Profile Aggregation and Dissemination: A Framework for Personalized Service Provisioning

Daisuke Morikawa Masaru Honjo Akira Yamaguchi Masayoshi Ohashi
KDDI Corporation
3-10-10 Iidabashi Chiyoda-ku, Tokyo 102-8460 JAPAN
+81 3 6678 5717
{da-morikawa, ms-honjou, no-kotsuka, ai-yamaguchi, ma-oohashi}@kddi.com

ABSTRACT
This paper presents a framework for aggregating, updating and disseminating user-related profiles where each part of the profile comes from a different source. A user-related profile, called a "Personalized Profile" in this study, consists not only of time and location, but also various kinds of activities in daily life. We also present personalized service scenarios based on this "Personalized Profile".

Keywords
context-aware, service-provisioning, aggregation, profile

1 INTRODUCTION

Although a framework for adaptive networking and computing such as web content personalization and adaptive service composition have been researched, various kinds of profiles are required to create these services.

Various kinds of digital information are transmitted in ubiquitous information environments with a wide range of networking, computing, and distributed contents and services. User-related profiles and/or context information is required to provide relevant context-aware services. Figure 1 shows a schematic illustration of the relation between user's daily activities and related profile information.

User-related profile and/or context information is managed independently by various owners such as service providers (SPs) where the user is registered, the company where the user works, and by users themselves. SPs now provide services to users based exclusively on their own user profile/context information that they have collected and managed. However, a shortage in the amount of profile/context information managed by an SP may make it difficult to provide relevant context-aware services to the user.

We have already proposed a context-aware system, called CASP, which offers a framework for aggregating and disseminating user context information [1]. The user profile/context information must be aggregated and managed from diverse souces in order to improve the relevance of context-aware services.

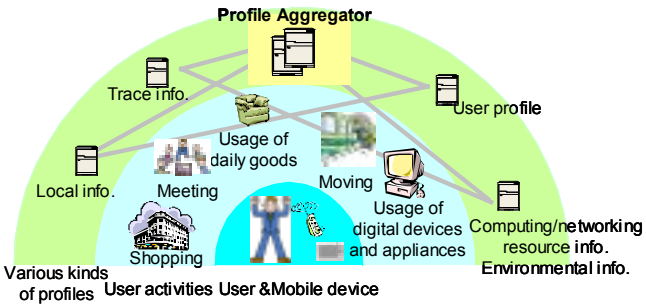


Fig. 1 Schematic Illustration of User's Daily Activities and Related Profile Information.

2 A FRAMEWORK FOR PROFILE AGGREGATION AND DISSEMINATION

2.1. Architecture

There are various kinds of digital information, including static and dynamic data, handled in a ubiquitous information environment with a wide range of networking, computing, and distributed contents and services. Thus user related profiles must be aggregated in order to provide users with relevant context-aware personalized services.

A basic model of profile aggregation, updating and dissemination is shown in Fig. 2. The Profile Collectors (PCs) indicate profile sources and each PC collects and maintains a specific part of the profile. Typical PCs dealt with in this study are described in Table 1. This table is divided into two categories: personal profile and environmental profile. Some parts of the profile are managed by the user, while others are provided by the external profile sources with which the user is registered.

The Profile Aggregator (PAs) aggregates various profiles from diverse PCs and manages them as a "Personalized Profile." The Collector Resolver (CRs) resolves the appropriate PC based on profile attributes.

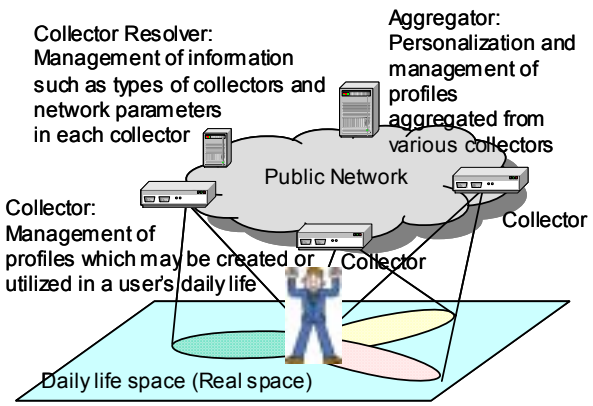


Fig. 2 Schematic Illustration of Profile Aggregation and Dissemination.

Table. 1 Example Profile Collectors

Profile Collectors for a Personal Profile
A data repository a user registers personal profiles and preferences with through a web-based interface.
A location server where a user is registered.
Personal terminals with a software module that monitors application usage history, web access, etc.
Personal terminals with a software module that manages purchase history.
Profile Collectors for Environmental Profile
A data repository involving networking/computing resources
An information server for local information.

2.2. Advantages

The proposed architecture, which aggregates user related profiles from diverse sources, has the following advantages.

- Construction of a "Personalized Profile"

  Profiles collected by environmental sensors, for example, have the same value for everyone, but are interpreted differently for each user. This system makes it possible to infer the user's context and preferences by interpreting general profiles based on user's intentions. Profiles with a high correlation are dynamically linked to each other.

- Personal repository of user experiences

  The user of terminals with different capabilities requires a centralized data repository. We assume that the user's "home server" is available as a PA and that useful data utilized and generated in the user's daily life is periodically synchronized with the PA. This system supports the management of user knowledge, information and the history of user experiences.

- Decision-maker for creating services

  Aggregated profiles are employed not merely for sharing such information between users but also for making decisions needed to dynamically create a context-aware service. A confederation of aggregators also helps in creating new services.

2.3. Requirements for the Personalized Profile

A Personalized Profile is essential for realizing ubiquitous computing applications. The following requirements should be satisfied.

Extensibility

There are various kinds of profile elements in order to represent and share not only static personal information but also user activities. A Personalized Profile is added according to the demand of application developers. Thus it should be easy to extend the data model and create new attributes for a Personalized Profile.

Interoperability

A Personalized Profile is constructed by aggregating from various profile sources so that aggregated profile information is represented in a well-formed structure. Some standardized vocabularies are necessary to maintain the interoperability in ubiquitous computing application deployment. This "ontology" enables application developers to reduce their efforts in creating context-aware personalized services.

Rich (Semantic) query capability

A Personalized Profile should be machine-readable so that application software can share, analyze and exchange a user profile information.

Dynamic update mechanism

A Personalized Profile handles user activity information as dynamic changes.

2.4. RDF-based Personalized Profile Model

A Resource Description Framework (RDF) [2] is a knowledge representation language with capabilities for modeling context information and is satisfied with the above requirements. Thus RDF, in this study, is used for describing semantics in the "Personalized Profile." The RDF statement is represented as a triple, which contains a subject, a predicate and an object node as shown in Fig. 3. This is illustrated by a node and directed-arc diagram, in which each triple is represented as a node-arc-node link. All subject nodes and some of the object nodes are represented as ovals containing their URIs, all arcs are labeled with properties, and literal nodes are described in rectangles.

Thus we have defined a set of templates that describe the base structure of the triples used in the "Personalized Profile" including the types of properties (representing a predicate node), the value range of each property and the resource types that have a given property (called "domain" and "range" in the RDF Schema [3], respectively).

2.5. Functions Required in Profile Aggregator

Functions required in the PA are described in this section.

Profile Aggregation from Profile Collectors

Communication between the PC and the PA is based on the message based on a message with an XML document on transport layer protocol such as SOAP.

Update Mechanism of Personalized Profile

RDF is designed for static descriptions, while no mechanism exists for dynamically updating triples. Thus we therefore designed and developed a dynamic update module in the PA.

We update the "Personalized Profile" based on this template. Figure 4 shows the procedure for constructing and updating the "Personalized Profile." In this study, we assume that the input profile data is described by XML. First, an XML based input profile is transformed into an RDF based input profile via the style reformat module and then the Personalized Profile is updated via the personalized profile update module. The style reformat module is described by XSL (eXtensible Stylesheet Language). In the personalized profile update module, the required functions here are 1) adding triples, 2) updating literal values, and 3) unifying the same representation of triples. This update module was implemented based on Jena (A Semantic Web Framework for Java) [4].

Query Mechanism for Personalized Profile

RDF Data Query Language (RDQL) [5] is an SQL-like query language for RDF, which is implemented in Jena. RDQL provides methods in which application developers write the declarative statement of what should be retrieved under given conditions. Thus we use RDQL to retrieve the target statements and/or values from the "Personalized Profile." The distributed query is for further study.

Trigger Notification Mechanism

We assume that multiple applications run simultaneously and access RDF statements in the PA. Changes in the profile data must be detected in a given condition in order to detect the trigger of service invocation. Then the context-based service providers (SPs) access context information when receiving trigger information from the PA.

2.6. Privacy and Security Issues

Users are concerned about privacy issues, which arise particularly in collecting and exchanging user profiles. Appropriate profile information must be exchanged with service providers in order to provide context-aware services. This reflects the tradeoff between allowing personal information access and enjoying context-aware services. Here, we summarize privacy requirements.

- In aggregating profiles from various sources

  Communication channels (e.g., communication between the PC and the PA, between the PAs, and between the PA and the SP) are securely preserved.

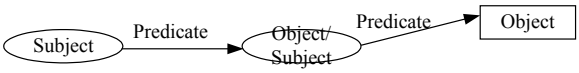
- In disseminating the Personalized Profile

  Personalized Profile should be effectively managed. However, it is difficult for users to control what data may be shared with whom and in what situations.

Various research regarding privacy control (e.g., tools for enabling end-user control of privacy, policy-based matching engines [6, 7]) has been reported and these studies are applicable to our framework. Privacy issues in accessing the Personalized Profile are beyond the scope of this paper.

3. PERSONALIZED PROFILE DESIGN

We developed a prototype of a shopping navigation service in order to demonstrate the feasibility of our framework. In this section, we first describe the service scenario and then show the Personalized Profile according to this service scenario.



All arcs (Predicates) and ovals (all Subjects and some of the Objects) are represented by URIs.
All rectangles (some of the Objects) are represented by literals.

Fig. 3 Schematic Illustration of RDF Model.

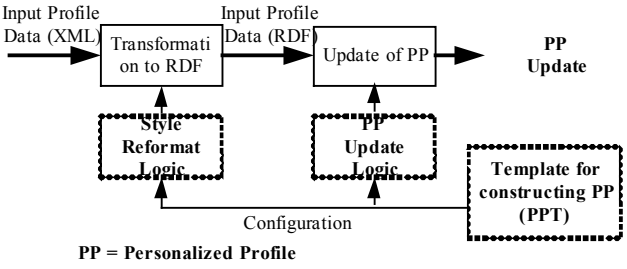


Fig. 4 Update Method of Personalized Profile Managed in Profile Aggregator.

3.1. Outline of Service Scenarios

- The user edits *a shopping list* and registers it to the user's PA. Items inferred from *the previous purchase history* are automatically added to *the shopping list*.
- When the user goes shopping, an appropriate shopping location is recommended based on the "Personalized Profiles" such as *the current location, shopping list*, *shop membership list* and *nearby shop information*. This shopping information is displayed in a suitable time registering a user's schedule.
- In the shopping mall, *the floor information database* is registered to the user's PA and then the user obtains shop information according to *the user's preference and location*. The user may consult that database when searching for locations and thus can easily navigate potential locations.
- The PA automatically checks *the shopping list* when buying an item. The PA notifies the user of an alert when the user leaves the shopping mall, if some items remain un-purchased.

3.2. Examples of Personalized Profile

Preferred Commodity Profile

We assume that passive RFID tags are attached to various objects and information corresponding to each object (i.e. electric ID, called object ID in this paper) is managed on a networked server. Mobile users have a networked mobile terminal equipped with an RFID tag reader in order to identify interacting objects. A user may view detailed information on commodities of interest and register what the user would like to annotate as the user profile. The RDF model of this commodity profile is described in Fig. 5. A user may create a shopping list through viewing detailed information on commodities in the user's vicinity. A shopping list is registered as a preferred commodity profile.

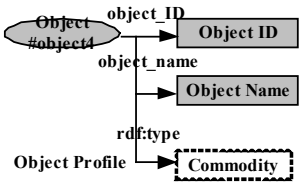


Fig. 5 Commodity Profile Model.

The RDF model of a preferred commodity profile is described in Fig. 6. *PreferredObject* and *Annotation* class are defined as a sub-class of *ProfileComponent* and *ProfileSubComponent*, respectively.

Resources described in the preferred commodity profile and resources described in the commodity profile can be unified. The unification of these two URIs is carried out if the value of the property, *Object_ID*, in the preferred commodity profile is equivalent to that in the commodity profile. The PA also registers the frequency of viewing the object and its property information as well as the timing and location profile when this generated preferred commodity profile is added. Related annotation information is added to the preferred commodity.

In this figure, the preferred object is displayed, but every time a user registers the preferred object, appropriate triples of RDF are also added. In this profile, object information is connected to the user annotation, while temporal-spatial information is also added. The linkage between profiles aggregated from several profile sources enables us to retrieve detailed information under a detailed condition.

Purchased Commodity Profile

This profile indicates information regarding the user's daily activities. In this service scenario, an electric receipt is published and transferred to the user's PA. Appropriate triples are added to the Purchased Commodity Profile, and a time and Location Profile is also added when electric receipt information is first received as an input profile. This profile is schematically illustrated in Fig. 7. *Activity* and *PurchasedCommodity* class are defined as a sub-class of *ProfileComponent* and *ProfileSubComponent*, respectively.

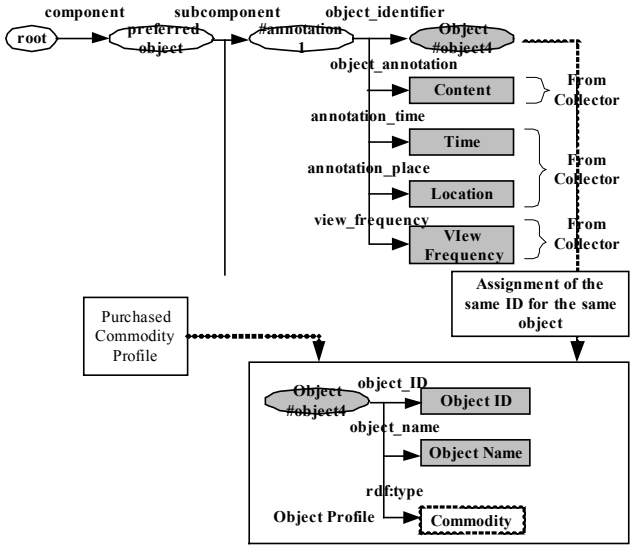


Fig. 6 Preferred Commodity Profile Model.

Resources in the purchased commodity profile and resources in the commodity profile can be unified. The unification of these two URIs is performed if the value of the property, *Object_ID*, in the purchased commodity profile is equivalent to that in the commodity profile. As a result, preferred commodity and purchased commodity may be linked through the *Object_ID*.

Preferred Place Profile

This profile indicates information that interests the user, and/or the user locates. This profile indicates user related information such as the access method to the required information database, etc. *PreferredPlace* class is defined as a sub-class of *ProfileComponent* and *CurrentPlace*, *Home* and *Shop1* class are defined as a sub-class of *ProfileSubComponent* as is shown in Fig. 8.

4. RELATED STUDIES

A number of context-aware services (e.g. [8, 9]) have been developed. These earlier studies focused mainly on building application-specific systems. Context toolkit [10] provided a concept for reusable components for managing sensor-based context information to support rapid prototyping of context-aware services.

Recent studies have attempted to create frameworks describing various kinds of user profiles. Ubisworld [11] proposes a basic semantics and RDF-based data structure for representing property attributes regarding physical objects (persons and objects), spatial and temporal information, and user activities through a web-based interface. This approach for describing semantics is similar to our proposal. However, this system is based only on Web-based input data so that an adaptive update method for collecting user related information from diverse sources is not considered. The Haystack Project [12] aims at the system to improve information retrieval for personal use. Information dealt with in the Haystack Project is e-mail client, files, calendar and address book and Web by using a desktop/laptop PC. SOUPA Project [13] defines the shared ontology, regarding beliefs, desires and intentions, time, space, events, user profiles, actions and policies for security and privacy. These are established through a Web interface.

Studies regarding middleware and infrastructure support for context-aware systems (e.g. Context Fabric [14] and GAIA [15]) have also been attempted. CoBrA [16] is a broker-based agent framework for building smart spaces, e.g., smart meeting rooms, and intelligent homes via SOUPA ontology mentioned above. Although our approach is also based on the infrastructure support approach and establishment of a common context/profile model, the point we would like to emphasize here is that our goal is to gather daily activities via mobile terminal devices.

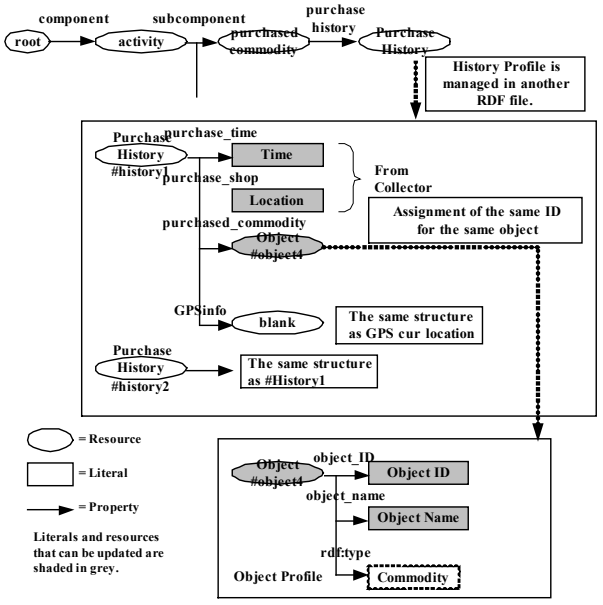


Fig. 7 Purchased Commodity Profile Model.

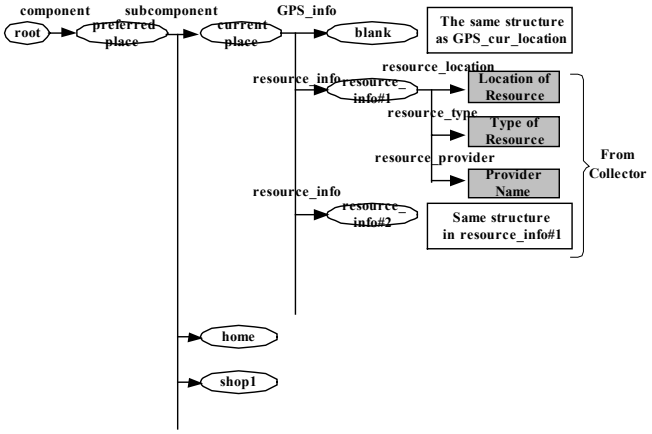


Fig. 8 Preferred Place Profile Model.

5. CONCLUSION

The first key contribution of this paper is to propose a framework for aggregating and updating profile information from multiple profile sources. The Profile Aggregator in this architecture has the role of both a personal information repository and a decision-maker.

The use of ontology is a key requirement for realizing context aware applications through knowledge sharing. However, it is difficult to determine the most appropriate data model for organizing defined vocabularies. We indicate the usefulness of the proposed ontology by prototyping a context-aware service based on a shopping navigation scenario. We also demonstrate that some of the profile is connected to a related profile and that the linkage between profiles aggregated from several profile sources

enables us to retrieve detailed information based on the design and implementation of prototype system. The proposed framework must be evaluated through prototyping various kinds of context-aware services and analyzing use-cases, while evaluated results should be fed back to the defined semantics. Privacy control issues in a context-aware service framework require further study and discussion.

ACKNOWLEDGEMENT

This work was performed under the research project of Ministry of Public Management, Home Affairs, Posts and Telecommunications of Japan.

REFERENCES

1. D. Morikawa et al., "Service Platform for Exchanging Context Information," In *Adjunct Proc. of Ubicomp 2003*, Oct. 2003.
2. D. Beckett, "RDF/XML Syntax Specification," W3C, Feb. 2004.
3. D. Brickley and R. V. Guha, "RDF Vocabulary Description Language 1.0: RDF Schema," W3C, Feb. 2004.
4. Jena – A Semantic Web Framework for Java, http://jena.sourceforge.net/
5. A. Seabome, "RDQL – A Query Language for RDF," W3C, Jan. 2004.
6. Hull, B. Kumar, D. Lieuwen, P. F. Patel-Schneider, A. Sahuguet, S. Varadarajan and A. Vyas, "Enabling Context-Aware and Privacy-Conscious User Data Sharing," In *Proc. of MDM 2004*, Jan. 2004.
7. Mark Ackerman, Trevor Darrell and Daniel J. Weitzner, "Privacy in context," Special Issue on Context-Aware Computing, Human-Computer Interaction, 16, 2001.
8. R. Want, K. Fishkin, B. Harrison and A. Gujar, "Bridging Physical and Virtual Worlds with Electronic Tags," In *Proc of ACM SIGCHI 99*, May 1999.
9. T. Kindberg and J. Barton, "A Web-based Nomadic Computing System," Computer Networks, 35(4): pp.443-456, 2001.
10. A. K. Dey, D. Salber, G. D. Abowd, " A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Computing," Human-Computer Interaction (HCI) Journal, Vol. 16, pp.97-166, 2001.
11. Ubisworld, http://www.u2m.org/
12. D. Huynh, D. R. Karger and D. Quan, "Haystack: A Platform for Creating, Organizing and Visualizing Information Using RDF," In *Proc. of Semantic Web Workshop*, May 2002.
13. SOUPA, http://pervasive.semanticweb.org/
14. Jason I. Hong and James A. Landy, "An Infrastructure Approach to Context-Aware Computing," Human Computer Interaction, Vol. 16, 2001.
15. A. Ranganathan and R. H. Campbell, "A Middleware for Context-Aware Agents in Ubiquitous Computing Environments," In *Proc. of International Middleware Conference*, June 2003.
16. H. Chen, T. Finin and A. Joshi, "Semantic Web in the Context Broker Architecture," In *Proc. of Percom 2004*, March 2004.