iSMS: An Integration Platform for Short Message Service and IP Networks

Herman Chung-Hwa Rao, ATT Laboratories-Research
Di-Fa Chang, University of Southern California
Yi-Bing Lin, National Chiao Tung University

Abstract

This article describes iSMS, a platform that integrates IP networks with the Short Message Service in mobile telephone systems. iSMS provides a generic gateway for creating and hosting wireless data services for mobile stations. Our approach does not require any modification to the mobile telephone system architecture. The iSMS system can be quickly developed and operated by a third party or end user without involvement of mobile equipment manufacturers and telecom operators. Based on the iSMS platform, we illustrate services such as e-mail delivery/forwarding, Web access (e.g., stock and train schedule query) and handset music services. The iSMS platform and the services have been implemented for GSM networks. With iSMS, users are able to use standard GSM handsets to access wireless Internet services, while other approaches like Wireless Application Protocol and SIM Toolkit services require function-enabled MSs.

Recently Short Message Service (SMS) [1–4] has become a mature wireless communication service. Most modern digital cellular phone systems, such as Global System for Mobile Communications (GSM) [1], Digital Advanced Mobile Phone Service (DAMPS) based on IS-136/time-division multiple access (TDMA) [5], and cdmaOne (IS-95/code-division multiple access, CDMA) [6], offer SMS, which is considered a profitable value-added service. A natural extension to SMS is to integrate this service with e-mail services, which provides linkage between mobile networks and IP networks. Furthermore, several Internet applications over SMS can be implemented on similar platforms. Before we elaborate on SMS-Internet integration, we use GSM as an example to introduce SMS.

GSM SMS provides a connectionless transfer of messages with low-capacity and low-time performance. Each message can contain up to 140 octets or 160 characters of GSM default alphabet [7]. It operates like a paging service with added capability so that messages can be passed in both directions. The short messages are transported on the GSM standalone dedicated control channel. Thus, messages can be received and sent while mobile users are in conversation (that utilize GSM traffic channels).

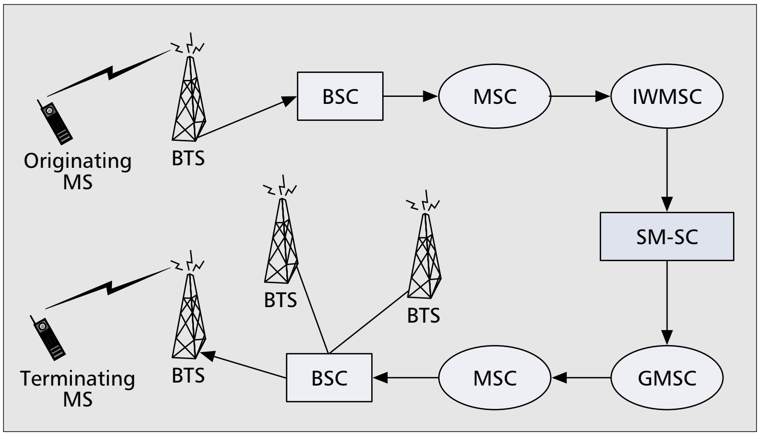
Two types of GSM SMSs have been defined. Cell broadcast service periodically delivers short messages to all subscribers in a given geographical area. Point-to-point service enables short messages to a specific user, which can be considered an enhanced two-way paging service. The GSM SMS network architecture is illustrated in Fig. 1. In this architecture, when a GSM mobile station (MS) sends a short message, this message is delivered to a mobile switching center (MSC) called an SMS interworking MSC (SMS IWMSC). The IWMSC passes this message to an SMS center (SM-SC). Upon receipt of the short message, the SM-SC may send an acknowledgment back to the originating MS (sender) if an acknowledgment request is specified in the short message. The SM-SC then forwards the message to the destination GSM network through a specific GSM MSC called the SMS gateway MSC (SMS GMSC).

Following the GSM roaming protocol [1], the SMS GMSC locates the serving MSC of the message receiver and forwards the message to that MSC. This MSC broadcasts the message to the base transceiver stations (BTSs), and the BTSs page the destination MS (receiver). Every short message contains a header in addition to the body. The header includes the originating MS address, the terminating MS address, the serving SM-SC address, a timestamp, and the length of the message body. Mobile station integrated services digital network (MSISDN) numbers (i.e., GSM telephone numbers) are used for addressing. An MS that supports SMS must contain special software to enable the messages to be decoded and stored. Today, SMS software is available in most standard GSM MSs. Every MS consists of two parts: the subscriber identity module (SIM) and the mobile equipment (ME). The SIM is protected by a personal identity number (PIN) code chosen by the subscriber, which contains the subscriber-related information. Besides the PIN codes, the SIM may contain a list of abbreviated and customized dialing numbers, short messages received when the subscriber is not present, names of preferred networks to provide service, and so on. Parts of the SIM information can be modified by the subscriber by using either the keypad of an MS or a PC connected to a SIM card reader.

The ME contains the non-customer-related hardware and software specific to the radio interface. When the SIM is removed from an MS, the remaining ME cannot be used to reach the service except for emergency calls. The received short messages can be stored in either the SIM or the memory of the ME, and are displayed on the standard MS screen.

There are three types of short messages: user-specific, ME-

0890-8044/01/$10.00 © 2001 IEEE



■ Figure 1. The *GSM SMS network architecture.*

specific, and SIM-specific. A user-specific message is displayed to the user. An ME-specific message is processed within the ME, which is not shown to the user. A special function created by the handset vendor can be triggered by the ME-specific message. For example, Nokia Smart Message is an ME-specific message which provides functions such as playing a ringing tone, displaying a business card, and modifying the default icon. Similarly, a SIM-specific message is processed at the SIM card. If a GSM operator designs a special function in the SIM card, the SIM-specific message can trigger this function. If the short message receiver is not attached to the network (e.g., the MS is turned off), the SM-SC will resend the message until success or the message is expired. There is a limitation on how many undelivered messages per receiver can be cached on the SM-SC. No acknowledgment message is sent back to the sender when a given message reaches the receiver, nor when the message is expired in the SM-SC. Furthermore, sequencing on message delivery is not guaranteed in the GSM SMS protocol; SMS is a best-effort store-and-forward mechanism. The reader is referred to [1] for an SMS tutorial.

As a wireless data service, SMS has distinct features such as handset alert capability and support of ME-specific and SIM-specific data. The MS is considered an "always-on" device that facilitates instant information exchange. No dialup modem connection is required to access SMS. Furthermore, SMS provides message storage when the recipient is not available, and allows simultaneous transmission with GSM voice, data, and fax services. On the other hand, SMS has drawbacks such as narrow bandwidth and long latency of end-to-end transmission. To design an SMS-IP system, the above SMS strengths and limitations cannot be ignored. Especially, the IP server itself may also be mobile (e.g., the iSMS gateway proposed in this article can move when it provides services), and usage of the limited wireless bandwidth must be carefully addressed. To facilitate the research and development of various SMS applications over Internet, we need a generic gateway to interwork GSM and IP networks, and many specific data format converters between IP application contents and ME-specific/SIM-specific/text SMS byte codes.

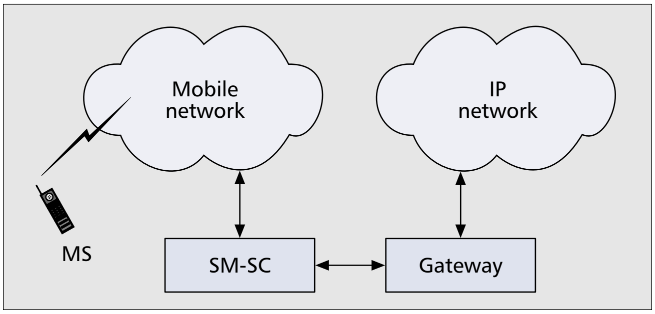
In most existing implementations [8–11], SMS and IP networks are integrated through an SM-SC (or an SM-SC-like component), as illustrated in Fig. 2. In this figure, a gateway interworks the SM-SC to the IP network where a specific protocol is essential for communication between the SM-SC and the gateway. Since the SM-SC is not defined in the GSM specifications and its implementation is vendor-specific, the SM-SC-based SMS-IP integration solution heavily depends on SM-SC vendors. Furthermore, the SMS-IP gateway is maintained and controlled by GSM operators. It is very difficult (if not impossible) for the third party to deploy new services via SMS without having full cooperation from GSM operators. Also, from a GSM operator's viewpoint, maintaining a reliable, secure, scalable interconnection platform between individual service providers and an SMS-IP gateway will not be an easy task. The SMS-IP approach typically utilizes a centralized gateway, where performance, scalability, and reliability issues must be carefully considered.

To address these issues and to further support an environment for quick prototyping and hosting wireless data service, we propose iSMS, an endpoint SMS-IP integration solution that is transparent to the existing SM-SC and GSM network. This article first introduces the iSMS system architecture. Then we describe the iSMS communication protocols and application programming interface (API). We discuss the designs and implementations of several iSMS applications and point out potential future applications. Finally, we conclude our work by summarizing the strengths and limitations of iSMS.

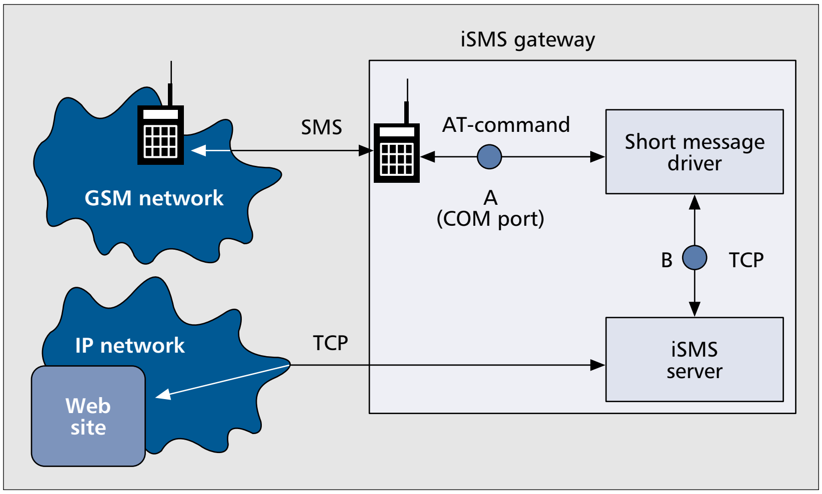
iSMS System Architecture

Figure 3 illustrates the iSMS architecture. In this architecture, an iSMS gateway is introduced. No components in the GSM and IP networks are modified. The MS is a commercial handset product without installing any new software.

A major difference between the iSMS and SM-SC-based architectures is that the iSMS gateway connects to an MS instead of the SM-SC. This MS serves as the GSM-compliant modem that provides iSMS wireless access to the GSM network. We refer to this MS as the *MS modem*. In our implementation, the iSMS gateway is a PC running on Windows 95/98 or NT. (A UNIX version of iSMS should be trivial.) The PC-based gateway can be a desktop with high reliability and availability. The PC can also be a notebook. In this case, the gateway becomes a mobile server, which may move around just like a GSM MS. Several MS models including Nokia Card Phone, Nokia 6150 [12], Ericsson GC25, and Ericsson SH888 [13] have been used as MS modems in the current iSMS implementation. The MS modem and iSMS gateway are connected by an RS232 port, an infrared port (e.g., IrDA), or a PCMCIA interface. In iSMS, the data sent from the IP network to the GSM network are automatically packaged into short messages. The short messages can be multicast to up to 65,535 receivers. Similarly, an MS can broadcast short messages to several servers connected to the iSMS gateway. The iSMS system is identified by the IP network through the IP address assigned to the gateway, and is addressed by the GSM network through the MSISDN of the MS modem attached to the gateway.



■ Figure 2. *SMS-IP integration with SM-SC.*



■ Figure 3. *iSMS system architecture (an end-point SMS-IP integration solution).*

In the iSMS model, a user of an iSMS application is also a GSM customer. There are several alternatives to access iSMS services. For example, a user may make a request by sending a short message. This short message is terminated at the MS modem connected to the iSMS gateway. Upon receipt of the short message, the iSMS gateway executes the corresponding routines and returns the results to the user. The iSMS gateway may also be triggered by predefined customized events. In a home security application, for example, if someone rings the doorbell when the user is not at home, the iSMS gateway may send an alerting short message to the GSM MS of the customer. The iSMS gateway consists of two parts: iSMS servers responsible for service provisioning, and a short message driver responsible for communication between the GSM network and iSMS servers in IP networks (Fig. 3). The communication protocol between the MS modem and the short message driver (reference point A in Fig. 3) is implemented using the SMS AT command set [3]. The communication functions between the iSMS servers and the short message driver (reference point B) is implemented through the iSMS communication API based on the TCP socket. Details of these communication mechanisms will be given later.

During iSMS system initialization, the short message driver opens COM port driver (reference point A) for sending/receiving short messages to/from the GSM network (via the MS modem). The short message driver also opens and listens on a predefined TCP port (reference point B) for server connection requests. For each connection request, the server will register the telephone numbers of its customers' GSM MSs to the short message driver. Messages from registered senders will then be forwarded to the server.

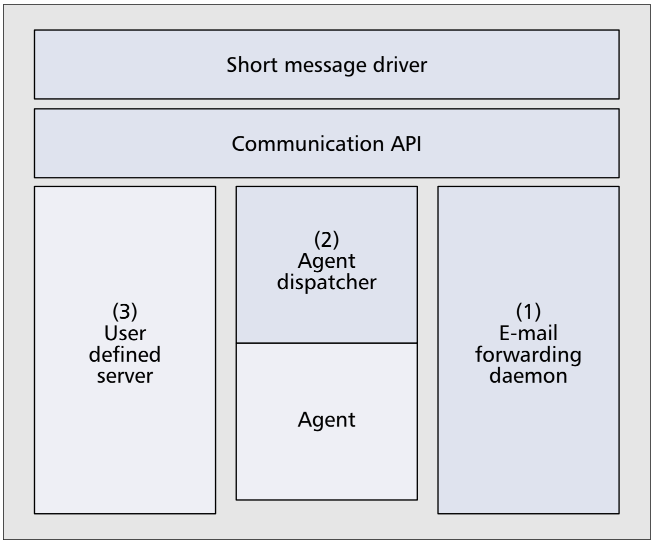
The short message driver performs conversion between the TCP socket API (the interface to the iSMS server) with the SMS AT command set (the interface to the MS modem). The short message driver receives incoming short messages from the COM port and passes these messages to the iSMS servers according to the registration table. Depending on the registration status, the short message driver may forward a message to several iSMS servers or drop the message if no server has registered the sender of the message. For outgoing short messages, the driver receives messages from servers, transforms them into short message format, and then sends them out to the GSM network via a COM port.

An iSMS server may run on the same host as the short message driver or on a remote site. For security reasons, the driver authenticates the servers for communication sessions. A command sent from the MS is pure text in the short message. The short message driver dispatches the message to the appropriate iSMS server based on the MSISDN of the MS (caller ID) and the mapping in the registration table. The server parses the command in the message body and then invokes corresponding internal functions or external agents to execute the messages. Functions are executed on the same address space as the server, while agents run on different processes. The caller ID is used to identify the user in the current iSMS version. iSMS security is implemented at the application level. For example, passwd may be required as a parameter of every command sent from the MS to the iSMS gateway.

To develop a new service, one implements iSMS servers that communicate with the short message driver by using the functions defined in the iSMS communication API. In the current iSMS version, we have implemented two kinds of servers, the E-mail Forwarding Daemon and Agent Dispatching Server. Furthermore, the iSMS platform allows developers to implement new server types. The relationship among various server types and the short message driver is illustrated in Fig. 4. In this figure, the E-mail Forwarding Daemon ((1) in Fig. 4) relays messages between the MSs and e-mail systems on the IP domain. It supports interface to Microsoft Exchange Server as well as to standard SMTP/POP3. The daemon converts a short message sent from an MS to e-mail and forwards it to an SMTP server for delivery. The daemon may periodically query a mail server (e.g., via POP3 interface), pick up important e-mails (according to user profiles), and send out SMS notifications to users' MSs. The daemon software can easily be generalized to support other mail systems such as AOL Instant Message.

A server of the agent-dispatching type ((2), Fig. 4) consists of an agent dispatcher and several agents where each of the agents is an iSMS server. The dispatcher invokes the agent corresponding to the SMS message header and passes the message body as the parameters to the agent. Each agent implements one function. When the agent finishes processing a message, the agent dispatcher collects the results and sends them back to the short message driver.

Depending on the implemented services, each agent-dispatching-type server may implement its own message parsing



■ Figure 4. *iSMS driver and server structure.*

rules and maintain a command table with function/agent pairs. The current iSMS version has implemented a general-purpose agent dispatcher platform. In this platform, details of communication between the short message driver and the server are hidden from service developers. Instead, the developer only needs to specify the agent dispatching rules and implement the agents that carry out the services.

The iSMS platform allows service developers to implement new server types ((3), Fig. 4). In this case, the developer needs to implement the interaction between the short message driver and the servers. We provide a communication API that allows the developer to quickly deploy new servers. This communication API will be elaborated on in a later section.

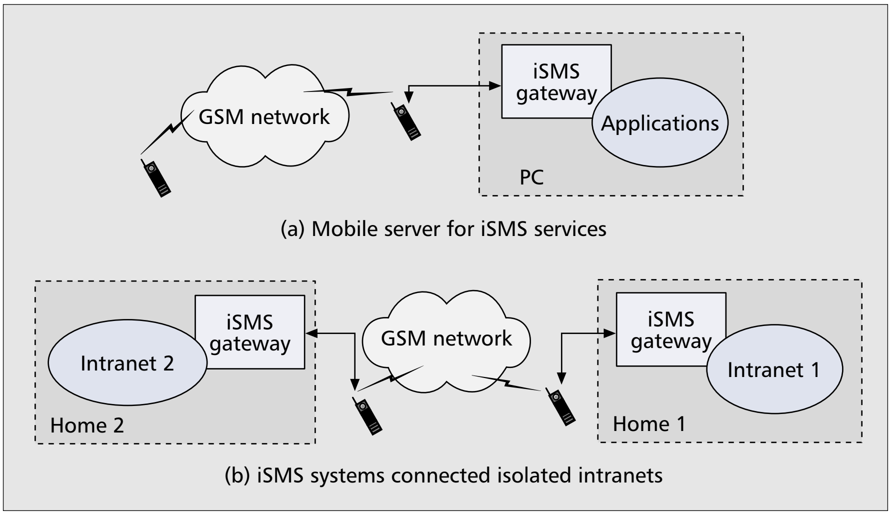


Figure 5. *Variations of iSMS configuration.*

The system also supports group broadcasting and smart message delivery including the ringing tone, music, and icons. The smart messaging protocol defines the format of ASCII stream that can be passed via different transport protocols. Smart messaging is considered in our implementation because it has been adopted by major GSM handset suppliers for SMS as well as messaging for infrared among personal digital assistants (PDAs) and even for Bluetooth [14].

If all iSMS applications and the iSMS gateway are implemented in a portable notebook that is not connected to any IP network, the iSMS system becomes a mobile server, as illustrated in Fig. 5a. In this configuration, mobility management of the iSMS server is automatically maintained by GSM Mobile Application Part (MAP) [1]. In other words, the existing GSM mechanism will transparently track the moving iSMS server, and iSMS does not need to implement a location tracking mechanism. A mobile iSMS server can be used, for example, in a mobile library application where the library truck moves around a city. The iSMS server connects to the library database in the truck. An iSMS customer can use the MS to check the status of a book from the library database and the location of the library truck.

Figure 5 shows another iSMS configuration where the iSMS systems connected to different isolated intranets (e.g., homes), which can query each other through a mobile network.

iSMS Communication Protocols

Two communication protocols have been developed in iSMS. A communication protocol between the MS modem and the short message driver is implemented using the SMS AT command set [3]. iSMS also implements a communication API that provides functions to support communication between the short message driver and the iSMS servers through TCP connections.

With the above communication protocols, several types of MS (customer) and agent (service) interaction are implemented in iSMS:

- Query-response services. A query (in SMS format) from an MS is sent to an iSMS server, which in turn invokes corresponding agents and sends back results to the original MS. Examples of query-response services include querying stock, train schedule, and UPS package status.
- Relaying services. An iSMS server forwards messages and information (e.g., e-mails and AOL Instant Messages) to corresponding mobile users whenever messages are available. The server maintains a user profile and actively collects information from different Internet servers.
- Notification services. An iSMS server delivers notification to a user's mobile. Notifications are triggered by events specified by users. For example, a user may specify information (headline news, stock quotes, and weather information) she/he wants to receive every day to a scheduling server (like cron service in UNIX) in iSMS. As another example, an iSMS agent (e.g., stock/auction monitor agents) running on behalf of a user monitors information on the Internet and delivers alert messages to the user.
- Group communication service. The iSMS server allows specifying groups, each of which contains more than one MS members. Messages to a group (via SMS or from other services) will be forwarded to all members in the group.

This section shows how iSMS communication protocols are implemented based on GSM. To accommodate iSMS for non-GSM systems, if the systems support AT commands, we need only replace the GSM card phone with the appropriate card phone (e.g., cdmaOne card phone). If the mobile systems do not support AT-commands, the short message driver must be modified. With the communication API structure, the iSMS server and all iSMS service applications need not be changed.

SMS AT Command Set

The short message driver connects to the MS modem by RS232 or PCMCIA card. In the current iSMS version, the messages are delivered through serial port number 4. Two serial ports are used to test the driver in the simulation mode. The NULL port always accepts outgoing short messages successfully. The `LOOPBACK` port always sends back outgoing messages as incoming short messages. A variable `MOBILE_COM_PORT` is used in the driver to indicate which port is connected.

The communication protocol between the MS modem and the short message driver is based on the SMS AT command set [3]. To run this protocol for a specific MS model, one should specify the type of MS modem. MS modem setup is achieved by two variables, `MOBILE_TYPE` and `MOBILE_INIT_STRING`. Some of the AT commands used in the short message driver are listed in Table 1. Every command sent from the short message driver begins with "AT" (e.g., "`AT+CMGS`"). The MS modem's responses are without "AT" (for example, "`+CMGS`").

When the short message driver receives a message from an iSMS server, the driver divides the message into several segments of length less than 140 octets. For each receiver, the driver generates a set of SMS packets from the message segments. For example, if the message is divided into four seg-

AT command	Description
+CNMI	New message indications to TE
+CSCA	Service center address
+CMGD	Delete message
+CMGS	Send message
+CMGL	List message
+CSMP	Set text mode parameters
+CMT	SMS message received

■ Table 1. *AT commands used in iSMS (a partial list).*

ments and there are three receivers, the drivers will generate 12 SMS packets. The driver pushes these SMS packets into a FIFO queue and transmits them sequentially. For every SMS packet, the driver issues the SMS AT command that instructs the MS modem to submit a short message. There are two command modes for the MS: text mode and packet data unit (PDU) mode. The parameters of AT commands for these two modes are different. Most MSs in the market support the PDU command mode. In the PDU mode, the parameter for sending a short message is the entire short message packet. In Table 1, the AT command is +CMGS with the following format (in packet mode):

```
AT+CMGS=<length><CR><pdu>,
```

where <length> is the length of the actual data unit in octets, and <pdu> is the short message to be delivered. Details of other AT commands can be found in [3].

iSMS Communication API

We have implemented an iSMS API using VC++. Through this API, servers and agents for specific applications can easily be developed. For every application, there is an iSMS server that communicates with the short message driver through a TCP connection (TCP port number 1122 in the current iSMS version), and several agents may be created to interact with the iSMS server through command execution. The relationship among the short message driver, iSMS servers, and agents is illustrated in Figs. 3 and 4.

In iSMS API, a class CsmsdServer implements the following communication functions between a server and the short message driver.

Connect() sets up a communication link from a server to the short message driver. This function takes two arguments: the IP address of the short message driver and the TCP port number of the driver. This function returns the status of the connection establishment. Disconnect() terminates the TCP link between the server and the short message driver.

SetTimeout() sets a timeout period. When a server issues an operation to the short message driver, a timeout period is set. If the socket to the driver is not ready before the specified timer expires, the operation fails.

Register() specifies the customers of a server with their GSM telephone numbers. The function takes two arguments: an array of phone numbers and the size of the array.

Status() returns the communication status between the server and the driver. The status SMCMD_READABLE indicates that the server is ready to retrieve a short message from the driver. SMCMD_WRITABLE indicates that the server is ready to send a message to the driver. SMCMD_ACK indicates that the message from the server to the driver is successful. SMCMD_CLOSED indicates that the connection is closed.

Send() sends data to one or more customers (i.e., GSM MSs). Recv() is invoked by a server for receiving data from a GSM MS. RecvACK() returns an acknowledgment from the driver for the status of message transmission.

Besides the Send function, the CsmsdServer provides two more functions for facilitating text messages and unstructured binary data delivery. SendText() is used to send a message with a null-terminated string of ISO-8859-1 characters or traditional Chinese (BIG 5) characters. SendData() is used to send the unstructured binary data using GSM 8-bit coding.

Examples of Services

To further illustrate iSMS functionality, this section describes several services already implemented and running on the iSMS platform. Special commands are defined for these services. An online menu can be built in the MS so that a customer can check the menu to figure out the meaning of a special command. Note that we have built a platform to test the implemented services without actually involving the GSM network (the platform simulates the interaction of the user MS and the iSMS wireless modem).

Accessing the Web from GSM MSs

Based on iSMS, we have implemented a service that allows users to surf the Web using standard GSM MSs. The Web access architecture is illustrated in Fig. 6. In this architecture, a customer (i.e., a GSM MS holder) sends out a Web query in SMS format to a proxy server connected to the iSMS server. The proxy server and iSMS server may be running on the same host or on different hosts. Based on the customer's preference, the proxy maps the input query to a proper HTTP call and forwards the call to the corresponding Web server. The proxy then filters and converts data from the Web server to SMS format, and returns the SMS message to the original caller. This implementation has several interesting characteristics:

- Standard GSM MSs are used as terminals for surfing the Web. No special devices or software are needed.
- The communication medium is SMS. Both queries sent from the MSs (GSM customers) and data received by the MSs are in SMS format. A single URL access is performed by two SMS messages: one from the MS to the proxy containing a URL query and another from the proxy to the MS including the result data. The "instant transmission" property of SMS allows the customer to send an instant Web query without a long machine setup time as in dialup service to a PC.
- Functionality to map SMS queries to HTTP calls and converting data in HTML format to that in SMS format is implemented in the proxy. This application allows customers to tailor the call mapping and data conversion functions. The proxy picks the proper mapping and filter functions based on the customer's profile. The customers may also utilize the default mapping and conversion functions supported by the proxy.

The current iSMS version implements functions for querying Internet information, like stock quote, currency exchange rate, and delivery status of FedEx packages. The stock quote query command is of the form

```
QUO     {symbol1} {symbol2} …
```

The QUO command is explained with the following example:

```
QUO http://investor.msn.com/quotes/quotes.asp?
         Symbol=1/bin/quotefilter
```

The first field, QUO, is the key word for query. The second field, {symbol1}, specifies HTTP call mapping. The third field, {symbol2}, defines the filter/conversion function. If {symbol1} and {symbol2} are not specified, the default call mapping and conversion functions are used.

With the above setup, a customer enjoys Web querying through his/her GSM MS. Note that the customer can turn the MS off and then on, and the setup is still valid. Suppose the customer has a GSM MS with telephone number +886-936-105-401, and the MS modem in the iSMS gateway has the phone number +886-931-144-401. The customer sends out an SMS message to +886-931-144-401 as follows:

```
QUO T
```

This command queries AT&T stock. When the iSMS server (running on phone number +886-931-144-401) receives the message, it forwards the message to the proxy server, which in turn converts the SMS query into an HTTP call as follows:

```
http://investor.msn.com/quotes/quotes.asp?
Symbol=T
```

The proxy sends the HTTP call to the destination Web site. When receiving data from the Web site, the proxy invokes a filter function (i.e., `/bin/quotefilter`) that reformats the data received from the Web site. The proxy returns the output of the filter function to the customer's MS 0936-105-401. The SMS data looks like:

```
T Last 85 7/8
Change +1 9/16 (+1.85%)
Volume 7.708 M
```

where the first line indicates the last price of AT&T stock, the second shows the price change since the last closing, and the third gives the transaction amount. Consider a currency exchange rate query as another example. The currency rate query command issued by the customer is of the form

```
CUR from to
```

The system returns the currency rate between the currency of the country from and the currency of the country to. For example, the short message:

```
CUR USD TWD
```

queries the currency exchange rate between U.S. dollar and Taiwan dollar.

In the market, there are several proposed solutions for mobile phones to surf the Web:

- UP.browser from Unwired Planet allows accessing Web pages written in a special language different from the standard HTML. Content providers support two different formats of Web pages. Special terminals are used to access these services. AT&T PocketNet adopts this approach.
- Microsoft announced that they will implement a microbrowser running on phones to access the Web. In this approach, users establish data connection and log in to an ISP from cellular phones in order to access the Web, the same procedure users do from regular PCs/notebooks with an analog modem to dial into the telephone network. The system will run on Window CE devices.
- 3COM Palm VII introduced a notion Web Clipping where a set of predefined agents located on a Palm can interact with content providers, which also supports Web pages in a format different from HTML. It runs on top of Palm devices.
- Wireless Application Protocol (WAP) [15] is a protocol suite on top of mobile networks. New software has been developed for the MSs to browse Web pages in Wireless Markup Language (WML) format, which is a variation of HTML.

With certain modifications to the existing systems and handsets, the above approaches have been successful in various aspects. Compared with these solutions, our approach is transparent to Web sites and MSs.

Handset Music Service

Existing MSs can store several music tones in the memory. An MS may play various alert tones for incoming calls sent from different caller IDs (either mobile or fixed telephones). Teenager MS holders are particularly interested in having fancy music tones such as the latest pop hits. Nokia, for example, has developed a messaging protocol for this purpose. Based on SMS, the protocol, called Smart Messaging [4], specifies a set of predefined message headers for sending music tones, business cards, and so on. Each smart message contains a header for message type, and Nokia MSs recognize the header and perform different actions on the message body. Two issues regarding Smart Messaging deserve attention. First, each smart message contains a non-ASCII binary header, and it is not trivial to input a smart message from regular MSs. It may not be convenient to send a music tone from one MS to another. Second, only Nokia MSs understand how to interpret short messages in Smart Messaging format.

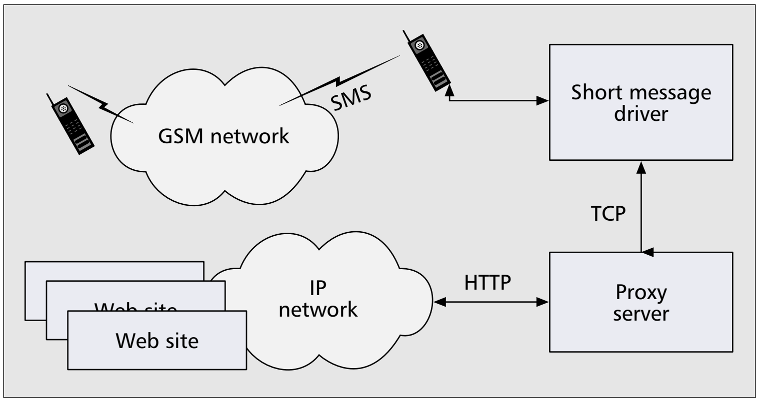
To address these issues, we have developed a Simple Tone Language (STL) to represent the music notes which are translated into Smart Message format by the iSMS music agent. More precisely, a GSM customer inputs a short message for the music tone request, containing the music agent name, a receiver of the music tone, and the music tone body in STL. The message is sent to the iSMS server. Upon receipt of the request, the music agent encodes the music notes into a short message and sends it to the receiver. The handset can then store and play the received music tone.

Other iSMS Services

In addition to services described in the previous sections, other examples of services running on the current version of iSMS include the following.

Train Schedule Service — This service allows mobile users to query a local train schedule via SMS. The query is based on destination, time period (for departure or arrival time), and train class. It's also possible to link this service to train reservation systems and to allow mobile users to book tickets. This service can also be generalized for other ticket services, such as airline reservation.

Personal Profile — iSMS maintains a personal profile repository for individual registered users. Profiles are organized on a per-user basis (according to phone number). Each entry has format "`keyword=value`" where `value` could be for phone number, address, personal note, or anything. A mobile user



■ Figure 6. *iSMS web accessing architecture.*

can add an entry to his/her personal profile by sending a short message to iSMS. For example, a mobile user may send the following short message to iSMS:

```
PB Robin +19179075010.
```

This message instructs the iSMS to add a new entry,

```
Robin=+19179075010,
```

to his/her profile. Then the user can query the entry with a keyword by sending the following message to iSMS:

```
PQ Robin.
```

In response, iSMS returns the following message:

```
Robin=+19179075010
```

A Web interface has been provided, and a user may update his/her profile using a regular browser as well. This repository serves not only as the mobile user's repository on the network, but also as the user profile for other services. For example, e-mail services use a user's profile to locate a sender's e-mail address by name.

Group Communications — iSMS implements a group communication mechanism. A group is identified by a unique name, and contains a set of phone numbers. A message sent to a group will be forwarded to all members in the group. iSMS supports the following group communication commands:

- Creating a group with founding members
- Querying members in a named group
- Adding/deleting members to/from an existing group
- Sending messages to a named group

This group mechanism can also be used by other services. For example, a multi-user game based on SMS may communicate with its mobile players using this group communication mechanism.

E-mail Service — iSMS implements an agent to SMTP/POP3 servers for relaying e-mail and SMS. A short message from a mobile user to iSMS with the following format:

```
EMA {e-mail-address} {message}
```

is transferred to an e-mail and sent to a SMTP server for delivery. On the other hand, the agent consults the mail server via a POP3 interface on behalf of a mobile user periodically, and delivers notifications of new messages to the user's mobile via SMS. The user reads an e-mail just like he/she reads a short message.

Conclusions

We propose iSMS, a platform that integrates IP networks with the short messaging mechanism of mobile networks. It supports a middleware for creating and hosting wireless data services based on SMS. The current implementation of iSMS integrates GSM and IP networks. The iSMS hardware architecture can easily be established with standard mobile stations and personal computers. We have developed agent-based middleware with an API, which results in a lightweight solution to allow quick deployment of value-added data services. Compared to solutions that integrate IP networks through a short message service center, iSMS has the following advantages:

- Easy installation. The iSMS hardware components are the standard GSM MS and the PC. The iSMS software can be installed on the PC without special treatment.
- Generic platform with personalization. Several applications have been developed for iSMS, including e-mail delivery/forwarding, Web access (e.g., stock and train schedule querying), and handset music service. iSMS users can easily develop new services and tailor the existing ones to fit their needs.
- Transparency. The iSMS services transparently run on GSM networks without any maintenance effort of GSM operators.
- Immediacy. Unlike dialup service to a PC, iSMS offers instant information exchange between the MS and the servers. This property is desirable for a customer to perform IP computing when he/she is moving.

For a GSM operator, iSMS is targeted for small companies that subscribe to the Closed User Group [1] service. In each of the small companies, a PC-based iSMS gateway is installed. The iSMS is designed for two types of services:

- **Type I:** existing Internet services
- **Type II:** customized services

For Type I services, an iSMS gateway serves as an access point for the GSM user to receive existing Internet services (e.g., stock quotes or Yahoo surfing). On the other hand, Type II services (e.g., private mail) are tailored for each iSMS gateway. This type of services is exclusive to a closed user group of an iSMS gateway, which cannot be accessed by users outside the gateway (although different CUGs may have the same customized servcies). For both Type I and II services, the iSMS gateways do not communicate with each other. Thus, there is no need to manage and administer the distributed iSMS gateways. Our iSMS gateway approach has several advantages.

Scalability — We can easily accommodate new iSMS customers by installing PC-based iSMS gateways at their company sites. Since the radio paths for these iSMS gateways are different, adding new customers does not increase traffic to a specific radio link. It is clear that our iSMS approach is scalable for Type I services. To accommodate more users for Internet services, we need only add more iSMS gateways. For other approaches such as WAP, all access to the Internet should go through a centralized WAP gateway that may need great effort for maintenance, and may become a bottleneck. In iSMS, access to the Internet is performed through thousands of routes using independent iSMS gateways.

Reliability — When an iSMS gateway at a company fails, we need only fix this failed gateway. This failure does not affect other iSMS gateways. To enhance the reliability of iSMS service in a company, we can have a duplicate configuration on an iSMS gateway site. In this configuration, an MS (the customer) may register to an operational iSMS gateway and a standby iSMS gateway. This setup is performed at gateway initialization (when the customer subscribes to iSMS). For a standard GSM MS, the phone numbers of both gateways are stored in the phone book of the MS. The MS first tries the operational iSMS gateway for service. If it does not work, the MS tries the standby iSMS gateway. Message resending can be automatically performed by the GSM divert service. If the GSM MS is equipped with the SIM TookKit feature, when the operational gateway fails, the standby gateway will instruct the MS to switch the gateway over the air (by modifying the gateway phone number stored in the MS). When the operational gateway is restored, the gateway phone number in the MS is switched back by the operational gateway over the air. Gateway restoration is transparent to users.

Performance — Since customer traffic is distributed to individual iSMS gateways, good performance (e.g., short latency time) can be expected.

Due to limited SMS bandwidth, most iSMS applications are transaction-oriented. When high-bandwidth mobile data infrastructures (e.g. General Packet Radio Service [16] and third generation wideband wireless systems [17]) are available, the data rate of iSMS can be significantly increased and development of general data services will be essential.

Acknowledgment

We would like to thank the three reviewers for their valuable comments. Their suggestions have significantly improved the quality of this article. Lin's work was sponsored in part by MOE Program of Excellence Research under contract 89-E-FA04-4, FarEastone, National Science Council under contract NSC 89-2213-E-009-203, the Lee and MTI Center for Networking Research, NCTU.

References

[1] Y. B. Lin and I. Chlamtac, *Mobile and Wireless Network Architectures*, Wiley, 2001.

[2] ETSI/TC. "Technical Realization of the Short Message Service Point-to-Point," v. 4.6.0. Tech. rep., Rec. GSM 03.40, 1993.

[3] ETSI/TC, "Use of Data Terminal Equipment-Data Circuit Terminating; Equipment (DTE-DCE) Interface for Short Message Service (SMS) and Cell Broadcast Service (CBS)," v. 5.3.0, Tech. rep., Rec. GSM 07.05, 1997.

[4] Nokia, "Smart Messaging Specification," v. 1.0.0.Tech. rep., Nokia, 1997.

[5] EIA/TIA, "800 MHz TDMA Cellular Radio Interface — Mobile Station — Base Station Compatibility — Digital Control Channel," Tech. rep. IS-136, 1994.

[6] EIA/TIA, "Mobile Station-Base Station Compatibility Standard for Dual-mode Wideband Spread Spectrum Cellular System," Tech. Rep. IS-95, 1993.

[7] ETSI/TC, "Alphabets and Language-Specific Information. Technical Specification," v. 5.6.0, phase 2+, Tech. rep. Rec. GSM GSM 03.38, 1997.

[8] T. Y. C. Woo, T. F. LaPorta, and K. K. Sabnani, "Pigeon: A Wireless Two-way Messaging System," *IEEE JSAC*, vol. 15, no. 8, 1997, pp. 1391–1405.

[9] http://www.esms.com

[10] http://www.smsnordic.com

[11] http://www.geonet.de

[12] http://www.fenix.co.uk/Marketing/id/16.htm

[13] Ericsson, http://mobile.ericsson.com/

[14] http://www.bluetooth.com

[15] http://www.wapforum.org

[16] ETSI/TC, "GPRS Service Description Stage 2," v. 7.0.0, phase 2+, Tech. rep. Rec. GSM 03.60, 1998.

[17] T. Ojanpera and R. Prasad, "An Overview of Third-Generation Wireless Personal Communications: A European Perspective," *IEEE Pers. Commun.*, vol. 5, no. 6, Dec. 1998, pp. 59–65.

Biography

HERMAN CHUNG-HWA RAO (herman@research.att.com) is a principal member of technical staff in AT&T Laboratories-Research. His research interests are in the areas of the Internet, distributed file systems, distributed systems, mobile computing, operating systems, computer networks, and software tools. In particular, he has been involved in the design and implementation of an Internet file system, Middleware for Web Client, and GRADIENT (Global Research and Development Envlronment). He received his Ph.D. and M.S. in computer science from the University of Arizona and his B.S. in mechanical engineering from National Taiwan University.

DI-FA CHANG (difac@usc.edu) is a Ph.D. student in the Computer Science Department, University of Southern California. He received his B.S. and M.S. from the Department of Computer Science and Information Engineering, National Taiwan University, in 1994 and 1996, respectively. From 1998 through 1999 he was a research engineer at FarEasTone Telecommunication Co., Taiwan. His current research interests include network dynamics, simulation of networking protocols, and wireless ad hoc routing.

YI-BING LIN (liny@csie.nctu.edu.tw) received his B.S.E.E. degree from National Cheng Kung University in 1983, and his Ph.D. in computer science from the University of Washington in 1990. From 1990 to 1995, he was with the Applied Research Area at Bell Communications Research (Bellcore), Morristown, New Jersey. In 1995 he was appointed a professor of the Department of Computer Science and Information Engineering (CSIE), National Chiao Tung University (NCTU). In 1996 he was appointed deputy director of the Microelectronics and Information Systems Research Center, NCTU. During 1997–1999, he was elected chair of CSIE, NCTU. His current research interests include design and analysis of personal communications services networks, mobile computing, distributed simulation, and performance modeling. He is an associate editor of *IEEE Network*, an editor of *IEEE J-SAC: Wireless Series*, an editor of *IEEE Personal Communications*, an editor of *Computer Networks*, an area editor of *ACM Mobile Computing and Communication Review*, a columnist for *ACM Simulation Digest*, and an editor of *International Journal of Communications Systems*, *ACM/Baltzer Wireless Networks*, *Computer Simulation Modeling and Analysis*, and *Journal of Information Science and Engineering*. He is a co-author with Imrich Chlamtac of the book *Wireless and Mobile Network Architecture* (Wiley). He received 1998 and 2000 Outstanding Research Awards from the National Science Council, ROC.