

# INTEGRATION OF IOT SENSORS TO 3D INDOOR MODELS WITH INDOORGML

A. Sarmiento<sup>1</sup>, A. A. Diakit <sup>2</sup>

<sup>1</sup> Faculty of Engineering, School of Civil and Environmental Engineering, University of New South Wales, Australia – a.sarmiento@student.unsw.edu.au

<sup>2</sup> Faculty of Built Environment, University of New South Wales, Australia – a.diakite@unsw.edu.au

## Commission VI, WG VI/4

**KEY WORDS:** Internet of Things, Sensors, Indoor Spatial Models, IndoorGML, GIS, Standards

### ABSTRACT:

Sensors are the vehicle through which Internet of Things (IoT) applications collect timely data of which are communicated to objects, or “Things”, to make them aware of their environment. With multiple sensors within an IoT system sending continuous streams of data, the potential scale of data is large, so efficient data management and useful representation is a key concern. As the information required from sensors benefit from a spatial context, 3D indoor models, such as IndoorGML, have been identified to support this condition. As it stands, a standardised structure to the amalgamation of sensors with IndoorGML has not been defined. The goal of this paper is to explore this opportunity by firstly, reviewing previous approaches to the integration of the two systems. Research into the interpretations of sensor information through existing standards is conducted before narrowing these attributes down into a minimal profile according to identified functional requirements of sensor applications. Finally, this knowledge is organised into a conceptual data model and presented as a thematic module in IndoorGML.

## 1. INTRODUCTION

The Internet of Things (IoT) is set to disrupt industries as we know it with the number of internet-connected devices already surpassing the total human population and is continuing to grow exponentially (Dave et al., 2018). As the name suggests, it involves a network of connected devices, sensors and actuators (“Things”) that are able to make dialogue with one another over web communication protocols (“Internet”). Through this system, it is aimed that data can be gathered and analysed, and responses to events in the physical world be created autonomously with minimal human intervention (Gubbi et al., 2013).

A key proponent of this technology are the sensors. A sensor is a small device capable of monitoring and collecting data on real-world conditions including audio, visual, chemical, environmental, positioning, and proximity information through sensing physical phenomena such as temperature, light, humidity, and vibrations. Their capability to provide real-time, context-rich insights about the environment have opened up opportunities in the smart buildings space, with sensing technologies being used in an ever-growing list of applications from proactive facility management systems which are able to self-regulate building services according to the feedback and needs of its occupants, as well as can play a role in reactive emergency response (Jia et al., 2019). With the notion of Internet of Things quickly becoming a household name, and even more so in industry, the wealth and timeliness of information assembled by IoT sensor sources has a need to be communicated, stored, and managed efficiently.

Spatial models are a significant opportunity to accomplish these goals as (1) they are visual representations of spaces, (2) they are proven urban information analysis and management tools, and (3) geometric and semantic descriptions of space can be discerned. When it comes to representing the indoor environment, two of the most renowned spatial models are the Industry Foundation Classes (IFC) from the Building

Information Modelling (BIM) domain and Indoor Geography Markup Language (IndoorGML) from the Geographic Information System (GIS) domain.

IndoorGML was devised by the Open Geospatial Consortium (OGC) to overcome the absence of space modelling and topological relationships of BIM standards, such as IFC, which focus on feature representation (Kang, Li, 2017). IndoorGML places magnitude on the semantics and relationships of indoor spaces, of which are significant to describe location and activities. It accomplishes this by adopting a cellular space model where spaces are embodied as cells mapped to nodes within the topology space, which have defined connectivity and adjacency to other nodes through edges and are further semantically categorised (e.g. navigable and non-navigable) (Lee et al., 2016). In this work, we have chosen to focus on IndoorGML because, unlike BIM standards, IndoorGML was developed with the idea to support Location Based Services (LBS), which naturally involves the inclusion of IoT sensors to support activities such as indoor navigation. Although previous versions of the standard provided limited support to such options, the version 2, which we are targeting here, is bringing several improvements for a stronger LBS support, including for example POI definitions (Claridades et al., 2019), modelling of indoor features (Diakit  and Zlatanova, 2016).

However, the current state of knowledge in this area is limited to conceptual proposals (Tang et al., 2019). Standards, including SensorML and ISO 19115, have been developed that are capable of organising sensor information, but there exists no open method that have integrated them with spatial information models. This is critical as sensor data is prone to being misinterpreted (Dawes et al., 2008) if sensor readings, sensor metadata and contextual awareness are not analysed in synchrony as these aspects are usually correlated. For example, if a temperature reading of a certain room is much higher than that of adjacent rooms, by gaining an understanding of the space’s conditions such as its occupancy levels, existence of heating devices etc., it can provide

a critical layer of reasoning. Additionally, a lack of standardisation in representation makes sensor interoperability a challenge due to the proliferation of varying sensor types, usages, manufacturers, protocols and software, garnering it difficult to deploy and manage large-scale sensor networks (Ding et al., 2013) without extensive fiddling into their interfaces. This may limit sensors to operate only within its application domain.

Taking everything into account, a gap in the research has been identified in the development of an open standard that integrates 3D spatial models with IoT sensors. It is the objective of this study to bridge this gap by proposing a conceptual framework uniting both IndoorGML and sensor information. As an indoor-navigation centric spatial model, IndoorGML currently does not have any well-defined modules that cater to the adoption of sensor systems, which leads to inefficiencies in setting up GIS-IoT systems and adversely affects information flow among heterogeneous devices and networks.

In this paper, a review on previous work towards the integration of sensors into indoor spatial models, as well as two existing standards that can describe sensors will be undertaken in Sections 2 and 3. Section 4 presents a study towards discovering the fundamental properties of sensors. A proposition of the integrated data model is presented in Section 5 before the paper is concluded with a summary of research project and any future directions in Section 6.

## 2. PREVIOUS WORK TOWARDS THE INTEGRATION OF SENSORS INTO 3D INDOOR MODELS

The need to integrate building models with sensor data in a generic way is unmet. However, evaluation of related work can be used to further push research further into this direction.

### 2.1 Integration of IFC with SensorML

After study into numerous existing sensor metadata standards, a prototype of a SensorML based integration with IFC was decided by Liu and Akinci (2009). Their methodology consisted of comparing six existing standards including IEEE 1451, ANSI N42.42, SensorML, TransducerML, oBix and IFC in relation to how they described the metadata properties of location, measured object(s), measurement, calibration, sensor readings, interface, and functional/spatial aggregation.

SensorML was able to represent most richly each of these criteria, though it was not able to infer the measured object (i.e. what space/objects are being measured). It was also realised that IFC was adept at embodying the environment and location of the sensor, whilst SensorML can accurately pinpoint sensor readings and metadata, and so form a compatible pairing that marries the strengths of BIM with the strengths of IoT sensors. A prototype using IFC Parser to edit an IFC model and SensorML Model Editor to input sensors according to their data model were able to validate their claims of the feasibility of a sensor-integrated model under the IFC standard.

Due to the potential of SensorML for describing sensors as recognised in their work, it was deemed worthy to review it, as given in Section 3.1.

### 2.2 Extending IFC for specific sensor support

Instead of combining two representations of BIM and sensors into the one model, Theiler et al. (2017) have opted for an approach that defines an extension, which they named IFC

Monitor, to the existing IFC standard. This was executed by defining a subset of new classes. Three main entities – structural health and monitoring and control system, sensor network and sensor nodes – were established into new IFC classes – *IfcMonitoringControlSystem*, *IfcSensorNetwork* and *IfcSensorNode*, and integrated within the existing hierarchal structure of IFC (see Figure 1). For example, the first two are both able to be satisfactorily represented by the *IfcSystem* class as they aggregate smaller components, sensor networks and nodes respectively, for a common function. Accordingly, they have an inheritance relationship with that class.

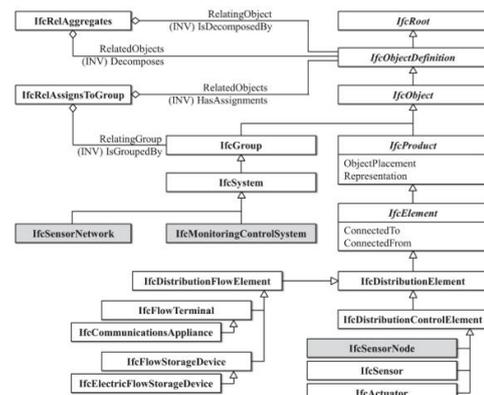


Figure 1. IFC Monitor model (from Theiler et al., 2017)

A prototype of the IFC-based system was tested to investigate its effectivity in sensing changes in the test structure to mimic monitoring of structural health. The sensors were able to detect the deformations and so their proposed schema was effective.

In a similar vein, Rio et al. (2013), took the existing sensor property sets of IFC, and adapted it for their structural sensors, for application within the structural health monitoring area, as well. Because the type of sensor this study used, a vibrating wire strain gauge sensor, was not already defined in IFC, a custom set of properties *Pset\_SensorTypeVibratingWireSensor*, that addressed all the sensor data's requirements and was compatible with *IfcSensorType*, was crafted. As part of their validation process, they created a BIM including a virtual vibrating wire strain gauge sensors and were able to exhibit that their prototype was successful in visualising sensors within a building information model and performing structural analysis. Whilst the approach was viable, it was deemed as inconvenient due to the differing requirements of BIM applications.

Whilst the concern for this investigation is to support sensors in a universal means, the methodologies and results of both works are still valuable, even if they have been applied to the field of civil engineering. This is because they demonstrate the ability of existing spatial models (i.e. IFC) to be revised to accommodate specific sensors and their use cases through (1) the creation of new classes that inherit from existing elements within the standard or (2) adapting existing profiles to accommodate for that sensor, revealing possible approaches the problem.

## 3. REVIEW OF EXISTING STANDARDS FOR SENSOR REPRESENTATION

### 3.1 Sensor Model Language (SensorML)

**3.1.1 Description of SensorML:** Sensor Model Language (SensorML) is an OGC standard to encode sensor attributes to enable their discovery, selection, processing, and actuation.

Processes are designed in SensorML to classify components such as physical and virtual hardware (e.g. detectors, actuators, and sensors) and actual processes (e.g. computations, mathematical functions). Given an input, such as a reading of a phenomena by a sensor, a method can be applied over it using passed parameter values to generate an output which is either numerical or descriptive in nature. Optionally, metadata can be defined to aid in documentation and provide more context.

A sensor for our purposes can be best represented as a *Component*, specifically a *PhysicalComponent* class – real devices in which spatiotemporal position is significant. A model of this class is given in Figure 2.

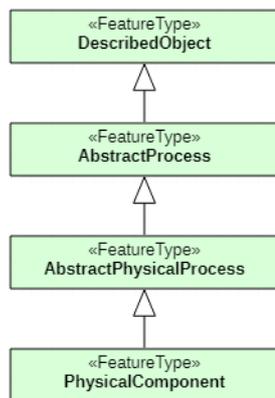


Figure 2. Data model for the *PhysicalComponent* class

At the base layer is a *PhysicalComponent* which has a single attribute – a method. This feature inherits properties related to the spatial and temporal contexts (e.g. *attachedTo*, *position*, *timePosition*) from *AbstractPhysicalProcess* of a sensor component. The *AbstractPhysicalProcess* is also a type of *AbstractProcess* comprising of a set of descriptions including *inputs* (e.g. wind), *outputs* (e.g. wind chill, wind speed), *parameters*, *featuresOfInterest*, *configuration* and *modes*, of which also inherits properties from *DescribedObject*, a class to define metadata.

Metadata, which is optional and should not be required for the proper execution of a process, is annotated through lists of descriptive attributes including keywords, identification (e.g. *shortName*, *processorID*, *serialNumber*), classification (e.g. *sensorType*, *intendedApplication*), constraints (e.g. *validTime*, *securityConstraints* and *legalConstraints*), characteristics (e.g. dimensions, weight, power demands, lifetime), capabilities (e.g. measurable range, sampling frequency, operational and survivable limits), references (e.g. *contacts* and *documentation*), and *history*.

**3.1.2 Evaluation of SensorML:** SensorML embraces a strong focus on geolocation by targeting multiple approaches to expressing location – it can be described through text description, point, vector (e.g. latitude, longitude, and altitude), trajectory and process (e.g. orbital model). Therefore, even dynamic components are supported, and a true heading and pitch degrees together can provide a sensor’s orientation. This trait makes it fit for incorporation into 3D models.

Other advantages of the standard is that it is rich in information that can classify a sensor according to its associated keywords, identification, references and history fields and qualify its data due to the facility to specify constraints, characteristics and capabilities. Additionally, its physical inputs can be processed via the *method* property of *PhysicalComponent* to compute outputs without the need for extra software. For example, windchill can be derived from temperature and wind speed.

Though, a major shortcoming with the standard is that the definition of the sensing area and coverage is not encapsulated in the standard, wherein the sensor data’s location is usually attributed to that of the sensor itself. This means that there is no way to determine the spatial extent of data or to geolocate an individual reading, where a query such as “What is the current humidity at (x, y, z)?” is difficult to answer.

Overall, SensorML is a sensing standard that is highly applicable to the deployment and management of sensors with spatial models due to fact that it effectively combines both sensor-related requirements (i.e. sensor reading and sensor metadata) into the one model, and adapts to multiple applications.

## 3.2 ISO 19115

**3.2.1 Description of ISO 19115:** The International Organization for Standardization created an information standard, ISO 19115, which was targeted at structuring and describing geographic metadata including information about its identification, extent, data quality, spatial reference, distribution, access and rights. In 2013, a revision, ISO 19115-2, was produced to add further capabilities in defining the acquisition and processing of data captured from other sources, such as remote sensing, as opposed to just imagery. This extension came about as information about the instrument and its measuring and computational methods was deemed valuable to support the process of transforming raw data into geographic information. The standard is composed as an XML schema, akin to the other sensor standard reviewed in this study, SensorML, and the indoor spatial model of reference, IndoorGML. Although it is not strictly a sensor standard, it has elements that capture properties about sensors’ metadata, observations, and locality, and hence was still regarded as relevant and worthwhile to examine.

ISO 19115-2 is divided into many packages that each express a specific area of description. The root package is *Metadata (MI\_Metadata)*, which provides a high-level overview of all the information to represent geographic data. Metadata may be quantitative or qualitative as denoted by *contentInfo*. A range of descriptive parameters are defined within *identificationInfo* that further assists in classifying data according to its purpose, usage, constraints and spatiotemporal extent, as well as providing more avenues for sensors to be discovered through definition of keywords and topics. The quality of the data may be assessed on varying scales, according to metrics such as positional and quantitative accuracy. In addition, a history of the data’s activities (e.g. source) can be recorded. This is all achieved within *dataQualityInfo* of the schema. The *acquisitionInformation* (see Figure 3) attribute introduced in the second iteration of ISO 19115 is where the instrument, such a sensor, and platform information can be obtained. An acquisition can be defined in terms of its operational status and its objective which can be made up of a series of events.

**3.2.2 Evaluation of ISO 19115:** It is clear that ISO 19115 is observation-centric, as opposed to being a sensor standard, since nearly all groups of information revolve around the geographic data itself. Taking the example of *maintenanceInfo*, this parameter does not describe the instrument’s maintenance habits, it revolves around the upkeep of the resource or metadata records.

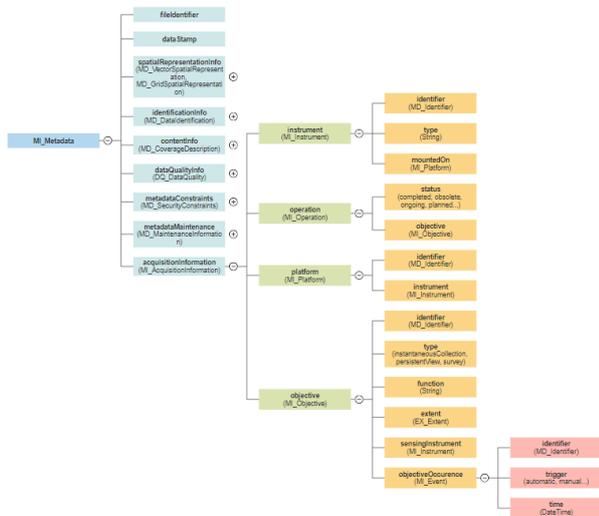


Figure 3. Organisation of *acquisitionInformation* attribute

A big concern is recognised as the lack of representation in positioning a sensor – a prominent factor required for integration with spatial models. This inevitably raises issues into how and where to locate the device.

What ISO 19115 does well is through its comprehensive methods to classify data, such as by keywords, categories and usage, making the process of selecting the most appropriate sensor(s) to extract data from more precise. Its verbose log of a dataset’s lineage is also valuable when extracting changes or patterns in data over time.

**3.3 Comparison of SensorML and ISO 19115**

It is notable to point out that, whilst both standards function in somewhat opposite regards, SensorML is primarily used to describe sensors and secondarily, its data, whereas the focus of ISO 19115 is on the data representation but has references to its source – at their core, they describe sensors comparatively to each other. The key disparity between the two standards, which renders SensorML more suitable to use “out of the box”, is in their interpretation of sensor location, or lack thereof for ISO 19115. This is a major disadvantage against ISO 19115 as this is a fundamental property required to localise a sensor into a spatial model.

Both have their own strengths, from the wealth of metadata information that ISO 19115 supports and SensorML’s suitability for sensor definition. However, there is still a gap in the information. Moving forward, if concepts of either or both standards is reused, it will also need to be supplemented with representations of the sensing coverage.

**4. RELEVANT PROPERTIES OF IOT SENSORS**

The significant task of identifying the most relevant information items in view of standardisation, whilst maintaining a profile that is comprehensive enough that functional requirements of all application domains are met.

**4.1 Survey of Sensor Applications – Functional Requirements and Key Information**

A review of use cases involving sensors deployed in multiple industries from disaster relief to agriculture was conducted to extract the information that were considered critical to satisfy the application’s functional requirements. This exercise provided an understanding into the universal characteristics of sensor systems.

Sha et al. (2006) proposed FireNet, a sensor network application used in fire response which can track firefighters working on the field, providing commanders at remote fire departments with information to detect potential hazards, unsafe conditions or events for informed real-time decision making and planning. Although it is not constrained to indoor environments, it is still relevant in answering this research question. This is because the principal function of this application, in detecting phenomena that is location-dependent, is equivalent to the motivations behind a sensor’s inclusion into spatial models.

One of the key information items required is the location of the firefighter since commanders need to know where firefighters are in the field to direct them appropriately. This stream of data has to be received in a timely fashion and be to a high degree of accuracy since positions of the firefighters change rapidly and responses to events have to be made promptly and precisely due to the high risk nature of the work. The information about the firefighter such as their age and role, is useful for identification and tracking but does not correlate with an attribute of a sensor, rather it can be regarded as the measured object. Information about events that occur during the firefighting procedure (e.g. death of a firefighter) are observed as well. Because of the extreme environments that these sensors may operate in, the requirement for robustness is vital so properties such as operating requirements are captured to detect situations where nodes may fail or malfunction. If so, the status of each device should be closely observed as any faults with the sensor can result in distorted data.

A similar mapping exercise between the characteristics and equivalent sensor properties was performed on further use cases and are given in Appendix A.

In Table 1, a matrix with the applications on one axis and the list of sensor attributes on the other is made for comparison. Attributes are grouped under one of three categories – Observation, Metadata or Spatial Context – following the implied categories represented by the *AbstractProcess*, *DescribedObject* and *AbstractPhysicalProcess* classes of SensorML, respectively.

Matrix of sensor information according to application	Fire rescue	Soil moisture distribution mapping	Habitat monitoring	Indoor air quality monitoring	Indoor emergency monitoring and navigation
Observation: Value and Unit	Y	Y	Y	Y	Y
Observation: Timestamp	Y	Y	Y	Y	Y
Observation: Event	Y				Y
Metadata: Observation Type	Y		Y	Y	
Metadata: Measured Object(s)	Y	Y			
Metadata: Physical Qualities	Y				
Metadata: Status	Y		Y		
Metadata: Operating Requirements	Y		Y	Y	
Metadata: Accuracy Parameters				Y	
Metadata: Accessibility					
Spatial Context: Location	Y	Y	Y	Y	Y
Spatial Context: Orientation and Motion					
Spatial Context: Coverage					

Table 1. Matrix of sensor key information according to application

From the sample of use cases examined, trends between their requirements and, consequently, their data models are made apparent:

- **Sensors are data collectors:** A universal factor between all systems was the need to detect a type of phenomenon, whether it be the occupancy of a location or the humidity of an indoor environment. Thus, the principal role of the sensor nodes was to collect these data values which is associated with the property of “Observation: Value and Unit”.
- **Sensors and their data should have a temporal context:** The concept of time is recurrent in every use case, albeit in its own utilisation. Applications either require access to information in a time-constrained manner such as is typical in emergency response. The sensor network designed for fire rescue relies on being able to obtain real-time data on the firefighters, the field and emergent events (Sha et al., 2006) as they are integral to fulfill objectives of live rescue monitoring. On the other hand, real-time data collection is not the requirement for soil moisture distribution mapping or habitat monitoring. Rather, they utilise historical data for the intent of trend analysis. For the latter use case, by recording the times in which data is captured, predictive models built upon patterns in burrow usage according to time of day or how usage fluctuates throughout the breeding season (Mainwaring et al., 2002) can be generated. A similar case can be made in agriculture, where field conditions, such as soil water content, can be monitored across a specified temporal interval to detect environmental patterns which can assist in the development of automated scheduling systems for activities such as irrigation (Ayday, Safak, 2009). Thus, observations would benefit from being associated with a time property (“Observation: Timestamp”).
- **Sensors and their data should have a spatial context:** The importance of spatial context is another persistent trend amongst the applications for two main purposes. Firstly, as with time, data can change along space such as where rooms within a building or areas across a soil field can possess varying microclimate conditions. A sensor in the use case of habitat monitoring needed to be specified a location to offer knowledge about which burrow was being observed (Mainwaring et al., 2002). Giving a sensor a location adds values in terms of context about its surroundings and can clarify what the feature(s) of interest are. In addition, users often want to know information about a specific region or point. It is not possible to do so if sensors, and by extent, their data, are not geo-referenced in any way. All applications referenced some form of location, which was either a standard Cartesian coordinate or a symbolic representation, which is parallel to the property “Spatial Context: Location”.
- **Heterogenous sensors must be distinguishable:** In systems in which different types of sensors are employed, there needs to be a way to classify observations captured by the network. An observation value by itself is just a number with no other identifying characteristics and in a database where sensor data from various sources may be stored, it is difficult to assume a selected observation is the corresponding data type to a query. Even with a unit, there may still be some misunderstandings. Such as in the case of the measurements of CO<sub>2</sub> and VOC

concentrations in the indoor air quality monitoring application, wherein both forms of sensors provide outputs in units of parts per million (ppm) (Abraham, Li, 2014). The trend follows that where there may exist different output types (“Metadata: Observation Type”), it is recommended represent these categories.

- **Accuracy of sensor data is a crucial quality:** Multiple use cases mention the need to obtain accurate information such as being able to precisely locate firefighters in rescue missions. When discussing sensor networks, a major characteristic of these systems is in the need for them to be fault-tolerant (Gupta et al., 2006), which refers to the idea that the system must still be able to operate to achieve its objectives and maintain its correctness even whilst a subset of sensor nodes fail. By involving a sensor’s working status (“Metadata: Status”), it forms a method to filter out misrepresentative data procured from faulty devices.
- **Sensors sometimes need to detect and respond to events:** Systems that provide autonomous services where information is periodically sampled or acquired in real-time from sensors are also typically tasked to detect events. The definition of an event is dependent on the application. Both emergency related use cases mention the ability to identify special events, such as dramatic changes of temperature in the case of fire emergencies. However, for this framework it is assumed that all processing of data will happen in the application side of the integrated system, rather than on the sensor-perception layer where the focus is on the capture and classification of raw data.

To summarise, three crucial properties related to sensors are realised: the sensor’s observation value (and unit) as well as the spatial and temporal characteristics of this data. Supplementary attributes which contribute to the correct dissemination and assessment of data, include the type of data the sensor senses, and the sensor’s working state.

#### 4.2 Importance of Spatial Context on Data Classification

One aspect that was not overtly remarked on is that of the sensing area or coverage of the sensor. Coverage is defined in this paper as the bounded spherical region around a sensor calculated as a function of its sensing radius and location, which is in line with the disk model, the most renowned sensor coverage model in literature (Wang, 2010). Whilst the sensor’s location is mentioned, this is not necessarily the same as the observation’s location. Since observations are not taken at discrete points, rather, the state of the surrounding area of a sensing device is taken, we take this space (i.e. the sensor’s coverage) as the observation’s locality. To obtain this critical insight, a sensor should have a defined coverage model within which it is able to operate. Thus, we consider a “Spatial Context: Coverage” property. This is the one of the greatest advantages of incorporating information from spatial models and what differentiates this schema from existing standards SensorML and ISO 19115.

Furthermore, since a sensor might be located at a specific position and have a specific coverage, this placement will have influences on the its measurements. Reasons for this include impacts by obstructions or other objects (Tsai, 2008) in the space within its coverage. For example, longitudinal mechanical waves, such as those detected by acoustic sensors, are influenced by properties of materials and obstacles, such as walls, it passes through. Without reference to this spatial contextual information

as supplied by 3D models, it is not possible for computers to discern the effects of nearby objects to judge data appropriately.

### 4.3 Sensor Property Profile

To answer the question “What are the fundamental properties to describe sensors?”, a sensor property profile is presented in Table 2.

Type of Information	Information Needed	Description	Rationale
Observation	Value	Value of the observation	To comprehend the input that a sensor has collected.
	Unit	Measurement unit of the observation	To comprehend the input that a sensor has collected.
	Timestamp	Date and time at which the observation was captured	To identify the temporal context of a reading and track changes in data over an interval.
Metadata	Observation Type	Category of data that the sensor measures (e.g. “occupancy count”)	To describe and classify the sensor according to the information it provides for sensor/observation discovery.
	Status	Working status of the sensor (e.g. “up”, “down”)	To identify sensors and their observations for sensor discovery and is also useful for system maintenance and monitoring.
	Sensing Range	Maximum radius of the sphere around the sensor’s centre to which it can reliably measure	To indicate the sensing capabilities of a sensor and as an input to compute its’ coverage region.
Spatial Context	Location	Description of the sensor’s position	To position a sensor and its’ observations in space to provide georeferenced data and as an input to compute its’ coverage region.
	Coverage	Description of the space surrounding the sensor where it can sense phenomena	To geolocate data according to the measured space its’ corresponding sensor can detect.

Table 2. Description of sensor properties to be used in the conceptual framework

### 4.4 Comparison with Query Decomposition Approach

The main interface with which users can interact with sensors is through queries, such as “How many meeting rooms in UNSW Main Library have been empty in the past hour?”. In their paper, Liu and Akinci (2009) identified two types of information queries – those related to sensor reading discovery and those related to contextual information discovery. Through decomposing both types of queries, they conclude that a query is made up of spatial constraints, functional constraints, temporal constraints, constraints related to target information, and a processing method. To assist with efficiency of searching, another category functional/spatial aggregation was also considered.

The shortlisted attributed listed in Section 4.4 can be associated back to the query constraints in Figure 4.

The two properties lists are comparable with the main differences due to the inclusion of accuracy-based traits and the exclusion of hardware associated interfaces in this study’s proposition, which is not a concern since the aim in contributing to a standard is to abstract those details out.

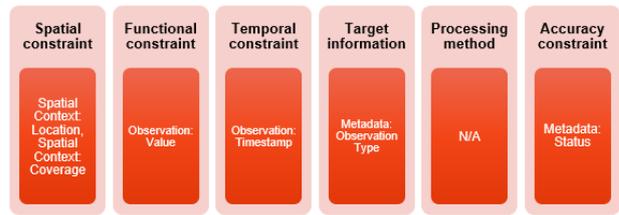


Figure 4. Relationship between sensor information and query constraints

## 5. INTEGRATED INDOORGML-SENSOR DATA MODEL

### 5.1 Sensor Module for IndoorGML

To adopt these sensor descriptions, we define a thematic extension module *Sensor* for IndoorGML. A conceptual data model for the integration of this module with IndoorGML is proposed in the form of a UML class diagram (see Figure 5), with an XML schema to be left for future work. The version of IndoorGML that was considered was version 2.0.

The *Sensor* extension covers the semantic representation of sensors within indoor spaces.

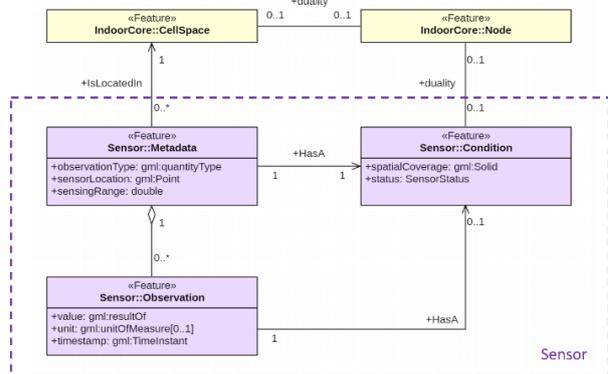


Figure 5. UML schema of proposed *Sensor* extension module for IndoorGML

Three main classes form the basis of the *Sensor* module:

1. *Metadata*,
2. *Observation*, and
3. *Condition*.

*Metadata* represents characteristics of a sensor device which have been deemed pivotal to its classification and description as justified in Section 4. It is composed of an *observationType*, which is a description of the type of physical quantity that the sensor measures such as “occupancy”, “temperature”, or “voltage”; a *sensorLocation* defines the position of the sensor with respect to a 3D coordinate system; and includes a *sensingRange* to designate the maximum distance from the sensor’s position in which it has the ability to reasonably sense.

The *Observation* class represents the reading of a sensor. A sensor may detect zero or more readings. Accordingly, an instance of *Metadata* can have zero or more *Observation* instances. Conversely, a sensor reading can only have ever been observed from one sensor, and so the relationship between *Observation* and *Metadata* has been assigned a multiplicity of 1. The *Observation* class includes a *value*, a type of *gml:resultOf*

which can specify a numeric quantity or have a reference to a file such as an image for camera-based instruments (e.g. CCTV). The *unit* (i.e. unit of measurement) is an optional parameter given a cardinality of none or one as in some instances, such as in occupancy counting, no measurement unit is involved. Each *Observation* must have a *timestamp* to denote the date and time at which it was measured.

A sensor's *Condition* is comprised of attributes which both are relevant to a sensor and its observations but may not necessarily be equivalent to one another due to the dynamic nature of the sensor. Firstly, *spatialCoverage* defines the geometry of the sensing or the sensed region for a *Metadata* and *Observation*, respectively. This will take the form of a sphere with centre at the *sensorLocation* and a boundary formed at the radius or the *sensingRange* in all directions. The coverage of a sensor might not always be the same as its observations, as is the case with mobile sensors whose location, and thus, coverage, can change with time. Secondly, a *status* attribute is prescribed to denote the *Metadata*'s changing operating condition or, for an *Observation*, it would be representative of the corresponding *Metadata*'s *status* at the time of reading and will remain constant. This is given as an enumeration of values UP, DOWN, PAUSED, UNUSUAL, and UNKNOWN<sup>1</sup>. As with coverage, a sensor's *status* is not necessarily static and so its observations cannot simply inherit this parameter from it. As a result, the new class was formed as opposed to defining coverage and status properties within both the *Metadata* and *Observation* classes to avoid duplication. It is mandatory for a sensor to have a *Condition* as enforced by the one-to-one association between the *Metadata* and *Condition* classes. However, for *Observation* and *Condition*, if an *Observation* does not have its own *Condition*, it would use the corresponding *Metadata*'s values by default. Accordingly, the association between an *Observation* and a *Condition* is given a multiplicity of none or one.

The *Sensor* module is integrated into the core module of IndoorGML via two main associations: *Metadata* to *CellSpace* and *Condition* to *Node*. In the former, the association of a *Metadata* to the *CellSpace* has a multiplicity of one as a sensor is only ever located in a single indoor space. On the other side of the relationship, a *CellSpace*, such as a room or corridor may have no sensor object within it. The association between the two classes provides a sensor with a symbolic (i.e. "Sensor 1 is in CellSpace A") representation of location, in addition to its geometrically-defined locality as stipulated through the *sensorLocation*. The second key relationship is the duality relationship between a *Node* in IndoorGML and a *Condition*. Through this association, the geometry of a sensor's coverage can be mapped into a node in the *DualSpace* as a *SpaceLayer* to signify what CellSpaces (which has an association of duality with *Node*) that the sensor covers (to some extent).

## 5.2 Use Case for Indoor Navigation

The proposed data model benefits from the consideration of a spatial coverage component for both the sensor itself and its readings. In this way, the system can be queried according to discrete positions in coordinate space, rather than just based on the semantic space descriptions.

Taking the use case of indoor navigation, the primary application domain of IndoorGML, a question that might be asked is "Can a person at (x, y, z) in Room A be localised by its sensors right

now?". If an object/feature can be localised, then navigation services such as pathfinding can be coordinated. This query may be answered through the following pseudocode in Figure 6. The procedure assumes that all localisation sensors are static at the current time, so the *Condition* parameter of the *Metadata* is used instead of that of the *Observations*.

This query may be adjusted to verify whether a given space can be covered completely by its sensors and hence, whether localisation of any object in that space is possible. This can be achieved by employing Boolean vector operations with the *Metadata*'s *Condition* instances and *CellSpace*'s geometry property. This is a relevant question to the sensor coverage problem (Argany, 2011).

```
cs = 'Room A'
person = (x, y, z)

for Metadata m in CellSpace cs:
    if m.observationType is 'localisation':
        for each Condition c of Metadata m:
            if person is within c.spatialCoverage and c.status is 'UP':
                return true

return false
```

Figure 6. Pseudocode for localisation related query

## 6. CONCLUSIONS AND FUTURE WORK

The two featured components of this research – IoT sensors and 3D indoor models – form a mutualistic relationship. From the sensor's perspective, it gains from the pairing through the offered geometric, semantic and topological interpretation of space which brings with it an understanding of the context in which a sensor operates. On the other hand, 3D indoor models, such as IndoorGML, can have access to real-time datasets about its spaces and objects to become dynamic and visual repositories of information, in what is traditionally seen as a static entity.

With the current state of research in this field in its early stages, a review of related work has exhibited that links exist to tie sensor information with existing standards for indoor space, chiefly IFC. Thus, the integration of 3D models to IoT Sensors under IndoorGML is feasible. However, to our knowledge, there is no definitive solution yet in the form of an open standard to integrate the representation of the two elements. The main contribution of this study is then to explore this possibility.

The research process towards this objective is described through a study of the core characteristics needed to represent sensors with respect to objectives of sensor and observation discovery. With this insight, we present a Sensor data model to be defined as an extension module of IndoorGML. Three main classes are developed – *Metadata*, *Observation* and *Condition* – which encompass descriptions of the chosen sensor attributes, which are associated with IndoorGML through the *CellSpace* and *Node*.

Future work is required to discover to what extent IndoorGML can support the integration of IoT sensors under this proposition through the creation of a prototype to be tested under a range of independent variables including types of sensors and use cases as well as the sensors' opportunity to be represented within a multi-layered space model. Topics such as the effect of incorporating a sensor's capabilities as part of the model for the evaluation of sensor data in terms of accuracy and reliability, an investigation into how to assess whether a sensor is able to transmit through

<sup>1</sup> Sensor status adopted from PRTG Manual ([https://www.paessler.com/manuals/prtg/sensor\\_states](https://www.paessler.com/manuals/prtg/sensor_states))

walls with respect to its sensing method and properties of the wall, and the utilisation of other sensor coverage models (e.g. sector model) are also possible areas of further research.

### ACKNOWLEDGEMENTS

The authors would like to thank Ordnance Survey GB (<https://www.ordnancesurvey.co.uk>) and 1Spatial (<https://1spatial.com/>) for sponsoring the publication of this paper.

### REFERENCES

Abraham, S., Li, X. 2014. A Cost-effective Wireless Sensor Network System for Indoor Air Quality Monitoring Applications. *FNC/MobiSPC*, 165-171.

Argany, M., Mostafavi, M. A., Karimipour, F., Gagné, C. 2011. A GIS based wireless sensor network coverage estimation and optimization: A Voronoi approach. *Transactions on Computational Science XIV*, 151-172. Springer, Berlin, Heidelberg.

Ayday, C., Safak, S., 2009. Application of wireless sensor networks with GIS on the soil moisture distribution mapping. *Symposium GIS Ostrava*, 1-6.

Claridades, A.R., Park, I. and Lee, J., 2019. Integrating IndoorGML and Indoor POI Data for Navigation Applications in Indoor Space. *한국측량학회지*, 37(5), pp.359-366.

Dave, B., Buda, A., Nurminen, A., Främling, K., 2018. A framework for integrating BIM and IoT through open standards. *Automation in Construction*, 95, 35-45.

Dawes, N., Kumar, K. A., Michel, S., Aberer, K., Lehning, M., 2008. Sensor metadata management and its application in collaborative environmental research. *2008 IEEE Fourth International Conference on eScience*, 143-150. IEEE.

Diakité, A.A. and Zlatanova, S., 2016. Extraction of the 3D Free Space from Building Models for Indoor Navigation. *ISPRS Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-2/W1, 241–248, 2016.[doi.org/10.5194/isprs-annals-IV-2-W1-241-2016](https://doi.org/10.5194/isprs-annals-IV-2-W1-241-2016)

Ding, Z., Xu, J., Yang, Q., 2013. SeaCloudDM: a database cluster framework for managing and querying massive heterogeneous sensor sampling data. *The Journal of Supercomputing*, 66(3), 1260-1284.

Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M., 2013. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7), 1645-1660.

Gupta, H., Zhou, Z., Das, S. R., Gu, Q., 2006. Connected sensor cover: Self-organization of sensor networks for efficient query execution. *IEEE/ACM transactions on networking*, 14(1), 55-67.

Jia, M., Komeily, A., Wang, Y., Srinivasan, R. S., 2019. Adopting Internet of Things for the development of smart buildings: A review of enabling technologies and applications. *Automation in Construction*, 101, 111-126.

Kang, H. K., Li, K. J., 2017. A standard indoor spatial data model—OGC IndoorGML and implementation approaches. *ISPRS International Journal of Geo-Information*, 6(4), 116.

Lee, J., Li, K., Zlatanova, S., H.Kolbe, T., Nagel, C., Becker, T., 2016. *OGC IndoorGML Version 1.02*. Document No.14- 005r5.

Liu, X., Akinci, B., 2009. Requirements and evaluation of standards for integration of sensor data with building information models. *Computing in Civil Engineering (2009)*, 95-104.

Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., Anderson, J., 2002. Wireless sensor networks for habitat monitoring. *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, 88-97.

Rio, J., Ferreira, B., & Martins, J. P. P., 2013. Expansion of IFC model with structural sensors.

Sha, K., Shi, W., Watkins, O., 2006. Using wireless sensor networks for fire rescue applications: Requirements and challenges. *2006 IEEE International Conference on Electro/Information Technology*, 239-244. IEEE.

Tang, S., Shelden, D. R., Eastman, C. M., Pishdad-Bozorgi, P., Gao, X., 2019. A review of building information modeling (BIM) and the internet of things (IoT) devices integration: Present status and future trends. *Automation in Construction*, 101, 127-139.

Theiler, M., Dragos, K., Smarsly, K., 2017. BIM-based design of structural health monitoring systems. *Proceedings of the 11th International Workshop on Structural Health Monitoring, Stanford, CA, USA (Vol. 12)*.

Tsai, Y. R., 2008. Sensing coverage for randomly distributed wireless sensor networks in shadowed environments. *IEEE Transactions on Vehicular Technology*, 57(1), 556-564.

Wang, B., 2010. Sensor coverage model. *Coverage Control in Sensor Networks*, 19-34. Springer, London.

### APPENDIX

#### Appendix A: Functional requirements and key information by application

Application	Functional requirements	Key information	Equivalent sensor properties
Soil moisture monitoring	<ul style="list-style-type: none"> <li>Ability to apply sensors in way that takes the soil properties (e.g. soil type), which can affect moisture content, into consideration.</li> <li>Ability to visualize data as a soil moisture distribution map using GIS functions to aid in location of dry areas in need of irrigation.</li> <li>Ability for sensor network to be applied to varying climate regions, soil types and plant types.</li> </ul>	<ul style="list-style-type: none"> <li>Real-time observed soil moisture</li> <li>Soil property of the field</li> <li>Location of sensors</li> </ul>	<ul style="list-style-type: none"> <li>Observation: Value and Unit</li> <li>Observation: Timestamp</li> <li>Metadata: Measured Object(s)</li> <li>Spatial Content: Location</li> </ul>
Habitat and environmental monitoring	<ul style="list-style-type: none"> <li>Ability to schedule irrigation activities with respect to data collected.</li> <li>Ability to acquire various environmental data (e.g. usage patterns of heating burners, environmental differentials between burrows and surface), that can be delivered real-time or archived for offline-use, at appropriate time intervals depending on the season, time of day etc.</li> <li>Ability for sensor system to be employed for long durations at a time and be managed and maintained remotely.</li> <li>Ability for individual sensors to wirelessly communicate and coordinate with other nodes for data transmissions, aggregation etc. purposes.</li> </ul>	<ul style="list-style-type: none"> <li>Real-time environmental data at specified sampling rates</li> <li>Type of sensor</li> <li>Sensor's condition and operating requirements</li> <li>Location of sensors</li> </ul>	<ul style="list-style-type: none"> <li>Observation: Value and Unit</li> <li>Observation: Timestamp</li> <li>Metadata: Type</li> <li>Metadata: Status</li> <li>Metadata: Operating Requirements</li> <li>Spatial Content: Location</li> </ul>
Indoor air quality monitoring	<ul style="list-style-type: none"> <li>Ability to accurately detect periodic air quality conditions with spatial and temporal references.</li> <li>Ability to support and categorize data from range of sensors types including chemical, temperature and humidity sensors.</li> </ul>	<ul style="list-style-type: none"> <li>Various air quality data from different locations at specified intervals</li> </ul>	<ul style="list-style-type: none"> <li>Observation: Value and Unit</li> <li>Observation: Timestamp</li> <li>Spatial Content: Location</li> <li>Metadata: Type</li> </ul>
Indoor emergency monitoring and navigation	<ul style="list-style-type: none"> <li>Ability to monitor the indoor environment and detect emergency event promptly and reliably.</li> <li>Ability to identify hazards and regions that may be unsafe because of the emergency event.</li> <li>Ability to provide safe navigation paths for occupants to exit the building.</li> <li>Ability to classify sensors (i.e. exit sensor, stair sensor, normal sensor) into roles based on their symbolic location.</li> <li>Ability for sensors to communicate information, such as if an emergency event has been detected, with surrounding sensors.</li> </ul>	<ul style="list-style-type: none"> <li>Environmental information</li> <li>Information about emergency events</li> <li>Location of sensors (symbolic)</li> </ul>	<ul style="list-style-type: none"> <li>Observation: Value and Unit</li> <li>Observation: Timestamp</li> <li>Observation: Event</li> <li>Spatial Content: Location</li> </ul>