Location-based E-campus Web Services: From Design to Deployment*

Simon G. M. Koo, Catherine Rosenberg, Hoi-Ho Chan, and Yat Chung Lee
School of Electrical and Computer Engineering and
Center for Wireless Systems and Applications
Purdue University, West Lafayette, IN 47907-1285, USA
{koo, cath, hchan, lee85}@ecn.purdue.edu

Abstract

In our previous work, we have designed, using a network-based approach, and are currently deploying on Purdue wireless Infrastructure, a Web service for location discovery of 802.11-based mobile devices. This paper presents a novel web-based application called Remote Printing Service (RPS) which is entirely built on top of our Location Discovery Service (LODS). RPS is not only capable of locating the nearest printers but also allows mobile users to print without having to install any printer drivers. Most of the time users can print even without having to download the file to the mobile device. We have also designed a prototype for a network-based Personal Paging system that provides active paging and active email notification capability for mobile users. These value-added wireless services contribute to the building and promotion of an e-campus community.

Keywords: Location Discovery, Remote Printing, Personal Paging, E-campus, Web Service.

1. Introduction

Location-based services have gained a lot of attention recently and have become the basis for exciting new telecommunication services. These new services distinguish themselves from other network services by needing to know the users' geographical locations in the network. Cellular systems have the ability to locate particular users because they have to be able to connect them to incoming calls, and this makes the provision of location-based services in cellular systems relatively easy. In a client-server paradigm like the one usually used in a Wireless Local Area Network (WLAN), there is no such need for locating users, as hosts usually initiate a connection as a client, and, unless the host also wants to be a server, it will not be passively connected. This limits *a priori* the need for a location discovery mechanism. Such a mechanism, however, would be needed to create value-added applications based on location discovery such as, for example, those mapping the current location of a mobile device to the nearest points of interest such as the nearest printer or vending machines.

In our previous work [13], we have presented a newly designed Web Service called Location Discovery Service (LODS) which allows users with 802.11-enabled wireless devices to find their approximate locations within a campus. Based on the location of a user, applications can suggest the nearest points of interest, e.g., printers, elevators, vending machines, etc. LODS highly enhances the mobility of the hosts within a campus, and, more importantly, it is easy to deploy and does not require large investment in infrastructure. LODS is accessible directly via common Web browsers so virtually all mobile hosts can use it without purchasing additional hardware (e.g., a GPS receiver) or software. Our solution is significantly different from the existing mobile-device-based solutions ([2], [10]) which locate users by using an application installed on the device to collect local information like signal strengths observed by the wireless modem from different Access Points (APs), and then perform triangulation either by sending the collected information to a dedicated server and letting the server process it, or by processing it locally in the mobile device. These solutions face hardware- and software-dependent problems, as wireless modems from different vendors require different methods to obtain such information, and the application for collecting signal strengths must have different versions for different operating systems.

We believe that solutions, which are designed to take advantage of the underlying wireless infrastructure configuration, will utilize the network better, and impose fewer problems on the client end. Our solution is also scalable to campus-wide networks and provides services to the user at virtually no cost at the user end. Furthermore, our solution

*This work has been supported in part by the 21^{st} Century funded Indiana Center for Wireless Communications and Networking.

Proceedings of the First IEEE International Conference on Pervasive Computing and Communications (PerCom'03)
0-7695-1893-1/03 $17.00 © 2003 IEEE

is hardware- and software-independent. We will revisit the design of LODS and our network-based solutions in Section 4.

Purdue University has invested a lot in providing wireless access to her community to promote an e-campus environment. Information Technology at Purdue (ITaP) of Purdue University is currently deploying a campus-wide wireless network, the Purdue Air Link (PAL), to make 802.11b wireless network access broadly available throughout the campus. Upon the completion of the project in May 2003, over eighty buildings on the West Lafayette campus will have wireless connectivity. Currently the system has about 150 APs over twenty-plus buildings. In order to provide more value-added services to the e-campus community, we are currently deploying a Web Service called the Remote Printing Service (RPS), which works with LODS to allow mobile users to print PS, PDF, HTML and most image formats to the printer(s) located closest to the users. This is especially useful in the case where the mobile user is away from his office, e.g., in a different building, and wants to get a hardcopy of some information on the web, minutes of previous meetings, etc. The only requirement to use RPS is to have a generic web browser set to use a dedicated proxy server. There is no additional software or hardware installation required in the mobile device, not even printer drivers. RPS also reduces the amount of data flow between mobile hosts and the network (see later), which highly reduces the power consumption of the mobile device. We will present the details of RPS in Section 5.

Another location-based application we are currently prototyping is a network-based Personal Paging system for WLAN. Currently there are a lot of Instant Messaging (IM) applications available on the market which can provide paging functionality, and most of them are IP-based, i.e., they "locate" users by their IP addresses. However, for most campus-wide wireless networks, Dynamic Host Configuration Protocol (DHCP) is used to dynamically assign IP addresses to mobile hosts. As a result, IP address of a mobile device may not be the same throughout the whole session. In the PAL network setting (see Section 3), for which we design our solution, the campus is divided geographically into different Virtual LANs (VLAN), and each user will obtain a new IP address when it moves from one VLAN coverage to another. Therefore, locating users and finding them simply based on their IP addresses is not feasible, unless some connection re-establishment mechanisms between the IM server and mobile devices are used to keep device-to-IP relation updated, which will inevitably increase the power consumption of the mobile device. We have created a prototype which relies on the knowledge of the underlying network infrastructure as we did for LODS, so that we are allowed to take full advantage of the network and reduce the amount of signaling needed to locate users. However, our Personal Paging system does not directly use our LODS service for reasons to be discussed later. This design, moreover, can also handle the change of IP address problem mentioned above. Details about our Personal Paging system will be discussed in Section 6.

This paper is organized as follows. Related work will be reviewed in Section 2. We will describe the wireless infrastructure of PAL on which we based our design on in Section 3[1]. We will then revisit in Section 4 the design and deployment of LODS, which acts as a building block for some of our location-based services. Remote Printing Service and Personal Paging system will be discussed in details in Section 5 and 6 respectively. We will conclude the paper and provide directions for future development in Section 7.

2. Previous Work

Previous work on location management and mobility ([7], [16], and [19]) mainly considered telecommunication networks settings, which have infrastructure in place to support passive connectivity, paging, and location-based services. They have focused on passive connectivity for which mobile devices, when idle, still have to listen to some control information, either periodically or using certain policy in case an incoming call arrives. Since in telecommunication networks, the architecture and client connectivity are fundamentally different from those in wireless LAN, those schemes do not necessarily work well for a wireless LAN setting. Our design differs from previous work in that we deal with active connectivity, i.e., users actively trigger LODS when they need it. This is why our Personal Paging system cannot use LODS directly. However it reuses most of the concepts developed during the design of LODS.

Hightower and Borriello [11] surveyed some radio frequency (RF) and GPS based location systems, which provide active connectivity. These solutions require additional RF components such as proprietary tag hardware and base stations or a GPS module, together with driver software installed on the mobile and, in some cases, the network infrastructure. ActiveCampus project [10] and Ekahau [2] on the other hand use signal strength experienced by the mobile to determine a user's position through the process of triangulation. A software module developed specifically for each particular operating system and a wireless modem interface must be installed on the mobile before a user can obtain such information. These systems provide solutions that have the advantage of being network independent in that they work on different wireless settings. However, both these systems require the installation of extra modules to the mobile host, which imposes extra costs on the users, and face the problem of hardware and software dependence.

[1] Note that we have presented in [13] a design of LODS over two different wireless infrastructure configurations.

Finally, in the case of Ekahau, administrators need to do on-site calibrations before the system can actually be put into service.

The idea of Remote Printing on an 802.11-based wireless domain uses a different concept. To the best of our knowledge, there is no existing work on this topic. There is an Internet Draft by Ryutov and Neuman [18] on the framework for providing access control to remote application, including printing, but authentication and security are their main concerns. We believe that our solution, the Remote Printing Service, on wireless campus-wide network is the first of its kind to address such a specific need and we will present it in Section 5.

There are many instant messaging (IM) software systems available in the market, and each of them uses a different protocol. ICQ [4] is a popular example of such IM software. Jabber [6] is an attempt to unify all these IM systems so that users only need to install one program to use the services each of these IM applications provides. Instead of having a direct peer-to-peer infrastructure like some IM applications do, it uses an IP-based, client-server model approach. The connection between the Jabber client and the server stays on for the life of the client's session, therefore the client does not have to open any listening ports on the machine, which might introduce security risks. Both ICQ and Jabber are IP-based, so in a DHCP domain where the IP address of a device could change from time to time, they will need extra signaling to keep track of whether the connections are alive. This can be achieved either by continuously probing the server, or by actively re-establishing a connection once the original connection is lost (due to a change in the IP address assigned to the device). In both cases, significant power and bandwidth are wasted on signaling. We will present our network-based solution for Personal Paging in Section 6, where the amount of signaling is kept minimal. Our solution also considers the fact that the IP address of a device in a DHCP domain may not stay the same throughout the session, and we will provide a solution for such scenario.

3. Wireless infrastructure (PAL)

As we have mentioned in the previous sections, a solution that is not aware of the underlying network configuration faces the problem of not being hardware and system independent. In this section, we will present one of the most popular settings for 802.11-based networks, which we called the virtual private networks (VPN) setting. This configuration consists of multiple VLANs under the same administrative domain. It usually comprises thousands of APs, and spans over a large geographical area. Purdue Air Link (PAL) network is an example of such a configuration. PAL is a campus-wide wireless network access service provided by ITaP (Information Technology at Purdue) at Purdue University, and the configuration of PAL is shown in Figure 1.

In PAL network, users are assigned private IP addresses through a DHCP server logically located inside the VPN, and these private IP addresses are valid only in one VLAN area. Once a device leaves a VLAN coverage (say coverage A) to move to another VLAN coverage (say B), it needs to get another private IP address reflecting that it is in VLAN B and the old IP address of the mobile while it was in VLAN A is released to be used by other mobile devices under VLAN A coverage. PAL does not use IP masquerading or other similar techniques to provide Internet access to users. Instead, a VPN box (Figure 1) is put at the edge of the VPN to provide a private-to-real IP mapping. All the traffic that runs across the VPN and is destined to the Purdue Intranet or the Internet will pass through the VPN box, and all the IP packet headers will be modified by the VPN box to provide real-to-private IP mapping (the reverse for the traffic sent to mobile devices). It is also scalable since adding new VLANs will not increase the local traffic on the existing VLANs. Note that users must provide a Purdue account username and password to the system before the VPN box performs any mapping services for them. The system will check this username/password pair with the user authentication server located in the Purdue Intranet, and once the user is authenticated, the VPN box will perform the private-to-real IP mapping and the reverse mapping throughout the session.

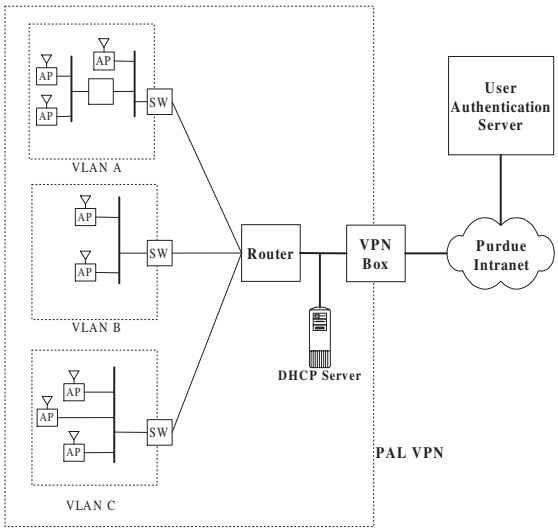


Figure 1. PAL Wireless VPN Configuration

4. Location Discovery Service: a Web Service

4.1. Concept of the service

The Location Discovery Service we designed and implemented is a Web Service consisting of a location discovery engine and an API for accessing the service. Upon either a direct request from the user or indirectly through the call to a location-based service, the location discovery engine will determine which access point (AP) the user's mobile device is currently connected to, and return the ID of the AP as an estimation of the mobile's position. The entity that requested the LODS service can then make use of the returned ID to provide the location-based service through a database mapping the positions of the APs to say, the position of the printers. The number and the positioning of the APs determine the preciseness of the locating process and in our networks, each AP is responsible for a thirty- to fifty-foot radius area, so LODS provides a good estimation of the location of the mobile device considering that our primary objective is to use LODS to suggest neighboring resources like printers, elevators, etc. to the mobile user.

Most APs are designed and configured to be transparent bridges and operate at the link level. In that case, an AP only knows the MAC addresses (and not the IP addresses) of the clients that it is associated with. However, as a Web service, LODS services communicate with clients or other location-based services at the application level using IP addresses and MAC addresses are generally not available. Thus, we are trying to create an IP-to-AP mapping (i.e., the Web Service provides the IP address of the device to locate and LODS returns the ID of the associated AP), and this mapping cannot be obtained directly. In fact, the LODS engine must be able to convert an IP address to the corresponding MAC address, and then find the AP that is associated with this MAC address (see Figure 2). The readers should notice that the IP-to-MAC mapping and the MAC-to-IP mapping could be obtained by different methods.

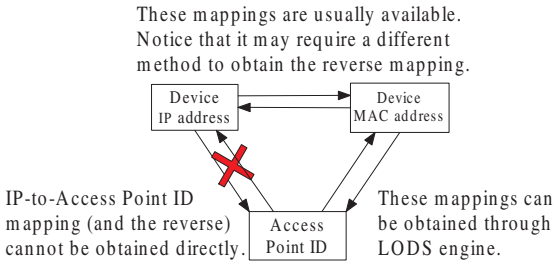


Figure 2. Relationship between IP address, MAC address and Access Point ID in LODS

4.2. Our network-based solution

In our previous work [13], we have compared three different approaches to obtain the ID of the AP with which a mobile user is currently associated. We concluded that the RADIUS approach, is the most suitable one for our campus-based network and it is the one we are currently deploying.

In a network where a RADIUS server [17] is deployed, each time a device tries to associate with a new AP, an authentication process starts in which the MAC address of the device is sent through Extensible Authentication Protocol (EAP) [8] to the AP that sends it to the RADIUS server for authentication. RADIUS server will reply with an ACCEPT or REJECT message to the AP based on the information it supplied. Note that the device has to go through the authentication process again when it wants to associate with a new AP. If the authentication process is successful, the following information is kept in a log file in the RADIUS server: the time at which the authentication request has been made, the ID of the AP that has made the request, and the MAC address of the device that was authenticated. By inspecting the log file of the authentication (RADIUS) server, it is possible to determine to which AP a given device is currently associated. Using this information, we can determine the approximate location of the device. Figure 3 shows the use of the RADIUS server as a mean to locate a mobile device. Currently most APs have built-in support for RADIUS.

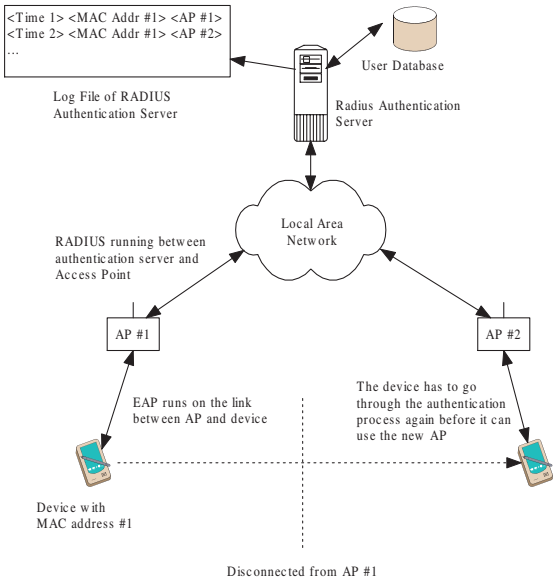


Figure 3. The RADIUS approach

With a centralized RADIUS server, LODS can check the log file of the server and easily determine the latest association between the mobile and APs. However, ITaP, the

unit that administers PAL, decided not to have a centralized RADIUS and EAP service to check for authenticated MAC addresses. Instead, mobiles with any MAC addresses are allowed to talk to all the APs without blocking, but the user must have a Purdue account and needs to supply login information (as mentioned in Section 3) before getting a real-to-private (and a private-to-real) IP mapping from the VPN box.

For the purpose of enhancing scalability, we dedicated a "reduced" RADIUS server to one or more VLANs, depending on the expected LODS load from the VLANs. A reduced RADIUS server, which we re-named Location Collector, consists of a RADIUS server program, which can be run on a low-end Linux box. It does not need to perform authentication as it is supposed to. Its only function is to log the access requests to APs made by the mobile users in its coverage and maintain the mappings of MAC addresses to the most current associated APs. This is equivalent to voiding the RADIUS authentication processes by always returning ACCEPT to the AP. It is important that the reliability of the Location Collectors should not affect that of the PAL network, so EAP should be configured to never block a modem even if it fails to get a RADIUS ACCEPT message. With these Location Collectors, RADIUS traffic is limited to each VLAN or group of VLANs and each Location Collector will be responsible for the MAC-to-AP mapping in its region. Now we have the components required to build a scalable LODS system.

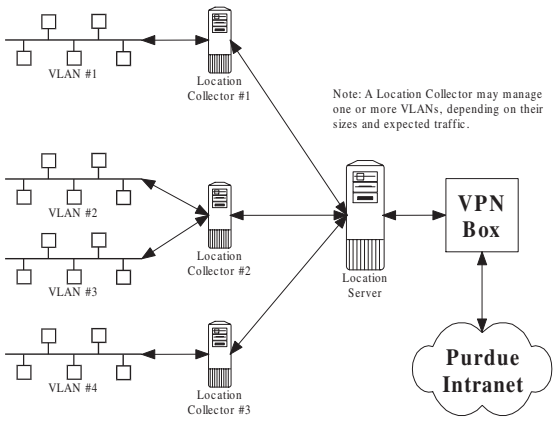


Figure 4. Schematic of LODS for Scalable VLAN Configuration

This scalable LODS works in the following way (Figure 4): When a user or a Web application requests service from LODS, the IP address of the mobile initiating the LODS request will be extracted from the message header of the query. The Location Server will, based on the private IP address of the mobile, determine which VLAN the mobile device is in, thus which Location Collector to contact (there is a straightforward mapping between a mobile device private IP address and the VLAN it is in). As each Location Collector contains only the MAC-to-AP mappings, LODS needs to first figure out the MAC address of the device. This can be done in two ways: First, the default gateway of each VLAN will have the IP-to-MAC mapping of the mobile since the APs are configured to be transparent bridges, so LODS can make an SNMP query to the correct gateway to get the MAC address corresponding to an IP address. This is the most straightforward way to obtain IP-to-MAC mapping in the sense that there is no overhead except an SNMP query and an SNMP reply messages, which are generated on-demand.

Alternatively, it is also possible to make use of the log of the DHCP server. The log file of the DHCP server contains the assignment of IP addresses to MAC addresses, so by studying the log file, the IP-to-MAC mapping can be obtained. For more efficient response to queries, it is recommended that a database should be built which also parses the log file in the background. The database can be implemented in the same machine as the Location Server, so that the query time and traffic will be minimal, and the only signaling traffic introduced will be the one generated when the log file is being updated. One advantage of this scheme over querying the gateway is that with this database, not only the IP-to-MAC mappings are available, but also the MAC-to-IP mappings, which cannot be easily obtained by querying the gateways since we do not know which gateway(s) to query. The advantage of having the MAC-to-IP mappings is that the network is now able to provide 802.11-based paging services and user tracking services, which we will describe in the next section.

Once the Location Collector obtains the MAC address of the mobile from either the gateway or the DHCP log, it will return the ID of the AP to which the mobile device is currently associated back to the requesting entity.

This design solves the scalability problem in that it takes full advantage of the modular architecture of the VLAN configuration. The downside of this design is that it requires additional hardware (the Linux boxes) for the Location Collectors (i.e., the RADIUS servers). One such server will need to be installed in every group of VLANs. The Cistron RADIUS server [1] is using a free software which has all the features needed. Since each Location Collector will only be responsible for the RADIUS traffic in its own region, the wireless network could expand by introducing more VLANs, and more Location Collectors thereafter, without posing scalability problem to LODS. With network-aware LODS deployed in the PAL network, it is possible for mobile users in the Purdue community to enjoy location service by using any mobile device with a generic browser. It is also possible to build location-based applications on top of

this new Web Service. We will introduce in the next section a first innovative application of LODS, namely the Remote Printing service that is currently being deployed on the PAL network.

5. Remote Printing: A Case study of a successful design and deployment

5.1. Description of the service

Remote Printing Service (RPS) is a project supported in part by Hewlett Packard. It is a Web-based printing service for PDAs and laptops that enable them to print virtually any document that can be accessed through a Web browser (i.e., HTML, PS, PDF, and virtually all types of images) using any printer connected to the network without downloading the document. The advantages of this service are that there is no need to install any printer driver in the mobile device; that files for which there is no viewer installed in the PDA can still be printed from a PDA; and the mobile devices do not need to download, say, a huge postscript file, before printing it. This last advantage reduces the consumption of power and memory in the mobile device and is bandwidth friendly for the wireless LAN. In our network-based solution each mobile user wanting to use the Remote Printing Service is required to configure his browser to use a designated proxy server in order to receive this service. This proxy server will look at every URL the user requests, and if it has a PS, PDF, or any image suffix the system supports, it will redirect the request to the Remote Printing page together with the original URL. In that page the mobile user can choose whether to save/view the file or directly print it. If the user chooses to print the file, he has to select which printer to send it to, and then the Remote Printing Service will download the file directly from the content providing site, convert the file to a printable format, and print it to the desired printer (See Figure 5). If the requested page is an HTML document, the proxy will add a small piece of code which generates a small box with a printing icon on the upper left hand corner of the webpage (Figure 6). Users can have the box removed by clicking the ”close” button on the box. If the user decides to print that page, he would just need to click the printing icon, and the proxy will redirect the request to the Remote Printing page, again with the URL of the page that the user is browsing. The user can then choose which printer to use and the Remote Printing Service will download, convert and print the HTML file in a similar fashion as printing PS and PDF files.

Currently RPS supports TXT, PDF, PS, HTML, and virtually all formats of pictures. It uses ImageMagick [5] to do the image conversion, HTMLDOC [3] to do HTML conversion, and Adobe Acrobat Reader to do PDF conversion. The whole process from downloading to printing does not involve any signaling or data transfer between the mobile device and the Remote Printing server.

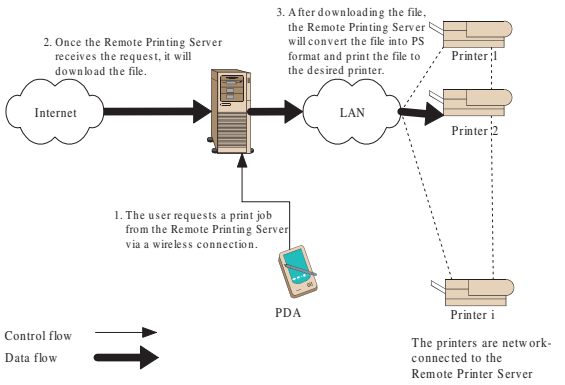


Figure 5. Remote Printing Service

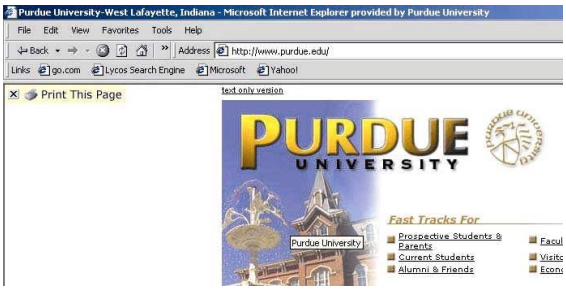


Figure 6. Screenshot of Remote Printing Service

The Remote Printing Service works well if the user knows to which printer to print to. However, if the mobile user wants to use the printer, which is the closest to him right now, assuming that he is away from his own building, he may not know where the closest printer is. This means that without a location-based printing service, the user may have to get his document printed to an unnecessarily distant printer, even when a closer one is available. With LODS, the remote printing service can know approximately where the user is, and suggest printers that are closer to the user. This is illustrated in Figure 7.

When Remote Printing is called by the user (directly or indirectly through a redirecting proxy), the service will know the IP address of the device. The Remote Printing Service will then pass the IP address of the device to the network-based LODS service discussed in the previous section to determine which AP the device is currently using, and the AP's ID will be used to query a Location Database which contains the AP-to-printer mappings and is located in the same machine as the Remote Printing service. A list of closest printers will be returned to the user and the user can

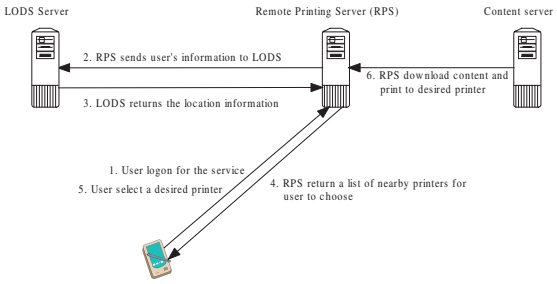


Figure 7. Remote Printing Service using LODS

choose a printer from the list. After the user selects the desired printer, RPS will directly download the content from the content provider and print to the selected printer.

6. Personal Paging System

6.1. Description of the service

Personal Paging System is a newly proposed system that we have just prototyped and are planning to deploy onto PAL in the near future. This system consists of a cross-platform client application written in Python, and a paging server. Users are required to install the client application on their device and logon to the paging server before can be paged. Currently in the prototyping and testing phase, the functionalities provided by the client are limited. The mobile device can actively receive pages, page and reply to someone, and actively get notified if any emails for the user have arrived. More features including "ignore list" of users, chatting, and locating the people who paged you (in collaboration with LODS) will be included in the next version.

Our Personal Paging system differs from existing products in the sense that it is network-based. Unlike other IP-based instant messaging system, our paging system fully utilized the underlying network architecture, which makes it more adaptable to the limitations existed on WLAN.

6.2. Limitations on WLAN

As mentioned earlier, tracking a mobile user is not a compulsory requirement for WLAN. Unlike cellular networks, there is no unique, network-wide accessible ID like the telephone number in WLAN that can be used to find the user. Each modem has a unique MAC address, but this form of ID is not accessible to hosts that are physically not on the same network as the device. IP address could be a solution, but in most DHCP based wireless network, the IP address of the device could change during the session as it is the case in PAL when a user moves from one VLAN to another. For client applications running on the mobile device which maintain a connection to the server during the life of the application, the connection could be disconnected at any time due to a change in IP addresses, and a re-establishment is required. This eliminates IP-based products from being effective and power-saving solutions.

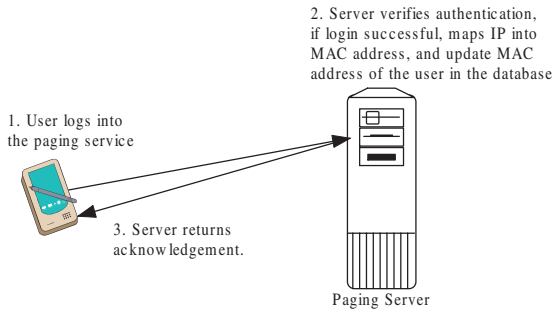
6.3. Proposed Design

Our application uses a network-based approach. We have decided to use the MAC address of the device as the ID instead of the IP address, which means we have to develop tools that are able to do translations between MAC addresses and up-to-date IP addresses, and vice versa. From our design experience of LODS, this can be achieved by making a query to the DHCP log file. The system works as follows. When the client first establishes a connection with the server, the server translates it's the client IP address to a MAC address and stores this address to its database. Now, when a page request for a logon user arrives, the system needs to know the latest IP address of that user in case it has changed. It will make a query to the DHCP log file using the MAC address of the user as the index and find the up-to-date IP address of the client. The server can then establish a connection with the client, and send the page.

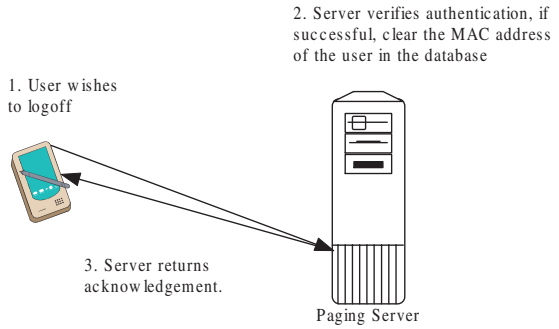
Login and Logoff processes (Figure 8)

In order to use the Personal Paging System, a user must first login to the paging server. On the paging server, a database is used to provide authentication to users and store important connection information about users that are currently logged on. The database should have at least three fields, the first two being the login name and password of the user, the last field being the MAC address of the user when he logs on. We choose to store MAC address instead of IP address because in our WLAN configuration, when moving from one VLAN to another, the IP address will change. When the user wants to log on, the client application sends the username and password to the paging server, and the server will check this authentication against the database and if it matches, the system will translate the IP address of the connection into the MAC address and store it in the database. The process to convert MAC address to IP address (and the reverse) is done by making queries to the DHCP log file. The server will also send a positive acknowledgement to the client if login is successful or a negative acknowledgement otherwise. This connection is not maintained throughout the life of the client session. Instead, it is closed down immediately after the server sends the acknowledgement. Our client programs do not send any periodic probes to maintain a constant connection with the server, so it is not possible for the server to determine the client's up-to-date IP address based on his/her previous IP address. However, because the

MAC address is unique, it will not be changed unless the user changes a wireless modem, so storing the user's MAC address can give us a unique identification. When the client is paged, the system can rely on the stored MAC address to determine the up-to-date IP address of the client via SNMP query.



(a) Logon



(b) Logoff

Figure 8. Login and Logoff processes

When the user logs off, it connects to the server and supplies its username and password. If the authentication is successful, the server sets the MAC address field of the user's record in the database to empty and returns a positive acknowledgement. A negative acknowledgement is returned otherwise.

Receiving page from general users (Figure 9)

Here we define "general users" to be those who choose not to use the client program we provide to page our users. In that case, the paging will be done by sending an email to a specific email address. For example, if one of our registered users has the username johndoe, he will have an email address johndoe@paging.ecn.purdue.edu to receive pages. We assume that the paging message is contained in the subject of the email, and any content inside the body or any attachments will be discarded. When a "paging email" arrives to our system, the Mail Transport Agent (MTA) will pick it up. We have configured the MTA to deliver the email to our own delivery agent, which will extract only the From and Subject information, construct a message containing this information, and page the recipient. Note that besides taking out the pertinent information from the email, the delivery agent can also do other actions on the email body. For example, checking if the email is from a spam source, or translating the message into another language.

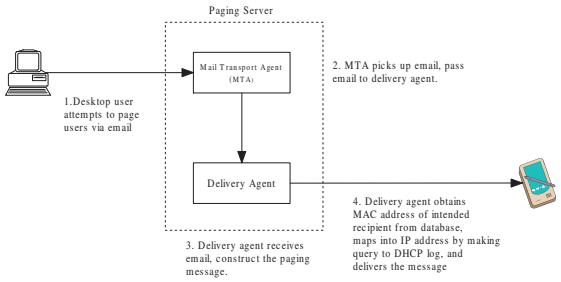


Figure 9. Receiving page from general users

After doing the processing on the email, the system queries its database for the MAC address of the intended recipient. If the recipient is logged on, we then obtain his up-to-date IP address by querying the DHCP log. On the client program, there is a process listening on a specific port known to the server that only accepts connections originating from our server and rejects any other connections. The server will contact the recipient at this specific port. If the server failed to obtain the IP address of the user because the user is offline, or the client fails to respond because of application shutdown, the message is discarded and the sender of the page is informed. If the connection is established successfully, the server will start sending the message it constructed from the email it received, and then shuts down the connection after the transmission. After the connection is closed, the client program will display a notification (e.g. a popup box) on the screen to notify the user.

Client-to-client paging (Figure 10)

In most Instant Messaging applications, client applications usually contact another client directly to deliver messages, but because our client application could be running on a laptop or a PDA, we wish to reduce the complexity of the client application. Therefore, client to client communication is done using the server as an intermediate agent. An inter-client delivery agent will be listening on a specific port running on the server to wait for paging requests from users. When a user wishes to deliver a page to another user, the client application from the sender side would contact the inter-client delivery agent, and upon establishment of connection, the application will authenticate itself and make a request to send the message to the intended user. The inter-client delivery agent will pass the message to the delivery agent mentioned above to deliver the message to the intended recipient.

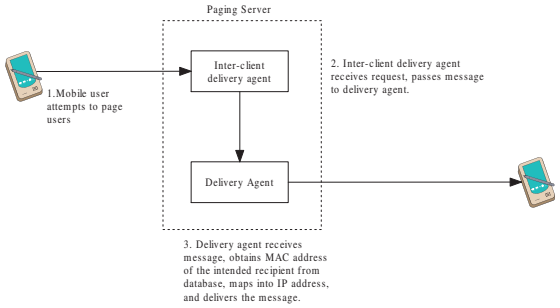


Figure 10. Client-to-client paging

7. Conclusion and Future Works

In this paper we have reviewed the Web Service we deployed called the Location Discovery Service (LODS), which serves as a building block for some of our location-based applications. Remote Printing Service is currently being deployed in the Purdue Air Link network. This service reduces the data flow between mobile hosts and the network, eliminates the need for installing printer drivers in the mobile device, and suggests the printer nearest to the user with the use of LODS. Personal Paging system is a network-aware, non-IP-based paging system specifically designed for WLAN. It takes full advantage of the underlying network architecture and overcomes the limitations of WLANs.

Providing these value-added services will promote an exciting and interactive e-campus environment. Future work includes providing more features for the Personal Paging system, like "ignore list" of users, chatting, and locating the people who paged you via LODS. Security of the client application, which has listening ports, is another issue on which we are currently working.

Acknowledgement

The authors would like to thank Mr. Bill Simmons of Engineering Computer Network (ECN) at Purdue University for his help in deploying LODS in ECN and Mr. Scott Ballew, Mr. Steve Mayo and Mr. Jim Bottum of ITaP at Purdue for their help with the PAL network. The authors would also like to thank Hewlett-Packard for their support through the mobile laboratory grant.

References

[1] Cistron RADIUS Server. $< http://www.radius.cistron.nl/ >$.
[2] Ekahau. $< http://www.ekahau.com >$.
[3] HTMLDOC. $< http://www.easysw.com/htmldoc >$.
[4] ICQ. $< http://www.icq.com >$.
[5] ImageMagick. $< http://www.imagemagick.org/ >$.
[6] Jabber. $< http://www.jabber.org >$.
[7] A. Bar-Noy and I. Kessler. Tracking Users in Wireless Communications Networks. In *Proceedings of IEEE INFOCOM*, pages 1232–1239, San Francisco, CA, 1993.
[8] L. Blunk and J. Vollbrecht. PPP Extensible Authentication Protocol (EAP) . RFC 2284, Mar. 1998.
[9] R. Droms. Dynamic Host Configuration Protocol. RFC 2131, Mar. 1997.
[10] W. G. Griswold, R. Boyer, S. W. Brown, T. M. Truong, E. Bhasker, G. R. Jay, and R. B. Shapiro. ActiveCampus - Sustaining Education Communities through Mobile Technology. Technical Report CS2002-0714, University of California, San Diego, 2002.
[11] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *IEEE Computer*, (8):57–66, 2001.
[12] E. Isaacs, A. Walendowski, and D. Ranganathan. Hubbub: A sound-enhanced mobile instant messenger that supports awareness and opportunistic interactions. In *Proc. Conference on Human Factors in Computing Systems (CHI 02)*, Minneapolis, Minnesota, Apr. 2002.
[13] 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 appear.
[14] S. Patel, G. Henderson, and N. D.Georganas. Multimedia Fax-MIME Interworking. *IEEE Multimedia*, (4):64–70, 1994.
[15] B. Raman, R. Katz, and A. Joseph. Universal Inbox: Providing Extensible Personal Mobility and Service Mobility in an Integrated Communication Network. In *Proc. of the Workshop on Mobile Computing Systems and Applications (WMSCA'00)*, Monterey, Dec. 2000.
[16] R. Ramjee, T. F. L. Porta, S. Thuel, K. Varadhan, and S. Y. Wang. HAWAII: A Domain-Based Approach for Supporting Mobility in Wide-area Wireless Networks. In *Proceedings of IEEE Intl. Conference Network Protocols*, Toronto, Canada, 1999.
[17] C. Rigney, A. Rubens, W. Simpson, and S. Willens. Remote Authentication Dial In User Service. RFC 2138, Apr. 1997.
[18] T. Ryutov and C. Neuman. Access Control Framework for Distributed Applications. Internet Draft, draft-ietf-cat-acc-cntrl-frmw-05.txt, Nov. 2000.
[19] R. Shankaran. A distributed location management scheme for mobile hosts. In *Proceedings of ICPADS*, KyongJu City, Korea, June 2001.
[20] Y. W. Thomas, T. F. L. Porta, and K. K. Sanbani. Pigeon: A Wireless Two-Way Messaging System. In *Proceedings of IEEE Symposium on Personal, Indoor and Mobile Radio Communications*, pages 693–697, Taipei, Taiwan, ROC, October 15–18 1996.