A Platform for Physical Triggers in Ubiquitous Environment

Goro KUNITO Kenji SAKAMOTO Ken'ichi YAMAZAKI
Network Laboratories, NTT DoCoMo, Inc.

1 Introduction

Computing and telecommunications are maturing, and **Ubiquitous computing** has become quite interesting as a new environment in which to seek new services. The goal of Ubiquitous computing is the enhancing computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user[1], and the goal of services in ubiquitous environment is not making either services or appliances the center of attention but enhancing users' lives. A new application platform is needed to support this vision, and we have proposed Ubiquitous Daemon (UD)[2]. This paper presents issues of handling physical information to achieve services in ubiquitous environment.

2 Invisible Service

Currently, most users think of services provided by information technologies as associated with their computers, however such computers should vanish into the background in ubiquitous environment. Of course, such disappearance is a fundamental consequence of human psychology. Services also should be disappeared and assist users without intruding their conscious, and we call such services 'invisible service'.

The model of the invisible service consists of three steps as follows:

- recognition of physical world:
  It is necessary to keep monitoring users, objects and environment.
- selection of appropriate services:
  In the ubiquitous environment, a huge number of objects in the world are expected to communicate, and a great many services related to these objects are expected to be provided by various service providers. Therefore it is important to choose appropriate services according to detected physical world information.
- adaption of result to physical world:
  It is necessary to adapt the service result to the physical environment of the user.

Services are provided in the logical world. It is very important to monitor users, objects and environment to prevent services from intruding on the users' consciousness. We focus on RFID tags as one of the most promising tools to monitor and connect physical world with the logical world.

2.1 Ubiquitous Daemon

According to the model of service described above, we proposed the concept of the Ubiquitous Daemon (UD)[2]. UD exists, correspondent to a tag, on a platform in the logical world, and each UD consists of three components, Ubiquitous Daemon Input (UDI), Ubiquitous Daemon Engine (UDE) and Ubiquitous Daemon Output (UDO) (Fig 1).

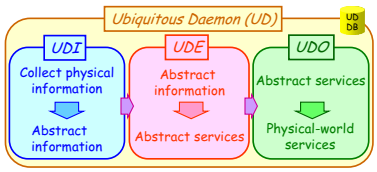

Figure 1: Ubiquitous Daemon

When the ID of the tag which is stuck on a object is detected by a tag reader, the ID is sent to its UD and the UD is activated. UDI extracts information of the object from a database and recognizes the import of the physical information. UDE searches and executes some appropriate services based on the physical world information, and UDO converts the service result service to the physical world.

2.2 Service Model

Figure 2 shows service model from detecting tags to providing services. When a tag reader detects a tag, it send ID of the tag to its correspondent UD and activates the UD. The UD looks up attributes of the ID from the ID database, and the UD seeks services in service DB, which can be provided various service providers. Since services related to a tag are unfixed, they are customizable and flexible. The UD executes the service, and looks up contents in a DB, which is actually sent to users as the service result to the user. Various content providers can provide the contents, and one service provider may create both service and contents. Finally the UD offers the contents to the user.

Separation of tags and services realizes two aspects, for users and businesses. First services are user cus-

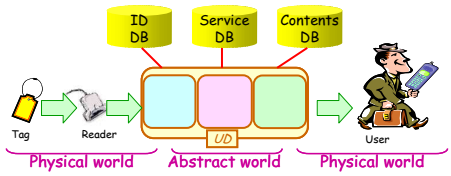

Figure 2: Service Model

tomizable; the user can receive different services as triggered by a single tag that suit the user's context. Second, the separation can be the supports for business models, admitting of horizontal service provision. In addition, the separation enables emerging business models to connect physical objects with services.

3 Interaction between UDs

3.1 Multiple Tags Detection

There can be so many objects in the world that some problems are arose as follows (Fig.3):

- Trigger Storm:
  In ubiquitous environment it is expected that a tag reader detects a great many tag at the same time, and they would cause huge number of UDs activated. Grouping is the one of way to reduce number of tags activated, and methods of grouping can be divided broadly into two categories, physical way and semantic way. Physical grouping means categorizing based on physical context, while semantic grouping means categorizing based on attributes of tags. Although tag reader can categorize based on its physical ability[3], semantic grouping can divide logically and flexibly beyond tag's ability. Further consideration is required.
- Neighbor discovery Storm:
  Since some services are expected to require information of multiple objects, UDs have to discover other UDs and to collect such information. If UDs have to broadcast queries, storm of neighbor discovery would occur. One of the solution is described in following section.
- Service Storm:
  If all activated UDs provide services to the same user, the user would receive to many services, and the services would make congestion on the output device. Furthermore the user might be charged for such troublesome services, and that is service storm problem. To solve this problem further consideration is required.

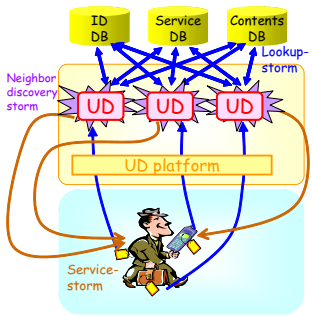

Figure 3: Storm Problems

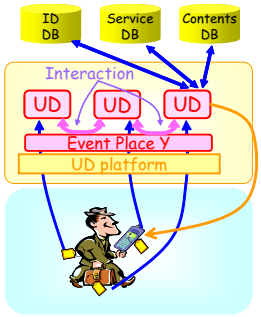

Figure 4: Event Place

3.2 Event Place

We introduce a Event Place on UD platform (Fig.4). UDs on a same event place can communicate each other and handle triggers from tags. Since UDs which have common attributes gather on same Event Place, the Event Place has some attributes and that enables UDs to provide services flexibly.

4 Examples of Invisible Services

Figure 5 shows an example of invisible services. In this example, a woman is shopping in a convenience store. Tags are stuck on all items, the shopping cart has a tag reader, and the shop cooperates with a cooking book company. When she put some items into shopping cart, the tag reader detects the tags of items and the woman. UD correspondent to one of the items in the cart finds some recipe services in DB of the cooking book company, detects all items in the cart, and she receives a voice message service containing a shopping advice such as "Won't you buy a fresh tomatoes? And you can cook one more dish" with her favorite actress's voice.

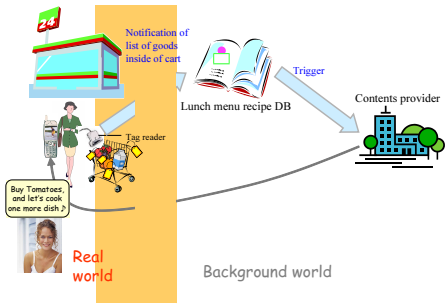

Figure 5: An example of invisible service

5 Conclusion

In order to make services invisible it is necessary to keep monitoring physical world, and we focus on RFID tags. A huge number of tags are expected to be stuck around us, and tag readers may detect many tags at the same time. We have proposed an architecture of Ubiquitous Daemon and this paper described some issues to build services in detecting multiple tags.

Ubiquitous Daemon provides a very challenging platform in which to investigate applications that can harness the emergent behavior of ensembles of simple tags. Dealing with building invisible services triggered by multiple tags detected simultaneously poses critical research challenges for the ubiquitous environment. We will keep on studying issues in more details.

References

[1] Mark Weiser, "Hot Topics: Ubiquitous Computing," *IEEE Computer*, Oct 1993.

[2] Kenji Sakamoto and Goro Kunito and Kenichi Yamazaki, "Toward Real-World Management Platform for Invisible Services," in *Communications Society Conference of IEICE*, vol. 2, pp. 507–508, Sep. 2002.

[3] Kenji Sakamoto and Goro Kunito and Kenichi Yamazaki, "Processing Physical Triggers in Ubiquitous Environment," in *The 65th IPSJ Conference*, no. 3H-4, Mar. 2003.