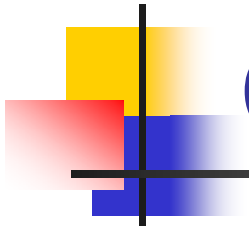
Chapter 1
Short Message Service and IP Network Integration

Prof. **Yuh-Shyan Chen**
Department of Computer Science and Information Engineering
National Taipei University

Abstract

- **Chapter 1** presents two platforms, NCTU-SMS and iSMS, that integrate IP networks with the short message mechanism of mobile networks.
  - The NCTU-SMS platform was developed at National Chiao Tung University (NCTU).
  - The iSMS platform was developed by AT&T and FarEasTone. iSMS supports a middleware for creating and hosting wireless data services based on SMS. The iSMS hardware architecture can be easily established with standard Mobile Stations (MSs) and personal computers.
  - They have developed agent-based middleware with API, which results in a lightweight solution to allow quick deployment of added-value data services in iSMS.

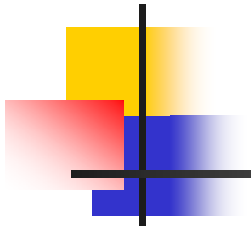


Outline

- 1.1 SMS-IP Integration with SM-SC
  - 1.1.1 NCTU Short Message System
  - 1.1.2 Statistics for SMS Delivery
- 1.2 iSMS System Architecture
- 1.3 iSMS Communication Protocols
  - 1.3.1 SMS AT Command Set
  - 1.3.2 iSMS Communication API
  - 1.3.3 Implementation of an Echo Server
- 1.4 Examples of Services
  - 1.4.1 Accessing the Web from GSM MSs
  - 1.4.2 Handset Music Service
  - 1.4.3 Train Schedule System
  - 1.4.4 Other iSMS Services
- 1.5 Caching for iSMS-Based Wireless Data Access

Short Message Service and IP Network Integration

- ***Short Message Service* (*SMS*)** is a mature wireless communication service.
- Most modern digital cellular phone systems offer SMS, which is considered a profitable added-value service.
- A natural extension is to integrate SMS with electronic mail services, which provides linkage between mobile networks and IP networks.
  - Furthermore, several Internet applications over SMS can be implemented on similar platforms.
- This chapter uses the *Global System for Mobile Communications* (*GSMC*) SMS as an example to illustrate SMS-Internet integration.



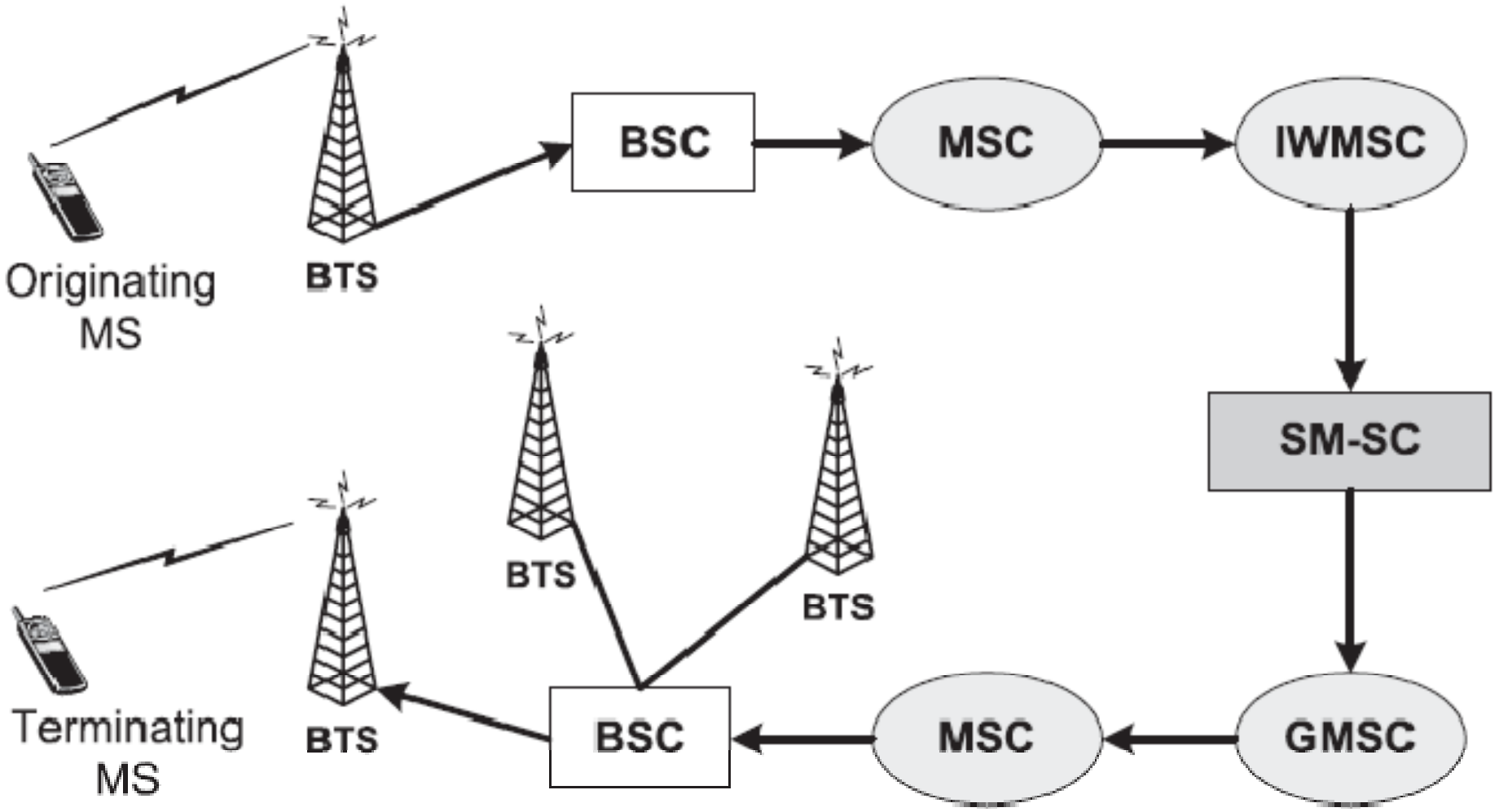
Introduction

- 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 the GSM default alphabet
  - The short messages are transported on the GSM*Stand-alone Dedicated Control Channel* (*SDDC*)
  - Since a voice session utilizes GSM radio traffic channels, short messages can be received and sent while the mobile users are in conversation.

The GSM SMS network architecture (Fig. 1.1)

- Mobile Station (MS) sends a short message, this message is delivered to the GSM radio system
  - That is, a *Base Transceiver Station* (*BTS*) and then a *Base Station Controller* (*BSC*).
- The radio system then forwards the message to a *Mobile Switching Center* (*MSC*) called SMS *Inter-Working MSC* (*IWMSC*).

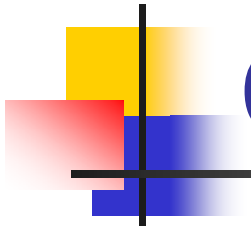
Fig. 1.1 GSM Short Message Service Network Architecture



The GSM SMS network architecture

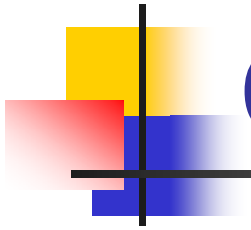
- The IWMSC passes this message to a *Short Message Service Center* (*SM-SC*).
  - Upon receipt of a short message, the SM-SC may send an acknowledgment back to the originating MS (sender) if the acknowledgment request is specified in the short message.
- The SM-SC then forwards the message to the destination GSM network through a specific GSM MSC called the SMS *Gateway MSC* (*SMS GMSC*).

- Following the GSM roaming protocol (see Section 9.2.1), 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 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 time stamp, and the length of the message body.
  - *Mobile Station ISDN Numbers* (*MSISDNs;* the GSM telephone numbers), are used for addressing.



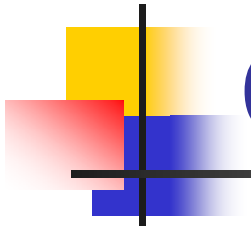
Cont.

- As a wireless data service, SMS has distinct features, such as handset alert capability and support for ME-specific and SIM-specific data.
  - SMS provides message storage when the recipient is not available.
  - It also allows simultaneous transmission with GSM voice, data, and fax services.
  - SMS has drawbacks, such as narrow bandwidth and long latency of end-to-end transmission.



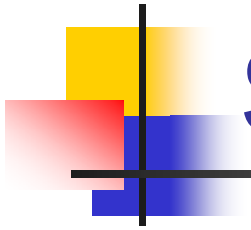
Cont.

- To design an SMS-IP system, the aforementioned SMS strengths and limitations cannot be ignored.
- The IP server itself may also be mobile
  - for example, the iSMS gateway described in this chapter can move when it provides services.



Cont.

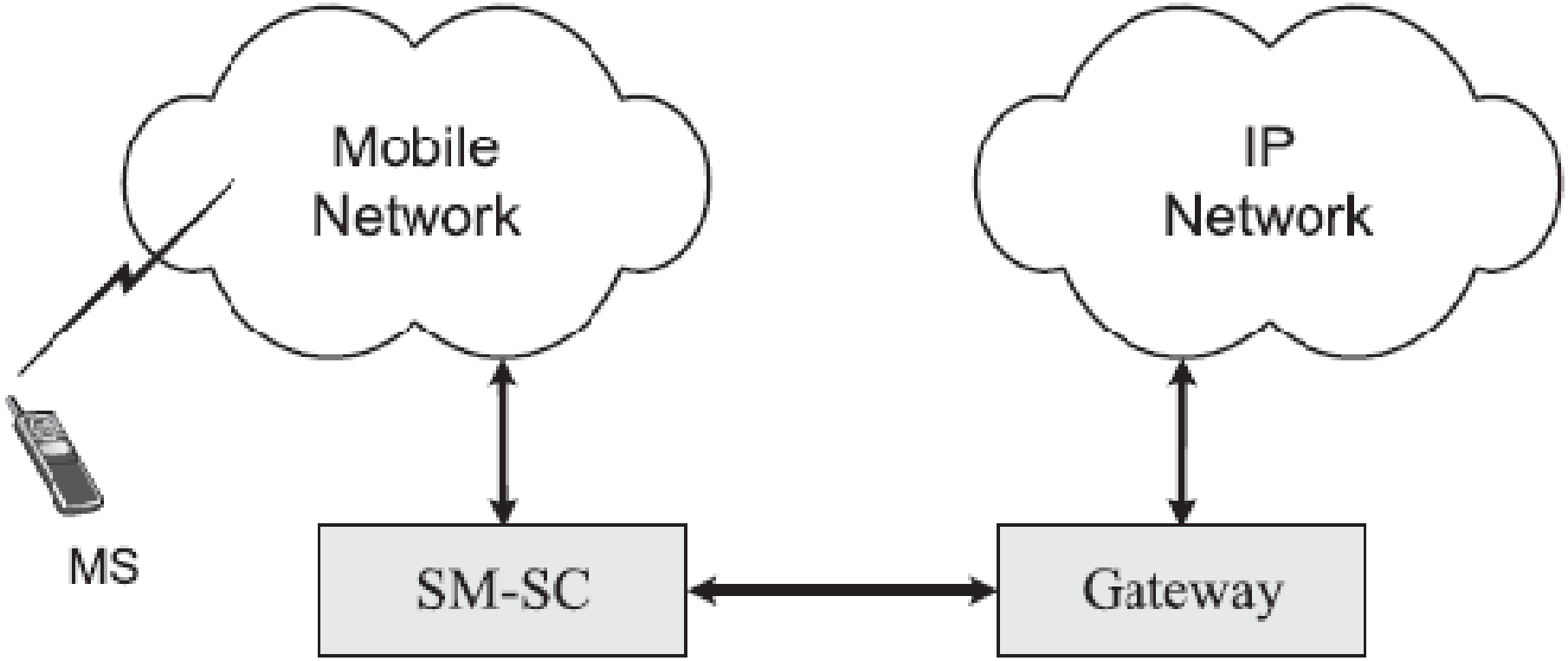
- To facilitate the development of various SMS applications over the Internet, we need a generic gateway to interwork the GSM network and IP network, and many specific data format converters between IP application contents and the SMS byte codes.

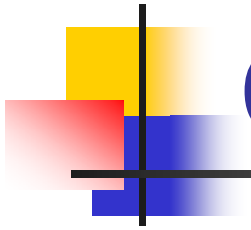


SM-SC (Fig. 1.2)

- In most commercial implementations, SMS and IP networks are integrated through SM-SC.
  - 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 SM-SC implementation is vendor-specific, the SM-SC-based SMS-IP integration solution depends heavily on the SM-SC vendors.

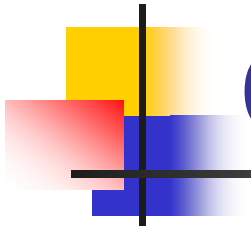
Fig. 1.2 SMS-IP Integration with SM-SC





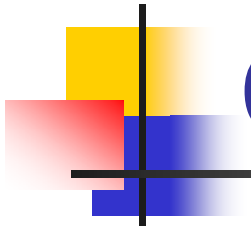
Cont.

- The SMS-IP gateway is maintained and controlled by GSM operators.
  - From the GSM operator's viewpoint, maintaining a reliable, secure, scalable interconnection platform between individual service providers and the 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.



Cont.

- To address these issues and to further support an environment for quick prototyping and hosting wireless data service,
  - an endpoint SMS-IP integration solution called iSMS was proposed in [Rao01a].
  - This solution is transparent to an existing SM-SC and GSM network.



Cont.

- This chapter first describes a SM-SC-based SMS-IP system called NCTU-SMS.
- Then we introduce iSMS, a non-SM-SC based SMS-IP system, and elaborate on the designs and implementations of several iSMS applications.

1.1 SMS-IP Integration with SM-SC

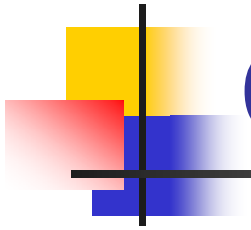
- The NCTU-SMS platform was developed at National Chiao Tung University (NCTU).

NCTU-SMS System

- In a joint project between the FarEasTone Telecommunications Corporation and National Chiao Tung University (NCTU),
  - They developed a web-based short message system called NCTU-SMS [Hun 04].
  - NCTU-SMS integrates SMS with IP through SM-SC.
  - With a user-friendly *Graphical User Interface* (*GUI;* Figure 1.3),

Fig. 1.3 The Web-based Graphical User Interface for NCTU-SMS

(1) Personal data (2) Calendar (3) Shared disk (4) Phonebook (5) Message delivering options (6) Logout (7) Destination phone number (8) Message text



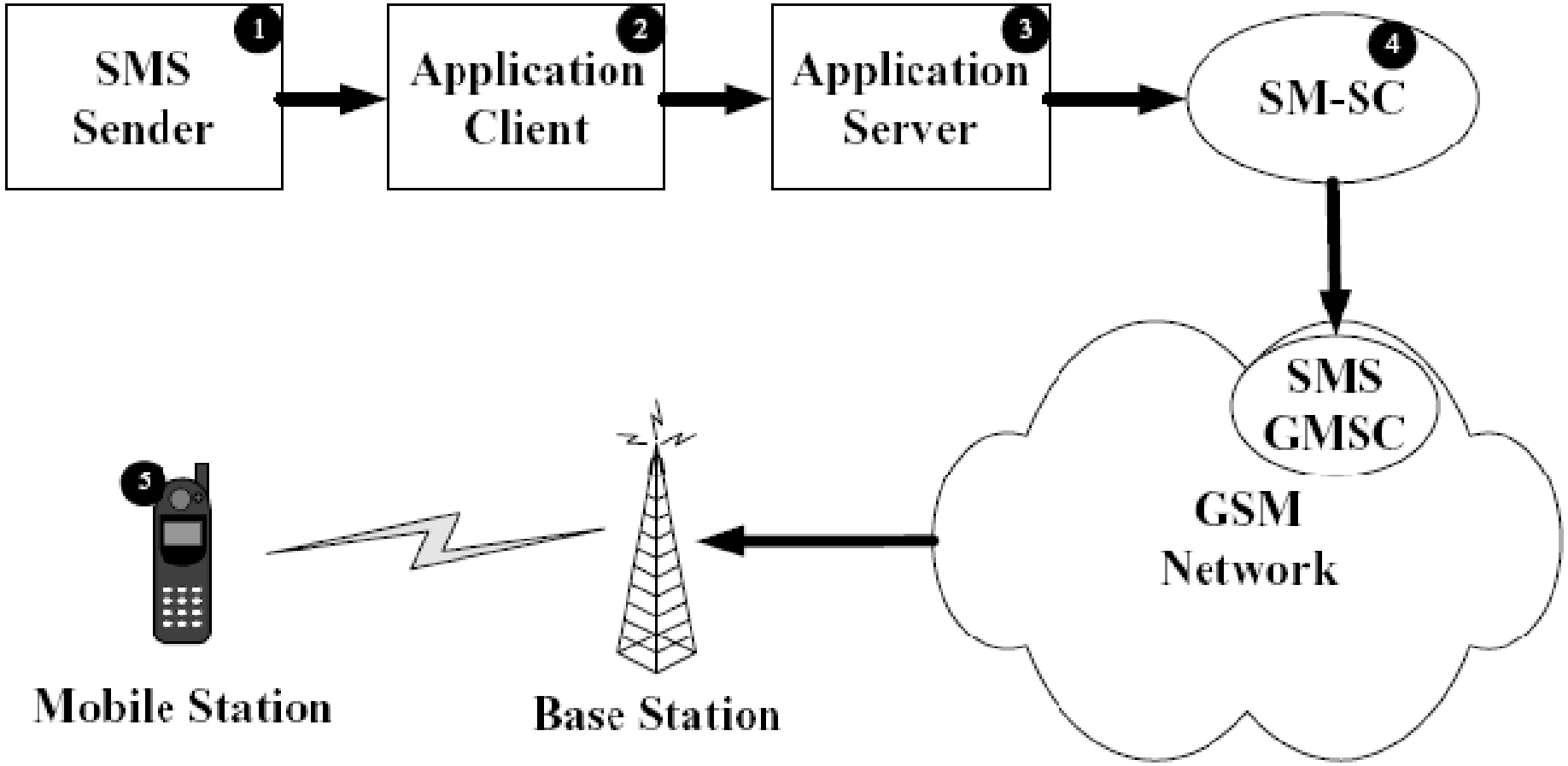
Cont.

- NCTU-SMS allows users to input short messages through a web site.
- At NCTU, students who take the personal communication course receive their test scores and final grades through short messages generated by NCTU-SMS.

1.1.1 NCTU Short Message System

- The NCTU-SMS architecture is illustrated in Figure 1.4, and consists of an *Application Client* (*AC*), an *Application Server* (*AS*), and the SM-SC.
- A delivery state tree (see Figure 1.5) is used to indicate the status of every short message delivery.
  - The delivery state of a short message is recorded in the Application Client.

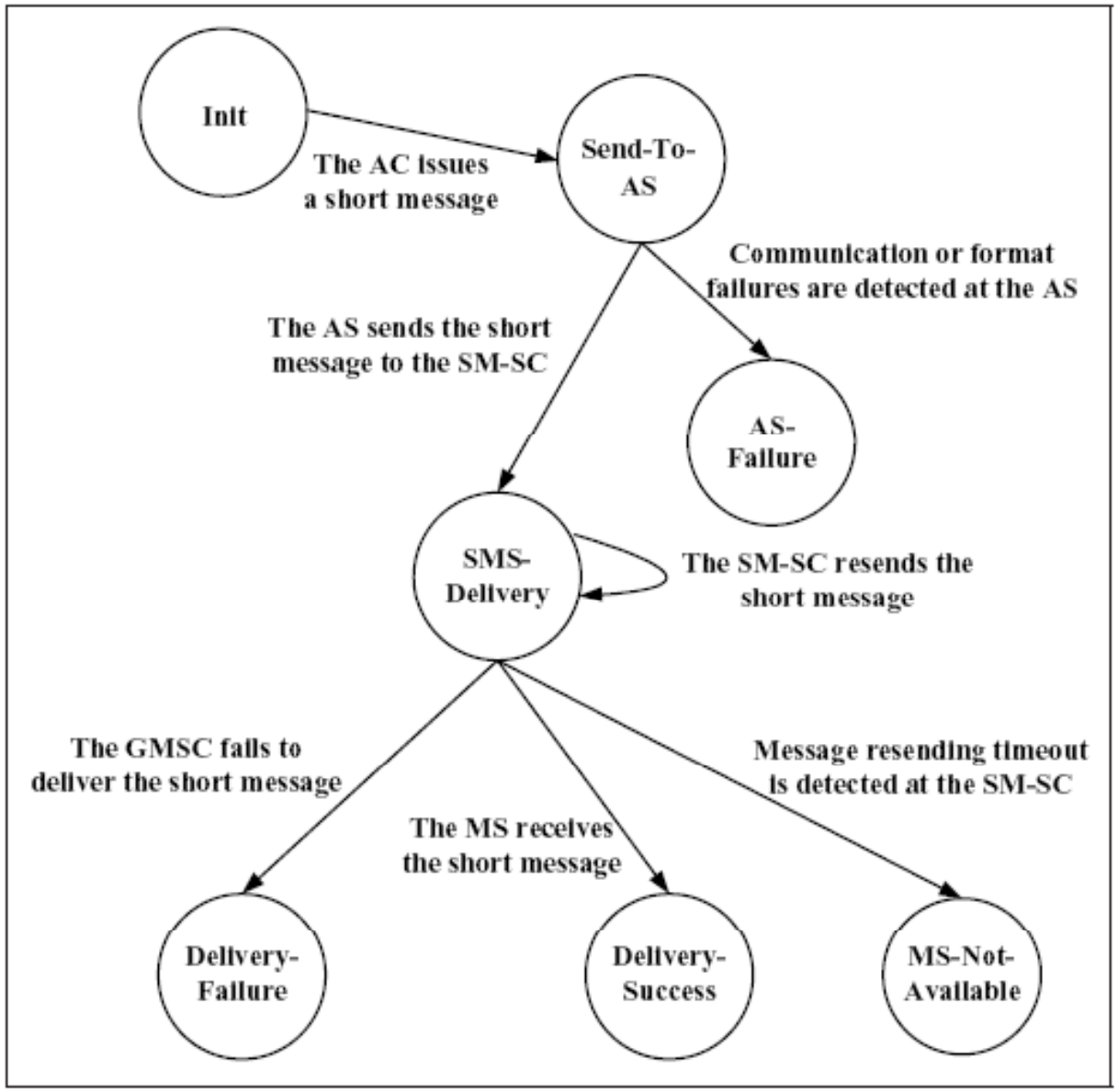
Fig. 1.4 The NCTU-SMS Architecture



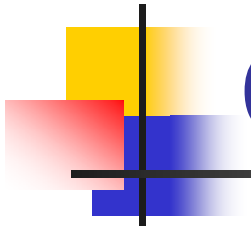
The procedure of sending a short message

- **Step 1.**
  - A user (Figure 1.4 (1)) issues a short message through the web based GUI. Specifically, the user inputs the destination phone number (MSISDN) and types the message text. When the user presses the OK button, the Application Client (Figure 1.4 (2)) generates a message delivery record that records.
    - the destination MSISDN
    - the time *Ti* when the message is issued,
    - the current delivery state,
    - the time *Ts* associated with the delivery state, and
    - other parameters.
- At this step, the delivery state is **Init** and $Ts = Ti$.

Fig. 1.5 The Delivery State Tree

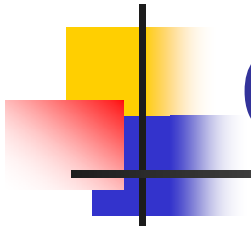


AC: Application Client
AS: Application Server



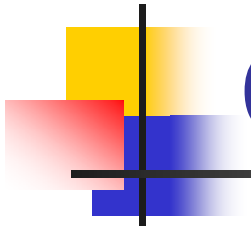
Cont.

- **Steps 2 and 3.**
  - The Application Client forwards the short message to the Application Server (Figure 1.4 (3)), and the delivery state is changed to **Send-To-AS**.
  - If the communications link between the Application Client and the Application Server is disconnected, the Application Client sets the delivery state to **AS-Failure**.



Cont.

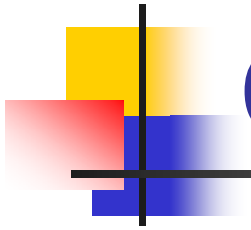
- Otherwise, the Application Server receives the message delivery request from the Application Client and performs format checking (for example, the destination phone number format, the message text format, and so on).
  - If the format check fails, an error message is sent back to the Application Client. In this case, the Application Client updates *Ts* to the current time, the delivery state is set to **AS-Failure**, and the procedure exits.



Cont.

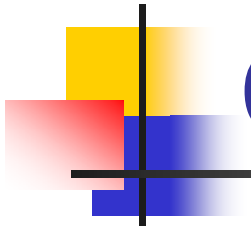
- If the format check succeeds, then an acknowledgment message is sent back to the Application Client, and the Application Server forwards the short message to the SM-SC (Figure 1.4 (4)). When the Application Client receives the acknowledgment, it updates *Ts* and sets the delivery state to **SMS-Delivery**.

- **Step 4.**
  - When the SM-SC receives the short message, it delivers the message following the standard SMS procedure illustrated in Figure 1.1.



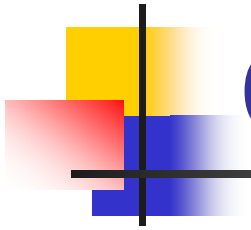
Cont.

- Note that the message may not be actually delivered if errors occur
  - for example, the mobility database *Home Location Register* (*HLR;* see Chapter 2 and Section 9.2.1) cannot identify the destination MS.
  - In this case, an error message is sent back to the Application Client.
  - The delivery state is set to **Delivery-Failure**, and the procedure exits.



Cont.

- **Step 5.**
  - When the destination MS receives the short message, an acknowledgment is sent back to the Application Client.
  - The delivery state is set to **Delivery-Success**, and the procedure exits.
    - Sometimes the short message may not be received by the destination MS
      - (for example, the destination MS is power off).



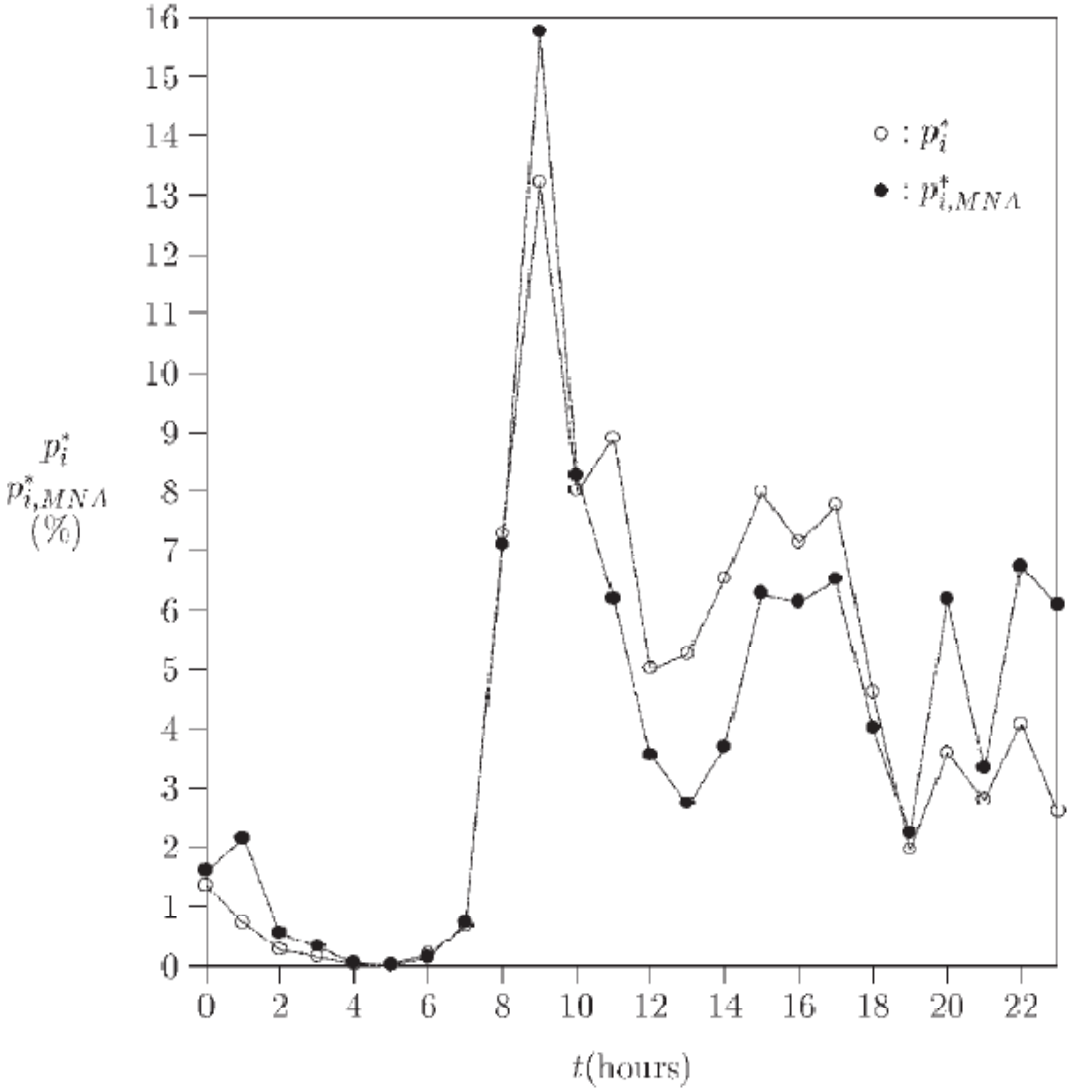
Cont.

- In this case, the short message will be periodically resent until either the destination MS receives the message or the SMSC gives up.
- In the latter case, an error message is sent back to the Application Client. The Application Client updates *Ts*, sets the delivery state to **MS-Not-Available**, and exits the procedure.

Statistics for SMS Delivery

- Several interesting statistics can be derived from the delivery records in the Application Client—for example, the hourly distribution $Ti*$ of Ti (the time when a short message is issued), where
  - $Ti* = Ti \bmod 24$.

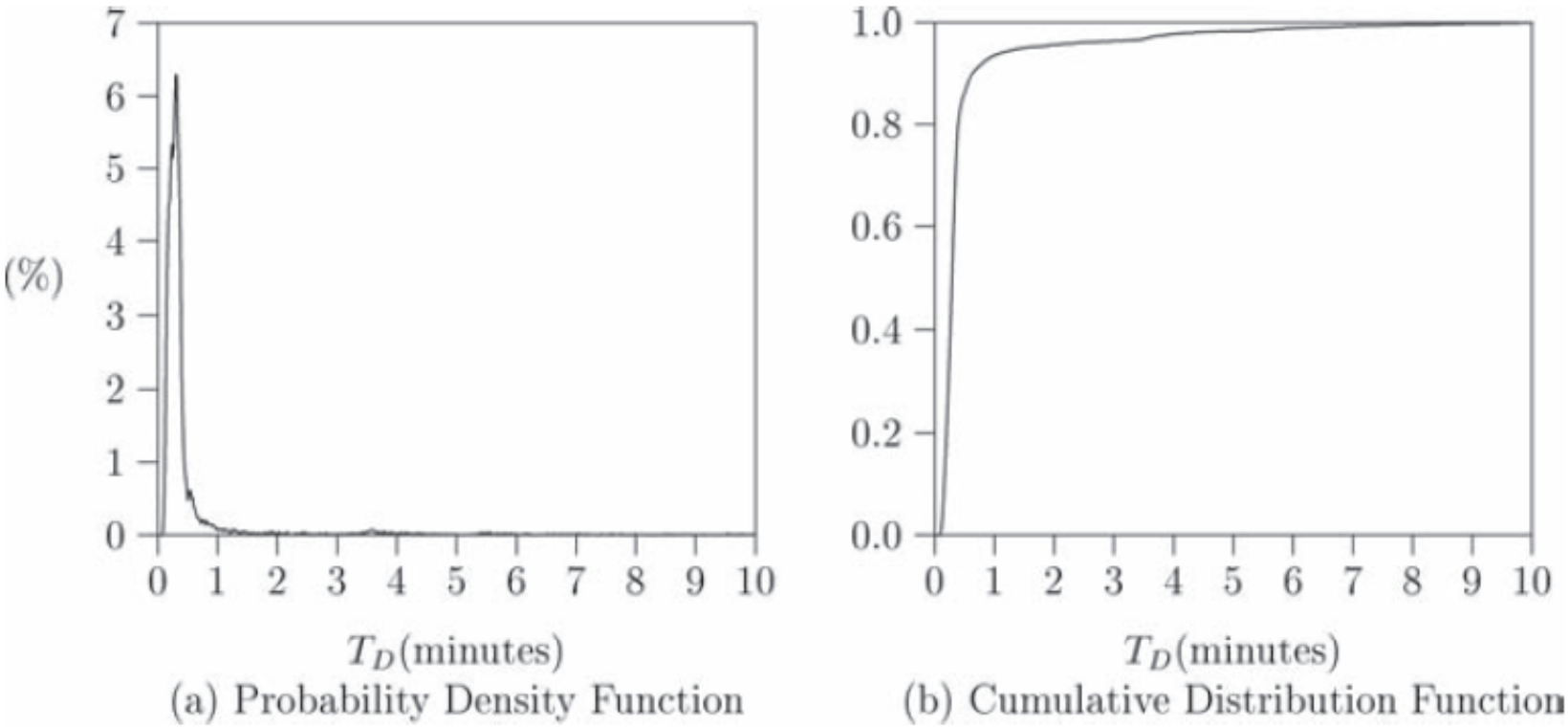
The T_i^* and $T_i^{*,MNA}$ Distributions



Numbers of deliveries, with various final states

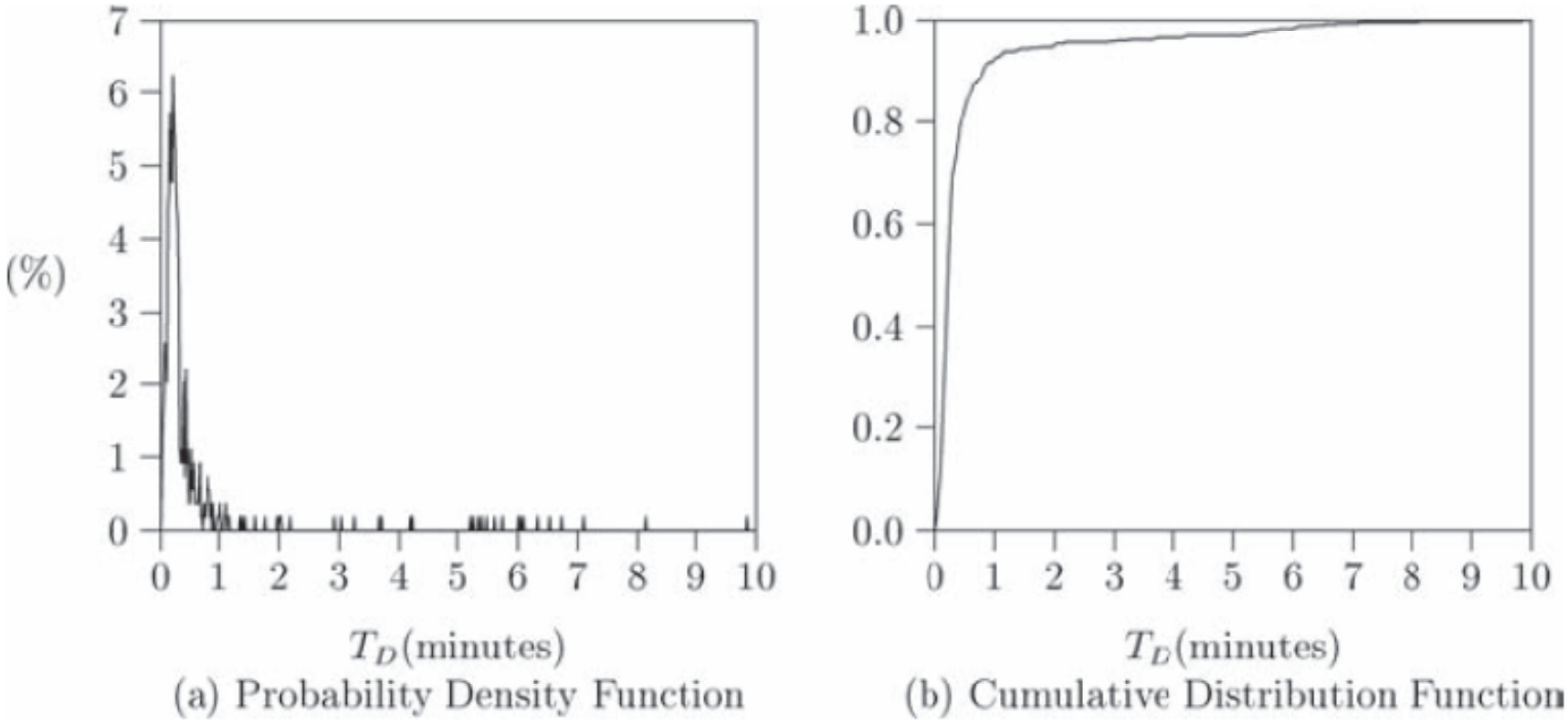
STATE	AS-FAILURE	DELIVERY-FAILURE
Number	423	1649
Percentage	0.92 %	3.57 %
STATE	**MS-NOT-AVAILABLE**	**DELIVERY-SUCCESS**
Number	4278	39,780
Percentage	9.27 %	86.23 %

The T_D Distribution

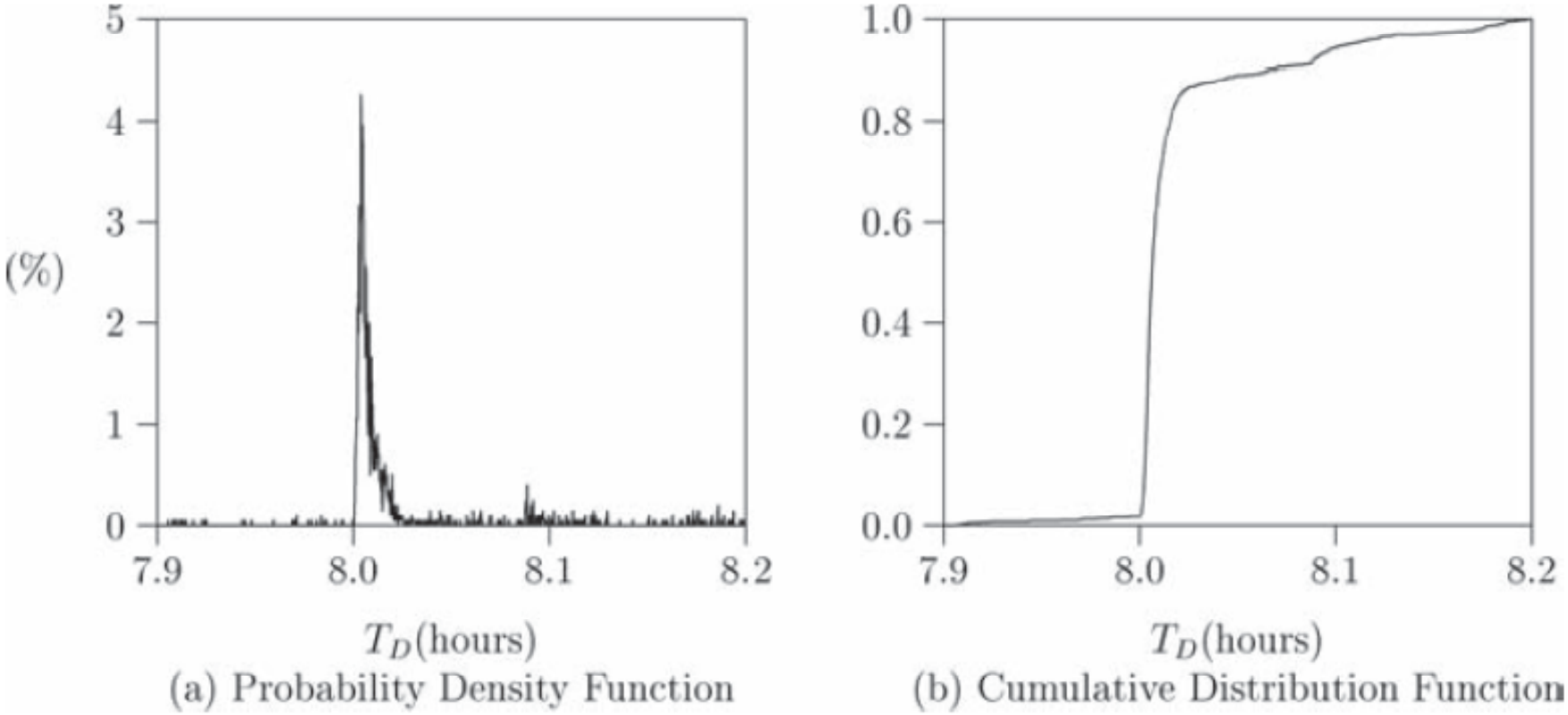


Delivery-Success; the normalized factor is 1.1

The T_D Distribution for **Delivery-Failure**



The T_D Distribution for **MS-Not-Available**



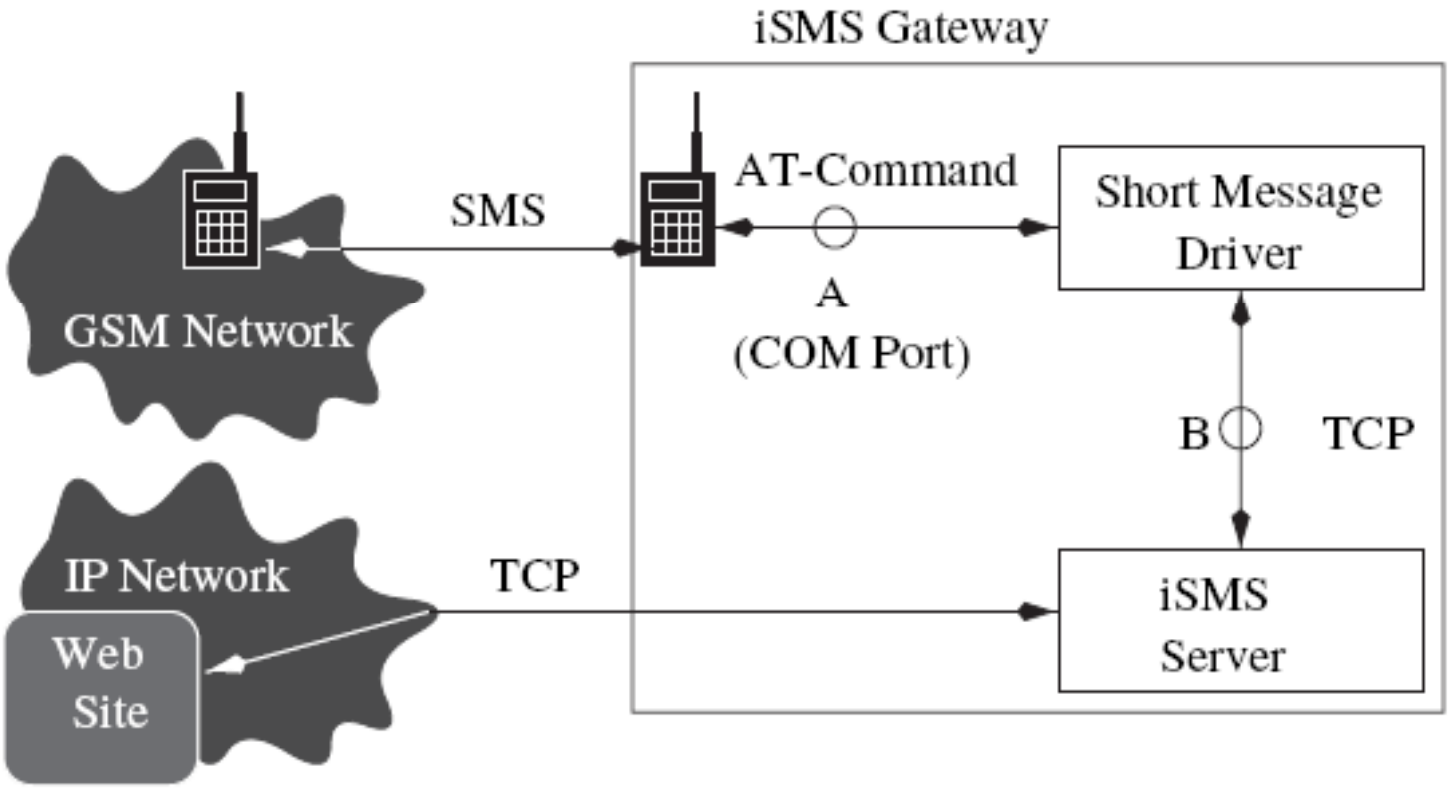
1.2 iSMS System Architecture

The iSMS platform was developed by AT&T and FarEasTone.

iSMS system

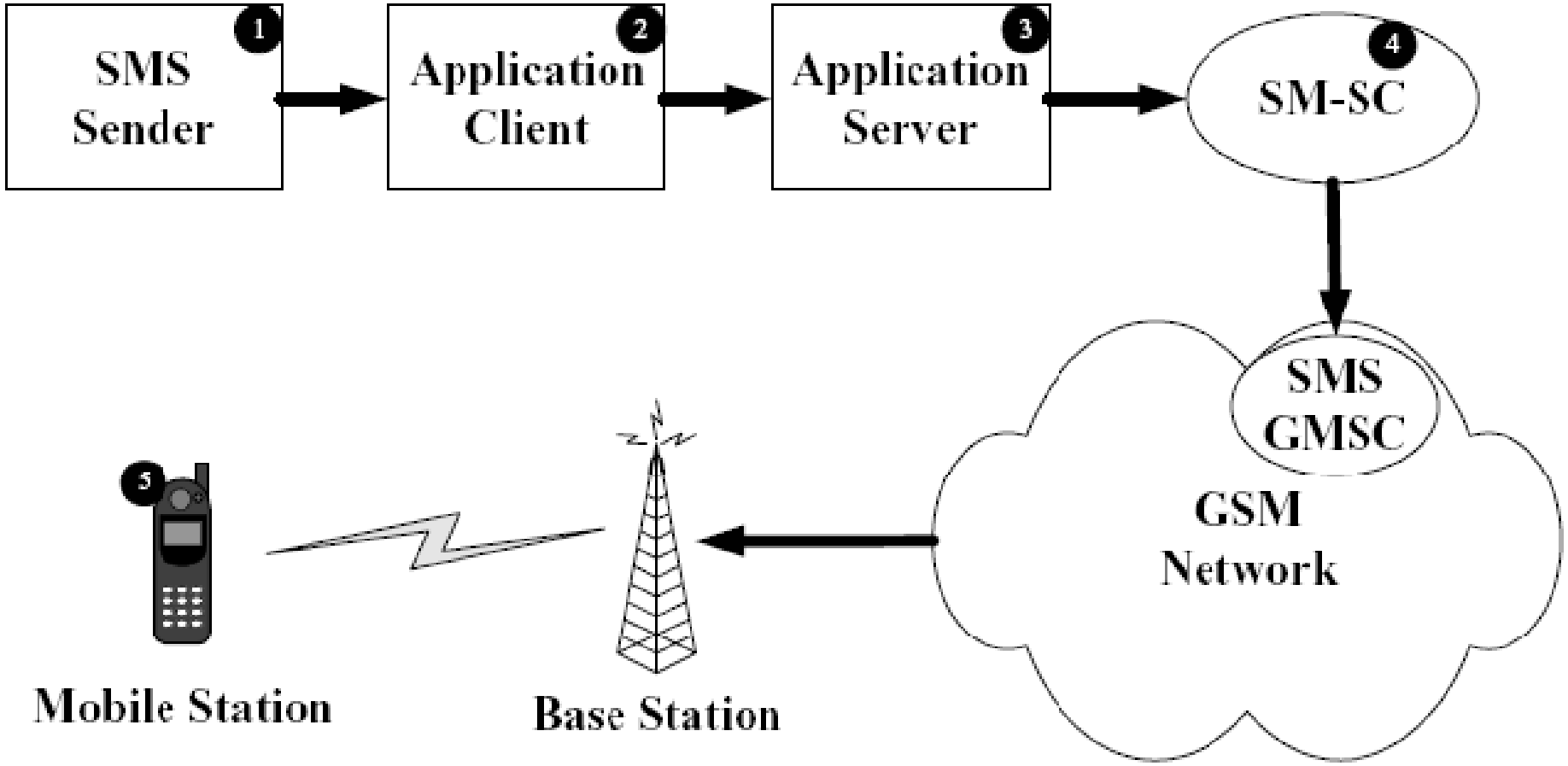
- Figure 1.10 illustrates the architecture for iSMS, a non-SM-SC based SMS-IP integration solution.
  - *iSMS gateway* is introduced.
  - No components in the GSM and IP networks are modified.
  - The MS is a commercial handset product that does not require installing any new software.

Fig. 1.10 iSMS System Architecture



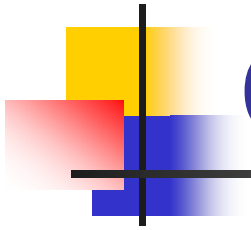
An End point SMS-IP Integration Solution

Fig. 1.4 The NCTU-SMS Architecture



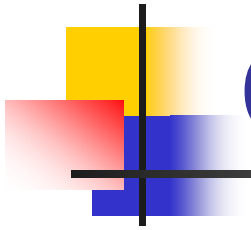
Cont.

- A major difference between the iSMS architecture and the SM-SC based architecture 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.*
  - the iSMS gateway is a PC running on Windows or UNIX operating systems.
  - The PC-based gateway can be a desktop with high reliability and availability.



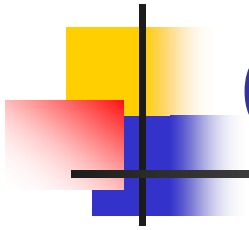
Cont.

- The PC can be replaced by a notebook.
  - In this case, the gateway becomes a mobile server, which may move around just like a GSM MS.
  - The MS modem and the iSMS gateway can be connected by the serial port, the Infrared port (for example, IrDA), the USB interface, or the PCMCIA interface.



Cont.

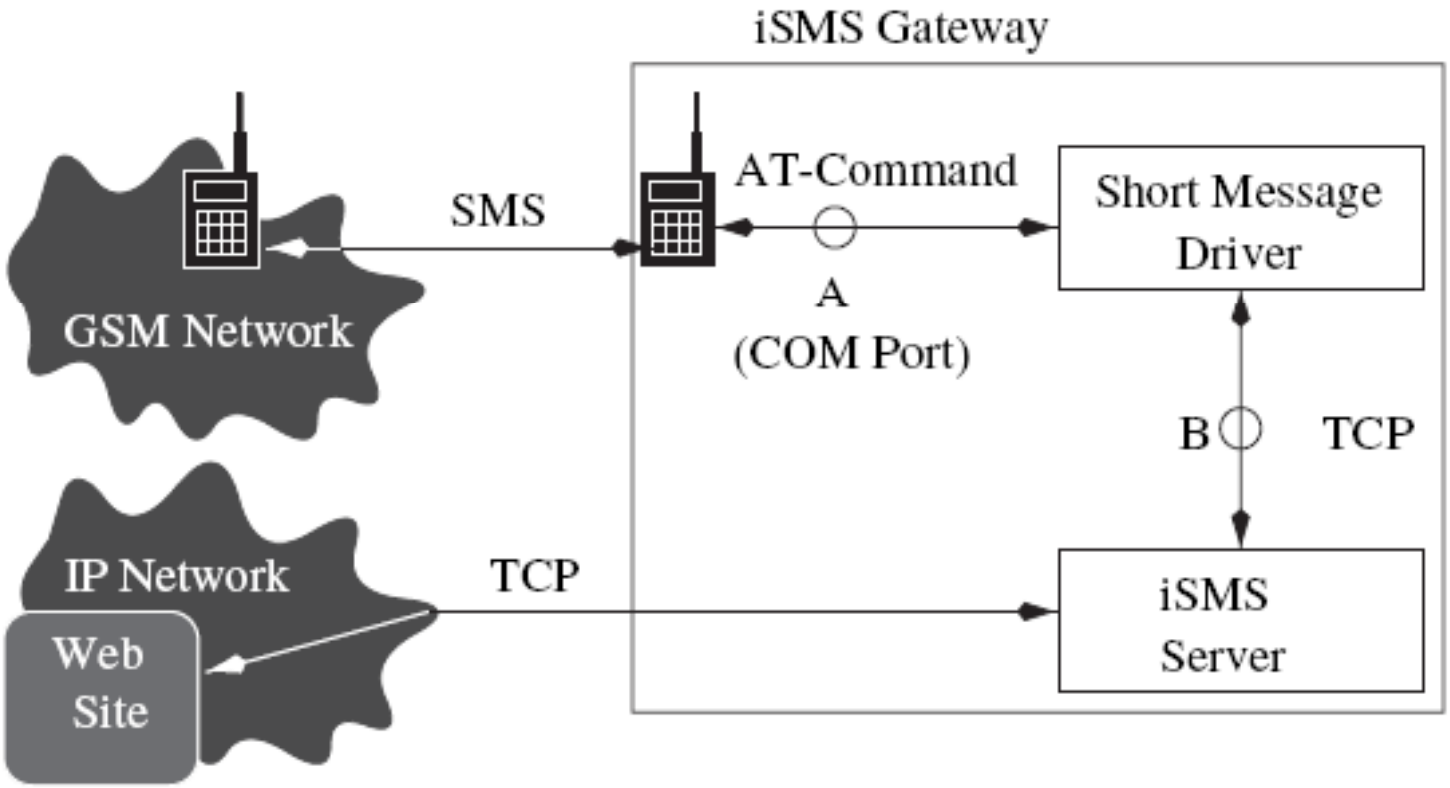
- In iSMS, the data sent from the IP network to the GSM network is automatically packaged into short messages.
  - The short messages can be multicast up to 65,535 receivers.
- An MS can broadcast short messages to several servers connected to the iSMS gateway.
  - The iSMS system is identified by the 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.



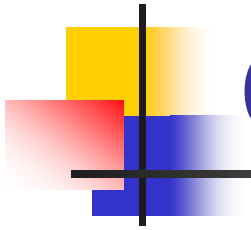
Cont.

- 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
  - **Short message driver:** Responsible for communication between the GSM network and iSMS servers in the IP networks

Fig. 1.10 iSMS System Architecture

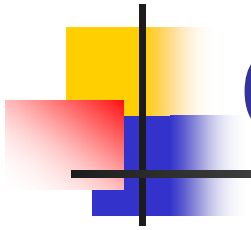


An End point SMS-IP Integration Solution



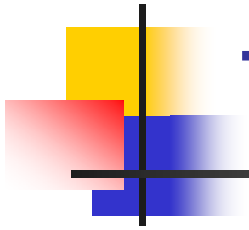
Cont.

- The communication protocol between the MS modem and the short message driver (the reference point **A** in Figure 1.10) is implemented by using the SMS *AT Command Set*
- The communication functions between the iSMS servers and the short message driver (the reference point **B**) is implemented through the iSMS communication *Application Programming Interface* (*API*) based on the TCP socket.



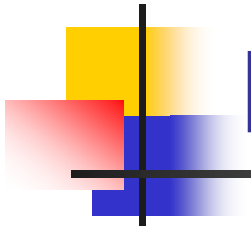
Cont.

- The short message driver receives incoming short messages from the serial port and passes these messages to the iSMS servers according to a *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.



To develop a new service

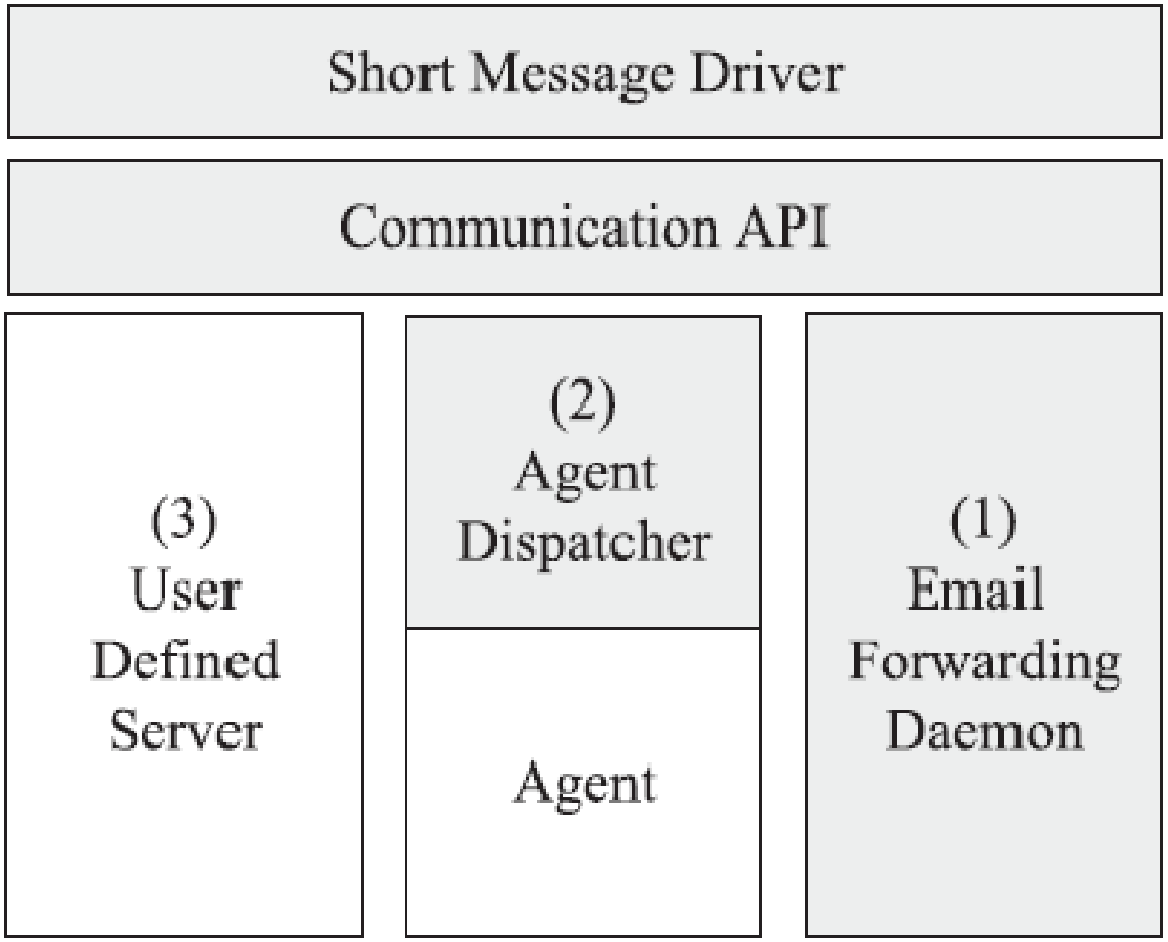
- One implements iSMS servers that communicate with the short message driver by using the functions defined in the iSMS communication API
  - Email Forwarding Daemon
  - Agent Dispatching Server

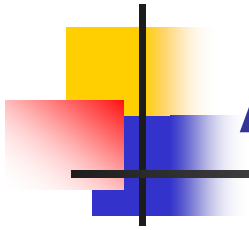


Email Forwarding Daemon

- The Email Forwarding Daemon ((1) in Figure 1.11) relays messages between the MSs and email systems on the IP domain.
  - It supports an interface to Microsoft Exchange Server as well as to standard *Simple Mail Transfer Protocol* (*SMTP*) and Post Office Protocol Version 3 (POP3)
  - The daemon converts a short message sent from an MS to email and forwards it to a SMTP server for delivery. The daemon may periodically query a mail server (for example, via a POP3 interface), pick up important emails (according to user profiles), and send out SMS notifications to users'
  - mobile stations.

Fig. 1.11 iSMS Driver and Server Structure





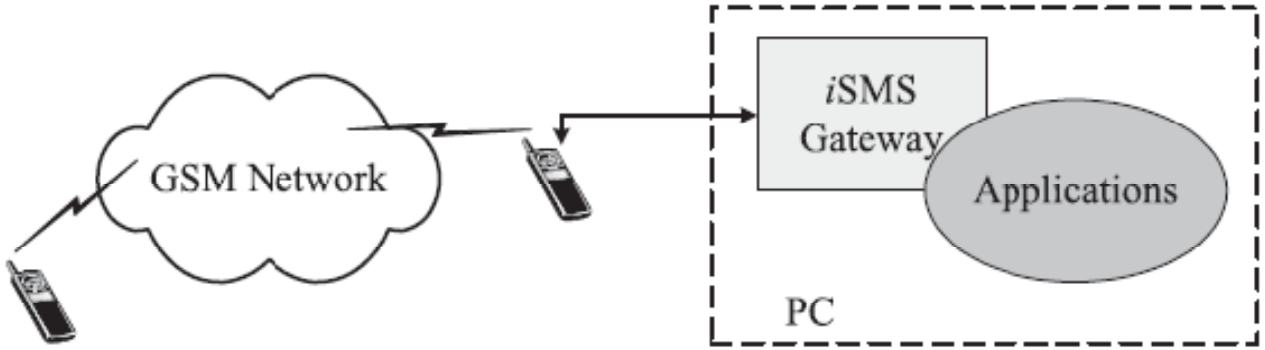
Agent Dispatching Server

- A server of an agent-dispatching type ((2) in Figure 1.11) 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 parameter to the agent.
  - Each agent implements one function. When the agent finishes processing of a message, the agent dispatcher collects the results and sends them back to the short message driver.

Cont.

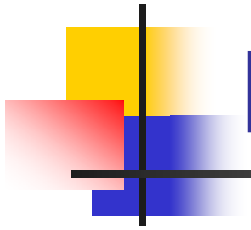
- If all iSMS applications and the iSMS gateway are implemented in a portable notebook that is not connected to any IP network, then the iSMS system becomes a mobile server, as illustrated in Figure 1.12 (a).
  - In this configuration, mobility management of the iSMS server is automatically maintained by the GSM Mobile Application Part
  - In other words, the existing SM mechanism will transparently track the moving iSMS server, and iSMS does not need to implement any location tracking mechanism

Fig. 1.12 Variations of iSMS Configuration



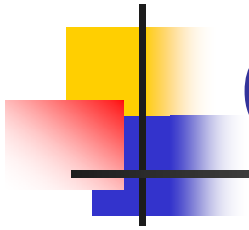
(a) Mobile Server for *i*SMS Services

(b) *i*SMS Systems Connected to Isolated Intranets



Example

- A mobile iSMS server can be used
  - for example, in a mobile library application in which 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 book mobile.



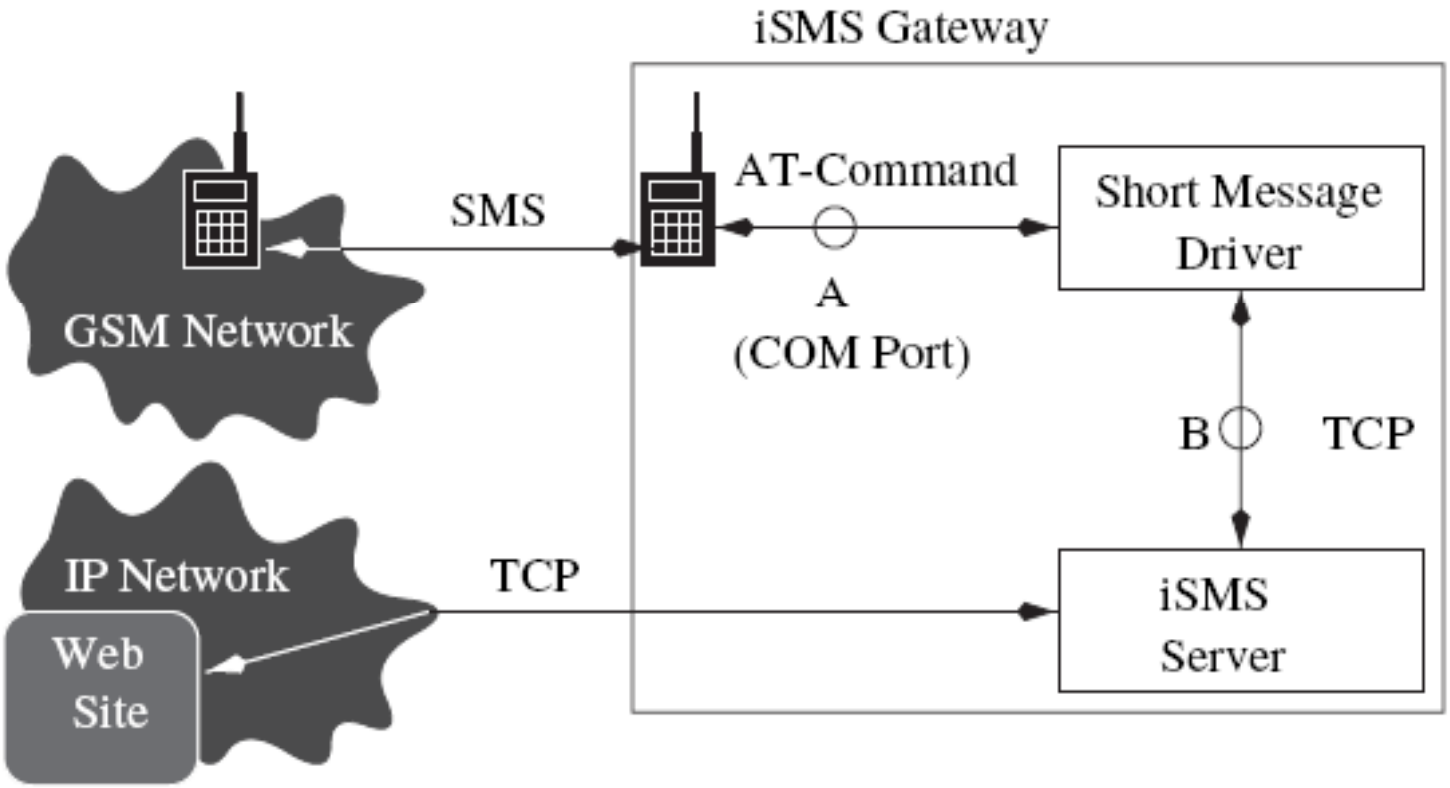
Cont.

- Figure 1.12 (b) shows a different iSMS configuration in which the iSMS systems are connected to different isolated Intranets (e.g., homes), which can query each other through a mobile network.

1.3 iSMS Communication Protocols

- AT commands
- iSMS Communication API

Fig. 1.10 iSMS System Architecture

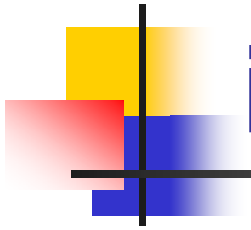


An End point SMS-IP Integration Solution

Table 1.2 AT commands used in iSMS (a partial list)

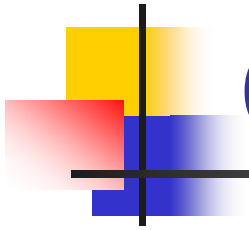
AT COMMAND	DESCRIPTION
+CNMI	New Message Indications to TE
+CSCA	Service Center Address
+CMGD	Delete Message
+CMGS	Send Message
+CMGL	List Message
+CSMP	Set Text Mode Parameters
+CMT	SMS Message Received
+CPMS	Preferred Message Storage
+CNMA	New Message Acknowledgement to ME/TA
+CMGR	Read Message
+CMGC	Send Command
+CMGW	Write Message to Memory
+CMSS	Send Message from Storage
+CSMP	Set Text Mode Parameters
+CMTI	SMS Message Received Indication
+CBMI	New CBM (Cell-Broadcast Message) Indication
+CDSI	New SMS-STATUS-REPORT Indication

ME: Mobile Equipment
TA: Terminal Adaptor



iSMS Communication API

- Connect()
  - sets up a communication link from a server to the short message driver.
- Disconnect()
- SetTimeout()
- Register()
- Status()
  - SMCMD_READABLE indicates that the server is ready for retrieving short message from the driver.
  - SMCMD_WRITABLE indicates that the server is ready for message to the driver.
  - SMCMD_ACK indicates that the message from the server to is successful.
  - SMCMD_CLOSED indicates that the connection is closed.



Cont.

- Send()
- Recv()
- RecvACK()
- 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.

A Simple Echo Server Program

```
#include <stdlib.h>
#include <smsio.h>
1 char *phone_list[] = { "+886936000001", "0931000001" };
2 int main() {
3 class CSmsdServer server; octet data[1024],dcs,option;
4 char sender[22]; char* da[1]; u_long host;
5 int port, ret, length; LPHOSTENT lphost;
6 lphost = gethostbyname("localhost");
7 if(lphost!=NULL) host=((LPIN_ADDR)lphost->h_addr)->s_addr;
8 port = 1122;
9 server.SetTimeout(3, 0);
10 if (server.Connect(host, port) != INET_SUCCESS)
11 { printf("Failure: connect to smsd\n"); _exit(1); }
12 server.SetTimeout(0, 50);
13 if (server.Register(phone_list, 2) != INET_SUCCESS)
14 { printf("Failure: register valid users\n"); exit(1); }
15 while (1) {
16 Sleep(1000);
17 ret = server.Status();
18 if (ret & SMCMD_CLOSED) break;
19 if (!(ret & SMCMD_READABLE)) continue;
20 ret = server.Recv(sender, data, &length, &dcs, &option);
21 if (ret != INET_SUCCESS) break;
22 printf("Sender: %s\nMessage: %s\n", sender, (char*)data);
23 while (!(server.Status() & SMCMD_WRITABLE)) ;
24 da[0] = sender;
25 server.SendText(da, 1, (const char*)data);
26 while (!(server.Status() & SMCMD_ACK)) ;
27 ret = server.RecvACK();
28 if (ret == SMCMD_NACK_SENDSM) break;
29 printf("\n Sending SMS....successful \n");
}
30 return 0;
}
```

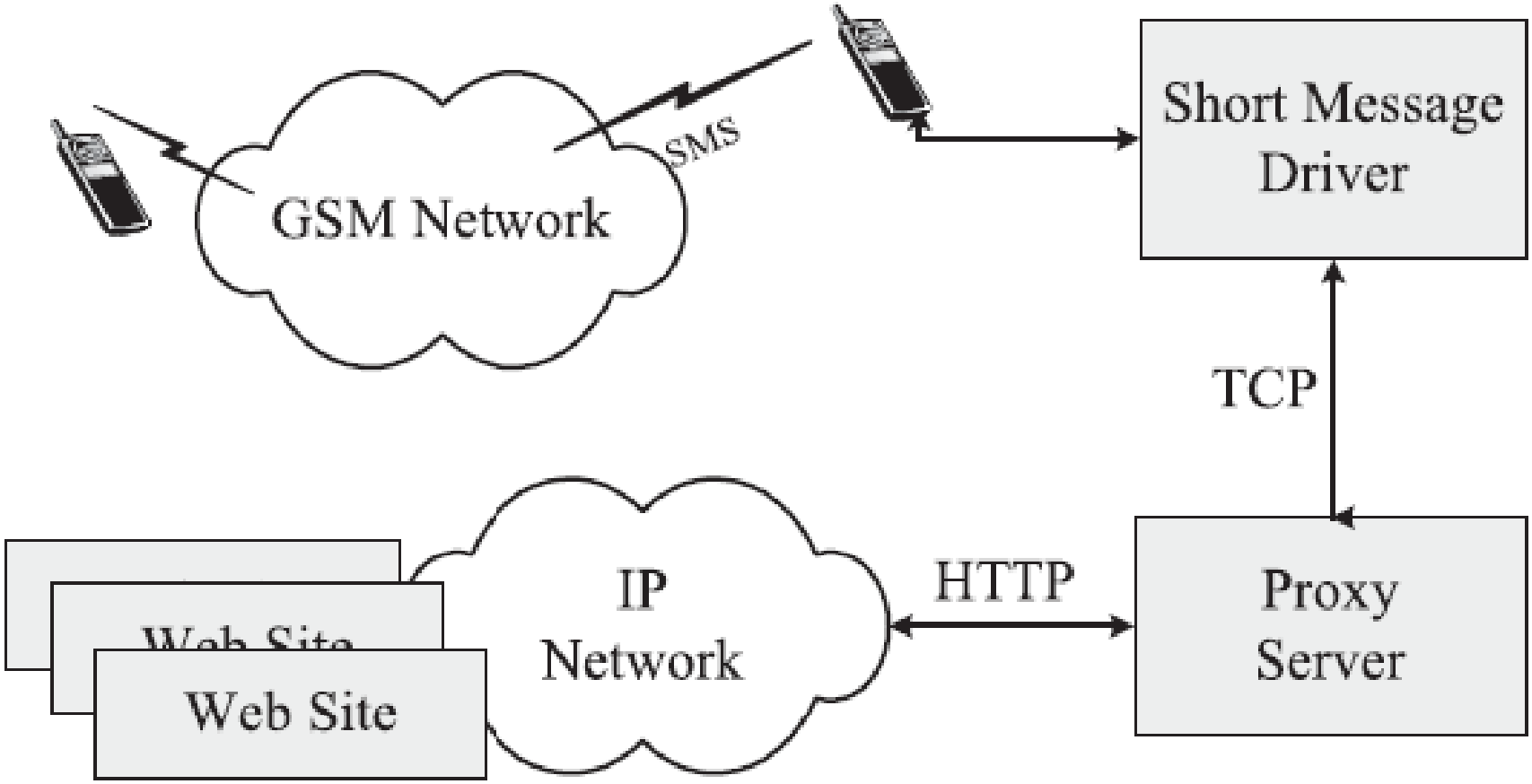
1.4 Examples of Services

This section describes several services implemented in iSMS.

1.4.1 Accessing the Web from GSM MSs

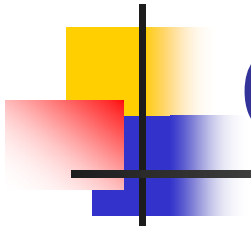
- iSMS supports a service that allows users to surf the web using standard GSM MSs.
- In this architecture, a customer (that is, a GSM MS holder) sends out a web query in SMS format to a proxy server connected to the iSMS server.
- The proxy server and the iSMS server may be running on the same host or on different hosts.

Fig. 1.14 iSMS Web Accessing Architecture



Cont.

- Based on the customer's preference, the proxy maps the input query to a proper ***Hypertext Transfer Protocol* (*HTTP*) call** and forwards the call to the corresponding web server.
- The proxy then filters and converts the data returned 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.



Cont.

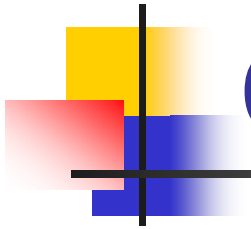
- iSMS implements functions for querying Internet information,
  - such as stock quotes, currency exchange rates, and delivery status of FedEx packages

Cont.

- ***Stock Quotes***
- 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 keyword for query.
- The second field, {symbol1}, specifies HTTP call mapping.
- The third field, {symbol2}, defines the filter/conversion function.

1.4.2 Handset Music Service

- Teenage MS holders are particularly interested in having fancy music tones such as the latest pop songs.
  - Nokia, for example, has developed a messaging protocol for this purpose.
  - Based on SMS, the protocol, called *Smart Messaging* [Nok 97], specifies a set of pre-defined message headers for sending music tones, business cards, etc.
  - Nokia MSs recognize the header and perform different actions on the message body.



Cont.

- tone = [style] [tempo] [volume] [repeat] (noteexpression)+
- note-expression = note [scale][duration][durationspecifier]

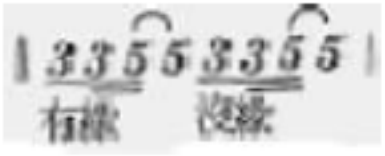
- duration = "A" ; full-note
  |"B" ; reserved
  |"C" ; 1/2-note
  |"D" ; 1/4-note
  |"E" ; 1/8-note
  |"F" ; 1/16-note
  |"G" ; 1/32-note

Lengths of 1/4 note

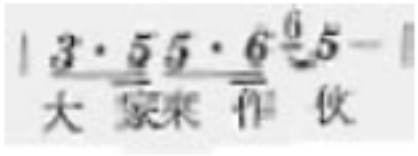
SYMBOL	LENGTH OF 1/4 NOTE	SYMBOL	LENGTH OF 1/4 NOTE
T00	2.40 sec.	T08	0.95 sec.
T01	2.14 sec.	T09	0.85 sec.
T02	1.90 sec.	T10	0.76 sec.
T03	1.70 sec.	T11	0.67 sec.
T04	1.51 sec.	T12	0.60 sec.
T05	1.35 sec.	T13	0.54 sec.
T06	1.20 sec.	T14	0.48 sec.
T07	1.07 sec.	T15	0.43 sec.

The STL Representation of a Taiwanese Song

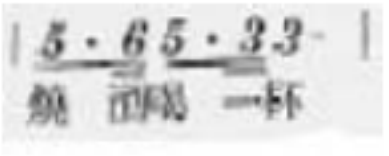
Notes:



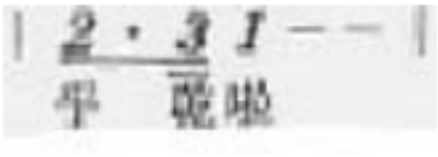
STL: `t13` `3e 3f 5 3e 3f 5`



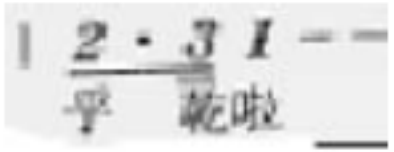
`3e 5f 5e 6f 5`



`5e 6f 5e 3f 3`



`2z 3f 1c`



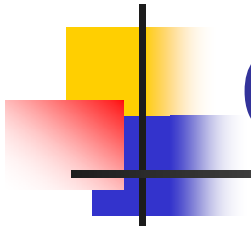
`2z 3f 1c`

1.4.3 Train Schedule System

- Consider the iSMS train service that provides train schedule query information and ticket reservations.
- This train schedule server consists of an agent dispatcher and two agents:
  - **The query agent** allows the customers to query particular train schedules.
  - **The reservation agent** allows the customers to reserve train tickets.

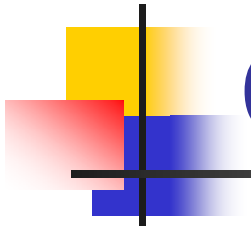
1.4.4 Other iSMS Services

- **Online Help Service:** An iSMS user types the h command to query the services available in the iSMS system.
- **Personal Profile:** iSMS maintains a personal profile repository for individual registered users.
- **Broadcast Message Service:** An iSMS user can broadcast a message to several destinations with the following command:
  - bc {receivers} {message}
  - In this command, {receivers} is a list of MSISDNs of the receivers and *{*message*}* is the message to be delivered.



Cont.

- **Group Communications:** iSMS implements a group communication mechanism.
- **Mobile Dictionary Service:** The command
  - dic {English word}
  - returns the Chinese meaning of the English word {English word}.



Cont.

- **Email Service:** iSMS implements an agent to SMTP/POP3 servers for relaying email and SMS. A short message from mobile users to iSMS with the format
  - ema {email-address} {message}
- **Forwarding Service:** An iSMS user can execute a command *{*command*}* and then forward the results to another user *{*receiver*}*. The command is
  - fwd {receiver} {command}

1.5 Caching for iSMS-Based Wireless Data Access

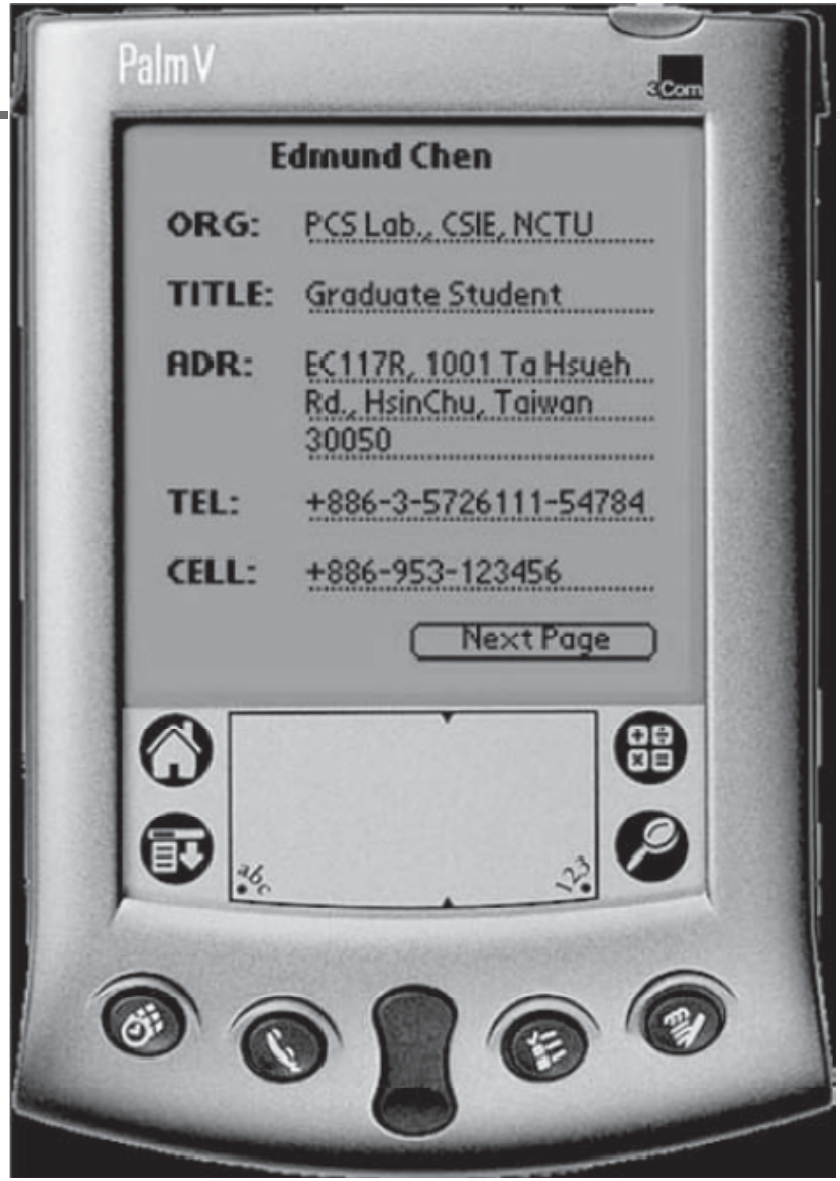
This section uses a business card application as an example to illustrate how cache-based wireless data access can be implemented in an SMS-IP platform such as iSMS.

The vCard format in the iSMS-based business card service

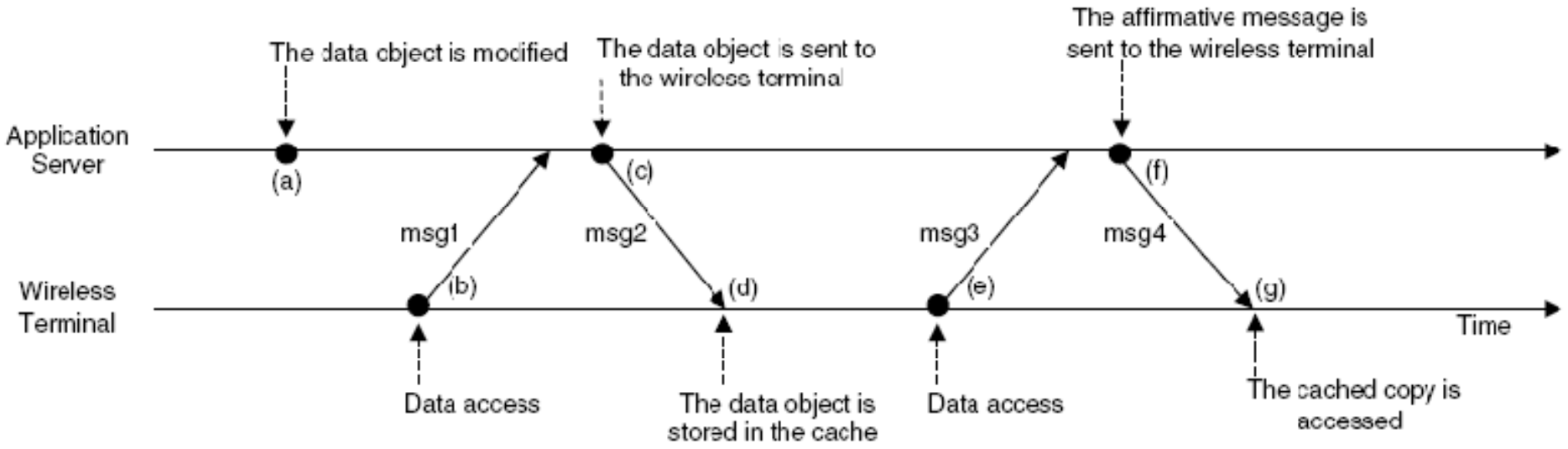
FIELD	DESCRIPTION	LENGTH
VERSION	Version of vCard	13 bytes
FN	vCard object name	30 bytes
N	Name information	40 bytes
ORG	Organization information	50 bytes
TITLE	Job title	50 bytes
ADR	Address	120 bytes
TEL	Phone number	130 bytes
EMAIL	Email address	50 bytes
URL	Uniform resource locator	50 bytes
CALENDAR*	Public calendar event	100 bytes

*The CALENDAR field is not defined in the vCard standard.

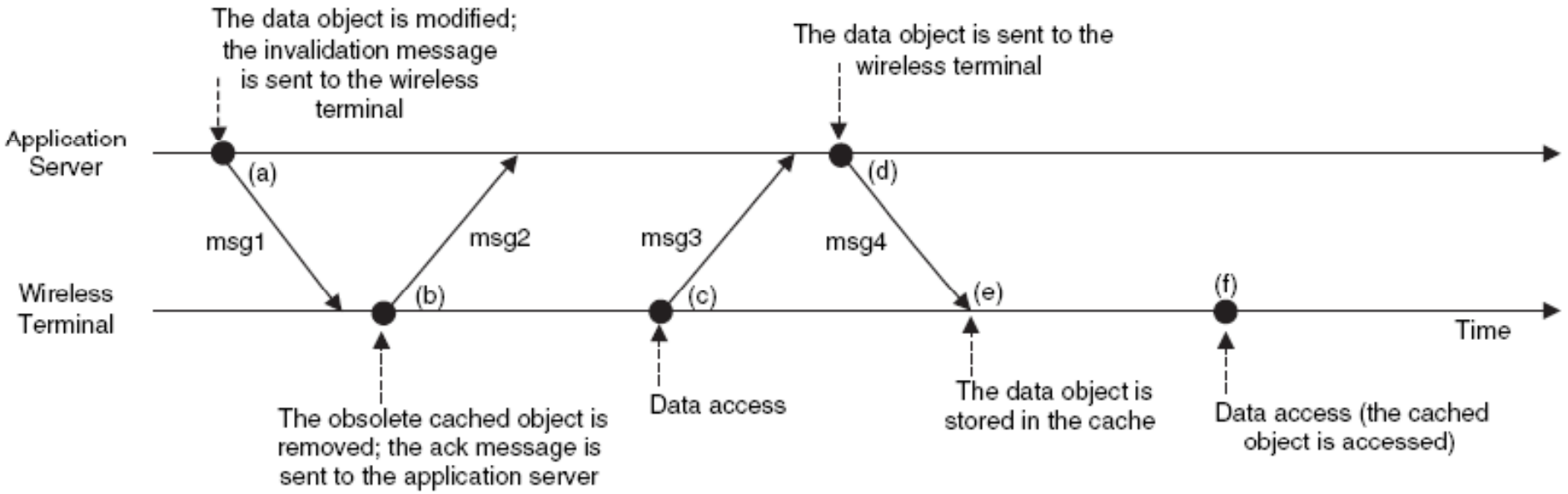
An iSMS Business Card Shown in PDA

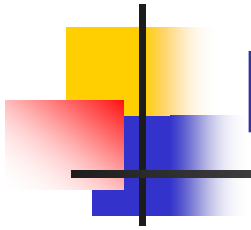


Data Access in Poll-Each-Read



Data Access in Call-Back





Homework

- **1-1** Try to develop a new service of SMS and IP network integration.