SVGOpen Conference Guide: An overview

Kavitha Muthukrishnan,Nirvana Meratnia,Georgi Koprinkov,Maria Lijding and Paul Havinga

University of Twente, Faculty of Computer Science
Computer Architecture Design and Test for Embedded Systems group
P.O.Box 217, 7500 AE, Enschede, The Netherlands
{k.muthukrishnan,n.meratnia,g.t.koprinkov,m.e.m.lijding,p.j.m.havinga}@ewi.utwente.nl

Abstract

Context-aware applications are emerging on a daily basis and location information proves to be one of the key components in this domain. 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. Availability of campus-wide WLAN infrastructure at University of Twente (UT) and the fact that SVGOpen 2005 was scheduled to be held at UT, were two strong driving forces towards building a location-aware conference guide. In this paper, a privacy-sensitive, location-aware service architecture is presented, which utilizes a calibration-free localization technique. The presented architecture uses existing WLAN infrastructure for cost efficiency, and uniquely incorporates the location information into Jini service discovery platform. Vector graphics provide better support for highly dynamic interface. Among all available vector formats, SVG proves to be a better choice to design the dynamic user interface and hence it was used in our implementation.

1 Introduction

Equipped with 650 individual wireless network access points, with each point having a range of about 100 meters, in June 2003, University of Twente (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 [5]. Availability of such an infrastructure and the fact that SVGOpen 2005 conference was scheduled to be held at UT, were two strong driving forces towards building useful applications.

One such application is FLAVOUR, i.e., Friendly Location-aware conference Assistant with priVacy Observant architectURe, which is used for the first time during SVGOpen conference [3]. Services offered by FLAVOUR can be categorized into:

- *Pull services, in which location of attendants play an important role as the attendants' request will be replied by the system on the basis of their whereabouts. Examples of offered pull services at the moment are:*
  - *Finding fellow attendants*
  - *Locating and using resources available in the infrastructure such as printers, copiers, coffee machines etc.,*
- *Push services, in which individual and bulk messages are sent to the attendants. This will enable the attendants to:*

- *Be notified about important events by conference organizers*
- *Communicate with their contacts, i.e., colleagues, friends, etc.*

2 Architecture

To support the aforementioned services, we identify the following issues as high-level technical requirements:

- 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 user interface should be lean due to the shortage of resources on the user's devices.

Figure 1 shows our proposed architecture. The *Location Manager* provides services using the *Jini* platform [4]. 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)*. It also publishes to all the *subscribers* relevant changes in the location of its owner.

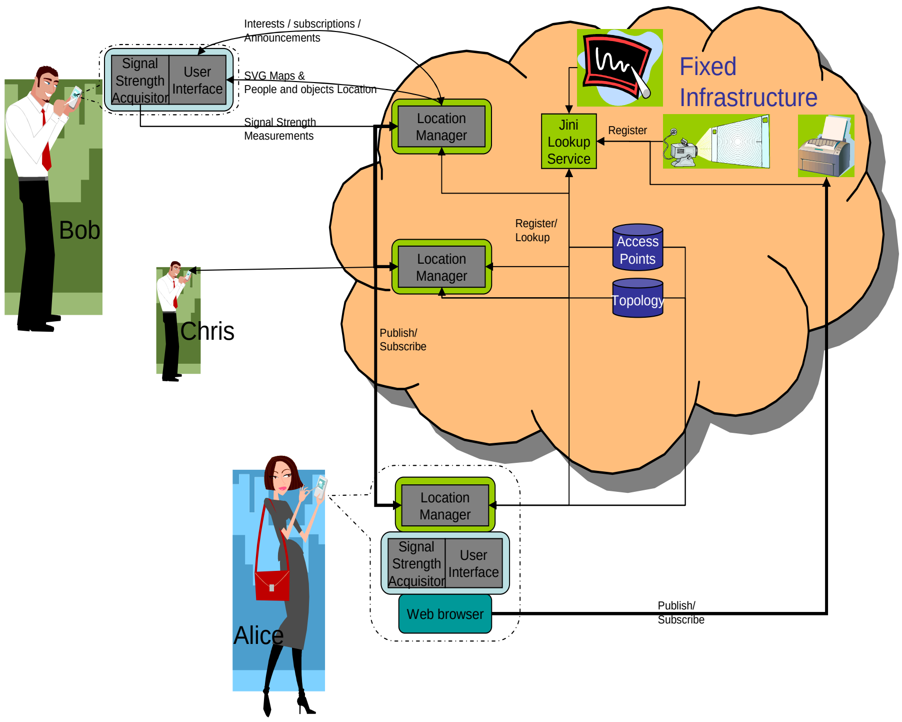

Figure 1: System Architecture

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 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. 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, or to chat.

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 off-line. 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.

So far, we have been addressing conference attendants carrying laptops or PDAs. One particular service, that we are aiming at is providing support to conference participants 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 attendants 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 requested directions to the user to be able to reach his specified destination. An important issue here is that smart signs require *user identification*, therefore we supply the participant 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

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, compared to other radio techniques such as bluetooth or RFID, the range covered by WLAN is more, ranging approximately $50 - 100\ m$. 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.

There are numerous applications, being employed in various environments and used by diverse user groups, which utilize WLAN-based localization. However the main drawback of WLAN-based localization is the *calibration* phase of signal strength as a function of a particular location. The calibration needs enormous amount of manual labor and should be performed repeatedly. Little work has addressed the issue of reducing the amount of calibration effort. As a rule of thumb, there is a trade-off between the amount of effort put on the reducing the calibration and the accuracy obtained.

3.2 Methodology

Our Wi-Fi based positioning algorithm is dependent on an initial mapping stage, in which the coordinates of all the access points of the conference venue in 3D coordinate system were recorded in a database. Mapping stage was not costly, since this work was done as part of University's administration.

As in all the other WLAN-based indoor positioning systems, our localization algorithm relies on the observed signal strength distribution as its input to determine the location. Our main goal is to have calibration-free localization preserving the quality and accuracy.

The Wi-Fi card periodically scans its environment to discover access points in the vicinity. During the access point scanning phase, the MAC address of the heard access point and its corresponding signal strength is recorded. Table 1 shows an example of such recorded data.

Table 1: An example of access points scan

AP BSSID	SignalStrength	SSID
000b5fd00de8	-75	WLAN
000b5fbcc0e0	-91	WLAN
000b5fd7f214	-88	WLAN
000b5fd00d2e	-82	WLAN
000b5fd7f1c5	-45	WLAN
000b5fd7f1d6	-61	WLAN

At any unknown location n in the conference venue, the variation of the signal strength will be;

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

However, the signal strengths are often associated with noise. In order to have a better estimation of the actual location, we employ *exponential moving average filter* to smoothen the signal strength. Equation 1 shows the formula used in doing so, in which $\alpha = 0.125$ and SS denotes the observed signal strength:

$$CurrentSS = \alpha * (1 - CurrentSS) + \alpha(previousSS) \quad (1)$$

To compute the users location, the top three access points having the best signal strength are chosen. An estimation of users location is carried out through applying triangulation technique. If the device gets only one access point in the scan, it estimates its location in respect to that access point. However, since triangulation technique is used, the localization always leads to a better result if at least three access points in the vicinity are heard.

4 SVG's role in map visualization

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.

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, static maps with meta data about points of interest can be used. Associating these maps with functionalities such as zooming (in/out) and panning will increase user interaction with the system.

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 [2]. SVG proves to be a better choice to design the user interface among all other available vector formats, simply because it inherits all XML capabilities such as, interoperability, tools to create geometric objects, extensibility, easy manipulation and transformation [1]. 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.

Figure 2 shows a snapshot of our GUI.

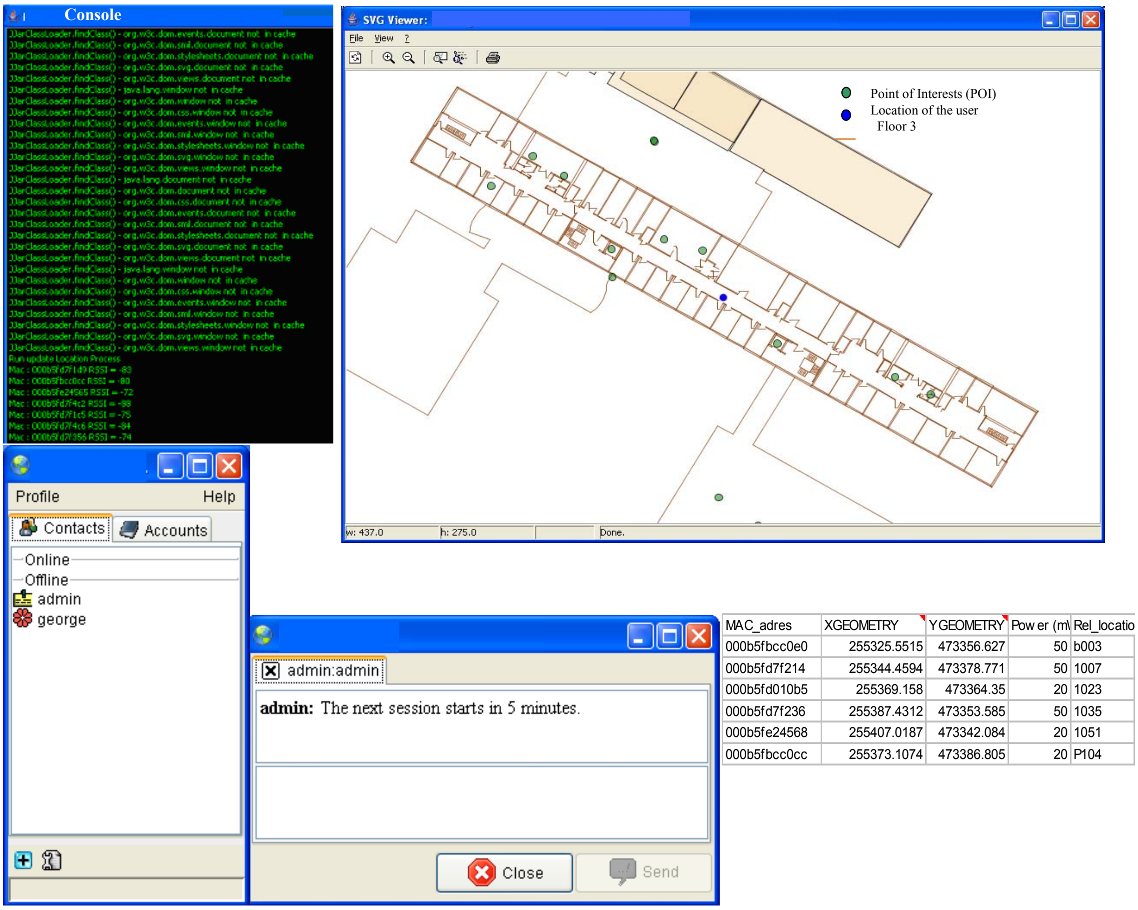

Figure 2: Graphical Interface of FLAVOUR

5 Conclusions and future work

In this paper, we present, a privacy-sensitive, location-aware service architecture for conference environment. Our architecture uniquely incorporates location information into the Jini service discovery platform to provide conference participants with service sharing based on their location. It also facilitates the availability of location information even when the user is off-line. Another advantage of the presented architecture is *cost-efficiency* because it uses existing WLAN infrastructure. A *calibration-free* localization technique based on triangulation was presented. Among all available vector formats, SVG proves to be a better choice to design the dynamic user interface and hence it was used in our implementation.

On-going work includes enhancing the accuracy of localization technique. Not to end up with a big-brother scenario, we plan to incorporate more privacy policies in the future. Last but not the least, we also aim at extending FLAVOUR to encompass campus-wide location-based system.

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.

Authors would like to thank Department of Information Technology, Library & Education (ITBE) of University of Twente and Drs. Barend Köbben from ITC, respectively, for manual mapping of the access points and providing enormous help in respect to geo-database.

References

[1] Hegde, S.: Potential of SVG for a cartographic interface to a route optimization model for the transport of hazardous material. Master thesis. ITC. 2004

[2] http://www.w3.org/TR/SVG/intro.html

[3] Kavitha Muthukrishnan, Nirvana Meratnia and Maria Lijding: FLAVOUR- Friendly Location-aware conference Aid with priVacy Observant architectURe, CTIT Technical report TR-CTIT-05-28, June 2005, 16 pp.

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

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