Campus-Wide Indoor Tracking Infrastructure

Heinrich Schmitzberger and Wolfgang Narzt
Department of Business Informatics - Software Engineering
Johannes Kepler University Linz
Altenberger Straße 69, 4040 Linz, Austria
{heinrich.schmitzberger, wolfgang.narzt}@jku.at

Abstract — **In the context of large-scale indoor spaces location-based services still lack a feasible technology for localization and tracking in terms of ubiquitous operation. In this article, we present a tracking system based on wireless local area network infrastructure that is capable of simultaneously tracking numerous and diverse mobile clients (i.e., cell phones, laptops, personal digital assistants and alike) in multistory buildings within a campus facility at a near real time resolution and without client software to be installed. In the course of a real-life project at the campus of the University of Linz we have been studying deployment issues and environmental influences on infrastructure-based tracking in a large-scale setup that comprises various types of building structures and architecture. We contribute our findings regarding effects of arbitrary environmental conditions on radio signal based person tracking and present our current results. Furthermore, we demonstrate the feasibility of integrating infrastructural tracking technology into a location-based services platform called "Digital Graffiti" that handles user and privileges management while sustaining the privacy of the individual.**

Keywords-indoor infrastructural wlan tracking; scalability; large scale; real time; location-based services.

I. INTRODUCTION

At the core of mobile computing is the most prominent context information about the user: location. The user's current location as well as the awareness of the location of friends and things of interest have been a decisive driver for the ongoing trend towards smart phones and mobile applications up to date. The integration of the Global Positioning System (GPS) into commercially available mobile devices (smart phones and personal digital assistants for instance) carved the way for a broad variety of location-based services (LBS) covering outdoor spaces and public places in the form of navigation systems, location-based social networks and city guide applications. However, still no convincing counterpart regarding indoor location acquisition has prevailed in real-life LBS scenarios. In the last decade, several approaches towards realizing a ubiquitous indoor localization technology that could compete with the quality and reliability of outdoor GPS have been presented. The underlying sensor technologies were manifold (Bluetooth, Infrared or Ultrasonic, just to name a few). Concluding from the GPS story of success, a key factor of influence is the broad availability of the respective technology in a common mobile device. As we demonstrated in [1], WLAN (Wireless Local Area Network) technology seems to have the most promising potential in this context. We pointed out that the approach of exerting tracking infrastructure for the localization of a large number of concurrent users has shown feasible, even in terms of large-scale setups covering vast building complexes (e.g., the campus of a university). Using WLAN in this context offers certain benefits concerning costs, accuracy, scalability and deployment compared to other popular radio localization technologies, not least because of its availability in modern mobile phones [2].

Recently, commercial WLAN localization products have been introduced [3][4] following a localization method having the mobile device acting as sensor or even estimating its position itself (client-based). In this article, we propose a converse system where sensor hardware as well as position estimation is decoupled from the client, but achieved by a backend server combined with WLAN infrastructure within a complex of buildings (infrastructure-based).

This setup allows us to look into the subject of indoor localization from a different point of view, being able to support a vast range of client devices. Instead of providing software for different client platforms and sensor arrangements our infrastructure comprises a network of homogeneous, permanently active sensors that assure accurate measurements for convenient location estimation of numerous clients operating on several platforms. By this means we avoid forcing CPU prerequisites since infrastructure bears the processing load. Consequently, power consumption is reduced on the client side.

Another benefit of our setup is its robustness with respect to a constantly changing WLAN environment, a disadvantage that affects accuracy in client-based setups because they mainly rely on stable signal strength fingerprints of access points (APs) in the vicinity. Since the system solely depends on measurements of client signals additional WLAN signal sources have no consequences on the position estimation process.

The system presented in this article meets the following prerequisites: first of all, localization accuracy of room-level granularity is reached. The system is able to provide a mapping of geo-coordinates to symbolic names at detailed spatial level (such as "*Office 306, Management Center, University of Linz*"), which facilitates users' perception of location information. In order to attain a ubiquitous user experience, the system operates 24 hours a day, 7 days a week and is open to public access from within the wireless campus network. Accordingly, our architecture has to attend to scalability in terms of providing a near real time service to a potentially unbounded amount of users. Furthermore, we have developed a service based on our LBS framework that enables the finding and tracking of friends given their anytime revocable permission. In this context we refer to the notion tracking almost similarly as to localization, with one distinctive exception: localization describes the process in which the current location of a certain client is estimated. Tracking on the other hand uses previously calculated location estimates to create a traceable path that is further used to render the client's position more precisely and to conclude to the client's course.

The rest of this article is organized as follows. Section 2 discusses related work to the broad topic of indoor localization with respect to WLAN signal strength based approaches concentrating on infrastructure. In Section 3 we describe the architecture of our system in detail, focusing on its 3 main components in particular. Subsequently, Section 4 deals with our real-life setup in the course of the *Smart Information Campus* project at the University of Linz, discussing detailed facts of campus-wide deployment as well as the users' view of the system. In Section 5, we compare tracking results surveyed in different types of buildings at the campus. Finally, Section 6 concludes with a discussion on our efforts to realize the described system and gives a brief outlook on future research questions in context of our setup.

II. Related Work

In the last years, numerous contributions have discussed the subject of indoor localization and tracking on WLAN basis as an alternative to obtaining positions using GPS. Most of them concentrated on signal strength localization algorithms that emphasized a client-based application, i.e., the client device is operated as sensor collecting signal strength information of nearby APs. These systems broadly depend on special hardware in the form of tags [4][5] or native libraries supporting certain network interface card (NIC) features at client-side [6][7][8].

We envision a client-device independent setup in order to not constrain the usage of our system to certain hardware. To this end we have studied an infrastructure-based approach to achieving WLAN signal strength localization. In this context the notion "infrastructure-based" refers to a setup that comprises stationary sensor nodes for measuring client radio transmissions [9][10][11].

Early contributions to this approach are presented in [12]. The RADAR system uses a setup of ordinary PCs as stationary signal strength sensors. This experimental setup showed the feasibility of WLAN for indoor localization. In [9], the LEASE system is proposed, an infrastructure-based framework using sniffers and reference-emitters. The sniffers are constructed as embedded sensor platforms. The emitters were used as signal strength references to constantly rebuild an active radio profile. The main focus of this work concentrated on deployment issues and consequences of radio propagation for location estimation in potential real life scenarios. A detailed description of the sniffers used for LEASE is given in [13]. In this article, the issues of capturing conventional WLAN traffic as well as the placement of sniffer nodes for convenient location estimation are explored.

Embedded devices have also been the basis for Pinpoint [14], a system that uses the Time-Of-Arrival localization method. In order to achieve accurate estimations, the main functionality of their devices was to maintain high precision clock synchronization. The usage of Linksys WRT54 (the previous generation of the devices used in our work) as platform for localization applications has been a matter of research as well. In [15], customized kismet software is used on a Linksys device to report signal strength measurements. The authors report experiments in a calibration-free localization setup. In [16], another calibration-free fingerprinting system is proposed that applies probabilistic methods for constructing radio profile model and position estimation. Due to heavy computational load it uses PCs as sniffers. In this work another focus lies on WLAN channel characteristics and fluctuations in signal strength.

However, all the systems depicted above discuss the infrastructural localization on a prototypical basis. Most of them solely cover a 2-dimensional area of interest; often a dedicated test bed is constructed to make a proof of concept. The work presented in this article reports experiences of a campus-wide real life setup comprising multistory buildings of various characteristics at the *Johannes Kepler University* in Linz. In this context we experimented with commercial products as well. Since the campus-wide WLAN network uses Cisco Aironet 1250 devices we evaluated a trial setup of the Cisco Wireless Location Appliance (Version 3.1.35.0) based on RSS fingerprints [17]. Our findings were that the update frequency of the clients' location was too low (one estimate in 30 seconds up to 1 minute) for our purposes, since it solely listens for client probe request frames. Our system in contrary uses a near real time resolution (updated every 3 seconds). Furthermore, we strongly focus on a scalable, robust system addressed to a public audience in order to provide a convenient LBS experience.

III. WLAN-based Tracking Infrastructure

Our main design objective was to implement a system capable of concurrently tracking multiple mobile clients within multistory buildings across a campus areal. We aim at serving a vast variety of mobile devices not requiring specific hardware but a WLAN interface. This had two profound implications on system architecture. First of all, the sensor component of the system is implemented infrastructural to avoid heterogeneity of measuring data. And second, the client's radio communication with the system can be reused as signal data for position estimation.

Even though indoor WLAN localization systems have been widely studied in the last decade most contributions concentrated on a client-based approach. With few exceptions (cf. Section 2) infrastructure-based WLAN localization has been relegated to a niche existence [13]. This is mainly due to the cost for an area-wide infrastructure deployment (i.e., purchasing sensor hardware, permanent power consumption, setup and maintenance costs and the alike) on the one hand [11], as well as privacy and security concerns on the other hand [18]. In fact, it is easier to sustain user privacy in a client-based localization setup since the respective information is computed on the client side. But if client-side contextual data is transmitted to an untrusted server for LBS consumption (as in most public application scenarios), this advantage does not remain existent.

Another important concern is system scalability. If location estimation for each client is calculated on a central backend server and not on each client device individually, the computation load poses an obvious bottleneck. A main focus of our research lies on this aspect. To overcome the scalability issue we explored the potential of the sensor infrastructure itself, further discussed in Section 3B.

Opposing these difficulties that arise as a consequence of our design decisions we have to point out some benefits as well. First and most of all, the system is able to operate without any client pre-requisites but a WLAN communication interface. At the moment we solely support WLAN 802.11b/g/n (802.11a/n support is a work-in-progress at the moment). This implies that every mobile device equipped with such an interface currently on the market is able to use our tracking service. Since consuming the localization service can basically be done via a web request, it poses an energy saving alternative compared to GPS for instance. Furthermore, no additional software needs to be installed at client-side. The system presented in this article consists of three main components (cf. Fig. 1) discussed in the following sections.

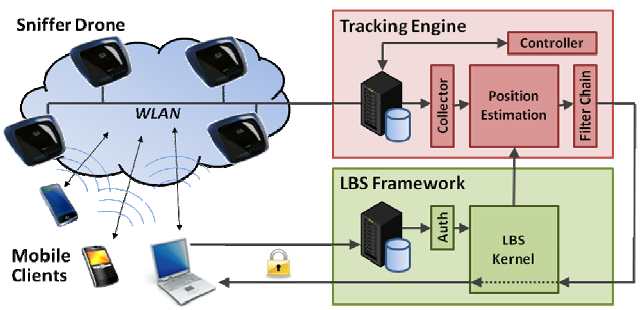

Figure 1. System Architecture

A. *Tracking Engine*

Our architecture bases on a two-phase signal strength fingerprinting system implemented in Java and hosted on a server backend. Every physical position used for location estimation is represented as a vector consisting of several tuples of a certain signal strength paired with a MAC address (cf. Table 1).

TABLE I. POSITION VECTORS

Position			Client Fingerprints	
LON	*LAT*	*ALT*	*Sniffer MAC*	*RSS*
14.3182073	48.3363918	5.0	00:23:69:3B:2C:A7	-42
			00:23:69:3B:2C:FF	-62
			00:23:69:3B:2E:E7	-78
14.3180591	48.3363577	5.0	00:23:69:3B:2C:FF	-58
			00:23:69:3B:2E:AF	-62
			00:23:69:3B:6D:77	-81

Traditionally, the MAC address identifies an AP that is observed by the client. Since the infrastructure-based setup estimates positions observing client transmissions on the contrary, the MAC address identifies the sniffer device that reads the client's signal strength. A database holding all of these positions (candidate points) is created in the initial training or offline phase, representing a radio profile for the targeted physical space. Fig. 2 (left) shows a clipping of an office floor with its respective candidate points. The colors of the candidate points indicate the quality of the signal level (green: ≥ 5 sniffer receptions, blue: ≥ 3, red: < 3). As the figure illustrates, each office comprises 2 to 4 candidate points with varying signal quality. To cope with several different transmitter characteristics (antenna properties, transmitting power) we created an individual radio profile for each supported device type (laptop, cell phone, PDA).

In the online phase, location estimation is computed with the commonly used Nearest-Neighbor-In-Signal-Space (or kNN) algorithm [12] that queries the database for the k best matches with the least Euclidean distance to the client's current signal vector. This approach has been discussed and proven feasible for position estimation in numerous publications [19][20][21] and won't be explained further. An important factor for the estimation quality especially when dealing with large-scale setups in this context is the accurate weighting of both the client's signal vector entries and the candidate points vector entries, which decisively accounts for the localization result. This weighting has to reflect the density of sensors in the vicinity of the candidate point. Consequently, it uses *(i)* the amount of different sniffer entries forming one fingerprint vector, *(ii)* the averaged signal level of the fingerprint and *(iii)* the actual strength of each entry in the vector to reflect the probability of the appearance of the signal at the respective position. If a vector entry is missing in comparison with the database, it has to be taken into account as well.

Since system scalability is of most importance we don't use probabilistic localization approaches that might provide better accuracy for account of CPU load, as the HORUS system [8] does for instance.

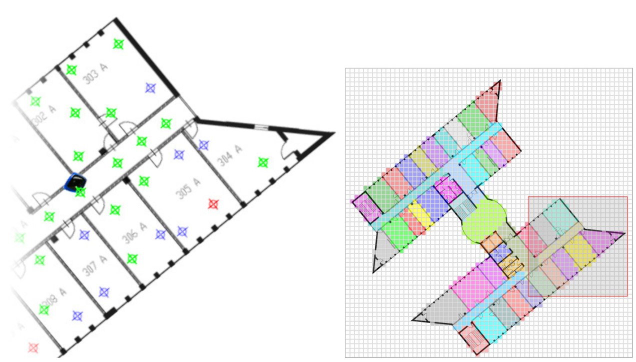

Figure 2. Candidate points (left), floor sectors (right)

Using signal strength (i.e., the RSSI value of the sensor device's WLAN NIC) for client localization has practical implications on the tracking engine. First of all, signal strength measurements underlie fluctuations. As [12][16] reported, multipath signal propagation and other propagation effects such as reflection, refraction, and scattering falsify the signal strength measurement. If not filtered (cf. Section 3B) or handled using appropriate mathematical models as the Wall Attenuation Factor (WAF) model [12], these effects can produce grave estimation outliers that have to be compensated later in the processing chain. Additionally, a person or a group of people between the measurement device and the transmitter can pose a dynamic signal attenuation component in the measuring process affecting especially radio signals at 2.4GHz frequency. Consequently, a daytime-dependent fluctuation in public spaces can be observed [22]. We are currently developing a mechanism to compensate such fluctuations by monitoring AP beacon signals.

Within an interval of 3 seconds (which corresponds to two channel hop cycles, cf. Section 3B) the engine calculates a position estimate. We represent a position according to the World Geodic System WGS84 as spherical polar coordinates in longitude, latitude and altitude. The estimate then passes a chain of filters. The first filter evaluates the plausibility of the estimate by testing if the position is within an accessible area of a building modeled in the database. Each building has at least one floor represented by a north aligned floor map and partitioned into several sectors (cf. Fig. 2 right). Each sector models a distinct type of an enclosed area, as rooms, hallways, stairs or elevators. If the position estimate is within a sector modeled as invalid, the estimate is identified an outlier and is not considered further.

The next filter bases on a set of rules that specify possible transitions from one sector to another. The rule set takes into account the building's characteristics, modeling direct paths, sector connections and next hop neighborhoods. Changing floors is only possible within a stairway or elevator sector. At this point, the track of the client's last positions is considered for stabilizing purpose as well, i.e., to ensure that a client is not considered moving if he is not.

Alternatively, we experimented with a particle filter algorithm [11][23]. Results showed that the filter did not perform much better in compensating outliers. Due to a large increase in computational demand we consider this alternative as not feasible for a large scale setup.

B. Sniffer Drones

The work presented in this article emphasizes enhancing WLAN infrastructure with a sensor overlay consisting of of-the-shelf network devices. In related publications these devices are often referred to as sniffers [9][16] or sniffer drones [15]. Our architecture comprises a network of such sniffer drones, forming the core component of the system. In general, sniffers can be denoted as passive components, meaning that they do not emit radio signals themselves since they use an Ethernet backbone to drain their measurement data. Currently, we're running a campus wide sniffer network consisting of custom Linksys WRT610N APs with some modifications to firmware and software.

Basically, these devices can be considered as embedded systems, operating a MIPS32 platform at 466MHz with 8MB RAM and providing two separate WLAN interfaces to support both 2.4GHz and 5GHz radio. A Linux kernel 2.6 [24] allows developing applications at system level. Our sniffing software (implemented in C) uses the low-level packet capturing library *libpcap* [25] that addresses a feature of the WLAN interface driver to collect signal strength measurements, the so called monitoring mode. Monitoring mode describes an alternative mode of operating a WLAN interface (such as the master mode for access point operation or the managed mode for client operation) and is a mandatory feature that facilitates using access points as sensors. This way sniffing the wireless medium in real time is possible, which is crucial to our architecture. First experiments we conducted made use of SNMP (Simple Network Management Protocol) for collecting signal strength measurements, as demonstrated by [26], for instance. A benefit was to be able to abstract from AP hard- and software as long as they supported SNMP. This approach however did not satisfy our demand for conducting real time measurements (cf. Section 2).

As the sniffer drones network is conceived as an overlay to existing WLAN infrastructure, we consequently have to deal with the usage of three non-overlapping channels in the 2.4GHz frequency band (typically the channels 1, 6 and 11 or 2, 7 and 12) - the 5GHz band will not be covered in this article. Hence, we can't assume the transmission channel of a mobile client precisely. Therefore, all possible channels are subsequently iterated while monitoring each prospective channel for a certain period of time. In order to assure collection of sufficient measurements per channel 500 milliseconds of channel dwell time have proven applicable. A completed iteration will further be addressed as a channel hop cycle.

In Section 3A we pointed out that a measurement vector consists of several entries, one for each drone that detects the client's signal at a certain place. To calculate a position these entries are proportionally weighted according to several properties, such as the signal intensity. Hence, the absence of a presumably intensive drone's measurement in this vector can lead to a grave estimation error. If it is not assured that adjacent drones concurrently listen on the same channel it is likely that the measurement vector is incomplete. To avoid this effect we implemented a synchronization mechanism that concurrently triggers restarting of the channel hop cycle at every sniffer drone in the network. The triggering component resides on the server backend, centrally orchestrating the sniffer network by sending a UDP restart broadcast every 60 seconds. In this context, we experimented with an alternative approach to avoid a hop cycle restart every minute. The Precision Time Protocol (PTP), as defined by the relevant IEEE 1588 standard [27], provides clock synchronization accuracy of less than one microsecond. Our embedded hardware platform on the contrary offers a system timer resolution (often referred to as *Jiffy*) of just 10 milliseconds. If we applied a predefined, hard coded hop cycle schedule on each Sniffer Drone along a daemon process updating the system clock via PTP on a daily basis, the maximum drift between the networked Sniffer Drones could be reduced to these 10 milliseconds, in theory. In practice, this has turned out not to work satisfactorily. Due to a varying workload on each separate sensor device correlated to differing radio environments, a measureable internal clock drift within the whole sensor network might appear already after a few minutes, especially between idle sensors and busy ones. Considering the protocol overhead produced by the PTP synchronization, an every minute clock update is not a better choice compared to the UDP restart solution.

During normal sensor operation, each single sniffer continuously reports its measurement data to a collector process within the tracking engine while constantly switching through the channel hop cycle. The sniffer is able to apply packet filters to reduce the subsequent processing load for the engine. System architecture implies that a user has to request a position from a frontend (cf. Section 3C) that operates on a special high port. This port is used as indicator for WLAN traffic to be tracked; traffic on other ports is filtered out. Filter functionality can also be activated remotely by a controller process within the tracking engine, for instance if a certain MAC address has to be blacklisted. As another additional load reduction effort, each sniffer groups measurement data by MAC addresses and averages them before sending.

C. *LBS Framework*

In the previous sections we explained how an infrastructural sensor overlay is used to estimate a user's location. This section deals with how to deliver this information to the user to achieve additional benefit. For these purposes we use the Digital Graffiti system.

Digital Graffiti is a stand-alone framework for location-based services developed in the course of a research project between Siemens Corporate Technology Munich, the University of Linz and the Ars Electronica Futurelab Linz. Conceived as a system to manage and visualize localized information within the context of a mobile user (respectively a mobile device) [28], it has been enhanced with functionality to fulfill the demands for a social network system as well. It comprises a map server, an elaborated user and privileges management concept that additionally handles communication encryption and a messaging component.

Similar to conventional cellular telephony the system uses a distributed provider model for the server-side component where users all over the world can join the provider of their choice in order to take part in the mobile location-based information service. This proven model distributes the load ensuing from (asynchronously) communicating users and guarantees scalability of the service all over the world as each provider only handles a limited number of clients. Information elements (graffiti) are stored in corresponding databases at the providers.

The clients are supposed to be executed on any mobile platform, either as a native application particularly designed for the device or as a web application (utilizing the novel W3C standard and HTML5 for accessing GPS out of a browser). In the context of infrastructure-based indoor tracking, we accentuate the web application, for it complies with our requirements of a bare device without the needs of installing client software. Fig. 3 illustrates the architectural layers of the Digital Graffiti client framework.

The framework utilizes Java for maximizing the variety of potential target platforms (e.g., Symbian, Android). For Java-incompatible systems (such as the Apple iPhone) the framework comprises a Java-based proxy (built upon the same kernel, although without a user interface) which runs on a web server and dynamically transmits data via Ajax to a web browser for display. Thus the service can also be consumed on platforms that are not natively supported, and it abstains from tedious download and installation procedures.

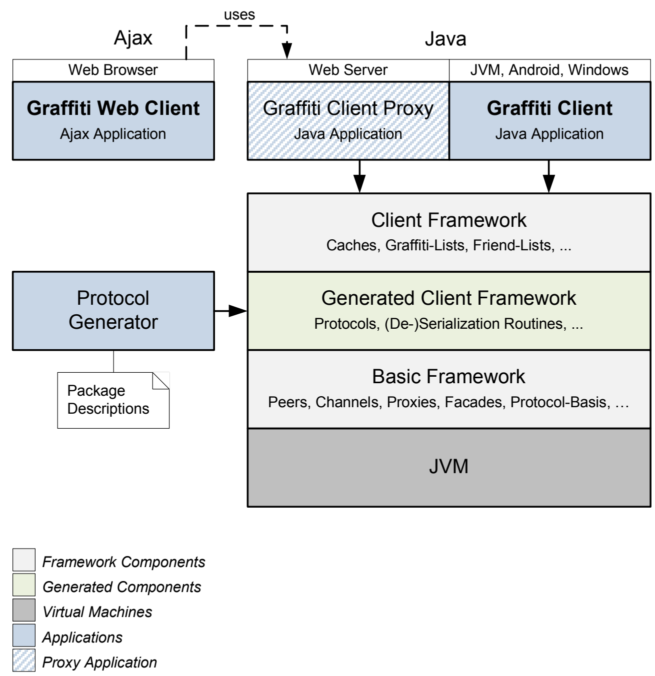

Figure 3. LBS Framework architecture

The architecture envisioned in this article employs the Digital Graffiti framework as third component. The framework acts as frontend for the tracking system, allowing any device equipped with an 802.11b/g/n interface and a web browser to consume location. Once registered and logged in, the user is visualized as an avatar at his exact residing position in front of a map and his geographical position is textually resolved into a human readable address. Digital Graffiti therefore provides a "*spatial coding*" component to map geographical addresses to corresponding names at detailed spatial levels (i.e., buildings, floors and even distinct rooms). Alternatively to indoor WLAN localization the system supports a seamless transition from and to outdoor GPS tracking as well. The position of the user is updated at a near real time frequency (due to the hop cycle length every three seconds; cf. Section 3B). Alongside user's own position, the system also offers to track the position of the user's friends, provided that the respective friend has granted permission. To sustain privacy this permission can be revoked by one click in the user interface.

IV. APPLICATION

The applicability of the infrastructural approach of WLAN tracking using the Digital Graffiti framework as a user frontend is being demonstrated within a campus-wide live system at the University of Linz, called the Smart Information Campus (SIC) Project [29]. It covers an area of about 800x300 meters campus space (cf. Fig. 4) including 15 multistory buildings varying from 1 to 11 stories equipped with the proposed sniffer technology, which in total results in 320 access points (Cisco Aironet 1250) and the same amount of co-located sniffer drones (Linksys WRT610N). Co-locating the sensor drones has been decided due to practical reasons such as the availability of power and Ethernet connections, but is still a matter of discussion [13] in terms of localization accuracy.

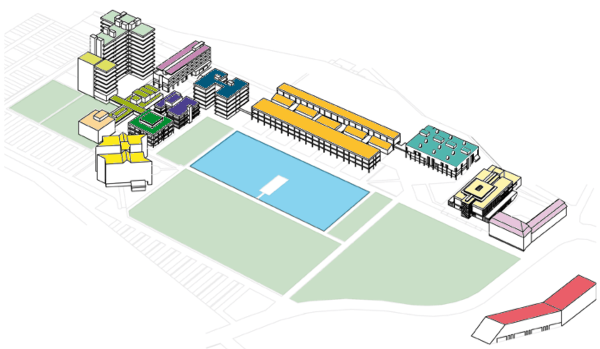

Figure 4. Overview of the building complex at the JKU campus

Table 2 gives an overview of the complete indoor areal of the campus site, showing the abbreviations of each building and a description of the respective area correlated to deployment numbers of the tracking infrastructure (amount of deployed sensor nodes, total number of calibrated candidate points, costs measured in working hours). Due to the system rollout procedure, some details are still to be determined at the time of this writing. As on every typical university campus, these buildings host office spaces, laboratories, meeting rooms and lecture halls, varying in room dimensions, furnishing and technical equipment. The campus has been built starting in the year 1964 and is continuously expanding since then. Consequently, the structural design and building characteristics reflect the ideology of the decade in which each respective building has been realized. In terms of indoor tracking, this has two important implications. For one, the amount of sniffer drones to be deployed varies correlated to wall construction substances since each substance shows its own attenuation characteristics. To assure an appropriate coverage, a traditional radio site survey gives a first guide number for planning the deployment of tracking infrastructure. The other implication for the campus-wide tracking setup is the divergence of tracking accuracy with respect to wall attenuation and the respective room layout. In Section 5 the tracking accuracy within three exemplary buildings on the campus are compared to highlight the effects of differing building design and structure on infrastructural tracking.

TABLE II. CAMPUS COVERAGE STATISTICS

Building	Floors	Area (m^2)	Nodes	Candidate Points	Effort (h)
MZ	5	10.500	26	517	39
KE	3	19.293	50	818	47
HF	5	6.650	26	424	29
ScP	7	64.232	32	978	40
SL	3	3.255	8	172	7
UC	4	6.600	11	275	15
BI	3	6.720	12	386	22
HP	4	2.600	8	213	10
ME	1	450	3		
HT	3	2.925	3		
BA	4	3.840	10		
PH	4	3.844			
JU	5	11.970			
Turm	11	18.810			
KG	7	4.256			
		165.945	***189***	***3.783***	***209***

The Smart Information Campus system offers several means of consuming the provided location-based services on a broad variety of mobile platforms. To this end, we focus on four basic directions of implementation. We offer a native Windows desktop application for laptop and netbook service consumption based on *Microsoft WPF*. Java compatible mobile phones (such as Symbian based phones, etc.) are supported by the J2ME client implementation using the Kuix UI framework [30]. For Android based smart phones we additionally implemented a touch screen application utilizing the sophisticated features this modern platform is offering. To provide access for the remaining mobile platforms (e.g., Apple iPhone and iPad, Blackberry, Linux, etc.) we implemented a web application client based on AJAX. Fig. 5 and 6 show screenshots of all four client variants.

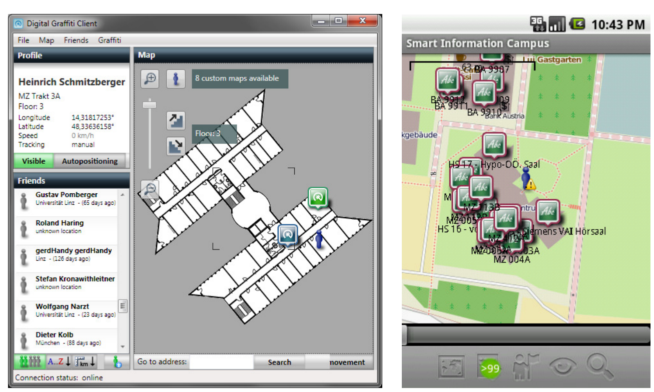

Figure 5. Client Snapshots (Desktop and Android)

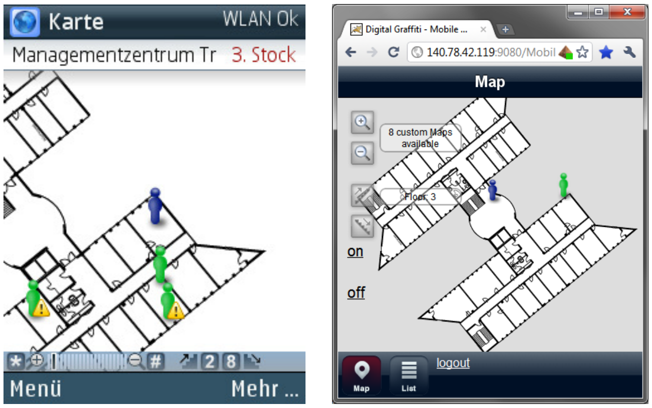

Figure 6. Client Snapshots (J2ME and web application)

The system has started its beta phase in January 2010 and is since then publically available 24 hours a day, 7 days a week [28]. Students as well as lectors, administrative staff and guests (a total number of about 16.000 people) are able to track themselves and their selected friends (Fig. 7 right presents a snapshot of the SIC application revealing our own position by a blue avatar and those of the friends by green ones) and perceive and post location-based information at the university campus due to their detected position within buildings and outside. For instance, students may find their way to their lecture halls displaying the current lecture type and times for these rooms; the event management announces upcoming activities or presentations to the users related to their location; teachers are able to ad-hoc exchange documents with students just because of their geographical attendance in a lecture room; etc. Generally, the SIC is supposed to enhance social, scientific and organizational networking within the campus, e.g., enabling the creation of communities and providing a practical research platform for location-based service issues.

As of this writing, the system is at the end of phase 2 of 3 rollout phases, providing indoor localization for 8 of 15 buildings, already covering about 72% of the whole indoor area. Due to their age these buildings have dissimilar construction characteristics that imply different radio propagation properties. To provide accurate localization in such an environment, the application of an architecture based on an *a priori* off-line training process has proven to be feasible. Consequently, a radio profile was taken at a larger number of spots in every building (cf. Table 2).

V. RESULTS

The system setup is under constant assessment by 200 selected beta users at the university, with an average number of 33 concurrent position requests per second during a regular working day. In order to provide a satisfying LBS experience our tracking system is set up to assure the correct symbolic reference rather than the exact position. For most places on the campus this does not make a perceivable difference since the position has room level accuracy. If localized within one of the bigger lecture halls though, our approach might result in a bigger estimation error.

Fig. 7 shows a live snapshot of the J2ME client running on a Nokia E52 on an office floor. The position is indicated by the blue avatar in the center of the map. The symbolic name of the actual position appears at the top of the display, indicating the name of the building along with the floor. Since located in a corridor, no office name is displayed.

Figure 7. Tracking snapshot at office floor on J2ME client

The accuracy of the system primarily depends on the density of sniffer drones, their positions and the buildings' characteristics. Consequently, we encounter different localization precision at different sites. The following Figures 8, 9 and 10 depict exemplarily captured tracks of a Nokia E52 client walking along a predetermined path (indicated by the red line) in 3 different buildings on the campus of *Johannes Kepler University*. All position measurements (indicated by the red crosses in the figures) were taken weekdays around midday. As the data was collected during the academic semester, the offices and hall ways were averagely crowded with students and university staff. The effects of the time of day on an indoor localization system have already been studied by Tao et al. in [31]. Their studies showed that the signal strength histograms of measurements vary noticeably as a function of the time of day, with significantly more noise when more people are in the building. Consequently, our sample tracks were captured at the same time of day. However, the bias provoked by an arbitrary amount of people interfering with signal measurements is not compensated in our framework yet. Thus, we have to keep in mind that our test cases are compared under slightly divergent environmental conditions. A more detailed investigation on effects derived from building characteristics and people-depending signal strength variations has been presented in [32] along with an approach of compensating for consequential positioning errors.

Fig. 8 shows the first test track recorded on the ground floor of the *Kepler Building* (KE). It comprises a base area of 11782m^2 on 3 stories and is one of the oldest buildings on the campus, consisting of several lecture halls, libraries and public meeting places. Since its wall structure is thick and radio absorbing (concrete and brick walls) it is equipped with 50 Sniffer Drones to cover the whole building (cf. Table 2). The actual track through the building started and ended at the lower left quadrant of the figure and took 4:11 minutes for a distance of 328m, using a client transmission interval of 1 second. It comprises 80 position updates, all of which correctly located at the ground floor.

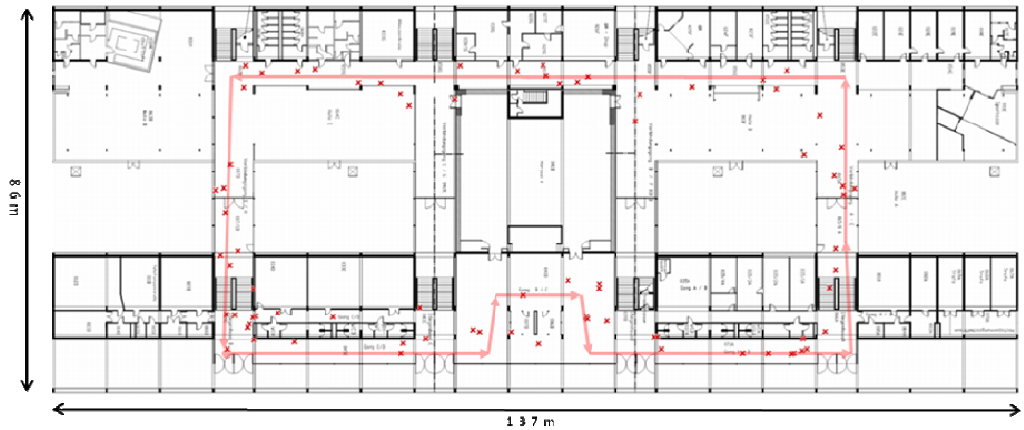

Figure 8. Track through auditorium building (KE)

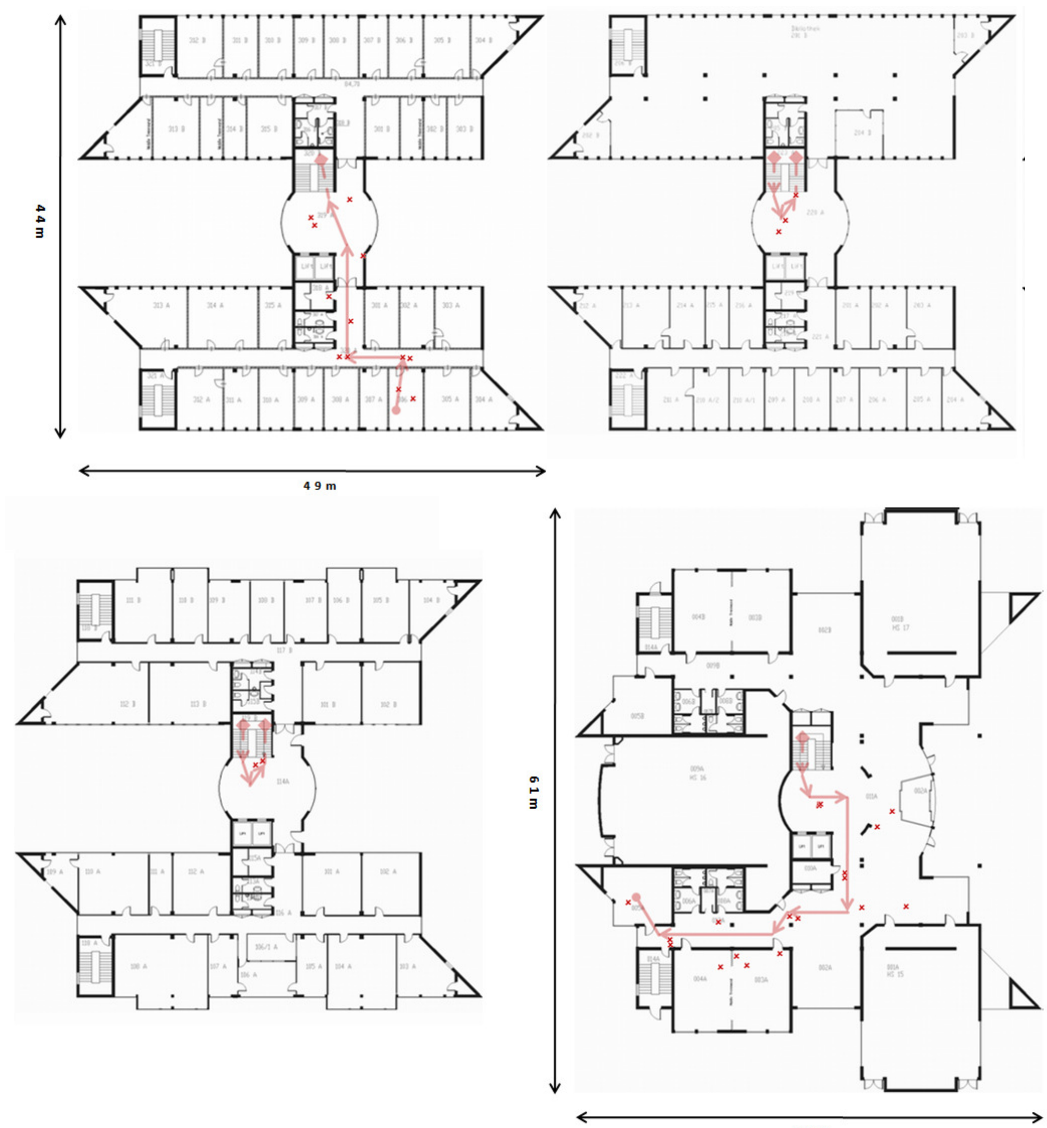

Figure 9. Track through office building (MZ)

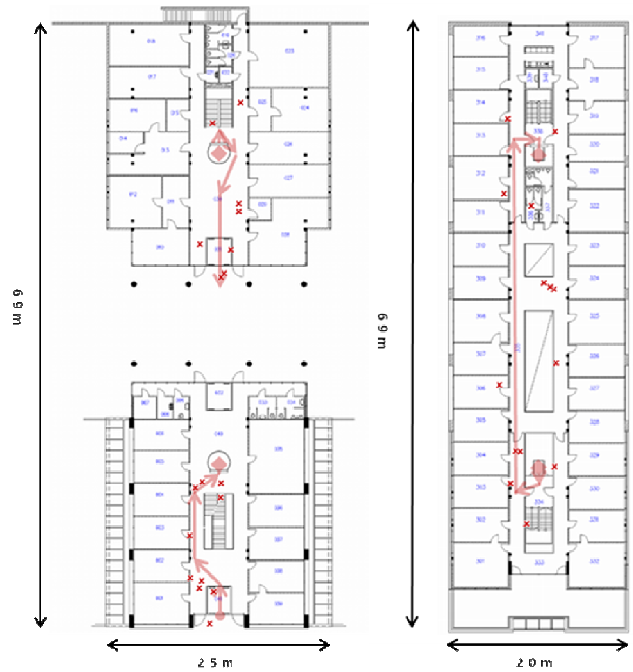

Figure 10. Track through laboratory building (HF)

In Fig. 9, the track is captured within the *Management Zentrum* (MZ), a typical office building comprising meeting rooms and smaller lecture rooms as well. It has been built in 1990 using light-frame construction. The small circle in the upper left picture marks the starting point in an office at the third floor. The client then descends to the ground floor using the stairway and passing the second and the first floor (cf. upper right and lower left picture). Each floor transition has been estimated correctly. The track finally ends in a small lecture room indicated by another red circle in the lower left quadrant of the lower right picture. It took 3:28 minutes for a distance of 126m, featuring 37 position updates at the same transmission interval as above but at a lower walking pace.

The third track (cf. Fig. 10) was measured inside the *Hochschulfond Building* (HF), which has been built in the year 2003. Because it is mainly dedicated to practical research, it houses technical laboratories, offices and storage spaces. The constructive form of the building can be characterized as modern architecture, emphasizing glass as building substance and a large galleria section spanning from the first to the third floor. The track started at the south entrance (at the bottom of the left picture) on the ground floor. The diamond marks the entering of an elevator that ascended to the third floor (right picture). The track led alongside the galleria towards the north elevator. After descending again, the track ended on the ground floor (in the middle of the picture on the left). Since the elevator is surrounded by a glass construction, the tracking engine was able to estimate each floor and position correctly even within the elevator. The track was finished after 3:56 minutes, even though the covered distance was only 96m including 33 position updates. The long duration was mainly caused by elevator waiting times. Transmission interval and walking pace correspond to track two.

TABLE III. ERROR DISTANCE STATISTICS

	KE	MZ	HF
Average	4.11 m	2.40 m	2.90 m
Std. deviation	2.30 m	1.38 m	1.69 m
25th percentile	2.33 m	1.30 m	1.75 m
Median	4.08 m	2.15 m	2.47 m
75th percentile	5.21 m	3.38 m	3.53 m
90th percentile	7.59 m	4.14 m	5.07 m
Maximum error	10.09 m	6.47 m	7.13 m

Detailed results of these three test tracks based on the error distance are summarized in Table 3. Fig. 11 depicts a comparison chart that clearly highlights the differences in accuracy referring to the respective building. Track one as well as Track three both showed two phases without a position update indicating that the new estimate would have been an outlier. For the first track, this can be explained by a sudden appearance of a larger group of people since these positions were near a lecture hall (Fig. 8, upper right) and a cafeteria (Fig. 8, upper left). In the third track, this phenomenon is caused by the open space from the galleria (Fig. 10, in the middle of the right picture). Due to the gap in the middle section ranging from the first up to the third floor, the signal strength fingerprints tend to be very similar on both sides of the galleria.

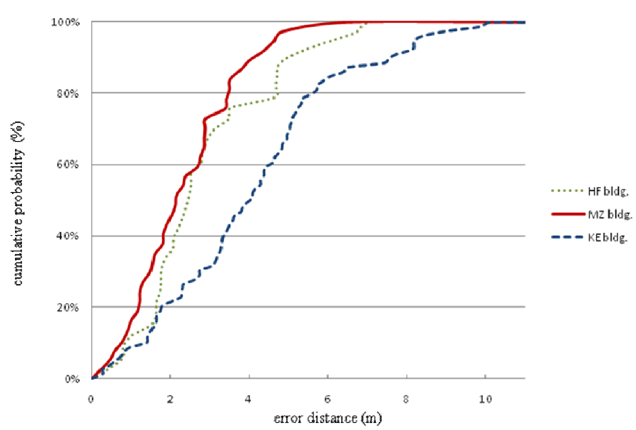

Figure 11. Comparison of the tracking accuracy in exemplary buildings

Concluding from these results, the tracking framework presented in this article shows a clear tendency towards better performance and accuracy within younger buildings. As reported in the context of client-based indoor localization setups [32], this can be explained with building characteristics such as radio absorbing wall substances or large spaces that provoke indistinguishable fingerprints. Out of all types of campus facilities, office floors turn out to provide the most promising environment for indoor tracking because of their room arrangement diversity that results in unambiguous fingerprints.

In terms of overall accuracy, the presented system still leaves room for improvement (compensating mechanisms for dynamic attenuation provoked by people for instance). Whereas newer contributions report mean estimation errors under 2m [18][22], our setup reveals values from 2.4m up to 4.11m (depending on the building properties). However, comparable tracking systems are commonly investigated under laboratory conditions (i.e., small areas of 500-1000m^2, tailored radio coverage, homogeneous building structures, single-story localization, etc.), not considering real-life scenarios in a continuous operation mode, as well as economic factors of comprehensive infrastructure deployment.

VI. CONCLUSION AND FUTURE WORK

In this article, we presented a system capable of concurrently tracking numerous clients with no special client-side hardware prerequisites within a large-scale indoor setup. The proposed architecture has been deployed in a real-life setup at the University of Linz and is part of a project providing campus-wide LBS to an academic audience, outdoor as well as indoor. As the campus comprises diverse buildings of manifold types of architecture, the tracking system has to cope with a variety of different radio distribution characteristics. By comparing the performance of our system in three exemplary test cases that reflect the three most diverging radio environments on the campus site, we pointed out the feasibility of our architecture. The results clearly indicate that system accuracy benefits from the more modern style of building construction on the one hand, as well from an office floor layout. Overall, the system performance can compete with other related systems, even under real-life conditions and on a campus-wide scale.

We reported detailed deployment numbers and costs (in working hours) that were invested in our large-scale setup. In this regard, a clear disadvantage still exists in the form of the tedious training process (approximately 40 working hours for the full sensor coverage of an average building) that precedes the life system. Therefore, we are exploring alternatives to our two-phased fingerprinting architecture. We envision a benefit from making use of beam-forming technology that is one of the most promising features of the new 802.11n (still draft) standard for WLAN localization. Unfortunately, the driver used by the sniffer drones does not support obtaining lower level antenna reception information yet. Further experiments need to be conducted relating to the 802.11a standard in order to cope with the greater amount of alternative radio channels, that common 5GHz WLAN networks make use of (at JK University we're using 12) and that imply a greater hop cycle length in our current system. Since some WLAN NIC drivers capable of both 2,4GHz and 5GHz communication tend to favor the 5GHz band, the respective NIC drivers have to be configured explicitly to use 2,4GHz first. In the future we hope to avoid this user inconvenience.

The compensation of signal strength fluctuation effects provoked by people passing by is another important issue we are investigating at the time of this writing. Our focus lies on achieving a solution for our network consisting of 320 Sniffer Drones that solely runs decentralized, i.e., exclusively on the sensor platform to avoid a potential bottleneck at the backend. To this end, a mechanism to reliably detect spontaneously emerging crowds of people has to be realized that relies on pure radio environment fluctuations and does not depend on emitters carried by the people within the crowd [22]. This way, an even more accurate and temporarily stable tracking experience could be provided.

REFERENCES

[1] H. Schmitzberger and W. Narzt, "Leveraging wlan infrastructure for large-scale indoor tracking," in *Wireless and Mobile Communications (ICWMC), 2010 6th International Conference on*, 2010, pp. 250 – 255.

[2] M. Hazas, J. Scott, and J. Krumm, "Location-aware computing comes of age," *Computer*, vol. 37, no. 2, pp. 95–97, 2004.

[3] "Skyhook wireless." [Online]. Available: http://www.skyhookwireless.com/ [last accessed on June, 28. 2011].

[4] "Ekahau rtls." [Online]. Available: http://www.ekahau.com/-products/real-time-location-system/overview.html [last accessed on June, 28. 2011].

[5] "Aeroscout." [Online]. Available: http://www.aeroscout.com/ [last accessed on June, 28. 2011].

[6] T. Sohn, W. G. Griswold, J. Scott, A. LaMarca, Y. Chawathe, I. Smith, and M. Chen, "Experiences with place lab: an open source toolkit for location-aware computing," in *ICSE '06: Proceedings of the 28th international conference on Software engineering*. New York, NY, USA: ACM, 2006, pp. 462–471.

[7] J. Wierenga and P. Komisarczuk, "Simple: developing a lbs positioning solution," in *MUM '05: Proceedings of the 4th international conference on Mobile and ubiquitous multimedia*. New York, NY, USA: ACM, 2005, pp. 48–55.

[8] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services*. New York, NY, USA: ACM, 2005, pp. 205–218.

[9] P. Krishnan, A. S. Krishnakumar, W.-H. Ju, C. Mallows, and S. Ganu, "A system for lease: Location estimation assisted by stationery emitters for indoor rf wireless networks," in *Proceedings of IEEE Infocom 2004*, 2004.

[10] M. B. Kj, "A taxonomy for radio location fingerprinting," in *Proceedings of the Third International Symposium on Location and Context Awareness*, 2007.

[11] J. Krumm, Ed., *Ubiquitous Computing Fundamentals*. CRC Press, 2010.

[12] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM*, 2000, pp. 775–784.

[13] S. Ganu, A. S. Krishnakumar, and P. Krishnan, "Infrastructure-based location estimation in wlan networks," in *Proceedings of the IEEE WCNC 2004*, 2004.

[14] M. Youssef, A. Youssef, C. Rieger, U. Shankar, and A. Agrawala, "Pinpoint: An asynchronous time-based location determination system," in *MobiSys '06: Proceedings of the 4th international conference on Mobile systems, applications and services*. New York, NY, USA: ACM, 2006, pp. 165–176.

[15] J. H. H. Lim, L. Kung and H. Luo, "Zero-configuration, robust indoor localization: Theory and experimentation," in *INFOCOM*, 2006.

[16] L. F. M. de Moraes and B. A. A. Nunes, "Calibration-free wlan location system based on dynamic mapping of signal strength," in

MobiWac '06: Proceedings of the 4th ACM international workshop on Mobility management and wireless access. New York, NY, USA: ACM, 2006, pp. 92–99.

[17] "Cisco wireless location appliance." [Online]. Available: http://www.cisco.com/en/US/products/ps6386/ [last accessed on June, 28. 2011].

[18] J. Indulska and P. Sutton, "Location management in pervasive systems," in *ACSW Frontiers '03: Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003*. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2003, pp. 143–151.

[19] J. Yim, S. Jeong, K. Gwon, and J. Joo, "Improvement of kalman filters for wlan based indoor tracking," *Expert Systems with Applications*, vol. In Press, Corrected Proof, pp. –, 2009.

[20] n.-B. Galo Nu and J. M. Páz-Borrallo, "A new location estimation system for wireless networks based on linear discriminant functions and hidden markov models," *EURASIP J. Appl. Signal Process.*, vol. 2006, no. 1, pp. 159–159, 2006.

[21] N. Swangmuang and P. Krishnamurthy, "Location fingerprint analyses toward efficient indoor positioning," *Pervasive Computing and Communications, IEEE International Conference on*, vol. 0, pp. 100–109, 2008.

[22] M. Youssef, M. Mah, and A. Agrawala, "Challenges: device-free passive localization for wireless environments," in *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2007, pp. 222–229.

[23] K. Vandikas, A. Katranidou, L. Kriara, H. Baltzakis, T. Papakonstantinou, and M. Papadopouli, "Empirical-based analysis of a cooperative location-sensing system," in *Autonomics '07: Proceedings of the 1st international conference on Autonomic computing and communication systems*. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, pp. 1–11.

[24] "Dd-wrt." [Online]. Available: http://www.dd-wrt.com/ [last accessed on June, 28. 2011].

[25] "Libpcap." [Online]. Available: http://www.tcpdump.org/ [last accessed on June, 28. 2011].

[26] G. E. Violettas, T. L. Theodorou, and C. K. Georgiadis, "Netargus: An snmp monitor & wi-fi positioning, 3-tier application suite," in *ICWMC '09: Proceedings of the 2009 Fifth International Conference on Wireless and Mobile Communications*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 346–351.

[27] "Ieee standard for a precision clock synchronization protocol for networked measurement and control systems," *IEC 61588:2009(E)*, pp. C1 –274, 2 2009.

[28] W. Narzt, G. Pomberger, A. Ferscha, D. Kolb, R. Müller, H. Hörtner, and R. Haring, "Addressing concepts for mobile location-based information services," in *UAHCI'07: Proceedings of the 4th international conference on Universal access in human-computer interaction*. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 507–516.

[29] "The smart information campus," 2011. [Online]. Available: http://sic.jku.at/ [last accessed on June, 28. 2011].

[30] "The kuix user interface framework," 2011. [Online]. Available: http://www.kalmeo.org/projects/kuix [last accessed on June, 28. 2011].

[31] P. Tao, A. Rudys, A. M. Ladd, and D. S. Wallach, "Wireless lan location-sensing for security applications," in *Proceedings of the 2nd ACM workshop on Wireless security*, ser. WiSe '03. New York, NY, USA: ACM, 2003, pp. 11–20.

[32] K. Kaemarungsi, "Design of indoor positioning systems based on location fingerprinting technique," Ph.D. dissertation, University of Pittsburgh, 2005.