6-5 A Study of a Naming Scheme for User-Centric Environment

MURAKAMI Homare, OLSEN Rasmus Løvenstein, SCHWEFEL Hans-Peter, and PRASAD Ramjee

We will be able to access to our all resources from anywhere through Personal Network (PN) connecting a user's Private Personal Area Network (P-PAN) and his/her clusters in a secure manner.

We describe in this paper requirements on the naming scheme for the user-centric environment. Hereafter we propose a naming scheme, named New Naming Scheme (NNS). The naming scheme is developed based on Domain Name System (DNS) and satisfies the requirements. The naming scheme introduces two-layer concept to divide name space into private flat name space and public hierarchical name space.

Keywords
Naming, NNS, DNS, Personal network (PN), User-centric

1 Introduction

To realize new-generation mobile environments, it is extremely important that we pursue research into the services and their usability in an actual usage environment, in addition to research on improving transmission rates and connectivity, including the issue of vertical handover between two or more different wireless systems. For users, the essential issue lies in the services they will be able to use; the provision of the most effective services is thus the key to the successful spread of new-generation mobile environments.

As the Internet continues to grow and electric devices and home appliances are getting to be connected to the network, the role of wireless communication will be extended to allow for connection among all the resources connected to the network, not only to provide connectivity to mobile users. And while the world of communication services has to date revolved around. Network operators must reconsider and reconstruct their business from operator-centric infrastructure model to user-centric service model.

A user-centric environment requires deploying a function to find devices and services that users wish to connect to. Currently, the identity of devices is managed by the network operator by assigning a number to each device (as in the current system of telephone-number assignment). But, this must be changed to a distributed manner: a user assigns a name to his or her own devices and registers the names in public shared servers. Accordingly, we have investigated a naming technique as one part of a distributed management method of assigning device names, under the auspices of the MAGNET[1]project (one of a number of European IST projects) since 2004. Specifically, we have investigated this issue in collaboration with Aalborg University, Denmark, a leading member of this project.

This article discusses a naming method

proposed as a result of this investigation. In particular, Section **2** describes the nature of the user-centric environment, Section **3** provides an overview of the currently proposed naming schemes, and Section **4** discusses the details of the New Naming Scheme (NNS), the naming method that we now propose.

2 User-centric mobile environment

Today, real-world information acquired by various RF-IDs and sensors can be obtained on the Internet. On the other hand, some home appliances are starting to be equipped with an Internet connection port to accept remote control via the Internet. A ubiquitous communication environment, aimed at the interaction between the real world and networks, is thus rapidly being deployed. On the other hand, both wired and wireless communications networks have rapidly improved in terms of transmission rate and capacity, and have started to be offered with flat-rate fees. The combination of these trends makes it possible for users to access all the information stored in a personal device at his or her home, office, or car from anywhere.

While the environment gives greater convenience to users, it also increases the security risk. For example, if the complexity of the system demands that the user have specialized knowledge, it may lend itself to mistakes in configuration, and the system may allow unauthorized access by third parties to personal devices or information. As a result the system to be implemented must allow for easy configuration (or automatic configuration), and must be designed to avoid the security risks.

The MAGNET Project[1], one of the European IST projects, deals with research into systems offering heightened convenience while avoiding these security problems. While the concept of a Personal Area Network (PAN), consisting of a group of devices placed near the user, has been considered at the (wireless) link level, this project adds the definition that a group of devices owned by a user

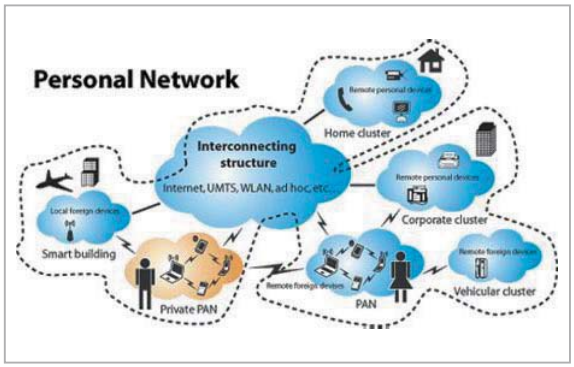

Fig.1 *Personal Network*

as a Private Personal Area Network (P-PAN). The devices in the P-PAN are considered trustworthy to each other and are thus allowed to send and receive data freely, and are mutually accessible within the P-PAN. On the other hand, communication with devices outside of the P-PAN is restricted.

A small-sized network will be organized by the devices owned by a single user where the user stays for a relatively long period—at home, in the office, or in the car, for example. This local group of devices is referred to as a "cluster". As with the P-PAN, this group of devices is considered secure within this cluster, and communication within the cluster is freely allowed. However, communication outside the cluster is restricted.

By dynamically connecting the P-PAN and cluster using a secure method such as VPN in accordance with the user's requests, the user can access all devices and data that he or she owns from anywhere. The user can handle the remote devices and data as if they are local. The virtual network composed of the P-PAN and the cluster is referred to as the Personal Network (PN). Figure 1 shows a conceptual illustration of this structure.

3 Need for a naming technique

If many devices are connected to the network, it is important to provide a method of finding and identifying the service that a user wants to access. For example, when a user forgets to set a video deck to record a TV program on a particular night, he or she will try to

program it remotely. However, even though the video deck may be connected to the Internet, it will be difficult for the user to find and operate the deck on the network if the device is identified only by numbers or meaningless strings, as with an IP address.

There are two different solutions for this problem. One is to assign names that the user can easily understand to all the devices, in addition to the IP addresses. Packets are routed based on the IP addresses, so the binding of the name and the IP address of devices must be managed somewhere on the Internet. The name is translated to its address, and vice versa, at the user's request. This scheme is referred to simply as "naming". This system avoids forcing users to memorize meaningless numbers and strings.

Another solution is to register the services provided by each device (in other words, what the device can do) in a specific server in advance. When the user sends out a request to the server—for example, "record a TV program"—the server searches for the device that provides the service that matches the request. In this manner, the user can access the device he or she is looking for. This scheme is referred to as "Service Discovery". Generally, the Service Discovery technique is more complicated than the naming technique, and thus is considered more difficult to implement in terms of scalability.

These two techniques are not mutually exclusive, and combining them can enhance user convenience. For example, among the techniques proposed in the past, the Intentional Naming System (INS)[2] features characteristics of both of these schemes. In INS, each device has a name which shapes a tree structure, and the tree contains elements such as location, types of supporting services, and device accessibility. Therefore users can easily find the desired service nearby. However, as information on names is shared hop-by-hop, it is difficult to ensure scalability on the scale of the Internet.

The present Internet widely uses the Domain Name System (DNS)[3] as a naming scheme. The DNS features a hierarchical name space. Here a name is the combination of a device name and a domain (organizational) name. For example, in the name "www.nict.go.jp", the initial "www" is the device name, and the rest, "nict.go.jp", constitutes the domain name. The domain name has a hierarchical structure (in the example above, jp→go→nict). The name servers in each organization are also arranged to construct the same hierarchical structure. In this manner, the device name can be managed independently within each organization. Even when the numbers of domains and devices increase, DNS can manage the whole name space with the distributed servers using this hierarchical structure, thus offering Internet-wide scalability.

For the PN architecture, the widely-used Internet is the most promising candidate for the backbone infrastructure, e.g. for connecting P-PANs and clusters in the PNs. Although we have seen attempts at the construction of new managed networks—the Next Generation Network (NGN)[4], for example—these networks also focus on low-cost construction based on IP technology and are expected to be highly compatible with Internet applications.

From this point of view, it seems a sensible approach to continue using DNS on an IP-based infrastructure for the PN architecture. However, DNS as is cannot fulfill a number of requirements for the PN architecture.

First, a PN must support its user mobility. In particular, each device in the P-PAN frequently changes its point of attachment to the Internet (or to an IP-based backbone network) as it changes Radio Access Network (or RAN; access technologies such as Ethernet, wireless LAN 802.11g, and W-CDMA, for example). Here, the new point of attachment does not always belong to the same domain as the previous point of attachment. In DNS, the device name depends on the point of attachment, which essentially does not meet to the requirement of device mobility.

Second, with DNS, all device information is more or less open to the public. The advantage of DNS is that users can access each

device with a user-friendly, simple name when all devices are connected to the network. However, it means that unexpected users can also see the names registered in the DNS. If the device has a general or obvious name, it may attract unauthorized access from intruders. In principle this should not cause a problem, so long as the access control settings are appropriate. Nevertheless, it is preferable from a management point of view that the devices to be used privately conceal this sort of information from third parties.

In light of these considerations, this article investigates a naming scheme based on a DNS, but extended to avoid the two problems described above. We refer to this extended DNS method as the New Naming Scheme (NNS).

4 New Naming Scheme (NNS)

4.1 Two-layered naming system

As discussed above, any device names registered in a DNS server are open to all users, which is not preferable in terms of security. Thus, we divide the name space of the DNS into two layers. One is only accessible by the owner of the devices, and the other is open to the public just like the current DNS name space. The former layer is referred to as the PN layer, while the latter is referred to as the IP layer. Figure 2 illustrates the concept behind this structure.

The IP layer is the present DNS architecture itself. The names of devices that accept access by others are recorded in a hierarchical name space and are globally effective. On the other hand, the PN layer constructs a name space that only the owner of the devices can access. In other words, while there is a single global IP layer shared by everyone, a PN layer is established for each user, and user A's PN layer and user B's PN layer are different.

There are advantages other than security in constructing a name space for each user in the PN layer. As the name space of each user is independent, the same device names can be used by different users without causing a problem. For example, the names "tv", "pc", and "printer" will be assigned to devices by many users because these designations are intuitive and easy to understand. As the name spaces of users A and B are independent, both users can name their PCs "pc" without a problem, and the system can bind different IP addresses to the two PCs. However, this is true only for the PN layer. In the name space of the IP layer, the device names need to be independent on the level of a "fully qualified domain name" (or FQDN, the full name of the device,

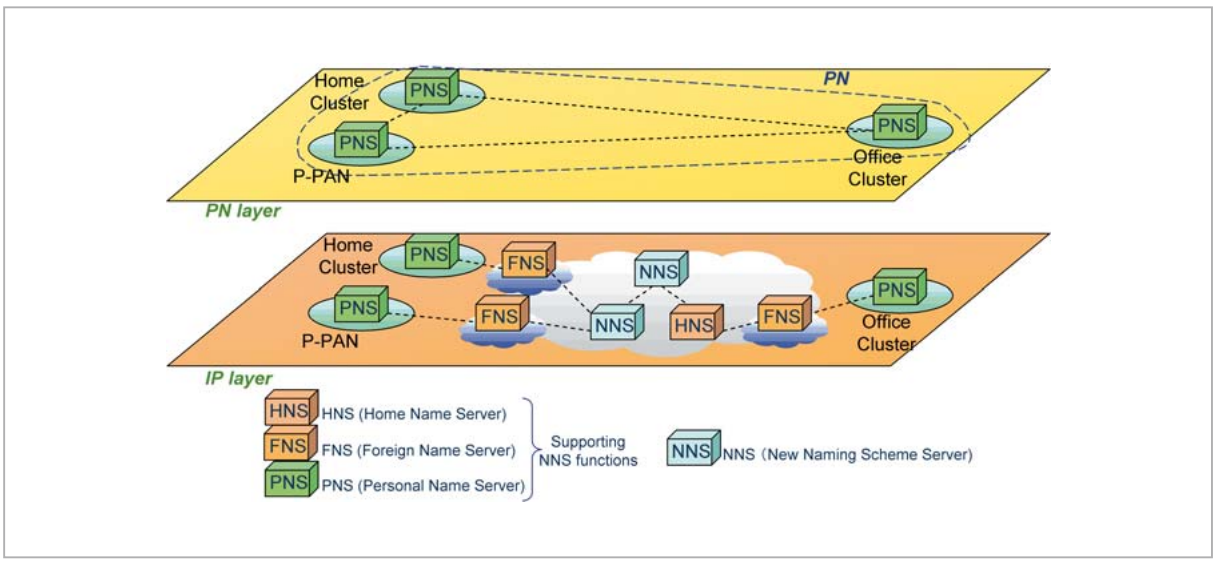

Fig.2 NNS architecture

including the domain name). For example, when users A and B are in the same domain and they register their PCs to the IP layer as open to the public, they need to assign different names to their respective PCs.

4.2 Role of PNS

The NNS server consists of two parts; a basic component of the original DNS function and an extended component installed with additional functions for the NNS. If the extended component is not used, it behaves in the same manner as the existing DNS server. It is practically impossible to change all DNS servers to NNS servers at once, so we can view the new structure as effective insofar as it is deployed gradually.

The NNS servers are assumed to be arranged hierarchically in a manner similar to the arrangement of current DNS servers. In addition, a subset of NNS functions must be placed in a device in each P-PAN/cluster along with the primary NNS server.

The three servers indicated in Fig. 2—namely, the Home Name Server (HNS), the Foreign Name Server (FNS), and the Personal Name Server (PNS)—are designated differently according to their roles, but they are all NNS servers (and are also subset functions). The following discusses in detail the roles of these three servers.

4.2.1 Communication within P-PAN/cluster

We assumed that all the devices employing the PN architecture had a fairly small program with a subset of NNS functions. We select a device with relatively rich resources (for example, a PC or mobile-phone terminal) in the P-PAN/cluster. The selected device works as a server. The program operates as the client on the other devices. Specifically, the device that acts as the server is referred to as the PNS. Figure 3 shows the message flow between the PNS and a client in the P-PAN/cluster.

The PNS functions as the so-called master node in the P-PAN/cluster. The PNS regularly broadcasts an advertisement packet throughout its P-PAN/cluster to let the other devices know of its presence. The advertisement packet contains its unique ID (e.g. cluster_name) in the PN. It is best to set the cluster name manually (with names that the user can easily understand; for example, “home” and “office”).

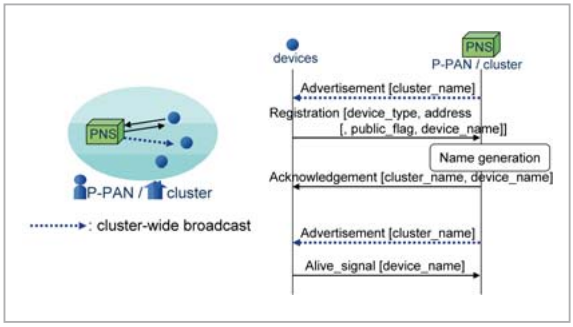

Fig.3 *Message flow within P-PAN/cluster*

On the other hand, devices other than the PNS register their information with the PNS using the client function. The clients know which P-PAN/cluster they belong to by the cluster name broadcasted by the PNS. When the device is not registered with any PNS (directly after the power is turned on, for example), or when the device detects it has moved to a new P-PAN/cluster, the client software transmits the registration packet to the PNS. This registration packet must contain (1) indication of the type of device and (2) its IP address. If the user wants to open the device to the public, then (3) a public-device flag must also be specified. Since many devices are connected to the network, it is preferable, for purposes of convenience, that the name of the device be generated automatically. However, if the user wants to assign arbitrary names to his/her devices, then (4) the device name can also be explicitly specified. Items (1) and (2) are mandatory. Items (3) and (4) are arbitrary. When the public-device flag (3) is set, the device name (4) is also required, to maintain the uniqueness of the device name. Here, a client once registered in the PNS periodically transmits a low-frequency “live” signal, as long as there are no modifications or reassignments of addresses.

In this manner, the PNS acquires information on all devices in its P-PAN/cluster. Based

on the acquired information, the PNS generates names for the devices if they are not explicitly specified with arbitrary names in (4). An example of name generation using information on the device type (1) and the cluster name is

```
device_type[serial_number].cluster_name
```

Specifically, the PNS generates names such as "tv01.livingroom" or "printer03.office". These names contain information on device type and locations, thus this method provides names that the user can easily understand. Here, a serial number is used to avoid overlapping of names when two or more devices of the same type exist within the P-PAN/cluster. If the names do not overlap, a serial number is not necessary. However, when overlap is avoided by the use of serial numbers, as in "tv01" and "tv02", it is difficult to distinguish which of the two television sets is referred to as "tv01". It would be best if a characteristic difference between the two sets (such as manufacturer name or the size of the display) were to be incorporated into the name, based on the detailed device information acquired in collaboration with the Service Discovery function or through context management technology.

The PNS keeps the device name acquired in the process described above and the address of the device in the binding list for a certain period of time. If the PNS does not receive a periodic live signal from a device during this time, the PNS judges that the device has moved out of the P-PAN/cluster or has ceased to be active (for example, if the device has run out of batteries), and it removes the name from the list.

4.2.2 Communication in PN layer

The device information collected in a PNS is shared among all PNSs within the user's PN layer to allow full access. Figure 4 shows the message flow when this information is shared among the PNSs.

The name space in the PN layer has a flat structure, so the PNSs share horizontal relationships. Thus, the key of the naming scheme in the PN layer is how to establish the relationship and how to share the device information among PNSs.

We choose a proactive method in this article, meaning that the PNSs mutually exchange name space information before the user starts actual data transfer. Each PNS broadcasts an advertisement packet (a "hello" packet) throughout the PN to find the other PNSs. Here, the PNS includes its ID (cluster_name) unique to the PN in the advertisement packet. The other PNSs that have received this broadcast verify whether the cluster name received is known or unknown. If the cluster name is unknown, it means that the receiving PNS does not have the device information for the cluster of the issuing PNS, and vice versa, so these PNSs need to exchange information. Thus, the PNS that has received the broadcast returns an acknowledgement packet including its cluster name to the PNS that sent the hello

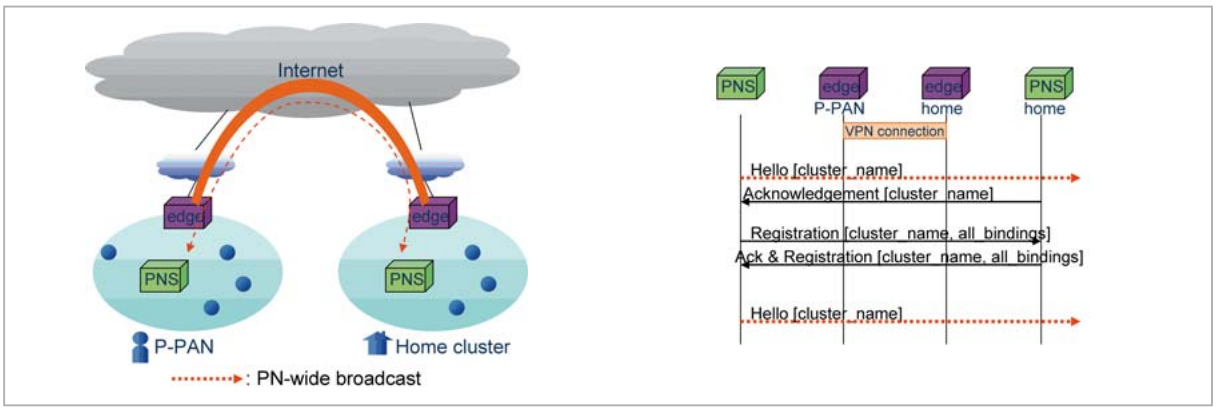

Fig.4 Message flow for PN layer

packet. After exchanging their cluster names, they exchange information on the devices that are within their respective P-PANs/clusters.

By repeating these procedures between all PNSs, each PNS gathers the complete set of name spaces in the PN. In other words, each PNS can resolve all queries by itself, without the need to transfer the query to another PNS.

If the device information is updated in a P-PAN/cluster after the name space is constructed through this exchange process—for example, when a new device is connected—the PNS of the updated P-PAN/cluster instantly sends a signal out via unicast (or multicast if possible) to notify the other PNSs in the PN of the change. This process maintains a state in which all PNSs share the latest information.

Each PNS stores the exchanged device information, with the addition of the names of existing clusters and timer information. Each PNS broadcasts the "hello" packet regularly, as stated above. If this broadcast is not received within set time (i.e., the set value of the timer), the information on the devices in the corresponding P-PAN/cluster is discarded.

Although this article uses a proactive scheme, a reactive scheme may be more efficient in terms of communication in an environment in which few queries are generated. A reactive scheme in this case would not allow for the exchange device information in advance, even when a PNS finds another PNS through a "hello" packet. Instead, when a PNS has received a query, it transfers this query to other PNSs to be resolved. If the names are assigned automatically following the manner described above, a query which searches for a device name—for example, "tv01.livingroom"—obviously needs to be forwarded to the PNS of the "livingroom" cluster to be resolved. This structure would thus maximize the efficiency in which requests are resolved.

However, this method presents a problem. Let us assume, for example, that a user brings a digital camera named "camera02.home", to his/her office. In order to route the query to the correct PNS of the existing P-PAN/cluster by the device name, the name of the camera needs to be changed as it moves: "camera02.home" at home, "camera02.p-pan" while the user carries it, "camera02.office" in the office. The device name thus loses its effectiveness as a pointer, an undesirable result.

4.3 Role of HNS / FNS and communication in the IP layer

The name space in the IP layer has a hierarchical structure, similar to the name space of the existing DNS. All devices belong to specific domains, which makes distributed management possible. As the names themselves include a representation of this hierarchical structure, everyone can reach the name server of the domain to which the device belongs by iterative query.

Today, most devices on the Internet registered in a DNS are not mobile and rarely change their domain or its IP address. Schemes have been proposed for dynamic updating of the DNS registration [5] and implemented for users assigned dynamically changing IP addresses, such as dial-up users. However, we can assume that the frequency of this updating is not high.

On the other hand, the next-generation network must accommodate services designed for high mobility, including mobile-phone services. When a device moves at high speed, changing its points of attachment to the network, the IP address of the device frequently changes, and the name server will also require frequent updates. We thus require a modification of the naming scheme to accept frequent binding updates.

As the devices registered in the IP layer of the NNS can be seen by all users, a minimal number of devices—in other words, only those that other users are permitted to use—should be registered. For this reason, only devices with a "public device flag" set to "on" within the NNS client software (as described in ***4.2.1***) are registered.

The devices with public device flags must have unique names specified by the user (for example, "pc01"). Combining the domain name of the user's so-called home network

(for example, "nict.go.jp") produces a unique FQDN ("pc01.nict.go.jp" in this example). Even if the device attaches to another network, the name will not be changed.

If someone attempts to access this "pc01.nict.go.jp", the user must resolve the IP address of "pc01.nict.go.jp" through the name server of the device's home network, "nict.go.jp". The NNS server that manages the domain of the user's home network is specifically referred to as the Home Name Server (HNS). Figure 5 shows the name servers in the IP layer and the configuration of the name space.

When the IP address of the device changes, the PNS transmits the latest address to the HNS, which updates the binding recorded in the HNS. However, when the HNS and the PNS are located far apart (in terms of the network) or when the communication rate between the P-PAN and the Internet is low, frequent binding updates may occupy bandwidth that ought to be used in data transmission. Thus we introduce a function to reduce the frequency of binding updates.

The function is implemented in the NNS of the "attaching" network—i.e., the NNS server for the network to which the P-PAN/cluster is connected. This NNS server is referred to as the Foreign Name Server (FNS). Instead of performing a binding update between the PNS and the HNS as originally intended, this function divides the update into two parts, one between the PNS and FNS and the other between the FNS and HNS. The binding update is transferred from the PNS to the FNS, instead of the HNS. At the same time, the FNS informs the HNS of its address and the name of the device in the P-PAN/cluster for which it is receiving the update information in place of the HNS (see red arrows in Fig. 6). The frequency of the signal is reduced, and in addition the amount of the packet size of the signal is smaller than the original bind-

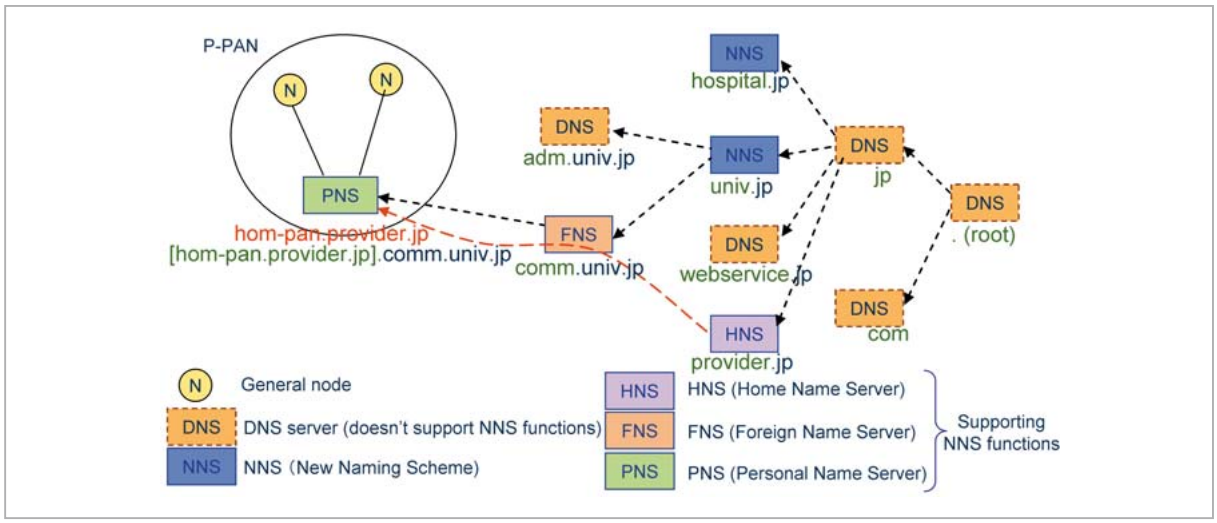

Fig.5 *Servers in IP layer and configuration of name space*

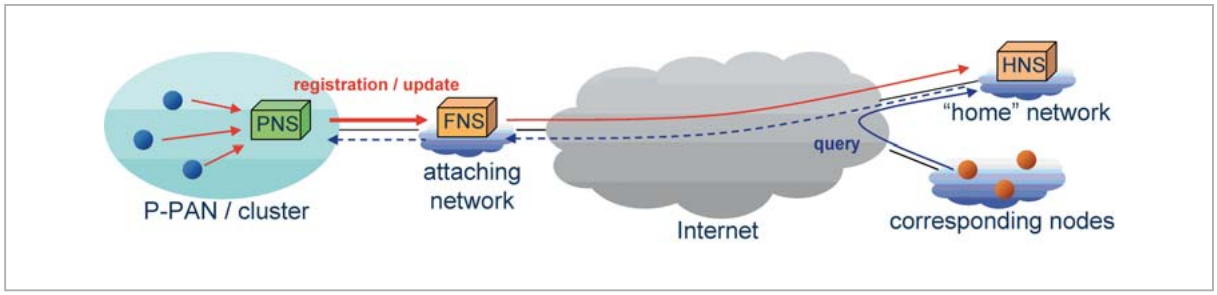

Fig.6 *Collaboration of three types of servers*

ing update.

On the other hand, a user who wants to access a public device (corresponding node in Fig. 6) can reach the HNS by iterative query. To route this query to the FNS, the HNS notifies the IP address of the FNS to the corresponding node as the name server to access as the next step of the iterative query.

In the example of Fig. 5, to access the device, "hom-pan.provider.jp". the corresponding node reaches the name server of the "provider.jp" domain, which is the HNS in the original method of DNS query resolution. Instead of returning the address of the device, the HNS urges the corresponding node to access the name server of the "comm.univ.jp" domain, which is the FNS to which the P-PAN is presently connected. Finally this FNS resolves the address "hom-pan.provider.jp".

As shown in Fig. 5, the name server configuration of the IP layer can contain DNS servers without causing a problem. The system only needs to include the HNS of the home network and the PNS in the P-PAN/cluster in order to operate. The corresponding node does not even need to support NNS. This is because the query transmitted to the HNS or to the FNS and the response received from them are exactly the same as a query to a DNS server for the corresponding node. The only differences between the DNS and the IP layer of the NNS are the update scheme between HNS–FNS–PNS and an additional iterative process stage (HNS→FNS) when the FNS is involved.

5 Conclusions

Here we have described the investigation of a two-layered naming scheme. A PN layer for personal use provides a name space for each user and automatically generates device names. This layer is focused on ensuring user convenience. On the other hand, the IP layer for public use is highly compatible with existing DNS technology and is designed to include a minimum number of devices, for reasons of security.

In the future, we would like to implement the proposed method and evaluate its feasibility and scalability. We would also like to extend and modify the proposed method to handle the situations currently not included: collaboration between PNs of two or more users (PN Federation[1]), devices shared by two or more users, and public terminals, such as KIOSK terminals.

Acknowledgement

We would like to thank the participants of MAGNET,* with whom we had many fruitful discussions and who provided numerous helpful proposals.

* About MAGNET/MAGNET Beyond

MAGNET—My personal Adaptive Global NET—is a worldwide R&D project within Mobile & Wireless Communication beyond 3G. MAGNET will introduce new technologies, systems, and applications that are at the same time user-centric and secure. MAGNET will develop user-centric business model concepts for secure Personal Networks in multi-network, multi-device, and multi-user environments. The MAGNET consortium contains 37 partners from 17 countries, combining highly acknowledged Industrial Partners, Universities, and Research Centres.

MAGNET: FP6-IST-IP-507102.

MAGNET Beyond: FP6-IST-IP-027396.

References

1 IST-MAGNET consortium, IST-MAGNET / MAGNET Beyond project webpage, http://www.ist-magnet.org/

2 W. Adjie-Winoto, et al., "The design and implementation of an intentional naming system", Proc. of ACM SOSP'99, pp.186-201, Dec. 1999.

3 P. Mockapetris, "DOMAIN NAMES-CONCEPTS AND FACILITIES", IETF RFC 1034, Nov. 1987.

4 International Telecommunication Union, Next Generation Network Global Standards Initiative (NGN-GSI) webpage, http://www.itu.int/ITU-T/ngn/index.phtml

5 P. Vixie, et al., "Dynamic Updates in the Domain Name System (DNS UPDATE)", IETF RFC 2136, Apr. 1997.

MURAKAMI Homare

Researcher, Ubiquitous Mobile Communication Group, New Generation Wireless Communications Research Center

IP Mobility, Wireless Transport Protocols, Naming

***SCHWEFEL Hans-Peter**, Ph.D.*

Associate Professor, Aalborg University

OLSEN Rasmus Løvenstein

Research Assistant, Aalborg University

Context-Aware Service Discovery

***PRASAD Ramjee**, Ph.D.*

Professor, Aalborg University