Location-Based Push Services in Wireless LAN Hotspots

Yen-Cheng Chen and Gang-Ming Fan
Department of Information Management
National Chi Nan University
Puli, 545 Nantou, Taiwan
ycchen@ncnu.edu.tw, berger@im.ncnu.edu.tw

Abstract

Due to the small cell size in Wireless LANs (WLANs), it is practical to develop location dependent services based upon the awareness of the WLAN access points (AP) that mobile clients currently access. One possible E-commerce application in public WLAN hotspots is the delivery of location dependent information to mobile users. In this paper, we will discuss how to provide location-based push services in WLAN hotspots. It will be shown that the location of a mobile client can be determined effectively from the SNMP traps sent by APs. APs identify each mobile client by its MAC address. To deliver information to the mobile client, we also have to know the current IP address of the mobile client. We propose an effective technique for the mappings between IP addresses and MAC addresses. A web service framework is then proposed to enable location-based push services in WLAN hotspots.

Keywords: Wireless LAN, location-based service, push service, SNMP

1. Introduction

IEEE 802.11 [1] wireless local area networks (WLANs) have been successfully deployed in corporate networks and campus networks. Recently, Internet service providers (ISPs) start to provide Internet wireless access services by installing WLAN access points (APs) in public hotspot areas, such as airports, coffee shops, and conference centers. Therefore, as in cellular telecommunication systems, there is an increasing demand for location-based services on WLANs. That is, any IEEE 802.11-enabled mobile client can obtain location dependent information based on its current location within a WLAN. For example, before departure in an airport, a user can use a PDA equipped with a WLAN adapter to obtain special discount information of the duty free shops in the airport. An efficient way to deliver such location dependent information to mobile users is the employ of "push" services. That is, an original server providing contents can asynchronously send location dependent information to a WLAN client without any explicit request from the client. In this paper, we will discuss the enabling technologies for providing location-based push services in WLAN hotspots.

WLAN hotspots are usually located in indoor environments. Hence, the coverage area of an IEEE 802.11 AP installed in a WLAN hotspot will be small, compared with the size of cells in cellular telecommunication systems. Therefore, if we can identify the AP that a mobile client accesses currently, we can determine that the mobile client is in the neighborhood of the AP. Hence, it is practical to provide location dependent information to the mobile client based on the location of the AP. In an IEEE 802.11 WLAN, when a mobile client attempts to access the network, the device will first make an association with its nearest AP. The IEEE 802.11

standard defines three association-related services: *association*, *re-association*, and *disassociation*. For effective monitoring of association-related services, many APs are capable of sending association notifications to their network manager via SNMP traps. SNMP (Simple Network Management Protocol) [] is the current standard protocol for the management of TCP/IP networks. Since most APs support SNMP, we can make use of those association-related SNMP traps to obtain the locations of users.

Whenever a mobile client successfully makes an association with an AP, the AP will immediately send an SNMP association trap to notify its management station. Similarly, the re-associations and disassociations between mobile clients and APs will also trigger the transmissions of re-association traps and disassociation traps, respectively. By receiving the SNMP traps sent from APs, we can know which mobile client is entering or leaving the coverage area of an AP. Our location determination technique doesn't require a any particular hardware or software in mobile clients. Therefore, location-based E-commerce services can be widely deployed in various mobile clients with WLAN adapters.

As well as the location of a mobile client, we should also know the current IP address of the mobile client to push E-commerce messages to the mobile client over the Internet. Currently, mobile IP mechanisms are not widely supported in WLANs. The IP address of a mobile client is usually assigned by DHCP. That is, the assigned IP address may vary with time. Besides, the IP address is also dependent on the subnet where the mobile client is connected. Therefore, to enable push services in WLAN hotspots, we also need an effective mechanism to obtain the current IP addresses of mobile clients in WLAN hotspots. From the association-related traps sent by APs, what we can identify a mobile client is its MAC addresses. In fact, the IP address of a mobile client can be seen in the default router of the mobile client. Based upon the observation, we develop an effective technique for MAC-to-IP address mappings. We first find the default router of a mobile client from the AP that is associated with the mobile client. Then, we use SNMP to retrieve management information about address mappings stored in the default router. Finally, the IP address of the mobile client can be determined. It can be seen that both location determination and address mapping can accomplished by the utilizing the inherent SNMP supported in APs and routers. Therefore, no additional hardware and software are required in network devices. As a result, by our approach, location-based push services can be developed independently of the network infrastructures. Various E-commerce applications can be developed based upon our approach. To further separate the development of location-based E-commerce applications from the location determination mechanism, we will propose two generalized web service frameworks according to the ways of providing location-based E-commerce push services.

The remainder of this paper is organized as follows. Section 2 is a review of previous location determination techniques in IEEE 802.11 WLANs. In section 3, we will describe our location determination approach based on the SNMP protocol. In section 4, we will present two web service frameworks for WLAN location-based push services. Section 5 describes a prototype implementation of a location-based push service based on our web service frameworks. Finally, the conclusion and future work are given in section 6.

2. Related Work

Location determination techniques for cellular systems have been developed in recent years. The interested reader may refer to [3] for details. In addition, there are a number of location sensing systems that require particular sensor hardware such as radio frequency (RF) and GPS components. Since special hardware is required in mobile clients, hardware-based approaches are not preferred for the wide deployment of location-based push services. A comprehensive survey of such location systems can be found in [4]. In this section, we focus on previous work that determines the location of a mobile client simply by the IEEE 802.11 WLAN infrastructure.

2.1. RSS-Based Approaches

Several previous studies [5]-[7] make use of the strengths of signals received by a mobile client to infer its current location. Triangulation [8] is the most common technology for those location determination approaches based on received signal strengths (RSSs). Theoretically, triangulation can be used to accurately determine the geographical location of a mobile client, if the distances between the mobile client and APs can be precisely estimated from the signal strengths. Unfortunately, probably due to obstructions and multipath fading effects in indoor environments, there is a considerable variation in the strengths of signals received by a mobile client at the same spatial point. Therefore, triangulation based on variable signal strengths may give inaccurate results. The deficiency can be possibly overcome by the use of a database of signal strength data, measured earlier in each possible location. Then, the location of a mobile user can be inferred by finding the most likely location whose corresponding signal strength data in the database best match received signal strengths. This kind of location determination technique is called RSS-based location fingerprinting [9]. To build the database of signal strengths, RSS-based location fingerprinting needs tedious measurements of signal strengths in the all coverage areas of WLANs. In addition, both triangulation and RSS-based location fingerprinting also require denser installation of APs to ensure that each location is within the converge areas of three APs. Such installations of APs are usually not found in current public WLAN hotspots. In addition, both techniques also require additional software in each mobile client to collect signal strengths from different APs.

2.2. Network-Based Approaches

Unlike the previous RSS-based techniques, several approaches determine the location of a mobile client by finding the AP that the mobile client is associated with. S.G.M. Koo et al. in [10] proposed a network-based approach, called the RADIUS approach, based on the use of a centralized RADIUS server. Remote Authentication Dial-In User Service (RADIUS) [11] is a service to provide centralized authentication, authorization, and accounting for network access. The RADIUS approach assumes that a RADIUS server is used for authenticating WLAN users. Each mobile client attempting to access the WLAN will send an authentication request to an AP. The AP then forwards the request to the RADIUS server. In case of a successful authentication, the time, the ID of the AP, and the MAC address of the mobile client will be recorded in the log file of the RADIUS server. By inspecting the log file, the RADIUS approach can determine which AP a mobile client is currently associated with. For performance considerations, it was recommended that location determination be implemented in the RADIUS server. Since the format of the RADIUS log file is not standardized, the log format may vary in different RADIUS servers. Therefore, the implementation of the RADIUS approach depends on the RADIUS server used.

Also in [10], an SNMP approach was proposed to provide a standard and platform-independent solution for location determination. In the SNMP approach, an SNMP polling program is used to periodically query each AP in a WLAN to obtain the MAC addresses observed by the AP. Essentially, APs are configured as transparent bridges. For forwarding frames, the MAC addresses listened by an AP will be stored in the forwarding table of the AP. In general, an MAC address can be stored in the forwarding table for 15 to 20 minutes even if the corresponding device has stopped an association with the AP. As a result, it is possible that the same MAC address may appear in several APs. This complicates the location determination process using the SNMP approach. In addition, the periodic polling by SNMP may consume considerable network resources and may result in longer response time than the RADIUS approach.

Both the RADIUS approach and the SNMP approach use the MAC addresses of mobile clients to perform location determination. As described previously, the IP addresses of mobile clients should also be known for location-based services. In [12], the IP address of a mobile client is found in the log of the DHCP server for a WLAN. Whenever the DHCP successfully assigns an IP address to a mobile client, the IP address as well as the MAC address of the mobile client will be recorded in the log file. Therefore, MAC-to-IP address mappings can be obtained by inspecting the log file. Since there is no standard format for the log file of a DHCP server, the DHCP approach also depends on a particular implementation of the DHCP server. In addition, in public WLAN hotspots, hotspots are located in different places. Thus, a number of DHCP servers will be used. For efficiency, we should first know which DHCP server has the correct address mapping. The previous approach didn’ t consider this issue.

3. A New SNMP Approach

Platform independence is an important factor to facilitate the wide use of location-based push services in public WLAN hotspots. Since RSS-based approaches for location determination require specialized software in mobile clients, they are platform dependent. RADIUS approaches can only be adopted in WLANs using RADIUS authentications. Furthermore, it is also platform dependent, since it should be integrated with a particular implementation of a RADIUS server. Among previous approaches, the SNMP approach is the most promised one to achieve platform independence. However, as described in the previous section, the periodic polling by SNMP may generate heavy signaling traffic and results in poor performance. In this section, we will propose an alternative solution for location determination based on the trap mechanism of SNMP. We will show that no any periodic polling is required if APs have the capability of sending SNMP traps to report the association-related events in the APs.

3.1. SNMP Trap Approach

Before a mobile client is allowed to send data messages via an AP, the mobile client shall first become associated with the AP. This can be accomplished by the association service specified by IEEE 802.11. IEEE 802.11 also specifies the reassociation service and the disassociation service. The reassociation service is invoked to enable a current association to be transferred from one AP to another. The disassociation service is invoked whenever an existing association is to be terminated. IEEE 802.11 recommends that an SNMP trap should be sent when a disassociation occurs. For effective management of associations, many enterprise-level APs [14]-[17] will also send out SNMP traps after associations and reassociations take place. Thus, sending association-related traps is a common feature found in most APs. The information carried in an association-related trap includes the MAC address of the mobile client involved in an association-related service. The IP address of the AP sending the trap is also in the agent address field of an SNMP trap message. As a result, by interpreting the traps sent from all APs, we can determine which AP a mobile client is currently connected to or disconnected from. Fig. 1 illustrates our SNMP trap approach.

In the SNMP trap approach, only a location server is required in the wired network. The location server is responsible for receiving traps and performing location determinations. Each AP is configured so that all traps are sent to the location server. In the location server, a trap listener program runs as a daemon process to receive and handle incoming traps. The location server also contains a location database to store location information. In the location database, an AP table is used to store the information about each AP, including its identifier (ID), IP address, and a textual location description. The location database also contains an association table. A record in the association table consists of the MAC address of a mobile client and the ID of the AP that the mobile client is currently associated with. The association table is updated according to the types of traps received:

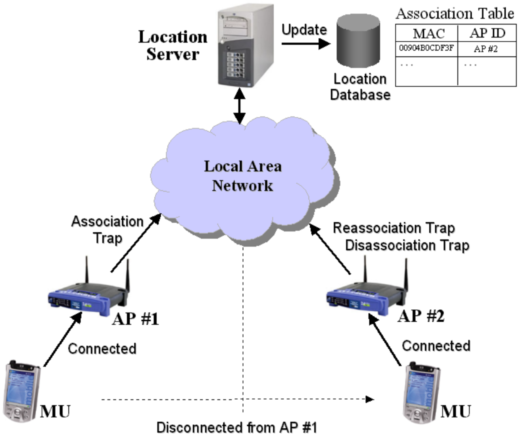


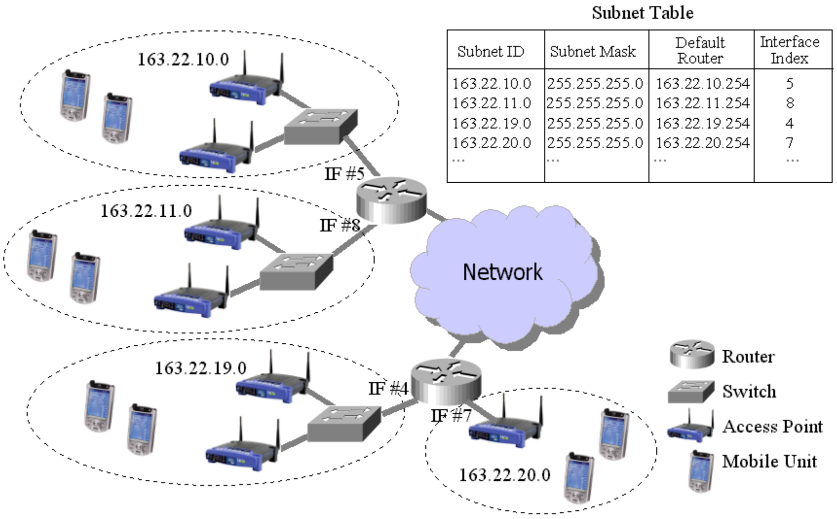
Fig. 1 The SNMP trap approach

(1). *Association Trap*: Insert a new record to store the MAC address of a mobile client and the ID of the AP that sent the trap.

(2). *Reassociation Trap*: Update the AP ID field of the record that has an MAC address same as the one shown in the reassociation trap.

(3). *Disassociation Trap*: Delete the record that has an MAC address same as the one shown in the disassociation trap.

By the above traps, we can effectively maintain MAC-to-AP mappings in the association table without introducing additional network traffic as incurred in the previous SNMP polling approach. Furthermore, unlike the use of ARP caches in previous approaches, the MAC-to-AP mappings derived from SNMP traps are always most up-to-date. Therefore, the location of a mobile client can be determined correctly.



Subnet Table

Subnet ID	Subnet Mask	Default Router	Interface Index
163.22.10.0	255.255.255.0	163.22.10.254	5
163.22.11.0	255.255.255.0	163.22.11.254	8
163.22.19.0	255.255.255.0	163.22.19.254	4
163.22.20.0	255.255.255.0	163.22.20.254	7
...	...	...	...

Fig. 2 A WLAN configuration and its subnet table

3.2. MAC-to-IP Address Mapping

Essentially, WLAN APs are layer 2 devices. They identify each mobile client by its MAC address. That is why we only obtain the MAC address of a mobile client from the traps sent by APs. Location-based push services send location dependent information to a mobile client without any explicit request from the mobile client. When a location-based push service attempts to “push” a message to a mobile client, the location-based service must know the IP address currently used by the mobile client. Up to now, we can obtain the MAC address of a mobile client from the association table. Therefore, we need MAC-to-IP address mappings to get the IP address of a mobile client. In the following, we will show how the address mappings can be obtained via SNMP.

To efficiently perform addressing mapping, we first prepare a subnet table in the location database. The subnet table consists of four fields: Subnet ID, Subnet Mask, Default Router, and Interface Index. Each record in the subnet table corresponds to a subnet that could contain mobile clients. In general, when a mobile client accesses a network via an AP, the mobile client is in a subnet that also contains the AP. Therefore, the subnets containing APs will also be those ones that can contain mobile clients. By the AP table, we can know which APs are involved in the location-based push services. Therefore, we can find the subnets from the APs. For each AP, we send an SNMP get-request to the AP to get its subnet mask, which will be stored in the Subnet Mask field of the subnet table. The subnet mask is available in the *ipAdEntNetMask* object of the *ipAddrTable* table, defined in MIB II. After the Subnet Mask is determined, the Subnet ID can be obtained by the bit-wised AND of the Subnet Mask and the IP address of the AP. The Default Router field indicates the default router of the mobile clients in a subnet. In fact, the default router of an AP is also the one of the mobile clients in the same subnet. For an AP, the IP address of its default router can be obtained in the *ipRouteTable* table of MIB II. In *ipRouteTable*, the entry with a value of 0.0.0.0 in the *ipRouteDest* object indicates a default route. The *ipRouteNextHop* object in the default route entry will be the IP address of the default router. The Interface Index field indicates the index number of a network interface in the default router. The network interface is used to connect to a subnet that can contain mobile clients. We can determine the index number of the network interface from the *ipAdEntIfIndex* object in the *ipAddrTable* table of MIB II. Figure 2 shows a typical WLAN configuration with its corresponding subnet table.

Given the MAC address of a mobile client, we determine the current IP address of the mobile client by the following steps.

(1). Determine the AP that the MAC address is associated with. This information is available in the association table.

(2). Using the IP address the AP, we search the subnet table to find the subnet that contains the AP.

(3). After the subnet is found, we identify the default router, which will contain the IP address of the mobile client.

(4). Use *Interface Index* and the *Subnet ID* of the subnet table as the start instance identifier of the SNMP Get-Next-requests followed.

(5). We use Get-Next-requests to perform an IP address search in the *ipNetToMediaTable* table of the default router until the IP address is found.

Note that the *ipNetToMediaTable* table is indexed by interface index and IP address. The interface index has been known from the subnet table. Obviously, the IP address is unknown. Therefore, we cannot determine the IP address of the mobile client in a single SNMP request. To avoid retrieving the entire *ipNetToMediaTable* table, we use *Interface Index* and the *Subnet ID* of the subnet table as the start instance identifier of SNMP queries. Then, we can quickly find the resulting IP address in a few SNMP get-next requests. As a result, only a very limited part of the *ipNetToMediaTable* should be retrieved. If SNMPv2 is supported in the router, the use of get-bulk requests can further reduce the SNMP traffic in step (5).

4. Web Service Frameworks

To hide the complexity of the location determination mechanism from upper location-based push services, we will implement the location determination as a web service. We propose two generalized web service frameworks based on how location-based E-commerce push services can be provided in the real world. In general, location-based push services can be provided in two ways: *user-oriented* and *location-oriented*. User-oriented push services send a mobile client location-dependent messages that are only interesting for the client. That is, mobile clients may receive different messages even they are in the same location. Personalized push services are user-oriented. On the other hand, location-oriented push services push identical messages to all the clients that are present in a specific location. Location-dependent advertisements or broadcasts are typical examples of location-oriented push services. That is, location-oriented push services don’t care who receive the push messages, but require that all the mobile clients in a location be notified.

4.1. A User-Oriented Web Service Framework

The user-oriented web service framework, as shown in Figure 3, is composed by mobile clients, WLAN APs with default routers, push initiators, and a location server with a location database. Mobile clients receive location-dependent push messages via a *push user agent* program. Push initiators prepare location-dependent messages and push them to mobile clients via a push protocol. The WLAN APs with their default routers together provide necessary management information for location determinations. The location server is the place where location determinations are performed. All location information is stored in the location database. To provide a standard way to access location information, we design a web service in the location server. The web service provides location query services using the Simple Object Access Protocol (SOAP) [18]. Push initiators can obtain location information via SOAP messages. SOAP is an XML-based protocol for exchange of information in web environments. The SOAP protocol between push initiators and the location server is implemented over HTTP. Four SOAP messages are defined in the user-oriented web service framework. The *PushEnable* SOAP message, sent by location server, is used to inform a push initiator that a certain mobile client is entering a specific location. The *IPRequest* SOAP message, sent by a push initiator, is used to query the location server for the IP address of a certain mobile client. The *IPResult* SOAP message, sent by the location server, is used to carry the query result of a previous IP query request. The *PushDisable* SOAP message, sent by the location server, informs a push initiator that a certain mobile client is leaving a location.

In the user-oriented web service framework, when a mobile client makes an association with an AP or moves to a new location covered by another AP, a push initiator will be notified. The push initiator will first check whether there is any location-dependent information for the mobile client. If any, the push initiator will query the location server to get the current IP address of the mobile client. Then, the push initiator can push messages destined for the IP address. The interactions among the components of the web service framework are described in the following typical scenario, as illustrated in Fig. 3.

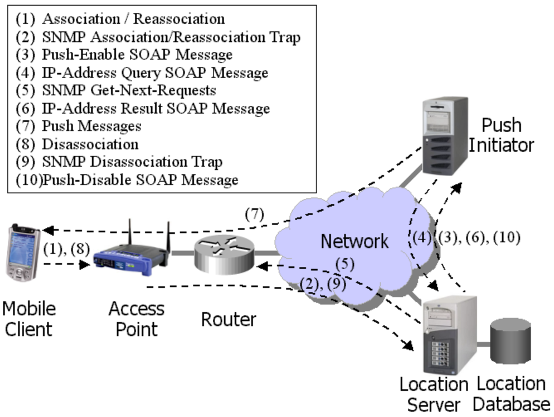

(1) Association / Reassociation
(2) SNMP Association/Reassociation Trap
(3) Push-Enable SOAP Message
(4) IP-Address Query SOAP Message
(5) SNMP Get-Next-Requests
(6) IP-Address Result SOAP Message
(7) Push Messages
(8) Disassociation
(9) SNMP Disassociation Trap
(10)Push-Disable SOAP Message
Push Initiator
(7)
(1), (8)
Network
(4)
(3), (6), (10)
(5)
(2), (9)
Mobile Client
Access Point
Router
Location Server
Location Database

Fig. 3 A user-oriented web service framework

(1). A mobile client makes an association (reassociation) with an AP.

(2). After the association (reassociation) is made, the AP sends an SNMP association (reassociation) trap to the location server. The location server stores the MAC address of the mobile client with the ID of the AP in the association table of the location database.

(3). After receiving an association (reassociation) trap, the location server will send a *PushEnable* SOAP message to the push initiator.

(4). Upon receiving the *PushEnable* SOAP message, the push initiator determines whether there are push messages for the mobile client. If any, the push initiator sends an IPRequest SOAP message to the location server to obtain the IP address of the mobile client.

(5). The location server determines the default router of the mobile client by referring to both the association table and the subnet table stored in the location database. Then, the location server sends SNMP get-next requests to the default router to find out the IP address of the mobile client.

(6). After obtaining the IP address of the mobile client, the location server returns the IP address to the push initiator via an IPResult SOAP message.

(7). The push server pushes location-dependent messages to the mobile client.

(8). The mobile client is disassociated from an AP.

(9). The AP sends an SNMP disassociation trap to the location server. The location server then deletes the record for the mobile client in the association table.

(10). Since the mobile client has been disassociated from the AP, the location server sends a PushDisable SOAP message to the push initiator to tell that the mobile client is leaving a specific location.

4.2. Location-Oriented Web Service Framework

Figure 4 illustrates the location-oriented web service framework for location-based push services in WLAN hotspots. Two SOAP messages are defined in the web service framework. The *MUQuery* SOAP message, sent by a push initiator, is used to query the location server for the IP addresses of the mobile clients in a given location. The *MUQueryResult* SOAP message, sent by the location server, is used to return the query result of IP addresses.

When a push initiator needs to send a location-dependent message to all the mobile clients in a specific location, the push initiator will query the location server for the IP addresses of the mobile clients in the given location. The location server gets the IP address from the default router of the AP installed in the location. Then, the location server returns the IP addresses to the push initiator via a SOAP message. Finally, the push initiator starts to push messages to mobile clients. The interactions among the components of the web service framework are described in the following typical scenario, as illustrated in Fig. 4.

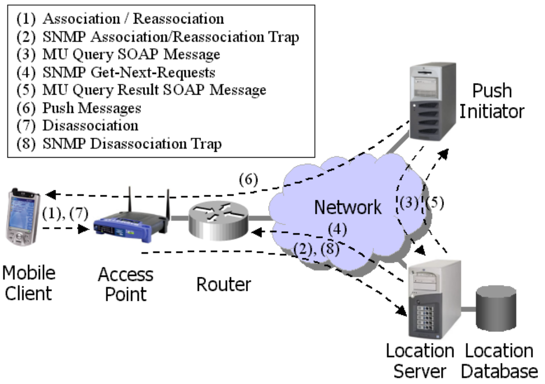

(1) Association / Reassociation
(2) SNMP Association/Reassociation Trap
(3) MU Query SOAP Message
(4) SNMP Get-Next-Requests
(5) MU Query Result SOAP Message
(6) Push Messages
(7) Disassociation
(8) SNMP Disassociation Trap
Push Initiator
(6)
(1), (7)
Network
(3)
(5)
(4)
(2), (8)
Mobile Client
Access Point
Router
Location Server
Location Database

Fig. 4 A location-oriented web service framework

(1). A mobile client makes an association (reassociation) with an AP.

(2). After the association (reassociation) is made, the AP sends an SNMP association (reassociation) trap to the location server. The location server stores the MAC address of the mobile client and the ID of the AP into the association table of the location database.

(3). When a push initiator has location-dependent information for mobile clients, the push initiator will send a *MUQuery* SOAP message to the location server to get a list of the IP addresses of the mobile clients in a given location.

(4). From the AP table, the location server first finds the AP installed in the given location. Then, from the association table, the location server finds the MAC addresses of all the mobile clients associated with the AP. Then, the location server gets the corresponding IP addresses from the default router of the mobile clients.

(5). The location server returns the list of IP addresses to the push initiator via a *MUQueryResult* SOAP message.

(6). The push initiator pushes information to mobile clients.

(7). A mobile client is disassociated from an AP.

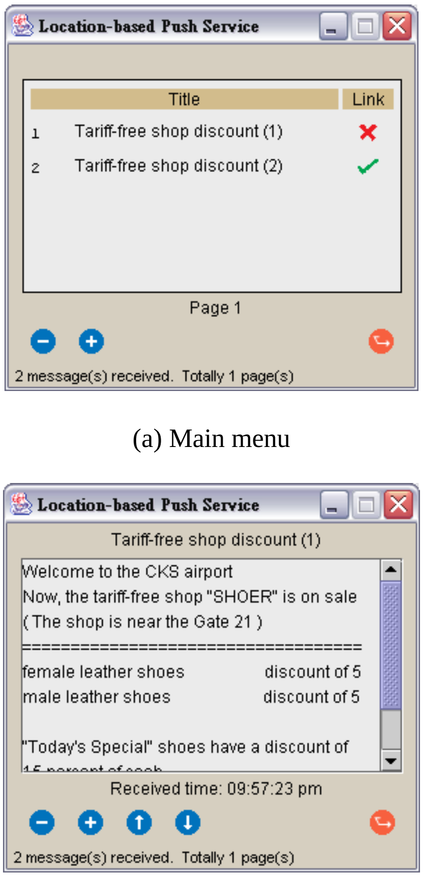
(8). The AP sends an SNMP disassociation trap to the location server. The location server then deletes the record for the mobile client in the association table.

5. A Prototype Implementation

In this paper, we have proposed a novel location determination technique based on the SNMP supports in APs and routers. To verify the correctness of the location determination method, we have implemented a location server in the campus WLAN of the National Chi Nan University. The WLAN is composed by Cisco Aironet 350 APs [14], supporting the IEEE 802.11b WLAN standard. We also implemented a prototype of location-based push services based on the web services provided in the location server. In the prototype implementation, we choose twelve APs to simulate public WLAN hotspots. Three routers are used to connect the APs. A push initiator is developed to provide both location-oriented and user-oriented push services. As Figure 3 and 4 illustrated, the implementation consists of a mobile client application, a push initiator and a location server, all written in Java. The required SNMP communication in the implementation is developed using AdventNet SNMP APIs.

5.1. Mobile Client Application

To receive asynchronous push messages from the push initiator, we implement a *push user agent* in the mobile client. The push user agent is a program that interprets the incoming push messages and presents the messages in a user interface, as shown in Figure 5(a) and 5(b).



Location-based Push Service

	Title	Link
1	Tariff-free shop discount (1)	✗
2	Tariff-free shop discount (2)	✓

Page 1

2 message(s) received. Totally 1 page(s)

(a) Main menu

Location-based Push Service

Tariff-free shop discount (1)

```
Welcome to the CKS airport
Now, the tariff-free shop "SHOER" is on sale
( The shop is near the Gate 21 )
==================================
female leather shoes          discount of 5
male leather shoes            discount of 5

"Today's Special" shoes have a discount of
15 percent of each
```

Received time: 09:57:23 pm

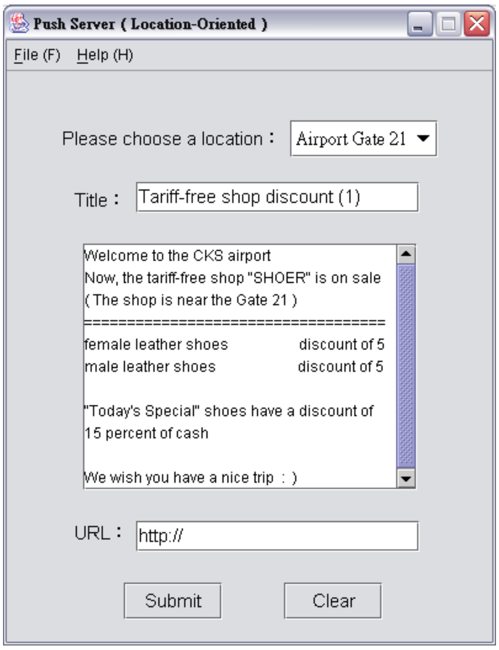
2 message(s) received. Totally 1 page(s)

(b) A push message

Fig. 5 User interface of the push user agent

5.2. Push Initiator

A push initiator pushes location-dependent messages to mobile clients. In real applications, user interfaces are not required in the push initiator. In our prototype implementation, to simulate the issuing of location-dependent messages in a location-oriented push service, we implement a user interface, as shown in Figure 6, to allow us to test whether messages are correctly pushed to mobile clients in a specific hotspot. Figure 6 shows the simulation of location-based push services in hotspots located within an airport.



Please choose a location : Airport Gate 21

Title : Tariff-free shop discount (1)

```
Welcome to the CKS airport
Now, the tariff-free shop "SHOER" is on sale
( The shop is near the Gate 21 )
==================================
female leather shoes          discount of 5
male leather shoes            discount of 5

"Today's Special" shoes have a discount of
15 percent of cash

We wish you have a nice trip  : )
```

URL : http://

Fig. 6 Simulating location-oriented push services

5.3. Location Server

The location server consists of an SNMP trap listener and a web service program. The SNMP trap listener is responsible for receiving SNMP traps sent from APs. The web service program is responsible for receiving and sending SOAP messages. A location database is also located in the location server. The location database stores three tables: *AP table*, *association table*, and *subnet table*. All the three tables are maintained by the location server.

6. Conclusions and Future Work

Previous studies [7], [10] indicated that the MAC addresses of mobile clients cached in APs might be invalid for location determinations. Instead of the use of the cache, we have proposed an effective approach based on the use of SNMP traps to obtain most up-to-date MAC addresses associated with APs. To enable push services in the application level, we have also proposed an effective approach for mapping MAC addresses to IP addresses. Consequently, location-dependent messages can be pushed to mobile clients by knowing their IP addresses.

We have presented two practical web service frameworks to enable location-based push services in WLAN hotspots. In our prototype implementation, we have shown that location-dependent messages can be properly sent to mobile clients. An issue in the implementation of location-based push services is the lack of a standard push protocol in WLAN environments. Current available push protocols are developed for cellular systems. For example, the WAP PAP (Push Access Protocol) protocol is developed for GSM or GPRS networks. 3GPP also defines push services for 3G systems. Our future work will focus on is the development of a standard push protocol designed for WLANs. Another important issue is how to deploy location-based push services for mobile commerce (M-commerce) applications. We can foresee that WLAN accesses will be popular in public hotspots. This will raise the demand for location-based push services. We have much interest in how service providers can get profit from location-based push services in public WLAN hotspots.

References

[1] IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Std 802.11 1999. The Institute of Electrical and Electronics Engineers, 1999.

[2] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", RFC 1157, May 1990.

[3] "Location Technologies for GSM, GPRS and WCDMA Networks," White Paper,

SnapTrack, A QUALCOMM Company, November 4, 2001.

[4] Jeffrey Hightower and Gaetano Borriello, “A Survey and Taxonomy of Location Systems for Ubiquitous Computing,” Extended paper from Computer, 34(8) p57-66, August 2001.

[5] P. Bahl, V. N. Padmanabhan, “RADAR: An In –Building RF-Based User Location and Tracking System,” in *Proc. IEEE INFOCOM* 2000, Vol. 2, pp. 775-784, March 2000.

[6] Paul Castro, Patrick Chiu, Ted Kremenek, and Richard Muntz. "A probabilistic location service for wireless network environments," in *Proc. Ubicomp* 2001, pp. 18-24., September 2001.

[7] Smailagic, A., Kogan, D. “Location Sensing and Privacy in a Context-Aware Computing Environment,” *IEEE Wireless Communications*, vol. 9, no. 5, Oct 2002

[8] K. Pahlavan and P. Krishnamurthy, *Principles of Wireless Networks: A Unified Approach*, Prentice Hall PTR, 2002.

[9] P. Prasithsangaree, P. Krishnamurthy, and P. K. Chrysanthis. “On Indoor Position Location with Wireless LANs,” in *Proc. the 13th IEEE Int'l Symposium on Personal, Indoor, and Mobile Radio Communications*, Lisbon, Portugal, Sept. 2002.

[10] S. G. M. Koo, C. Rosenberg, H. -H. Chan, and Y. C. Lee. “Location Discovery in Enterprise-based Wireless Networks: Implementation and Applications,” in *Proc. the 2nd IEEE Workshop on Applications and Services in Wireless Networks* (ASWN 2002), Paris, France, Jul 3-5, 2002.

[11] Rigney, C., Willens, S., Rubens, A. and W. Simpson, "Remote Authentication Dial in User Service (RADIUS)", RFC 2865, June 2000.

[12] S. G. M. Koo, C. Rosenberg, H. Chan, and Y. C. Lee. “Location Discovery in Enterprise-based Wireless Networks: Case Studies and Applications,” *Annals of Telecommunications*, to be published.

[13] K. McCloghrie, T. Rose, “Management Information Base for Network Management of TCP/IP –based internets,” RFC 1213, March 1991.

[14] Cisco Aironet 350 Series Access Points, http://www.cisco.com/warp/public/cc/pd/witc/ao350ap/

[15] Symbol Technologies' Spectrum24 4131 Access Point, http://www.symbol.com/products/wireless/ap4131.html

[16] Intel PRO/Wireless 2011B LAN Access Point, http://www.intel.com/network/connectivity/products/2011_lan_access.htm

[17] WAP11 - Instant Wireless Network Access Point, ftp://ftp.linksys.com/datasheet/wapIIds.pdf

[18] Simple Object Access Protocol, http://www.w3.org/TR/SOAP/