IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 October 2022, accepted 27 October 2022, date of publication 31 October 2022, date of current version 7 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3218301

RESEARCH ARTICLE

A Framework for Indoor Positioning Including Building Topology

ANDREA BRUNELLO, ANGELO MONTANARI, AND NICOLA SACCOMANNO
Department of Mathematics, Computer Science and Physics, University of Udine, 33100 Udine, Italy

Corresponding authors: Nicola Saccomanno (nicola.saccomanno@uniud.it) and Andrea Brunello (andrea.brunello@uniud.it)

The work of Angelo Montanari and Nicola Saccomanno was supported by u-blox AG, for the projects ''Towards a uniform modeling framework for fingerprint-based positioning'' and ''Advanced solutions for indoor positioning,'' respectively.

ABSTRACT In many application domains, position information is of fundamental importance. However, unlike the case of outdoor positioning, producing an accurate position estimation in the indoor setting turns out to be quite difficult. One of the most common localisation strategies makes use of fingerprinting. Research in this area has been faced with a number of challenges, leading to the proposal of a number of localisation algorithms, sampling strategies, benchmark datasets, and representations of building information. This proliferation made the modeling of the indoor positioning domain quite hard from both a theoretical and a practical point of view. In this paper, we propose a general and extensible framework, based on a relational database, that pairs fingerprints with building information. We show how the proposed system successfully deals with a number of problems that affect indoor positioning, supporting a large set of relevant tasks. The source code of the framework is available online, as well as an implementation of it, that provides an interactive open repository of indoor positioning data.

INDEX TERMS Building topology, fingerprinting, indoor positioning, multi-sensor data, relational databases.

I. INTRODUCTION

Position information plays a major role in everyday life. Many applications rely on it including, e.g., those in the fields of logistics, navigation, access control, and emergency response. In the outdoor setting, high-precision localization can be achieved thanks to the Global Navigation Satellite System (GNSS). Nevertheless, as pointed out in many papers (see, e.g., [1]), people spend most of their time indoors, where the localization task is much more difficult due to signal perturbation, masking effects, and lack of standards [2], [3]. Several approaches to indoor positioning have been proposed in the literature. Among them, fingerprinting is by far the most common one [4]. It consists of two phases. During the first one (*offline* phase), a survey of the considered site is performed, aimed at the collection of information recorded by different sensors, like WiFi, cellular, and Bluetooth signals, and GNSS and inertial measurement unit (IMU) data. The sampling is done at different locations, either predefined or casual, and it produces a collection of tuples *(location, sensed data)*, called *radio map*. In the second one (*online* phase), a user samples data at an unknown location. The observations are then compared against those stored in the radio map, and an estimate of the user's position is obtained by executing a specific algorithm, such as, e.g., (k-)Nearest Neighbor [5].

The lack of standards, the high heterogeneity of collected data, and the inherent variety of indoor premises are major challenges for the development of indoor positioning solutions. This led to an extensive research effort that resulted in the proposal of several localization algorithms [2], [4], [6], [7], [8], [9], sampling strategies [10], [11], [12], benchmark datasets [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], and building modeling approaches [20], [21], [22], [23], [24].

On the positive side, such an effort resulted in a deeper understanding of indoor positioning and the achievement of reasonably accurate estimations. On the negative side, the research has been faced with a number of issues, including: *(i)* the difficulty in retrieving a comprehensive collection of datasets for the experiments; *(ii)* the need of reconciling data representations using heterogeneous formats and

The associate editor coordinating the review of this manuscript and approving it for publication was Kegen Yu.

conventions; and, *(iii)* the problems in comparing existing contributions, that used the same datasets in different ways, e.g., with respect to the training/test split selection. The matter is even more serious in an industrial setting, where the design and deployment of a positioning system involve solving non trivial and time demanding tasks such as determining the extent of information that should be modeled and the best way to accommodate needs that may change over time. Last but not least, there are cross-cutting themes which are halfway between research and industry as the effective modeling and integration of advanced elements, like, for instance, device trajectories and information about the topology of buildings. These data can contribute significantly to the realisation of state-of-the-art positioning systems; however, combining them in a uniform framework, taking into account all the aforementioned aspects, is not trivial at all.

The present work lies at the intersection of all the above dimensions: we propose a comprehensive, yet general and extensible, framework that poses as a tool easily adoptable by both the research community and industrial practitioners, which allows one to jointly handle fingerprint and building information. We first provide an abstract model of the considered domain, and then we turn it into a concrete relational database. A relational database stores information by means of fixed-length records, that are collected within a set of tables. Operationally, the development of a database begins with the definition of a conceptual schema, typically formalized by means of the Entity-Relationship (ER) modelling language, which is used to explicitly represent all domain requirements. The ER schema is then translated, using well-established mapping rules, into a logical schema, containing the definition of the tables in terms of attributes and their domains, and constraints, including primary and foreign keys. Finally, the tables are implemented into a physical RDBMS (Relational DataBase Management System) instance by making use of suitable SQL (Structured Query Language) instructions [25]. The choice of relying on such a DBMS is based on a number of reasons, including: *(i)* its ease of deployment into an industrial setting, thanks to the widespread mastery of this technology, *(ii)* the availability of *SQL*, an easy-to-learn language that supports a user-friendly interaction with the system, *(iii)* the existence of a streamlined design process, from the conceptual design to the physical implementation of the database, and, *(iv)* the possibility of natively handling domain constraints, so as to guarantee data quality requirements.

The source code needed to deploy the proposed solution is available online [26]. In addition, we provide access to an implementation of the system [27], already populated with data coming from well-known indoor positioning datasets, that demonstrates the full potential of the proposed solution; the idea is that of evolving over time the latter system into a centralised, open repository of indoor positioning data available to the research community. Overall, we believe that the combination of the above two elements fosters the wide adoption of the framework: a company or a research group can first become familiar with its online implementation, at no cost; then, the source code provides a quite straightforward manner to set up a production-ready running local instance. In addition, as we will see, the proposed solution does not force any strong constraint on the type of (fingerprint-based) localization system to possibly employ on top of it. Thus, it poses as a general backbone approach to support indoor positioning, capable of providing a clear, structured, customizable, and unified interface to access and exchange information [28].

The paper is organised as follows. Section II discusses the motivations and challenges of the work. Section III outlines an Entity-Relationship conceptual schema of the proposed framework. The logical schema of the database is given in Section IV. Section V illustrates some notable use case scenarios. Section VI briefly analyses related work. We conclude the paper with some final remarks.

II. MOTIVATIONS AND CHALLENGES

In fingerprint-based indoor positioning, modeling is commonly recognized as a very complex activity (see, e.g., [29]). In this section, we discuss the most pressing issues we addressed in developing the proposed framework, that aims at supporting all tasks involved in the offline and online phases of a localization system.

The main challenge is the intrinsic dynamic nature of the domain. As an example, in WiFi fingerprinting, access points may be added, removed, or replaced. As for cellular data, mobile cells can be merged or relocated [30]. In addition, new wireless technologies are continuously being developed. Even the indoor premises themselves can be modified in their arrangement and architectural characteristics over time. Alongside, not only the sources of information may change, but also the way in which they are perceived and recorded by devices. This is true for both the kinds of sensor they are equipped with and the effectiveness of their sensing capabilities. As a consequence, information stored in radio maps undergoes constant evolution: new fingerprints are added, and old ones are updated or even discarded. For these reasons, the framework must be designed to grant ongoing and long-term support to the collection and maintenance of radio map fingerprint data.

A second aspect pertains to the high heterogeneity of the domain. As shown in the literature [22], an indoor scenario can be described at different levels of detail. Determining the right abstraction level is not trivial, as it involves reasoning over the possible kinds of premises and the topological relationships among them, including reachability aspects. The heterogeneity also applies to fingerprint data under two dimensions. First, fingerprints may consider several types of signal source, such as WiFi, Bluetooth, cellular information, data from inertial sensors and GNSS receivers, as well as their combinations. A well-designed solution should offer comprehensive support to multiple kinds of observation, and allow for an easy extension to new ones. Second, fingerprints may be collected according to several strategies, e.g., following a well-planned survey plan, or relying on a crowdsourced effort. In addition, different sampling strategies may coexist within a given premise, e.g., as a result of repeated survey campaigns performed over the same building.

This brings to light the issue of information sharing: if the same object appears in more than one dataset, it makes sense to store its data only once in the system, providing pointers to the original sources so as to maintain data lineage. This may be the case, for instance, with a room, a mobile network cell, or an access point, that has been considered in more than one study. From a practical point of view, this may also increase the overall amount of available information. Let us assume that, within the same building, on two adjacent floors, two independent positioning systems are deployed. Suppose that they sense WiFi data, and since they are close, they might be based on the same access points (detected through the ceiling/floor). However, as the two systems are fully disjoint, there is no way to combine the data related to one floor and those related to the other. Organising data by means of the proposed framework allows one to recognize that the same access points are used by both systems, and to fruitfully exploit such a knowledge, e.g., by producing a richer radio map. In fact, information sharing goes beyond sensors/emitters as, for instance, it may also support the combination of data about the topology of a building.

It is worth pointing out that the problem raised by redundant, and possibly inconsistent, information is not only tied to the combination of several data sources. Such inconsistencies may, indeed, be already present within a single dataset: modeling topological aspects is quite complex, and requires the knowledge of and the adherence to a large set of domain constraints. As an example, a floor should not - typically - be contained into multiple buildings. Thus, the framework must also provide a simple and uniform manner to enforce such constraints, so as to guarantee data quality requirements.

The remaining issues pertain to the online usage of the system. In the most general case, a user may submit a single fingerprint to a positioning system to obtain a position estimate. Thus, a first question is how fingerprints can be related to information about the structure of buildings to support localization. The problem becomes more complex when the positioning algorithm relies on a sequence of fingerprints, that is, a trajectory, arising from user navigation within the premises, as dealing with both single points and trajectories is far from being simple. Finally, the framework should be able of supporting the (possibly concurrent) usage of multiple prediction algorithms, which may generate different outputs. As an example, the exact position coordinates may be estimated, or the fingerprint can be matched to a single logical location, like a room, as well as to multiple ones, e.g., by means of a probability distribution.

As we will see, all the above issues have been taken into account in the design of the proposed solution.

III. DOMAIN MODELING

In this section, we provide a high-level modeling of the considered domain. First, we propose a way to represent information about indoor premises that can be easily paired with positioning data. Then, we integrate the resulting model with all the aspects that are relevant to a positioning system by means of an Entity-Relationship diagram, which is the cornerstone of the relational database at the core of the proposed framework.

A. PAIRING INDOOR TOPOLOGY AND FINGERPRINTS

To represent information about indoor premises, we rely on a relatively simple representation, that allows one to describe the topology of indoor environments without explicitly encoding elements like walls, windows, and objects in the rooms. The guiding principle is that of building a system as general as possible so that researchers and practitioners may use it regardless of the adopted fingerprinting methodology and with only a few, possibly none, data about the premises.

To this end, we conceptually model the indoor setting as a heterogeneous (directed) graph. This choice is supported by at least three arguments: *(i)* in the indoor setting, there are different types of element, like buildings, floors, and rooms, which have different properties and, thus, are better modeled by different types of node, *(ii)* there are relationships among elements of the same type which are worth modeling, e.g., the relations of adjacency or walkability, that is, adjacency + traversability, over places, and *(iii)* there are relationships among elements of different type which are worth modeling as well, like, e.g., the intrinsic hierarchical organisation of indoor environments (a building consists of multiple floors, which are in turn composed of multiple rooms). Such a modeling is highly flexible: it makes it easy to capture relevant topological properties that may be useful for positioning purposes, while allowing to customize the detail at which to encode the structure of a given indoor scenario (information about most of the relationships is optional). In the following, we describe the characteristics of each element of the model as well as their relationships.

Building is the top object of the hierarchy. It can be adjacent to other buildings and contains a set of floors. We consider a building as a structurally independent element of the modeled domain, e.g., a separate construction possibly connected to others by indoor elements, such as bridges or underground tunnels. Each *floor* can be related to other floors to model the vertical ordering among them. Since the vertical dimension is fundamental in indoor positioning and navigation, information about it must be as sound and complete as possible. To this end, it is worth including in the model information about all the floors of a building, even if they are not explicitly involved in the positioning process. This is not a mandatory requirement, although the number of floors in a building is generally easy to obtain. A floor may consist of various elements, such as rooms, stairs, elevators, corridors, and so on, that we collectively name sites. A *site* is (a portion of) an indoor environment that one has decided to model explicitly. The core element linking position information (*fingerprints*, described next) and topology is the *tile*. A set of tiles in a floor is the result of a tessellation procedure, which defines

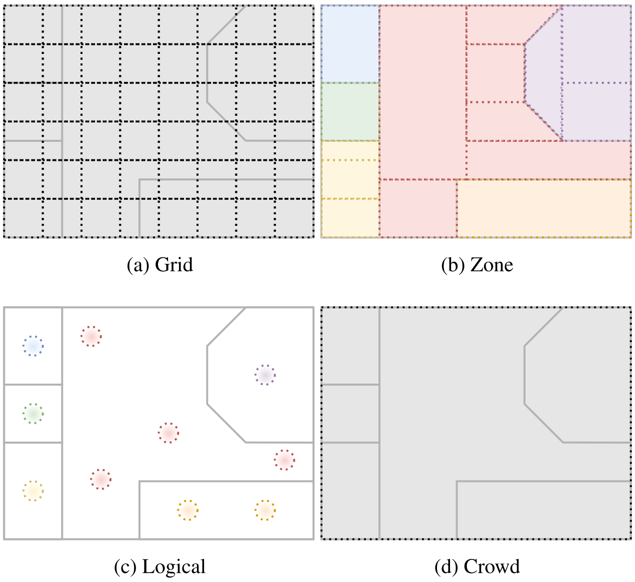


FIGURE 1. **Different tessellations for an indoor scenario. Dashed lines denote tiles; Tiles of the same colour refer to the same parent place (e.g., a site or a floor) in the hierarchy.**

the granularity at which the positioning task is performed. Different types of tessellation are possible.

Grid: a fixed size regular grid is superimposed on the floor map, generating a set of tiles. Here the notion of tile is not associated with a site, since a grid-based partitioning does not take these pieces of information into account, and a grid cell can cross site boundaries. Thus, a grid-generated tile is linked directly to a floor (Figure 1a).

Zone: an irregular grid is defined to partition a site or a floor. Each of the resulting areas is a tile. Each tile can be associated with at most one site or floor, and its coverage area can be arbitrarily large (Figure 1b).

Logical: each tile can be considered as a semantic label associated with a meaningful location of the considered site or floor (Figure 1c); the tile has no geometrical shape, although it can optionally be characterised by a single pair of coordinates identifying a specific point in space for instance, the geometrical centre of a considered site, or a point of interest.

Crowd: a single tile, devoid of geometrical data, is directly associated with a floor, acting as a general container of fingerprints in order to support crowdsourcing tasks (Figure 1d).

The relationships of adjacency and walkability over the set of places are encoded at the tile level, except for the building case, that relies on an ad-hoc relationship. The latter allows us to model the adjacency between buildings that are structurally separated and not connected in any manner, without making use of "outdoor" tiles. On the basis of information about tiles, it is possible to derive complex notions, such as paths (possibly traversing different floors) and adjacency relationships between rooms and corridors, according to the level of granularity of the chosen representation. A graphical account of an indoor scenario modeled by means of the proposed heterogeneous directed graph is reported in Figure 2.

Turning to the *fingerprint* data, we must bear in mind (see Section I) that a fingerprint can be acquired during both the offline and the online phase of a positioning system. In the first case, it may be collected either via a planned survey conducted by experts or by users in a crowdsourced fashion. Depending on the collection modality, position information associated with a fingerprint may take on different forms. In the most standard setting, the position of a fingerprint is given as a vector of coordinates in a given reference system. In addition, a fingerprint can be logically associated with a specific element of the indoor environment, such as a floor, a site, or a portion of it, through the notion of tile.

Here, it is worth noticing the different semantics associated with *grid*, *zone*, and *logical* tiles and with *crowd* tiles. In the first case, each tile groups fingerprints related to a specific, predefined area. This is the case, for instance, when multiple fingerprints have been sampled for the same location. In the second case, a tile contains fingerprints collected from a given floor without any specific constraint. Note that, although *crowd* and *logical* tessellations may look similar, they play a very different role: the only way to model a crowdsourced scenario by a logical tessellation would be that of generating a distinct logical tile for each collected fingerprint, which is both counterintuitive and inefficient.

Finally, we also deal with the case in which no position information at all is associated with a fingerprint. This happens, for instance, with surveys where data gets labeled at a later stage, or when fingerprints are collected and exploited in an unsupervised fashion, or simply when the fingerprint has been collected during the online phase, e.g., no position estimation algorithm has been applied yet, or its results have not been stored in the system.

As a last remark, we observe that tessellations serve a dual purpose: on the one hand, they define a fingerprint collection strategy; on the other, they allow one to model topological relationships. As an example, a given floor may rely on a *crowd* tessellation for the former purpose, and on a *zone* tessellation for the latter one.

Overall, the proposed model is highly flexible with respect to the type and the amount of data that a user is required to add, especially with respect to building topology. As an example, little to none information may be inserted about the latter, e.g., only the labels representing the building and the floor, and a crowdsourced approach can then be followed for the radio-map construction, with a minimum implementation effort. On the contrary, a person could be interested in modeling the indoor topology in its full detail: in such a case, complete information about the building components and their relationships can provided, and a fine grade tessellation, using logical or zone tiles, can be adopted.

To conclude, we would like to observe once more that we do not provide any means to model furniture and similar objects, as we aim at providing an account of building topology and integrating it with positioning data.

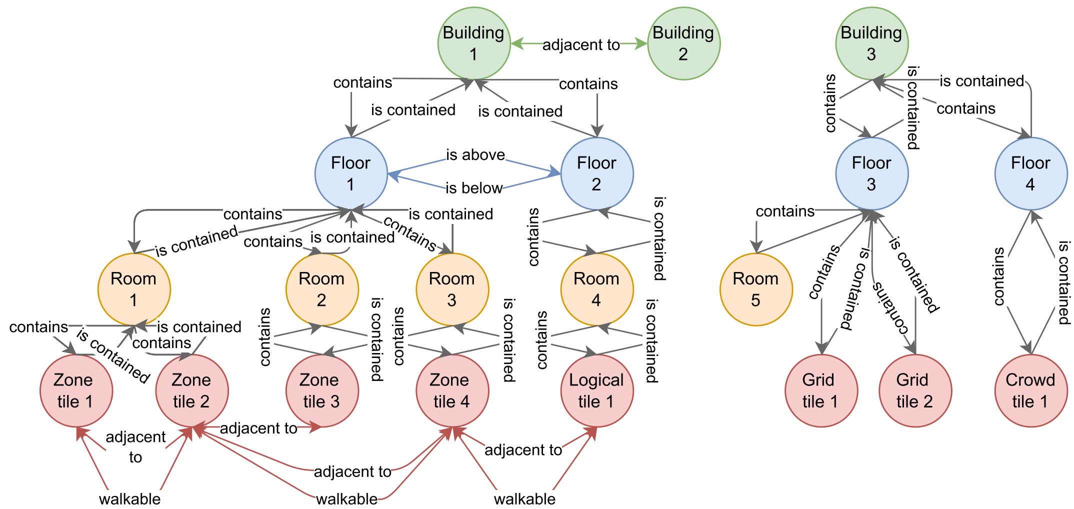


FIGURE 2. Proposed heterogeneous directed graph modeling a generic multi-building, multi-floor scenario using different types of tessellation.

B. OVERALL CONCEPTUAL SCHEMA

The Entity-Relationship diagram of Figure 3 provides a conceptual representation of information of interest about fingerprints, building topology, and their connection.

The schema consists of four distinct sub-schemas, each one focused on a specific portion of the domain, namely:

- the *Data source* sub-schema, which is responsible for preserving the data lineage, that is, it keeps track of the original sources of the data (fingerprints, observations, places) inserted in the database;
- the *Fingerprint* sub-schema, that records information about the fingerprints acquired by users/devices. Each fingerprint is collected at a given place and holds a set of observations, that is, the actual measurements performed by the device at the specific place. In addition, this sub-schema supports position estimation tasks;
- the *Observation* sub-schema, that stores detailed information about the data sensed by devices, which can be of various forms, like, e.g., the received WiFi or Bluetooth signal);
- the *Place* sub-schema, that models topological information about indoor scenarios, used for fingerprint collection and positioning tasks.

As we will see, the proposed design is general and flexible enough to be used in several contexts and scenarios.

1) DATA SOURCE SUB-SCHEMA

This sub-schema provides data lineage capabilities, that is, it allows one to track the original data sources of all pieces of information stored in the system. Each *Data source* is uniquely identified by a *name*, e.g., the name of the original dataset collecting the data (for instance, UJIIndoorLoc [10]). Optional information include the *URL* of the original dataset and some free textual *Notes*. Several many-to-many relationships connect the data source to other schema entities, namely, *Cell_source*, *AP_source*, *BT_source*, *Device_source*, *User_source*, *Fingerprint_source*, and *Place_source*. In such a way, for instance, we can record information about the relationships between a data source and several access points. *Data_source* has an optional participation to all such relationships (for instance, no access point may be present in the dataset), except for *Fingerprint_source* and *Place_source*. In a given data source, indeed, at least information about the fingerprints and the premises where they were collected must be present.

2) FINGERPRINT SUB-SCHEMA

This sub-schema stores information about fingerprints. Each *Fingerprint* has a *Code*, which uniquely identifies it within its data source, that is, *Fingerprint* is a weak entity with respect to *Data source*. In addition, it features the attribute *Timestamp*, that records the date and time at which the fingerprint was collected, the optional attribute *ML_purpose*, that encodes the intended use of the fingerprint in the original dataset, i.e., training, validation, or test purposes, and, possibly, some free textual *Notes*. A fingerprint can be preceded or followed by another fingerprint, as in the case of trajectories. This piece of information can be expressed by means of the relationship *Follows*. Some of the fingerprints may belong to the radio map, which is modeled through a partial specialization. For fingerprints belonging to the radio map, we store the following data about position information: the optional coordinates *X*, *Y*, and *Z* of the point of collection (*Z* is in its turn optional, as in many datasets only 2D spatial coordinates are considered); and the *Tile* where the fingerprint was acquired.

Each fingerprint is collected by means of a single physical *Device*, although information about it may be not known. A device is uniquely identified within a dataset by its *Code*. With a device we may associate further information which is stored within *Device model*, including *Developer* and *Name*,

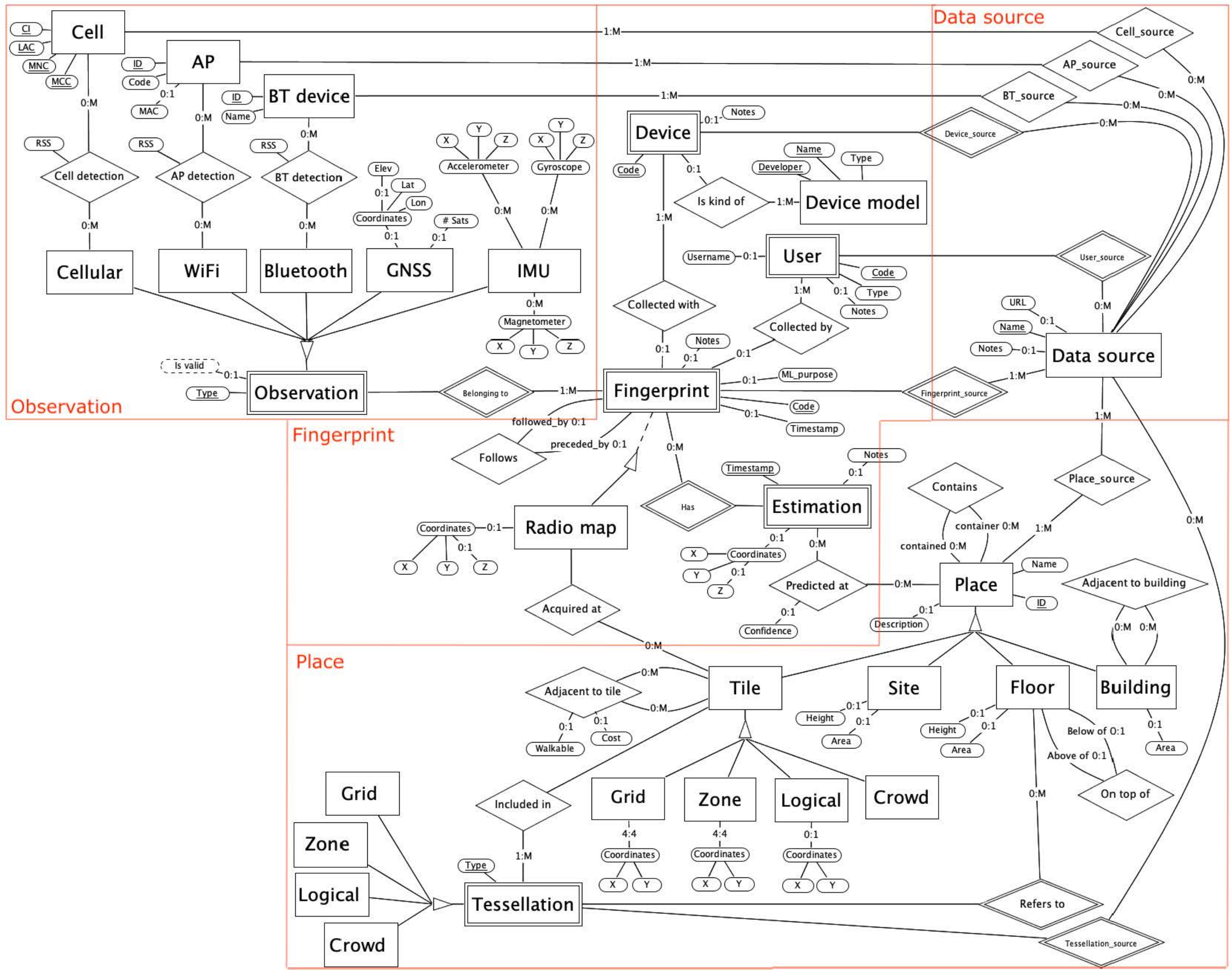


FIGURE 3. **The Entity-Relationship diagram. The notation is very close to the one originally proposed by Chen [31].**

that together identify a device model, and *Type*, e.g., smartphone, tablet, or other. As an example, in the dataset there may be two distinct smartphones, characterized by codes *A* and *B*, both of the same device model *Samsung S22*.

Each fingerprint is collected by a *User*. As it happens with the device, information about the user may be missing. A user is uniquely identified within a dataset by its *Code*, and characterized by its *Type*, e.g., *trusted*, for known users that assembled the radio map, or *online*, for those using the system for localization purposes. In addition, a user may have a *Username*.

To take into account the online usage of the positioning system, we introduce the entity *Estimation*. A fingerprint may be related to zero or more estimations, e.g., produced by different algorithms. In turn, an estimation is linked to a specific fingerprint, and may be discerned from the others that are associated with the same fingerprint thanks to its *Timestamp*. Each estimation may have some free textual *Notes*, and at least one among the following position data: the coordinates *X*, *Y*, and (possibly) *Z* of the point of prediction; and the *Place*(s)where the fingerprint was predicted. Observe that we can associate more than one place with a given estimation: this is the case with an algorithm that provides a probability distribution, where the single probabilities can be encoded by means of the optional attribute *Confidence*.

3) OBSERVATION SUB-SCHEMA

This sub-schema deals with information about observations, that is, the data sensed about the environment, that compose a fingerprint. A *Fingerprint* may be associated with one or more observations, and an observation refers to one and only one fingerprint. Each *Observation* has an attribute *Type*, that specifies the kind of observation among those included in the specialization. Since such an attribute is a partial identifier with respect to *Fingerprint*, the latter may have at most one observation for each type. In addition, an observation may have a *Validity* value, that indicates whether its data are still to be relied upon, e.g., for positioning tasks. Observational data may indeed lose their reliability over time. This is the case, for instance, with WiFi fingerprints, that collect information about the detected access points, which may be turned off or relocated. Since the validity of an observation can be computed from the related *Fingerprint*'s *Timestamp*, it is actually a derived attribute.

An observation may be of different kinds (total and disjoint specialization). Think of the fact that, for instance, a typical smartphone is able of simultaneously collecting several types of data. A *Cellular* observation consists of zero or more *Cells*, each one detected with its own *RSS* (Received Signal Strength). A cell is identified by the combination of *CI* (Cell Identifier), *LAC* (Local Area Code), *MNC* (Mobile Network Code), and *MCC* (Mobile Country Code). Similarly, a *WiFi* observation consists of zero or more *AP*s (Access Points), each one identified by an *ID* and characterized by a *code* and, possibly, a *MAC* number. Finally, a *Bluetooth* observation consists of zero or more detected *BT devices*, each one identified by an *ID* and characterized by a *Name*. Note that each cell, access point, and Bluetooth device may belong to one or more data sources. Again, this is quite natural, as the same cellular antenna may be detected in several scenarios. To accommodate for that, we introduced the surrogate key *ID* both for AP and Bluetooth devices. Such an attribute also allows us to discriminate, e.g., between two different APs that have been given the same *Code* in different datasets. The surrogate key is not necessary for *Cell*, since such an entity set already features a global identifier.

The modeling of GNSS and IMU data undergoes a different logic, as they do not store information about a signal pattern received from external beacons. Specifically, a *GNSS* observation has an associated optional set of latitude (*Lat*), longitude (*Lon*), and elevation (*Elev*) coordinates (the latter is optional, as it depends on the number of available satellites, *# Sats*). As for *IMU* observation data, they can refer to different kinds of device, typically hosted on a same module: *Accelerometer* information, tracking the acceleration along axes *X*, *Y*, and *Z*; *Gyroscope* information, detecting the orientation with respect to axes *X*, *Y*, and *Z*; and *Magnetometer* information, measuring the magnetic field for axes *X*, *Y*, and *Z*. Note that an IMU observation may have several sets of data associated with it. This is the case, for instance, with a single fingerprint which contains multiple accelerometer samples that have been sensed several times over a period of time. In this case, information is modeled in a differential manner with respect to a previous fingerprint in a trajectory. Thus, a fingerprint with an associated IMU observation should also participate into the relationship *Follows*, in order to keep track of the preceding one.

To conclude, we observe that the above design choice allows us to store empty observations related to a fingerprint. This is the case, for instance, with a cellular scan which detected no cells (this may happen when the device is in an underground location).

4) PLACE SUB-SCHEMA

This sub-schema models topological information about indoor scenarios. As previously pointed out, the intended goal is not that of storing extremely detailed data with which to reconstruct the exact appearance of the considered premises, but, rather, to keep track of information that may be useful for positioning purposes.

The main entity of the sub-schema is *Place*, which represents a generic spatial concept. Each instance of place is uniquely identified by a surrogate key *ID* (following the same reasoning pattern as that of *AP* and *BT device*), and characterized by a *Name* and, possibly, a *Description*. A place may belong to more than one data source (this is the case, for instance, with several datasets collected over the same premises at different times). A place is then partitioned (total and disjoint specialization) into *Building*, *Floor*, *Site*, and *Tile*. The relationship *Contains* allows one to keep track of a hierarchical structure among places.

Building represents the coarsest level of the hierarchy. Each building may have an associated *Area*, and can be adjacent to zero or more other buildings (relationship *Adjacent to building*). A building may include one or more floors.

A *Floor* may have an associated *Area* and a *Height*, and is contained in a single building. To preserve the vertical ordering of floors, we make use of the relationship *On top of*. It is worth pointing out that such a modeling decision requires one to specify all intermediate floors between any two given levels, in order to correctly and completely maintain the vertical relationships. However, such a constraint does not limit the flexibility of the model as if some floors are not present in a dataset, they can still be added as empty (dummy) levels, without any sites or tiles included in them. A floor may contain zero or more sites, or, directly, tiles, as in the case of grid and crowd tessellations.

A *Site* represents a specific spatial area in an indoor scenario, such as a room, a corridor, a bridge between buildings, or some stairs between floors. A site may have a *Height* and an *Area*, and may belong to one or more floors. An example of the second case is an auditorium, that may have entrances on two or more different floors. A floor being used for positioning purposes always contains at least one tile.

A *Tile* represents the most basic piece of spatial information that can be stored within the system, and it acts as a bridge between the topological knowledge of the premises and the fingerprints. In accordance with the discussion in Section III, a tile can be of four different types: (i) *Grid*, characterized by the four 2D *Coordinates* of its associated regular grid cell, (ii) *Zone*, characterized by the four 2D *Coordinates* of its polygon, (iii) *Logical*, possibly characterized by a pair of 2D *Coordinates*, and (iv) *Crowd*, with no other associated information. Information about adjacency is captured by the relationship *Adjacent to tile*, which may possibly track the walkability between tiles (attribute *Walkable*) and the associated traversing cost (attribute *Cost*).

Each tile is included in one and only one *Tessellation*, which should be of the same kind as that of the tile. The specialization of tessellation is total and disjoint, and thus a given tessellation contains only one kind of tiles. Note that, similarly to the case of *Observation*, *Tessellation* has an attribute *Type* as its partial identifier, and the entity set is weak with respect to *Floor* and *Data source*. The overall result is that, within a floor belonging to a given data source, we may have at most one tessellation for each kind. Thus, a floor may have a grid and a zone tessellation associated with it, each with its own tiles, but it cannot have two distinct grid tessellations.

IV. RELATIONAL DATABASE DEVELOPMENT

In this section, we focus on the development of the relational database for indoor positioning. In particular, Section IV-A discusses its logical schema, which has mainly been derived from the Entity-Relationship diagram of Section III-B. Then, Section IV-B deals with the problem of representing fingerprint ground truth information that can be useful for the evaluation of indoor positioning systems, an aspect which, as we will see, deserves specific attention. Although we do not describe here all the details regarding the physical implementation of the database, which include the definition of a set of data consistency constraints not directly expressible in the logical schema, the interested reader may find the source code on GitHub [26].

A. LOGICAL SCHEMA

The conceptual schema illustrated in Section III-B can be translated into a relational schema by applying the standard mapping rules [25]. The resulting schema, depicted in Figure 4, includes all the tables obtained via such rules, as well as some additional tables that were required to store, for instance, type information. It is worth observing that the relational counterparts of the four ER sub-schemas can be easily identified: *Data source*, *Fingerprint*, *Observation*, and *Place*. In the following, we will describe the most important choices we took during the logical design process. In general, all composite attributes were handled by keeping their components, and we tried to reduce the size of foreign keys as much as possible in order to avoid complex join conditions and unnecessary usage of space.

The translation of the Data source sub-schema does not present any particular problem.

In the Fingerprint sub-schema, the *Fingerprint* specialization was translated by keeping just the parent entity and adding the Boolean attribute *is_radio_map* to it. As for the attributes *coordinate_x*, *coordinate_y*, *coordinate_z*, and *acquired_at_tile_place_id*, they pertain just to those instances belonging to the radio map (*is_radio_map* = *True*). The attribute *ml_purpose* has a dedicated domain, just including the strings *'training'*, *'validation'*, and *'test'*. As for the primary key, we introduced the surrogate *id*, which allows us to use simpler foreign keys when referring to fingerprint instances. Then, to enforce data consistency, we placed a uniqueness constraint over the pair of attributes *code* (the fingerprint identifier in the original dataset) and *data_source_id*. Similar considerations were made for the primary key of *estimation*, where we introduced the surrogate *id* and imposed a uniqueness constraint over the pair of attributes *timestamp* and *fingerprint_id*. In the table *device*, we find the usual surrogate key *id* and a uniqueness constraint defined over the pair of attributes *code* and *data_source_id*. Information about the device model and its type was recorded by means of two dedicated tables, to avoid unnecessary data replication and to allow for an easy extension of their allowed values. The same logic was followed for the table *user*, with the uniqueness constraint placed over the pair *code* and *data_source_id*, and the provision of a dedicated table used to store the user type.

Turning to the Observation sub-schema, we translated the *Observation* specialization by keeping just the children entities. In the table *cell*, we introduced the surrogate key *id* to avoid to deal with foreign keys consisting of four attributes. Multi-valued attributes belonging to the entity *IMU* were handled by introducing three separate tables, each one with the pair of attributes *fingerprint_id* and *epoch* as the primary key. The latter attribute can be used to sort multiple data coming from the same sensor within a single *IMU* observation. Being numeric, it can be used both as a kind of timestamp (thus implicitly conveying information about the sensor sampling frequency) or as a simple ordering integer. As for the domains of the attributes, *rss* stores negative numerical values corresponding to the decibel-milliwatts (*dBm*) of the received signals. Along each axis, acceleration in *observation_imu_accelerometer* is measured in metre per second squared (m/s^2), angular velocity in *observation_imu_gyroscope* is encoded in radian per second (rad/s), and magnetic field intensity in *observation_imu_magnetometer* is recorded in microtesla (μT).

As for the Place sub-schema, we kept all entities involved in the *Place* specialization, given the presence of several distinct relationships involving them. As for the specialization of *Tile* and *Tessellation*, we instead kept just the parent entity and relied on the attribute *type*. The latter has the same custom domain both when used in the table *tile* and in the table *tessellation*, consisting of just the strings *'grid'*, *'zone'*, *'logical'*, and *'crowd'*. In the table *tile*, we explicitly listed all four pairs of coordinates. Such a decision allows us to have fine-grained control over the consistency of their usage. In the table *tessellation*, we introduced the surrogate key *id*, and we placed a uniqueness constraint on the triplet of attributes *type*, *floor_place_id*, and *data_source_id*. As for the attribute types, *height* and *area* are encoded in meters and square meters, respectively.

B. GROUND TRUTH INFORMATION MANAGEMENT

We conclude the section with a short note on the management of fingerprint ground truth information. In contrast with our design choices, indeed, it may happen that a non-radiomap fingerprint is associated with some partial

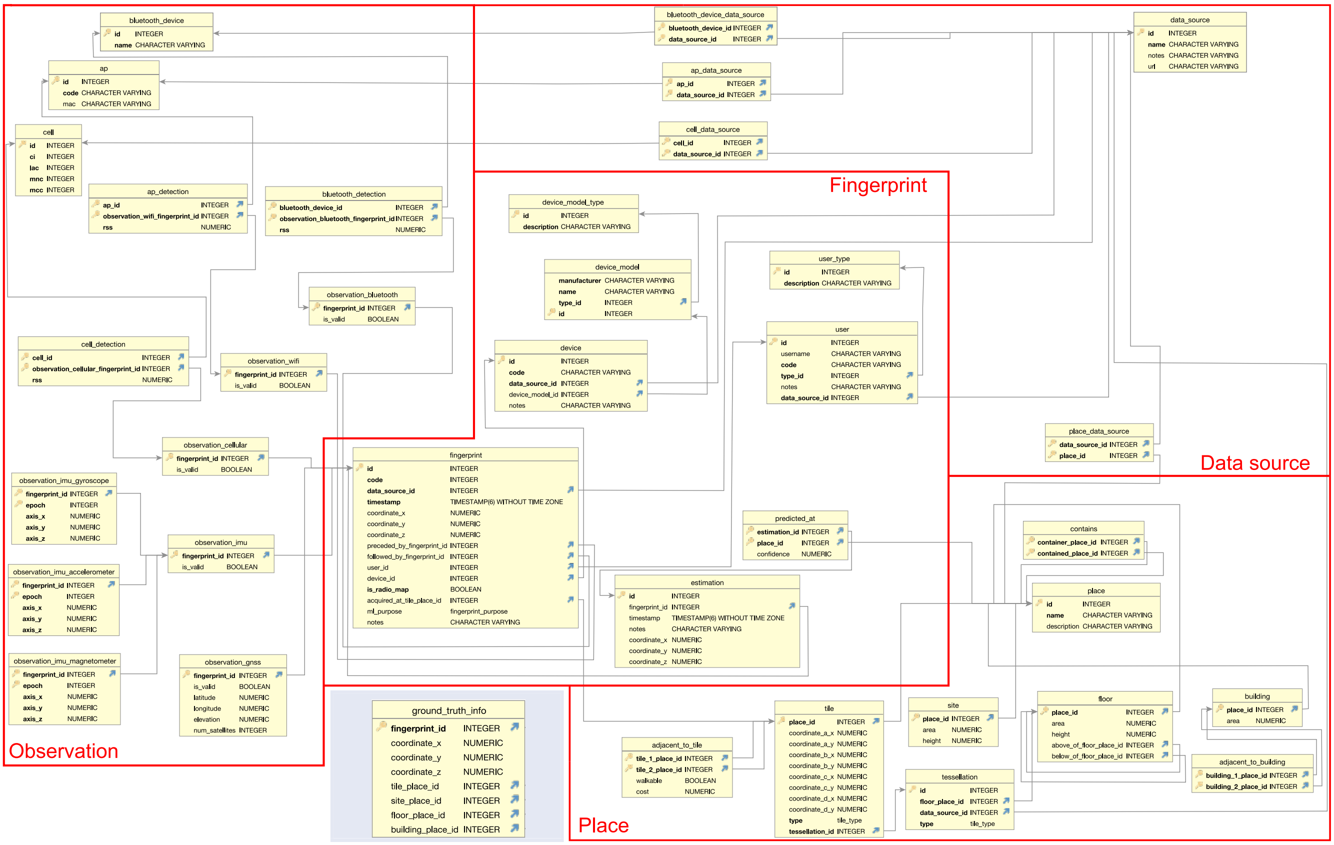


FIGURE 4. **Logical schema of the indoor positioning relational database. Arrows represent foreign key directions. The red sub-schemas make up the *public* schema. The blue area depicts the *evaluation_support* schema (consisting of one table only).**

spatial information. As an example, this is the case with test set fingerprints in the dataset UJIIndoorLoc [10] that, although being considered as ''online'' fingerprints, still possess ground truth coordinate data, but are not associated with their tile or site of collection. Since information of this kind may be useful for the evaluation of indoor positioning systems, we decided to accommodate it into a separate schema, called *evaluation_support*. Within such a schema, we defined the table *ground_truth_info*, depicted in Figure 4, blue shaded area. No constraint on the absence of null values was enforced on the table, and thus, given a fingerprint, it allows us to store any kind of (possibly fragmentary) spatial data associated with it. Primary key attribute *fingerprint_id* is a foreign key with respect to the attribute *id* of the table *fingerprint*, while attributes *tile_place_id*, *site_place_id*, *floor_place_id*, and *building_place_id* are foreign keys pointing to the attribute *place_id* of tables *tile*, *site*, *floor*, and *building*, respectively. Such a table is particularly useful to foster reproducibility in indoor positioning experiments, as it allows one to collect and highlight the data which should be used for evaluation purposes.

V. USAGE OF THE SYSTEM

In this section, we first show how the proposed model is flexible enough to accommodate various indoor scenarios. Then, we discuss how the system can support non trivial tasks in the indoor positioning domain, other than promoting the research on and the deployment of novel localization approaches.

The source code of our implementation, which includes the definition of some useful SQL queries and user defined functions (UDFs), is available on the GitHub page of the project [26]. Examples of UDFs that easily allow one to retrieve complex information are `MinimumShortestPath` (computed for pairs of heterogeneous elements, e.g., a tile and a floor, or a fingerprint and a room), `FPDistances` (measuring the distance between two fingerprints or estimations with several metrics), and `CharacterizingFP` (that determines the average fingerprint of a tile).

An online, freely accessible [27] implementation of the system has been developed in PostgreSQL [32]. Users may submit custom queries through the PGAdmin (web) interface [33] as well as relying on well-known database connectivity APIs (e.g., JDBC [34]). The database already stores more than 15 well-recognized datasets for indoor positioning [10], [11], [12], [13], [15], [16], [17], [18], [19], which can be retrieved and used as pleased. We remind all the potential users of the tool to give proper credit also to the original collectors of the datasets.

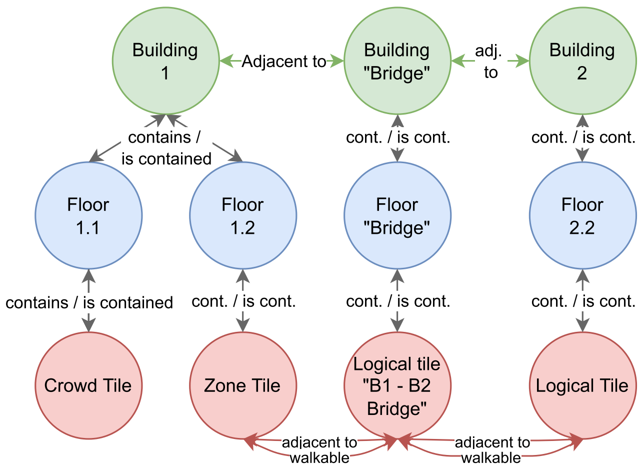


FIGURE 5. **Inter-building connectivity modeling example (for simplicity, only one tile per floor is depicted).**

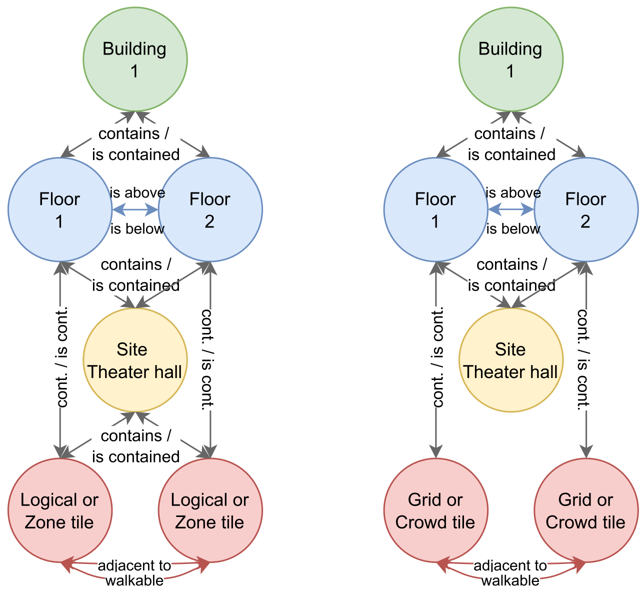


FIGURE 6. **Premises spanning over several floors (theater hall) modeling example. Left-hand side denotes the case of logical or zone tessellations; right-hand side reports the case for crowd or grid ones (for simplicity, only one tile per floor is depicted).**

A. REPRESENTATION OF NOTABLE INDOOR SCENARIOS

In this section, by means of a series of use cases, we show how the flexibility of the proposed model allows us to represent various indoor scenarios.

1) LOGICAL TILES

Assume that we want to perform a simple tessellation of a given floor of a conference building, featuring different rooms, without resorting to the definition of complex grid or zone tiles. In such a case, we opt for a much lighter logical approach. A first solution might be that of associating with each of the rooms its logical tile, without any pair of coordinates. As an example, we may have a single logical tile representing "conference room A", acting as a purely logical link between the place (for which the semantics is defined by the tile label) and the fingerprints associated with it. However, if conference room A is quite large and has a stage far from the audience space, it might be sensible to further refine our tiling approach, for instance, to support more precise positioning tasks. We can thus specify two distinct logical tiles, the first one representing the audience space, and characterised by a pair of coordinates equal to the geometrical centre of such a space, and the second one tied, for instance, to the coordinates of the main tribune on the stage.

2) INTER-BUILDING CONNECTIVITY

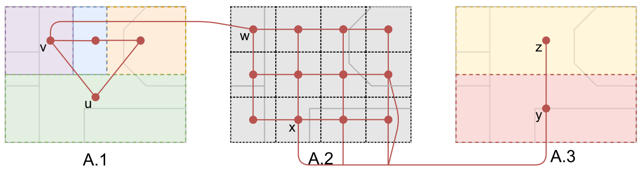
Let us consider two separate buildings connected by a bridge-like structure. The latter can be modeled as a building on its own, with a single floor, and some tiles adjacent and walkable with respect to tiles of the other two buildings. This solution can be easily generalized to a set of buildings connected by a network of underground tunnels, possibly structured on more than one level. A possible model of such a scenario is depicted in Figure 5.

3) PREMISES SPANNING OVER SEVERAL FLOORS

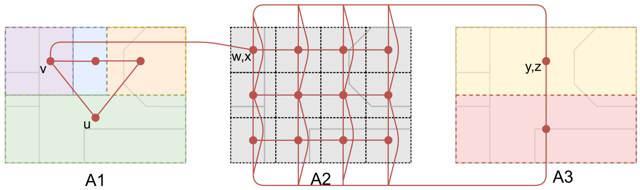
A theatre hall may have entrances on different floors. Here, also the tiles representing the hall belong to different floors. This is true for the tessellations of type *zone* and *logical*, where the tiles are connected to the *site* that corresponds to the hall (which in turn belongs to more than one floor), as well as for the tessellations of type *grid* and *crowd*, where the tile(s) are paired directly with the floors spanned by the hall. A possible representation of the resulting scenario for both cases is given in Figure6.

4) CONNECTIVITY INVOLVING DIFFERENT TESSELLATION STRATEGIES

Let $A.1$ and $A.3$ be two zone-tessellated floors of a building A, interleaved by a floor $A.2$ that follows a grid tessellation approach. Despite the heterogeneity of tessellations, it is still very easy to model adjacency and walkability relationships among tiles, so as to represent a possible path starting from a tile u on floor $A.1$ and ending at a tile z on floor $A.3$. More precisely, the path would start from the tile u; then, it would follow the walkability relationships among zone tiles till it reaches a tile v which is walkable with respect to a grid tile w belonging to the floor $A.2$. In the most general case, each grid tile belonging to the floor $A.2$ is adjacent (and walkable) to each of its neighbouring tiles, and thus the walkway would then follow a shortest path till a given tile x, vertically connected to a tile y of floor $A.3$, and finally reach the tile z according to the relationships defined over the zone tiles of floor $A.3$. An example of topological information for such a scenario is reported in Figure 7.



(a) Grid tiles have only specific inter-floor connections



(b) All grid tiles are connected with those physically adjacent

FIGURE 7. Two connectivity scenarios involving different tessellation strategies across three different floors. Floors *A*.1 and *A*.3 use zone tiles, while floor *A*.2 uses grid tiles. Red links denote adjacency and walkability relationships at the tile level (some have been compacted for simplicity).

5) CROWDSOURCED SAMPLING

A crowdsourced collection of fingerprints performed on a floor can be supported by defining a tessellation of type *crowd* on it, consisting of just a single tile that acts as a general container. Then, collected fingerprints would all be connected to such a tile, and possibly be complemented by information about their position coordinates.

B. SUPPORT FOR INDOOR POSITIONING TASKS

The proposed modeling of the indoor positioning domain makes it possible to support a large array of interactions, ranging from very simple queries to rather advanced use cases. As we did in the previous section, we introduce some notable use case scenarios that demonstrate the potential of the system.

1) MULTI-SOURCE COMPOSITIONALITY

Starting from the information stored within the *Place* sub-schema, it is rather easy to extract knowledge about the structure of an indoor positioning scenario in terms of, for instance, composition and adjacency relationships among sites, floors, and buildings. In doing that, information coming from different data sources can be merged (while still preserving data lineage) in order to obtain a complete picture of an indoor scenario, that may have been considered in a fragmented fashion within different studies. Information coming from multiple sources can be exploited also with respect to fingerprints. As an example, different radio maps pertaining to the same places can be merged to provide a higher number of (possibly temporally updated) observations, that can be used during the online phase of the system. In this respect, Figure 8 shows, for two datasets collected at the same premises one year apart from each other, how the available APs and their average RSS may change dramatically.

2) MULTI-SENSOR POSITIONING

Following the proposed approach, it is quite easy to combine WiFi and Bluetooth signals, which are the most useful information sources for fingerprinting. Such sensors, together with other point-based ones, are efficiently managed even if they are sampled at different rates and in the presence of IMU data and trajectories. This one allows to seamlessly exploit a rich set of information for positioning purposes.

3) MORE ROBUST LOCALIZATION

Given a fingerprint submitted by a device, it is possible to filter the radio map so as to consider only fingerprints collected by similar devices or within a specific time window. In principle, such an ability has the potential to improve positioning performance, since devices may sense and record observation data in different manners, due to hardware or software differences. Figure 8b shows, for the UJIIndoorLoc dataset, how different devices indeed exhibit, for the same location, different average RSS patterns. In addition, focusing on WiFi fingerprinting, signal propagation is likely to vary in a cyclical fashion over a day or a week, due to the distribution of people within premises, which may have a perturbation effect on the signals. Restricting the attention to fingerprints recorded within specific time windows is also likely to allow for a more sensible comparison among observation data.

4) COMPLEX AND MORE EXPRESSIVE METRICS

Storing topology information supports the development and usage of error metrics for indoor positioning that are more advanced than the commonly considered 2D distance. As an example, predicting the location of a user with a 3 meter error radius is way more serious if this brings to an uncertainty over the floor the user belongs to, with respect to an uncertainty over the position of the user within a single room. To account for that, reachability information stored by means of the *adjacent_to_tile* relation could be leveraged. As an example, it is possible to apply the UDF `FPDistances` to the two fingerprints with identifiers 525373 and 525373 (belonging to a version of the dataset UJIIndoorLoc enriched with adjacency information) to calculate, in addition to the 2D and 3D Euclidean distances, the minimum shortest path between them, that might represent the traveling effort for a user from a wrong to a right position estimate.

```
SELECT * FROM FPDistances(525373, 525405);
[178 msec, Fig. 8c]
```

5) TOPOLOGY-AWARE POSITIONING ALGORITHMS

Topology information also fosters the research on state-of-the-art positioning algorithms, based, for instance, on Graph Neural Networks or on other machine learning techniques, like, e.g., Hidden Markov Models, that could effectively leverage the graph structure of indoor premises in order to provide better position estimates.

6) TRAJECTORY AND PERSONALISED LOCALISATION

The system allows one to store subsequent online fingerprints provided by the same device or user over time. In such a way, it actually generates a trajectory of predictions, which can be exploited by a positioning algorithm to reduce the prediction error, discard outlier fingerprints, or reason about users' typical patterns. For instance, employing the UDF `TrajectoriesInPlace`, one can retrieve in an array-like format all the trajectories, that is, sequences of fingerprints, passing through a given place, e.g., tile, site, or floor. Below, we consider the tile with id 520815 (IPIN 2021 Competition Track 3 dataset).

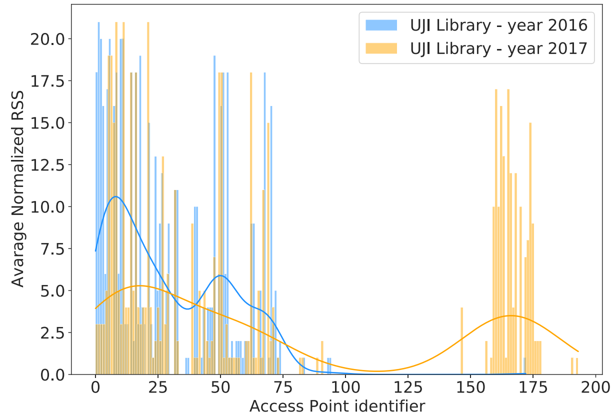
```
SELECT * FROM TrajectoriesInPlace(520815);
[130 msec, Fig. 8d]
```

7) NAVIGATION

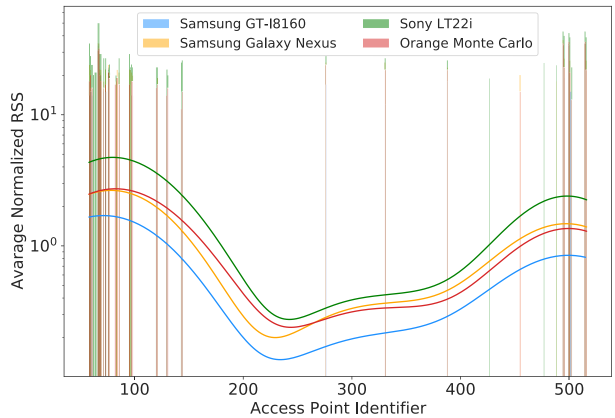
The system supports navigation tasks as well. As an example, a user may find the shortest path (in terms of the number of traversed tiles or associated traversability costs) to a specific location within the indoor premises, starting from his/her predicted location. This comes down to finding the shortest path between two nodes in the graph that represents the indoor scenario.

8) KNOWLEDGE DISCOVERY

The richness of information stored in the system, in terms both of the data and the relationships among them, makes it possible to develop a large number of unsupervised analysis tasks. For instance, usage patterns of the system could be investigated in order to discover regularities with respect to specific classes of users or time windows. Similar analyses may bring to the discovery of issues within a radio map, e.g., due to outlier fingerprints, that can then be corrected or removed. As a final example, consider the case of WiFi fingerprints collected within the same premises at different time points by different devices. By analyzing their signal patterns, it may be possible to discover regularities and differences in the access points or in the propagation of their signals. At this stage, a step further could be that of developing a normalization strategy to cope with such variations, so to consider a "time-and-device-corrected" version of fingerprints which may bring to a better online phase usage of the radio map.



(a) Average RSS observed for the same premises one year apart.



(b) Average RSS patterns of different devices for the same tile.

	euclidean2d numeric	euclidean3d numeric	msp numeric
1	63.9531078212779	[null]	18

(c) `FPDistances(525373, 525405)`. The 3D Euclidean distance is *null* as z-coordinates are not provided for the two involved fingerprints.

	trajectory integer[]
1	{520938,520939,520940,520941,520942,520943,520944,520945,520946,520947,520948,520949,520950,520951}
2	{520952,520953,520954,520955,520956,520957,520958,520959,520960,520961,520962,520963,520964,520965}
...	...
9	{520924,520925,520926,520927,520928,520929,520930,520931,520932,520933,520934,520935,520936,520937}
10	{520910,520911,520912,520913,520914,520915,520916,520917,520918,520919,520920,520921,520922,520923}

(d) `TrajectoriesInPlace(520815)`.

FIGURE 8. Outcomes of some interactions with the system.

VI. RELATED WORK

A great amount of research has been done on indoor positioning systems, their applications, and related tools. Here, we focus on two areas that are closely related to the present work. A comparative summary of notable features, with a ranking of their importance and support capabilities by the considered systems/frameworks, is given in Table 1.

A. MODELING INDOOR PREMISES

Several proposals have been made on how to model indoor premises [23], [35], [36], although a general consensus is still missing. Many of the most advanced formats and standards, e.g., IFC – Industrial Foundation Classes [37] or CityGML [38], pay a special attention to the description of scenarios with all their details [22]. IndoorGML [21], [39], instead, mainly focuses on the description of the structure of a premise, with a focus on the arrangement of spaces and their relationships. All these approaches (especially the latter) are quite flexible and provide some support to positioning tasks through additional application layers.

Our contribution differs from them in several respects. First, its main goal is to store and support (fingerprint-based) positioning, and only in accordance with that to provide topological information about the premises. From this point of view, the proposed solution is close in spirit to IndoorGML, which also has the latter capability. However, our modeling goes in the direction of what should be implemented within the prospective IndoorGML 2.0, allegedly designed to also work with topological information only [40]. Another key difference with respect to existing solutions is the possibility to easily interact via SQL (in fact, the proposed one is the first and only relational database-centric system). For the sake of completeness, it is worth remarking that IndoorGML is better than our approach in handling indoor modeling/mapping and navigation tasks. However, as already remarked, it is important to keep in mind that our motivations and goal are quite different.

B. COMPARING POSITIONING SYSTEMS

Mainly motivated by the high diversity in metrics, datasets, and result reporting approaches in the literature, several authors have studied how to enable a fair comparison of indoor positioning systems, also implementing several tools [41], [42]. The EvAAL framework [43] has been largely adopted by indoor positioning competitions [44], [45], [46], [47]. Its main goal is to enable fair, realistic, and systematic comparison of positioning solutions, especially in the case when they rely on different methodologies and sensors data. It achieves that through its core principles: natural movement of an actor, realistic environment, realistic measurement resolution, and, third quartile of point Euclidean error. Localisation Systems Repository [48] aims at supporting continuous, reliable, and accurate positioning on smartphone devices, providing a large benchmark suite and a repository for localisation systems source code. Web platforms for the evaluation and comparison of indoor localisation algorithms have been proposed in [49] and [50], respectively focusing on radio frequency-based and fingerprinting data. Our work largely differs from all previous ones, that are basically oriented towards storing tabular datasets and comparing algorithmic performances as: (*i*) it supports the industrial-level deployment of indoor positioning systems, (*ii*) it stores data in a normalised way, highlighting their relationships and making them easier to exploit, (*iii*) it embeds information on building topology, and (*iv*) it supports advanced concepts such as trajectories and multi-sensor data.

VII. CONCLUSION AND FUTURE WORK

In this work, we presented a comprehensive, yet general, and extensible framework to support indoor positioning tasks, whose core is a relational database made available online. Its main characteristic is that it allows one to represent topological information of indoor premises, which can be seamlessly combined with fingerprint positioning data. The flexibility of the system makes it capable of accommodating several indoor scenarios and supporting a large number of tasks, both concerning its industrial deployment, as well as considering its usage within the research community. As for future developments, besides including more datasets to foster the development of an open repository for the community, we plan to investigate the interoperability between the proposed system and mapping standards such as IndoorGML, and to develop a graph database version of the framework to compare the performance and capabilities of the two solutions.

TABLE 1. Summary of the notable features and their importance and support capabilities by the considered systems/frameworks.

System / Framework	Compare IPSs	Model Indoor Positioning	Multi-sensor Support	Building Topology	Indoor Mapping	Detailed Scenarios	Enforce Data Consistency
IndoorGML [39]		*	*	***	***	*	
IFC [37]					***	***	
EvAAL [43]	***		***				
LSR [48]	***		*				
Web tool [49]	**						
Web tool [50]	**						
our	*	***	***	***			***

*** = prominent feature of the system, full/good support capabilities; ** = feature of the system but there are better competitors, some support capabilities; * = not an aim of the system, which anyway provides limited support capabilities; IPS = Indoor Positioning System.

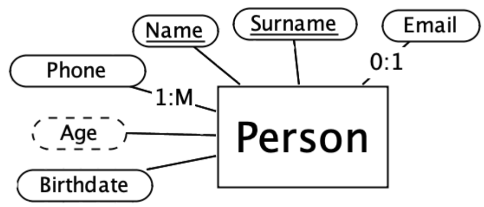


FIGURE 9. Strong entity set notation.

APPENDIX A ENTITY-RELATIONSHIP DIAGRAM NOTATION

Here we describe, by means of a series of examples, the notation employed in the Entity-Relationship diagram of Figure 3. Figure 9 depicts a strong entity set named *Person*, that has a primary key composed of the attributes *Name* and *Surname*. Each entity of *Person* may have at most one *Email* address, and one or more *Phone* numbers. In addition, it always has a *Birthdate*, based on which the value for the derived attribute *Age* is established.

Figure 10 reports the case of a weak entity set, named *Song*, that has the attribute *Title* as its partial identifier. Its identifying relationship is *Belongs to*, thus, the title of a song is unique within a given album. The entity set *Album* has *Name* as its primary key. Each album contains one or more songs, and a song belongs to one and only one album (the constraint 1 : 1 is assumed by default by our notation, and thus has been omitted on the *Song* side of the relationship).

Figure 11 shows the notation for a total and disjoint specialization. Each entity of entity set *Professor* is uniquely identified by its *SSN*, and it corresponds to either a *Full* or an *Associate* professor.

Finally, Figure 12 represents the case of a partial specialization. Here, an entity of entity set *Employee*, uniquely identified by its *SSN*, can also be an entity of entity set

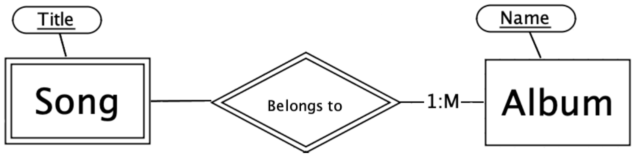


FIGURE 10. Weak entity set notation.

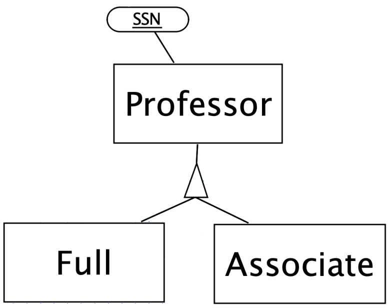


FIGURE 11. Total specialization notation.

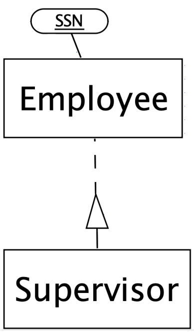


FIGURE 12. Partial specialization notation.

Supervisor. This is quite natural, since supervisors are themselves employees, but not all employees are supervisors.

APPENDIX B EXEMPLARY SQL QUERIES

In the following, for illustrative purposes, we report some simple SQL queries that can be used to extract relevant information from the relational database for indoor positioning. For each query, we also report the number of returned rows and the average running time in milliseconds when executed against the working demo of the system [27]. The server running the demo is hosted on a virtual machine equipped with 4 dedicated cores (Intel(R) Xeon(R) CPU X5550 running at 2.67 GHz) and 20 GB main memory.

QUERY 1

It extracts topological information regarding the dataset UJI-IndoorLoc [10]. Specifically, for each building, it retrieves its structuring into floors and sites [905 rows, 180 msec].

```
SELECT
 building_place.name AS building_name,
 floor_place.name AS floor_name,
 site_place.name AS site_name
FROM place AS building_place
JOIN place_data_source ON place_data_source.place_id = building_place.id
JOIN data_source ON data_source.id = place_data_source.data_source_id
JOIN building ON building_place.id = building.place_id
JOIN contains AS contains_floor ON
 contains_floor.container_place_id = building_place.id
JOIN contains AS contains_site ON
 contains_site.container_place_id = contains_floor.contained_place_id
JOIN place AS floor_place ON
 floor_place.id = contains_floor.contained_place_id
JOIN place AS site_place ON
 site_place.id = contains_site.contained_place_id
WHERE data_source.name = 'UJI1'
ORDER BY building_place.name, floor_place.name, site_place.name;
```

QUERY 2

It extracts the *id* of all the tiles that are (directly or indirectly) reachable from the tile with *id = 524465*. Note that, in order to perform such an "unlimited" visit of the graph, we need to rely on a recursive strategy [110 rows, 140 msec].

```
WITH RECURSIVE reachable AS (
SELECT
 adjacent_to_tile.tile_2_place_id
FROM adjacent_to_tile
WHERE walkable AND tile_1_place_id = 524465
UNION
SELECT
 succ.tile_2_place_id
FROM reachable AS prev
JOIN adjacent_to_tile AS succ ON
 succ.tile_1_place_id = prev.tile_2_place_id AND succ.walkable
)
SELECT tile_2_place_id AS reachable_tile_id
FROM reachable;
```

QUERY 3

It extracts the WiFi portion of the fingerprint with *id = 520857*. Observe that, since in the database only information pertaining to the detected access points is stored, in order to recover the full WiFi fingerprint (with respect to all access points in a *data_source*) outer join operations are necessary [544 rows, 245 msec].

```
SELECT
 ap.id,
COALESCE(ap_detection.rss, -110) AS rss
FROM fingerprint
JOIN observation_wifi ON
 observation_wifi.fingerprint_id = fingerprint.id AND fingerprint.id = 520857
JOIN ap_detection ON
 ap_detection.observation_wifi_fingerprint_id = observation_wifi.fingerprint_id
RIGHT OUTER JOIN ap ON ap_detection.ap_id = ap.id
RIGHT OUTER JOIN ap_data_source ON ap.id = ap_data_source.ap_id
RIGHT OUTER JOIN data_source ON
 ap_data_source.data_source_id = data_source.id AND data_source.name = 'UJI1'
ORDER BY ap.id;
```

QUERY 4

Given the (WiFi) fingerprint with *id = 533530*, the code extracts all sites containing fingerprints that have at least one access point in common with it, together with the number of such fingerprints (a *zone* or *logical* tessellation is assumed) [545 rows, 10 sec].

```
SELECT
contains.container_place_id AS site_id,
COUNT(DISTINCT finger_2.id) AS num_fingerprints
FROM fingerprint AS finger_1
JOIN ap_detection AS ap_detection_1 ON
ap_detection_1.observation_wifi_fingerprint_id = finger_1.id
JOIN ap_detection AS ap_detection_2 ON
ap_detection_1.ap_id = ap_detection_2.ap_id
JOIN fingerprint AS finger_2 ON
finger_2.id = ap_detection_2.observation_wifi_fingerprint_id
JOIN contains ON
contains.contained_place_id = finger_2.acquired_at_tile_place_id
JOIN site ON contains.container_place_id = site.place_id
WHERE finger_1.id = 533530 AND finger_2.is_radio_map AND finger_1.id != finger_2.id
GROUP BY contains.container_place_id
ORDER BY num_fingerprints DESC;
```

APPENDIX C USAGE OF THE ONLINE DEMO OF THE SYSTEM

The demo of the system can be accessed at the address http://158.110.145.70:5050/. Upon connection, users will find a *pgAdmin* web server interface, asking for the login data. A read-only user, that has the privileges to perform `SELECT` operations over the *public* and *evaluation_support* schemas of the database *Open_Fingerprinting* has been provided, with the following credentials:

$$\begin{aligned} \textit{username} &= \text{tester@indoor.uniud.it} \\ \textit{password} &= \text{tSUD22\$Indo0r} \ . \end{aligned}$$

The database comes already populated with information originating from several datasets.[1] Moreover, some user-defined functions aimed at easing the interaction with the system have been implemented[2] as well as various examples of queries on the database.[3]

ACKNOWLEDGMENT

(Nicola Saccomanno and Andrea Brunello are co-first authors.)

REFERENCES

[1] *Report to Congress on Indoor Air Quality: Assessment and Control of Indoor Air Pollution*, US Environmental Protection Agency, Washington, DC, USA, 1989.
[2] A. Khalajmehrabadi, N. Gatsis, and D. Akopian, "Modern WLAN fingerprinting indoor positioning methods and deployment challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1974–2002, 3rd Quart., 2017, doi: 10.1109/COMST.2017.2671454.
[3] A. Pérez-Navarro, J. Torres-Sospedra, R. Montoliu, J. Conesa, R. Berkvens, G. Caso, and C. Costa, "Challenges of fingerprinting in indoor positioning and navigation," in *Geographical and Fingerprinting Data to Create Systems for Indoor Positioning and Indoor/Outdoor Navigation*. San Diego, CA, USA: Academic, 2019, pp. 1–20.
[4] G. M. Mendoza-Silva, J. Torres-Sospedra, and J. Huerta, "A meta-review of indoor positioning systems," *Sensors*, vol. 19, no. 20, p. 4507, 2019, doi: 10.3390/s19204507.

[1] https://github.com/dslab-uniud/Database-indoor/tree/main/Datasets
[2] https://github.com/dslab-uniud/Database-indoor/tree/main/Database#implemented-user-defined-functions
[3] https://github.com/dslab-uniud/Database-indoor/blob/main/Database/exemplary_SQL.sql

[5] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. 19th Annu. Joint Conf. IEEE Comput. Commun. Societies*, Tel Aviv, Israel, Mar. 2000, pp. 775–784.
[6] S. Xia, Y. Liu, G. Yuan, M. Zhu, and Z. Wang, "Indoor fingerprint positioning based on Wi-Fi: An overview," *ISPRS Int. J. Geo-Inf.*, vol. 6, no. 5, p. 135, Apr. 2017.
[7] S. He and S.-H. G. Chan, "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 466–490, Jan. 2015.
[8] X. Feng, K. A. Nguyen, and Z. Luo, "A survey of deep learning approaches for WiFi-based indoor positioning," *J. Inf. Telecommun.*, vol. 6, no. 2, pp. 1–54, 2021.
[9] P. Roy and C. Chowdhury, "A survey of machine learning techniques for indoor localization and navigation systems," *J. Intell. Robot. Syst.*, vol. 101, no. 3, p. 63, Mar. 2021, doi: 10.1007/s10846-021-01327-z.
[10] J. Torres-Sospedra, R. Montoliu, A. Martinez-Uso, J. P. Avariento, T. J. Arnau, M. Benedito-Bordonau, and J. Huerta, "UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Busan, South Korea, Oct. 2014, pp. 261–270.
[11] E. S. Lohan, J. Torres-Sospedra, H. Leppäkoski, P. Richter, Z. Peng, and J. Huerta, "Wi-Fi crowdsourced fingerprinting dataset for indoor positioning," *Data*, vol. 2, no. 4, p. 32, Oct. 2017.
[12] G. Mendoza-Silva, P. Richter, J. Torres-Sospedra, E. Lohan, and J. Huerta, "Long-term WiFi fingerprinting dataset for research on robust indoor positioning," *Data*, vol. 3, no. 1, p. 3, Jan. 2018.
[13] E. S. Lohan, "Additional TAU datasets for Wi-Fi fingerprinting-based positioning," Zenodo, May 2020. [Online]. Available: https://zenodo.org/record/3819917#.Y1-4CXbP2Uk
[14] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg, "CRAWDAD dataset mannheim/compass (V. 2008-04-11)," Crawdad, 2008. [Online]. Available: https://crawdad.org/mannheim/compass/20080411/
[15] A. Moreira, I. Silva, and J. Torres-Sospedra, "The DSI dataset for Wi-Fi fingerprinting using mobile devices," Zenodo, Apr. 2020. [Online]. Available: https://zenodo.org/record/3778646#.Y1-4UHbP2Uk
[16] J. Torres-Sospedra and A. Moreira, "Analysis of sources of large positioning errors in deterministic fingerprinting," *Sensors*, vol. 17, no. 12, p. 2736, Nov. 2017.
[17] S. Shrestha, J. Talvitie, and E. S. Lohan, "Deconvolution-based indoor localization with WLAN signals and unknown access point locations," in *Proc. Int. Conf. Localization GNSS*, Turin, Italy, Jun. 2013, pp. 1–6.
[18] P. Richter, E. S. Lohan, and J. Talvitie, "WLAN (WiFi) RSS database for fingerprinting positioning," Zenodo, Jan. 2018. [Online]. Available: https://zenodo.org/record/1161525#.Y1-4dHbP2Uk
[19] X. Song, X. Fan, C. Xiang, Q. Ye, L. Liu, Z. Wang, X. He, N. Yang, and G. Fang, "A novel convolutional neural network based indoor localization framework with WiFi fingerprinting," *IEEE Access*, vol. 7, pp. 110698–110709, 2019.
[20] M. Goetz and A. Zipf, "Extending OpenStreetMap to indoor environments: Bringing volunteered geographic information to the next level," in *Urban and Regional Data Management: UDMS Annual*, 2011, pp. 47–58. [Online]. Available: https://www.taylorfrancis.com/chapters/edit/10.1201/b11647-7/extending-openstreetmap-indoor-environments-bringing-volunteered-geographic-information-next-level-goetz-zipf and https://www.taylorfrancis.com/books/edit/10.1201/b11647/urban-regional-data-management-massimo-rumor-sisi-zlatanova-hugo-ledoux-elfriede-fendel?refId=5905a3c2-5b4f-496f-82d9-7098b512063a&context=ubx
[21] H.-K. Kang and K.-J. Li, "A standard indoor spatial data model—OGC IndoorGML and implementation approaches," *ISPRS Int. J. Geo-Inf.*, vol. 6, no. 4, p. 116, Apr. 2017.
[22] K.-J. Li, S. Zlatanova, J. Torres-Sospedra, A. Perez-Navarro, C. Laoudias, and A. Moreira, "Survey on indoor map standards and formats," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sep. 2019, pp. 1–8, doi: 10.1109/IPIN.2019.8911796.
[23] F. Mortari, E. Clementini, S. Zlatanova, and L. Liu, "An indoor navigation model and its network extraction," *Appl. Geomatics*, vol. 11, no. 4, pp. 413–427, Dec. 2019.
[24] J. Yan, S. Zlatanova, and A. A. Diakité, "A unified 3D space-based navigation model for seamless navigation in indoor and outdoor," *Int. J. Digit. Earth*, vol. 14, no. 8, pp. 985–1003, 2021, doi: 10.1080/17538947.2021.1913522.

[25] A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. New York, NY, USA: McGraw-Hill, 2019.
[26] A. Brunello, A. Montanari, and N. Saccomanno. (May 2022). *GitHub Page of the Relational Database for Indoor Positioning Project*. Data Science and Automatic Verification Laboratory. University of Udine. [Online]. Available: https://github.com/dslab-uniud/Database-indoor
[27] A. Brunello, A. Montanari, and N. Saccomanno. (May 2022). *Live Implementation of the Relational Database for Indoor Positioning*. Data Science and Automatic Verification Laboratory. University of Udine. [Online]. Available: http://158.110.145.70:5050/
[28] F. Furfari, A. Crivello, P. Baronti, P. Barsocchi, M. Girolami, F. Palumbo, D. Quezada-Gaibor, G. M. M. Silva, and J. Torres-Sospedra, "Discovering location based services: A unified approach for heterogeneous indoor localization systems," *Internet Things*, vol. 13, Mar. 2021, Art. no. 100334, doi: 10.1016/j.iot.2020.100334.
[29] N. Saccomanno, A. Brunello, and A. Montanari, "What you sense is not where you are: On the relationships between fingerprints and spatial knowledge in indoor positioning," *IEEE Sensors J.*, vol. 22, no. 6, pp. 4951–4961, Mar. 2022.
[30] A. Viel, P. Gallo, A. Montanari, D. Gubiani, A. D. Torre, F. Pittino, and C. Marshall, "Dealing with network changes in cellular fingerprint positioning systems," in *Proc. Int. Conf. Localization GNSS (ICL-GNSS)*, Nottingham, U.K., Jun. 2017, pp. 1–6, doi: 10.1109/ICL-GNSS.2017.8376255.
[31] P. P.-S. Chen, "The entity-relationship model-toward a unified view of data," *ACM Trans. Database Syst.*, vol. 1, no. 1, pp. 9–36, 1976.
[32] The PostgreSQL Global Development Group. (May 2022). *PostgreSQL*. [Online]. Available: https://www.postgresql.org/
[33] pgAdmin Development Team. (May 2022). *pgAdmin*. [Online]. Available: https://www.pgadmin.org/
[34] Oracle Corporation. (May 2022). *The Java Database Connectivity (JDBC)*. [Online]. Available: https://www.oracle.com/java/technologies/javase/javase-tech-database.html
[35] J. Lee, "A spatial access-oriented implementation of a 3-D GIS topological data model for urban entities," *GeoInformatica*, vol. 8, no. 3, pp. 237–264, Sep. 2004, doi: 10.1023/B:GEIN.0000034820.93914.d0.
[36] I. Afyouni, C. Ray, and C. Claramunt, "Spatial models for context-aware indoor navigation systems: A survey," *J. Spatial Inf. Sci.*, no. 4, pp. 85–123, Jun. 2012.
[37] buildingSMART. (May 2022). *Industry Foundation Classes Standard*. [Online]. Available: https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/
[38] Open Geospatial Consortium. (May 2022). *CityGML Encoding Standard*. [Online]. Available: https://www.ogc.org/standards/citygml
[39] Open Geospatial Consortium. (May 2022). *IndoorGML—OGC Standard for Indoor Spatial Information*. [Online]. Available: http://www.indoorgml.net/
[40] A. A. Diakité, S. Zlatanova, A. F. M. Alattas, and K. J. Li, "Towards IndoorGML 2.0: Updates and case study illustrations," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vols. 2020, pp. 337–344, Aug. 2020.
[41] F. Potortì, S. Park, A. R. J. Ruiz, P. Barsocchi, M. Girolami, A. Crivello, S. Lee, J. H. Lim, J. Torres-Sospedra, F. Seco, R. Montoliu, G. M. Mendoza-Silva, M. D. Carmen Pérez, C. Losada-Gutiérrez, F. Espinosa, and J. M. Guarasa, "Comparing the performance of indoor localization systems through the EvAAL framework," *Sensors*, vol. 17, no. 10, p. 2327, 2017, doi: 10.3390/s17102327.
[42] J. Torres-Sospedra, I. Silva, L. Klus, D. Quezada-Gaibor, A. Crivello, P. Barsocchi, C. Pendao, E. S. Lohan, J. Nurmi, and A. Moreira, "Towards ubiquitous indoor positioning: Comparing systems across heterogeneous datasets," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Lloret de Mar, Spain, Nov. 2021, pp. 1–8, doi: 10.1109/IPIN51156.2021.9662560.
[43] P. Barsocchi, S. Chessa, F. Furfari, and F. Potortì, "Evaluating AAL solutions through competitive benchmarking: The localization competition," *IEEE Pervasive Comput. Mag.*, vol. 12, no. 4, pp. 72–79, Oct./Dec. 2013.
[44] F. Potortì et al., "Off-line evaluation of indoor positioning systems in different scenarios: The experiences from IPIN 2020 competition," *IEEE Sensors J.*, vol. 22, no. 6, pp. 5011–5054, Mar. 2022.
[45] F. Potorti et al., "The IPIN 2019 indoor localisation competition—Description and results," *IEEE Access*, vol. 8, pp. 206674–206718, 2020.
[46] V. Renaudin, "Evaluating indoor positioning systems in a shopping mall: The lessons learned from the IPIN 2018 competition," *IEEE Access*, vol. 7, pp. 148594–148628, 2019.
[47] J. Torres-Sospedra, J. A. R. Jiménez, A. Moreira, T. Lungenstrass, W.-C. Lu, S. Knauth, and G. M. Mendoza-Silva, "Off-line evaluation of mobile-centric indoor positioning systems: The experiences from the 2017 IPIN competition," *Sensors*, vol. 18, no. 2, p. 487, 2018.
[48] I. E. Radoi, D. Cirimpei, and V. Radu, "Localization systems repository: A platform for open-source localization systems and datasets," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, F. Potorti, V. Renaudin, K. O'Keefe, and F. Palumbo, Eds., Sep. 2019, pp. 1–8, doi: 10.1109/IPIN.2019.8911748.
[49] F. Lemic, V. Handziski, N. Wirstrom, T. Van Haute, E. De Poorter, T. Voigt, and A. Wolisz, "Web-based platform for evaluation of RF-based indoor localization algorithms," in *Proc. IEEE Int. Conf. Commun. Workshop (ICCW)*, London, U.K., Jun. 2015, pp. 834–840, doi: 10.1109/ICCW.2015.7247280.
[50] R. Montoliu, E. Sansano, J. Torres-Sospedra, and O. Belmonte, "IndoorLoc platform: A public repository for comparing and evaluating indoor positioning systems," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sapporo, Japan, Sep. 2017, pp. 1–8, doi: 10.1109/IPIN.2017.8115940.

ANDREA BRUNELLO received the Ph.D. degree in computer science from the University of Udine, Italy, in 2020. He then joined as a Postdoctoral Researcher with the Department of Mathematics, Computer Science and Physics, University of Udine, where he is currently an Assistant Professor. He has published various papers in international conferences and journals, and he is a supervisor of theses. His main research interests include data modeling and integration, data mining, and machine learning.

ANGELO MONTANARI received the Ph.D. degree in logic and computer science from the University of Amsterdam, The Netherlands, in 1996. He is currently a Full Professor in computer science with the University of Udine, Italy, where he chairs the Data Science and Automatic Verification Laboratory. He has published over 250 contributions in international journals, conferences, and handbook chapters. His main research interests include formal methods, artificial intelligence, and spatio-temporal databases.

NICOLA SACCOMANNO received the master's degree in computer science from the Department of Mathematics, Computer Science and Physics, University of Udine, Italy, jointly with the University of Klagenfurt, Austria, in 2019. He is currently pursuing the Ph.D. degree with the Department of Mathematics, Computer Science and Physics, University of Udine. He has published several works in international conferences, journals, and he co-supervised multiple students. His main research interests include indoor positioning and applied machine and deep learning, the latter especially to the healthcare domain.

Open Access funding provided by 'Università degli Studi di Udine' within the CRUI CARE Agreement