Unified Communications Platform

TSUTSUI Kensaku, ARAO Shinya, SERADA Teruharu, HOKARI Makoto

Abstract

Integration of communications services used by an enterprise on an IP network allows the voice service to be embedded easily in other communication tools such as E-mail and IM. Although this is the generally accepted definition of Unified Communications (UC), the definition of UC adopted by NEC is characteristic in that it also combines telephony and other communications services in the business systems of enterprises for their internal use.

This paper describes the technological aspects of the platform that NEC proposes for the implementation of UC.

Keywords

unified communications, SIP, SOC, SOA, presence, one-number

1. Introduction

When an enterprise using a PBX (Private Branch Exchange) introduces an IP telephony system, the telephone network is included in the IP network. The voice communications that have previously been transmitted via private analog or digital circuits are then transmitted as IP packets flowing in the IP network. The voice service may thus be integrated in the network of other communication tools such as E-mail and IM (Instant Messaging) [1].

The integration of communications services used by an enterprise on an IP network allows and facilitates the voice service to be embedded easily in other communication tools such as E-mail and IM. Although this is the general definition of the Unified Communications (UC), the definition of UC by NEC is characteristic in that it also combines telephony and other communications services in the task management systems of enterprises for their internal use (see "UNIVERGE Pursuing "People-Centric" Communications Environments" on pp. 15-18).

We use SIP (Session Initiation Protocol) as the core technology of our UC platform because it is a communication protocol capable of implementing voice services on the IP network at the same time as implementing many communications services such as IM and presence services. In order to embed the communications services in the job systems of enterprises, our UC platform adopts the SOA (Service Oriented Architecture) framework, which makes it possible to embed the communications services provided by NEC in the task management systems of various vendors.

2. SIP and SOA

(1) SIP

SIP is a communication protocol standardized by an organization called IETF (Internet Engineering Task Force). SIP makes it possible to provide a variety of means of communication such as voice, IM and presence services via a single system.

The services implemented by SIP are based on coordinated operations of the SIP terminals and the SIP server.

The use of SIP also enables interconnection with SIP terminals and SIP application servers (such as UMS and call center systems) provided by multiple vendors.

The users can then use the desired terminals and SIP applications by freely combining them.

This paper positions the SIP terminals as the UC terminals.

(2) SOA

SOA is a system architecture that constructs and also arranges software components and functions according to the configuration of business processes and publicizes them on the network so that they may be linked mutually for the construction of flexible enterprise systems and inter-enterprise business process execution systems [2]. Mutual linkage of the software components on the IP network requires a standardized communication interface, and it is usually the web services that assume this role.

The web services are the technologies for calling software components provided by remote servers and exchanging messages with them using standard Internet-related technologies such as HTTP and SOAP. In a narrower meaning of the term, it sometimes refers to the services described in WSDL (Web Services Description Languages) among the

above described services. WSDL describes the API of publicized services in XML. When WSDL is loaded in a software development tool, the module codes for use in calling the web services are generated automatically.

When the communications services provided by the UC platform are publicized using SOA, the business processes consider and handle them as a single software component specialized in communications. As a result, the above makes it easy to combine business processes with the task management systems.

3. UC Platform

In this section, we will discuss the UC platform that uses SIP as the core technology and SOA as the means of linkage with task management systems.

The UC platform is composed of multiple UC-common functions (**Fig. 1**). The following subsections describe these functions individually.

(1) Presence Server

Presence is the means for simultaneous representation of the status of the other party of a communication, the list of terminals of the other party and their attributes (types of communication tools, indications of communication availability, etc.). The main functions of the presence server includes the compilation of information collected from various terminals of individual users into comprehensive presence information and the relay of notifications of the presence status change information between users.

The presence server also references the information provided from sources other than UC terminals, such as the location information and schedule of each user and integrates it into the presence information. For instance, when the presence server references a device capable of detecting the location information of Mr. A and finds that he is in meeting room 401, it automatically sets the status "Mr. A is in conference in meeting room 401" as the presence information of Mr. A.

From the standpoint of the user, the presence information can be referenced in the buddy list, which is a list of persons that a user communicates with frequently. Since it is natural today that a user uses multiple terminals, it is necessary to enable the buddy list to be referenced from multiple terminals. For this purpose, the presence server is sometimes designed to retain the buddy lists of individual users.

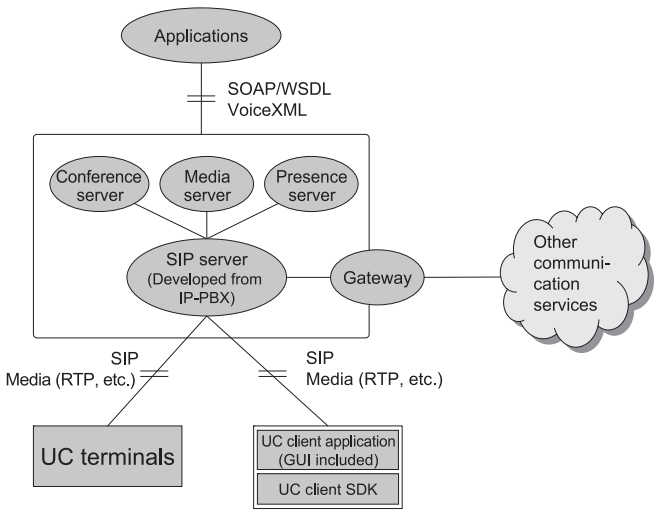

Fig. 1 UC platform.

(2) Media Server

The media server is the function with which the machine accepts phone calls and communication with the user in place of a human operator. The representative functions are the voice mailing and the IVR (Interactive Voice Response) functions.

The media server can be regarded as a user interface for assisting the UC platform to communicate with users by voice. Therefore, in Europe and North America, the voice recognition function is often used together with IVR. An example of the implementation of the media server is a system in which the users can instruct playback and deletion of voice mails by voice.

With the aim of opening the IVR function, W3C (World Wide Web Consortium) has defined an XML-based description language called VoiceXML. Using VoiceXML makes it easy to develop applications that interact with users by controlling IVR.

(3) Conference Server

The conference server provides the function of the conference bridge in conference telephony systems. Conventional conferencing systems have mainly used teleconferencing that uses only the voice medium and the use of video conferencing, data conferencing and multimedia conferencing has recently been advancing.

At present, the main communication protocols used in video conferencing and data conferencing is the H.323/T.120 protocol standardized by ITU-T (International Telecommunication Union Telecommunication Standardization Sector) and the vendor-specific protocols. However, for the

integration of conferencing services, it is ideal to manage the conference sessions in a shared manner using SIP.
The chat server required for IM may also be regarded as one of the functions of the conference server.

(4) Protocol Conversion Gateway

Although it is considered to be ideal if all of the terminals and services can be controlled using one unique standardized communication protocol or API, the reality is not so. For example, IETF had to standardize XMPP (Extensible Messaging and Presence Protocol) as the protocol for implementing the presence and IM functions in addition to SIP. There are also communication terminals and services that do not use IP, such as the analog telephone and ISDN.
A gateway for converting various protocols into SIP is required to handle various terminals and communications services on a single UC platform.

(5) UC Client SDK

The software for IP phone terminals running on PC is called the Softphone. The latest Softphone is gaining the status of an integrated communications client capable of using various communication tools including buddy lists, IM and application sharing in addition to IP telephony. In the following, we will refer to this status of the Softphone as the UC client.
In the world of UC, the UC clients are embedded for use in other applications or web pages. Various communication tools are selected and embedded as required. Also, even when a UC client operates on its own, the interface is customized and provided according to the requirement of each user.
The UC client SDK (Software Development Kit) is provided in order to facilitate the development of UC clients according to various needs. The SDK is composed of the libraries and APIs of the communication tools.
We have discussed customization of the UC clients in the above. On the other hand, there are also cases in which it is necessary to develop a UC client that can provide common services to terminals running on different OSs running different OSs. In order to make this possible, the communication libraries of UC clients are offered for various OSs.

(6) SOA

An API defined using SOAP/WSDL is offered in order to make the communications services available with the UC platform publicize to various applications including business systems. When an application developer loads WSDL in a software development tool, the application being developed will be enabled to easily call communications services on the UC platform.

4. UC terminals

In section 3 above, we have described mainly the UC platform. In this section, we will describe the user interfaces of the UC terminals. Since the UC is provided by combining various services, it is usual that the method of user interface packaging varies between individual UC services.

The user interfaces of UC terminals can roughly be classified into the Stimulus and Functional interfaces (**Fig. 2**).

With the Stimulus user interface the UC terminal behaves like a dumb terminal by following instructions from the application server. The UC terminal therefore does not recognize the application currently sending instructions to it. All it does is to follow instructions from the application servers, display the instruction information for the users and report the event input by the user to the applications.

For example, for the information display function it shows a text message in the LCD panel of an IP phone and for the reporting of user-input events it detects one of the specific patterns that are input from the keypad of an IP phone and reports on it to the application. A representative example of such specific patterns is the PBX service special numbers such as "*8". The special numbers are numbers used for special purposes and are different from ordinary telephone numbers. An example is "*8", which allows the user of the telephone on which it is entered to pick up a call incoming to another telephone in the same group.

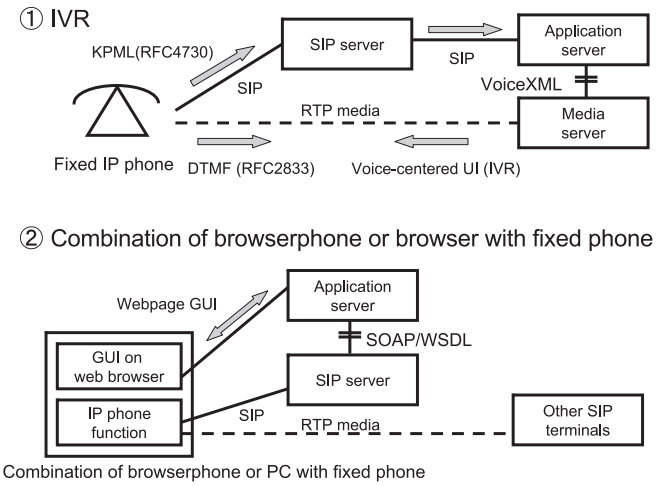

Fig. 2 Stimulus user interface.

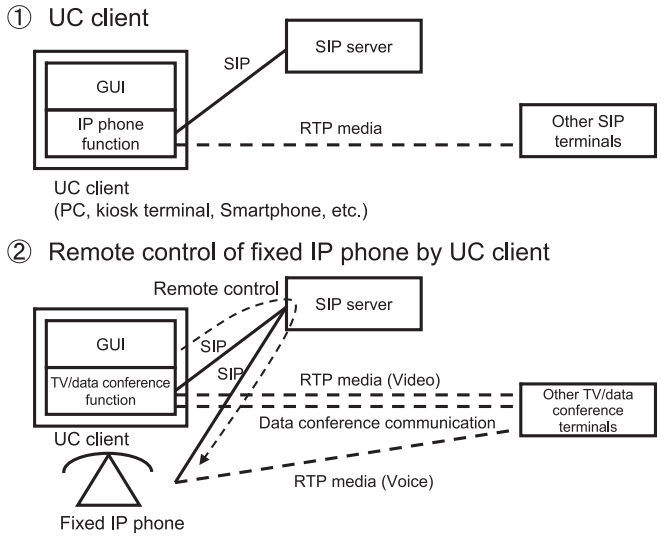

Fig. 3 Functional user interface.

On the other hand, with the Functional User Interface, the UC terminals identify the applications that they are running on themselves, so they can make decisions on the GUI to be presented for the users and the inputs to be obtained from the users (**Fig. 3**).

5. Core Technology of the SIP–UC Platform

In this section, we will discuss further details on the SIP, which is the core technology of the UC platform.

SIP has been standardized with the intention of providing services like UC from the beginning. The key SIP functions in the UC are the presence and one-number functions.

(1) Presence

SIP offers a mechanism for exchanging presence information among specified users. The presence information is expressed in the form of XML data, which expresses the status of a specific user, the list of terminals and the attributes of these terminals (such as the type of communication tool and communication availability).

In **Fig. 4** , when the terminal of Hanako detects a change in the presence information on Hanako, the terminal reports about it to Hanako's PA (Presence Agency) in the presence server. Then, Hanako's PA notifies the users registered in Hanako's watcher list of the change in Hanako's presence information using the NOTIFY message of SIP.

Example; Presence information in Jiro's buddy list

Name	Status	Comment	Available Communication Tools
Taro	Presence in office		Telephone/Mail
Hanako	Meeting room 405	IM whenever possible	Telephone/Mail/IM
Saburo	Dispatched overseas		Mail

When Hanako's status changes from "Present in office" to "Meeting room 405":

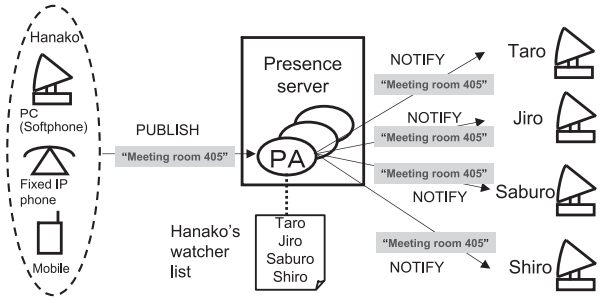

Fig. 4 Presence function of SIP.

(2) One-Number

1) SIP Forking

In the world of SIP, the identifiers of terminals and users are called SIP URI (Uniform Resource Identifiers) and are expressed in a format similar to the E-mail address, for example "sip:alice@nec.co.jp".

Among the SIP URIs, those representing the users themselves are called AORs. In the example above, "alice" in "sip:alice@nec.co.jp" is the name of a person so it is AOR. If "alice" is replaced with a telephone number, the telephone number itself represents the persons and therefore functions as her one-number.

The SIP Forking function of the SIP server distributes the session start signal (INVITE) destined to AOR to multiple terminals under the AOR. It is sometimes distributed simultaneously (ringing multiple terminals simultaneously) or sequentially (ringing multiple terminals one after another) depending on the services.

Although the one-number is convenient, it sometimes results in an impossibility of identifying individual terminals under the AOR by outsiders (see below). This problem can be solved by GRUU (Globally Routable User Agent URIs) and Caller Preference.

2) GRUU

We will describe about GRUU by taking an example. Now, assume that a mobile terminal, fixed IP phone and Softphone are registered under the AOR of Carol.

When Alice places a phone call to Bob ([2]-(a) in **Fig. 5**) and Bob wants to transfer Alice's call to Carol, Bob holds Alice's call and calls Carol, who answers the call on the Softphone ([2]-(b) in Fig. 5). When Bob tells Carol, "I will

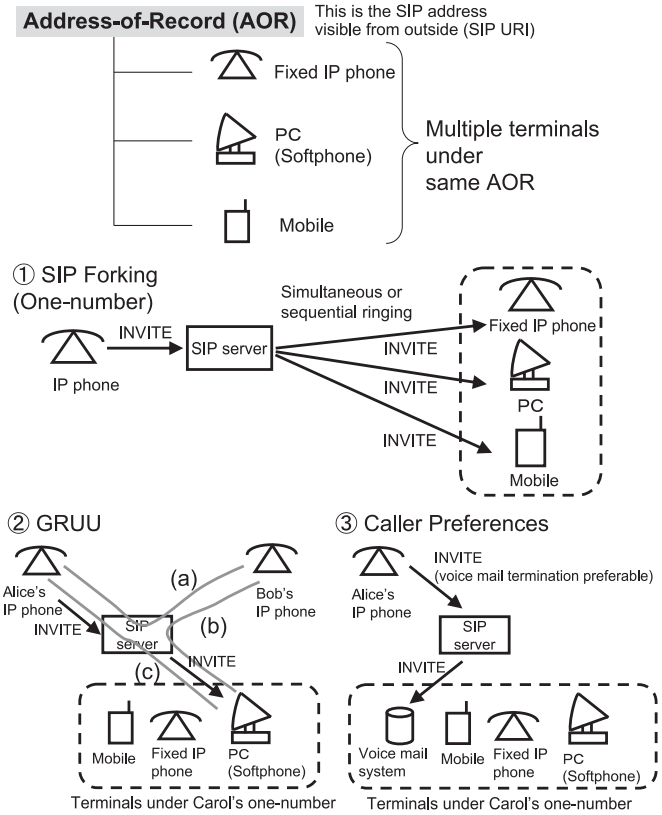

Fig. 5 One-number function of SIP.

transfer Alice's call" and pushes the transfer button, Alice's call is transferred to Carol. At this time, it is required to ensure the connection of Alice to Carol's Softphone ([2]-(c) in Fig. 5).

GRUU is the special SIP URI attribute to each terminal for use in identification of a specific terminal under AOR. With Carol, for example, unique GRUUs are assigned to the mobile terminal, fixed IP phone and the Softphone of Carol.

In the example above, it is necessary to send the INVITE message from Alice to the GRUU of Carol's Softphone to complete transfer.

(3) Caller Preferences

In the case of [3] shown in Fig. 5, Alice thinks, "I want to tell Carol of the idea I hit upon right now, but I should leave a message in Carol's voice mailbox because it is too late in the night." In this case, Alice's terminal can add the Preference to "terminate at the voice mailbox" to the INVITE signal it transmits. Then, the SIP server follows this preference and routes the INVITE message only to the voice mail. Caller Preferences allow users to express their preferences in various ways in addition to this example.

6. NEC's UC Platform

Sphericall is the UC platform of NEC. It provides the UC common functions using the SIP server as the core.

Some customers like highly reliable vertically-integrated products focusing on PBX services and some are more oriented toward the horizontal division combining terminals and applications from multiple vendors freely. There may also be customer enterprises that want to focus on telephone services for the present but also wish to introduce advanced UC functions gradually in order to improve their efficiencies.

Sphericall makes it possible to set the balance between the vertical integration orientation and horizontal division orientation according to the needs of each customer.

One of the features of Sphericall is that it can provide UC services based on appropriate allocation between the Functional UI and the Stimulus UI (**Fig. 6**). The Functional UI is the user interface generated by an application built into the UC terminal, and the Stimulus UI is the user interface generated by an application on the side of Sphericall. The GUI may be subject to change following an addition of functions to the application, but does not need upgrading of the software in the UC terminal.

The Stimulus UI is sometimes more convenient with additional or optional services because it enables an addition of functions by simply upgrading the software in Sphericall or in the application server.

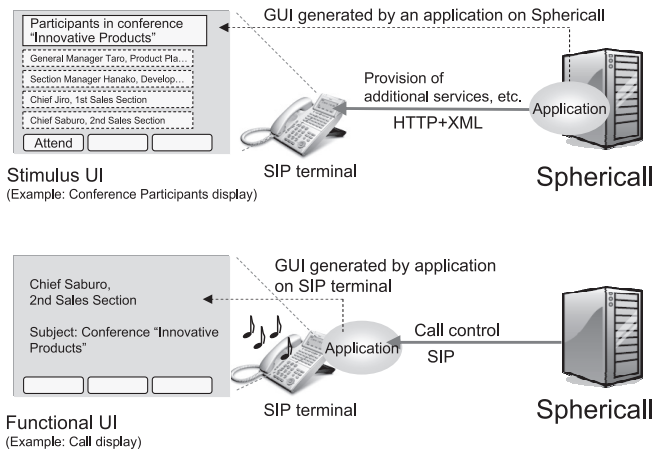

Fig. 6 Stimulus/functional UI in sphericall.

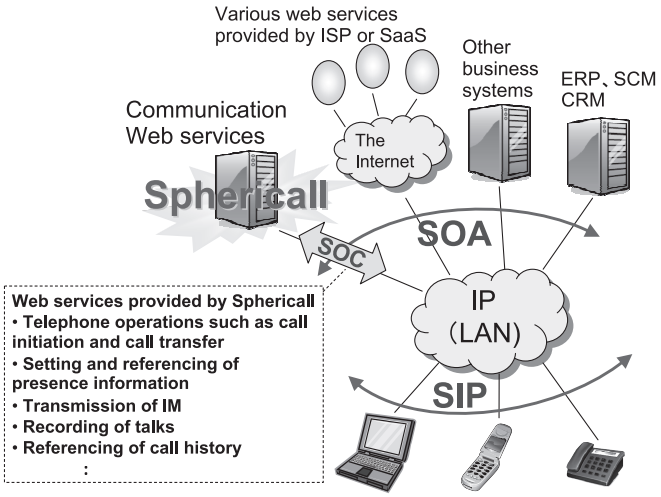

Fig. 7 Enterprise communications system with SOA.

At NEC, we are proposing the concept of SOC (Service Oriented Communications) (see "UNIVERGE Pursuing "People-Centric" Communications Environments" on pp. 15-18).

Publicizing the communications services as web services to the public in the framework of SOA facilitates their embedding in other task management systems. To implement SOC, Sphericall opens the communications services using the SOAP/WSLD-based API (**Fig. 7**).

7. Conclusion

UC keeps on progressing with the aim of integrating a variety of tools and services while linking them to business processes in order to improve their efficiency. At NEC, we plan to make full use of both the IT and network technologies to implement advanced UC that does not become merely a simple extension of PBX but is a platform that is fused with task management systems in order to contribute to the effective advancement of customer enterprise.

References

1) "IT Roadmap 2008", Nomura Research Institute, Toyo Keizai Inc., 2008.
2) http://www.atmarkit.co.jp/aig/04biz/soa.html
3) J. Rosenberg, "A Framework for Application Interaction in the Session Initiation Protocol (SIP)", draft-ietf-sipping-app-interaction-framework-05, IETF, July 2005.
4) E. Burger, M. Dolly, "A Session Initiation Protocol (SIP) Event Package for Key Press Stimulus (KPML)", RFC4730, IETF, November 2006.
5) J. Rosenberg, "Obtaining and Using Globally Routable User Agent (UA) URIs (GRUU) in the Session Initiation Protocol (SIP)", draft-ietf-sip-gruu-15, IETF, October 2007.
6) J. Rosenberg, P. Kyzivat, "Guidelines for Usage of the Session Initiation Protocol (SIP) Caller Preferences Extension", RFC4596, IETF, July 2006.

Authors' Profiles

TSUTSUI Kensaku
General Manager,
UC Software Development Division,
Enterprise Solutions Operations Unit,
NEC Corporation

ARAO Shinya
Manager,
UC Software Development Division,
Enterprise Solutions Operations Unit,
NEC Corporation

SERADA Teruharu
Assistant Manager,
UC Software Development Division,
Enterprise Solutions Operations Unit,
NEC Corporation

HOKARI Makoto
Vice President,
NEC Sphere Communications, Inc.