

A COLLABORATIVE-BASED APPROACH FOR CONTEXT-AWARE SERVICE PROVISIONING IN SMART ENVIRONMENT

W. J. QIN*

*Department of Computer Science and Technology, Tsinghua University
Beijing, 100084, China*

*Handicom Lab, TELECOM&Management, SudParis,
Evry, 91000, France*

**E-mail: qinweijun.99@mails.tsinghua.edu.cn
weijun.qin@it-sudparis.eu*

D. Q. ZHANG and M. MOKHTARI

*Handicom Lab, TELECOM&Management, SudParis,
Evry, 91000, France*

E-mail: daqing.zhang@it-sudparis.eu, mounir.mokhtari@it-sudparis.eu

Y. C. SHI

*Department of Computer Science and Technology, Tsinghua University
Beijing, 100084, China*

E-mail: shiyc@tsinghua.edu.cn

K. J. DU

*School of Computer Science, Northwestern Polytechnical University
Xi'an, 710072, China*

*Handicom Lab, TELECOM&Management, SudParis,
Evry, 91000, France*

E-mail: kejundu@yahoo.com.cn

The novel mobile scenarios encourage the design and development of context-aware middleware that provides on demand appropriate semantic services services to mobile user's access in a dynamic smart environment, since wireless hotspots and mobile devices are permeating our workplace, home and public places. As mobile users will be immersed in thousands of different kinds of pervasive services, it's of paramount importance to provide appropriate personalized services to the right person in the right form. In this paper we present a collaborative-based service provisioning approach which can enable effective service provision based on social semantic characteristics of user profile, device capability, situation context and services in WLAN enabled environment. We

illustrate our ideas through a scenario of service discovery and provision in unfamiliar public places (e.g. airport), and present an OSGi-based middleware to instantiate the service provisioning process based on semantic matchmaking.

Keywords: Smart Environment; Service Provisioning; Context Awareness; Collaborative Filter.

1. Introduction and Overview

The increase in usage of wireless hotspots and mobile devices implies that it becomes easier and pervasive to permeate the globe bringing interesting services and spontaneous connectivity to mobile users. This result in a more dynamic and information-rich smart environment beyond traditional physical spaces (e.g. home, office, airport, hospital, shopping mall, hotel, etc.) where dependent people can be supported across the heterogeneous environments and be fully facilitated into the society with better quality of life and independence, and increases the needs to allow a client device or service to discover other relevant devices and services in the network.¹ Examples of service discovery protocols (SDPs) have been investigated including Bluetooth, Jini, Universal Plug and Play (UPnP), Service Location Protocol(SLP), and Salutation.

In order to provide services in various open smart environments, impromptu service discovery and provision with heterogeneous mobile devices become critical. There are two possible ways to make the services accessible to mobile users. One way is to get connected to a global service provider and access the location-based services through the service provider. In this case, the service provider needs to aggregate all the services in the smart spaces and have the indoor/outdoor location of the user. Due to the evolutionary nature of the autonomous smart spaces, it is unlikely to have such a powerful service provider in the near future. The other way, which we are in favor of, is to enable the mobile user interact with individual smart environment directly when the mobile user physically enters the space, that is, the mobile user can automatically discover and provision the services provided by the smart spaces. In the latter case, the mobile user needs to get connected directly to the individual smart space using short range wireless connectivity (such as WLAN or Bluetooth).

There are two key challenges to achieve impromptu service discovery and provision. The first challenge is how to automatically discover the relevant services upon entering the smart environment, making reasonable assumptions on mobile devices and service providers who offer services in the specific smart environment;² The second challenge is how to automat-

ically provide appropriate services to the right person with the right form with the consideration of contextual information in smart environment. In this paper, we aim to propose context-aware service provisioning middleware based on user/device/situation context/service profile metadata with collaborative-based filtering approach to select appropriate services to newcomer user from similar neighbors.

The rest of the paper is organized as follows: Section 2 starts with use scenarios for context-aware service provisioning at the airport environment, from the use case six system requirements are identified for supporting impromptu service discovery and provision in unfamiliar open smart environments. After presenting the theoretical framework for context modeling and collaborative-based service provisioning in Section 3, the initial implementation are described in Section 4. Finally, some concluding remarks are made.

2. Motivating Scenarios

To introduce some impromptu service discovery and provision issues for open smart environment, let us start by considering the service provisioning scenarios in public places (e.g. airport, station, hotel, museum, shopping mall, etc.). In the public places, social characteristics among the users (e.g. career, friendship, common interests, and shared professional activities, etc.) play important roles for providing services in a open smart environment. In this paper, we focus on the issues of context-aware service provision with the consideration of social characteristics and service interaction history within a social community.

2.1. *Service Provisioning Scenarios in Public Places*

Let us consider the following service provisioning scenarios at the Aéroport international Charles de Gaulle where the users with different social characteristics use mobile devices to access personalized services in an unfamiliar smart environment.

- (1) Mike, who is a young backpacker from USA, visits the airport for the first time. When he arrives, he wants to find the way to the nearest bus station to the downtown. Besides, he wants to collect some tourist information about Paris. The system can provide the *FollowMe Service* with French version to guide him to the destination, and specialized services for the travellers, e.g. *Tourist Information Service*, *Weather*

Forecast Service, Map Service, etc. The system especially provides *Social Network Service* to help Mike find journey partner.

- (2) Alice, who is a young lady, is waiting the flight at the airport. She wants to buy some gifts for her friends before boarding. The system can provide *Shopping Service* to recommend some perfumes based on her shopping preference and *FollowMe Service* to guide her to the perfume shop.
- (3) George, who is a disabled elder with latent heart disease, visits her daughter in Paris. When he walks along the aisle in the airport, a heart attack happens suddenly. The system should provide *Emergency Call Service* for him to call for help. When he triggers this service, the airport crews analyze his health status, and predict the heart attack. And then the crews take the medical treatment to him immediately on following his location.

2.2. Requirement Analysis

From the scenarios described above, we can elicit the requirement for *user and environment context consideration* that support impromptu, secure and context-aware service discovery and provision with mobile devices. In order to present the right services to the right person in the right form, user and environment context should be taken into account for service discovery and provision in each smart environment. Besides, social characteristics and service interaction history among the users are useful for improving service provisioning for a social community.

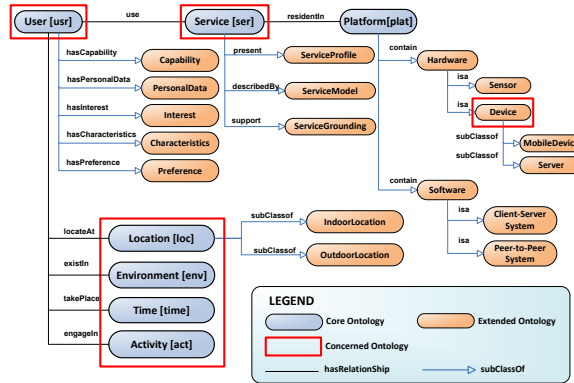
To address the above-mentioned requirements, we propose and implement a impromptu service discovery and access framework which enables the services in unfamiliar environments accessible to any WLAN enabled mobile device, according to individual user's preference, situation and device capability. From the scenarios above, we believe social semantic characteristics and service interaction history can facilitate service provision and reuse in a social community. For instance, when a traveler is waiting for the flight at the departure lounge, he might find partners with the same destination and share their journey experience. The system can provide *social networking services* that support human social community interaction. In the next section, we focus on collaborative-based service provisioning to deliver appropriate services to the end users with the consideration of semantic social characteristics and service interaction history.

3. Collaborative-based Service Provisioning

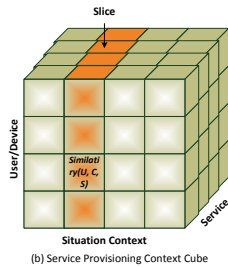
3.1. Modeling Context in Open Smart Environment

Context is the key feature in open smart environment to identify semantic entities to describe the situation of interaction between users and services.³ To distinguish the specific properties or attributes of various contexts, ontology-based modeling is adopted to describe and classify context concept and properties, including physical contexts (e.g. time, location), personal contexts(e.g. demographics, identity, preference, mood), device contexts(e.g. display size, power), activity contexts(e.g. journey plan, meeting schedule). Besides those, other types of information are still considered as crucial context which may be invisible to the participants e.g. systematic contexts (e.g. CPU power, network bandwidth), and application contexts (e.g. agents, services), environmental contexts (e.g. light, temperature), etc. In this paper, we present a partial ontology-based context model based on our previous work shown in Figure 1(a) which is divided into two layers: *core context ontology* for general conceptual entities in smart environment and *extended context ontology* for domain-specific environment. To build up *core context ontology*, we investigate seven basic concepts of User, Location, Time, Activity, Service, Environment and Platform.

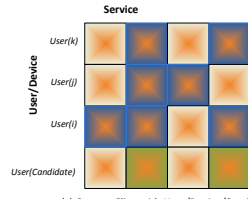
To illustrate the features of context-aware service provisioning issue derived from the scenarios above, we build up a context cube to identify the key elements of service provisioning in a smart environment, which is composed of User, Device, Service and SituationContext(including the environmental contexts, e.g. PhysicalEnvironment, Location, Time, Activity, etc.) shown in Figure 1(b). In the *Service Provisioning Context Cube*, the User ontology consists in classification of users type and demographics features of users (e.g. name, nationality, language, gender, age, etc.), and each categorized class will be linked with associated information, such as preferences, knowledge, capabilities, social relationship, and so on. The Device ontology describe the parameters of the user's mobile device capability, e.g. display device, networking connect, media support, memory size, etc.. The Service ontology defines the multi-level service specifications in order to support service discovery and composition. Each instance of Service presents a Service-Profile description, be describedBy a ServiceModel description, and supports a ServiceGrouping description in OWL-S specification. The Condition property and the additional hasPrecondition parameter in OWL-S specify one of the preconditions of the service which can be considered as the instance defined according to the restriction of user's situation contexts. The Situ-



(a) Context Ontology in Open Smart Environment



(b) Service Provisioning Context Cube



(c) Context Slice with User/Device/Service Profiles, when Situation Context is given

Fig. 1. Modeling context in open smart environment: (a) a partial context ontology that delineates the concepts and relationships between user’s situation context, user profile, device capability and service context; (b) service provisioning context cube to identify an array of user profile/device capability, service profile and situation contexts which is concerned as essential entities for service provisioning; (c) a slice returns the information on user profile, device capability and service profile in the current situation context (e.g. in the given co-location) to depict social semantic relation between users and services

ationContext ontology combines the metadata definition of Location, Time, Activity and PhysicalEnvironment ontologies to identify the user’s physical situation.

A *Slice* is derived from the *Service Provisioning Context Cube* to describe the social characteristics and semantic relationship between users and services when the active user’s situation is fixed (shown in Figure 1(c)). For instance in the scenario, when Mike is waiting for the flight at the airport lounge, he wants to find partners and information about the tour. In this case, the middleware collects all the co-located participates’ user/device profile and their in-use service profiles as a context slice, and recommends appropriate services to Mike from the slice with collaborative-based filtering approach.

3.2. Collaborative-based Service Provisioning

This section introduces the theoretical framework of collaborative-based service provisioning which is derived from user-based or memory-based collaborative filtering algorithm. For M services and K users, the joint profiles with user, device and situation context are presented in a $K \times M$ context slice user/device-service matrix X (Figure 2(a)). Each element $x_{k,m} = r$ indicates that user k rated service m by r , and $x_{k,m} = \emptyset$ means that the rating is not set. The matrix can be decomposed into row vectors:

$$X = [u_1, \dots, u_K]^T, u_k = [x_{k,1}, \dots, x_{k,M}]^T, k = 1, \dots, K \quad (1)$$

where T denotes transpose. Each row vector u_k^T corresponds to a user profile and represents a particular user's service ratings.

Collaborative-based service provisioning process based on user-based collaborative filtering algorithms is depicted as follows which predicts a test user's interest in a test service based on rating from similar user profiles:

- Step 1: Given the user/device profile set and situation context captured from the system, the service provision process first select appropriate user profile based on current user's situation context (e.g. location).
- Step 2: Each user profile (row vector in Figure 2(b)) is sorted by its dissimilarity towards the test user profile with *Pearson's correlation coefficient*:

$$w_{i,k} = \frac{u_i \cdot u_k}{\|u_i\| \times \|u_k\|} = \frac{\sum_{j=1}^p \alpha_j \beta_j}{\sqrt{\sum_{j=1}^p \alpha_j^2 \sum_{j=1}^p \beta_j^2}} \quad (2)$$

where $u_i = (\alpha_j), u_k = (\beta_j), (j \in 1..p)$ and $w_{i,k}$ returns the relevance weight of two contexts over all the item ratings from the system.

The set of similar users can be identified by selecting top- N (shown in Figure 2(b)). The set of top- N similar users towards user k can be generated as follows:

$$S_N(u_k) = \{u_a | \text{rank}_{w_{a,k}} \leq N, x_{a,m} \neq \emptyset\} \quad (3)$$

where $|S_N(u_k)| = N$ and $w_{a,k}$ is the similarity between user k and a .

- Step 3: We combine all the N neighbors' ratings into a prediction by computing the weighted average of the ratings, using the correlations as the weights as follows:

$$\hat{x}_{k,m} = \bar{u}_k + \frac{\sum_{u_a \in S_N(u_k)} w_{a,k} \cdot (x_{a,m} - \bar{u}_a)}{\sum_{u_a \in S_N(u_k)} w_{a,k}} \quad (4)$$

where \bar{u}_k and \bar{u}_a denote the average rating made by users k and a . The missing ratings can be replaced by a 0 score.

Step 4: We sort the rating of the predicted value $\hat{x}_{k,m}$ and select the top- H services for user interface generation:

$$S_H(s_m) = \{s_a | rank_{\hat{u}_{k,a}} \leq H\} \tag{5}$$

Though the weakness of user-based collaborative filtering is discussed in,⁴ the sparsity and scalability of user/device-service matrix is efficiently under control with the assumptions of use scenarios in practical smart environment.

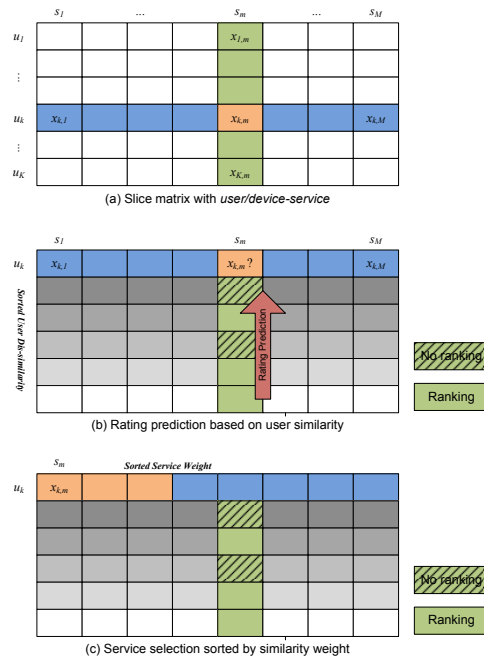


Fig. 2. A context slice matrix indicates user/device-service similarity relationship: (a) Slice matrix with user/device-service; (b) Rating prediction based on user similarity; (c) Service selection sorted by similarity weight

4. Initial Implementation

This section presents some initial implementation details based on collaborative-based filter to illustrate the context-aware service provisioning scenarios.

4.1. Environment Setup

We implemented a prototype as Figure 3 shown for the impromptu service discovery and provision framework using the Knopflerfish service platform.⁵ All the services are wrapped as OSGi bundles in the framework.

In each logical smart environment, we install the captive portal on a Linksys WRT54GL WLAN router. On entering the smart space, a mobile device (Apple iPhone or Samsung Q1 UMPC) is detected with a WLAN card and connected automatically to the router. After launching the web browser in the iPhone, it is automatically directed to the service portal showing the offered services. For aggregation of devices, we implemented a UPnP wrapper service built over Kono's CyberLink for Java UPnP library, providing web access capabilities. Links generated in the service portal point to the presentation URLs of the UPnP services.

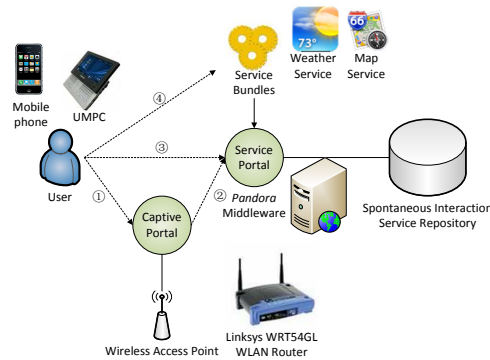


Fig. 3. Physical environment setup of the middleware for service provisioning. When the user enters a smart space, his mobile device is detected by the Captive Portal and connected to the wireless access point(1), and after launching the web browser, his web browser is directed to the Service Portal(2). He can then browse the available services(3), and invoke a service through his mobile device(4).

4.2. Aggregation of Services and Contents

The system contains a core with the functionality to install services as modules. Whole ranges of services can be added when aggregators based on existing protocols are added to the core. Some services can also be aggregator services which collate sub-services. However, these sub-services may not be visible directly to the core. In the case of the Media Service, it is an aggregator service that also aggregates media content from various

media sources, while being a service itself in the public place environment.

4.3. *Semantic Service Matching*

Using the aggregation of situation contexts and user profiles, rating or scoring process can be done to generate the vector description set. The system can calculate the semantic similarities with collaborative-based filtering algorithm of all services existed in the smart environment and provide appropriate numbers of personalized services to end users according to the capabilities of mobile device, such as *screen display size*, *media support type*, and so on. We use Taste collaborative filtering library⁶ as plug-in for semantic service matching engine and Pellet⁷ as OWL DL reasoner for context reasoning.

5. Conclusion

This paper presented an architecture for context-aware service provisioning based on user/device/situation context/service semantic profile with the consideration of users social relations and interaction history. It was demonstrated that several components based on OSGi framework could cooperate into efficient workflow to allow mobile clients with a web browser and wireless connectivity to spontaneously discover and access services with personalized service view.

References

1. M. K. S. Cuddy and H. Lutfiyya, Context-aware service selection based on dynamic and static service attributes, in *Proceedings of IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, (Montreal, Canada, 2005).
2. A. M. D. Q. Zhang and M. Mokhtari, A trustworthy framework for impromptu service discovery with mobile devices, in *Proceedings of the 4th Mobility Conference*, 2007.
3. T. Strang and C. Linnhoff-popien, A context modeling survey, in *Workshop on Advanced Context Modelling, Reasoning and Management*, (Nottingham, England, 2004).
4. J. K. B. Sarwar, G. Karypis and J. Reidl, Item-based collaborative filtering recommendation algorithms, in *Proceedings of the 10th international conference on World Wide Web*, (Hong Kong, 2001).
5. Knopflerfish <http://www.knopflerfish.org/>.
6. Taste project - collaborative filtering for java <http://taste.sourceforge.net/>.
7. Pellet - the open source owl dl reasoner <http://pellet.owldl.com/>.