

# Digital Library System by Advanced Distributed Agent Platform

Young Im Cho

Dept. of Computer Science, Pyongtaek University

yicho@ptuniv.ac.kr

## Abstract

I propose a personalized digital library system (PDL) based on an advanced distributed agent platform. The new platform is developed by improving the DECAF (Distributed Environment-Centered Agent Framework) which is one of the conventional distributed agent development toolkits. Also, a mobile ORB (Object Request Broker), Voyager, and a new multi agent negotiation algorithm are adopted to develop the advanced platform. The new platform is for mobile multi agents as well as the distributed environment, whereas the DECAF is for the distributed and non-mobile environment. From the results of the simulation, the searched time of PDL is lower, as the numbers of servers and agents are increased. And the user satisfaction is four times greater than the conventional client-server model. Therefore, the new platform has some optimality and higher performance in the distributed mobile environment.

**Key words :** Distributed agent platform, digital library

## I. Introduction

Recent developments of the internet and network technologies evoke the technical change of the data processing from a conventional centralized and local processing system to the distributed processing system. The research about this network and the various approaches have been studied in order to efficiently manage mutual operations in such a network environment.

Many studies have been actively carried out in a distributed processing environment by using agent systems for efficient network management. An agent system has the following characteristics: multi agents in the distributed environments promote efficiency by solving one problem through any cooperation. Each agent manages the problem by dividing a common work into the number of agents, or each agent manages it independently, and then they solve the problems by analyzing the results. In addition, it has some advantages such that intelligent agents reflecting the tendency of users make no limitation of movement in a network, and it remarkably decreases the network traffic [1, 2,3,4].

There are so many application areas of agents in the real world. One of these areas is a digital library system. The digital library is called an *electronic library* or a *virtual library*. This is a library developed to replace the conventional library, in order to serve information from databases on the web to users, according to the development of computers and the related fields.

However, there are several problems in the searching of data of the existing digital libraries. First, as the searching method is one dimension and distinguishes the existence of the searching keyword from the database, the result is very

simple. Secondly, the results may contain unnecessary information under a condition that was not given the prior information about the user. Thirdly, whenever a client connects to the servers, he has to receive the certification and be under the dominant power of the influence of network.

To overcome such problems, I proposed a new platform of mobile multi agents for a personal digital library in this paper. For developing a new platform, I combined the existing DECAF (Distributed Environment Centered Agent Framework) multi agent framework [5] with Voyager which is a mobile ORB (Object Request Broker). Also a new negotiation algorithm and a scheduling algorithm are proposed, so that I developed a PDL (*Personalized Digital Library System*) using this new platform. Although the partial studies for a personal digital library have been carried out, there has been none about the integrated and systemized personal digital library.

For the higher relationship among searched documents from mobile servers, an unsupervised neural network is applied. For the user's preference, some modular clients are applied to a neural network. A multi agent platform and a mobile agent platform are combined to develop a new mobile multi agent platform so as to decrease a network burden. Also, a new negotiation algorithm and a scheduling algorithm are activated for the effectiveness of PDL.

PDL is different from the electronic paper service system which is supplied only to members. It is a more intelligent system that is capable of establishing a database in users' computer by learning the interests of those individuals. In this paper, I tried to set up the theoretical structure of the multi mobile agents and develop an algorithm of the modified intelligent negotiation agent for inducing interaction among multi agents.

This paper is composed of five chapters. Multi agents and DECAF framework is explained in chapter 2. PDL based on a new mobile multi agent platform is explained in chapter 3.

The simulation results of PDLs are explained in chapter 4, and finally the conclusions are in chapter 5.

## 2. Distributed Agent System

### 2.1 DECAF and Voyager

DECAF (Distributed Environment-Centered Agent Framework) is a conventional framework to design a lot of intelligent agents [5]. DECAF is a kind of operating system including agent communication, planning, scheduling, monitoring, coordination, diagnosis, and learning among agents. DECAF makes a socket program by itself, and presents some building blocks which makes messages and communicates between agents. Therefore, users or programmers can produce agents without having some knowledge about API approaches. Also, users or programmers do not need to make the communication codes directly to communicate among agents. DECAF produces a KQML protocol automatically which sends messages and searches other agents and interacts between agents. Agent systems have been developed using various languages and platforms, and they are classified into so many types by purpose. In DECAF, many agents' tasks are divided by both GPGP (Generalized Partial Global Planning) and TAEMS (Task Analysis Environment Modeling and Simulation) algorithms.

GPGP is for improving of PGP which acts as a coordination algorithm of multi agents [6]. The first advantage of GPGP is that it reduces the system overhead which occurs by overlapping interaction among agents. And the second advantage of GPGP is that it is independent from some specific domain areas. Therefore, GPGP can make heterogeneous multi agents system having different functions. User's requirements can be decomposed by GPGP, and be structured by TAEMS [7]. In TAEMS task structure, the root task can be decomposed into subtasks, and the subtasks can be decomposed into methods. The leaf node acts as a method that becomes an activated element.

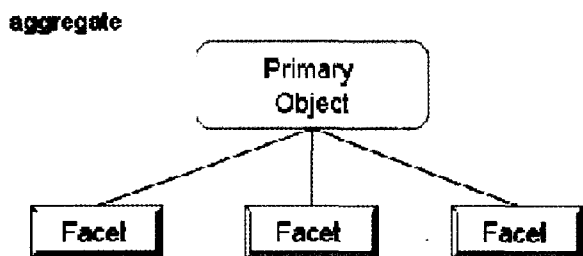


Figure 1. Aggregation with 3 Facets in Voyager

Voyager [8] is a distributed mobile agent's framework for developing agent's applications, whereas DECAF is a non-mobile agent's framework. Voyager is an interactive framework with Java programming. Also, Voyager can activate any Java class in remote sites, and it makes use of network bandwidths effectively. Voyager has 3 facets in Figure 1. Voyager has dynamic aggregation properties using facets. Facet is an sub object, which is adhere it to the primary

object during run time. And then they act as an object. This is called an aggregation.

### 2.2 Agent based Digital Library

A digital library serves a lot of information on-line [9,10]. The advantages of digital libraries are user friendly, on-site service and accessibility. However, in case of not having standardized platform, the search of heterogeneous information from digital libraries may be hard, as well as impossible. If it does not have or learn about the user's information, unnecessary or useless information will appear in the searched results from the digital library.

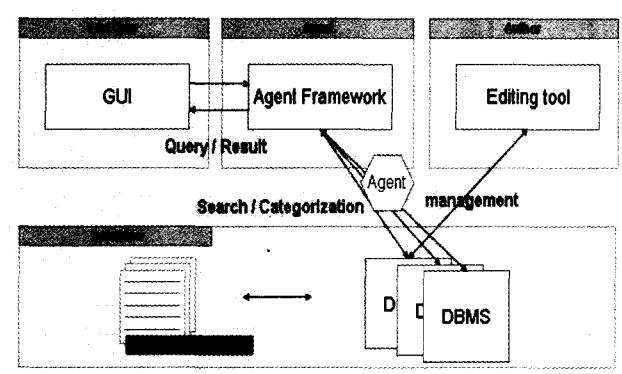


Figure 2. Agent based Digital Library

Figure 2 shows the concept of an agent based digital library which is proposed in this paper. This is based on the proposed mobile multi agent framework to search many servers concurrently, using multi agents. Also user's profile can be produce into a database in this system.

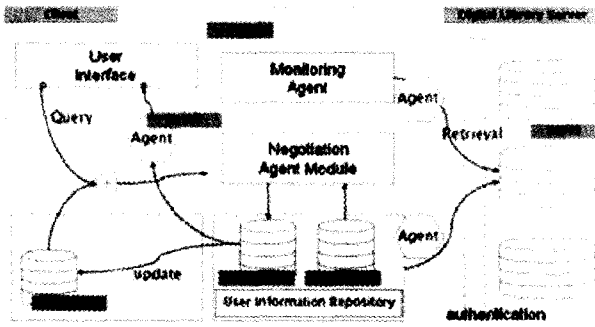
## 3. Digital Library System by Agent Platform

### 3.1 PDLs Overview

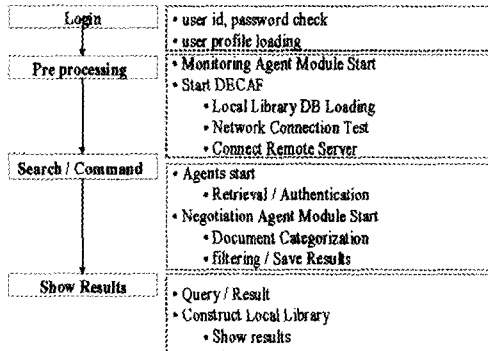
The proposed system, in Figure 3, is a Personalized Digital Library System (PDLs) based on a new multi mobile agent platform. The system combines a mobile system and a distributed processing system to make an optimization of behaviors in a distributed environment.

To establish a distributed environment, DECAF is used, and to activate a mobile framework, Voyager is used here.

The PDLs is composed of two parts: client group and server group. The client group is composed of three modules. First, a user interface module lets users make use and control the library. Second, a user profile control module learns the user's preferences by neural network (SOM [11]), and makes databases accordingly. Third, a PLA (Personal Library Agent) module makes multi agents in real time, and searches information from the library according to the user's profile, and stores the searched results into a database. The interactions among detailed modules in PDLs are explained in Figure 2(b).



(a) PDL S Overview



(b) Interactions among Detailed Modules in PDL S  
Figure 3. PDL S Structure and Interactions among Modules

### 3.2 PLA

As shown in Figure 2, PLA has two modules and two databases. The monitoring agent module is composed of Voyager and DECAF, and it monitors the agents' movements and controls their executions. When the user's requirements are transferred to the PLA, the monitoring agent module checks whether the servers are available or not. After that, it makes some agents, and passes them to the servers. The searched results are saved in a temporary repository. They are filtered by negotiation agents, and the final results are saved in the result repository.

In the proposed platform, the relationship among multi agents in negotiation agent module is in Figure 4. Agent\_Task Group generates Task\_1, Task\_2 and Task\_3 according to Agent 1, Agent 2, and Agent 3. Tasks are automatically decomposed into methods and do their assigned tasks. Each method has five types of methods' relationships. Add\_R is to add the results of actions to the results of the other methods. Activate\_R is to let the running method run continuously. Compensate\_R is the relationship that the results among methods need to be compensated. Replace\_R is to replace the results of receiving methods with the results of sending methods. Contradict\_R is to disregard the results of receiving methods. Also, there are lots of relationships between methods and tasks and between methods and resources, such as Enable, Facilitate, Produce, Consume, Limits and so on. In the negotiation algorithm, if the agents in the same levels do the different actions, then max operation is operated to produce the output of the agents, and if the agents in the lower levels

do the different actions, then min operation is operated.

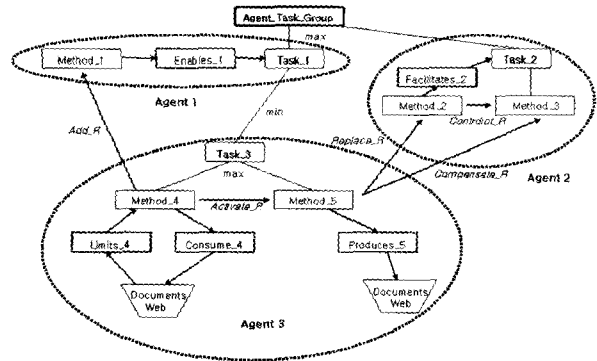


Figure 4. Negotiation Algorithm among Multi Agents

### 3.3 User's Profile

The construction of the initial user's profile is constructed by the user's first input information. According to the user's searched results, PDL S endows the user's keywords to weight values, and updates user's profile information by SOM (self organizing map) network in real time in Figure 5.

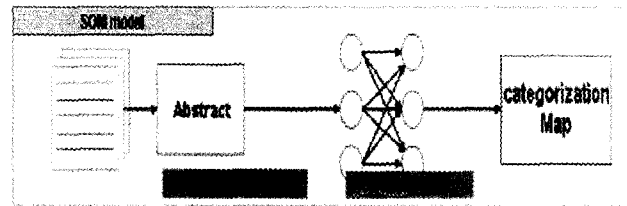


Figure 5. SOM Network

The user interface is composed of four windows. The user's window is for entering the user's information and for recalling the user's profile from databases. The monitoring window is for checking agent's activities. The remote window shows the final results, and finally the local window shows the constructed hard disk information of the user's computer by PDL S.

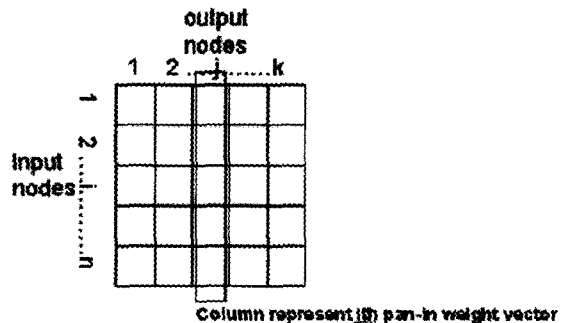
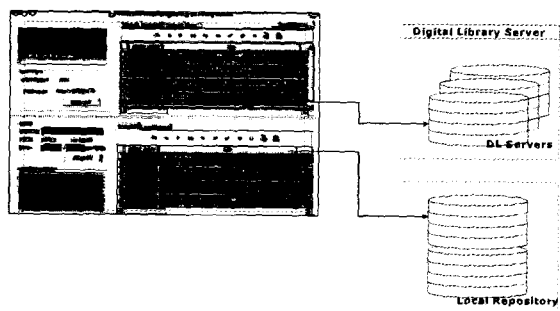


Figure 6. Weight Vector in SOM

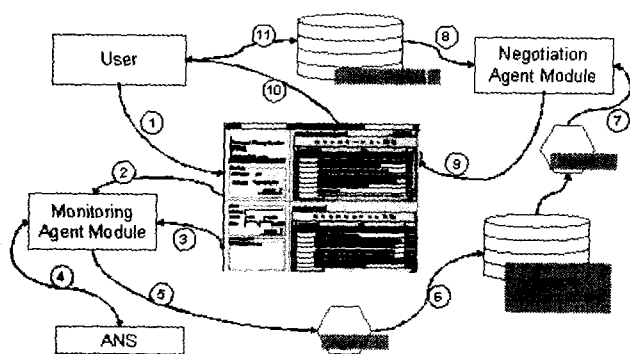
Figure 6 is weight vectors in SOM network. In SOM,  $j$  th fan-in weight vector, which has the smallest Euclidian distance among other vectors, is the winner take all of  $i$  th input vector. The weight update function in SOM is the followings.

### 4. Simulation Results

The user interface is composed of four panes, and each pane is interconnected. The login window and query window is for user login. The user pane is for checking the agents' activation states including monitoring. The remote pane is for representing the information of remote servers and searched results. The local pane is for representing the building states of a local library. The scenario for PDLs simulation is in Figure 7.



(a) User Interface and Module Relations



(b) PDLs Simulation Scenario

Figure 7. PDLs Interface and Simulation Scenario

PDLs scenario is the followings.

- ① user login
- ② PDLs execute the monitoring agent(MA), and MA checks remote digital library. The user profile is reading from database.
- ③ Queries is sending to PLA according to user's query
- ④ MA produce search agent and register to ANS.
- ⑤ The produced agent 1 is moved to remote library.
- ⑥ The agent 1 is searching by the searching condition from PLA
- ⑦ The searching results is sent to the negotiation agent(NA) with library name, index value, and abstract of documents.
- ⑧ NA can be clustered the results using SOM
- ⑨ The clustered values are displayed the remote pane of PDLs.

- ⑩ User can acquire the necessary information from remote pane, and catch them. User can move the necessary information to user's local computer.
- ⑪ The update of user profile is performed.

Table 1 shows keywords in PDLs. The keyword from <http://citeseer.hj.nec.com> according to frequencies of users.

Table 1. Keywords in PDLs

Category Number	Agent System	Neural Network	Digital Library
1	framework	learning	virtual
2	mobile	neural	library
3	java	architecture	indexing
4	multi	network	structure
5	personalized	simulation	distributed
6	distributed	layer	agent
7	environment	agent	retrieval
8	neural	artificial	autonomous
9	architecture	associative	neural
10	autonomous	algorithm	multi

To represent input vector, each word is assigned as a vector, and it makes 30 dimensional input vectors. The learning process is the followings. 3 remote servers are used here, and each server is developed a database using 100 documents without the classification of category, namely randomly. The adopting parameters in SOM is the followings.

- Output dimension : 5 \* 5 dimension
- Neighborhood radius : 2
- Number of training : 500 times
- Training rate : 0.05

The specification of client/server in this simulation is CPU 1.8 GHz and windows NT using MySQL.

From the simulation results of SOM network, it is classified according to the above categories like Figure 8.

36	1			7	21	1								7
22	50	4	29		7	36		3			10	4	26	
		32					15				6	15		
25	60	1	4				2				24		2	
10	16			8	9	6				1				5

(a)Neural Network (b)Agent Systems (c)Digital Library

Figure 8. Simulation Results in SOM

Figure 8 shows that there is not distributed in the simulation results. However, The ratio of documents of each category shows some trend. The numbers in box represent the number of documents about categories.

The searching times between the proposed PDLs and the traditional client-server model are shown in Figure 9.

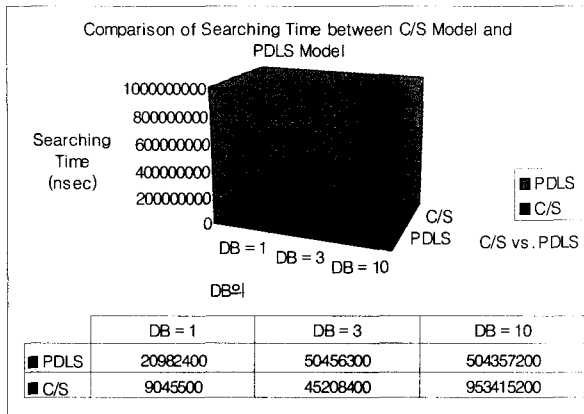
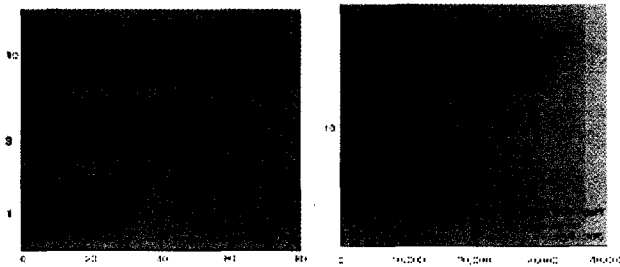


Figure 9. Simulation Results

As time passed, PDL showed a faster search time as well as a much safer search than the client-server model. The result showed that as the numbers of servers were increased, the searching time was decreased in PDL. The searching time is reverse proportionally increased according to the agent numbers.

Figure 10 shows the maximum CPU usage and the maximum memory usage during execution of PDL.



(a) Max CPU usage (b) Max Memory Usage  
Figure 10. Max CPU and Memory Usage

In Figure 10, the libraries based client-server model are not changeable according to the increasing of servers. However, PDL using mobile multi agents in increases the CPU usage during execution. This is for the reproduction of multi agents in PDL, whereas client-server model is using remote procedures. However the gap is not serious now. Also, as the performance of CPU is powerful, the gap can be overcome.

### 5. Conclusions

In this paper, I proposed a Personal Digital Library System. PDL is designed based on a new mobile multi agent platform using Voyager and DECAF agent framework. The new platform is a hybrid system of a mobile and a distributed system in order to achieve an optimality in distributed Environments, and to make it operate effectively by the propose of a new negotiation algorithm and a new scheduling algorithm. From the simulation results of PDL, the performance and the user's satisfaction of this system is higher

than any other information search systems as of now. Also, as the numbers of servers and agents are increased, the searched time of PDL is lowered. And the degree of the user's satisfaction is increased four times than the conventional client-server model. In the future, PDL needs to be compensated in order to be activated in the real world.

### References

- [1] David C., Colin H., Aaron Kershenbaum, "Mobile Agents: Are They a Good Idea?", *IBM Confidential*, March 13, 1995
- [2] Russell, *Artificial Intelligence a Modern Approach 2/E* Chap 2. *Prenticehall International Edition.*, 1996
- [3] Hyacinth S. Nwana, "Software Agents: An Overview" , *Knowledge Engineering Review*, 11(3), pp.1-40, Sept 1996.
- [4] Finin T., Fritzson R., McKay D., and McEntire R., "KQML as an agent communication language," *Proc. of CIKM '94*, pp 126-130, 1994.
- [5] John R. Graham, Keith S. Decker, Michael Mersic, DECAF - A Flexible Multi Agent System Architecture, *Autonomous Agents and Multi-Agent Systems*, 2003
- [6] Keith S. Decker, Victor R. Lessor Generalizing the partial global algorithm. *Intelligent Cooperative Information Systems*, 1(2), pp.319~346. 1992
- [7] Keith S. Decker, Task Environment Centered Simulation, Ph..D. thesis, 1996
- [8] ObjectSpace, Voyager Core Technology 2.0 User Guide, ObjectSpace, <http://www.objectspace.com>. 1998
- [9] Sanchez, "AGS: Introducing Agents as Services Provided by Digital Libraries", *2nd ACM International conference on Digital Library*, Philadelphia, Penn, July, 1997
- [10] Jonas Holmstrom, "A Framework for Personalized Library Services", <http://www.abo.fi/~jholmstr/publications/personalized.pdf> October 2002
- [11] Kohonen, *Self Organizing Feature Map*, <http://www.nd.com/models/sofm.htm>, 1995



#### Young Im Cho

Dept. of Computer Science, Assistance Professor. Dept. of Computer Science, Korea University, Bachelor(1988). Dept. of Computer Science, Korea University, M.Sc. (1990). Dept. of Computer Science, Korea University, Ph.D (1994). 1995-1996 Samsung Electronics Co., Dept. of Multimedia, Senior Researcher.

1999-2000 University of Massachusetts, at Amherst, Dept. of Computer Science, Post-doc. 2001-Present Korea Engineering Education Society, Editor. 2003-Present Korea Fuzzy and Intelligent System Society, Editor. 2003-Present Korea Woman Information Society, Administrator.

e-mail : yicho@ptuniv.ac.kr