

PlaceComm: A framework for context-aware applications in place-based virtual communities

Tuan Nguyen*, Seng W. Loke, Torab Torabi and Hongen Lu

Department of Computer Science & Computer Engineering, La Trobe University, VIC, 3086, Australia

E-mail: {t.nguyen,s.loke,t.torabi,h.lu}@latrobe.edu.au

Abstract. There is increasing interest in context-aware applications, ranging from traffic routing, location-based services, friend finding, context-aware messaging, virtual tour guides, to physical world search. We make three observations: (i) many context-aware applications use knowledge about the individuals (and groups of people), places and objects within a particular place, that is, they are place-based, (ii) such knowledge about a place can be shared by different applications and reused, without each application having the same knowledge reengineered (there is at least overlap among such knowledge across different applications), and (iii) applications can not only be consumers of knowledge but also be contributors of knowledge about the place. We propose the PlaceComm framework for building such context-aware applications with “place” as a key abstraction, where place knowledge can be collected and shared. The knowledge is structured using an ontology called the Place-Based Virtual Community ontology. Our framework utilizes a multiagent architecture for modularity, separation of concerns and extensibility. We also describe three applications built using our approach: PlaceGuide, PlaceAware and PlaceSense.

Keywords: Ontology, place-based computing, place-based virtual communities

1. Introduction: The concept of the place-based virtual community

A virtual community typically transcends geographical boundaries. However, recent technologies with limited range connectivity such as Wi-Fi and Bluetooth have led to geographically based connectivity, on which virtual communities can be built upon, which are based on the local vicinity. Such Wi-Fi hotspots are increasingly rapidly today. A place-based virtual community (or PBVC, for short) comprises the collection of people, objects, buildings, devices, services, history of movements, activities and interactions at a place, basically a virtual community rooted at a physical place, or a digital augmentation of an area. A place is not just a location but contains interactions among people and experiences that make the place meaning-

ful, essentially, making a location a place. In crowded areas, such as urban areas, typically, one could be in multiple place-based virtual communities at the same time [10], that is, one is effectively located within a “stack” of communities, and moving in and out of such communities happens as a person moves around.

A place can be a stadium, a shopping mall, a street (e.g., Orchard Road in Singapore or the Bourke Street Mall in Melbourne, Australia), a neighbourhood, the city centre (e.g., Central Business District of a city), a museum, a park, a university campus, a town, a collection of buildings, and so on. Each place can be associated with at least one PBVC which has information about the physical place and community of people living in or being at that place. In general, the boundaries around a place may be specified by an n -sided polygon, where each point is represented by a set of n GPS coordinates, or a circle with a given GPS coordinates for a point and a radius. Note that our notion of PBVC does not aim to replace but can complement ex-

*Corresponding author.

isting virtual communities that transcend boundaries. More generally, a place can be a union of polygons (i.e., a set of disconnected areas). For example, a place labelled as “myCommonPlaces” is a union of the university campus, my home, my parent’s home and two shopping areas frequented.

Our vision is that each PBVC contains knowledge about a place, and the community at that place, a shared repository which we call a *place knowledge base* (or place KB, for short). Each person, via his or her device(s) might use such knowledge (or such knowledge used by a context-aware application on the device) or might contribute such knowledge back to the place KB. Different applications running on users’ mobile devices can consume and/or contribute knowledge to the place KB. A place KB may contain knowledge of a more static nature, such as locations of buildings, or knowledge of a more dynamic nature, such as the current visitors at the place. It can also contain knowledge about physical features of a community, including restaurants and shops, available physical and digital services (e.g., electronic directory or virtual tour guide for a place), where people are assembling at different points in time within the place, as well as, with permission, even locations of particular people, a representation of relationships between people, and people and objects (e.g., ownership). Even a history of how people move through the place can be stored.

Since the effective area of a PBVC is bounded by the place, there could be multiple overlapping PBVCs, each with their own KBs (e.g., the PBVC of a cafe is located within the PBVC of a street on which the cafe is located, and a city’s PBVC contains the PBVC of a university within the city). A person (with a device) may be standing in more than one PBVC at a time, and so, can opt-in to join one or more PBVCs. A PBVC may provide different context-aware services to the people. For example, an application could be a guide for the blind, where as a person traverses through different parts of a place (e.g., buildings, bus stops, and sites), verbal communication is provided to help the blind gain a better understanding of the place.

Consider a range of examples of context-aware applications applicable to a place, such as a virtual guide for the blind, a virtual tourist guide, a shopping-recommender system, a crowd-seeker, an ATM Bank Teller finder (with real-time reports of lengths of queues in front of them), and a navigation application. Some of these applications may share knowledge about a place, benefiting from a knowledge base

about physical features and objects at a place. Some of these features may vary over time, depending on human movement at the place or new objects and information being introduced.

2. Background and related work

Indeed, there has been tremendous work on community computing,¹ where software is used to help people know each other and share preferences and also support interaction at events. In previous work, a key idea is the notion of digital cities, with a three layered architecture: a) the information layer integrating sensor information and Web information related to the city, b) the interface layer providing 2D and 3D graphical views of the city, and c) the interaction layer assisting social interactions among people in the city. The Nexus project² is working on context models which can be placed in a shared space for a wide variety of applications. Our work is in a similar spirit, of super-imposing a digital abstraction over a physical place, where a knowledge base about the place can be shared by different applications. However, we focus on ontological-based representation of entities within a community and have the purpose of supporting developers in building myriad context-aware applications via shared knowledge about a locality.

Site-specific services accessed via smartphones were considered in [19], but they do not consider the notion of a community and a place KB. Also, our work differs from earlier context-aware middleware and toolkits (e.g., [6]) in that these toolkits and middleware generally do not support reuse of a common knowledge base across different applications or allow applications to contribute knowledge to be shared via a common KB.

Moreover, there has been numerous work on context-aware frameworks [6], context ontologies [4,5] and context modelling and reasoning [3], but our work suggests that the notion of “place” should be central, and knowledge about a place can be built even without committing to any application initially. Other context ontologies can indeed complement our work. Also, a key difference in building context-aware applications for PBVCs as opposed to general context-aware systems is the need to capture people and their relation-

¹<http://www.ai.soc.i.kyoto-u.ac.jp/~ishida/community-computing.htm>

²<http://www.nexus.uni-stuttgart.de/index.en.html>

ships, device and their presence and usage, people's interactions with and utilization of a place, and services that are specific to social and place characteristics.

In [16], the role of place in virtual communities is discussed where a sense of place is due to people's activities and interactions at a location, and socio-cultural bonds within a community leads to the notion of a place. Indeed, our proposal majors on the notion of place. Because a place is what it is due to users' activities, behaviours, and interactions, and sensors can detect in part these aspects, we propose the idea of keeping real-time sensor-acquired knowledge of a place to be used in context-aware applications. PBVCs can be either member-initiated or organization-sponsored, though there is a need to have our PlaceComm framework (or similar) installed and running, after which members and organizations can choose to add to the place's knowledge base.

Other urban sensing projects [8] utilize the mobile device to collect information about the environment (from location to pollution information), and analyse such data to understand urban behaviour and environments. Our work also considers individuals using applications to collect data, but such data in our framework is to be processed and added to a knowledge base about the place. We believe a novelty of our proposal is building historical knowledge about a place over hours or even years, whether knowledge of a more static nature or dynamic real-time knowledge acquired via sensors, and then making this knowledge available to be used by context-aware applications. We contend that an ontology should be used as a schema for this knowledge. This ontology should be publically available and consistent for different places.

The CenceMe application [9] makes use of sensors that come with today's mobile phones for detecting user presence and activities (including walking, standing, etc.). Sensor data is processed on the device itself as well as uploaded to be processed on backend servers. Data classification and processing on the backend servers allow activities involving social context to be determined, including whether a person is in a conversation (from audio sound samples), the determination of nearby CenceMe buddies, whether a person is in a meeting, alone or in a party. Our approach differs from CenceMe in that we consider the use of sensors not only on the mobile but also embedded in the environment, in particular, situated at a place, and that we make use of an ontology to represent core knowledge of a place. Such an ontology acts as a schema for queries and allows different places to store similar

kinds of knowledge, thereby supporting uniform expectations for users and applications.

3. The PlaceComm framework

The system we have built to support context-aware applications is the PlaceComm Framework. The layered architecture of the system is given in Fig. 1. Applications in the application layer make use of services enabled by agents in the agent community layer. Services in the community layer include services for leaving and joining the community, communicating with other people in the community, managing connections with mobile clients, and monitoring services for obtaining information about the community. Web services are also supported as well as agent protocols. PlaceComm includes different types of agents such as the PBVC Community Agent, the Place Agent, and agents to manage context and person information. The PBVC Community Agent acts like a concierge for participants who join the PBVC, and have access to the place KB. The Place Agent is an agent that, based on the user's current location (e.g., GPS coordinates), maps raw positioning information to meaningful place information. Context information is gathered and processed via sensors³ in the Context Processing Layer and Context Gathering Layer.

The place KB effectively stores what we call context information (information about entities within the community, e.g., locations, etc., which are time-stamped) and also the more static information about the community (which might be entered at development time). The place KB is accessed via agents in the agent community layer. In our prototype, we use the Sesame API and Google Maps (if required) in managing information, and accessing maps. PlaceComm has been prototyped using a multiagent architecture with the JADE platform [1]. In the previous work, we implemented the protocols for members joining and leaving a PBVC [10].

4. The PBVC ontology

A key component of the PlaceComm framework is the PBVC core ontology summarized in Fig. 2. The key terminological concepts are boxes and arrows

³We take a broad definition of sensor as any device that can provide useful context information.

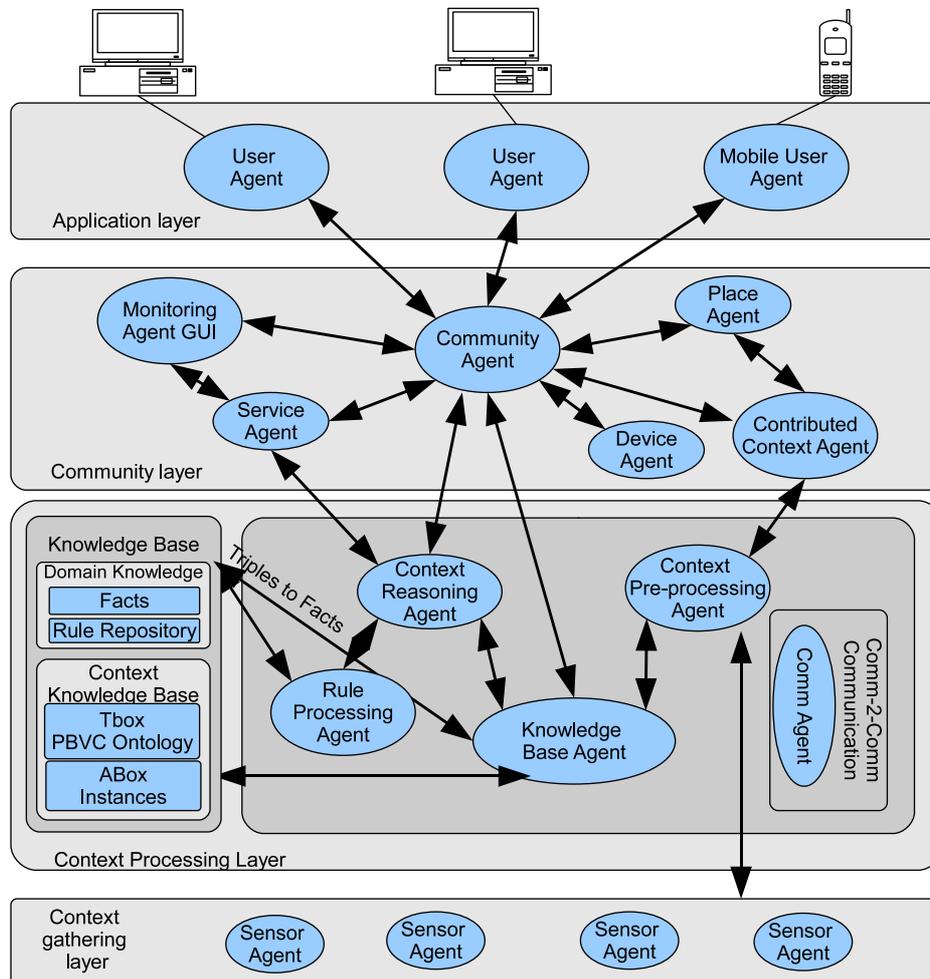


Fig. 1. Conceptual architecture of the PlaceComm framework.

indicate relationships among the concepts. We have used OWL to represent our ontology. The ontology forms the core schema for the place KB. The PBVC core ontology includes six main concepts: Agent, Device, Context, Place, Service and Place-Based Virtual Community. The Agent concept is imported from the Friend of a Friend ontology.⁴ The Device ontology is inherited from the SOUPA ontology.⁵

Time factor is taken into account in our system. Whenever an instance is created, the time-stamp is automatically generated and added to the knowledge base. Time-stamping helps to resolve inconsistencies, e.g., the location of a person varies at

different times. Time-stamps also allow historical queries.

The ontology standardizes the set of agents and basic services for the PBVC, and includes (not shown in the figure) definitions of a physical place, generic places, and personal places. Each place is defined by a name, a tag, or GPS coordinates, and through this correspondence, allows place-aware or location-aware queries. For example, when a “Where am I?” question is posed to the KB, the Place Agent returns a meaningful place for the user, using the user’s most recent location (within a validation period, or the location is considered unknown if the information is too old). Moreover, different types of places such as “home”, “office”, “shopping mall”, “university campus”, and “park” are in an “is-a” (or subclass) relationship with the concept “place”. These typology of places is still

⁴<http://xmlns.com/foaf/spec/>

⁵<http://ebiquity.umbc.edu/paper/html/id/165/The-SOUPA-Ontology-for-Pervasive-Computing>

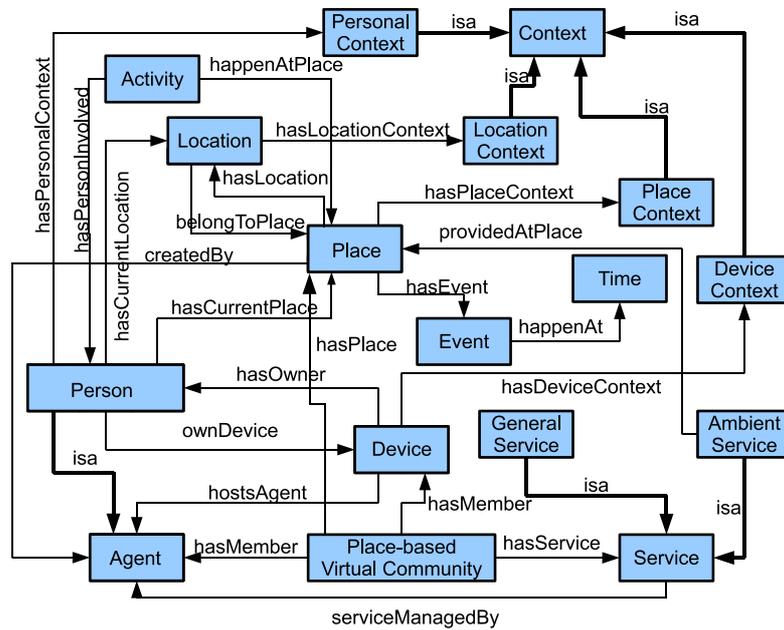


Fig. 2. The PBVC core ontology.

abstract since they do not specify a specific address as a home, office or shopping mall; instead, instances of these classes would have the specific addresses and details. One can consider place-specific services and place-specific extensions to the PBVC core ontology, and also categories of places. For example, all university campuses might have the same PBVC campus ontology; all cafes might have the same PBVC cafe ontology, which is different from the PBVC campus ontology, but the PBVC cafe and PBVC campus ontologies have the common PBVC core ontology, and similarly for shopping centers.

Many ontologies have been developed [20] but we tailor our ontology to specify the minimal information the place-based virtual community KB should contain.

A key novel idea of our work is that knowledge of a place can be collected, at first independently of applications, but later, parts of which can be used in multiple context-aware applications. The knowledge we collect are of a more static nature such as buildings, geographical features or fixed physical features, or of a more dynamic real-time knowledge such as who is at the place at the time, what devices are present currently, and what events are happening. With time-stamping of collected data (e.g., ranging from readings of a vibration sensor installed on an outside wall to, as in our prototype, time-stamped Bluetooth device ad-

resses discovered at an area), a history of knowledge about a place can be built up, perhaps over years. Such knowledge can then be queried using a language such as SPARQL, be used with a set of rules in rule-based reasoning, or be mined for patterns using data mining techniques. The ontology provides a schema for the knowledge of the place, and can be extended for particular places or PBVCs.

Activities. We have also considered an extension of the PBVC ontology to include activities at a place. We introduce the concept of Activity as shown in Fig. 3 below. This concept is linked to the concept Place in Fig. 2 through a property `happenAt`, modelling an activity that happens at a place.

The Place-Based Activity Ontology (PAO) contains four subclasses: Social Activity, Shopping Activity, Transport Activity and Room Activity. The SocialActivity is a place-based activity that happens when more than two people are gathered at a place and interact. The Place-Based Activity has two important properties: `hasPersonInvolved` and `happenAt`. The Social Activity is a subclass of Place-Based Activity with another characteristic being that people must interact with each other.

Example of room monitoring for activities. A PC with Bluetooth dongle scans the room every 20 sec-

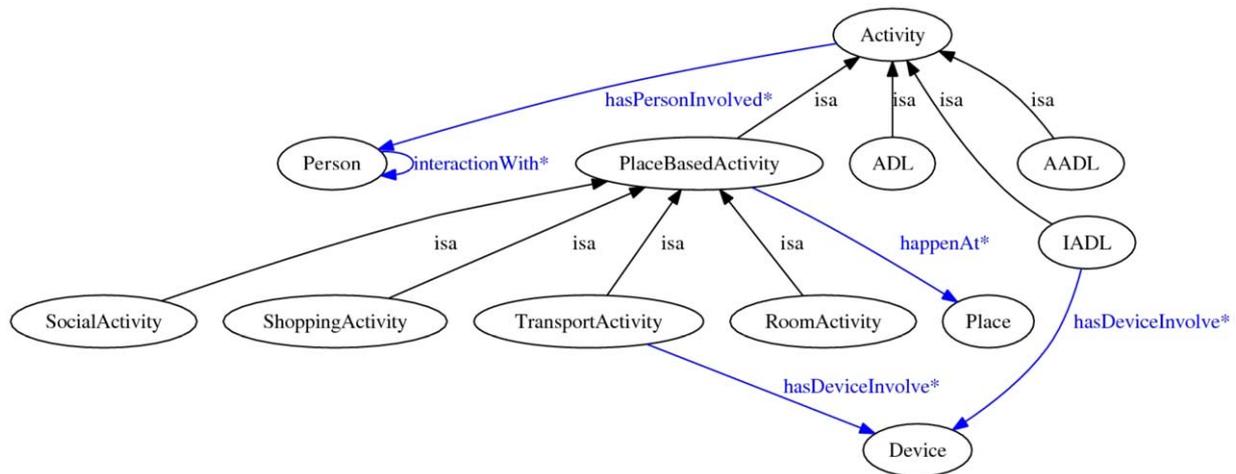


Fig. 3. Place-Based Activity Ontology (PAO).

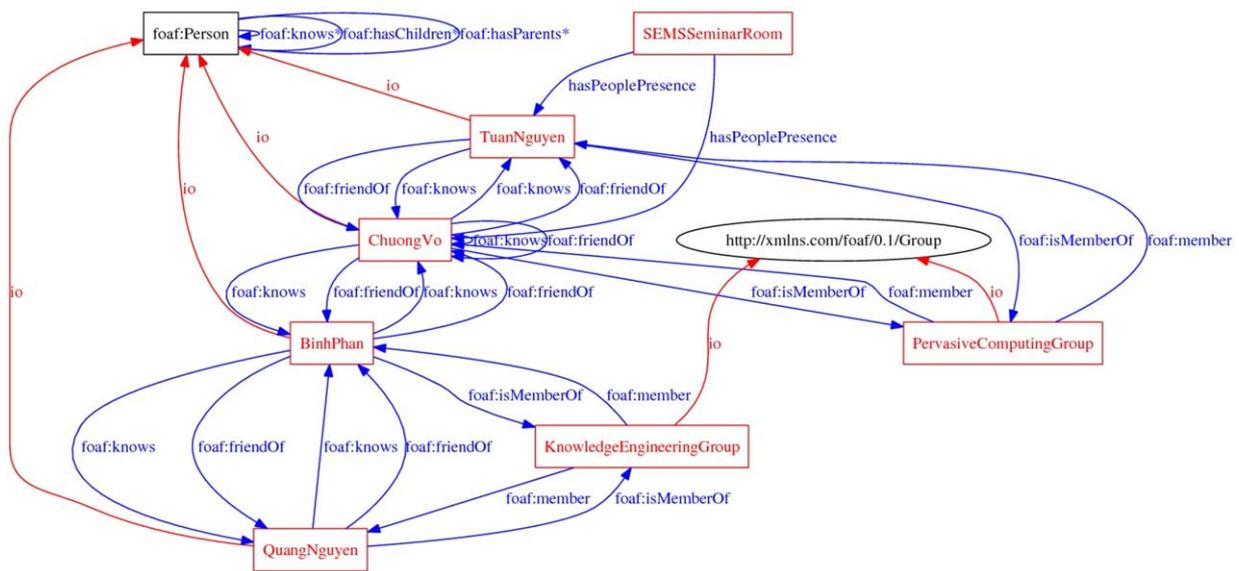


Fig. 4. Special interest group meeting activity in the SEMS seminar room.

onds.⁶ In [13], we discovered the room's routine, that being, the room is most crowded on Tuesdays and Thursdays in the middle of semester, by monitoring the seminar room for the presence of Bluetooth devices.

Based on the PAO, we can build a knowledge base to model activities at a place. Then, by analyzing the sensed data using the knowledge base, we can discover when a group seminar activity is occur-

ring in the SEMS Seminar Room, as an instance of RoomActivity. Figure 4 illustrates a representation of the group seminar activity and persons involved (including the people's friend relationships using concepts borrowed from the FOAF ontology⁷ – which are not part of the CORE PBVC ontology). Note that the `hasPeoplePresence` relationships are detected via Bluetooth and established at run-time, and are transient though time-stamped. This is done by detecting the presence of people using Bluetooth discovery

⁶According to Bluetooth specifications, the query cycle timeout is 20 seconds. Therefore we ensure that any devices appear in the room can be detected.

⁷<http://xmlns.com/foaf/spec/>

(or sensing), and then from detected Bluetooth device IDs, we can infer the identities of the people and the fact that these people belong to the Pervasive Computing Group, by consulting the knowledge base (ontology) as shown in Fig. 4.

5. Example applications

We describe three example applications we have developed based on the PlaceComm approach: PlaceGuide, PlaceAware and PlaceSense. Here, we focus on PlaceGuide, and only highlight key features of PlaceAware and PlaceSense since they are described in detail in [11] and [12], respectively.

5.1. PlaceGuide

The main idea is to provide a voice feedback from the environment as detailed as possible to users in order to help them get to know a place (or particular Points-of-Interest (POIs)) more easily. PlaceGuide was inspired by Paulos and Jenkins's work [15] on place-based ringtones. In their work, the authors describe a Hullabaloo system that scans nearby Bluetooth phones and generate a mix of audio tracks. In the PlaceGuide system, a program that runs on the user's mobile phone scans Bluetooth devices attached to fix objects (or POIs) in the environment. Then, it will play a sound file for each Bluetooth device (or POI) detected in order to inform users about the POI. The sound file may contain a verbal explanation about the POI, or alternatively, aesthetically, the sound files can be used to create a sense of "liveliness" or mood about a place.

PlaceGuide can run in two different modes: the stand-alone mode and the community-based mode. In the stand-alone mode, PlaceGuide senses the bluetooth devices around the users and provide an audio feedback to them. The community-based mode is more complex, where users are monitored and aided by the multiagent-mediated PBVC.

The system is low cost, and scalable. A Bluetooth-enabled mobile phone now is fairly affordable for many people. The Bluetooth-tagged environment can be expanded to room scale, street scale, campus scale, or even city scale.

We used a bunch of Bluetooth-enabled phones in our prototyping. The phone plays two roles. The first role is a tag or a beacon. We use old mobile phones (with Bluetooth capability), named them, and attached them to objects (i.e., stationary POIs) that we want to

tag. The second role is that (some of) the phones serve as the computer that hosts the User Agent program. On the phone carried by the mobile user, we built the PlaceGuide User Agent that continuously scans the environment around the user. When the agent discovers any other Bluetooth-tagged POIs, it will effectively "speak" to the users the name (and a description) of those POIs.

Users can join a PBVC via the User Agent. The User Agent on the mobile phone can connect to the place KB via a User Proxy Agent on a stationary server hosting the PlaceComm server and the place KB. The place KB is built on top of the PBVC Ontology. The place KB contains all information about the PBVC. The place KB can be updated and enriched by members.

Scenario. We consider a university and its associated PBVC. John is a user of the PlaceGuide system. John is visiting a new university today for a meeting. His place is 10km away from the university so he has to take the Hurstbridge train line and gets off at Macleod station. He then walks for 5 minutes to take bus route 561 and gets off at the bus interchange in the university. With the PlaceGuide application he can choose the correct bus or train to get in (by consulting information about transportation for that place). During his trip, he discovers his classmate is also in that train, just 10m away from him, so he can call his friend to chat. At university, all corridors and buildings are tagged, so he can be directed to the office for the meeting. When he is in the library (where the meeting takes place), his phone is fully connected to the community. He can upload to the community server data associated with the university that he created during his trip, including, with his permission, Bluetooth tagged POIs he passed by or even his own location (if he chooses to do so, or in anonymized form).

Figure 5 shows the user's travel surroundings in the campus scenario. The blue circles represent locations of the Bluetooth phones (tags).

PlaceGuide implementation. As visualized in Fig. 1, the PlaceGuide system is build based on PlaceComm, and consists of three parts:

- The sensing takes the form of discovering Bluetooth tags, which are old phones, Bluetooth dongles (connected to laptops) or commercial Bluetooth tags⁸ on objects (POIs) in the environment

⁸<http://www.bluelon.com/>

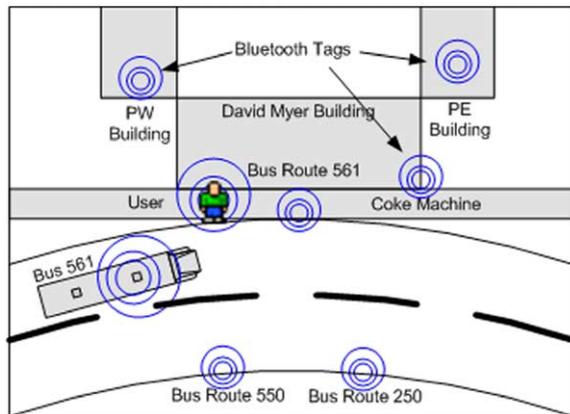


Fig. 5. User travel map.

(some shown in Fig. 6, and a deployment illustrated in Fig. 7) such as bus stops, drink machines, building entrances or even trees. In our trials, we built a corresponding audio file to describe each tagged POI.

- A User Agent runs on the user’s mobile phone. This application continuously scans for Bluetooth devices. When it finds the pre-known Bluetooth tags, it will look-up and play the corresponding sound files. If the detected tag is unknown, the User Agent will issue a specific beep to acknowledge this to the user.
- A PlaceComm server runs on a stationary computer and the mobile user (via the User Agent) connects to it in order to interact with services and the place KB.

In the place KB, there is knowledge about objects (and POIs) at the place. Each object or POI is identified via a Bluetooth deviceID of the phone (or Bluetooth tag) attached to the POI. In the place KB, we assign each bluetooth tag address to a specific sound file. The sound file is a simple voice description about the object. For example, we have the file `bus561.wav`, a voice recording that contains this sentence “bus five hundred and sixty one”, associated with a Bluetooth tag address on Bus 561. A Library building, a tree, a coke machine, and BusStop 60 are examples of other tagged entities, each associated with a Bluetooth deviceID, for their identification. Note that Bus 561 is a mobile object (or POI) but the others are stationary. We used the FreeTTS,⁹ a Java text-to-speech engine, to generate the sound file. This engine is also used to gen-

erate a simple voice reply when the User Agent queries the place KB. For example, the User Agent sends the tag name “David Myer Building” to the Community Agent, and can then receive the sound file via a wireless network (if not already preloaded) or even a Bluetooth connection.

In the scenario, John belongs to the La Trobe University PBVC. To optimise performance, we preloaded the sound files and the above mapping on his device. We also add his friends’ Bluetooth ID to the database so that he can find his friends.

The User Agent is a JADE/LEAP¹⁰ agent running on a JavaME mobile phone that can continuously scan for Bluetooth devices and communicate with the User Proxy Agent in the PBVC. When a user is traveling, s/he can save all strangers’ devices or obstacles that were not pre-loaded into his/her mobile phone. When s/he arrives at the campus, s/he can connect to the unsecured campus wireless network to join the PBVC in the area, and ask about the unknown objects detected.

Evaluations. We measured how long it takes for the User Agent (on the user’s mobile device) to detect a Bluetooth tagged object once within range. The Bluetooth zone according to specifications is 10 metres, but in the pilot study, in an outdoor environment, in open spaces, the zone was found to be up to 18 metres. In the testing scenario, we tried 100 times stepping in and stepping out of the zone, and recorded the time it took for the mobile phone to discover the Bluetooth tag. The average discovery time is 1.6 seconds (see Fig. 8, each vertical line representing one try). This means that with people’s walking speed at about 5km/h or 1.4m/s, they can easily discover an object before they can really reach that object. This test suggests that people will not miss objects when they are passing by.

The combination of information of a place such as sights, sound, noise, smell, road surface texture, are able to form a fingerprint for that place or a POI within the place. For example, when hearing a bell from the church in her/his town, one knows that s/he is close to home. While the sound of church bell can be heard from kilometers away, in PlaceGuide, sounds can be heard within 10 meters of a POI. Triggering various sounds (played on the mobile user’s device) could help create a sense of familiarity with the place (or POI), adding to the place charm or character (or even be used by the blind or vision-impaired as additional clues). Moreover, different sounds, at different times, could

⁹<http://freetts.sourceforge.net/docs/index.php>

¹⁰<http://jade.tilab.com/>



Fig. 6. Some devices we used in our trials.



Fig. 7. Pilot study.

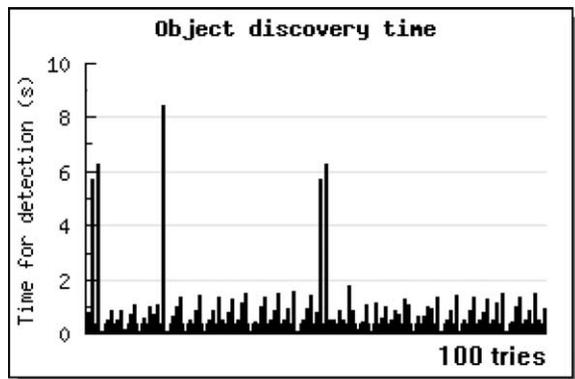


Fig. 8. Bluetooth tag object discovery time.

convey different moods or feel about a place or POI, or mark occasions and events.

Indeed, while PlaceGuide is not specifically designed for the vision-impaired, we note that there has been a range of work done for helping the vision-impaired. Helal et al. [7] implemented Drishti, a wireless pedestrian navigation system. The Drishti system integrated many technologies such as a wearable computer, voice recognition and synthesis, wireless network, GIS and GPS for helping the blind. The system continuously guides the user regard to the user's environment. The Cyber Crumbs project used RFID and IR technology to implement a system that guides users along the walk by detecting users' locations and giving suggestions [18]. The "Cyber Crumbs" concept is the idea that RFID tags that can be put along the walking path, like a trail of crumbs to follow. The RFID receiver is embedded in the blind person's cane. When the user travels, the system gives a spoken phrase "turn left", "turn right" or "go straight" to the user. Similarly, Bellotti et al. [2] built a location-aware tour-guide based on RFID for helping visually impaired people in an exhibition. The authors set up 99 RFID tags in a place of about 30,000 square metres exhibition rooms to cover 99 points of interest (POI). When the user approaches a POI, the PDA with RFID reader recognizes the RFID tags and plays the description about that POI. In PlaceGuide, we make use of standard Bluetooth technology, but RFID and GPS can be integrated if needed. When the users approach the zone of a POI, a voice message can be sent directly to the users phone (either through Bluetooth or WLAN).

In future versions of PlaceGuide, when users discover any new device, they send the name of the device to the PBVC's place KB so that they can get voice

descriptions about that device (or the tagged POI). A PlaceComm server and PlaceGuide can be deployed in different places such as in shopping centers, in supermarkets or in museums. Also, the objects (or POIs) can also be annotated by other users and shared via the place KB. So, the place KB can contain information about members, tagged POIs, annotations, locations of POIs and people (if they so allow, or locations stored for analysis but anonymized). Such knowledge can be queried via SPARQL¹¹ and reused in different applications.

5.2. PlaceAware

PlaceAware is an application that one uses on a mobile device in order to query the place KB about people. A set of preprogrammed queries (mostly in SPARQL) are presented to the user in a user friendly manner, as depicted in Fig. 9. Selecting one of these queries (e.g., "Where are my friends?") and issuing it to the place KB will result in a response (as far as the contents of the KB will allow). Note that "my" can be mapped to the current user (in this example, "Tuan-Nguyen"). The following SPARQL query is used for this, which traverses the ontology, following the graph to determine the whereabouts of "TuanNguyen".

```

PREFIX pbvc:<http://www.tuannguyen.mobi/ontologies/
pbvc/2009/4/pbvc.owl#>
SELECT ?location ?myFriends
WHERE {
  ?location rdf:type plc:Location.
  ?location pbvc:hasTimeInstant ?timestamp.
  ?timestamp pbvc:timestampMillisecond ?value.
  ?location pbvc:belongtoPersonalContext
    ?PersonalContext.
  ?PersonalContext pbvc:createdByPerson ?myFriends.
  ?myFriends foaf:friendOf ?myselft.
  ?myselft foaf:name ?myName
  FILTER ((?value >=1235 && ?value <= 9999) &&
    (?myName= "TuanNguyen"^^xsd:string))
}

```

Note that the result reports on three individual in this example, with known places. Given the geographical boundaries of a PBVC, the answer to this query might merely be "not in this place". In this example, either the user specifies a location when s/he leaves the PBVC (e.g., the user is leaving the campus and tells the PBVC that s/he is going home), or boundaries of the PBVC's area can include not just a campus but also locations of homes of pre-specified members.

A query such as "Who has been here?" can be answered and would fall easily within the boundaries of a PBVC. Figure 10 depicts such a query.

¹¹<http://www.w3.org/TR/rdf-sparql-query/>



Fig. 9. PlaceAware application running on a Nokia phone – showing a list of preprogrammed queries, then issuing a query about where my friends are, and then a result.

For the more technical users, rather than fixed preprogrammed queries, a parameterized (semi-custom) SPAQL query can be issued (e.g., “Where is <parameter>?”, where the <parameter> is a string entered by the user), or for experts in SPARQL, a free form query can be issued as shown in Fig. 11.

Although we mainly used SPARQL, we have also experimented with SeRQL queries using the “IN” op-

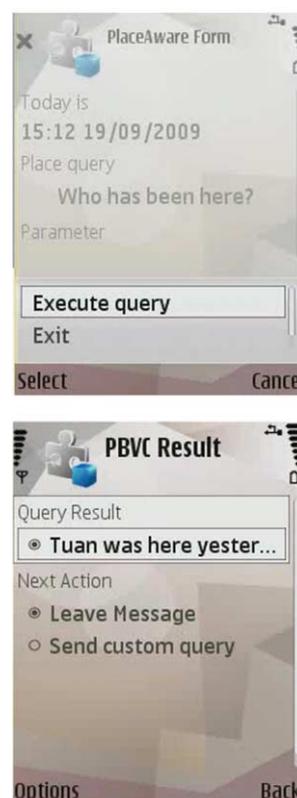


Fig. 10. Who has been at this place? Query and result.

erator for queries such as “What services in my interests are present here?” (where “my” is mapped to “TuanNguyen”). The query consults knowledge about services at a place and the categories in which the services fall, returning only services at the place which fall into the categories of interest to “TuanNguyen”.

```

SELECT Service , place, category
FROM {Service} rdf:type {pbvc:Service} ;
      pbvc:serviceAtPlace {place} ;
      pbvc:serviceCategory {category}
WHERE category IN
( SELECT currentPersonalInterest
  FROM {PersonalContext} rdf:type
      {pbvc:PersonalContext} ;
      pbvc:currentPersonalInterest
      {currentPersonalInterest} ;
      pbvc:createdByPerson {Person}
  WHERE Person LIKE "*TuanNguyen"
)

```

We note also that such knowledge about people can be used in other context-aware applications by accessing the place KB, for example, to generate appropriate advertising.



Fig. 11. More customized queries: parameterized (input parameter shown is the string “Tuan”) and free form custom querying.

5.3. PlaceSense

PlaceSense was an early application of PlaceComm, used to probe, via Bluetooth, everyday environments. A number of experiments were conducted from measuring utilization of a seminar room (which we did using Bluetooth discovery of mobile phones), to observing the community of a group of postgraduate students which had a priori registered to form a PBVC over the union of the university campus and a nearby market. We found that they usually went shopping at the same market near to the university every Friday afternoon, based on detecting their Bluetooth phones.

In a similar way, note that we can “observe” customers at a cafe, and see if the same people frequently visit the cafe or how often particular individuals visit the cafe – but all without knowing who the persons are, preserving their anonymity and privacy – we only observe the same Bluetooth deviceID occurring at the cafe each morning (so-called “familiar strangers” [14]). However, for deviceIDs of registered

PBVC members (that is, the PBVC corresponding to the cafe), with appropriate authorization, their names can be known to each other or to the cafe owner. Possibilities exist, therefore, for a context-aware application to query such knowledge in order to notice and respond to frequent customers – for example, by triggering particular advertising or greetings. In other experiments, we note observations about commonly visited locations by the same group of people (we do not know their identity or need to, but can detect patterns as a group) – the same group of people tend to take certain trains from certain stops and arrive at certain places. Such sensed knowledge can then be queried and shared by applications via the place KB.

6. Conclusion and future work

We have presented the concept of Place-Based Virtual Communities (which can complement general Internet-scale virtual communities) and the PlaceComm middleware for context-aware applications in PBVCs. A key idea is the sharing of a place KB (supported by an ontology) by applications. The ontology provides a “standard” view or schema of the minimal knowledge contained in, or which can be expected from, a PBVC.

In future work, we intend to develop probes that allow developers of applications to see what types of knowledge are available at a PBVC and provide a standard set of APIs for developers. We also hope to build further context-aware applications for our prototype university PBVC, and also to deploy more instances of PlaceComm to create several PBVCs over different categories of places. Also, we will continue to investigate scalable databases for large place KBs and Semantic Web rules reasoning with community knowledge.

Our ontology for activities that is integrated into the PBVC ontology is still fairly simple – we hope to extend this activity ontology with ideas from activity-based computing.¹² The work in [17] attempts to integrate statistical inferencing with ontology based reasoning using an activity ontology called COSAR-ONT. However, this work focuses on recognizing the activity of individuals such as jogging, walking down the stairs, standing still, strolling, brushing teeth and riding a bicycle (including an individual’s possible activities in different places) rather than what happens

¹²<http://activity-based-computing.org/>

at a place. The City Sense project¹³ supports visualizing in real-time of city activities, with maps of where crowds are in the city. An avenue of future work is to consider such visualizations, and how our approach can be combined with machine learning techniques.

We have not discussed privacy issues in this paper. Other work have attempted to develop systems to control sharing of context information,¹⁴ which will be useful in our work. The issue of reliability can arise in the sense of errors in sensing or failure to detect users, either due to unexpected user behaviour (user walks too quickly to be discovered) or hardware sensor failure (which in turn confuses the users who are expecting some response). While hardware failure can be helped with redundant sensors, tackling unexpected user behaviour will be more difficult – it might require users to adjust to system behaviour (in that users learn how to behave to get a response) but leads to inflexibility for users – more experimentation is required towards this end.

Acknowledgement

The first author is sponsored by Project 300, Ho Chi Minh City, Vietnam. The first author would like to acknowledge uncle Tran for providing his own experiences on how the blind uses sounds to navigate in reality.

References

- [1] F. Bellifemine, A. Poggi and G. Rimassa, Jade: A FIPA2000 compliant agent development environment, in: *Proceedings of the 5th International Conference on Autonomous Agents*, ACM, 2001, pp. 216–217.
- [2] F. Bellotti, R. Berta, A. De Gloria and M. Margarone, Guiding visually impaired people in the exhibition, in: *Gior-natanazionale su Guide Mobili Virtuali, Virtuality 2006*, 2006, <http://hclilab.uniud.it/sigchi/doc/Virtuality06/Bellotti&al.pdf>.
- [3] C. Bettini, O. Brdiczka, K. Henriksen, J. Indulska, D. Nicklas, A. Ranganathan and D. Riboni, A survey of context modelling and reasoning techniques, *Pervasive and Mobile Computing* **6**(2) (2010), 161–180.
- [4] M. Blackstock, R. Lea and C. Krasic, Managing an integrated ubicomp environment using ontologies and reasoning, in: *PERCOMW '07: Proceedings of the Fifth IEEE International Conference on Pervasive Computing and Communica-tions Workshops*, IEEE Computer Society, Washington, DC, USA, 2007, pp. 45–52.
- [5] H. Chen, F. Perich, T. Finin and A. Joshi, SOUPA: Standard ontology for ubiquitous and pervasive applications, in: *Proceedings of the International Conference on Mobile and Ubiquitous Systems: Networking and Services*, 2004, pp. 258–267.
- [6] C. Endres, A. Butz and A. MacWilliams, A survey of software infrastructures and frameworks for ubiquitous computing, *Mobile Information Systems* **1**(1) (2005), 41–80.
- [7] A. (Sumi) Helal, S.E. Moore and B. Ramachandran, Drishti: An integrated navigation system for visually impaired and disabled, in: *Proceedings of the 5th International Symposium on Wearable Computers*, IEEE Computer Society, 2001.
- [8] N.D. Lane, H. Lu, S.B. Eisenman and A.T. Campbell, Cooperative techniques supporting sensor-based people-centric inferencing, in: *Pervasive*, J. Indulska, D.J. Patterson, T. Rodden and M. Ott, eds, Lecture Notes in Computer Science, Vol. 5013, Springer, 2008, pp. 75–92.
- [9] E. Miluzzo, N.D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S.B. Eisenman, X. Zheng and A.T. Campbell, Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application, in: *SenSys '08: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, ACM, New York, NY, USA, 2008, pp. 337–350.
- [10] T. Nguyen, S.W. Loke and T. Torabi, The community stack: Concept and prototype, in: *AINAW '07: Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops*, IEEE Computer Society, 2007, pp. 52–58.
- [11] T. Nguyen, S.W. Loke and T. Torabi, PlaceAware: a tool for enhancing social interactions in urban places, in: *I-SPAN'09: Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms and Networks*, 2009.
- [12] T. Nguyen, S.W. Loke and T. Torabi, PlaceSense: a tool for sensing communities, in: *ISWPC'09: Proceedings of the 4th International Symposium on Wireless Pervasive Computing*, IEEE Computer Society, 2009, pp. 1–5.
- [13] T. Nguyen, S.W. Loke, T. Torabi and H. Lu, Placesense: A tool for sensing communities, *Wireless Pervasive Computing, 2009, ISWPC 2009, 4th International Symposium on*, Feb. 2009, pp. 1–5.
- [14] E. Paulos and E. Goodman, The familiar stranger: anxiety, comfort, and play in public places, in: *CHI '04: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, New York, NY, USA, 2004, pp. 223–230.
- [15] E. Paulos and T. Jenkins, Place based ringtones, in: *Workshop Position Paper for Imaging the City Workshop at ACM SIGCHI*, 2007.
- [16] C.E. Porter, A typology of virtual communities: A multi-disciplinary foundation for future research, *Journal of Computer-Mediated Communication* **10**(1) (2004).
- [17] D. Riboni and C. Bettini, Context-aware activity recognition through a combination of ontological and statistical reasoning, in: *UIC '09: Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing*, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 39–53.
- [18] D.A. Ross, Cyber crumbs for successful aging with vision loss, *IEEE Pervasive Computing* **3**(2) (2004), 30–35.

¹³<http://www.sensenetworks.com/citysense.php>

¹⁴<http://mcom.cs.cmu.edu/user-controllable-security-and-privacy/>

- [19] E. Toye, R. Sharp, A. Madhavapeddy and D. Scott, Using smart phones to access site-specific services, *IEEE Pervasive Computing* **4**(2) (2005), 60–66.
- [20] J. Ye, L. Coyle, S. Dobson and P. Nixon, Ontology-based models in pervasive computing systems, *Knowledge Engineering Review* **22**(4) (2007), 315–347.