FLAVOUR - **F**riendly **L**ocation-aware Conference **A**id with Pri**V**acy **O**bservant Architect**UR**e

Kavitha Muthukrishnan, Nirvana Meratnia and Maria Lijding

Faculty of Computer Science
University of Twente
P.O.Box 217, 7500 AE, Enschede, The Netherlands
{k.muthukrishnan,n.meratnia,m.e.m.lijding}@ewi.utwente.nl

Abstract

Context-aware applications are emerging on the daily basis and location information proves to be one of the key components to develop context-aware applications upon. This stems from the fact that location information enables and facilitates reasoning about what users are doing (user's behavioural patterns) and what users are interested in. Over the years many localization algorithms have been evolved to determine the location of entities using various technologies. In this paper, a novel localization technique using the geometrical properties of the building to increase accuracy, thus eliminating the need for calibration is presented. In addition a privacy-sensitive, location-aware service architecture called FLAVOUR is proposed, which utilizes the presented localization technique. FLAVOUR uses existing WLAN infrastructure for cost efficiency, and uniquely incorporates the location information into Jini service discovery platform. A new method of classifying existing WLAN localization techniques is introduced and the important evaluation criteria for assessing such techniques are discussed.

Keywords: location-aware, context-aware, privacy-observant, localization, WLAN, Jini

1 Introduction

We all have occasionally experienced being alone in a foreign territory. Naturally, it had come to our mind it would have been nice if we were accompanied by a trustworthy native person who knows a great deal about the area, places worth visiting, and how to find our way and our interest points, etc. As unrealistic as it may sound, that is exactly what this research aims at, i.e, building a mobile guide to (temporarily) be your best friend when you are attending a conference.

The idea is built on top of already existing *wireless campus* at the University of Twente (UT), in the Netherlands. Equipped with 650 individual wireless network access points, with each point having a range of about 100 meters, in June 2003, UT announced the launch of its wireless campus. In short, spread over 140-hectare campus, UT offers its staff, students, as well as its visitors, i.e., anyone with a desktop, laptop, handheld or wireless fidelity (Wi-Fi) devices to wirelessly access the university's network and the internet from everywhere on the campus even from the university's pool [2]. Availability of such infrastructure is a strong driving force towards building useful applications as well as practical use cases upon. One of such use cases is our conference aid, which will be provided for the first time to the participants of the 4^{th} Annual Conference on Scalable Vector Graphics (SVGopen 2005) taking place in August 15-18 at UT.

Appearing under different names, the most basic and popular services for conference attendants can be grouped into the following three categories:

1. *Finding fellow attendees*. Colleagues attending the conference may want to participate in many of the parallel sessions they are interested as possible. Thus, by locating colleagues who share the same interests, one can check whether his/her friend joins one of the presentations and can update him/her about. People also want to find colleagues and friends during the conference in order to have lunch or a coffee together. We do not want to provide an anonymous tracking functionality by which attendants can be tracked without being aware of. Instead, the attendee decides who, when and how long can be aware of his location and people are given access rights accordingly.

2. *Locating and using resources*. Easily finding out about available resources is always useful. Some resources are as simple as location of static points of interest, such as the restaurant, the conference rooms where the talks are taking place, Internet access rooms and coffee machines. We also want to provide easy use of resources available in the infrastructure such as projectors and printers. The user can then seamlessly send a document to print and be shown the location of the printer that has his document. The system also provides additional information about the resources, e.g. the current presentation in a certain room or if there is a computer available in the Internet access room.

3. *Receiving messages and notifications*. Instead of using an announcement board, conference organizers can use a messaging mechanism to reach all attendants. This is handy for organizational announcements, such as changes in schedules, session cancellation or diversions, as well as social events announcements.

In the first two categories, *location* of attendants is an important player in locating and guiding towards their points of interest as well as fellow attendants. The first two categories are examples of *pull* services, in which attendants request some information from the system and the system, in return, provide them with respective information based on their location. Therefore, such location should be first determined in the best possible extent. This brings us to the issue of *localization*, which will be addressed in a separate section.

The last category is an application of *bulk messaging* to conference participants. Depending on the nature of sent messages, smart approaches can be followed. For instance, emergency messages should target *everyone*, while a session cancellation message may be sent only to those interested in or subscribed to that particular session. In this case text messages are *pushed* to attendants without any trigger/request from them.

Our proposed architecture to achieve our goals and being able top offer these services is addressed in Section 2. Section 3 describes the methodology we would adopt to get the users' location information. In Section 4 the conceptual design of the GUI is presented. Finally, the Section 5 concludes the paper.

2 Architecture

In this section, we first identify the following issues as high level technical requirements for the system:

- the users do not have to be equipped with specialized hardware,

- the system should be able to determine users's location indoors as well as outdoors with a reasonable accuracy and the transition of such data should be transparent to the user,

- the system should keep the user's location private in such a way that the privacy involved do not restrict the users from using the services provided. The users can decide who has access to their location information, when, and how long,

- the system's user interface should be lean. This stems from shortage of resources on the user's devices.

We have decided to base our system on the use of the WLAN infrastructure as devices equipped with a WLAN card (e.g. laptops and PDAs) can determine their location without any additional hardware. As it will be shown later, we only need to install a software (*Signal Strength Acquisitor*) on the devices to measure the signal strength of the *access points* in the vicinity of the device. This has another advantage of making the client "light-weight" with regard to memory allocation. Since this application does not consume much of the program memory, the users can run other applications without facing any conflicts.

In turn, the *Location Manager*, which can either reside on the mobile device or in the fixed network infrastructure, uses the signal strength measurements and information about the access points to compute the location of the user. Figure 1 illustrates the information flow to compute the location of the user. To be able to obtain the location of the device as accurate as possible, while keeping an eye on the calibration effort, we propose a method which will be explained in details in section 3.5.

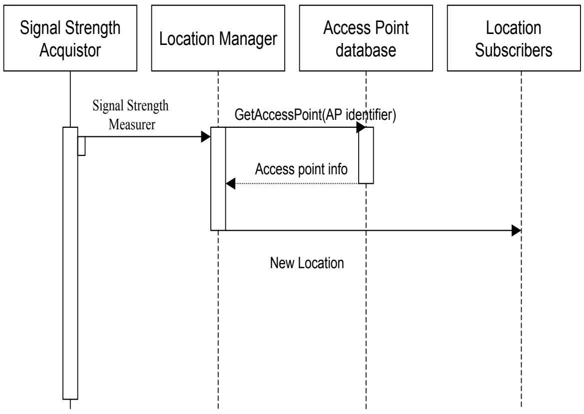

Figure 1: Information flow of user's location computation

Despite of running the *Location Manager* on the fixed infrastructure our proposed architecture is *decentralized* since each user has his own *Location Manager* computing his personal location. At present we cannot guarantee absolute privacy of the data when the *Location Manager* runs in the infrastructure, as in principle it is possible that the administrator of the system hosting the software 'spies' on the user. However at this stage, since our main focus is on providing a test bed for demonstrating the architecture and the localization mechanism, we will assume that the system administrator is "trust-worthy".

Figure 2 shows our proposed architecture of the system. The *Location Manager* provides services using the *Jini* platform [7]. Each *Location Manager* registers with the *Jini Lookup Service* to offer the location of the user it represents. Interested users can look up the service on the *Jini Lookup Service* and subscribe to the location of a given conference participant. This is done using *publish-subscribe* mechanism. The *Location Manager* uses a *privacy policy* to decide if a client is allowed to subscribe to the location of its owner *(publisher)*. The *Location Manager* publishes to all the *subscribers* relevant changes in the location of its owner.

We also use the Jini architecture to provide other kind of services, such as the *message board* to which every conference participant can subscribe. The message board is used by the conference organization to publish changes in the schedule, information related to the social event, etc. Additionally the participants can specify their interests (e.g. conference tracks, particular talks, special events, etc.), which in turn is used by organizers to send appropriate messages only to those who are interested.

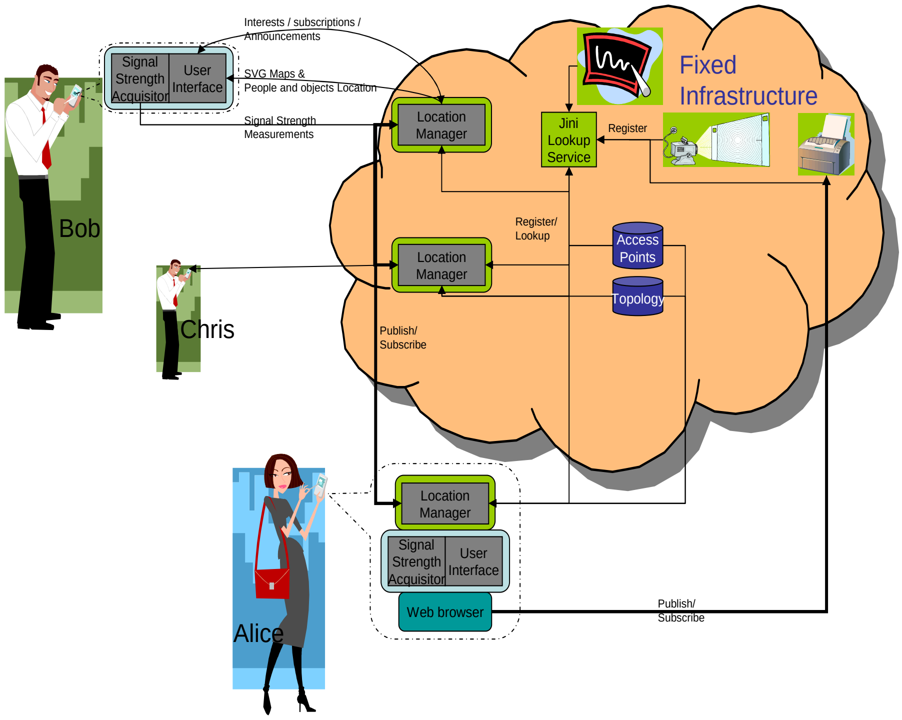

Figure 2: System Architecture

This prevents the user from being spamed. The participants can also use the message board to make announcements to the other participants, as for example asking about lost objects.

If the *Location Manager* is running on the fixed infrastructure, it is able to provide the last known location of its owner (and the time when he was last seen), even when the mobile device of the user is switched off and thus it is not reachable. It also allows subscriptions to take place when the user is offline. If the *Location Manager* is running on the mobile device, the service will disappear every time the device is switched off and the subscriptions to it will be terminated. The latter situation implies that the participant will not be visible when his device is switched on again.

The user can subscribe to the location of other participants using his *Location Manager*. Additionally, the *Location Manager* composes the SVG maps that show the user where he is and where the people and objects he is interested in are. The *user interface* can thus be seen as a client of the user's *Location Manager*. The user interface is a topic by itself and hence, although rather brief, will be addressed in section 4.

One should note that the communication between the *Signal Strength Acquisitor* residing in the client device and the *Location Manager* residing in the infrastructure is not accomplished using Jini, but using an ad-hoc protocol.

For the sake of clarity, now let us assume that Bob, Alice and Chris are three conference attendees using the services provided by our architecture. As it can be seen from Figure 2 Bob and Chris run the *Location Manager* on the fixed infrastructure, while Alice runs it on her mobile device. If they want to be aware of each others whereabouts, they need to subscribe to each others *Location Manager*. Each subscription triggers a request message which can be either *accepted* or *rejected*. In case of acceptance, the publisher has the right to specify when and for how long his/her location can be transparent to the subscriber. All this information is kept in *Jini lookup* service. In addition, Alice is also using Jini to print pages from her web browser on a fixed printer available at the conference.

So far, we have been addressing conference attendees carrying laptops or PDAs. One particular service, that we are aiming at is providing support to conference attendees without personal devices using

Smart Signs. Smart signs are small, static sensor nodes with limited interface capabilities. Using a fixed desktop on the infrastructure, the conference attendee can pose a query asking for directions towards a specific point of interest. Smart signs will in turn act as an interface and provide the needed directions to the user to be able to reach his specific point of interest. An important issue here is that smart signs require *user identification*, therefore we supply the attendees with RFID tagged sensor nodes. Smart signs can be quite chaotic when multiple users querying at the same time. Thus, use of sensors with small displays for multiple users requires further studies.

3 Localization

Localization is defined as a mechanism to find spatial relationship between objects [4]. Fundamentally speaking, to function location systems require some kind of inputs. Such input can, for instance, come from sensors, i.e., coordinates, or information from access points (APs) such as signal strength. Regardless of what the source of this information is, it is then used as an input to any of the location techniques such as triangulation, proximity, or scene analysis to derive objects location (either absolute or relative location). A detailed survey on the techniques and technologies that are enabling localization in indoor and outdoor environments can be found in [9]. There is a growing demand for indoor location systems, mainly because people tend to stay most of the time indoors- either at work or at home. So lots of research is focussed on indoor location technologies.

There are different types of indoor positioning technologies. Depending on the required range, propagation speed, cost, precision, bandwidth, etc., one can choose the required technology for a specific application. Examples include infrared-based, ultrasonic-based, electromagnetic-based, inertial-based, optical-based, and radio frequency-based systems. Depending on the type of frequency range used, Radio Frequency can be categorized into RFID (Radio Frequency IDentification), WLAN (IEEE 802.11b), Bluetooth (IEEE 802.15), wide area cellular, and Ultra Wide Band. Each of the technology will have a specific technique of measuring the location information. More details can be found in [9].

3.1 Why Wi-Fi based Localization

There are various reasons for heading towards Wi-Fi based localization. One of which is the fact that it provides economical solution. Since the wireless network infrastructure already exists, without adding any additional hardware localization can be done by a software-only method. Secondly, when compared to other radio techniques like bluetooth or RFID, the range covered by WLAN is more, reaching approximately 50-100m. Thirdly, it is scalable as wireless networks are being deployed at all the important places like universities, airports, offices, shopping malls, etc. Finally, line of sight condition does not exist.

Due to the aforementioned advantages, Wi-Fi based localization methods have received enormous amount of attention.

3.2 Related work on WLAN based localization

In this section we review some of the important research developed in the area of WLAN based localization.

RADAR is a RF-based location system, which is mainly used to track users inside the buildings. It operates by recording signal strength information from multiple base stations. In addition a centralized system is used to perform triangulation and consequently to compute the location of the user. RADAR system combines empirical measurements with signal propagation modeling to determine the user location [10]. User's location can also be computed by probabilistic approaches [11] or neural network model [5]. Joint clustering [14] and Bayesian Networks [8] are similar to RADAR. They all use a training session to get many fingerprints and from them they try to predict the location. The main disadvantage of these methods is their lack of scalability due to the need for extensive training.

Ekahau positioning system [1] is a software tool, which is able to locate targets and provides the coordinates (x,y, and floor) corresponding to each client. The main positioning module is run on the server or a PC. It gives an accuracy of about 1m, however it requires quite a lot of calibration effort.

Place lab [6] is a new initiative developed by Intel, which allows the commodity hardware clients like laptops, PDAs and cell phones to locate themselves by scanning for radio beacons such as 802.11 access points, GSM cell towers and fixed bluetooth devices. It does not involve much calibration, as information about the access point, GSM cell towers are collected through the process of *war driving*. However it has been demonstrated only for outdoor environments where radio propagation is not harsh. The reported accuracy ranges from 13 to 20m [6]. It is maintaining a privacy observant architecture to compute the locations of the users, by performing all the computations at the client device.

The problem in these aforementioned state-of-the-art is that the location systems either involve too much of calibration efforts or give too little accuracy. So we set our research objective to be reducing the calibration effort and incorporating better accuracy. At first, we present our taxonomy of WLAN-based positing systems.

3.3 Taxonomy of WLAN localization

Figure 3 illustrates the taxonomy of the existing location systems.

In principle, localization system based on WLAN infrastructure can be taxonomized as *Hardware-based systems* and *Software-based systems*. Hardware-based systems use additional hardware on top of the existing infrastructure. Examples include systems based on *Time of Arrival (TOA)* or *Angle of Arrival (AOA)* techniques. On the other hand, software-based systems provide cost effective solutions, as they do not require anything in addition to the existing infrastructure. They are based on signal strength technique. The *Cell of Origin* method reports only coarse-grained accuracy. *Signal Strength* method can be grouped as either being based on *Radio map* or *Model*. Radio map based method requires location fingerprinting done at various places and requires extensive calibration effort. The benefit that is obtained with regard to extensive calibration is the accuracy. It can also be called as *trainable algorithms*. In *empirical method*, an RF map is created before location positioning phase. This map stores the signal strengths at each location from all access points that can be read from that area into a database. When a device requests a location, it sends the signal strengths from all access points to the server, which holds the database. The server now searches the database to find the closest match and returns this match as a location. However, this requires a large amount of manual effort. Also of importance is the time of the day when the RF map is created, since the radio wave properties in an indoor environment can vary greatly depending on the number of people inside the building. A solution to this could be to create several RF maps and choose the most appropriate one, however intense manual labour is involved. *Propagational method*, is based on the fact that as radio wave propagates in an environment it loses signal strength and the amount of signal strength the radio wave loses is dependent on the type of environment. The loss of signal strength can be modelled using propagational models like Hata-Okumura model. The accuracy of the propagational model can be improved by increasing the complexity of the model used. While model-based method (also called as *non-trainable algorithms*) requires only the access point information and includes the internal geometry of the building to predict the location. Thus the amount of calibration effort is negligible.

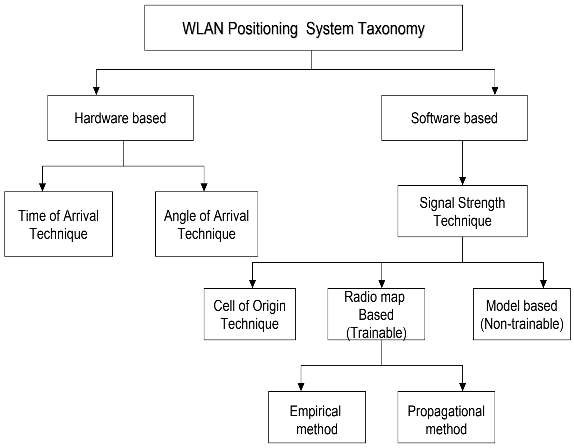

Figure 3: Taxonomy of existing WLAN based location systems

Table 1: Overview of the implemented location system using WLAN infrastructure

Positioning Technique and Feature	**Cell of Origin**	**Signal strength (Triangulation)**	**Signal strength (Fingerprinting)**
Technique	It makes use of the BSSID of the access point to which the client device is associated with	It measures the signal strength of the neighboring access points and estimates the distance either using probabilistic technique or empirical methods	It measures the current signal strength and matches the entry with the pre-computed signal strength obtained in the offline phase
Minimum Access point requirement	1	3	3-4
Accuracy obtained	10-25m	5m	1-2m
Calibration effort	No calibration	Calibration effort significant	Laborious calibration needed

3.4 Evaluation criteria for WLAN location systems

We have defined the following parameters, which can be used as guidelines to compare and evaluate several indoor location/positioning systems.

1. *Accuracy and Precision* of estimated location are the key metrics for evaluating a localization technique. Accuracy is defined as, how much the estimated position deviates from the true position and is denoted by an accuracy value and *precision* value (e.g. 15 cm accuracy over 95% of the time). The precision indicates how often we expect to get at least the given accuracy. The accuracy of a positioning system is often used to determine whether the chosen system is applicable for a certain application.

2. *Calibration* is also very important. The uncalibrated ranging readings are always greater than the true distance and are highly erroneous due to transmit and receive delays [13]. Device calibration is the process of forcing a device to conform to a given input/output mapping. Often there is a tradeoff between the accuracy and the calibration effort.

3. *Responsiveness* is defined as how quickly the location system outputs the location information. It is an important parameter, especially when dealing with mobility. However, this parameter is mostly ignored in the description of the existing systems.

4. *Scalability* is a significant parameter, as the proposed design should be scalable for large networks. If an approach is calibration intensive then eventually it is not a scalable solution.

5. *Self-organization* is of great importance, as it is infeasible to manually configure the location determination processes for a large number of mobile devices in random configurations with random environmental characteristics.

6. *Cost* is also a crucial issue. It includes the cost of installation, deployment, infrastructure and maintenance.

7. *Power Consumption* is of great concern when running the system in a real environment. It is clearly not feasible to change or recharge batteries very often when scaling up to thousands or millions of autonomous small devices. Thus *energy efficiency* should be a goal of any localization mechanism meant for a large scale system.

8. *Privacy* arises major concerns and should be definitely taken into account since its conception is important. Using localization it is very easy to create a Big Brother infrastructure that track users movements and allow to deduce patterns of behavior. However, this issue is being generally overlooked in the design of systems and considered as an after thought only. Centralized systems are particularly weak with regard to privacy.

3.5 Methodology of obtaining location information

In this section we provide our methodology for estimating the user's location. Our approach differs significantly from the others, as we propose a *Model-based approach*. Model-based approach is mainly introduced in order to reduce the complexity involved in the training sequence. In all the existing indoor positioning systems, a trade-off between system accuracy and the training workload can be seen. Our main goal is to have less calibration while not degrading the quality and the accuracy. We introduce the knowledge of area topology in the algorithm, which will result in considerable improvement in the accuracy of our indoor positioning system.

As in all the other WLAN based indoor positioning systems, our positioning algorithm relies on the observed signal strength distribution as its input to determine the location. The access point in the test area is marked as;

```
AP= {ap1, ap2 ,ap3...apj}
```

And the location of the access points in 3D coordinate system is maintained in a database as a part of University's administration. At any other unknown location *n* in the test area, where the user location needs to be obtained, the client device scans for access points in the vicinity. At each of these unknown points the variation of the signal strength readings will be;

```
At each n ;
Signal strength varies,
0>=SS<=MAX
```

Table 2 shows an example of a scanning.

In a simple case, we can assume that the user is located in the vicinity of the location of the strongest scanned access point. If the device gets only one access point in the scan, then it estimates its location in regard to that access point by taking into account the heard access point. But if the device scans two or more access points, then based on the strong signal received, it can employ either triangulation or proximity technique and can thus infer user's location. The location coordinates of the access point whose signal strength is the highest, is obtained by looking into a database of access points and picking up the location of the access point as the location of the user. This algorithm will essentially use only the signal strength information received from the access point. But due to inherent variation in the signal strength at indoor environments this method is not reliable.

The accuracy of this method can, however, be improved by introducing the knowledge of internal geometry of the building and the previous traces (history) of the users.

Consider a user, whose prior position estimations will follow a certain probability distribution. The knowledge of his pre-determined location can be used to predict the users movement in a certain area. For e.g. , assume that there are already *'p'* predetermined positions prior to the users new location *'q'* such as:

```
p={p1,p2,p3..}
```

there is a high probability that the new location 'q' is near the vicinity of the predetermined positions.

Supposing that due to the fluctuations in the signal strength, during scanning phase, it returns a wrong access point as the one with the strongest signal (refereing to table 2), the algorithm computes the difference in the distance between the previously connected access point with the currently connected access point. If the difference is too large then it is clearly not possible for a walking user to go that far within short amount of time. This introduces the role of geometry of the building and the location of the access point in the location estimation process.

Table 2: An example of access points scan

AP BSSID	SignalStrength	SSID
000b5fd00de8	-75	WLAN
000b5fd00de7	-55	WLAN
000b5fd00fe9	-60	WLAN

4 Graphic User Interface (GUI) design

User interface is important for both the user and the system. On the one hand, it is used as an output device, i.e. a presentation tool for the system, to provide users with appropriate services. On the other hand, it is used as an input device through which the user can interact with the system.

For the basic desired services described earlier in Section 1, following three target levels can be identified:

- Services without maps, such as bulk messaging,
- Services with not so intelligent maps, such as locating points of interest and other people, and
- Services with intelligent maps, such as navigation towards points of interest and fellow conference attendees.

Providing users with some coordinates representing their location is not very informative. That would even be meaningless to most, if not all, users. Therefore, some sorts of map representations are desired. To do so, a base map of the conference venue is needed. However, not all services require the same level of detail. As far as topology and detailed meta data about base map are not needed, dumb maps, i.e., static maps with limited meta data about points of interest as well as the base map, can be used in which special symbols are used to show points of interests as well as location of other people on the map. Figure 4 illustrates the conceptual design of the GUI.

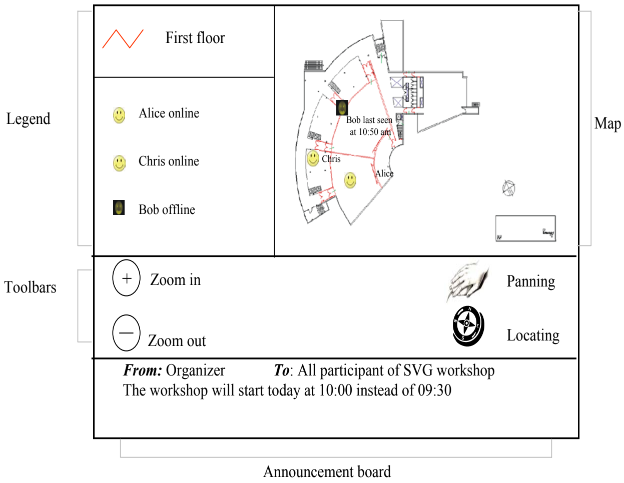

Figure 4: Conceptual design of the GUI

As mentioned earlier, the GUI is a users's gateway to the system to received push services and trigger pulled services. Using the GUI users pass their queries to the system. As soon as the system receives the user's request, the *data processing* phase is started, in which the user's query is processed and the corresponding results are produced. Consequently, the next phase, i.e., *map generation*, is carried out to prepare a graphic visualization of the results produced by the data processing phase. The generated map is then sent to user's device and is illustrated on GUI. Figure 5 illustrates such information flow.

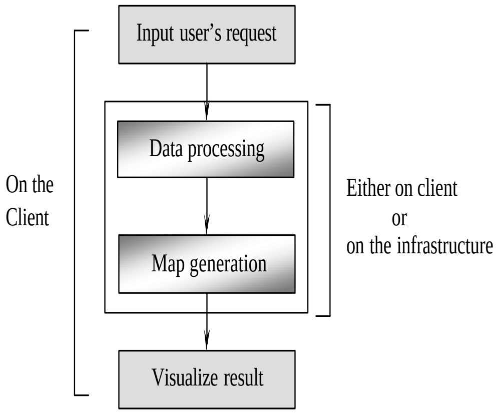

Figure 5: Information flow in pull services

4.1 SVG's role in map visualization

Regardless of their contents, digital maps can mainly be classified into two classes—static and dynamic (interactive). The former is nothing more than traditional paper maps now stored in a computer. Associating static maps with functionalities such as zooming (in/out), panning or hyperlinks results in the latter group.

Besides lack of interactivity, unchangeable resolution and large sizes, static maps suffer from possibly long download time while transferring map, specially when having slow connections. Like any other dynamic application, ours requires an highly interactive interface. In this regard, vector graphics are superior compared to image files in raster format. Among all vector formats available, we decide to use SVG (Scalable Vector Graphics).

SVG is a language for describing two-dimensional graphics in XML, which allows dynamic and interactive drawings [3]. SVG proves to be a better choice to design the user interface among all other available vector formats, simply because it inheres all XML capabilities such as, interoperability, tools to create geometric objects, extensibility, easy manipulation and transformation [12]. These together with SVG's own features such as small file size, support for geometric elements on various platforms, scalability and support for zooming without degradation make SVG a suitable tools for map visualization interface.

Users can view SVG maps using either *(i)* a stand-alone SVG viewer or *(ii)* an embedded SVG viewer as a plug-in in web browser.

5 Conclusions and future work

In this paper, we present FLAVOUR, a privacy-sensitive, location-aware service architecture for the conference environment. FLAVOUR uniquely incorporates location information into the Jini service discovery platform, to provide conference participants with navigational support and service sharing based on location. The proposed architecture facilitates the availability of location information even when the user is offline. In conference environment, compatibility with heterogenous platform is needed, as devices can operate in different operating systems.

We also propose a localization technique which uses geometrical properties of the building to increase accuracy, therefore eliminating the need for extensive calibration. FLAVOUR is a cost-efficient architecture because it uses existing WLAN infrastructure. We also introduce a new method of classifying existing WLAN localization techniques.

On-going work includes prototyping FLAVOUR and enhancing the localization accuracy. At present, privacy is incorporated by having a personal agent running as the location manager in the Jini infrastructure, in future we would incorporate more privacy policies. Also we aim at extending FLAVOUR to encompass campus-wide location-based service.

6 Acknowledgements

This work is part of the *Smart Surroundings* project, funded by the Ministry of Economic Affairs of the Netherlands under the contract no. 03060.

References

[1] Ekahau positioning system.

[2] http://www.newscientist.com/article.ns?id=dn3834.

[3] http://www.w3.org/tr/svg/intro.html.

[4] Nirupama Bulusu. *Self-Configuring Location Systems*. PhD thesis, University of California, Los Angeles, 2002.

[5] J.Small, A.Smailagic, and D.P.Siewiorek. Determining user location for context aware computing through the use of wireless lan infrastructure.

[6] Anthony LaMarca, Yatin Chawathe, Sunny Consolvo, Jeffrey Hightower, Ian Smith, James Scott, Tim Sohn, James Howard, Jeff Hughes, Fred Potter, Jason Tabert, Pauline Powledge, Gaetano Borriello, and Bill Schilit. Place lab: Device positioning using radio beacons in the wild. In *Proceedings of Pervasive'05*, May 2005.

[7] Sun Microsystems. http://www.sun.com/jini. Technical White Paper, December 1999.

[8] Andrew M.Ladd, Kostas E.Bekris, Algis Rudys, Guillaume Marceau, Lydia E.Kavraki, and Dan S. Wallach. Robotics based location sensing using wireless ethernet. In *Proceedings of the Eigth ACM International Conference on Mobile Computing and Networking (MOBICOM)*.

[9] Kavitha Muthukrishnan, Maria Lijding, and Paul Havinga. Towards smart surroundings: Enabling techniques and technologies for localization. In *Proceedings of the international workshop on location and context awareness (Loca2005)*, 2005.

[10] P.Bahl and V.Padmanabhan. RADAR: An inbuilding RF based user location and tracking system. In *IEEE Infocom*, volume 2, pages 775–784, March 2000.

[11] P.Castro, P.Chiu, T.Kremenek, and R.Muntz. A probabilistic room location service for wireless networked environments. In *Proceedings of Ubiquitous computing*, 2001.

[12] Hegde S. Potential of svg for a cartographic interface to a route optimization model for the transport of hazardous material. Master's thesis, ITC, 2004.

[13] Kamin Whitehouse. The Design of Calamari: an Ad-hoc Localization System for Sensor Networks. Master's thesis, University of California at Berkeley, 2002.

[14] Moustafa Youssef, Ashok Agrawala, and Udaya Shankar. Wlan location determination via clustering and probability distributions. In *Proceedings of IEEE PerCom 2003 (PerCom03)*, march-2003.