Technology Reports

Systems for Personal Data Storage Services

Makoto Hamatsu, Tetsuo Sato and Eiji Yano

As part of DoCoMo's efforts toward "secure and safe" mobile terminals, we developed an personal data storage service system. With this development, it is now possible to avoid the loss of user data such as phone books due to the loss or failure of mobile terminals.

1. Introduction

Until now, user data stored within FOMA terminals has been assumed to be backed up using Secure Digital (SD) memory and other external storage media at the user's own initiative, and backup services were not included in the service guarantee.

However, as mobile terminals are becoming increasingly sophisticated with higher added value, the loss of user data due to the mobile terminal being damaged or lost has begun to have greater impact at the social level as well. Since DoCoMo takes all of its social responsibilities seriously, we decided to address this issue by launching personal data storage services as a way to achieve "security and safety in times of emergency."

The personal data storage services allow users to save user data such as phone books, images, and mail saved within a mobile terminal on the servers provided by DoCoMo, and recover the data whenever needed.

This article provides an overview of the personal data storage services, along with the implementation method and the functions overview.

2. Services Overview

Figure 1 shows an overview of the personal data storage services. These services support saving the following three types of user data.

- Phone books

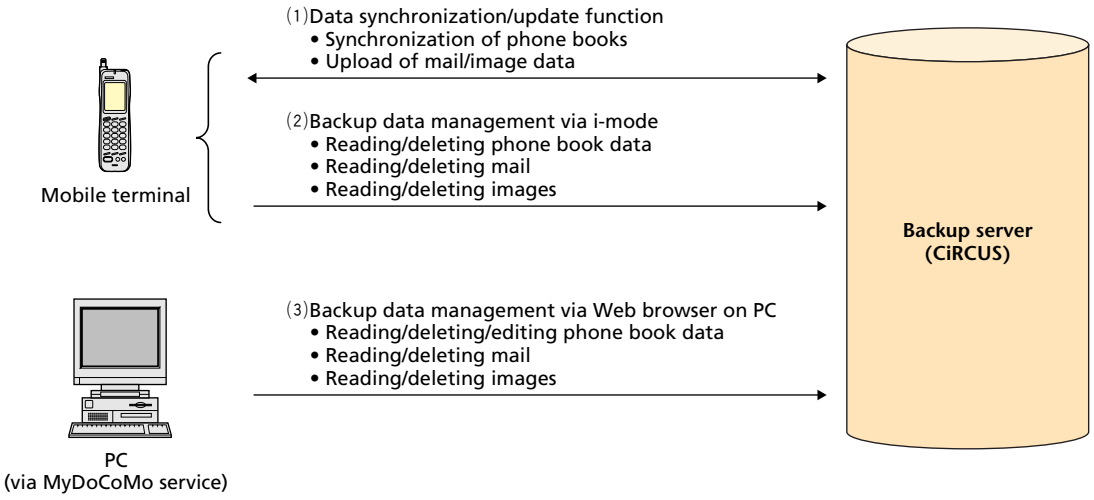


Figure 1 Overview of the personal data storage services

- Mail (i-mode mail and Short Message Service (SMS)[*1])
- Images (primarily images taken with built-in cameras)

Note that the personal data storage services system provide the following three prominent types of functions:

1) Data Synchronization/Update Function

This function receives requests to synchronize phone books and upload mail and/or images from mobile terminals, and handles the updating of data stored on the backup server.

2) Backup Data Management via i-mode

This function allows the reading/deleting of phone books, mail, and images stored on the backup server from mobile terminals via i-mode browsers. It also provides functionality for saving/recovering phone book data stored on the backup server and enables various user settings related to the phone book backup services, such as setting the intervals for automatic synchronization between a mobile terminal and the backup server.

3) Backup Data Management via Web Browser on PC

This function receives requests sent from Web browsers on PCs to the MyDoCoMo service[*2] to read, delete, and edit phone books, mail, and images stored on the backup server. It also provides functionality for saving and recovering phone book data, enables various user settings related to the phone book backup services in the same way as for management via i-mode browsers, and also allows phone books to be uploaded and downloaded in the form of Comma Separated Values (CSV) files[*3].

3. Method of Implementing Services

User data such as phone books and mail are changed on a daily basis. For instance, new data is added, other data modified or deleted, thus resulting in a constantly changing "current status." For this reason, in order to recover such data at any time in preparation for unexpected loss or failure of a mobile terminal, the timing at which to back up the data becomes an important factor. Moreover, it is necessary to consider ways to reduce the network traffic and user communication fees as well.

The services thus adopts Open Mobile Alliance (OMA)[*4]-Data Synchronization (DS) technology, an international standard for data synchronization that allows efficient communication by transferring only incremental updates due to changes made since the last connection [1]. OMA-DS is a client-server-type application protocol, and the server functions are implemented in treasure Casket of i-mode service, high Reliability platform for CUStomer (CiRCUS)[*5]. OMA-DS is implemented in the services as the host protocol of HTTP and the network architecture shared with the i-mode browsers. Note that the Access Point Name (APN)[*6] is also the same as for i-mode.

---

*1 SMS: Service for sending/receiving short text-based messages mainly between mobile terminals. It can also be used for sending/receiving control signals for mobile terminals.

*2 MyDoCoMo service: A Website where DoCoMo's customers can login using PCs. The site accepts the payment of fees as well as changes to the terms and conditions of subscription.

*3 CSV files: A file format where data is arranged in columns separated by commas (","). Typically used for data storage in spreadsheet and database software.

*4 OMA: An industry standards body to enable technology for services and applications in mobile communications and for ensuring interoperability.

*5 CiRCUS: An i-mode gateway system.

4. Realized Functions and Implementation on Mobile Terminals

4.1 Overview of OMA-DS Protocol

The methods of data synchronization and incremental updates between mobile terminals and the server used for the personal data storage services are based on OMA-DS version 1.2. OMA-DS prescribes six usage patterns for data synchronization (**Table 1**). However, in order for users to use these types of synchronization optimally in a given usage scenario, they must understand each concept. To alleviate this requirement, the services are designed so that the usage conditions are detected and an appropriate synchronization type is selected automatically on the mobile terminal side, thus facilitating a wider range of service users who need not be aware of the various concepts of synchronization. For example, "Two-way sync" is executed once connection to the server has been established, while "Slow sync" is executed automatically when information about changes made since the last update managed within a mobile terminal is no longer meaningful, such as when an phone book has been overwritten by copying data from miniSD or other external storage media to the mobile terminal. When saving image and mail data, the user simply selects which data is to be saved on the server side; therefore, it is not necessary to synchronize between the mobile terminal and CiRCUS databases, with "One-way sync from client only" selected automatically in this case.

Table 1 Types of data synchronization in OMA-DS

Synchronization type	Overview
Two-way sync	The basic type of data synchronization. Incremental updates due to changes made since the previous synchronization are exchanged between client and server.
Slow sync	Data on the client side is compared field-by-field with data on the server side, and data missing on either side is copied so that all data exists on both sides.
One-way sync from client only	Incremental updates due to changes made since the last synchronization are sent from the client side to the server side.
Refresh sync from client only	Data on the client side is sent to the server side, and the corresponding data on the server side is overwritten.
One-way sync from server only	Incremental updates due to changes made since the last synchronization are sent from the server side to the client side.
Refresh sync from server only	Data on the server side is sent to the client side, and the corresponding data on the client side is overwritten.

4.2 Recovery after Disconnection

In today's diverse mobile communication environments, it is necessary to consider cases where communication is cut in the middle, such as when the user enters a tunnel or elevator. OMA-DS adopts the concepts of Suspend and Resume, whereby Suspend (the suspension of data synchronization processing) is assumed in the following two ways:

1) Synchronization is Suspended due to User Operations

In this case, communication is not disconnected until the client and server have exchanged information on what data has thus far been received correctly during the communication.

2) Sudden Disconnection of Communication

In this case, communication is disconnected without the client and server having the time to exchange information on the progress of synchronization with each other.

In case 1), since information about the data already received correctly on the receiving side has been exchanged with the transmitting side, both sides know the point from which to resume synchronization. Conversely, in case 2), the information is not shared. In order to share information at the same level as in case 1) to the greatest extent possible, the services adopt a function on the mobile terminal side called Cached Map, for which implementation is arbitrarily defined in OMA-DS. The Cached Map function stores information about the data received from the server before disconnection and notifies the data registered as having thus far been received to the server when communication is resumed. Since OMA-DS allows transmission results to be determined at individual data levels when sending multiple data items, it is relatively easy for the client itself (mobile terminal) to manage which data has been sent correctly. Another feature is that the services also use the Cached Map function to allow similar management of received data as well

*6 APN: Similar to a telephone number, this identifies a connection destination and is used in packet communication.

(i.e., whether or not "already registered received data" has arrived at the server).

4.3 Interface Design Considering Differences among Individual Mobile Terminals

There are a many vendors who provide mobile terminals. Consequently, the numbers of mail addresses and telephone numbers that can be registered to one memory dial number typically differ from implementation to implementation. For example, when five telephone numbers are registered for one record[*7] on the server side and only three numbers can be registered per record on the mobile terminal side as a result of changing the model, only three numbers are saved on the mobile terminal after synchronization. Thus, in such cases, executing synchronization after some changes are made to specific records on the mobile terminal may result in the data on the server being overwritten by the data notified as incremental updates, and the data on the server side that could not be previously registered on the mobile terminal side being deleted. In order to prevent such unintended data loss from occurring, we applied the concept of Field Level Change in the interface design. Field Level Change is an approach whereby only changes made to a specific record are notified to the server at an incremental update, and data not included in the notification is treated as "data left unchanged since the last synchronization" on the receiving side. In other words, data that could not be actually saved in a mobile terminal is disguised as if it was saved, resulting in the server regarding the data sent from the mobile terminal as being strictly limited to "incremental updates due to changes made since the previous synchronization." As a result, this feature prevents the unintended deletion of data.

5. Functions Implemented on the Server

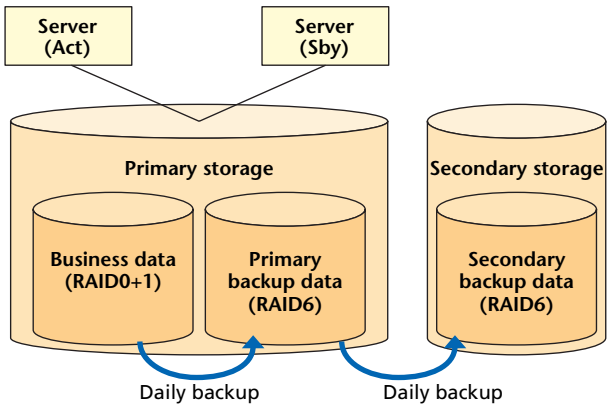
5.1 Overview

Since the personal data storage services are intended "to take care of the users' data," these services must achieve high reliability in terms of data security. For this reason, backup server functions are implemented in the highly reliable CiRCUS i-mode server.

5.2 Securing Reliability

In order to ensure high reliability, the backup server executes the remote real-time synchronization of data between the main center and Mirror center. The backup server also runs daily backup routines within the main center as well to create secondary backups in case of failure during backup. The backup server within the main center adopts an Active/Standby (Act/Sby) cluster configuration[*8]; the primary and secondary storage enclosures are connected outside the server, and the primary and secondary backups are saved in the primary and secondary storage enclosures, respectively. **Figure 2** shows the server and storage connection configuration.

Moreover, to further enhance security, the server encrypts



RAID (Redundant Arrays of Inexpensive Disks): A device that manages multiple hard disks at the same time.

Figure 2 Configuration of backup server and storage connection

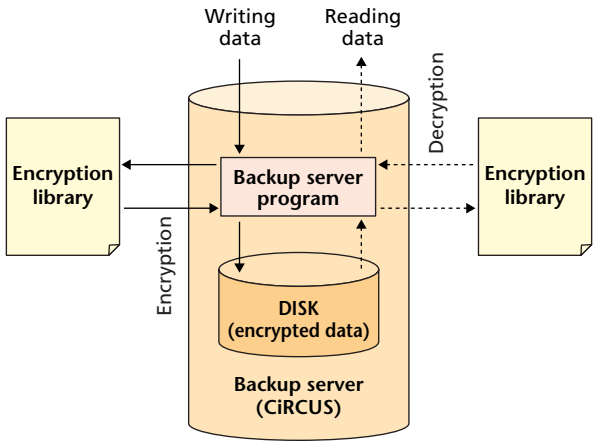


Figure 3 Overview of encryption function

*7 Record: In this article, a record is a unit of data management in a database, where names, telephone numbers, mail addresses, and other information are organized into one data structure.

*8 Cluster configuration: A system configuration that prevents system down by adopting several subsystems instead of a single system, and uses subsystems if trouble occurs in the subsystem currently in use.

all data before being written to the backup server. When writing the data, the backup server calls the common encryption library[*9] in CiRCUS, encrypts the data to be written, and then writes the data to the DISK. Similarly, when reading data, the backup server calls the encryption library, decrypts the data, and then creates a message to be sent to the receiver. **Figure 3** shows an overview of the encryption function.

5.3 Phone Book Data Save/Recovery Functions

The backup server system is composed of two areas: a normal and a storage area. Each area is equipped with a function for saving phone book data in the normal area to the storage area, and a function to recover phone book data from the storage area to the normal area (**Figure 4**). The normal area is used for OMA-DS synchronization communication with mobile terminals, whereas the storage area is not updated in OMA-DS synchronization communication with mobile terminals.

The user can execute operations to save data kept in the normal area to the storage area, and recover data from the storage area to the normal area, either via mobile terminals or PC browsers. Moreover, at initial synchronization after subscription, the phone book data created in the normal area is automatically saved to the storage area. Synchronization communication immediately after recovery from the storage area is always performed using "Slow sync" so that the phone book data on the server side is securely reflected on the mobile terminal side.

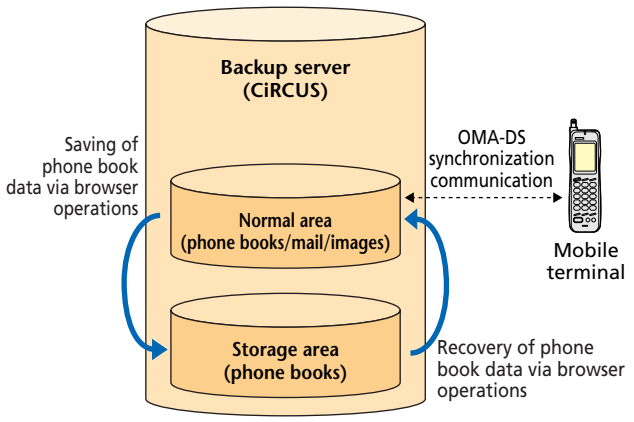


Figure 4 Overview of save/recovery functions

These save/recovery functions are provided to enable the recovery of data in case of emergencies, such as when a user mistakenly deletes phone book data by executing the wrong operation.

5.4 Automatic Synchronization Function

In order to synchronize the phone book data in a mobile terminal and the phone book data on the backup server, the backup server is equipped with a function to regularly and automatically send reception notifications to the mobile terminal. Upon receiving a reception notification, the mobile terminal requests the server to synchronize the phone books. When OMA-DS synchronization communication between the backup server and mobile terminal is completed, the phone book data within the mobile terminal and the phone book data on the server are synchronized.

With the network load taken into consideration, reception notifications are automatically transmitted during late-night time slots when the volume of traffic is low. When reception notifications are sent, the target mobile terminals are properly selected so that automatic transmission of reception notifications is not concentrated on particular subscription companies.

The intervals at which to automatically transmit reception notifications can be set or changed for each subscriber (e.g., daily transmission, weekly transmission, no automatic transmission). This automatic transmission is subject to congestion control. For example, if all reception notifications cannot be sent at the specific automatic transmission time on the date specified by a user due to an increased number of subscribers or unevenly distributed settings, transmission is postponed until the next day.

6. Conclusion

This article described an overview of the personal data storage services, the methods of implementation, and the functions provided. It is possible to expand various other services with backup functionality as well by simply applying the functions implemented in these services. We expect that the development of these services will form the basis for the introduction of new

*9 Library: A collection of general purpose software programs in a reusable form.

services from now on.

In the future, we intend to add varieties of supporting data, as well as making communication even faster and more efficient by improving the protocol.

REFERENCES

[1] SyncML Data Sync Protocol;
http://www.openmobilealliance.org/release_program/index.html