

Ubiquitous Services: Enhancing Cyber-Physical Coupling with Smart Enablers

Hideyuki TOKUDA^{†a)}, Jin NAKAZAWA[†], *Members*, and Takuro YONEZAWA[†], *Nonmember*

SUMMARY Ubiquitous computing and communication are the key technology for achieving economic growth, sustainable development, safe and secure community towards a ubiquitous network society. Although the technology alone cannot solve the emerging problems, it is important to deploy services everywhere and reach real people with sensor enabled smart phones or devices. Using these devices and wireless sensor networks, we have been creating various types of ubiquitous services which support our everyday life. In this paper, we describe ubiquitous services based on a HOT-SPA model and discuss challenges in creating new ubiquitous services with smart enablers such as smart phones, wireless sensor nodes, social media, and cloud services. We first classify various types of ubiquitous service and introduce the HOT-SPA model which is aimed at modeling ubiquitous services. Several ubiquitous services, such as DIY smart object services, Twitthings, Airy Notes, and SensingCloud, are described. We then address the challenges in creating advanced ubiquitous services by enhancing coupling between a cyber and a physical space.

key words: *ubiquitous services, smart objects, environmental monitoring, sensing clouds*

1. Introduction

In early UbiComp research projects, to realize ubiquitous computing environment [1], researchers developed unique devices such as Active Badge [2], Active Bat [3], MediCup [4], Smart Furniture [5], Activity Recorder [6], and various testbeds such as EasyLiving [7], AwareHome [8], Smart Space lab. [9]. In many cases, these rooms equipped with UbiComp enablers such as sensors, devices and embedded computers are used to detect human activity and provide various services. However, the cost and time for building such smart spaces are high and the lack of ubiquitous connectivity to a real space is a barrier to the deployment of the ubiquitous computing environment everywhere.

After two decades in sharing Mark's vision [1], many computing and communication enablers became commercially available for creating ubiquitous services everywhere. In particular, many enablers can support for unique services in real as well as cyber spaces by coupling two spaces. We classify these enablers into two types: "personal" enablers and "global" enablers. Personal enablers allow us to empower our daily activities. These are smart devices such as smart phones, smart pads, active and passive RFIDs, smart cards and ultra low cost wireless sensors. These devices allow us to link any object in a real space into a cyber space easily. Global enablers allow us to link personal services

with existing global services in a cyber space without concerning the scalability issues. Global enablers are cloud computing services, such as amazon's EC2, S3, Google's public cloud services, and social network services such as Facebook and Twitters. These global enablers offer simple APIs to allow us to link many kinds of existing services without significant costs.

To create ubiquitous services effectively, we build a ubiquitous computing system with these enablers based on so called a "HOT-SPA" model. In the HOT-SPA model, the system consists of sensing (S), processing (P) and actuation (A) components. The "HOT" stands for "Here or There" in the SPA model and indicates the actual location of the component in a real space. We denote Sh, Ph, Ah, St, Pt, At for components located here or there respectively.

Sensing components (Sh+St) provide sensing functions of real objects or phenomena in a real space and capture the sensed data into a cyber space. Processing components (Ph+Pt) interpret captured sensed data and convert them to higher level information and decide a suitable action or a set of actions for the actuator. Actuation components (Ah+At) execute the requested action as the outcome of the processing components. Then, the requested action may change the status of real objects and it may initiate a new SPA cycle.

In the remainder of this paper, we present various ubiquitous services with smart enablers based on the HOT-SPA model and discuss the challenges in creating advanced ubiquitous services. In Sect. 2, we first describe the definition of ubiquitous services. The HOT-SPA model is discussed in details with the issues in realizing various types of ubiquitous services in Sect. 3. Section 4 describes case studies of ubiquitous services we have implemented in a real space. Section 5 addresses the challenges in creating advanced ubiquitous services and Sect. 6 summarizes the paper.

2. Ubiquitous Services

We define ubiquitous service as a service which can offer a set of functions or actions to support our everyday activities in real space. While web services can be initiated in a cyber space, ubiquitous services are initiated in a real space based on a user's situation or context.

For instance, a remote nursing care system for an elderly person offers a ubiquitous service. It may consist of a sensor attached to the elders mug cup (St), a care taker's

Manuscript received March 3, 2011.

[†]The authors are with the Faculty of Environment and Information Studies, Keio University, Fujisawa-shi, 252-0882 Japan.

a) E-mail: hxt@ht.sfc.keio.ac.jp

DOI: 10.1587/transinf.E94.D.1122

mobile phone (Ph), and a skypephone placed at the elderly person’s home (At). Suppose that, one day, the care system detects no movement of the elderly person’s mug cup during breakfast time, then the system may send a warning message and initiate a phone call from the care taker’s smart phone to the elderly person’s skypephone for checking out the person’s condition.

Similarly, a runners’ running support system also offers a ubiquitous service. A tiny sensor (Sh) attached to a running shoe or smart phone (Sh+Ph) can track the current location of the runner and share the runners’ running records. A global enabler such as a runners web page (At), like nike+’s challenge page [10], makes us allow to link a personal data to a team data to participate the longest-running team ranking battle. In this way, a simple connection between personal and global enablers makes us a richer ubiquitous services for supporting our activities.

2.1 Classification of Ubiquitous Services

The well known feature of ubiquitous services is pervasive nature of the service functions or actions. A ubiquitous service can be used at anytime, from anywhere and by anyone. We call this as an “Any3 (any-cube)” type service. Services are not designated for specific users in space and time. However, another type of services is often designated to a person or a specific group of people in space and time. We call this as an “Only3 (only-cube)” type service.

For instance, a group of people who bought a product X at a supermarket can get 50%-off of product Y. If the supermarket’s owner wants to offer further incentive to use e-cash, then the store can give further discount to users who used to pay the total amount by e-cash using their mobile phone. In this way, the store can reduce the congestion at the cashier area and customers who used e-cash could check out without additional waiting time.

We can further classify ubiquitous services based on the region covered by the services as shown in Fig. 1.

We divided the regions into 5 zones, namely i-, a-, b-, c- and d-zones. i-zone is around our body. a-zone is up to a room and b-zone covers up to a building. c-zone covers up to an outdoor space where a user can access the Internet and GPS. d-zone is an outdoor space without the Internet

and GPS connectivity. In an urban area, there are many tiny d-zone spots within c-zone and a ubiquitous service often faces difficulties with unexpected disconnection from infrastructure networks.

For instance, a location of a car, x as shown in Fig. 1, on a road can be mapped properly in a car navigation system as x' in a cyber space. Although the car navigation system with a smart phone can also provide a navigation service in c-zone, it may not be used in d-zone due to the lack of access to GPS connectivity. Similarly, when a user uses a personal indoor navigation system to find a location of a room in b-zone, the system needs to interact with an indoor location system such as active badge [2] or 3D bat [3].

2.2 Application Domains

Ubiquitous services can be created by coupling objects, people, places or phenomena in a real space with a cyber space. By just coupling two entities in real and cyber spaces, a service can offer a various kind of awareness to the users of the service. For instance, a medical training service for mental disorders, may offer a promising way of therapy by teaching patients to recognize and manage early warning signs by their own. Similarly, in personal mobile sensing applications, all members’ mobile sensors are connected to a global common web page. The service aggregates sensed data and visualize in the web page, then the service may improve the member’s awareness of the target environment. In urban traffic control systems, processing of sensed data and controlling of traffic signals and flows are essential functions. In general, processing functions provide mining, analysis, or optimization functions. Anonymizing functions for large volumes of sensed data is also a key function of such services.

These services are enhanced by using personal and global enablers in many application domains. Examples of useful service domains of ubiquitous services are as follows.

- Health Care and Medicine
- Education and Training
- Traffic and Transportation
- Agriculture
- Logistics
- Marketing
- Games

3. HOT-SPA Model

Designing a ubiquitous computing system for creating a ubiquitous service must overcome many systems issues. The HOT-SPA model allows us to model a ubiquitous system with a simple set of sensing, processing and actuation components. In this section, we describe the HOT-SPA model and discuss the issues with these components.

3.1 SPA and HOT-SPA Model

In a ubiquitous computing system, there are three basic com-

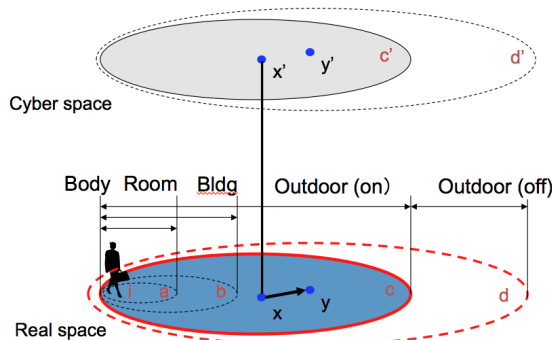


Fig. 1 Classification of ubiquitous services.

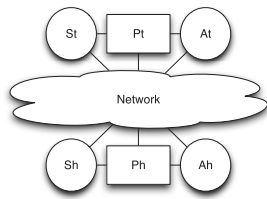


Fig. 2 HOT-SPA Model.

ponents: sensing (S), processing (P) and actuation (A) components. Sensing components consist of a set of wirefull or wireless sensors. These are often attached to real objects, people, building or places to capture the state of the target objects or phenomena in a real space. Processing components consist of embedded PCs, servers, PDAs and personal smart phones. These components process the sensed data to analyse the status of the target objects or phenomena, and generate a set of actions for actuation components. Actuation components are various types of appliances and devices such as information appliances, displays, speakers, PCs, smart phones, and network robots. By taking the S-P-A steps, the system can offer a sensible action and by taking the step repeatedly the service can enhance the coupling between a cyber and physical spaces.

In traditional ubiquitous computing systems, S-P-A components are self contained and the systems do not depend upon other resources and services. In modern ubiquitous computing systems, S-P-A components split into two side of a network: components on this side and the other side of the network. We call components on this side as “HERE” components, such as Sh, Ph, Ah, and on the other side as “THERE” components, such as St, Pt, At as shown in Fig. 2.

On the HERE side, “personal enablers” like a smart phone or wireless sensor nodes are popular devices. These components can provide sensing, processing and actuation functions easily. On the other side of the network, “global enablers” such as sensor web services (eg. SenseWeb [11], Pachube [12], airsage [13]), cloud services (eg. ec2 and sss), and social media (eg. Facebook and Twitters) are very popular components. These allow us to provide global sensing, processing and actuation functions easily.

3.2 Issues in HERE Components

Although many smart phones became commercially available for supporting ubiquitous services, ultra low cost wireless sensor nodes are still limited commercially.

One of the most important issues in HERE components is heterogeneity of wireless sensor nodes and their interoperability. Since there are many wireless sensor communication protocols, it is difficult to exchange raw data among heterogeneous sensors without using a gateway. In a pervasive health service, it would be better if a smart phone could communicate with various types of biomedical sensors without having additional cables or gateway boxes. Similarly, in

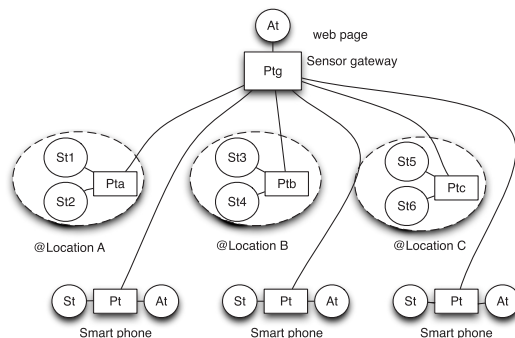


Fig. 3 Sensor gateway with remote embedded sensors.

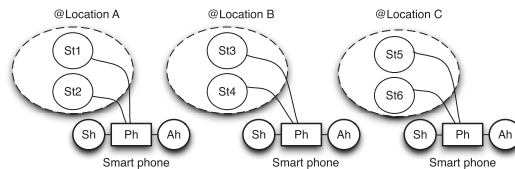


Fig. 4 Peer-to-peer interaction with remote embedded sensors.

personal mobile sensing applications, embedded urban sensors’ data must be uploaded to a web page (At) via sensor gateway (Ptg) as shown in Fig. 3. It would also be better if a user’s smart phone (Sh+Ph+Ah) could communicate with embedded urban sensors (eg. St1, . . . ,St6) such as weather, CO2, traffic sensors as shown in Fig. 4.

We can summarize the issues in HERE components as follows.

- Interoperability among heterogeneous sensors
- Local data and connectivity management
- Processing capability and battery life of smart phones
- Extensibility of smart phones

3.3 Issues in THERE Components

Global enablers such as sensor web services, cloud services, and social media are key to make a ubiquitous service available to more extended regions. By connecting processing components, Ph and Pg, a service can utilize a remote actuator At without extra cost as shown in Fig. 3. However, there are many issues in using global enablers to enhance the ubiquitous services.

In a personal mobile sensing applications, sensor data could be uploaded to a sensor gateway (Ptg) as shown in Fig. 3 by using an enabler like SenseWeb or Pachube. However, communication delay and jitter of update processing time of web page (At) may not be bounded. So, it is not easy to use these information for mission critical real-time applications.

Use of social media like Facebook or Twitter allows us to globally transmit various types of messages that indicate current status of people, objects or cities. However, it faces real-time issues as well.

We can summarize the issues in THERE components

as follows.

- Delay and jitter control of remote St, Pt and At
- Placement of remote data and code in St, Pt and At
- Interoperability among local and remote components (S-P-A)
- Security and Privacy enhancement

3.4 Issues in Connecting between HERE and THERE

SenseWeb and Pachube provide a simple way to integrate heterogeneous sensors from various spaces using XML-based interface. Cloud services also provide unique interface at IaaS-, SaaS-, PaaS-level functions. However, we face a lack of standard description of services, contexts and user profiles for ubiquitous services.

We can summarize the issues in connecting between HERE and THERE components as follows.

- QOS and routing control of communication networks
- Interface definition of S-P-A components
- Standard description of services, contexts and user profiles
- Security and privacy enhancement in HOT-SPA

4. Case Studies

In this section, we present examples of ubiquitous services we have demonstrated in various places [14],[15],[17],[18].

4.1 DIY Smart Object Services

By attaching a tiny wireless sensor node to a user’s belongings, the user can augment an object digitally in a cyber space and associate the object with various services such as status monitoring or preventing lost property. The notion of DIY (do-it-yourself) smart objects services is that non-expert users may create a smart object services by themselves without concerning an association between a real object and corresponding identifier of a wireless sensor node. Figure 6 shows a HOT-SPA model of the DIY smart object services with a user’s smart phone or PC.

An instantiation of smart object service requires a three-step process: 1) attaching a sensor node to every object, 2) making semantic associations between the sensor nodes and the objects, and 3) configuring an application to accommodate the objects with a preferred setting.

When using smart object services, a semantic association between a sensor node and a domestic object must be made before initiating any services. At a home environment, however, professional assistance with such instantiation may be either unavailable or too costly. To solve this problem, we proposed Spot & Snap interaction [14] that eases of such association with use of a USB camera with an LED spotlight (see Fig. 5). With Spot & Snap, non-expert users can register their belongings to preferred services without experts. Consequently, it provides application

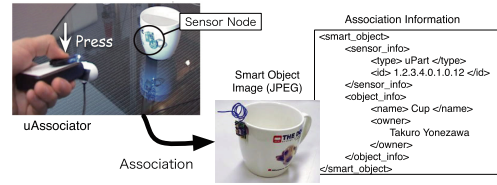


Fig. 5 Spot & Snap: Association between an object and a sensor node.

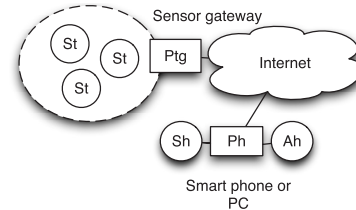


Fig. 6 HOT-SPA model of smart objects services.

framework for programmers to create various smart object services.

When creating smart object services, users should define events, such as beverage in a cup turning cold or someone sitting down on a chair, using physical values from sensors. The most common event definition for end-users is simply describing threshold of sensor values and boolean operation. When they want to define more complex events, such as multiple people sitting down on chairs or a user starting studying using a pen and a notebook, they need to use a programming language. To define such complex event easily without a complex programming language, we developed a new event description language called SOEML [16] based on temporal relation among simple events. We implemented a user interface for defining events and action and associating them (see Fig. 7). By using the interface, users can load, visualize its structure, edit and save the services as XML code. The interface provides a structure of the components visually with animation. As common problem running through all of hierarchical description method, it is difficult to manage elements as the number of them grows. To reduce this problem, the interface employs Zooming User Interface to manage services easily.

A simple service, called uViewer, assists a user in monitoring his or her personal belongings even when these objects are out of sight. It reports the changing status of the objects visually with images created by uAssociator (see Fig. 8). For instance, when an object moves, its image also moves. This visual presentation of the object status enables the user to intuitively understand the object’s whereabouts. In addition to this basic service, uViewer is capable of presenting the history of a particular object’s status. With this type of information, the user can tell how frequently the object is being used. All in all, uViewer extends the user’s ability to manage their personal belongings.

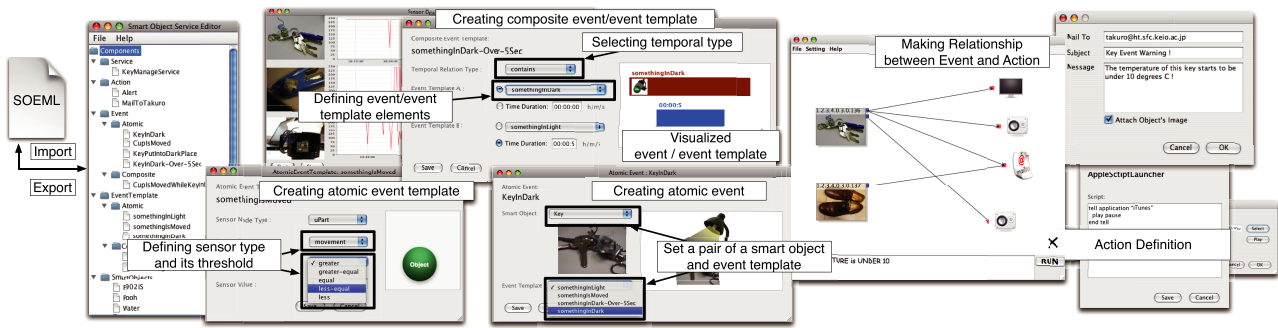


Fig. 7 Event and action association for smart objects.

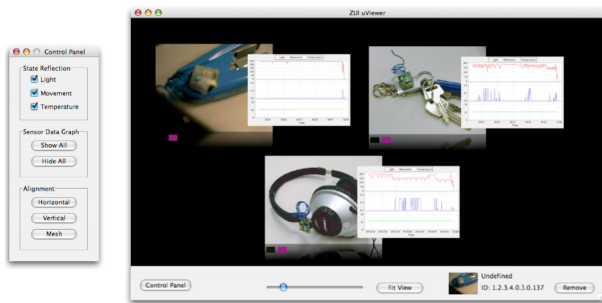


Fig. 8 uViewer: a simple object status viewer.

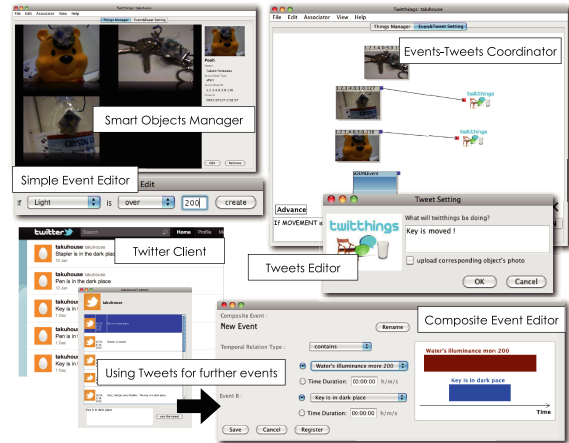


Fig. 10 Twitthings screen image.

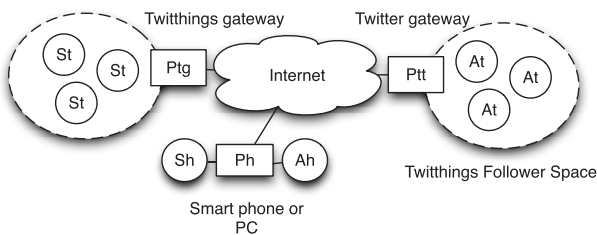


Fig. 9 HOT-SPA model of Twitthings.

4.2 Twitthings

Twitthings service is an extension of the smart object service by using social media, namely Twitter, as a group of remote actuators. The smart object service model shown in Fig. 6 is limited to a personal smart phone or PC as the actuator (Ah). However, by connecting Ptg with Twitter (Ptt) as a global enabler, Twitthings can utilize a group of actuators (At) and provides a service extensibility as shown in Fig. 9

Similar to the event and action association process shown in the previous section, a user can define an event and tweet messages associated with Twitthings as shown in Fig. 10. If you follow the user in Twitter, you could see all incoming messages. These messages are automatically generated by digitized real objects in a cyber space. The users can also define new events with the messages. For example, assume that a user receives messages regarding to a laboratory, “All chairs are used” and “Projector is used” from Twitthings. The user can define a new event “meet-

ing” by associating it with those two messages. Once the event is registered to Twitthings platform, the events will be detected automatically, and users can use the events for various action such as posting messages to Twitter or controlling appliances. The advantage of Twitthings is that it enables users to share and define smart object event seamlessly.

4.3 Airy Notes

In Airy Notes project [17], we monitored environmental condition using ultra low cost wireless sensor modules and showed the characteristics of places intuitively on a map in web (At) as well as on user’s cell phone (Ah). Unlike a web-based monitoring system, we have attached a 2-D bar code with each sensor, so that a user can sense the data in front of the sensor using a cell phone’s camera. A HOT-SPA model is a hybrid model of a sensor gateway and P2P model as shown in Fig. 11

The Airy Notes system is designed to help easy understanding of the effect of greening by enabling high density placing of sensor nodes. We conducted the experiments of the system in Shinjuku Gyoen with 160 sensors. The garden is known as taking part of keeping urban environmental condition calm with its green. We monitored the change of temperature, and observed that the temperature of the inner

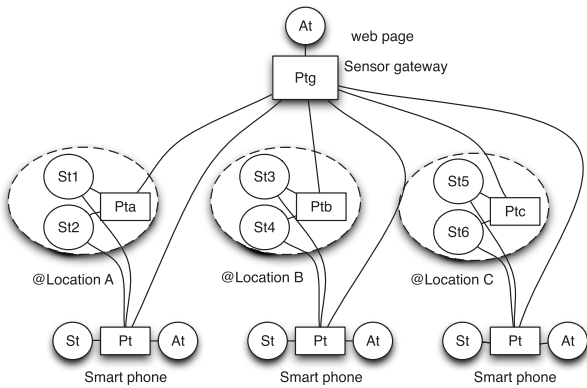


Fig. 11 HOT-SPA model for Airy Notes.



Fig. 12 Access from a cell phone.

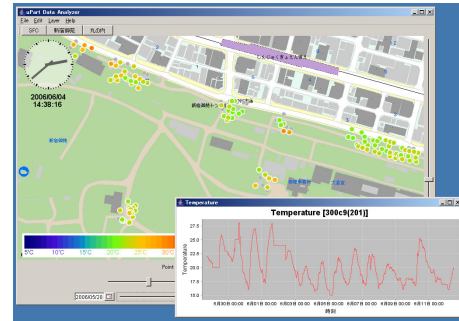


Fig. 13 Map-based Real-Time Monitor on Web.

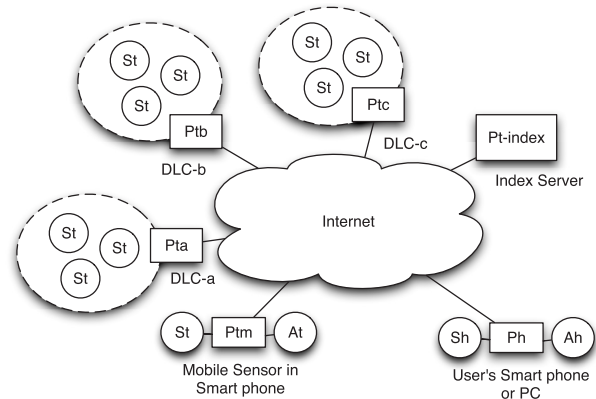


Fig. 14 HOT-SPA model for SensingCloud.

place was lower than the edge of the garden.

On-site Observation: Ah

Airy Notes System enables on-site observation of current temperature and its temporal change on a cell phone of any user. As shown in Fig. 12, users can casually see the characteristics of the environment by checking QR-code on the sensor at the point they are in. They would understand the environmental condition with their body sensation.

Real Time Monitoring on Web: At

The data shown on the viewer are updated in real-time. Unlike existing environmental monitoring systems which usually use a sensor module with a local storage unit, Airy Notes System utilizes wireless sensor nodes, and an on-line database system. Users can see the latest condition of the target region anytime. Figure 13 is a map on which live temperature are shown as colors.

4.4 SensingCloud

SensingCloud service enables applications to subscribe sensor data from arbitrary membership sensor nodes. While Airynote uses a set of homogeneous sensor nodes, SensingCloud is aimed at heterogeneous sensor networks with a single view. Core components of SensingCloud are Distributed Process Client (DPC), Index Server and various types of

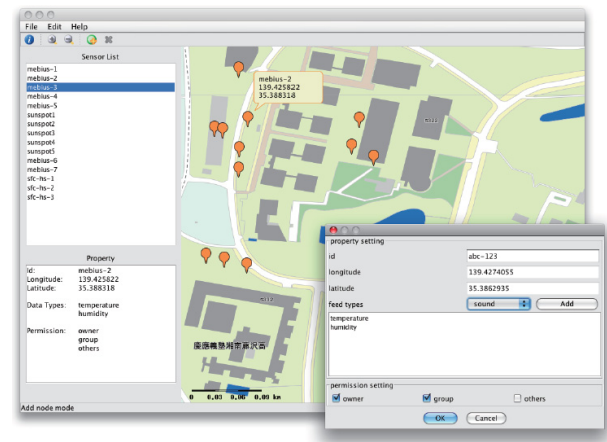


Fig. 15 UI part of Distributed Process Client.

sensors including mobile sensors in smart phones in a global real space. A HOT-SPA model of SensingCloud is shown in Fig. 14

Distributed Process Client is a GUI based sensor network management tool running as a bridging node to the Internet as shown in Fig. 15.

Although the front end of this client is GUI based network manager, it bridges the data stream from local sensor network to the Internet. Also, they dynamically turns to be a server which collects data stream from other clients and

provide sensor feeds to the applications. DPC is a part of distributed computational resource in SensingCloud. Index Server resolves IP addresses of relevant DPCs from meta-level requests from applications and sends control commands to them. A user of SensingCloud can interact with the heterogeneous sensor networks scattered all over the Internet as a virtual large scale sensor network.

Based on Distributed P2P Model with hybrid P2P architecture, we have designed SensingCloud's scaling mechanism. When the user application requests for sensor feeds via a smart phone or a PC, 1) it accesses to index server to resolve which sensor network provides target feeds. 2) The index server resolves the IP addresses of the DPCs connected to the target sensor networks, and nominates a master node from the nodes geologically close to target DPCs. 3) The index server sends Aggregation Request, which includes contents of query, IP address of the source sensor networks and that of the destination, which is the IP address of the application that has requested the feeds. 4) Then, the master node sends Feed Request to the DPCs connected to the target sensor networks. Feed request contains IP address of the master DPC and types and conditions of the feeds to be forwarded to the master DPC. After this step, 5) the DPCs forward the feeds from their sensor networks to the master DPC. 6) The master DPC aggregates the feeds and sends to the user application.

5. Challenges in Enhancing Cyber-Physical Coupling

We have presented examples of ubiquitous services that were improved by using personal and/or global enablers such as smart phones, ultra low cost wireless sensors, social media and P2P computing components. However, we are still facing many challenges for creating advanced ubiquitous services by enhancing cyber-physical coupling.

- Scalability in HOT-SPA architecture

A global enabler allows us to connect many Sh-Ph-Ah components with global St-Pt-At components easily. However, as we presented in Twitthings and SensingCloud, scalability and real-time issues are still remaining. For instance, a large event handing process is migrated into Twitter system, however, a user of Twitthings does not have any control over their response time. In command and control applications, it may face a problem.

- Cross-domain Mashup

In web service development, a mashup is a well known method to create a new web service by combining data or functions from two or more existing web services on demand. Mashup is important to make existing services and resources more useful. Among ubiquitous services, it is still difficult to create a mashup based on existing S-P-A components. In particular, many sensors and actuators are used in different application domains, we must overcome the lack of interoperability among sensors and actuators. Network robots have a

potential to be personal or global enablers, however, we do not have common APIs nor a common directory to find a suitable robot for a specific functionality as an actuator.

- Interoperability among heterogeneous components

There are many useful sensors and actuators for creating advanced ubiquitous services. Among cooperating sensors and actuators, command and control protocols must be matched between a sender and receiver. In order to improve the interoperability of components, we need to create a better protocol conversion mechanism as well as universal components description language. By improving interoperability among heterogeneous sensors and actuators, a ubiquitous service could be created in a new application domain.

- Security and Privacy Enhancing Technology

Social media, such as Facebook and Twitter, allows a ubiquitous service to utilize large amount of user profiles for creating highly personalized or context-aware services. However, for instance, real-time location information of a group of people has a potential to lead location-based discrimination in a small community. These privacy related information should be controlled by individuals to maintain time and space accountability. In particular, auditability and visibility of such data flows should be managed.

6. Conclusion

We presented ubiquitous services with smart enablers based on the HOT-SPA model. The HOT-SPA model allows us to model a ubiquitous computing system with a simple set of sensing, processing and actuation components. Unlike traditional ubiquitous computing systems, modern systems split S-P-A components into "HERE" and "THERE" sides of the network. Personal and global enablers, such as smart phones, web, cloud and social media, are key to provide global sensing, processing and actuation functionalities. We also addressed that the scalability, cross-domain Mashup, interoperability, security and privacy issues are key challenges in creating future advanced ubiquitous services.

Acknowledgments

We thank all members of Tokuda Lab. for their dedicated efforts to develop better ubiquitous services.

References

- [1] M. Weiser, The Computer for the 21st century, *Scientific American*, vol.265, no.3, pp.66-75, Sept. 1991.
- [2] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The active badge location system," *ACM Trans. Information Systems*, vol.10, pp.91-102, Jan. 1992.
- [3] A. Ward, A. Jones, and A. Hopper, "A new location technique for the active office," *IEEE Pers. Commun.*, vol.4, no.5, pp.42-47, Oct. 1997.

- [4] H-W. Gellersen, M. Beigl, and H. Krull, "The MediaCup: Awareness technology embedded in an everyday object," 1st Int. Sym. Handheld and Ubiquitous Computing (HUC99), Karlsruhe, Germany, 1999.
- [5] H. Tokuda, K. Takashio, J. Nakazawa, K. Matsumiya, M. Ito, and M. Saito, "SF2: Smart furniture for creating ubiquitous applications," IEEE Proc. International Workshop on Cyberspace Technologies and Societies (IWCTS2004), pp.423-429, Jan. 2004.
- [6] J. Yura, H. Sakakibara, J. Nakazawa, and H. Tokuda, "Activity recorder: A device to record user's activities using RFIDs and sensors," IEICE Trans. Commun., vol.E90-B, no.12, pp.3480-3495, Dec. 2007.
- [7] B. Brumitt, B. Meyers, J. Krumm, A. Kern, and S. Shafer, "Ea-sy-living: Technologies for intelligent environments," Handheld and Ubiquitous Computing, 2000.
- [8] K. Cory, R. Orr, G. Abowd, C. Atkeson, I. Essa, B. Mac-Intyre, E. Mynatt, T. Starner, and W. Newstetter, "The aware home: A living laboratory for ubiquitous computing research," International Workshop on Cooperative Buildings, 1999.
- [9] T. Okoshi, S. Wakayama, Y. Sugita, S. Aoki, T. Iwamoto, J. Nakazawa, T. Nagata, D. Furusaka, M. Iwai, A. Kusumoto, N. Harashima, J. Yura, N. Nishio, Y. Tobe, Y. Ikeda, and H. Tokuda, "Smart space laboratory project: Toward the next generation computing environment," IEEE 3rd International Workshop on Networked Appliances, pp.115-121, March 2001.
- [10] <http://nikerunning.nike.com/>
- [11] A. Kansal, S. Nath, J. Liu, and F. Zhao, "Senseweb: An infrastructure for shared sensing," IEEE Multimedia, vol.14, no.4, pp.8-13, Oct.-Dec. 2007.
- [12] Pachube "Data infrastructure for the internet of things," <http://www.pachube.com/>
- [13] Airsage, "Live, real-time traffic information, real-time travel times, location, movement data," <http://www.airsage.com/>
- [14] T. Yonezawa, H. Sakakibara, J. Nakazawa, K. Takashio, and H. Tokuda, "Spot & snap: A bootstrap interaction for DIY smart object services," 8th International Conference on Ubiquitous Computing (UBICOMP2006) Video Session, California, USA, Sept. 2006.
- [15] T. Yonezawa, H. Sakakibara, K. Koizumi, S. Miyajima, J. Nakazawa, K. Takashio, and H. Tokuda, "uPackage: A package to enable do-it-yourself style ubiquitous services with daily objects," 4th International Symposium on Ubiquitous Computing Systems (UCS 2007), Tokyo, Japan, 2007.
- [16] T. Yonezawa, J. Nakazawa, G. Kunito, T. Nagata, and H. Tokuda, "SOEML: Smart object events modeling language based on temporal intervals," International Journal of Multimedia and Ubiquitous Engineering, vol.4, no.3, pp.73-91 July 2009.
- [17] M. Ito, Y. Katagiri, M. Ishikawa, and H. Tokuda, "Airy Notes: An experiment of microclimate monitoring in Shinjuku Gyoen Garden," 4th International Conference on Networked Sensing Systems (INSS 2007), pp.260-266, Braunschweig, Germany, June 2007.
- [18] M. Ito, Y. Katagiri, M. Ishikawa, and H. Tokuda, "Airy Notes project: Creating landscape planning method for the Ubicomp World," International Workshop on Hybrid Design Practice: Situating Ubicomp's Interdisciplinarity, Conjunction with Ubicomp 2009, pp.89-90, Sept, 2009.
- [19] J. Nakazawa, J. Yura, S. Aoki, M. Ito, K. Takashio, and H. Tokuda, "A description language for universal understandings of heterogeneous services in pervasive computing," IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC 2010), Newport Beach, California, USA, June 2010.



Hideyuki Tokuda was born in 1952. He is Dean of Graduate School of Media and Governance and a Professor in the Faculty of Environment and Information Studies, Keio University. His research interests include ubiquitous computing systems, smart spaces, sensor networks, distributed real-time operating systems and embedded systems. He is a member of ACM, IEEE, IPSJ and JSSST.



Jin Nakazawa was born in 1975. He is a lecturer at Graduate School of Media and Governance, Keio University. His research includes mobile code architecture, distributed component software architecture, and home area networks. He is a member of ACM, IEEE, and IPSJ.



Takuro Yonezawa was born in 1981. He received Ph.D. degrees in Media and Governance from Keio University in 2010. He was engaged in the study of ubiquitous computing and context-aware system. His current major activity is system and interaction with internet of things. He is a member of IPSJ.