Service Delivery Platform Technologies

● Takafumi Chujo ● Masafumi Katoh ● Akira Chugo
● Hisayuki Sekine

(Manuscript received April 8, 2009)

The penetration of broadband networks is playing a key role in bringing about major changes in the way that network-based services are provided. New schemes such as software as a service (SaaS) and platform as a service (PaaS) let users access a variety of services over networks. Applications, computers, storage, and other resources combined over the network enable the rapid delivery of diverse services. Fujitsu is working to provide a low-cost, quick, and highly reliable SaaS service based on the technologies it developed in constructing information technology systems and on operational know-how accumulated at data centers. In response to this shift to a service economy, it will become increasingly important for the infrastructure of the network society to increase the value of the network and build a service platform that can deploy new services promptly. In this paper, we introduce future directions of network services, describe a service delivery platform architecture, and present the research and development of its key technologies.

1 Introduction

Broadband networks are penetrating deeply into homes and enterprises, and the spread of video delivery and Internet protocol television (IPTV) services is expected to increase network traffic by 1.4 times every year.[1] From a usage viewpoint, the way in which network-based services are delivered is undergoing major changes. Schemes like software as a service (SaaS)[2] and platform as a service (PaaS) are giving users access to a wide variety of services via the network, and it is becoming possible to provide diverse services by combining whatever resources are needed—applications, computers, or storage—through the network. In response to these changes, it is becoming increasingly important to build a service platform that can transform network value and provide new services promptly as an infrastructure of the network society.

In this paper, we introduce future directions of a service-oriented network and present Fujitsu's R&D activities related to a service delivery platform suitable for these trends.

2. Shift toward a service-oriented network

A service-oriented network interconnects services, terminals, and devices to provide value-added services. In order to create new markets and services, we aim to transform the value of a network in the following two ways.

The first transformation, as shown in **Figure 1**, would be to enhance the effectiveness of information and communications technology (ICT) by gathering information from diverse locations via sensors as a service front-end and using that information to obtain a detailed understanding of situations in the field. In agriculture and medical care, the use of sensing technologies and wireless technologies in this way could increase the effectiveness of ICT significantly.

The second transformation would be to provide a place for information sharing among service providers. Information brought together from various channels would be interrelated and integrated to draw a big picture of the real world. This value-added information could then be provided to service providers, resulting in further evolution of ICT. An example in the retail business[3] is shown in **Figure 2**. In this field, multi-channel retailing using multiple sales channels such as retail stores, Web sites, and catalogs is now showing an upward trend. However, each of these channels has its own operations department (or system), which prevents information from being shared among them sufficiently. This situation makes it difficult to understand rapidly changing customer needs. If various types of data and messages originating from a customer are interrelated across the channels, it should be possible to uniformly manage and visualize the customer behavior across all channels. For example, point of sale (POS) data at the retail store, hypertext transfer protocol (HTTP) requests to the Web site, and voice-over-Internet-protocol (VoIP) calls to the call center are interrelated using a commonly specified customer attribute as a key. In short, obtaining an understanding of the customer experience over all channels should

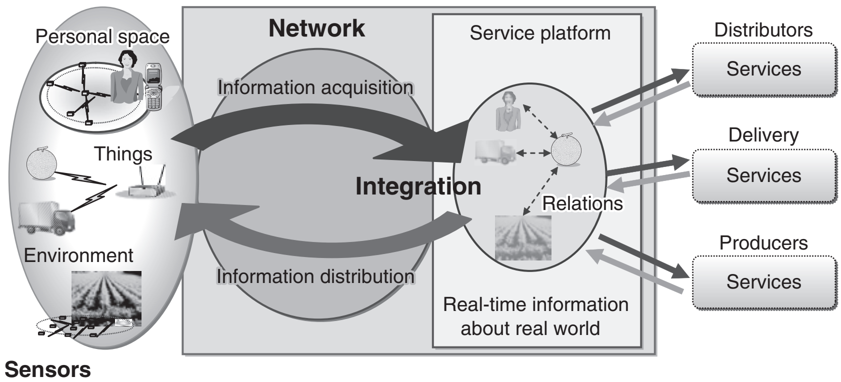

Figure 1
Creation of new markets through transformation of network values.

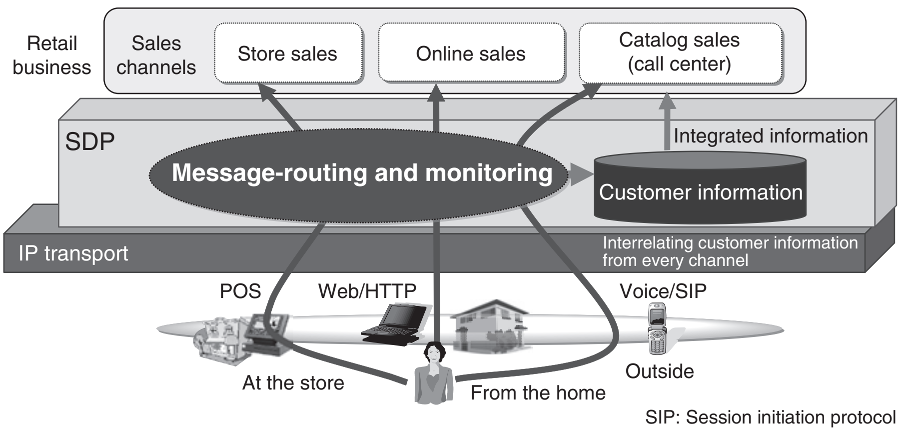

Figure 2
Example of services: cross-channel visibility.

make it possible to improve the brand image of the retailer.

Fujitsu is now researching and developing such a service-front-end function as a service delivery platform (SDP). Its architecture and enabling technologies are introduced in the next section.

3. Service delivery platform: SPACE

The Service Platform for Awareness, Creation and Experience (SPACE) incorporates terminals and networks as service components with the aim of enhancing *awareness* of situations in both the virtual and real worlds, *creating* new services through the use of service components, and enabling users to *experience* the world of these new services.

3.1 Architecture of SPACE

The SPACE is a platform for creating and executing a wide range of services and plays the role of a *place* for growing and disseminating new services and values. Its total architecture is shown in **Figure 3**.

The SPACE provides a wide variety of common functions in the form of service components such as those related to network control so that application developers can develop applications without advanced or specialized knowledge of terminals and networks. It also aims to reduce service development costs by opening up these service components to service providers and other involved entities (stake holders) and then by enabling many players to share those service components.

In the SPACE, services and terminals are

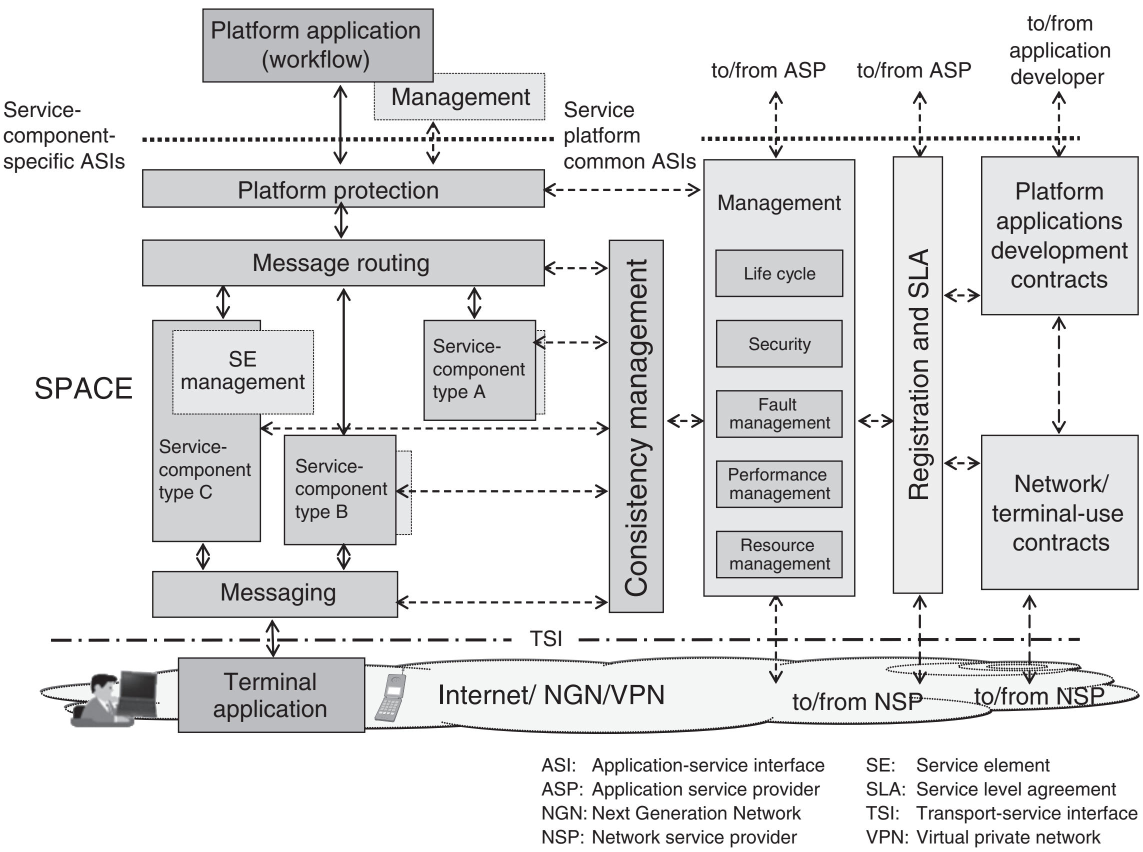

Figure 3
Architecture of SPACE.

virtualized so that applications can use these service components easily. Since servers that execute services can be expected to change dynamically to optimize work loads and electric power consumption at data centers, services must be virtualized in such a way that they are called not by the server address but by the service name.

The SPACE is not only concerned with information in the virtual world, but also gathers various types of real-world information from terminals and devices connected to the networks and enables service providers to utilize and exchange that information. This corresponds to a place where service providers can share this information and work together closely. However, if such explosively growing information were presented in its original, chaotic form, it would not by itself be able to increase the value of this place. It is important to present that huge volume of information after it has been organized in various ways so that the people receiving it can experience revelations from it. With the aim of transforming information into value, the following aspects are being studied for the SPACE.

1) Quantitative analysis

Obtains data from sensors to collect events that could not be observed in the past and analyzes those events in a quantitative and scientific manner

2) Time-based statistics

Presents the temporal fluctuation of accumulated information in a statistical manner to suggest various types of transitions such as changes in future demand

3) Space-based statistics

Presents characteristics of individual geographical regions so as to provide a basis for optimizing and customizing services for a specific geographical region

4) Correlation analysis

Determines steady states in order to present variations from steady-state and cause-and-effect relationships between observed events

Below, we describe three enabling technologies for realizing these SPACE concepts: service componentization, service virtualization, and ID information repository.

3.2 Enabling technologies

3.2.1 Service componentization

In the SPACE, service components are designed to have a layered configuration consisting of a basic service component layer and an abstract service component layer. The purpose of the basic service component layer is to hide the complexity of the protocols and terminals while that of the abstract service component layer is to gather real-world information in real time and to provide that information in an application-friendly form. This two-layer structure lets application developers use either abstract or basic functions as they require. In addition, functions and capacity can be expanded in response to changing demand by using both component types as units for expansion. As shown in Figure 3, various types of interfaces are provided depending on the level of application abstraction and three types of service components are defined, as described below.

1) Service component type A

This in an abstract service component that abstracts information collected over the networks or from terminals and functions to facilitate the execution of the application. As its main functions, it gathers information widely dispersed over the networks, reshapes it into an easy-to-use form, and delivers necessary information to terminals. It communicates via high-level logical interfaces (labeled ASI for application-service interface in the figure) in order to provide information and functions with a high-level of abstraction to a platform application.

2) Service component type B

This is a basic service component that hides the intrinsic technologies of the networks and terminals and intermediates the access of type-A service components and applications to the networks and terminals. As its main functions, it terminates lower-layer communications protocols dependent on the networks and terminals; manages network and terminal status during application execution; and converts protocol, code, and media. It terminates the communications protocol between a network/terminal and a service component via a transport-service interface (TSI).

3) Service component type C

This is a composite service component that combines type-A abstract functions and type-B basic functions. This one-layer model can be effective in simplifying the structure in an early stage when the numbers of service components and also applications are small.

3.2.2 Service virtualization

In the SPACE, a platform application combines service components and terminal applications by message routing to compose a service. To provide a persistent connection environment for external platform applications and terminal applications, internal mechanisms such as service component execution are "virtualized". As described below, external function entities can access function entities within the SPACE by a persistent method even if the SPACE's internal system configuration changes dynamically, for example, as a result of load balancing, changes in service components, or system switching in the event of failure.

1) Load balancing

Multiple function entities are prepared within the SPACE to avoid the performance bottleneck of the system. An external entity that wishes to execute a particular function entity specifies it using its unique name. On the basis of that name, the SPACE allocates one of the internal function entities with the smallest load for load balancing. Therefore, the specific internal function entity that performs the execution of that particular function entity is hidden to the outside.

2) Changes in service components

In the SPACE, different function entities can be arranged into a single group and only a proxy for that group is presented to external entities. When the proxy of a group is called by an external function entity, the SPACE will select a function entity within the group to be executed. This means that software upgrades to service components and the addition of new service components can be performed without having to inform the outside of such changes.

3) System switching in event of failure

If a system failure occurs and a certain service component cannot be executed, that service component will be switched over to a standby service component.

To achieve these virtualizations, the message routing function in the SPACE analyzes the XML-based message (XML: extensible markup language) received from a platform application or terminal application and forwards that message to the logical entity determined by that analysis.

During the execution of an application program as a combination of multiple processes, the failure to pass states and parameters between the processes correctly can result in a logical inconsistency. To prevent such logical inconsistencies from occurring, the message routing function has a consistency management function.[4] The message routing identifies one entity to execute a process and transfers messages to that entity during the execution of that application. In short, once an initial request message has been assigned to a certain entity, the message routing transfers all subsequent messages to that entity. In this way, an application developer can devote his or her efforts to developing an application program without having to be aware of the internal execution mechanism in the SPACE.

3.2.3 ID information repository

The ID information repository component

stores various kinds of information collected from the field such as sensor and presence information as ID information and delivers it to applications. It is thus a key element for transforming the value of the network. It combines the following three technologies[5] to store such ID information from the field in a form usable by applications for each person and thing. Here, an ID (identifier) refers to a object like a person or thing and ID information (identity) refers to all attribute information, including various types of information from the field associated with that ID.

1) ID information integration

ID information integration technology integrates and systematically stores "field information that changes in real time" and "attribute information that changes relatively infrequently" for each person and thing (**Figure 4**). Providing this technology as a common platform for multiple applications eliminates the need for application developers to develop new programs for processing, consolidating, and storing such data in each application. This technology also provides a common platform for the protocols to connect to ubiquitous sensor-equipped devices and for data processing of sensor information and enables application developers to make use of information gathered from the field through the use of an easy-to-program Web interface. In the above manner, applications can be easily developed without specialized knowledge or skills about individual sensor characteristics and the network protocols.

2) Relationship management

Relationship management technology enables information about one object to be reused with information about another object, based on the relationship between the two objects. For example, as shown in **Figure 5**, the location management of a package that is currently being loaded onto a truck will not only use the radio-frequency identification (RFID) tag attached to the package, but also reuse location information obtained from the truck's global positioning system (GPS).

3) Event generation and notification to reduce

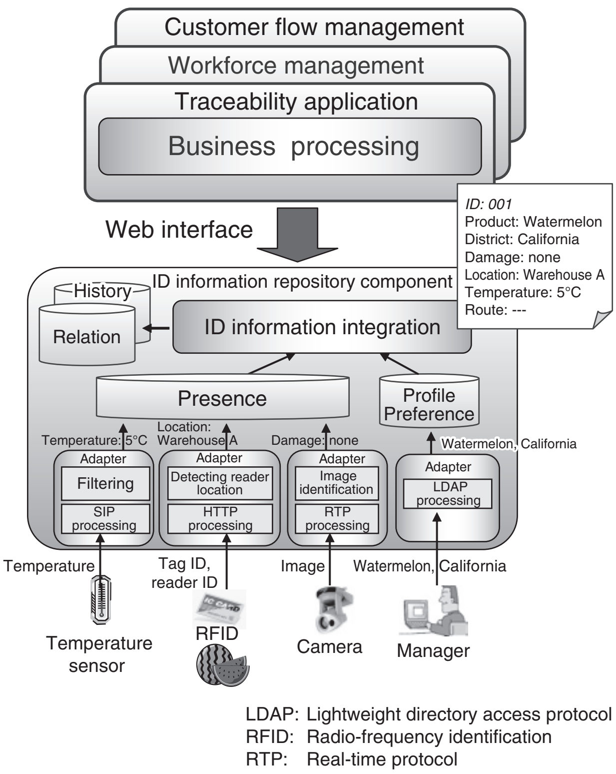

Figure 4
ID information repository technology (ID information integration).

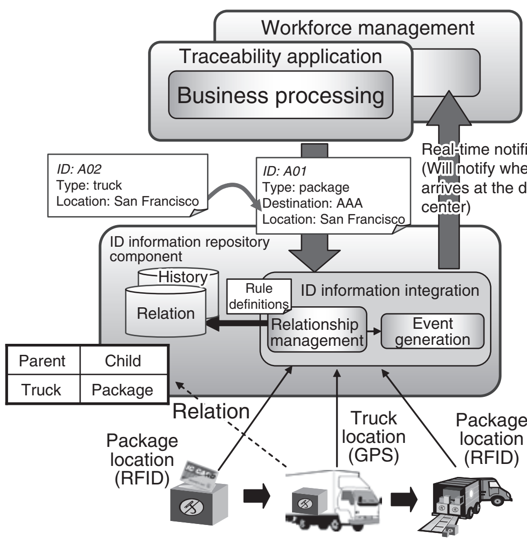

Figure 5
ID information repository technology (relationship management).

system load processing

To detect changes in information from the field, the SPACE incorporates a real-time event notification engine for applications. This eliminates the need for applications to make periodic checks of changes, so it reduces the system's processing load.

4. Conclusion

In this paper, we introduced Fujitsu's R&D activities related to a service delivery platform with the aim of transforming network value and creating new services. This transformation opens up many opportunities. For example, we foresee the expansion of ICT application areas and the creation of new business fields; more innovation in the workplace based on real-time information dealing with people, things, and the environment; and the creation of novel and collaborative services between service providers. To make these possibilities a reality, Fujitsu will continue to research and develop enabling technologies while promoting the value of network use in the field and supporting and expanding its solution business.

References

1) National Institute of Information and Communications Technology: Diversity and Inclusion: Networking the Future. (in Japanese), September 2008.
http://nwgn.nict.go.jp/report/NWGN-Vision-NICT-Report-V1-2008.pdf
2) K. Satake: Fujitsu's Activities for SaaS. (in Japanese), *FUJITSU,* Vol. 59, No. 1, pp. 14–19 (2008).
3) M. Hisatomi et al.: Application of Service Delivery Platform for Supply Chain Management. WTC 2008, 2008.
4) K. Matoba et al.: Architecture and key technologies of the next generation service platform. WTC 2008, 2008.
5) J. Maeda: Toward Advanced ID Management Systems. (in Japanese), Proceedings 2 of the 2008 Communications Society Conference of IEICE, BS-6-1, S-43-S-44, 2008.

Takafumi Chujo
Fujitsu Laboratories Ltd.
Mr. Chujo received a B.E. degree in Electronics Engineering from Kanazawa University in 1976 and an M.E. degree in Electronics Engineering from Nagoya University in 1978. He joined Fujitsu Laboratories Ltd., Kawasaki, Japan in 1978 and has been engaged in research and development of a broad range of network systems, including SDH, ATM, and WDM transport systems, network management systems, broadband wireless systems, and next-generation service platforms.

Akira Chugo
Fujitsu Laboratories Ltd.
Mr. Chugo received B.E. and M.E. degrees in Electrical Engineering from Waseda University, Tokyo, Japan in 1981 and 1983, respectively. He joined Fujitsu Laboratories Ltd., Kawasaki, Japan in 1986. Since then, he has been engaged in research and development of next-generation network systems. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan.

Masafumi Katoh
Fujitsu Laboratories Ltd.
Mr. Katoh received B.E. and M.E. degrees in Information Engineering from Yokohama National University, Yokohama, Japan in 1979 and 1981, respectively. He joined Fujitsu Laboratories Ltd., Kawasaki, Japan in 1981, where he has been engaged in research and development of switching systems for ISDN and ATM. He has recently been involved in fields such as ubiquitous computing services and the NGN. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan.

Hisayuki Sekine
Fujitsu Ltd.
Mr. Sekine graduated from Meiji University in 1989. He joined Fujitsu Ltd., Japan in 1989 and engaged in switching system software development. He now works in the Business Promotion Division of the Network Solution Business Unit, where he has been engaged in service platform strategy planning since 2007.