NTT DOCOMO

Technical Journal

Vol.17 Jul.2015 | No.1

NTT docomo

Efforts to Apply Virtualization Technology to Mobile Network Nodes

Kazuo Sugiyama

Managing Director of Core Network Development Department

NTT DOCOMO R&D is tackling the development of systems that apply virtualization technology to mobile network nodes.

Virtualization enables hardware such as servers to be decoupled from their underlying physical structure and be utilized logically (For example, one physical server can be treated as multiple logical servers). Currently, network functions in a mobile network (for example, the Evolved Packet Core [EPC]) are realized by software operating on hardware dedicated to each node that forms the mobile network. By applying virtualization technology to these network nodes, software for each network node can operate on general-purpose hardware (called a network virtualization platform). Multiple network nodes can share the network virtualization platform.

The benefits of applying virtualization technology to network nodes are as follows:

(1) When traffic temporarily increases due to a disaster or large-scale event, the processing capacity on network nodes become insufficient. As a result, communication becomes congested and it becomes difficult for calls to connect. In such a case, the processing capacity on network nodes can be increased by incorporating the automatic addition of virtual resources on a network virtualization platform. In this way, calls connect more easily when communication is clogged.

(2) Customer service interruption can be prevented by establishing a redundant operational/backup hardware architecture for network nodes. However, even with such a design, once a hardware breakdown occurs, the architecture becomes in essence a single structure until the hardware is repaired and restored. By applying virtualization technology, new virtual resources can be automatically assigned from the network virtualization platform to take the place of the defective hardware. Thus, redundancy can be constantly maintained and the reliability of communication services against hardware failures is improved.

(3) To realize new network nodes/services, necessary software can be developed and deployed at once on the already prepared network virtualization platform. In this way, new functions can be realized without adding specialized hardware, which had been the case until now. New services can thus be provided promptly.

(4) Multiple network nodes can share general-purpose hardware that make up the network virtualization platform, thus improving efficiency of use. In this way, the number of hardware units can be reduced, and network facilities can be built economically.

To maximize these advantages, it is necessary for hardware and software to work together in a mixed environment consisting of equipment from multiple vendors. For this effort to succeed, cooperating with many vendors is necessary.

In October 2014, NTT DOCOMO released out a press announcement entitled "DOCOMO Successfully Trials NFV Using Multi-vendors' Virtualization Systems" [1]. This test demonstrated not only NTT DOCOMO's success in using multiple vendors' equipment to realize virtualization technology, it also proved that it is possible to build a system in which multiple vendors collaborate. Through this trial, we confirmed once again that to skillfully build and operate a multi-vendor environment, it is necessary to standardize a unified inter-system interface. At NTT DOCOMO, we are currently collaborating with major vendors and actively contributing documents toward standardization decisions.

In addition, we are seeking to commercialize virtualized Evolved Packet Core (vEPC) within FY2015. This is our first application of virtualization technology to network nodes. The LTE network is continuing to expand as customers increase. By applying virtualization technology to EPC, which provides core network functions to LTE services and has great processing loads, we can maximize the advantages of virtualization technology. Furthermore, as the next step, we are also researching and expanding the application of virtualization technology to other network nodes that make up current mobile networks.

As described in this article, standardization of the application of virtualization technology to network nodes is being intensively deliberated. We are convinced that virtualization technology will make our core network, which bears the "mission of NTT DOCOMO," even more stable, economical, and attractive. We are also actively incorporating new technologies in the development of future core networks to provide a satisfying communication environment to our customers.

Reference

[1] NTT DOCOMO Press Release: "DOCOMO Successfully Trials NFV Using Multi-vendors' Virtualization Systems," Oct. 2014. https://www.nttdocomo.co.jp/english/info/media_center/pr/2014/1014_00.html

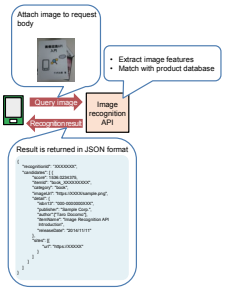
NTT DOCOMO
Technical Journal

Vol.17 No.1 Jul. 2015

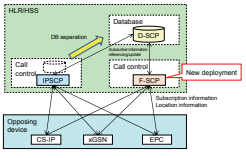
Contents

DOCOMO Today

Technology Reports



(P.10)

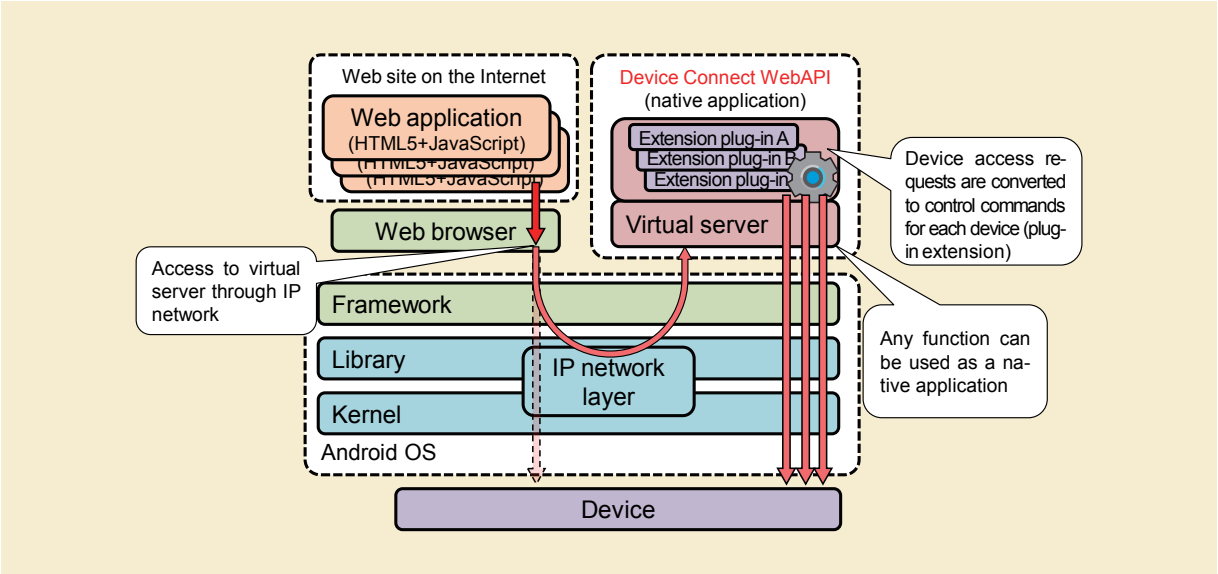


(P.31)

Standardization

News

(P.41)



Technology Reports　Device Connect WebAPI —Web Interface for Variety of Smartphone-linked Devices— (P.4)

Using a device from the DeviceConnect WebAPI (Android OS example)

IoT WebAPI Standardization

Technology Reports

Device Connect WebAPI —Web Interface for Variety of Smartphone-linked Devices—

*A variety of devices able to connect with smartphones have entered the market recently. However, the development environments for each product are different, and developing content for each OS environment and individual device is becoming an issue. With device specifications dependent on communication protocols such as Bluetooth®*1, affinity for Web services is also low. As such, we have developed a Web interface technology that operates on smartphones and has strong generality and extensibility. This article describes development of this technology and efforts toward standardization.*

Service Innovation Department *Takafumi Yamazoe*
Hiroaki Hagino

1. Introduction

A variety of devices, such as wrist bands and cameras, that are able to connect with smartphones and exchange information or control them have entered the market.

However, in order to develop services using these devices, original, manufacturer-specific specifications must be supported, which can create difficulty. Standardization of wearable and healthcare devices is advancing, but it is limited in scope, and usually there is no compatibility between specifications. Because of this, implementations specific to individual devices and standards are needed when developing a service using a variety of devices.

One way to implement content that is not dependent on the environment is the Web application (Web content that operates as an application within a Web browser). As part of the standardization of HTML5*2 at the World Wide Web Consortium (W3C)*3, Application Programming Interfaces (APIs)*4 are being consolidated to use devices from Web applications. However, they assume the commoditization of functions and devices, so while they are generic and general purpose, they do not allow the specialized functions provided by individual devices, which differentiate them from each other, to be used. Even hybrid applications, which enable Web applications to operate like native applications, are limited in that they are dependent on the state of device support in the existing framework. Even with hybrid applications, if the user has multiple applications that use varions devises installed on the same terminal, each application will have to have dedicated functions to access the devices. As with native applications, this issue has still not been resolved.

As such, NTT DOCOMO has developed the Device Connect WebAPI, combining standard Web technologies and

©2015 NTT DOCOMO, INC.
Copies of articles may be reproduced only for personal, noncommercial use, provided that the name NTT DOCOMO Technical Journal, the name(s) of the author(s), the title and date of the article appear in the copies.

*1 **Bluetooth®:** A short-range wireless communication specification for wireless connection of mobile terminals, notebook computers, PDAs and other portable terminals. Bluetooth is a registered trademark of Bluetooth SIG Inc. in the United States.
*2 **HTML5:** An enhanced version of HTML formulated by WHATWG and W3C (see*3).

general smartphone functions to create a mechanism that enables any content or service, whether native or Web application, to use any device by accessing the WebAPI. This article describes the features of the Device Connect WebAPI and related initiatives.

2. Features and Mechanism

The main features of the Device Connect WebAPI are a common method for device access, device abstraction through functions, and strong generality and extensibility.

2.1 Common Method for Device Access

Ordinary native applications (applications downloaded from an application store for the OS and running on a smartphone) access devices using methods provided by the OS, as shown in **Figure 1**. This enables any function to be implemented, but also requires applications to be implemented for each OS, and each device. Also, accessing devices from a Web application depends on functions provided by the Web browser, as shown in **Figure 2**. Currently, even though W3C has standardized APIs for accessing basic devices, only some of the functions have been implemented in Web browsers on smartphones. In contrast to this, the Device Connect WebAPI runs as a virtual server on the smartphone and provides a WebAPI to access devices freely, even from a Web browser, so that devices can be used from Web applications. Specifically, access to devices is provided by the following procedure, as shown in **Figure 3**.

(1) The Web application sends a request for device access to the WebAPI on the virtual server through the internal IP network on the smartphone.

(2) The virtual server receives the device access request, and converts it to a control command for the OS and individual device.

Generally, WebAPIs are used through an IP network, so the Device Connect WebAPI can be accessed from either native or Web applications. Also, it is not dependent on the OS development

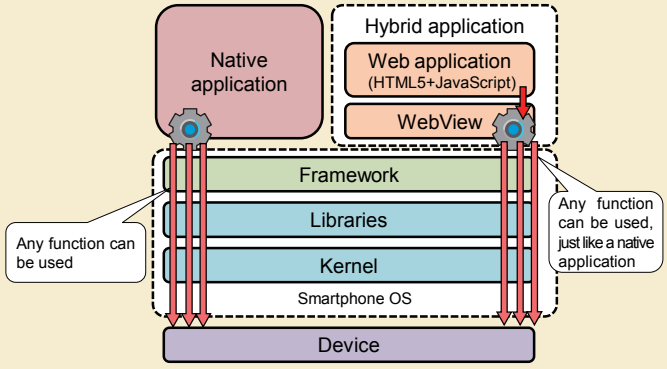


Figure 1 Using a device from a native or hybrid application

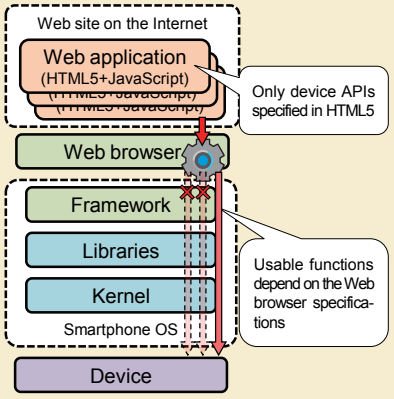


Figure 2 Using a device from a general Web application (Web browser)

*3 **W3C:** An international organization that promotes the standardization of technologies used on the WWW.

*4 **API:** General-purpose interfaces for using functions and data.

environment, run-time environment or device being connected to, and general-purpose Web or native application implementations can be used.

2.2 Device Abstraction by Function

The Device Connect WebAPI abstracts individual devices, with original functions specified by each manufacturer, using the set of functions provided by each device. For example, devices with a function to turn on a light, whether a light switch, a camera, or a toy, all perform a common operation with a common code description such as "turn on the light." **Figure 4** shows an example of controlling devices with different purposes in similar ways from a Web browser.

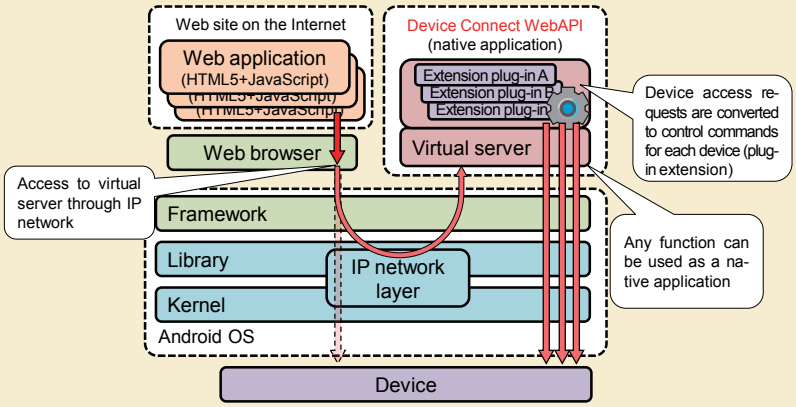


Figure 3 Using a device from the Device Connect WebAPI (Android OS example)

2.3 Strong Generality and Extensibility

The Device Connect WebAPI is composed of the core, which operates as a virtual server, and plug-ins, which connect to and control devices, as shown in **Figure 5**.

As described in section 2.1, the core

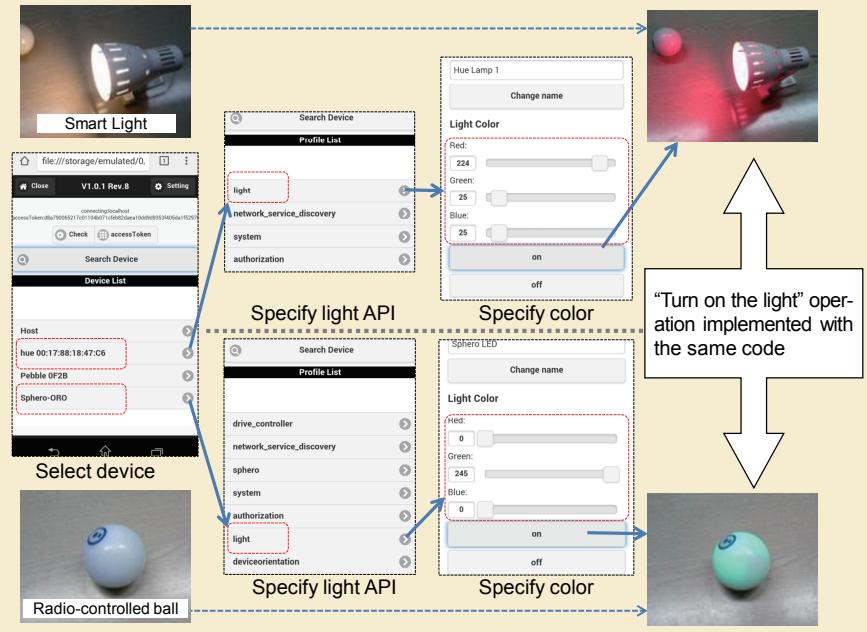


Figure 4 Example of operation from a Web browser

relays API requests received through the IP network to the plug-ins. As such, the core does not have any real functionality besides system administration, and does not have individual API specifications.

Conversely, the plug-ins have the actual functions for accessing the device, and provide API specifications for using the various functions of the device.

Since the core does not implement any functionality or specify APIs, any new device can be supported by adding a new plug-in. For the convenience of developers, general and generic functions are pre-defined in the API specifications, but any additional APIs can be defined in the plug-in, so that special and unique functionality in each device can be used. This enables the Device Connect WebAPI to achieve strong generality and extensibility.

3. Security Measures

The Device Connect WebAPI is designed to extend functionality, allowing access to various device functions from the smartphone OS or Web browser, so ensuring security is an issue (**Figure 6**).

As such, it implements various measures to protect against malicious applications, interception, tampering or impersonation (spoofing) of the virtual server.

3.1 User Consent for Use of Functions by Services

When a user installs a native application on a smartphone OS, the user is presented with a screen to authorize the application to use certain functions, to prevent the application from behaving in ways unintended by the user. However, applications using the Device Connect WebAPI are installed separately from the Device Connect WebAPI, and the functions are used through the IP network, so the confirmation screen cannot be displayed for the Device Connect WebAPI, which has already received permission to use the functions.

Thus, to prevent unintended access to functions, the Device Connect WebAPI virtual server uses OAuth authentication*5, which is a security model used widely on the Internet. When a service first accesses the virtual server, it checks a list of functions that it will use with the user, preventing access to functions not intended by the user, as with conventional applications.

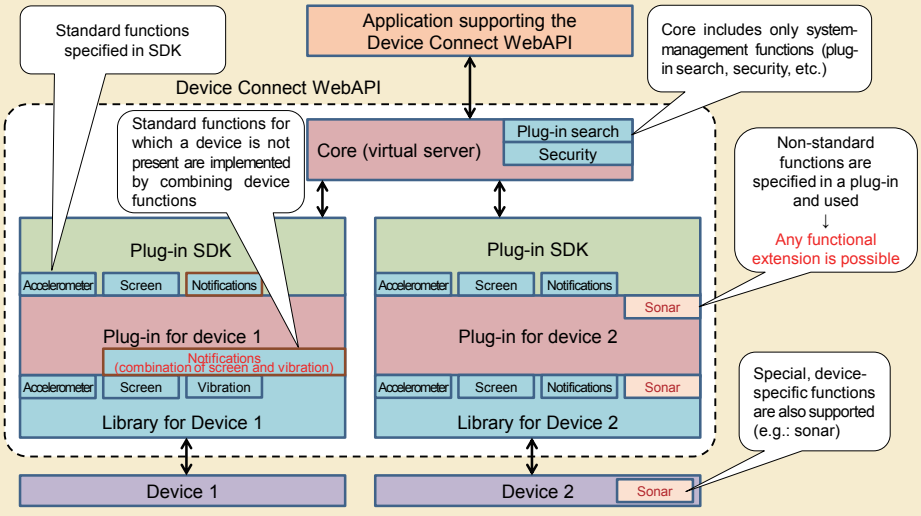


Figure 5 Device Connect WebAPI architecture

*5 **OAuth authentication:** A standard method for certifying secure APIs used in Web services and other applications.

3.2 Interception or Falsification of Communication

Since the application and virtual server communicate through the IP network, there is a risk of interception or falsification. Ideally, communication should be encrypted using Secure Sockets Layer (SSL)[*6] or other means, but the virtual server is installed and used on the smartphone, so encryption key information cannot be protected from reverse engineering[*7], and dynamically generated keys cannot be used from the Web browser. Thus, for our smartphone security model, we verified the secrecy of HTTP[*8] communication, and confirmed that various information regarding HTTP communication within the terminal cannot be obtained without having root permissions.

Communication outside the terminal, which requires a separate mechanism for secrecy, is designed to be handled by plug-ins, so it is outside of the scope of the Device Connect WebAPI. Plug-ins operate as ordinary native applications, so they are able to use dynamically generated key information, unlike the Web browser, and they can be used for encrypting communication outside of the terminal. For information passed between the virtual Web server core and plug-ins, security can be preserved by various security models provided by the smartphone OS.

3.3 Impersonation of Virtual Servers

Virtual servers are implemented as ordinary native applications, so the possibility of an application terminating the virtual server and then impersonating the virtual server must be handled.

Thus, in addition to monitoring the state of the virtual server, the virtual server is explicitly launched from the Web layer, and during processing, an Intent[*9] URL scheme is used as a mechanism to protect against impersonation. This is the mechanism that enables native applications to be launched from the standard

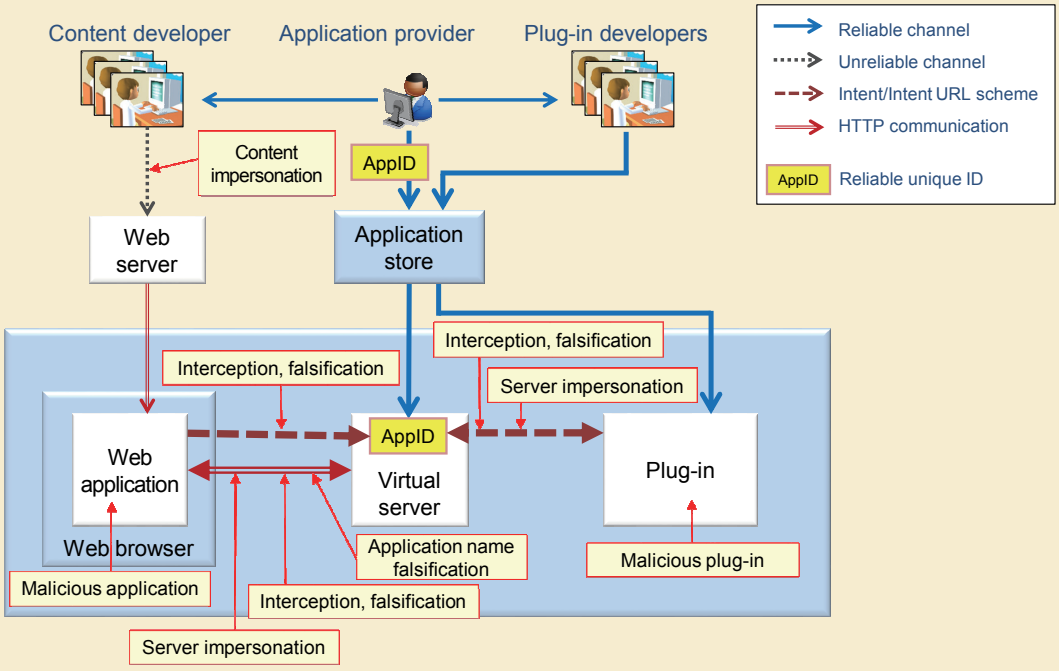


Figure 6 Security measure perspective

*6 **SSL:** A protocol for encrypted communications.

*7 **Reverse engineering:** A process of analyzing the configuration and operation of software or hardware to clarity manufacturing methods and operating principles.

*8 **HTTP:** A communications protocol used to send and receive HTML and other content between Web browsers and Web servers.

*9 **Intent:** A mechanism provided by the Android OS for programs to exchange parameters. Used between components within an application and between applications to exchange information.

Android™*10 Web browser, by specifying a package name*11, allowing information to be passed explicitly from the Web layer to the native application layer. However, the opposite, passing information from the native application layer to the Web layer, is not possible. Information verifying that impersonation is not occurring is sent as a parameter to the native application layer the first time only, and for all subsequent interaction, information proving that it is not fake is exchanged in the Web layer only in order to detect any impersonation.

3.4 Malicious Plug-ins

Plug-ins contain the implementations of functions, and can be considered the same as ordinary native applications. Thus, using the smartphone security model, it is possible to check whether a malicious application is using functions not intended when the plug-in is installed or on the settings screen.

4. Initiatives for Development of the Device Connect WebAPI

The source code for the overall architecture of Device Connect WebAPI, including a virtual server implementation, has already been published on GitHub*12 as open source software [1]. A Web interface specification as a REpresentational State Transfer (REST) API*13, as well as Software Development Kits (SDK)*14 for Android, iOS*15, and JavaScript*16 environments to make content development easier, and SDKs for Android and iOS plug-in development are provided.

As described in section 2, many features of the Device Connect WebAPI, such as generality through a Web interface and extensibility through plug-ins, are based on the architecture. Standardization of device access has been done with various goals in the past, but they have focused on building a closed model with vertically-integrated protocol stack, or conversely, specified only a limited part of the communication protocol. Very little effort has emphasized generality and extensibility of the architecture. For example, in setting near-field communication protocols, several organizations have their own specifications, as a means of creating exclusive lock-in, but they effectively lose any compatibility between devices, and reduce convenience for users.

On the other hand, by not specifying the protocol stack there is no lock-in with the Device Connect WebAPI, and by using a Web interface that abstracts functionality, the architecture emphasizes incorporation of other specifications. Thus, even with a flood of different specifications, it can be used as an interface for integrating all of them. NTT DOCOMO has already contributed the Device Connect WebAPI specification to the Open Mobile Alliance (OMA)*17 Generic open terminal API (GotAPI)*18 specification, and it will be released as a standard in March 2015.

5. Conclusion

We have developed the Device Connect WebAPI as an architecture enabling use of all functions of a variety of devices linked to smartphones from content. The technology developed has been released as open source software on GitHub using the MIT license*19, and is being standardized at OMA as GotAPI. Publishing details of the technology contributes to expanding its use and improving security, and will promote use in a variety of content and support for more devices.

Reference

[1] GitHub: "Device Connect." https://github.com/DeviceConnect

*10 **Android™:** A software platform for smartphones and tablets consisting of an operating system, middleware and major applications. A trademark or registered trademark of Google Inc., in the United States.

*11 **Package name:** Information identifying an application. Uniqueness is guaranteed by smartphone application stores.

*12 **GitHub:** A sharing Web service for software development projects.

*13 **REST API:** A style of software architecture used on the Web.

*14 **SDK:** A tool or set of tools used for software development.

*15 **iOS:** A trademark or registered trademark of Cisco Corp. in the U.S.A. and other countries.

*16 **JavaScript:** A script language appropriate for use in Web browsers. JavaScript is a registered trademark or trademark of Oracle Corporation, its subsidiaries and affiliates in the United States and other countries.

*17 **OMA:** An organization that promotes standardization of mobile-related applications.

*18 **GotAPI:** A general-purpose Web interface specified by the OMA and based on a study of the DeviceConnect WebAPI implementation.

*19 **MIT License:** A software license providing no guarantee, but permitting unlimited use, free of charge, by just adding the license descriptor.

Image recognition | Specific object recognition | Open API

Technology Reports

High-speed, Large-scale Image Recognition and API

We have developed an image recognition system that makes it possible to recognize items (objects) in a photograph by instantly searching for similar images in a large-scale database containing over five million images. By implementing this real-time image recognition algorithm, we have realized a new human interface which takes an image as an input to the system instead of text or voice data. In this article, we describe our high-speed image recognition algorithm developed by NTT DOCOMO, as well as an image recognition API provided by NTT DOCOMO.

Service Innovation Department
Hayato Akatsuka
Teppei Inomata
Toshiki Sakai

1. Introduction

Image recognition refers to technologies to identify objects within images.

In the history of image recognition, character recognition was developed first, and has been useful in improving work efficiencies in commercial and industrial fields. In Japan, with the introduction of the postal code system in 1968, Toshiba Corporation implemented automatic mail sorting equipment [1] that incorporates the first ever hand-written character recognition. The equipment mechanized the sorting of mail by postal code, which had been done by hand before this invention. More recently, as computing power has increased, development of image recognition algorithms has become more active because the image recognition requires processing power. Reduction in size and price of cameras has enabled ordinary consumers to experience the benefits of image recognition in their daily lives. For example, Toyota Motors Corporation is providing the Night-View system [2], as part of initiatives to achieve a traffic-accident-free society, using image processing. The Night-View system detects pedestrians and notifies the driver in real time in order to improve safety when driving at night. In the gaming industry, Microsoft developed the Kinect™*1 [3] for Xbox360®*2 in 2010, which enabled natural game play through gestures, without using a physical controller. In e-commerce, Amazon.com introduced real-time product identification from images using object recognition, and developed Amazon Firefly*3 [4] in 2014, which direct users to its online shopping site through object recognition. These are just a few examples, but they show how image processing has permeated our daily lives in the half-century since its introduction. In the future, as smartphones and wearable devices become more common, we expect the need for instant recognition of all kinds of objects in photographs will increase still further.

To this end, NTT DOCOMO is working to develop and improve our own image recognition technologies. We have

©2015 NTT DOCOMO, INC.
Copies of articles may be reproduced only for personal, noncommercial use, provided that the name NTT DOCOMO Technical Journal, the name(s) of the author(s), the title and date of the article appear in the copies.

*1 **Kinect™:** A registered trademark or trademark of Microsoft Corp. in the United States and other countries.
*2 **Xbox360®:** A registered trademark of Microsoft Corp. and related companies.

already provided our Utsushite Honyaku[*4] service, which performs character recognition to provide foreign language translations of Japanese by simply holding the camera over the text. Currently, we are also developing an image recognition engine that can recognize more complex objects than text. For recognition of complex objects, images of the objects to be recognized had to be registered in the database beforehand, and the main task of the image recognition is to identify items (or objects) by comparing input images with images stored in the large-scale database in real time. One big challenge for image recognition is handling the large-scale database in real time. As the number of items registered in the database increases, the number of items which share similar image characteristics increases as well, and this causes a drop in recognition accuracy. Also, as the number of images registered in the database increases, it takes more time to look up items in the database, and this causes a drop in processing speed. NTT DOCOMO solved these issues that commonly exist in image recognition by improving our algorithms, and realized highly accurate image recognition from a large-scale database of several million images in less than one second. Our image recognition algorithm is based on specific object recognition.

This article describes the algorithms used in image recognition technology developed by NTT DOCOMO that recognizes items (objects) in photographs. These algorithms result in the accuracy and processing speed of the image recognition. This article also gives an overview of the image recognition Application Programming Interface (API), which NTT DOCOMO began offering in October 2010, through docomo Developer support [5], in order to create open innovation and to support developers.

2. Image Recognition Details

2.1 Image Recognition Algorithms

The image recognition algorithm used in the image recognition engine developed by NTT DOCOMO (hereinafter referred to as "the algorithm") mainly focus on objects which have distinctive patterns on their planar surfaces. It identifies what the items in the photograph are (e.g., if it is a book, then the book itself can be recognized, so that specific information about the book can be obtained). The image recognition process is divided roughly into the following three phases (**Figure 1**).

(1) Keypoint detection

Points that indicate characteristics of the object (keypoints) are extracted from the image entered by the user (the query image) in real time. The keypoints in images of objects stored in the database are similarly extracted beforehand. These images stored in the database hereinafter will be called "reference images."

(2) Image feature[*5] description

For each keypoint extracted from the query and reference images in (1), a vector describing the characteristics of the keypoint ("image features") is computed from information such as the distribution of brightness at and around the point. This process is done in real time for the query image, and beforehand, off-line for the reference images.

(3) Image feature comparison

The image features for the query and reference images are compared, and the reference image which has image features that are the most similar to those of the query image is selected.

Each of these phases is described below in more detail.

1) Keypoint Detection

To identify an object in a photograph by image recognition, image characteristics of the object must be extracted from the image data. With specific object recognition, a set of keypoints extracted from the image characterizes the object.

It is desirable that the same keypoints can be extracted invariantly from the image, regardless of various photographic conditions and shooting methods. Typically, scale-invariant keypoints appear at corners or at the intersections of lines. We have combined several corner detection methods to implement more reliable keypoint detection.

Keypoints are extracted from the

*3 **Amazon Firefly:** A trademark of Amazon.com in the United States and other countries.

*4 **Utsushite Honyaku:** The name of character recognition service provided by NTT DOCOMO.

*5 **Feature:** A feature consists of numerical values. Sets of features capture unique image characteristics of an object that can represent the object. In particular, our feature is computed based on the brightness distribution surrounding detected keypoints.

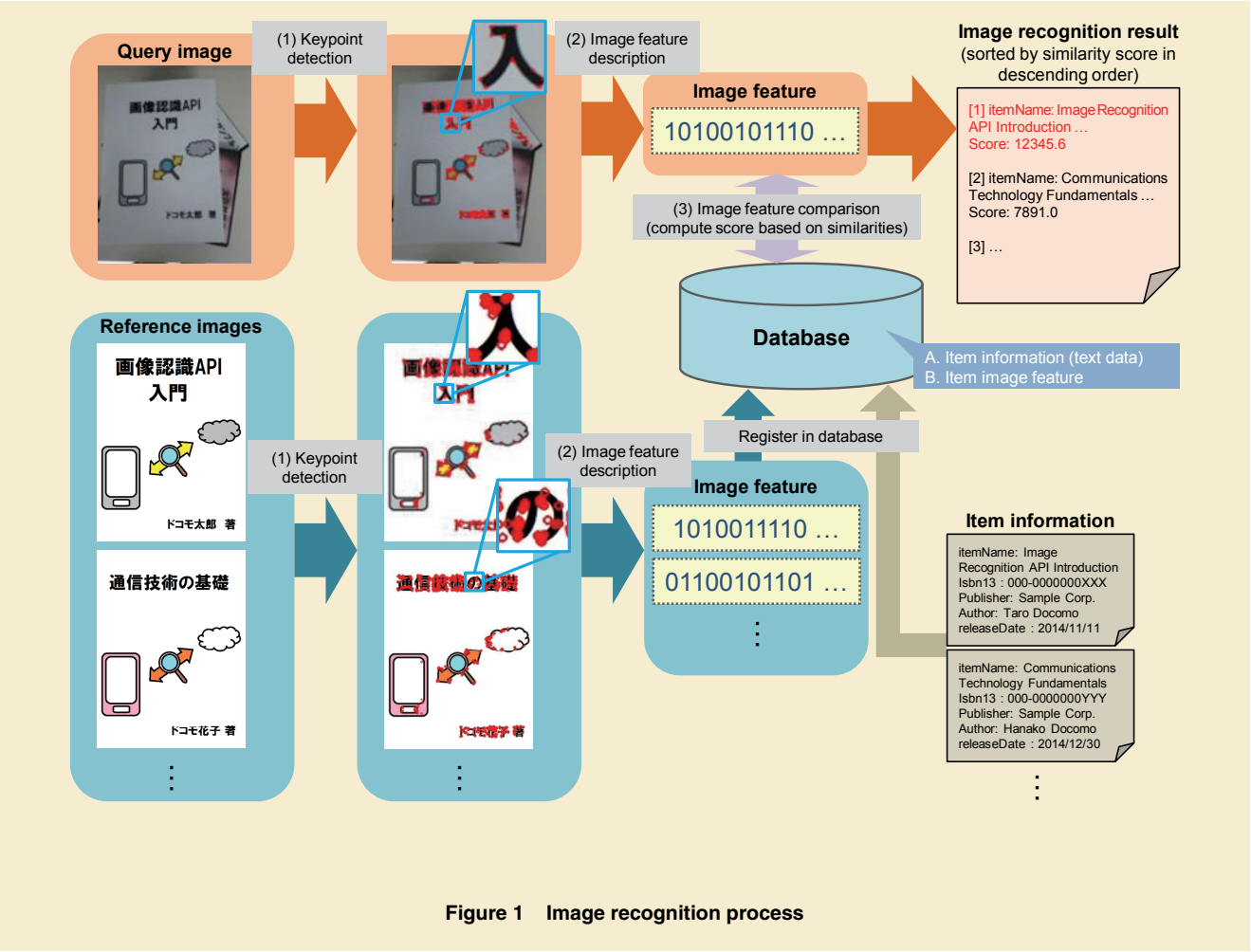


Figure 1 Image recognition process

query image in real time and from the reference images off-line beforehand. It is desirable that the same keypoints are detected in the query image and the reference images, but at certain degree of discrepancy can be expected due to differences in photographic conditions and shooting methods, even though the keypoints detection is robust.

2) Image Feature Description

An image feature is computed for each keypoint detected in the previous phase. If the objects shown in the query image and a reference image are the same, we expect many of the keypoints in the query image to correspond to keypoints in the reference image. We compute a unique image feature for each keypoint in order to compute the similarity between the query image and reference images.

The image feature used in this algorithm is typically called a "local feature" in specific object recognition. The local feature is a vector that describes the distribution of brightness in the area at and around the keypoint. Our algorithm defines the image feature such that for a given keypoint, the feature is invariant, regardless of changes in scale or rotation between photographs. Comparing image features makes it possible to find similar images.

There are several standard ways of describing local features in computer vision. One is called Scale-Invariant Feature Transform (SIFT) [6], and another is called Speeded Up Robust Features (SURF) [7]. Both algorithms produce local features that are scale and orientation invariant. For our algorithm, we have kept scale and roll invariance

as with SIFT and SURF, but coded the local feature using binary*6 vectors to speed up the image feature comparison, as described in 3).

3) Image Feature Comparison

In this phase, the set of local features from the query image are compared with those of reference images in order to retrieve highly similar objects from the database. Our algorithm is advanced compared to others in the computer vision industry today, because it can perform local feature comparisons much faster.

The database contains both product information and local features for each object. The product information includes title, author, and publication date, for example. The local features are calculated for each reference image as described in 2). These local features are used for recognition because binary comparison based on local features is much faster than brute-force comparison based on raw image data.

The local features from the query image are compared with all of the local features stored in the database in order to extract matching keypoint pairs. If the same object appears in both the query image and a reference image, many keypoint pairs will be found. After extracting matching keypoints, posture estimation is performed to eliminate a set of keypoint pairs which do not follow the majority of them. A similarity score is then computed based on the number of remaining keypoint pairs and the image feature similarity of the pairs.

With millions of reference images, computing the similarities by brute force would take more than one minute to perform just one image recognition or process just one query image.

To solve this problem, we developed a faster comparison technique using Locality Sensitive Hashing*7 (LSH). LSH summarizes local features in a hash space with fewer dimensions, so that similar data can be searched efficiently. LSH is a probabilistic search technique, so it is not theoretically guaranteed to find the optimal solution, but in most real cases, it does find an appropriate solution. It is also able to complete a comparison with several million reference images in less than one second.

2.2 Recognition Performance

In order to check the performance of our image recognition algorithm, we conducted evaluation tests. Here, we describe the details of these tests and summarize the results in terms of both recognition accuracy and processing speed.

1) Recognition Accuracy

We conducted experiments to evaluate recognition accuracy using eight different types of query images and approximately one million reference images. Types of query image include (1) *FRONT* (clear photographs of the object taken from the front, filling up the photograph frame), (2) *BLURRED* (blurred photographs), (3) *NOISY* (photographs with random noise), (4) *ROLLED* (photographs of the object taken from the front, but images are rotated along line-of-sight axis), (5) *ENLARGED* (photographs of the object taken from the front, but enlarged so that only part of object is shown in the frame), (6) *REDUCED* (photographs of the object taken from the front, but zoomed out), (7) *PANNED* (photographs of the object taken from the left), (8) *TILTED* (photographs of the object taken from below) (**Figure 2**).

After image recognition, we take the top-three items from each image recognition result, with items sorted by similarity score. If there are any correct items within in the top-three items, we increment the number of correct image recognitions by one. Then, we divide the number of correct image recognitions by the number of image recognitions attempted to calculate the recognition accuracy. Note that the number of image recognitions attempted is same as the number of query images, and the number of correct image recognitions cannot exceed the number of query images (**Figure 3**). In our evaluation results, we achieved accuracy of over 90%, and we found that some types of query image do not degrade image recognition accuracy. These included *FRONT*, *BLURRED* and *NOISY* images. Conversely, images whose appearance is different from the corresponding reference image showed degraded image recognition accuracy. These included *PANNED* and *TILTED* images. For both *ENLARGED*

*6 **Binary:** A format for expressing numerical values in base two using strings of 0s and 1s.

*7 **Hash:** A technique to map data of arbitrary size to data of fixed size. In this article, it is used to speed up the data comparisons.

and *REDUCED* images, the recognition accuracy decreases because fewer key-points are extracted from such query images. For *ENLARGED* images, even though they share similar image-feature properties with their reference images, parts of the object are not captured within the photograph. For *REDUCED* images, details in image data are lost when they are reduced and compressed (Fig. 2).

So far, we have identified two factors that contribute to decreasing recognition

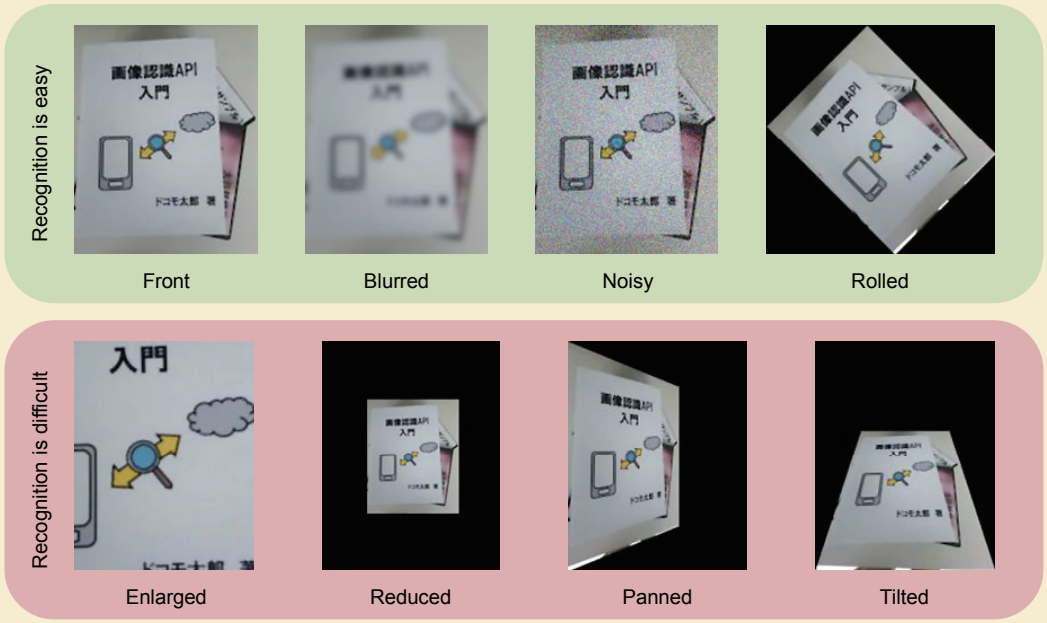


Figure 2 Eight types of query image for evaluation

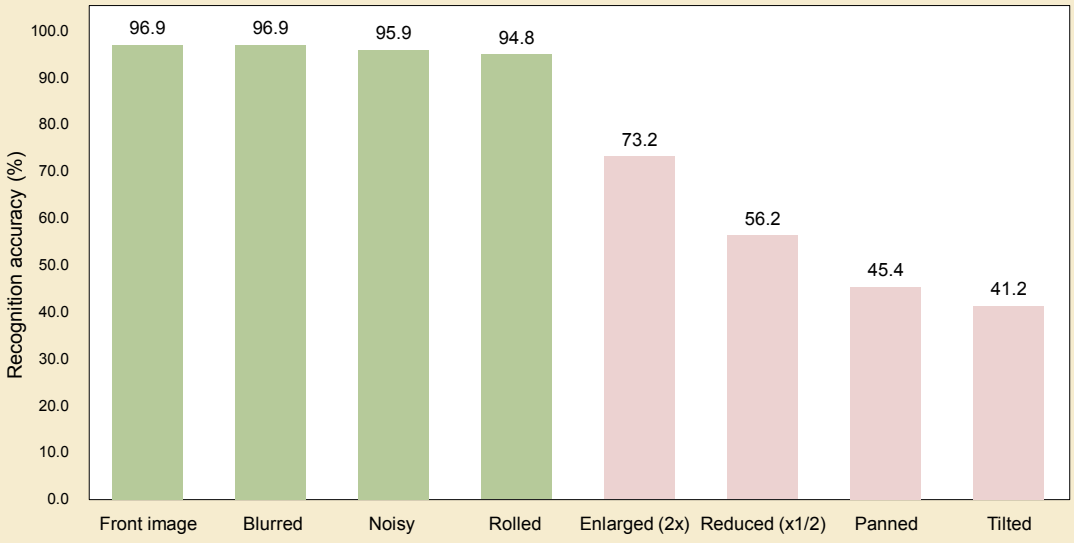


Figure 3 Recognition accuracy

accuracy in evaluation tests. The first is changes in the appearance of the object in the query images due to different scale or orientation. When the appearance of the object differs between input image and reference image, keypoints extracted from the two sources also vary. This causes fewer keypoints to be matched between the query and reference images and introduces a drop in image recognition accuracy. The second is difficulty in selecting the correct reference image when there are many similar reference images. For example, there may be a series of books whose appearance differs only in title and otherwise look the same. In such a case, since books in the series share a common appearance, image recognition accuracy can drop. NTT DOCOMO is currently working on solving these two issues in order to further improve recognition accuracy.

2) Processing Speed

For brute-force matching methods, the number of comparisons between each query image and the reference images increases in proportion to the number of reference images. However, since we have applied the high-speed matching technique using LSH to compare the query images and reference images, image features are compressed into a lower-dimensional vector space and this makes the increase in the number of comparisons much less than with brute-force matching. In the results of our evaluation tests, recognizing one query image using 100,000 reference images required an average of 0.24 seconds of processing time, while using one million reference images required an average of 0.64 seconds. Increasing the number of reference images by a factor of ten resulted in an increase in computing time by a factor of 2.7. Devices such as glasses-type wearable devices are currently spreading quickly, so we are working to further increase processing speed in order to realize more seamless recognition in real time.

3. Image Recognition API Service Details

The image recognition technology described above is provided on docomo Developer support [8] to application and service developers as "Image Recognition API". This image recognition API is a RESTful (REpresentational State Transfer) API*8 and is available to registered members of docomo Developer support.

3.1 Image Recognition API Features

"Image Recognition API" allows computers to perform image based recognition of the packaging of products sold in Japan. Books, DVDs, CDs, PC software, game software, and foods are supported. The image recognition API has a database of image and product information for more than five million products on the market. It compares the features of an input image with those of reference images in the database using the algorithm described above, and returns the product information associated with reference images similar to the query image.

Most other image recognition services, such as GAZIRU®*9 [9] provided by NEC Corporation, and the object recognition software [10] from PUX Corporation, only provide a recognition engine, and users of the image recognition service must gather the database, including images and the names of the objects in those images (information regarding what is in the images), and store it in the database on their own. Our image recognition API provides both the image recognition engine and a database of over five million items that NTT DOCOMO has gathered. Since developers can use the API with less labor for gathering data, they can develop and build mashup*10 applications and services using image recognition easily. Our API design concept is that it can be used by simply inputting an image, as described below, and developers need not be concerned with the internal image recognition processes when using it. This makes it easy to develop image recognition applications and services without any knowledge of the image recognition mechanisms.

3.2 Usage

Figure 4 shows the typical steps when using the image recognition API, from inputting an image to the image recognition API to receiving the recog-

*8 **RESTful API:** An API conforming to REST. REST is a style of software architecture developed based on design principles proposed by Roy Fielding in 2000.

*9 **GAZIRU®:** A trademark of NEC Corp.

*10 **Mashup:** To create and provide a service by combining the content and services from several other, different services.

nition result. Users of the image recognition API input a query image using the Hyper Text Transfer Protocol (HTTP) POST method*11, attaching it to the request body*12, and receive the recognition result in reply.

The result is returned as JavaScript Object Notation (JSON)*13 format text data, including the name of the product in the query image, a certainty score (similarity between query and reference image), and product details. Product details can include, for example, the publisher, publication date and author for a book, or links to e-commerce sites where the product is sold.

The API also provides end-points for feedback, so users can provide feedback on the suitability of recognition results. Feedback is used to improve recognition accuracy and to update the database.

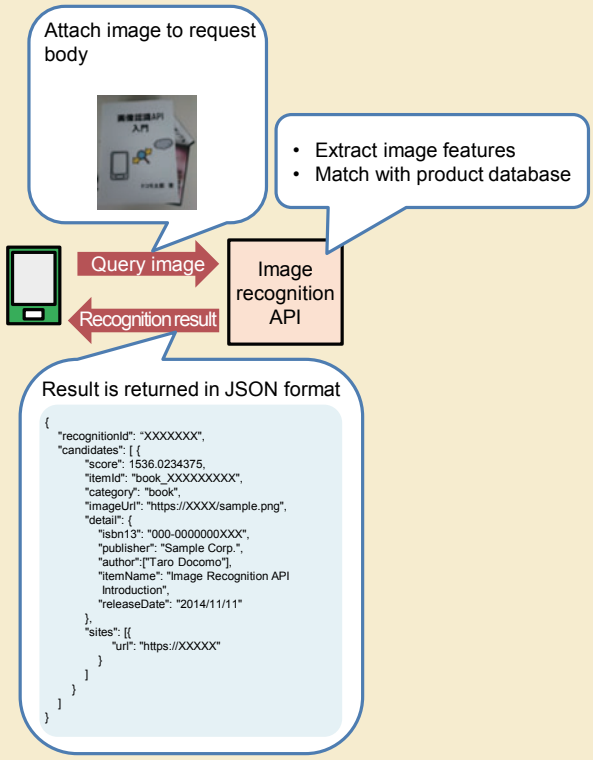


Figure 4 Image recognition process in a service (or application) with the image recognition API

3.3 Service Examples

Users of the image recognition API can develop new image recognition services by combining their own ideas with the information returned by the image recognition API. Possible examples include applications that provide product reviews and display prices retrieved from the Internet based on the product name and e-commerce site links returned from our API, or that allow users to take a picture of a product and then immediately purchase it on an e-commerce site, like Amazon Firefly [4]. Various other applications are possible, such as image based inventory management.

The image recognition API is also very compatible with Augmented Reality (AR), making it possible to recognize products and display information overlaid on the image or video based on the result of image recognition. In particular, eyeglass-type wearable devices such as Google Glass are very compatible. On these devices, image recognition could be performed on images captured with the attached camera and the information displayed on the screen of the glasses, enabling the user to get information seamlessly. After the release of the API service, developers on docomo Developer support became more active developing image recognition applications using AR technology and wearable devices.

4. Conclusion

In this article, we have described our image recognition algorithm, and our image recognition API service.

The accuracy of image recognition depends on how an object is photographed, and the speed of image recognition depends on how many objects are registered in the database. Through experiments, we have shown that high image

*11 **POST method:** A method for sending data from a client to a server when using HTTP communication.
*12 **Request body:** The part of the POST method containing the data sent from the client.
*13 **JSON:** A data description language based on JavaScript object notation.

recognition accuracy can be achieved when objects are photographed from the front, and high-speed image recognition can be achieved even on a scale of one million reference images stored in the database.

The image recognition algorithm currently developed by NTT DOCOMO mainly recognizes planar objects, but we are continuing to work on implementing high-speed, large-scale image recognition for 3D objects (such as landmarks, celebrity, clothes, and food) as well.

References

[1] Toshiba: "Toshiba Science Museum: World's First Automatic Mail Processing Equipment." http://toshiba-mirai-kagakukan.jp/learn/history/ichigoki/1967postmatter/index_j.htm

[2] Toyota Motor Corp.: "Toyota | Safety Technology | Night View." http://www.toyota.co.jp/jpn/tech/safety/technology/technology_file/active/night_view.html

[3] Microsoft Research: "Human Pose Estimation for Kinect – Microsoft Research." http://research.microsoft.com/en-us/projects/vrkinect/default.aspx

[4] Amazon.com: "Understanding Firefly - Amazon Apps & Services Developer Portal." https://developer.amazon.com/public/solutions/devices/fire-phone/docs/understanding-firefly

[5] NTT DOCOMO: "Support for Developers with docomo Developer support," NTT DOCOMO Technical Journal, Vol. 16, No. 2, pp. 48-50, Oct. 2014.

[6] D. G. Lowe: "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, Vol. 60, No. 2, pp. 91-110, 2004.

[7] H. Bay, T. Tuytelaars and L. V. Gool: "SURF: Speeded Up Robust Features," 9th European Conference on Computer Vision, 2006.

[8] NTT DOCOMO: "Image Recognition | docomo Developer support | NTT DOCOMO." https://dev.smt.docomo.ne.jp/?p=docs.api.page&api_docs_id=102

[9] NEC: "Service Implementation | GAZIRU Image Recognition Service | NEC." http://jpn.nec.com/solution/cloud/gazou/service.html

[10] PUX Corp.: "PUX Developers site." https://pds.polestars.jp/contents/technology.html

oneM2M 3GPP-MTC Interworking

Technology Reports

Interworking Functions for oneM2M Service Middleware Functions and 3GPP-MTC Transport Networks

In recent years, machine communication (M2M, MTC, IoT) has been used in a growing range of applications. But despite the wide diversity of areas where M2M is being applied, these applications have so far used vertically integrated architectures where each carrier constructs its own individual systems, resulting in issues such as increased development costs and development timescales. The oneM2M international standardization organization is working to mitigate such issues by adapting international standard specifications for the provision of common service functions via a standardized platform. In this article, we present an overview of the oneM2M organization and describe the first edition of the technical specifications and the interworking functions that use a 3GPP-MTC transport network.

Core Network Development Department *Takashi Koshimizu*
M2M Business Department *Ryohei Kurita*
Mei Hasegawa
Communication Device Development Department *Kohta Fujimura*

1. Introduction

In recent years, a growing number of applications have been making use of machine communication (Machine to Machine (M2M)*1, Machine Type Communication (MTC)*2, Internet of Things (IoT)*3). Despite the wide diversity of these M2M applications, it has so far been that case that M2M services are individually constructed by each service provider, and only provide services within a closed range of industries. It has been said that this vertically integrated service structure not only leads to increased development costs, longer development timescales, and duplication of development work, but also presents a barrier to the new entry of service providers who are trying to introduce M2M services.

The oneM2M international standardization organization [1] is working to resolve these issues by drawing up international standard specifications for the provision of a platform that supports the Common Service Functions (CSF) required by all M2M services. This will reduce the time and expense involved in the introduction of new services and the exchange of M2M data, thereby facilitating horizontal market expansion, the creation of innovative services that use big data, and the development of new businesses that extend beyond the confines of individual business sectors (**Figure 1**).

NTT DOCOMO is actively participating in oneM2M standardization, and

©2015 NTT DOCOMO, INC.
Copies of articles may be reproduced only for personal, noncommercial use, provided that the name NTT DOCOMO Technical Journal, the name(s) of the author(s), the title and date of the article appear in the copies.

*1 **M2M:** A general term for communication between machines without human control or intervention.
*2 **MTC:** Machine-type communication. A collective term for 3GPP machine communication with no intervening communication operations performed by humans.

has proposed interworking*4 functions that are needed when using 3rd Generation Partnership Project (3GPP)-MTC [2] transport networks. These functions were realized in the first edition of the oneM2M technical specification.

In this article, we describe the background to the formation of oneM2M and the configuration of this organization, and we present an overview of the first edition of the technical specification approved in August 2014 and their benefits. We also present an overview of interworking function between the oneM2M platform (oneM2M-PF) and the 3GPP-MTC network, and we describe the triggering scheme for functional cooperation between oneM2M-PF and 3GPP-MTC, which is a detailed practical function of this specification.

2. Overview of oneM2M and First Edition of Specifications

2.1 Background to the Establishment of oneM2M

In July 2011, to deal with the anticipated growth of activity on the international standardization of M2M services, the European Telecommunications Standards Institute (ETSI)*5 initiated a study aimed at founding a global integrated organization for the unification of standardization initiatives for M2M services. Seven organizations participated in this study: the Association of Radio Industries and Businesses (ARIB)*6 (Japan), the Alliance for Telecommunications Industry Solutions (ATIS)*7 (US), the China Communications Standards Association (CCSA)*8, ETSI (Europe), the Telecommunications Industry Association (TIA)*9 (US), the Telecommunications Technology Association (TTA)*10 (South Korea), and the Telecommunication Technology Committee (TTC)*11 (Japan).

In December 2011, a basic agreement was reached regarding the establishment of an international joint organization for M2M standardization, which became officially known as oneM2M in January 2012. In July 2012, it began forming international standard specifications for an M2M service platform.

2.2 Organizational Structure of oneM2M

As shown in **Figure 2**, the oneM2M organization broadly comprises a Steering Committee (SC), a Technical Plenary

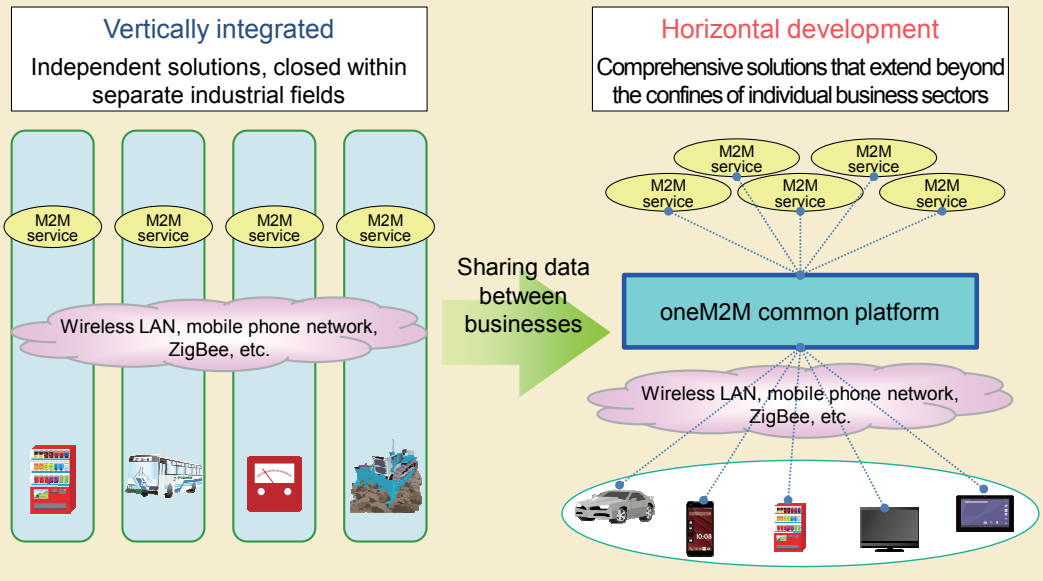


Figure 1 Direction of the M2M markets targeted by oneM2M

*3 **IoT:** General term for a style of control and communication where various "things" are connected via the Internet or cloud services.
*4 **Interworking:** Interaction between communications systems.
*5 **ETSI:** A standardization organization concerned with telecommunications technology in Europe.
*6 **ARIB:** An organization subordinate to the MIC that sets standards for systems that use the radio spectrum in the fields of communications and broadcasting in Japan.
*7 **ATIS:** Alliance for Telecommunications Industry Solutions (US).
*8 **CCSA:** China Communications Standards Association.
*9 **TIA:** Telecommunications Industry Association (US).
*10 **TTA:** Telecommunications Technology Association (South Korea).
*11 **TTC:** Telecommunication Technology Committee (Japan).

(TP), and various Working Groups (WGs).

The SC is responsible for overall administration of the oneM2M organization, draws up the organization's operating rules and objectives, and defines the scope of its studies. Based on the policies of the SC, the TP manages the progress of standardization work, and is responsible for approving the specifications examined by each of the subordinate WGs.

The scope of the activities of each WG are outlined below (**Figure 3**).

WG1: Responsible for the requirements on which the study of standardized specifications is based. Specifically, it collects M2M service use cases from various industries, extracts the common functions that are needed by each industry, classifies them into various classes of requirements (security, billing, device management, etc.), and prescribes functional requirements based on these results.

WG2: Responsible for the oneM2M architecture. Prescribes the layer model needed to implement the concept (application layer, service layer, network layer) and reference points*12 between each function in the architecture.

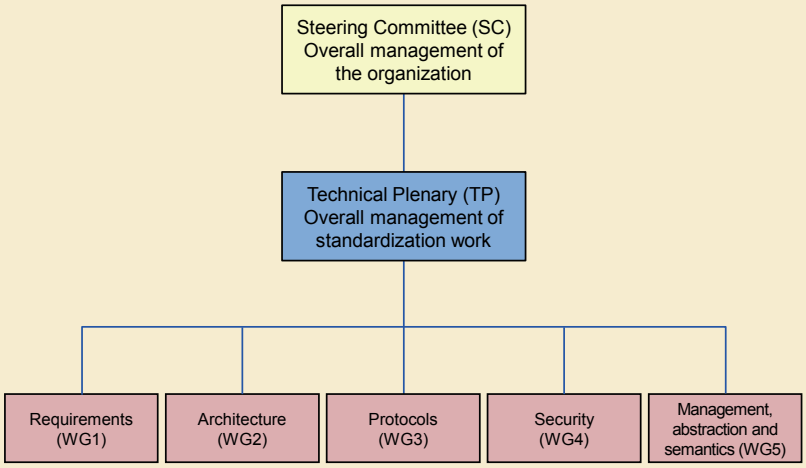


Figure 2 oneM2M organizational structure

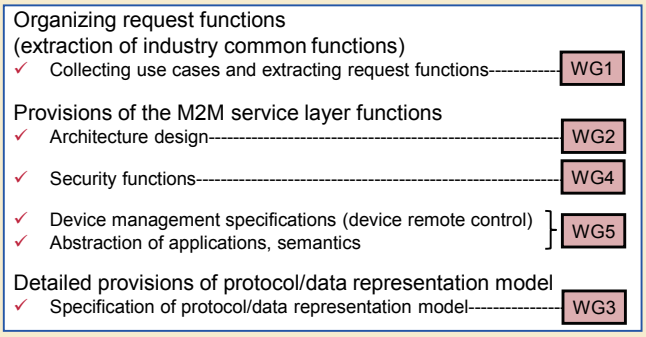


Figure 3 Principal study details in each WG

*12 **Reference point:** The interface part between elements.

WG3: Responsible for the protocols used in oneM2M. Prescribes the data representation methods used in communication protocols (data types, message parameters, resource types, etc.), and status codes associated with normal/quasi-normal procedures.

WG4: Responsible for security functions. Prescribes various security functions for M2M services (access management, service authentication, management of security credentials, ID management, etc.) and sequences for the establishment of secure communication paths between M2M nodes, etc.

WG5: Studies the abstraction of device management and applications, and prescribes device management functions that can manage and control multiple device management protocols of other standardization organizations (Open Mobile Alliance (OMA)*13/Broad-Band Forum (BBF)*14, etc.) by using an integrated interface.

2.3 First Edition of oneM2M Specifications

1) oneM2M Release-1

In August 2014, two years after its establishment, oneM2M published the first edition (Release-1) of its specification set (**Table 1**). Following the review comments that were received from outside following this publication, a revised specification was published in February 2015. There was also a release event held in December 2014 to demonstrate the first release of the oneM2M standard.

2) oneM2M Common Platform

This presents an overview of the common platform defined in the first edition of the oneM2M specification (**Figure 4**). The main scope of studies conducted by oneM2M is a common platform at the service middleware*15 layer situated between the application layer and the underlying transport layer. By adopting a standard specification for service middleware control functions in M2M services, it is possible for service providers to control a wide variety of M2M devices and applications with a single service middleware layer.

Based on the requirements extracted from use cases, the service middleware control functions are defined by a total of thirteen functional modules for purposes such as device management, application management, data management, positional information control, security control, and service billing functions (**Table 2**). The logical node that makes up the common platform is called the Common Service Entity (CSE), and the thirteen functional modules that make up the CSE are called CSFs (fig. 4).

Of these CSFs, the area mainly corresponding to the proposal submitted by NTT DOCOMO is the network service cooperation CSF associated with mobile communication networks, which is one of the functions handling the control of network congestion*16 and device triggering*17 (fig. 4, Table 2).

Table 1 List of specifications in the first edition of the oneM2M specification

Specification number	Title	Responsible WG
TS 0001	M2M Architecture	WG2
TS 0002	M2M Requirements	WG1
TS 0003	oneM2M Security Solutions	WG4
TS 0004	oneM2M Protocol Technical Specification	WG3
TS 0005	oneM2M Management Enablement (OMA)	WG5
TS 0006	oneM2M Management Enablement (BBF)	WG5
TS 0008	CoAP Protocol Binding Technical Specification	WG3
TS 0009	HTTP Protocol Binding Technical Specification	WG3
TS 0011	Definitions and Acronyms	ALL WGs

*13 **OMA:** An industry forum for standardizing and enabling technology for services and applications in mobile communications and for ensuring interoperability.

*14 **BBF:** An international organization that aims to promote the spread of broadband forums.

*15 **Service middle:** Middleware platform functions that are used in common by many different services.

*16 **Congestion:** Impediments to communications services due to communications requests being concentrated in a short period of time and exceeding the processing capabilities of the service control server.

*17 **Device triggering:** Transmitting information to a device in order to trigger an application that runs inside it.

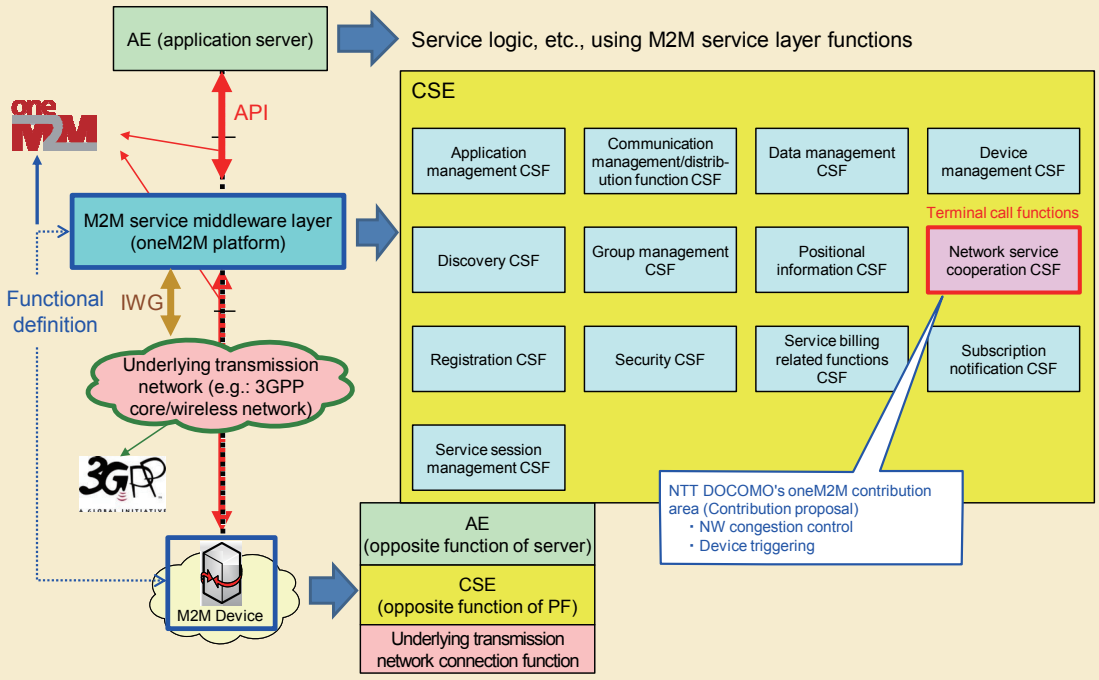


Figure 4 Constituent elements of oneM2M-PF and the underlying transmission network

Table 2 Constituent CSFs of the oneM2M CSE

Function (CSF)	Overview
Application management CSF	CSE self-configuration and diagnostic functions, AE software management functions
Communication management/distribution function CSF	Functions for management of communication between entities (CSE/AE/NSE)
Data management CSF	Functions for storage and mediation of application data, subscriber information, positional information, device information, etc.
Device management CSF	Device remote management functions (using, e.g., OMA-DM/LWM2M, TR-069, etc.)
Discovery CSF	Functions for retrieval of information resources based on requests from AEs and other CSEs
Group management CSF	Group management functions
Positional information CSF	Functions for the provision of terminal position information
Network service cooperation CSF	Access functions for the network service functions of each NSE (network)
Registration CSF	Functions for registering AEs/other CSEs/device terminals etc. to a CSE
Security CSF	Security functions (data encryption, security management, authentication, authorization, etc.)
Service billing related functions CSF	Billing management
Subscription notification CSF	Service session management
Service session management CSF	Functions for the management of subscriptions to resources and notifications when resources change

3) Conceptual Implementation of oneM2M Functions

The conceptual implementation of oneM2M functions is shown in **Figure 5**. The logical node CSE that provides oneM2M service middleware control functions is implemented as a software function in M2M devices and at each server node. An Application Entity (AE) is an application, and the figure shows how these two entities[*18] are located within the nodes. An Infrastructure Node (IN) is a server positioned on the transmission network side, which is called the infrastructure domain in oneM2M. On the other hand, in the field domains, a Middle Node (MN), such as M2M devices and M2M gateway that aggregates those M2M devices are included. The M2M devices are called the Application Service Nodes (ASNs), Application Dedicated Nodes (ADNs) and Non-oneM2M devices according to differences in the positioning of AEs and CSEs. The difference between and ADN and an ASN is that an ASN has a CSE that allows it to take charge of Non-oneM2M devices. A Non-oneM2M device has no AE or CSE and is thus unable to provide oneM2M functions, but the connection of such devices is envisaged.

2.4 Advantages of oneM2M Technical Specifications

The main benefits of using the oneM2M standard specifications are summarized below.

First, it reduces the business costs of providing services. M2M service providers do not need to prepare a separate platform for each service, and can use a common platform to initiate business by providing only applications and devices. This reduces the start-up costs and allows smaller businesses to compete, and is thus expected to result in market expansion.

Second, the oneM2M CSE is designed to be used with any underlying network (mobile, fixed, local-area, etc.), and thus offers the benefits of versatility across different transport networks and communication methods.

Third, it ensures interconnectivity with existing M2M protocols, which is a major benefit. In oneM2M, only the core protocols are prescribed, and the specification is designed assuming that interconnections will also be made with existing protocols such as HTTP that do not conform with the oneM2M spec-

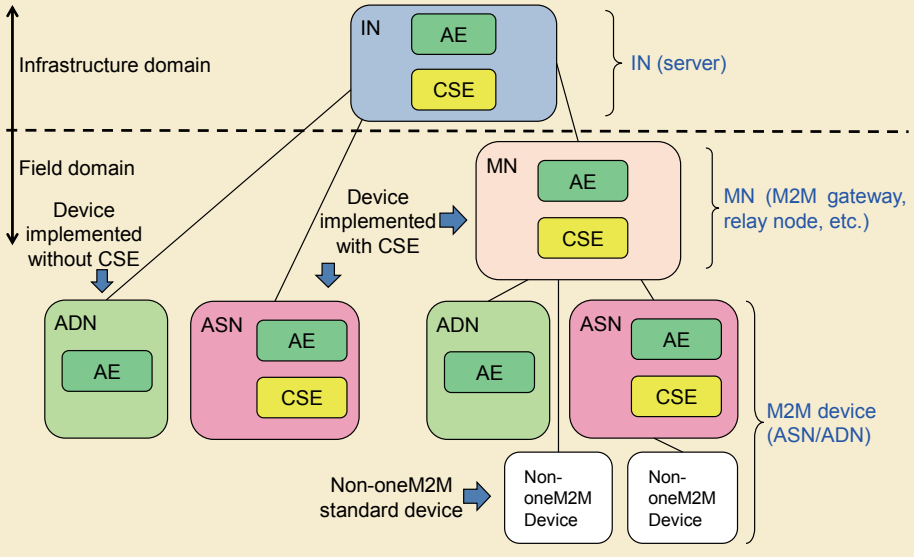


Figure 5 Conceptual implementation of oneM2M functions

*18 **Entity:** A structural element that provides functionality within a logical architecture.

ification.

Finally, in oneM2M, standards are being studied for the utilization of data such as semantics*19 that can be re-used by horizontal deployment between different business fields. For example, it is expected to promote the use of stored data and the creation of new businesses across the boundaries between business fields, for example by realizing smart cities through the cooperative use of diverse business fields such as ITS and electric power.

3. Interworking between oneM2M-PF and 3GPP-MTC Networks

The constituent functions of oneM2M-PF are first implemented by cooperation with transmission networks as shown in fig. 4. When viewed from the oneM2M-PF, any underlying transport network should work, but as a mobile communication network operator, it is essential that these new functions operate integrally in concert with our existing (3G and LTE) networks. NTT DOCOMO has been particularly active in this regard in the standardized specifications of oneM2M Release-1. In fig. 4, it is envisaged that any network can be applied as the underlying network, but in 3GPP, since MTC network functions have already been specified independently, the functions on the oneM2M side had to be designed so as to be highly compatible with these functions. Therefore, in order to realize the interworking of MTC networks in oneM2M and 3GPP, we tried to take a broad view of the technical content of both systems and promoted activities to enhance their compatibility. The following sections describe the features of M2M devices and the considerations and aims necessary for functional cooperation, and presents a summary of the 3GPP Release-11 MTC network platform and its triggering scheme.

3.1 Features of M2M devices, and Considerations and Aims of Functional Cooperation

1) M2M Device Features

There are a wide variety of examples of machine communication (M2M, MTC, IoT, etc.), but these are considered to involve terminal devices that are much smaller and whose functions are much more limited than ordinary mobile phones and the like. These modes of use are considered to encompass many applications where, for example, sensors are distributed around a room to form miniature terminals with limited sensing capabilities as a compact and inexpensive way of measuring environmental parameters such as temperature, humidity and dust levels. These devices may even be so small as to not require a power supply [3] [4].

Environment sensing M2M devices such as these tend to be small, functionally limited, and have a limited power supply capacity. Therefore, these devices only operate for short periods of time, and once they have finished making measurements, they typically transition to a dormant*20 state or transmit only small amounts of data. On the other hand, as a use case of which IoT is a typical example, it is envisaged that a huge quantity of devices will be used, so studies are being conducted on ways of efficiently accommodating these M2M devices in a wide area network [5] [6].

2) MTC-GW Proxies

The terminals that are measurement devices with characteristics as discussed in 1) above, and functions that act as intermediates between these terminals and a public network or other wide area network are MTC gateways (MTC-GW) called device proxies. As their name suggests, they perform functions on behalf of devices. **Figure 6** describes these functions and effects by way of an example. In the uplink communication of measured values from the measurement devices to the network side (e.g., temperature measurements taken once per hour), the measurement devices do not communicate during other time periods and thus transition into a dormant state, while the measured values are stored in the MTC-GW proxy shown in fig. 6. The measurement devices are also arranged so that communication with the GW can be performed using short-range radio communication so as to reduce their power consumption. On the other hand, when the server side that lies on the network wants to obtain the device measurement

*19 **Semantics:** Arrangements relating to the meaning conveyed by data.

*20 **Dormant:** One of the states of communication equipment in which it is standing by to receive communication.

values, it accesses the MTC-GW proxy that is always connected, from where it retrieves the stored data. This sort of basic operation is being studied as a way of accommodating large numbers of inexpensive sensing devices that have limited functions, but in cases where real-time performance is not needed [4]. These are a large part of the background in the study of oneM2M (fig. 5), resulting in a configuration including an MN that aggregates devices.

3.2 Overview of MTC Related Functions in 3GPP

1) Connectivity via an Underlying 3GPP Transmission Network

Figure 7 shows a simplified view of the sort of InterFace (IF) through which connectivity is provided by the IN on the server side and the MTC-UE (ASN or MN) when using a 3GPP network as the underlying transmission network. The M2M devices considered for oneM2M correspond to the UE as shown in the figure on a 3GPP architecture including middleware (MTC-GW and MTC proxy) nodes.

We will first describe the MTC-UE shown on the left side of fig. 7. According to the oneM2M standard, an AE and CSE are situated in the UE. As in fig. 5 above, although the ASN and MN have this sort of oneM2M function, they appear as a single UE from the viewpoint of the 3GPP network. Thus the connections between ASN/MNCSE and the 3GPP network are connected via Uu-IF (3GPP air IF*21) [7].

On the other hand, the connectivity between the oneM2M network server functions defined by 3GPP as SCS: Service Capability Server*22 (CSE) and the 3GPP network is configured as shown at the right side of fig. 7, where the Infrastructure Node-CSE (IN-CSE) and 3GPP underlying transmission network, where the User Plane (U-Plane)*23 information is connected by the Mcc-IF*24, and the Control Plane (C-Plane)*25 is connected by the Tsp-IF*26.

The 3GPP underlay network in fig. 7 is a simplified representation of how IP connectivity is provided between the

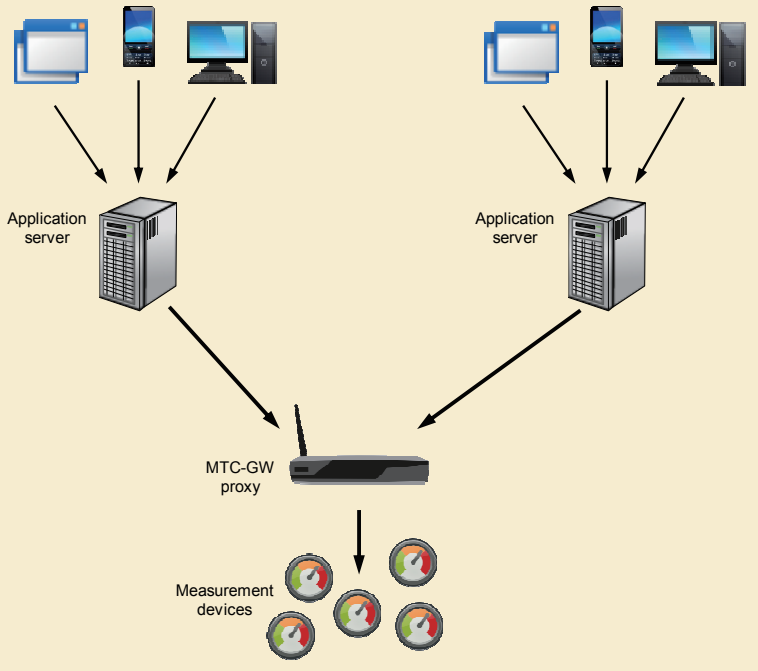


Figure 6 The effect of an MTC-GW proxy

*21 **Air IF:** General term for a radio communication link from a mobile terminal to a mobile base station.
*22 **SCS:** Equipment that provides a termination point for communication with terminals, implemented by an M2M application.
*23 **U-Plane:** A path for the transmission of user data to the C-Plane, which is a control signal transmission.
*24 **Mcc-IF:** One of the reference points in the oneM2M specification; the interface between CSE and the underlying network.
*25 **C-Plane:** This refers to the control plane, a series of control processes that is executed when a call is established and other such times.
*26 **Tsp-IF:** One of the reference points in the 3GPP MTC specification; the interface between MTC-IWF and SCS.

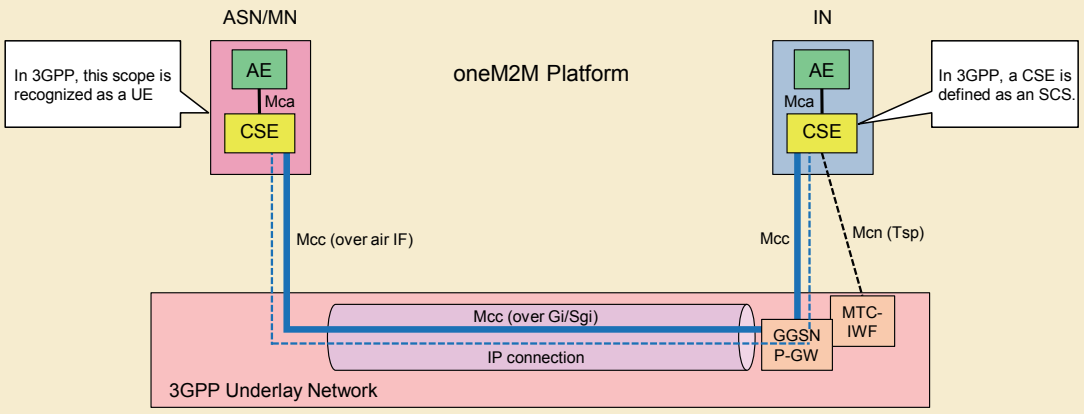


Figure 7 Providing connections via the underlying 3GPP transmission network

ASN/MN-CSE (MTCUE) and IN-CSE. In effect, this connectivity is provided by the MTC network platform in 3GPP Release-11. Its functions are shown below.

2) Underlying Transmission Network, Taking the 3GPP Release-11 MTC Network Platform as an Example

The original oneM2M discussions included a discussion of which 3GPP release's MTC network platform should be selected. NTT DOCOMO proposed using the 3GPP Release-11 MTC network platform based on its maturity and functional content, and as a result this proposal was incorporated into the oneM2M Release-1 specification. The overall architecture of the MTC network platform in 3GPP Release-11 is shown in **Figure 8** [8]. The locations labeled in red are the parts where functional entities are added as studied for oneM2M in this MTC network platform.

As the primary function, the SCS prescribed on the 3GPP side corresponds to the oneM2M IN-CSE described in chapter 2 (fig. 5). Also, the Mcc-IF that carries the U-Plane corresponds to Gi/SGi-IF*27, and the oneM2M Mcn-IF that carries the C-Plane signals corresponds to the Tsp-IF in 3GPP. On the terminal side, the oneM2M devices and middleware (ASN, ADN, etc.) also correspond to the 3GPP UE. When aligning the two standardized specifications, it initially took some time to achieve convergent discussions because the two standards had been developed by different groups in different locations. However, with the participation of experts on both sides, we finally managed to adapt and integrate these functions.

3) SMS Incoming Routes

The triggering of the 3GPP Release-11 MTC network platform by incoming traffic is implemented using only SMS*28. We will describe an example where these incoming message routes and principal function entities are received by a specific terminal from Application Server (AS)*29 (1) in fig. 8.

The AS initiates a call to the SCS by applying unique identifying IDs called External IDs (EXT-IDs) to the call signals. The SCS (oneM2M IN-CSE) judges where the corresponding terminals are placed in the 3GPP network, and transfers the message to the 3GPP Machine Type Communication-Inter Working Function (MTC-IWF)*30 via the Tsp-IF. This MTC-IWF queries a Home Subscriber Server (HSS)*31 and extracts an Internal ID (INT-ID) needed for delivery inside the 3GPP network. Since incoming calls to MTC terminals in a Release-11 MTC network are SMS messages, an Mobile Station International Integrated Services Digital Network Number (MSISDN)*32 for delivering an SMS message to a specific terminal is essentially extracted as the INT-ID. This information is handed over via the T4-IF to an entity that is

*27 **Gi/SGi-IF:** One of the reference points in the 3GPP EPC specification; the interface between SGSN/PGW and an external server.
*28 **SMS:** A service for transmitting/receiving short text-based messages. SMS is also used for transmitting/receiving mobile terminal control signals.
*29 **AS:** A server that runs an application to provide a service.
*30 **MTC-IWF:** Equipment that implements functions including Device Triggers and authentication of connection request and control-plane signals on a 3GPP network.
*31 **HSS:** A subscriber information database in a 3GPP mobile network that manages authentication and location information.
*32 **MSISDN:** The phone number assigned to each subscriber as specified by the 3GPP.

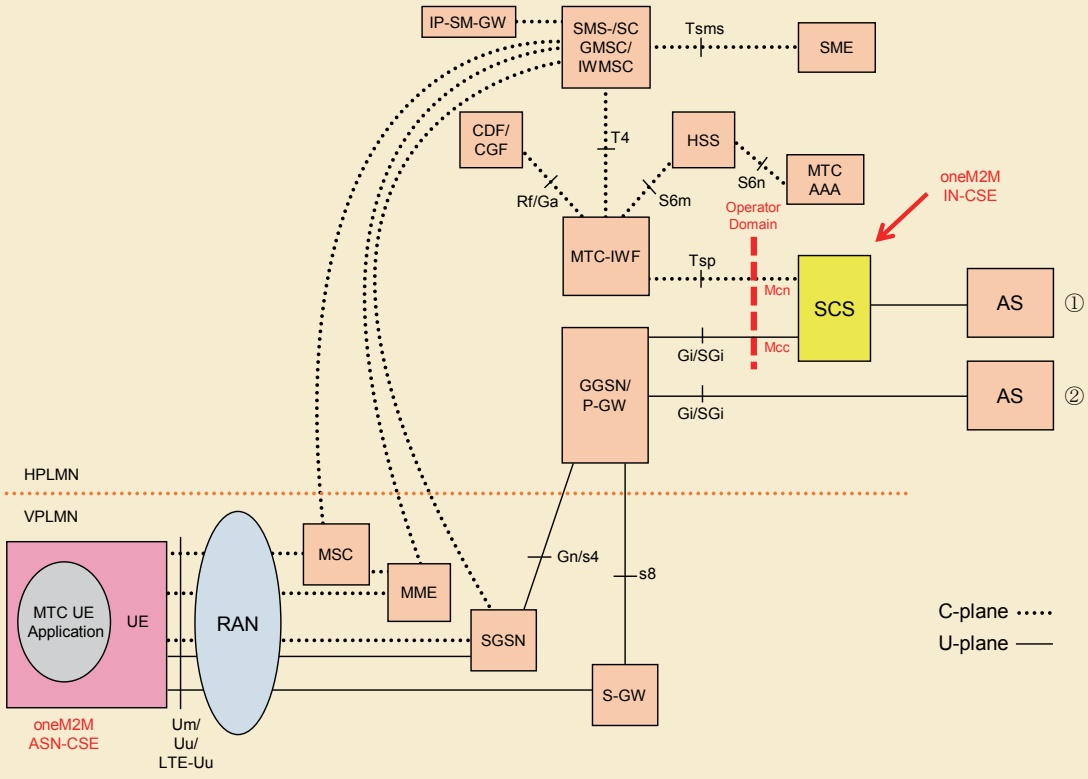


Figure 8 MTC platform (underlying transmission network) of 3GPP Release-11

required for SMS start-up (SMS-SC, etc.), and is delivered as an SMS message request to an Mobile Switching Center (MSC)*33 for 3G-CS, an Serving General packet radio service Support Node (SGSN)*34 for 3G-PS, or an Mobility Management Entity (MME)*35 for LTE. After performing paging*36 operations and the like, the message is delivered to the terminal's MTC-UE [8].

Although the route of an inbound SMS text is described above, it is necessary to register suitable information in the SCS (IN-CSE) before delivery is performed, and a reception operation must be started up only when the IN-CSE has obtained the status of a UE. With these mechanisms, it becomes possible for the first time to perform control without unnecessary actions such as forcing the reception of messages when the power is switched off.

3.3 Triggering Method based on Functional Cooperation between oneM2M and 3GPP Network

In a 3GPP network as described above, highly reliable message delivery operations can be implemented by ascertaining details about the UE state (i.e., switched off, dormant, busy, etc.). On the other hand, since a newly created oneM2M-PF does not have this sort of UE state information, it can lead to problems such as initiating reception to a terminal that is not switched on, for example. To address this sort of problem, NTT DOCOMO proposed a function that works by securing and holding the minimal state of a UE even in a oneM2M-PF, and finally allows the IN-CSE to report/hold these UEs when there is a change in the status of the UE. In addition, incoming content from IN-AE is

*33 **MSC:** A logical node having CS functions specified by 3GPP.
*34 **SGSN:** A logical node having packet communication functions specified by 3GPP.
*35 **MME:** A logical node for C-Plane control that accommodates an eNB and provides mobility control functions and other functions.
*36 **Paging:** Calling all mobile terminals at once when there is an incoming call.

also considered and connected to a method for selecting the reception behavior. The registration function and incoming call operations are described below.

1) Procedure for Forming an IN-CSE Connection from the Terminal Side

Figure 9 shows the procedure whereby an MTC-UE (ASN/MN-CSE) registers a UE destination address in the IN-CSE via the attach*37 procedure, and then forms a connection. In this figure, steps (1)-(3) are taken directly from the procedure prescribed by 3GPP [2], and through these steps, a connection is formed between the UE and IN-CSE at step (4).

Through this procedure, the IN-CSE maintains a set of destination addresses of this terminal and the ID specified by an arbitrary UE (M2M-Ext-ID). When a change has occurred in the UE's own state (e.g., it is powered down or becomes dormant), it has a mechanism that reports this to the IN-CSE. The IN-CSE is a part that saves this state information (including while connected), considers the incoming content from the IN-AE, and manages the procedure for selecting the reception behavior. If there is a change in the destination address (Point of Attachment (PoA)) of the UE after step (4), this is reported to the IN-CSE at step (5) [7].

2) Procedure for Forming a Connection from the IN-CSE to a Terminal

After completing the above registration procedure, it becomes possible to receive the first message on the terminal. This procedure is shown in **Figure 10**.

First, the IN-AE sends the IN-CSE an incoming call request including an M2M-Ext-ID that points to a particular terminal (fig. 10 (1)). On receiving this request, the IN-CSE identifies the destination on the underlying transmission network by consulting the Domain Name System (DNS)*38 based on this ID (fig. 10 (2)).

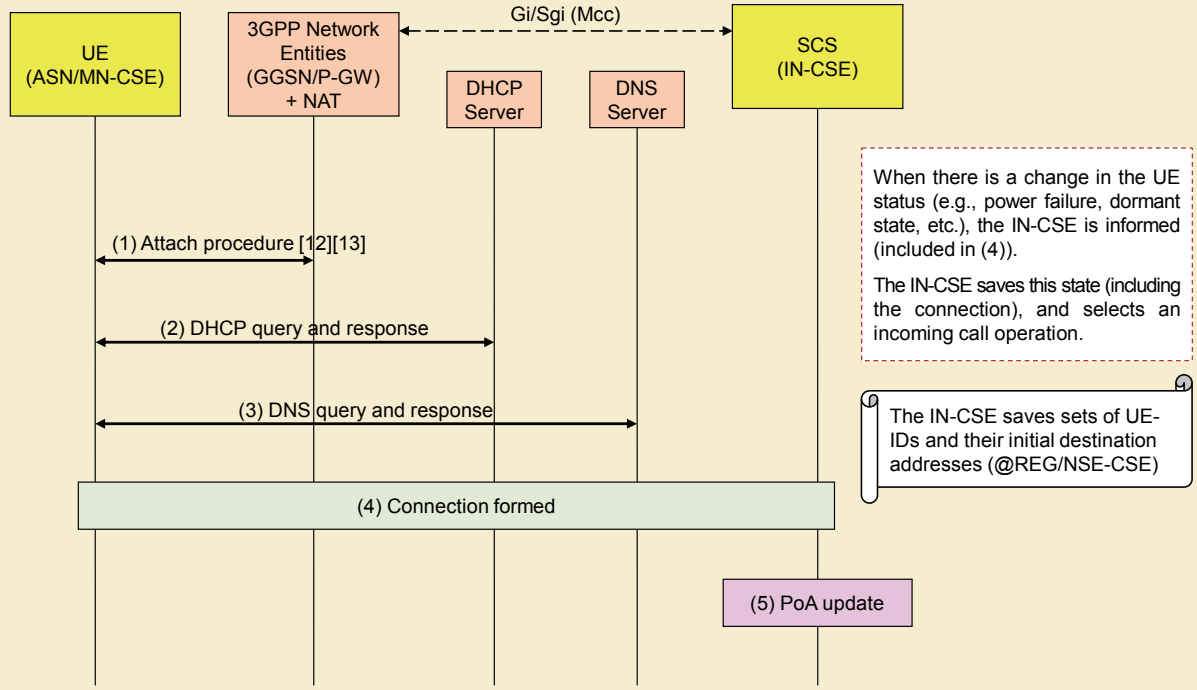


Figure 9 Connection procedure from a terminal to IN-CSE (UE Initiated)

*37 **Attach:** A procedure, and the status thereof, for registering a terminal on the network when, for example, its power is switched on.

*38 **DNS:** A system that associates host names with IP addresses on IP networks.

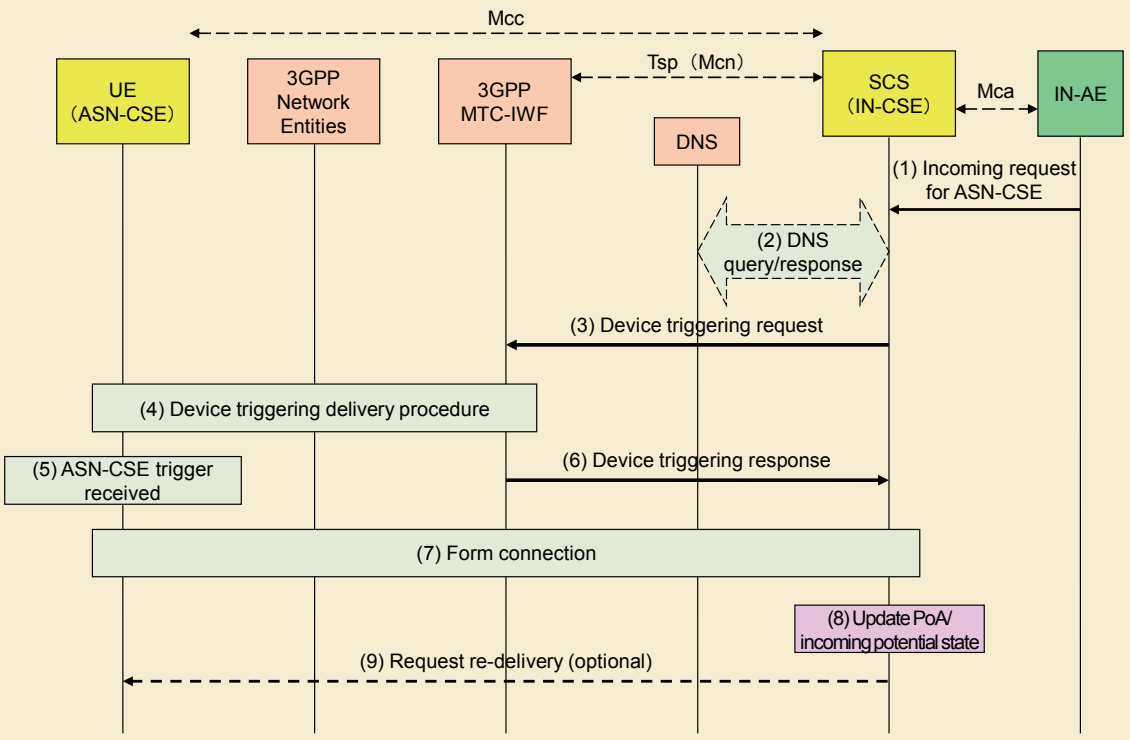


Figure 10 Procedure for forming connections from the IN-CSE to a terminal (NW Initiated)

Next, the IN-CSE transmits an incoming call request to the MTC-IWF of the underlying network to which it should be transferred (in this case, the 3GPP network) (fig. 10 (3)). The subsequent device triggering procedure (fig. 10 (4)) conforms to the 3GPP-MTC specification [8], and is used to deliver the trigger to the ASN-CSE terminal (fig. 10 (5)). On receiving this trigger, the UE initiates the procedure to establish a connection with the IN-CSE (fig. 10 (6), (7)).

When this happens, the IN-CSE is able to update the state necessary for reception and PoA, which is the transmission destination (fig. 10 (8)) [7].

4. Conclusion

In this article, we began by presenting an overview of the oneM2M organization, the first edition of its specification, its functions, and so on. We also reviewed the characteristics of M2M devices and an overview of the 3GPP-MTC network, and finally we described the interworking functions needed for the integral operation of oneM2M with 3GPP networks and MTC transmission networks. The method for triggering devices between oneM2M and 3GPP-MTC networks is one part of the contribution made by NTT DOCOMO in cooperation with other companies. By implementing this sort of function in oneM2M, it will be possible to integrate oneM2M-PF with the 3GPP underlying transmission network, and to make use of cooperative control.

In oneM2M, now that the Release-1 specification has been completed, discussions have started on the requirements for the next release. Also, the activities for Release-12/13 of 3GPP include studies of AESE (Architecture Enhancement for Service Exposure) [9] for enhanced cooperation with external servers such as oneM2M, MTC group control functions [10], and functions related to mon-

itoring [11].

The triggering technique has forestalled the proposal of other methods to replace SMS, yet is expected to result in a more convenient triggering scheme for the provision of M2M services once further functional improvements have been made. M2M services have a broad range of applications, and it is thought that they will also make further advances in applications to, for example, the IoT. Service applications such as these are evolving, and are expected to continue developing in the future, so we hope to continue following these trends. Also, from the viewpoint of commercial use and the like, we intend to focus on these trends including partnerships with other companies.

References

[1] The oneM2M Web site. http://www.onem2m.org/

[2] 3GPP TR23.888 V11.0.0: "System improvements for Machine-Type Communications (MTC), Release-11," Dec. 2012.

[3] T. Kinoshita: "Build a smart society with wireless M2M!" Hitachi, New Generation M2M Consortium Symposium, Nov. 2013. http://ngm2m.jp/m2m/files/kouen_kinoshita.pdf

[4] D. Boswarthik, O. Elloumi and O. Hersent: "M2M Communications: A systems approach,"1st Edition, Wiley, 2012.

[5] T. Fujita, Y. Goto and S. Koike: "M2M architecture and technology challenges; Special Issue: ICT to support M2M services," IEICE Journal, Vol. 96, No. 5, May 2013.

[6] K. Hori, M. Hattori, T. Yoshihara, A. Ikenoue and N. Yamazaki: "M2M Area Networks; Special Issue: ICT to support M2M services," IEICE Journal, Vol. 96, No. 5, May 2013.

[7] oneM2M TS-0001 V1.0.0: "oneM2M Functional Architecture," Dec. 2014.

[8] 3GPP TS23.682 V11.5.0: "Architecture enhancements to facilitate communications with packet data networks and applications, Release-11," Sep. 2013.

[9] 3GPP TR23.708 V1.0.0: "Architecture Enhancements for Service Capability Exposure, Release-13," Dec. 2014.

[10] 3GPP TR23.769 V1.0.0: "Group based Enhancements, Release-13," Dec. 2014.

[11] 3GPP TR23.789 V1.0.0: "Monitoring Enhancements, Release-13," Dec. 2014.

[12] 3GPP TS23.401 V11.11.0: "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access, Release-11," Dec. 2014.

[13] 3GPP TS23.060 V11.12.0: "General Packet Radio Service (GPRS); Service description; Stage 2, Release-11," Dec. 2014.

F-SCP D-SCP Round robin selection

Technology Reports

"F-SCP" Service Control Equipment Providing Higher Reliability Services

IPSCP provides important functions on the NTT DOCOMO mobile communications network such as subscriber information management, call sending and receiving, and providing additional services. Therefore, these systems require a high level of reliability. To improve reliability, we separated IPSCP into F-SCP, which controls IPSCP, and D-SCP, which is the DB section. By separating these sections, opposing devices such as exchange equipment can access subscriber information (in D-SCP) no matter which F-SCP the opposing devices access. Thus, higher reliability can be ensured by distributing load across F-SCPs and controlling access if an F-SCP malfunction occurs. This article describes F-SCP.

Core Network Development Department

Tomonori Kagi
Jun Kakishima
Kohei Yamamoto
Toru Hasegawa

1. Introduction

On its mobile communications network, NTT DOCOMO uses IP Service Control Point (IPSCP) to achieve Home Location Register (HLR)*1 and Home Subscriber Server (HSS)*2 functions to manage user subscriber information and location information, control call sending and receiving, and location registration. As well as the recent spread of M2M (Machine to Machine)*3 terminals that has increased subscriber numbers that must be managed, new services such as Voice over LTE (VoLTE) are also projected to increase traffic for IPSCP. For this reason, we separated DB functions from IPSCP as Database SCP (D-SCP) to efficiently scale out*4 equipment to handle subscriber increases [1].

Currently, we use IPSCP as control sections, however, we will deploy Front end SCP (F-SCP) as the successor to IPSCP to cope with future increases in traffic, and to enable processing to continue and provide users with reliable services during disasters or malfunction events (**Figure 1**).

This article describes F-SCP device configuration, load distribution and methods of improving reliability, as well as issues with separation and their countermeasures.

2. Improving Reliability by Round Robin F-SCP Selection

Up to now, all F-SCP-opposing devices selected the destination IPSCP based on subscriber telephone numbers etc. Because the F-SCP to be deployed does not store subscriber information, opposing devices do not need to select F-SCP based on subscriber information. For this reason, the signal destination F-SCP will be selected by round robin selection*5 to distribute load and risk (**Figure 2** (a)). Since the conventional

©2015 NTT DOCOMO, INC.
Copies of articles may be reproduced only for personal, noncommercial use, provided that the name NTT DOCOMO Technical Journal, the name(s) of the author(s), the title and date of the article appear in the copies.

*1 **HLR:** A logical node defined by the 3GPP with functions for managing subscriber information and call processing.
*2 **HSS:** A subscriber information database on a 3GPP mobile network that manages authentication and location information.

IPSCP adopts an ACT/SBY configuration*6, if both ACT/SBY units malfunction the relevant subscriber information becomes inaccessible. However, with round robin selection, services can be continued even if a number of F-SCPs malfunction. Also, when there is an increase in signals for certain numbers due to traffic spike*7, the load can be distributed across a number of F-SCPs.

Furthermore, if an F-SCP malfunction occurs, each opposing device removes

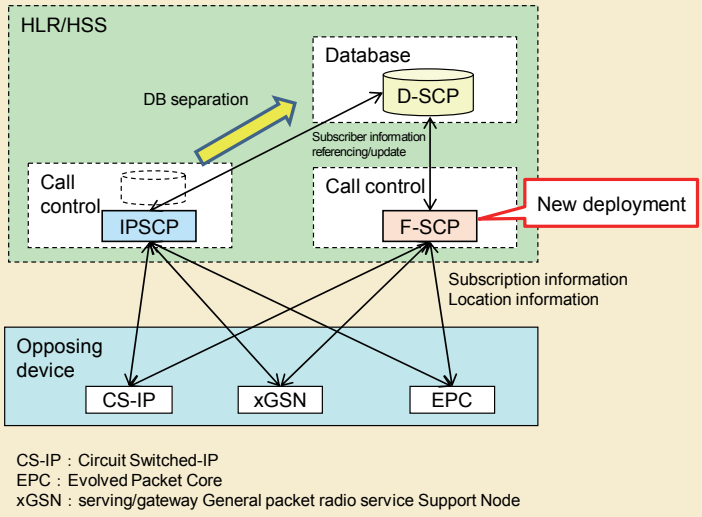


Figure 1 F-SCP and D-SCP network configuration

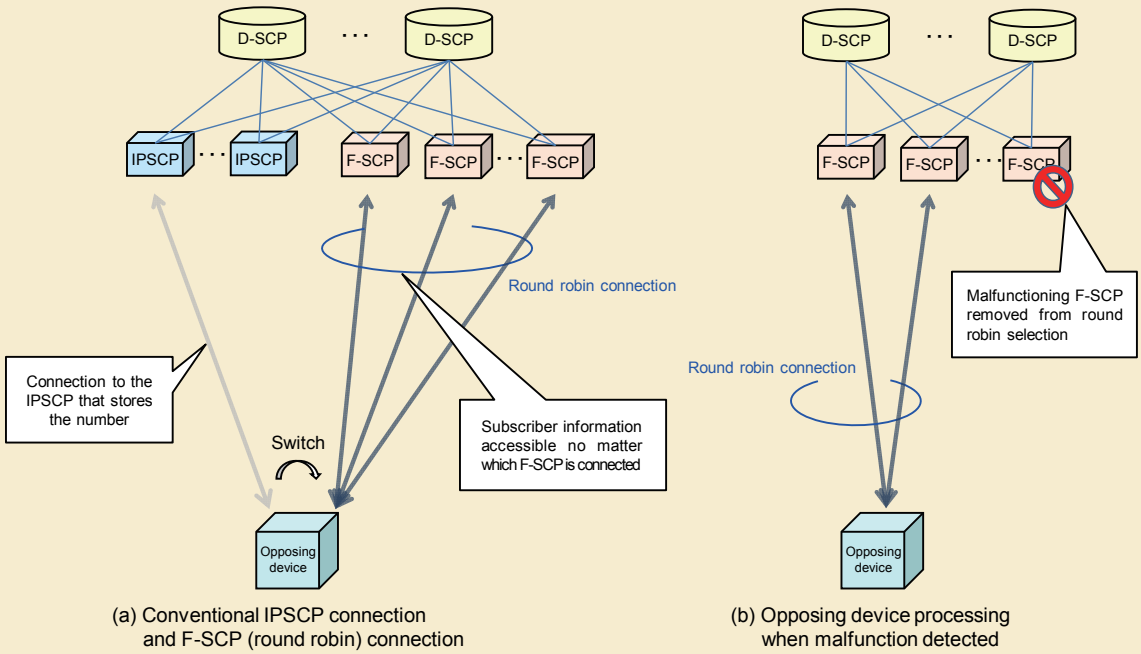


Figure 2 Round robin overview

*3 **M2M:** Machine-to-Machine Communications between machines. Systems that enable machines to communicate with each other without any human mediation.

*4 **Scale out:** Adding and assigning new resources to reinforce processing capacity when service requests increase and there is insufficient processing capacity on the network.

*5 **Round robin selection:** A selection method using a round robin. A Round robin is one way to distribute load over a network. It entails preparation of a number of devices capable of the same processing, and allocating requested processes to them in sequence.

*6 **ACT/SBY configuration:** A system configuration in which two servers perform the same function with one server in active mode (ACT) and the other in standby mode (SBY). If the ACT server malfunctions, the SBY server immediately takes over to prevent service outages. The ACT state is always retained in the SBY in readiness for switching over.

*7 **Traffic spike:** A sudden increase in traffic.

the F-SCP judged to have failed from the round robin selection, stops sending signals to it, and continues services (Figure 2 (b)).

3. F-SCP Equipment Configuration

1) Equipment Configuration

The hardware configurations of IPSCP and F-SCP are shown in **Figure 3**.

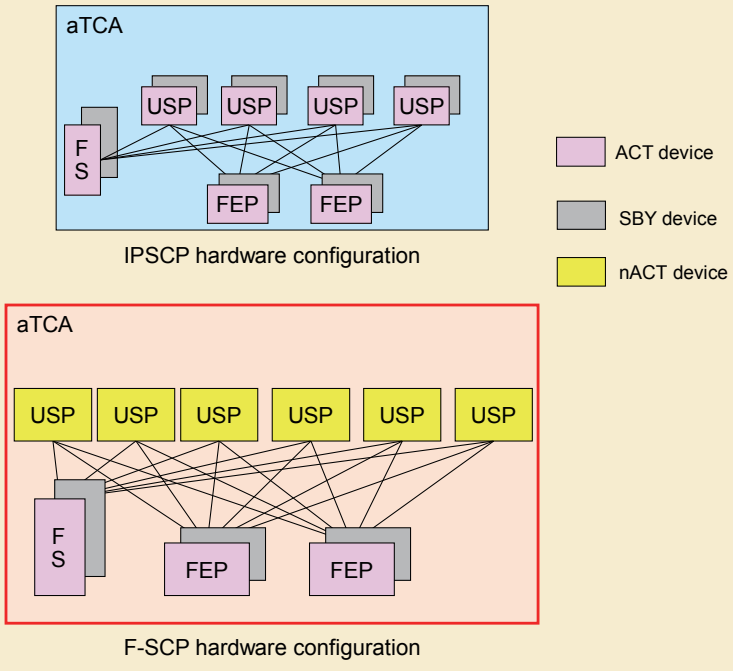
Both IPSCP and F-SCP are equipped with a Front End Processor (FEP), File Server (FS) and User Service Processor (USP).

FEP is a blade*8 that has Diameter*9/GSM-MAP (Mobile Application Part)*10 protocol termination, while FS is a blade that has HTTP/SO (Service Order)*11 protocol termination and regulatory control. These functions are the same in IPSCP and F-SCP. Furthermore, to ensure high reliability, both FEP and FS are configured for ACT/SBY.

USP is a blade that resolves the address to determine D-SCP and processes calls to provide services based on subscriber information. Because IPSCP retains subscriber information in USP, user call processing must be performed in a specific USP. High reliability is ensured by configuring USP for ACT/SBY, by switching to the SBY system if there is a USP malfunction. In contrast, because F-SCPs do not retain subscriber information in USP, it's possible to switch to another USP if one malfunctions. For this reason, the nACT configuration*12 was adopted for F-SCP because processing is possible even if more than one USP fails.

2) Round Robin Selection

The opposing device selects the FEP and FS individually using round robin selection. Round robin selection is also performed for blades in F-SCP equipment for load distribution and improved reliability. The FEP or FS that receives signals from an opposing device performs USP round robin selection. An overview of USP round robin selection is shown in **Figure 4**.



aTCA : advanced Telecommunications Computing Architecture

Figure 3 IPSCP and F-SCP hardware configurations

*8 **Blade:** A device inserted into a blade server case. Mainly refers to servers equipped with a CPU and memory.

*9 **Diameter:** An extended protocol based on Remote Authentication Dial In User Service (RADIUS), and used for authentication, authorization and accounting in IMS.

*10 **GSM-MAP:** A communications protocol used between HLR and SGSN.

*11 **SO:** A protocol used for transmitting and receiving signals with customer information management systems.

*12 **nACT configuration:** Distributes the load across n number of servers operating in parallel. If a server malfunctions, its processing can be taken over by another server.

4. Isolation Control

F-SCPs must be able to handle call processing functions and continue services even when malfunctions occur. This chapter describes typical F-SCP and USP system isolation functions for service continuity.

Isolation is an opposing device or opposing blade function that disables round robin selection for the relevant F-SCP or USP.

4.1 F-SCP System Isolation

If there is an F-SCP system malfunction, F-SCP autonomously performs system isolation processing. After that, opposing devices detect that the F-SCP has been isolated, and delete it from the round robin selection.

The following describes triggers for F-SCP system isolation.

(1) Three or more USP units have malfunctioned

(2) Both FS ACT and SBY systems have malfunctioned (double-system failure*13) and restarted (service is suspended during restart (device reboot))

(3) Both FEP ACT and SBY systems have malfunctioned (double system failure) and restarted

Recovery is only possible manually using commands. This is because while the line between F-SCP and the opposing device is unstable there is a risk of repeated isolation and recovery, and because the stability of F-SCP must be confirmed.

The main criteria used to judge F-SCP isolation in an opposing device are as follows:

- The link is disconnected (GSM-MAP connection)
- When Stream Control Transmission Protocol (SCTP)*14 Association*15 is not established, Abort*16 is returned by an F-SCP for a connection request from an opposing device, or, the opposing device detects an abnormality with the F-SCP health check*17 (Diameter connection)
- The opposing device Load Balancer (LB)*18 is informed from the F-SCP that connection is not possible (HTTP connection)

An example of the malfunction detection and recovery sequence is shown in **Figure 5**.

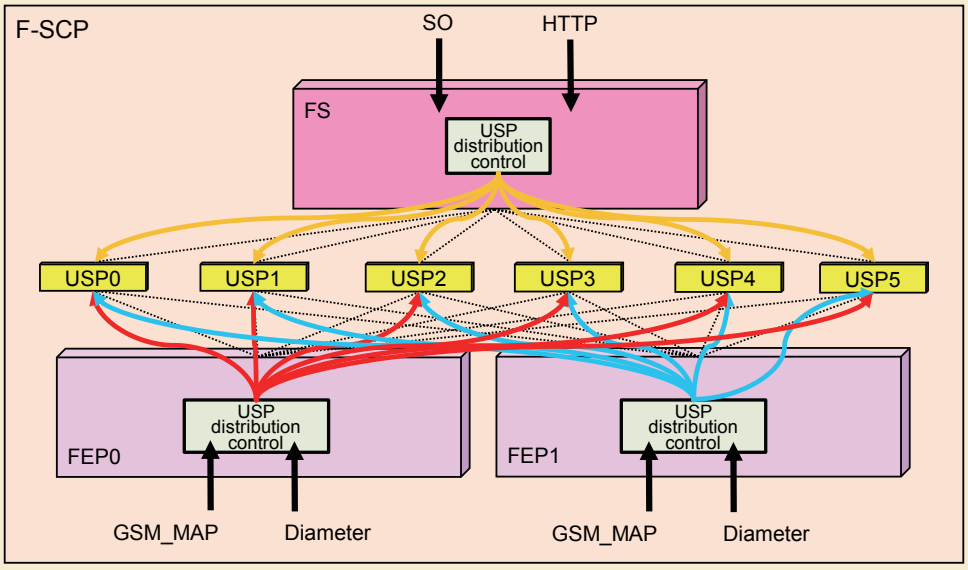


Figure 4 USP round robin overview

*13 **Double-system failure:** A failure that occurs in both the active and standby systems in a redundant configuration.

*14 **SCTP:** A transport layer protocol created to transmit telephone network protocols over IP.

*15 **Association:** A communications route established between a client and server with SCTP.

*16 **Abort:** Refusal of a request signal or suspension of communications.

When F-SCP automatically performs isolation, SCTP communications are instantly cut to prevent opposing device signal buffer overflow and impacts on resources.

4.2 USP Isolation

If a USP is malfunctioning, FEP/FS delete it from the round robin selection to ensure service continuity.

FEP/FS use the following detection triggers. If services are judged not to be continuous, they isolate the relevant USP.

(1) USP restart event

(2) USP malfunctions and errors in communications with USP detected periodically

The USP isolates itself with any of the following detection triggers:

(3) All health checks abnormal between D-SCPs (DBP: Data Base Processor*19)

Recovery triggers are as follows:

- USP restart has finished
- USP communications status is normal for a certain amount of time
- At least one health check possible between D-SCPs (DBP)

It is also possible to isolate/recover a specific USP manually with commands.

5. Issues with Round Robin Selection

1) Issues

If call processing becomes concentrated on a number range stored in a particular D-SCP when performing round robin selection with separated F-SCPs and D-SCPs, there is a possibility of signals from multiple F-SCPs converging on one D-SCP, which could cause the D-SCP to become congested*20. To solve this issue, F-SCPs are equipped with functions control flow to D-SCPs.

2) Flow Control

An overview of flow control is shown in **Figure 6**. With a D-SCP, the amount of traffic is periodically notified to FS from each DBP. Monitoring is performed in FS in D-SCP so that the amount of DBP traffic does not exceed its threshold. If the amount of traffic exceeds the threshold per unit time, the D-SCP FS notifies of the congestion to all F-SCP

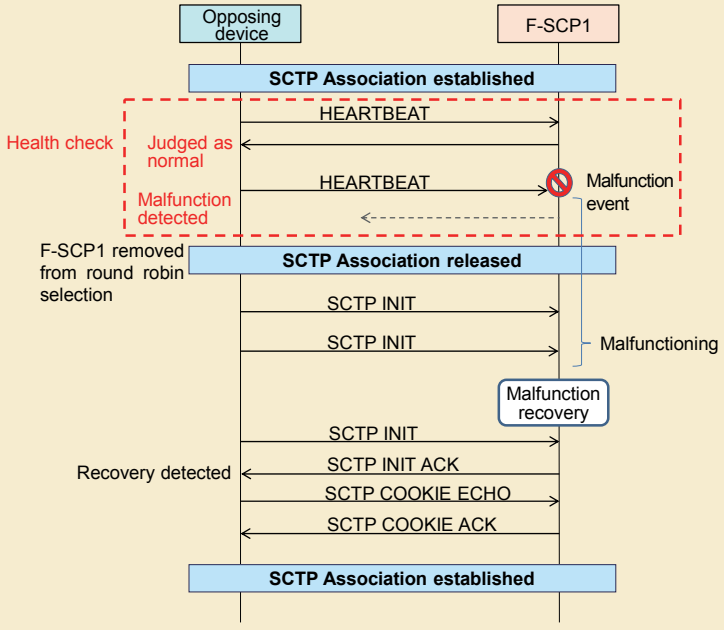


Figure 5 Malfunction detection and recovery sequence example (link disconnected)

*17 **Health check:** A periodic check of the operation of adjacent devices to detect abnormalities if they occur.

*18 **LB:** A device that distributes the load of request signals from clients across a number of servers, and detects malfunctions.

*19 **DBP:** The blade that stores subscriber information in a D-SCP. Notifies subscriber information to an F-SCP for reference requests from the F-SCP, and updates subscriber information for update requests.

*20 **Congestion:** Impediments to communications services due to communications requests being concentrated in a short period of time and exceeding the processing capabilities of the communications control server.

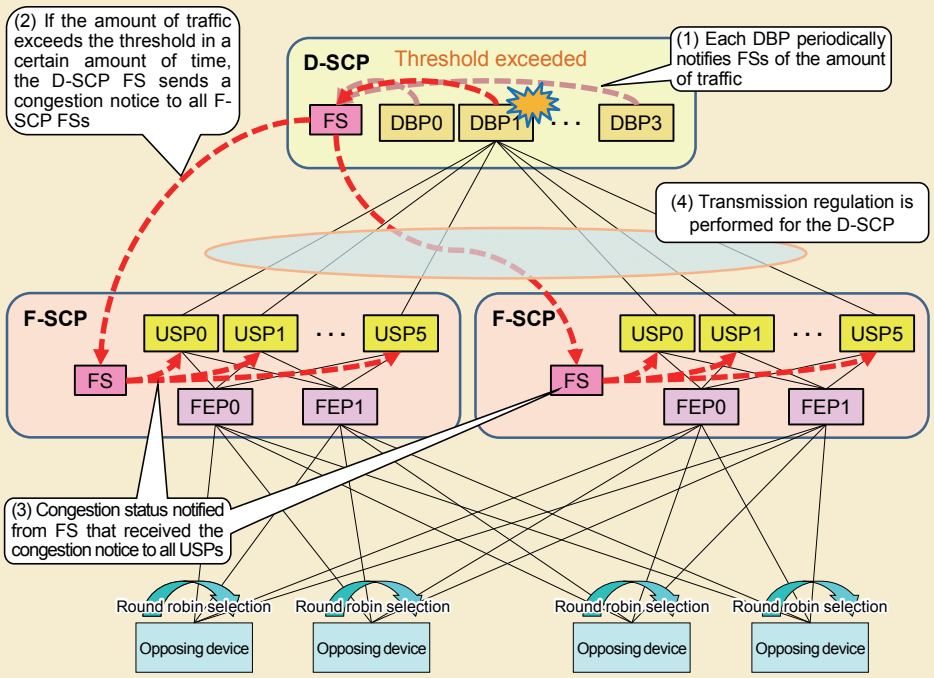


Figure 6 Overview of flow control

FSs. The F-SCP FSs then notify all USPs in the system of the congestion and control transmission to the D-SCP.

6. Conclusion

With F-SCP positioned as successor technology to IPSCP, this article has described related round robin selection functions between opposing devices, device configurations, round robin selection functions in devices, isolation control functions and flow control functions for D-SCP congestion. Deploying F-SCPs and D-SCPs will enable the quick response and flexibility needed to handle future increases in subscribers and traffic. These systems will also enable NTT DOCOMO to provide users with uninterrupted services when there are malfunctions such as during disasters.

We plan to move all opposing device connections from IPSCP to F-SCP to further improve performance and add more functionality.

Reference

[1] K.Otsuka, et al.: "Enhanced Service Control Equipment Supporting Diverse NTT DOCOMO Services," NTT DOCOMO Technical Journal, Vol. 14, No. 4, pp. 37-42, Apr. 2013.

VoLTE S8HR Roaming

Standardization

Standardization of New VoLTE Roaming Architecture

The GSM Association recently endorsed a new VoLTE roaming architecture, S8HR, as a candidate for VoLTE roaming. Unlike previous architectures, S8HR does not require the deployment of an IMS platform in VPLMN. This is advantageous because it shortens time-to-market and provides services universally without having to depend on the capability of VPLMN. This article covers the technical characteristics, basic call controls, technology trials, and future development of S8HR.

Core Network Development Department

Motohiro Abe
Itsuma Tanaka
Shin-ichi Isobe

1. Introduction

In the 3G era, operators provide voice and SMS services via circuit switch functions. However, the LTE network does not have such functions, so on LTE, operators must provide voice and SMS services via the IP Multimedia Subsystem (IMS), which is specified by the 3rd Generation Partnership Project (3GPP) [1] [2]. To replicate the 3G voice model, 3GPP and the GSM Association (GSMA) have previously specified a Voice over LTE (VoLTE) roaming architecture called Local Break Out (LBO) [3]. However, the latest demands of the operators, such as less CAPEX/OPEX and shorter time-to-market in order to compete against OTT's VoIP services, have re-opened the debate on VoLTE roaming architecture. In response, GSMA re-evaluated the VoLTE roaming architecture and a new architecture, S8 Home Routed (S8HR). As a result, GSMA endorsed S8HR as a new candidate for VoLTE roaming architecture [4].

2. Technical Characteristics of S8HR Architecture

The architecture of S8HR is shown in **Figure 1**, where IBCF/TrGW/BGCF/MGCF refers to the functions necessary for interconnect services. With S8HR, a VoLTE roaming service is provisioned on top of the LTE data roaming framework using a VoLTE capable terminal, as specified by GSMA PRD IR.92 [5]. The architecture has the following technical characteristics:

(1) Bearers for IMS services are established on the S8 reference point, just as LTE data roaming.
(2) All IMS nodes are located at Home Public Land Mobile Network (HPLMN), and all signaling and media traffic for the VoLTE roaming service go through HPLMN.
(3) IMS transactions are performed directly between the terminal and P-CSCF at HPLMN. Accordingly, Visited Public Land Mobile Net-

©2015 NTT DOCOMO, INC.
Copies of articles may be reproduced only for personal, noncommercial use, provided that the name NTT DOCOMO Technical Journal, the name(s) of the author(s), the title and date of the article appear in the copies.

Standardization

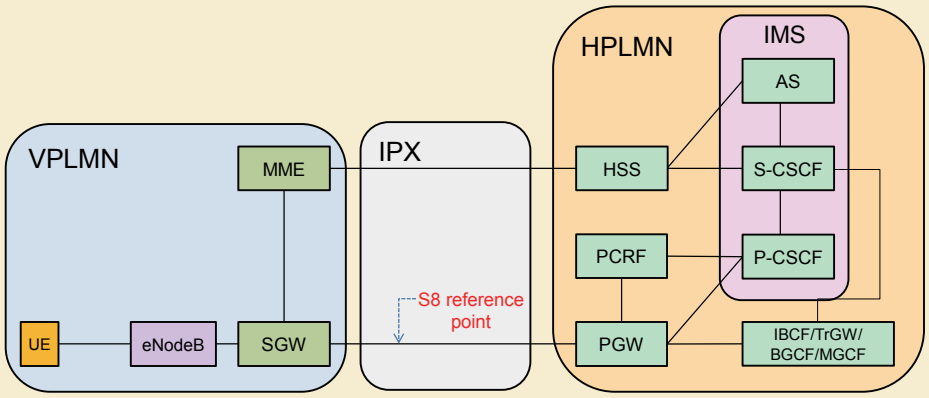


Figure 1 S8HR architecture

work (VPLMN) and interconnect networks (IPX/GRX) are not service-aware at the IMS level. The services can only be differentiated by APN or QoS levels.

These three technical features make it possible to provide all IMS services by HPLMN only and to minimize functional addition to VPLMN. As a result, S8HR shortens the time-to-market for VoLTE roaming services.

3. Basic Call Controls for S8HR

3.1 Attach Procedure

Figure 2 shows the attach procedure for S8HR VoLTE roaming. From Steps 1 to 3, there is no significant difference from the LTE data roaming attach procedure. In Step 4, HSS sends an update location answer message to MME. In order for the MME to select the PGW in HPLMN (Step 5), the MME must set the information element VPLMN Dynamic Address "Allowed," which is included in the subscribed data, to "Not Allowed." In Step 6, the bearer for SIP signaling is created between SGW and PGW with QCI=5. MME sends an attach accept message to the terminal with an IMS Voice over PS Session Support Indication information element, which indicates that VoLTE is supported. The information element is set on the basis of the MME's internal configuration specifying whether there is a VoLTE roaming agreement to use S8HR. If no agreement exists between two PLMNs, the information element will not be set.

3.2 IMS Call Control

After the attach procedure is completed, the terminal performs an IMS registration procedure. There is no difference between roaming and non-roaming VoLTE procedures, and all IMS services procedures, e.g., voice call, video call, SMS, or any kind of IMS service, are also the same as non-roaming procedures.

3.3 Emergency Call

According to 3GPP specifications, emergency calls must be connected

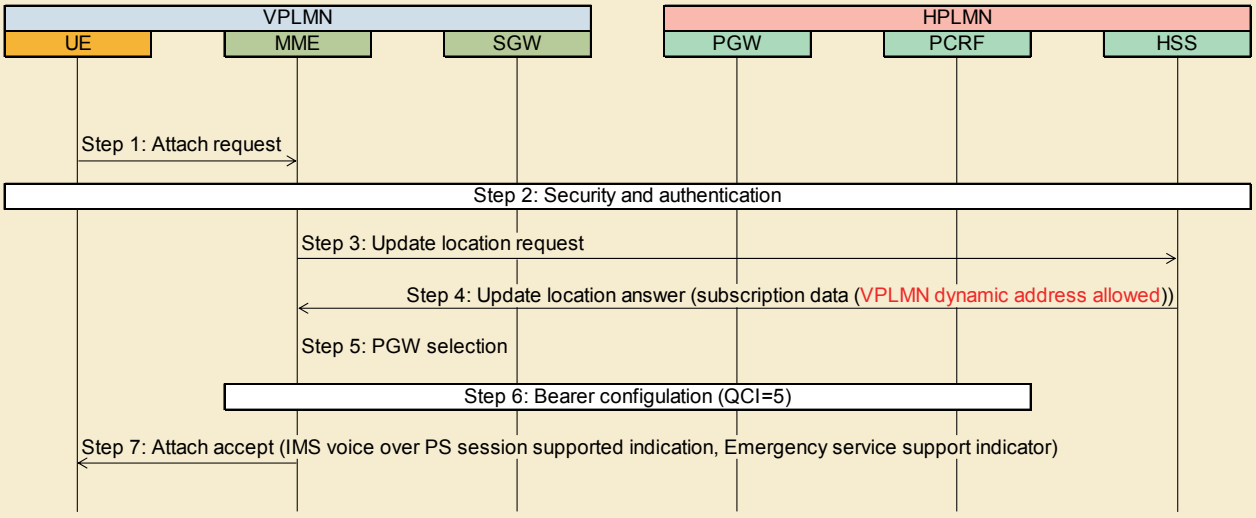


Figure 2 S8HR attach procedure

to the local emergency service. This means that the call routing procedure for emergency calls must be completed within VPLMN. To fulfill these regulations, S8HR must support at least one of the following methods:

(1) CS fallback

(2) IMS emergency call over LTE access

Which method to use is determined by the terminal on the basis of information about the capability of VPLMN, which is sent from MME to the terminal during the attach procedure. The terminal selects IMS emergency call with LTE network if it receives an attach accept message with an Emergency Service Support Indicator information element in Step 7. Otherwise, the terminal selects CS Fallback for emergency call via 2G/3G network, if it made a combined location registration to VPLMN.

When IMS emergency call is selected, the terminal is required to perform an emergency registration procedure for user authentication (**Figure 3**). For this procedure, the IMS layer connection between VPLMN's P-CSCF and HPLMN's S-CSCF is required to exchange authentication information. However, if the emergency registration fails for whatever reason, the terminal can continue the procedure by sending an INVITE message to VPLMN's P-CSCF with the information element "Anonymous," which indicates that S-CSCF could not authenticate the user [6].

4. Trials to Validate Voice Quality

As mentioned above, with S8HR, all IMS signaling and media go through HPLMN. To investigate the effect on delays and voice quality, NTT DOCOMO, Korea Telecom, and Verizon Wireless performed experiments in cooperation with GSMA [7]. The trials were conducted in an environment that replicates commercial networks. Results showed that VoLTE calls with S8HR have better quality

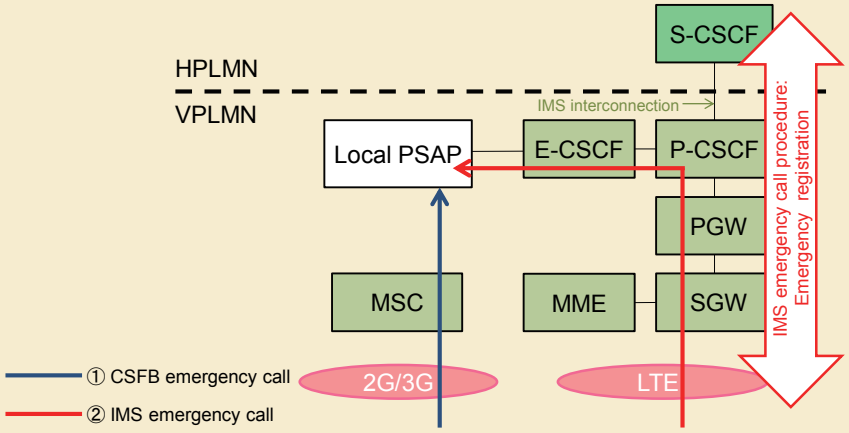


Figure 3 S8HR emergency call procedure

than 3G roaming voice calls, even for long-distance roaming. More participation from other operators will be expected, and individual results will be used for GSMA evaluation criteria to progress the discussion on S8HR.

5. Conclusion

This article outlined the characteristics of S8HR VoLTE roaming. At the time of publication, there is no official agreement as to which roaming architecture will be the standard for the mobile industry. GSMA and 3GPP will continue technical discussions on VoLTE roaming and many operators will also continue discussion on the basis of the trial results. NTT DOCOMO has contributed to both 3GPP and GSMA as chair to lead the discussion and will continue to drive the industry's discussion of VoLTE roaming and to develop related standards, aiming at the accelerated global deployment of VoLTE services.

References

[1] I. Tanaka, et.al: "Overview of GSMA VoLTE Profile," NTT DOCOMO Technical Journal, Vol.13, No.4, pp.45-51, Mar. 2012.

[2] K. Tokunaga, et.al: "VoLTE for Enhancing Voice Services," NTT DOCOMO Technical Journal, Vol.16, No.2, pp.4-22, Oct. 2014.

[3] I. Tanaka, et.al: "VoLTE Roaming and Interconnetion Standard Strategy," NTT DOCOMO Technical Journal, Vol.15, No.2, pp.37-41, Oct. 2013.

[4] 3GPP SP-150139: "LS on VoLTE Roaming Architecture," Feb. 2015.

[5] GSMA PRD IR.92: "IMS Profile for Voice and SMS," Apr. 2015.

[6] 3GPP TS23.167: "IP Multimedia Subsystem (IMS) emergency sessions," Jun. 2014.

[7] NTT DOCOMO Press Release: "DOCOMO Successfully Verifies VoLTE Roaming in Commercial Environment," Feb. 2015. https://www.nttdocomo.co.jp/english/info/media_center/pr/2015/0226_00.html

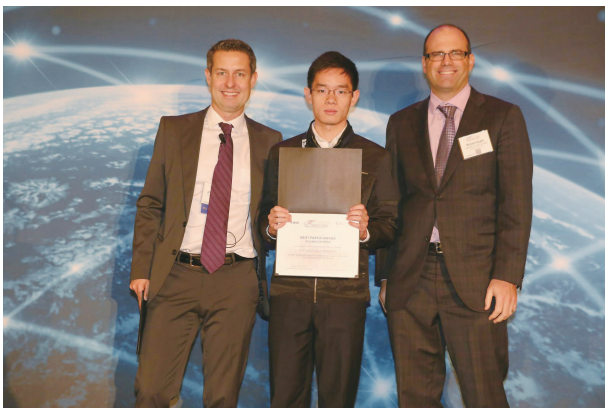
Received "Best Paper Award" in IEEE Globecom 2014

Hiroyuki Ishii† of DOCOMO Innovations, Inc., and Bo Yu and Liuqing Yang of Colorado State University received the Best Paper Award at the Institute of Electrical and Electronics Engineers Global Communications Conference 2014 (IEEE Globecom 2014) held in Austin, Texas (U.S.) from December 8-12, 2014.

IEEE GLOBECOM is one of two flagship conferences of the IEEE Communications Society (ComSoc), together with IEEE ICC. Each year the conference attracts about 3,000 submitted scientific papers [1], of which approximately 35% are accepted for publication. Among the published papers, 14 papers received the Best Paper Award for their outstanding performance.

The title of the award-winning paper was "3D Beamforming for Capacity Improvement in Macrocell-Assisted Small Cell Architecture." 5G, which is the next phase of mobile telecommunication standards beyond the current LTE/LTE-Advanced, is being studied all over the world for commercialization in the 2020's. 5G aims to achieve a 1,000-fold system capacity per km^2 and a 100-fold increase in user-experienced throughput compared to 2010. It is expected to significantly improve the above system capacity and user-experienced throughput by means of three approaches; namely, "spectrum efficiency enhancement by utilizing 3D Beam-forming/ Massive MIMO," "spectrum extension by utilizing frequency bands higher than 3 GHz," and "network densification by utilizing small cells."

This paper proposes the capacity enhancement for small cells under "*Phantom Cell*® *1 architecture" utilizing the flexible 3-dimensional (3D) Beam-forming*2 facilitated by the adoption of the Active Antenna System (AAS) at Base Stations (BSs). Phantom Cell is a macrocell-assisted small cell architecture proposed by DOCOMO. The aim of Phantom Cell architecture is to provide high system capacity and robust mobility while reducing the cell planning efforts. One of the key features for Phantom Cell architecture is the C-plane/U-plane Split configuration. C-plane is supported by macro cell layer to maintain good connectivity and mobility using lower existing frequency bands while the U-plane is supported by small cells to provide higher throughput and more flexible and energy efficient operations using higher frequency bands (e.g., 3.5 GHz band). With such configurations in Phantom Cells and the assistance from macrocells, conventional cellular network problems, such as coverage holes or handover failure, need not be considered. This enables more dynamic and flexible 3D Beam-forming using an extremely narrow beam, and thereby the received signal quality can be improved and the interference can also be controlled more effectively. The system level simulations demonstrate the significant gain of capacity enhancement with 3D Beamforming over the conventional sectorization with fixed



down-tilt in terms of both the cell average capacity (up to 124.8% gain) and the cell edge user throughput (up to 454.3% gain).

The simple configuration of "combining 3D Beam-forming and small cells on the basis of the Phantom Cell concept" has verified that a large capacity/user throughput gain can be obtained and has demonstrated the potential performance of 5G, which was evaluated and resulted in receiving the award.

Reference

[1] IEEE GLOBECOM 2014: "Welcome to IEEE 2014."
http://globecom2014.ieee-globecom.org/about.html

† Currently Service Innovation Department

*1 **Phantom Cell™:** A trademark of NTT DOCOMO, INC.

*2 **3D Beam-forming:** Beam-forming is a signal processing technique used to control the directionality of the transmission and reception of radio signals. The improvement compared with omnidirectional reception/transmission is known as receive/transmit gain (or loss). 3D Beam-forming is a technology that dynamically performs beam-forming in both horizontal and vertical domains.

NTT DOCOMO
Technical Journal Vol.17 No.1

Editorship and Publication
NTT DOCOMO Technical Journal is a quarterly journal edited by NTT DOCOMO, INC. and published by The Telecommunications Association.

Editorial Correspondence
NTT DOCOMO Technical Journal Editorial Office
R&D Strategy Department
NTT DOCOMO, INC.
Sanno Park Tower
2-11-1, Nagata-cho, Chiyoda-ku, Tokyo 100-6150, Japan
e-mail: dtj@nttdocomo.com

Copyright
©2015 NTT DOCOMO, INC.
Copies of articles may be reproduced only for personal, noncommercial use, provided that the name NTT DOCOMO Technical Journal, the name(s) of the author(s), the title and date of the article appear in the copies.