2003 International Symposium on Advanced Intelligent Systems
September 25-28, 2003, Jeju, Korea

A Personal Agent for Combining the Home Appliance Services and Its Learning Mechanism

Yuji Takeda and Kazumi Sakamaki
Tokyo Metro. Industrial Technology Research Institute
Nishigaoka 3-13-10, Kita, Tokyo 115-8586, Japan
Email: Yuuji_Takeda@member.metro.tokyo.jp

Kanemitsu Ootsu, Takashi Yokota and Takanobu Baba
Utsunomiya University
Yoto 7-1-2, Utsunomiya, Tochigi 321-8585, Japan

***Abstract* - In this paper, we propose a new personal agent for generating the combinational services from using history of appliances in the home network environment. In such environment, it is required that flexible services can be provided by combining services of appliances and unskillful users can use these services without knowledge. So, it is needed to satisfy following: (1) combinational services can be suggested automatically and (2) the increase of services can be followed. Then, we propose a new personal agent that suggests combinational services by learning the lifestyle. Its learning mechanism is based on Self-Organizing Map (SOM), and can follow the increase of services. We implemented the the agent, and use history of a user for two weeks was made to learn. As the result, we confirmed that the agent can extract services related with time or location and can suggest combinational services.**

I INTRODUCTION

Recently, the network of home electronics and information terminals, called appliances, is progressing by HAVi[1], ECHONET[2], Jini[3] and so on, and the provided services have become diversified. In such environment, it is required that flexible services can be provided by combining the services that are equipped in appliances, and unskillful users can use these services without knowledge. However, many combinational services are given beforehand by the developers that construct the system components, and it needs much knowledge to customize them according to his/her lifestyle in the present age which their lifestyle have been various.

On the other hand, although the GUI based modeler has been proposed and developed for generating combinational services by unskillful users, it is necessary to understand all services for using this modeler. So, it is difficult to customize services while ubiquitous appliances advance. Therefore, it is needed to satisfy following: (1) combinational services can be suggested automatically and (2) the increase of services can be followed.

Then, we propose a new personal agent that suggests combinational services by learning the lifestyle. This agent is assigned to each user, and resides permanently on network. And the learning mechanism is based on Self-Organizing Map (SOM), and can follow the increase of services by the features of SOM. We implemented the personal agent with above learning mechanism, and use history of a user for two weeks was made to learn. As the result, we confirmed that the agent can extract services related with time or location and can suggest combinational services.

II HOME APPLIANCES ENVIRONMENT

Fig. 1 shows an image of the home networked appliances environment. Many appliances that include sensors and lights are connected by network, and each appliance provides one or more services. For example, a phone provides call service, address book service, fax service, and so on. So, We can use the phone as a printer by connecting a display to the phone, and use the display as a memo pad by connecting the call service to the display through a voice to text conversion service. Also, Not only content data like image but also event triggers, such as a switch of the light, can be made to cooperate, and it can enable remote control such as water of a bath is boiled while going home.

In the home environment, unskillful user must be able

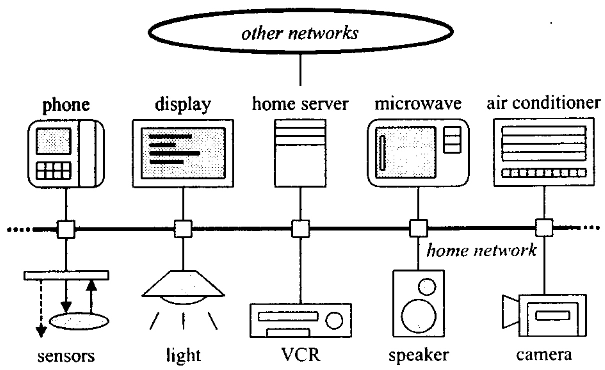

Fig. 1 Home Networked Appliances Environment

to use them without knowledge for using the system. So, many systems[4][5][6] have been proposed and developped in order to realize it. VNA (Virtual Network Appliances) [4] is a system for managing networked appliances, and can make virtual networks of services by defining function templates described in XML. Moreover, it can enable plug and play, and can provide dynamic connection services. uBlocks[5] is a system that a user can construct ubiquitous applications, and the applications can be constructed by the GUI modeler. iHOME[6] is a multi-agent system that can manage appliance resources dynamically, and controls QoS on multiple user environment.

However, in these systems, a user needs much knowledge for customizing them, and it is impossible to combine the services when the appliances increase because the user must know about all services.

III. SYSTEM ARCHITECTURE

For enabling combinational services by unskillful users, it is needed to raise connection flexibility of services, to make virtual networks, and to personalize services to be used. Then, we wrap each function by appliance management agent (AMA), virtual network management agent (VNMA), and personal agent (PA) respectively as shown in Fig. 2.

VNMA and PA work permanently on the home server, and AMA is assigned to each appliance. AMA not only provides services but also supervises operations of the user. AMA negotiates other AMAs, and sends content data and event triggers. VNMA manages locations of AMAs, and connects input and output ports of services on the network. If the content data type which services can send or receive differs, conversion services are inserted. And, VNMA communicates with VNMAs to manage other networks. One PA exists for one user respectively. PA learns his/her lifestyle, and activates combinational services instead of the user.

By wrapping by agents, the personal information in the multi-user environment can be protected, and the maintenancebility of the system becomes easy. Moreover, Agent Communication Language (ACL) recommended by FIPA[7] can improve the connection ability, and the cooperation with other agent systems can be expected.

IV PERSONAL AGENT

PA consists of rule execution engine, communication unit, user history records, Self-Organizing Map (SOM) [8], service evaluation functions, and scheduler. The communication unit receives a message with service ID and location ID from AMA for acquiring the user situation when using a service. These situations with current time are stored in the use history records at any time. The records are given to SOM at every fixed time as one day/week, and PA learns the lifestyle.

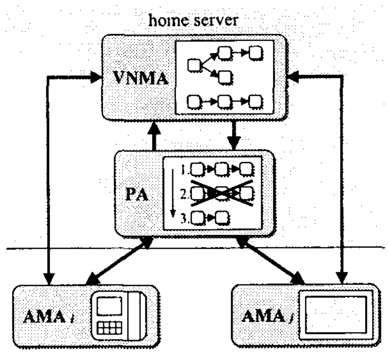

Fig. 2 System Architecture

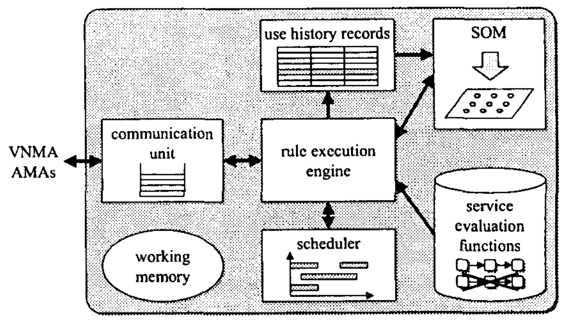

Fig. 3 Personal Agent

The features of SOM are to extract characteristics of input data following certain distributions, and to create a feature map that approximates it. So, relational services can be extracted by association memory. The user operations can be reduced by learning without teacher, and can be followed for the increase of services without changing the previous characteristics.

After learning, PA acquires the user situation when using a service, and extracts a service by giving the situation to the SOM. Then, PA shows the user new combinational service including the extracted service through the AMA. PA orders VNMA to create virtual networks if the user confirms it. The service is registered in the scheduler if always using is permitted, and is activated on decided conditions.

V LERNING MECHANISM

SOM consists of an input layer and an output layer on two-dimensional plane, and all neurons on the output layer are bound to the input layer. The neuron i has reference vector $\boldsymbol{m}_i = \{m_{i1}, m_{i2}, \cdots, m_{in}\} \in R^n$. When input vector $\boldsymbol{x} \in R^n$ is given to the SOM, winner neuron $\boldsymbol{m}_c$ ($\| \boldsymbol{x} - \boldsymbol{m}_c \| = \min_{\text{for all } i} \| \boldsymbol{x} - \boldsymbol{m}_i \|$) is selected, and all reference vectors of neighbor neurons N_c for $\boldsymbol{m}_c$ approach $\boldsymbol{x}$ as follows:

$$\boldsymbol{m}_i(t+1) = \boldsymbol{m}_i(t) + \alpha(t)[x(t) - \boldsymbol{m}_i(t)]$$
$$\alpha(t) = \alpha_0(1 - t/T)$$
$$N_c(t) = N_c(0)(1 - t/T)$$

where α_0, $N_c(0)$, and T represent the initial region of neighbor neurons, the initial learning rate, and the maximum number of learning iterations respectively.

By applying this feature to the user records in section IV, a service approaches another service if it occurs in many cases that they are activated at the same time. So, PA can know a service that should combine another service depending on services, a location, and time.

A. Input Vectors

The user records must be converted to numeric vectors for learning by SOM. Table I shows an example of conversion from the user records to input vectors. The input vector consists of a service vector, a location vector, and time value. The service vector represents whether services are activated or not at that time with 1/0. And the location vector represents whether there is the user at the corresponding location or not with 1/0. The time value is within the range from 0.0 (00:00) to 1.0 (24:00), and is regularized by 1.

B. Service Extraction

The neurons mapped services which are used frequently are gathered, and their distance become short by competitive neighborhood learning of SOM. So, the agent extracts another service which should combine with the service by following algorithm:

1. vectorize current user situation as well as the input vector when the user activate a service. Where the obtained vector is defined as $\boldsymbol{x}_e$.
2. ask for $\boldsymbol{m}_c$ which is the winner neuron for $\boldsymbol{x}_e$ from SOM.
3. calculate the adaptation value $s_a(j)$ of service j by follows:

$$s_a(j) = \sum_{i \in N_e} \frac{\lfloor m_{ij} + 0.5 \rfloor}{(r_{ci} + 1) \| \boldsymbol{m}_i - \boldsymbol{x}_e \|}$$

TABLE I
INPUT VECTORS FOR SOM

active services	location	time	description
0 1 0 0 0 0 0	0 1 0 0 0	0.292	Television@Living [07:00]
0 1 0 0 0 1 0	0 0 1 0 0	0.295	Microwave@Kitchen [07:05]
0 1 0 0 0 0 0	0 0 1 0 0	0.299	Microwave@Kitchen [07:09]
0 0 0 0 0 0 0	0 1 0 0 0	0.333	Television@Living [08:05]
1 0 0 0 0 0 0	1 0 0 0 0	0.833	Light@Entrance [20:00]
0 0 0 0 0 0 0	1 0 0 0 0	0.837	Light@Entrance [20:05]
0 0 0 1 0 0 0	0 1 0 0 0	0.840	Light@Living [20:10]
0 1 0 1 0 0 0	0 1 0 0 0	0.843	Television@Living [20:15]
:	:	:	:

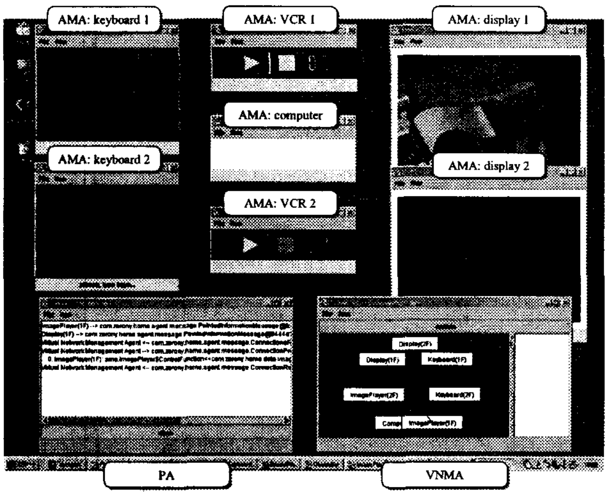

Fig. 4 Screen Shot of the All Agents

4. apply 3. for all services, and extract the service having the maximum value.

Where N_e represents the set of neurons satisfying condition $\| \boldsymbol{x} - \boldsymbol{m}_c \| \le D$ to avoid that the neurons with extremely far distance are selected, and D limits the distance. And r_{ci} represents the distance ($0, 1, 2, \cdots$) between neuron c and neuron i on the map. Finally, services which are close to the current user situation are extracted, and PA suggests combinational services.

VI DISCUSSION

We implemented the agents as shown in Fig. 4 by Jini[3]. The frame of PA shows agent communication logs and extracted services. Then, the use history of services for two weeks was made to learn for confirming that suitable services can be extracted from the user situations. The user has following characteristics:

1. He usually goes out at 08:00 and goes back at 20:00.
2. An air conditioner and television are activated if he stays in a home.
3. Sometimes, he listens to radio before sleeping at 23:00.

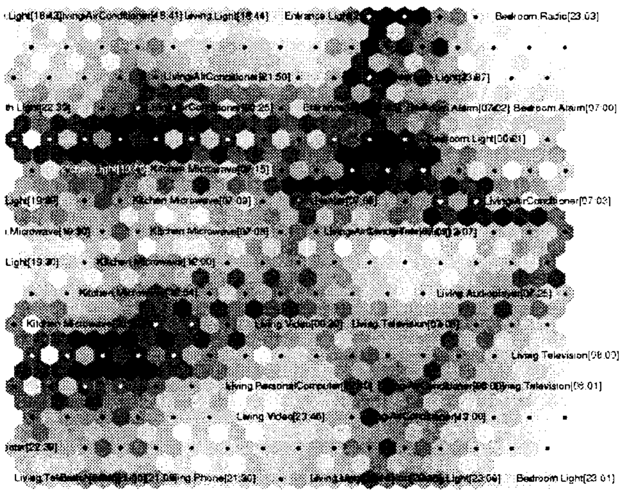

Fig. 5 Acquired Sammon Map

TABLE II
EXTRACTED SERVICES FOR USER SITUATIONS

situation	active services	s_a
(a)	AirConditioner@Living	**4.12**
	Light@Living	3.41
	Light@Entrance	0.70
	Light@Bedroom	0.64
	Television@Living	0.13
(b)	AirConditioner@Living	37.32
	Light@Living	33.96
	Television@Living	**4.70**
	Light@Bathroom	2.40
	Heater@Bathroom	0.27
(c)	Light@Bedroom	4.83
	Light@Entrance	**0.37**

These records were learned by PA that the size of SOM is 16 x 16 neurons, and its learning parameters are $\alpha = 0.2$ and $T = 10,000$ experimentally. Fig. 5 shows the sammon map acquired from the use history by [9].

Next, following situations are given to the agent:

(a) turn on a light at living room at 20:20.

(b) in addition, turn on an air conditioner at the same time.

(c) turn on a light at bedroom at 23:00.

Table II shows each result. In the column of s_a, the numbers with underline represent to be current active services, and the bold numbers represent to be maximum adaptation value without them.

In (a), it is shown to turn on the air conditioner priors to the light. So, it shows that the user characteristic (2) can be reflected and extracted. And, in (b), it is extracted to turn on the television, and it shows (3). However, In (c), only small value for a light at the entrance is acquired, and the user characteristic (3) cannot be caught.

VII CONCLUSION

In this paper, we proposed a new personal agent for generating the combinational services from using history of appliances in the home network environment. The agent acquires the user situation, and they are learned by SOM. The agent can extract another service by giving current user situation. We implemented the above learning mechanism to the agent, and use history of a user for two weeks was made to learn. As the result, we confirmed that the agent can extract services related with time or location and can suggest combinational services.

REFERENCES

[1] Sony, Matsushita, Philips, Thomson, Hitachi, Toshiba, Sharp, and Grundig, *Specification of the Home Audio/Video Interoperability (HAVi) Architecture*, May 1998. http://www.havi.org

[2] ECHONET Consortium, *ECHONET Specification*, Version 1.0, Sep. 2000.

[3] Sun Microsystems, Inc., *Jini* Architecture Specification, Version 2.0, Jun. 2003. http://wwws.sun.com/software/jini/specs/

[4] J. Nakazawa, Y. Tobe and H. Tokuda, *On Dynamic Service Integration in VNA Architecture*, IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E84-A, No.7, Jul. 2001.

[5] M. Iwai, J. Nakazawa and H. Tokuda, *uBlocks: Enabling User-side Composition of Distributed Ubiquitous Application*, IPSJ Tech. Report, Vol.2002-IAC, No.4, pp.29-36, Nov. 2002.

[6] V. Lesser, M. Atighetchi, B. Benyo, B. Horling, A. Raja and R. Vincent, *A Multi-agent System for Intelligent Environment Control*, Proc. 3rd International Conference on Autonomous Agents, pp.291-298, 1999.

[7] FIPA (Foundation for Intelligent Physical Agents), *FIPA Communicative Act Library Specification*, Aug. 2001.

[8] T. Kohonenn, *The Self-Organizing Map*, Proc. the IEEE, Vol.78, No.9, pp.1464-pp.1480, 1990.

[9] T. Kohonen, J. Kangas and J. Laaksonen, *SOMPAK: The Self Organizing Map Program Package*, Apr. 1995.