What is next for Indoor Localisation? Taxonomy, protocols, and patterns for advanced Location Based Services

Francesco Furfari, Antonino Crivello*, Paolo Barsocchi, Filippo Palumbo and Francesco Potortì
ISTI-CNR (Information Science and Technologies Institute - National Research Council), Pisa, Italy
{name.surname}@isti.cnr.it, Potorti@isti.cnr.it

***Abstract*—Indoor localisation systems have been studied in the literature for more than ten years and nowadays are starting to approach the market. While technology is not mature yet, we can argue that the single biggest obstacle to wide adoption is the lack of standard ways to integrate different systems together. The missing pieces are a common taxonomy, definition of services, protocols. This work is an attempt to define what is next for indoor localisation systems in order to promote market adoption. It is a first high-level attempt at defining a taxonomy of indoor positioning systems, at outlining the main phases of a protocol for the utilisation of different cooperating indoor localisation systems, and at drawing a vision of services and applications in the close future.**

***Index Terms*—Indoor localisation taxonomy; Location Based Services; Indoor localisation integration; standards**

I. Introduction

In the last decade, several Indoor Localisation Systems (ILSs) have been proposed, with several differences in terms of data sources used and methods exploiting them. The most promising ILS data sources are based on the opportunistic exploitation of radio communication systems like BLE and Wi-Fi and on usage of MEMS like inertial, pressure and magnetic sensors. All of these are available in most mobile devices [1], [2]. Recently, camera based systems are emerging as an additional data source on smartphones [3]. Data sources are managed using a wide variety of methods and then fused together with different approaches. Kalman filters and particle filters the most used fusion methods, with various machine learning approaches, including deep learning, gaining momentum [4].

Generic Location Based Services (LBS) platforms have started to appear. Here we mention IndoorAtlas, Anyplace, Quuppa, Google Maps.

IndoorAtlas is an indoor service platform operating in the health care, retail, and transportation domains. It serves the Mumbai Airport assisting the travellers through the terminals and guiding them to the nearest facilities and gates. IndoorAtlas [5] exploits magnetic field, Wi-Fi, BLE, pressure and inertial sensors, and map information. The platform provides a set of tools for indoor positioning, enabling the opportunity to build one's own indoor location-based services such as search, navigation and proximity marketing.

* Antonino Crivello is the contact author

978-1-7281-1788-1/19/$31.00 © 2019 IEEE

Anyplace is a service-oriented localisation architecture that leverages Wi-Fi fingerprinting and inertial sensors [6]. The platform provides the possibility to define PoIs (Points of Interest), to manage maps and the fingerprinting phase, and it gives a RESTful API/SDK to developers.

Quuppa is a software platform used for planning, deploying and configuring the indoor localisation system using Angle-of-Arrival technology based on roof-mounted flat antennas and BLE devices carried by the entity to be tracked [7]. The platform offers a standard JSON/REST push/pull API, it enables both real-time viewing and recording/replaying of data and creates heat-maps, trajectories, and zone history tables for analytic purposes.

Indoor Google Maps is based on Wi-Fi fingerprinting and provides routes, places, and PoIs after a survey of the indoor area and once the indoor map is provided.

As we can expect the advent of more LBS platforms in the future[8] [9], the need is emerging for commonly accepted performance evaluation methods. Several efforts have been made in order to fill the gap between research and shared standards [10]. ISO/IEC 18305:2016 was the first official standard to appear in this area. It describes a methodology for evaluating indoor localisation systems and Test&Evaluation (T&E) procedures for Localisation and Tracking Systems [11], [12].

We argue that the single most important gap to market acceptance is the lack of accepted methods for ILS integration and cooperation. For example, in a large shopping mall several ILSs may coexist, each store having its own independent ILS. In this scenario, different ILSs could cooperate in overlapping coverage areas to make transition transparent in terms of quality of service and technology (handoff) and seamless (handover) in terms of user experience. Moreover, all the ILSs installed in a common space may provide a consistent level of service in the overall places in which they are deployed. In a scenario like this, the coordination among these systems becomes necessary.

This paper is intended to make the first steps towards defining some common concepts: a taxonomy (Section II); the abstract components of a reference architecture (Section

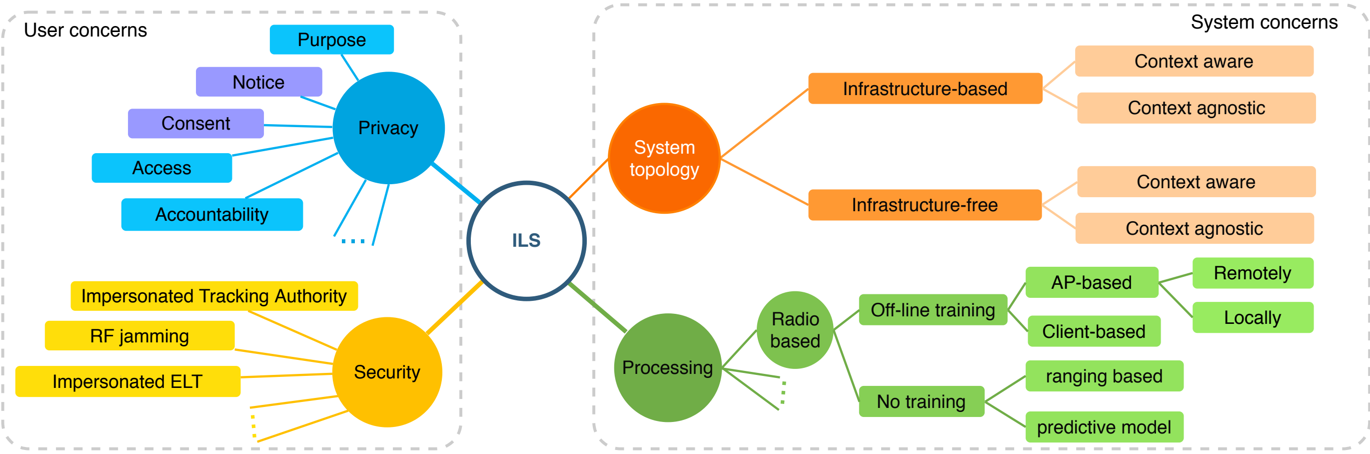

Fig. 1. ILS taxonomy draft

III); and the information exchanged during service discovery (Section IV). The main goal is to have a common view on how to integrate different ILSs in a shared environment in a way that is transparent to the user. To this end, in the last section we briefly present some scenarios that exploit cooperation among different ILSs and applications.

II. ILS TAXONOMY

An ontology is a formal, schematised representation of a field of interest; it contains the set of concepts (entities, attributes, etc.) and the relationships between concepts that are necessary to describe the knowledge of a particular domain. An ontology can be realised through different conceptual maps that highlight the business model, the technological solutions, and the use cases in which the domain stakeholders can be involved. Capturing all the knowledge of an application domain is an expensive task, in general, and is made more difficult for ILS given the continuous evolution of technological solutions. Efforts in this sense have been made in the form of survey papers [13], [14], classifications of localisation methods [15] and ontologies [16] that, however, have been introduced mainly to improve the estimation of the position.

We advocate the importance of standardisation efforts not only for ILSs, but for all of the components involved in supplying location based services as described in Section III. As a small step in this direction, we start by introducing a non-exhaustive example of taxonomy which we use to discuss about few fundamental concepts to be further detailed and to be agreed upon. Empty arcs in Figure 1 represent concepts like inertial sensors or camera-based systems that are not included because of space limits.

Any standardisation process should start by defining a clear terminology, or dictionary, to solve language ambiguities. ISO/IEC 18305 proposes some standard terms, which we borrow in the following:

- ELT – entity to be located or tracked
- locating – knowing the position of an ELT
- positioning – an ELT knowing its own position
- localisation – used for both positioning and locating
- tracking – following the position of an ELT over time
- navigating – computing a path to a final destination

Privacy in figure 1 has children highlighting various aspects of privacy concerns. Among these, *Notice* and *Consent* are coloured differently because they require user interaction: the system should notice when user data is being collected, and user should approve data storage and possible disclosure to third parties.

Accountability is the possibility of identifying a responsible in charge of privacy requirements. All the information related to these aspects has to be managed and eventually stored by the user when she enters the coverage area of an ILS. Notice and consent information is exchanged during the *Accessing phase* described below in the *Protocol in Action*, while the rest of the information is advertised during the *Discovery phase*.

Security is an important aspect of indoor localisation, so ILSs should be protected by cyber attacks, such as impersonated ELT and impersonated tracking authority.

Topology classification is used to identify the system with its infrastructure, if any.

Processing is used to describe the workflow used by the system: when and how information is exchanged between the system and the user (see *ILS* in the Architecture section).

Information relative to security, privacy, topology and processing is exchanged during the Discovery phase (see figure 5) and subsequently used to properly interact with the ILS of interest.

System topology and data processing techniques define the inner working of a system and are well investigated in the literature [13]. Systems can benefit from exchanging information about their inner workings. For example, a fingerprinting-based system can be improve its fingerprint database by using data coming from anchors of another system. Similarly, RFID-based systems can share the position of RFID readers.

We propose a two-dimensional binary classification. The first dimension concerns context knowledge, the second one concerns the presence of ad hoc infrastructure. An ILS is *context aware* if it has knowledge of the environment, for example through access to building maps, position of Wi-Fi access points, position of BLE anchors and so on; conversely, an ILS is *context agnostic* if it uses no knowledge of the environment, as it is the case for people or robots accessing unknown environments. On the second dimension, an ILS is *infrastructure-free* if it is able to exploit information from the environment, for example by using cameras and SLAM, listening to Wi-Fi access points or using pedestrian dead reckoning; conversely, it is *infrastructure-based* if it relies on location information purposely provided to this end, as it is the case for UWB systems of BLE systems based on roof-mounted AoA scanners. The two dimensions define four different classes, which are not exclusive: an ILS may belong to more than one class.

III. LBS Architecture

As mentioned in the introduction, indoor localisation systems are already available, and increasing number of vertical solutions will appear next years bundled with smartphone apps. By entering such smart environments, customers have only to launch the proper application and use its services. A natural consequence of such an approach is that sooner or later smartphones would be clogged with fairly identical applications and yet a customer able to navigate the indoor environment in the departure airport may be not able to get the same localisation service in the arrival airport because a different ILS is deployed and the right app is not installed on the smartphone. The solution is to define an abstraction layer, shared by all smartphone applications, which enables access to heterogeneous indoor localisation systems in a standard way.

Localisation APIs and location providers are already available in current smartphone platforms, what we need is an extension mechanism able to integrate new indoor localisation systems which may be based on custom technology working with different workflows. A minimal abstraction layer should provide a common discovery protocol to recover the characteristics of indoor localisation systems and to access their specific interfaces (Figure 2). For example, we can have either local or global discovery process: as an example of local search consider the UPnP standard enriched with new ILS-type devices which announce their description. When the user starts the application, an UPnP query is automatically launched and all the ILS-type devices respond by offering their description. Then, from the system description, the application (from here on User Agent), can select the proper ILS and access the physical channels adopted by the system (e.g. BT or Wi-Fi channels) to use its features. Alternatively, a global search can takes place through the Internet: googling the name of the site with a smartphone, the main information are retrieved, such as website, street address, telephone and, among these, also the URL to access the deployed ILS and its features.

The previous discovery example takes into account only two components: the ILS and the User Agent, but in order to capture systems' heterogeneity, all the components involved in the implementation of LBS should be considered. The main

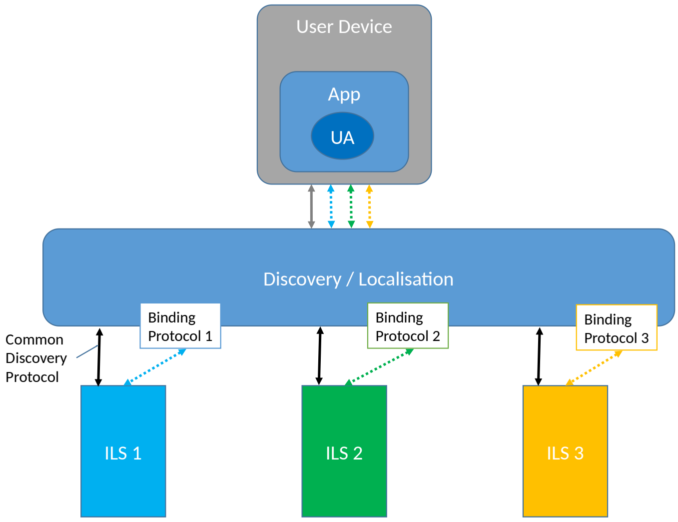

Fig. 2. Simple abstraction layer

components that characterise an LBS platform can be grouped into six abstract components: *ILS*, *Map Server*, *Navigator*, *Translator*, *User Agent* and *Communication Protocols*. We first focus on the characteristics of these abstract components and then describe a protocol example in more detail.

Abstract components can be implemented in one or separate physical components, they can be associated in different ways and can be distributed over several tiers. In the next section we describe use cases with abstract components residing both in one and in several physical components. Abstract components are all optional: for example if the ILS component is omitted it is still possible to use the rest of the components to guide the user in an unknown environment.

- An **ILS** consists of hardware and software components in charge of locating and tracking an entity (ELT), including possible components installed on the ELT and the environment. A taxonomy is used to uniquely identify the system's category and the workflows used by the system. For instance, in RSS-based ILS the position can be estimated by processing the RSS received either by the access points or by the client. The application running on the smartphone during the discovery process has to detect the type of system, e.g. Wi-Fi based, and how to interact with it: AP-based vs Client-based.
  Thus, ILSs have to self-describe both their properties and interfaces. In particular two classes of interfaces should be considered. The first one towards clients requesting positioning information; endpoints and protocols used to communicate entities' position as well as data streams for tracking. The second one for M2M (machine-to-machine) communication with other ILSs placed on adjacent zones. In fact, multiple ILS can be present in the same big area to solve scalability issues, or because independent organisations have deployed their own ILS at different times, i.e. shops inside air-terminals or independent institutes in the same research area. When moving from the coverage area

of one ILS to another one, a client could need switching among different technologies.

- The **Map server** provides maps and associated context information such as PoIs, visual representations for 3D rendering and position of reference points. Reference points include radio anchors (Wi-Fi AP, BLE beacons, UWB transmitters, RFID tags) and visual tags (QR or bar codes, written labels). Maps and the associated context information are used for several purposes. They are used as input context data for localisation purposes by the ILS; they are used by the User Agent to provide rich map display on user interface; they are used as context information to create navigation paths. A well known example of this kind of service is Google Maps, which offers fairly rich APIs, including indoor maps. Currently, some airports are available with a detailed multifloor indoor map. In fast-changing environments like supermarkets map updating in a centralised server may be inadequate and local map server could be more appropriate. Map servers may provide a wide variety of information, which should be notified to the User Agent during the discovery process. For instance, a local map server could maintain very updated photographic layers of the environment and the ability to provide augmented reality services.
- The **Navigator** is the component that uses distance graphs to calculate routes to a specific target inside the coverage area, including special requirements such as avoiding stairs when carrying a load or supporting a group of people moving together. Currently the route updating is done only relatively to the mobility of the User Agent reaching some fixed target, but more generally we may need to update routes to several moving participants. All these capabilities need to be advertised during the discovery process to the User Agent. In the last section we propose some usage patterns.
- The **Translator** is the abstract component that performs a linguistic rendering of the information displayed on a map. In fact, we are used to a sound feedback that shows us the path to follow without being distracted while we are driving a car. Instead, as input tool the translator should facilitate the searching for a particular entity, and a user can refer it by using either a generic name or the name of a brand that identifies it. It has to be extensible with ad hoc vocabularies and be based on NLP techniques.
- The **User Agent** is the abstract component that interacts with localisation systems on behalf of the user, taking on the heterogeneity of the systems. It discovers the available ILSs and configures the user application to run properly with the rest of components made available locally, if any. It is the core of the user application running on the smartphone that integrates Navigator, maps and language skills.

  The metaphor that we consider most appropriate for this application is that of instant messaging chats with which users are used to interact. Thus a shop offering location based services can be impersonated as a chat user with chat bots and contextual menus exposing shortcuts for accessing such new services. LBSs as new navigation patterns should be announced during the discovery process.
- Finally, with **Protocols** component we mean the set of standards used to allow interoperability of indoor LBSs. In general, standard agreement will be needed about:
  - the discovery of localisation systems,
  - description of workflows,
  - data streaming,
  - user interfaces for LBSs and navigation patterns.

A. Protocols in Action

Seamless transition between outdoor and indoor navigation is certainly among the requirements of a standard for indoor navigation, moreover it should facilitate the integration of both proprietary and open systems. Here we highlight the 5 phases that should be model by a standard. We avoid describing the phases related to *deployment*, *configuration* and *commissioning* of a localisation system that are preliminary to its use.

As reference scenario, imagine a visitor arriving at a research area, invited by his guest from a research institution which is part of the area. The area is equipped with an ILS through which the visitor will be guided towards his guest.

- **Initiation**: this phase describes how an indoor navigation is initiated. We can distinguish at least 3 cases:
  - *User-initiated*: the simplest case where the user himself starts an application when he is aware of being inside an area where location services might be available (smart space).
  - *Environment-initiated*: here the environment is aware of the arrival of the new user; a typical example could be the ILS detecting Wi-Fi probes emitted by a smartphone and, following some dedicated protocol, sends a welcome message to the user.
  - *Invited*: this is the case of the reference scenario, where the user is invited into an area where it is possible to be guided. The invitation could include area coverage coordinates so that the user's outdoor navigator defines geo-fences for launching indoor navigation handshaking when entering the indicated area.
- ***Discovery***: the User Agent retrieves the description of one or more ILSs deployed in the area. We could have local versus global discovery: i.e. internet-based. In the case of local discovery a pivotal protocol should be selected for a reference network, otherwise given the heterogeneity of the systems, multiple networks scan should be done (i.e. BT and WiFi). Example of local discovery protocols are UPnP, ZeroConf, SLP.
- ***Access***: in this phase the user gives consent to be located and finalises the handshaking with the system. The user exchanges and stores privacy information: who is monitoring her, reference website and contact point, supervisor and so on. This is the phase where any missing

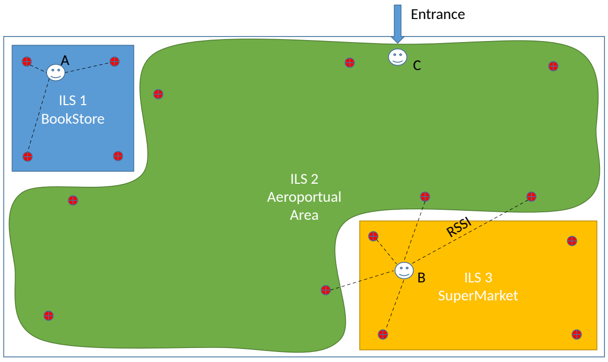

Fig. 3. Air terminal scenario

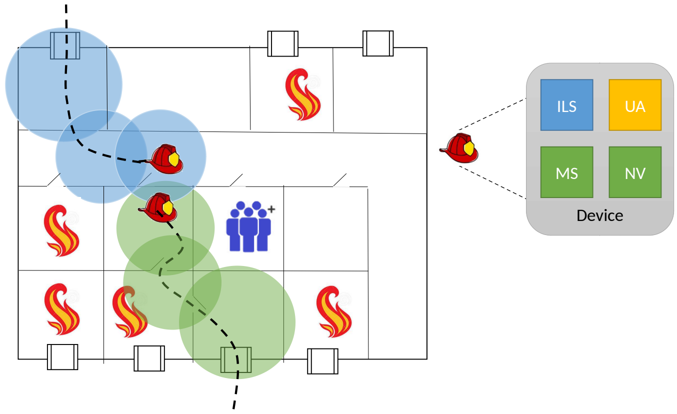

Fig. 4. Emergency and rescue scenario

information is managed. As an example, consider the case where the discovered ILS uses a new brand technology, a well self-describing system should offer alternatives to match correctly its workflow; suppose, for example, that during the discovery phase the User Agent learns that the ILS needs a ZigBee interface which is not present on the user's smartphone. Workflow description may enable the User Agent to select alternatives: for example to co-locate the visitor with an external device, that is the ZigBee badge given to him at the entrance of the area, but still receive notifications via her smartphone. Thus a smart User Agent may prompt the user to scan the badge QR code in order to communicate it to the ILS and finalise the handshaking.

- ***Localisation and Tracking***: this phase is tied to the user experience and therefore to the user interfaces provided by the User Agent. The services available in the area should be exposed in some way to the user, as well as maps. Navigation patterns should emerge at the level of interfaces, if equivalent functionalities are not already available in the smartphone. Generally, the APIs provided with instant messaging chats are adequate for the mashup of high level services. Through a chat you can enter text, or send voice commands via audio to request an entity location. You can access the video camera to navigate with augmented reality or open an inline web page to access instantaneously to Google Map or any Map Server provided by the local indoor system. You can share and forward information with other users or with the indoor system and manage the visibility of your messages. Standardisation of contextual menus, bot commands and interface shortcuts are essential to realise a common user experience on different smartphone platforms.
- ***Leaving***: cleanup of allocated resources. This phase is initiated by timeouts on the client and server side and reinforced by checking exit corridors defined in the area coverage.

IV. Traversing ILSs

Companies such as those mentioned in the introduction promote the advantages of adopting their own indoor navigation system in various sectors such as: retail sales, airports, universities, hospitals, congress buildings, stadiums, and so on. Main benefits are the data analytics production to improve the work organisation and greater freedom and more security for the users who would prefer companies offering such services. In this section we focus mainly on two scenarios to highlight aspects that emerge when a person moves in a large area where different indoor location systems are deployed. A third one scenario is reported to show how the abstract components introduced in the Architecture section can be combined to solve very different use case. Multiple ILSs may impact on the modality of the discovery process and on the type of resources shared among ILSs.

A. Scenarios

1) University campus: a large area with different buildings belonging to one or more departments. ILSs based on different technologies have been installed at different times, according to the policies of each department. They cooperate to offer a fluid service to visitors who must find their way on the campus for the first time: switching between GNSS outdoors and different ILS installed in different buildings.

2) Airport terminal: stores in the terminal may deploy their own localisation system. Each store has acted independently at different times to install its own ILS. Coverage areas are multifloor and normally overlapping, both between stores and with the common areas managed by the airport authorities (see figure 3). Note that in figure 3 User A is localised only by the Bookstore ILS, and position accuracy of user B could be improved because RSSI belonging to anchors of different ILSs are available.

3) Emergency and rescue: firefighters (figure 4) look for people trapped in a burning building. Even if the building were equipped with a stationary ILS, it could be completely or partially out of order, and generally no map is available. Firefighters open one or more passages in an environment usually saturated with smoke and with poor visibility. Therefore it is necessary that the fire brigade dynamically build a map as they explore the environment avoiding dangerous areas. In the figure two firefighters are trying to reach people through two different paths, of which the green one soon becomes unusable due to the spread of fire. In similar situations it is important that the local maps created by each fireman advancing in the building can be merged to give a global view of the building and have more chances to find escape routes. Abstract components installed in the support device of the firefighters are depicted on the right side of the figure. Each firefighter is equipped with a mobile ILS that must coordinate with other ILSs to update the maps by an automated and transparent M2M process.

The main difference between the two first scenarios is that the first scenario is *multibuilding*, so the common areas are covered by GNSS, while the second one is *purely indoor*, so common areas are covered by an ILS. In both first and second scenarios different *stationary* ILSs installed in the environment coexist, while in the third scenario only *mobile* ILSs exist.

Common to all scenarios is the necessity of coordination among ILSs. In the third scenario, coordination is a *functional requisite*, and is built in the system right from the start. In the two first scenarios we observe infrastructered cases where independent ILSs exist, which are generally installed at different times, with different capabilities and managed by administratively independent entities. We highlight three *pragmatic reasons* why we expect a natural propensity to coordinate such independent and heterogeneous systems:

- first, the need to simplify human-machine interaction favours the aggregation of information for which user consent is required: it would be impractical for the user to connect and give consent to interaction with each ILS in a given area;
- second, dividing a complex area into zones covered by different subsystems reduces the complexity of the system, distributes the load and reduces congestion points of a centralised system, in short scalability is eased;
- third, various subsystems being managed by autonomous organisations can be freely reconfigured according to individual needs without impacting on remaining subsystems, for example as consequence of changes in sales, lesson patterns, furniture.

B. Aggregating ILS information

Discovery information in complex scenarios could be provided either by each individual ILS in response to a query issued by a User Agent, or collected within the area by a gateway device and provided as aggregated info; in this second case information on topology could also be provided, such as geo-fences, how various systems cover the entire area (see figure 5 . We use this second case as a reference scenario (figure 3) where inside the terminal there are three ILSs: the first one, in green, covers the common area of the airport space, and the two other ones are installed inside some large shops.

Discovery information is described in figure 5, represented as a tree. At the first level there is the list of available ILSs, complete with topology information. Looking at the exploded

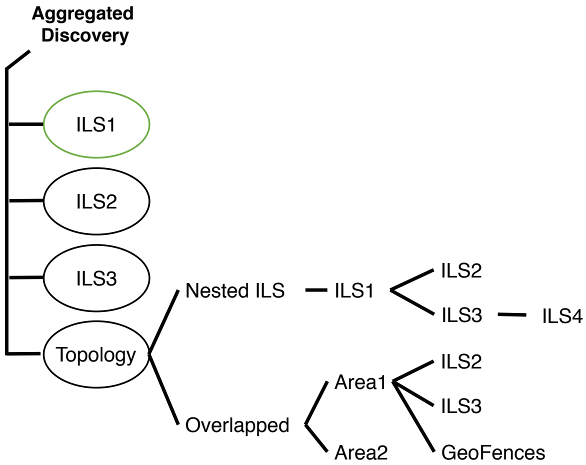

Fig. 5. Aggregated discovery information

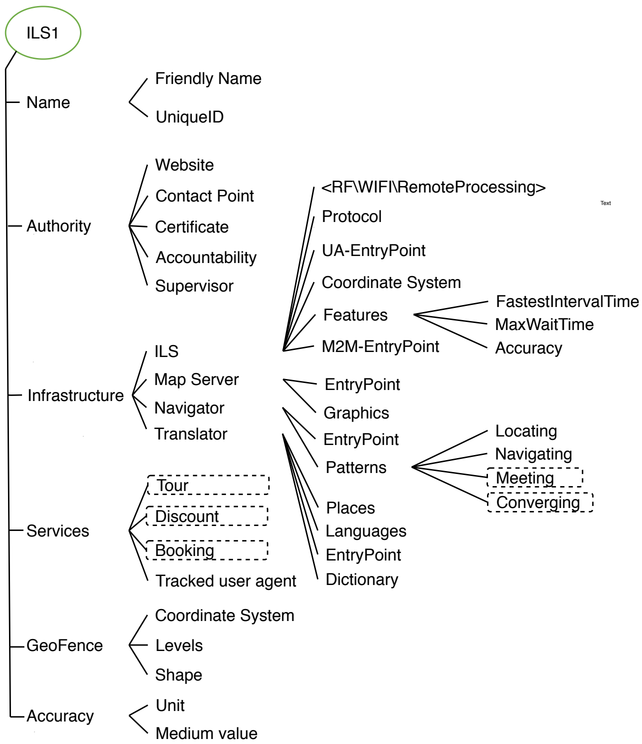

Fig. 6. LT1 exploded discovery information

system ILS1 (see figure 6) we can see 6 different types of information nodes:

- The ***Name*** of the system to use both for communications with the user and for machine to machine communication.
- The ***Authority*** who supervises the system consisting of: the web site, the contact points, public certificates to be used for authentications, accountability information for verifying proper privacy management and references to the person responsible for the system (see taxonomy draft).
- ***Infrastructure*** information which describes the main

components of the system: the ILS, the Map server, the Navigator and the Translator server.

  - The *ILS* node shows the technology on which the system is based. In the figure this information is represented as a path of the taxonomy introduced in section 2 (e.g. W3C XPath standard). In this example we assume that the system is Wi-Fi based and that remote processing is applied to estimate the position. The Protocol node describes access information for data exchange: port and workflow; analogously the UA-EntryPoint specifies the interfaces used by a user agent to query the system. Other published properties are the Coordinate System, and features of the localisation service: Accuracy, Maximum Waiting Time, Interval with which location data are provided. Finally, M2M-EntryPoint are the interfaces with which the system interacts with other ILSs, for the publication and updating of data.
  - *Map server, Navigator, Translator*: the Map Server node is an optional feature, as already described the system could be integrated with an external map server, or provide references and description of a local map server. Equivalently, for the Navigator various patterns implemented by the server such as Locating, Navigating and Meeting, have been highlighted. The last component is the Translator node which provides dictionaries for uncommon places and dictionary of the shop items.

- The *Services* node publishes information about LBSs provided in the area: for instance, if it is possible to locate other users present in the area (Tracked user agent), if there are recommended Tours, if there are offers with Discounts, support for Booking facilities and so on.
- Finally, ***Geofence*** and ***Accuracy*** nodes offer a quick access to relevant information about coverage area and metric and error in the position estimation

C. Advanced Location Based Services

Here we discuss about two nodes nested in the discovery tree: patterns@Navigator and services@ILS, in order to provide example patterns and services.

1) Patterns:

- ***Meeting***: with this pattern the navigator does not have to elaborate a route to a specific place (navigating), but must continuously update the route between two people on the move (meeting = navigating + navigating).
  In a typical scenario two or more people make a generic appointment and when they are in the predetermined area they try to reach each other guided by the navigator. Another scenario is when they split off, for example in a large shopping centre, each one to make their own purchases, with the intention to meet after some time. The application that can act as mediator and initiate this pattern may be a messenger app; for instance it could be possible to start the *meeting* pattern with the person with whom you are chatting through a contextual menu enabled after the discovery phase. In this scenario we can envision both a decentralised and centralised distribution of localisation information. In the first case it will be the responsibility of the User Agent of each person to send their own coordinates, received from the local system to the User Agent of the other person through an offline protocol. In the second case it is the same localisation system that, once the position of the entities are computed, takes care of distributing the same information to both. This mechanism could be generalised using publish-subscribe-type features for anyone who has expressed a willingness to be visible to other clients; in this sense the node item ”Tracked User Agents” included in the discovery tree is a property of the system that could be exhibited to favour public interactions.
- ***Converging***: this pattern could be of particularly interesting where surveillance is a critical aspect, like in airports. Imagine that a person representing a potential threat should be intercepted by a group of agents. In this case the navigator does not have to calculate the shortest navigation of each agent to the target, but must make the group of agents converge through different paths closing all possible ways of escape of the person (i.e. Pacman metaphor). This should work as part of an early warning service, after the person has been identified as a potential threat and before he tries to disappear. This illustrates a situation typical of security scenarios, where the importance of integration between ILSs emerges clearly, together with the need for specific M2M protocols. An entity, for example, could exit a system's coverage area (central hall) and enter an area covered by a different system (shop area); therefore it is important that a query on the presence of a certain entities can be spread to all the ILSs and their maps used to compute possible escape routes.

2) Services:

- ***Tours***: it is generally a service that allows users to follow a thematic path within the environment; it could be a guided tour about artworks inside a museum, or the path to find the news that are present in the environment, or a path to buy only the products that are on offer. During the discovery phase, inspecting the services node of each ILS, it would be possible to recover the tours present in the system and communicate them to the user with a chat message when they enter the area of the store that offers the tour. The user is therefore free to ignore the message and configure the chat with the store according to his needs as he usually does with his circle of friends. The tour should be configured as a list of items with summary information and with the ability to dynamically open and inline the navigator to follow the entire route or go directly to an item on the list.
- ***Shopping List***: it is the classic message that is exchanged in the context of family chats to communicate things that

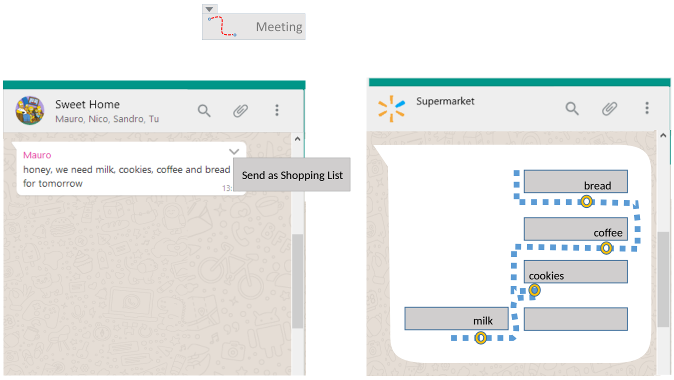

Fig. 7. Shopping list chat

need to be bought at the store. In the presence of such a service, activated during the discovery phase, the user has to do nothing more than communicate the shopping list to the system navigator to obtain an optimised route to the right shelves. From an interaction point of view, the message received on the home chat should be forwarded to the chat of the store, which automatically parses the data as a query to the store system (see figure 7). Alternatively a contextual menu could appear to send it directly to the Navigator to obtain an optimised route.

- ***Booking***: a modern version of the ticket dispenser service at the counters, which could allow you to book the service and be informed when it is time to go to the counter. In this case the computer system measures the average waiting time and therefore warns the user accounting for his distance from the counter; it may also plan a route for the shopping list in order to be in the proximity of the counter when the time is going to expire. Similar booking services enabled in railway stations or airports could alert passengers, accounting for their distance from the gate.

V. Motivation, Purpose and Future Work

We argue that time is ripe for discussing Indoor Localisation Systems at the system level, with the long-term purpose of creating a universal, seamless location and tracking service which works both indoor and outdoor using standard interfaces.

This paper is intended as a teaser, in the sense that it tries in no way to be exhaustive or to consider all significant cases or interaction, but to give the flavour of what an overall systematic view of location and tracking systems could be and how and why they should interact.

We hope that this work inspires the localisation and tracking community to discuss how to integrate systems and to create cases for interoperable location and tracking systems. The next steps include expanding and deepening the views proposed in this paper, but most importantly creating and analysing use cases and business cases, to be implemented in proof-of-concept integrated systems. This work can be intended as a contribution to the activities of the International Standards Committee of IPIN.

References

[1] H. Liu, H. Darabi, P. Banerjee, and J. Liu, “Survey of wireless indoor positioning techniques and systems,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1067–1080, 2007.

[2] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, “Zee: Zero-effort crowdsourcing for indoor localization,” in *Proceedings of the 18th annual international conference on Mobile computing and networking*. ACM, 2012, pp. 293–304.

[3] A. Mulloni, D. Wagner, I. Barakonyi, and D. Schmalstieg, “Indoor positioning and navigation with camera phones,” *IEEE Pervasive Computing*, vol. 8, no. 2, pp. 22–31, 2009.

[4] C. Mensing, S. Sand, and A. Dammann, “Hybrid data fusion and tracking for positioning with gnss and 3gpp-lte,” *International Journal of Navigation and Observation*, vol. 2010, 2010.

[5] S. Särkkä, V. Tolvanen, J. Kannala, and E. Rahtu, “Adaptive kalman filtering and smoothing for gravitation tracking in mobile systems,” in *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2015, pp. 1–7.

[6] A. Konstantinidis, P. Irakleous, Z. Georgiou, D. Zeinalipour-Yazti, and P. K. Chrysanthis, “Iot data prefetching in indoor navigation soas,” *ACM Transactions on Internet Technology (TOIT)*, vol. 19, no. 1, p. 10, 2018.

[7] M. F. S. van der Ham, S. Zlatanova, E. Verbree, and R. Voûte, “Real time localization of assets in hospitals using Quuppa indoor positioning technology,” *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. IV-4/W1, pp. 105–110, 09 2016.

[8] P. Bellavista, A. Küpper, and S. Helal, “Location-based services: Back to the future,” *IEEE Pervasive Computing*, vol. 7, no. 2, pp. 85–89, April 2008.

[9] A. Dey, J. Hightower, E. de Lara, and N. Davies, “Location-based services,” *IEEE Pervasive Computing*, vol. 9, no. 1, pp. 11–12, 2009.

[10] F. Potortì, S. Park, A. Jiménez Ruiz, P. Barsocchi, M. Girolami, A. Crivello, S. Lee, J. Lim, J. Torres-Sospedra, F. Seco *et al.*, “Comparing the performance of indoor localization systems through the evaal framework,” *Sensors*, vol. 17, no. 10, p. 2327, 2017.

[11] ISO/IEC 18305:2016, “Information technology – Real time locating systems – Test and evaluation of localization and tracking systems,” ISO/IEC JTC 1/SC 31, Standard ISO/IEC 18305:2016, 2016.

[12] F. Potortì, A. Crivello, P. Barsocchi, and F. Palumbo, “Evaluation of indoor localisation systems: comments on the iso/iec 18305 standard,” in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2018, pp. 1–7.

[13] J. Xiao, Z. Zhou, Y. Yi, and L. M. Ni, “A survey on wireless indoor localization from the device perspective,” *ACM Computing Surveys (CSUR)*, vol. 49, no. 2, p. 25, 2016.

[14] G. Deak, K. Curran, and J. Condell, “A survey of active and passive indoor localisation systems,” *Computer Communications*, vol. 35, no. 16, pp. 1939–1954, 2012.

[15] C. Langlois, S. Tiku, and S. Pasricha, “Indoor localization with smartphones: Harnessing the sensor suite in your pocket,” *IEEE Consumer Electronics Magazine*, vol. 6, no. 4, pp. 70–80, 2017.

[16] M. Shchekotov, M. Pashkin, and A. Smirnov, “Indoor navigation ontology for smartphone semi-automatic self-calibration scenario,” in *2019 24th Conference of Open Innovations Association (FRUCT)*. IEEE, 2019, pp. 388–394.