TR—85—005

由資訊驅動觀點建立辦公室資訊系統的通用模式

計　劃　報　告

AN AGENT SOCIETY MODEL FOR OFFICE INFORMATION SYSTEMS

主　持　人：郭　德　盛[1]

協同主持人：劉　寶　鈞[2]

研究助理：何　正　信

洪　炯　宗

陳　稼　興

郭　鈞　宗[3]

1. 中央研究院資訊科學研究所
2. 國立中央大學電機研究所
3. 國立台灣大學電機研究所

中華民國　七十四　年　九　月

AN AGENT SOCIETY MODEL FOR OFFICE INFORMATION SYSTEMS

CONTENTS:

摘　　要

辦公室資訊系統一直廣被視爲計算機科學甚難克服的領域之一。部分困難在於缺乏廣泛合理的辦公室資訊系統理論。本研究報告旨在爲辦公室資訊系統建立理論雛型。

從辦公室運作行爲與成長原則的探討過程中，我們提出了“工作員社會”模式作爲辦公室資訊系統的塑模基礎。我們相信惟有透過辦公室行爲內在精神的描模，才有可能發展出一個能應付實際問題的辦公室資訊系統。

在工作員社會模式中，辦公室實體經視爲工作員，工作員則以互相影響的智識源所構成的微型社會來模擬。在該微型社會內，所有知識源均利用一套微知識交換協定在微知識交換系統上作知識的交流與動作的協調。辦公室則模塑成諸多工作員所構成的高度發展的社會。諸工作員利用其知識進行社會活動以協定完成辦公室的目標。

利用單一化觀點來模塑工作員及辦公室實是本工作員社會模式的一大特色。因爲單一化觀點代表著簡易，富彈性，與通用性。這些特性均是一良好模式所需具備者。

ABSTRACT

The field of office information systems has been widely recognized as one of the hardest problems that computer science may attack. Part of the difficulty is that a comprehensive theory of office information systems is not yet developed. This report is a first step toward the theory of office information systems.

In this report, we propose an agent society model for offices, which is derived from the study of the behavior and development principles of offices. We believe that only through the modeling of the inside spirit of the behavior of an office, we may develop an office system which is capable of dealing with all the problems we may encounter in real offices.

In the agent society model, office objects are viewed as agents. And an agent is modeled as a micro society of interacting kowledge sources. Within the micro society, there exists a micro knowledge exchage system, which provides a set of micro knowledge exchange protocols as a coordination system among those knowledge sources during their cooperative reasoning process. An office is then modeled as a highly developed society of various interacting agents, who use their knowledge as means for social activities to cooperatively complete the office goals.

This unified viewpoint to model agents and offices is a unique feature of the agent society model, which stands for simplicity, flexibility, and generality. All of these features are well known as necessary conditions of a good model.

1. INTRODUCTION

Office information systems have invited a lot of researches and investments from both industries and acadamies over the past few years [ELLI80]. Initially, most of the involving industries concentrate themselves on the developments of new "business machines" to upgrade or replace old, inefficient tools [KING83]. In contrast to them, most of the involving acadamies focus their researches on how to make an integrated office information system which can integrate various office facilities to accomplish so called office missions. These developments implicitly classified office automations into following categories [HONG82]:

1. Business equipments mechanization: This category of automation advocates the replacement or purchasing of office equipments to mechanise activity processing. The developments of electronic workstations, electronic mailing systems, teleconferencing, etc., are all classified in this category. The central idea is to cope with explosive information and the speed requirment in information processing by (individually) mechanized facilities [ELLI82, KING83,ZLOO81].

2. Integrated task automation: This category of automation is usually known as application-oriented. It emphasizes on how to integrate office various facilities to accomplish office missions, also known as goals [MACF83]. Some examples are office procedure automation software, multiple fonts man-machine interfacing, etc.. The primary idea is to eliminate unnecessary human interventions which are well known as error-prone, and to alleviate time consumption, during which office workers transfer from one kind of facilities to others, by automated task invocation based on and derived from predefined office goals [ELLI82,KING83,NEWM80].

3. Office management supporting system: This category of automation provides appropriate and sufficient information for office decision-makers to facilitate decision-making, office policy making, office operation quality monitoring, and so forth. The main purpose is to develop an office information system which is capable of gathering information from various sources, analyzing those information, and proposing some suggestive conclusions to help office managers manage their offices.

[ELLI82] shows a sign of catergory 2 eventually accomodating category 1. As a matter of fact, many developments in category 2 use those facilities developed in category 1 [ZISM77], which reveals an escalating relationship from category 1 (level 1) to category 2 (level 2). Basically, included in level 2 are the researches of finding out a suitable model to describe offices or office task procedures, and analyzing modeled objects to discover potential inconsistency, incorrectness, inefficiency, deadlcok, and so on. Office models are the key roles to integrate different facilities to accomplish office goals. The success of office automation usually havily relies upon a "good model" [NEWM 80].

Roughly speaking, an ideal model should be able to reflect a real world situation faithfully (semantically), at the same time allow syntactical analysis. But unfortunately, these two aspects are usually in conflict. That is, a model may be designed to have preferred analytic power or describing power but not both. Many recent researches in office modeling have selected "syntactical" approach [ELLI79,HONG82,LADD80,ZISM77]. They have intentions of deriving analytic algorithms to help office workers make tasks much more consistent and efficient. In summary, level 2 addresses the office modeling: how to capature much more knowledge in real world and how to analyze the modeled objects. Then, what is the relationship between level 1 and level 2? The answer is: the "modeled" office is to be executed through the supporting of office facilities [ZISM77].

How about category 2 and 3? The situation seems a little different. Many models cited in [BARB82,NEWM80] which are suitable for category 3 manifest themselves having taken totally different viewpoints upon offices. That makes category 2 and category 3 much more like two different approaches rather than two related levels like that of level 1 and level 2 as expected. It would be very frustrating if category 2 cannot be extended to cover category 3 or category 3 cannot get support from category 2. A brief comparison of existent models may give some idea on how to accommodate both categories 2 and 3.

We may roughly partition office models into two basic types: those that emphasize the declarative aspects and those that emphasize procedural aspects. In the declarative approach to office modeling, one tries to capture different types of office objects existing in an office, the relationships among them, and the operations on these objects. In the procedural approach, one tries to model how the office objects are transmitted along the workstations in the office and what processing each workstation does on the objects. [GIBB82] is a typical example of declarative type of model. [CHAN82,ELLI79, LADD80,ZISM77] are typical examples of procdural types of models. [BAUE79, LEBN82] are not exactly procedural types, but in the sense of programming an example procedure or a decision table, they may also be classified as procedural types of models.

All of the procedural type models have a common limitation, namely, exception handling. It stems from the fundamental concept that those models employ, i.e., they think an office procedure is a "predefined" sequence of activites., and they do physically explicitly or implicitly express an office procedure as a predefined connection of relevant activities. But offices are "real-worlds", which encompass so many unpredictable factors that predefined procedures are incapable of capturing all knowledge. The worst case with those models happens when a procedure is in halfway and an undefined situation emerges. Without specifying, in advance, this undefined situation in ICN [ELLI79], FFM [LADD80], alerters [CHAN82], or Petri nets [ZISM77], these models will fail to continue the processing of the procedure.

Besides, viewing office tasks as sequences of activities explains what, and more or less how, a task is, but not why we use the procedure to represent the task. That is, no very informative information can be derived from such an office information system.

These limitations, among others, make these models not flexible enough to accommodate knowledge needed in category 3. That's why a totally different approach has been used to attack the category 3.

As to declarative types of models, they also suffer an inherent limitation of being awkward in capturing dynamic knowledge. That is, they seem more suitable for category 3 than category 2. This limitation has invited lots of extensions to declarative approach. For example, [AIEL84] has focused on the office structures modeling instead of office objects only. The concept of agents introduced in [AIEL84] incorporates much more dynamic functions. [BARB82] is a pioneer in its intention of introducing techniques in Artificial

Intelligence [BARR80,NILS80,RICH83,WALT82] to solve office tasks by viewing office task execution as a problem solving task. The fact (static) knowledge about office objects is embedded into the knowledge base by "descriptions" or "viewpoints" (for aggregated objects), and the dynamic knowledge is embodied in the form of "sprites" which serve as triggers of actions while assertions are made or goals are posted in the knowledge base. [CHEN84,CHOU83,DU84] depict a system which interactively acquire knowledge from office designers for an office automation system. The knowledge, both static and dynamic, is represented in ORAL language. Basically, ORAL is a logic programmimg language augmented by data typing on objects defined or referenced in logic programs. This approach has the advantage of being able to interpret ORAL programs both in declarative and procedural ways. These extensions are either toward some pre-defined directions [BARB82] or unable to capture some important knowledge, for example, knowlege of regulations [AIEL84] and inexact knowledge [CHEN84,CHOU83,DU84].

In summary, an integrated office system needs to attack problems like exception handling, reasoning about tasks, knowledge growing or updating to meet new requirements, and so on. We found that most of the existing models either skip the problems or address it with no solutions. We have also found that lack of sufficient knowledge is the reason that those systems cannot successfully address the problems.

Now, what we need is to develop a suitable model which can capture as much as real-world knowledge of an office, and which is expected to incrementally "grow" by acquiring knowledge from external worlds (e.g., office specialists). Owing to the much more knowledge embodied and the way the knowledge is used, such a system will be expected to avoid those problems mentioned above. That is, in addition to satisfying the requirement in level 2, this system will behave as a herald to explore problems in category 3.

We have proposed a project to develop such an office information system. In the following section, we will present an agent society model as a base of the office information system under development. The model is expected to be able to address all problems mentioned in this section. Based upon this model, our next research topic would be to develop a building system for office information systems.

2. OFFICES AND THEIR BEHAVIOR

To make an office model able to faithfully reflect a realworld office, we have to investigate offices in more detail, expecially on their structure, their behavior, and their nature. In this section, we first give a more formal definition of offices and derive two principles which governs offices behavior and growth procedure. Based upon these observations, we then propose an agent society model in subsquent sections to model offices. The model is unique in its interpretation of agents, its emphasis on knowledge, its strategy to model office activities, and its striking similarity to real-world offices.

Definition 1

An office, being an information center of an organization, is a 8-tuple (M,B,A,C,F,S,R,K), where

M: is a nonempty set of missions. It stands for the meaning of existence of the office and serves as a driving source of all office tasks;

B: is the body of the office. It refers to office space, probably containing more than one floors or buildings;

A: is a nonempty set of agents. The word "agent" is used here in a most general sense. Three types of agents are readily identified, namely, personnel, office objects, and their processing tools. Personnel includes several types of workers, such as executives, managers, professionals, clerks, etc., Examples for office objects are order forms, reports, documents, letters, etc., And the processing tools include even wider types of facilities, naming a few, word processors, data pracessing, typewriters, teletypewriters, OCR's, PC's, workstations, and so forth;

C: is a nonempty set of communication channels, which form the neural tract of the office. It is the communication channels that make agents within an office able to cooperate to achieve the office missions. It is also the communication channels that make an office able to communicate with its environment to faithfully respond to and activate the outside word;

F: is a nonempty set of information storage facilities, which serve as both perpetual and temporal storage area of office objects;

S: is a nonempty set of structures, which set agents in proper roles of some office structure, hence in turn, characterize the architecture of an office. Several structures exist simultaneously within an office, for instance, hierarchically ligitimate status structure, communication structure, responsibility structure, etc.;

R: is a nonempty set of regulations. We can divide it roughly into two groups. One is regulations enforced by the organization, e.g., organizational policy. The other is legistrations constrained by external environments, for example, the labor standard law; and

K: is a nonempty set of knowledge bases. Knowledge may be codified or uncodified. Codified knowledge refers to the knowledge which is "physically codified" in text, graph, or any other forms. Examples of them are knowledge of the office structure, office regulations, statistics of agents' expertise, market information, etc., Uncodified knowledge instead contains all knowledge which is not "yet" codified. Some of them is so called meta knowledge [DAVI84,LENA82], e.g., how to make an effective plan to solve a problem [STEF81]. Some are different types of task solving knowledge, for instance, accounting expertise, management expertise, and so on.

Among the components of an office, office communication channels play a rather salient role in the office behavior. For example, office communication channels provide suitable mechanisms by which agents can cooperatively work together to come up with a comparatively cost effective way to complete a mission. In the suquel, we will look more closely at these office communication channels.

No mater what type of channels we may use in office communications, either in voice or text, the main purpose of the communication is knowledge exchange. For example, an agent may try to gain more knowledge from other agents to help him solve a task. Based on how an agent is working on a task, we find out three different methods of knowledge exchange mechanisms in offices.

In the simplest case, supposed an agent is working alone on a task. That is, he is **thinking** of how to solve the task. We call that the agent is **musing,** or he is in a **muse mode.** In other words, he is attempting to solve the task by exercising his own thought.

Now supposed that he fails to complete the task due to lack of relevant knowledge, normally he will ask another agent (who he knows may help some) either to "teach" him how to continue the task, or to take the responsibility of the task. In general, if the incompleted part of the task is beyond the responsibility of the agent, he will simply pass down the task to another agent. Otherwise, the agent will attempt to gain more knowledge from another agent to complete the task. In either case, when two (and only two) agents are engaged to solve a task, we call that they are in a **conversation mode.** The typical reasons for two agents to go into a conversation mode are personal tutoring, responsibilty compromising, cooperatively working out a plan, and so forth.

If, unfortunately, the second agent still cannot help the first agent. The first agent normally will continue, in turn, to engage conversation modes with other agents, until he is unable to find out any more help. That is, he has encountered a task that neither muse nor conversation mode can solve. A third mode called **session mode** is then requested.

The session mode of knowledge exchange is used to work out a plan, develop new knowledge, compromise responsibilities, etc., among more than two agents. The result of session may be knowledge redistribution, new knowledge generation, new agents recruiting, office structure reconfiguration, etc..

In summary, depending on how hard a task is, an agent may attempt to solve the task through a muse mode, a conversation mode, or a session mode.

The above discussions suggest that an office communication system have to provide mechanisms which allow agents to perform musing, conversation, and session kinds of social activities for the purpose of knowledge exchange in order to cooperatively achieve office goals. Hence, we may define the office communication system as follows.

Definition 2

An office communication system is a knowledge exchange system among office agents. It provides a set of knowledge exchage protocols, and forms the basis for office agents to cooperatively solve tasks, which are expected to achieve delegated goals of the office. Three basic modes of knowledge exchange protocols are: muse, conversation, and session, which serve as mechanisms to facilitate the knowledge exchange among office agents.

The above definitons make us able to derive a basic principle for office behavior. We state it below as an office behavior law.

First principle (Office behavior law):

Office behavior is subject to the principle of mission-oriented, stimulus-driven, knowledge-based and comparatively cost effective law.

The behavior of an office is characterized by its response to stimuli either from within the office or from the external environment in which the office is involved. The office tends to exploit all its knowledge, either centrally located or distributedly located, through the help of its neural tract and under the constraint of regulations, to solve these stimuli-driven tasks, which will singlely or aggregately lead to one of its goals. Furthermore, the procedures that the office finally comes up with to solve the tasks are normally one of the most cost effective ways. This phenomenon is called the law of mission-oriented, stimulus-driven, knowledge-based, and comparatively cost effective behavior. Since the time that stimuli ever happen is in general unexpected, "dynamicness" hence becomes the nature of an office. Trying to characterize statically the work flow of an office is therefore not just impractical, sometimes even impossible. Examples for internal stimuli are workforce change, time schedule, and mission shift. Customer request, market growth, and cost fluctuation are instead examples of external stimuli.

The second principle deals with the office developments.

Second principle (Office development law):

Offices are evolutionally developed.

Offices are subject to evolutionary growth as well as organizations. Generally, at initiatory stage, an office is located in a single place with all components of an office concentrated on one or two agents. As the office gets into operations, the behavior law will make the office adjust itself, which at the preliminary stage normally causes it to expand. This expansion will eventually grow the office into an equilibrium state (roughly thought as a mature stage). During the evolution, all components of an office will grow up and get scattered, and finally form a preliminary model of a "society". By this we mean that many communities (or subsocieties) will be shaped as the behavior law needs them. This phenomenon is called the evolutionary development law. Several typical factors which cause the behavior law to grow an office are market growth, multi-directional productions, etc..

3. THE AGENT SOCIETY MODEL

We now propose an agent society model for offices which views an office as a **"highly developed society"** of interacting **agents** using **knowledge** as a means for **social activities.** All the social activities are basically esteemed to be beneficial to the **goals** of the society.

By **"highly developed society"** we mean that a society is constructed by communities (or subsocieties), which in turn can be constructed by subcommunities, all enforced by office legitimate structures. A community is a set of agents who are functionally related. Applied on offices, a community can be thought as a department. Under this hierarchical community structure, agents associated with a specific community may communicate with those agents who are in the same community, formally or informally. But to formally communicate with agents in other communities, one has to pass the "message" up to the agents of highest level of the community, who are then responsible to communicate with other communities. By formally we mean that the activity of the communication is recognized as a legitimate action.

Goals are often denoted as missions in offices. Since the structure of an office is a society with community hierarchy, goals normally consist of several subgoals, specific to different communities. This refinement doesn't limit on a single level. Hence the achievement of a goal normally implies the completions of many levels of subgoals.

Social activities refer to all activities which directly or indirectly make contributions to office goals. Conversation, discussion, meeting, typing, data processng, accounting, etc., are obvious examples.

Knowledge is considered as a means to complete social activities. We can grossly partition social activities into two groups, namely, information processing and communication (or knowledge exchange). It has been well recognized that an agent reqiures knowledge (e.g., idea, skill, expertise, etc.) to process information. But when he feels that his knowledge is insufficient to finish the task, he is going to ask other agents to help him. That is, he may invoke a communication event to get more knowledge form other agents. The knowledge over the communication channel then becomes the key point for cooperative processing.

The word **agent** is used here to mean an intelligent entity, which in some sense like an actor [BYRD82,HEWI77], or an expert system [HAYE83, MELL84,SHOR84,STAL79]. He is capable of reasoning based upon his knowledge to solve a problem. For offices, agents may refer to personnel, intelligent systems, intelligent objects, etc..

To make our model able to accommodate most general sets of agents, we have chosen to model an agent as a **"micro society"** of interacting **knowledge sources** [ERMA80]. Inside the micro society, there is a **micro knowledge exchange system** used as a coordination system among knowledge sources to facilitate the reasoning process inside an agent. The micro knowledge exchange system will provide a set of protocols, which are intuitively very similar to those protocols defined in definition 2. The definition of agents is as follows.

Definition 3

An agent is a micro society of interacting knowledge sources. That is, an agent is a 5-tuple (Gai, Kai, Sai, Dai, Xai), where

Gai: is a nonempty set of goals subject to agents' roles. Each role in the structure of an office assumes some responsibilities to achieve some goals (or subgoals from the standpoint of the whole office). Gai is the set of goals that an agent is supposed to achieve if he is put into some role of office structures;

Kai: is a nonempty set of knowledge sources (ks). A knowledge source is defined as a primitive action. It is actually a partial miniature of an agent, being confined to a more specific function. For example, a single step in a speech understanding system [ERMA80], or a brain cell in the society of human mind [MINS79]. Basically, it consists of a processor and a knowledge base. The knowledge base contains some domain-specific knowledge. The processor is related to the knowledge representation method used in the knowledge base. It is responsible for processing something based on the knowledge drawn from the knowledge base according to the environment status. Several knowledge sources are mandatory: agent descriptor, acquaintance relationship ks, common message processing ks, micro planner, etc.. Wherein, the agent descriptor specifies the agent's identifier (e.g., his/her name), agent's roles (e.g., his/her position), agent's expertise, and other attributes related to the agent. The acquaintance relationship ks specifies the relationships between the agent and other agents. For example, the community relationship apecifies what community the agent belongs to, hierarchy relationship specifies whom the agent obeys or orders to, responsibility relationship specifies what events the agent has to take responsibility for, etc.. The common message processing ks serves as an interface to the knowledge exchange system, i.e., it interpretes messages recieved from other agents and sends out messages to other agents. The micro planner is a special ks. Its major job is to make a plan to solve a subgoal as the driving source for the subgoal emerges. Its knowledge base contains heuristics (or meta knowledge) of how to make plans for some pre-recognized subgoals. For those subgoals that the micro planner itself can not make plans, the micro planner will resort to micro-conversation or micro-session mechanism (see below)attempting to make a plan. Once the plan is made, it will distribute each step through the micro knowledge exchange system to other ks's for distributed processing. Many other knowledge sources are agent-dependent, which facilitate the operations of office activities. We will see later that a set of ks's that are functionally related forms a micro community. Hence, effectively, a micro community is the unit of knowledge which enables office activities to be done;

Sai: is a nonempty set of structures, which specify the structural relationships among knowledge sources. One basic structure is the micro community hierarchy which partitions the whole knowledge sources within an agent into several micro communities. A micro community is a set of knowledge sources which are functionally related. For agents, a micro community can be thought as a "package" like meeting scheduler, decision support system, data entry system, etc.;

Dai: is a nonempty set of databases. In general, a database is locally associated with each knowledge source, which uses the database to reflect the environment status that the knowledge source is currently involved in; and

Xai: is a "micro knowledge exchange system", which serves as a coordination system of knowledge exchange within an agent (or among knowledge sources), thus serves as a mechanism for agents to perform **musing.** It provides a set of micro knowledge exchange protocols to facilitate the knowledge exchange among ks's. These basic protocols, called "micro mechanisms", are a simplified version of office knowldge exchange protocols, i.e., micro-muse, micro-conversation, and micro-session modes:

* Micro-muse: When a knowledge source is exercising its own knowledge to solve part of a task, we say it is in a micro-muse. Since a knowledge source is, as defined, a primitive action, no further clarifications are required;

* micro-conversation: When two (and only two) knowledge sources are working together to work out something, we call that they are in a micro-conversation. The micro-conversation mode provides a micro-mechanism for these two ks's to set up a micro-conversation, proceed the micro-conversation, and clear up the micro-conversation, which form a basis for two ks's to achieve synergy [CHAN81]; and

* micro-session: When more than two ks's are working together to solve some problems, we call that these ks's are in a micro-session. The micro-session mode provides a mechanism for these ks's to rally a micro-session, progress the micro-session, and dismiss the micro-session, which form a basis for ks's to achieve unison.

The detailed description of these micro-mechanisms will be delayed until section 4.

Sometimes, an agent only contains a knowledge source. That is, no micro knowldge exchange system is required. This kind of agents is called **degenerated agents.**

Definition 4

A degenerated agent is an agent who has only one knowledge source (with an associated database of course).

We now give the definition of the agent society model for offices.

Definition 5

The agent society model for offices is a 7-tuple (G, A, S, R, K, D, Xk), where

G: is a nonempty set of goals. For offices, goals are often denoted as missions and normally decomposed into several subgoals based upon various office structures. Hence, the achievement of a single goal normally implies the completions of many subgoals specific to various office structures;

A: is a nonempty set of agents as defined in definition 3. In offices, all entities are modeled as agents. For example, personnel such as executive, manager, professional, etc. are different types of agents. Processing tools with associated objects such as database managers, decision support systems, data processing packages, form management packages, etc. are agents too. Furthermore, intellignce forms (messages) are all considered as agents;

S: is a nonempty set of office structures. Several structures exist simultaneously in an office, which put agents onto different roles of different structures. One basic structure is community hierarchy, which partitions a whole office into functionally different communities. Communities can be thought as departments in an office, which may in turn contain subcommunities such as sections under departments. All agents are put into some roles of communities (or subcommunities). Hence an agnet is capable of communicating with his supervisors and subordinates by keeping part of this structure relevant to himself. Another type of office structures is acquaintance relationship among agents. Any agent in an office may keep an acquaintance list to facilitate his formal and informal communications, which have been well recognized as important channels for knowledge development among agents;

R: is a nonempty set of regulations. Regulations are rules that confine the behavior of an office. For example, office internal policy is one type of regulations called internal regulations. Laws enforced by the government is another type of regulations called external regulations. There are lots of distinctions between these two types of regulations. First, internal regulation specify more detailed requirements that office agents must observe but they are subject to a confinement of never conflicting with external regulations. Furthermore, internal regulations may be constantly amended as organization grows. On the contrary, external regulations are more stable in the sense that to amend them is in general not very easy;

K: is a nonempty set of knowledge bases. It can be thought as U Kai U Ks U Kr U Kg. Wherein, Kai is the knowledge base of anent i (see definition 3), Ks is the knowledge of office structures, Kr is the knowledge of regulations, and Kg is the knowledege of goals;

D: is a nonempty set of databases. It can be thought as $\bigcup_i$ Dai with all agents modeled as agents. Wherein, Dai is the database associated with agent i (see definition 3); and

Xk: is an office knowledge exchange system, which serves as a coordination system of knowledge exchange among agents in an office. It provides a set of knowledge exchange protocols to facilitate the knowledge exchange among agents. Three basic protocols are muse, conversation, and session modes:

* muse: When an agent is working alone to solve a task he is actually using a muse mode to solve problems. Externally, no communiations with other agents are necessary while an agent is in a muse. But internally, the agnet is busy in exercising his inside thinking mechanisms to attack problems, which normally characterizes the (problem solving) behavior of an agent. Since an agent is defined in definition 3 as a micro society, the muse mode is in fact the behavior of the micro society;

* conversation: When two (and only two) agents are cooperating together to solve a task, they are actually using a conversation mode to exchange knowledge. Basically, the conversation mode provides a sequence of procedures for two agents to set up a conversation, proceed the conversation, and clear up the converstaion, which form a basis for two agents to achieve synergy during their cooperating period; and

* session: When more than two agents are cooperating together to solve a task, they are actually using a session mode to exchange knowledge. Basically, the session mode provides a sequence of procedures for agents to rally a session, progress the session, and dismiss the session, which form a basis for agents to achieve unison during their cooperating period.

The detailed description of these mechanisms will be given in the following section.

4. MESSAGES AND KNOWLEDGE EXCHANGE PROTOCOLS

Conventionally, messages are used as carriers of information. While applied on office information systems, over which knowledge plays an important role, we find that the content of message has to be augmented if they are still used as **carriers of knowledge.** To make it clear that knolwledge is "generative information" [NEWE82], rather than only information about facts, we define that **a message is an agent,** which is the most general concept of a message. In other words, the message may contain inferential knowledge and one processor, in addition to the information (including fact knowledge) usually existent in a conventional message. The processor may be a specific process associated with a particular message, or an instance of a common used process supported by office communication systems. For simplicity, we assume the last case for the time being.

To make sure that agents in a society can correctly exchange knowledge with one another we have to carefully deal with both syntax and semantics aspects of "knowledge exchange protocols". We have chosen using message agents as the media of knowledge exchange protocols. That is, in the syntactical aspect we will assume that a common message framework is used as a skeleton of message agents. Besides, a common language used by agents to express knowledge onto messages during knowledge exchange events will aslo be assumed. As to the semantics aspect (procedure aspect, or knowledge exchange mechanism) of the knowledge exchange protocol, we will adopt the basic knowledge exchage mechanisms described in definition 2 and describe them more clearly in the following subsections.

4.1 Muse-mode

We have pointed out in definition 3 that there are three micro-mechanisms to simulate the musing process of an agent. Among them, the micro-muse is a primitive action, which requires no further explanation. Hence we only have to describe micro-conversation and micro-session modes here.

4.1.1 Micro-conversation Mode

Basically, a micro-conversation protocol is composed of three main states: **setup, engaging,** and **clearup.** In the following discussion, we will first talk about set up procedures, followed by engaging procedures with clearup procedures as the last.

SETUP

Depending upon whether or not the originating ks knows the destination ks, three different types of micro-conversation setup procedures are provided. They are **direct contact, focused broadcast,** and **universal broadcast.**

If a ks knows the destination ks, it will directly send out a micro-conversation SETUP message with the address of the destination ks and a description about the micro-conversation topic. The destination ks will respond with a ACK nowledge after receiving the SETUP message. We call this type of setup the direct contact. We will assume a time limit is set. Any ACK received beyond the limit is thought to be ineffective. Hence, it provides to the originating ks a mechanism to switch to other agents once the chosen destination ks is unavailable to converse with. If the destination ks is incapable of fully understanding the topic contained in the SETUP message, it can use ACK to ask more information from the originating ks and decide whether or not going into the engaging state. The originating ks, after receiving the ACK with more information solicited, will send out an EXPLAIN message to give more information to the destination ks. This procedure will continue until both sides of ks's are ready to go into the engaging state. Fig. 1 shows the setup procedure of a direct contact with a typical engaging state of monitor-executer mode (see below) for task sharing [DAVI83].

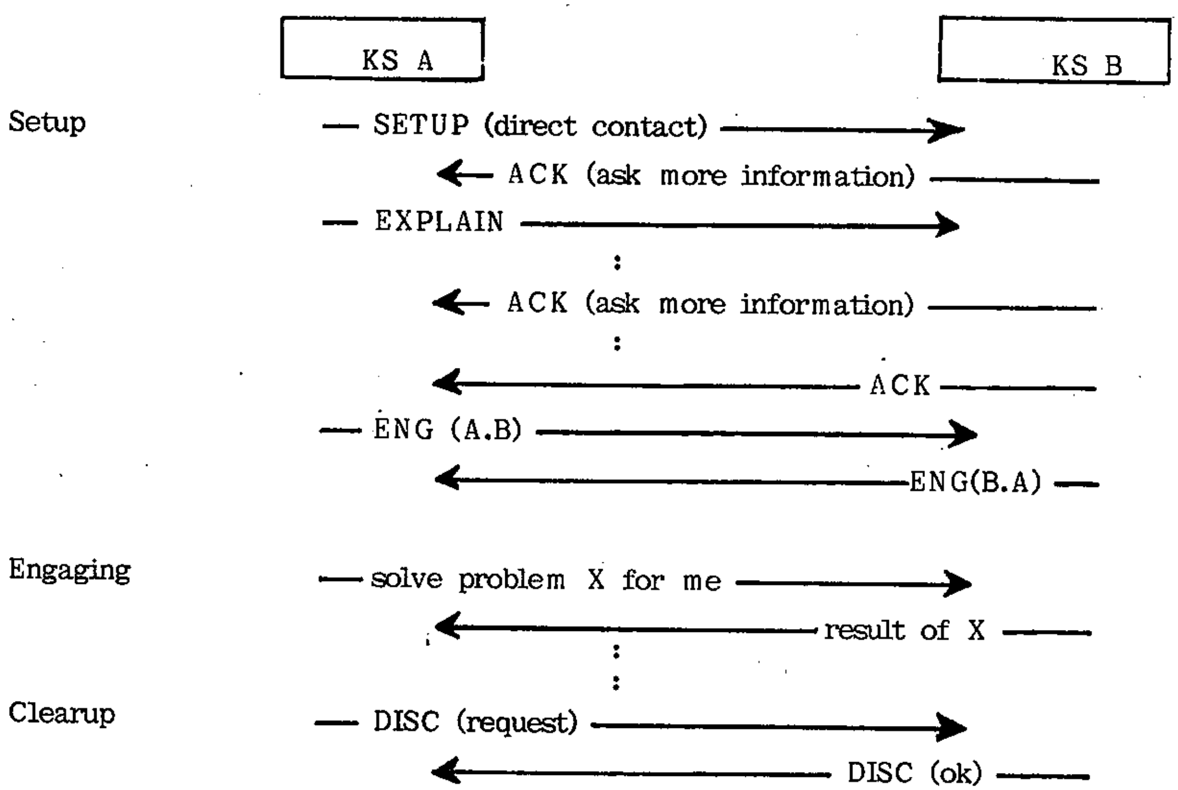
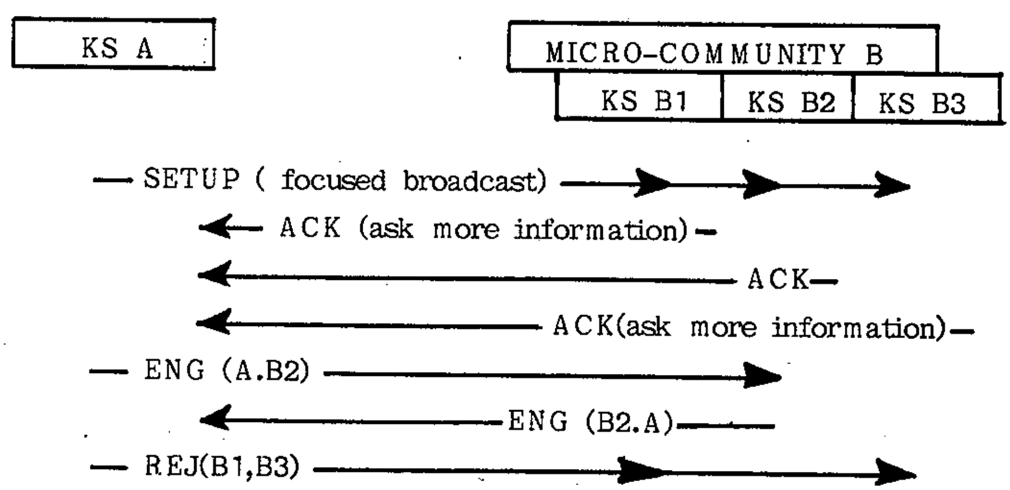


Fig. 1

If the ks only knows, say, the micro-community name of potential destination ks's, it can use a focused broadcast procedure to set up a micro-conversation. The ks first sends out a SETUP message with the address of the target micro-community and a description about the theme of the micro-conversation. This SETUP message will be broadcasted to all ks's of the micro-community after it arrives the micro-community. Any ks that is capable of participating in the requested micro-conversation may respond with an ACK message containing the address of the respondent and sometimes a request for more information as in direct contact modes. We also assume that a time limit exists. Only those ACK's which arrive before the time limit are recognized as legal candidates to converse with. The originating ks will then keep these ACK's and pick one of them (hence reject the rest of them) for engagement. If the ACK is the type of no more information requested, the two ks's then directly go into the engaging state. Otherwise, an EXPLAIN message will be sent from the originating ks to the destination ks. If the micro-conversation doesn't help, the originating ks will disconnect it and enter direct contact modes with the rest of respondents one by one. Fig. 2 gives an annotated diagram of the focused broadcast setup.



* If the micro-conversation doesn't help, KS A disconnects B2 and enters direct contact mode with B1.

Fig. 2

Originally, those ACK's are temporally stored. But, they may finally become knowledge of the originating ks if the same records occur very often.

If the ks knows nothing about the destination ks, he can use universal broadcast to set up a micro-conversation mode. A universal broadcast is in most facets the same as the focused broadcast except that SETUP messages with a micro-conversation topic is broadcasted to all ks's within an agent instead of just the focused group of ks's. Any ks that is familiar with the topic can respond with an ACK containing his own identifier and an optional request for more explanation to the originating ks. Once the originating ks get those responses, the rest of the procedures are basically no different from those of direct contact or focused broadcast.

ENGAGING

Depending upon how two ks's are cooperating together in a micro-conversation, we may have three general modes of engagement. They are **monitor-executer, student-consultant,** and **discussion** modes.

An originating ks may download problems to the destination ks and monitor the execution of the destination ks. In this case, the destination ks serves as an executer to execute tasks downloaded from the originating ks, who then plays the role of monitor of the destination ks. We call it a monitor-executer mode. Fig. 3 shows an example of the engaging state of monitor-executer modes.

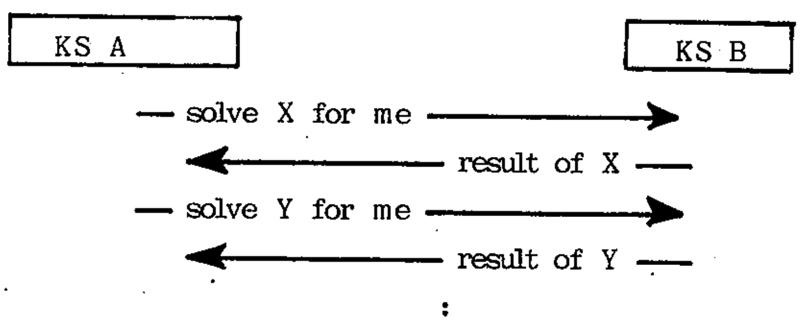


Fig. 3

An originating ks may also ask knowledge from the destination ks to help solve problems. Under this condition, the originating ks is a student to gain instructions from a consultant, i.e., the destination ks. We call this type of engaging state a student-consultant mode. Fig. 4 shows a simple example of an engaging state of student-consultant mode.

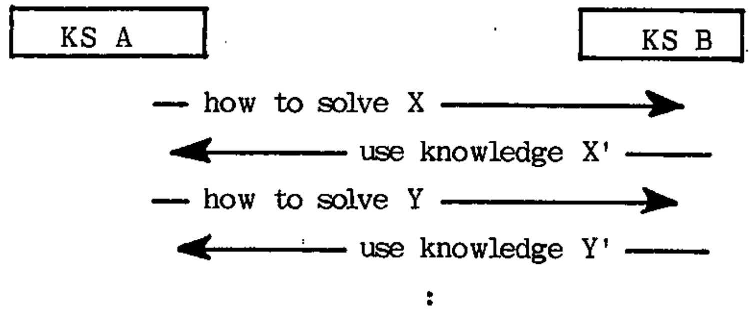


Fig. 4

If the knowledge of both ks's are not dominant over each other, they may use discussion to help each other. For example, ks A solves part of a problem and ks B solves the rest of the problem. Then both of them look at the result of the problem to see if any modification is needed. This procedure may repeat until a satisfactory result is reached. We call this type of engaging state a discussion mode. Fig. 5 gives an example of a discussion mode.

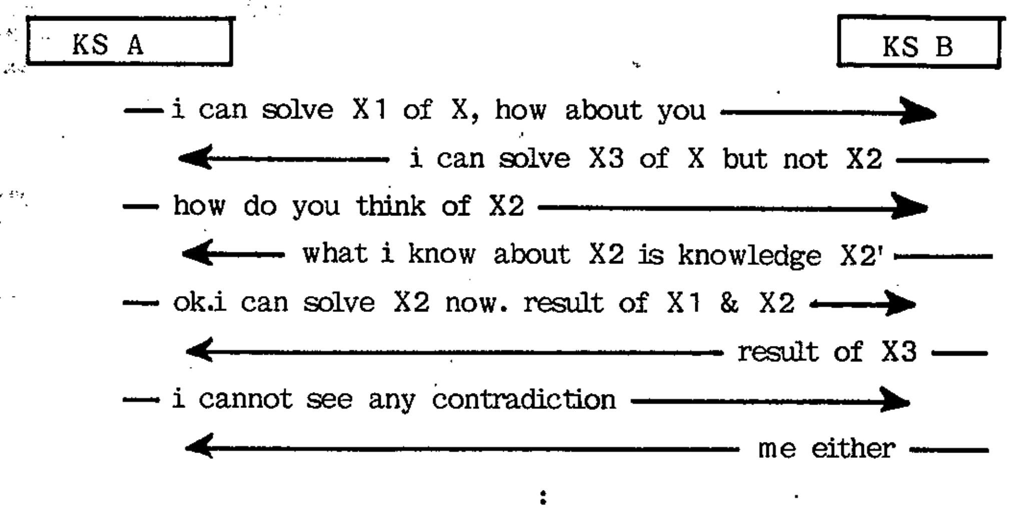


Fig. 5

CLEARUP

Each side of a micro-conversation may request to disconnect the micro-conversation. But it has to await the grant response from the other before really disconnecting the micro-conversation. The clearup procedure in Fig. 1 is an example. To prevent perpetually waiting due to an abormal disconnection, several mechanisms, e.g., time out, could be used. Since these mechanisms are not specific in our knowledge protocol model, we will skip them in the report.

4.1.2 Micro-session Mode

As in micro-conversation modes, there are also three main states in micro-session protocols. They are **rally, meeting,** and **dismissal** states. These names are chosen mainly for easy discrimination from micro-conversation modes.

RALLY

Depending upon whether or not the initiating ks, that normally is the chairman of the micro-session, knows the attendants of the micro-session, two types of rally procedures are provided. They are **selective rally** and **universal rally.**

A ks may use selective rally to call for a micro-session if it knows the address of every ks to be attendants, the groups of ks's to be attedants, the micro-communities, members of which are going to be attendants, or any combination of the above. First, the ks will initiate the micro-session by sending out RALLY messages directly to known ks's or micro-communities, which then serve as relay stations to broadcast the RALLY messages to all their members. A RALLY message contains the address of destination ks or micro-community, major topic of micro-session, the name of the chairman, etc.. Any ks, that receives a RALLY message, may decide whether attending or not majorly based upon the micro-session topic. If the decision is to attend, an ACK message is sent back to the initiating ks. We still assume a time limit is set. Only those ACK's arriving in time are legal candidates to attend the micro-session. As in micro-conversation modes, ks's are allowed to use ACK to get more explanation from the initiating ks. But to avoid over loading the communication system, the initiating ks may decide not to send an EXPLAIN message, i.e., instead send a REJect message. When the initiating ks is satisfied with those ACK's, it will send a MEET message to each of the respondent ks asking them to enter meeting mode. REGister messages are then used by all to-be-attendants to signal the readiness to the meeting mode. Fig. 6 gives an example of a selective rally with a typical meeting state of 1:n mode (see below).

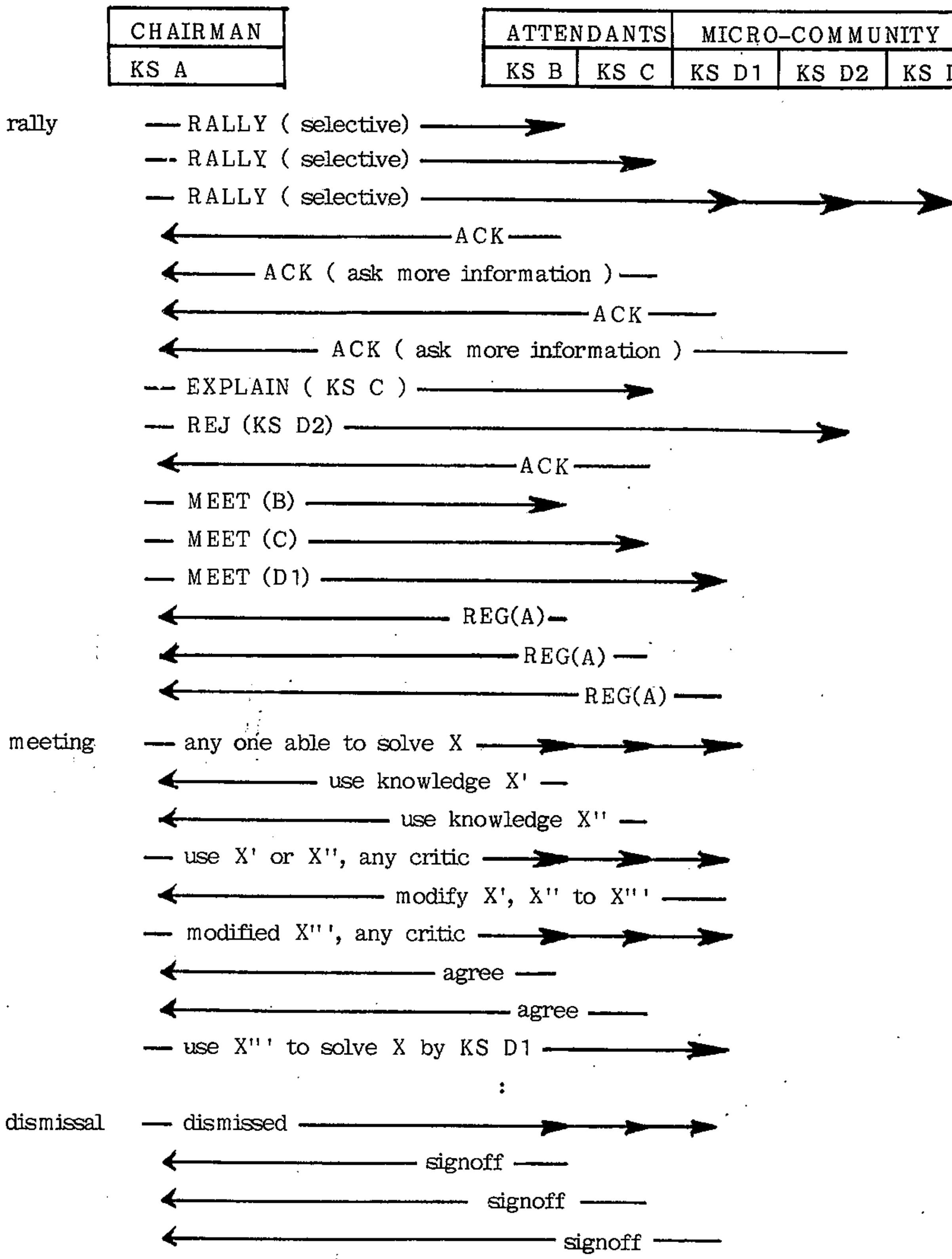

CHAIRMAN
KS A
ATTENDANTS
MICRO-COMMUNITY D
KS B
KS C
KS D1
KS D2
KS D3
rally
RALLY (selective)
RALLY (selective)
RALLY (selective)
ACK
ACK (ask more information)
ACK
ACK (ask more information)
EXPLAIN (KS C)
REJ (KS D2)
ACK
MEET (B)
MEET (C)
MEET (D1)
REG(A)
REG(A)
REG(A)
meeting
any one able to solve X
use knowledge X'
use knowledge X''
use X' or X'', any critic
modify X', X'' to X'''
modified X''', any critic
agree
agree
use X''' to solve X by KS D1
:
dismissal
dismissed
signoff
signoff
signoff

Fig. 6

A universal rally is used to call for a micro-session if the initiating ks knows nothing about potential attedants. The universal rally is in most facets the same as the selective rally except that RALLY messages with topic of the micro-session are broadcasted to all ks's within an agent instead of those selected ones. Any ks that is interested in the topic of the micro-session may respond with an ACK containing his own indentifier and an optional request for more explanation to the initiating ks. Once the initiating ks gets those responses, the rest of the procedures are basically no different from those of selective rally.

MEETING

Depending upon how members of a micro-session are working together, we may have two general modes of meeting. They are **1:n** and **1:1** modes.

Normally, a chairman uses a 1:n mode to progress a meeting. During the 1:n mode, the chairman broadcasts information either to all members or to a specific group of members. A time limit is assumed here to set an expired time before which the target attedants are supposed to respond. One example of 1:n mode is that the chairman solicit opinions of attendants. The meeting mode shown in Fig. 6 is such an example.

A chairman may also temporarily dedicate himself to somebody on a specific topic. We call that they are in a 1:1 mode. Examples of 1:1 mode are that during a micro-session the chairman may delegate something to someboday and ask him to speak out difficulties, the chairman may appoint somebody to "stand up" and explain his opinion, etc.. Fig. 7 is an example of 1:1 mode meeting.

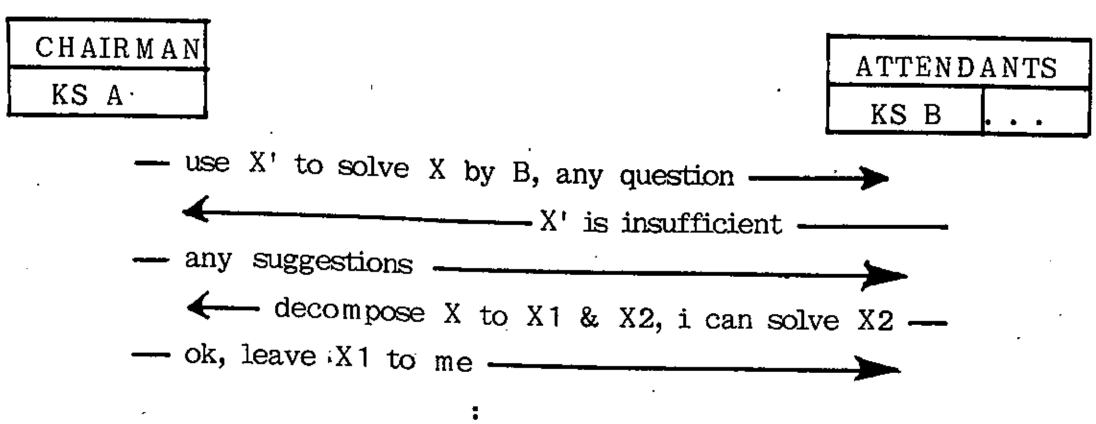


Fig. 7

In a real-world session, we may have several meeting states, which are slightly different from these two modes described above. For example, an attedant may be assigned "to speak out his opinion to all attedants". That is, we need a mode of "somebody able to broadcast his opinion to all members", which is neither 1:1 nor 1:n mode. However, we can achieve the same effect by a 1:1 mode followed by a 1:n mode with the chairman as a "relay" station. This arrangement not only prevents latent problem of more than one attedants broadcasting messages without coordination and making a mess (a Nihilistic state) but also improves the efficiency of the communication system if the chairman plays the role of a "filter" rather than just a "relay". By filter we mean a chairman may enter a 1:1 mode to get more exact information from somebody and broadcast it to all members (1:n mode) after processing the information.

DISMISSAL

When the chairman is satisfied with the result of a micro-session, he may declare that the micro-session is dismissed. And after all attedants have signed off, the micro-session is successfully closed. Any attedant also may request that the micro-session be dismissed, but the chairman is supposed to resort to all attedants for the agreement before declaring the dismissal.

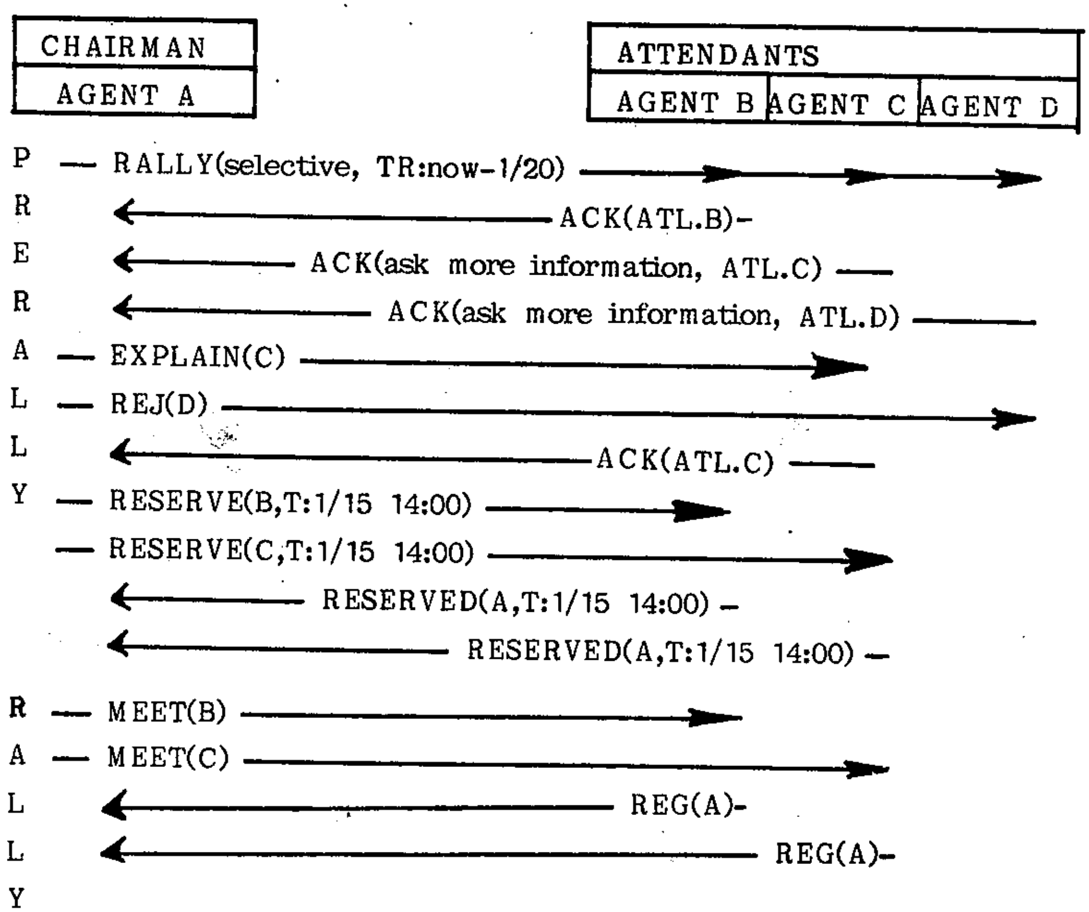
4.2 Conversation and Session Modes

Conversation and session modes are two mechanisms provided by office knowledge exchange systems to facilitate, respectively, the conversation and session activities among agents. These two office modes are essentially the same as their corresponding micro mechanisms, i.e., micro-conversation and micro-session modes, provided by micro knowledge exchage systems. That is, most of the mechanisms described in micro-conversation (micro-session) modes are applicable to the converstaion (session) modes if we replace knowledge sources and agents in micro modes by agents and offices respectively. Hence, we won't repeat the whole mechanisms over here. What we are going to do in the following is to depict some of the major differences between them.

One major difference between micro modes and office modes resides in the content of messages used as media for communication among knowledge sources (in micro modes) or agents (in office modes). In office modes, messages are agents containing not only fact but also inference knowledge. In micro modes, however, messages are intentionally chosen as a limited version of message agents, i.e., they contain only fact knowledge. This is due to the prejudice that knowledge needed to exchange among ks's over micro knowledge exchange systems is not so complicated as that in office knowledge exchange systems. The reason is this: if an agent is unable to solve a task, we expect that he can "learn" complicated knowledge from other agents through the help of office knowledge exchange system. In fact, inside an agent learning means that some knowledge sources are upgraded or a new ks is fostered. But if a ks is unable to solve task, what it has to do all the time is just ask other ks's to solve the task rather than attempt to learn new knowledge and solve the task by itself. Of course, asking knowledge from other ks's to solve a task is not necessarily impossible for a ks, but in general only simple fact knowledge is exchanged. This choice is in one side to make ks's as simple as mind cells inside a real agent and in the other side to make micro society easier to implement.

Another difference consists in the setup procedure of a communication event between micro modes and office modes. In general, agents in offices are much more knowledgeable than knowledge sources. Hence they are allowed to speak out more opinions according to their individual situations. For example, during the rally procedure, selective or universal, of a session, the initiating agent is in general unable to predict the available time of other agents. If he decides the session time in advance without the knowledge of the availabilability of other agents, the number of attendants to the session would not be so optimistic that the session result may be comprehensively acceptable. One way to solve this difficulty is to ask any to-be-attendant to speak out his available time before making the decision of the session time. The rally procedure for the session mode is hence slightly complicated. First, a RALLY message with a possible time range to open a session is sent out. Any one receiving the RALLY message may respond with an ACK message containing **his available time,** an optional request for more information, and his own identifier (in the universal case only). After the initiating agent having processed the explanations to those ACK's which request more information, he will decide the opening time of the session based upon those available time listed in the respondent ACK's. If the opening time is immediate, he will send out MEET messages to all to-be-attendants. All agents receiving the MEET messages are supposed to send back REG messages immediately and enter the meeting mode. Otherwise the initiating agent will send out RESERVE messages to all to-be-attendants. Any agent receiving a RESERVE message must send back a RESERVED message and leave the rally state. The initiating agent will not leave the rally state until all RESERVED messages are received.

Once the opening time comes, the initiating agent will send out MEET messages to all to-be-attendants and enter the meeting state after he has received all REG messages from them. Fig. 8 shows a typical selective rally state of a session mode. In the figure we have partitioned the state into two stages of PRE-RALLY and RALLY to represent, respectively, the substate before and after the logical session opening time. As to the universal rally this same modifications can be applied to the micro mode and get the corresponding office mode.



* TR: Time Range
* ATL: Available Time List
* T: Time reserved

Fig. 8

We can also attribute this time decision feature to conversation modes. Fig. 9 is an example of a direct contact to set up a conversation. Upon receiving a SETUP message, the destination side may respond with an Available Time List (ATL) along with an indication of being busy to the originating side, who then decides whether or not to request a time reservation for future contact. If not, a REJ message is sent out and the setup is aborted. Otherwise, a RESERVE message is sent to the destination, from whom a RESERVED message is expected. Both sides then leave the (pre) setup state until the reserved time comes.

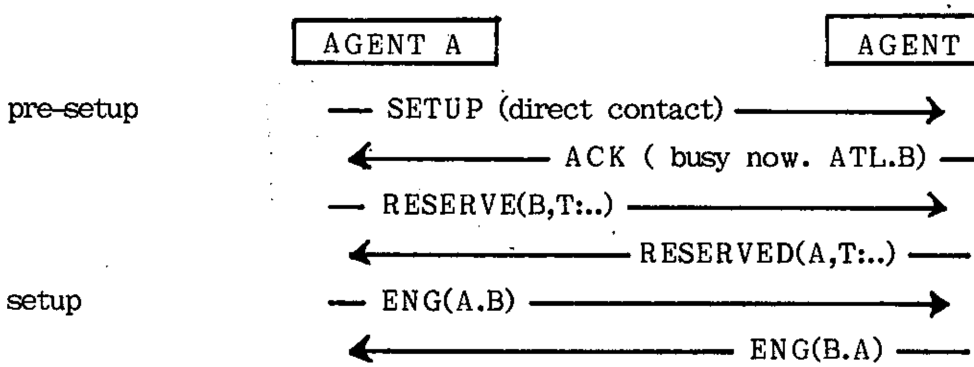
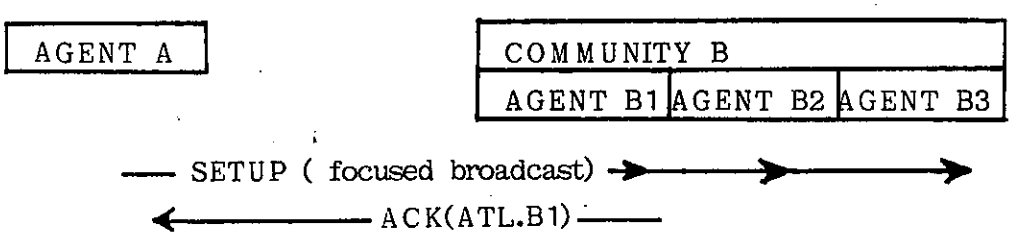


Fig. 9

Fig. 10 shows an example of a focused broadcast to set up a conversation. The example, along with Fig. 2, has fully explained itself, hence we won't make further comments. We will also skip the description of the universal broadcast of office modes since it can be derived with a similar modification on the corresponding micro mode.



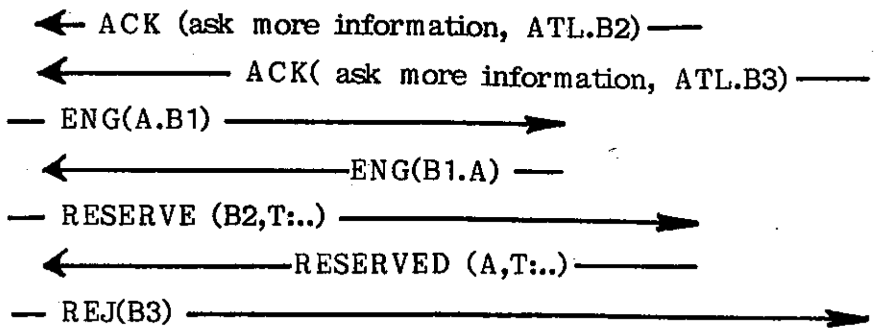

ACK (ask more information, ATL.B2)
ACK(ask more information, ATL.B3)
ENG(A.B1)
ENG(B1.A)
RESERVE (B2,T:..)
RESERVED (A,T:..)
REJ(B3)

Fig. 10

5. CONTRIBUTIONS

The development of the agent society model makes contributions to many fields, for example, office automation, artificial intelligence, cognitive science, and so on. The following arguments summarize these contributions.

1. Contributions to office automation: We have developed an agent society theory for office systems. The agent society theory bases itself on the observations of office behavior and tries to capture the spirit behind the behavior. It provides a uniform and general framework for modeling agents and offices. Under the framework one may capture declarative (static) and procedural (dynamic) knowledge of an office both in the set up stage and the evolving stage of an office information system. The theory hence supports any level of office automation on various types of offices and can be customized to various local application areas.

2. Contributions to artificial intelligence: Several aspects of contributions to artificial intelligence deserve notice.

2.1 Contributions to expert systems: We have proposed a micro society theory for an agent, which can be an intricate person, a specific expert system, or a degenerated knowledge source. In other words, the micro society provides a general framework for the architecture of expert systems[STEF82]. Since we have intentionally skipped the specification of knowledge representation for each knowledge source, the modeling level of the micro society is high enough to accommodate various kinds of knowledge sources. The common communication interface of these ks's is through the micro knowledge exchange system which provides micro knowledge exchange mechanisms to deal with all problem solving requirements for the problem solving process of those ks's. Besides leaving the knowledge representation of each ks to be a decision of implementation could be more suitable to the versatile requirements of different kinds of knowledge sources in an expert system developing process.

2.2 Contributions to distributed problem solving: We have proposed a set of knowledge exchange mechanisms to facilitate the cooperative problem solving for agents of an office, which in fact is a distributed problem solving system. [DAVI83] has pointed out four basic steps to solve a problem by distributed problem solvers. They are planning, distribution, solving, and synthesis. Specifically, planning refers to decomposing a problem into several subproblems, distribution refers to distributing subproblems to appropriate problem solvers, solving refers to solving the subproblem by each problem solver, and synthesis refers to synthesizing the result from all problem solvers. The proposed knowledge exchange mechanisms seem to provide a suitable strategy to commit each stage of the distributed problem solving. An office information system based on this model is being developed, which is expected to demonstrate this point.

3. Contributions to cognitive science: The micro society theory used to model agents also make a contribution to cognitive science. The basic idea is this: We believe that our mind is a micro society with various kinds of knowledge sources. Each knowledge source is of different knowledge processing capability. They are not necessarily uniform in their knowledge representation. But they can communicate with one another through a subtle communication system. We have adoped the micro knowledge exchange system to emulate the communiation system. For example recalling mechanism can be achieved by the micro conversation mode. By this, the micro society may serve as a general framework to understand the reasoning process of human minds.

6. CONCLUSION

We have proposed an agent society model to describe an office. The agent society model may capture as much as knowledge of an office. Besides it can evolutionally grow as offices develop. Some important features that the agent society model owns are listed below.

1. It recognizes the importance of knowledge. The basic idea behind this model is that knowledge is what an office's behavior is based upon. For example an agent is modeled as interacting knowledge sources which cooperatively characterize the behavior of the agent. The behavior of an office is then the cooperative behavior of all agents in the office. This concept reveals the fact that the basic governing source of office behavior is knowledge.

2. A new approach is used to model office entities. Logically all entities in an office are viewed as agents. An agent can be as fancy as a person or as simple as an intelligent form. This agent-oriented approach is intentionally to capture both declarative and procedural aspects of an office and serve as a new technique to attack real-world problems (especially nondeterministic features) existing in an office.

3. A single modeling concept is used both in modeling agents and offices. Basically both of them are modeled as a society of individuals confined by suitable structures and social requlations. This uniformity makes the model easy to understand, simple to follow, and general enough to accommodate various types of offices [TEGE83]. Besides the knowledge exchange mechanisms introduced in the model make it flexible enough to recruit new agents (new knowledge sources), foster new regulations, and reconfigure office structures, all of which make the model evolve as offices grow up.

4. The model tries to mimic offices through the modeling of their behavior. This attempting to capture the spirit of the behavior of an office in some sense makes the modeling level of the model high enough to evolve as the office technology improves [LOCH83]. For example in the agent society model we only provide knowledge exchange protocols like muse, conversation, and session, rather than talk about what technology is used in what knowledge exchange mechanism. One may use teleconferencing in session protocols, but

in general not necessarily so. Hence the model not only can accommodate various kinds of offices as pointed out above, but also can survive various kinds of office technology.

REFERENCES:

[AIEL84] Aiello, L., Nardi, D. and Pauti, M., "Modeling the Office Structure: A First Step Towards the Office Expert System," ACM Proc., 1984, 25 - 32.

[BARB82] Barber, G., "Supporting Organizational Problem Solving With a Workstation," ACM SIGOA Proc., June 1982.

[BARR80] Barr, A. and Feigenbaum, E.A.(eds.), "The Handbook of Artificial Intelligence," (Vol. 1, 2 & 3, Computer Science Department, Stanford Univ. Stanford, C.A. 1980.)

[BAUE79] Bauer, M.A., "Programming By Examples," Artficial Intelligence 12 (1979), 1 - 21.

[BYRD82] Byrd, R. J., Smith, S. E. and de Jong, S. P., "An Actor-Based Programming System," ACM SIGOA Proc., June 1982.

[CHAN81] Chang, S. K., "A Model for Information Exchange," International Journal of Policy and Information Systems, Vol. 5, No. 1981, 67 - 93.

[CHAN82] Chang, J. M. and Chang, S. K., "Database Alerting Techniques for Office Activities Management," IEEE Trans. on Communication, COM - 30, No. 1, Jan. 1982.

[CHEN84] Cheng, P. W., "Design of Interactive Office Automation System," Master Thsis, NTU, R.O.C., 1984.

[CHOU83] Chou, M. S., "Office Information System Requirement Specification and Analysis," Master Thsis, NTU, R.O.C., 1983.

[DAVI83] Davis, R. and Smith, R. G., "Negotiation as a Metaphor for Distributed Problem Solving," Artificial Intelligence 20 (1983) 63 - 109.

[DAVI84] Davis, R. and Buchanan, B. G., "Meta-Level Knowledge," in Rule-Based Expert Systems, Buchanan, B. G. and Shortliffe, E. H., (eds.), Addison-Wesley, Reading, MASS. 1984.

[DU84] Du, W. C., "Office Information System Design Methodology for Interactive Office Automation System," ICS Proc. Taipei R.O.C. 1984.

[ELLI79] Ellis, C. A., "Information Control Nets: A Mathematical Model of Office Information Flow," ACM Proc. Conf. Simulation, Modeling and Management of Computer Systems, Aug. 1979, 225 - 240.

[ELLI80] Ellis, C. A. and Nutt, G. T., "Office Information Systems and Computer Science," ACM Comput. Surveys, 12, 1, 3, 1980.

[ELLI82] Ellis, C. A. and Bernal, M., "Officetalk - D: An Experimental Office Information System," ACM SIGOA Proc., June 1982.

[ERMA80] Erman, L. D., Hayes - Roth, F., Lesser, V. R. and Reddey D. R., "The Hearsay - II Speach Understanding System: Integrating Knowledge to Resolve Uncertainty," Comput. Surveys 12, 2, 1980, 213 - 253.

[GIBB82] Gibbs, S, J., "Office Information Models and the Representation of 'Office Objects'," ACM SIGOA Proc., June 1982.

[HAYE83] Hayes - Roth, F., Waterman, D. A. and Lenat, D. B., "An Overview of Expert Systems," in Building Expert Systems, Hayes - Roth, F., Waterman, D. A. and Lenat, D. B.(eds.), Addison-Wesley, Reading, MASS. 1983.

[HEWI77] Hewitt, C., "Viewing Control Structure as Patterns of Passing Messages," Artificial Intelligence 8 (1977) 323 - 364.

[HONG82] Hong, Y. C., Ho, C. S., Ho, Y. W. and Kuo, T. S., "Office Systems Modeling, Analysis, and Design," ICS Proc., R.O.C. 1982, 680 - 696.

[KING83] King, K. J. and Maryanski, F. J., "Information Management Trends in Office Automation," IEEE Proc., Vol. 71, No. 4, April, 1983, 519 - 528.

[LADD80] Ladd, I. and Tsichritzis, D. C., "An Office Form Flow Model," NCC Proc., 1980, 533 - 539.

[LEBN82] Lebnsold, J., Radhakrishnan, T. and Janorski, W. M., "A Modeling Tool for Office Information Systems," ACM SIGOA Proc., June, 1982, 141 - 152.

[LENA82] Lenat, D. B., "The Nature of Heuristics," Artificial Intelligence 19 (1982) 189 - 249.

[LOCH83] Lochovsky, F. H., "Improving Office Productivity: A Technology Perspective," IEEE Proc., Vol. 71,No. 4, April, 1983, 512 - 518.

[MACF83] Macfariane, D., "What You Get When You Buy Office Automation," Datamation, Feb., 1983.

[MELL84] van Melle, W., Shortliffe, E. H. and Buchanan, B. G., "EMYCIN: A Knowledge Engineer's Tool for Constructing Rule - Based Systems," in Rule - Based Expert Systems, Buchanan, B. G. and Shortliffe, E. H., (eds.), Addison - Wesley, Reading, MASS. 1984.

[MINS79] Minsky, M., "The Society Theory of Thinking," in Artificial Intelligence: An MIT Perspective, Winston, P. H. and Brown, R. H.(ed.) Vol. 1, The MIT Press, MASS. 1979.

[NEWE82] Newell, A., "The Knowledge Level," Artificial Intelligence 18 (1982) 87 - 127.

[NEWN80] Newman, W. M., "Office Models and Office System Design," in Integrated Office Systems, Naffah, N. (ed.), North - Holland, 1980.

[NILS80] Nilsson, N. J., "Principles of Artificial Intelligence," (Tioga, Palo Alto, C. A. 1980.)

[RICH83] Rich, E., "Artificial Intelligence," (McGraw - Hill, N. Y. 1983.)

[SHOR84] Shortliffe, E. H., "Details of the Consultation System," in Rule - Based Expert Systems, Buchanan, B. G. and Shortliffe, E. H., (eds.), Addison - Wesley, Reading, MASS. 1984.

[STAL79] Stallman, R. M. and Sussman, G. J., "Problem Solving About Electrical Circuits," in Artificial Intelligence: An MIT Perspective, Winston, P. H. and Brown, R. H.(ed.) Vol. 1, The MIT Press, MASS. 1979.

[STEF81] Stefik, M., "Planning and Meta-Planning (MOLGEN:Part 2)," Artificial Intelligence 16 (1981) 141 - 170.

[STER82] Stefik, M., Aikins, J., Balzer, R., Benoit, J., Birnbaum, L., Hayes-Roth, F. and Sacerdoti, E., "The Organization of Expert Systems: A Tutorial," Artificial Intelligence 18 (1982) 135 - 173.

[TEGE83] Teger, S. L., "Fators Impacting the Evolution of Office Automation," IEEE Proc., Vol.71, No. 4, April, 1983, 503-511.

[WALT82] Waltz, D. L., "Artificial Intelligence," Scientific American, Oct. 1982.

[ZISM77] Zisman, M. D., "Representation, Specification and Automation of Office Procedures," Ph. D. Dissertation, Wharton School, Univ. of Pennsylvania, 1977.

[ZLOO81] Zloof, M. M., "QBE/OBE: A Language for Office and Business Automation," IEEE Computer, may 1981.