

# Noah: Environment for Distributed Agents

## (1) — Basic Concepts —

3M-3

Hiroshi KOZUKA, Fumiaki SATO, Kazuya MIYAZAKI,  
Hisao FUKUOKA, Junichiro TSUJI

Computer & Information Systems Laboratory, Mitsubishi Electric Corporation

### 1 Introduction

Distributed computing system has been getting more and more popular under the wave of down-sizing. In such environments, high cost performance and reliable services are expected. We propose Noah: Network oriented applications harmony as an architecture of distributed environment. Noah is suitable for resource allocation, fault-tolerant applications, system management, and applications for cooperative works.

### 2 Issues of distributed computing system

Distributed computing system has several new problems that a conventional centralized system does not.

- System administration :  
In distributed environment, management tasks of addresses, names, peripheral devices, and networks are much more complex than in centralized systems.
- End user's operation :  
As resources and services are distributed, end users have to be conscious of their physical locations. Furthermore, it is necessary to change the commands or sequences of operations on the resources and the services according to their locations.
- Application development :  
In order to describe applications that effectively use the distributed environment, it is necessary to distribute functions of the applications and to program some complex processes such as communication among distributed modules, synchronization of them, and access control of resources. Besides, adding more values such as concurrency of the modules and multiplicity of the functions for improvement of the reliability makes programming tasks extremely complicated.

To cope with these problems, some countermeasures are taken. On the system management of distributed environment, central management systems are constructed

to manage resources using centralized management information sent by system management processes at all the computers. But such systems are very brittle against a damage of the central management system.

On distributed application development, development environments such as distributed object-oriented languages succeed in giving the users some transparency of access to networks and resources. But in conventional environments, relations between objects are fixed, so it is impossible to add new services flexibly. Also it is impossible to change the way to provide the services.

### 3 Concept of Noah

Based on the issues described above, we propose Noah — Network oriented applications harmony, the cooperation mechanism among applications in distributed computing environments. The major objectives of Noah are to provide the following:

- Programming environment for autonomous program modules i.e. agents.
- Cooperative mechanism for multiple agents.
- Communication mechanism for distributed agents.
- Management mechanism to control agents.

These features improve fault tolerance of systems and availability of services, and make it easy to develop reliable and efficient software using potentiality of services of distributed environment.

#### 3.1 Applications harmony

“Applications harmony” means an application control mechanism to solve some constraints defined in the “field” (operational environment) at which applications or functional units work and interact autonomously.

“Cooperation field” consists of a goal, an evaluation criteria to measure a distance to the goal, and agents that achieve the goal cooperatively. The cooperation field specifies a cooperation process for agents to achieve the

Noah: Environment for Distributed Agents (1) — Basic Concepts —

Hiroshi KOZUKA, Fumiaki SATO, Kazuya MIYAZAKI, Hisao FUKUOKA, Junichiro TSUJI  
Computer & Information Systems Laboratory, Mitsubishi Electric Corporation.

goal. According to the specified process, agents work cooperatively. In addition, the cooperation field specifies the members of agents joining some cooperational tasks, qualifications for the membership, and the process for the participation.

The term "agent" or "cooperation" has different meanings since it has been used in many different contexts. Nakajima et al.[4] defines these terms as follows:

**Agent:** An agent is a program unit which has a certain capability. Although it is similar to the concept of a module, it does not imply information hiding. An agent is not necessarily intelligent. Contrary to the traditional concept of a program unit that performs a predefined task upon invocation, an agent works autonomously to some extent.

**Cooperation:** Cooperation means consistent behavior of agents. It includes cooperation, negotiation, compromise and so on.

### 3.2 Conceptual Model of Noah — Lake/Canal Model

The fundamental concept of Noah computing architecture is based on the Lake/Canal model shown in Fig.1. In this model, a distributed application consists of several active processing units, i.e. agents, and each agent takes actions to its environment to proceed on its task. Basically, the Lake/Canal model is an improvement on the Linda model[5] developed at Yale University.

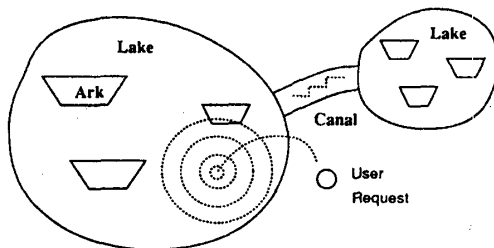


Figure 1: Lake/Canal Model

The Lake/Canal model consists of the following abstract entities.

**Lake:** A Lake is a field where the communication among agents takes place. A set of computers of the same architecture form a Lake.

**Canal:** A Canal is a gateway which connects two Lakes. It performs the data/message transformation to connect Lakes of different architecture.

**Ark:** An Ark is a function unit of an application and can be regarded as floating on a Lake. It is an active agent, which performs its task sensing the status change of the Lake and let the Lake transit to the other state by giving the processing results to it.

Applications in Noah can cooperate by communication via the Lake, the communication field. Since agents, which compose an application, can perform their own tasks autonomously to some extent, the cooperation can be achieved in a relatively easy manner.

Fig.2 shows the position of Noah in a layered software architecture as a vertical section of "Lake Noah."

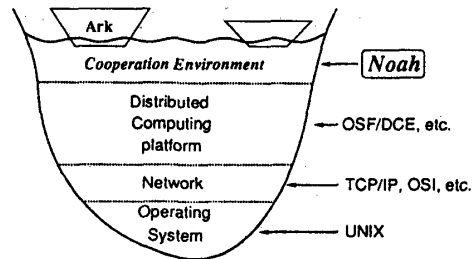


Figure 2: Vertical section of Lake Noah

The Noah architecture provide hierarchical cooperation fields for the communication among agents. Each cooperation field supports inheritance of the cooperation mechanism[2].

## 4 Summary

This paper proposed Noah: Network oriented applications harmony as an architecture of distributed environment. Noah is suitable for resource allocation, fault-tolerant applications, system management, and applications for cooperative works.

Noah makes it easy to develop reliable and efficient software using the potentiality of the distributed environment services. The examples of services will provided by Noah environment are described in [3].

Future work includes design of programming interface, language and development environment. Design and development of a "shell" to manage interactions between users and the environment are also for future work.

## References

- [1] H. Kozuka et al. "An Architecture of Distributed Computing Environment — Noah: Network Oriented Applications Harmony," Proc. IPSJ 58th DPS, 1993.
- [2] F. Sato et al. "Noah: Environment for Distributed Agents (2) — Architecture Design —," Proc. IPSJ 46th meeting, 1993.
- [3] K. Miyazaki et al. "Noah: Environment for Distributed Agents (3) — Applications —," Proc. IPSJ 46th meeting, 1993.
- [4] Nakajima et al. "Current Status of Researches on Cooperative Architecture," (in Japanese), Electrotechnical Laboratory Research Report No.221, 1991.
- [5] Sudhir Ahuja, Nicholas Carriero, David Gelernter, "Linda and Friends," IEEE COMPUTER, Vol.19, No.8, Aug.1986.