# Smart Buildings Evacuation Ontology

Smart Building Evacuation Ontology (SBEO) is an ontology that couples the information about any building with its occupants such that it can be used in many useful ways. For example, indoor localization of people, detection of any hazard, a recommendation of shopping or stadium seating routes, a recommendation of safe and feasible emergency evacuation routes, or all together.

The core SBEO uses the three major concepts: topology of the building, situation awareness in the building, and the information about the physical characteristics and preferences of each person located in the building.

There are eight classes in SBEO: Person, BuildingSpace, BuildingSpaceConnection, Incident, Device, SpaceCharacteristic, Route and Distance. As the reusing the vocabulary is the essence of linked data and semantic web, therefore in SBEO, we have also reused three external vocabularies, i.e. Person from FOAF(http://xmlns.com/foaf/spec/), and both BuildingSpace and BuildingSpaceConnection from SEAS Building Ontology (https://ci.mines-stetienne.fr/seas/BuildingOntology-1.0). On the other hand, we have introduced some classes in SBEO which are the subclasses of these existing vocabularies.

**Overview of Classes and Properties:**

Here is a very brief description of these classes and how they interact with each other. Note that we have introduced eight general classes along with some subclasses within each class. The purpose of creating more subclasses is to give an idea that the vocabulary has possibilities of extension according to a specific application.

**Person:** The instances of this class are all the occupants of the building. For describing the characteristics and type of each person, Person class has further subclasses named Impaired, Normal, Visitor, Dependent, Responsible, Injured, Family. To describe more about person's Impairment, the Impaired class has further two sub-classes; PhysicalImpairment and Sensory Impairment. PhysicalImpairment has two more types; LowStamina and Wheelchair. Later on, Wheelchair class has been divided into two more sub-classes; Motorised and Non-Motorised. Similarly, SensoryImpairment class has also two subclasses; Hearing and Visual that represents the type of sensory impairments one is using.

The above-mentioned classification of persons helps to take decisions according to the type of each person. For example, a person who has a physical impairment and using a non-motorised wheelchair will be an instance of Non-Motorised class. Also, if the same person has any family tie with any other occupant of the building, then both persons will also be the instances of Family class having the same familyID. On the other hand, if some person is responsible for another person(s), one will be an instance of Responsible class. Note that the usage Responsible and Dependent classes are beyond the family ties. It means one may be Responsible or Dependent on other persons despite the fact that they have any family ties among them.

In addition, if a person is visiting any building, then one will be an instance of Visitor class along with the other characteristics one may have.

As the main motivation of SBEO is to conform the building topology to the physical characteristics and preferences of persons, therefore most of the properties of this ontology are applied on Person class. For example, firstly, basic things such as, first name (foaf:firstName), last name (foaf:lastName) and age (foaf:age), are taken. Later on, each person is provided by a unique ID(sbeo:hasID). Also, if two or more persons have family ties among them, all of them are provided by a unique familyID so that they can be traced out easily. Secondly, persons who are bound to stay together are assigned by a unique groupID as well as some other properties such as sbeo:reponsibleTo and sbeo:dependentOn, which are inverse of each other. Thirdly, an extra layer of information is added for Person class that tells either a person is familiar with the building (sbeo:familiarityWithBuilding) or needs any supervision (sbeo:needsSupervision). Lastly, there are some properties that can be used to describe to momentary situation of each person such as fitness level (sbeo:fitnessLevel), momentary location (sbeo:isLocatedIn) and in case of any emergency evacuation, the evacuation status (sbeo:evacuationStatus) can be known.

Some of the properties can be used with timestamp property, for example, the location of a person or evacuation status of a person at a specific time.

**BuildingSpace and BuildingSpaceConnection:** We have reused these classes from the vocabulary of SEAS Building Ontology. In this regard, the instances of subclasses of BuildingSpace and BuildingSpaceConnection classes can be used to describe the topology of a building such as rooms, offices, halls, corridors, doors, etc. In addition, we have introduced two new subclasses of Room class that is a direct child class of BuildingSpace class. These new classes are KidsArea and DayCare.

To get the information about the topology of the building and how does it relate with its occupants, these two classes and their subclasses are used as domains or ranges of different properties. For example, sbeo:connectedTo property is used to show a connection between two spaces or space connections. Similarly, the adjacency of two spaces are also describes using sbeo:adjacentTo property. In addition, the location of each person is also described by using sbeo:locatedIn property that has a range of building space. Similarly, building space is also used as a range of sbeo:installedIn property to mention the position of any sbeo:Device.

Furthermore, there are some advanced properties that are used describe the total accommodation capacity and the current occupancy of any space in terms of persons (i.e. sbeo:accommodationCapacity and sbeo:currentOccupancy).

**Incident:** The instance of this class represents an incident that occurred in the building. To classify the type of incidents, the Incident has further subclasses: ChemicalSpill, TerroristAttack, Congestion, Fire, Panic, Earthquake. As the Panic may occur due to several reasons, therefore we have classified it using a Stempeding class.

Properties such as sbeo:locatedIn, use the instances of the subclasses of Incident class as a domain whereas the range is either building space or building space connection. It means a specific incident is occurred at a particular part of the building. The instances are also

associated with a timestamp property (sbeo:atTime) to get the exact time of the occurrence of any incident.

**Device:** This is a very general class to represent all physical devices such as Sensor, MobilePhone, DisplayScreen, IncidentProtection, etc. The breadth of Device class and its subclasses can be extended to any level with respect to the depth of application. Each subclass of Device class has different types of instances. For example, the Sensor class has three further subclasses: LocationSensor, TemperatureSensor, SmokeSensor. Similarly, in IncidentProtection class, there is a FireProtection class that has further two subclasses; ActiveFireProtection and PassiveFireProtection. In ActiveFireProtection class, there is a FireDoor class wherein PassiveFireProtection class, FireExtinguisher class.

The instances of the subclasses of Devices use different properties with respect to their nature of use. For example, the instances of IncidentProtection class only use sbeo:installedIn and sbeo:locatedIn to mention where these devices are located. On the other hand, the instances of Sensor class use sbeo:hasValue sbeo:hasSmoke properties because the values of these devices can be used according to the application. In addition, the timestamp property (sbeo:atTime) is also used to add the timestamp to the values detected by sensors.

**SpaceCharacteristic:** This class contains characteristics that a BuildingSpace or a BuildingSpaceConnection may have. The purpose of coining this class in SBEO is to add an additional layer for the instances of BuildingSpace and BuildingSpaceConnection classes. It includes Available, Unavailable, Entrance, Exit. The Available class has SafeZone class as well that has two types of classes in it; WaitingZone and AssemblyPoint. The Exit class has subclass called EmergencyExit that is a specific case of it as it is not as usable as normal exits.

The instances of building space and building space connections classes as well as their subclasses can also be the instances of the subclasses of SpaceCharacteristic class. For example, in case of any hazard, space or space connection can become an instance of Unavailable class. Note than Available and Unavailable classes are disjoint with each other that means the instance of one class cannot be the instance of others at the same time.

In addition, SafeZone is a subclass of Available class that means space can be SafeZone that must be Available all the time. The SafeZone has further two subclasses: WaitingZone and AssemblyPoint. These concepts are specifically used for emergency evacuation purposes.

**Route:** The instances of this class represents the routes that start from any building space and end in any other building space or building space connection.

The instances of Route class are used to describe the details of a route in which the starting and the ending space or space connections are mentioned using sbeo:startsFrom and sbeo:endIn respectively. Also, the complete path is given using sbeo:hasPath property in each instance of a Route class.

**Distance:** The instances of this class represents the distance between two BuildingSpaces or BuildingSpace and BuildingSpaceConnection.

The instances of Device class use sbeo:distanceOf property to describe the distance between two building spaces, a building space and building space connection or two building space connections. Each instance uses a sbeo:distanceOf property twice(once for each space or space connection) to mention the distance.

**Data Properties:**

1. **sbeo:hasLength**
   It represents the length of a route in terms of time (in seconds) or distance (in meters). More explanation of the route can be found in sbeo:Route class.

   **Domain Includes:** sbeo:Route
   **Range:** xsd:integer
   **Defined by:** sbeo:

2. **sbeo:hasID**
   It represents an unique identification number for each occupant of the building. This ID is used to calculate all the functionalities for a specific person, e.g. location, characteristics, preferences, etc.

   **Domain Includes:** foaf:Person
   **Range:** xsd:integer
   **Defined by:** sbeo:

3. **sbeo:familyID**
   It represents a unique identity number (ID) for persons having family ties. All the members of a family have one common familyID.

   **Domain Includes:** foaf:Person
   **Range:** xsd:integer
   **Defined by:** sbeo:

4. **sbeo:groupID**
   It represents an unique identification number (ID) for persons who have a same group for moving from one space to another. All the members of a group have one common groupID.

   **Domain Includes:** foaf:Person
   **Range:** xsd:integer
   **Defined by:** sbeo:

5. **sbeo:fitnessLevel**
   This property is used to represent the fitness level of a person. For example, there are 5 levels. Upper three levels are appropriate and considered as safe whereas the below two levels are considered as dangerous and a particular person needs assistance. The person is considered as sbeo:Injured, if the levels are changed from upper three to lower two.
   **Domain Includes:** foaf:Person

**Range:** xsd:integer
**Defined by:** sbeo:

### 6. *sbeo:atTime*

It is used to represent a timestamp with any particular event. It is a calculated as Unix epoch. For example, when a specific person is located in a specific space, or a specific sensor detects a specific value, etc.

**Domain Includes:** foaf:Person, sbeo:Incident, sbeo:Sensor
**Range:** xsd:long
**Defined by:** sbeo:

### 7. *sbeo:needsSupervision*

This property is used to mention either a person needs supervision or not. As its value is in Boolean, therefore, the true means that a person needs the supervision whereas a false means (s)he does not.

**Domain Includes:** foaf:Person
**Range:** xsd:boolean
**Defined by:** sbeo:

### 8. *sbeo:evacuationStatus*

It represents the status of evacuation of each person. This property is specially used for evacuation purposes. Its value can be either 'not started' or 'in progress' or 'completed'. The value remains 'not started' in two cases. The first case, when there is no hazard and emergency evacuation has been started. The second case, when a hazard is detected and the emergency evacuation has not been started by person; both intentionally and unintentionally. The value of this property becomes 'in progress' when a person starts to move from one's initial position and remains as it is until one has not arrived its assigned exit. Once a person arrives the exit, the evacuation status changes to 'completed', that represents the evacuation process has been finished for that specific person.

**Domain Includes:** foaf:Person
**Range:** xsd:string
**Defined by:** sbeo:

### 9. *sbeo:familiarityWithBuilding*

It represents whether a person has some level of familiarity with the building geometry or not. As its value is in Boolean, therefore, the true means that a person is familiar with the building whereas a false means (s)he is not.

**Domain Includes:** foaf:Person
**Range:** xsd:boolean
**Defined by:** sbeo:

### 10. *sbeo:accommodationCapacity*

It represents the accommodating capacity (in terms of persons) of any particular space.

**Domain Includes:** seas:BuildingSpace, sbeo:SafeZone
**Range:** xsd:integer
**Defined by:** sbeo:

### 11. sbeo:currentOccupancy

It represents the current occupancy (in terms of persons) of any particular space.

**Domain Includes:** seas:BuildingSpace, sbeo:SafeZone
**Range:** xsd:integer
**Defined by:** sbeo:

### 12. sbeo:hasDistance

It represents the distance (in meters) between any two space connections or any exit/entrance and a space connection.

**Domain Includes:** seas:BuildingSpaceConnection, sbeo:Exit, sbeo:Entrance
**Range:** xsd:integer
**Defined by:** sbeo:

### 13. sbeo:hasValue

It represents the value of detected by any sensor or device.

**Domain Includes:** sbeo:Device
**Range:** xsd:integer
**Defined by:** sbeo:

### 14. sbeo:hasSmoke

This property is used to describe if a smoke sensor detects smoke or not. As its range is Boolean, therefore the true means the smoke is detected and the false means there is no smoke detected by a smoke sensor.

**Domain Includes:** sbeo:SmokeSensor
**Range:** xsd:boolean
**Defined by:** sbeo:

### 15. foaf:lastName

The last name of the person.

**Domain Includes:** foaf:Person
**Range:** xsd:string
**Defined by:** foaf:

### 16. foaf:firstName

The first name of the person.

**Domain Includes:** foaf:Person
**Range:** xsd:string
**Defined by:** foaf:


### 17. *foaf:age*
The age of the person.

**Domain Includes:** foaf:Person
**Range:** xsd:integer
**Defined by:** foaf:


## Object Properties:

### 1. sbeo:endsIn
This property is used to express the last building space or building space connection where any route ends. Generally, these are exits but if two routes are coupled together then the sbeo:endsIn of one route can be the sbeo:startsFrom of another route or the sbeo:endsIn of a route can be any element of sbeo:hasPath.

**Domain Includes:** sbeo:Route
**Range Includes:** sbeo:Exit, sbeo:Entrance, seas:BuildingSpace, seas:BuildingSpaceConnection, sbeo:Route
**Inverse of:** sbeo:startsFrom
**Characteristics:**
**Defined by:** sbeo:


### 2. sbeo:startsFrom
This property is used to express the building space or building space connection where any route starts.

**Domain Includes:** sbeo:Route
**Range Includes:** sbeo:Exit, sbeo:Entrance, seas:BuildingSpace, seas:BuildingSpaceConnection, sbeo:Route
**Inverse of:** endIn
**Characteristics:**
**Defined by:** sbeo:


### 3. sbeo:hasPath
This property is used to express the set of connected building spaces or building space connections that makes a route. In this regard, when this property is used as a predicate in a triple, the object of that triple must be a rdf:list.

**Domain Includes:** seas:BuildingSpace, seas:BuildingSpaceConnection
**Range Includes:** seas:BuildingSpace, seas:BuildingSpaceConnection
**Characteristics:**
**Defined by:** sbeo:

#### 4. *sbeo:assignedRoute*

This property is used to represent a route (rdf:list of connected spaces) that is assigned to any  person or a group of persons.

**Domain Includes:** foaf:Person
**Range Includes:** sbeo:Route
**Inverse of:**
**Characteristics:**
**Defined by:** sbeo:

#### 5. *sbeo:connectedTo*

This property is used to mention the connectivity of any building space or building space connection with each other.

**Domain Includes:** seas:BuildingSpace, sbeo:BuildingSpaceConnection, sbeo:Exit, , sbeo:Entrance
**Range Includes:** seas:BuildingSpace, sbeo:BuildingSpaceConnection, sbeo:Exit, , sbeo:Entrance
**Characteristics:** symmetric
**Defined by:** sbeo:

#### 6. *sbeo:responsibleTo*

This property is used to mention the responsibility for a particular person to his/her dependent persons. When this property is used as a predicate in any triple, the object of that triple has the IRI of the instance of dependent person associated with the subject of this responsible person.

**Domain Includes:** foaf:Person
**Range Includes:** foaf:Person
**Inverse of:** sbeo:dependentOn
**Characteristics:** transitive
**Defined by:** sbeo:

#### 7. *sbeo:dependentOn*

This property is used to mention the dependency of a particular person on his/her responsible person. When this property is used as a predicate in any triple, the object of that triple has the IRI of the instance of responsible person who has the responsibility of this dependent person.

**Domain Includes:** foaf:Person
**Range Includes:** foaf:Person
**Inverse of:** sbeo:responsibleTo
**Characteristics:** transitive
**Defined by:** sbeo:

#### 8. *sbeo:accompanying*

This property is used to mention how who is accompanying who in a common building space.

**Domain Includes:** foaf:Person
**Range Includes:** foaf:Person
**Characteristics:** transitive, symmetric
**Defined by:** sbeo:

### 9. sbeo:acquaintanceOf

This property is used to represent any friend or acquaintance tie among persons.

**Domain Includes:** foaf:Person
**Range Includes:** foaf:Person
**Characteristics:** symmetric
**Defined by:** sbeo:

### 10. sbeo:adjacentTo

This property is used to mention the building spaces that are adjacent to a specific space by a wall. This property is helpful for security purposes, for example, in case of any emergency situation when the building space connection, i.e. door, windows, are not accessible.

**Domain Includes:** seas:BuildingSpace
**Range Includes:** seas:BuildingSpace
**Characteristics:** symmetric
**Defined by:** sbeo:

### 11. sbeo:locatedIn

This property is used to mention the existence of anything (e.g. person, or stairs, doors, etc.in a particular building space.

**Domain Includes:** foaf:Person, seas:BuildingSpaceConnection, seas:BuildingSpace, sbeo:Incident
**Range Includes:** seas:BuildingSpace, seas:BuildingSpaceConnection
**Characteristics:** transitive
**Defined by:** sbeo:

### 12. sbeo:installedIn

This property is used to mention any device, system, or instrument, installed in a building space.

**Domain Includes:** seas:BuildingSpace, seas:BuildingSpaceConnection, sbeo:Device
**Range Includes:** seas:BuildingSpace, seas:BuildingSpaceConnection
**Characteristics:** transtive
**Defined by:** sbeo:

### 13. sbeo:distanceOf

This property is used to represent the distance between any two connected building spaces or building space connections.

**Domain Includes:** sbeo:Distance
**Range Includes:** seas:BuildingSpace, sbeo:BuildingSpaceConnection, sbeo:Exit, , sbeo:Entrance
**Characteristics:**
**Defined by:** sbeo:

### 14. *sbeo:notAccessibleTo*

This property is used to prohibit the access of any specific person to a specific building space or building space connection.

**Domain Includes:** seas:BuildingSpace, seas:BuildingSpaceConnection
**Range Includes:** foaf:Person
**Characteristics:**
**Defined by:** sbeo: