information market somewhat inefficient and sets the stage for the emergence of information "intermediaries" in the market. We can use software agents to act as "intermediaries" in support of customer requirements.

In complex systems, the most powerful tools for handling complexity in software development are modularity and abstraction. Agents are a powerful tool for making systems modular. With a multi-agent system, a designer can partition a complex task into several small and relatively independent subtasks. Each agent then performs a specific subtask. For example, several agents work collaboratively to perform portfolio analysis for the stock trader in the WARREN system (Decker et al., 1998).

Successful application of intelligent agents in support of decision-making processes is contingent on two critical phases. First, one needs to identify the decision-making processes that can best be supported by the agent methodology. The second phase requires the appropriate use of technology in the development of pertinent agent systems. This chapter provides an overview of intelligent agent research-and-development environments. Section 2 elaborates on the characteristics of agents. There are almost as many opinions on the definition of agents as there are agents themselves. The diversity of agent definition can be attributed to the range of applications that can use this technology to enhance decision-making processes. In this chapter, agent and intelligent agent refer to the same type of application system and are used interchangeably. Agents have to interact with each other as well as with environmental entities (e.g., human decision-makers and databases) to achieve their goals. Three basic agent architectures are described in section 3. Section 4 describes an agent communication language called Knowledge Query and Manipulation Language (KQML). One of the basic problems facing designers of multi-agent systems for open and complex information environments such as the Internet is that of connection: finding the other agents who might have the required information to deal with a decision problem. To this end, section 5 describes the architecture of multi-agent systems and pertinent coordination strategies. In section 6, we discuss the current architecture and existing limitations. Section 7 provides concluding remarks and examines the challenges inherent in the development of agent-based systems in support of decision-making processes.

3.2 Decision-making and Intelligent Agents

Decision-making is a process of choosing among alternative courses of action for the purpose of attaining a goal or goals (Turban and Aronson, 1998). According to Simon (1977), there are three major phases involved in the decision making process: intelligence, design, and choice. The decision-making process starts with the intelligence phase, where reality is examined and problem is identified and defined. In the design phase, a model that represents the system is constructed. This is done by making assumptions that simplifies reality and by writing down the relationship among all variables. The choice phase includes selection of a proposed solution to the model (not to the problem it represents). Once the proposed solution seems to be reasonable, we are ready for the last phase: implementation. Successful implementation results in solving the real problem.

Various kinds of technologies are developed to support each phase of decision-making process. Decision support systems (DSS) allow people at many different levels to systematically analyze problems before making a decision. In the process, these systems extend the range and capability of the decision-making process, increasing its effectiveness (Gallegos, 1999). Especially, the intelligence phase is a primary target for DSS and for other computer-based information systems that deal with nonstructured problems (Lucas, 1995). The primary requirement of decision support for the intelligence phase is the ability to scan external and internal information sources for opportunities and problems and to interpret what the scanning discovers. Nowadays, the worldwide marketplace provides not only more customers, suppliers and competitors, but also increased complexity for the decision-making process (Sauter, 1999). Internet-based electronic transactions take place actively worldwide and the transaction amount is continuously increasing day by day (Kang and Han, 2003). As an excellent information source, the Internet provides significant opportunities for people to obtain information. Electronic information services are pitched to a wider range of decision-`makers, from CEOs and CIOs on down to the end users themselves (Curle, 1998). At the same time, the Internet also brings about the problem of information overload (Chen et. al, 2002).

Information overload results from the inability of living systems to process excessive amounts of information. The cognitive limitations of humans make it impractical to consider all possible alternatives to a particular problem. Even if we could review all relevant alternatives, we would not be able to assimilate all the information so that we could make an appropriate decision (Marakas, 2003). As the complexity of the task or information load increases, the human information processor tries to reduce cognitive effort by changing to a more effective information-processing strategy. People try to minimize the effects of information overload by employing conscious or even unconscious strategies to reduce information load (Grise and Gallupe, 1999). We tend to "simplify reality" by focusing our energy on finding a solution that meets our preconceived notion of what an acceptable solution looks like. Upon finding such a solution, we immediately adopt it and stop to looking for a better one (Marakas, 2003).

On the Internet, the staggering amount of information has made it extremely difficult for users to locate and retrieve information that is actually relevant to their task at hand. Given the bewildering array of resources being generated and posted on the

WWW, the task of finding exactly what a user wants is rather daunting. Although many search engines currently exist to assist in information retrieval, much of the burden of searching is on the end-user. A typical search results in millions of hits, many of which are outdated, irrelevant, or duplicated (Ram, 2001). One promising approach to managing the information overload problem is to use "intelligent agents" for search and retrieval (Ram, 2001). Agents will interpret user requests and automate manual processes. Agents will allow users to delegate simple tasks. Users will have time to solve complex, abstract problems, while agents use their knowledge of user preferences, standard domain defaults, and networked information sources to make simple decisions and even take action on behalf of the user (Dyer, 1999). For example, an agent might remind or automatically prompt a person to email Joe, find an article on IBM's new chip, or buy Yahoo stock when it drops to $24.00. In a more technical vein, agents are atomic software entities operating through autonomous actions on behalf of the user without constant human intervention (Ma, 1999).

3.3 Characteristics of Intelligent Agents

There is currently no general consensus on the definition of an agent (Serugendo, 2001). Different researchers have given different definitions based on their practices and understandings. Here, we will discuss agents mainly from a practical view and investigate the major characteristics of agents.

Intelligent agents work in open and complex information environments (Jennings and Wooldridge, 1998). In complex systems, the most powerful tools for handling complexity in software development are modularity and abstraction. The agent paradigm and multi-agent systems (MAS) are widely recognized as suitable abstractions to deal with complex application environments, especially when the openness and unpredictable dynamics of the environment make traditional approaches less effective (Ricci et al., 2001). In such an information environment, the structure of the system itself is capable of dynamically changing. In order to achieve the goal of the user, the agent performs the following actions (Reticular, 1999): executes autonomously; communicates with other agents or the user; and monitors the state of its execution environment. Its components are not known in advance, can change over time, and may be highly heterogeneous. To be an intelligent agent, Newell argues that software should possess the following capabilities or attributes (Newell, 1988):

- Be able to exploit significant amounts of domain knowledge.
- Be tolerant of errorful, unexpected, or wrong input.
- Be able to use symbols and abstractions.
- Be capable of adaptive, goal-oriented behavior.
- Be able to learn from the environment.
- Be capable of operation in real-time.
- Be able to communicate using natural language.

In fact, not all of the above features are needed for all intelligent agents. Hayes-Roth (1995) views intelligent agents as having the capability to perform three necessary functions:

- To perceive dynamic conditions in the environment.

- To take action to affect conditions in the environment.
- To reason in order to interpret perceptions, solve problems, draw inferences, and determine actions.

Researchers have described the characteristics of, and classified agents in, numerous ways. Nwana (1996) provides a typology that defines four types of agents based on their abilities to cooperate, learn, and act autonomously. Autonomy refers to the principle that agents can operate on their own without the need for human guidance. With cooperation capability, agents can interact with each other and possibly humans via some communication language and coordinate their actions without cooperation. The key attribute of any intelligent being is its ability to learn. Smart agent systems would have to *learn* as they react and/or interact with their external environment. Nwana (1996) terms these smart agents, collaborative agents, collaborative learning agents, and interface agents (Nwana, 1996). Figure 3.1 depicts how these four types of agents utilize the capabilities described next.

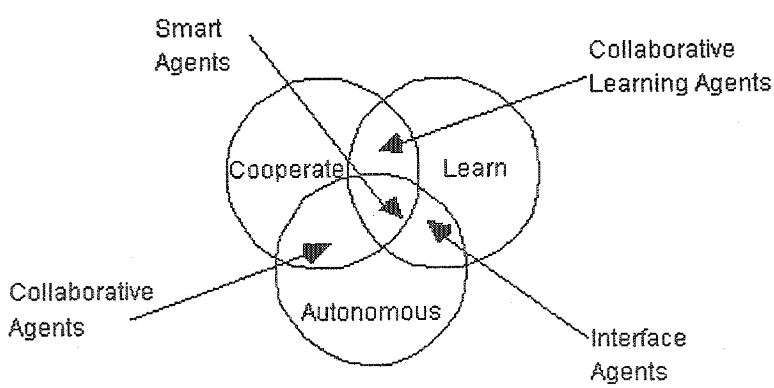


Figure 3.1: A Typology of Agents (Nwana, 1996)

- **Collaborative Agents**

To deal with complex real world problems, it is desirable to have different types of agents specializing in different types of tasks to collaborate with others to solve a problem (Ram, 2001). Collaborative agents emphasize autonomy and cooperation to perform tasks by communicating, and possibly negotiating, with other agents to reach mutual agreements. For example, the Collaborative Agent Interaction and synchronization (CAIRO) (PenÄa-Mora et. al, 2000) system provides an environment for structured information exchange across the Internet in real-time. The major component of the CAIRO system is the set of distributed artificial intelligence based software agents. The agent removes some level of direct involvement in running a meeting. The CAIRO system allows designers and engineers to work together in virtual teams by supporting multi-media interactions over computer networks. CAIRO aids the concurrent engineering effort by relaxing the physical, temporal and organizational constraints experienced in traditional design meeting environments.

- **Interface Agents**

Interface agents are autonomous and utilize learning to perform tasks for their users. This class of agent is used to implement assistants as well as guides, memory aids, and filters; perform matchmaking and referrals; or buy and sell on behalf of the user (Conway and Koehler, 2000; Reticular, 1999). For example, Letizia (Lieberman, 1997) is a user interface agent that assists a user browsing the World Wide Web. As the user operates a conventional Web browser such as Netscape, the agent tracks user behavior and attempts to anticipate items of interest by doing concurrent, autonomous exploration of links from the user's current position.

- **Collaborative Learning Agents**

A typical example of Collaborative Learning Agents is a robotic soccer system by Stone and Veloso (1998). In this collaborative system, teams of agent players must work together to put the ball in the opposing goal while at the same time defending their own. Learning in this system is divided into two levels. First the agent players learn to acquire some low-level skills that allow them to manipulate the ball. Second, they must learn to work together to achieve the common goal of winning.

Besides the above classification, researchers also classfied agents based on other dimensions such as their mobility (i.e., by their ability to move around telecommunication networks. This yields the classes of static or mobile agents). Mobile agents are computational processes capable of moving over a network (e.g., a wide area network such as the Internet or World Wide Web); interacting with foreign hosts; gathering information on behalf of the user; and returning to the user after performing their assigned duties. Mobile agents are increasingly used in various Internet-based applications such as electronic commerce, network management, and information retrieval (Kim et al., 2001). For example, TabiCan (www.tabican.ne.jp), one application of IBM's Aglet (www.trl.ibm.co.jp/aglets), offers several merchant agents for companies selling tickets online. When a user accesses TabiCan, a consumer agent is created and interacts with the merchant agent to find travel information.

The most popular uses for intelligent agents are finding, analyzing and retrieving large amounts of information (Krishnan et al., 2001; Rhodes and Maes, 2000; Tu and Hsiang, 2000). Information agents are tools to help manage the tremendous amount of information available through communication networks. Information agents access the network looking for particular kinds of information, filter it, and return it to their users. For example, WARREN (Decker et el., 1996) has six information agents: two stock ticker agents using different WWW sources; a news agent for Clarinet and Dow-Jones news articles; and an agent that can extract current and historical sales and earnings-per-share data from the SEC Edgar electronic annual reports.

3.4 Agent Architecture

Agent architectures are essentially design methodologies: they are technological frameworks and scaffolding for developing agents (Bryson and Stein, 2001). The architecture of an agent describes its modules and capabilities. Usually three types of architectures are distinguished according to the agent paradigm as follows (Botti et al., 1999; Müller, 1997):

- Reactive agents

  Agents that are built according to the behavior-based paradigm, that have no or only a very simple internal representation of the world, and that provide a tight coupling of perception and action.

- Deliberative agents or belief-desire-intention (BDI) systems

Agents in the symbolic artificial intelligence tradition that have a symbolic representation of the world in terms of categories such as beliefs, goals, or intentions, and that possess logical inference mechanisms to make decisions based on their world model.

- Hybrid agents

  Agents that are built from two or more subsystems. One is deliberative (i.e., containing a symbolic world model) and the other is reactive.

Each BDI agent has a sophisticated reasoning architecture that consists of different modules that operate asynchronously. Reactive agents do not have representations of their environment and act using a stimulus-response type of behavior; they respond to the present state of the environment in which they are situated. Reactive systems are mainly used in rapidly changing environments. Nonetheless hybrid agent systems can be used for most application problems since neither a purely deliberate nor purely reactive architecture is appropriate (Sycara, 1998).

3.5 Agent Communication Language

The central idea underlying software agents is that of delegation. To delegate is to entrust a representative to act on one's behalf (Norman and Reed, 2001). The user delegates a task to the agent and the agent autonomously performs that task on behalf of the user. For delegation to be successful, there must be a relationship between the agent delegating the goal or task and the agent to whom it is delegated (Norman and Reed, 2001). The act of delegating can be carried out by the performance of communication. In multiagent systems, if agents are not designed with an embedded knowledge about the beliefs, intentions, abilities and perspectives of other agents, they need to exchange information to improve their social activities (Dragoni et al., 2001). Knowledge Query and Manipulation language (KQML) is designed to support interactions among intelligent software agents. KQML offers an abstraction of an information agent (provider or consumer) at a higher level than is typical in other areas of computer science. KQML assumes a model of an agent as a knowledge-based system (KBS) (Finin et al., 1994b). The KBS model easily subsumes a broad range of commonly used information agent models, including database management and hypertext systems, server-oriented software (e.g., finger demons, mail servers, HTML servers), and simulations. Figure 3.2 summarizes the possible components of an agent; they are grouped into representation components, communication components, and components that are not directly related to shared understanding (Finin et al., 1997).

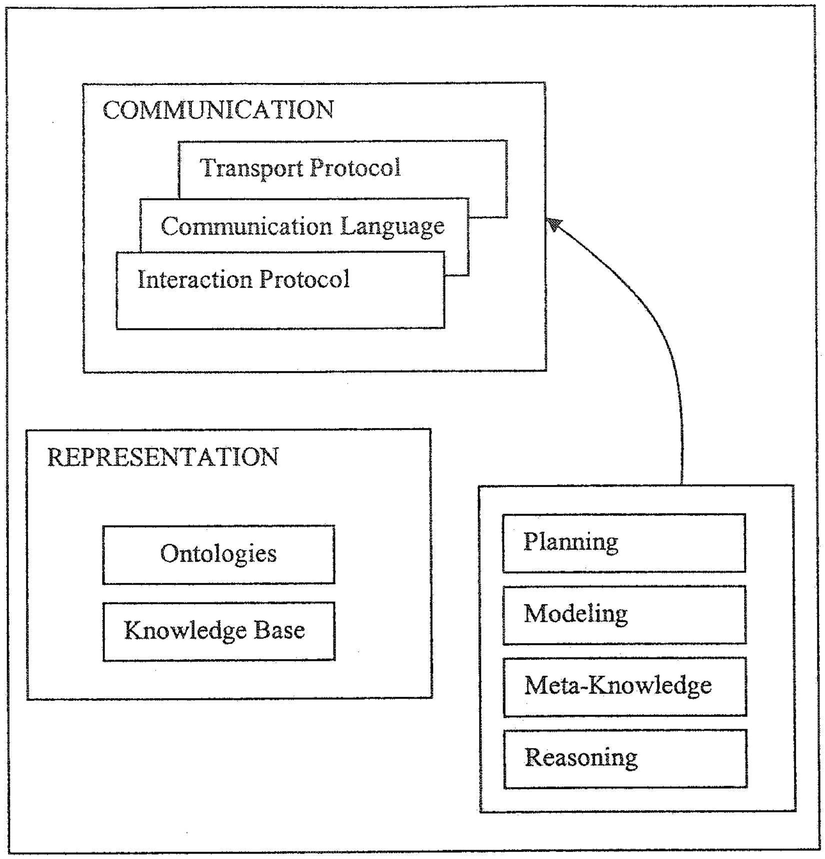


Figure 3.2: An abstract model for interoperating software agents with three classes of components: representation components, communication components, and components not directly related to shared understanding. (Finin et al., 1997)

KQML is most useful for communication among agent-based programs, in the sense that the programs are autonomous and asynchronous. Autonomy means that agents may have different, and even conflicting, agendas. Thus, the meaning of a KQML message is defined in terms of constraints on the message sender rather than on the message receiver. This allows the message receiver to choose a course of action that is compatible with other aspects of its function.

KQML language can be viewed as being divided into three layers: the content layer, the message layer and the communication layer. See Figure 3.3.

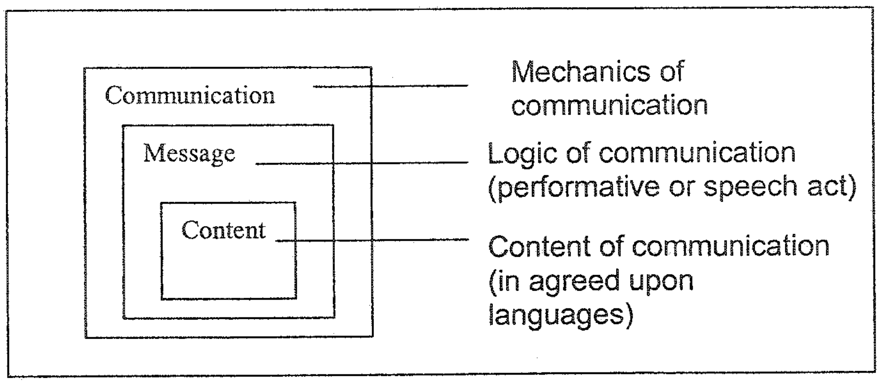


Figure 3.3: The three layers of the KQML communication language (Reticular, 1999)

- The content layer is the actual content of the message in the program's own representation language. KQML can carry any representation language, including languages expressed as ASCII strings and those expressed using a binary notation. All of the KQML implementations ignore the content portion of a message except to the extent that they need to determine its boundaries.
- The communication layer encodes a set of features to the message which describe the lower level communication parameters, such as the identity of the sender and recipient, and a unique identifier associated with the communication.
- The message layer forms the core of the language. It determines the kinds of interactions one can have with a KQML-speaking agent. The primary function of the message layer is to identify the protocol to be used to deliver the message and to

supply a speech act, or performative, which the sender attaches to the content. The performative signifies that the content is an assertion, a query, a command, or any other function in a set of known performatives. Because the content is opaque to KQML, this layer also includes optional features which describe the content: its language; the ontology it assumes; and some type of more general description, such as a descriptor naming a topic within the ontology. These features make it possible for KQML implementations to analyze, route, and properly deliver messages even though their content is inaccessible.

A KQML message consists of a performative, its associated arguments which include the real content of the message, and a set of optional arguments. The main focus of KQML is on its extensible set of performatives, which defines the permissible operations that agents may attempt on each other's knowledge and goal states at run time. The performatives comprise substrata on which to develop higher-level models of inter-agent interaction such as contract net and negotiation.

The contribution that KQML makes to Distributed AI (DAI) research is to offer a standard language and protocol that intelligent agents can use to communicate among themselves as well as with other information servers and clients. Permitting agents to use whatever content language they prefer makes KQML appropriate for most DAI research.

3.6 Architecture of Multi-Agent Systems

Multi-Agent systems are groups of agents that work together as a single system to integrate their functionality. They consist of a group or groups of agents that interoperate, cooperating to execute large complex tasks (Nodine et al., 2001). In the open and dynamic environment, each agent needs to collaborate with other agents. Therefore a fundamental agent requirement is the ability to coordinate its own actions with those of other agents (Durfee, 2001). Coordination entails managing dependencies between activities (Schumacher, 2001). There are many kinds of research on the coordination problem related to organizations or even virtual organizations. For example, Fernandez and Wijegunaratne (1998) studied the cooperation approach in distributed applications, Bernus and Uppington (1998) demonstrated the coordination in a virtual enterprise, and Flores et al. (2001) developed the architecture for multi-agent coordination and cooperation. But of all these multi-agent system architectures, the major basic structures are two: centralized and decentralized (Sikora and Shaw, 1998). Next, we explain these two main kinds of control structures used in multi-agent systems coordination.

3.6.1 Control Structures and Coordination Mechanisms

One of the basic problems facing designers of open multi-agent systems for the Internet is the connection problem (Davis and Smith, 1983)--finding the other agents who might have desired preferences and capabilities. Preference is (meta) knowledge about what types of information have utility for a requester, both in form (e.g., John follows the price of IBM) and in other characteristics (e.g., John wants only free information; John wants stock quotes at least every 35 minutes). Capability refers to (meta) knowledge about what types of requests can be serviced by a provider (e.g., Mary can provide the current price of any NASDAQ stock, delayed 15 minutes, for free, at a rate of 10 quotes per minute). There are basically two kinds of control structures in multi-agent systems (Mařík et al., 1999; Sikora and Shaw, 1998): centralized control and decentralized control. In centralized control, there is a central coordinator to whom everyone communicates solutions. The coordinator, therefore, handles the interdependencies among the agents. Usually, either they reply on some quantitative measures of utility, or their replies are based on a qualitative notion of interrelation (Ossowski, 1999). Decentralized control is the most common form of control structure in distributed systems. There is no coordinator in decentralized control. A solution to a coordination problem constitutes equilibrium, a compromise that assures somehow "maximal" attainment of the different interests of all involved individuals (Ossowski, 1999).

3.6.1.1 Centralized Control

In centralized control, there is a central coordinator called middle agent that handles interdependences among agents (See Figure 3.4) (Finin et al., 1994).

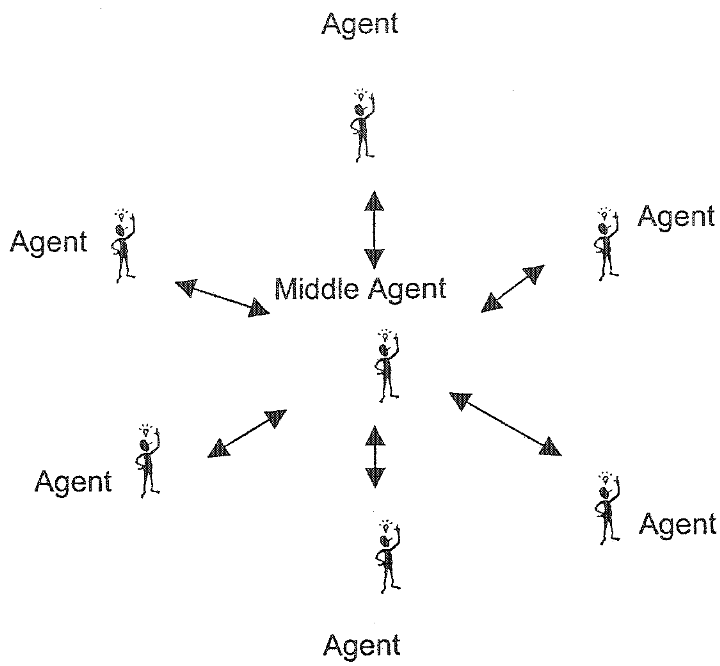


Figure 3.4: Agents Communicating through a Middle Agent

There are several types of middle agents, including matchmaker agents, blackboard agents, broker agents and yellow pages agents (Sycara et al., 1997).

- A broker agent protects the privacy of both preferences and capabilities, and routes both requests and replies appropriately.
- A matchmaker/yellow-pages agent stores capability advertisements that can be queried by requesters. The requesters then choose and contact directly any provider they wish.

- A blackboard agent keeps track of requests. Requesters post their problems and providers can then query the blackboard agent for events they are capable of handling.

For example, Retsina (Sycara et al., 1996) uses a distributed collection of software agents that cooperate asynchronously to perform goal-directed information retrieval and integration for supporting a variety of decision-making tasks. Each user in the Retsina framework is associated with a set of agents that collaborate to support the user in various tasks and act on his or her behalf. The agents are distributed and run across different machines. They have access to models of the task and information-gathering needs associated with different steps of the task. Based on this knowledge, the agents decide a) how to decompose and delegate tasks; b) what information is needed at each decision point, and c) when to initiate collaborative searches with other agents to get, fuse, and evaluate the information. Retsina uses three types of agents: interface, task, and information as follows.

- Interface agents interact with the user by receiving user specifications and delivering results. They acquire, model, and utilize user preferences to guide system coordination in support of the user's tasks. The main functions of an interface agent include: (1) collecting relevant information from the user to initiate a task; (2) presenting relevant information including results and explanations; (3) asking the user for additional information during problem solving; and (4) asking for user confirmation, when necessary.

- Task agents support decision making by formulating problem-solving plans and carrying them out through query and exchange of information with other software agents.
  - Information agents provide access to a heterogeneous collection of information sources. These agents have models of the associated information resources, and strategies for source selection, information access, and conflict resolution and information fusion.

Agents are distributed across different machines in Retsina that use a matchmaker structure. Agents that can provide services advertise their capabilities to the matchmaker. An agent queries the matchmaker when looking to find another agent with a specific capability-- one that can supply particular information or achieve a problem-solving goal. The matchmaker either returns appropriate lists of agents matching the query description, or returns "null" if it finds no match.

3.6.1.2 Decentralized Control

The majority of MAS work deals with systems in which agents are peers of each other (Turner and Jennings, 2001) with a common form of decentralized control (Sikora and Shaw, 1998). See Figure 3.5. The agents have to interact among themselves, exchanging information and coordinating their interdependencies without the help of a middle agent. In practice, however, due to the communication costs and information overload, each agent is allowed to communicate only with a small subset of other agents

(Sikora and Shaw, 1998). The information about capabilities and behavior of other agents is stored in each individual agent. We call the stored information about other agents an "acquaintance list." In the acquaintance model, the individual agent contains information on the current capabilities of peers of the agent (Mařík et al., 1999).

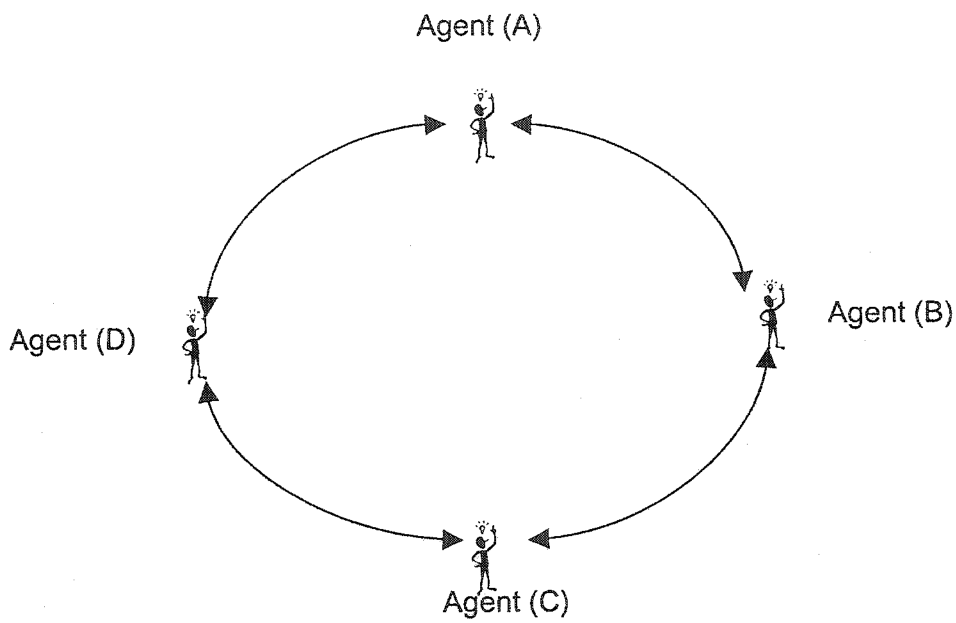


Figure 3.5: Agents Communicating with Each Other through Peer-to-Peer Mode

Mařík et al. (1999) suggested the tri-base acquaintances model. It can be viewed as a specific knowledge-based system that is able to combine permanent and temporary knowledge with facts stored in distributed databases. This approach suggests organizing the relevant information about cooperating agents into three separate information bases in the agent's wrapper as follows:

- *The Co-operator Base* stores static information on the peer agents, such as their IP addresses, communication means and predefined responsibilities. Moreover, it specifies subscribed agents – agents that are subscribed to report their status change.
- *The Task Base* contains knowledge concerning possible task decomposition with respect to problem solving processes. It is further split into two separate sections. The problem section contains general knowledge on possible task decomposition and contingent time precedence and prerequisites. The plan section stores deduced plans on how to solve particular tasks through co-ordination of sub-contracted helping agents.
- The *State Base* reflects the actual peer agents' states that may evolve rapidly in time. The agent section of the state base reflects the internal states of the peer agents like their current load, attainability and trust, as well as capabilities (e.g., speed and price of processing) and schedule of the considered agent. The task section describes the current states of the solution of the tasks that have been contracted and are coordinated by the agent. The peers are expected to report the solution progress.

This structure makes it possible for the task base to have up-to-date information on the current capabilities of the peers. This facilitates directing the co-operation requests to the most suitable agent in the community. Therefore, the communication traffic is significantly reduced and the system response is very fast since non-addressable task announcements are avoided.

3.6.2 Coordination Mechanisms

Co-operative multi-agent systems offer a novel approach to handle complex integration tasks. All participants in a coordination process have interdependencies. Coordination entails managing dependencies between activities (Schumacher, 2001). Ossowski (1999) has classified agent coordination into three groups: multi-agent planning, negotiation, and organization.

- Multi-agent planning

  With multi-agent planning, agents form plans that specify all their future actions and interactions with respect to a particular objective: all agents involved in a multi-agent plan commit to behave in accordance with it. This plan describes all actions that are required to achieve the respective goals of agents. The planning can be centralized or decentralized.

- Negotiation

  Negotiation is seen as a method for coordination and conflict resolution (Sycara, 1998), and a process by which two or more parties make a joint decision (Zhang et al., 2001). The parties first verbalize and then move toward an agreement through a process of concession formation; or they search for new alternatives (Mueller, 1996). Negotiation processes dynamically generate an agreement, which usually lasts shorter than a priori commitments that organizational structures imply. Still, agreements can be re-negotiated. The most promising application areas for agent negotiation include retail e-commerce, electricity markets, bandwidth allocation, manufacturing planning and scheduling in subcontracting networks, distributed vehicle routing among independent dispatch centers, and electronic trading of financial instruments (Sandholm, 1999).

- Organization

  Organization is usually seen as a metaphor for a set of long-term structural relationships between roles. A role determines the expectations about the agent's individual behavior by describing the agent's responsibilities, capabilities and authority inside the MAS. When an agent agrees to play certain roles within an organization, they commit to comply with the behavior that these roles and their relationship imply.

Agents can improve coherence by planning their actions. Planning considers the constraints that the other agents' activities place on an agent's choice of actions, the constraints that an agent's commitments to others place on its own choice of actions, and the unpredictable evolution of the world caused by other unmolded agents. One direction of research in cooperative multiagent planning has been focused on modeling teamwork explicitly (Sycara, 1998). The joint-intentions framework (Cohen & Levesque, 1991) focuses on characterizing a team's mental state, called joint intention. A team jointly intends a team action if team members are jointly committed to completing that team action, while mutually believing that they are doing it. The model of SHAREDPLAN

(Grosz and Sidner, 1990) is not based on a joint mental attitude but rather on a new mental attitude intending that an action be done. However, an individual agent's intention is directed towards its collaborator's actions or towards a group's joint action. Intention is defined using a set of axioms that guide a teammate to take action or enter into communication that enables or facilitates its teammates to perform assigned tasks (Tambe, 1997).

Negotiation is a coordination mechanism used in the distributed environment. Negotiation means a discussion in which the interested parties exchange information and come to an agreement (Davis and Smith, 1983). For example, CAP II is an office automation agent for time management. The agent works in the background as a personal digital assistant. Since much of the office work is performed with the cooperation of different people, the agent also models the workflow, including simple sequence work and complex negotiation work. CAP II performs the negotiation of meetings between the attendees by sending email messages back and forth. Agents operate strictly locally in a purely reactive manner without any planning. Each agent uses its owner's calendar but does not have access to the calendars of other participants. All synchronization works through communication and negotiation. Planning (i.e., deliberative behavior) is not necessary. Once the user has indicated the desire for a meeting, the CAP II agent generates a proposal and sends it to all invited attendees. If the recipients possess a CAP II system, their agent may negotiate until they find a commonly accepted date. Replies are called "bids." A CAP II agent accepts Yes, Not-then, Maybe, or No bids. If an agreement has been found, an additional "handshake" procedure of sending confirmation and validation message follows. In case those responses are missing, a time-out handler generates and sends remainder messages or, if unsuccessful, notifies its user so that he can contact the attendee directly (Bocionek, 1995).

An organization provides a framework for agent interaction through the definition of roles, behavior expectations, and authority relations. An organization gives each agent a high-level view of how the group solves problems and indicates the connectivity information to the agents so that they can distribute subproblems to competent agents. Examples of organizations include the following (Sycara, 1998):

- Hierarchy

  The authority of decision making and control is concentrated in a single problem solver (or specialized group) at each level in the hierarchy. Interaction is through vertical communication from superior to subordinate agent; and vice versa. Superior agents exercise control over resources and decision making.

- Community of experts

  This organization is flat, where each problem solver is a specialist in some particular area. Agents coordinate through mutual adjustment of their solutions so that overall coherence can be achieved.

- Market

  Control is distributed to the agents that compete for tasks or resources through bidding and contractual mechanisms.

- Scientific community

Solutions to problems are locally constructed, then they are communicated to other problem solvers that can test, challenge, and refine the solution.

3.7 Discussion

In multi agent systems, agents communicate and cooperate with each other to solve problems. There are basically two kinds of control structures in multi-agent systems (Mařík et al., 1999; Sikora and Shaw, 1998): centralized control and decentralized control. In the centralized control structure, there is a central coordinator to whom all agents communicate their solutions. The service provider agents advertise their capabilities to the middle agent, and the middle agent takes the responsibility to dispatch the task to the right agent when it receives a service request. This control structure is based on the advertised agents' capabilities. As in the multi-agent system developed by Pouchard and Walker (2001), different agents in the system are distinguished according to their roles and responsibilities. When all the agents' roles and capabilities are similar or difficult to differentiate, the central control structure won't work. For example, in music fan societies, all music fans have an interest in and knowledge of music, and their interest and knowledge are changing with time – thus making it difficult for a middle agent to keep track of all possible music that would match with the changing interest of each user.

MASs are best suited for use in open systems with a large and dynamic number of agents (Turner and Jennings, 2001). Pouchard and Walker (2001) contend that the central control agent (CA) may create a bottleneck since the CA controls all information exchange for all other agents when the number of users increases. It is believed that the CA system can scale up to 100 users. Therefore, getting the right team of agents and controlling them is of prime interest in the decentralized control structure for a large number of users (Dignum et al., 2001). There is no middle agent in a decentralized control structure. Therefore, agents use an acquaintance list to communicate only with a small subset of agents (Sikora and Shaw, 1998). In the acquaintance model, the individual agent contains information on the current capabilities of peers of the agent. For example, PoliTeam is a groupware support system that makes use of intelligent agent technology and cased-based reasoning technique towards information sharing among team members (Bordetsky and Mark, 2000). In this system, feedback control relationships are captured into a multilayered model of organizational memory and transferred to users by agent-facilitators. The approach is based on a system dynamic approach to organizational learning when the group members constitute a small finite set with similar information needs. The question arises as to how we can extend the functionality of a system such as PoliTeam to share information among a very large number of decision makers who are unaware of each other's existence and/or information needs. Response to this question is the objective of this research.

4. Proposed Methodology of MAS

4.1 Multi-Agent Systems

With the development of agent technology, the need for a system that consists of multiple agents, which communicate in a peer-to-peer fashion, is becoming apparent. Characteristics of MASs are as follows (Sycara, 1998):

- Each agent has incomplete information or capabilities for solving the problem
- There is no system for global control
- Data are decentralized
- Computation is asynchronous

MASs can solve problems that are beyond the individual capabilities of a single agent. The motivations for the increasing interest in MAS research include the ability of MAS to do the following:

- Solve problems that are too large for a centralized agent to solve
- Allow for the interconnection and interoperation of multiple existing legacy systems.
- Provide solutions to problems that can naturally be regarded as a society of autonomous interacting component-agents
- Provide solutions that efficiently use information sources that are spatially distributed (e.g., information gathering from the Internet)

- Provide solutions in situations where expertise is distributed (e.g., health care, stock market)

To communicate effectively, each agent in a multi-agent system needs to know the characteristics of the other agents that can best serve its requirements. A popular model to facilitate communication among agents is by means of a middle-agent (also called "Matchmaker" or "Broker"). All agents register with the middle-agent. For example, let us assume that the agent A has a request (i.e., ASK(X)). To perform this request, the middle agent (F) can use one of the following two procedures.

(1) Recommend performative: A asks F to "recommend" an agent to whom it would be appropriate to send the performative ASK(X). Once F learns that B is willing to accept ASK(X) performatives, it replies to A with the name of agent B. A is then free to initiate a dialog with B to answer this and similar queries. See Figure 4.1 (Finin et al., 1994).

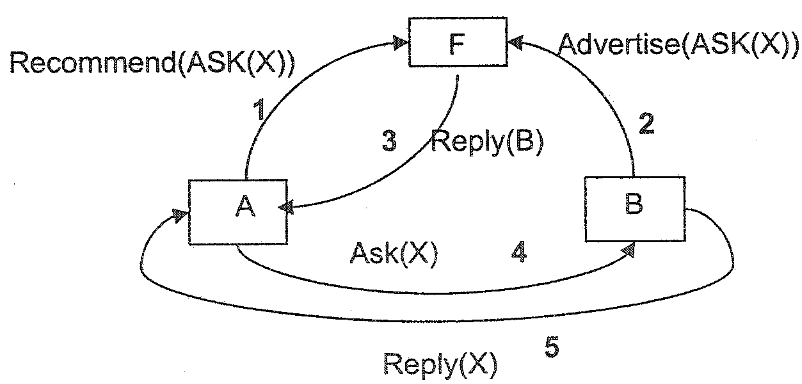


Figure 4.1: The recommend performative is used to ask a facilitator agent to respond with the "name" of another agent which is appropriate for sending a particular performative (Finin et al., 1994)

(2) Broker performative: A asks F to find an agent that can process an ASK(X) performtive. B independently informs F that it is willing to accept performatives matching ASK(X). Once F has both of these messages, it sends B the query, gets a response and forwards it to A. See Figure 4.2 (Finin et al., 1994).

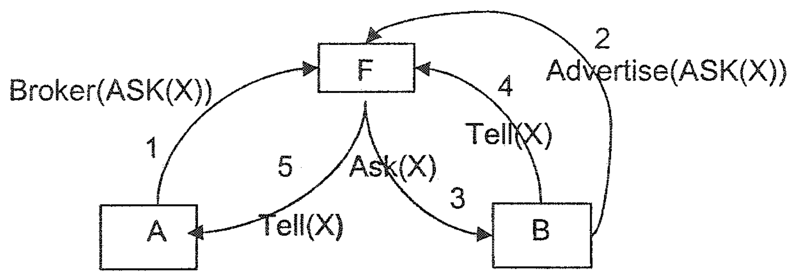


Figure 4.2: The broker performative is used to ask a facilitator agent to find another agent which can process a given performative and to receive and forward the reply (Finin et al., 1994)

The above methodology is applicable when the characteristic of the environment represented by each agent is simple, and the number of agents in need of cooperation is small. This structure becomes difficult to apply effectively in complex decision environments such as stock market portfolio selection.

In the stock portfolio selection decision environment, investors are distributed globally. This community is not controlled centrally, and each investor chooses his/her portfolio independently. Each investor has a local database storing personal stock portfolio that represents his/her personal knowledge and judgment of selected stocks. In addition, investors tend to share their knowledge to improve the quality of their decision processes in selecting their portfolio (e.g., see investor community at ETRADE.COM). Thus, the stock selection decision environment can be regarded as a society of autonomous investors that tend to share knowledge to improve decision-making performance. Let us assume that there are N agents (serving N investors), and that agent 1 (i.e., A1) would like to know if others could recommend a stock similar to IBM (See Figure 2.3). Thus, A1 sends a message to other agents. Other Agents (A2 … An) search their own portfolio (case-base) and select a few similar stocks. These selections are sent back to agent A1. After receiving all the responses, A1 is in a position to assess them and select the one that best matches IBM stock. This is an acceptable process as long as there is no cost involved in sending, receiving, and processing data. Let us assume that A1 would like to send its message to those agents that are most likely to give a good response within the shortest time. We call these agents "buddy agents of A1". The objective of this research is to develop a methodology for identifying the "buddy" agents.

4.2 Assessment of Buddy Agents

Our basic assumption is that a message sent by agent A1 to find stocks similar to IBM is best answered by the agents of those investors whose portfolio (case-base) is "more" similar to the portfolio represented by agent A1. Thus, the objective of this research is to identify agents (buddies) who can best respond to the request of another agent. This objective is based on the assumption that in order to solve a new problem, one should first try using methods similar to those that have worked on similar problems. This is termed as "reinforcement learning" that is based on a set of specific processes (Sutton and Barto, 1998). A reinforcement process is one in which some aspects of the behavior of a system are caused to become more (or less) prominent in the future as a consequence of the application of a "reinforcement operator". In this research, reinforcement operator specifies how the agent should changes its view of its buddy agents. In response to a request from the decision maker D1, agent A1 sends messages to a number of other agents to satisfy the request from D1. Next, the responses of other agents are presented to decision maker D1. The decision maker (Trainer) D1 sends to the agent A1 his/her degree of satisfaction (i.e., positive or negative reinforcement signal) with each of the responses received. As can be noted, the agent (reinforcement operator) does not initiate behavior, but merely selects that which the decision maker likes from which has occurred. The reinforcement model for the agents is presented next (Figure 4.3).

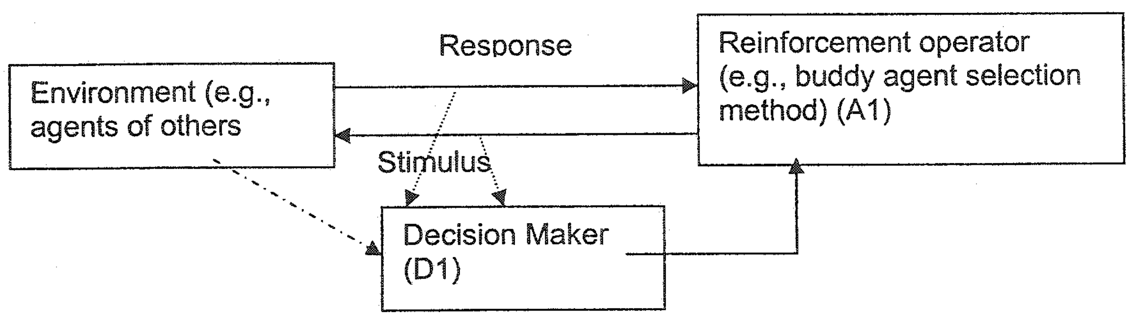


Figure 4.3: Reinforcement Learning Model

Our proposed methodology (i.e., reinforcement operator) to assess the degree of membership of buddy agents is based on fuzzy set modeling. The objective is to select buddy agents that are expected to meet a set of criteria in responding to a request. Let us assume that the two criteria related to stock selection request are as follows.

(1) Response time (T): Amount of time it takes for each agent to respond to a request. Therefore, an agent tends to select buddies that respond quickly to its requests (i.e., minimize T`).

(2) Response quality (Q): The quality of the response (recommendation) received. This is the degree of match between requested stock and those recommendations offered by an agent. We use a range 0-1 where 1 indicates a perfect match and 0 represent no match at all. Thus, the objective is to maximize Q (i.e., select agents as buddy agent with Q close to 1).

We can use a variation of Yager fuzzy intersection to assess the value of goal attainment by each agent as follows:

- Identify the goal attainment (μ) by agent x for each criteria using the following formula:

$$\mu\ (x) = \frac{1}{\left(1+(x-a)^2\right)} \qquad 4.1$$

is used to calculate the membership function of the vicinity **x** to its desired limit ***a***, where x is a possible buddy agent and ***a*** is the value of each criteria. Thus, the goal attainment for T (in which ***a*** is desired to take lower limit of zero) and Q (in which ***a*** is desired to take the upper limit of 1) are derived as follows:

$$\mu\ (T) = \frac{1}{\left(1+T^2\right)} \qquad 4.2$$

$$\mu\ (Q) = \frac{1}{\left(1+(Q-1)^2\right)} \qquad 4.3$$

The goal attainment t, q for all the agents is computed as follows:

$$(G_t(x_i))^{wt} = \{(x_1, \mu(t_1)),(x_2, \mu(t_2)),(x_3, \mu(t_3))\}^{wt} \quad 4.4$$

$$(G_q(x_i))^{wq} = \{(x_1, \mu(q_1)),(x_2, \mu(q_2)),(x_3, \mu(q_3))\}^{wq} \quad 4.5$$

Where *wt* and *wq* are the weights assigned by the decision maker as to the significance of buddy agents' response time and quality of response for buddy membership assignment. For example, a decision maker may assign timeliness of response to be $wt = 2$ and quality of response to be less significant with a value of $wq = 1.2$. The final membership value (D) for each agent is computed by the intersection of all the criteria that they should attain as follows:

$$D = \{[\, x_i, \min_j (\, G_j (x_i)^{wj})] \text{ where } i = 1,\ldots,n; j = t, q\} \quad 4.6$$

For example, if

$$\left(\tilde{G}_t(x_i)\right)^2 = \{(x_1,0.7^2),(x_2,0.5^2),(x_3,0.4^2)\}$$
$$= \{(x_1,0.44),(x_2,0.2),(x_3,0.12)\} \quad 4.7$$

$$\left(\tilde{G}_q(x_i)\right)^{1.2} = \{(x_1,0.3^{1.2}),(x_2,0.8^{1.2}),(x_3,0.6^{1.2})\}$$
$$= \{(x_1,0.24),(x_2,0.76),(x_3,0.54)\} \quad 4.8$$

Then

$$D = \{(x_1, 0.24), (x_2, 0.2), (x_3, 0.12)\} \qquad 4.9$$

This means degree of membership for $x_1 > x_2 > x_3$. Thus, based on the number of possible agents, we can set a limit to the choice of the top n selection to be selected as the buddy agent.

- **Rationale behind this method:**

The goal attainment (μ) in all goals having little importance (Wt , Wq <1) becomes larger, and while those in objectives having more importance (Wt, Wq >1) become smaller. This has the effect of making the membership function of the decision subset D, which is the minimum value of each X over all objectives, being more determined by the important objectives.

4.3 Research Issues and Hypotheses

In order to evaluate the effectiveness of our proposed agent-based buddy finding methodology, we analyze it from two aspects. First, we prove that our methodology is theoretically sound. Second, we use an empirical experiment to test the effectiveness of agent-based buddy finding for human subjects.

Research Issue 1: Is the proposed agent-based buddy finding methodology theoretically sound?

Segmentation is key to marketing (Levin and Zahavi, 2001). Segmentation means to partition the market into groups, or segments, of "like" people, with similar needs and characteristics (Levin and Zahavi, 2001). As a procedure that is appropriate for grouping objects (respondents) into groups (segments) (Gehrt and Shim, 1998), cluster analysis is widely used to identify these like–minded customer segments (Gallagher and Mansour, 2000; Gehrt and Shim, 1998; Lin et al., 1999; March, 1997). The unsupervised classification of patterns (observations, data items, or features vectors) into groups (clusters) has wide applications such as image segmentation, object and character recognition, document retrieval, and data mining (Jain et al., 1999; Kiang and Kumar, 2001). Cluster analysis groups data with similar characteristics and identifies homogenous clusters or groups that are significantly different with regard to several attributes (Hyman and Shingler, 1999; Modha and Spangler, 2002). As a well-established research method (Walsh et al., 2001), cluster analysis is the most useful analytical tool available for aggregating discrete units (e.g., consumers) into groups (i.e., segments) based on their similarities (Iacobucci et al., 2000; Knoke and Kuklinski, 1982).

Cluster analysis is a fundamental technique of unsupervised learning in machine learning and statistics (Duda and Hart, 1973; Hartigan, 1975). In using cluster analysis, the users have all the data in a database to measure the distance between data to form clusters with similar properties (Jain et al., 1999). Therefore, we can use cluster analysis to find clusters of music lovers with similar music interest only if the music interest of the population is known in advance and is available in a database. However, complete availability of data in distributed and dynamic systems such as Internet is not possible.

For example, music lovers use Napster in a distributed environment to share music with each other in a peer-to-peer mode. In this case, each individual music lover does not have access to the centralized data of all other music lovers. Therefore, a music lover cannot use cluster analysis to identify buddies (a cluster of other music lovers) with similar musical taste. Our agent-based methodology is intended to help users find buddies even when there is no access to centralized data for clustering. Nonetheless, to use cluster analysis as a benchmark, we assume availability of a complete data set to test the following hypotheses:

H1: There is no significant difference between the buddies (clusters) identified from the proposed methodology and the buddies derived from cluster analysis.

We also need to test if the proposed methodology identifies buddies acceptable by human subjects. Obviously, this can only be achieved through empirical investigation. In this case, our research objective would be as follows.

Research Issue 2: Is the proposed agent-based buddy-finding methodology useful to human subjects?

The central idea underlying software agents is that of delegation (Hu, 2001; Norman and Reed 2001; Reticular, 1999). In our proposed methodology, subjects delegate their buddy finding task to agents and the agents find buddies for the subjects. The major concern of users is about the quality of agent recommendation results. In the scenario of buddy finding, users expect that agents can find the same buddies as if they searched for buddies by themselves. In our subject experiment, we use the subjects' manually found buddies as the benchmark to evaluate the effectiveness of agent-based methodology. In order to evaluate the effectiveness of the proposed agent-based methodology, our second hypothesis can be formulated regarding the effectiveness of agent delegation as follows.

H2: There is no significant perceived difference between the buddies found through the proposed agent-found methodology and buddies identified by the subjects.

5. Empirical Evaluation Using Stock-Market Portfolio Selection

5.1 Objectives

The objective of this empirical investigation was to test the first hypothesis: there is no significant difference between the buddies (clusters) identified from the proposed methodology and buddies derived from cluster analysis. Section 5.2 describes the decision environment used to test the stated hypothesis. Section 5.3 describes the tool developed for use in our empirical investigation. Section 5.4 presents our experimental design and Section 5.5 presents our experimental result.

5.2 Decision Environment

We have developed a multi-agent system (MAS) to assist the investors to receive (and provide) advice about stocks market securities. The decision environment faced by an investor is highly ill-structured, so much so that security prices are posited to follow the random walk hypothesis, which stated that at any point in time the size and direction of the next price change is random with respect to the state of knowledge available at that point in time (Dyckman et al., 1975). The major cause of this random behavior is caused by the following:

(1) the large number of causal variables;

(2) the fact that variables are highly stochastic; and

(3) the unknown significance of causal relationships among the variables.

Consequently, in such an extremely complex and rapidly changing environment, the forecast of security prices is expected to rely heavily on the analyst's (investor's) cognitive efforts (e.g., intuition, training). Therefore, one would expect that the pertinent decision processes involved should not only vary among different investors, but also should be context dependent (Simon and Hayes, 1976; Tversky and Kahneman, 1982). In addition, among significantly large number of stocks, an investor can only cover a small subset of the stocks. Thus making it highly desirable for the investors to share their knowledge of specific stocks (See ETRADE.COM investor community). This decision environment resembles the society of minds as hypothesized by Minsky (1988), amenable for support by means of MAS because (1) decisions are distributed; (2) each decision maker is autonomous; and (3) decision makers need to share their knowledge to improve their individual decision performance.

5.3 Tool

We have developed a multi-agent system (MAS) using the methodology proposed in this research (see Appendix 1). Our MAS performs the following functions: (1) enables selecting buddy agents; (2) broadcasts the requirements of an investor for new stocks to other investors' agents; (3) facilitates local comparison of stocks for selection of most similar stocks by means of distributed case-based reasoning systems (CBR); and (4) ranks and presents of the stock information received from other agents. The procedures used in the development of the CBR systems was adopted from previous research (Gupta, 1996; Montazemi and Gupta, 1996). AGENTBUILDER software was used in support of communication protocol among agents. An overview of the proposed system is depicted in Figure 2.3 (see Appendix 1 for detailed description).

The following section presents the experimental design for the evaluation of objectives presented in section 5.1.

5.4 Experimental Design

The effectiveness of the methodologies for selecting the degree of membership of the buddy agents was assessed by means of two test scenarios. The first test scenario consisted of eight portfolios with different stocks. This test made it possible to measure our methodology under extreme conditions. However, in reality, different investors can carry the same stock in their portfolio. Therefore, in our second test scenario we assess the sensitivity of proposed methodology when the characteristics of the portfolios in Test 1 are changed. To this end, we made sure that each portfolio would have five stocks common to another portfolio closest to it (i.e., most similar to it). The details of these two tests scenarios are provided next.

5.4.1 Test 1

Test 1 consisted of the following steps:

Step I:

Information about 5000 stocks was collected. Each stock was represented by the 17 financial attributes that are generally used to select stocks (See Table 5.1).

Next, K-Means cluster analysis was used to identify eight groups of stocks with similar characteristics. Stocks with highest loading in each cluster were selected to represent the portfolio of an investor. We were limited to eight portfolios because there were eight microcomputers available to perform this experiment. Each microcomputer represented an investor with an assigned portfolio of stocks in the form of a case-base of a CBR system, and an agent to assist in the selection of stocks. Based on this, we selected eight clusters (portfolios), each with 48 stocks. The stocks in each cluster were entered in the case-base of a CBR system. Thus, we created eight case-bases to represent eight investors.

K-Means cluster analysis attempts to identify relatively homogeneous groups of cases (e.g., stocks) based on selected characteristics (e.g., attributes used to represent a stock), using an algorithm that can handle large numbers of cases. However, the algorithm requires us to specify the number of clusters. Euclidean distances between the final cluster centers provide information about dissimilarities of the clusters: greater distances between clusters correspond to greater dissimilarities (e.g., portfolio of stocks). This can be taken as the closeness of portfolios to each other or the degree of membership of one portfolio with others. Thus, we can use distance between clusters as a good benchmark for assessing the merits of our proposed buddy-agent membership assignment.

We used cluster analysis to identify 8 portfolios as well as the degree of closeness (similarity) among those portfolios using statistical techniques. Here we are using statistical techniques as an acceptable benchmark. Next, we used our methodology to assess the degree of closeness among the portfolios. Kendal Tau was used to compare degree of closeness provided by our methodology with the benchmark (i.e., distance among clusters derived from statistical methods).

Table 5.1: Attributes used to represent selected stocks

Abbreviation	Definition
ASK	Latest ask price.
BID	Latest bid price.
DIVIDEND	Annual dividend payment representing either the latest fiscal year or indicated annual rate based on the most recently announced dividend payments.
EARNINGS	Annual earnings per share representing either the latest fiscal year or indicated annual rate based on the most recent published earnings.
LAST	Last trade price or value.
NET CHANGE	Difference between latest trading price or value, and the historic closing value or settlement price.
OPENING PRICE	Today's opening price or value.
PE RATIO	Ratio of stock price to earnings per share.
PERCENT CHANGE	Percentage change in the latest trade price or value from the historic close.
TODAY'S HIGH	Today's highest transaction value.
TODAY'S LOW	Today's lowest transaction value.
YEAR HIGH	Highest value in the year.
YEAR LOW	Lowest value in the year.
YIELD	For equities, dividend per share expressed as a percentage of the price.
LAST	Last trade price or value.
HISTORIC CLOSE	Most recent non-zero closing value or settlement price.
TRADE VOLUME	Transactional volume of the trade price reported in the LAST field.

We cannot use cluster analysis to assess the agent-membership in a large and dynamic decision environment because cluster analysis requires all the records (data) to be available prior to computation of the clusters. However, in reality this is not possible because of the following conditions:

1. We don't know about all the records at once (i.e., records are distributed throughout the system), and agents (decision makers) join/leave the system at will.
2. The nature of each portfolio (i.e., type of stocks) can change at will.
3. The number of possible agents could be millions -- thus, making the complete portfolio too huge to perform cluster analysis as soon as a change is made to each local portfolio.

Our proposed methodology takes dynamic nature of the decision environment into account. It is based on the following premises:

1. The degree of closeness between two portfolios is based on nature of records (stocks) at any point in time.
2. Assessment of the relationship between two portfolios does not require knowledge about all the portfolios in the system.
3. The degree of closeness among agents is based on pair wise interactions -- thus, making it computationally manageable to assess degree of closeness among agents.

4. We can use as many dimensions (e.g., quality of response time, response quality, cost of response) as required to assess buddy agent memberships.

Step II:

The multi-agents system was used to assess the degree of membership of agents. This was achieved by having each of the 8 agents request recommendation about 48 stocks similar to those in its case-base. Quality of response received from agents was used to compute the degree of membership of remaining agents (i.e., seven buddies) for each agent.

Step III:

Degree of membership of the buddy agents for each agent should be similar to the distance between clusters. Kendal Tau was computed to assess this similarity.

5.4.2 Test 2

The experimental design for this test was the same as Test 1, except that we added five stocks from each portfolio to another one that was most close to it. For example, if the stocks of portfolio 1 were close to the stocks of portfolio 8, we added 5 of the stocks from portfolio 8 to portfolio 1. Our assumption is that the same stocks can be shared between two portfolios that are most similar to each other.

5.5 Results and Analyses

5.5.1 Analysis of the Buddy-Agent Membership for the Test Scenario 1

The objective of testing the MAS was to determine the effectiveness of the proposed procedures to assess the degree of membership of the buddy agents. To this end, 5000 stocks were selected at random from NASDAQ and financial data for each stock was collected. Cluster analysis, based on 17 financial attributes, identified 8 clusters of stocks to exist among the 5000 selected stocks. The data for each cluster was saved in a case-base of a CBR system to represent the portfolio of an investor. Next, 48 requests were generated from each case-base. The MAS was responsible for sending the requests to other agents and returning the responses to the requesting agents. Here, we used quality of response as our only criteria for assessing the degree of membership of the agents. The reason for ignoring other attributes such as response time is that our basis of assessing the goodness of computed membership was its comparison with the distance between clusters (see Table 5.2). The rank-order of the portfolios (clusters) in relation to distance to each other is presented in Table 5.3. Table 5.4 presents the degree of membership of agents based on 48 requests sent to other agents and pertinent feedback received. The content of Table 5.3 and 5.4 are depicted in Table 5.5 for ease of comparing the two sets of rankings. Our contention is that the goodness of our proposed buddy-agent membership model can be assessed by comparing it with the distance among clusters.

The correlation between membership ranking of two methods (distance between clusters and our proposed method) shows that the two types of assessments are correlated (Kendall's tau = 0.313 ($p < 0.01$)) as shown in Table 5.6. This confirmed our first hypothesis.

H1: There is no significant difference between the buddies (clusters) identified from the proposed methodology and the buddies derived from cluster analysis.

Table 5.2: Distance between the 8 portfolios (clusters)

	Portfolio 1	Portfolio 2	Portfolio 3	Portfolio 4	Portfolio 5	Portfolio 6	Portfolio 7	Portfolio 8
Portfolio 1		1.85	3.39	2.87	3.18	1.83	3.07	3.74
Portfolio 2	1.85		4.39	3.9	3.77	2.01	3.62	4.14
Portfolio 3	3.39	4.39		2.27	2.26	3.21	3.09	4.68
Portfolio 4	2.87	3.9	2.27		2.09	2.4	1.82	2.82
Portfolio 5	3.18	3.77	2.26	2.09		1.95	1.36	3.34
Portfolio 6	1.83	2.01	3.21	2.4	1.95		1.65	2.72
Portfolio 7	3.07	3.62	3.09	1.82	1.36	1.65		1.99
Portfolio 8	3.74	4.14	4.68	2.82	3.34	2.72	1.99	

Table 5.3: Rank order of the distance between clusters

	Portfolio1	Portfolio 2	Portfolio 3	Portfolio 4	Portfolio 5	Portfolio 6	Portfolio 7	Portfolio 8
Portfolio 1		1	5	6	5	2	5	5
Portfolio 2	2		6	7	7	4	7	6
Portfolio 3	6	7		3	4	7	6	7
Portfolio 4	3	5	2		3	5	3	3
Portfolio 5	5	4	1	2		3	1	4
Portfolio 6	1	2	4	4	2		2	2
Portfolio 7	4	3	3	1	1	1		1
Portfolio 8	7	6	7	5	6	6	4	

Table 5.4: Rank order of the membership of the buddy agents

	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Agent 6	Agent 7	Agent 8
Agent 1		5	2	3	5	5	5	5
Agent 2	1		7	7	4	2	4	4
Agent 3	7	7		1	7	7	7	7
Agent 4	6	6	1		6	6	6	6
Agent 5	5	4	4	4		3	1	2
Agent 6	4	1	6	6	3		3	3
Agent 7	2	3	5	5	1	1		1
Agent 8	3	2	3	2	2	4	2	

Table 5.5: Ranking comparison between cluster analysis and buddy membership

Agent \ Portfolio	1	2	3	4	5	6	7	8
1		1 / 5	5 / 2	6 / 3	5 / 5	2 / 5	5 / 5	5 / 5
2	2 / 1		6 / 7	7 / 7	7 / 4	4 / 2	7 / 4	6 / 4
3	6 / 7	7 / 7		3 / 1	4 / 7	7 / 7	6 / 7	7 / 7
4	3 / 6	5 / 6	2 / 1		3 / 6	5 / 6	3 / 6	3 / 6
5	5 / 5	4 / 4	1 / 4	2 / 4		3 / 3	1 / 1	4 / 2
6	1 / 4	2 / 1	4 / 6	4 / 6	2 / 3		2 / 3	2 / 3
7	4 / 3	3 / 3	3 / 5	1 / 5	1 / 1	1 / 1		1 / 1
8	7 / 2	6 / 2	7 / 3	5 / 2	6 / 2	6 / 4	4 / 2	

Table 5.6: Correlation between the Agent rank order of buddy agents and the distance between portfolios (clusters)

	Rank Correlation Cofficient (Kendall's tau_b)
Agent Rank-Cluster Rank	0.313

*positively correlated at the 0.01 level (2-tailed).

5.5.2 Analysis of the Buddy-Agent Membership for the Test Scenario 2

The second test scenario was conducted in the same manner as the first one, except for adding 5 stocks from one portfolio to the portfolio closest to it. Therefore, we had 53 stocks in each portfolio after adding 5 stocks to the original 48 stock-portfolio. We used the distance among clusters derived in test scenario 1 as a proxy for the closeness of the portfolios. We recomputed the distance matrix among these new sets of portfolios (See Table 5.7). The rank-order of the new portfolios in relation to distance to each other is presented in Table 5.8.

Data about the characteristics of stocks for each new portfolio was also saved in a case-base of a CBR system to represent the portfolio of an investor. Next, 53 requests were generated from each case-base. Our MAS was responsible for sending the requests to other agents and returning the responses to the requesting agents. Table 5.8 presents the degree of membership of agents based on 53 requests sent to other agents and feedback received.

The correlation between membership ranking of two methods (distance between new portfolios and our proposed method) shows that the two types of assessments are correlated (Kendall's tau = 0.601 ($p < 0.01$)), as shown in Table 5.10. This indicates that our proposed model for assessing membership of the buddy agents works well when there are overlaps among portfolios of investors. Again, this confirmed our first hypothesis when there are overlaps among portfolios of investors.

H1: There is no significant difference between the buddies (clusters) identified from the proposed methodology and the buddies derived from cluster analysis.

It is noteworthy that the correlation among membership ranking of two methods increased from 0.313 (when there was no overlap of stocks between the portfolios in Test 1) to 0.601 when there were overlaps among portfolios. This is expected because, as we increase the overlap among the portfolios, the clusters become more similar so much so that at the extreme when all portfolios have exactly the same set of stocks then the correlation among the two methods should be 1.00.

Table 5.7: Distance between the 8 portfolios (clusters)

	Portfolio 1	Portfolio 2	Portfolio 3	Portfolio 4	Portfolio 5	Portfolio 6	Portfolio 7	Portfolio 8
Portfolio 1		2.733036	6.551111	4.819408	3.382902	3.763938	3.309554	3.969797
Portfolio 2	2.733036		8.798087	7.066524	3.639997	3.580886	3.421188	3.852538
Portfolio 3	6.551111	8.798087		3.213969	7.108595	8.337911	7.673482	8.64416
Portfolio 4	4.819408	7.066524	3.213969		5.387848	6.299526	5.574286	6.114861
Portfolio 5	3.382902	3.639997	7.108595	5.387848		1.838053	1.208193	3.1324
Portfolio 6	3.763938	3.580886	8.337911	6.299526	1.838053		1.009194	2.290964
Portfolio 7	3.309554	3.421188	7.673482	5.574286	1.208193	1.009194		2.045471
Portfolio 8	3.969797	3.852538	8.64416	6.114861	3.1324	2.290964	2.045471	

Table 5.8: Rank order of the distance between clusters

	Portfolio 1	Portfolio 2	Portfolio 3	Portfolio 4	Portfolio 5	Portfolio 6	Portfolio 7	Portfolio 8
Portfolio 1		1	2	2	4	5	4	5
Portfolio 2	1		7	7	5	4	5	4
Portfolio 3	7	7		1	7	7	7	7
Portfolio 4	6	6	1		6	6	6	6
Portfolio 5	3	4	3	3		2	2	3
Portfolio 6	4	3	5	6	2		1	2
Portfolio 7	2	2	4	4	1	1		1
Portfolio 8	5	5	6	5	3	3	3	

Table 5.9: Rank order of the membership of the buddy agents

	Agent1	Agent2	Agent3	Agent4	Agent5	Agent6	Agent7	Agent8
Agent1		1	2	3	5	1	5	5
Agent2	1		5	5	6	3	6	6
Agent3	7	7		1	7	7	7	7
Agent4	6	6	1		3	6	4	4
Agent5	5	5	4	4		4	1	2
Agent6	4	3	7	7	4		3	3
Agent7	2	4	6	6	1	2		1
Agent8	3	2	3	2	2	5	2	

Table 5.10: Correlation between the Agent rank order of buddy agents and the distance between portfolios

	Rank Correlation Cofficient (Kendall's tau_b)
Agent Rank-Cluster Rank	0.601

*positively correlated at the 0.01 level (2-tailed).

6. Empirical Evaluation Using Music Selection

6.1 Objective

The objective of this empirical investigation was to test the second hypothesis: There is no significant difference between the agent-found buddies (based on our proposed methodology) and buddies identified by the subjects. The decision environment used to test this hypothesis was the selection of buddies who can recommend music titles based on one's music interest.

The remainder of this chapter is structured as follows: section 6.2 describes the decision environment. Section 6.3 describes the experimental tools. Section 6.4 describes the subjects used in the experiment. Section 6.5 describes data collection. Section 6.6 describes experimental procedures. Section 6.7 describes the consistency check of user-entered preferences. Section 6.8 describes the statistical model. Section 6.9 describes the results and analyses.

6.2 Decision Environment

We developed a multi-agent system (MAS) to assist music lovers to find buddies in support of our experimental test. Justification for selecting this decision environment to test the stated hypothesis is as follows. First, music lovers are available everywhere, and listening to and evaluating songs or a piece of music does not require significant training. Therefore, it is easier to recruit subjects for our experiment. Second, there are many well-grounded research findings in the retrieval and classification of music (e.g., www.moodlogic.com; www.musclefish.com; Chai and Vercoe, 2000) that we utilized towards an empirical test of our second hypothesis. Third, when human listeners are confronted with musical sounds, they can rapidly and automatically orient themselves with the music within seconds (Scheirer, 2000). Even musically untrained listeners have an exceptional ability to make rapid judgments about music from very short examples, such as determining the music's style, performer, beat, complexity, and emotional impact (Scheirer, 2000). Therefore, we expected a reliable outcome from having common people evaluate a song within a short period of time (e.g., MoodLogic users take 30 seconds to listen to and choose a piece of music). A short time limit is very important in assuring the reliability of these experiment results, since a long duration to complete the experiment can cause subjects to become tired and impatient.

6.2.1 Buddy Finding in a Music Domain

There are many music stations on the Internet, and it is very popular to listen and download music from these stations (e.g.: http://www.mp3.com/). Within those music stations, music lovers not only listen to music, but also communicate with other music lovers through message boards. Through their communication, music lovers look for people with similar interests (i.e., buddies) to share music and related information. Finding buddies through message boards is already a popular practice among music lovers. There are hundreds of music lovers on one message board, and it is a very time consuming and sometimes frustrating process for people to find buddies manually. As a platform for music lovers to communicate with each other and find buddies manually, existing message boards do not solve the problem of information overload.

As we already described in the previous experiment, this decision environment resembles the society of minds as hypothesized by Minsky (1995), amenable for support by means of MAS because (1) decisions are distributed; (2) each decision maker is autonomous; and (3) decision makers need to share their knowledge to improve their individual decision performance. Our proposed agent-based buddy finding methodology provides a convenient decision support tool for users to find buddies in this virtual environment.

6.2.2 Music Features Classification

Traditionally, one important reason to classify music is for music retrieval. Category-based browsing and/or text-based searching is the most traditional and simple way of music retrieval. To find a piece of music, the user needs to provide information such as the music title, composer, and artist, to enable the search engine to search the database; otherwise, the user needs to browse the whole category (Chai and Vercoe, 2000). Audio and multimedia applications would benefit listeners significantly if they could interpret the content of the audio with a feature-based filtering system (Wold et al., 1996). For a content-based search or feature-based filtering system, one important problem is to describe the music by its parameters or features, to enable search engines or information filtering agents to measure the similarity of the target (user's query or content preference) and the candidate music (Chai and Vercoe, 2000). The classifications of music features are varied. Digital-signal processing is one major approach used to classify music. As in MUSCLE FISH (http://www.musclefish.com.), the music can be analyzed based on the following aspects of sound: loudness, pitch, brightness, bandwidth, and harmonicity (Wold et al., 1996):

- Loudness is measured by the signal's root-mean-square (RMS) level in decibels, which is calculated by taking a series of windowed frames of the sound and computing the square root of the sum of the squares of the windowed sample values.
- Pitch is estimated by taking a series of short-time Fourier spectra. For each of these frames, the frequencies and amplitudes of the peaks are measured and an approximate greatest common divisor algorithm is used to calculate an estimate of the pitch.
- Brightness is computed as the centroid of the short-time Fourier magnitude spectra, again stored as a log frequency. It is a measure of the higher frequency content of the signal.
- Bandwidth is computed as the magnitude-weighted average of the differences between the spectral components and the centroid.
- Harmoncity distinguishes between harmonic spectra (such as vowels and most musical sounds), inharmonic spectra (such as metallic sounds), and noise (spectra that vary randomly in frequency and time). It is computed by measuring the deviation of the sound's line spectrum from a perfectly harmonic spectrum (Wold et al., 1996).

Digital signal processing approach is from the technical perspective. According to Wold et al. (1996), the major application areas of this approach are for professional music work, such as audio databases and file systems; audio database browsers; audio editors; surveillance and automatic segmentation of audio and video. We did not use these features of classification in the agent application, because generally music lovers have little knowledge of these measurements. It is very difficult to build computer systems that can understand the things that human listeners understand immediately and unconsciously when they hear music, such as the musical genre, or the instruments that are being played, or the beat in the music (Scheirer, 2000).

Music is a way for people to communicate emotion and feeling, and in fact, this is often viewed as the primary function of music (Scheirer, 2000). Music is not the simple combination of various sounds or digital signals. Chai and Vercoe (2000) classified music features as quantitative and qualitative:

- Quantitative features: background information (composer, artists, age, genre, etc.), time signature, key signature, tonality, tempo, instruments, music structure, pitch, loudness, melody, rhythm, and all the other parameters that could be derived from the music, etc.
- Qualitative features: rating, functionality, emotional features, etc.

MoodLogic (http://www.moodlogic.com), an Internet music application station, considers both quantitative features and qualitative features of music. MoodLogic believes that the most reliable way to know how consumers perceive each song in the music universe is to ask them systematically and repeatedly. Since March 2000, MoodLogic has attracted more than 40,000 music fans of all kinds to listen to songs and evaluate them carefully at a public website called Jaboom (www.jaboom.com). MoodLogic has gathered more than 700 million data points about music by inviting music lovers to fill out an extensive demographic/psychographic survey when they register at Jaboom. To date, MoodLogic has gathered metadata concerning consumer perceptions of over 500,000 song titles in the most popular genres, across all relevant decades since about 1950. MetaDB is the MoodLogic database of music metadata. Dr. Robert Gjerdingen, the world's leading expert on music perception, manages this database of music metadata. MoodLogic extracted a set of key information describing each song:

- Song ID tag, Song, Album and Artist names.
- Genre, Mood, Decade, Tempo, Beat, Popularity, Vocal Style, Lead Vocal Style. See Table 6.1.

Table 6.1: Music Features Classification from MoodLogic

Genre	Mood	Decade	Tempo	Beat	Popularity	Vocal Style	Lead Style
Rock	Upbeat	1960s	Very Slow	Light	Top Picks	Smooth	Male
R&B/Soul	Happy	1970s	Slow	Medium	Popular	Neutral	Female
Country	Romantic	1980s	Medium	Heavy	Well known	Raspy	Mixed
Electronica	Mellow	1990s	Fast		Split Decision		Instrumental
Rap/Hip-Hop	Sentimental	Current	Very Fast		Niche		
Jazz	Sad						
New Age	Brooding						
Alternative	Aggressive						
Easy Listening							
Reggae							
Folk							
Blues							
Gospel							
Latin							
World							

Finding buddies in a music domain is an emotional matching process. It is not just a simple digital signal processing. In the research reported here, music buddies is the term used to denote people with similar music interests and music attributes preferences. We also consider objective features, like tempo and beat, as well as qualitative features like mood and popularity, to be important attributes of music in finding music buddies. The music attributes data from MoodLogic are suitable to use in the agent-based buddy finding methodology.

6.2.3 Music Similarity Measures

SaxEx is a Case Based Reasoning System (CBR) system capable of generating expressive performances of melodies based on examples of human performances (Mántaras and Arcos, 2001). The purpose of SaxEx is to endow the automatically generated music with the impressiveness that characterizes human performance. In this research, we also use CBR methodologies to represent music and to select similar music. The attributes for our musical CBR system are depicted in Table 6.1.

For example, our CBR system would be able to respond to a question such as, What is the similarity between "Here" authored by The Beatles, and "The Story in Your Eyes," authored by The Moody Blues? We measure music similarity by calculating the difference between users' preferences for different songs. The values of preference descriptors are measured by Likert-type scales (1-9). Next, we explain the CBR matching methodology and its application in a music domain.

In case based reasoning, the Nearest-Neighbor matching function has been widely used in existing systems (Gupta, 1996). The overall similarity (OS^{NN}) of the new case (song) "n" and the k^{th} previous case (song) p_k, using the NN matching function, is as follows:

$$OS^{nn}(n, p_k) = \frac{\sum_{i=1}^{m} w_i sim(a_i^n, a_i^{p_k})}{\sum_{i=1}^{m} w_i} \qquad 6.1$$

where $sim(a_i^n, a_i^p)$ is the similarity of the new song to the previous song along a descriptor (i.e., attribute) pair, a_i^n is the i_{th} descriptor of the new song, a_i^p is the i_{th} descriptor of the previous song, and w_i is the importance of the i_{th} descriptor; the superscripts n and p refer to the new song and the previous song respectively, and *sim*() is a function, rule, or heuristic that determines the pair-wise similarity along a descriptor.

The nearest-neighbor (NN) matching function has been adopted from the pattern matching literature (Gupta, 1996). In pattern matching, all previous cases are represented by the same set of descriptors and their importance is determined by means of an inductive matching learning technique which minimizes classification error (Duda and Hart, 1973). The NN matching function assesses overall similarity by a weighted linear combination of similarities along descriptors. The weights represent the degree of importance of the descriptor towards the goal of the decision problem.

In using agent-based buddy finding methodology, we need to judge the similarity between songs based on the degree of preference for each music attribute (descriptor) from the user's input. This enables us to measure the difference between the user's preferences for two different songs. The "preference" for each music attribute is measured on a nine-point Likert-type scale, from extremely little like to extremely like. All attributes have the same acceptable range R_i. Let A^n be the set of attributes in the problem-schema of the new case such that:

$$A^n = \{a_i^n\} \quad i = 1 \ldots m \qquad 6.2$$

Where a_i is the i^{th} attribute of the new music, and x_i^n is its score on the scale. Let there be k previous music samples that are candidates for matching, and let A^{p_k} be the set of attributes for the problem-schema of the k^{th} candidate previous song:

$A^{p_k} = \{a_i^{p_k}\}$ where $i = 1....m$, and $k = 1...l$ (candidate previous music) 6.3

Where $a_i^{p_k}$ is the i^{th} attribute of the k^{th} candidate previous song, and $x_i^{p_k}$ is its score. The degree of closeness (c_i^k) along the i^{th} attribute for k^{th} candidate previous song is determined as follows:

$$c_i^k = 1 - \frac{\left|x_i^n - x_i^{p_k}\right|}{R_i} \quad \forall \quad i \text{ and } k \qquad 6.4$$

Where R_i is the range of the scoring scale of i^{th} attribute; here it is 8. c_i^k is the similarity between two songs along the i^{th} attribute(i.e.; $sim()$). Next, we use an example to demonstrate the process of calculating the overall similarity between two songs.

Let us assume that the subject entered the importance of each music attribute in evaluation as follows (Table 6.2):

Table 6.2: User Input Importance of Music Attribute

Genre	Mood	Decade	Tempo	Beat	Popularity	Vocal Style	Lead Vocal
8	5	2	7	3	6	9	1

The sum of all importance values is 41. Therefore, we use 41 to divide every importance value to compute the weight of each attribute as follows (Table 6.3):

Table 6.3: Weights of Music Attributes

Genre	Mood	Decade	Tempo	Beat	Popularity	Vocal Style	Lead Vocal
0.20	0.12	0.05	0.17	0.07	0.15	0.22	0.02

Next, the subject assesses his/her preference for different values of each of the music attributes. For example, the subject's preference for different types of Genres is depicted in Table 6.4. The weight of Genre is 0.2 from Table 6.3.

Table 6.4: Preference for Genre

Rock	R&B/Soul	Country	Electronic	Rap	Jazz	New Age	Alternative	Easy	Reggae	Folk	Blues	Gospel	Latin	World
3	7	9	4	6	2	6	3	6	4	4	4	7	3	3

We now use two different songs as an example to illustrate the calculation of the similarity between them based on the subject's input.

Song 1: Let Me Let Go　**Author:** Faith Hill

Genre: Country　**Mood:** Romantic　**Decade:** Current

Tempo: Slow　**Beat:** Light　**Popularity:** Popular

Vocal Style: Neutral　**Lead Vocal**: Female

Song 2: Free Bird　**Author:** Lynyrd Skynyrd

Genre: Rock　**Mood:** Mellow　**Decade:** 70's

Tempo: Medium　**Beat:** Medium　**Popularity:** Well Known

Vocal Style: Raspy　**Lead Vocal:** Male

Since the Genre of "Let Me Let Go" is Country Music, we can find from Table 6.4 that the preference for Country is 9. The Genre of "Free Bird" is Rock, and we can find from Table 6.4 that the preference for Rock is 3.

The preference similarity along attribute Genre between these two songs is:

$$c_1 = 1 - \frac{|9-3|}{8} = 1 - \frac{6}{8} = 0.25 \qquad 6.5$$

Next, let us look at the their preference similarity in Mood. The preference for each kind of mood is depicted in Table 6.5:

Table 6.5: Preference of Mood

Upbeat	Happy	Romantic	Mellow	Sentimental	Sad	Brooding	Aggressive
9	7	6	2	2	1	3	5

The mood for "Let Me Let Go" is Romantic, and we can find from Table 6.5 that the preference for Romantic is 6. The mood for "Free Bird" is Mellow, and we can find from Table 6.5 that the preference for Mellow is 2.

Therefore, we can compute preference similarity along attribute Mood between these two songs to be as follows:

$$c_2 = 1 - \frac{|6-2|}{8} = 1 - \frac{4}{8} = 0.5 \qquad 6.6$$

The weight of Genre is 0.2 and the weight of Mood is 0.12 (see Table 6.3). The weighted sum of these two preference similarities is computed thus:

$$0.2 * c1 + 0.12 * c2 = 0.2 * 0.25 + 0.12 * 0.5 = 0.11 \qquad 6.7$$

As preference similarities of these two songs along other attributes (i.e., Decade, Tempo, Beat, Popularity, Vocal Style, and Lead Style) are computed, the overall preference similarity between these two songs is determined by aggregating all these weighted preference similarities.

6.3 Tools

In order to evaluate the effectiveness of agent-based buddy finding methodology, we developed a web-based system to allow the user to perform the following four major tasks: (1) enter music attribute preferences; (2) select favorite music from a music station and create a music collection; (3) communicate with other subjects and manually find buddies; (4) evaluate the goodness of subject-found buddies and agent-found buddies.

This test system consists of several major components -- (I) Music browser; (II) Message board; and (III) Agent-based buddy finding system -- as follows:

- Music browser

We used a commercial music browser from MoodLogic (www.moodlogic.com). MoodLogic, Inc. is a provider of software and metadata for digital music media. The HTML-based music browser (http://browser.moodlogic.com/B/So/667/) enabled subjects to search for music based on their music taste and preference; browsing choices included the following attributes: (i) genre, (ii) decade, (iii) mood, (iv) tempo, (v) beat strength, (vi) vocal arrangement, (vii) vocal style, and (viii) popularity. The subjects selected the attribute value from the dropdown list and were able to listen to different music to understand various attribute values. It is possible that a subject might have had little knowledge of some of the above eight attributes. Thus, to capture this, we asked the subjects to rate the importance of each of the attributes in selecting music (see section 6.6).

- Message board

We provided a message board by ezboard.com for subjects to communicate with each other. Ezboard.com has an important "Search" function that enables a user to find another specific user.

- Agent-based buddy finding system

We used the same agent-based buddy finding methodology as we used in our first empirical evaluation (i.e., stock portfolio selection (see Chapter 4)).

To test the second stated hypothesis, the performance of agent-based buddy finding methodology was compared with the subject's choice of buddies. There are many ways for users to communicate with each other and to find their buddies on the Internet, like email, instant messaging and chat rooms, newsgroups and message boards. With email, the user can contact some users for whom he/she already has addresses. Apparently this method is limited to only a small number of people. With instant messaging and chat rooms, users can contact only those users who login at the same time. A very popular peer-to-peer means of sharing music on the Internet is through a message board (e.g., www.mp3.com). Message board systems support one-to-many asynchronous communication. Compared with other communication approaches, the advantages of message boards are apparent:

- users can post their requests on the message board and check them whenever they have time (asynchronous communication)
- requests can be seen by all others who login to this message after posting (one-to-many)
- users can search others' postings and reply to their postings at a different time (asynchronous communication)
- users can search others' replies to their postings

Message boards such as http://www.mp3.com/ provide music lovers a place to exchange information about music of their interests. We also developed a message board for the subjects to share music information. The subjects could post a message and reply to a message. From those who offered the best recommendations to requests, the subjects chose their top five buddies.

There are many ways to create the profile of user preference in an agent-based system. One can simply ask users to manually enter their preferences for various music attributes. This is a very common and effective method used in developing intelligent agent filtering (Good et al., 1999). Details of procedures that we followed in collecting the pertinent data are presented in section 6.6.

6.4 Subjects

The subjects were recruited voluntarily from three undergraduate and graduate classes at the University of Illinois. All subjects were MIS majors or MIS minor students. Each subject was given five extra bonus points added to his/her final grade for completing the tests. Thirty eight students out of 46 participated in the experiment. Four students did not complete all the required steps of the experiment. Therefore, the total number of subjects for final evaluation is 34. The subjects were asked to answer three questions about their music-related habits as follows: (1) Time spent listening to music; (2) Money spent on purchasing music CDs per year; and (3) Time spent downloading music from the Internet. From the subjects' input, we see that the average time for music listening is 2.6 hours/day. Average money spent on purchasing music CDs is $4.73. The average number of times downloading music from the Internet is 2.2 times per year. Figures 6.1, 6.2 and 6.3 show variation of these attributes among the subjects. This indicates that subjects had experience in listening to music, and downloading music from the Internet. Thus, they all had a varying interest in music.

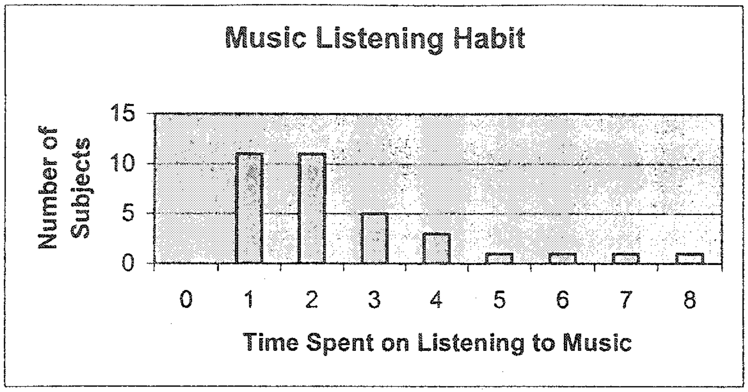


Figure 6.1: Hours spent listening to music/day

0: None; 1: 1 hours; 2: 2 hours; 3: 3 hours; 4: 4 hours; 5: 5hours; 6: 6 hours;

7: 7 hours; 8: 8 hours

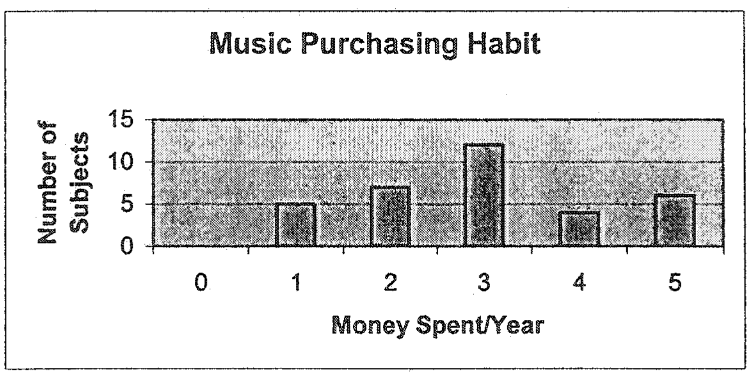


Figure 6.2: Money spent on music / Year

0: none; 1: $10- $20; 3: $30- $40; 5: $50- $60; 7: $70- $ 80; 9: >$80

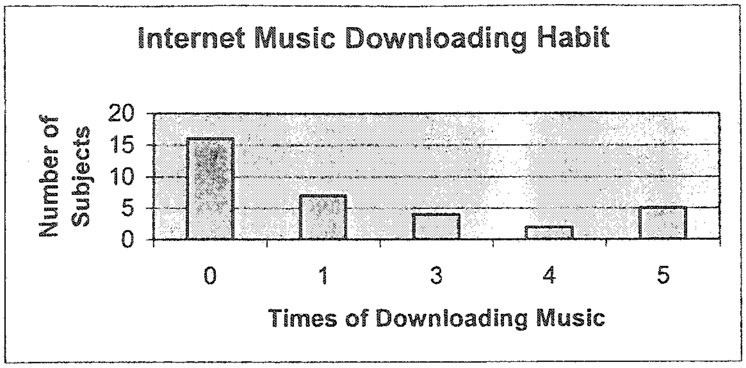


Figure 6.3: Times downloading music from the Internet

0: None; 1: 10 -20 times; 2: 30- 40 times; 3: 50 - 60 times; 4: 70 times-80 times and above

6.5 Data Collection

The data were collected electronically and stored in a database. The data we collected were the following:

- Subjects' evaluation of subject-found buddies
- Subjects' evaluation of mixed buddies from subject-found buddies and agent found buddies
- Subjects' evaluation of songs contained in three items: (1) subject-found buddies; (2) agent- found buddies; and (3) final subject-found buddies.
- Subjects' music related data consisting of (1) Time to listen to music; (2) Money spent on purchasing music CDs; and (3) Times downloading music from the Internet
- Subjects' comments on the evaluation process

The procedures followed to collect these data are detailed next.

6.6 Experimental Procedures

Following are the procedures used by subjects in the experiment.

1. Subjects entered their level of preference for music attributes: Subjects were asked to use the weighted score method to assess the factors that determine their music preference based on the music attributes presented on moodlogic.com using a Likert-type scale of 1-9 (from 1: extremely little important to 9: extremely important) to assess the importance of each category (i.e., Genre, Mood, Decade, Tempo, Beat, Popularity, Lead vocals, Vocal style). The relative importance was calculated by dividing each category by the sum of the scores. Within each category, subjects gave favorite scores of 1-9 (from 1: extremely little like to 9: extremely like) to each item (e.g., items such as Smooth, Neutral, and Raspy in the Category of Vocal Style), shown in Figure 6.4.

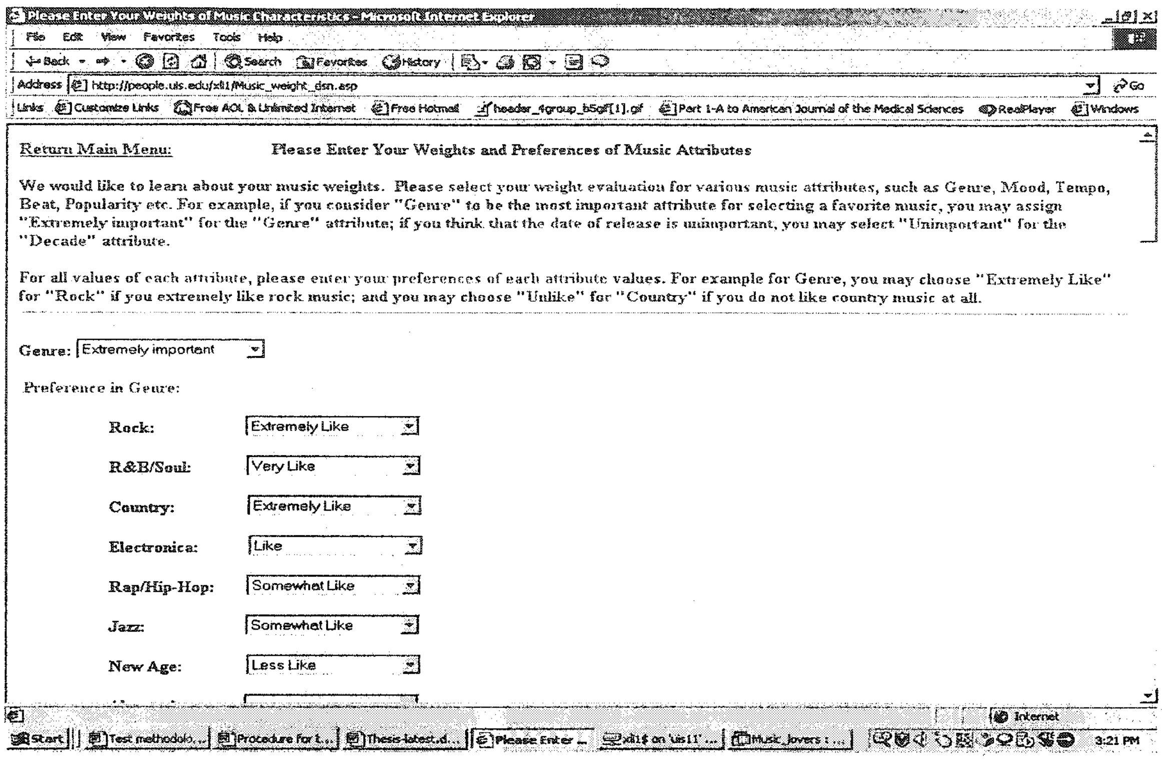


Figure 6.4: Subjects used this form to enter their level of preference for music attribute

2. Subjects created their music collections by selecting their favorite music from the music station: Subjects were asked to identify their 20 most-favored music titles from moodlogic.com. Selecting 20 songs was a suitable task for the subject to complete in about one hour. Moodlogic (www.moodlogic.com) is a music station and it stores the values of music attributes, shown in Figure 6.5. For each music title, the subject not only needed to enter the music name and artist name, but also needed to enter a value for each attribute. Subjects saved the characteristics of their selected music in a database, shown in Figure 6.6.

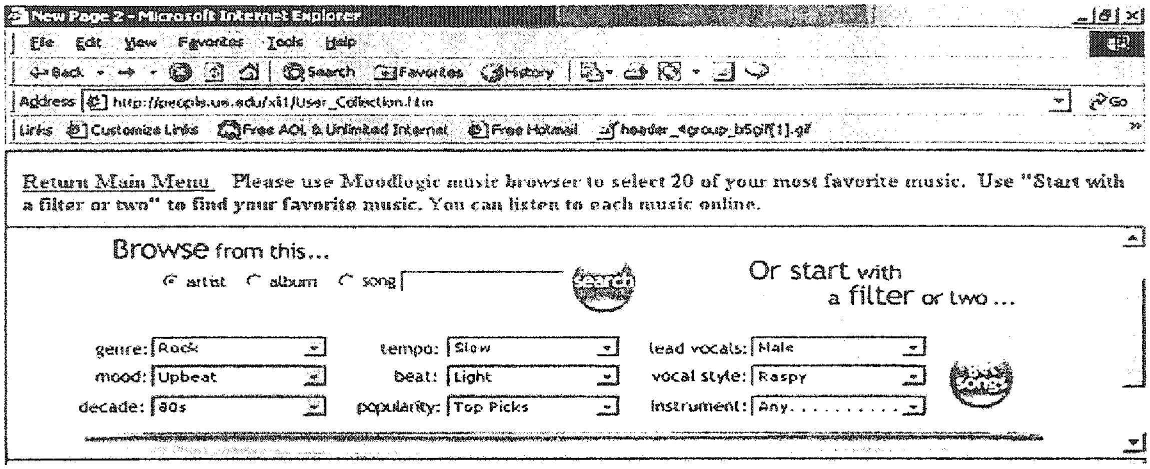


Figure 6.5: Subjects used this form to select their favorite music from www.moodlogic.com

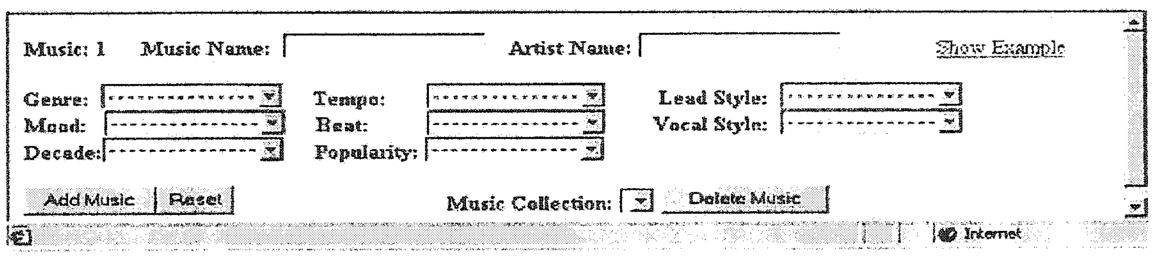


Figure 6.6: Subjects used this form to save the characteristics of their selected music in a database

3. First, subjects posted their favorite music from their music collection on the message board and found their buddies. Next, each subject was asked to announce the music titles from his/her collection of 20 music titles on the message board and to ask for music-titles similar to them. Each subject had to provide a recommendation of at least 10 requests from other people on the message board. Recommendations should have been based

on his/her list of 20 music titles selected in step 2. Based on the recommendation received, each subject selected five subjects as his/her buddies with closest music interests to his/her own. The subject could also add some comments to each selected buddy, indicating his/her perception about the usefulness of the music suggestions from that buddy as shown in Figure 6.7. We call this "the manual process of buddy searching."

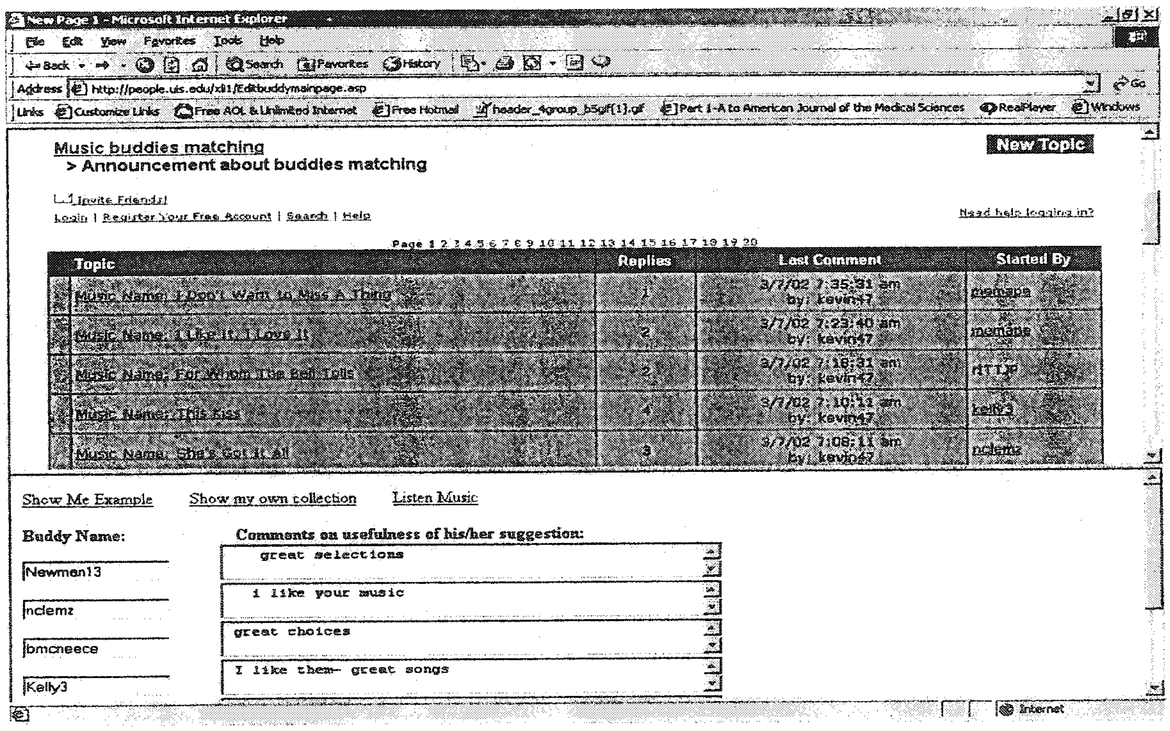


Figure 6.7: Subjects used this form to assess the goodness of their buddies

4. Subjects evaluated their buddies: Each subject was asked to rank-order the top-five buddies based on the usefulness of their recommended music (see

Figure 6.8). With this result, we compared this rank with the rank from the proposed agent-based methodology to see if these two ranks are correlated.

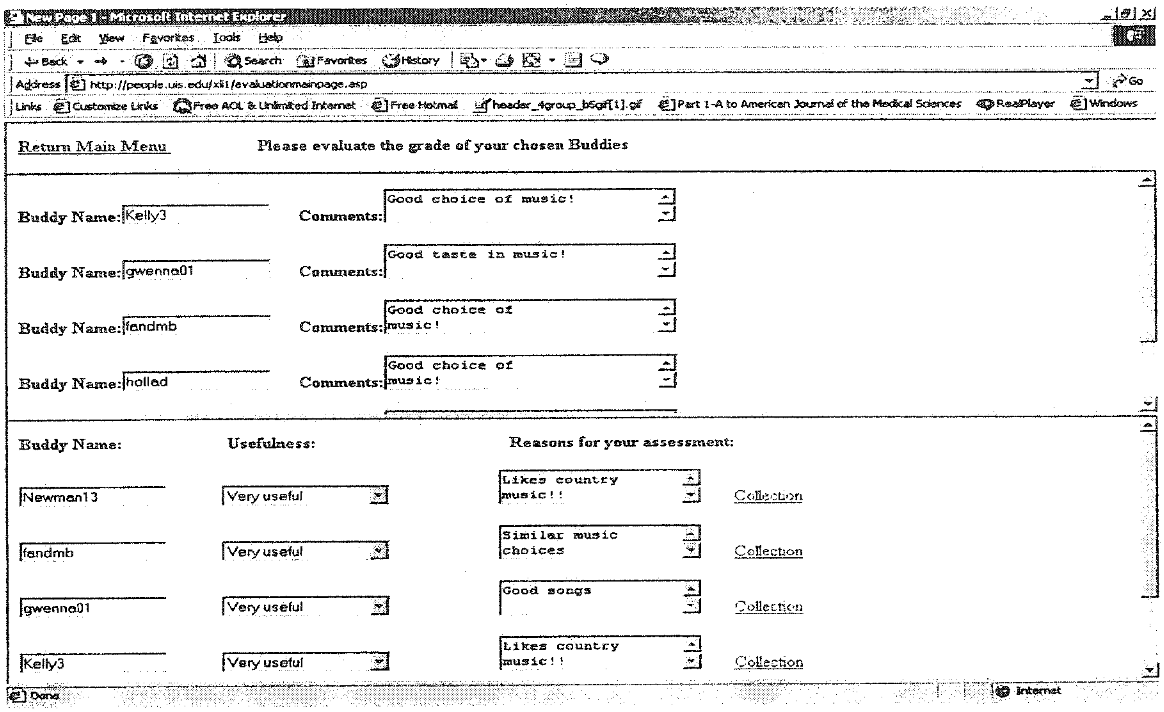


Figure 6.8: Subjects used this form to evaluate their buddies

5. Subjects assessed the goodness of buddies selected by themselves as well as those selected by our agent methodology: In order to analyze the recommendation quality of agent results, we examined whether there was a significant difference between the "agent found buddies" and "subject found buddies". To this end, subjects were presented with the top-five "subject-found-buddies" (from step 4 above) and the top-five "agent found buddies" in a random order. Next, the subjects were asked to assess the

buddies using a Likert-type scale of 1-9, shown in Figure 6.9. The top five buddies from this assessment are called the "final subject-found buddies."

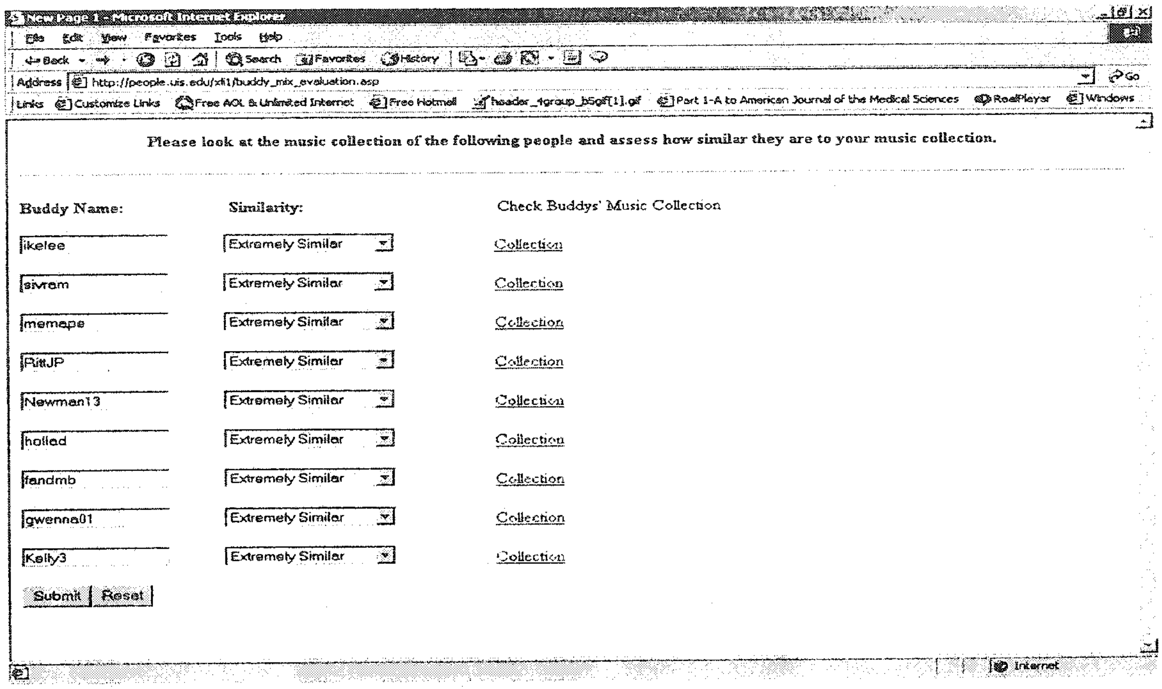


Figure 6.9: Subjects used this form to assess the goodness of buddies selected by themselves as well as those selected by our agent methodology

6. With the results from the above steps, we presented these three sets of buddies to the subjects:

    - Agent found buddies
    - Subject-found buddies
    - Final subject-found buddies

7. Subjects evaluated the goodness of three possible groups of buddies. Subjects were asked to make evaluations on three buddy sets: 1) agent found buddies; 2) subject-found buddies; 3) final subject found buddies,

and to give comments about each set (see Figure 6.10). Since our major concern in designing multi agent systems is the quality of agent recommendation, we need to know to what extent we can trust the recommendation from the agent. This comparison enables us to once again compare the goodness of agent-found buddies with those that subject selected on their own (i.e., subject-found buddies)

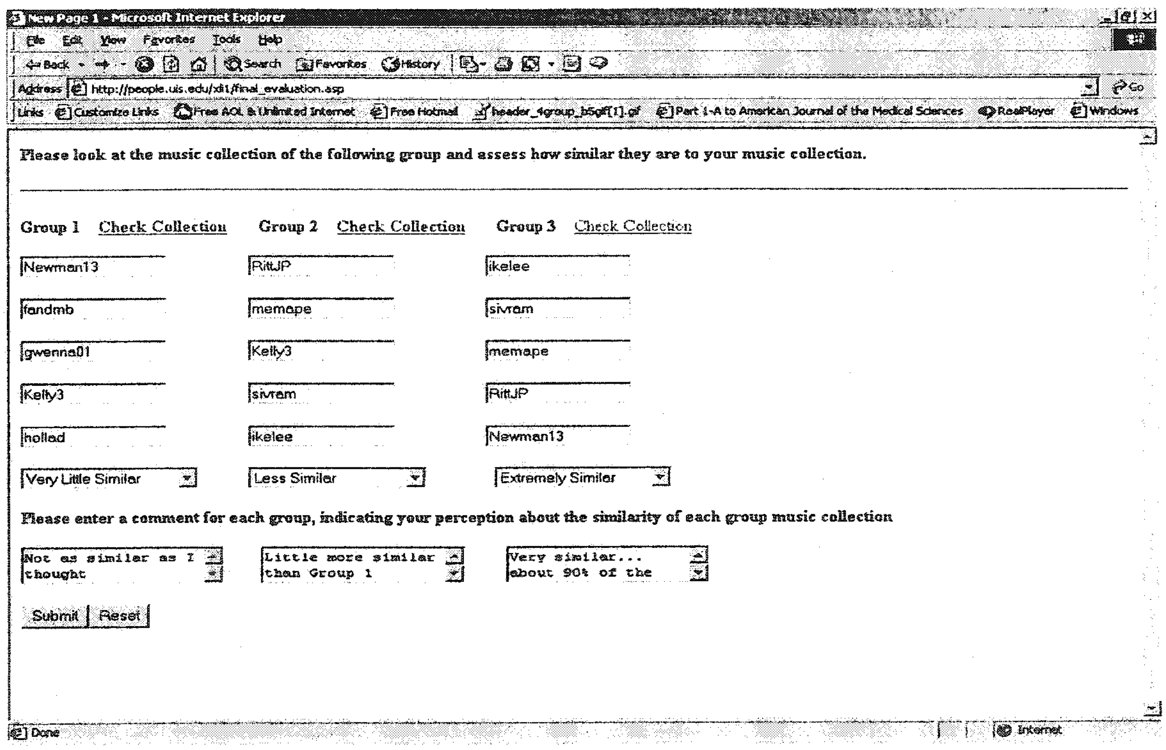


Figure 6.10: Subjects used this form to evaluate the goodness of three possible groups of buddies

The experiment was completed in a computer lab. The author administered the whole process. Since the experiment was 3-6 hours long, it was divided into two parts that were completed within two weeks. The first part covered steps 1 and 2 (i.e., the subjects entered their music preferences and selected their favorite music) and the second part consisted of steps 3 through seven inclusive (i.e., identifying buddies manually and subsequent evaluations).

6.7 Validity Check of Subject Found Buddies

In the test, the subject found their buddies through the message board manually. In the validity check of subject found buddies, we check the commonality between the subjects and subject found buddies to see if there are common attributes existing in each subject-buddy pair. From the Table 1 in Appendix 3, we can see 90% of the subject-buddy pairs share music genre, that is, the subject and the buddy like the same type of music. For the other 10% of the subject-buddy pairs, the subjects and buddies do not share the same music genre, but they have common interest in other attributes, like mood, tempo, beat, etc. For example, both the subject and the buddy like romantic music, so they have common interest in this attribute.

6.8 Results and Analyses

To prove the effectiveness of the agent-based buddy finding methodology, we analyze the test results from the following perspectives: (1) evaluation of the overlap between the music contained in agent found buddies (B_a) (see Table 6.6 for complete definition of symbols used in this section) and music contained in subject found buddies (B_s); (2) comparison of the overlap between agent found buddies (B_a) and final subject found buddies (B_{sf}) with the overlap between subject found buddies (B_s) and the final subject found buddies (B_{sf}); (3) comparison of agent-found buddies (B_a) and subject-found buddies (B_s); (4) evaluation of consistency between Agent Ranking Orders (R_a) and Subject Ranking Orders (R_s); (5) evaluation of items (music) contained in agent-found buddies (E_{ia}) and items (music) contained in subject-found buddies (E_{is}).

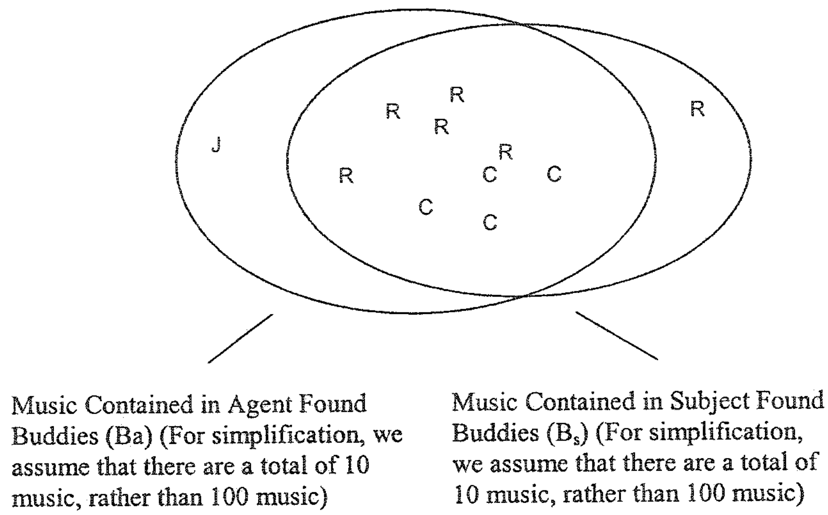
Table 6.6: Notation of Abbreviations

Abbreviation	Definition	Explanation
B_s	Subject-found buddies	buddies that the subject got from the message board through the manual method (Step 3 of experimental procedures), such as $B_{s1}, B_{s2}, B_{s3}, B_{s4}, B_{s5}$. See Figure 5.7.
B_a	agent-found buddies	buddies that the subject got from our proposed agent-based buddy finding methodology, such as $B_{a1}, B_{a2}, B_{a3}, B_{a4}, B_{a5}$.
B_{sf}	final subject-found buddies	buddies that are the top five buddies selected from the mixed set of subject-found buddies and agent-found buddies (step 5 of experimental procedures). See Figure 5.9. For example, one subject has 5 buddies from the manual method, like $B_{s1}, B_{s2}, B_{s3}, B_{s4}, B_{s5}$ (i.e., B_s), and five buddies from the proposed agent methodology, like $B_{a1}, B_{a2}, B_{a3}, B_{a4}, B_{a5}$ (i.e., B_a). After mixing these buddies together, the subject will have a maximum 10 buddies (if there are overlaps between B_s and B_a, the number of buddies in mixed set will be less than 10. Here we assume there is no overlap between B_s and B_a), such as $B_{s1}, B_{s2}, B_{s3}, B_{s4}, B_{s5}, B_{a1}, B_{a2}, B_{a3}, B_{a4}, B_{a5}$. In step 5 of the experimental procedure, the subject ranked these 10 buddies based on their goodness, and got an order of them from high to low, like $B_{a3}, B_{s3}, B_{a4}, B_{s2}, B_{s4}, B_{s5}, B_{a1}, B_{s1}, B_{a2}, B_{a5}$. The set of top five buddies is $B_{a3}, B_{s3}, B_{a4}, B_{s2}, B_{s4}$. We call this buddy set B_{sf}.
R_s	subject ranking orders	subjects' ranking of subject found buddies (Step 4 of experimental procedures). See Figure 5.8. For example, the subject ranked five subject-found buddies (B_s) based on their goodness as $B_{s3} > B_{s2} > B_{s4} > B_{s5} > B_{s1}$. We call this rank R_s.
R_a	agent ranking orders	Agents' ranking of subject found buddies (B_s). For example, in step 4 of the experimental procedures, subjects ranked five subject found buddies as $B_{s3} > B_{s2} > B_{s4} > B_{s5} > B_{s1}$. In our proposed agent methodology, the agent computed fuzzy membership values for these five buddies, so we got another rank order of them with agent methodology, such as $B_{s5} > B_{s3} > B_{s4} > B_{s1} > B_{s2}$. We call this rank R_a.
E_{ia}	evaluation of items (music) contained in agent-found buddies	from step 7 of the experimental procedures. The subject was presented the music collection from the group of agent-found buddies. The evaluation is the similarity of the music contained in this group of buddies to music contained in the music collection of this subject. See Figure 5.10.
E_{is}	evaluation of items (music) contained in subject-found buddies	from step 7 of the experimental procedures. The subject was presented the music collection from the group of subject-found buddies. The evaluation is the similarity of the music contained in this group of buddies to music contained in the music collection of this subject. See Figure 5.10.
E_{isf}	evaluation of items (music) contained in final subject-found buddies	from step 7 of the experimental procedures. The subject was presented the music collection from the group of final subject-found buddies. The evaluation is the similarity of the music contained in this group of buddies to music contained in the music collection of this subject. See Figure 5.10.

6.8.1 Music Overlap Analysis

The music overlap analysis was used to compare the music type commonality between the music contained in agent found buddies (B_a) and the music contained in subject found buddies (B_s). With this test, we calculate the overlap rate between the music contained in agent found buddies (B_a) and the music contained in the subject found buddies (B_s) based on their common music types. This result would reveal whether there is a significant difference between the music types contained in agent found buddies (B_a) and the music types contained in subject found buddies (B_s).

Next we use an example to illustrate the calculation of music overlap for one subject. Of all 100 music (each subject has 20 music in his/her music portfolio, so there are totally 100 music contained in five buddies' portfolios) contained in five agent found buddies (B_a) of this subject, 50 music are Rock music, 40 are Country music and 10 music are Jazz music. And of all 100 music contained in five subject found buddies (B_s) of this subject, 60 music are Rock music, 40 music are Country music, and zero music are Jazz music. The total music overlap between music portfolios of these two groups of buddies is 90%. See Figure 6.11 and Table 6.7.



(C: Country Music; R: Rap Music; J: Jazz Music)

Figure 6.11: Overlap between the Music Contained in Agent Found Buddies (B_a) and the Music Contained in Subject Found Buddies (B_s)

Table 6.7: Example of Music Type Overlap of One Subject

Genre	Music contained in Agent Found Buddies (B_a)	Music contained in Subject Found Buddies (B_s)	Overlap
Rock	50	60	50
Country	40	40	40
Jazz	10	0	0
Total:	100	100	90

We calculated the overall music overlap of this subject by the music overlap in all selected attributes. For example, for genre we find 70% overlap between the music contained in agent found buddies (B_a) and the music contained in subject found buddies (B_s). For the remaining 30% that don't match we use mood to see overlap. This identifies another 20% of the music with overlap. Thus, the total music overlap for genre and mood is 90%. Similarly, we calculate the music overlap based on other remaining attributes, such as tempo, vocal style, etc.

From Table 6.8 and Figure 6.12, the statistical analysis for the music overlap showed that the mean overlap is 99.69%. This finding indicated, generally, that the music contained in the agent found buddies (B_a) and the music contained in subject found buddies (B_s) have significant similarity in the selected attributes.

Table 6.8: Statistics of the Music Overlap

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
Music Overlap	35	98%	100%	99.69%	0.005298
Valid N (listwise)	35				

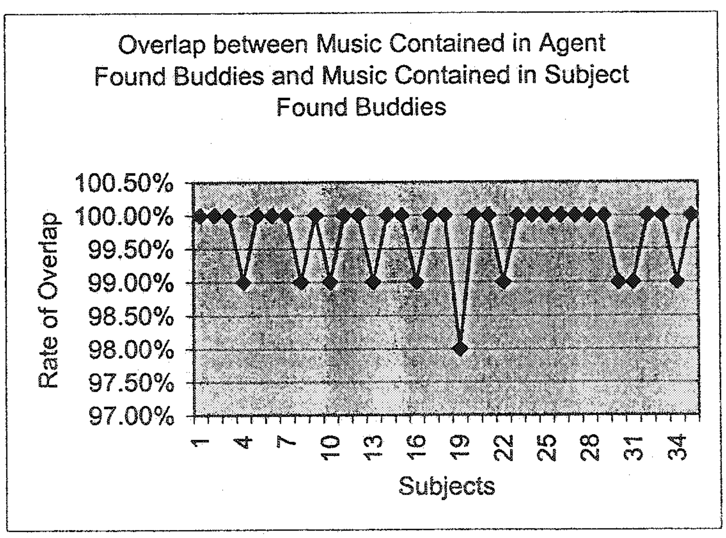


Figure 6.12: Overlap Analysis between the Music Contained in Agent Found Buddies (B_a) and the Music Contained in Subject Found Buddies (B_s)

6.8.2 Buddy Overlap Analysis

The buddy overlap analysis was used to analyze the significance of agent found buddies (B_a) and subject found buddies (B_s) to the final subject buddy selection (B_{sf}). To this end, we assess the final pooled agent-found buddies (B_a) and subject-found (B_s) to see how many final subject selected buddies (B_{sf}) come from each of the two groups (B_a and B_s). This enables us to assess the significance of each of the two groups of buddies in providing the best buddy-suggestions. See Figure 6.13.

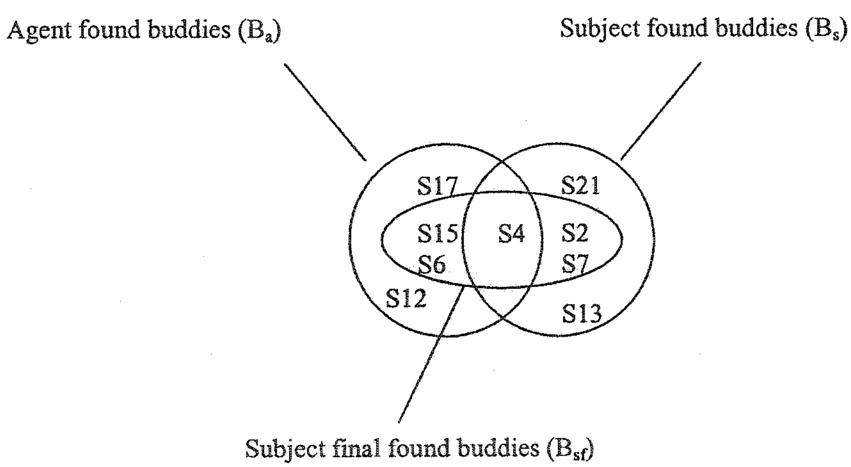


Figure 6.13: Overlap between Agent Found Buddies (B_a) and Subject Found Buddies (B_s)

A pair wise T-test was conducted to test overlaps between agent-found buddies (B_a) and final subject-found buddies (B_{sf}) and overlaps between subject-found buddies (B_s) and final subject-found buddies (B_{sf}) (i.e., to test $B_a \cap B_{sf}$ and $B_a \cap B_{sf}$). This result would reveal whether there is a significant difference between the significance of agent found buddies (B_a) and subject found buddies (B_s) to the subjects' final buddy selection (B_{sf}).

The mean value of the overlap between agent found buddies (B_a) and final subject found buddies (B_{sf}) is 55%, and the mean value of the overlap between subject found buddies (B_s) and final subject found buddies (B_{sf}) is 59% (see Table 6.9). The pair wise T-test for the evaluation of the overlap values of these two groups of buddies showed no significant difference between them ($T=-0.734$, $p=0.468$) (See Table 6.10). This result shows that we cannot reject the second hypothesis in terms of the buddy overlap:

H2: There is no significant perceived difference between the buddies found through the proposed agent-found methodology and buddies identified by the subjects.

Table 6.9: Buddy Overlap with the Final Subject Buddy Selection

	Mean	N	Std. Deviation	Std. Error Mean
Overlap between Agent Found Buddies (B_a) and Final Subject Found Buddies (B_{sf})	55%	34	0.18	0.03
Overlap between Subject Found Buddies (B_s) and Final Subject Found Buddies (B_{sf})	59%	34	0.19	0.03

Table 6.10: Comparison of Buddy Overlap with the Final Subject Buddy Selection (Pair wise T-test)

Paired Samples Test

	Paired Differences					t	df	Sig. (2-tailed)
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference				
				Lower	Upper			
Pair: Agent-Subject	-0.04324	0.34339	0.05889	-0.163	0.0766	-0.734	33	0.468

6.8.3 Evaluation of Agent-Found Buddies

A pair wise T-test was conducted to test users' evaluations of agent-found buddies (B_a), and subject-found buddies (B_s). To do this, we first ranked the order of the agent found buddies and subject found buddies for each subject based on the subject's evaluations of these buddies. See Table 6.11. This result would reveal whether there is significant difference between subjects' evaluations of buddies found by agents (B_a) and subjects' evaluations of buddies found by subjects (B_s) manually.

Table 6.11: Sample of the Ranked Agent Found Buddies and the Ranked Subject Found Buddies for Subject S1

Subject	Agent Found Buddies (B_a)	Subject's (S1) Evaluations
S1	S3	8
S1	S10	7
S1	S11	6
S1	S7	5
S1	S9	4

Subject	Subject Found Buddies (B_s)	Subject's (S1) Evaluations
S1	S7	9
S1	S26	6
S1	S8	5
S1	S9	4
S1	S15	3

The mean value of the subject found buddies (B_s) is 6.27, and the mean value of the agent found buddies (B_a) is 6.05 (see Table 6.12). The pair wise T-test for the evaluation of these two groups of buddies showed no significant difference between them (T=-0.1355, p=0.177) (See Table 6.13). This result shows the second hypothesis cannot be rejected: **There is no significant perceived difference between buddies found through the proposed agent-found methodology and buddies identified by the subjects.**

Table 6.12: Mean Evaluation Values of Subject-Found Buddies (B_s) and Agent-Found Buddies (B_a)

	Mean	N	Std. Deviation	Std. Error Mean
Agent Found Buddies	6.05	170	2.35	0.18
Subject Found Buddies	6.27	170	2.39	0.18

Table 6.13: The T-test of Evaluation Score Difference between Subject-Found Buddies (B_s) and Agent-Found Buddies (B_a)

	Paired Differences					t	df	Sig. (2-tailed)
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference				
				Lower	Upper			
Pair: Agent-Subject	-0.218	2.094	0.161	-0.5347	0.09938	-1.355	169	0.177

6.8.4 Consistency Analysis between Agent Ranking Orders (R_a) and Subject Ranking Orders (R_s)

Rank consistency was used to compare the agent ranking (R_a) and subject ranking (R_s) for the same set of subject found buddies (B_s). The rank consistency index is based on the correlation coefficient between these two rankings. For example, subject S3 manually found five buddies from the message board, e.g., S1, S5, S6, S7, S9. These five subject-found buddies (B_s) can be ranked from high to low by the evaluation values suggested by the subjects. See Table 6.14. Similarly, these five subject-found buddies (B_s) can be ranked by the buddy membership values acquired from proposed agent-based buddy finding methodology (See Table 6.15). Rank correlation analysis was used to assess the degree of consensus between the subject ranking (R_s) and the agent ranking (R_a). A positive significant correlation would mean the proposed agent-based buddy finding methodology is consistent with the subject manual buddy finding methodology in evaluating buddies.

Table 6.14: Samples of the Conversion from the Subject's Satisfaction Values to the Rank Values

Subject	Subject Found Buddies (B_s)	Subject's Evaluation
S3	S1	7
S3	S5	6
S3	S6	9
S3	S7	4
S3	S9	5

=====>

Subject	Subject Found Buddies (B_s)	Converted Rank Values
S3	S1	2
S3	S5	3
S3	S6	1
S3	S7	5
S3	S9	4

Table 6.15: Samples of the Subject Found Buddies Ranked by the Proposed Agent-based Buddy Finding Methodology

Subject	Subject Found Buddies (B_s)	Agent Rank
S3	S1	3
S3	S5	1
S3	S6	2
S3	S7	4
S3	S9	5

From Table 6.16, the rank correlation coefficient between the agent ranking orders (R_a) and subject ranking orders (R_s) is 0.168. Therefore, the agent ranking orders (R_a) are positively correlated with the subject ranking orders (R_s). This finding also shows that the second hypothesis cannot be rejected in terms of buddy rank consistency.

H2: There is no significant perceived difference between the buddies found through the proposed agent-found methodology and buddies identified by the subjects.

Table 6.16: Rank Correlation Coefficient between Agent Ranking (R_a) and Subject Ranking (R_s)

	Rank Correlation Cofficient (Pearson); N=170
Agent Rank-Subject Rank	0.168

*positively correlated at the 0.05 level (2-tailed).

6.8.5 Evaluation of Items (Music) Contained in Agent-Found Buddies (B_a)

A one-way ANOVA test was used to test the users' satisfactions with the items (music) contained in the three groups of buddies: (1) agent-found buddies (B_a); (2) subject-found buddies (B_s); (3) final subject-found buddies (B_{fs}). This result would reveal whether there is a significant difference among subjects' evaluations of music contained in three possible buddy groups. See Table 6.17 (please check Appendix 5 for complete list).

Table 6.17: Sample of the Subjects' Satisfaction with the Items (music) Contained in Three Groups of Buddies (E_{ia}, E_{is}, E_{isf})

Subject	Evaluations of Music Contained in Subject Found Buddies (E_{is})	Evaluations of Music Contained in Agent Found Buddies (E_{ia})	Evaluations of Music Contained in Final Subject Found Buddies (E_{isf})
S1	2	5	9
S2	7	6	7
S3	8	7	7
S4	8	5	7
S5	9	7	8

From Table 6.18, the mean satisfaction value of items (music) contained in subject-found buddies (E_{is}) is 6.97; the mean satisfaction value of items (music) contained in agent-found buddies (E_{ia}) is 6.29; and the mean satisfaction value of items (music) contained in the final buddy selection (E_{isf}) is 6.88. The ANOVA test (see Table 6.19) for the evaluation of items (music) contained in three different buddy selections (E_{ia}, E_{is}, E_{isf}) showed no significant difference ($F=1.714$, $p=0.185$). This indicated that the music contained in the agent-found buddies (B_a) received the same satisfaction level as the music contained in subject-found buddies (both B_s and B_{sf}).

Table 6.18: Evaluation of the Items (Music) Contained in Different Buddy Selections

	Mean	Std.
E_{is}	6.97	1.62
E_{ia}	6.29	1.49
E_{isf}	6.88	1.79

Table 6.19: Comparison of Evaluation of Three Groups of Recommendation (ANOVA)

Source	D.F.	Sum of Squares	Mean squares	F	p
Between groups	2	9.2	4.6	1.71	0.19
Within groups	99	265.56	2.68		
Total	101	274.76			

This finding also indicates that we cannot reject the second hypothesis:

H2: There is no significant perceived difference between the buddies found through the proposed agent-found methodology and buddies identified by the subjects.

6.8.6 Concluding Remarks and Subjects' Comments

In the test, we analyzed the test results from several perspectives. First, we compared the subjects' satisfactions to agent found buddies (B_a) with subjects' satisfactions to subject found buddies (B_s). The pair wise T-test for the evaluation of these two groups of buddies showed no significant difference between them (T=-0.1355, p=0.177). Secondly, we compared the ranking from the agent-based methodology (R_a) with the ranking from the subject manual method (R_s) to the same set of subject found buddies (B_s). The rank correlation coefficient between the agent ranking orders (R_a) and subject ranking orders (R_s) is 0.168. The result indicated that the agent ranking order (R_a) is positively correlated with the subject ranking order (R_s). Thirdly, we compared the subjects' satisfactions in the music contained in agent found buddies ($E_{ia,}$) with subjects' satisfactions to the music contained in subject found buddies (E_{is} and E_{isf}). The ANOVA test for the evaluation of music contained in three different buddy selections ($E_{ia,}$ $E_{is,}$ E_{isf}) showed no significant difference (F=1.714, p=0.185). Lastly, we compared the significance of agent-found buddies (B_a) to the subject's final buddy selection (B_{sf}) with the significance of subject found buddies (B_s) to the subject's final buddy selection (B_{sf}). The pair wise T-test for the evaluation of these two groups of buddies (B_a and B_s) showed no significant difference between their significances in providing the final buddies selection (B_{sf}) (T=-0.734, p=0.468). From several perspectives, we proved the second hypothesis: that there is no significant perceived difference between the buddies found through the proposed agent-found methodology (B_a) and buddies identified by the subjects (B_s).

In evaluating the performance of the system, it is valuable to look at how the students actually felt about their choices, as reflected in the comments entered as part of the simulation. The findings from the subjects' comments are highly consistent with the results from statistical analysis. For example, with the music overlap analysis, we can see that 99.69% of the music contained in the agent found buddies (B_a) and the music contained in the subject found buddies (B_s) have similar characteristics. The ANOVA test for the evaluation of music contained in three different buddy selections (E_{ia}, E_{is}, E_{isf}) showed no significant difference ($F=1.714$, $p=0.185$). Consistent with these findings from statistical analysis, subjects commented thus on the music contained in the agent found buddies (B_a): "*this group is very similar to me,*" "*similar,*" "*Somebody has same experience in music,*" "*it is close but good songs,*" etc. Compared with the music contained in the collections of subject found buddies (B_s), subjects found the music contained in the agent found buddies (B_a) to be as good as the music contained in the collections of subject found buddies. For example, one subject commented: "*good one, I like your choices very much*"; and another subject remarked that "*Although there were different songs in this list, I recognized many of them. So, I felt this list was about as good as the list for group 1* [group 1 is the subject found buddy group]." More directly, some subjects simply considered that the agent found buddies (B_a) have the same music tastes as his/her own. For example, one subject commented that "*Somebody has same experience in music.*" Subjects' comments showed their acceptance of agent found buddies (B_a) through the recognition of the similarity between the music tastes of agent found buddies (B_a) and the music tastes of subject found buddies (B_s).

Thus, the above analysis shows that that there is no significant perceived difference between the agent-found buddies (B_a) and buddies identified by the subjects (B_s).

7. Conclusions and Future Research

7.1 Introduction

This thesis has presented a variety of buddy finding technologies, our proposed agent-based buddy finding methodology, and two experiments to testify to the effectiveness of our proposed methodology. This chapter discusses the conclusions from the findings of the empirical test and some future research issues. The text is organized as follows: Section 7.2 summarizes results of our research conducted in the thesis; Section 7.3 presents implication of practice of our proposed methodology; Section 7.4 discusses implications for research, and limitations.

7.2 Summary of Results

The central idea underlying software agents is that of delegation. The user delegates a task to the agent and the agent autonomously performs that task on behalf of the user (Norman and Reed, 2001). Therefore, our most important research objective is to find out whether the proposed agent-based buddy finding methodology can mimic human buddy finding behavior properly. To do this, in the first test, we selected cluster analysis as the benchmark in buddy finding. We compared our proposed methodology with cluster analysis. The result proved that our proposed agent-based buddy-finding methodology is as good as clustering analysis in buddy finding. The difference is that our methodology is used for finding buddies in a decentralized environment in which complete knowledge of population is not possible, while cluster analysis is applied in a centralized environment with complete knowledge of population. Thus, our methodology can be employed in a decentralized peer-peer environment. Furthermore, our proposed methodology can use several indicators to identify a buddy agent; for example, we can use quality of response, duration, and cost to determine the most appropriate buddies.

In our second test, we compared our proposed methodology with human subjects in the scenario of music buddy finding. The purpose of this test was to find out whether our proposed methodology can simulate human perception in assessing the goodness of buddies. The first important discovery from this test is that the agents can work as well as human subjects in finding music buddies. The second important discovery from this test indicates that subjects' satisfaction with the items (songs) contained in agent-found buddies is significantly similar to the items (songs) contained in subject found buddies. These test results support our hypothesis that led to the conclusion that our proposed methodology is useful in helping subjects finding buddies in distributed environments. Users can use the proposed methodology to find buddies in the virtual world.

7.3 Implication for Practice

Our methodology, presented in this thesis, can be used to facilitate knowledge sharing among large numbers of users in a distributed environment. For example, many music lovers share music through Napster (www.napster.com), Morpheus (www.morpheus.com), Kazaa (www.kazaa.com/us/index.php), etc. Napster enables a music lover to connect to a community of millions of other music lovers. It allows each music lover to search and browse for music files in MP3 and WMA formats and to chat with any other member of the Napster Community. People search music by artists and title, and then download matched music from other music lovers' collections. With Napster, people may need to sift through hundreds of music files to find the one that they have not heard before but matches their taste; our proposed buddy-agents-membership methodology can reduce an individual's search effort by recommending the best-matched music lovers' group. In this application, the music style preferences are represented by parameters of composer, rhythm, price, etc. Our buddy-agent-membership methodology can identify the best members (i.e., buddy-group members) that can offer information about music to a person seeking advice. This helps music lovers to narrow down their selection and keep informed about new music of their interest. Sharing music and pertinent knowledge among individuals is one form of using our agent-based methodology.

We can use the proposed multi-agent methodology in support of knowledge management within and between firms. Firms in technologically intensive fields rely on collaborative relationships to access, survey and exploit emerging technological opportunities (Powell, 1998). For example, strategy-consulting firms such as Bain, Boston Consulting Group, and McKinsey emphasize a personalization strategy in knowledge management. To make their personalization strategies work, firms like Bain invest heavily in building networks of people. Knowledge is shared not only face-to-face but also over the telephone, by e-mail, and via video-conferences. McKinsey fosters management of its knowledge workers in different ways: by transferring people between offices; by supporting a culture in which consultants are expected to return phone calls from colleagues promptly; by creating directories of experts; and by using "consulting directories" within the firm to assist project teams (Hansen et al, 1999). It is commonly believed that learning is enhanced when knowledge workers are encouraged to collaborate with like-minded individuals (Hansen et al., 1999). Our methodology makes use of a combination of agent technology and distributed CBR systems in support of knowledge sharing among like-minded DMs.

7.4 Limitations and Future Research

The contribution of our proposed methodology to literature is the agent coordination strategy in support of multi-agent systems. Since there is no middle agent in a decentralized control structure, agents use an acquaintance list to communicate only with a small subset of agents (Sikora and Shaw, 1998). Getting the right team of agents and controlling them is of prime interest in the decentralized control structure for a large number of users (Dignum et al., 2001). As we declared in chapter one, the existing acquaintance structure of agent coordination is fixed--which means the agent system builders need to create the acquaintance list when they are implementing the multi-agent system. While the results of empirical tests of our proposed methodology are encouraging, there are still some limitations as follows.

First, we need to further test the proposed buddy finding methodology with a large number of possible users. After all, the real advantage of multi-agent systems is in reducing information overload for the environments in which there are a very large number of users in need of sharing information with each other (e.g., sharing music on the napster.com message board). In this case, due to large number of possible users, it is impossible to have one-to-one communication among all the participants. Therefore, identification of the best buddies becomes an asset. Using our proposed buddy finding methodology, the agent can dynamically identify a small finite set of buddy agents. This enables information sharing among a very large number of decision makers who are unaware of each other's existence and/or information needs. We conjecture that in an empirical test with very large number of subjects (e.g. more than 200 people), our proposed buddy-finding methodology would outperform human selection. Thus a formal testable hypothesis to be tested with a very large sample size would be: The buddies found through the proposed agent–found methodology are significantly perceived to be better than buddies identified by the subjects.

Second, our controlled empirical tests enabled us to assess the stated hypotheses. In our test, subjects were able to see recommendations from other people and compare them with those filtered by the agent. This comparison enabled the subjects to assess whether the recommendation made by their agents are as good as those offered by the human through a message board. The question arises as to what happens if the subjects did not have access to human recommendations through the message board: Would they trust the recommendations of their agents and accept them at face value in the real virtual community?

Trust, within the context of information systems, represents the reliance of business actors on other actors or information systems (Wilikens et al., 2002). In virtual communities, trust is especially important where the absence of workable rules makes essential the reliance on the socially acceptable behavior of others (Ridings et al., 2002). Trust has a major impact in relationships among group members (e.g., users and their buddies); it encourages a climate conducive to sharing of knowledge (Nelson and Cooprider, 1996). Therefore, virtual teams require their members to rely heavily on trust in coworkers (Morris et al., 2002; Ridings et al., 2002). The lack of trust results in individual work with little collaboration, worker dissatisfaction, and team attrition (Johnson, 2001).

Users, in online communities, build trust with each other mainly by cooperative interactions through message boards (Hoffman et al., 1999; Ridings et al., 2002). However, when using our proposed buddy finding methodology, the users would receive the recommended buddies directly from agents (i.e., without interactions through a message board). Consequently, lack of prior interactions between the users and the recommended buddies might influence the users' trust about the usefulness of the recommended agent-found buddies. Research findings indicate that it is indeed possible to create trust between users without prior interactions (Ba and Pavlov, 2002). For example, in an online community, online feedback mechanisms help build trust among users by allowing users to rate the quality of the service (Ba and Pavlov, 2002). Developing a feedback mechanism to help users build trust with the agent recommended buddies is another empirical research issue that we intend to pursue in the future.

Third, our proposed buddy-finding agent methodology does not provide adaptability to the change of users' preferences. Due to the dynamic nature of knowledge sharing environments, responses from some of the buddy-agents may become so poor that they lose their status as a "buddy." We need to allow the user to specify a threshold as a performance benchmark. Thus, as soon as the number of buddy-agents deteriorates below a threshold, new buddy-agents must be recruited. This threshold obviously cannot be less than one. However, the threshold could be set by the user according to the minimum confidence level in the other agents consulted. Therefore, one user may be comfortable with responses from at least six buddy-agents and another user may set the minimum threshold at 20. One drawback of our methodology is that when the number of buddy-agents goes below a threshold, a request must be sent to all other agents to recruit new buddy-agents. This could be inefficient (e.g., computing workload increases proportionally as the number of agents increases). Future research may identify methodologies that utilize information about the buddy-agents of the requesting agents to assess new buddy-agents' membership.

Another issue in need of attention is selection procedures for responding to a request. We used the FIFO (first-in-first-out) procedure by a responding agent to serve the requesting agents in the queue. This can be an inefficient procedure since it is based on an assumption that is equally likely to give valuable recommendations to all the requesting agents. Markov decision process modeling (Sutton and Barto, 1998) may provide a better procedure for serving requests from buddy-agents. It is our hope that our findings will generate fruitful discussion and provoke further research.

Finally, our stated hypotheses were tested in only two domains. There are hundreds of potential application domains for knowledge sharing that benefit greatly from using our multi-agent buddy finding approach. Obviously, application of the proposed methodology needs to be assessed in other domains such as (1) sharing complex medical knowledge within the communities of common interest, (2) sharing problem-solving skills among software developers, and (3) sharing socio-political views among citizens with similar interests

References

Alavi, M., and Leidner, D. E., "Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues", *MIS Quarterly,* Volume 25, Number 3, March 2001, pp. 107-136.

Arcos, J. L., "T-Air: A Case-Based Reasoning System for Designing Chemical Absorption Plants," *Case-Based Reasoning Research and Development, LNAI 2080, 4th International Conference on Case-Based Reasoning, ICCBR 2001, Vancouver, BC, Canada, July/August 2001,* Aha, D. W. and Watson, I. (Eds.), Springer-Verlag, Berlin, 2001, pp. 576-588.

Augier, M., and Vendelo, M.T., "Networks, Cognition and Management of Tacit Knowledge," *Journal of Knowledge Management,* Volume 3, Number 4, 1999, pp. 252-261.

Austin, J. R., "A cognitive framework for understanding demographic influences in groups," *International Journal of Organizational Analysis,* Volume 5, Number 4, Oct 1997, pp.342-359.

Awad, E. M., *Building Expert Systems: Principles, Procedures, and Applications*, West Publishing Co., Los Angles, CA, 1996.

Ba, S., and Pavlov, P. A., "Evidence of the effect of trust building technology in electronic markets: price premiums and buyer behavior," *MIS Quarterly*, Volume 26, Number 3, 2002, pp. 243-268.

Bagshaw, M., and Bagshaw, C., "Leadership in the Twenty-First Century," *Industrial & Commercial Training*, Volume 31, Number 6, 1999, pp. 236-239.

Barnatt, C., "Virtual Communities and Financial Services-on-line Business Potentials and Strategic Choice," *International Journal of Banking Marketing*, Volume 16, Number 4, 1998, pp. 161-169.

Bellifemine, F., Poggi, A., and Rimassa, G., "Developing Multi-agent Systems with Jade," *Intelligent Agents VII, Agent Theories Architectures and Languages, 7th International Workshop, ATAL 2000, Boston, MA, USA, July 2000 Proceedings, LNAI 1986*, Castelfranchi, C. and Lespérance, Y. (Eds.), Springer-Verlag, Berlin, 2001, pp. 89-103.

Bernus, P., and Uppington, G., "Co-ordination of Managemnet Activities-Mapping Organizational Structure to the Decision Structure," *Coordination Technology for Collaborative Application - Organizations, Processes, and Agents, W.Conen and G.Neumann (Eds), LNCS 1364*, Springer-Verlag, Berlin, 1998, pp. 25-38.

Bocionek, S.R., "Agent systems that negotiate and learn," *International Journal of Human-Computer Studies*, Volume 42, 1995, pp. 265-288.

Bordetsky, A., and Mark, G., "Memory-Based Feedback Controls to Support Groupware Coordination," *Information Systems Research, 2000 INFORMS*, Volume 11, Number 4, December 2000, pp. 366-385.

Botti, V. J., Carrascosa, C., Julián, V., and Soler, J., "Modeling Agents in Hard Real-Time Environments," *MultiAgent System Engineering, 9th European Workshop on Modeling Autonomous Agents in a Multi-Agent World, MAAMAW '99*, Valencia, Spain, June 30 - July 2, 1999, Proceedings, LNCS 1647, Springer 1999, pp. 63-76.

Bryson, J., and Stein, L. A., "Architectures and Idioms: Making Progress in Agent Design," *Intelligent Agents VII, Agent Theories Architectures and Languages, 7th International Workshop, ATAL 2000, Boston, MA, USA, July 2000 Proceedings, LNAI 1986*, Castelfranchi, C. and Lespérance, Y. (Eds.), Springer-Verlag, Berlin, 2001, pp. 73-78.

Byrd, D., and Crawford T., "Problem of music information retrieval in the real world," *Information Processing & Management*, Volume 38, 2002, pp. 249-272.

Cassell, J., and Bickmore, T., "External manifestations of trustworthiness in the interface," *Communications of the ACM*, Volume 43, Number 2, 2000, pp. 50-56.

Catterall, M. and Maclaran, P., "Researching consumers in virtual worlds: A cyberspace odyssey," *Journal of Consumer Behaviour*, Volume 1, Number 3, Feb 2002, pp. 228-237.

Chai, W., and Vercoe, B., "Using User Models in Music Information Retrieval Systems," *Proceedings of International Symposium on Music Information Retrieval*, Plymouth, Massachusetts, Oct. 2000.

Chakrapani, C., "Cluster Analysis," *Marketing Research*, Volume 14, Number 1, Spring 2002, pp. 52.

Chase, R.L., "Knowledge Management Benchmarks," *The Journal of Knowledge Management*, Volume 1, Number 1, September 1997a, pp. 83-92.

Chase, R.L., "The Knowledge Based Organization: An International Survey," *The Journal of Knowledge Management*, Volume 1, Number 1, September 1997b, pp. 38-49.

Cheetham, W., "Benefits of Cased-Based Reasoning in Color Matching," *Case-Based Reasoning Research and Development, LNAI 2080, 4th International Conference on Case-Based Reasoning, ICCBR 2001, Vancouver, BC, Canada, July/August 2001,* Aha, D. W. and Watson, I. (Eds.), Springer-Verlag, Berlin, 2001, pp. 589-596.

Chen, H., Chau, M., and Zeng, D., "CI Spider: a tool for competitive intelligence on the Web," *Decision Support Systems*, Volume 34, Issue 1, December 2002, pp. 1-17.

Civi, E., "Knowledge Management as a Competitive Asset: A Review," *Marketing Intelligence & Planning*, Volume 18, Number 4, 2000, pp.166-174.

Cohen, P. R., and Levesque, H. J., "Confirmation and Joint Action," in *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, 1991, Sydney, pp. 951-957.

Conway, D. G., and Koehler, G. J., "Interface Agents: Caveat Mercator in Electronic Commerce," *Decision Support Systems,* Volume 27, Issue 4, Jan 2000, pp. 355-366.

Cothrel, J., and Williams, R. L., "On-line Communities: Helping Them Form and Grow," *Journal of Knowledge Management*, Volume 3, Number 1, 1999, pp. 54-60.

Curle, D., "Filtered news services: Solutions in search of your problem?" *Online*, Volume 22, Number 2, Mar/Apr 1998, pp.14-24.

Daniel, E., and Wilson, H.M.A., "Adoption of e-commerce by SMEs in the UK," *International Small Business Journal*, no. 3 (Aug 2002): pp. 253-270.

Davis, R., and Smith, R.G., "Negotiation as a Metaphor for Distributed Problem Solving," *Artificial Intelligence*, Volume 20, Number 1, January 1983, pp. 63-109.

Decker, K., Sycara. K., and Zeng, D., "Intelligent Agents in Portfolio Management," *Agent Technology: Foundations, Applications, and Markets*, Jennings, N. R., and Wooldridge, M.J. (Eds.), Springer-Verlag, Heidelberg, 1998, pp. 267-281.

Dignum, F., Dunin-Keplicz, B., and Verbrugge, R., "Agent Theory for Team Formation by Dialogue," *Intelligent Agents VII, Agent Theories Architectures and Languages, 7th International Workshop, ATAL 2000, Boston, MA, USA, July 2000 Proceedings, LNAI 1986*, Castelfranchi, C. and Lespérance, Y. (Eds.), Springer-Verlag, Berlin, 2001, pp. 150-166.

Dixon, N. M., "The changing face of knowledge," *The Learning Organization*, Volume 6, Number 5, 1999, pp. 212-216.

Dougherty, V., "Knowledge is about people, not databases," *Industrial and Commercial Training*, Volume 31, Number 7, 1999, pp. 262-266.

Dragoni, A. F., Giorgini, P., and Serafini, L., "Updating Metal States from Communication," *Intelligent agents VII, Agent Theories Architectures and Languages, 7th International Workshop, ATAL 2000, Boston, MA, USA, July 2000 Proceedings, LNAI 1986*, Castelfranchi, C. and Lespérance, Y. (Eds.), Springer-Verlag, Berlin, 2001, pp. 15-30.

Dron, J., Mitchell R. and Siviter P., "From Answer Garden to Answer Jungle," *Education + Training*, Volume 40, Number 8, 1998, pp. 353-358.

Duda, R., and Hart, P., *Pattern Classification and Scene Analysis*, Wiley, New York, NY, 1973.

Durfee, E. H., "Distributed Problem Solving and Planning," *Multi-Agent Systems and Applications, LNAI 2086*, Luck, M., Marik, M., Stepankova, O. and Trappl, R. (Eds.), Springer-Verlag, Berlin, 2001, pp. 118-149.

Dyckman, T.R., Downes, D.H., and Magee, R.P., *Efficient Capital Markets and Accounting*, Prentice-Hall, Englewood Cliffs, NJ, 1975.

Dyer, D. E., "Multiagent Systems and DARPA, " *Communications of the ACM*, Volume 42, Number 3, Mar 1999, pp.53.

Edmonds, E. A., Candy, L., Jones, R., and Soufi, B., "Support for Collaborative Design: Agents and Emergence," *Communications of the ACM*, Volume 37, Number 7, July 1994, pp. 41-47.

Fernandez, G., and Wijegunaratne, I., "A Cooperation Approach to Distributed Application Engineering," *Coordination Technology for Collaborative Applications - Organizations, Processes, and Agents, Lecture Notes in Computer Science 1364*, Conen, W., Neuman, G., Goods, G., Hartmanis J. and Leeuwen, J. V. (Eds.), Springer-Verlag, Berlin, 1998, pp. 39-48.

Finin, T., Fritzson, R., McKay, D., and McEntire R., "KQML as an Agent Communication Language," *The Proceedings of the Third International Conference on Information and Knowledge management (CIKM'94),* ACM Press, November 1994, pp. 456-463.

Finin, T., Fritzson, R., McKay, D., and McEntire, R., "KQML - A Language and Protocol for Knowledge and Information Exchange," *Technical Report No. CS-94-02,* University of Maryland, Department of Computer Science, 1994b.

Finin, T., Labrou, Y., and Mayfield, J., "KQML as an Agent Communication Language," *Software Agents*, Bradshaw, J.M. (Ed.), The MIT Press, Cambridge, MA, 1997, pp. 291-316.

Flores, R. A., Kremer, R.C., and Norrie, D. H., "An Architecture for Modeling Internet-Based Collaborative Agent Systems," *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems, LNAI 1887*, *International Workshop on Infrastructure for Scalable Multi-Agent Systems, Barcelona, Spain, June 2000, Revised Papers,* Wagner, T. and Rana, O. (Eds.), Springer-Verlag, Berlin, 2001, pp. 56-63.

Gallagher, M., and Mansour. A., "An Analysis of Hotel Real Estate Market Dynamics," *Journal of Real Estate Research,* Volume 19, Number ½, Jan-Apr 2000, pp. 133-164.

Gallegos, F., "Decision support systems: an overview," *Information Strategy*, Volume 15, Number 2, Winter 1999, pp. 42-45.

Gehrt, K. C., and Shim, S., "A Shopping Orientation Segmentation of French Consumers: Implications for Catalog Marketing," *Journal of Interactive Marketing*, Volume 12, Number 4, Autumn 1998, pp. 34-46.

Geyer, F., "Virtual Communities in Cyberspace," *Kybernetes*, Volume 25, Number 4, 1996, pp. 60-66.

Giampapa, J. A. and Sycara, K., "Conversational Case-Based Planning for Agent Team Coordination," *Case-Based Reasoning Research and Development, LNAI 2080, 4th International Conference on Case-Based Reasoning, ICCBR 2001, Vancouver, BC, Canada, July/August 2001*, Aha, D. W. and Watson, I. (Eds.), Springer-Verlag, Berlin, 2001, pp. 189-203.

Globerman, S. and Roehl, T.W. S. S., "Globalization and electronic commerce: inferences from retail brokering," *Journal of International Business Studies,* Volume 32, Number 4 , 2001, pp. 749-768.

Good, N., Schafer, J. B., Konstan, J. A., Borchers, A., Sarwar, B., Herlocker, J., and Riedl, J., "Combining Collaborative Filtering with Personal Agents for Better Recommendations," *Proceedings of the Sixteenth National Conference on Artificial Intelligence and Eleventh Conference on Innovative Applications of Artificial Intelligence*, July 18-22, 1999, Orlando, Florida, USA. AAAI Press / The MIT Press, 1999, pp. 439-446.

Gordon, R. S., "Online discussion forums," *Link – up,* Volume 17, Number 1, Jan/Feb 2000, pp12.

Gottschalk, P., "Knowledge Management in the Professions: The Case of IT Support in Law Firms," *Proceedings of the 33rd Hawaii International Conference on System Sciences*, Volume 3, Maui, Hawaii, January 4-7, 2000.

Gould, D., "Virtual organization," in *Leading Virtual Teams* (online) 2, www.seanet/com/-daveg/ltv.htm, September 1999.

Gray, P. H., and Meister, D. B., "Anomaly Reconciliation in Electronic Discussion Groups," *Twenty-Second International Conference on Information Systems*, New Orleans, LA, USA, 2001, pp. 339-348.

Grise, Mary-Liz, and Gallupe, R. B., "Information overload: Addressing the productivity paradox in face-to-face electronic meetings," *Journal of Management Information Systems,* Volume 16, Number. 3, Winter 1999/2000, pp. 157-185.

Grosz, B. J., and Sidner, C. L., "Plans for Discourse," *Intentions in Communication*, in *Cohen, P. R., Morgan, J., & Pollack, M., (Eds.),* MIT Press, Cambridge, MA, 1990, pp. 417-445.

Gupta, K., "A Framework for the Design and Development of Diagnostic Case-Based Reasoning Systems," *Ph.D. Thesis*, School of Business, McMASTER University, 1996.

Gupta, K.M., and Montazemi, A.R., "Empirical Evaluation of Retrieval in Case-Based Reasoning Systems Using Modified Cosine Matching Function," *IEEE Transaction on Systems, Man, and Cybernetics*, Volume 27, Number 5, 1997, pp. 601-612.

Haldin-Herrgard, T., "Difficulties in Diffusion of Tacit Knowledge in Organizations," *Journal of Intellectual Capital*, Volume 1, Number 4, 2000, pp. 357-365.

Hanson, J.E., and Kephart, J.O., "Combating Maelstrom in Networks of Communicating Agents," *AAAI-99 Sixth National Conference on Artificial Intelligence*, Orlando, Florida, July 18-22, 1999, pp.17-23.

Hansen, M. T., Nohria, N., and Tierney, T., "What is Your Strategy for Managing KM?" *Harward Business Review*, March 1999, pp.105-116.

Hartigan, J. A., *Clustering Algorithms*. John Wiley & Sons, 1975.

Harvard Management Update, "Do We Know How to Do That? Understanding Knowledge Management," February 1999, pp. 1-5.

Häubl, G., and Trifts, V., "Consumer Decision-Making in Online Shopping Environments: The Effects of Interactive Decision Aids," *Marketing Science*, Volume 19, Number 1, Winter 2000, pp. 4-21.

Hayes-Roth, B., "A Blackboard Architecture for Control," *Artificial Intelligence*, Volume 26, Number 3, 1985, pp. 251-321.

Hayes-Roth, B., "An Architecture for Adaptive Intelligent Systems," *Artificial Intelligence*, Volume 72, Number 12, 1995, pp. 329-365.

Hayes, C., Cunningham, P., and Smyth, B., "A Case-Based Reasoning View of Automated Collaborative," *Case-Based Reasoning Research and Development, LNAI 2080, 4th International Conference on Case-Based Reasoning, ICCBR 2001, Vancouver, BC, Canada, July/August 2001,* Aha, D. W. and Watson, I. (Eds.), Springer-Verlag, Berlin, 2001, pp. 234-248.

Hellström, T., Kemlin, P., and Malmquist, U., "Knowledge and Competence Management at Ericsson: Decentralization and Organization Fit," *Journal of Knowledge Management*, Volume 4, Number 2, 2000, pp. 99-110.

Hoffman, D. L., Novak, T. P., and Peralta, M., "Building Consumer Trust Online," *Communications of the ACM*, Volume 42, Number 4, 1999, pp.80-85.

Hogberg, C., "A Design for Futurizing Knowledge Networking," *Journal of Knowledge Management*, Volume 2, Number 3, December 1998, pp. 81-92.

Holmström, H., "Virtual Communities as Platforms for Product Development: An Interpretive Case Study of Customer Involvement in Online Game Development," *Twenty-Second International Conference on Information Systems*, New Orleans, LA, USA, December 16-19, 2001, pp. 299-306.

Horvath, J. A., "Working with Tacit Knowledge," *The Knowledge Management Yearbook 2000-2001*, Cortada, J. W. and Woods, J. A (Eds.), Butterworth Heinemann, Boston, MA, 2000, pp. 34-51.

Hu, Y., "Some Thoughts on Agent Trust and Delegation," *Proceedings of the Fifth International Conference on Autonomous Agents, AGENTS'01*, Montreal, Quebec, Canada, May 28-June 1, 2001, pp. 489-496.

Hyman, D., and Shingler, J., "The Hierarchy of Consumer Participation and Patterns of Economic, Social, and Political Participation," *The Journal of Consumer Affairs*, Volume 33, Number 2, Winter 1999, pp. 380-407.

Iacobucci, A., Arabie, P., and Bodapati, A., "Recommendation Agents On the Internet," *Journal of Interactive Marketing*, Volume 14, Number 3. Summer 2000, pp. 2-11.

Im, I., and Hars, A., "Finding Information Just for You: Knowledge Reuse Using Collaborative Filtering Systems," *Twenty-Second International Conference on Information Systems*, New Orleans, LA, USA, 2001, pp. 349-359.

Irma, B., and Rajiv, S., "Organizational knowledge management: A contingency perspective," *Journal of Management Information Systems*, Volume 18, Number 1, Summer 2001, pp. 23-55.

Jain, A. K., Murty, M. N., and Flynn, P.J., "Data Clustering: A Review," *ACM Computing Surveys*, Volume 31, Number 3, September 1999, pp. 264-323.

Jennings, N.R., and Wooldridge, M., *Agent Technology: Foundations, Applications and Markets*, Springer-Verlag, Berlin, 1998.

Johnson, C. M., "A survey of current research on online communities of practice," *The Internet and Higher Education*, Volume 4, Issue 1, 1st Quarter 2001, pp. 45-60.

Johnson, P., Heimann, V., and O'Neill, K., "The Wonderland of Virtual Teams," *Journal of Workplace Learning*, Volume 13. Number 1, 2001, pp. 24-29.

Johnson, R. E., "The Role of Cluster Analysis in Assessing Comparability under the U.S. Transfer Pricing Regulations," *Business Economics*, Volume 36, Number 2, April 2001, pp. 30-38.

Kang, N. and Han, S., "Agent-based e-marketplace system for more fair and efficient transaction," *Decision Support Systems*, Volume 34, Issue 2, 2003, pp.157-165.

Kankanhalli, A., Tan, B., and Wei, K., "Seeking Knowledge in Electronic Knowledge Repositories: An Exploratory Study," *Twenty-Second International Conference on Information Systems*, New Orleans, LA, USA, 2001, pp. 123-134.

Kiang, M. Y., and Kumar, A, "An Evaluation of Self-Organizing Map Networks as a Robust Alternative to Factor Analysis in Data Mining Applications," *Information*

Systems Research 2001 INFORMS, Volume 12, Number 2, June 2001, pp. 177-194.

Kim, I., Hong, J., Bae, D., Han, I., and Youn, C., "Scalable Mobile Agents Supporting Dynamic Composition of Functionality," *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems, LNAI 1887, International Workshop on Infrastructure for Scalable Multi-Agent Systems, Barcelona, Spain, June 2000, Revised Papers,* Wagner, T., and Rana, O. (Eds.), Springer-Verlag, Berlin, 2001, pp. 199-213.

Knoke, D.A., and Kuklinski, J. H., *Network Analysis*, Beverly Hills, CA: Sage, 1982.

Krishnan, R., Li, X. and Steier, D., "On Heterogeneous Database Retrieval: A Cognitively Guided Approach, " *Information Systems Research,* Volume 12, Number 3, Sept. 2001, pp. 286-301.

Kwak, M., "Web Sites Learn to Make Smarter Suggestions," *MIT Sloan Management Review*, Volume 42, Number 4, Summer 2001, pp. 17.

Lang, J.C., "Managing in Knowledge-Based Competition", *Journal of Organizational Change Management*, Volume 14, Number 6, 2001, pp. 539-553.

Lashkari, Y., Metral, M., and Maes, P., "Collaborative Interface Agents," *Proceedings of the Twelfth National Conference on Artificial Intelligence*, Vol. 1, AAAI Press, Seattle, WA, August 1994, pp. 444-449.

Lesser, E., and Prusak, L., "Communities of Practice, Social Capital, and Organizational Knowledge," *The Knowledge Management Yearbook 2000-2001*, Cortada, J. W. and Woods, J. A.(Eds.), Butterworth Heinemann, 2000, pp. 251-259.

Lesser, E. L. and Storck, J., "Communities of practice and organizational performance," *IBM Systems Journal*, Volume 40, Number 4, 2001, pp. 831-41.

Levin, N., and Zahavi, J., "Predictive Modeling Using Segmentation," *Journal of Interactive Marketing*, Volume 15, Number 2, Spring 2001, pp. 2-22.

Lewis, M., "Designing for Human-Agent Interaction," *AI Magazine*, Volume 19, Number 2, Summer 1998, pp. 67-78.

Liang, T. P., and Huang, J. S., "A Framework for Applying Intelligent Agents to Support Electronic Trading," *Decision Support Systems*, Volume 28, Issue 4, 2000, pp. 305-317.

Lieberman, H., "Autonomous Interface Agents," *Proceedings of the ACM Conference on Computers and Human Interface*, CHI-97, Atlanta, Georgia, March 1997, pp. 67-74.

Lieberman, H., and Selker, T., "Out of Context: Computer Systems that Adapt to, and Learn from, Context," *IBM Systems Journal,* Volume 39, Number 3/4, 2000, pp. 617-632.

Liebowitz, J., "Expert Systems - An Integral Part of Knowledge Management," *Kybernetes*, Volume 27, Number 2, 1998, pp. 170-175.

Lin, A., Lenert, L. A. H., Mark, A M., Kathryn, M., et al. "Clustering and the Design of Preference-Assessment Surveys in Healthcare," *Health Services Research,* Volume 34, Number 5, December 1999, pp. 1033-1045.

Lodge, G.C., and Walton, R.E., "The American Corporation and Its New Relationships," *California Management Review*, Volume 31, 1989, pp. 9-24.

Lucas, H.C., *Information Systems Concepts for Management*, New York, McGraw-Hill, 1995.

Lueg, C., "Information, Knowledge, and Networked Minds," *Journal of Knowledge Management*, Volume 5, Number 2, 2001, pp. 151-160.

Lynne, M. M., "Toward a Theory of Knowledge Reuse: Types of Knowledge Reuse Situations and Factors in Reuse Success," *Journal of Management Information Systems*, Volume 18, Number 1, Summer 2001, pp. 57-93.

Ma, M., "Agents in e-commerce," *Communications of the ACM*, Volume 42, Number 3, March 1999, pp.78-80.

Maes, P., "Agents that Reduce Work and Information Overload," *Communications of the ACM*, Volume 37, Number 7, July 1994, pp. 30-40.

Maes, P., "Smart Commerce: The Future of Intelligent Agents in Cyberspace," *Journal of Interactive Marketing*, Volume 13, Number 3, Summer 1999, pp. 66-76.

Maes, P., Guttman R.H., and Moukas A.G., "Agents that Buy and Sell: Transforming Commerce as We Know It," *Communications of the ACM*, Volume 42, Number 3, March 1999, pp. 81-87.

Malhotra, Y., " Enabling knowledge exchanges for e-business communities," *Information Strategy*, Volume 18, Number 3, Spring 2002, pp. 26-31.

Maltz, D., and Ehrlich, K., "Pointing the way: active collaborative filtering", *Conference proceedings on human factors in computing systems, 1995, Denver, Colorado,* ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1995, pp. 202-209.

Mántaras, R L., and Arcos, J.L., "The Synthesis of Expressive Music: A Challenging CBR Application," *Case-Based Reasoning Research and Development, LNAI 2080, 4th International Conference on Case-Based Reasoning, ICCBR 2001, Vancouver, BC, Canada, July/August 2001,* Aha, D. W., and Watson, I. (Eds.), Springer-Verlag, Berlin, 2001, pp. 16-26.

Marakas, G. M., *Decision Support Systes In the 21st Century*, 2nd Edition, Upper Saddle River, NJ, Prentice Hall, 2003.

March, K., "A Blueprint for Consumer-Driven Marketing," *Medical Marketing & Media,* Volume 32, Number 3, March 1997, pp. 78-84.

March, S., Hevner, A., and Ram, S., "Research Commentary: An Agenda for Information Technology Research in Heterogeneous and Distributed Environments," *Information Systems Research, 2000 INFORMS,* Volume 11, Number 4, December 2000, pp. 327-341.

Mařík, V., Pěchouček, M., Lažanský, J. and Roche, C., "PVS'98 Agents: Structures, Models and Production Planning Application," *Robotics and Autonomous Systems*, Volume 27, 1999, pp 29-43.

McCampbell, A. S., Clare, L. N., and Gitters, S. H., "Knowledge Management: the New Challenge for the 21st Century," *The Journal of Knowledge Management*, Volume 3, Number 3, 1999, pp. 172-179.

Mertins, K., Heisig, P., and Vorbeck, J., *Knowledge Management-Best Practices in Europe*, Springer-Verlag, Berlin, 2001.

Michelle, M., "Creating a Community," *Association Management,* Volume 53, Number 8, August 2001, pp. 82-83.

Millen, D. R., Fontaine M. A., and Muller M. J., "Understanding the Benefit and Costs of Communities of Practice," *Communications of the ACM*, Volume 45, Number 4, April 2002, pp. 69-73.

Minsky, M. L., *Society of Mind,* Touchstone Books, March 1988.

Modha, D. S. and Spangler, W. S., "Feature Weighting in k-Means Clustering," *Machine Learning*, 2002. Forthcoming.

Montazemi, A.R., and Gupta, K.M., "An Adaptive Agent for Case Description in Diagnostic CBR Systems," *Journal of Computers in Industry*, Volume 29, 1996, pp. 209-224.

Montazemi, A.R. and Gupta, K.M., "An Adaptive Agent for Case Description in Diagnostic CBR Systems," *Journal of Computers in Industry*, Volume 29, 1996, pp. 209-224.

Montazemi, A.R., and Gupta, K.M., "On the Effectiveness of Cognitive Feedback from an Interface Agent," *OMEGA, International Journal of Management Sciences*, Volume 25, Number 6, 1997, pp. 643-658.

Morgan, A. P., Cafeo, J. A., Gibbons, D. I., Lesperance, R. M., Sengir, G. H., and Simon, A. M., "CBR for Dimensional Management in a Manufacturing Plant," *Case-Based Reasoning Research and Development, LNAI 2080, 4th International Conference on Case-Based Reasoning, ICCBR 2001, Vancouver, BC, Canada, July/August 2001,* Aha, D. W., and Watson, I. (Eds.), Springer-Verlag, Berlin, 2001, pp. 597-610.

Morris, S. A., Marshall, T. E., and Rainer, R. K. J., "Impact of user satisfaction and trust on virtual team members," *Information Resources Management Journal*, Volume 15, Number 2, 2002, pp. 22-30.

Mueller, H. J., "Negotiation Principles," *in Foundations of Distributed Artificial Intelligence,* O'Hare, G. M. P. and Jennings, N. R. (Eds.), John Wiley & Sons, Inc, 1996.

Müller, J. P., "Control Architectures for Autonomous and Interacting agents: A Survey," *Intelligent Agent Systems: Theoretical and Practical Issues: Based on a Workshop Held at Pricai'96, Cairns, Australia, August 26-30, 1996, Lecture Notes in Artificial Intelligence 1209*, August, 1996, Cavedon, L., Rao, A., and Wobcke, W. (Eds.), Springer-Verlag, Berlin, April 1997, pp. 1-26.

Murray, J. A., O'Driscoll, A., and Torres, A., "Discovering diversity in marketing practice," *European Journal of Marketing*, Volume 36, Number 3, 2002, pp. 373-390.

MUSCLE FISH, *www.musclefish.com.*

Nardi, B. A, Miller, J. R, and Wright, D. J, "Collaborative, Programmable Intelligent Agents," *Communications of the ACM,* Volume 41, Number 3, March 1998, pp. 96-104.

Nelson, K. M., and Cooprider, J. G., "The contribution of shared knowledge to IS group performance," *MIS Quarterly*, Volume 20, Number 4, 1996, pp. 409-432.

Newell, A., "Putting It All Together," *Complex Information Processing: The Impact of Herbert Simon,* Klahr, D. & Kotovsky, K. (Eds.), Hillsdale, NJ: Lawrence Erlbaum, 1988.

Nodine, M., Chandrasekara, D., and Unruh, A., "Task Coordination Paradigms for Information Agents," *Intelligent Agents VII, Agent Theories Architectures and Languages, 7th International Workshop, ATAL 2000, Boston, MA, USA, July 2000 Proceedings, LNAI 1986*, Castelfranchi, C. and Lespérance, Y. (Eds.), Springer-Verlag, Berlin, 2001, pp. 167-181.

Norman, T. J., and Reed, C., "Delegation and Responsibility," *Intelligent agents VII, Agent Theories Architectures and Languages, 7th International Workshop, ATAL 2000, Boston, MA, USA, July 2000 Proceedings, LNAI 1986,* Castelfranchi, C. and Lespérance, Y. (Eds.), Springer-Verlag, Berlin, 2001, pp. 136-149.

Notess, G. R., "Communications and community on Web sites," *Online,* Volume 23, Number 4, July/August 1999, pp. 65-68.

Nwana, H. S., "Software Agents: An Overview," *Knowledge Engineering Review*, Volume 11, Number 2, 1996, pp. 1-40.

O'Dell, C., Grayson, C. J., and Essaides, N., *If Only We Knew What We Know: The Transfer of Internal Knowledge and Best Practice*, New York, The Free Press, 1998.

O Donnell, D., and O Regan, P. C. B., "Intellectual capital: a Habermasian introduction", *Journal of Intellectual Capital*, Volume 1, Number 2, 2000, pp. 187-200.

Ossowski, S., *Co-Ordination in Artificial Agent Societies: Social Structures and Its Implications for Autonomous Problem-Solving Agents, Lecture Notes in Artificial Intelligence 1535*, Springer-Verlag, Berlin, 1999.

Pemberton, J. D., and Stonehouse G. H., "Organizational Learning and Knowledge Assets- an essential partnership," *The Learning Organization*, Volume 7, Number 4, 2000, pp. 184-194.

Pan, S. L., and Scarbrough, H., "A Socio-Technical View of Knowledge-Sharing at Buckman Laboratories," *Journal of Knowledge Management*, Volume 2, Number 1, September 1998, pp. 55-66.

Pauleen, D. F., and Yoong, P., "Facilitating Virtual Team Relationships via Internet and Conventional Communication Channels," *Internet Research: Electronic Networking Applications and Policy*, Volume 11, Number 3, 2001, pp. 190-202.

PenÄa-Mora, F., Hussein, K., Vadhavkar, S. and Benjamin, K., "CAIRO: a concurrent engineering meeting environment for virtual design teams," *Artificial Intelligence in Engineering* , Volume 14, Issue 3, July 2000, pp. 203-219.

Pilkington, R.M., *Intelligent Help: Communicating with Knowledge-Based Systems*, Oxford University Press, Oxford, U.K., 1992.

Plaza, E., Luis, J., and Martin, F., "Cooperative Case-Base Learning," *Distributed Artificial Intelligence Meets Machine Learning: Learning in Multi-Agent Environments: Ecai'96 Workshop Ldais Budapest, Hungary, August 1,* Weiss, G. (Ed.), *Lecture Notes in Artificial Intelligence*, Springer-Verlag, Berlin, 1997.

Pouchard, L., and Walker D. W., "A Community of Agents for User Support in a Problem-Solving Environment," *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems, LNAI 1887, International Workshop on Infrastructure for Scalable Multi-Agent Systems, Barcelona, Spain, June 2000, Revised Papers,* Wagner, T., and Rana, O. (Eds.), Springer-Verlag, Berlin, 2001, pp. 192-198.

Powell, W.W., "Learning from Collaboration: Knowledge and Networks in the Biotechnology and Pharmaceutical Industries," *California Management Review,* Volume 40, Number 3, 1998, pp. 228-240.

Prahalad, C.K., and Hamel, G., "The Core Competence of the Corporation," *Harvard Business Review*, May-June 1990, pp 79-91.

Preece, J., "Supporting community and building social capital," *Communications of the ACM*, Volume 45, Number 4, April 2002, pp. 51-55.

Ram, S., "Intelligent agents and the World Wide Web: Fact or fiction?" *Journal of Database Management,* Volume 12, Number 1, January-March 2001, pp. 46-47.

Regan, E. A., and O'Connor, B. N., *End-User Information Systems: Implementing Individual and Work Group Technologies*, 2nd ed., Prentice Hall, New Jersey, 2002.

Reticular, *User Guide*, Version 1.1, Rev. 4, www.reticular.com, 1999.

Rhodes, B J., and Maes, P., "Just-in-time information retrieval agents," *IBM Systems Journal,* Volume 39, Number ¾, 2000, pp. 685-704.

Ricci, A., Denti, E., and Omicini, A., "Agent Coordination Infrastructures for Virtual Enterprises and Workflow Management," *Cooperative Information Agents V, LNAI 2182, 5th International Workshop, CIA 2001 Modena, Italy, September 2001 Proceedings, Klusch, M., and Zambonelli, F. (Eds.),* Springer-Verlag, Berlin, 2001, pp. 235-246.

Ridings, C. M., Gefen, D., and Arinze, B., "Some antecedents and effects of trust in virtual communities," *Journal of Strategic Information Systems*, Volume 11, *Issues 3-4,* 2002, pp. 271-295.

Roxanne, H., S., and Turoff, M., "What Makes Learning Networks Effective?" *Communications of the ACM*, Volume 45, Number 4, April 2002, pp. 56-59.

Ruggles, R., "The State of the Notion: Knowledge Management in Practice," *California Management Review*, Volume 40, Number 3, 1998, pp. 80-89.

Sandholm, T., "Automated Negotiation", *Communications of the ACM*, Volume 42, Number 3, March 1999, pp. 84-85.

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J., "Item-Based Collaborative Filtering Recommendation Algorithms," *Proceedings of the Tenth International World Wide Web Conference*, WWW 10, Hong Kong, P.R. China, May 1-5, ACM, 2001, pp. 285-295.

Sauter, V., *Decision Support Systems*, New York, John Wiley & Sons, 1999.

Scheirer, E., "Music-Listening Systems," *PhD dissertation*, Massachusetts Institute of Technology, 2000.

Schumacher, M., *Objective Coordination in Multi-Agent System Engineering: Design and Implementation, Lecture Notes in Artificial Intelligence 2039*, Springer-Verlag, 2001.

Serugendo, G. D. M., "A Formal Development and Validation Methodology Applied to Agent-Based Systems," *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems, LNAI 1887, International Workshop on Infrastructure for Scalable Multi-Agent Systems, Barcelona, Spain, June 2000, Revised Papers,* Wagner, T., and Rana, O. (Eds.), Springer-Verlag, Berlin, 2001, pp. 214-225.

Seufert, A., Krogh, G. V., and Bach, A., "Toward Knowledge Networking," *Journal of Knowledge Management*, Volume 3, Number 3, 1999, pp. 180-190.

Shani, A. B., Sena J. A., and Stebbins M. W., "Knowledge Work Teams and Groupware Technology: Learning from Seagate's Experience," *Journal of Knowledge Management*, Volume 4, Number 2, 2000, pp. 111-124.

Shaw, N. G, Mian, A., and Yadav, S. B, "A Comprehensive Agent-Based Architecture for Intelligent Information Retrieval in a Distributed Heterogeneous Environment," *Decision Support Systems,* Volume 32, Issue 4, March 2002, pp. 401-415.

Sikora, R., and Shaw, M. J, "A Multi-Agent Framework for the Coordination and Integration of Information Systems," *Management Science,* Volume 44, Number 11 (Part 2), November 1998, pp. S65-S78.

Simon, H., *The New Science of Management Decision*, Englewood Cliffs, NJ, Prentice Hall, 1977.

Simon, H.A., and Hayes, J.R., "Understanding complex task instructions," *Cognition and Instruction (Chap. 14),* Klahr, D. (Ed.), Hillsdale, NJ, Erlbaum, 1976.

Smith, M., "Tools for Navigating Large Social Cyberspace," *Communications of the ACM*, Volume 45, Number 4, April 2002, pp. 51-55.

Stoddart, L., "Managing Intranets to Encourage Knowledge Sharing: Opportunities and Constraints," *Online Information Review*, Volume 25, Number 1, 2001, pp. 19-28.

Stone, P., and Veloso, M., *RoboCup-97: Robot Soccer World Cup I, (Lecture Notes in Computer Science, 1395)*, Kitano, H., Siekmann, J., and Carbonell, J.G. (Eds.), Springer-Verlag, Berlin, 1998.

Sutton, R.S., and Barto, A.G., *Reinforcement Learning: An Introduction*, The MIT Press, Cambridge, MA, 1998.

Swan, J., Newell, S., Scarbrough, H., and Hislop, D., "Knowledge Management and Innovation: Networks and Networking," *Journal of Knowledge Management*, Volume 3, Number 4, 1999, pp. 262-275.

Sycara, K., "Multiagent Systems," *AI Magazine*, Volume 19, Number 2, 1998, pp. 79-92.

Sycara, K., Decker, K., and Williamson, M., "Middle-Agents for the Internet," *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence, IJCAI 97*, Nagoya, Japan, August 23-29, Volume 1, Morgan Kaufmann, 1997, pp. 578-583.

Sycara, K., Decker, K., Pannu, A., Williamson, M., and Zeng, D., "Distributed Intelligent Agents," *IEEE Expert*, December 1996, pp. 36-46.

Syed J. R., "An Adaptive Framework for Knowledge Work," *The Journal of Knowledge Management*, Volume 2, Number 2, 1998, pp. 59-69.

Tambe, M., "Toward Flexible Teamwork," *Journal of Artificial Intelligence Research,* Volume 7, Number 1, 1997, pp. 83-124.

Tu, H. C., and Hsiang, J., "An Architecture and Category Knowledge for Intelligent Information Retrieval Agents," *Decision Support Systems*, Volume 28, Issue 3, May 2000, pp. 255-268.

Tung, L. L., Tan, P. L. J., Chia, P.J.T. and Koh, Y. L., "An Empirical Investigation of Virtual Communities and Trust," *Twenty-Second International Conference on Information Systems*, New Orleans, LA, USA, 2001, pp. 307-320.

Turban, E., and Aronson, J.E., *Decision Support Systems and Intelligent Systems*, 5th ed. Prentice-Hall, NJ, 1998.

Turner, P. J., and Jennings, N. R., "Improving the Scalability of Multi-Agent Systems," *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems, LNAI 1887, International Workshop on Infrastructure for Scalable Multi-Agent Systems, Barcelona, Spain, June 2000, Revised Papers,* Wagner, T., and Rana, O. (Eds.), Springer-Verlag, Berlin, 2001, pp. 246-262.

Turowski, K., "Agent-Based e-Commerce in Case of Mass Customization," *International Journal of Production Economics,* Volume 75, Number 1-2, January 10, 2002, pp. 69-81.

Tversky, A., and Kahneman, D., *Judgment under Uncertainty: Heuristics and Biases,* Kahneman, D., Slovic, P., and Tversky, A. (Eds.), Cambridge University Press, Cambridge University, 1982.

Von Krogh, G., Nonaka, I., and Ichijo, K., "Develop Knowledge Activists!" *European Management Journal,* Volume 15, Number 5, 1997, pp. 475-483.

Walsh, G., Henning-Thurau, T., Wayne-Mitchell, V., Wiedmann, K., "Consumers' decision-making style as a basis for market segmentation," *Journal of Targeting, Measurement & Analysis for Marketing,* Volume 10, Number 2, Dec 2001, pp. 117-131.

Watson, I., *Applying Case-Based Reasoning: Techniques for Enterprise Systems*, Morgan Kaufmann Publishers, Inc., San Francisco, CA, 1997.

Wenger, E., *Communities of Practice: Learning, Meaning, and Identity*, Cambridge University Press, Cambridge, 1998.

Wilikens, M., Morris, P., and Masera, M., "Trust requirements in e-business," *Communications of the ACM*, Volume 43, Number 12, 2000, pp. 81-87.

Wold, E., Blum, T., Keislar, D., Wheaton, J., "Content-Based Classification, Search, and Retrieval of Audio," *IEEE MultiMedia*, Volume 3, Number 3, FALL 1996, pp. 27-36.

Wooldridge, M., "Verifying that Agents Implement a Communication Language," *AAAI-99 Sixth National Conference on Artificial Intelligence*, Orlando, Florida, July 18-22, 1999, pp. 52-57.

Youll, J., "Agent-Based Electronic Commerce: Opportunities and Challenges," *5th International Symposium on Autonomous Decentralized System with an Emphasis on Electronic Commerce*, March 26-28, 2001, Dallas, Texas, USA, pp. 143.

Zack, M., "Managing codified knowledge," *Sloan Management Review*, Summer 1999, pp. 45-58.

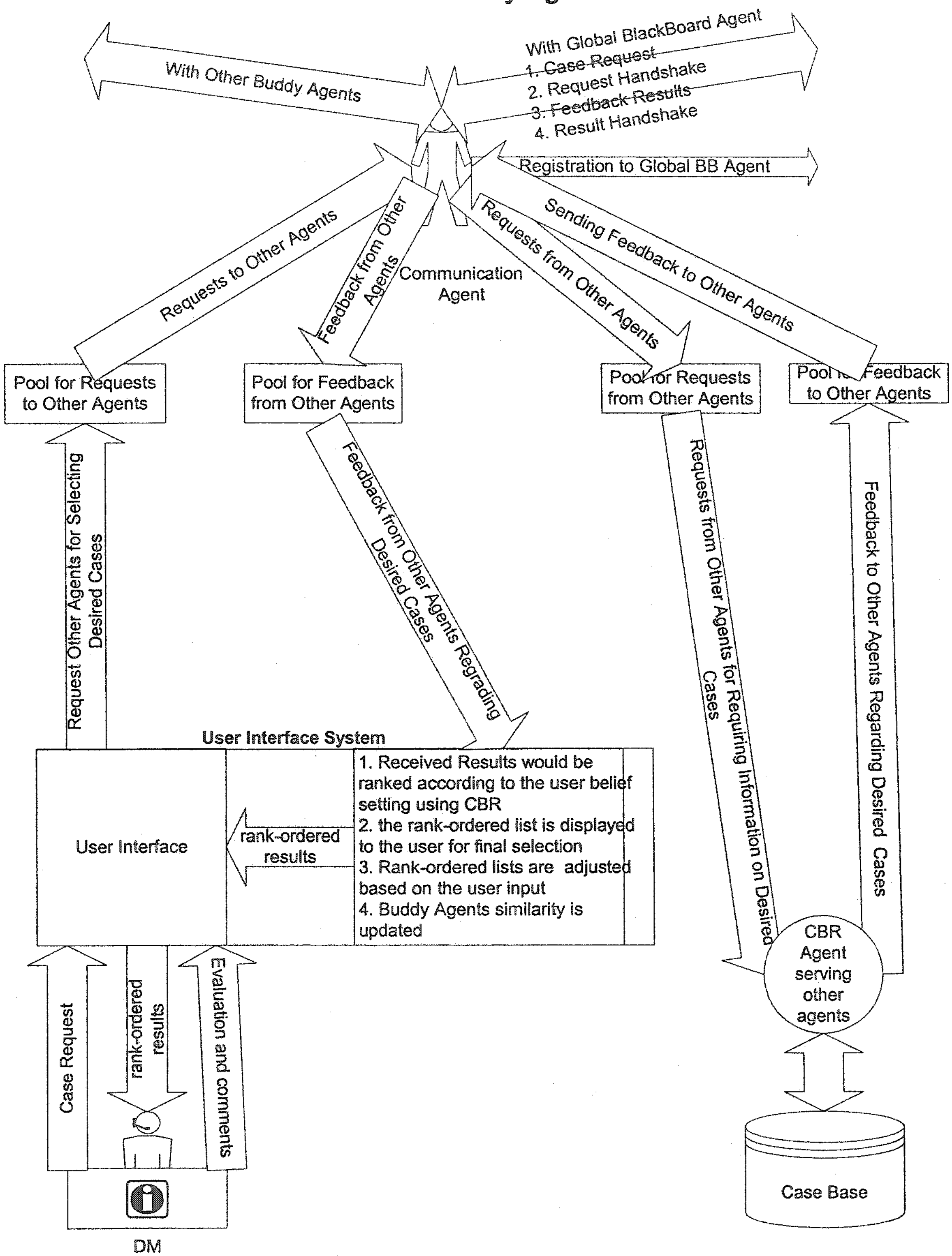
Zetlin, M., "Putting communities to work," *Computerworld,* Volume 36, Number 23, June 3, 2002, pp. 40.

Zhang, X.Q., Lesser, V., and Podorozhny, P., "New Results on Cooperative, MultiStep Negotiation Over a Multi-Dimensional Utility Function," *AAAI Fall 2001 Symposium on "Negotiation Methods for Autonomous,"* AAAI Press, January 2001, pp. 1-10.

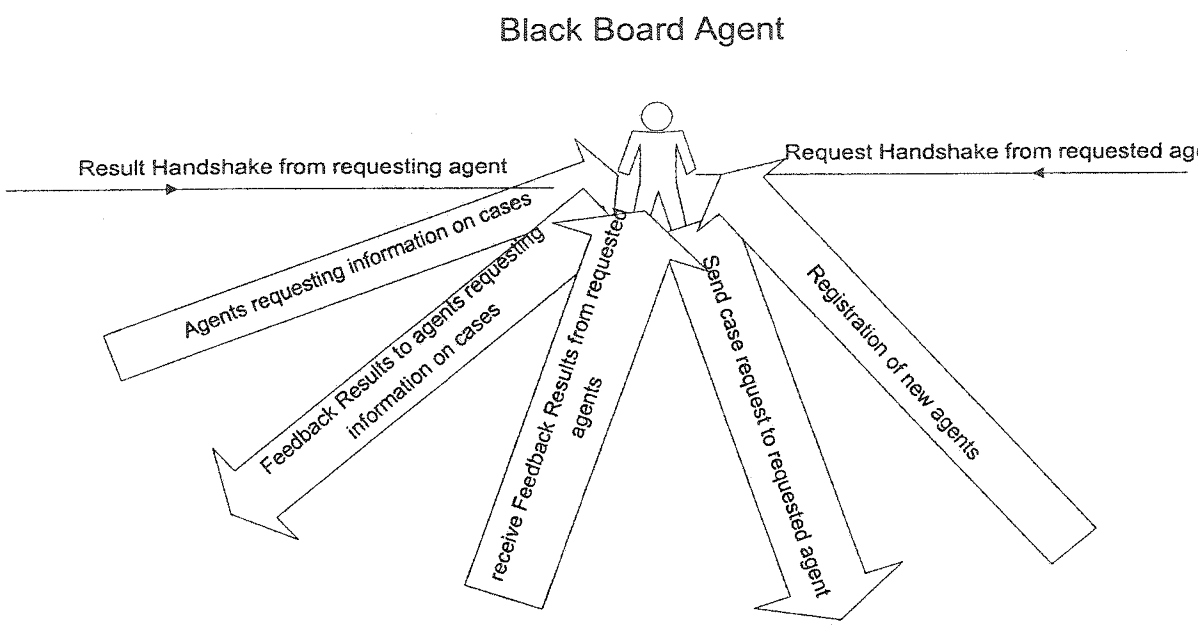
Zimmermann, H.J., *Fuzzy sets, decision making, and expert systems,* Boston, Kluwer Academic Publishers, c1987.

APPENDIX 1: Detailed Flowchart of MAS

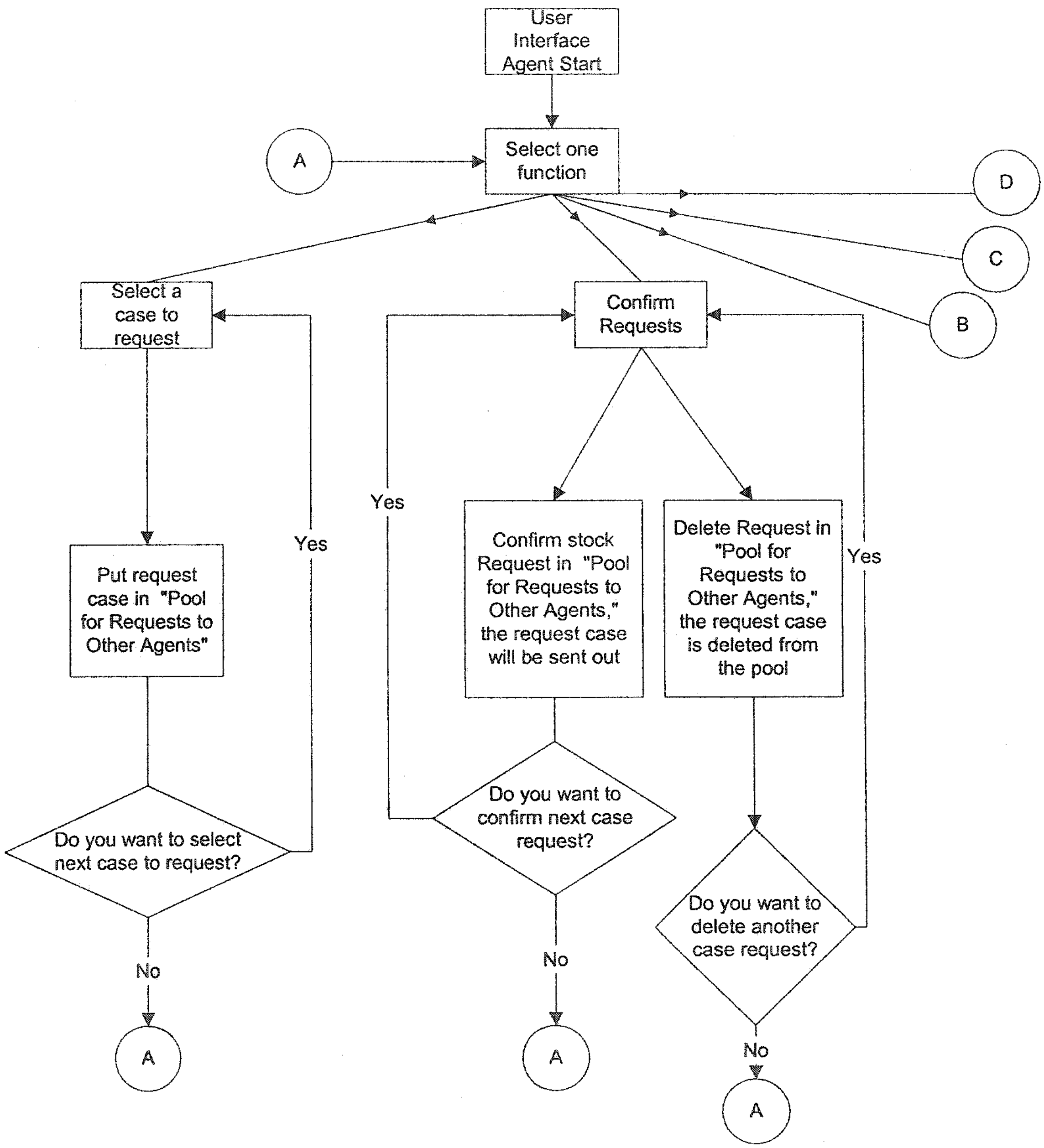
Architecture of Buddy Agent



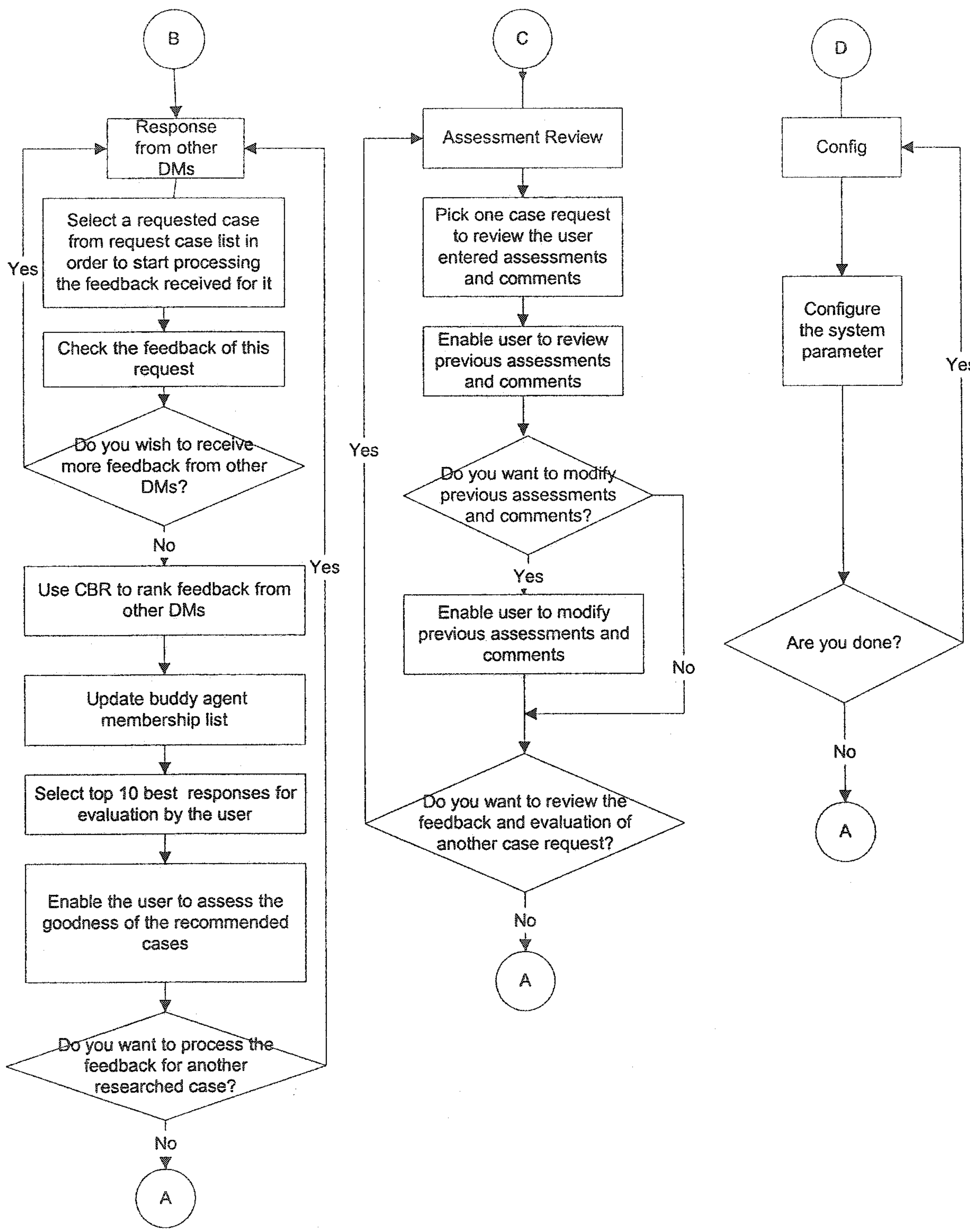
System Architecture of BlackBoard Agent



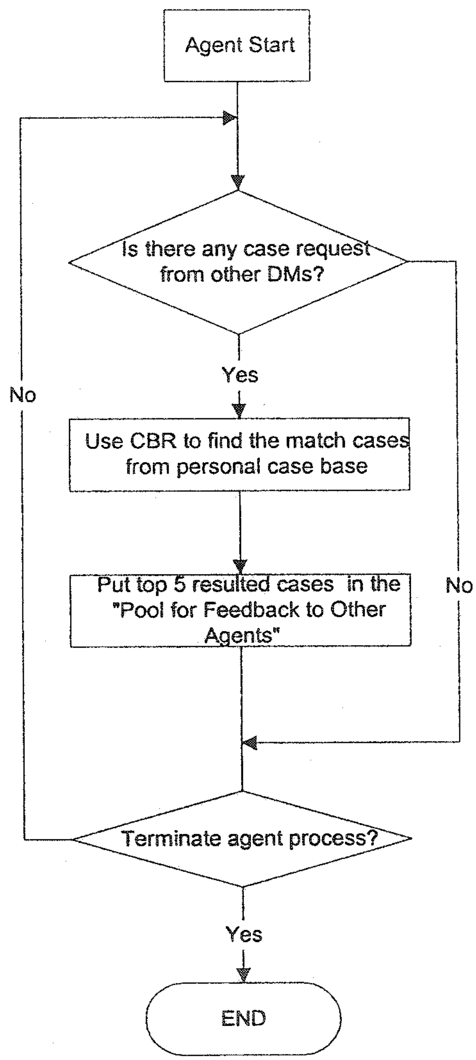
Buddy Agent -- Flowchart of user interface system(1)



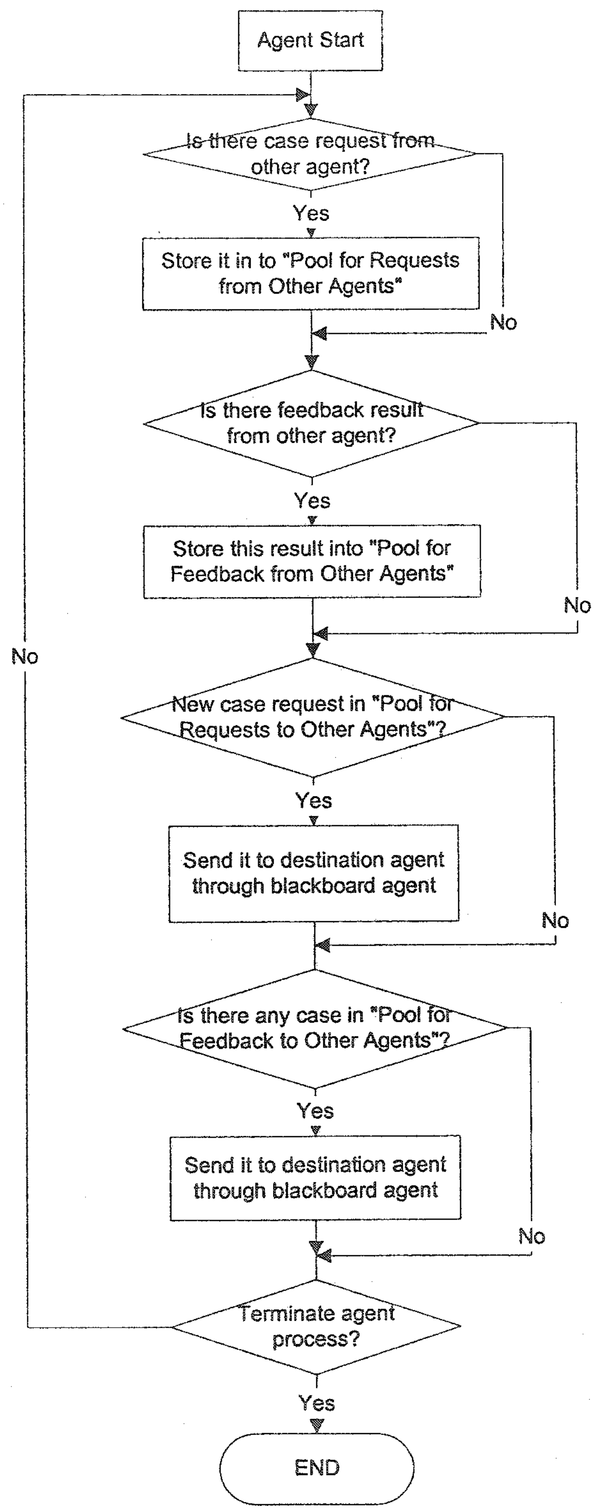
Buddy Agent --- Flowchart of user interface system (2)



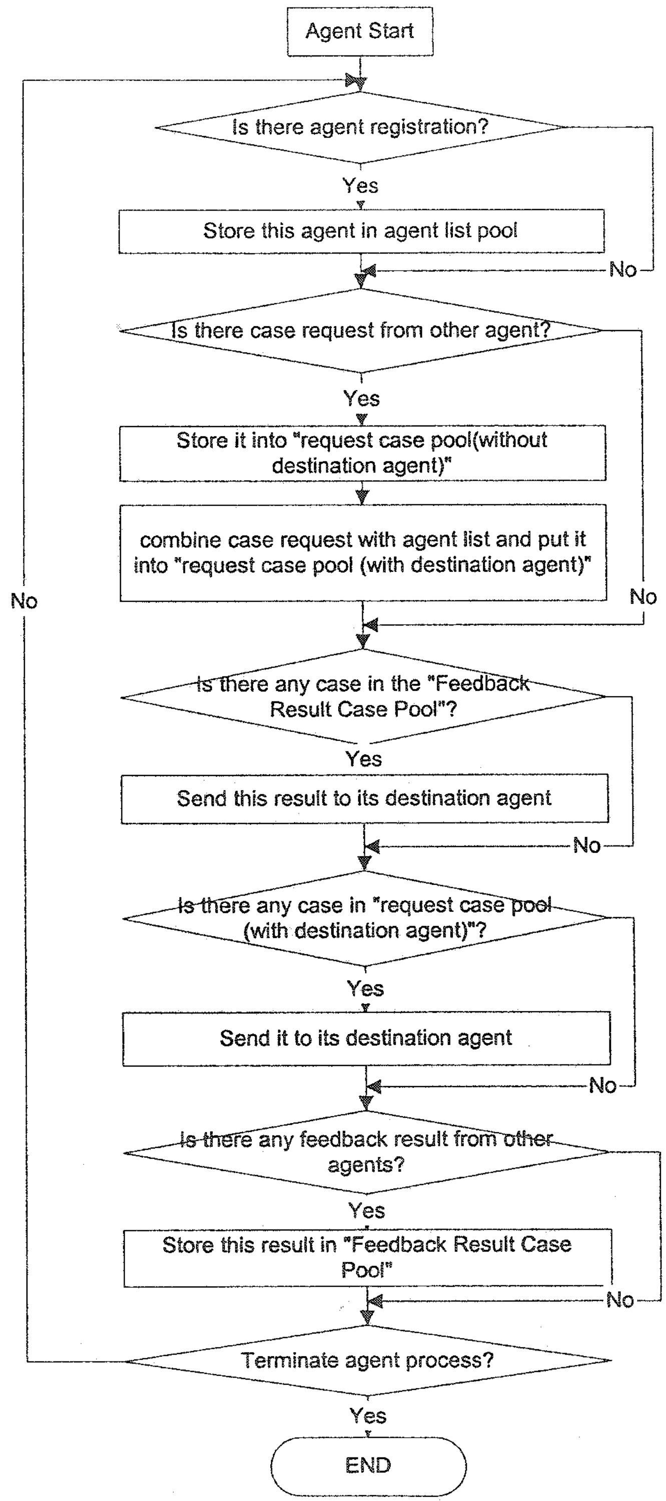
Buddy Agent --- Flowchart of the CBR agent serving other agents



Buddy Agent --- Flowchart of communication agent



Flowchart of blackboard agent



Appendix 2: AgentBuilder Pro- An Integrated Toolkit for Constructing Intelligent Software Agents

AgentBuilder (www.agentbuilder.com) is an integrated tool suite for constructing intelligent software agents. Agents constructed using AgentBuilder communicate using the Knowledge Query and Manipulation Language (KQML) and support the performatives defined for KQML. In addition, AgentBuilder allows the developer to define new interagent communications commands that suit his particular needs. AgentBuilder is currently available in two different versions to meet a wide variety of developer needs: AgentBuilder Lite and AgentBuilder Pro. AgentBuilder Lite is an entry-level product for agent software developers. AgentBuilder Pro is designed for serious multi-agent development. It builds upon AgentBuilder Lite with additional powerful tools for developing multi-agent systems. Since AgentBuilder Pro was used in the development of our multi-agent system, the following describes the development tools and run-time of AgentBuilder pro.

AgentBuilder consists of two major components - the Toolkit and the Run-Time System. The AgentBuilder Toolkit includes tools for managing the agent-based software development process, analyzing the domain of agent operations, designing and developing networks of communicating agents, defining behaviors of individual agents, and debugging and testing agent software. The Run-Time System includes an agent engine that provides an environment for execution of agent software.

AgentBuilder Pro Development Tools

AgentBuilder Pro provides graphical tools for supporting all phases of the agent construction process. Programming software agents (sometimes called Agent-Oriented Programming) is accomplished by specifying intuitive concepts such as the beliefs, commitments, behavioral rules and actions of the agent. AgentBuilder Pro makes it much easier to create, debug and test multi-agent systems. The software developer need only model the communication dialogs between agents using the protocol tools and AgentBuilder will automatically construct the required behavioral rules for implementing these conversations.

Run-Time System

The Run-Time System consists of the agent program created using the AgentBuilder Pro graphical tools and the Run-Time Agent engine. The agent program includes the agent definition file and PAC libraries. The agent program is executed by the Run-Time Agent engine; the combination of the agent program and the agent engine create an executable agent.

System Requirements

AgentBuilder Pro is coded in Java and produces Java-based agents. AgentBuilder is distributed with the JRE (Java Runtime Environment) for each supported platform. Both the AgentBuilder Toolkit and the Run-Time System execute on the Java Virtual Machine included with the JRE.

AgentBuilder Pro distributions are available for the following platforms: Solaris, Windows 95/98/2000/XP, Windows NT, Linux and IRIX.

Appendix 3

Table 1: Commonality between Subjects and the Subject Found Buddies

Subject	Buddy	Common Genre (+: has common Genre; -: no common Genre)	Other common attributes
S1	S4	+	
S1	S5	+	
S1	S11	+	
S1	S14	+	
S1	S26	+	
S2	S1	-	mood: upbeat; decade: current; tempo: medium; beat: medium; vocal style: neutral; lead vocal: male, female
S2	S5	+	
S2	S25	+	
S2	S26	+	
S2	S28	+	
S4	S1	+	
S4	S12	+	
S4	S15	+	
S4	S26	+	
S4	S32	+	
S6	S9	+	
S6	S13	-	mood: romantic, sentimental, aggressive; decade: current; tempo: fast; beat: medium, heavy; popularity: toppicks; vocal style: smooth lead vocal: male, female
S6	S21	-	mood: romantic; decade: 90; tempo: medium, fast; beat: medium, heavy; popularity: toppicks; vocal style: smooth, raspy
S6	S31	+	
S6	S35	+	
S7	S2	-	mood: upbeat, mellow, sad; decade: 60,70, current; tempo: medium; popularity: popular, known; vocal style: smooth, neutral; lead vocal: male, female
S7	S5	+	
S7	S12	+	
S7	S23	-	Mood: upbeat, mellow, sad, aggressive; decade: 80, 90, current; tempo: veryslow, medium; beat: heavy; popularity: toppicks, popular, wellknown; vocal style: smooth; lead vocal: male, female
S7	S34	+	
S8	S9	+	
S8	S21	+	

S8	S27	+	
S8	S31	+	
S8	S35	+	
S9	S2	-	mood: upbeat; decade: current; tempo: medium; beat: medium; popularity: toppicks; vocal style: neutral; lead vocal: male, female
S9	S14	+	
S9	S23	+	
S9	S25	+	
S9	S31	+	
S10	S1	+	
S10	S2	+	
S10	S4	+	
S10	S14	+	
S10	S25	+	
S11	S1	+	
S11	S5	+	
S11	S6	-	mood: aggressive; decade: 90; tempo: medium, fast; beat: medium, heavy; popularity: toppicks; vocal style: raspy; lead vocal: male
S11	S14	+	
S11	S26	+	
S12	S5	+	
S12	S9	+	
S12	S11	+	
S12	S15	+	
S12	S35	+	
S13	S5	+	
S13	S6	-	mood: romantic, sentimental, aggressive; decade: current; tempo: fast; beat: medium, heavy; popularity: toppicks; vocal style: smooth; lead vocal: male, female
S13	S11	+	
S13	S31	+	
S13	S35	+	
S14	S1	+	
S14	S8	+	
S14	S11	+	
S14	S28	+	
S14	S35	+	
S15	S2	+	
S15	S4	+	
S15	S12	+	
S15	S14	+	
S15	S30	+	
S16	S1	-	mood: upbeat; beat: medium; popularity: toppicks; vocal style: neutral; lead vocal: male, female
S16	S11	+	

S16	S15	+	
S16	S21	-	decade: 80,90; tempo: fast; beat: medium, heavy; popularity: toppicks; vocal style: neutral; lead vocal: male, female
S16	S37	-	decade: 80,90; tempo: fast; beat: medium; popularity: toppicks; vocal style: neutral; lead vocal: male, female
S17	S2	+	
S17	S5	+	
S17	S10	+	
S17	S22	+	
S17	S28	+	
S18	S10	+	
S18	S15	+	
S18	S22	+	
S18	S23	+	
S18	S26	+	
S19	S2	-	mood: romantic; decade:90, current; tempo: slow, medium, beat: medium; popularity: popular, wellknown; vocal style: neutral; lead vocal: male
S19	S4	+	
S19	S10	+	
S19	S16	+	
S19	S25	+	
S20	S6	+	
S20	S9	+	
S20	S11	-	mood: upbeat; decade: 70,80,90; tempo: medium, fast; beat: medium, heavy; popularity: toppicks, popular; vocal style: raspy; lead vocal: male
S20	S21	-	Mood: upbeat, happy, romantic; decade:70, 80, 90; tempo: slow; medium, fast; beat: light, medium, heavy; popularity: toppicks, popular, wellknown; vocal style: smooth, raspy; lead vocal: male, female
S20	S31	+	
S21	S2	+	
S21	S4	+	
S21	S5	+	
S21	S12	+	
S21	S17	+	
S22	S5	+	
S22	S13	+	
S22	S17	+	
S22	S28	+	
S22	S36	+	
S23	S1	-	mood: upbeat; decade: current; tempo: medium; beat: medium; popularity: toppicks; lead vocal: male, female
S23	S18	+	
S23	S22	+	
S23	S24	+	

S23	S33	+	
S24	S2	+	
S24	S4	+	
S24	S17	+	
S24	S26	+	
S24	S31	+	
S25	S1	+	
S25	S2	+	
S25	S4	+	
S25	S11	+	
S25	S26	+	
S26	S1	+	
S26	S4	+	
S26	S25	+	
S26	S28	+	
S26	S36	+	
S27	S2	+	
S27	S9	+	
S27	S13	+	
S27	S34	+	
S27	S37	+	
S28	S4	+	
S28	S9	+	
S28	S10	+	
S28	S11	+	
S28	S23	+	
S30	S2	+	
S30	S4	+	
S30	S12	+	
S30	S15	+	
S30	S24	+	
S31	S9	+	
S31	S24	+	
S31	S27	+	
S31	S30	+	
S31	S37	+	
S32	S5	+	
S32	S8	+	
S32	S9	+	
S32	S11	+	
S32	S12	+	
S33	S2	+	
S33	S5	+	
S33	S13	+	
S33	S21	+	
S33	S28	+	

S34	S2	+	
S34	S6	-	mood: sentimental; decade: current; tempo: medium, fast; beat: medium, heavy; popularity: toppicks; vocal style: smooth, raspy; lead vocal :male, female
S34	S7	+	
S34	S12	+	
S34	S16	+	
S35	S5	+	
S35	S6	+	
S35	S8	+	
S35	S14	+	
S35	S37	+	
S36	S17	-	mood:upbeat, romantic, mellow; decade:current; tempo:medium, fast, veryfast; beat:light, medium, heavy; popularity:toppicks, popular; vocal_style:smooth, neutral, raspy;lead_vocal:male, female
S36	S20	+	
S36	S22	+	
S36	S30	+	
S36	S31	+	
S37	S8	+	
S37	S9	+	
S37	S14	+	
S37	S27	+	
S37	S34	+	

Appendix 4: Subjects' Preferences to Agent Found Buddies and Subject Found Buddies

The following table displays the subjects' evaluations of the agent found buddies and subject found buddies for each subject. The values in the upper corner of each cell are the subjects' evaluation of Agent Found Buddies; the values in the lower corner of each cell are the subjects' evaluation of Subject Found Buddies. The zero in the upper corner of each cell means that this subject is not an agent found buddy of the subject listed at the top. The zero in the lower corner of each cell means that this subject is not a subject found buddy of the subject listed at the top. Subjects S3, S5, S29 did not complete the buddy selection; therefore, there is no subject evaluation of their agent found buddies and/or subject found buddies.

Subjects

BUDDIES	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
S1		0 / 7		0 / 9						0 / 9
S2				6 / 0			7 / 7		8 / 8	0 / 5
S3		8						8 / 0		
S4	9 / 9	6					6 / 0	9 / 0	6 / 0	0 / 8
S5	0 / 5	0 / 6					0 / 8			
S6										
S7										
S8		3 / 0				6 / 0				2 / 0
S9						0 / 6		0 / 1		
S10										
S11	0 / 3						5 / 0			6 / 0
S12				0 / 8		8 / 0	0 / 7			
S13						0 / 7				
S14	0 / 2			7 / 0			6 / 0	2 / 0	9 / 9	0 / 6
S15				7 / 7						
S16										
S17	9 / 0									
S18										1 / 0
S19									7 / 0	
S20		3 / 0					5 / 0			
S21	9 / 0					0 / 7		0 / 4		
S22										
S23		3 / 0				6 / 0	0 / 8	8 / 0	0 / 6	
S24						4 / 0				
S25	9 / 0	0 / 7							0 / 6	0 / 8
S26	0 / 7	0 / 7		6 / 6				8 / 0		
S27								0 / 2		
S28		0 / 8								
S29				7 / 0						2 / 0
S30									8 / 0	4 / 0
S31	2 / 0					0 / 5		0 / 9	0 / 7	
S32				0 / 7						
S33										
S34							0 / 9			
S35						0 / 8		0 / 7		
S36						5 / 0				
S37										

Subjects

BUDDIES	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
S1	0 / 8			0 / 3		0 / 9				
S2	8 / 0	5 / 0		2 / 0	6 / 6		0 / 4	6 / 0	7 / 7	5 / 0
S3				2 / 0						
S4	6				2 / 2	6 / 0		3 / 0	9 / 9	8 / 0
S5	0 / 9	0 / 6	0 / 6				0 / 7			
S6	0 / 6		0 / 8							0 / 6
S7										
S8		6 / 0		0 / 6						
S9		0 / 8		2 / 0						0 / 3
S10							0 / 7	0 / 7	0 / 8	
S11		0 / 5	0 / 8	0 / 4		9 / 9	6			0 / 2
S12			6 / 0		0 / 8					
S13						9 / 0	8			
S14	0 / 1		6 / 0		0 / 4		4			
S15		9 / 9				0 / 9		9 / 9		7 / 0
S16									0 / 7	
S17										
S18	5 / 0					8 / 0			7 / 0	
S19										
S20		7 / 0								
S21			8 / 0			0 / 9				0 / 3
S22							0 / 7	0 / 7		
S23		9 / 0		2 / 0				0 / 7		9 / 0
S24	2 / 0									
S25			8 / 0						7	
S26	0 / 5							0 / 6	8 / 0	
S27										
S28	8 / 0			0 / 6			9 / 9			
S29					9 / 0					
S30					0 / 7			7 / 0		
S31			0 / 6		7 / 0					0 / 6
S32					8 / 0			7 / 0		1 / 0
S33			7 / 0			5 / 0	7 / 0			
S34										
S35		0 / 2	0 / 7	2 / 2						
S36									7 / 0	
S37						0 / 9				

Subject

BUDDIES	S21	S22	S23	S24	S25	S26	S27	S28	S29	S30
S1			0 / 1		0 / 9	0 / 9				
S2	8 / 8		4 / 0	6 / 6	1 / 1	3 / 0	0 / 2			8 / 8
S3				6						
S4	0 / 7		2 / 0	8 / 8	0 / 8	8 / 8	9 / 0	0 / 7		0 / 8
S5	0 / 7	0 / 8								
S6										
S7										
S8						1 / 0				
S9							0 / 2	0 / 1		
S10								5		
S11	7 / 0	8 / 0			0 / 8		9 / 0	7 / 7		
S12	0 / 3		9 / 0		7 / 0		6 / 0			0 / 7
S13		5 / 5					0 / 9	4 / 0		
S14										6
S15			5 / 0		4 / 0					0 / 8
S16		8								
S17	0 / 6	0 / 3		0 / 8				6 / 0		
S18		7 / 0	0 / 7			2 / 0				
S19										
S20			8 / 0							
S21								4 / 0		
S22			0 / 2							
S23	5 / 0			6 / 0	8 / 0			1 / 1	S23	6 / 0
S24			0 / 1				8 / 0			0 / 8
S25						0 / 2				
S26	7 / 0			0 / 7	9 / 9					
S27										
S28	8 / 0	5 / 5				0 / 7				
S29										4 / 0
S30							8 / 0			
S31				0 / 6						
S32										7 / 0
S33			0 / 3							
S34							0 / 4			
S35										
S36		0 / 3		2 / 0		9 / 9				
S37							0 / 8			

Subject

BUDDIES

	S31	S32	S33	S34	S35	S36	S37
S1							
S2	0 / 7		8 / 0	7 / 7		0 / 5	
S3		6			0 / 6		
S4		0 / 2			0 / 5		0 / 8
S5		8 / 0	7 / 0		8 / 0		
S6				3 / 0	4 / 0		
S7				9 / 0			
S8		3 / 0			4 / 0		8 / 8
S9	9 / 9	3 / 3			0 / 5		7 / 0
S10	6						
S11	0 / 9	9	0 / 6	0 / 8			
S12		6 / 0		9 / 0			
S13			2 / 0				
S14					8 / 8		7 / 0
S15							
S16				7 / 7			
S17						1 / 0	0 / 7
S18	0 / 6						
S19							
S20		0 / 2		0 / 9		4 / 0	
S21			8 / 0				
S22						1 / 1	
S23			0 / 5		0 / 3	0 / 7	
S24	9 / 0						
S25						2	0 / 8
S26						0 / 7	0 / 8
S27	9 / 0						8 / 0
S28			7 / 7				
S29		0 / 9					
S30	9 / 0					6 / 0	
S31			0 / 3	0 / 6		9 / 0	
S32							
S33							
S34							8 / 0
S35							
S36							
S37	6 / 0		0 / 2		6 / 0		

Appendix 5: Subjects' Satisfaction with the Items (music) Contained in Three Groups of Buddies (E_{ia}, E_{is}, E_{isf})

Subject	Evaluations of Music Contained in Subject Found Buddies (E_{is})	Evaluations of Music Contained in Agent Found Buddies (E_{ia})	Evaluations of Music Contained in Final Subject Found Buddies (E_{isf})
S1	2	5	9
S2	7	6	7
S4	8	7	7
S6	8	5	7
S7	9	7	8
S8	7	7	2
S9	6	7	6
S10	8	6	8
S11	9	3	6
S12	7	6	8
S13	6	7	8
S14	6	3	6
S15	6	8	7
S16	9	8	7
S17	7	8	9
S18	8	6	7
S19	7	8	6
S20	4	8	7
S21	6	7	7
S22	6	7	5
S23	7	4	9
S24	6	7	5
S25	8	6	8
S26	6	3	5
S27	8	7	9
S28	5	6	4
S30	9	7	8
S31	9	7	9
S32	7	9	9
S33	8	5	8
S34	7	6	5
S35	9	5	6
S36	4	7	3
S37	8	6	9