Chapter 9
Session Initiation Protocol

(text book: chapter 12)

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

Outline

- 12.1 An Overview of SIP
- 12.2 SIP-based GPRS Push Mechanism
- 12.3 SIP-based VoIP Prepaid Mechanism
  - 12.3.1 Prepaid Call Setup
  - 12.3.2 Forced Termination of a Prepaid Call
- 12.4 Concluding Remarks

Abstract

- **Chapter 12** introduces the Session Initiation Protocol (SIP).
  - We show how SIP supports user mobility, call setup, and call release.
- Based on the SIP protocol, we illustrate how the push mechanism and the prepaid mechanism can be implemented in GPRS/UMTS.
- In GPRS, the push feature is not supported. That is, an MS must activate the PDP context for a specific service before the external data network can push this service to the MS.
  - An example is VoIP call termination (incoming call) to an MS. However, maintaining a PDP context without actually using it will significantly consume network resources.

Cont.

  - Using the iSMS Internet platform, we describe a SIP-based push mechanism for SIP call termination of GPRS supporting private IP addresses.
- A major advantage of this approach is that no GPRS/GSM nodes need to be modified.
  - For another SIP application example, we describe a SIP-based prepaid mechanism to handle the prepaid calls in a VoIP system.
  - Integration of this prepaid mechanism with the existing VoIP platform can be easily achieved by reconfiguring the call server.

Introduction

- In the UMTS all-IP network, the *Session Initiation Protocol* (*SIP*) is the default protocol for the IP Multimedia Core Network Subsystem.
- As a standard for Internet telephony published in 1999 by the Internet Society, SIP is a general way for an application to make one computer user aware that another user is online and available for communications;
  - that is, it is the Internet's virtual dial tone.

Cont.

- SIP telephony can be integrated with applications such as games.
  - Today, the easiest way to make a free Internet phone call is with a network connected Xbox or by playing a multiplayer online video game

12.1 An Overview of SIP

- As an application-layer signaling protocol over the IP network, SIP is designed for creating, modifying, and terminating **multimedia** sessions or calls.
- The SIP message specifies the *Real-Time Transport Protocol/Real-Time Transport Control Protocol* (***RTP/RTCP***), which delivers the data in the multimedia sessions.

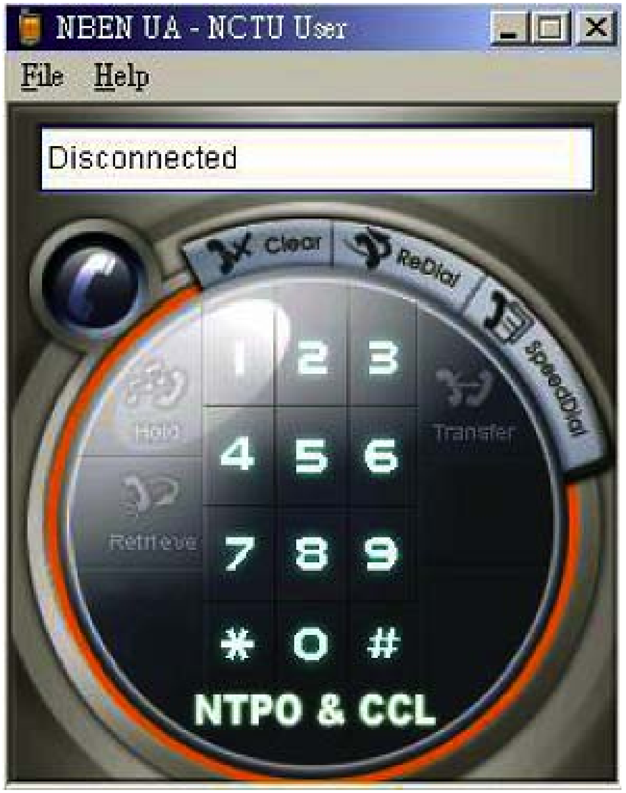
Cont.

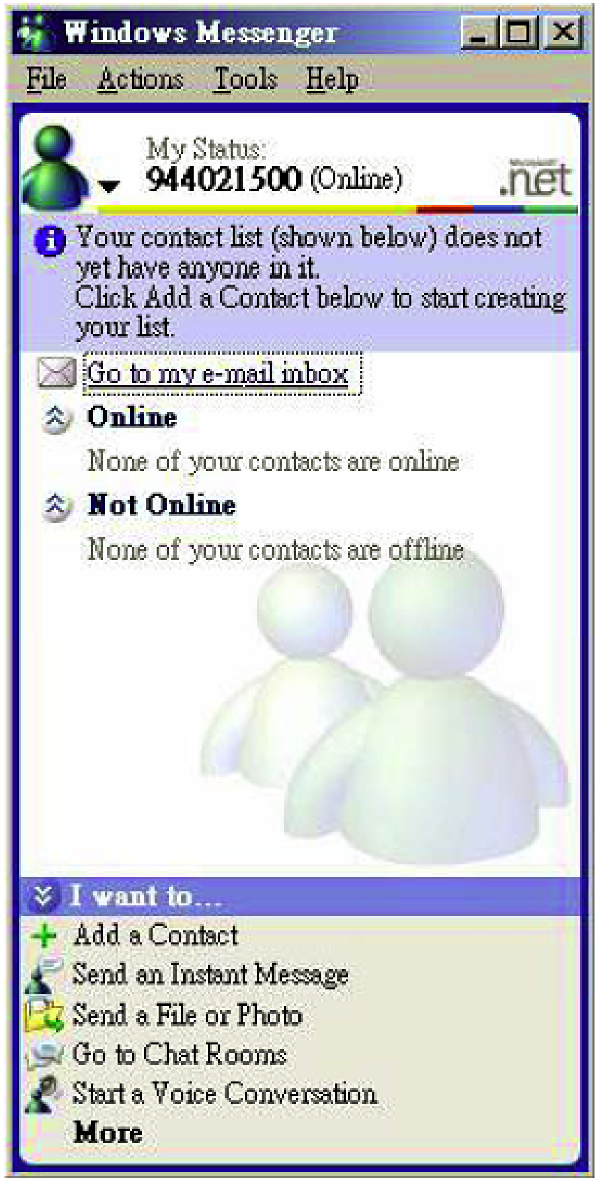
- RTP is a transport protocol on top of UDP, which detects packet loss and ensures ordered delivery.
- An RTP packet also indicates the packet sampling time from the source media stream. The destination application can use this time stamp to calculate delay and jitter.

Cont.

- Two major network elements are defined in SIP: *user agent* (UA) and *network server*.
  - The user agent resides at SIP endpoints (or phones).
  - Figures 12.1 and 12.2 give softphone and hardphone examples of UA.
  - A user agent contains both a ***User Agent Client*** (*UAC*) and a ***User Agent Server*** (*UAS*).
  - The UAC (or calling user agent) is responsible for issuing SIP requests, and the UAS (or called user agent) receives the SIP request and responds.

Fig.12.1 User Agents (**Softphone**)



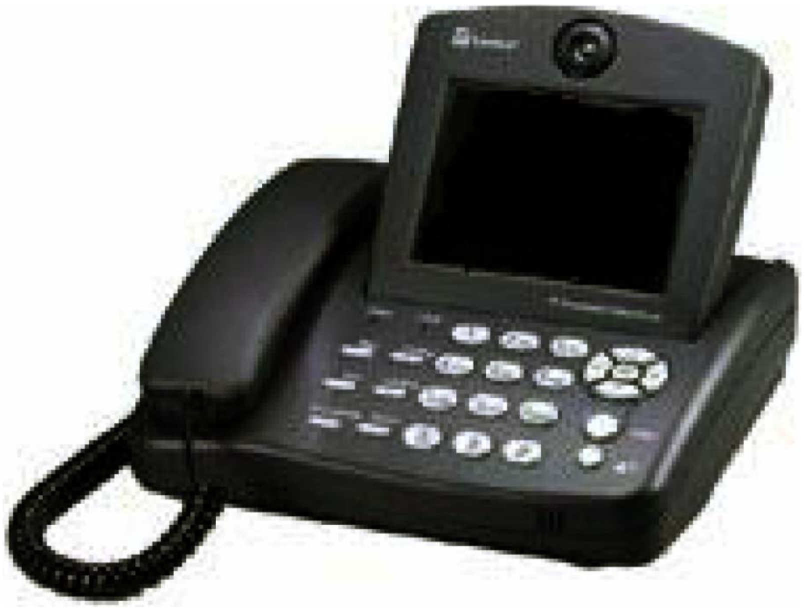


Your contact list (shown below) does not yet have anyone in it.
Click Add a Contact below to start creating your list.

None of your contacts are online

None of your contacts are offline

Fig.12.2 User Agent (Leadtek **Hardphone**)



Six basic types of SIP requests

- **REGISTER** is sent from a user agent to the registrar (a network server to be defined later) to register the address where the subscriber is located.
- **INVITE** is used to initiate a multimedia session. This request includes the routing information of the calling and the called parties, and the type of media to be exchanged between the two parties.
- **ACK** is sent from a UAC to a UAS to confirm that the final response to an INVITE request has been received.
- **OPTIONS** is used to query the user agent’s capabilities, such as the supported media type.
- **BYE** is used to release a multimedia session or call.
- **CANCEL** is used to cancel a pending request (i.e., an incomplete request).

Fig.12.3 SIP Registration, Call Setup, and Termination

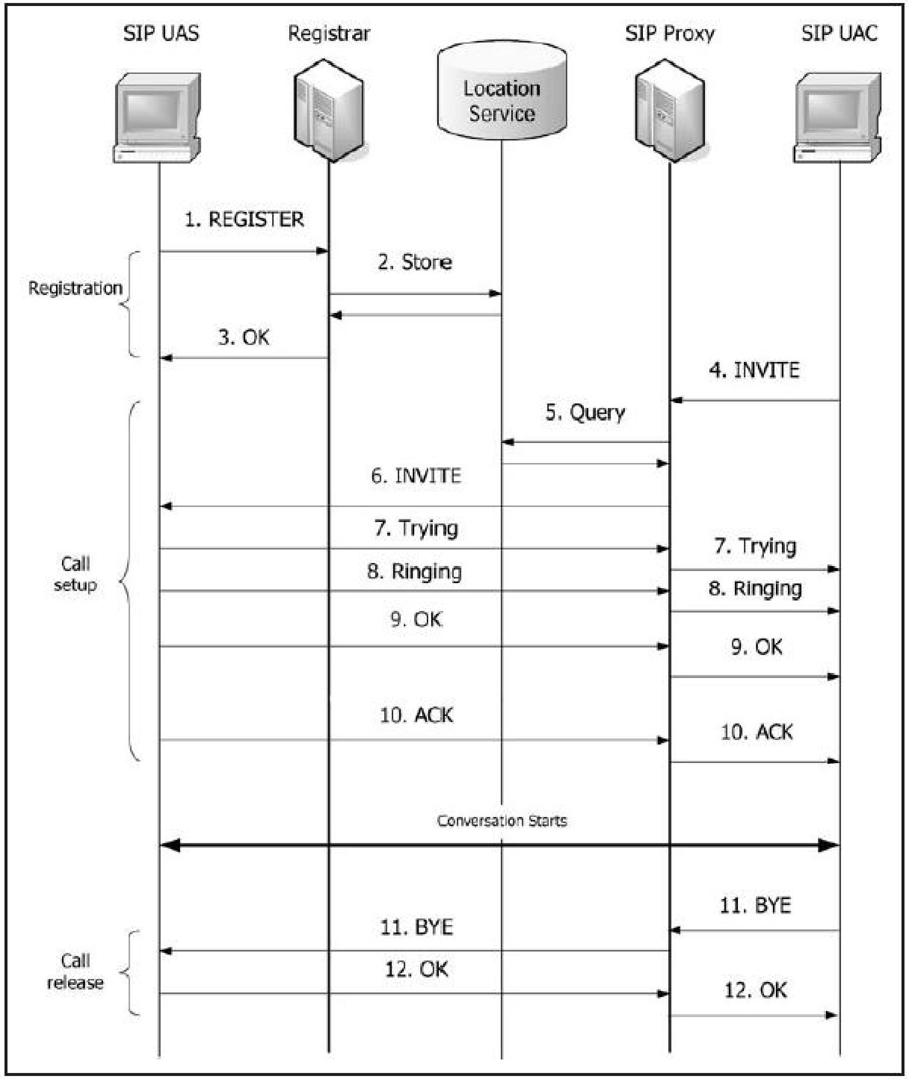


Figure 12.3 SIP Registration, Call Setup, and Termination

Cont.

- SIP also defines the **INFO** method for transferring information during an ongoing session.
  - For example, The *Dual-Tone Multifrequency* (*DTMF*) digits can be delivered during a VoIP call through the INFO message.
- We will show another example of INFO usage in Section 12.3.1.

Cont.

- After receiving a request message, the recipient takes appropriate actions and acknowledges with a *SIP response* message.
- The response message carries a **return code** indicating the execution result for the request.

Examples of the return code are

- **100** Trying: the request is currently being executed
- **180** Ringing: the called party is alerted
- **183** Session Progress (provisional)
- **200** OK: the request was executed normally
- **401** Unauthorized: the client is not authorized to make the request
- **404** Not Found
- **500** Server Failure: the request could not be executed because of server internal error

The SIP URI is of the format

- A SIP user is globally and uniquely identified by a *Uniform Resource Identifier* (*URI*)

  sip:username@hostname:port

- **sip** is the prefix to indicate that the whole string is a SIP URI,
- **username** is a local identifier of the SIP user on the server hostname,
- **hostname** is the SIP server for the user username, and
- **port** is the transport port to accept the SIP message on the hostname, which typically is “5060”.
  – A SIP URI example is “sip:yjlou@csie.nctu.edu.tw:5060”.

Cont.

- A *SIP transaction* consists of a request and one or more responses. The transaction is initiated by a *translation initiator*.
- The *target* of the transaction may or may not be the *recipient* of the request.
  - For example, in a registration transaction, a UA (the initiator) may register to a SIP registrar (the recipient) for another UA (the target).
  - As another example, in an invite transaction, a UA (the initiator) sets up a call to another UA (the target that is also the recipient).

Cont.

- SIP conjuncts with protocols such as *Session Description Protocol* (***SDP***) to describe the multimedia information.
  - While RTP transports the voice packets, SDP provides the RTP information such as the **network address** and the **transport port number** of the RTP connection.

SIP supports three types of network servers:

- *registrar*
- *redirect server*
- *proxy server*

Fig.12.3 SIP Registration, Call Setup, and Termination

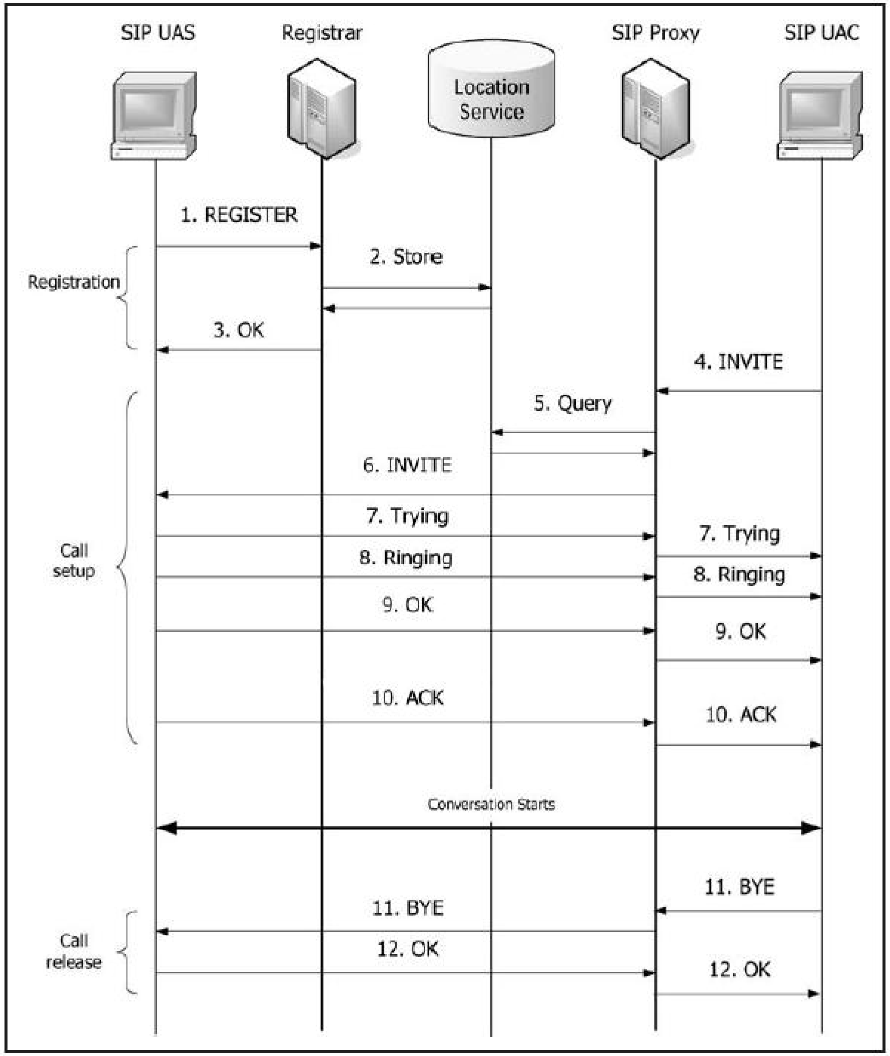


Figure 12.3 SIP Registration, Call Setup, and Termination

Cont.

- To support user mobility, the user agent informs the network of its current location by explicitly registering with a registrar.
  - The registrar is typically co-located with a proxy or redirect server.
  - A SIP UA can periodically register its SIP URI and contact information (which includes the IP address and the transport port accepting the SIP messages) to the registrar.
  - When the SIP UA moves to different networks, the registrar always holds the current contact information of the SIP UA. A registrar may store the contact information in a *location service* database.

Cont.

- A proxy server processes the SIP requests from a UAC to the destination UAS. The proxy server either handles the request or forwards it to other servers, perhaps after performing some translation.
  - For example, to resolve the SIP URI in the INVITE request, the proxy server consults the location service database to retrieve the current IP address and transport port of the SIP UAS. The proxy server then forwards the INVITE message to the SIP UAS.

Cont.

- A redirect server accepts the INVITE requests from a UAC, and returns a new address to that UAC. Similar to a proxy server, a redirect server may query the location service database to obtain the callee’s contact information.
- Unlike the proxy server, the redirect server does not forward the INVITE message. It only returns the contact information to the SIP UAC.

The interaction between the SIP UAC, the SIP UAS

- **Step 1.** When a SIP UAS is activated, the SIP UAS registers its SIP URI to the registrar by sending the REGISTER message.
- **Step 2.** The registrar stores the contact information in the location service database.
- **Step 3.** The registrar generates the 200 OK message and returns it to the SIP UAS.

Fig.12.3 SIP Registration, Call Setup, and Termination

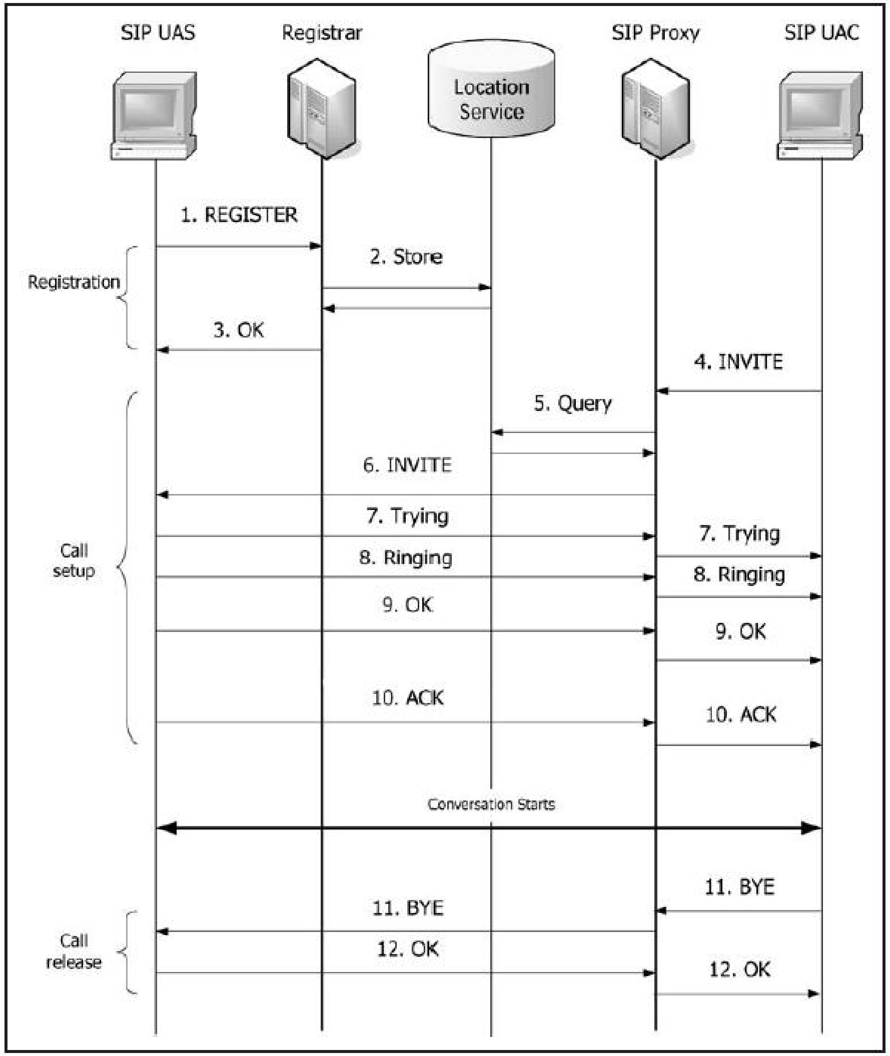


Figure 12.3 SIP Registration, Call Setup, and Termination

Cont.

- **Step 4.** The SIP UAC first sends the INVITE request to a proxy server, which is pre-configured by the SIP UAC.
  - The INVITE message contains the SIP URI of the SIP UAS and the **SDP** that describes the RTP information of the SIP UAC. The RTP information includes the IP address and port number for receiving voice data at the SIP UAC.
- **Step 5.** To resolve the SIP URI in the INVITE request, the proxy server may query the location service database to obtain the contact information of the SIP UAS.

Fig.12.3 SIP Registration, Call Setup, and Termination

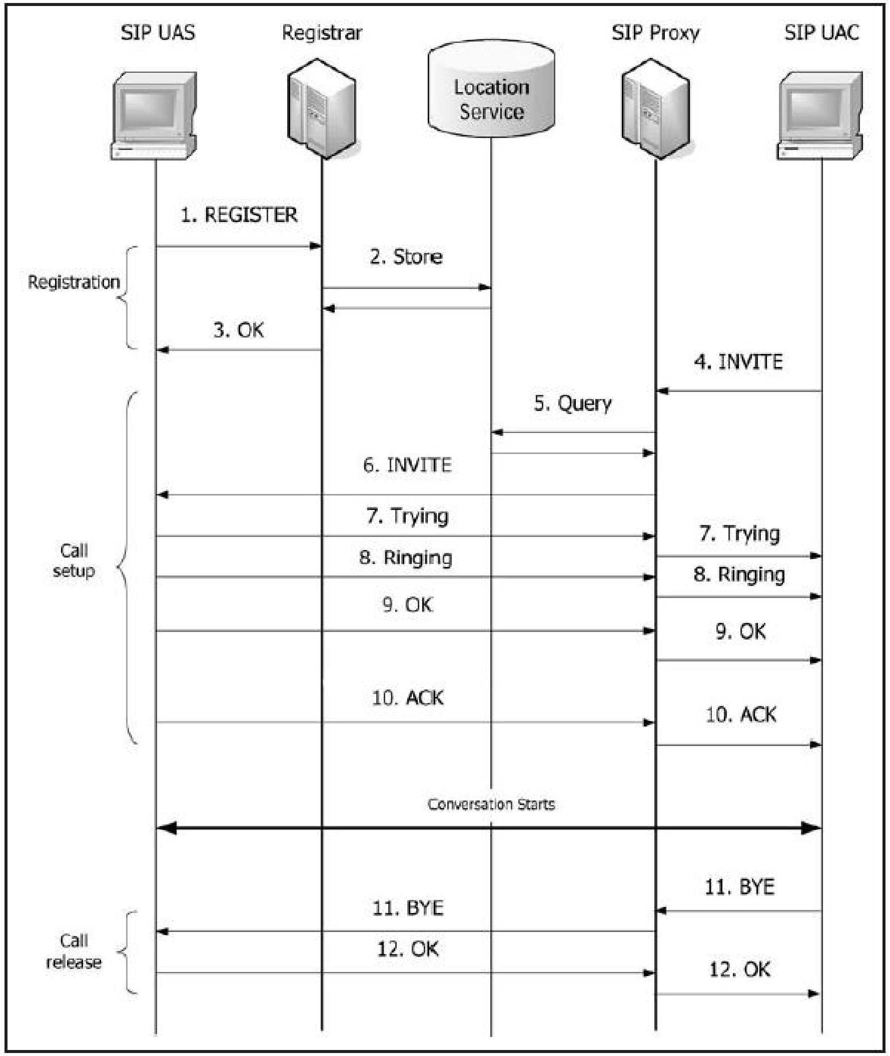


Figure 12.3 SIP Registration, Call Setup, and Termination

Cont.

- **Step 6.** Then the proxy server forwards the INVITE message to the SIP UAS.
- **Step 7.** Upon receipt of the INVITE request, the SIP UAS replies with a **100 Trying** response to indicate that the call is in progress.
  - This message is received by the SIP UAC through the proxy server.
- **Step 8.** The SIP UAS plays an audio ringing tone to alert the called user that an incoming call arrived, and sends the **180 Ringing** response to the SIP UAC through the proxy server. The SIP UAC plays an audio ringback tone to the calling user.

Fig.12.3 SIP Registration, Call Setup, and Termination

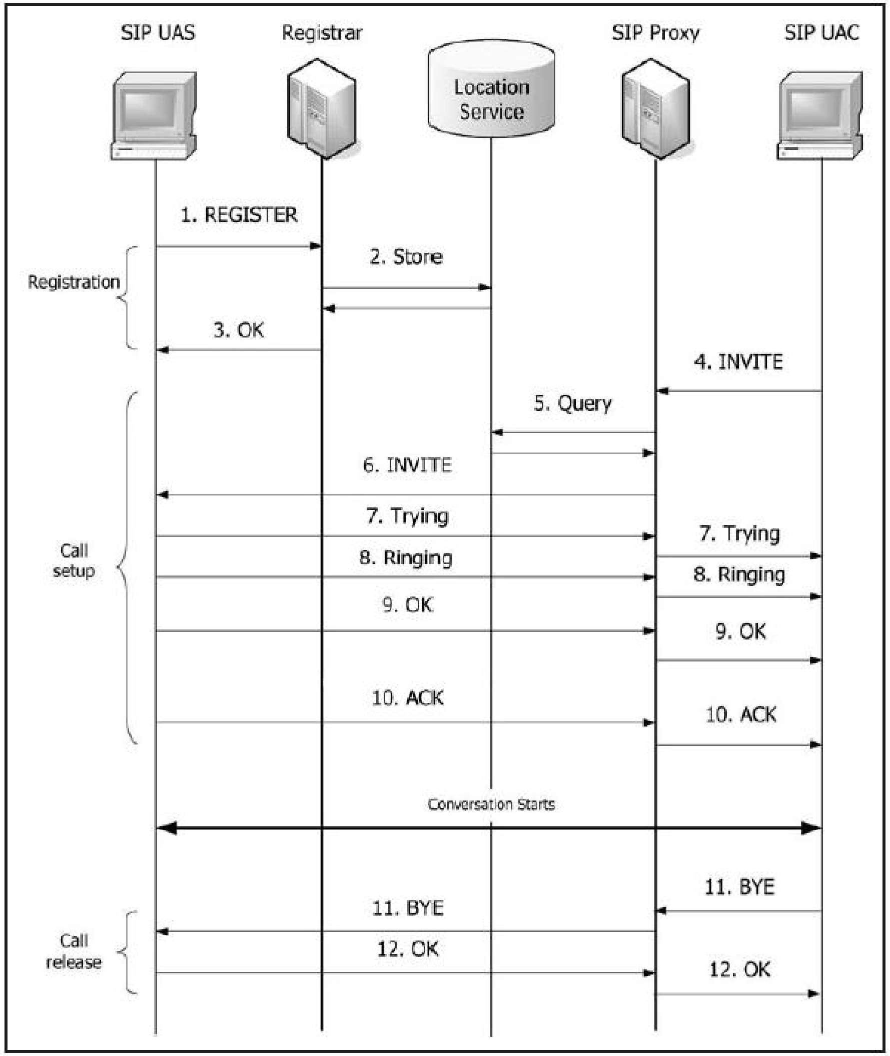


Figure 12.3 SIP Registration, Call Setup, and Termination

Cont.

- **Step 9.** When the called user picks up the handset, the SIP UAS sends the final **200 OK response** to the SIP UAC.
  - The OK message includes the SDP that describes the RTP information of the SIP UAS, such as the IP address and the transport port used by the SIP UAS.
- **Step 10.** Upon receipt of the OK response, the SIP UAC sends the ACK request to acknowledge the SIP UAS.

Fig.12.3 SIP Registration, Call Setup, and Termination

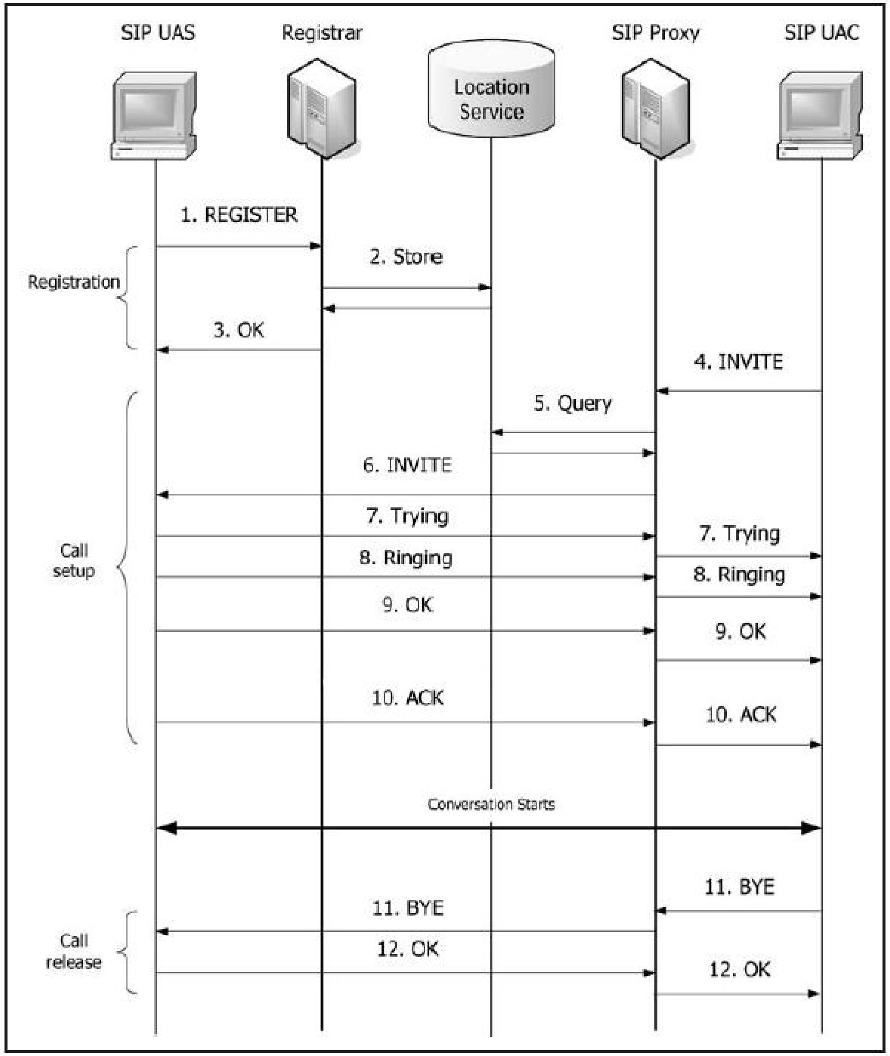


Figure 12.3 SIP Registration, Call Setup, and Termination

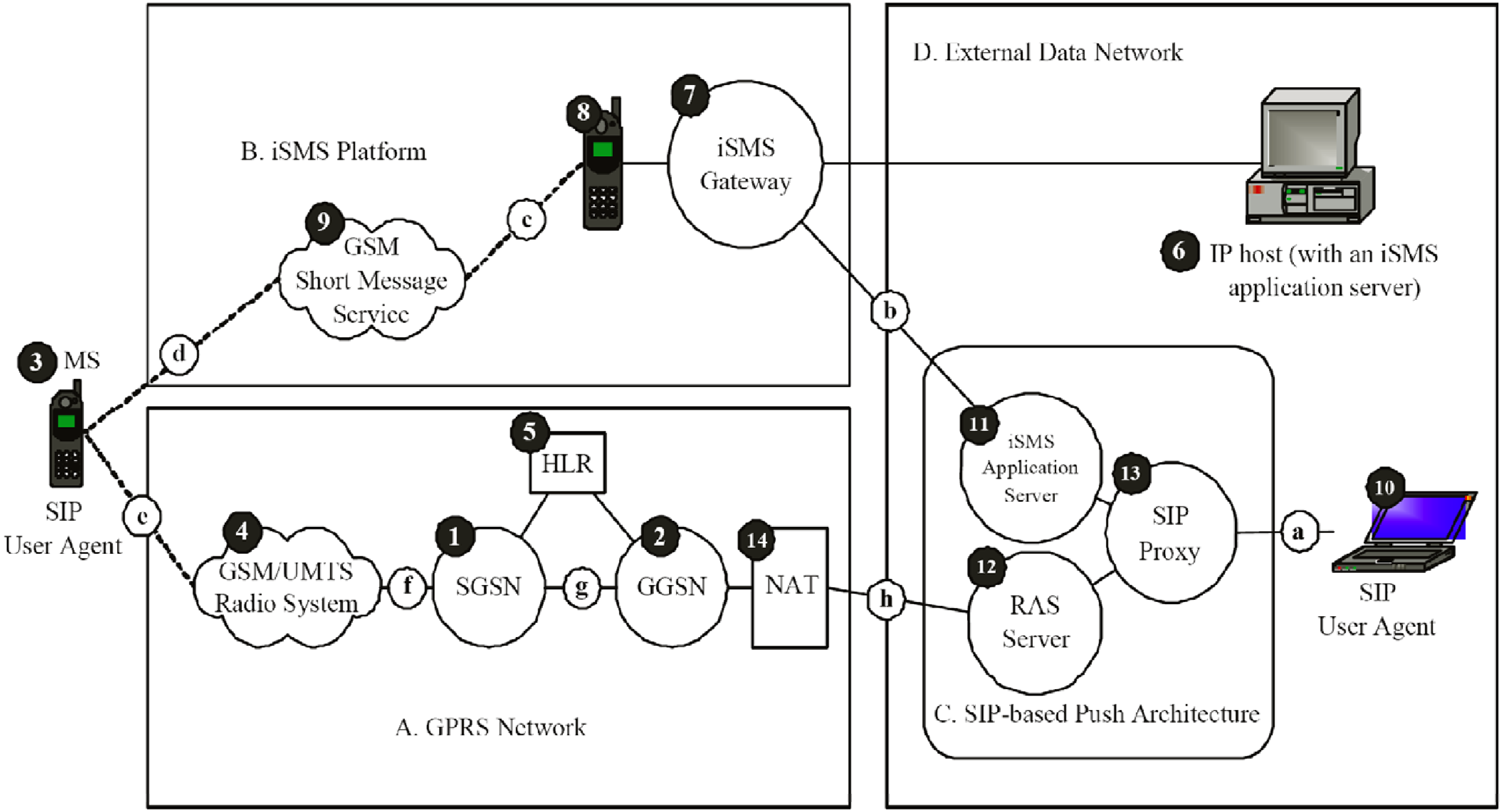
Cont.

- At this point, the conversation starts.
  - The SIP UAS sends the RTP packets to the SIP UAC according to the parameters described in the SDP of the INVITE message.
  - The SIP UAC sends the RTP packets to the SIP UAS according to the parameters described in the SDP of the final OK message.
  - Assume that the SIP UAC terminates the session after the conversation. The following steps in Figure 12.3 are executed:
- **Step 11.** The SIP UAC sends the BYE request to the SIP UAS through the proxy server.
- **Step 12.** The SIP UAS responds with the OK message to confirm the request, and the session is terminated.

12.2 SIP-based GPRS Push Mechanism

- Based on the SIP protocol described in Section 12.1, we elaborate on a push mechanism for GPRS.
- Figure 12.4 illustrates the SIP-based push architecture.
  - The GPRS network is shown in Figure 12.4 (A).
  - In several commercial GPRS implementations, an MS is dynamically assigned a private IP address.
- Therefore, a *Network Address Translator* (*NAT*) (Figure 12.4 (14)) is required to translate the IP addresses of the packets delivered between the public address realm (the external data network) and the private address realm (the GPRS network).

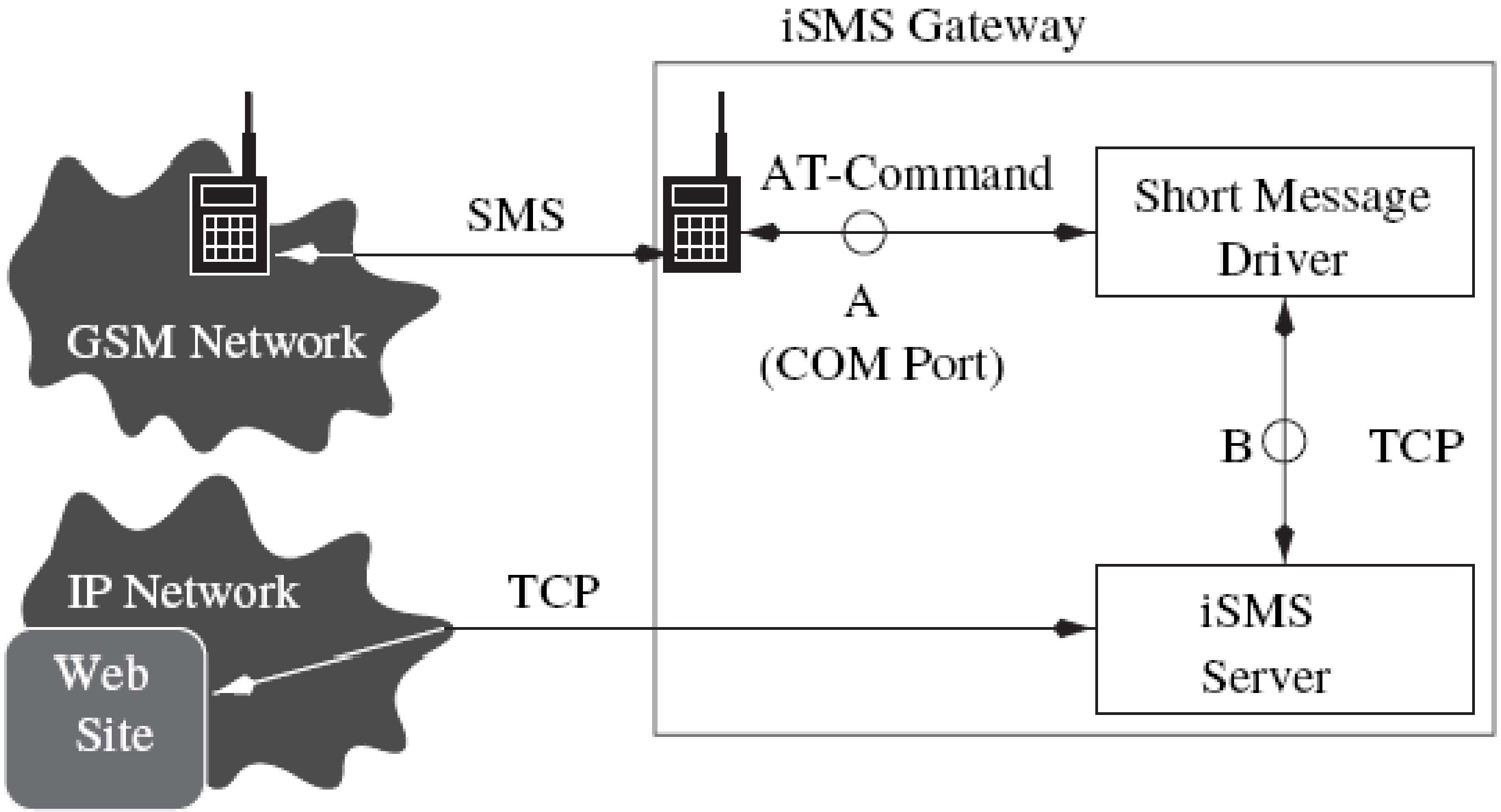
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

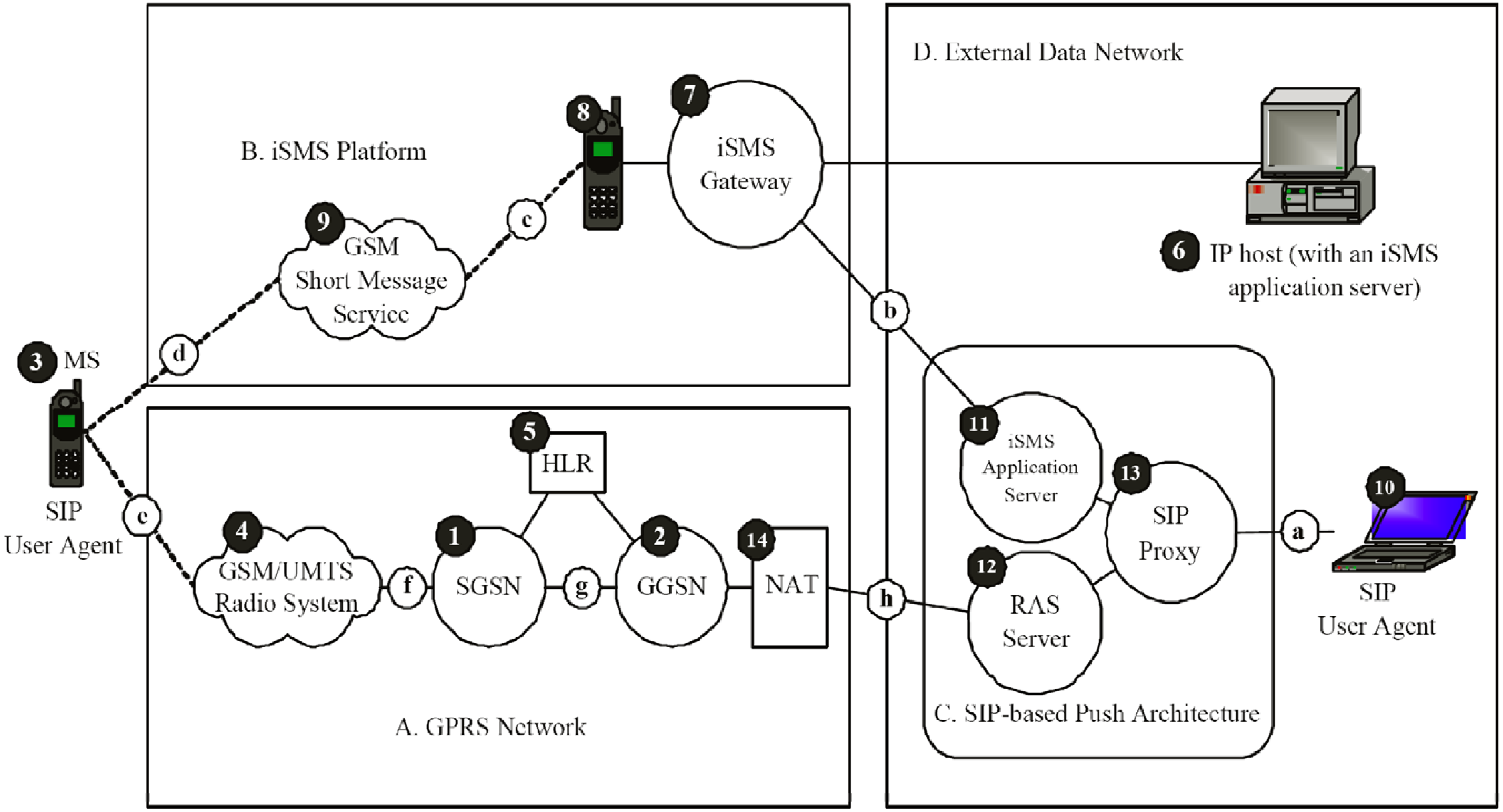
- Another issue in GPRS is that if an MS does not **activate the PDP context** for a specific service, the network cannot “push” this service to the MS.
- To resolve the above issues, this section describes a SIP-based push mechanism for GPRS supporting private IP addresses.
- This solution does not modify existing GPRS components.
  - We utilize the iSMS service platform (see Chapter 1) to implement the push mechanism. iSMS is an operator independent platform that integrates the IP network with the Short Message Service (SMS) in mobile telephone systems.

Fig. 1.10 iSMS System Architecture



An End point SMS-IP Integration Solution

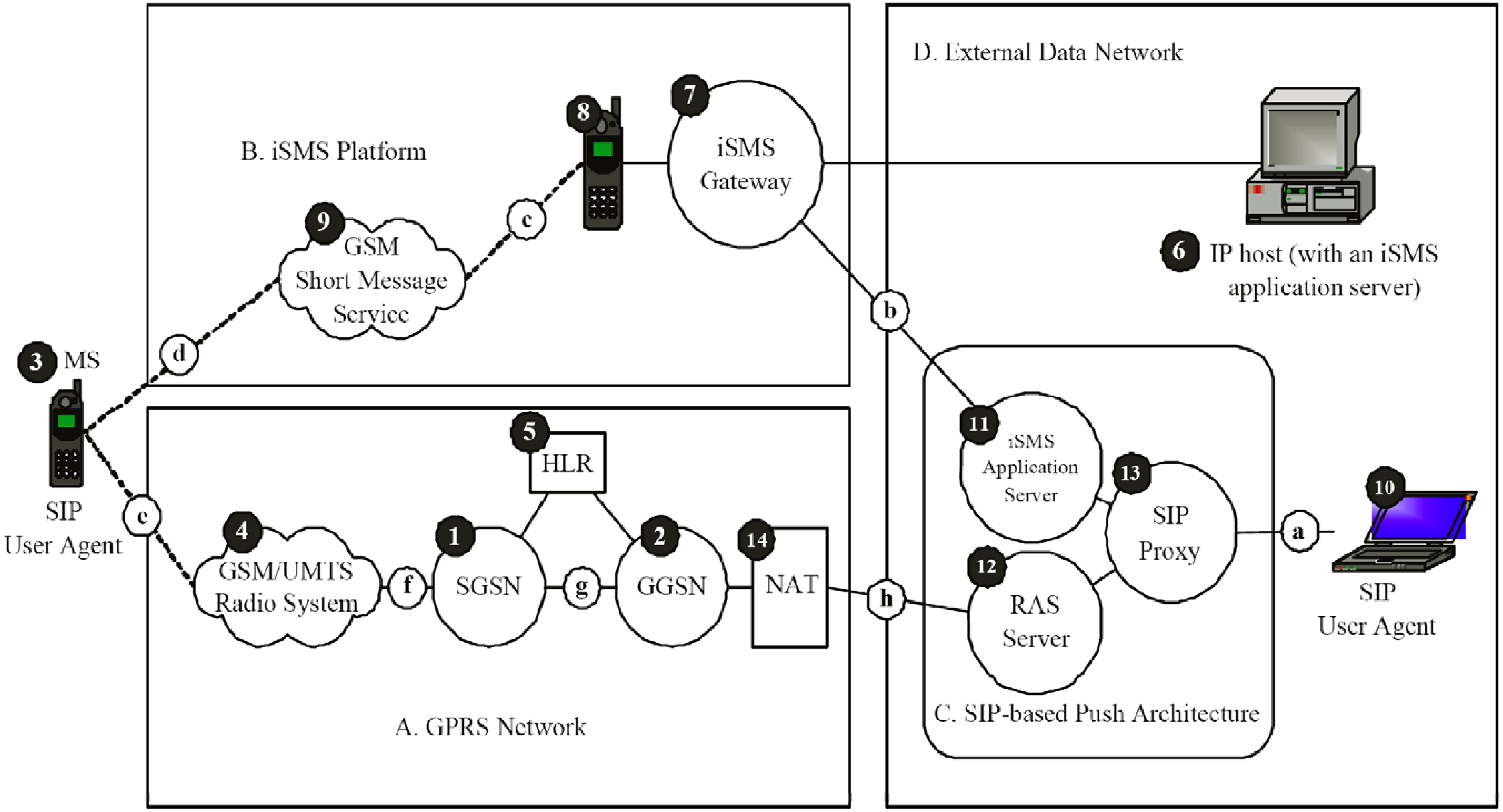
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

- Through iSMS (Figure 12.4 (B)), an IP host in the external data network (Figure 12.4 (6)) can offer Internet services to an MS (Figure 12.4 (3)).
- Specifically, messages are created by an iSMS application server run on the IP host, and then sent to the iSMS gateway (see Figure 12.4 (7)).
- The iSMS gateway is connected to a **GSM modem** (Figure 12.4 (8)) that delivers the messages to the MS through the short message service (Figure 12.4 (9)).
- Note that (9) and (4) in Figure 12.4 typically utilize the same radio system.
  - In the iSMS platform, no components in the GSM are modified. The iSMS gateway can be implemented by an off-the-shelf high-reliability PC or workstation connected to an MS.

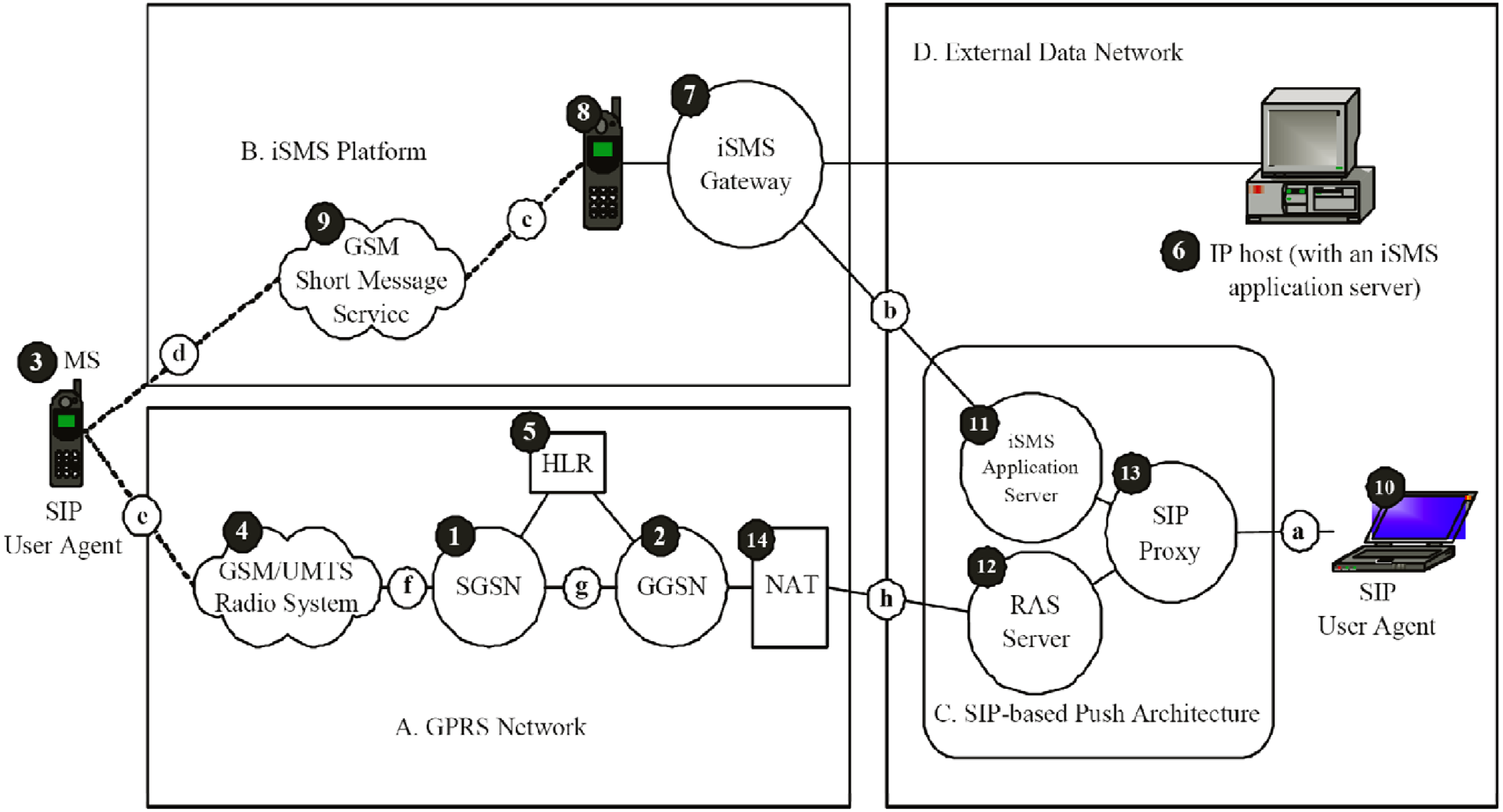
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Push Mechanism

- The idea behind the push mechanism is simple.
  - When the VoIP calling party in the IP network (Figure 12.4 (10)) initiates a call to an MS, the iSMS application server (Figure 12.4 (11)) issues a short message to the MS through the iSMS gateway. This short message instructs the MS to activate the PDP context for the VoIP service.
  - Therefore, the **network-requested PDP context activation** is performed without requiring the GGSN to interact with the HLR.

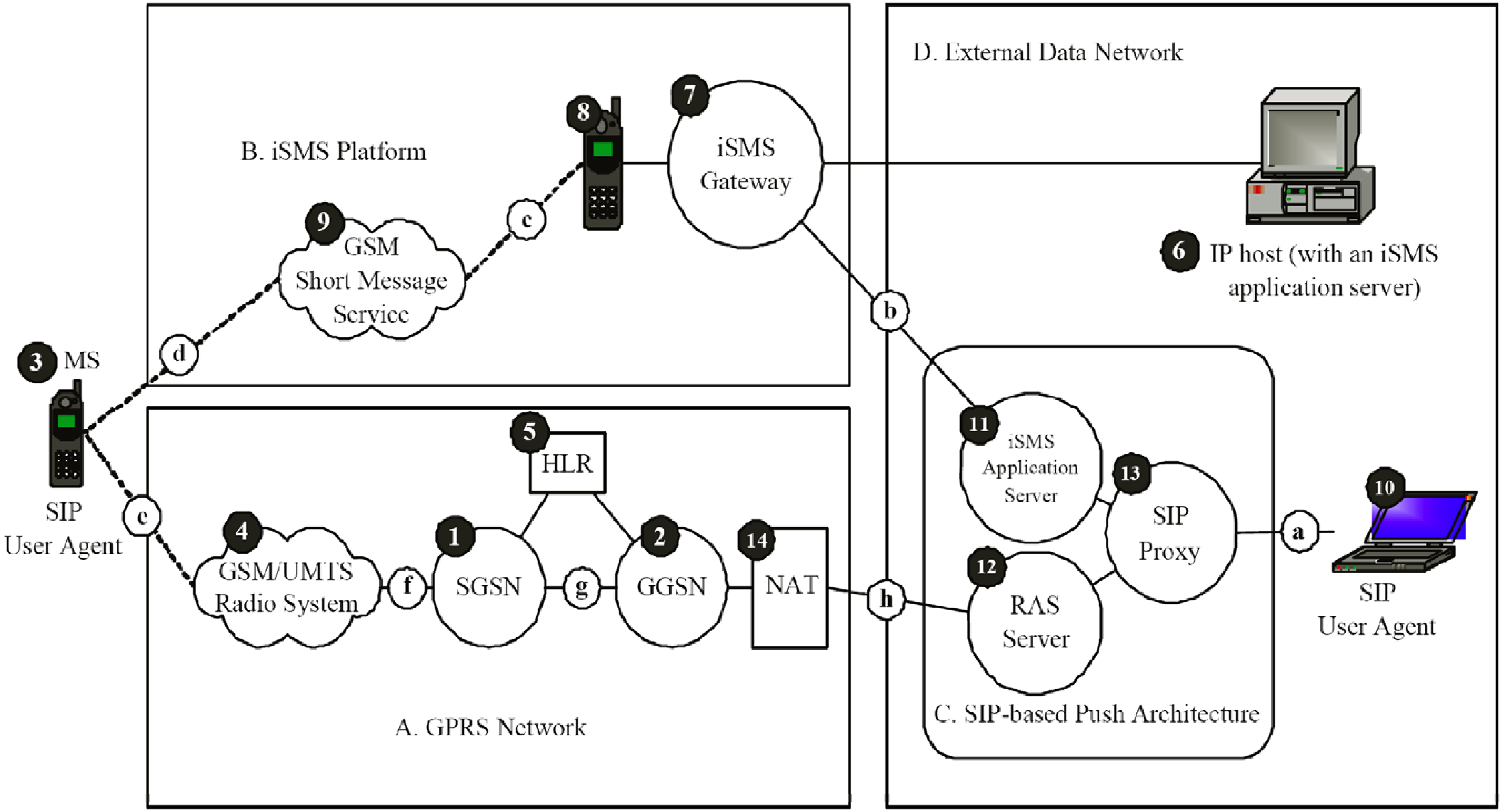
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

- At National Chiao Tung University (NCTU), we have implemented a *SIP-based Push Architecture* (*SPA*; see Figure 12.4 (C)) that uses the SIP described in Section 12.1 for **VoIP signaling**.
- In this architecture, a SIP proxy (Figure 12.4 (13)) connects to a SIP user agent (Figure 12.4 (10)) in the external data network and an MS (with another SIP user agent) through the GPRS network.

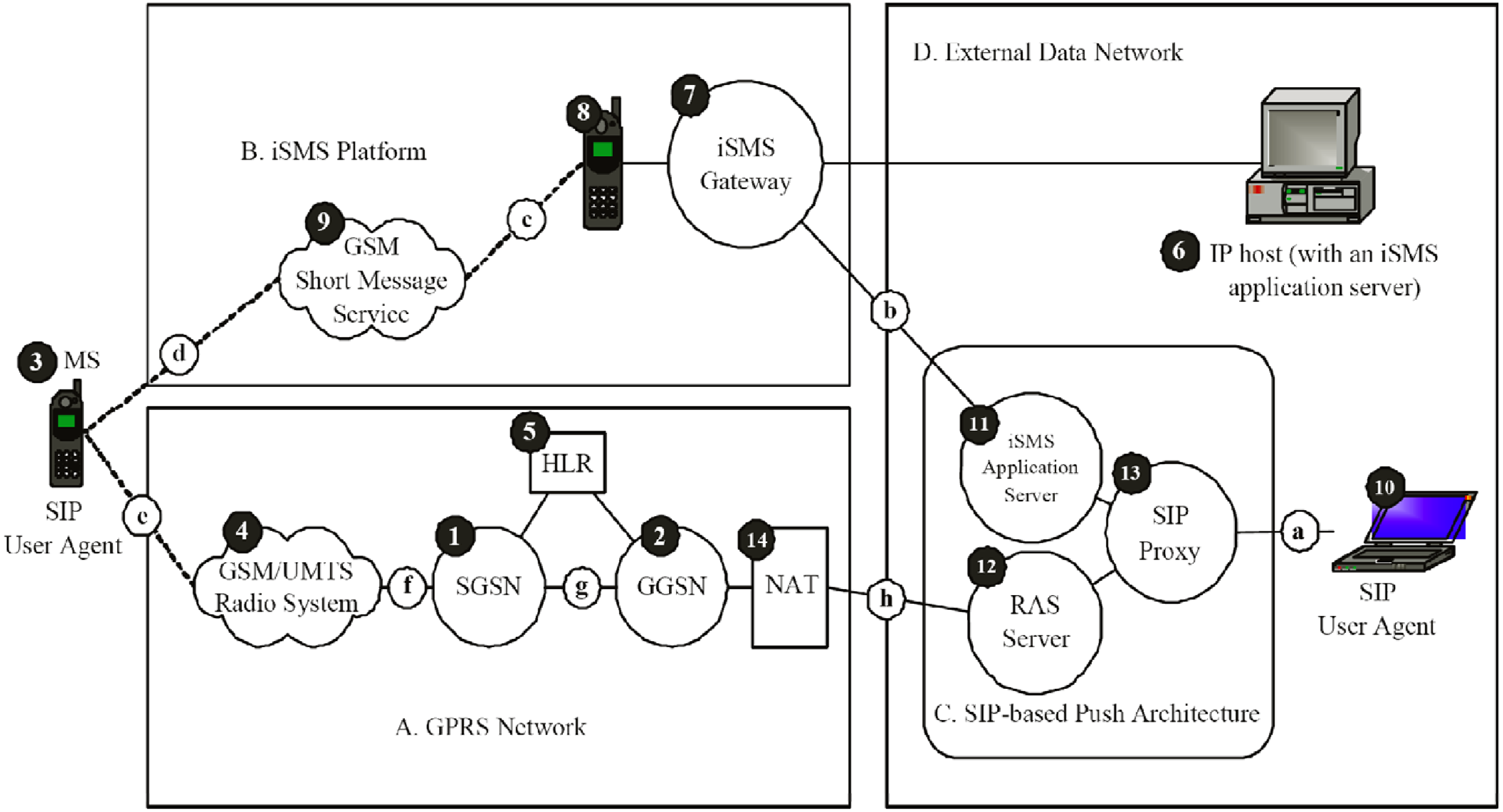
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

- An iSMS application server (Figure 12.4 (11)) is implemented in the SPA to support the push operation.
- Note that the port number for SIP applications is pre-defined.
  - Since the NAT server distinguishes the hosts by the port numbers, the fixed port number nature of SIP will result in wrong translation at the NAT server. Therefore, a *Remote Access Service* (*RAS*) server (Figure 12.4 (12)) is implemented to support the tunnel between the SIP proxy and an MS. This tunnel implementation is based on the *Layer Two Tunneling Protocol* (*L2TP*) described in Section 4.4 [IET99b].

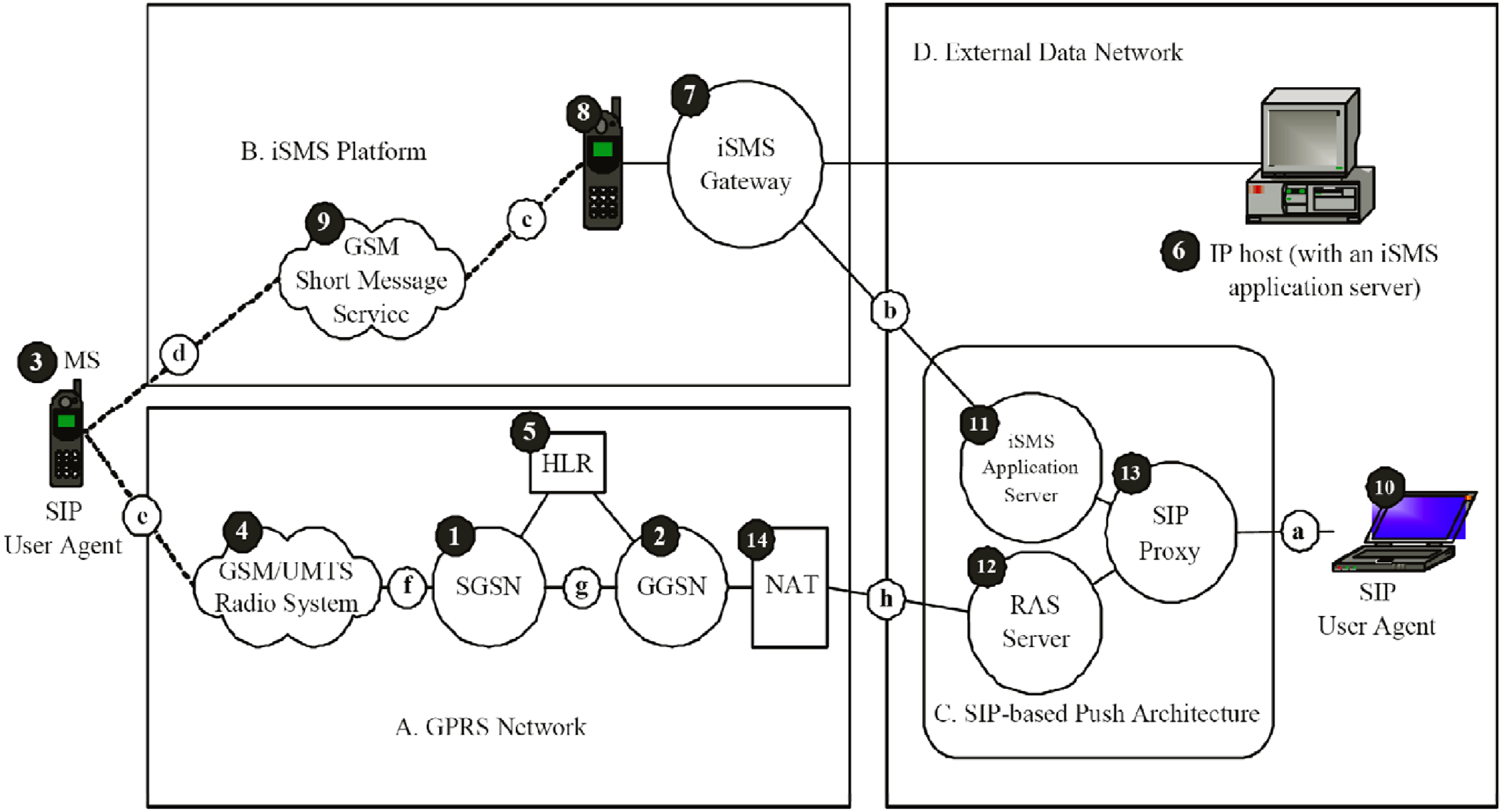
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

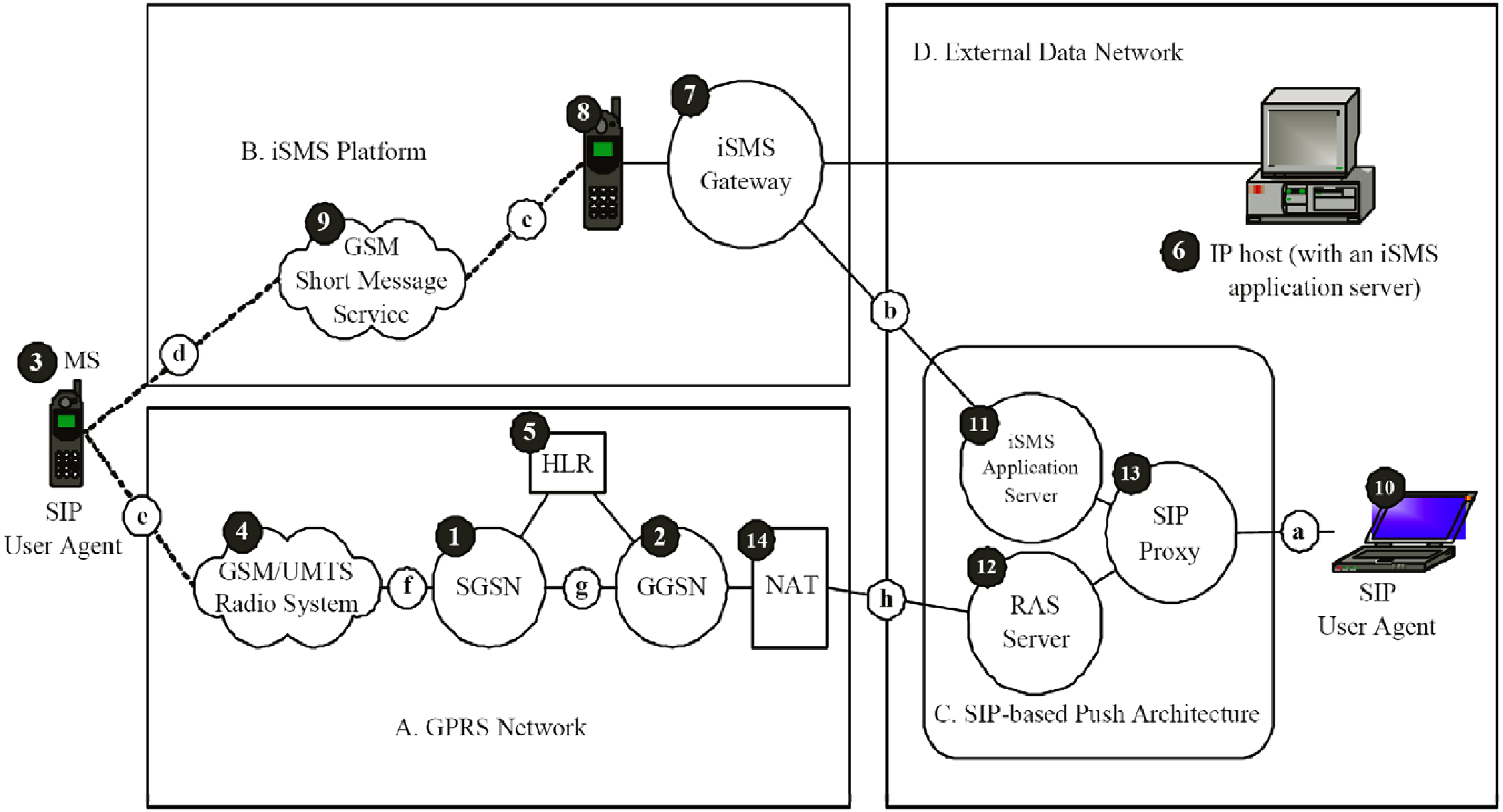
- Consider the SIP call setup procedure from an IP host in the external data network to an MS. The phone number of the MS is +886936105401 and the fully qualified domain name of the SIP proxy is fetnet.com.
- Before the call termination is initiated, the MS has not activated the VoIP service yet.
  - Steps 1–7 below are executed.

Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



- **Step 1.** The calling party initiates the call to the MS by issuing the SIP INVITE message. This message contains the SDP information that provides the RTP network address and transport port number of the calling party. In VoIP, a call party is identified by its IP address.
  - Since the IP address of the MS is dynamically assigned, this address is not available when the call termination is initiated.
- To resolve this issue, the MS is identified by the telephone number +886936105401 carried by the INVITE message using the SIP Request-URI with the following format:
- INVITE sip:+886936105401@fetnet.com

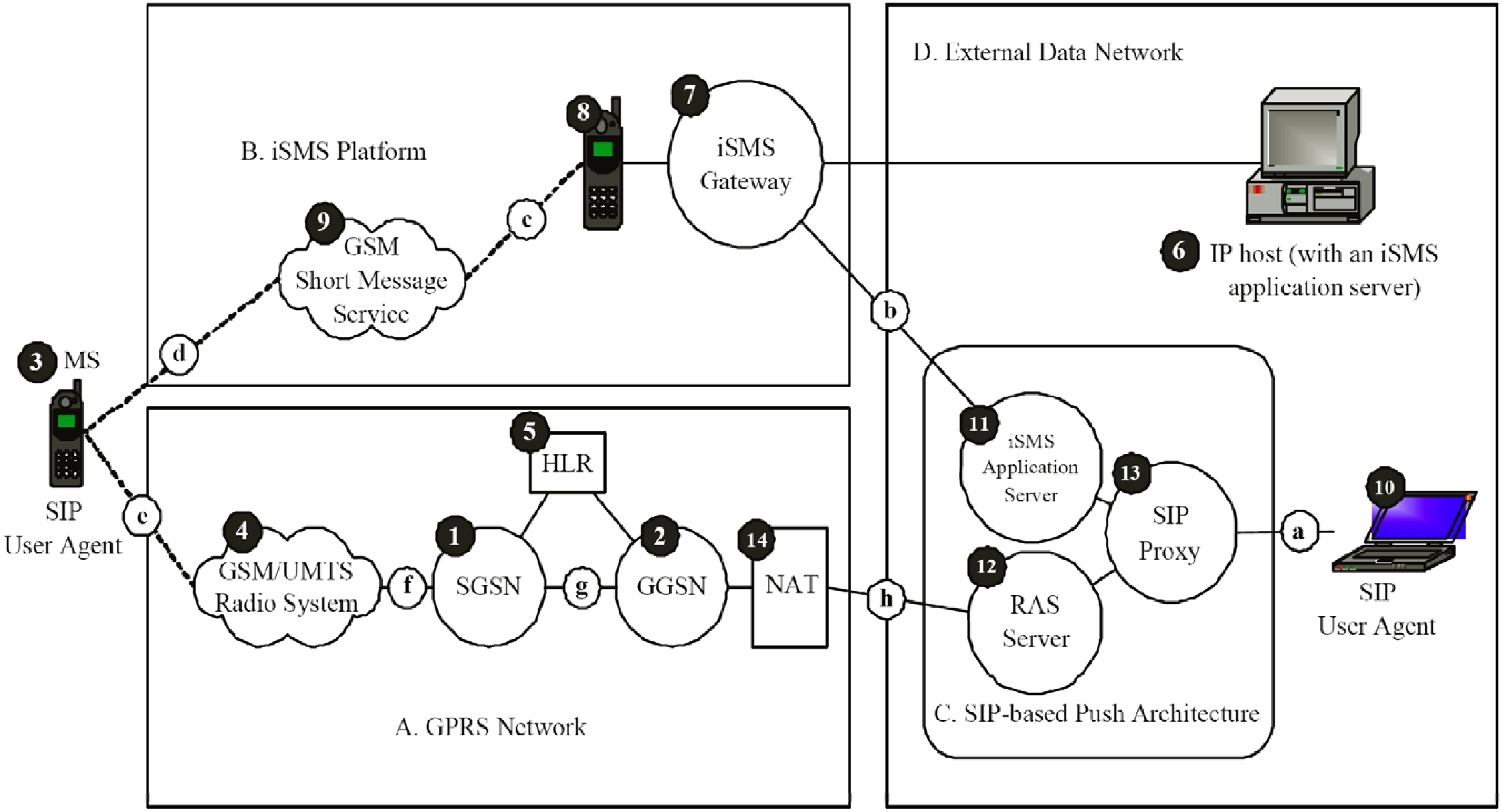
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

- The above message is routed to the SIP proxy (fetnet.com) through path (a) in Figure 12.4. Upon receipt of the INVITE message, the SIP proxy instructs the iSMS application server to send a short message to the number +886936105401 (path (b)→(c)→(d) in Figure 12.4).
- This short message carries the **public IP address of the RAS server** and **a tunnel IP address of the SIP proxy**, which will be used by the MS to establish the tunnel to the SIP proxy.

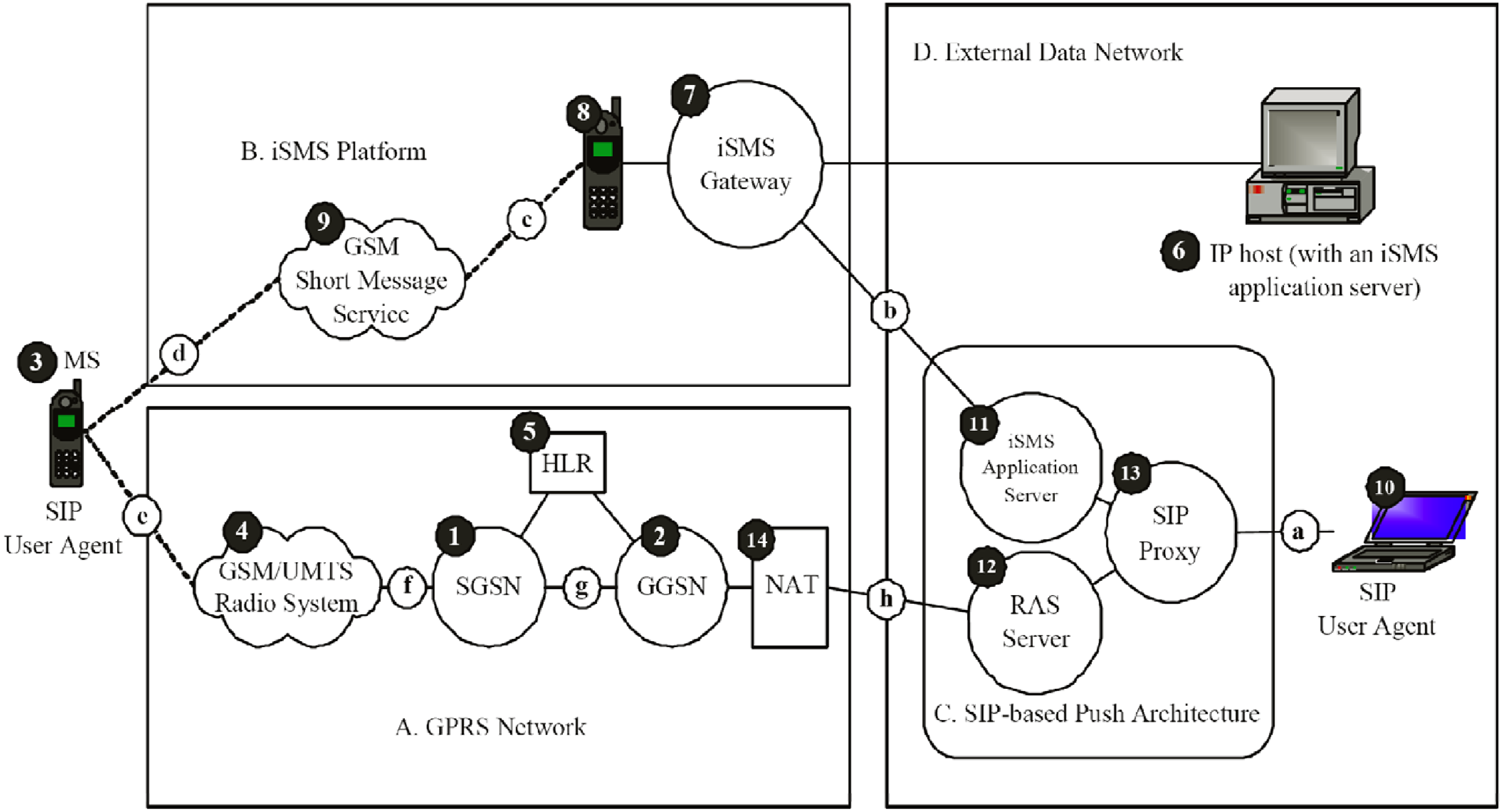
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

- **Step 2.** The short message triggers the MS to activate the PDP context for VoIP.
  - After the activation, the MS is assigned an IP address from the GGSN (see Chapter 4).

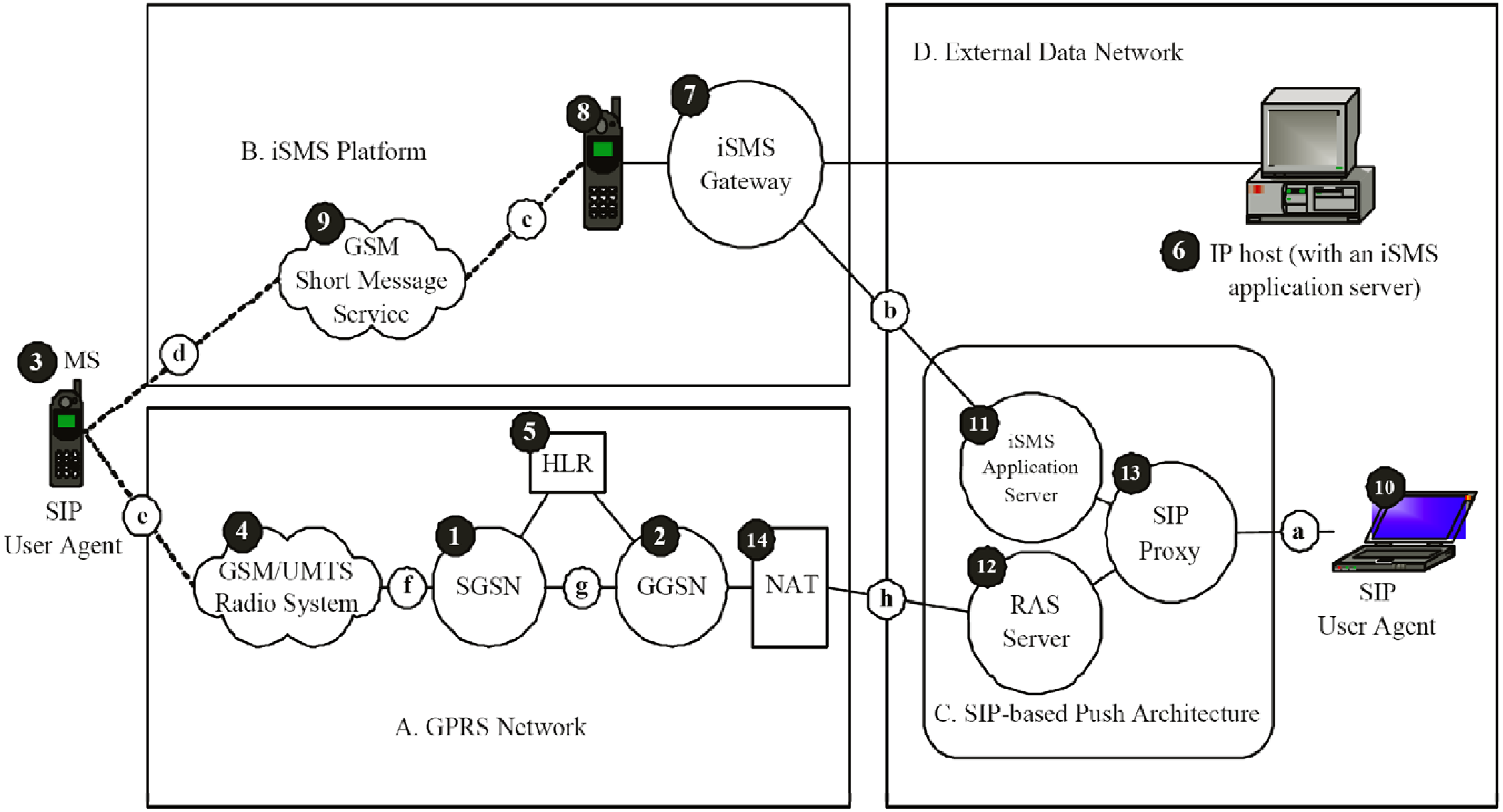
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

- **Step 3.** By using the RAS IP address and the MS IP address, the MS and the RAS server exchange L2TP messages to establish a tunnel between the MS and the SIP proxy (path (e)↔(f)↔(g)↔(h) in Figure 12.4).
- After the tunnel is established, the MS is assigned a tunnel IP address from the RAS server. Note that this tunnel IP address is different from the MS IP address assigned by the GGSN.

Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



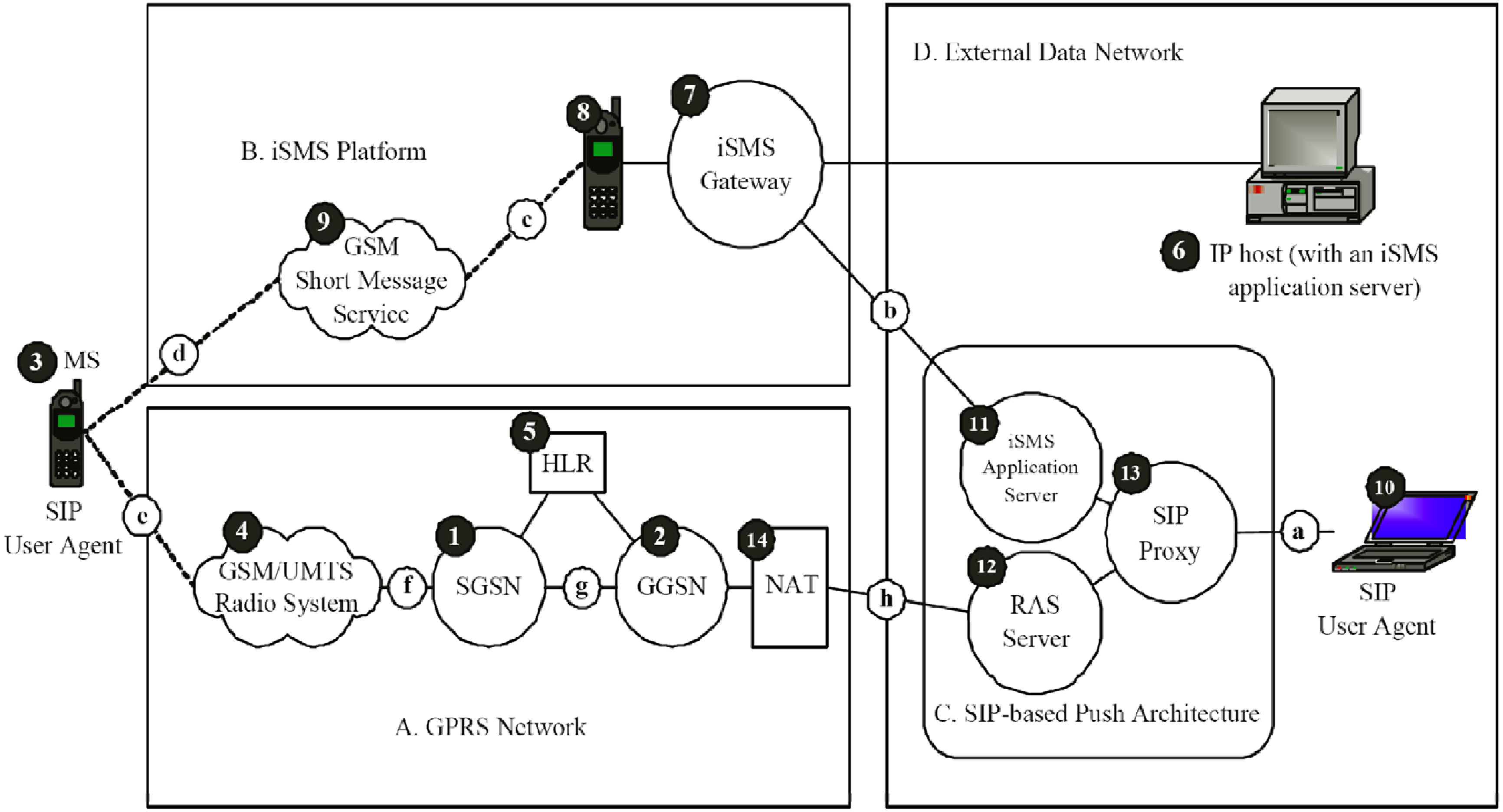
Cont.

- **Step 4.** Using the established tunnel, the MS sends its tunnel IP address and telephone number +886936105401 to the SIP proxy. This tunnel IP address and the phone number are saved in the SIP proxy.
- When the SIP proxy receives a SIP message with the destination phone number +886936105401, it will forward the message by using the tunnel IP address of the MS.

Cont.

- **Step 5.** The SIP proxy modifies the INVITE message received at Step 1.
- Specifically, the connection information field and the transport port field of the SDP are modified to the IP address and the port number of the SIP proxy. Therefore, the RTP packets will be routed from the MS to the calling party through the SIP proxy. The SIP proxy then forwards this message to the MS.

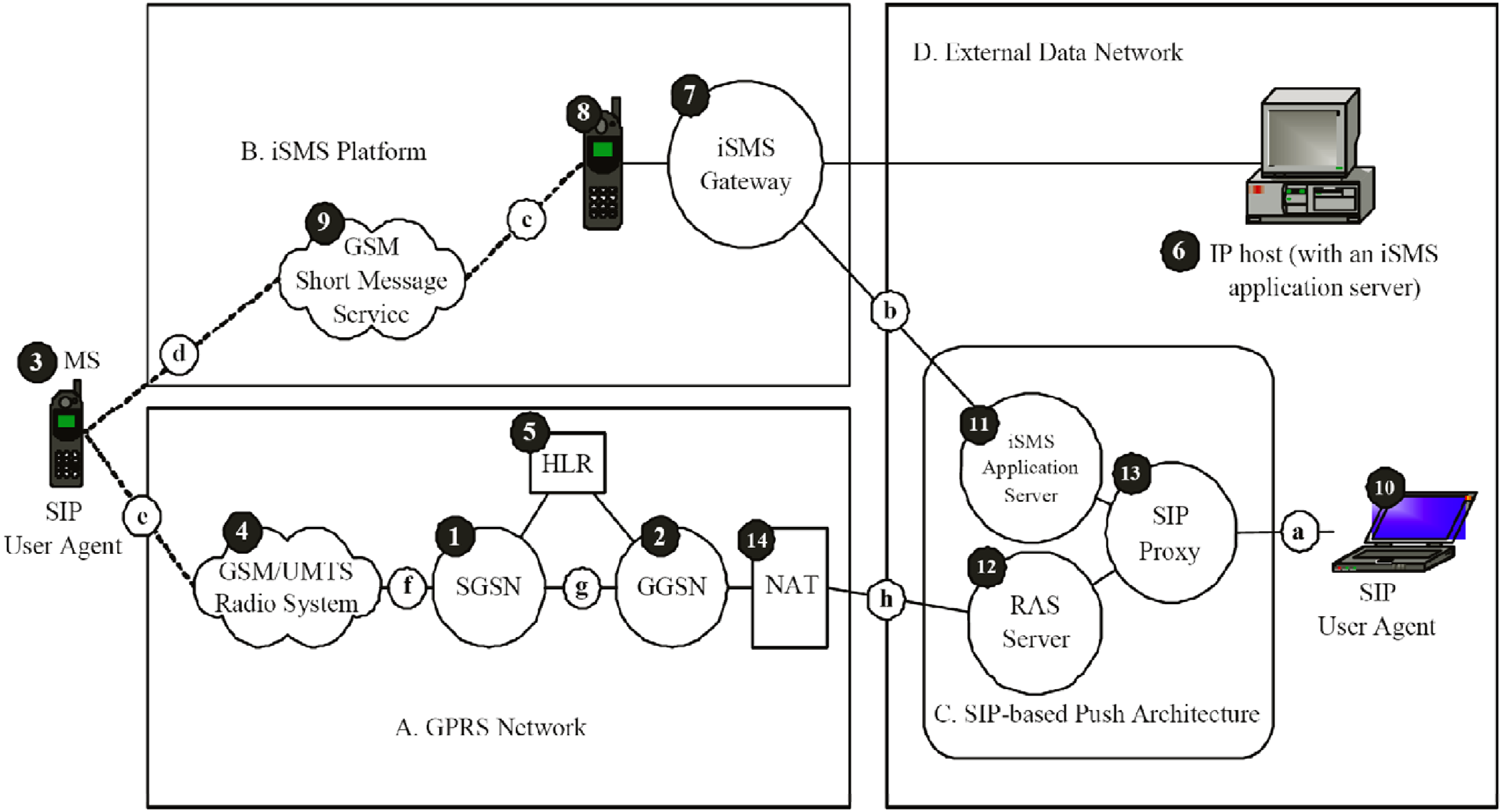
Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

- **Step 6.** Upon receipt of the modified INVITE message, the MS answers the call by sending the SIP OK response back to the SIP proxy through the tunnel established in Step 3.
  - – The SDP of this message contains the RTP information of the MS, which is also modified by the SIP proxy, similar to that in Step 5.

Fig.12.4 GPRS, iSMS and SIP-based Push Architecture



Cont.

- **Step 7.** The SIP proxy forwards the OK message to the calling party. The calling party confirms this session by sending the SIP ACK message to the MS through the SIP proxy. At this point, the conversation begins.
  - The RTP packets are routed through the path (e)↔(f)↔(g)↔(h)↔(a).

Fig.12.4 GPRS, iSMS and SIP-based Push Architecture

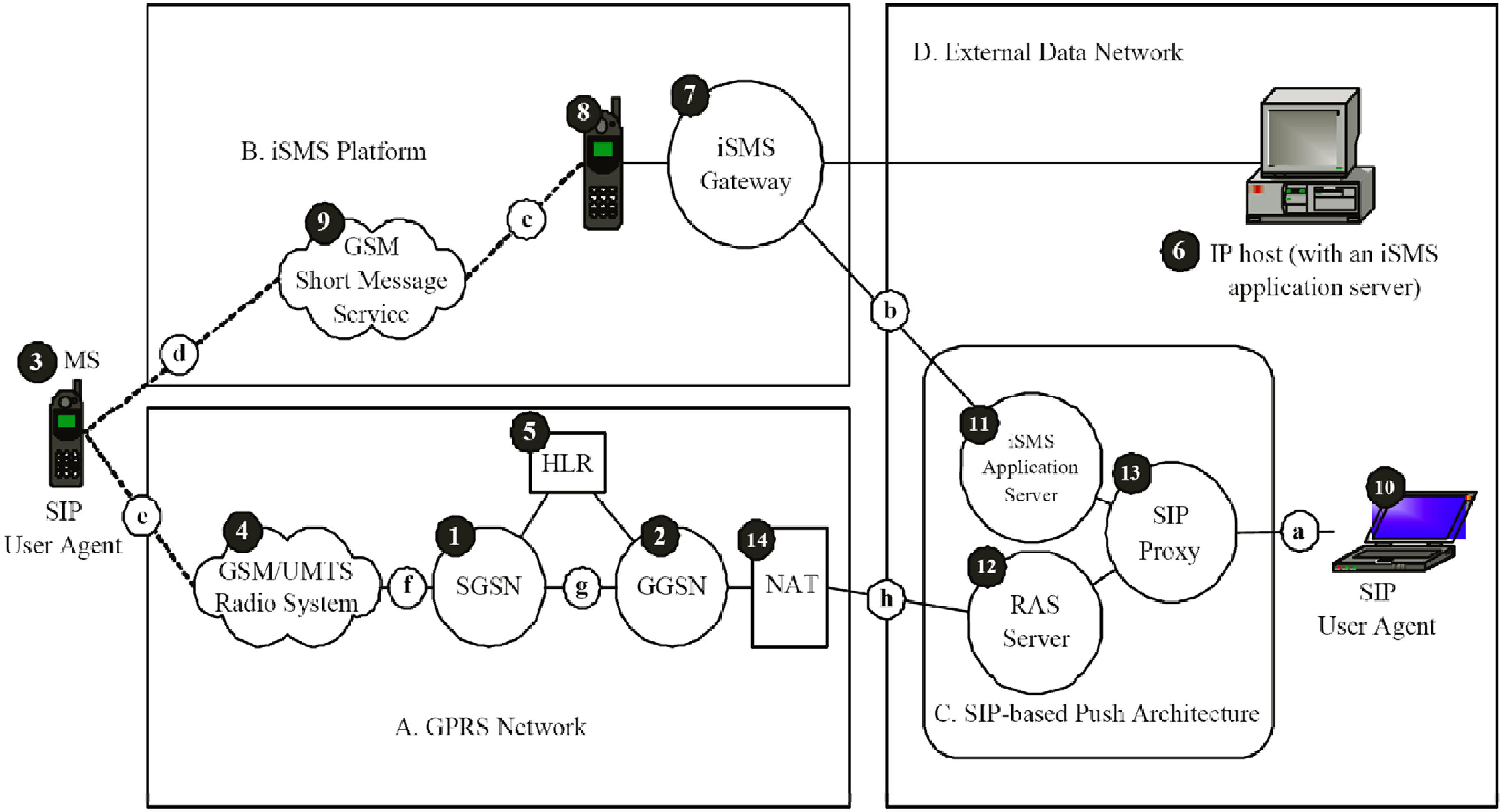


Fig.12.5 NCTU Prepaid System Architecture

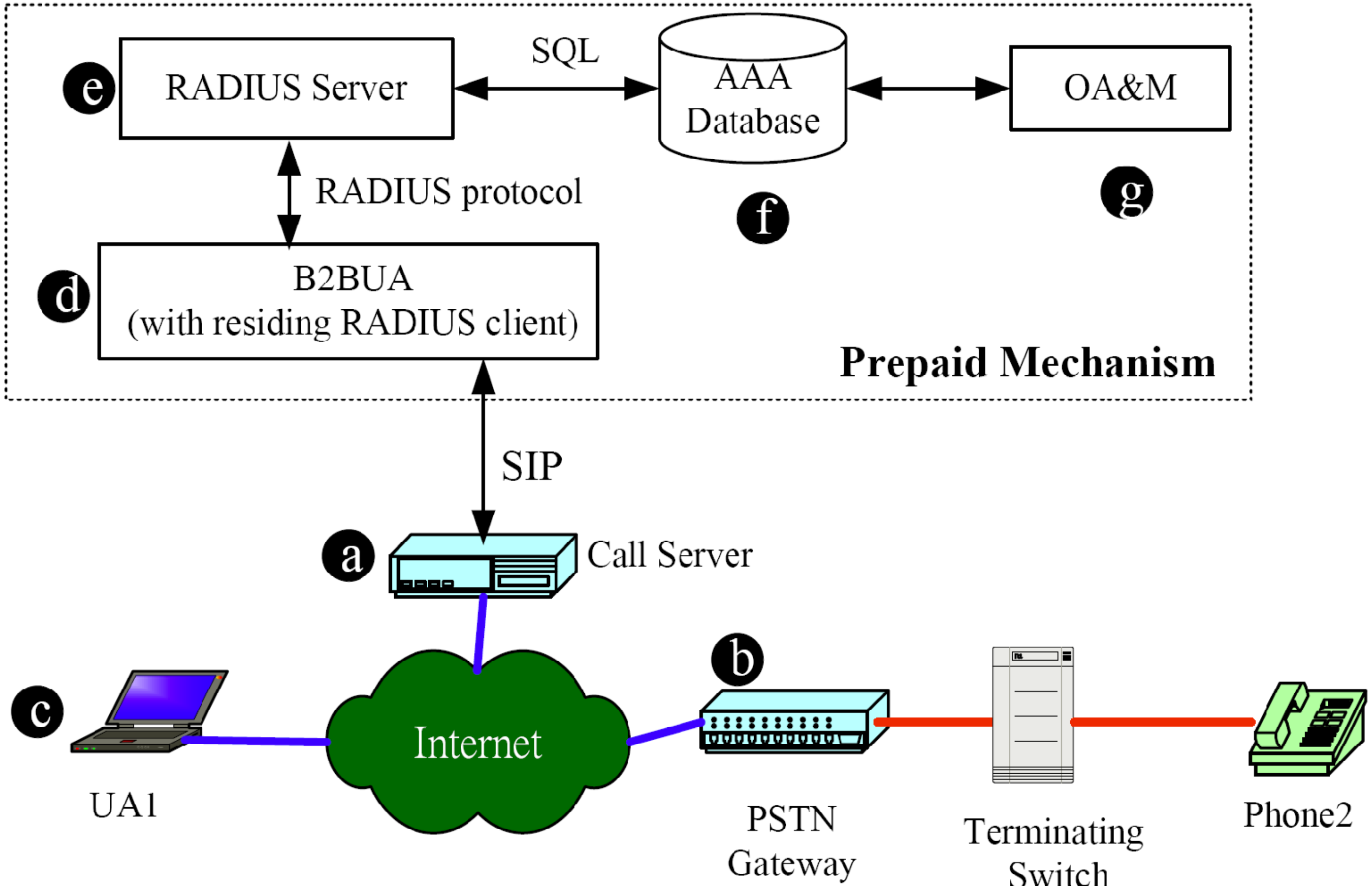


Fig.12.6 PSTN Gateway Developed in the ITRI

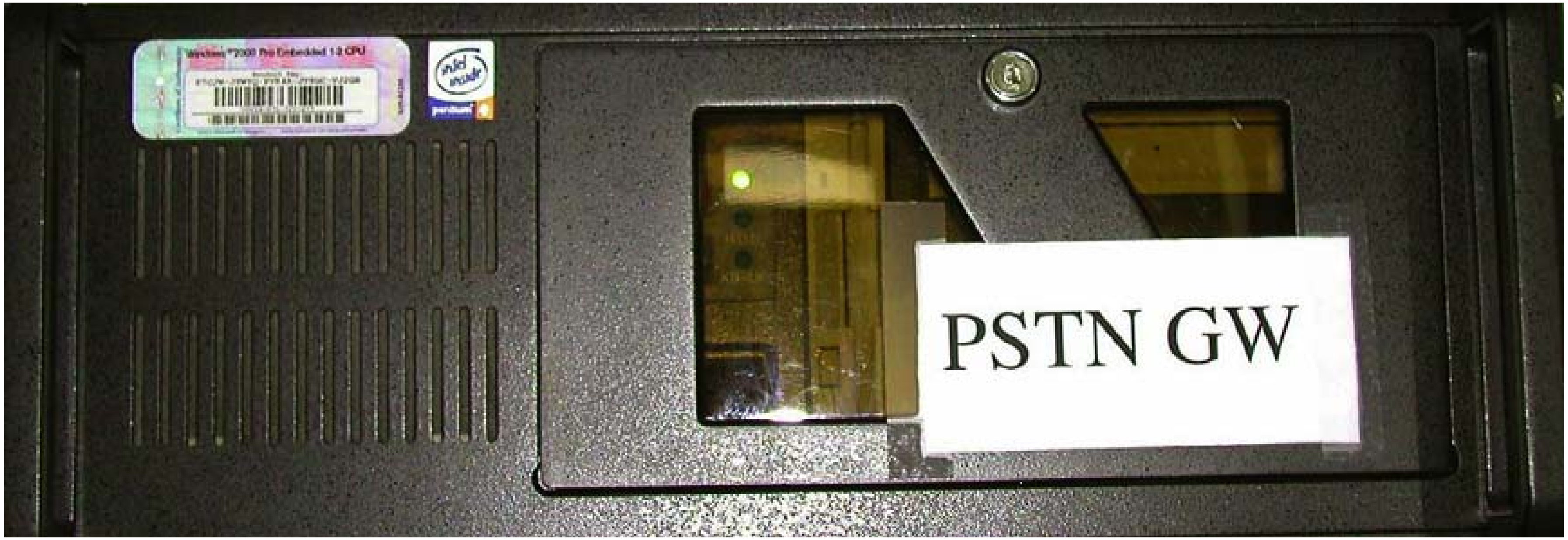
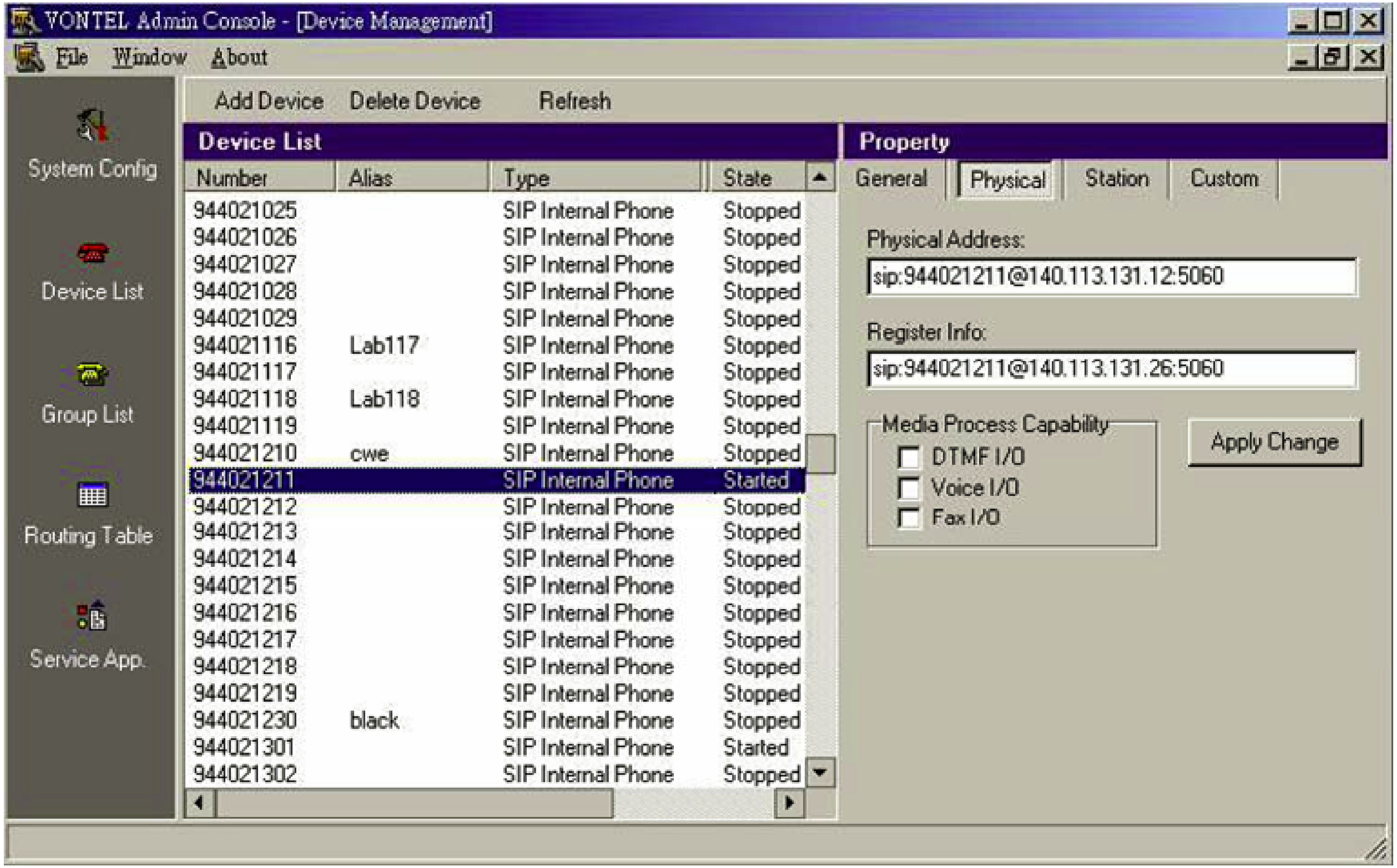


Fig.12.7 Administrative Console of the Call Server



Device List

Number	Alias	Type	State
944021025		SIP Internal Phone	Stopped
944021026		SIP Internal Phone	Stopped
944021027		SIP Internal Phone	Stopped
944021028		SIP Internal Phone	Stopped
944021029		SIP Internal Phone	Stopped
944021116	Lab117	SIP Internal Phone	Stopped
944021117		SIP Internal Phone	Stopped
944021118	Lab118	SIP Internal Phone	Stopped
944021119		SIP Internal Phone	Stopped
944021210	cwe	SIP Internal Phone	Stopped
944021211		SIP Internal Phone	Started
944021212		SIP Internal Phone	Stopped
944021213		SIP Internal Phone	Stopped
944021214		SIP Internal Phone	Stopped
944021215		SIP Internal Phone	Stopped
944021216		SIP Internal Phone	Stopped
944021217		SIP Internal Phone	Stopped
944021218		SIP Internal Phone	Stopped
944021219		SIP Internal Phone	Stopped
944021230	black	SIP Internal Phone	Stopped
944021301		SIP Internal Phone	Started
944021302		SIP Internal Phone	Stopped

Property

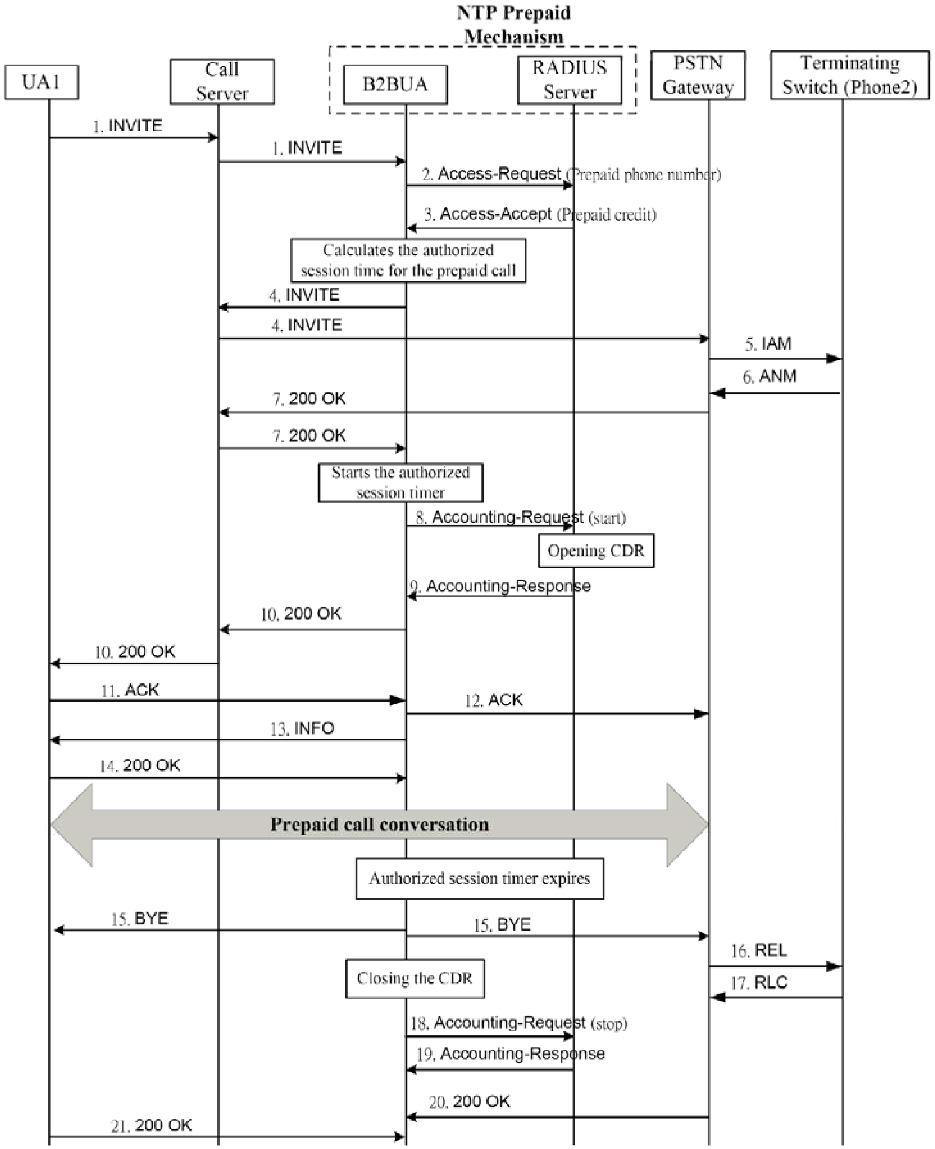
General | Physical | Station | Custom

Physical Address: sip:944021211@140.113.131.12:5060

Register Info: sip:944021211@140.113.131.26:5060

Media Process Capability
- ☐ DTMF I/O
- ☐ Voice I/O
- ☐ Fax I/O

Fig.12.8 Message Flows for Call Setup and Call Force-Termination


NTP Prepaid Mechanism
UA1
Call Server
B2BUA
RADIUS Server
PSTN Gateway
Terminating Switch (Phone2)
1. INVITE
1. INVITE
2. Access-Request (Prepaid phone number)
3. Access-Accept (Prepaid credit)
Calculates the authorized session time for the prepaid call
4. INVITE
4. INVITE
5. IAM
6. ANM
7. 200 OK
7. 200 OK
Starts the authorized session timer
8. Accounting-Request (start)
Opening CDR
9. Accounting-Response
10. 200 OK
10. 200 OK
11. ACK
12. ACK
13. INFO
14. 200 OK
Prepaid call conversation
Authorized session timer expires
15. BYE
15. BYE
16. REL
17. RLC
Closing the CDR
18. Accounting-Request (stop)
19. Accounting-Response
20. 200 OK
21. 200 OK