

FOMA 905i Application Functions

*An automatic software update function, an enhanced Flash^{®*1} function, a music player usage history transmission function, and a distributed speech recognition function have been developed as new applications for the FOMA 905i series.*

*Tetsuya Hamada, Akiko Tobe,
Akihiro Ichinose
and Naomi Torimoto*

1. Introduction

Mobile terminals must ensure the use with greater security and safety than previously possible, while providing a readily available service of advanced functions. For these reasons, the functionality of the FOMA 905i series has been expanded by development of an automatic software update function for higher safety and security in use, and enhanced Flash and distributed speech recognition functions for a readily usable service of advanced functions.

A music player usage history transmission function to send the histories of music downloaded by a user under the user agreement with the music distribution service provider (Content Provider: CP) has also been developed for marketing purposes.

This article gives an overview of these four functions.

2. Automatic Software Update Function

The automatic software update function of the FOMA 905i series is described below.

2.1 Background

The software update function allows the user to update the software of the mobile terminal as required.

Recent mobile terminal software development is often associated with a range of measures to improve quality to eliminate the possibility of defects (e.g., improvements in the software development process), however, issues do appear following market release due to the increasing scale of development required by addition of new functions.

Within this context, a wireless software update function to provide the user with software updates has been introduced in the FOMA 900i series. The FOMA 904i series supported an update notification function via a screen

icon to notify users of the need for such updates, however users were required to perform the updates manually. The automatic software update function eliminates this inconvenience.

2.2 Issues with Introduction and Solutions

The two issues encountered in introducing the automatic software update function, and the solutions adopted, are described below.

Issue 1: No mobile terminal functions are available while rewriting the software.

It is therefore desirable that the automatic software update function be run while the mobile terminal is not being used, which differ with each user. On the other hand, it is necessary to disperse the downloading of update files to prevent network traffic congestion, and thus reduce load on the network.

The processes of downloading update files and rewriting software are therefore separated, with the start of

*1 **Flash**[®]: Software for creating content incorporating a combination of voice and vector graphics animation, or the created content itself. Flash is a trademark or registered trademark of Adobe Systems Inc. in the United States and other countries.

downloading under network control and the start of software rewrite being determined by the mobile terminal user.

Issue 2: Software rewrite fails if the battery is removed during the process.

Since this function automatically updates the user's mobile terminal software, it is essential that the mobile terminal software rewrite process does not fail.

A fail-safe function has therefore been added to ensure that, even if the battery is removed during software rewrite, the operation may be resumed once power is applied again.

2.3 Automatic Update Procedure

Figure 1 shows an example of the automatic update setup screen. The user can select "Auto-update" for automatic update, "Only notification" for notification only without automatic update, or "Disable" for no update or notification. The user can also change the day and time at which to start the update. "Auto-update" is set as a default to ensure that

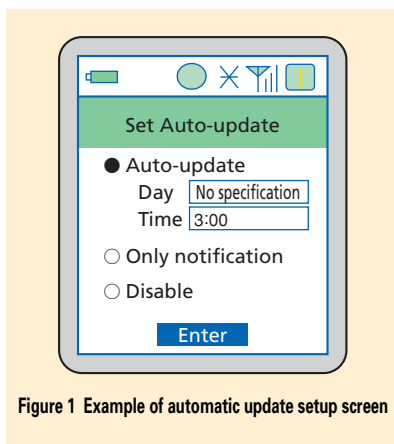


Figure 1 Example of automatic update setup screen

as many users as possible may use the function.

The automatic update is executed by the following three steps:

- 1) Reserve download
- 2) Download update file
- 3) Rewrite software

The mobile terminal receiving a signal from the software update server determines whether update is necessary, and makes reservation only when an update is necessary. Making a reservation involves determining the download time from the server, and acquiring the time. This function allows the dispersion of traffic to the server.

The mobile terminal starts the software update function automatically on the reserved day and time, and downloads the update file. When download is completed, the icon notifying the start time for software rewrite is displayed on the standby screen.

The mobile terminal then starts software rewrite automatically at the time specified on the automatic update setup screen. If software rewrite cannot be started (for example, due to another function in operation), it will begin at the next opportunity (defaults to 3 a.m. the next day). This operation is repeated until software rewrite is completed.

3. Enhanced Flash Function

3.1 Background

Flash is a function succeeded from

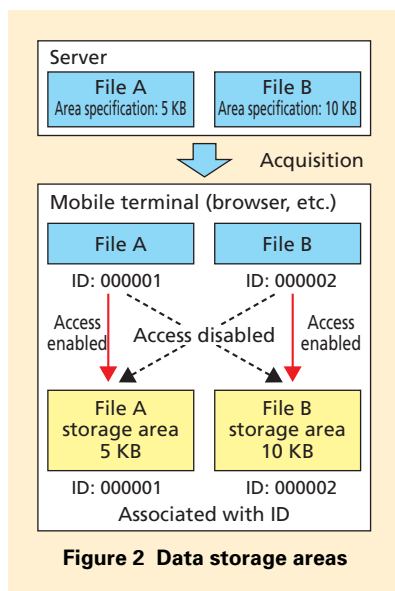
the mova 505i series for use over a diverse range of applications. This function is displayed on the i-mode browser, used in the i-channel service, and is available on standby screens and menus. However, there were certain restrictions, for example, the system did not return to the immediately previous focus position upon returning to the previous page, and it did not support e-mail attachments since uniform redistribution is not possible.

The enhanced Flash function was therefore added to enhance the usability and the expressiveness of users and content authors. Sections 3.2 and 3.3 below describe the main functions added.

3.2 Data Storage Function

This function creates and uses a file data storage area by specifying a file. The content author may specify an area size of up to 10 KB. Each Flash file is allocated to a single data storage area, and by associating it with an ID, use of the data storage area by another Flash file can be prevented (**Figure 2**). Saving the focus position during browsing is one example of using the data storage function.

Since the browser application saves the part of the previous page in the focus, the system is designed so that the focus returns to the previous focus position when a link is selected to move to a new page and the user returns to the previous page. On the other hand, with



Flash content, the browser application has no means of acquiring the focus position, so that the focus returns to the top of the page when the user returns to the previous page. The immediately previous focus position is therefore stored in the data storage area for subsequent display at the previous position, even when the user returns to the previous page in a manner similar to Web pages.

Since the content author may freely use the data storage area, a Flash content service previously unavailable (e.g., display of character strings in Flash content unique to the user) is possible in addition to the ability to store the focus position.

3.3 Use of Metadata

Previous Flash files only contained data needed for comprising a file, so that supplementary information for the

file could not be added.

A metadata^{*2} area is newly created in Flash files and added the following three types of information.

1) Non-redistributable Identifiers

Flash content is currently handled as uniformly non-redistributable content that the content author cannot specify. A metadata area has been newly created to specify whether redistribution is possible, permitting the content author to control external output, and thus provide a new service related to output of Flash content (e.g., e-mail attachments, saving external memory).

2) Transparent Background Screen

This allows a transparent display of the background within the Flash replay area. This permits the display of the Web page background (background color and still background images only), non-rectangular images, and more expressive Web pages.

3) Specifying Pictograph Default Color

Text color in Flash files must be specified when creating content. Since pictographs are also displayed in the color specified with the authoring tool^{*3} rather than in a default color, the content author must specify the color for each pictograph.

Since pictographs are displayed in the pictograph default color specified, rather than in the specified color, the time required for authoring Flash content is reduced.

4. Music Player Usage History Transmission Function

4.1 Background

Use of music players on mobile terminals is increasing. In addition to using mobile terminals to download music from a download service, users are increasingly using their mobile terminals to listen to music converted by ripping^{*4} from CDs and other sources. Content providers are expected to associate the music that users listen to (including music converted by ripping from CDs) with such information as genre and artist, and provide it according to user preference in an online sales format. Of course, users will also be able to readily select music according to personal preference.

In order to improve such music distribution services as providing music suited to the preference of individual users, and ranking music as listened to by many users, the music player usage history transmission function (hereinafter referred to as the “usage history transmission function”) was developed to transmit information about listening histories to the content provider. This function enables the transmission of usage history information when the user selects a usage history transmission tag^{*5} on the Web page of the music distribution service.

*2 **Metadata:** Information related to content data, not content data itself. For example, the camera position (latitude and longitude) for an image file.

*3 **Authoring tool:** Application software for editing different types of data (e.g., text and images), and creating a single item of software

and content.

*4 **Ripping:** Conversion of digital data recorded on music CDs and other sources into a format readily handled by PCs.

*5 **Tag:** A descriptive method of indicating headings and links on web pages.

4.2 Usage History Transmission Procedure

Figure 3 shows the operational flow of the usage history transmission function.

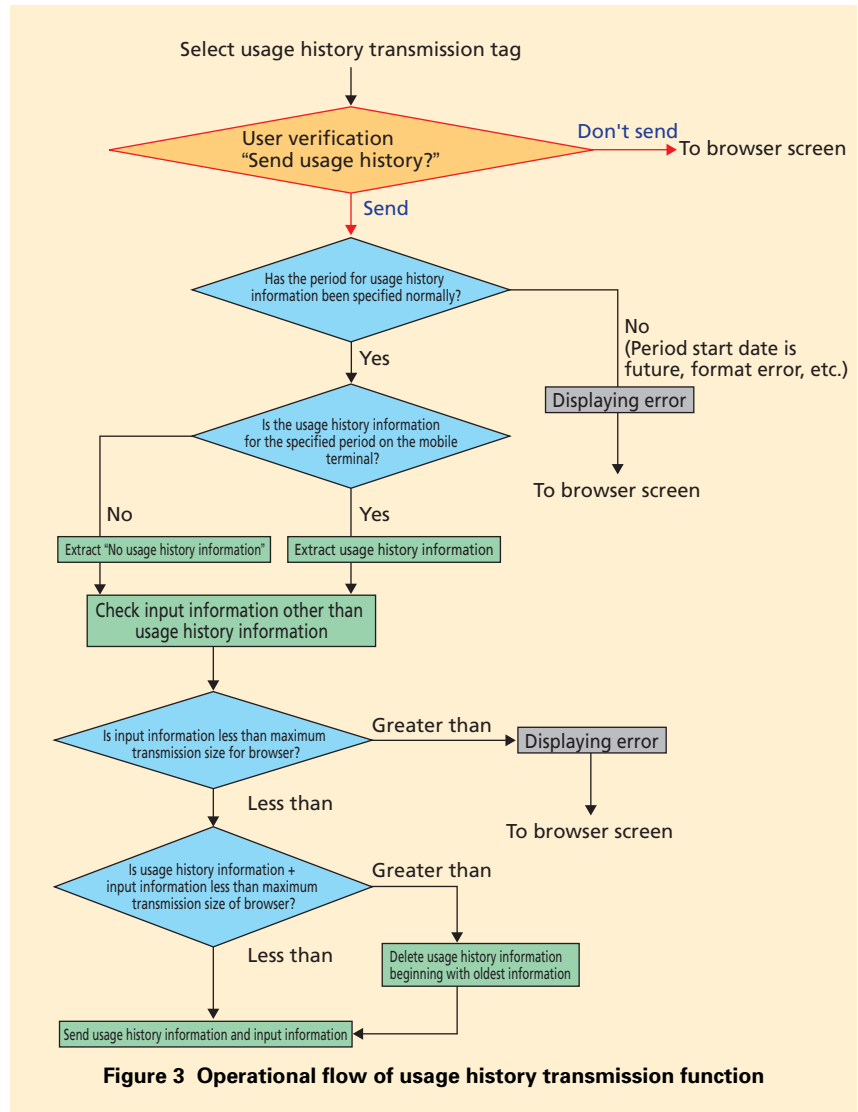
When the user selects a tag requesting usage history transmission, a screen appears allowing the user to select whether to transmit music player usage history. When the user selects “Do not send usage history” on this screen, the usage history information is not transmitted and the browser screen reappears. When the user selects “Send usage history,” the usage history information managed internally by the mobile terminal for the interval specified by the tag on the Web page is sent to the browser.

When the user transmits usage history along with other information entered on the Web page (e.g., information entered in comment fields of a questionnaire) and in other cases, the oldest information is deleted first to ensure that the entered information and usage history information can be contained within the maximum size for transmission with the browser, and the usage history is then transmitted to the content provider.

5. Distributed Speech Recognition Function

5.1 Background

The speech recognition function provides a means of input to the mobile terminal using the human voice. For



example, the speech of a user is converted into text for input to the mobile terminal.

The speech recognition function requires a dictionary for recognizing text from speech input. When only a few hundred words are to be recognized, the dictionary may be stored on the mobile terminal, however, when phrases are to be converted, the size of the dictionary will become too large (up

to 50,000-60,000 words or more), requiring speech recognition processing to be performed on the server. This method of performing speech recognition on both the mobile terminal and the server is called Distributed Speech Recognition (DSR).

The following were developed for use of the DSR function as a new input method for i-appli programs with the FOMA 905i series.

- Processing for speech feature extraction on mobile terminals.
- A new communications protocol to connect mobile terminals and the server.
- An Application Program Interface (API)^{*6} for i-appli programs.

An i-appli for Japanese-English translation of travel conversation, for example, permits the use of voice input to replace key input of text for travel conversations, and thus provides the user with greater convenience.

Figure 4 shows the internal configuration of the mobile terminal; Sections 5.2, 5.3 and 5.4 below describe the individual elements.

5.2 Processing for Speech Feature Extraction

Speech recognition requires processing to reduce sound (noise) other

than the user's voice, and processing to extract speech features from the user's voice for recognition. The section that performs this processing is referred to as the Front-End (FE) and represents a new feature in the FOMA 905i series.

In addition to speech feature extraction, the FE also determines the presence/absence of the user's voice, its volume, and size of the noise component, and then sends this information to the i-appli. Based on this information, the i-appli can provide advice to the user (e.g., "Please speak louder"). Furthermore, speech features can be held for a single user utterance lasting up to 30 seconds. Should a user utterance last longer than 30 seconds, the FE automatically interrupts voice input. When lasting less than 30 seconds, an utterance can be interrupted by the user or the i-appli. The FE extracts speech features for an utterance up to the point of interruption, and then sends these features to the i-appli.

5.3 A New Communications Protocol

DoCoMo DSR Protocol (DDP) has been developed as the protocol for communications between mobile terminals and the speech recognition server.

DDP is a protocol running over HTTP, and specifies commands from a mobile terminal in response to the progress of speech recognition processing, and responses from the server. It also specifies the output format for the

storage of speech features, and is based on the European Telecommunications Standards Institute (ETSI)^{*7} ES 202-050 [1] output format.

5.4 Use from i-appli Programs

A new API for i-appli programs has been developed to enable use of the DSR function. Figure 5 shows the case of user termination of an utterance as a basic example of the sequence using API for i-appli programs.

First, the i-appli acquires FE information from the FE (Fig. 5 (1)). Next, the FE information is used for negotiation^{*8} with the server (Fig. 5 (2)). An instruction is then sent to commence speech feature extraction based on the results of negotiation (Fig. 5 (3)). The i-appli displays the utterance start screen and prompts the user to begin speaking (Fig. 5 (4)). The features from voice input are extracted with the FE when the user speaks. When sufficient features have been accumulated, the fact is reported to the i-appli (Fig. 5 (5)). The i-appli then acquires the speech features from the FE (Fig. 5 (6)) and sends the data to the server (Fig. 5 (7)). This processing (Fig. 5 (5)-(7)) is repeated until the user instructs termination of the utterance (Fig. 5 (8)). The i-appli then instructs the FE to stop speech feature extraction processing (Fig. 5 (9)), and when all speech features extracted from input voice prior to the stop instruction are sent to the server, the recognition result chunk^{*9} from

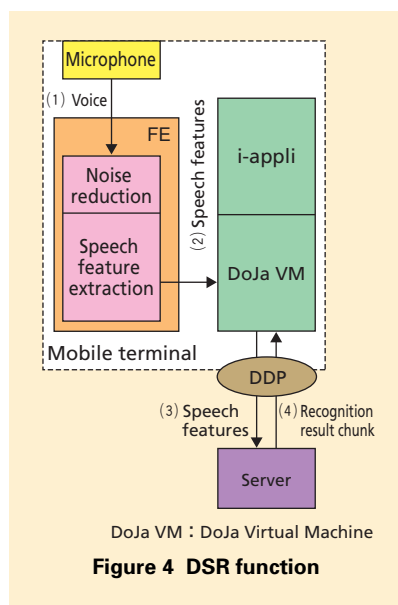
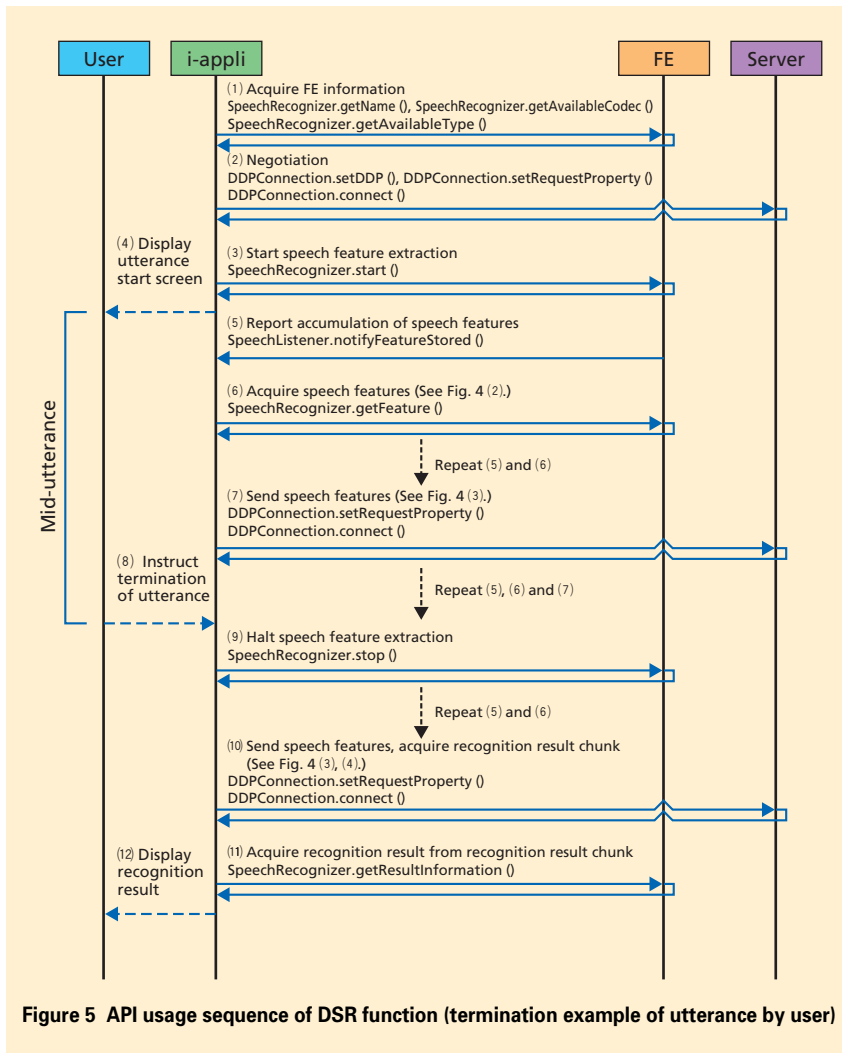


Figure 4 DSR function

*6 API: An interface to use functions supported by the OS and middleware for other software.

*7 ETSI: A European standardization organization engaged in telecommunications technology. Based in the Sophia Antipolis, France.

*8 Negotiation: Confirmation processing to match conditions used in speech recognition processing between mobile terminals and the server.



(Fig. 5 (12)).

6. Conclusion

The automatic software update function, enhanced Flash function, music player usage history transmission function, and distributed speech recognition function added to the FOMA 905i series have improved the update function, enhanced expressiveness, and much improved the ease of input to the mobile terminal.

Further improvements in the ease of using functions and more sophisticated I/O functions are being planned in order to develop mobile terminals desired by users.

REFERENCES

- [1] ETSI: "Speech Processing, Transmission and Quality Aspects (STQ); Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms," ES 202-050 v1. 1.5, Jan. 2007.

the server is output (Fig. 5 (10)). The recognition result chunk is analyzed in the FE and the recognition result is acquired (Fig. 5 (11)). The i-appli finally provides the recognition result to the user via the processing above

*9 **Recognition result chunk:** A recognition result described in accordance with the format specified by DDP. In addition to the recognized text information itself, it also includes the number of candidate recognition results, and the confidence value.