

# Technology Reports

## New FOMA Functions for 2009 — Evolving Web Applications —

*With the mobile terminal now an essential tool in everyday life and business, there is increasing demand to expand usage scenarios and provide more convenient and pleasant functions. In response to this need, NTT DOCOMO has developed an enhanced browser, an enhanced battery level indicator, a terminal-settings one-step backup function, individual charging function for i-appli content, and a hybrid speech recognition function for its 2009 summer models.*

Communication Device Development Department

*Satoru Morota*

*Naoto Okubo*

*Toshiki Takatsu*

*Hiroki Omoto*

*Hiroataka Furukawa*

### 1. Introduction

The mobile terminal has become an essential tool in everyday life and business. As a mobile communications operator, NTT DOCOMO recognizes that more convenient and pleasant functions need to be provided and that mobile-terminal usage scenarios need to be expanded by adding services that are more advanced and functional than those of previous models. Against this background, an “inline movie display function” for movie playback has been achieved within the i-mode browser as a browser enhancement for NTT DOCOMO’s 2009 summer models (**Photo 1**).

User-support functions have also

been enhanced in the form of an “enhanced battery level indicator” that displays remaining battery power in a more precise manner and a “terminal-settings one-step backup function” that eliminates worries when upgrading to a new terminal. Plus, by taking advantage of the fact that a mobile terminal is

owned and used by an individual, the existing “speech recognition function” has been expanded into a “hybrid speech recognition function” that can support personal names and proper nouns specific to individual users thereby enhancing the input interface for composing e-mail.



In addition to enhancing functions for individual users, it is also important to expand mobile-terminal usage scenarios so as to expand business opportunities for Content Providers (CPs). Traditionally, charging for individual pieces of content could only be per-

formed when downloading content by the i-mode browser, but the newly developed “individual charging function for i-appli content” makes it possible to charge for individual content even when downloading it from an i-appli.

This article described these five new functions that have been implemented in 2009 summer models divided into four conceptual series (STYLE, PRIME, SMART and PRO).

## 1. Enhanced Browser

### 1.1 Background

In 1999, mobile terminals were equipped with an i-mode browser function that enabled users to view content created for mobile terminals, and in 2005, they were equipped with the i-mode full browser that enables users to view content created for PCs. The 2009 summer models feature significant improvements to the i-mode browser function that enables rich content approximating PC content to be viewed even with the i-mode browser.

### 1.2 Enhancement of the i-mode Browser

The following functional improvements were made to the i-mode browser:

- Enlargement of content size
 

Content size has been expanded from 100 kB to 500 kB enabling users to view rich and information-heavy content that was difficult to view in the past.
- Screen capture function
 

This function enables the

results of executing dynamic content to be saved as an image. The existing screen memo function allows content itself to be saved but not the results of executing dynamic content. This function makes it possible to review those execution results the next time the browser is used without having to perform any data communications or content operations.

- Support of full browser functions
 

Elements that can be used with HTML tags have been added, support for Cascading Style Sheets (CSS)<sup>\*1</sup> and Flash<sup>®\*2</sup> Video has been provided, and viewing of rich content optimized for mobile terminals and on par with PC-oriented content has been enabled.
- Function promoting i-mode-browser operability (suppression of horizontal scroll)
 

Although the i-mode browser is now approaching the full browser in functionality, a layout adjustment

has been made that deems horizontal scrolling unnecessary from the viewpoint of preserving i-mode browser operability. In this adjustment, content in its entirety can be viewed by simply scrolling up or down. This adjusted content layout is shown in **Figure 1**.

These functions have enabled the provision of inline movie display on the i-mode browser. Specifically, they enable Flash Video to be laid out within a page and for a movie to be played back while text information described in HTML is displayed.

### 1.3 Enhanced Linkage between i-mode Browser and Full Browser

Users can now move seamlessly between PC-oriented content and mobile-terminal-oriented content. In particular, the use of the i-mode browser and full browser has been made more convenient by allowing for seamless

\*1 **CSS**: Specifications for describing how a Web page should be displayed.

\*2 **Flash**<sup>®</sup>: A trademark or registered trademark of Adobe Systems Inc. in the United States and other countries.

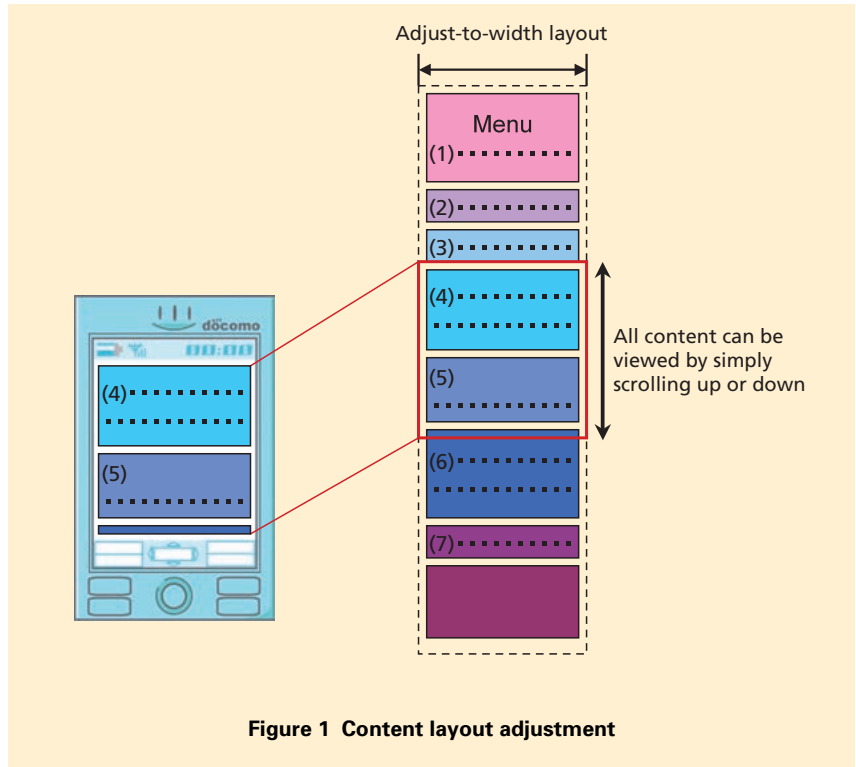
switching between them. This can be useful when a search performed on the full browser returns i-mode-oriented content as a search result, or when searching on the full browser is once again desired after i-mode-oriented content has been displayed.

- Support of ib attribute for switching from full browser to i-mode browser

Although the ifb attribute for switching from the i-mode browser to the full browser has been supported in the past, the 2009 summer models also support a function for switching from the full browser to the i-mode browser by selecting a link to which the ib attribute has been affixed. An example of describing the ib attribute is shown in **Figure 2**.

- Support of cache saving when switching between the i-mode browser and full browser

In the past, the i-mode-browser cache was deleted when switching from the i-mode browser to the full browser. Thus, when returning to the i-mode browser, it was necessary to obtain the content again by data communications. To deal with this issue, a history operation<sup>\*3</sup> has been implemented for saving the cache after switching thereby enabling content to be displayed



**Figure 1 Content layout adjustment**

Description for switching from full browser to i-mode browser and opening <http://yyy/yyy.htm>

```
<a href="http://xx.xx" ib="http://yyy/yyy.htm">
link</a>
```

**Figure 2 Example of describing the ib attribute**

when going back and forth between browsers without having to resort to data communications. This operation is shown in **Figure 3**.

### 1.4 Current Issues and Future Developments

The use of browsers on mobile terminals has been dramatically

improved by enhancing the functions on the i-mode browser and enhancing the linkage between the i-mode browser and full browser. Looking forward, we plan to make more functional enhancements to provide users with even greater convenience.

<sup>\*3</sup> **History operation:** A function for saving browser viewing history and for reviewing pages from that history.

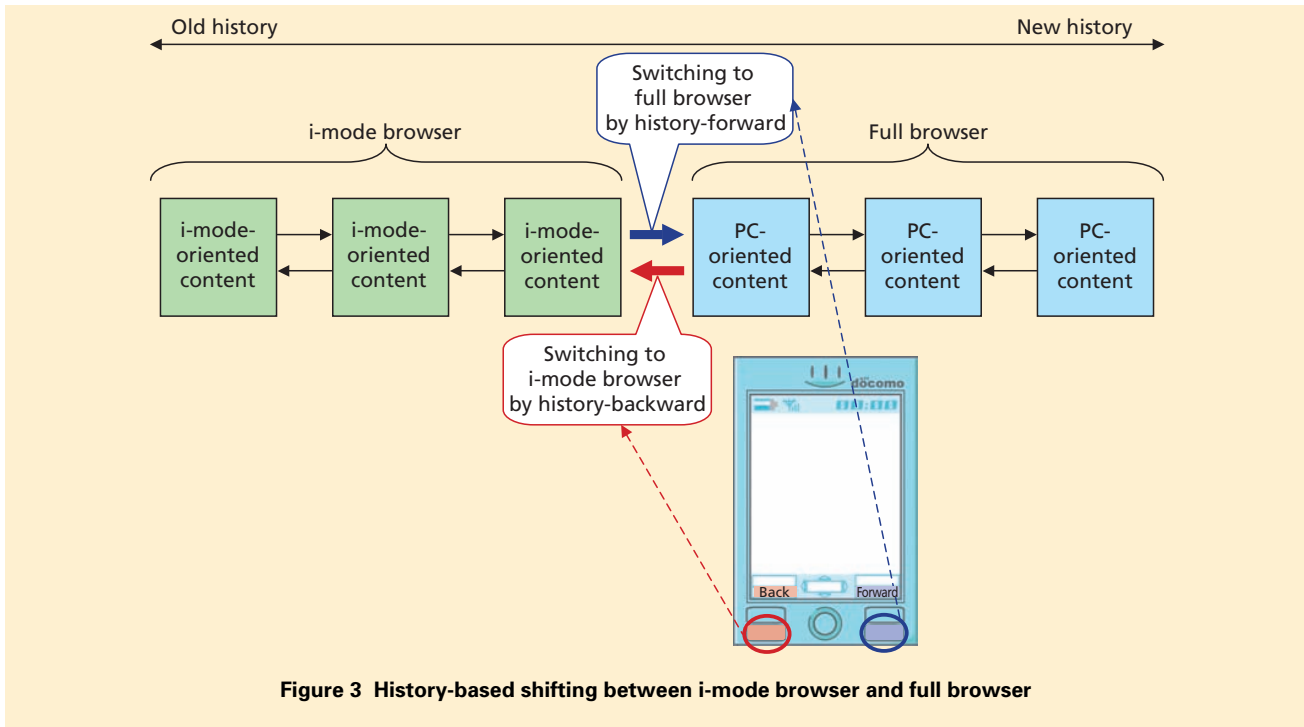


Figure 3 History-based shifting between i-mode browser and full browser

## 2. Enhanced Battery Level Indicator

### 2.1 Background

The battery level indicator in mobile terminals has been using a three-level icon still before the introduction of FOMA terminals. This indicator has been upgraded in the 2009 summer models to provide up to 100 levels that can aid the user in making a more informed decision on how best to use the mobile terminal given the amount of battery power remaining.

In the conventional three-level battery icon (Figure 4 (a)), the display of three bars represents a higher battery level and therefore a longer usage peri-

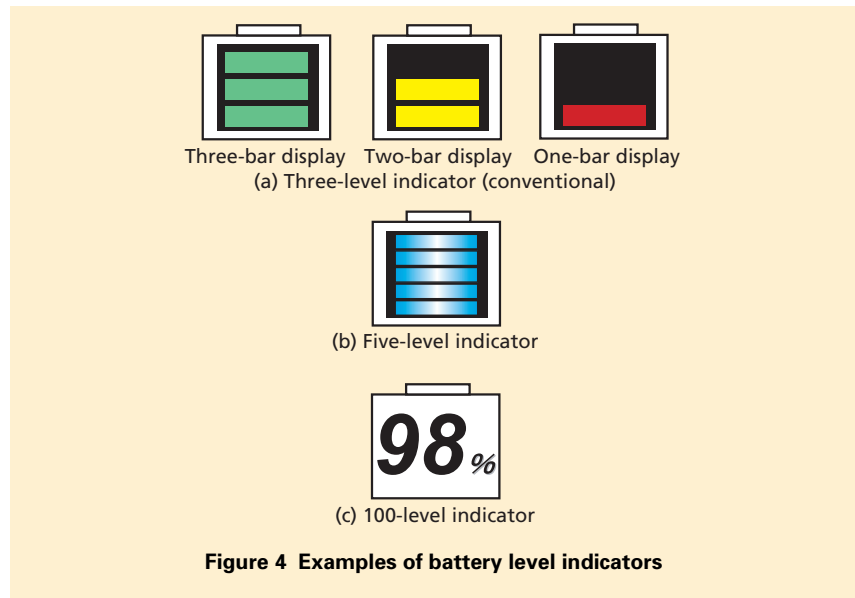


Figure 4 Examples of battery level indicators

od than the two-bar or one-bar displays. But even in the case of the three-bar display, the user is left to estimate whether the battery is near a fully charged state or about to switch to the two-bar display. For users that worry about reducing battery level and feel compelled to reduce power consumption, this implementation of a battery level indicator tends to curtail the use of services. Furthermore, when faced with no opportunity for charging such as when going out from one's home or office, a three-bar display in the conventional three-level battery icon makes it difficult to decide whether or not to charge up the terminal before going out. In any of the above cases, the user is left to make a decision, which is hardly desirable from the viewpoint of convenience and comfort. To eliminate this anxiety in using a terminal, the enhanced battery level indicator informs the user of the remaining amount of battery power in a more precise manner. This provides the user with an environment that makes it easy to judge whether services should be used or whether it's time for charging. Moreover, to provide even more convenience, the battery level is also displayed during the charging process.

## 2.2 Precise Battery Level Indicator

The conventional three-level indica-

tor has been upgraded to a more detailed indicator ranging from a minimum of five levels (Fig. 4(b)) to a maximum of 100 levels (Fig. 4(c)). In the conventional indicator, the percentage of battery power represented by each of the three bars was not uniform, but in the new specification, the total amount of battery power is divided equally among all levels. In other words, the five-level indicator is divided into five levels each representing 20% of battery power, and the 100-level indicator is divided into 100 levels each representing 1% of battery power. This equal division of battery power enables the user to determine the remaining amount of battery power as it truly is. There's no longer any need for the user to infer the imbalance in battery power indicated by each bar based on personal experience.

When using the conventional method of measuring battery level from terminal voltage, it has been difficult to display battery level accurately even when exceeding three levels as displayed values would tend to fluctuate continuously. However, by using a dedicated IC that can measure battery capacity or remaining power accurately from the battery voltage and integrated value of charge-discharge current as in battery level indicators on PCs, we have been able to achieve a high-accuracy

battery level indicator up to 100 levels.

## 2.3 Battery Level Display during Charging

In the conventional battery level indicator function, the battery icon was displayed in flashing mode during charging. At this time, however, the battery icon did not display the amount of remaining power, which meant that the user would either have to interrupt the charging to determine the battery level or simply wait for charging to complete. The enhanced function, however, displays the amount of remaining power on the battery icon even during charging so that the user can determine how far charging has progressed at any time. This should make it easier for the user to decide when to suspend charging when there's not enough time for a full charge.

## 2.4 Current Issues and Future Developments

Achieving a more precise battery level indicator and enabling the display of remaining battery power during charging have made the use of mobile terminals even more convenient for users. In future study, we plan to make the display of battery-related information even more convenient and comfort.

### 3. Terminal-settings One-step Backup Function

#### 3.1 Background

As mobile terminals become more sophisticated, the number of function-related items that a user needs to set to use a mobile terminal is increasing. Such settings of various types have had to be manually input when changing terminals, and this has been one factor behind the anxiety that users have felt when upgrading to a new model. For this reason, users have been calling for one-step transfer of time-consuming terminal function settings in the same way that phonebooks have been transferred when changing terminals.

In response to this need, a “terminal-settings one-step backup function” has been developed for 2009 summer models. This function makes it possible to

transfer the values of commonly used and normally time-consuming function settings at the time of a terminal upgrade.

Specifically, this function combines function-setting data set in the mobile terminal with the data targeted for backup in the “user memory one-step backup” function[1] developed for the 2007 autumn/winter models. Thus, the user only has to select “user memory one-step backup” from the menu to transfer terminal function settings together with Personal Information Manager (PIM) data<sup>\*4</sup>.

#### 3.2 Data Format for Function Settings Transfer

The manner in which function set-

tings are stored in current mobile terminals can differ from one terminal to another. Consequently, if such settings are simply output in the way that they are stored, a correct transfer can only be achieved between terminals of the same models or between certain types of terminals. However, if the proposed function is to have any value, it must be able to transfer terminal settings between different models of terminals without restriction. To this end, we established a format specification to be used in common by different manufacturers. The backup and restoration of function-setting data in a mobile terminal using this format is shown in **Figure 5**.

A basic mechanism for describing function settings has been specified in

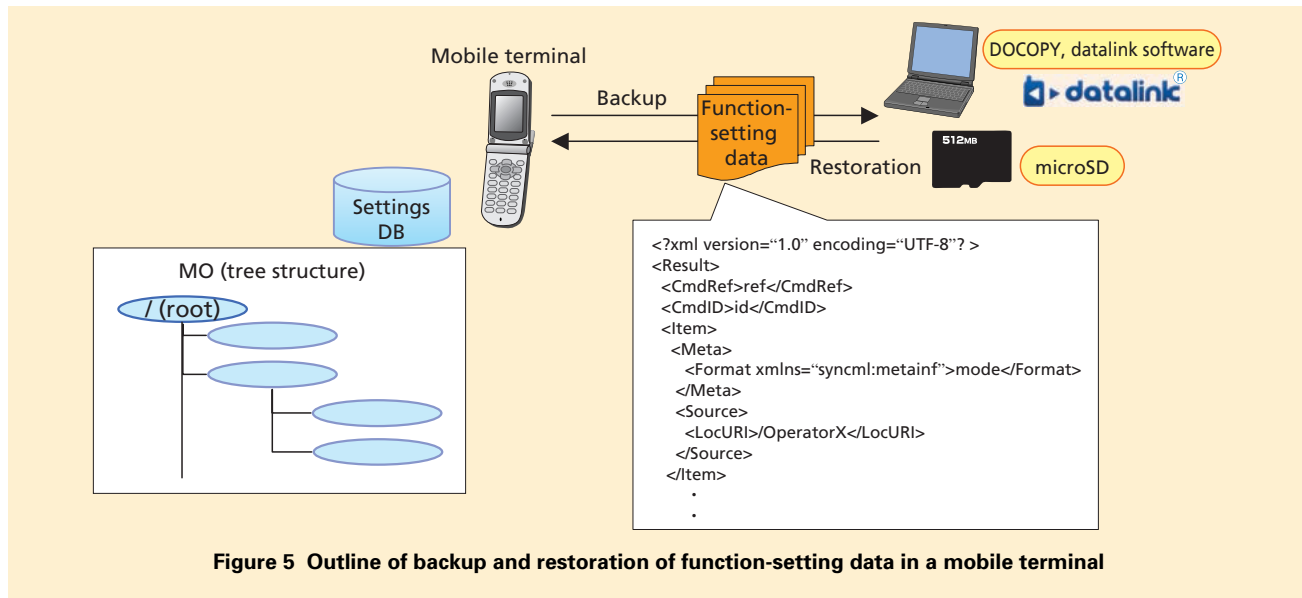


Figure 5 Outline of backup and restoration of function-setting data in a mobile terminal

\*4 PIM data: In this article, five types of data (phonebook, bookmark, mail, schedule, text-based memo).

Open Mobile Alliance (OMA)<sup>\*5</sup> - Device Management (DM)<sup>\*6</sup> - Tree and Description (TND)<sup>\*7</sup>, a standard for device management. A feature of this standard is its hierarchical description of device settings using the Extensible Markup Language (XML) format. A function setting described in XML format is called a Management Object (MO).

Taking into account the need for function extendibility in the future, we have adopted the OMA-DM-TND description method in the newly developed terminal-settings one-step backup function and have specified the description format for terminal settings targeted for backup.

In the 2009 summer models, data targeted for backup consist of mail function settings, calling/receiving function settings, alarm function settings, and text-input function settings. The data items in question are therefore subject to MO specifications.

At the time of backup, the function outputs XML data that incorporates the values of user settings in conjunction with MO specifications, and at the time of data restoration, the function reads in those user settings from the XML data and reflects them in the mobile terminal targeted for data transfer.

### 3.3 Input/output Paths of Function-setting Data

Function-setting data is output and input to and from a microSD by the

user data one-step backup function or output and input via software such as DOCOPY<sup>\*8</sup> or DOCOMO keitai datalink<sup>\*9</sup> via USB communications. This approach takes into consideration phone use by users who are familiar with mobile terminals and who have many customized settings by the time they change terminals as well as phone use by users who are unfamiliar with mobile-terminal operations and feel that making settings is always troublesome.

The reason for selecting these input/output paths lies in the fact that function-setting data can be restored together with PIM data. For example, in settings dealing with the accepting or rejecting of incoming calls corresponding to specific phone numbers, we consider the case in which the phone numbers specified as accept or reject by the user are inconsistent with the actual phonebook in the mobile terminal. In this case, an incoming call from a telephone number specified as “accept” by the user may be rejected, or an incoming call from a telephone number specified as “reject” by the user may be accepted, which can present a major issue to the user. To prevent such issue, specified telephone numbers are cross-referenced with telephone numbers registered in the PIM phonebook just before settings data and PIM data are restored together. This prevents erroneous settings from being made.

### 3.4 Display of Successfully Restored Items

To enable users to understand the outcome of a transfer when using this function to restore data that was previously backed up, a list of items that were successfully restored will be displayed on the mobile terminal’s screen.

This list of successfully restored items can be used to check whether certain function settings were correctly transferred. In addition, for data items having multiple records as in the history of calls made and received, the number of records restored will be displayed.

### 3.5 Current Issues and Future Developments

One issue to be taken up in future study is the expansion of function settings targeted for transfer. In the models presented here, the function settings targeted for transfer are mainly those common to all terminal manufacturers, which represents a relatively small number of items compared to all function settings in mobile terminals. To make the use of mobile terminals even more convenient for users, it is naturally desirable to extend function settings targeted for transfer to even items not common to all manufacturers. At the same time, some type of mechanism is needed for informing the user in an easy-to-understand manner about function settings that cannot normally be transferred.

We also consider that part of this

\*5 **OMA:** An industry standardization organization that aims to standardize service and application technology and achieve interoperability in mobile communications.

\*6 **OMA-DM:** OMA standard specifications for remote control of mobile devices.

\*7 **OMA-DM-TND:** OMA standard specifications for describing device information on mobile terminals.

\*8 **DOCOPY:** A terminal set up in NTT DOCOMO shops for backup purposes. It can be used to back up various kinds of data to a CD-ROM or

other media as well as to copy memory to a terminal or use a service like iC Ohikkoshi.

terminal-settings one-step backup function could be used to create a related service. For example, the formatted data created by this function for transferring function settings could be stored on a server by a “data storage service” to ensure that backup data exists even if

a terminal should be lost. However, to achieve such a service, it must be made possible to backup and restore functions settings without problem even if such backup data should become detached from PIM data. We can also envision the transmission of OMA-DM com-

mands from a server with respect to MOs created by this function to develop a service that remotely updates function settings for users unfamiliar with terminal operations.

## 4. Individual Charging Function for i-appli Content

### 4.1 Background

A function for charging for individual content downloaded via the i-mode browser (individual charging function) has already been introduced, but it has not been possible to perform charge processing for individual content obtained via i-appli communications within the i-appli itself. That is, for cases when charge processing was required when using an i-appli, it has been necessary to link to and start up the i-mode browser from the i-appli, have the i-mode browser perform charge processing, and then return to the i-appli. This has been inconvenient for both users and CPs not only because it breaks up operations on the i-appli but also because it is difficult to create a sense of uniformity in screen layout between the browser and i-appli. In response to this issue, we developed a charging function for individual data communications performed by i-applis to enable both content acquisition and charge processing within the i-appli

itself.

### 4.2 Function Outline

The individual charging function for i-appli content consists of the following three processes (**Figure 6**).

#### 1) Charge Preprocessing

In this process, the function checks whether the data to be obtained can actually be subjected to charge processing correctly and performs authentication processing with respect to the user on the basis of an i-mode password. If no problems are encountered during preprocessing, the function displays a screen asking whether the user would like data acquisition to begin.

#### 2) Data Acquisition Processing

Data acquisition begins if the user so agrees. Considering that data acquired via an i-appli is not restricted to content with a specific format such as a video file and may be of various types such as point-card and game items, map data, etc., no processing is performed here that is dependent on the

content of acquired data.

#### 3) Charge Processing

Charge processing begins once the server that performs charge processing is informed that all data has been downloaded successfully. On completion, the server informs the mobile terminal that charging has been completed. On receiving this notification of charge completion, the i-appli is now able to use the data in question as all charge-related processing is completed.

### 4.3 Examples of Using the Individual Charging Function for i-appli Content

As described above, data desired by a user and targeted for charge processing can be obtained during the data acquisition process described in step 2) above. However, another format can be considered in which the individual charging function is applied before any processing is performed by the i-appli. For example, full music tracks that cannot be obtained by the data acquisition

\*9 **DOCOMO keitai datalink**: Software that allows various types of mobile-terminal data to be viewed and edited on a PC after connecting the PC to the mobile terminal by USB cable. The datalink logo is a registered trademark of NTT DOCOMO.



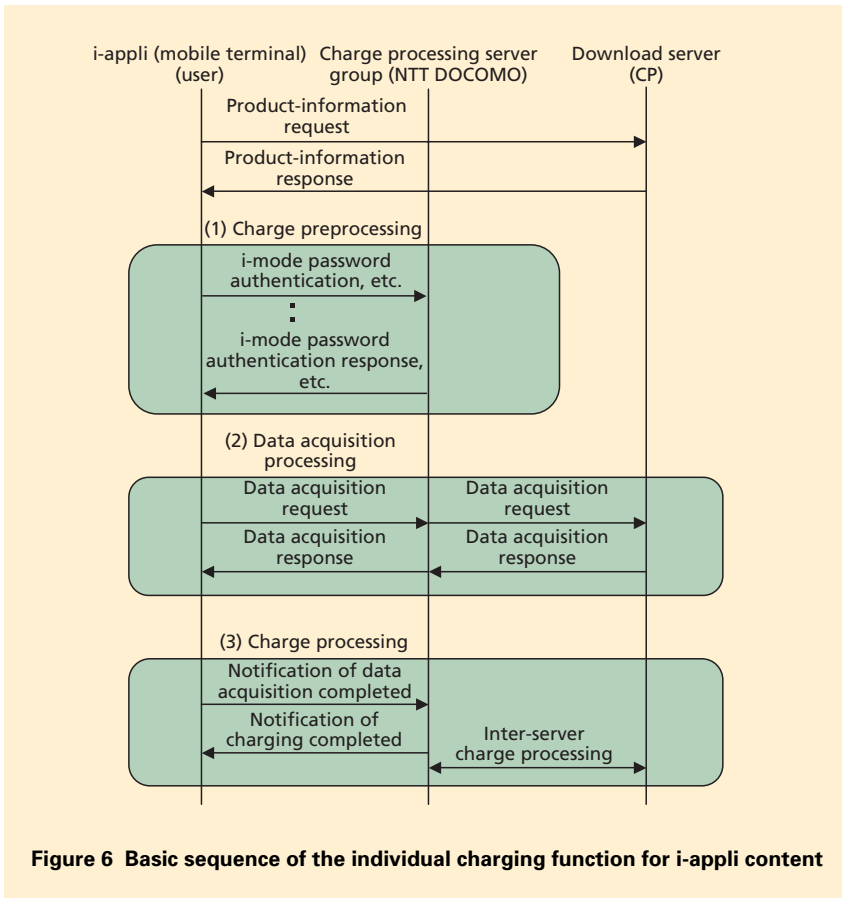


Figure 6 Basic sequence of the individual charging function for i-appli content

process, items within games, and the use of games themselves could be targeted for such a charging process, as illustrated by the following examples.

- i-appli for full-music-track purchasing

For its 2009 summer models, NTT DOCOMO has added a full-music-track download function together with the individual charging function for i-appli use. With these functions, an i-appli could be created that treats a URL pointing to a music track as a target of charge processing and obtains that URL in the data acquisition

process, and then, once charge processing is completed, obtains that music track by the full-music-track download function. In other words, charge processing and the acquisition of purchased music can be completed by a sequence of processes in the same i-appli. Such a scheme would make possible the purchasing and downloading of music tracks from an i-appli that introduces new music tracks or recommends existing ones.

- Use in game applications

When purchasing an item within a game, this function

enables the user to download the image data for that item in the data acquisition process and to be charged accordingly, or to make a settlement when advancing to the next stage even if there is no need to download specific data, all without having to momentarily stop the game.

This function could also be applied to services provided via i-applis that currently collect usage fees on a monthly basis. This would enable charges to be applied whenever the service is used and therefore provide another charging option that would provide more convenience for users and more business opportunities for CPs.

#### 4.4 Current Issues and Future Developments

The full-music-track downloading function was presented above as one example of a function combined with individual charge processing, but the capability of purchasing diverse content from i-applis should increase the types of content available from CPs. Development work is progressing on facilitating the download of large-capacity i-motion video clips (maximum capacity: 10 MB) that are presently difficult to access via i-applis due to limitations in the amount of memory and communication resources that i-applis can use. There is also no need to restrict com-

bined functions to the downloading of content—they can also be used to develop i-applis with even greater functional-

ity and therefore provide a more convenient mobile service environment for both CPs and users while expanding

business opportunities.

## 5. Hybrid Speech Recognition Function

### 5.1 Background

A speech recognition function with a large vocabulary is difficult to achieve using only the processing available on a mobile terminal. For this reason, NTT DOCOMO developed a Distributed Speech Recognition (DSR)<sup>\*10</sup> function that links the mobile terminal with a server and introduced it for i-appli use beginning with the FOMA 905i series. This function has found use, for example, in a Japanese-English translation i-appli for travelers [2]. In addition, the Raku-Raku Phone series beginning with Raku-Raku Premium provide a “voice input mail” service based on the FOMA 905i mechanism to enable users to input text by voice when composing e-mails.

However, the DSR service uses a common recognition dictionary on the server, and as such, it cannot necessarily realize optimal speech recognition for the individual user. In particular, it is difficult for a common recognition dictionary to register proper nouns such as personal names which, in Japanese, often have special readings. Such proper nouns become unknown words<sup>\*11</sup> that cannot be included in recognition

results and that contribute to the issue of recognition errors. Thus, to achieve voice input in conformance with the user’s personal information, it would be useful to supplement the common dictionary with words that are registered in a user dictionary prepared for each user. Such a combined dictionary approach should improve recognition accuracy for each user.

With this in mind, a “hybrid speech recognition function” has been developed for the 2009 summer models with the aim of providing speech recognition services appropriate for each user. This function incorporates a sub-function for extracting unknown words in the server-side recognition engine and interacts with Local Speech Recognition (LSR) in the mobile terminal to recognize unknown words using a user dictionary.

### 5.2 Recognition of Unknown Words

The key feature of hybrid speech recognition is the combined use of DSR and LSR speech recognition processing so that recognition of speech judged to be unknown by DSR can be recognized by LSR.

In general, the speech recognition process uses a language model based on phoneme<sup>\*12</sup> sequences that can be referred to dictionary data to identify words or phrases. Thus, by combining this model with a model for identifying unknown words in the Back End (BE) portion of DSR, it becomes possible to return information on unknown words in recognition results. Time information on each phrase (elapsed time from the beginning of an utterance) can also be obtained in the DSR BE and returned to the mobile terminal with recognition results.

After receiving the recognition results from the DSR BE, the mobile terminal examines the unknown-words information and phrase-time information included in those results. It then extracts speech features in the voice interval corresponding to the unknown words and passes them to the LSR BE that incorporates an unknown-words dictionary within the mobile terminal.

The recognition dictionary used by the LSR BE can be prepared from the phonebook and sent/received e-mail in the mobile terminal making it unnecessary for the user to create a dictionary

\*10 **DSR**: A speech recognition scheme in which the mobile terminal performs feature extraction from input speech and the server performs recognition and conversion processing from those speech features.

\*11 **Unknown words**: In this article, words not

included in speech recognition dictionaries, proper nouns in particular.

\*12 **Phoneme**: A minimal unit of sound used to discriminate meaning in a given language.

by manual input. Recognition dictionaries may also be switched according to the upper-level application enabling the most suitable dictionary to be used for the current application.

### 5.3 Flow of Speech Recognition Processing

Taking voice input including unknown words as an example, the basic flow of voice-input processing using hybrid speech recognition is shown in **Figure 7**.

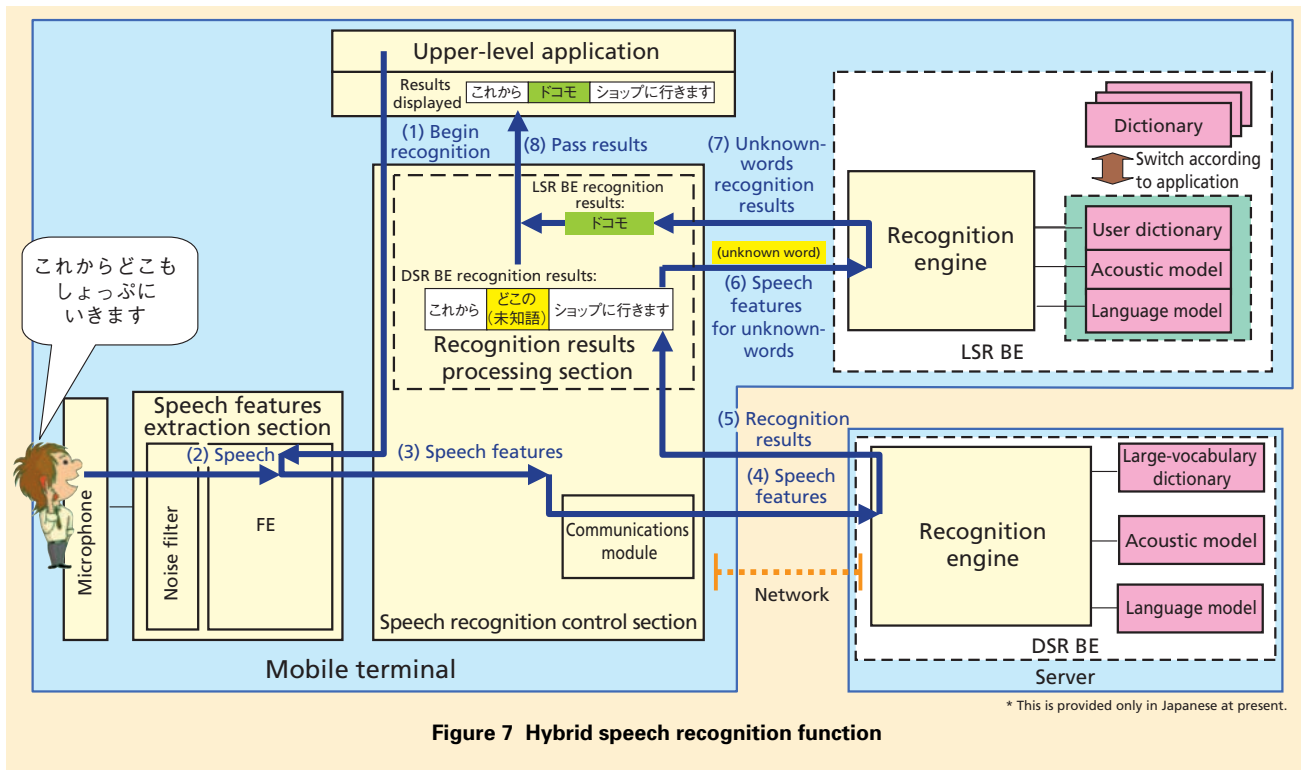
An upper-level application that uses voice input activates the voice-input function and speech recognition begins (Fig. 7 (1)). The speech recognition control section then notifies the Front End (FE) that speech recognition has

begun, and the receiving of voice input from the microphone begins (Fig. 7 (2)). Next, FE converts speech input from the microphone to speech features (Fig. 7(3)) and the speech recognition control section forwards these speech features to DSR BE by controlling the communications module (Fig. 7 (4)). The speech recognition control section then receives the results of recognition from DSR BE, and the recognition results processing section checks to see whether unknown words are included in those results (Fig. 7 (5)). In the event that unknown words are included in the recognition results received from DSR BE, the speech features corresponding to the unknown-words are passed to LSR BE for recognition processing

(Fig. 7(6)). The recognition results from LSR BE are then returned to the recognition results processing section in the same way as the results from DSR BE (Fig. 7 (7)). Finally, the recognition results processing section compares the results from DSR BE with those from LSR BE, deletes any duplicate candidates in recognition results, adjusts the display sequence taking the candidates from LSR BE as top candidates in the priority given, and passes the final results to the upper-level application (Fig. 7 (8)).

### 5.4 Current Issues and Future Developments

To deal with words specific to individual users that cannot be easily



**Figure 7 Hybrid speech recognition function**

recognized by a common recognition dictionary on a server, the common dictionary is linked with a user dictionary on the mobile terminal. This joint use of a common dictionary and a user dictionary optimized to the user and the upper-level application improves recognition accuracy. However, as words that can be used as speech recognition candidates increase, the probability of erroneous recognition due to similar words or homonyms likewise increases, and the issue arises as to how best to

present to the user the recognition candidates for the intended word.

In future research, we will work to improve recognition accuracy further by inferring recognition candidates based on current user context or infer the most suitable word from preceding and subsequent words. We can also envision an application that, instead of returning recognition results as text, links the results with a Web service. Such an application could expand the usage scenarios of voice input and pro-

vide users with even greater convenience.

#### REFERENCES

- [1] H. Suzuki et. al: "Application Functions for Autumn/Winter 2008 Models (1) i-concier and One-step User Memory Back-up Functions," NTT DOCOMO Technical Journal, Vol.10, No.4, pp.45-50, Mar. 2009.
- [2] T. Hamada et. al: "FOMA 905i Application Functions," NTT DoCoMo Technical Journal, Vol.9, No.4, pp.11-16, Mar. 2008.