About location-aware mobile messages.

Expert system based on WiFi spots

Dmitry Namiot
Lomonosov Moscow State University
Faculty of Computational Mathematics and Cybernetics
Moscow, Russia
dnamiot@gmail.com

Manfred Schneps-Schneppe
Ventspils University College
Ventspils International Radio Astronomy Centre
Inzenieru st 101, Ventspils, LV-3601 Latvia
manfreds.sneps@gmail.com

Abstract — **this paper describes a new model for proactive messages delivery to mobile phones. SpotEx application can use any Wi-Fi access point as presence sensor that could activate delivery for some user-generated messages right to mobile phones. The key idea is how to associate some user-defined messages and Wi-Fi access points. As a result we can build rule-based expert system that describes delivery of user-defined messages depending on visibility of Wi-Fi hotspots.**

Keywords – Wi-Fi, indoor positioning, checkin, productions, rules

Introduction

Let us start from the history and describe some applications that precede our SpotEx approach. At the first hand it is Active Badge. The Active Badge system provides a means of locating individuals within a building by determining the location of their Active Badge. This small device worn by personnel transmits a unique infra-red signal every 10 seconds. Each office within a building is equipped with one or more networked sensors which detect these transmissions. The location of the badge (and hence its wearer) can thus be determined on the basis of information provided by these sensors. [1]

The last statement is actually the key point here – a hardware trigger that defines some information context. Obviously, it is location-aware information.

The next issue is Bluetooth based proximity marketing. Bluetooth broadcast, simply or Bluetooth-based systems.

Bluetooth, a short-range wireless system supported by many mobile devices, is one transmission medium used for proximity marketing. The process of Bluetooth based proximity marketing involves setting up Bluetooth "broadcasting" equipment at a particular location and then sending information which can be text, images, audio or video to Bluetooth enabled devices within range of the broadcast server. Other standard data exchange formats such as Vcard can also be used. [2]

The main element here is the broadcasting device that actually discovers the mobile terminal. It used to be the case that due to security fears, or a desire to save battery life, many users keep their Bluetooth devices in OFF mode. Yes, some implementations of Bluetooth proximity marketing require users to run mobile applications on their phones to enable them receive content. This has the advantage that only those who choose to will receive content. But still for the many devices Bluetooth is either off or not in the “discoverable” mode. Also even some modern phones may not support the OBEX Push Bluetooth profile.

There are three main conclusions we can did at this point:

- nowadays we can use smart phones as Active Badges

- our system should be based on Wi-Fi just because it is more probably to see it “always ON” for modern phones

- the “badge” (phone) should discover information nodes (and not vice versa) as it is more safety for the users. And only those who choose to will receive content.

So based on that the next class of systems we can reuse some ideas from is indoor positioning.

As per Wikipedia, an indoor positioning system (IPS) is a network of devices used to wirelessly locate objects or people inside a building. Due to the signal attenuation caused by construction materials, the Global Positioning System (GPS) loses significant accuracy indoors. Instead of satellites, an IPS relies on nearby anchors (nodes with a known position), which either actively locate tags or provide environmental context for devices to sense. The localized nature of an IPS has resulted in design fragmentation, with systems making use of various optical, radio, or even acoustic technologies [3].

Nowadays, a great number of technologies are being used for indoor localization, such as Wi-Fi, RFID etc. However, all of them require the utilization of their own API with their own protocols. This can be a big challenge for developing heterogeneous scenarios where different localization systems have to be used for a location service.

For indoor-outdoor tracking, there is no such thing as a “one size fits all” technology; it takes a combination of technologies to tackle the typical use case. Solutions exist that integrate two complementary tracking technologies that leverage robust middleware and application software and provide a more complete hybrid solution that offers higher overall location system coverage and availability.

An Active RFID location system includes proprietary RFID scanners installed throughout a facility that interrogate either active (radio transceivers) or passive tags that attach to objects.

Active tags use batteries and allow up to a twenty foot range between the scanner and the tags. Passive tags don't use batteries, and they receive energy when being scanned. The radio waves emitted by an RFID scanner energize a passive tag long enough for the tag to transmit its code to the scanner. Passive tags, however, must be relatively close to the scanner. As a result, radio transceivers are the most common type of RFID tag (Active RFID Tag) found in positioning systems.

Active RFID tags contain electronic codes that identify one tag from another. A centralized station stores the tag codes that the scanners collect. Because the scanners are placed in known positions throughout a facility, the centralized station is able to identify and display the location of each tag (and of course the client device that the tag corresponds with).

Ultra-Wideband (UWB) positioning systems have proprietary scanners installed throughout the facility that continuously monitor UWB radio transceivers attached to clients. UWB systems, however, operate using radio signals having very wide bandwidth, and position calculations are made based on time-of-arrival techniques instead of signal strength. This leads to fairly good location accuracy. By reading the time of arrival of a beacon signal from a specific UWB radio transceiver from three or more scanners, for instance, the position of the tag and applicable object can be found.

A standard Wi-Fi based positioning system, such as the one offered by Ekahau [4], is completely software-based and utilizes existing Wi-Fi access points installed in a facility and radio cards already present in the user devices. Companies could deploy also Wi-Fi based radio tags that use industry standard components that adhere to the 802.11 standards. This approach allows for the use of commercial off-the-shelf hardware and drivers to produce a standards-based radio tag that can communicate bi-directionally over the 802.11 network.

Thus, a standard Wi-Fi based positioning system can realize any type of location-aware application that involves PDAs, laptops, bar code scanners, voice-over-IP phones and other 802.11 enabled devices. For embedded solutions, there is no need for the client to include a specialized tag, transmitter, or receiver.

Because of the entire use of standards-based hardware, such as 802.11b, 802.11g, and 802.11a, a standard Wi-Fi based solution rides the installed based and economies of scale of the networks and end user devices that are proliferating today. Without the need for additional hardware, a company can install the system much faster and significantly reduce initial and long-term support costs. A common infrastructure supports both the data network and the positioning system, something companies strive for. The positioning system works wherever there is Wi-Fi coverage.

In addition to cost savings in hardware, a standards Wi-Fi based positioning system significantly reduces the potential for RF interference. The total Wi-Fi positioning system shares the same network along with other network clients, so there is no additional installation of a separate wireless network (as RFID requires) that may cause RF interference with the existing wireless network. [4]

Wi-Fi location positioning is based on a grid of Wi-Fi hotspots providing, in general, 20-30m location accuracy. For more accuracy, there needs to be more access points with more Wi-Fi signals until a point of diminishing returns, i.e., you don't need 100% of access points to get the same accuracy with 75% of access points. In addition, better location accuracy can be achieved by knowing the actual (latitude, longitude) of the Access Point.

There are many articles devoted to Wi-Fi positioning. For example, a layered positioning system based on a model combining a reference point-based approach with a trilateration-based one. Several layers of refinement are offered based on the knowledge of the topology and devices deployed. The more data are known, the better adapted to its area the positioning system can be [5]

The model

As it follows from the review above all indoor positioning services are based actually on the map of tags (devices) with known locations. And what is very important, that this map should be created beforehand. Later, using trilateration or other similar technologies, we can calculate the approximate location for our own device.

Here is the starting point for our new approach. Two main ideas:

- we are going to support "ad_hoc" networks. In other words we need something that let us avoid the preliminary steps by positioning our "metering" tags (well known devices etc.)

- for many LBS applications (if not for almost all of them) the most important element is actually content related to the position, rather than position itself. In other words we are detecting positions in LBS services for getting (setting) some data related to this position rather than for getting simple (raw) location data.

Based on that let us present our SpotEx (Spot Expert) service [6]. What if we stop our traditional indoor positioning on the first stage: detection of Wi-Fi networks? This detection actually already provides some information about the location. And as the second step we will add the ability to describe some rules (if-then operators, or productions) related to the Wi-Fi access points. Our rules will simply use the fact that the particularly Wi-Fi network is detected. And based on this conclusion we will deliver some user-defined messages to mobile terminals. Note again, that we are talking only about the network detections. We are not going to connect mobile users to the detected networks and our suggestion does not touch security issues. We need only SSID for networks and any other public information.

So our service contains the following components:

- database (store) with productions (rules) associated with Wi-Fi networks

- rule editor. Web application (including mobile web) that lets users add (edit) rule-set, associated with some Wi-Fi network

- mobile applications, that can detect Wi-Fi networks, check the current conditions against the database and execute productions (e.g. deliver messages to mobile terminals in proactive mode)

So once again how does it work? We can take any exiting Wi-Fi network (or especially created for this service – the most interesting case, see below) and add some rules (messages) to that network. Message here is just some text that should be delivered to the end-user's mobile terminal as soon as the above-mentioned network is getting detected via our mobile application.

The possible use cases, including commercial deployment are obvious. Some shop can deliver deals/discount/coupons right to mobile terminals as soon as the user is near some predefined point of sale, campus admin can deliver news and announces, hyper local news could be tight to the public available networks and delivered via that channel too.

There are several services on the market that offer Wi-Fi proximity marketing. But they are trying to deliver commercial information to the users connected to Wi-Fi spots [7]. And our approach does not touch connectivity at all. We will use only a fact that some network is visible at this location.

More about SpotEx

Why do we think the rules based system around Wi-Fi points could be interesting? It is just because the access points could be opened right on the mobile phones. In other words, Wi-Fi networks (access points actually), we going associate our messages with could be created (opened, configured) on the mobile phones. Here for example is the screen shot for iPhone:

Figure 1. Wi-Fi access point iPhone

The similar service exists for Android and Symbian:

Figure 2. Wi-Fi access point Android

This feature actually opens the door for the new set of the services – dynamic LBS. Our messages (location aware data) are depending on the visibility of some Wi-Fi access point and this Wi-Fi access point could be not only switched on/off but moved also. Not only tag's holder is moving (active badge, indoor positioning) but the sensor itself is moving too.

For creating some rule set (read – for delivering messages in some particular area) we need just an ordinary phone. We (as content provider) can open Wi-Fi access point on that phone, describe rules (messages) to the opened network in the central repository and deliver own messages for all users nearby our phone.

And if this phone is staying during a day at the same location – we have triggers for this location. For example, trigger that works for some department in the big mall during the business hours. But such a "sensor" could be moveable of course: the phone (sensor) is moved with the owner. Wi-Fi access point could be simply switched off for example. This feature opens a lot of possibilities for the development: show messages in the local proximity, implement hyper-local news systems, develop some games (e.g. find all messages in the area) etc.

So this approach is leading actually to the wide set of various developments. And for starting we need actually just a smart phone with Wi-Fi access point (opened, configured) as well as our external database with rules.

How our productions data store looks like?

Each rule looks like a production (if-then operator). The conditional part includes the following objects: Wi-Fi network

SSID, signal strength (optionally), time of the day (optionally), client ID (see below). In other words it is a set of operators like:

IF network_SSID IS 'mycafe' AND time is 1pm – 2pm *THE*N present the coupon for lunch

Because our rules form the standard production rule based system, we can use old and well know algorithm like Rete [8] for the processing. A Rete-based expert system builds a network of nodes, where each node (except the root) corresponds to a pattern occurring in the left-hand-side (the condition part) of a rule. The path from the root node to a leaf node defines a complete rule left-hand-side. Each node has a memory of facts, which satisfy that pattern. This structure presents essentially a generalized tree. As new facts are asserted or modified, they propagate along the network, causing nodes to be annotated when that fact matches that pattern. When a fact or combination of facts causes all of the patterns for a given rule to be satisfied, a leaf node is reached and the corresponding rule is triggered [9].

So, this service finally includes the following components:

- database with rules (productions), defined for Wi-Fi networks

- client application for smart phones (currently – Android, in future version iPhone too) that lets execute rules against the current context

Database (rule set storage) has got web UI (including mobile web – it is actually an HTML5 application). The mobile web access would be fully enough for updating that database.

Additionally, this data set provides an open API (REST based requests) that lets third-party developers fill (update) database programmatically. We are keeping that also as a possible link for the enterprise usage for example. API helps automate rules uploading from ERP systems and so on.

Client side application actually uses the above-mentioned open API from database.

Also we would like especially highlighting the fact that as per suggested approach it is not mandatory to have just one database for all imaginable rules (all access points). With this approach we can easily see some vertical solutions – customized client side applications that work with particular database (with particular set of rules). With the existing API any set of rules is just an URL (end-point) for passing REST requests to.

Let us list shortly the possible use cases. The most obvious and probably most interesting commercially: deliver deals/discount/coupons right to mobile terminals as soon as the user is near your point of sale. And "point of sale" here is again the visibility (the defined rules) for some Wi-Fi access point (or points). And as we state above this access point could be an ordinary phone. We can describe such kind of deployment as "automatic check-in" for example.

With this service we can easily go to the hyper local news market too and deliver news/announces in campus and/or office complex. As per Wikipedia, Hyper-local content, often referred to as hyper-local news, is characterized by three major elements. First, it refers to entities and events that are located within a well-defined, community scale area. Secondly, it is intended primarily for consumption by residents of that area. Thirdly, it is created by a resident of the location (but this last point is discussed because for example a photo can be hyper-local but not locally produced) [10].

With SpotEx system, having just a mobile phone we can create local content (including multimedia files, created/recorded with this phone – photos, video etc.), describe this content in the rules set and after that show it (deliver to) people nearby this phone.

Technically this rule set (messages) could be described for any existing Wi-Fi network. But as a primary usage (at least for now) we see Wi-Fi access points opened right on the mobile phones. Actually a lot of interesting services could be based on the fact that access point (sensor in our approach) is an ordinary mobile phone. We can see here some like classifieds systems on demand. Notes from our classified system are linked to the particular Wi-Fi network, so it means that notes are actually linked to some particular phone. And it means that notes will "travel" with that phone etc.

Also some game applications could be developed with this schema. For example, some possible game scenarios: collect more messages from different people (game participants), find a particularly message on the streets (actually – among the data providers) etc.

Also the ability to fill rule set storage via API will help us to create dynamic applications.

Of course, our database (rule set) requires authenticated access. But for the clients, at least by our current vision, this approach does not require any authorization. So we see clients (consumers) to be completely anonymous in this system. The only element that we are going to add on the first stage is unique ID for the each client. It is very easy to implement, this ID could be assigned during the very first request to the system. It is really just an ID, there is no need to request and save any user-defined data. ID let us distinguish clients and deploy more sophisticated rules. For example, we can count how many times the particular client opened messages from the particular data provider and use that information in our rules. E.g. we can add rules similar to this:

IF offer from networkID 'mycafe' is opened 3-rd time during the week *THEN* offer frequent visitors pack.

The next set of enhancements could be linked with "multi-networks" rules. Actually, the same principles that are deployed now in positioning (trilateration) could be used here. Some of the productions could rely on more than one network in the conditional part.

The current implementation for mobile client based on Android OS. This application uses *WiFiManager* from Android SDK - the primary API for managing all aspects of Wi-Fi

connectivity. This API let us pickup the following information about nearby networks:

SSID - the network name

BSSID - the address of the access point

capabilities - describes the authentication, key management, and encryption schemes supported by the access point

frequency - the frequency in MHz of the channel over which the client is communicating with the access point

level - the detected signal level in dBm

Actually all these parameters could be used in our rule set. It is just a first version of SpotEx service [6] that deals with SSID only.

The future development might go deep into "idea of phone as a sensor" and add Bluetooth detection too. We've started with Wi-Fi only database mostly due to lack of security issues and Wi-Fi's popularity among smart-phone users. It is a general direction for this class of services: we look at what kind of things you can do using your phone as a proximity sensor. And the second key point for our offering – we are talking actually about proximity to other phones.

There are several applications (e.g. on Android Market), like *Locale* for example that can arrange some actions depending on your current position (location area): switch on/off Wi-Fi, switch off sound etc. In other words there are several applications that can automatically change an Android phone's settings based on its location. We can say that SpotEx solves actually the reverse task: what others can offer for you as soon as you are near some point.

Also we can note again, that instead of detecting user's location this service actually presents to the users context, related to the location. And it is actually the main task LBS services are getting developed for at the end of the day.

Conclusion

This paper describes a new location based service developed on the ideas of pseudo-indoor positioning with Wi-Fi networks. Service can use existing as well as the especially created (described) Wi-Fi networks as triggers for proactive delivering user-defined content to mobile phones. Service lets users link (associate) own messages with Wi-Fi hot spot, as well as read mobile messages from other participants. This service could be used for delivering commercial information (deals, discounts, coupons etc.), hyper-local news data, personal news etc. It could be used for creating context aware applications too. At this moment we are now aware about any other mobile service with the similar features.

References

[1] AT&T Laboratories Cambridge The Active Badge System http://www.cl.cam.ac.uk/research/dtg/attarchive/ab.html Washington

[2] Proximity Marketing: http://en.wikipedia.org/wiki/Proximity_marketing http://www.facebook.com/places/

[3] Wikipedia: Indoor positioning system http://en.wikipedia.org/wiki/Indoor_positioning_system

[4] Comparison of Wireless Indoor Positioning Technologies www.productivet.com/docs-2/Wireless_Comparison.pdf

[5] F. Lassabe, P. Canalda, P. Chatonnay and F. Spies "Indoor Wi-Fi positioning: techniques and systems" Annals of Telecommunications Volume 64, Numbers 9-10, 651-664, DOI: 10.1007/s12243-009-0122-1

[6] SpotEx service: http://spotex.linkstore.ru

[7] Mobile Markerintg: http://www.media-2go.net/Solutions

[8] Charles L. Forgy, "RETE: A fast algorithm for the many pattern/many object pattern match problem", Artificial Intelligence 19(1):17-37, September 1982.

[9] About the Rete Algorithm: http://en.wikipedia.org/wiki/Rete_algorithm

[10] Hyperlocal content: http://en.wikipedia.org/wiki/Hyperlocal