CITI-SENSE	SEVENTH FRAMEWORK PROGRAMME	Project title: Development of sensor-based Citizens' Observatory Community for improving quality of life in cities Acronym: **CITI-SENSE** Grant Agreement No: **308524**
		EU FP7- ENV-2012 Collaborative project

Deliverable D 7.2

Core ontology network for city observatories applications

Work Package 7

Date: 26.11.2013

Version: 2.11

Leading Beneficiary:	TECNALIA
Editor(s):	Marta González (TECNALIA)
Author(s) (alphabetically):	Arne J. Berre (SINTEF), Marta González (TECNALIA), Urtza Iturraspe (TECNALIA), Tony Lee (SALTLUX), Izaskun Mendía (TECNALIA), Dimitru Roman (SINTEF)
Dissemination level:	Public

Versioning and contribution history

Version	Date issued	Description	Contributors
1	Nov 12	TOC & First version	Marta González, Urtza Iturraspe
1.5	Feb 13	Inclusion of IOT-A, SenML and AirBase	Marta González
2	Feb 13	Annex II Guidelines for modelling ontologies within CITI-SENSE. Annex III: Upper ontologies,	Tony Lee
2.5	Feb 13	Inclusion of INSPIRE directive	Marta González
2.6	March 13	Inclusion of ISO 19150	Izaskun Mendía
2.7	April 13	Citizen Observatories Ontology implementation	Marta González
2.8	May 13	Format changed and inclusion of requirements from other WPs	Marta González
2.9	July 13	INSPIRE directive ontology implementation, chapter added about advantages of using ontologies and linked data	Marta González
2.10	September 13	General review	Marta González, Dimitru Roman.
2.11	November 13	Internal reviews incorporation	Marta González

Peer review summary

Internal review 1			
Reviewer	Jasmin Pielorz (UBIMET)		
Received for review	20.09.2013	Date of review	04.11.2013

Internal review 2			
Reviewer	Richard Rombouts (Snowflake)		
Received for review	20.09.2013	Date of review	15.11.2013

Executive Summary

This document describes the technical implementation of the "core ontology network for city observatories and applications" performed inside Task 7.1." Its goal is:

"(...) to reconcile various initiatives in ontology development for the creation of city observatories apps. This task will take as starting point the Semantic Sensor Network ontology developed in the context of the W3C incubator group on Semantic Sensor Networks and investigate possible extensions towards capturing aspects such as time, geography, city-related resources, etc. Besides, ontologies focused on the topics of the use cases will be developed in such a manner that they can be also applied in other similar contexts, that is, modularization will be considered as a key element in this process."[1]

One of the objectives of CITI-SENSE is to develop a concept for so-called city observatories. For this purpose diverse pilots have been defined and will be deployed, but several questions still need to be addressed: How do we deal with the data these pilots collect? And can it be gathered and provided in a standard way such that citizens and stakeholders can benefit or do we need to develop new concepts? In order to address this, we develop a new CITI-SENSE ontology which will be used to annotate the data gathered by sensors, calculated data (over sensor data) and responses to questionnaires, with the aim to disseminate these data and to provide interoperability among these heterogeneous data sources. The dissemination will be achieved by the publication of the CITI-SENSE observatories dataset under the Linked Data paradigm as open data, linking it to other linked datasets in the Linked Open Data Cloud[2].

This deliverable provides a detailed study of ontologies in the domain of data collected by sensors, as well as an overview of the INSPIRE Directive[3] which specifies how to share environmental data using common data specifications and internationally recognised open standards. In CITI-SENSE, the INSPIRE Directive has been partially formalised because no existing available ontologies, formalising INSPIRE Directive, have been found.

Requirements for the CITI-SENSE ontology come from work packages 2 and 3 who are in charge of the empowerment initiatives and the architecture to be defined in work package 7. The future users of the ontology will be the members of work packages 6 and 7.

In chapter 4 a shortlist of ontologies and other non-ontological resources is described. These resources have been selected according to their relevance in the current state of the art and suitability to the knowledge domain the city observatories must cover. This knowledge domain is covered by different aspects that match with the different subsections of chapter 4:

- Standards, resources and directives in sensors, observations and measurements and web services.
- Locations and space.
- Time modelling.
- Units of measure.
- Ontologies in the sensor domain.

[1] CITI-SENSE DoW
[2] http://linkeddata.org/
[3] http://inspire.jrc.ec.europa.eu/

Chapter 5 provides an overview of the different decisions taken to create the city observatory ontology reflecting the selected ontologies along with the reasons of such decision. This chapter also includes the guidelines for applying the ontology and how the ontology fulfils the requirements coming from chapter 3.

The city observatories ontology will adopt the Semantic Sensor Network (SSN) ontology as the upper ontology. The SSN ontology is considered as de facto standard in the sensor world, so this ontology ensures future enlargement of CITI-SENSE knowledge domain as well as makes possible links to other existing efforts in sensor data gathering and publication.

But the SSN ontology leaves the observed domain unspecified that is why other ontologies have been selected and aligned with the SSN ontology, where each ontology has been selecting according to a concrete necessity:

- **When are we measuring?** The answer to this question is modelled by the OWL Time ontology, a W3C-recommended ontology based on temporal calculus, that provides descriptions of temporal concepts such as instant and interval and which supports defining interval queries such as within, contains, and overlaps.
- **Where are we measuring?** The INSPIRE directive defines how the different locations should be modelled. The scope as defined in the INSPIRE directive includes two aspects: the environmental monitoring facility as a spatial object in the context of INSPIRE and Observations and measurements linked to the environmental monitoring facility.
- **What are we measuring**? The answer to this question covers the CITI-SENSE knowledge domain and it is represented by the requirements coming from the empowerment initiatives under development in Work packages 2 and 3.
- **Who is and how are we measuring?** The citizens actively involved in sensor measurements are also an important part because their social and personal context (e.g. age, occupation) can influence the analysis of the gathered sensor data.

The city observatories ontology provides the vocabulary to annotate the available resources for each pilot: the description of the place where to perform the observations and the person in charge of such observations, to describe the different sensors and their measurements as well as the calculated data values from sensor data.

The ontology has been implemented following the network ontology approach, trying to reuse as much as possible existing efforts in the ontology, linked data, sensor and Internet of Things (IoT) domains. One of the requirements was to be compliant with the INSPIRE Directive. From this directive, the data specifications, relevant to CITI-SENSE, have been implemented: Geographical Names, Utility and Governmental Services, Buildings, Addresses and Land Use. These data specifications, in some cases have not been completely implemented but only relevant parts.

The city observatories ontology is ready to annotate datasets with sensor observations, calculated data, people in charge of the observations, location and time published under the Linked Data paradigm[4] in the Linked Open Data Cloud[5]. Thanks to the reuse of well-known and widely used ontologies such as SSN[6] and GeoNames[7], the links to existing published datasets as GeoNames[8] and Linked Sensor Data[9] are possible without modifications of the ontology.

[4] http://www.w3.org/standards/semanticweb/data
[5] http://linkeddata.org/
[6] http://www.w3.org/2005/Incubator/ssn/ssnx/ssn
[7] http://www.geonames.org/ontology/documentation.html

The city observatories ontology provides different ways of addressing a location: it is either indicated its address, through an element of the GeoNames database or by the according GPS coordinates or any other geometry as defined by the GML standard[10].

The city observatories ontology is open for future evolution, allowing in particular the addition of new sensors, measure or unit of measure types where needed. Since this can be achieved by adding new instances to the ontology, these changes require no modification of its core structure.

The next step should be the inclusion in the CITI-SENSE architecture of a module able to annotate with the city observatories ontology, the data served by the different pilots. And establishing links from entities in CITI-SENSE dataset to other entities in already published datasets.

Future evolution of the ontology in the short-term is focused in the experience after its use in CITI-SENSE and once the incomplete requirements be defined. In medium/long term, the evolution is focused on the implementation of more data specifications from INSPIRE Directive.

[8] http://datahub.io/dataset/geonames
[9] http://datahub.io/dataset/knoesis-linked-sensor-data
[10] http://www.opengeospatial.org/standards/gml

Table of contents

Abbreviations and Acronyms

Abbreviation / Acronym	Description
GCI	GEOSS Common Infrastructure
GEOSS	Global Earth Observation System of Systems
HILUCS	Hierarchical INSPIRE Land Use Classification System
INSPIRE	Infrastructure for Spatial Information in the European Community
IoT	Internet of Things
ISO	International Organization for Standardization
LAeq	Equivalent continuous sound level with A-weighting
LAmax	Maximum value that the A-weighted sound pressure level reaches during a measurement period
LAmin	Maximum value that the A-weighted sound pressure level reaches during a measurement period
LOD	Linked Open Data
L90	The level just exceeded for 90% of the time of measurement or analysis
O&M	Observations and Measurements
OGC	Open Geospatial Consortium
OMG	Object Management Group
OWL	Web Ontology Language
RDF	Resource Description Framework
REST	Representational State Transfer
SenML	Media Types for Sensor Markup Language
SensorML	Sensor Model Language
(S)KOS	(Simple) Knowledge Organisation System
SPARQL	SPARQL Protocol And RDF Query Language
SQL	Structured Query Language
SSN	Semantic Sensor Network Ontology
SWE	Sensor Web Enablement
SWEET	Semantic Web for Earth and Environmental Terminology
UML	Unified Modelling Language
WP	Work package

1. Introduction

1.1 Purpose of the document

This document describes the technical implementation of the "core ontology network for city observatories and applications" performed inside Task 7.1." Its goal is

"(...) to reconcile various initiatives in ontology development for the creation of city observatories apps. This task will take as starting point the Semantic Sensor Network ontology developed in the context of the W3C incubator group on Semantic Sensor Networks and investigate possible extensions towards capturing aspects such as time, geography, city-related resources, etc. Besides, ontologies focused on the topics of the use cases will be developed in such a manner that they can be also applied in other similar contexts, that is, modularization will be considered as a key element in this process."[11]

The requirements for such ontology come from work packages 2 and 3 who are in charge of the empowerment initiatives, while work package 7 provides the framework for the architecture. Future users of the ontology will be the work packages 6 and 7.

The document is structured such that it reflects the process followed to obtain the "core ontology network for city observatories and applications":

- The document begins with an explanation of the benefits that ontologies add to a project like CITI-SENSE, benefits perfectly extrapolated to other IoT environments. Linked Data is introduced as a way to reach real citizen observatories: providing a public and standardized access to the gathered data by the different empowerment initiatives. (Chapter 2)
- In any IT environment it is mandatory to be compliant with the requirements, in the case of ontologies the same case appears. Collecting and analysing requirements from the various empowerment initiatives has been crucial. (Chapter 3)
- Once the requirements have been gathered it was necessary to look for existing, relevant and, when possible, standard ontologies or non-ontological resources that can be reused in order to fulfil with such requirements. (Chapter 4)
- With the analysis of the above resources in one hand and the requirements in the other, the city observatories ontology is implemented. (Chapter 5)
- Once the ontology is implemented a usage example will be developed in order to show how the ontology satisfies the requirements. (Section 5.4)

1.2 Detailed overview of objectives

The CITI-SENSE project will develop and test the concept of so-called "citizens' observatories" that empower citizens to contribute to and participate in environmental governance, to enable them to support and influence community and societal priorities and associated decision making.

To demonstrate this concept three pilot case studies (or empowerment initiatives) are developed. From the technical perspective the pilots focus on developing a range of services related to environmental issues of societal concern: combined environmental exposure and health associated

[11] CITI-SENSE DoW

with air quality; noise and development of public spaces, and indoor air at schools. These empowerment initiatives will be developed in work packages 2 and 3.

The core ontology network for city observatories and applications should be able to represent all data (sensor data, calculated data, personal perception) gathered by the different empowerment initiatives in order to make them exploitable by different applications. Based upon the requirements formulated by work packages 2 and 3, these applications will be developed in work packages 6 and 7.

Using a common ontology allows interoperability of the different empowerment initiatives, where heterogeneous sensors, human source data and calculations are foreseen. Chapter 2 is specifically aimed at explaining the benefits of using ontologies.

If the data produced by empowerment initiatives is annotated with the ontology developed in this deliverable, it will be possible to publish the CITI-SENSE dataset in the Linked Data Cloud[12]. In this case it will be mandatory to link our data to other published datasets that contain for instance environmental and/or forecasting information. Section 2.3 in this deliverable tries to widen this aspect.

This document collects and analyses the requirements coming from the empowerment initiatives. The result of the analysis is the input for the search, analysis and selection of existing and relevant ontological and non-ontological resources that allow coverage of the empowerment initiatives knowledge domain. When a non-ontological resource is selected, it is necessary to transform it to an ontology, this is applicable in the case of the INSPIRE Directive[13].

The final task consists of aligning all the ontologies in order to obtain a networked ontology that covers the whole knowledge domain and meets the requirements.

1.3 Audience

The intended audience of this document are all stakeholders of the CITI-SENSE project who are interested in the semantic aspects of CITI-SENSE in particular. Within this deliverable we distinguish between the following target groups:

1. CITI-SENSE project members / partners (in particular technical readers).
2. The European Commission, who supports, and partially funds, the CITI-SENSE project.
3. External stakeholders who want to understand the CITI-SENSE approach to semantics.

Guidelines for reading:

Purpose	Sections of interest
Quick overview	The Executive summary will be a good starting point together with chapter 2.
CITI-SENSE technical	Chapter 5 explains how the ontology has been implemented, while chapter 4 reflects the previous needed analysis labour.
CITI-SENSE non-technical	Chapter 2 and 3 give a good introduction to semantics and a global overview of the empowerment initiatives.

[12] http://lod-cloud.net/
[13] http://inspire.jrc.ec.europa.eu/

2. Ontologies and Linked Data in CITI-SENSE

The World Wide Web was built for human but not for machine consumption. Therefore everything on the Web is machine-readable but not necessarily machine-understandable. Humans can process only a tiny fraction of the information available on the Web. [1]

The ultimate goal of the **Web of data**[14] is to enable computers to do more useful work and to develop systems that can support trusted interactions over the network. The term "**Semantic Web**" refers to W3C's vision of the Web of linked data. Semantic Web technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Linked data are empowered by technologies such as RDF, SPARQL, OWL, and SKOS.

On the Semantic Web, **vocabularies**[15] define the concepts and relationships (also referred to as "terms") used to describe and represent an area of concern. Vocabularies are used to classify the terms that can be used in a particular application, characterize possible relationships and define possible constraints for using those terms. In practice, vocabularies can be very complex (with several thousands of terms) or very simple (describing one or two concepts only).

There is no clear division between what is referred to as "**vocabularies**" and "**ontologies**". The trend is to use the word "ontology" for more complex, and possibly quite formal collection of terms, whereas "vocabulary" is used when such strict formalism is not necessarily used or only in a very loose sense. Vocabularies are the basic building blocks for inference techniques on the Semantic Web.

2.1 Benefits of ontologies

An ontology is a formal specification of shared concepts within a domain and the relationships between those concepts. The word is based on the ancient Greek ontos and logos, and it basically means "the study of that which truly is." In the 18th century, ontology became the term used for the branch of philosophy that deals with the nature and structure of "reality."

Probably the first notable ontologist was Aristotle (384-322 BC) who among other things pursued the question of what can be known about something. His solution is presented in his "Categories" and can be seen as the first upper-level ontology. In his work he distinguishes between the following ten categories:

1. Substance
2. Quantity
3. Quality
4. Relation
5. Place
6. Time
7. Situation
8. Condition
9. Action
10. Affection

Which according to Aristotle's point of view suffice to say anything that can be known about something. They present the essential qualities that matter, everything else can be subsumed into one of those [23].

[14] http://www.w3.org/standards/semanticweb/
[15] http://www.w3.org/standards/semanticweb/ontology

Ontologies can be used to define the key concepts that enable interoperability, allowing heterogeneous applications to share a common understanding of the domain. The importance of standardization as a means to achieve interoperability is growing. Within this broad area, the topic of semantic information system (IS) standards and interoperability is relatively new[16]. The arrival of XML, a standard foundation, has boosted the development of B2B standards. Nowadays, XML based standards are common, since they involve fewer costs in comparison with Electronic Data Interchange (EDI)[17] standards for message interchange between organisations. Many of the latest trends like web services, service-oriented architectures, cloud computing, etc. are dependent on standards to fulfil their promise.

Semantic IS standards are designed to promote communication and coordination among the organizations; these standards may address product identification, data definitions, business document layout, and/or business process sequences. Besides specific horizontal (EDI, cXML, OAGIS, xCBL, UBL, GS1 XML, ebXML) and vertical (RosettaNet for electronics, CIDX for chemicals, etc.) semantic IS standards, there are also standards that can be used to describe (part of) the semantics that have to be defined by the standard. These include XML, UML, OWL, BPEL, BPMN and other similar types of standards.

The majority of the methodologies used in the corporate sector are focused on structured data integration (mainly databases), while semantics can be used to integrate all types of information assets, both structured and unstructured.

The key role of ontologies with respect to database systems is to specify a data modelling representation at a level of abstraction above specific database designs (logical or physical), so that data can be exported, translated, queried, and unified across independently developed systems and services. Successful applications to date include database interoperability, cross database search, and the integration of web services [24].

Typical questions asked in an enterprise setting are: "What business problem is semantics solving? What relevance is there between semantics and the real world? What value proposition for the conduct of business is there that is being addressed by semantics?"[2]. According to [3], ontologies allow the enterprise applications to benefit from different points of view:

- Innovative Business Scenarios: ontologies facilitate hitherto impossible or hard to achieve business scenarios.
- Increased Productivity of Information Workers: The productivity of an information worker can be increased by enabling more efficient access to required information.
- Improved Enterprise Information Management: Enterprise Information Management (EIM) is to find solutions for optimal use of information within organizations, for instance to support decision-making processes or day-to-day operations that require the availability of knowledge.
- Increased Productivity of Software Engineering. Full Automation of Web service discovery and composition. Ontologies are also applied to more established software engineering in order to achieve cost and time reduction and finally for quality Improvements which are often sought in software engineering.

[16] Folmer, Erwin and Verhoosel, Jack (2011) State of the Art on Semantic IS Standardization, Interoperability & Quality
[17] http://www.unece.org/trade/untdid/welcome.html

In the IoT world, ontologies appear as a suitable alternative to exchange knowledge [25], since as mentioned above they allow that heterogeneous applications exchange data, which is one of the objectives of IoT.

2.2 Benefits of the Linked Data approach

The web contains lots of information, but typically the raw data itself is not available - rather only HTML documents constructed from data, if a website is generated from a database at all. We are surrounded by data – data about the performance of our local schools, the fuel efficiency of our cars, a multitude of products from different vendors, or the way our taxes are spent. By helping us making better decisions, this data is playing an increasingly central role in our lives and driving the emergence of a data economy. Increasing numbers of individuals and organizations are contributing to this deluge by choosing to share their data with others, including Web-native companies such as Amazon and Yahoo!, newspapers such as The Guardian and The New York Times, public bodies such as the UK and US governments, and research initiatives within various scientific disciplines.

The semantic web seeks to change the shape of the internet with regard to this problem in a number of ways:

- Opening up the web of data to artificial intelligence processes (getting the web to do a bit of thinking for us).
- Encouraging companies, organisations and individuals to publish their data freely, in an open standard format.
- Encouraging businesses to use data already available on the web (data give/take).

The Linked Data approach offers significant advantages over current practices for creating and delivering data while providing a natural extension to the collaborative sharing models historically employed.

Linked Data and especially Linked Open Data are sharable, extensible, and easily re-usable. It supports multilingual functionality for data and user services, such as the labelling of concepts identified by a language-agnostic URIs. These characteristics are inherent in the Linked Data standards and are supported by the use of Web-friendly identifiers for data and concepts.

Like the linking that takes place today between Web documents, Linked Data allows anyone to contribute unique expertise in a form that can be reused and recombined with the expertise of others. The use of identifiers allows diverse descriptions to refer to the same thing.

2.3 Linked Data in CITI-SENSE

Too often[18] information is made available as lists of figures or spread sheets that only experts can interpret. To encourage and benefit from participation we need to present our information in a way everyone can understand.

They key to protecting and enhancing our environments is in the hands of the many, not the few. Although our political, economic and administrative structures may be designed to tackle our environmental concerns through scale and strategic decisions, it often leaves citizens as uninvolved and silent observers.

[18] http://www.eea.europa.eu/pressroom/speeches/global-citizen-observatory-the-role-of-individuals-in-observing-and-understanding-our-changing-world

One of the targets of CITI-SENSE is to develop and test the concept of city observatories. For this aim diverse pilots have been defined and will be deployed. But how do we deal with the data they collect? How can this data be gathered and provided in a standardized way so citizens and stakeholders can make profit from it?

Visual analytics is a key component for data interpretation, but data storage and access consequently, too. Linked Data is the technology that allows publishing data in a standardized way and linking to other data sources that extend it. For instance, while a pilot is gathering information about thermal comfort (objectively measured with sensors and subjectively asking people for their impressions) in a public park, what about linking this information with the current weather forecast or the wind speed or even the humidity published by the nearest weather station?

The CITI-SENSE ontology will be used to annotate the data gathered by sensors, calculate data (over sensor data) and answer to questionnaires, with the aim to publish this annotated dataset under the Linked Data paradigm as open data. In this way linking it to other linked datasets in the Linked Open Data Cloud[19] will be achieved.

During this deliverable a study of ontologies in the domain of data collected by sensors is detailed, as well as the INSPIRE Directive that indicates how to share environmental data. In CITI-SENSE, the INSPIRE Directive has been partially formalised as an ontology because no existing available ontologies – formalising INSPIRE Directive are available.

[19] http://linkeddata.org/

3. Requirements from CITI-SENSE

Within the project work packages 2 and 3 are in charge of developing the empowerment initiatives. So it is up to them to decide where the sensors will be placed and which type of observations will be captured. The Citizens' Observatories ontology must be capable to represent the information gathered by such initiatives in order to be publicly accessible in a standardized format.

The empowerment initiatives are divided, as indicated above, in two work packages dedicated to develop such initiatives in different scenarios:

- Work package 2 is devoted to develop and test methods for citizen's empowerment in the field of urban air quality. The city case studies will be deployed in 8 cities.
- Work package 3 has as aim to support citizen's participation in the management of public places in order to help ensuring a good environmental quality. This objective is concretised in two different pilots: a public park and several schools.

The aggregated requirements as defined the work packages 2 and 3 can be found in **Table 1**. In the following subsections of this chapter the different requirements are explained and classified under each work package and pilot that produced them, where each requirement is identified by the symbol Rx with the ID of the corresponding requirement inside.

In **Table 1**, some requirements have the label "UNDER DEVELOPMENT" attached, meaning that the exact requirement has not been fully defined. In general, due to the fact that pilots at the moment of writing this document are still under design, this causes the requirements to be still open. It is considered that at this moment of the project the pilots have enough maturity to outline an ontology that fulfils at least the 80% of the desired functionality and forces the ontology design to be open for the inclusion of future modifications.

Table 1. Requirements summary from CITI-SENSE pilots

ID	Requirement	Source
R1	Collect data from static outdoor sensors: NO, NO_2, CO, O_3,PM, Noise, Temperature, Humidity, PAH	WP2
R2	Collect data from personal sensors (indoor/outdoor): NO, CO, O_3, Temperature, Humidity	WP2
R3	Collect data from SmartPhone Data: GPS, Accelerometer, elevation	WP2
R4	Collect data from perception data: surveys UNDER DEVELOPMENT	WP2
R5	Collect data from air pollution monitoring and meteorological stations	WP2
R6	Collect Forecasting data on air pollution and meteorological data	WP2
R7	Collect data from User profile UNDER DEVELOPMENT	WP2
R8	Collect 360-degree photoscape (video or photo)	WP2
R9	Collect data from sensors (measures each minute): mean radiant temperature (Tmrt), wind speed, air temperature and relative humidity.	WP3/Public park
R10	Collect data from calculated value (for each measuring period): Heat index, Wind chill, Outdoor Wet Bulb Globe Temperature (WBGT), PET (Physiological Environmental Temperature)	WP3/Public park
R11	Collect data from sound measurements (MP3) for each measuring period	WP3/Public park
R12	Collect data from calculated value (each minute): LAeq, L90 LAmax and LAmin	WP3/Public park

ID	Requirement	Source
R13	Collect data from sound events: in MP3 recorded in R11, events: moment in the recording along with a label identifying the event.	WP3 – Public park
R14	Collect data from urban Landscape perception (answer to questionnaires about Local landscape perception, Participation and perception measurements and other data and photographs). UNDER DEVELOPMENT	WP3 – Public park
R15	Collect data from UV Exposure UNDER DEVELOPMENT	WP3 – Public park
R16	Collect data from sensor data: Temperature, Relative humidity, CO_2, NO_2, Dust, Noise, VOC, Radon.	WP3 – Schools Pilot
R17	Collect Location, School, operation hours, Time period	WP3 – Schools Pilot
R18	Compliant with INSPIRE directive	WP7
R19	Compliant with GEOSS standards (SKOS; SPARQL)	WP7
R20	Compliant with SenML standard	WP7

3.1 Work package 2 requirements

There are various types of information inputs coming from the 8 city case studies of work package 2, the following information will be collected and available during the case studies execution:

R1
- Static outdoor sensors. In each city 40 static sensors will be place to record information on a minute to minute basis for NO, NO_2, CO, O_3, PM, noise, temperature and humidity (- Ostrava and Belgrade will also record PAH).

R2
- Personal (indoor/outdoor) sensors. In each city (except Ostrava) 20 personal sensors will be distribute that measure NO_2, CO, O_3, temperature and humidity. We will likely recruit different groups of citizens (e.g. bicycle couriers / taxis, asthmatics …) and ask them to carry the sensors with them.

R3
- Smartphone data, in each city (except Ostrava) a limited number of smartphones will be distributed along with an. The app obtains positioning information from the in-built smartphone GPS (Satellite and network) and information from the in-built smartphone motion sensors (accelerometer) for physical activity (and perhaps elevation). The gathered data will have to be sent to a server on a minute-to-minute basis to obtain real-time positioning and physical activity data.

R4
- Perception data collected during short surveys via smartphone (app, SMS or email) will be gathered to obtain instantaneous information on people's perception of their urban environment.

R5
- Data from routine air pollution monitoring stations and meteorological stations that are present in the cities have to be extracted to a server for collation. Some cities have already websites that report this information on a 15 minute basis and it may be possible to obtain/link to this data to the server (see below). Otherwise, where this does not exist, we will have to decide on the time resolution the data will be available with. It may be beneficial to employ the EU Airbase with its homogenous data. Some example apps for air quality stations are available[20]. An app providing

[20] http://www.airqualitynow.mobi/
http://www.eea.europa.eu/mobile
Also near-real time data http://www.eea.europa.eu/themes/air/air-quality/map/real-time-map

visualisation of AQ data will be launched at EU level offering near-real time AQI information for specific locations.

R6

- Forecasting of air pollution and meteorological data for the next day, or following days, where available. Many cities now provide forecasts and it would be nice to obtain this data (see below for current location-specific examples). Producing forecasts of, e.g. air pollution and meteorological parameters, is an aspiration of WP6. WP6's first deliverable (D6.1) contains a list of services available (e.g. PASODOBLE MACC-II).

R7

- Consent for use of data; need for filtering home address geolocation (e.g. 50 m area coordinates / grid-system collated into one larger indicator). Health parameters expected to be available only to participant/patient and from investigator/physician. This may mean that in mandatory an acceptation process when using the mobile phone app and the personal sensors which would be related to a user profile.

R8

- Participants may be prompted to capture a 360-degree photoscape (as video or photo) of their environment at random/intermittent time-points or when pollution level thresholds have been breached by sensors / LUR map and location. Such a measure would suggest emission sources and microenvironments of influence; however, the video/image capturing would require extra memory and battery capacity so may not be practical for citizens.

3.2 Work package 3 - Public Park

WP3 will design, implement and evaluate a pilot case in a public park in Vitoria (Spain). The main aim of this pilot is to empower citizens in the process of designing public places from an environmental point of view including comfort criteria. The specific objectives are:

- to allow citizens or local communities to collect and share quantitative and qualitative information related to the environment of existing public places as well as their well-being in those places;
- to allow the city authorities to collect novel information about the ecosystem services provided in public places, and in this way
- supporting a dialogue between citizens/local communities and the authorities to adapt their planning process to improve or preserve the environmental conditions in sensitive public spaces.

The following aspects will be evaluated:

- Thermal comfort
- Acoustic comfort
- UV Radiation
- Urban landscape perception
- General satisfaction
- Other variables (safety, cleanness)

The scope of this pilot case is to empower the participants to evaluate during 2-3 days certain areas of the city using mobile sensors. Each measurement in each of the four areas will take at least 15 minutes. In the case of UV radiation continuous measurements are expected. The following figure shows an overview of the process:

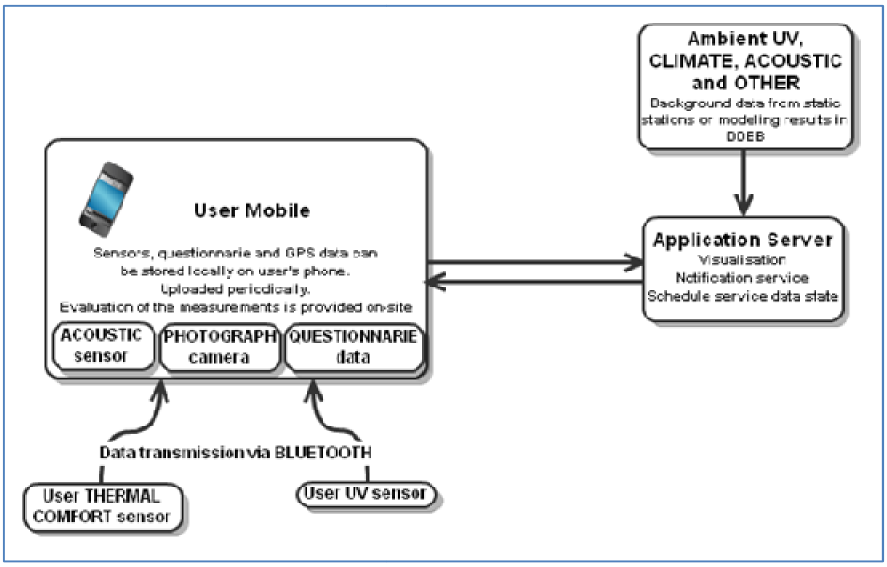

Figure 1. WP3 – Vitoria Pilot: Data acquisition for outdoor public spaces empowerment initiative (thermal comfort, acoustic and sound, UV radiation and urban landscape perception)

Mobile sensors carried by the participants (citizens) will provide information that will be collected with a smartphone (Figure 1). As soon as the smartphone receives data from the mobile sensors, an interface will prompt the user to rate the comfort (with the help of questionnaires), and take a photo with the in-build camera to evaluate the urban landscape perception.

In this empowerment initiative **three potential users** have been defined: participants/citizens, environmental public servants and decision-makers/politicians.

Measurements

R9

Thermal comfort

Information of different climatic variables (mean radiant temperature (Tmrt)), wind speed, air temperature and relative humidity) is required. Due to complexity of the measurement, Tmrt will be previously estimated / modelled as a function of cloudiness and solar radiation exposure and upload to an accessible database during the measurement campaigns. The rest of the variables, will be measured 'on-site' during the measuring period by the participants/citizens.

Initially, it is planned to use the commercial sensor **Kestrel® 4000 Pocket Weather Meter**.

Table 2. Specifications of Kestrel 4000 Weather Meter

SENSOR	ACCURACY (+/-)*	RESOLUTION	SPECIFICATION RANGE	OPERATIONAL RANGE
Wind Speed | Air Flow	Larger of 3% of reading, least significant digit or 20 ft/min	0.1 m/s 1 ft/min 0.1 km/h 0.1 mph 0.1 knots 1 B	0.6 to 40.0 m/s 118 to 7,874 ft/min 2.2 to 144.0 km/h 1.3 to 89.5 mph 1.2 to 77.8 knots 0 to 12 B	0.6 to 60.0 m/s 118 to 11,811 ft/min 2.2 to 144.0 km/h 1.3 to 134.2 mph 1.2 to 116.6 knots 0 to 12 B
Ambient Temperature	0.9 °F 0.5 °C	0.1 °F 0.1 °C	-20.0 to 158.0 °F -29.0 to 70.0 °C	14.0.0 to 131.0 °F -10.0 to 55.0 °C
Relative Humidity	3.0 %RH	0.1 %RH	5 to 95% non-condensing	0 to 100%
Pressure	0.03 inHg 1.0 hPa|mbar 0.01 PSI	0.01 inHg 0.1 hPa|mbar 0.01 PSI	8.86 to 32.49 inHg 300.0 to 1100.0 hPa|mbar 4.35 to 15.95 PSI and 32.0 to 185.0 °F 0.0 to 85.0 °C	0.30 to 48.87 inHg 10.0 to 1654.7 hPa|mbar 0.14 to 24.00 PSI and 14.0 to 131.0 °F -10.0 to 55.0 °C

The collection and analysis of thermal comfort data includes the following steps:

1. Temperature, relative humidity and wind speed will be stored in the internal memory of the smartphone (at least 15 minutes of data). During the measurements 1 minute mean values will be calculated and, at the end, mean values for the measurement period will be calculated and stored. Periodically (e.g. every minute), the recorded and calculated information will be sent to the platform.

R10

2. The following thermal comfort indexes will be calculated and stored: heat index, wind chill, Outdoor Wet Bulb Globe Temperature (WBGT) and physiological environmental temperature (PET) for the total measuring period. If this aspect (i.e. the post-processing) is done with the smartphone, the latter should have enough capacity to apply pre-defined algorithms on the stored data to obtain other results. Otherwise, the post-processing can be done on the platform and the results (thermal indexes) can be sent back to the smartphone.

Acoustic comfort

There are three aspects that affect the precision of the results and that are identified as challenges considering the review of some applications for smartphones that can analyse acoustic data:

- The precision of the microphone.
- The average of the measured data must be developed considering the logarithmic scale of dB.
- The smartphone capacity for collecting data (sampling time of the acoustic signal).

The **microphone of smartphone NEXUS** ***will be evaluated*** to see if the reliability of the measuring technique is sufficient for the aim of the project.

Collection and analysis of acoustic data includes the following steps:

R11
1. Sound measurements will be stored in the internal memory of the smartphone (at least 15 minutes of data, in mp3 format).

R12
2. The following acoustic indicators will be calculated and stored: LAeq and L90 with a 1 sec. sample as well as LAeq, L90 LAmax and LAmin for the total measuring period. Periodically (e.g. every minute), the recorded and calculated information will be sent to the platform.
3. The acoustic indexes will be combined to obtain information of sound events and a final acoustic comfort index. If this post-processing is done by the smartphone, the latter should have enough capacity to apply pre-defined algorithms on the stored data to obtain other results. Otherwise, the post-processing can be done on the platform and the results (sound events and indexes) can be sent back to the smartphone.

R13
4. When an acoustic event is detected by the smartphone an alert must be displayed so that the citizen can provide the required information regarding the source of the sound that generated the event.

R14
Urban landscape perception

In this case, the information of specific sites/areas inside the public space is required. It is expected that the participant will use the camera of the smartphone to take several pictures and following s/he will have to evaluate what the pictures represent:

- **UV exposure:** It is still under development.
- **Local landscape perception:** The assessment of the local landscape perception will be supported by photographs. Participants will be able to take two or three pictures that show what they are seeing at the measuring site. On the photographs, they must identify the elements and evaluate them on a scale of level of pleasantness. In this way, they will provide their urban landscape perception. This can be done either by selecting points on the picture/screen or drawing a polygon. In the first case, the number of points and their assessment will be stored and analysed. In the second case, percentages of picture area and its assessment will be stored and analysed. This aspect is still open.

  From each identified element the following information will be collected:

  - Type: a dropdown scale with possible response options will be supplied. Some of these could be:
    - green space,
    - fountains or other water feature,
    - cultural heritage,
    - buildings,
    - roads,
    - street furniture, etc.

    - "other" options will be possible for a new category.
- Pleasantness / unpleasantness: select on a 5 level scale (from little to very much).

Recorded data (i.e. photographs and the information associated to them) will be sent to the platform and stored for off-line post-processing mainly. The information sent will be periodically, i.e. every time a photograph is completely analysed.

- **Participation and perception measurements:** Citizens/Participants will provide their perception through questionnaires during the 15-20 minute period of each measurement in each site.

The questionnaire context is based on seven dimensions:

1. **Personal factors**: Socio-demographic variables (sex, age, education level…), residential factors (neighbourhood and age of residence…), health and life style factors (perceived health, emotions…), and psychosocial factors (sensibility to noise, heat and UV radiation, and global stress or annoyance to noise and heat).
2. **Places and comfort**: used spaces in the city or town
    a. General places (residential, work or study, social relationship, and entertainment and leisure): Acoustical and thermal comfort and global satisfaction
    b. Urban public spaces: place identification, frequency, characteristics and motivation for use, distance to general places, the most and least liked, and global place evaluation.
3. **Global experience and perception** of the selected place: frequency and characteristics of use (activity in the place…), place perception (Semantic Differential (SD): pleasantness, security, maintenance, land and soundspace, thermal…), the most and least liked, acoustic and thermal comfort and emotions.
4. **Sound Environment Perception or Soundscape**: perceived sounds and pleasant and congruence related, evaluation of soundscape (SD).
5. **Thermal comfort**: perception of temperature, humidity and wind, global heat stress, thermal preferences, clothing insulation and sweating.
6. **UV comfort, effects**: this part is still under development.
7. **Landscape perception:** selection of elements (green area, cultural heritage, road, buildings…) on a photograph and evaluate their pleasantness.
8. Beside the data collected in the questionnaire, other interesting information will be gathered by the smartphone system (e.g., time of start and end of the experience…)

The questionnaires will be integrated in a **unique smartphone application** that will handle the interaction between the participants/citizens and the capture, processing and storage of data. This application will be developed in the context of the empowerment initiative.

- **Other data**

Additionally, **urban maps** with different information are required. These should include urban morphology to define characteristics of the public space (building heights, areas), green areas, historical heritage, education issues, health issues etc. Also traffic intensity and other noise sources will be considered previously and after the empowerment measuring experience. This data will be provided by the municipality.

Regarding the social aspect, it would also be interesting to have information related with population and place: characteristics of the population that lives in nearby areas, municipal action plans affecting the selected area.

The usage of this already available data will complement data from the mobile sensors and questionnaires collected during the measuring period. Thus, definition of adequate databases previously to the measuring period will:

- provide extra information to the citizen/participant in the empowerment initiative and/or contribute to the 'on-line' perception evaluation.
- contribute to an 'off-line' evaluation of citizens thermal comfort, acoustics and UV radiation exposure after the measuring/experience period.

3.3 Work package 3 – Schools Pilot

R15

Visualization of real time data for these parameters:

- Temperature
- Relative humidity
- CO2
- NO2
- Dust
- Noise
- VOC
- Radon

R16

General queries

- Get sensors on school = return XML, JSON.
- Get locations on school = return XML, JSON.
- Get parameter types measured on school = return XML, JSON.
- Get measured data in this time period where parameter x`s value is greater than y = return XML, JSON.
- Get sensor data, Query with all possible combinations, return types= 1) plots 2)export to excel friendly format
    - Location
    - Parameter type/types
    - School
    - [operation hours J/N]

- Time period
- Flag

The main stakeholders in this pilot are the Pupils, Teachers, Technical staff and Headmaster.

The main purpose of the applications is to give the stakeholders tools/information that will enable them to work more actively and systematically with issues that will improve **indoor air quality**. And also, to raise awareness of the importance of indoor air environment on health, wellbeing and learning.

A CITI-SENSE school pilot portal will be developed as part of the project. Here the main structure and functionalities for the first pilot study are outlined.

The school should be able **to move the sensors from** one room to another during the pilot. The data must have a tag indicating the location of the sensors at all times.

The users will mainly be interested in exposures when the school children are present. This means that we need to be able to define operating hours individually for each school according to local needs.

3.4 Work package 7 –Architecture

The Deliverable D7.1 "CITI-SENSE Architecture" provides the foundation for the CITI-SENSE architecture. Figure 2Figure 3 shows the CITI-SENSE architecture for sensor and data management.

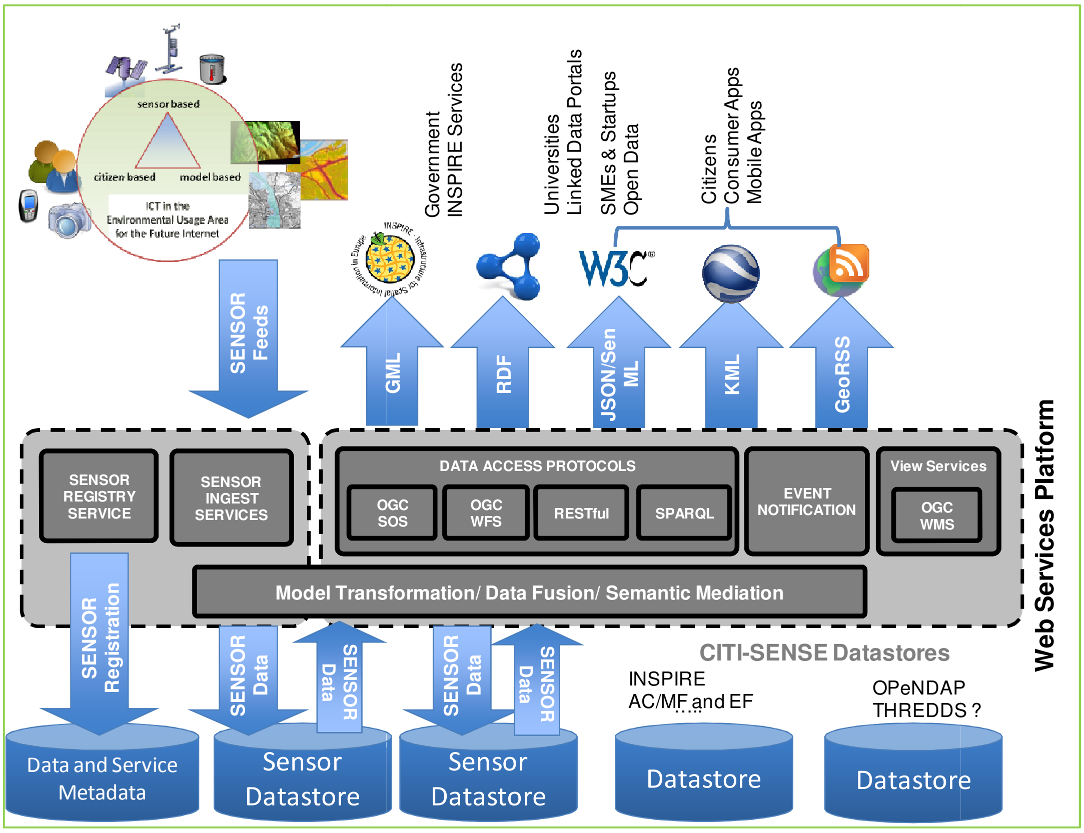

Figure 2. Architecture for sensor and data management

OGC web services will provide access to CITI-SENSE data for clients. This suite of standards is also used by GEOSS and INSPIRE and so by providing access via OGC services CITI-SENSE data will be easy to integrate into national, European and international data sharing frameworks.

R18

The platform architecture of CITI-SENSE is to be developed using the latest Open Standards for data encoding, exchange and services. The INSPIRE Directive will be followed for data observations encoding.

The CITI-SENSE project will contribute to the GEOSS Common Infrastructure (GCI) [21], but also the CITI-SENSE architecture will be aligned with the GEOSS Architecture. The CITI-SENSE project has been accepted as a contributor to the GEOSS AIP-6[22] effort, which takes place from March 2013 until March 2014. The aim is to present a Showcase pilot from CITI-SENSE following the principles of GEOSS.

The GEOSS strategy is to realize a system of systems through adoption of selected international standards that enable interoperability[23]. The mechanism that facilitates the interoperability is the GEOSS architecture, which is realized by the components implemented as part of the GEOSS Common Infrastructure (GCI). The GEOSS Standards and interoperability Registry[24] enables contributors to GEOSS to configure their systems so that they can share information with other systems. This Registry is vital to the ability of GEOSS to function as a true system of systems and to provide integrated and cross-cutting information and services. The standards can be found at the GEOSS Standard Registry[25]. Some of them are SKOS, SPARQL

R19

The sensor ingest component is responsible for recieving data from sensors and moving into the CITI-SENSE data store. This includes a number of functions.

The Sensor Ingest component of the CITI-SENSE architecture (see Figure 3) provides web services interfaces to allow sensors to upload data. It periodically polls sensors registered with it for updates if those sensors do not actively upload data themselves. It validates the incoming data and rejects invalid input.

R20

It translates the incoming **SenML** into the internal storage model of the CITI-SENSE platform and updates the store. The sensor ingest component also provides the sensor registration services since these are integral to the ingestion of sensor data. SenML is the specification for data encoding in the CITI-SENSE sensor API.

[21] http://earthobservations.org/gci_gci.shtml
[22] http://www.earthobservations.org/geoss_call_aip.shtml
[23] http://seabass.ieee.org/groups/geoss/
[24] http://www.earthobservations.org/gci_sr.shtml
[25] http://seabass.ieee.org/groups/geoss/index.php?option=com_sir_200&Itemid=157

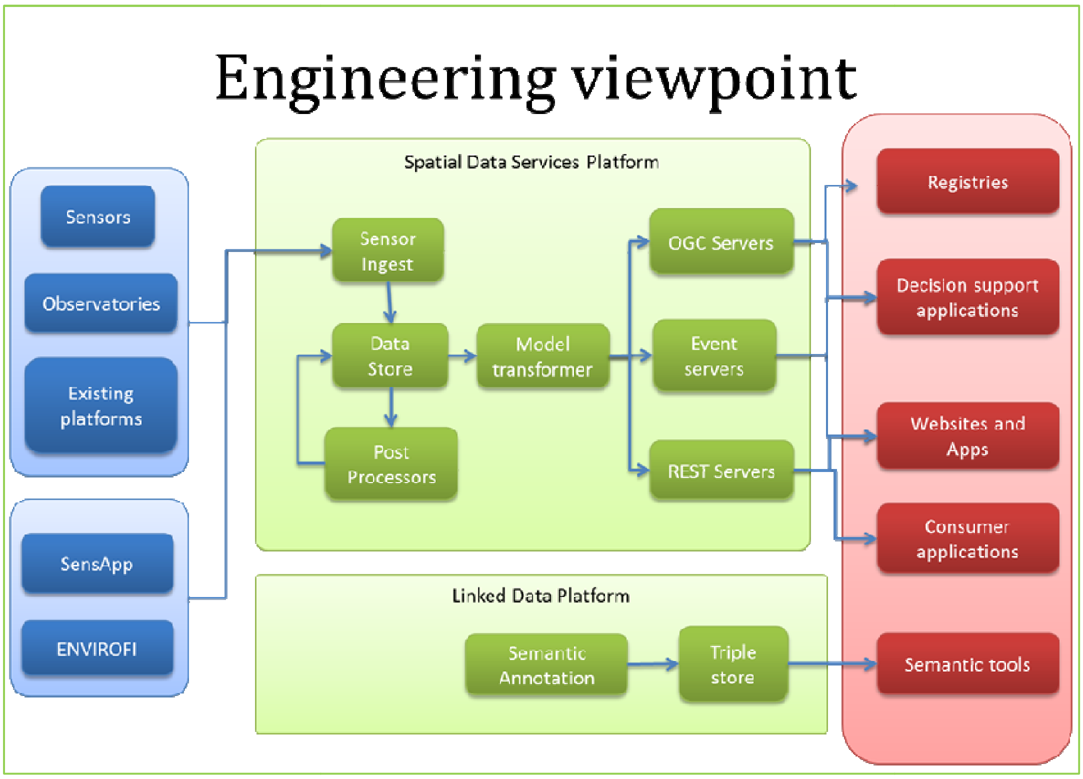

Figure 3. Engineering viewpoint of the CITI-SENSE architecture

4. Analysis of ontological and non-ontological sources

Sheth et al. in [6] defined semantics of sensor web within space, time, and theme scopes. There have been different approaches to provide semantic models for each of these attributes independently or in relation to sensors used. Some of the common ontologies are the SIMILE location ontology, the DAML location ontology for spatial attributes, OWL time ontology for time and common ontologies and vocabularies such as CyC, DBpedia for thematic data. A list of the most popular ontologies in the domain of sensors observations as well as references can be found in annex 2.

In this chapter a shortlist of ontologies and other non-ontological resources is discussed. These resources have been selected according to their relevance in the current state of the art and according to their suitability to the knowledge domain the city observatories must cover. This knowledge domain is covered by different aspects that match with the subsections of this chapter including in particular:

- Standards, resources and directives in sensors, observations and measurements as well as web services,
- Location and space,
- Time modelling,
- Units of measure, and
- Ontologies in the sensor domain.

Chapter 5 is devoted to the description of the different decisions that have to be made to create the city observatory ontology. These decisions reflect the selection process as well as the reasons leading to them. An overview of the different resources is described in the current chapter.

4.1 Standards and Resources

This subsection describes the analysis of knowledge domain related standards and resources. This analysis takes as input the outcomes of work in WP7 regarding the identification of the different standards and resources of application in CITI-SENSE in the different aspects of the knowledge domain of the whole CITI-SENSE project:

- Data Observations
- Sensor Data
- Metadata (Data / Services)
- Registries
- Notification/Event/Alert Service
- Visualization web services
- Access/Download Web Service
- Web Processing services
- Semantic Mediation

Table 3 shows the results of the identification process of the different standards and resources used in CITI-SENSE applications. The following subsections provide more detailed explanations of the most relevant standards and resources for describing sensor data that is applicable to the city observatories ontology:

Table 3. Standards and resources in CITI-SENSE

Scope	**OGC/ISO TC211**	**INSPIRE[26]*/ AQD[27]**	**W3C/OASIS**	**Other**
Data - Observations	ISO 19156:2012 Observations and Measurements[28] Observations and Measurements - XML Implementation[29] SWE Common 2.0[30]	**Source Observations:** INSPIRE Specialised Observations: Technical Guidance[11] Schema[31] INSPIRE Atmospheric and Meteorological Geographical Features Technical Guidance[12] Schema[32] **Aggregated Observations/Indicators:** INSPIRE Human Health and Safety Technical Guidance[33] Schema[34] INSPIRE Statistical Units Technical Guidance[35] Schema[36] **Source Observations/Modelled Observations/Aggregated Observations/Indicators** AQD e-Reporting Data Specification[37]	senML Semantic Sensor Network ontology[41]	

[26] INSPIRE builds upon the ISOTC211/OGC standards and AQD Data Specifications and requirements for network services build upon INSPIRE. http://inspire.jrc.ec.europa.eu/
[27] http://www.eionet.europa.eu/aqportal/datamodel
[28] http://portal.opengeospatial.org/files/?artifact_id=41579
[29] http://portal.opengeospatial.org/files/?artifact_id=41510
[30] http://portal.opengeospatial.org/files/?artifact_id=41157
[31] http://inspire.jrc.ec.europa.eu/draft-schemas/omso/2.9/SpecialisedObservations.xsd
[32] http://inspire.jrc.ec.europa.eu/draft-schemas/omso/2.9/SpecialisedObservations.xsd
[33] http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_HH_v3.0rc3.pdf
[34] http://inspire.jrc.ec.europa.eu/draft-schemas/hh/3.0rc3/HumanHealth.xsd
[35] http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_SU_v3.0rc3.pdf
[36] http://inspire.jrc.ec.europa.eu/draft-schemas/su-core/3.0rc3/StatisticalUnitCore.xsd
[37] http://www.eea.europa.eu/publications/reporting-and-exchanging-air-quality/at_download/file

Scope	OGC/ISO TC211	INSPIRE[26]*/ AQD[27]	W3C/OASIS	Other
		Guidelines for aggregation rules for e-Reporting[38] Data Model: Presentation Excel Mapping Sheet[39] Schema[40]		
Data - Sensors: Procedures Monitoring Facility	Procedures: SensorML[42] SWE Common 2.0[43] Monitoring Facilities: O&M Sampling Feature	Procedures INSPIRE Processes Technical Guidance[11] Schema[44] AQD Measurement Monitoring Facilities *Environmental Monitoring Facilities* Technical Guidance[45]Schema[46] *AQD Monitoring Stations (see links above)*	Semantic Sensor Network[47]	
Metadata (Data/Services)	ISO 19115/ISO 19139 - Data ISO 19119 – Services	INSPIRE Metadata[48] Implementing Rule Technical Guidelines[14]		
Registers/Registries	ISO 19135 - Procedures for Registration OGC Catalogue Service[49]: OWL Application Profile ebRIM Profile ISO Metadata	INSPIRE Discovery Services Implementing Rule[50] Technical Guidelines[51] INSPIRE Spatial Data Services (Invoke Services)[15]: Registries Gazeteeers	Linked Data Platform 1.0[52] Linked Data Registry[53]	

41 http://www.w3.org/2005/Incubator/ssn/XGR-ssn/
38 http://www.eionet.europa.eu/aqportal/guidelines/ETC_Aggregation_v0_6.7_latest.pdf
39 http://www.eionet.europa.eu/aqportal/datamodel/20121220_IPR_Mapping_v1.xlsx
40 http://dd.eionet.europa.eu/schemas/id2011850eu_v03/AirQualityReporting.xsd
42 http://www.opengeospatial.org/standards/sensorml
43 http://portal.opengeospatial.org/files/?artifact_id=41157
44 http://inspire.jrc.ec.europa.eu/draft-schemas/ompr/2.9/Processes.xsd
45 http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_EF_v3.0rc3.pdf
46 http://inspire.jrc.ec.europa.eu/draft-schemas/ef/3.0rc3/EnvironmentalMonitoringFacilities.xsd
47 http://www.w3.org/2005/Incubator/ssn/XGR-ssn/
48 http://inspire.jrc.ec.europa.eu/index.cfm/pageid/101
49 http://www.opengeospatial.org/standards/cat
50 http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32009R0976:EN:NOT
51 http://inspire.jrc.ec.europa.eu/documents/Network_Services/TechnicalGuidance_DiscoveryServices_v3.1.pdf
52 http://www.w3.org/TR/2012/WD-ldp-20121025/
53 https://github.com/der/ukl-registry-poc/wiki

Scope	**OGC/ISO TC211**	**INSPIRE[26]*/ AQD[27]**	**W3C/OASIS**	**Other**
	Profile			
Notification/Event/Alert Services	Open GeoSMS Sensor Event Service		Common Alerting Protocol (CAP)[54] WS Notification[55]	RSS GeoRSS ATOM
Visualization Web Services	ISO 19128 Web Map Service (WMS 1.3.0)[56] KML[57]	INSPIRE View Services Implementing Rule[58] Technical Guidelines[59]		
Access/Download Web Service	ISO 19142 Web Feature Service (WFS 2.0)[60] ISO 19143 Filter Encoding Specification (FES 2.0)[61] Sensor Observation Service (SOS 2.0) GeoSPARQL[62]	INSPIRE Download Services Implementing Rule[63] Technical Guidelines[64]	SPARQL	
Processing Web Services	Web Processing Service (WPS 1.0)[65]			
Semantic Mediation			OWL, RDF, SPARQL	

[54] http://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.html
[55] https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsn
[56] http://portal.opengeospatial.org/files/?artifact_id=14416
[57] http://www.opengeospatial.org/standards/kml
[58] http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32009R0976:EN:NOT
[59] http://inspire.jrc.ec.europa.eu/documents/Network_Services/TechnicalGuidance_ViewServices_v3.1.pdf
[60] http://portal.opengeospatial.org/files/?artifact_id=39967
[61] http://portal.opengeospatial.org/files/?artifact_id=39968
[62] http://portal.opengeospatial.org/files/?artifact_id=47599
[63] http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:02009R0976-20101228:EN:NOT
[64] http://inspire.jrc.ec.europa.eu/documents/Network_Services/Technical_Guidance_Download_Services_3.0.pdf
[65] http://portal.opengeospatial.org/files/?artifact_id=24151

4.1.1 Data Sensors – SensorML or Sensor Model Language

The OpenGIS® Sensor Model Language Encoding Standard (SensorML) specifies models and XML encodings that provide a framework for defining geometric, dynamic, and observational characteristics of sensors and sensor systems[66].

There are many different sensor types, from simple visual thermometers to complex electron microscopes and earth observing satellites. These can all be supported through the definition of atomic process models and process chains. Within SensorML, all processes and components are encoded as application schema of the feature model in the Geographic Markup Language (GML) Version 3.1.1. This is one of the OGC Sensor Web Enablement (SWE) (OGC_SWE) suite of standards[67]

The primary focus of SensorML is to define processes and processing components associated with the measurement and post-measurement transformation of observations.

The purposes of SensorML are to:

- Provide descriptions of sensors and sensor systems for inventory management.
- Provide sensor and process information in support of resource and observation discovery.
- Support the processing and analysis of the sensor observations.
- Support the geolocation of observed values (measured data).
- Provide performance characteristics (e.g., accuracy, threshold, etc.).
- Provide an explicit description of the process by which an observation was obtained (i.e., it's lineage).
- Provide an executable process chain for deriving new data products on demand (i.e., derivable observation).
- Archive fundamental properties and assumptions regarding sensor systems.

SensorML[68] provides a common framework for any process and process chain, but is particularly well-suited for the description of sensor and systems and the processing of sensor observations. Within SensorML, sensors and transducer components (i.e. detectors, transmitters, actuators, and filters) are all modelled as processes that can be connected and participate equally within a process chain or system, and which utilize the same process model frame as any other process.

SensorML can, *but generally does not*, provide a detailed description of the hardware design of a sensor. Rather it is a general schema for describing **functional models** of the sensor.

SensorML enables robust definitions of sensor models for providing **geolocation of observations from remote sensors.**

[66] http://www.opengeospatial.org/standards/sensorml
[67] http://schemas.opengis.net/sensorML/
[68] http://portal.opengeospatial.org/files/?artifact_id=21273

4.1.2 Data Sensors - OGC® SWE Common Data Model Encoding Standard

This standard [4] defines low level data models for exchanging sensor related data between nodes of the OGC® Sensor Web Enablement (SWE) framework. The generic SWE Common data model defined by this standard aims at providing verbose information to **robustly describe sensor related datasets.**

Considering **Sensor Data** as data resulting from the observation of properties of virtual or real world objects (or features) by any type of Observation Procedure. Sensor related datasets however are not limited to sensor observation values, but can also include auxiliary information such as status or ancillary data.

These models allow applications and/or servers to structure, encode and transmit sensor datasets in a self-describing and semantically enabled way. More precisely, the SWE Common Data Model is used to define the representation, nature, structure and encoding of sensor related data.

The SWE Common Data Model is intended to be used for describing static data (files) as well as dynamically generated datasets (on the fly processing), data subsets, process and web service inputs and outputs and real-time streaming data. All categories of sensor observations are in scope ranging from simple in-situ temperature data to satellite imagery and full motion video streamed out of an aircraft.

The Observations and Measurements Standard (O&M) also references the SWE Common data model, although the observation model defined in the O&M specification is decoupled from this standard. One goal of the SWE Common Data Model is thus to maintain the functionality required by all these related standards.

The SWE Common Data Model covers the following aspects of datasets description: Representation, Nature of data and semantics (by using identifiers pointing to external semantics), Quality, Structure and Encoding.

The SWE Common language is an XML implementation of this model and is used by other existing OGC® Sensor Web Enablement standards such as Sensor Model Language (SensorML), Sensor Observation Service (SOS), Sensor Alert Service (SAS) and Sensor Planning Service (SPS).

4.1.3 Data Sensors - INSPIRE data specification on Environmental Monitoring Facilities

The INSPIRE Directive (2007/2/EC) obliges national authorities of the EU-member states to contribute their spatial data according to over 30 harmonized themes (e.g. Hydrography, Protected Sites or Elevation), make them accessible and described via standardized Geo Web-Services.

Location and operation of environmental monitoring facilities[69] includes observation and measurement of emissions, of the state of environmental media and of other ecosystem parameters (biodiversity, ecological conditions of vegetation, etc.) by or on behalf of public authorities. The scope as defined in the INSPIRE directive includes two aspects the environmental monitoring facility as a spatial object in the context of INSPIRE and observations and measurements linked to the environmental monitoring facility.

[69] http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_EF_v3.0rc3.pdf

The application schema for Environmental Monitoring Facilities contains 4 spatial object types:

- Environmental Monitoring Programme
- Environmental Monitoring Activity
- Environmental Monitoring Network
- Environmental Monitoring Facility

In addition to the use of the Observations and Measurements standard, further elements of the OGC Sensor Web Enablement Suite (SWE) have been identified as useful for the encoding and provision of observation data. While further SWE specifications may be nominated for use in INSPIRE, some of them have been identified:

- Sensor Observation Service (SOS): service created for the provision of observational data;
- SensorML: Standard for the provision of procedural information;
- SWE Common: Includes result encoding options.

Please see http://www.opengeospatial.org/ogc/markets-technologies/swe for a wider view on this topic including sensor tasking, filtering, notifications from sensor measurements, etc.

Many types of spatial data can be structured using either Observation and Measurement. The goal is a feasible embedding of INSPIRE data in the Semantic Web. INSPIRE itself is based on the Geo Web technologies - which are ISO and OGC standardizations for Geo web-services. So INSPIRE Directive applies the ISO/OGC approach of modelling physical things in the GML.

INSPIRE Consolidated UML Model

INSPIRE has chosen the conceptual modelling language UML for the formal representation of abstraction of the reality. This international standard defines rules to convert UML static views of geographic information and application schemas into OWL ontologies in order to benefit from and support interoperability of geographic information over the Semantic Web.

4.1.4 DataSensors – ISO 19150

The ISO Technical Committee has defined a high level framework that structures the standards addressing specifically the semantics of geographic information through ontologies. The proposed other parts of the framework include:

- ISO 19150-2, Geographic information — Ontology — Part 2: Rules for developing ontologies in OWL, defines rules and guidelines for the development of ontologies in OWL-DL, including a mapping between UML class diagram elements and OWL-DL and rules for describing application schemas in OWL-DL;
- ISO 19150-3, Geographic information — Ontology — Part 3: Semantic operators, defines semantic proximity operators between concepts that complement geometric and temporal operators;
- ISO19150-4, Geographic information — Ontology — Part 4: Service ontology, identifies the framework for service ontology and defines the description of Web services for geographic information in an ontology language;
- ISO 19150-5, Geographic information — Ontology — Part 5: Domain ontology registry, defines an international registry of geographic information domain ontologies and its maintenance;

- ISO 19150-6, Geographic information — Ontology — Part 6: Service ontology registry, defines an international registry of geographic information service ontologies and its maintenance.

ISO 19150 was prepared by Technical Committee ISO/TC 211, *Geographic information/Geomatics.*

4.1.5 The Geospatial Semantic Web

The purpose of the Geospatial Semantic Web[70] is to push the development of the Semantic Web in order to enhance the interoperability of geographic information on the Web. It also enables automated knowledge inference, which significantly increases the amount of geographic information. Additionally, the Geospatial Semantic Web may allow the interoperability of information across all data sources on the Web by relating geographic and non-geographic information and providing more interaction between the different data sources on the Web. Challenges identified relating to the Geospatial Semantic Web are:

- ontologies of spatial concepts used across disciplines;
- geospatial-relations ontology;
- geospatial feature ontology;
- place names ontology;
- ontology for metadata;
- ontology for coordinate reference systems;
- ontology management: designing, developing, storing, registering, discovering, browsing, maintaining and querying;
- canonical form for geospatial data queries;
- matching concepts to ontologies;
- ontology integration;
- ontological description and annotation of geo-enabled Web Services.

4.1.6 Data Sensors - OGC Sensor Observation System (SOS)

The SOS [27] standard is applicable to use cases in which sensor data needs to be managed in an interoperable way. This standard defines a Web service interface which allows querying observations, sensor metadata, as well as representations of observed features. Further, this standard defines means to register new sensors and to remove existing ones. Also, it defines operations to insert new sensor observations. This standard defines this functionality in a binding independent way; two bindings are specified in this document: a KVP binding and a SOAP binding.

The SOS is one of a family of OGC standards that make up the OGC SWE framework.

The Sensor Observation Service (SOS) provides a standardized interface for managing and retrieving metadata and observations from heterogeneous sensor systems. Sensor systems contribute the largest part of geospatial data used in geospatial systems today. Sensor systems include for example

[70] http://www.opengeospatial.org/projects/initiatives/gswie

in-situ sensors (e.g. river gauges), moving sensor platforms (e.g. satellites or unmanned aerial vehicles) or networks of static sensors (e.g. seismic arrays). Used in conjunction with other OGC specifications the SOS provides a broad range of interoperable capability for discovering, binding to and interrogating individual sensors, sensor platforms, or networked constellations of sensors in real-time, archived or simulated environments.

SOS 2.0 relies on the OGC Observations and Measurements (O&M)[71] standard to encode data gathered by sensors.

4.1.7 Notification/Event/Alert Services - AirBase Network Communication Protocol

The AirBase Sensor Network suite represents a complete solution for remote air quality monitoring. The suite consists of a hardware platform for sensing and data collecting, a software suite for data storage, alerts, signal processing and information generation, and the WEB portal for geo-referenced information, maps, historical retrieval, and report generation.

The nodes can be deployed using local Wi-Fi networks, or GSM networks, and need only power supply.

The system has 4 basic levels of interface:

- CAP - Common Alerting Protocol
- Web service – to get detailed air quality measurements.
- Dynamic KML.
- Web pages (system and maintenance).

The Common Alerting Protocol (CAP)

The Common Alerting Protocol (CAP)[72] is an XML-based data format for exchanging public warnings and emergencies between alerting technologies.

Example of CAP alert

```
<?xml version="1.0" encoding="UTF-8"?>
<alert xmlns="urn:oasis:names:tc:emergency:cap:1.1">
<identifier>0f8fad5b-d9cb-469f-a165-70867728950e</identifier>
<sender>myairbase.com</sender>
<sent>2012-9-19T14:57:00-07:00</sent>
<status>Actual</status>
<msgType>Alert</msgType>
<scope>Public</scope>
<info>
<category>Security</category>
<event>AirPollution</event>
<urgency>Expected</urgency>
<certainty>Likely</certainty>
<severity>Moderate</severity>
<description>Air quality level moved from good to moderate. moderate levels of VOC</description>
<area>
<circle>32.9525,-115.5527 0</circle>
</area>
<!--0 - not working, 1 - good, 2 - moderate, 3 - high-->
```

[71] http://www.opengis.net/doc/om/2.0
[72] http://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.html

```
<parameter>
<valueName>O3</valueName>
<value>1</value>
</parameter>
<parameter>
<valueName>VOC</valueName>
<value>2</value>
</parameter>
<parameter>
<valueName>NO2</valueName>
<value>1</value>
</parameter>
<parameter>
<valueName>Dust</valueName>
<value>1</value>
</parameter>
<parameter>
<valueName>Noise</valueName>
<value>1</value>
</parameter>
</info>
</alert>
```


4.1.8 Data Observations - SenML

SenML (Media Types for Sensor Markup Language) is a specification[73] promoted by the IETF organisation.

SenML is defined by a data model for measurements and simple meta-data about measurements and devices. The data is structured as a single object (with attributes) that contains an array of entries. Each entry is an object that has attributes such as a unique identifier for the sensor, the time the measurement was made, and the current value. Serializations for this data model are defined for JSON, XML and Efficient XML Interchange (EXI).

Each representation carries a single SenML object that represents a set of measurements and/or parameters. This object contains several optional attributes described below and a mandatory array of one or more entries:

1. Base Name: This is a string that is prepended to the names found in the entries. This attribute is optional.
2. Base Time: A base time that is added to the time found in an entry. This attribute is optional.
3. Base Units: A base unit that is assumed for all entries, unless otherwise indicated. This attribute is optional. Acceptable values are specified.
4. Version: Version number of media type format. This attribute is optional, positive integer and defaults to 1 if not present.
5. Measurement or Parameter Entries: Array of values for sensor measurements or other generic parameters (such as configuration parameters). If present there must be at least one entry in the array. Each array entry contains several attributes, some of which are optional and some of which are mandatory.

[73] http://tools.ietf.org/html/draft-jennings-senml-08

6. Name: Name of the sensor or parameter. When appended to the Base Name attribute, this must result in a globally unique identifier for the resource. The name is optional, if the Base Name is present. If the name is missing Base Name must uniquely identify the resource. This can be used to represent a large array of measurements from the same sensor without having to repeat its identifier on every measurement.
7. Units: Units for a measurement value. Optional, if Base Unit is present or if not required for a parameter. Acceptable values are specified.
8. Value: Value of the entry. Optional if a Sum value is present, otherwise required. Values are represented using three basic data types: Floating point numbers ("v" field for "Value"), Booleans ("bv" for "Boolean Value") and Strings ("sv" for "String Value"). Exactly one of these three fields MUST appear.
9. Sum: Integrated sum of the values over time. Optional. This attribute is in the units specified in the Unit value multiplied by seconds.
10. Time: Time when value was recorded. This attribute is optional.
11. Update Time: A time in seconds that represents the maximum time before this sensor will provide an updated reading for a measurement. This can be used to detect the failure of sensors or communications path from the sensor. Optional.

The SenML format can be extended with further custom attributes placed in the base object, or in an entry. Extensions in the base object pertain to all entries, whereas extensions in an entry object only pertain to that.

4.2 Resources for Location and Space

The location of each sensor is crucial to provide a context to the observation it performs. It is not the same to observe a temperature of 30ºC in a park than in a school room. Different resources are in place to help formalising location and space as CityGML, GeoNames, GeoSPARQL, Mobile ontology, World Geodetic System 1984, NGEO and Core Location Vocabulary. Each of them are devoted to represent an exact location with GPS coordinates or geometry based on polygons and points for instance, while others provide geographical names to locate an object / person.

4.2.1 CityGML: City Geography Markup Language

CityGML[74] is an open data model and XML-based format for the storage and exchange of virtual 3D city models. It is an application schema for the Geography Markup Language version 3.1.1 (GML3), the extendible international standard for spatial data exchange issued by the Open Geospatial Consortium (OGC) and the ISO TC211. The aim of the development of CityGML is to reach a common definition of the basic entities, attributes, and relations of a 3D city model. This is especially important with respect to the cost-effective sustainable maintenance of 3D city models, allowing one the reuse of the same data in different application fields.

4.2.2 GeoNames ontology

The GeoNames Ontology[75] makes it possible to add geospatial semantic information to the World Wide Web. Up to now more than 83 million geonames toponyms have a unique URL with a corresponding RDF web service. Other services describe the relation between toponyms.

[74] http://www.citygml.org/index.php?id=1523
[75] http://www.geonames.org/ontology

The GeoNames geographical database[76] is available for download free of charge under a creative commons attribution license. It contains over 10 million geographical names and consists of over 8 million unique features whereof 2.8 million populated places and 5.5 million alternate names. All features are categorized into one out of nine feature classes and further subcategorized into one out of 645 feature codes. The data is accessible free of charge through a number of web services and a daily database export. GeoNames is already serving up to over 30 million web service requests per day. GeoNames is integrating geographical data such as names of places in various languages, as well as information of elevation, population etc from various sources. All lat/long coordinates are in WGS84 (World Geodetic System 1984). Users may manually edit, correct and add new names using a user friendly wiki interface.

The Linked Data representation of the database available as an RDF dump contains 6,520,110 features and 93,896,732 triples. The structure behind the data is the GeoNames ontology[77], which closely resembles the flat-file structure. The GeoNames Ontology makes it possible to add geospatial semantic information to the World Wide Web. All over 6.2 million geonames toponyms now have a unique URL with a corresponding RDF web service. Other services describe the relation between toponyms.

A typical individual in the database is an instance of type Feature and has a Feature Class associated with it. These Feature Classes can be administrative divisions, populated places, structures, mountains, water bodies, etc. Though the Feature Class is subcategorized into 645 different Feature Codes, the Feature Code is associated with a Feature instance and not as a specialization of the property feature Class (this is probably due to automatically exporting of existing relational data into RDF rather than building data conforming to an ontology). A Feature also has several other properties, such latitude, longitude, and an owl:sameAs property linking it to an instance from DBPEDIA.[19]

4.2.3 GeoSPARQL

This standard[78] defines a set of SPARQL extension functions, a set of RIF rules[79] , and a core RDF/OWL vocabulary for geographic information based on the General Feature Model, Simple Features ISO 19125-1[80], Feature Geometry and SQL MM.

The OGC GeoSPARQL[16] standard supports representing and querying geospatial data on the Semantic Web. GeoSPARQL defines a vocabulary for representing geospatial data in RDF and it defines an extension to the SPARQL query language for processing geospatial data.

The GeoSPARQL ontology defines two main classes: geo:SpatialObject and geo:Feature. The class Spatial Object represents everything that can have a spatial representation. It is a superclass of feature and geometry. The class Feature represents the top-level feature type. This class is equivalent to GFI_Feature defined in ISO 19156 Geographic information -- Observations and measurements[81].

[76] http://www.geonames.org/
[77] http://www.geonames.org/ontology
[78] http://www.opengeospatial.org/standards/geosparql
[79] http://www.w3.org/TR/rif-core/
[80] http://www.iso.org/iso/catalogue_detail.htm?csnumber=40114
[81] http://portal.opengeospatial.org/files/?artifact_id=41579

The class geo:Geometry represents the top-level geometry type. This class is equivalent to the UML class GM_Object defined in ISO 19107:2003 Geographic Information-Spatial schema[82], and it is superclass of all geometry types.

The GeoSPARQL vocabulary can easily be extended with other application/domain-specific vocabularies.

More details about the vocabulary namespace can be found at http://www.opengis.net/ont/geosparql.

OGC Geometry

It is a specialization of GeoSPARQL defining specific subtypes of Geometry following the GML Geography Markup Language.

The Geography Markup Language (GML) [83] is an XML grammar for expressing geographical features. GML serves as a modeling language for geographic systems, as well as an open interchange format for geographic transactions on the Internet. GML is also an ISO standard (ISO 19136:2007).

Vocabulary namespace: http://www.opengis.net/ont/gml

OGC Simple Features

The OGC Simple Features specifications describes the common architecture for simple feature geometry. The simple feature geometry object model is Distributed Computing Platform neutral and uses UML notation. The base Geometry class has subclasses for Point, Curve, Surface and GeometryCollection. Each geometric object is associated with a Spatial Reference System, which describes the coordinate space in which the geometric object is defined.

Vocabulary namespace: http://www.opengis.net/ont/sf

4.2.4 World Geodetic System 1984 (WGS 84) Vocabulary

WGS 84 is an earth fixed global reference frame, including an earth model. WGS 84 is the reference coordinate system used by the Global Positioning System (GPS).

The WGS 84 vocabulary[84], published by the W3C, is intended for representing latitude, longitude and altitude information in the WGS84 geodetic reference datum.

This vocabulary is used by 43 (2013, September) linked datasets[85], as the European Pollutant Release and Transfer Register[86], the European Nature Information System (EUNIS)[87] and DBpedia[88] in English and French.

This vocabulary allows the definition of coordinates in the WGS84 coordinate reference system, using the "latitude", "longitude" and "altitude" predicates. It also defines a "Point" class, which are

[82] http://www.iso.org/iso/catalogue_detail.htm?csnumber=26012
[83] http://www.opengeospatial.org/standards/gml
[84] http://lov.okfn.org/dataset/lov/details/vocabulary_geo.html
[85] http://lov.okfn.org/endpoint/lov_aggregator?query=PREFIX+rdfs%3A%3Chttp%3A%2F%2Fwww.w3.org%2F2000%2F01%2Frdf-schema%23%3E%0D%0APREFIX+voaf%3A%3Chttp%3A%2F%2Fpurl.org%2Fvocommons%2Fvoaf%23%3E%0D%0A%0D%0ASELECT+%3Fdataset+%3FdatasetLabel+%3Foccurrences%0D%0AWHERE%7B%0D%0A%3Chttp%3A%2F%2Fwww.w3.org%2F2003%2F01%2Fgeo%2Fwgs84_pos%3E+voaf%3AusageInDataset+%3Fusage.%0D%0A%3Fusage+voaf%3AinDataset+%3Fdataset.%0D%0A%3Fusage+voaf%3Aoccurrences+%3Foccurrences.%0D%0A%3Fdataset+rdfs%3Alabel+%3FdatasetLabel.%0D%0A%7D+ORDER+BY+DESC%28%3Foccurrences%29&format=HTML
[86] http://datahub.io/dataset/eprtr
[87] http://datahub.io/dataset/eunis
[88] http://datahub.io/dataset/dbpedia and http://datahub.io/dataset/dbpedia-fr

resources with WGS84 coordinates. Although the W3C Geo Vocabulary is widely used to describe WGS84 coordinates, it is does not allow users the description of geometric shapes, such as the border of a country.

Vocabulary namespace: http://www.w3.org/2003/01/geo/wgs84_pos#

4.2.5 NGEO - NeoGeo Vocabulary

Numerous providers publish geospatial data as Linked Data. However, no consensus has been achieved for developing a shared RDF vocabulary with enough descriptive power for modelling geographic regions. Every publisher (and every dataset) uses its own vocabulary, making integration difficult. The NeoGeo Vocabulary is the result of discussions at numerous VoCamps[89] related to geospatial data.

A common modelling choice in virtually all geospatial formats is a distinction between a Feature (a thing with spatial extent) and a Geometry (a geometric shape). The NeoGeo Vocabulary thus makes also the distinction, and provides spatial:Feature and geom:Geometry classes. The relation between a spatial:Feature and a geom:Geometry is geom:geometry.

NGEO Geometry Vocabulary

NGEO Geometry Vocabulary is a vocabulary for specifying geographical regions in RDF[90]. The NeoGeo Vocabulary is based on the GML Simple Features Profile as defined by the OGC (Open Geospatial Consortium). Simple geometries are described (along with their coordinates) explicitly in RDF, and composite geometries are described as an aggregation of simple geometries[91].

Vocabulary namespace: http://geovocab.org/geometry#

NGEOSpatial Vocabulary

It is a vocabulary for describing topological relations between features[92].

Vocabulary namespace: http://geovocab.org/spatial#

4.2.6 Core Location Vocabulary

The Location Core Vocabulary provides a minimum set of classes and properties for describing any place in terms of its name, address or geometry. The vocabulary is specifically designed to aid the publication of data that is interoperable with the EU INSPIRE Directive. It is closely integrated with the Business and Person Core Vocabularies. The Location Core Vocabulary was developed under the European Commission's ISA Programme[93].

The Core Location RDF Vocabulary can be used as a foundational RDF Vocabulary to harmonize address data that originates from disparate organizations and systems. The Core Location RDF vocabulary can be flexibly extended with experimental INSPIRE RDF vocabularies (i.e. transport networks and administrative units). HTTP URI sets can be derived from INSPIRE thematic and external object identifiers for address data, allowing one to create harmonized Web identifiers for respectively spatial things and spatial objects such as addresses.[26].

[89] http://vocamp.org/
[90] . http://geovocab.org/geometry#
[91] http://geovocab.org/doc/neogeo.html
[92] http://geovocab.org/spatial.html
[93] http://ec.europa.eu/isa/

This vocabulary comprises three classes:

- The Geometry Class [7] of the Location Core Vocabulary denotes the notion of geometry at a conceptual level, and can be encoded in different formats including WKT, GML, KML, RDF+WKT/GML (GeoSPARQL), RDF (WGS84 lat/long, schema.org) and GeoHash URI references.
- The Address class relies on the INSPIRE Address Representation data type[8].
- Locations can be described in three principal ways: by using a place name, a geometry or an address. The Location Core Vocabulary provides structure for all three. In addition to a simple (string) label or name for a Location, this vocabulary defines a property that allows one the definition of locations by a URI, such as a GeoNames or DBpedia URI.

4.3 Resources for Temporal modelling

4.3.1 OWL-Time

OWL-Time[94] is an ontology of temporal concepts, OWL-Time (formerly DAML-Time), for describing the temporal content of Web pages and the temporal properties of Web services. The ontology provides a vocabulary for expressing facts about topological relations among instants and intervals, together with information about durations and about date time information.

This vocabulary is used by 51 linked datasets[95] (by 2013, September).

Its main features are topological temporal relations, duration description, time zones and date time description.

Ontology namespace: http://www.w3.org/2006/time

4.3.2 iCalendar

iCalendar (Internet Calendaring and Scheduling Core Object Specification) is the Internet standard for exchanging calendar information (RFC-2445[96]). The iCalendar format is suitable as an exchange format between applications or systems.

Of particular interest in iCalendar is a well-developed model of recurring events. For example, "the meeting occurs every other week on Mondays at 3:00 pm for 30 minutes"[20].

RDF Calendar

RDFCal[97] is an application of the Resource Description Framework to iCalendar Data. RDF Calendar vocabulary is still a work-in-progress, but it provides anyone with RDF or XML tools a useful alternative to dealing with the character-level syntax of iCalendar.

hCalendar

hCalendar[98] is is a simple, open format for publishing events on the web, using a 1:1 representation of iCalendar VEVENT properties and values in HTML. hCalendar is one of several open microformat standards suitable for embedding data in HTML/HTML5, and Atom/RSS/XHTML or other XML.

[94] http://www.w3.org/TR/owl-time/
[95] http://stats.lod2.eu/vocabularies/18
[96] http://www.ietf.org/rfc/rfc2445.txt
[97] http://www.w3.org/TR/rdfcal/
[98] http://microformats.org/wiki/hcalendar

4.3.3 Date-Time Foundation Vocabulary

The Date-Time Vocabulary[99] is a OMG specification that models continuous time, discrete time, the relationship of events and situations to time, language tense and aspect, time indexicals, timetables, and schedules. It offers a "business vocabulary" (i.e. a linguistically-oriented ontology), intended for application by business rules for any business domain. The vocabulary is defined in SBVR, in UML + OCL, and partially in Common Logic and in OWL [20].

The Date-Time Vocabulary addresses the same general topics as the well-known OWL-Time. Beyond the basic difference that Date-Time provides a business vocabulary and a UML model in addition to an OWL ontology, there are a few more subtle distinctions.

Date-Time also provides a generic basis for defining any type of calendar (e.g., financial calendars, religious calendars) in addition to the standard calendar, whereas OWL-Time models only the standard calendar. Date-Time addresses language tense and aspect, which are not discussed in OWL Time.

4.4 Resources for User model and context

4.4.1 Mobile ontology

Mobile ontology[100] core vocabulary is a machine readable schema intended for sharing knowledge and exchanging information both across people and across services/applications. It covers domains related to mobile communications, such as persons, terminals, services, networks. Alignment with relevant standards makes SPICE outcomes interoperable with other efforts in the field. Supporting mechanisms and techniques facilitate usage of the vocabulary and other ontological standards.

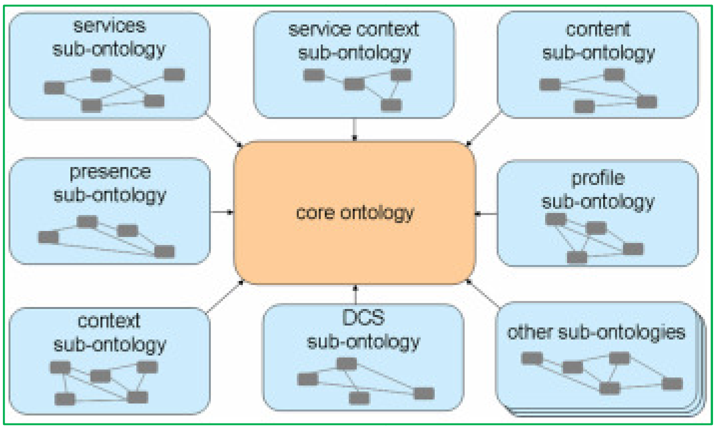

Figure 4. Mobile ontology sub-ontologies

The Mobile Ontology is structured in sub-ontologies that cover different domains and therefore an OWL file is provided for each one of these sub-ontologies. The main concepts to be defined are part of the Mobile Ontology Core and the other sub-ontologies that inherited from it. The Mobile Ontology makes also use of existing ontologies to represent concepts like time.

Mobile Ontology Core

- Mobile ontology Core v1.0.0

[99] http://www.omg.org/spec/DTV/
[100] http://ontology.ist-spice.org/mobile-ontology/1/0/core/0/core.owl#

Sub-ontologies

- Services - SPATEL v1.0.0
- Service Context v1.0.0
- Profile v1.0.0
- Presence v1.0.0
- Context v1.0.0
- Distributed Communication Sphere (DCS) v1.0.0
- Content v1.0.0
- Privacy v1.0.0

External ontologies imported into the Mobile Ontology are: Time ontology, FOAF, vCard, RDF Calendar.

4.4.2 Delivery Context ontology

The Delivery Context Ontology[101] provides a formal model of the characteristics of the environment in which devices interact with the Web or other services. The delivery context includes the characteristics of the device, the software used to access the service and the network providing the connection among others.

The delivery context is an important source of information that can be used to adapt materials to make them useable on a wide range of different devices with different capabilities. The ontology is formally specified in the Web Ontology Language -OWL.

4.4.3 CoBra (Context Broker Architecture) ontology

COBRA-ONT[102] is an ontology that defines some of the common relationships and attributes that are related to people, places and activities. The main objective of this ontology is to enable knowledge sharing and ontology reasoning within the CoBra (for Context Broker Architecture) infrastructure. COBRA-ONT defines key ontology categories such as action, agent, time, space, device, etc.

The latest version of the ontology extends ontologies from SOUPA -- Standard Ontologies for Ubiquitous and Pervasive Applications (e.g. time, space, person, policy, agents, meeting and geographical measurements).

4.4.4 CoDAMos (Context-Driven Adaptation of Mobile Services) ontology

The CoDAMoS ontology defines four main core entities: user, environment, platform, and service. This ontology has been designed with the aim of solving the following challenges: application adaptation, automatic code generation, code mobility, and generation of device-specific user interfaces.

Ontology namespace: http://www.cs.kuleuven.be/~davy/ontologies/2007/04/Context.owl

[101] http://www.w3.org/TR/2008/WD-dcontology-20080415/
[102] http://cobra.umbc.edu/ontologies.html

4.4.5 CONON

CONON (for Context Ontology) [21] defines general concepts such as location, activity, person or computational entity, whose terms are thought to be extensible in a hierarchical way by adding domain specific concepts. The authors divide their context model into an upper ontology and a specific ontology. On the one hand, the upper ontology is a high-level ontology that captures general features of basic contextual entities. On the other hand, the specific ontology defines the details of the general concepts and their features in each subdomain covered.

CONON is supposed to be used in pervasive computing environments to enable context modeling and logic-based context reasoning. Furthermore it supports interoperability of different devices as a common vocabulary. CONON defines generic concepts regarding context and provides extensibility for adding domain specific concepts. Logic reasoning is used in order to perform consistency checks and to calculate high-level context knowledge from explicitly given low-level context information.

4.4.6 GUMO: General User Model Ontology

The general user model ontology GUMO[103] is used for the uniform interpretation of distributed user models in intelligent semantic web enriched environments [5], [22]. GUMO, The General User Model (and Context) Ontology defines classes, instances and properties for describing the situation of the user, the system and the environment.

Ontology namespace: http://www.ubisworld.org/ubisworld/documents/gumo/2.0/gumo.owl

4.4.7 UUCM Unified User Context Model

B. Mehta, C. Niederee, A. Stewart, M. Degemmis, P. Lops, and G. Semeraro. Ontologically- ´Enriched Unified User Modeling for Cross-System Personalization. In Proc. of the 10th Int.Conference on User Modeling, UM '05, volume 3538 of LNCS, pages 119–123. Springer,2005

4.5 Units of measure

4.5.1 MUO - MyMobileWeb ontology of measurement units

MUO[104] is an ontology to represent units of measurement, it includes instances from UCUM (see subsection UCUM: The Unified Code for Units of Measure).

The classes considered by MUO ontology are: BaseUnit, ComplexDerivedUnit, DerivedUnit, MetricUnit, PhysicalQuality, Prefix, QualityValue, SIUnit, SimpleDerivedUnit and UnitOfMeasurement.

As explained above, the unified code for units of measure[105] has been instantiated in the MUO ontology.

The MUO ontology with instances from UCUM is http://purl.oclc.org/NET/muo/ucum/.

4.5.2 UCUM: The Unified Code for Units of Measure

The Unified Code for Units of Measure[106] is a code system intended to include all units of measures that are being contemporarily used in international science, engineering, and business. The purpose is to facilitate unambiguous electronic communication of quantities together with their units.

[103] u2m.org/2003/02/UserModelOntology.owl
[104] http://idi.fundacionctic.org/muo/
[105] http://unitsofmeasure.org/ucum.html

The Unified Code for Units of Measure is inspired by and heavily based on ISO 2955-1983, ANSI X3.50-1986, and HL7's extensions called "ISO+". The respective ISO and ANSI standards are both entitled "Representation of [...] units in systems with limited character sets" where ISO 2955 refers to SI and other units provided by ISO 1000-1981, while ANSI X3.50 extends ISO 2955 to include U.S. customary units. Because these standards carry the restriction of "limited character sets" in their names they seem to be of less value today, when graphical user interfaces and laser printers are in wide-spread use. For this reason, the European standard ENV 12435 in its clause 7.3 declares ISO 2955 obsolete.

UCUM is used by all INSPIRE themes to manage units of measure in a standard way.

4.5.3 QUDT - Quantities, Units, Dimensions and Data Types Ontologies

The QUDT[107] Ontologies, and derived XML Vocabularies, are being developed by TopQuadrant and NASA. Originally, they were developed for the NASA Exploration Initiatives Ontology Models (NExIOM) project, a Constellation Program initiative at the AMES Research Center (ARC). They now for the basis of the NASA QUDT Handbook to be published by NASA Headquarters.

The QUDT approach to specifying quantities, units, dimensions, data types, enumerations, and other data structures is to use precise semantically grounded specifications in an ontology model with translation into machine-processable representations.

QUDT semantics are based on dimensional analysis expressed in OWL. The dimensional approach relates each unit to a system of base units using numeric factors and a vector of exponents defined over a set of fundamental dimensions. In this way, the role of each base unit in the derived unit is precisely defined. A further relationship establishes the semantics of units and quantity kinds. By this means, QUDT supports reasoning over quantities as well as units. All QUDT models may be translated into other representations for machine processing, or other programming language structures according to need.

4.5.4 QUDV - Quantities, Units, Dimensions, Values

The conceptual model for Quantities, Units, Dimensions, Values (QUDV)[108] defines systems of units and quantities for use in system models. It is based on concepts of the International Vocabulary of Metrology (VIM)[109] and defined by the Object Management Group (OMG)[110] Systems Modeling Language (SysML).

QUDV has been formalized in UML and in OWL[111.]

4.6 Existing ontologies and other resources on the Sensor domain

This chapter describes existing ontological and non-ontological resources in the sensor domain and the specific knowledge domain each of them describe, when applicable.

---

106 http://unitsofmeasure.org/ucum.html
107 http://www.qudt.org/
108 http://www.omgwiki.org/OMGSysML/doku.php?id=sysml-qudv:quantities_units_dimensions_values_qudv
109 http://www.bipm.org/en/publications/guides/vim.html
110 http://www.omg.org/
111 http://www.omgwiki.org/OMGSysML/doku.php?id=sysml-qudv:qudv_owl

4.6.1 Semantic Sensor Network Ontology

The W3C Semantic Sensor Network Incubator Group has developed a formal OWL DL ontology[112] for modelling sensor devices (and their capabilities), systems and processes. This ontology describes sensors and observations, and related concepts. It does not describe domain concepts, time, locations, etc. These are intended to be included from other ontologies via OWL imports.

It is based in part on the **ISO 19156 "Observations and Measurements"** conceptual model.

The ontology is based around concepts of **systems, processes, and observations**. It supports the description of the **physical and processing structure of sensors**. Sensors are not constrained to physical sensing devices: rather a sensor is anything that can estimate or calculate the value of a phenomenon, so a device or computational process or combination could play the role of a sensor. The representation of a sensor in the ontology links together what it measures (the domain phenomena), the physical sensor (the device) and its functions and processing (the models).

The Semantic Sensor Network ontology revolves around the central Stimulus-Sensor-Observation pattern:

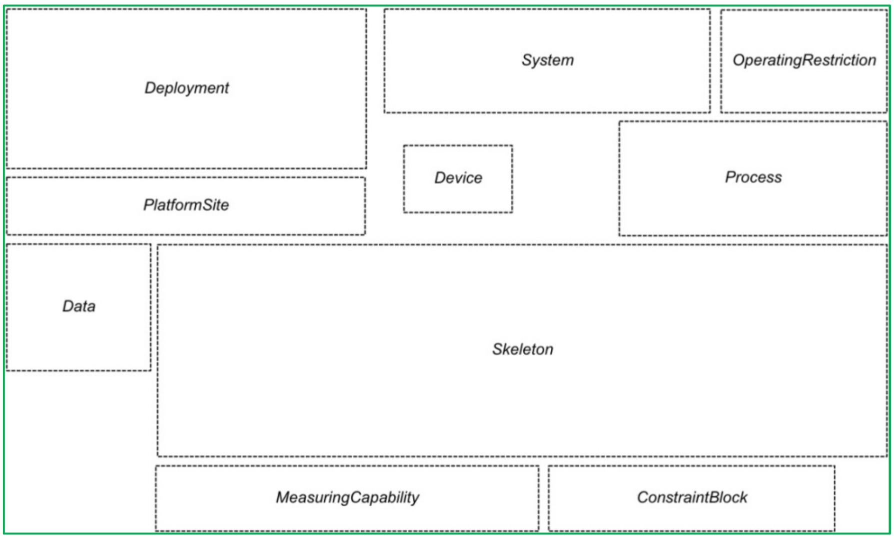

Figure 5. The SSN ontology, Overview of the Semantic Sensor Network ontology modules

The ontology can be used for a focus on any (or a combination) of a number of perspectives:

- A sensor perspective, with a focus on what senses, how it senses, and what is sensed;
- A data or observation perspective, with a focus on observations and related metadata;
- A system perspective, with a focus on systems of sensors; or,
- A feature and property perspective, with a focus on features, properties of them, and what can sense those properties.

The SSN ontology is organized, conceptually but not physically, into ten modules [17] as shown in Figure 6:

[112] Semantic Sensor Network Ontology: http://www.w3.org/2005/Incubator/ssn/ssnx/ssn
Name Space: http://purl.oclc.org/NET/ssnx/ssn

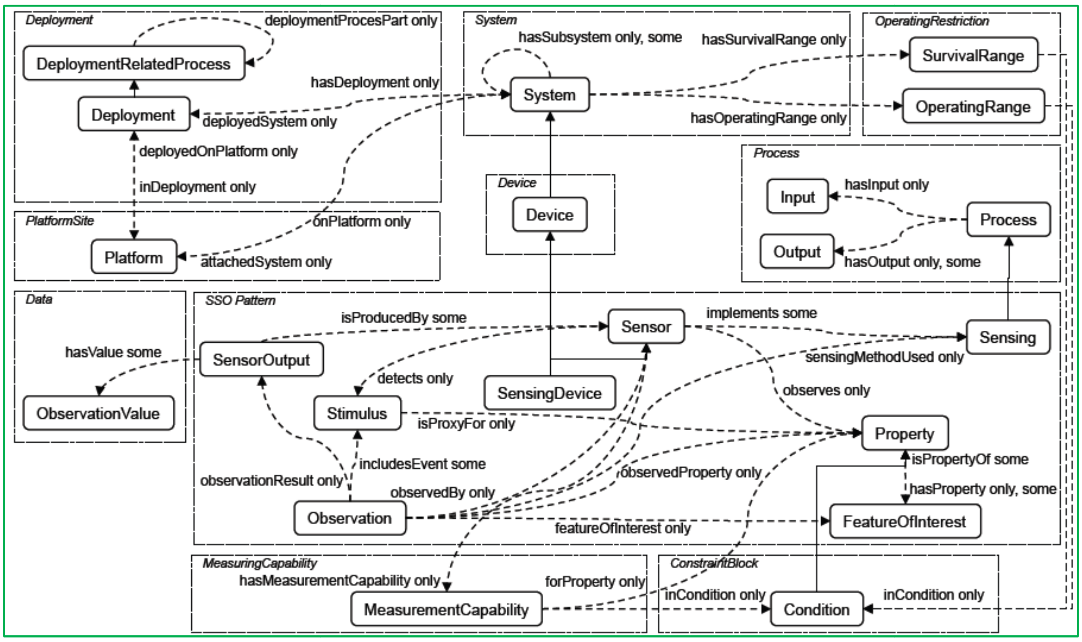

Figure 6. The SSN ontology, key concepts and relations, split by conceptual modules

The full ontology consists of 41 concepts and 39 object properties, directly inheriting from 11 DUL (Dolce Ultra Lite) concepts and 14 DUL object properties.

The ontology can describe sensors, the accuracy and capabilities of such sensors, observations and methods used for sensing. Also concepts for operating and survival ranges are included, as these are often part of a given specification for a sensor, along with its performance within those ranges. Finally, a structure for field deployments is included to describe deployment lifetime and sensing purpose of the deployed macro instrument.

Although the ontology **leaves the observed domain unspecified, domain semantics, units of measurement, time and time series, and location and mobility ontologies can be easily attached when instantiating the ontology for any particular sensors in a domain.** The alignment between the SSN ontology and the DOLCE Ultra Lite (DUL) upper ontology has helped to normalise the structure of the ontology to assist its use in conjunction with ontologies or linked data resources developed elsewhere[113].

The main classes of the Semantic Sensor Network ontology have been aligned with classes in the DOLCE Ultra Lite (DUL) foundational ontology to facilitate reuse and interoperability. Figure 7. shows in blue arrows the subclass properties used to align these two ontologies:

[113] http://www.w3.org/2005/Incubator/ssn/XGR-ssn-20110628/

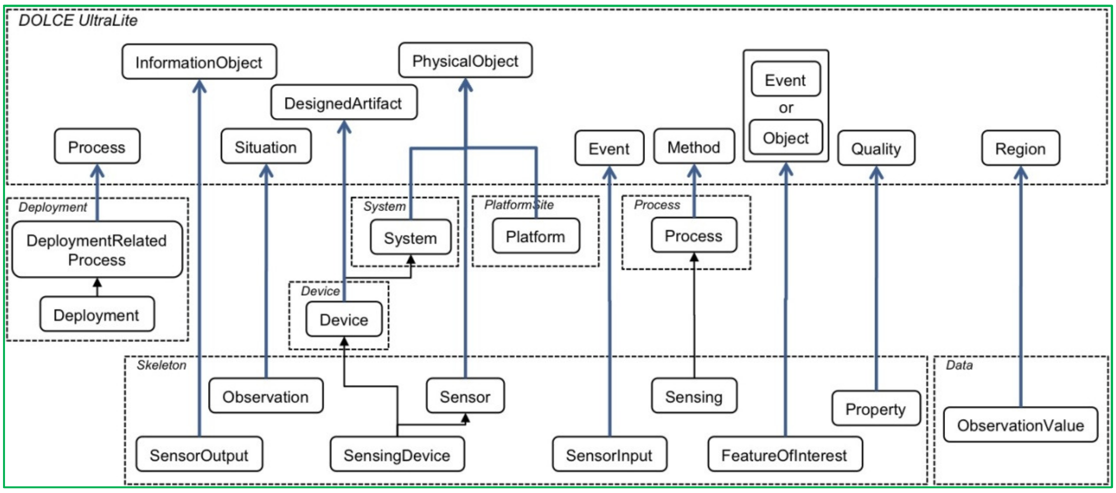

Figure 7. Alignment of the Semantic Sensor Network ontology to DOLCE Ultra Lite

The SSN ontology structure can be consulted at http://www.w3.org/2005/Incubator/ssn/XGR-ssn-20110628/#Ontology_structure.

4.6.2 Core ontology – SOFIA project

SOFIA (Smart Objects For Intelligent Applications) was an ARTEMIS project aiming the creation of an Open Innovation Platform (OIP) providing interoperability for interaction between multi-vendor devices. The defined scenarios to demonstrate the capabilities of the OIP are personal spaces, indoor spaces and cities. **The ontology to model the city** is wp3.owl[114]. This ontology models concepts as: Temperature, Vibration, Smoke, Presence, Lights, Humidity, Stress, GPS location, Noise, Mobile and Network device, etc. Making special incidence in events related for instance to high or low temperature, high or low humidity, high noise etc. The modelled sensors correspond to: Temperature, Vibration, Smoke, Presence, Humidity, StrainGauge.

4.6.3 R3COP Project

This ARTEMIS project developed an ontology to be used in the domain of algorithms for the reasoning and mission planning module of a robot model. The main ontology classes are shown in Figure 8. R3-COP Sensor ontology main classes:

[114] http://emb1.esilab.org/sofia/ontology/1.3/city/wp3.owl

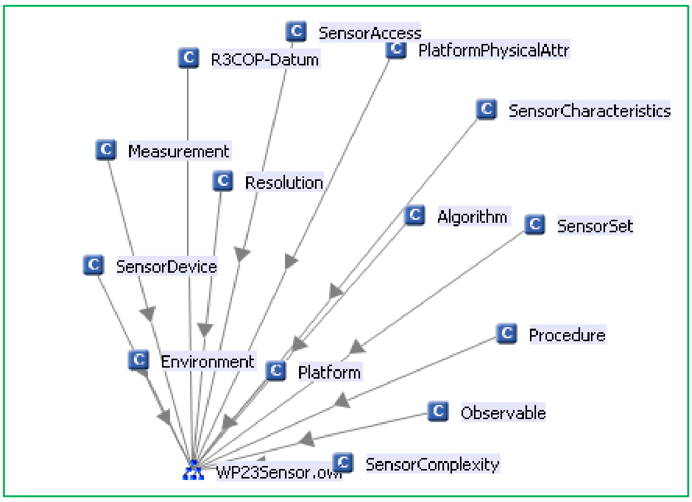

Figure 8. R3-COP Sensor ontology main classes

The sensor devices are classified according to its main features, as shown in Figure 9. SensorDevice Class:

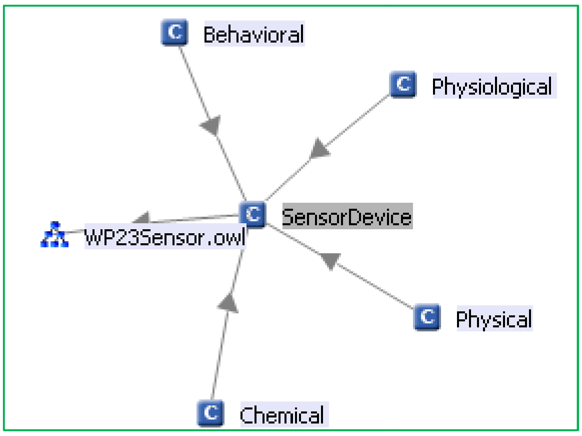

Figure 9. SensorDevice Class

4.6.4 Corelf Vocabulary

The corelf vocabulary from DERI[115] specifies terms for use by constrained devices to describe hosted resources, their attributes and other relationships between links. The semantics of the terms is based on the CoRE Link Format[116], the CoAP Profile Description Format[117], the Media Types for Sensor Markup Language (SenML)[118], as well as the Constrained Application Protocol (CoAP)[119].

[115] http://vocab.deri.ie/corelf.html
[116] http://tools.ietf.org/html/draft-ietf-core-link-format
[117] http://tools.ietf.org/html/draft-greevenbosch-core-profile-description
[118] http://tools.ietf.org/html/draft-jennings-senml
[119] http://tools.ietf.org/html/draft-ietf-core-coap

5. City Observatories Ontology

The City Observatories ontology will be implemented in order to address the requirements originating from a) the empowerment initiatives, b) existing standards in the sensor world, c) best practices in ontology reuse and d) current directives as INSPIRE that dictate how to share spatial data and trying to answer to what a citizen observatory is:

- What is it? A place where all the measures and observations taken by citizens are published and globally accessible by a unique access point.
- How to implement it? Publication under the Linked Data paradigm of the citizens' measurements and observations (coming for the Empowerment Initiatives: WP2 & 3).
- What is the aim? It allows sharing the data with the world and implementing new applications and policies by stakeholders.

This chapter begins with the definition of the core schema of the ontology. This core explains the different knowledge subdomains to be covered and the different decisions taken to cover them: the required alignments between selected ontologies and the new ontology implementation needed to fulfil all the requirements for a core ontology network for city observatories applications.

The final subsection provides further hints of the ontology usage for observations and measures annotations in the different pilots.

5.1 Core schema

The city observatories ontology will adopt the Semantic Sensor Network (SSN) ontology as the upper ontology. The SSN ontology is considered as de facto standard in the sensor world, so this ontology ensures future enlargement of CITI-SENSE knowledge domain as well as makes possible links to other existing efforts in sensor data gathering and publication and citizen empowerment.

But the SSN ontology leaves the observed domain unspecified that is why other ontologies have been selected and aligned with the SSN ontology, where each ontology has been selected according to a concrete requirement. The requirements are summarised below:

- **When are we measuring?** The answer to this question is modelled by the OWL-Time ontology, a W3C-recommended ontology based on temporal calculus, that provides descriptions of temporal concepts such as instant and interval and which supports defining interval queries such as within, contains, and overlaps. The primitive data types used by OWL-TIME ontology use lexical formats inspired by ISO 8601[120].
- **Where are we measuring?** The INSPIRE Directive defines how the different locations should be modelled. The scope as defined in the INSPIRE Directive includes two aspects: the environmental monitoring facility as a spatial object in the context of INSPIRE and Observations and Measurements linked to the environmental monitoring facility.
- **What are we measuring**? The answer to this question covers the CITI-SENSE knowledge domain and it is represented by the requirements coming from the empowerment initiatives under development in Work packages 2 and 3.

[120] http://www.w3.org/TR/xmlschema-2/#ISO8601

- **Who and how are we measuring?** The citizens actively involved in sensor measurements are also an important part because their social and personal context (e.g. age, occupation) can influence the analysis of the gathered sensor data.

Figure 10 shows the different aspects to be covered by the city observatories ontology:

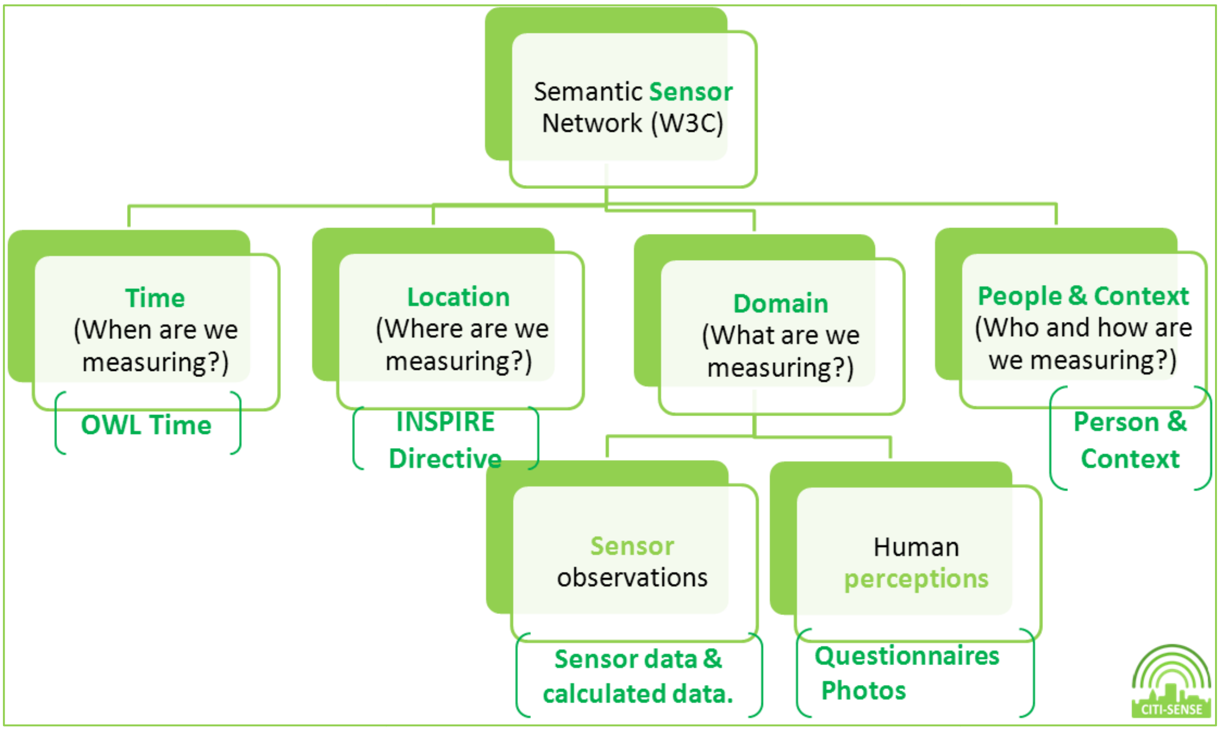

Figure 10. Core ontology network for city observatories applications

Ontology Namespace: http://www.citi-sense.eu/2013/cityobservatories

5.1.1 Ontology network

The following ontologies have been imported to the city observatories ontologies, for each ontology it is indicated the ontology namespace:

- **SSN: Semantic Sensor Network Ontology** (includes DUL): http://purl.oclc.org/NET/ssnx/ssn
- **OWLTime**: http://www.w3.org/2006/time. It is the current recommendation of W3C for Time modelling.
- **GeoNames Ontology**: http://www.geonames.org/ontology. This ontology allows linking a DUL:Entity to a location in GeoNames dataset.
- **UCUM**: http://purl.oclc.org/NET/muo/ucum/. UCUM is used by all INSPIRE themes to manage Units of measure in a standard way.[12]
- **LOCN: Core Location Vocabulary**: http://www.w3.org/ns/locn#. It is the vocabulary recommended for INSPIRE implementation and by ISA programme as best practice in the Joinup collaborative platform.[7]
- **GeoSPARQL** http://schemas.opengis.net/geosparql/1.0/geosparql_vocab_all.rdf. GeoSPARQL is an emerging standard from the OGC that aims to to query for geospatial relationships, providing

a common representation of geospatial RDF data and the ability to query and filter on the relationships between geospatial entities.[18].

- **WGS84_POS**: http://www.w3.org/2003/01/geo/wgs84_pos. As implementation of Point concept for GeoSPARQL.
- **FoaF:** http://xmlns.com/foaf/0.1/ As implementation of a person context.
- **SKOS:** http://www.w3.org/2008/05/skos Simple Knowledge Organization Scheme. Used to represent term lists, controlled vocabularies and thesauri.
- **INSPIRE ontology**, http://inspire.ec.europa.eu. Ontology developed during task 7.1 inside CITI-SENSE project, that considers a subset of the INSPIRE data specifications, see "5.2 INSPIRE directive implementation" for implementation details.

The following figure shows the ontology import hierarchy:

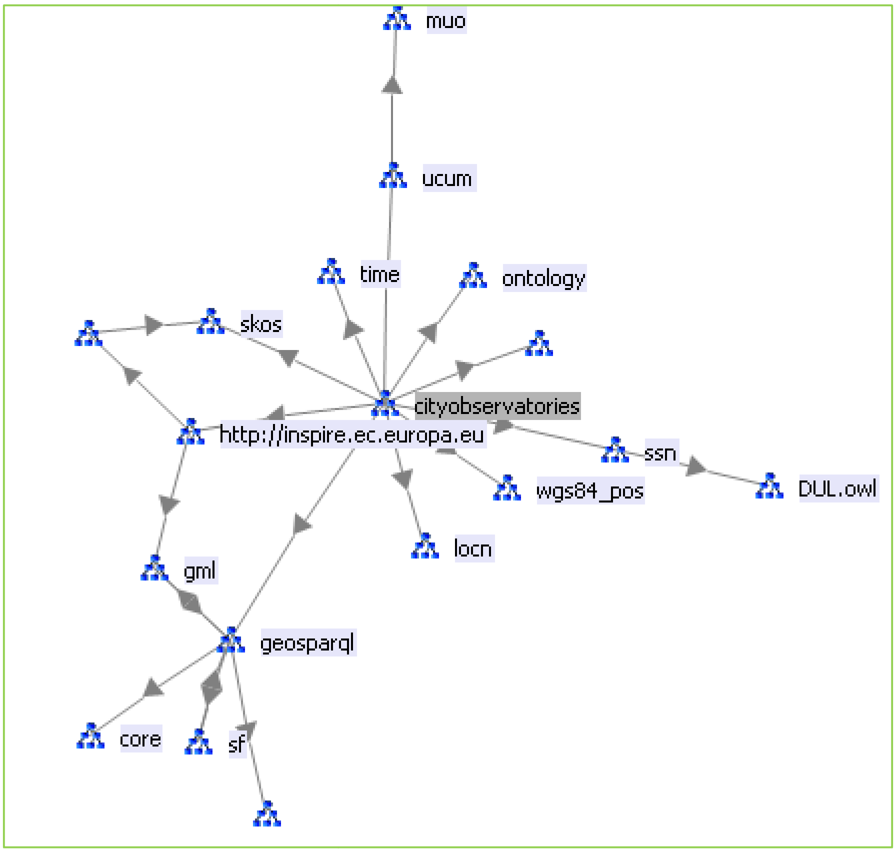

Figure 11. CITI-SENSE Ontology import hierarchy

5.1.2 Alignments and modifications done

Once the ontologies were imported it was necessary to align them in order to create the ontology network.

For each alignment it is indicated the classes and the relations used as well as the reason for such alignment.

- SSN - OWL-TIME. This alignment allows determining the instant where an observation takes place.
    - ssn:SensingDevice ssn:observationResultTime time:instant.
        - SensingDevice class of SSN ontology is related with Instant class of OWL-Time ontology by the relation observationResultTime from SSN ontology.
        - The result time shall describe the time when the result became available, typically when the procedure associated with the observation was completed. For some observations this is identical to the phenomenonTime. However, there are important cases where they differ.
- SSN - UCUM
    - ucum:PhysicalQuality subclassOf ssn:Property
        - ssn:Property is an observable Quality of an Event or Object. That is, not a quality of an abstract entity as is also allowed by DUL's Quality, but rather an aspect of an entity that is intrinsic to and cannot exist without the entity and is observable by a sensor. The SSN ontology does not contain[121] a vocabulary of possible properties which can be measured by sensors. Specific instances of the class ssn:Property have to be created by the user or (preferably) imported from an existing ontology. In this case UCUM is the selected ontology for this purpose.
    - ssn:SensingDevice ssn:observes ucum:PhysicalQuality
        - ss:observes models the relation between a Sensor and a Property that the sensor can observe. Note that, given the DUL modelling of Qualities, a sensor defined with 'observes only Windspeed' technically links the sensor to particular instances of Windspeed, not to the concept itself - OWL can't express concept-concept relations, only individual-individual. The property composition ensures that if an observation is made of a particular quality then one can infer that the sensor observes that quality.
        - ucum:PhysicalQuality is the class which instances are the different properties to be observed, for instance, temperature, wind speed and humidity.
- SSN - LOCN
    - ssn:Platform locn:geometry locn:Geometry.
        - The SSN ontology leaves the geometry implementation open, in this case LOCN vocabulary has been selected for this aim.
        - ssn:Platform is an Entity to which other Entities can be attached - particularly Sensors and other Platforms. For example, a post might act as the Platform, a buoy might act as a Platform, or a fish might act as a Platform for an attached sensor.
        - locn:geometry associates any location with the locn:Geometry class.
- LOCN – GEOSPARQL

[121] http://www.w3.org/2005/Incubator/ssn/wiki/SSN_Sensor

- locn:Geometry class equivalent geosparql:Geometry class
  - The locn:Geometry class denotes the notion of geometry at a conceptual level, and can be encoded in different formats including GeoSPARQL[7].
- LOCN-INSPIRE
  - locn:Address class equivalent to inspire:Address class.
    - Both classes are compliant with INSPIRE directive D2.8.I.5 INSPIRE Data Specification on Addresses.[8]
    - The INSPIRE data specifications are legally binding. For addresses, Core Location can be seen as a subset of the INSPIRE address specification, as it based on the INSPIRE AddressRepresentation class. The Core Location Vocabulary has an RDF Schema representation, whereas the INSPIRE data specifications are presented as W3C XML Schemas[26]
  - http://www.w3.org/ns/locn#geometry relates inspire:NamedPlace with locn:Geometry
- GEOSPARQL-WGS84_POS
  - http://www.opengis.net/ont/sf#Point sublclass of WGS84_POS:Point.
    - WGS84_POS: Point is the implementation of concept Point in GeoSPARQL.
- INSPIRE-DUL
  - inspire:GovernmentalService subclass of DUL:Organization and DUL:PhysicalPlace
  - Inspire:NamedPlace subclass of DUL:Place
  - inspire:GeographicalName subclass of DUL: InformationObject
  - inspire:SpellingOfName subclass of DUL:InformationRealization
  - inspire:AbstractConstruction subclass of DUL:DesignedArtifact
  - inspire:ExistingLandUseObject subclass of DUL:PhysicalPlace
- DUL:GEONAMES
  - GEONAMES:Feature subclass of DUL:PhysicalPlace.
- GEONAMES-INSPIRE
  - GEONAMEs:GeonamesFeature subclass of INSPIRE:NamedPlace.
- DUL-FOAF
  - foaf:Person class equivalent to DUL:NaturalPerson.
- DUL-TIME
  - dul:TimeInterval super class of time:TemporalEntity.
    - Dul:TimeInterval is any Region in a dimensional space that aims at representing time, time:TemporalEntity is super class of Instant and Interval of time.
- DUL-UCUM

- muo:UnitsOfMeasure equivalent class dul:UnitOfMeasure
  - The different units of measure defined by UCUM are equal to the concept unit of measure from DUL. The unit of measure will be assigned to the instance of dul:Amount that collects the measure and the time stamp.

The modifications performed, mainly as far as instances creation is concerned, are:

- Creation of new instances of class muo:PhysicalQuality. (See sub-section "5.3 How to use the ontology?").
- In case new units of measure be necessary, they can be added as instances of class muo:UnitOfMeasurement and, when applicable, directly to some of its subclasses.

5.2 INSPIRE directive implementation

For the ontological implementation of the INSPIRE directive some basic modelling principles[122] have been followed:

- Retain INSPIRE-naming
- Introduce resolvable HTTP-URIs
- Optimized data types for querying:
  - geometry storage: GeoSPARQL geo-vocabulary
  - measure values: Measure Unit Ontology (MUO)
- INSPIRE ontologies derived from INSPIRE UML-models
- The URIs should use the following structure,
  - in the case of code lists:
    - http://inspire.ec.europa.eu/codelist/<CodeListId> for code lists
    - http://inspire.ec.europa.eu/codelist/<CodeListId>/<valueId> for code list values
  - in the case of classes http://inspire.ec.europa.eu/<class-name>
  - in the case of properties and relations: http://inspire.ec.europa.eu/<propertyName>

In order to incorporate INSPIRE Directives to the ontology two new ontologies have been created: http://inspire.ec.europa.eu/ and http://inspire.ec.europa.eu/codelist/. The former imports the latter.

In the case of CITI-SENSE not all the INSPIRE themes are necessary, the following are the ones implemented as well as the corresponding code lists:

- Geographical Names.
- Utility and Governmental Services.
- Buildings.

The following chapters are devoted to detail how the different INSPIRE Data Specification themes, detailed above, have been modelled by ontology means.

[122] http://asio.bbn.com/terracognita2011/presentations/Semantic_Access_To_Inspire.pdf

5.2.1 INSPIRE Data Specification on Geographical Names

According to the INSPIRE Data Specification on Geographical Names document published the April 26th, 2010[123] classes formalising NamedPlace, GeographicalName and SpellingOfName have been defined. The CITI-SENSE ontology will model all of them. The following figure shows the UML diagram for Geographical Names.

Figure 12. UML class diagram: Overview of the Geographical names application schema

Ontology modelling

The INSPIRE classes being modelled are the following:

- Ontology **http://inspire.ec.europa.eu/**:
  - **NamedPlace**
    - inspireID: Data property http://inspire.ec.europa.eu#inspireID
    - name: http://inspire.ec.europa.eu#name relates NamedPlace and GeographicalName. Minimal cardinality = 1.

[123] http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_GN_v3.0.1.pdf

- type: http://inspire.ec.europa.eu#type relates NamedPlace and <http://inspire.ec.europa.eu/codelist#NamedPlaceTypeValue>. Minimal cardinality = 1.
- Alignment with CITISENTE ontology:
  - **NamedPlace** subclass of DUL:Place.
  - Geometry: http://www.w3.org/ns/locn#geometry relates NamedPlace with <http://www.w3.org/ns/locn#Geometry>

- **GeographicalName**
  - spelling http://inspire.ec.europa.eu #spelling relates with SpellingOfName. Minimal cardinality = 1.
  - pronunciation: http://inspire.ec.europa.eu#pronunciation related with PronunciationOfName. Minimal cardinality = 1.
  - namestatus: http://inspire.ec.europa.eu#nameStatus relates GeographicalName and NameStatusValue.
  - Alignment with CITI-SENSEontology:
    - **GeographicalName** subclass of DUL: InformationObject.
- **SpellingOfName** subclass of DUL:InformationRealization.
  - text: Data property http://inspire.ec.europa.eu#text
  - Alignment with CITI-SENSE ontology:
    - **SpellingOfName** subclass of DUL:InformationRealization
- **PronunciationOfName** subclass of DUL:InformationRealization.
  - pronunciationIPA: Data property http://inspire.ec.europa.eu#pronunciationIPA.

- Ontology **http://inspire.ec.europa.eu/codelist**:
  - **NamedPlaceTypeValue** instance of skos:Concept. The different codes are instances of NamedPlaceTypeValue.
  - **NameStatusValue** instance of skos:Concept. The different codes are instances of NameStatusValue.
  - The rest of code lists should be implemented in the same way:, Nativeness Value, GrammaticalGenderValue, GrammaticalNumberValue.

The ontology formalised the Geographical names application schema as shown below:

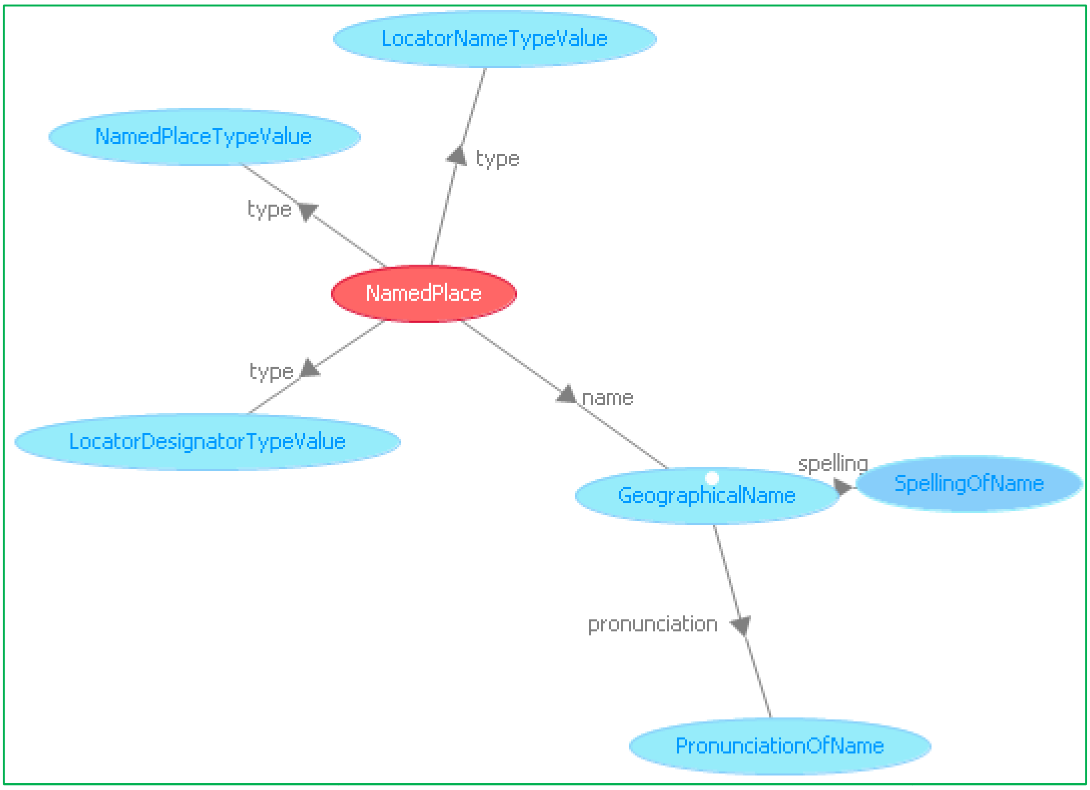

Figure 13. Implementation of Geographical names application schema

5.2.2 INSPIRE Data Specification on Utility and Governmental Services

The Utility and Governmental Services theme[124] is defined within the INSPIRE Directive as:

"Includes utility facilities such as sewage, waste management, energy supply and water supply, administrative and social governmental services such as public administrations, civil protection sites, schools and hospitals." [Directive 2007/2/EC].

The theme Utility and Government Services is divided in three sub-themes, dealing respectively with:

- Utility networks: include the physical constructions for transport of defined utility products (namely pipelines for transport of oil, gas, chemicals, water, sewage and thermal products), transmission lines and cables (included those for transmission of electricity, phone and cable-TV signals) and other network elements for encasing pipes and cases (e.g. ducts, poles and towers).
- Administrative and social governmental services comprise administrative and social governmental services such as public administrations, civil protection sites, schools and hospitals. The kind of sites that are commonly presented in governmental and municipal portals and map systems as "points of interest"-data (**POI**), and may be point-based location of a variety of categories of municipal and governmental services and social infrastructure.

[124] http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_US_v3.0rc3.pdf

- Environmental management facilities comprise several categories in order to identify the environmental protection facilities. Categories such as waste treatment sites, waste treatment facilities, regulated and illegal areas for dumping, mining waste and sewage sludge.

The following figure shows the different application schemas of the three sub-themes:

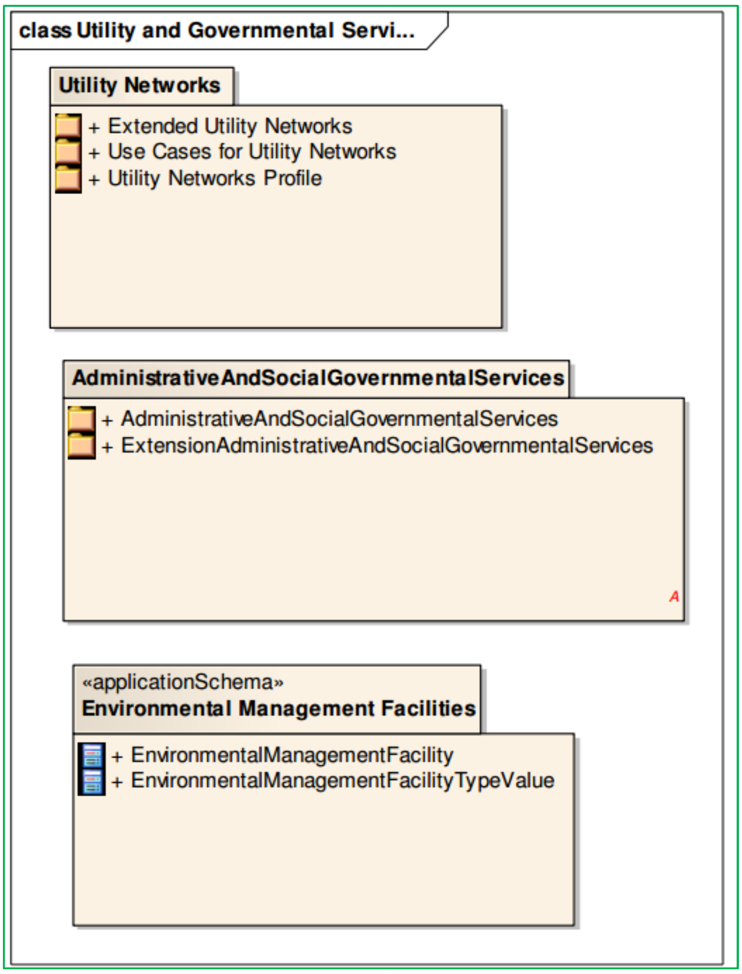

Figure 14. UML class diagram: Overview of the "Utility and governmental services" theme

As explained, this data specification defines three sub-themes, but in the case of CITI-SENSE the ontology is focused on the **AdministrativeAndSocialGovernmentalServices** sub-theme, according to the different pilot specification. The application schemas for **AdministrativeAndSocialGovernmentalServices** are:

- The "Administrative and Social Governmental Services" application schema that provides information concerning the location and the type of administrative and social governmental services;
- The "Extended Administrative and Social Governmental Services" application schema that provides more detailed information concerning administrative and social governmental services such as occupancy, resources and other specific descriptions;

The following figure shows the class diagram for AdministrativeAndSocialGovernmentalServices:

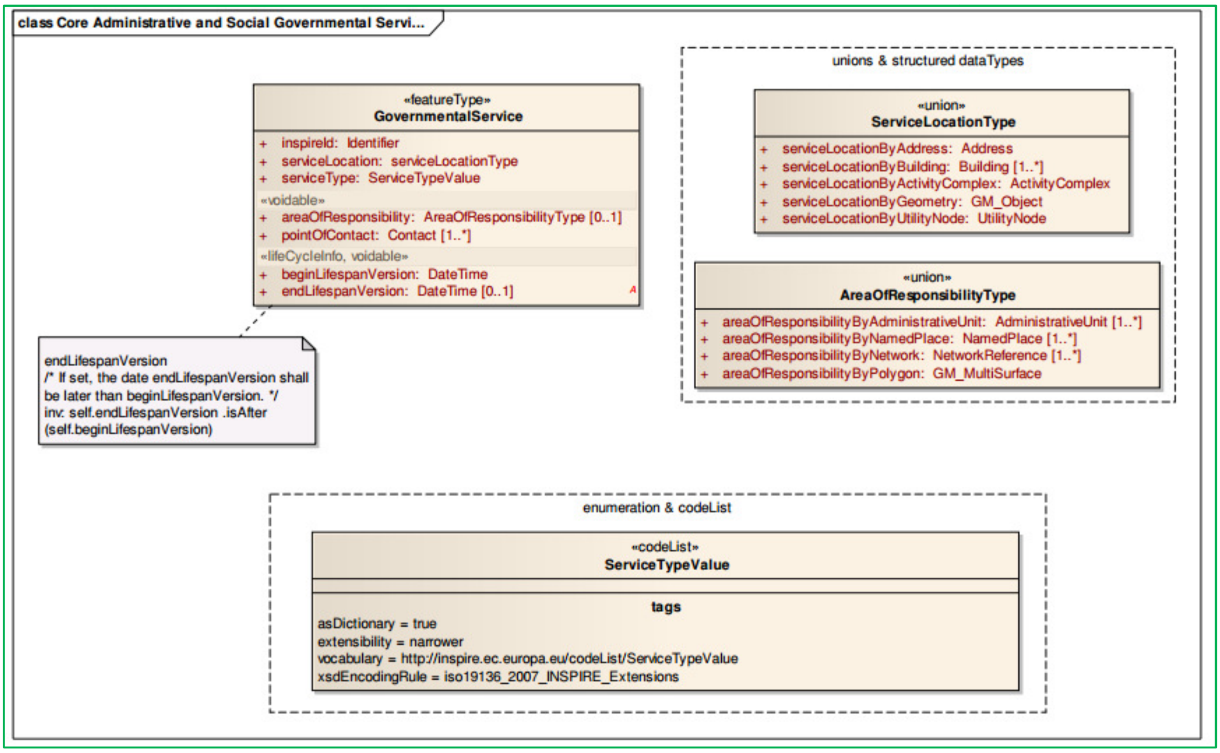

Figure 15. Class diagram: Overview of the "Administrative and Social Governmental Services"

Ontology modelling

- GovernmentalService
  - inspireID. Data property http://www.tecnalia.com/2013/citi-sense#inspireID
  - serviceLocation relates GovernmentalService with ServiceLocationType.
  - serviceType: http://inspire.ec.europa.eu#serviceType relates GovernmentalService and ServiceTypeValue.
  - Alignment with CITISENTE ontology:
    - GovernmentalService subclass of http://www.loa-cnr.it/ontologies/DUL.owl#Organization, and <http://www.loa-cnr.it/ontologies/DUL.owl#PhysicalPlace>
- ServiceLocationType
  - serviceLocationByAddress: http://inspire.ec.europa.eu#serviceLocationByAddress relates ServiceLocationType and Address.
  - serviceLocationByBuilding: http://inspire.ec.europa.eu#serviceLocationByBuilding relates ServiceLocationType and Building.
  - serviceLocationByGeometry: http://inspire.ec.europa.eu#serviceLocationByGeometry relates ServiceLocationType and http://www.opengis.net/ont/sf#Geometry.

The formalised ontology is shown below:

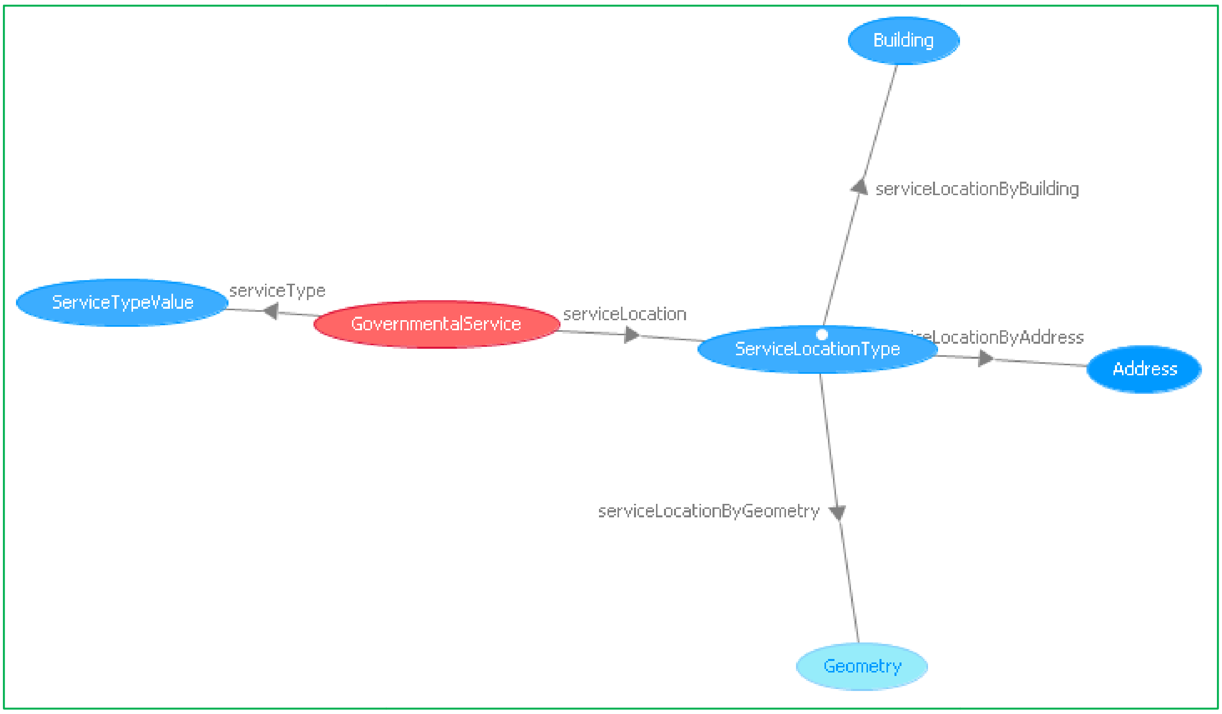

Figure 16. Implementation of GovernmentalService

5.2.3 INSPIRE Data Specification on Buildings

From the whole Data Specification [9] only a relevant part has been extracted to be modelled by CITI-SENSE ontology. So, some classes and code lists are not considered in the final ontology.

The following classes and code lists are the ones modelled by CITI-SENSE ontology:

- AbstractBuilding
- AbstractConstruction
- Building
- BuildingPart
- externalReference
- The code lists for classification of Buildings.

Following, the data extracted from the Data Specification explaining the above classes and code list is explained:

Buildings are enclosed constructions above and/or underground which are intended or used for the shelter of humans, animals, things or the production of economic goods and that refer to any structure permanently constructed or erected on its site.

A BuildingPart is a sub-division of a Building that might have been considered as a building and that is homogeneous related to its physical, functional or temporal aspects. It is up to each data producer to define what is considered as a Building and what is considered as a BuildingPart (if this concept is used). This information has to be provided as metadata.

Base application schema includes two abstract feature types: AbstractConstruction and AbstractBuilding:

- AbstractBuilding is an abstract feature type grouping the common properties of instanciable feature types Building and BuildingPart
- AbstractConstruction is an abstract feature type grouping the semantic properties of buildings, building parts and of some optional feature types that may be added to core profiles, in order to provide more information about theme Buildings. The optional feature types are described in extended application schemas.

Instanciable feature types Building and BuildingPart inherit both of the properties of abstract feature types AbstractConstruction and AbstractBuilding.

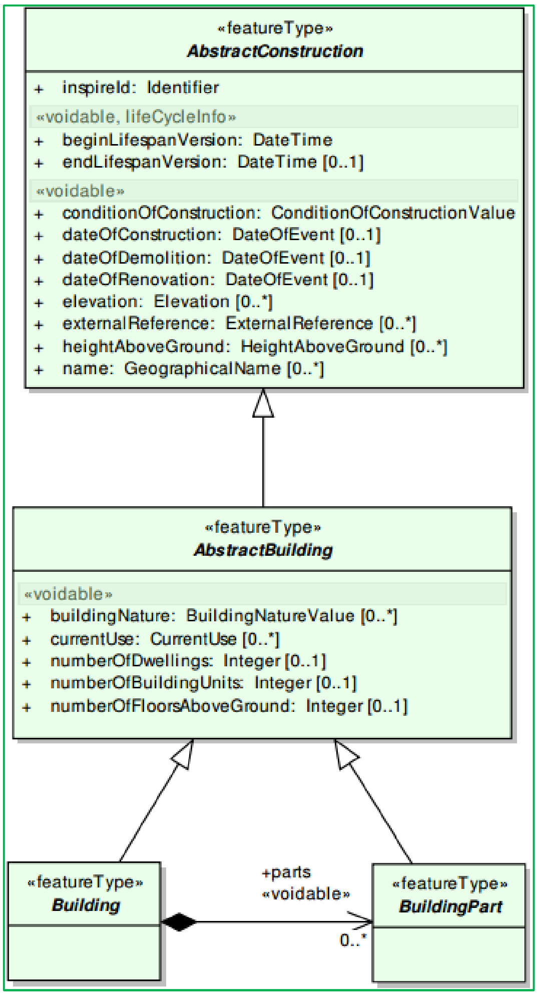

Figure 17. Feature types of Buildings Base application schema

Classification of buildings

The classification of buildings has to be done using two attributes:

- The attribute CurrentUse focuses on the activity hosted by the building. It aims to fulfil management requirements, such as computation of population or spatial planning; this classification aims to be exhaustive for the functional buildings hosting human activities.

- The attribute buildingNature that focuses on the physical aspect of the building; however, this physical aspect is often expressed as a function (e.g. stadium, silo, windmill); this attribute aims to fulfil mainly mapping purposes and addresses only specific, noticeable buildings. This is a rather short and simple list of possible values, with focus on two international use cases: air flights where buildings may be obstacles and marine navigation where buildings may be landmarks.

The code list for attribute buildingNature may be extended by Member States, in order to fulfil more mapping requirements. The attribute currentUse may take its possible values in a hierarchical code list. This hierarchical code list should enable easy matching from existing classifications to the INSPIRE classification:

- a data producer with simple classification may match at the upper level of INSPIRE classification (e.g. residential / agriculture / industrial / commerceAndService)
- a data producer with a more detailed classification may match at the lower levels of INSPIRE classification (e.g. moreThanTwoDwellings, publicServices, …).

The code list for attribute currentUse may also be extended by Member States, but only by providing more detailed values, under the hierarchical structure of the INSPIRE code list.

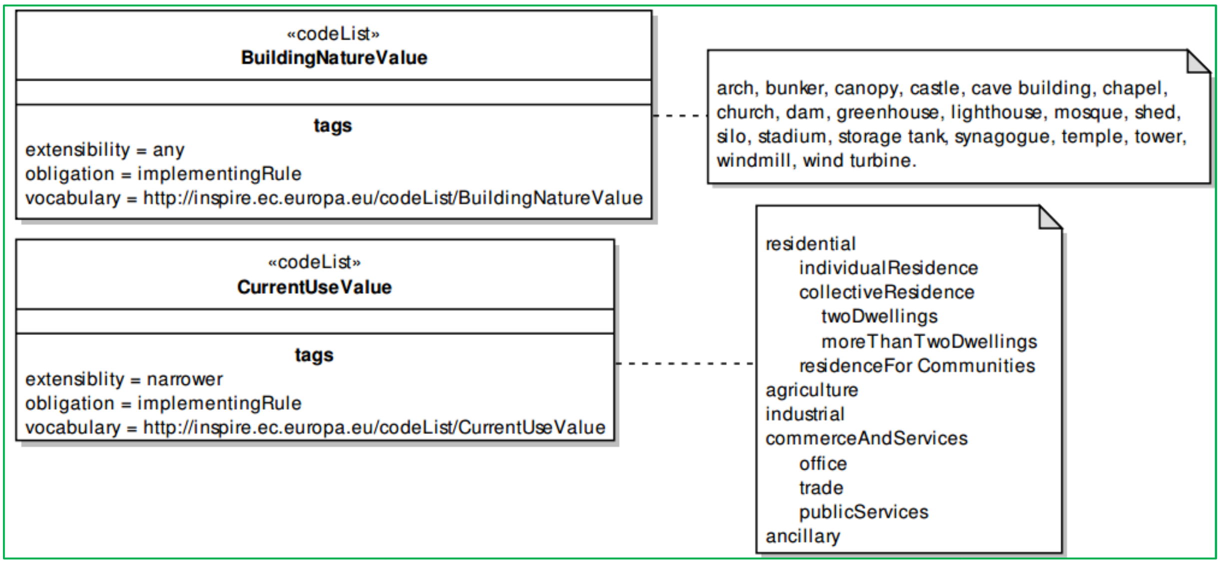

Figure 18. Code lists for classification of Buildings.

Attribute externalReference

This attribute aims to ensure the link to other information systems, for instance:

- another spatial data set including building data; in this case, the external reference contributes to ensure consistency between different views or different levels of detail on same real-world objects, that is an explicit requirement of the INSPIRE Directive.
- the cadastral register where information about owner, tenant, criteria of valuation (heating, toilet, …) may be found.

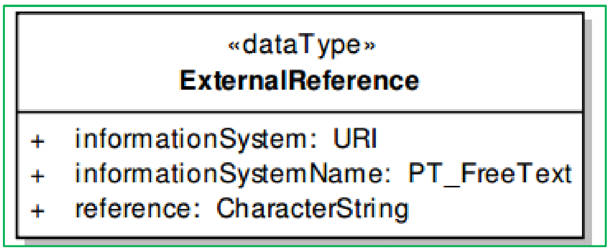

Figure 19. The attribute externalReference is defined as a data type.

Consistency between spatial data sets

There should be some consistency between the value of attribute currentUse in theme Buildings and the location of agricultural facilities, industrial or production facilities and governmental services.

Typically:

- Buildings within an agricultural or aquacultural facility should generally have value “agricultural” for attribute current use.
- Buildings within an industrial or production facility should generally have value “industrial” for attribute current use.
- Buildings within a governmental service should generally have value “public services” for attribute current use.

However, there may be exceptions (e.g. a residential building for guardian in a production site or for teacher in a school or for farmer in an agricultural facility); moreover, the geometry of facilities or governmental services may be represented in some cases just by a point and so, may not always enable to identified the related buildings. Consequently, no absolute consistency rule can be provided.

Ontology modelling

- **AbstractConstruction**.
  - inspireID. Data Property http://inspire.ec.europa.eu##inspireID.
  - name. http://inspire.ec.europa.eu#name relates AbstractConstruction and GeographicalName.
  - externalReference http://inspire.ec.europa.eu#externalReference relates AbstractConstruction with ExternalReference.
  - Alignment with CITI-SENSE ontology:
    - AbstractConstruction subclass of DUL:DesignedArtifact
- **AbstractBuilding** subclass of AbstractConstruction.
  - buildingNature. http://inspire.ec.europa.eu #buildingNature relates AbstractBuilding and codelist:buildingNature.
  - currentUse. http://inspire.ec.europa.eu#currentUse relates AbstractuBuilding and codelist:currentUse.
- **Building** subclass of AbstractBuilding.

- **BuildingPart** subclass of AbstractBuilding.
- **External Reference**
  - InformationSystem: Data property <http://inspire.ec.europa.eu#informationSystem>
  - InformationSystemName: Data property <http://inspire.ec.europa.eu#informationSystemName>
  - Reference: Data property <http://inspire.ec.europa.eu#reference>
- Ontology **http://inspire.ec.europa.eu/codelist**
  - BuildingNatureValue subclass of skos:Concept.
  - CurrentUse subclass of skos:Concept.

The formalised ontology is shown below:

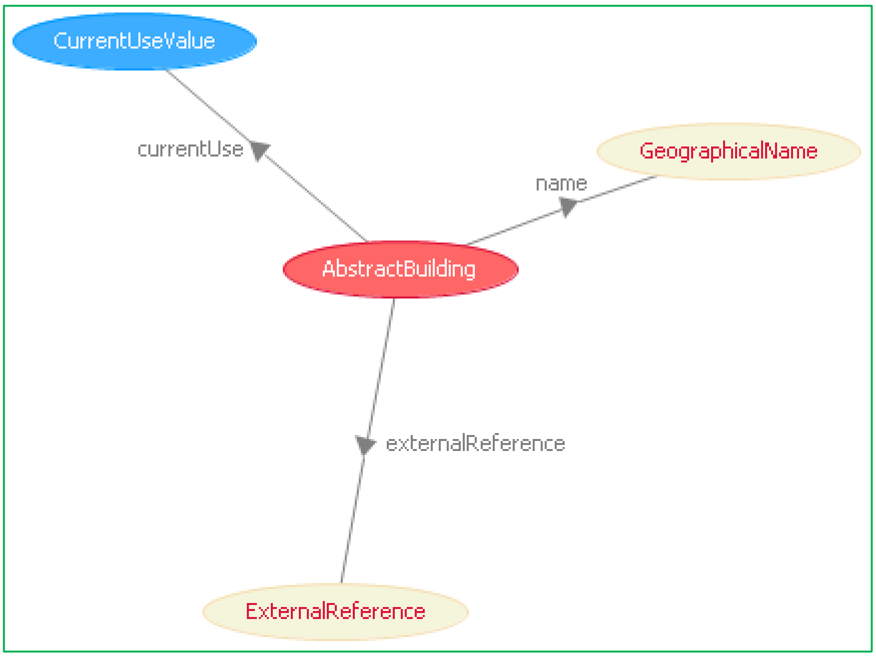

Figure 20. Implementation of AbstractBuilding (superclass of Building and BuildingPart, and subclass of AbstractConstruction)

5.2.4 INSPIRE Data Specification on Address

The INSPIRE Directive [8] defines the spatial data theme addresses as the: "Location of properties based on address identifiers, usually by road name, house number, postal code."

Addresses serve several generic purposes, including: location, identification, jurisdiction, sorting and ordering, and emergency response

The data specification defines an address as: "An identification of the fixed location of a property, e.g. plot of land, building, part of building, way of access or other construction, by means of a structured composition of geographic names and identifiers." Figure 21 shows the UML diagram of the Address application schema.

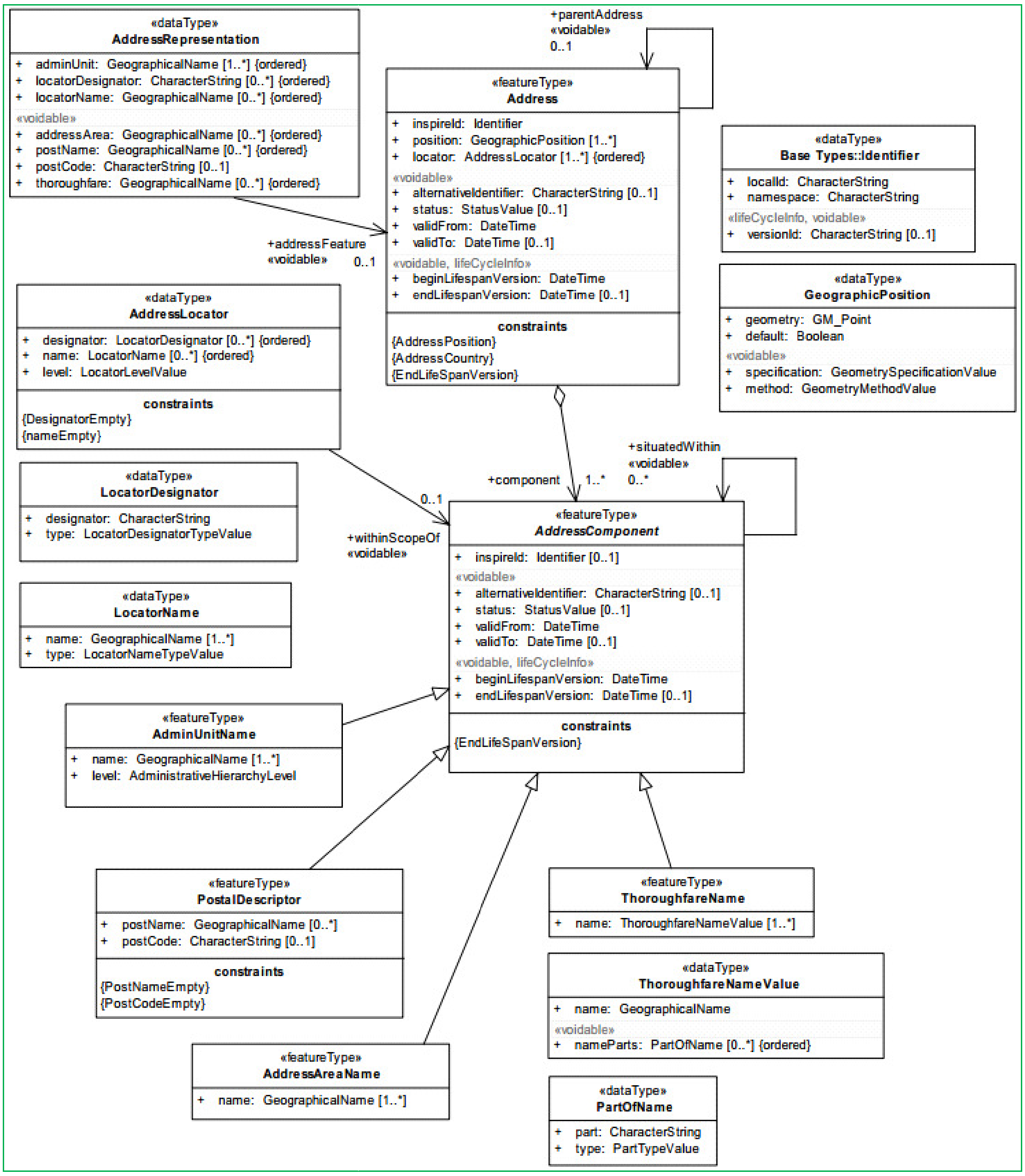

Figure 21. UML class diagram: Overview of the Addresses application schema [8]

Ontology modelling

- Address
  - inspireID. Data Property http://inspire.ec.europa.eu##inspireID
  - position: http://inspire.ec.europa.eu#position relates Address and GeographicPosition.
  - locator: http://inspire.ec.europa.eu#locator relates Address and AddressLocator.
  - parentAddress: <http://inspire.ec.europa.eu#parentAddress> relates Address with maximum one Address.
  - component: http://inspire.ec.europa.eu#component relates Address and AddressComponent. Min cardinality = 1.
- AddressComponet
  - inspireID: Data property <http://inspire.ec.europa.eu#inspireID>
- LocatorDesignator
  - designator: Data property http://inspire.ec.europa.eu#designator.
  - type: http://inspire.ec.europa.eu#type relates LocatorDesignator with LocatorDesignatorTypeValue.
- LocatorDesignatorTypeValue
  - Subclass of skos:Concept in http://inspire.ec.europa.eu/codelist ontology.
- GeographicPosition
  - geometry: http://inspire.ec.europa.eu#geometry relates GeographicPosition with http://www.opengis.net/ont/gml#Point
  - default: Data property http://inspire.ec.europa.eu#default
- AddressLocator
  - designator: http://inspire.ec.europa.eu#designator relates AddressLocator with LocatorDesignator.
  - name: http://inspire.ec.europa.eu#nameLocator relates AddressLocator with LocatorName.
  - level: http://inspire.ec.europa.eu#level relates AddressLocator with LocatorLevelValue
- AdminUnitName
  - Subclass of AddressComponent
  - name: http://inspire.ec.europa.eu#name relates AdminUnitName and GeographicalName.
  - level: http://inspire.ec.europa.eu#level relates AdminUnitName and AdministrativeHierarchyLevel.
- AdministrativeHierarchyLevel

    - This class has not been instantiated due its instances are the Levels of administration in the national administrative hierarchy. (see http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_AD_v3.0.1.pdf).

- AddressAreaName
    - Subclass of AddressComponent
    - Name http://inspire.ec.europa.eu#name relates AddressAreaName and GeographicalName.
- PostalDescriptor
    - Subclass of AddressComponent
    - postName: <http://inspire.ec.europa.eu#postName> relates PostalDescriptor and GeographicalName.
    - postCode: Data property <http://inspire.ec.europa.eu#postCode>
- LocatorLevelValue
    - Subclass of skos:Concept in http://inspire.ec.europa.eu/codelist ontology.
- LocatorName
    - name: http://inspire.ec.europa.eu#name relates LocatorName and GeographicalName.
    - type: http://inspire.ec.europa.eu#type relates LocatorName and LocatorNameTypeValue.
- LocatorNameTypeValue
    - Subclass of skos:Concept in http://inspire.ec.europa.eu/codelist ontology.

The formalised ontology is shown below:

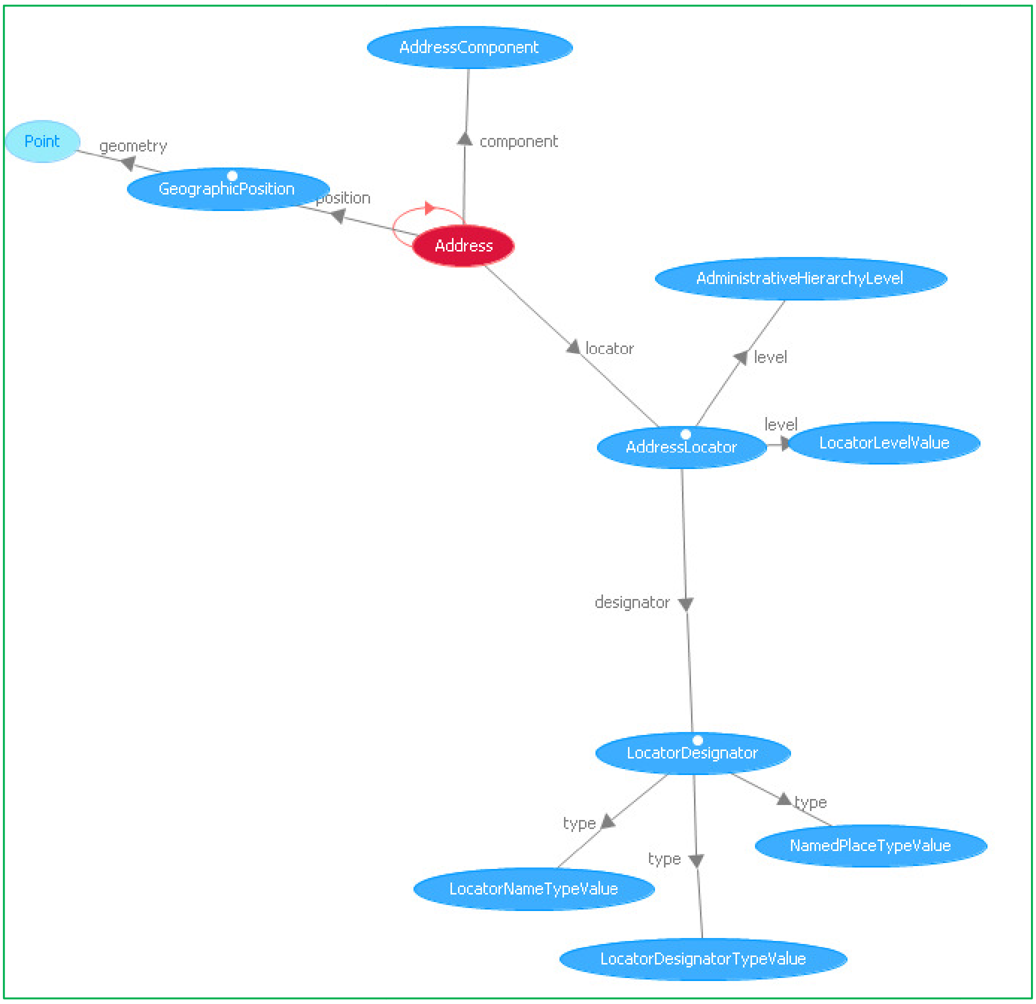

Figure 22. Implementation of INSPIRE Data Specification on Address

5.2.5 INSPIRE Data Specification on Land use

In the INSPIRE directive[10], Land Use is defined as **Territory** characterised according to its current and future planned functional dimension or socio-economic purpose (e.g. residential, industrial, commercial, agricultural, forestry, recreational). The inland water bodies as well as coastal waters are considered within the connected piece of land and planning of the use of sea and the use of seabed has been taken into consideration.

Land Use is itself split up into two different types:

- The **Existing Land Use** (current land use in the above definition), which objectively depicts the use and functions of a territory as it has been and effectively still is in real life. Geographical data-sets that provide Land Use information, at the time of observation, are modelled according to three application schemas:

- a. organized as a partition (in the mathematical sense) of a given area. Each element of the partition is homogeneous regarding the functional use of land. (ExistingLandUse application schema),
- b. organized as a set of discrete observation points informing on the functional use at the exact location and/or at its surrounding at the time of observation, (SampledExistingLandUse application schema),
- c. organized as a set of pixels informing on the functional use (GriddedExistingLandUse application schema)

- The **Planned Land Use** or PLU (future planned land use in the above definition), which corresponds to spatial plans, defined by spatial planning authorities, depicting the possible utilization of the land in the future. The scope of the INSPIRE Land Use Data Specification is giving the exact spatial dimension of all the elements a spatial plan is composed of. Planned Land Use application schema is mainly based on ZoningElement that depicts the zoning defined by spatial planners and SupplementaryRegulation that enables to inform on regulations that superimpose on the zoning.

The Land Use data specification supports two systems of classification:

- the (obligatory) Hierarchical INSPIRE Land Use Classification System (HILUCS) which is a multi-level, classification system that will apply to both the existing and planned land use
- the (optional) specific classification system in use in a member state.

For the purposes of CITI-SENSE project only the existing land use type modelled with the ExistingLandUse application schema will be considered while modelling the ontology. In CITI-SENSE the only usage of land is in pilot "Public Park" of WP3, where the necessity is to model the park at a certain time.

Figure 23 shows an UML overview of the Existing Land Use Application Schema, in which the ontology will be based.

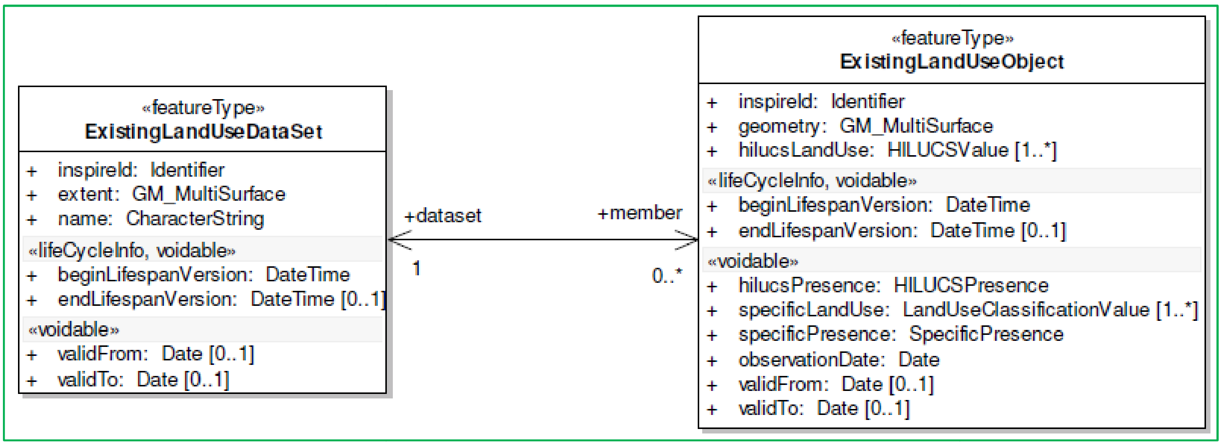

Figure 23. UML Overview of the Existing Land Use application schema

The Existing Land use application schema (see Figure 23) corresponds to a dataset that depicts the reality of the land use at a certain time. Usage of dataset depicting existing land use may require providing information on the same piece of land at different time. The application schema does not implement this requirement. It means that the existing land use on the same area at two different times will be provided as two different datasets.

The «featureType» ExistingLandUseDataset contains the ExistingLandUseObject that correspond to an area having a unique defined land use. An ExistingLandUseDataset may be created from several sources. Thus each ExistingLandUseObject may be associated with the time the documented land use corresponds to (attribute observationDate of «featureType» ExistingLandUseObject in Figure 23).

Ontology modelling

Only the ExistingLandUseObject is modelled in CITI-SENSE, because the requirements from the pilots only obly to model a public park, the rest of the cases are placed in buildings for different services.

- ExistingLandUseObject
  - inspireID. Data Property http://inspire.ec.europa.eu##inspireID
  - geometry. geosparql:hasGeometry relates ExistingLandUseObject with geosparql:Geometry
  - hilucsLandUse.: relates ExistingLandUseObject with http://inspire.ec.europa.eu/codelist#HILUCSLandUse. The code list is accessible at http://inspire.ec.europa.eu/codelist/HILUCSValue/
    - Alignment with CITI-SENSE ontology: ExistingLandUseObject subclass of <http://www.loa-cnr.it/ontologies/DUL.owl#PhysicalPlace>

The ontology formalised the Existing Land Use application schema as show below:

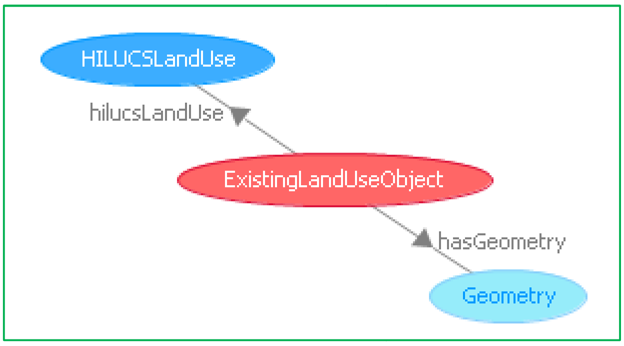

Figure 24. Implementation of INSPIRE Data Specification on Land Use

5.3 How to use the ontology?

For each of the necessities indicated in the pilots it is shown how they can be annotated with the city observatories ontology:

- Description of the sensors in each pilot:
  - A description of a sensor is created by defining an instance of the class ssn:SensingDevice. The relation ssn:observes allow indicating the physical quality (ucum:PhysicalQuality) that the sensor measures.
  - The sensors set in each pilot is an instance of ssn:Platform, where each sensor is an instance of ssn:SensingDevice. The relation between ssn:SensingDevice and ssn:Platform is ssn:onPlatform, the inverse relation is ssn:AttachedSystem.
- The observations made by every sensor:
  - Ssn:SensingDevice ssn:observes muo:PhysicalQuality, depending on the concrete SensingDevice the corresponding instance of muo:PhysicalQuality should be selected. The following instances have been added to muo:PhysicalQuality
    - http://www.citi-sense.eu/2013/cityobservatories#humidity
    - http://www.citi-sense.eu/2013/cityobservatories#noise
    - http://www.citi-sense.eu/2013/cityobservatories#No
    - http://www.citi-sense.eu/2013/cityobservatories#NO2
    - http://www.citi-sense.eu/2013/cityobservatories#CO
    - http://www.citi-sense.eu/2013/cityobservatories#03
    - http://www.citi-sense.eu/2013/cityobservatories#PM
    - http://www.citi-sense.eu/2013/cityobservatories#PAH
    - http://www.citi-sense.eu/2013/cityobservatories#elevation
    - http://www.citi-sense.eu/2013/cityobservatories#radiantTemperature
    - http://www.citi-sense.eu/2013/cityobservatories#windSpeed
    - http://www.citi-sense.eu/2013/cityobservatories#airTemperature
    - http://www.citi-sense.eu/2013/cityobservatories#relativeHumidity
    - http://www.citi-sense.eu/2013/cityobservatories#heatIndex
    - http://www.citi-sense.eu/2013/cityobservatories#windChill
    - http://www.citi-sense.eu/2013/cityobservatories#WBGT
    - http://www.citi-sense.eu/2013/cityobservatories#PET
    - http://www.citi-sense.eu/2013/cityobservatories#LAeq
    - http://www.citi-sense.eu/2013/cityobservatories#L90
    - http://www.citi-sense.eu/2013/cityobservatories#LAmax
    - http://www.citi-sense.eu/2013/cityobservatories#LAmin

      - http://www.citi-sense.eu/2013/cityobservatories#dust
      - http://www.citi-sense.eu/2013/cityobservatories#VoC
      - http://www.citi-sense.eu/2013/cityobservatories#Random
      - http://www.citi-sense.eu/2013/cityobservatories#photo
      - http://www.citi-sense.eu/2013/cityobservatories#sound
      - http://www.citi-sense.eu/2013/cityobservatories#soundEvent
  - The different units of measure must be instances of muo:UnitsOfMeasurement. Each instance is related to the concrete instance of muo:PhysicalQuality by property muo:MeasuresQuality.
  - The measured value is related to dul:Amount, by the data property dul:hasDataValue.
  - The unit of measure of the measured value is related to the instance of dul:Amount by the property isClassifiedBy that relates it to the exact unit of measure.
  - The representation of the instant or interval during which the value dul:Amount was observed is represented by the property dul:isObservableAt, that relates dul:Amount with time:Instant or time:Interval.
- Description of schools and public parks:
  - A description of a school is a inspire:GovernmentalService instance with a relation with the instance inspire/codelist:education of ServiceTypeValue and a related service location type to be selected between address, building or geometry.
  - A description of a public park is an instance of inspire:ExistingLandUseObject, indicating the HILUCS Land Use "3_4_4_OpenAirRecreationalAreas" and its situation.
  - inspire:GovernmentalService subclass of DUL:Organization and dul:PhysicalPlace and inspire:ExistingLandUseObject subclass of dul:PhysicalPlace
  - To locate a Platform is a physical place: ssn:Platform relates with dul:PhysicalPlace (so with inspire:GovernmentalService and inspire:ExistingLandUseObject) by dul:hasLocation property.
  - To locate the school and the park, inspire:GovernmentalService has relation inspire:serviceLocationByGeometry with geospaqrl:Geometry. inspire:ExisitngLandUseObject has relation inspire:Geometry with geosparql:Geometry.
      - In case GPS coordinates are used:
          - geosparql:Point is a subclass of inspire:Geometry and of wgs84:Point
          - inspire:GeographicPosition has relation inspire:geometry with geosparql:Point
      - In case a concrete address is used:
          - inspire:Address has relation inspire:position with inspire:GeographicPosition
      - In a case a place identified in GeoNames

    - geonames:GeonamesFeature is subclass of dul:PhysicalPlace, ssn:Platform has a relation dul:hasLocation with dul:PhysicalPlace.
    - geonames:GeonamesFeature is related with inspire:GeographicalName through inspire:NamedPlace, which is super-class of geonames:GeonamesFeature.
  - To define the park, inspire:ExisitngLandUseObject has relation inspire:hilucsLandUse with the instance inspire/codelist:3_4_4_OpenAirRecreationalAreas of inspire:HILUSLandUse in order to indicate the specific land use.
- Attachment of a person with the specific pilot
  - A person is an instance of foaf:Person.
  - foaf:Person relates with dul:Event through dul:isAgentInvolvedIn.
  - dul:Event is related with ss:Platform via dul:hasParticipant.

Figure 25 shows how the ontology can be used to implement the CITI-SENSE pilots:

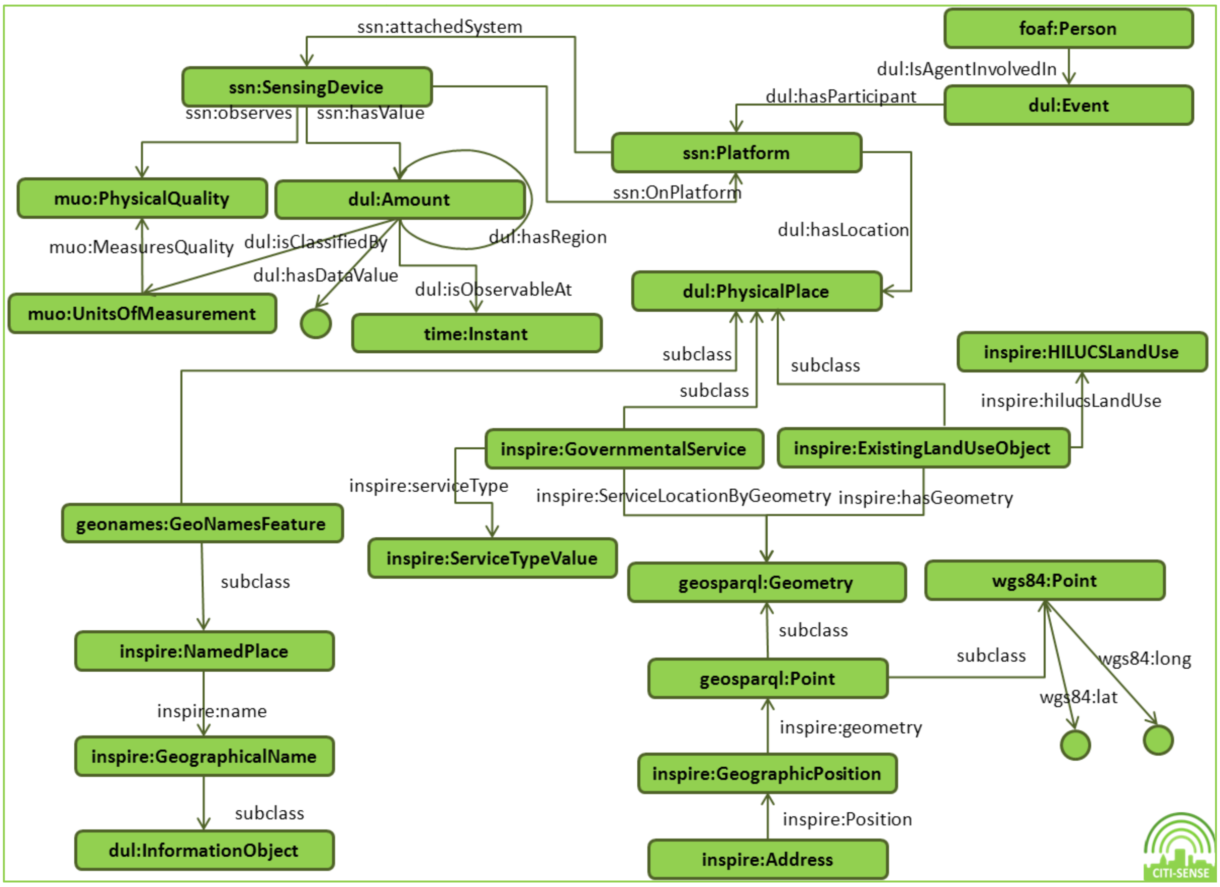

Figure 25. City observatories Ontology use cases implementation

5.4 Ontology requirements validation

Table 4 compiles the requirements, from chapter 3, and how the ontology satisfies each of them. In some cases the requirements are still undefined, in such cases it is indicated the future steps to be taken.

Table 4. Ontology requirement validation

ID	Requirement	Ontology aspect
R1	Collect data from static outdoor sensors: NO, NO2, CO, O3,PM, Noise, Temperature, Humidity, PAH	NO, NO2, CO, O3,PM, Noise, Temperature, Humidity, PAH are instances of muo:PhysicalQuality
R2	Collect data from personal sensors (indoor/outdoor): NO, CO, O3, Temperature, Humidity	NO, CO, O3, Temperature, Humidity are instances of muo:PhysicalQuality
R3	Collect data from SmartPhone Data: GPS, Accelerometer, elevation	GPS, Accelerometer, elevation are instances of muo:PhysicalQuality
R4	Collect data from perception data: surveys UNDER DEVELOPMENT	Future elaboration on foaf vocabulary, once the surveys are defined
R5	Collect data from air pollution monitoring and meteorological stations	ssn:Platform will be the class from where air pollution monitoring and meteorological stations will be instantiated.
R6	Collect Forecasting data on air pollution and meteorological data	Future links with other existing datasets in the Linked Open Data cloud.
R7	Collect data from User profile UNDER DEVELOPMENT	Future elaboration on foaf vocabulary once the user profile is defined.
R8	Collect 360-degree photoscape (video or photo)	Instance “photo” added to muo:PhysicalQuality. The photo is related to an instance of dul:Amount through the data property dul:hasAmount. The representation of the instant when the photo was taken is represented by the property dul:isObservableAt, that relates dul:Amount with time:Instant
R9	Collect data from sensors (measures each minute): mean radiant temperature (Tmrt), wind speed, air temperature and relative humidity.	The measured value is related to dul:Amount, by the data property dul:hasDataValue. The representation of the instant or interval during which the value dul:Amount was observed is represented by the property dul:isObservableAt, that relates dul:Amount with time:Instant or time:Interval
R10	Collect data from calculated value (for each measuring period): Heat index, Wind chill, Outdoor Wet Bulb Globe Temperature (WBGT), PET (Physiological Environmental Temperature)	The measured value is related to dul:Amount, by the data property dul:hasDataValue. The representation of the instant or interval during which the value dul:Amount was observed is represented by the property dul:isObservableAt, that relates dul:Amount with time:Instant or time:Interval
R11	Collect data from sound measurements (MP3) for each	Instance “sound” added to muo:PhysicalQuality. The measured sound is related to dul:Amount, by the data property

ID	Requirement	Ontology aspect
	measuring period	dul:hasDataValue The representation of the interval during which the value sound was recorded is represented by the property dul:isObservableAt, that relates dul:Amount with time:Interval.
R12	Collect data from calculated value (each minute): LAeq, L90 LAmax and LAmin	The measured value is an instance of dul:Amount, that has the data property dul:hasDataValue. The representation of the instant or interval during which the value dul:Amount was observed is represented by the property dul:isObservableAt, that relates dul:Amount with time:Instant or time:Interval
R13	Collect data from sound events: in MP3 recorded in R11, events: moment in the recording along with a label identifying the event.	Instance "sound" added to muo:PhysicalQuality. The measured sound file is related to dul:Amount by the data property dul:hasDataValue. The representation of the interval during which the value sound was recorded is represented by the property dul:isObservableAt, that relates dul:Amount with time:Interval. Instance "soundEvent" added to muo:PhysicalQuality. Each event label is related to dul:Amount by the data property dul:hasDataValue. The representation of the instant during which the value sound was recorded is represented by the property dul:isObservableAt, that relates dul:Amount with time:instant. The recorded sound file is related to its events by the dul:hasRegion property.
R14	Collect data from urban Landscape perception (answer to questionnaires about Local landscape perception, Participation and perception measurements and other data and photographs). UNDER DEVELOPMENT	Future elaboration on foaf vocabulary when the questionnaires be defined.
R15	Collect data from UV Exposure UNDER DEVELOPMENT	Once the parameters that define the UV exposure are defined, new instances should be added to muo:PhysicalQuality class.
R16	Collect data from sensor data: Temperature, Relative humidity, CO2, NO2, Dust, Noise, VOC, Radon.	Temperature, Relative humidity, CO2, NO2, Dust, Noise, VOC, Radon are instances of muo:PhysicalQuality
R17	Collect Location, School, operation hours, Time period	inspire:GovernmentalService class to describe school and its location.
R18	Compliant with INSPIRE directive	INSPIRE ontology developed, covering Geographical Names, Utility and Governmental Services, Buildings, Addresses and Land Use. UCUM used for units of measure. The Location Core Vocabulary is used as a complement for Addresses specification.
R19	Compliant with SenML standard	SSN ontology supports SenML standard.
R20	Compliant with GEOSS standards (SKOS; SPARQL)	The OWL ontology supports SPARLQ for querying.

6. Conclusions

The city observatories ontology provides the vocabulary to annotate the available resources at each pilot. This includes the description of the location where the observations are performed as well as the person performing the observations. In addition, it provides a way of describing the different sensors and their measurements as well as the calculated data values from sensor data.

The ontology has been implemented following the network ontology approach, trying to reuse where possible existing efforts in the ontology, such as linked data, sensor and IoT worlds. One of the requirements was to be compliant with the INSPIRE Directive. From this directive, the data specifications, relevant to CITI-SENSE, have been implemented: Geographical Names, Utility and Governmental Services, Buildings, Addresses and Land Use. These data specifications have not been completely implemented. In some cases only relevant parts were used. The ontology includes UCUM and the Core Location Vocabulary. UCUM is used by all INSPIRE themes to manage units of measure in a standard way.

The city observatories ontology is ready to annotate datasets. The city observatories ontology is ready to annotate datasets with sensor observations, calculated data, people in charge of the observations, location and time in order to be published under the Linked Data paradigm[125] in the Linked Open Data Cloud[126]. Thanks to the reuse of well-known and widely used ontologies such as the SSN[127] and GeoNames[128], the links to existing published datasets as GeoNames[129] and Linked Sensor Data[130] are possible without modifications of the ontology.

The city observatories ontology is ready to provide different ways to specify a location: indicating its address, through an element from GeoNames database, via GPS coordinates or any geometry as defined by the GML standard[131].

The city observatories ontology is open for future evolution where new sensors types, measure types or unit of measures are needed, without modification of its core structure, only adding new instances.

The next step should be the inclusion in the CITI-SENSE architecture of a module able to annotate with the city observatories ontology, the data served by the different pilots. And to establish links from entities in CITI-SENSE dataset to other entities in already published datasets.

Future evolution of the ontology in the short-term is focused in the experience after its use in CITI-SENSE and once the incomplete requirements be defined. In medium/long term, the evolution is focused on the implementation of more data specifications from INSPIRE Directive.

[125] http://www.w3.org/standards/semanticweb/data
[126] http://linkeddata.org/
[127] http://www.w3.org/2005/Incubator/ssn/ssnx/ssn
[128] http://www.geonames.org/ontology/documentation.html
[129] http://datahub.io/dataset/geonames
[130] http://datahub.io/dataset/knoesis-linked-sensor-data
[131] http://www.opengeospatial.org/standards/gml

References

Table 5. References

	References
01	Computer Science Corporation, CSC CatalystSM Ontology White Paper, V2011-07-15
02	B. Inmon. (2007) The semantics mystery. BeyeNETWORK, Article no. 4605, http://www.b-eye-network.com/print/4605, Jun 2007.
03	Oberle, Daniel (2009) How Ontologies Benefit Enterprise Applications
04	Open Geospatial Consortium (2011). OGC® SWE Common Data Model Encoding Standard v2.0.0. 4 January 2011
05	D. Heckmann et al. (2005) GUMO - the General User Model Ontology. In The 10th International Conference on User Modeling, pages 428–432, Edinburgh, UK, 2005.Springer, Berlin Heidelberg
06	Sheth, A., Henson, C., Sahoo, S (2008): Semantic Sensor Web. IEEE Internet Computing 78-83.
07	ISA European Commision, (2012) Core Vocabularies specification
08	D2.8.I.5 INSPIRE Data Specification on Addresses – Guidelines. Infrastructure for Spatial Information in Europe (INSPIRE). http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_AD_v3.0.1.pdf
09	D2.8.III.2 Data Specification on Buildings – Draft Technical Guidelines (INSPIRE) http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_BU_v3.0rc3.pdf
10	D2.8.III.4 Data Specification on Land use – Draft Technical Guidelines http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_LU_v3.0rc3.pdf
11	D2.9 Guidelines for the use of Observations & Measurements and Sensor Web Enablement-related standards in INSPIRE Annex II and III data specification development: http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/D2.9_O&M_Guidelines_V1.0.pdf
12	D2.8.III.13-14 Data Specification on Atmospheric Conditions Meteorological geographical features – Draft Technical Guidelines: http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_AC-MF_v3.0rc3.pdf
13	D2.8.III.7 Data Specification on Environmental monitoring Facilities – Draft Technical Guidelines: http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_EF_v3.0rc3.pdf
14	INSPIRE Metadata Implementing Rules: Technical Guidelines based on EN ISO 19115 and EN ISO 19119: http://inspire.jrc.ec.europa.eu/documents/Metadata/INSPIRE_MD_IR_and_ISO_v1_2_20100616.pdf
15	INSPIRE spatial data services and services allowing spatial data services to be invoked– Draft Implementing Rules: http://inspire.jrc.ec.europa.eu/documents/Spatial_Data_Services/Draft_IR_SDS_and_Invoke_3.0.pdf
16	OGC GeoSPARQL A Geographic Query Language for RDFhttp://www.opengis.net/doc/IS/geosparql/1.0
17	Compton et al. (2012) The SSN Ontology of the W3C Semantic Sensor Network Incubator Group
18	Battle, R., Kolas D.(2011) Enabling the Geospatial Semantic Web with Parliament and GeoSPARQL. Semantic Web Journal IOS Press.
19	Euzenat, J., Shvaiko, P(2007): Ontology matching. Springer-Verlag
20	Linehana M.H. et al. (2012) The Date-Time Vocabulary.
21	Wang, X.H.et al (2004) Ontology Based Context Modeling and Reasoning using OWL. Proc. 2nd IEEE Conf. Pervasive Computing and Communications, Workshop on Context Modeling and Reasoning, IEEE CS Press, 2004.
22	Heckmann D. et al (2007) The User Model and Context Ontology GUMO revisited for future Web 2.0

References

Extensions. C&O:RR, volume 298 of CEUR Workshop Proceedings, CEUR-WS.org, (2007)

23 Steffen Schulze-Kremer(2001) Ontologies for molecular biology and bioinformatics

24 Gruber T.(2009) Ontology. Encyclopaedia of Database Systems, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag, 2009.

25 IERC AC4 (2013) IoT Semantic Interoperability: Research Challenges, Best Practices, Solutions and Next Steps. IERC AC4 Manifesto 2012-2014

26 Colas et al(2013) Core Location Pilot. Interconnecting Belgian National and Regional Address Data

27 OGC® Sensor Observation Service Interface Standard (Approval data: 2012-04-16) http://www.opengis.net/doc/IS/SOS/2.0

ANNEX I ONTOLOGY SOURCES

The following table indicates different search engines that look for links to existing ontologies:

Table 6. Search engines looking for links to existing ontologies

URL	Category	Comments
http://swoogle.umbc.edu/	search engine	also REST API
http://swse.deri.org/	search engine	NOT ONLY FOR ONTOLOGIES
http://kmi-web05.open.ac.uk/WatsonWUI/	search engine	also SOAP and REST API
http://watson.kmi.open.ac.uk/	search engine	
Falcons http://ws.nju.edu.cn/falcons/objectsearch/index.jsp	search engine	a new Chinese engine; separate searches for objects, ontology, documents
TONES Repository: http://owl.cs.manchester.ac.uk/repository/browser	list with many columns	supports a RESTful interface; 232 ontologies
ORATE (Ontology Repository for Assistive Technologies) http://babel.informatik.uni-bremen.de/search/	Repository	55 ontologies, but some are just copies from their main site; also based on Bioportal; Ruby On Rails server.
Protege Ontologies Library: http://protege.cim3.net/cgi-bin/wiki.pl?ProtegeOntologiesLibrary protégé Ontology Ļibrary http://protegewiki.stanford.edu/wiki/Protege_Ontology_Library		
http://semanticweb.org/wiki/Ontology	hit-parade	
schemapedia.com http://schemapedia.com/	search engine	currently 277 schema referenced
vocab.deri.ie: http://vocab.deri.ie/		Vocabularies from DERI Sem. Web research center
Most-common-RDF-namespaces as of 2006: http://ebiquity.umbc.edu/resource/html/id/196/Most-	hit-parade	

common-RDF-namespaces		
http://olp.dfki.de/ontoselect		
http://www.schemaweb.info/	tree	
http://bioportal.bioontology.org/		
http://prefix.cc	Repository	nice web service to find ontology URI (and hence its definition) from the N3 (consensus) prefix, and vice-versa; used in EulerGUI
http://esw.w3.org/topic/SkosDev/DataZone		latest SKOS vocabularies (knowledge organization systems (KOS) such as thesauri, classification schemes, subject heading lists and taxonomies)

Popular vocabularies can be found on http://semanticweb.org/, and following is listed the popular ontologies:

- http://eulersharp.svn.sourceforge.net/viewvc/eulersharp/trunk/2003/03swap/index.html#ontologies
- FOAF - Friend Of A Friend ; OWL
- DAOP - Description Of A Project; OWL
- SIOC - Semantically-Interlinked Online Communities; OWL
- SKOS - from W3C, SKOS Simple Knowledge Organization System
- Dublin Core _ RDFS or OWL ; dcterms : Dublin Core terms - RDFS ; dcmitype ; RDFS Review, vCard (this is an example of recent smart advice from W3C, by which the same URL conveys through content negotiation either OWL or HTML).
- dbPedia : for OWL download: "DBpedia Ontology T-BOX (Schema) "
- SUMO (with MILO, etc), Dolce, BFO, UMBEL, Bremen, etc, are so-called upper level ontologies; A Comparison of Upper Ontologies .
- OpenCYC is a huge ontology (Open Source), coming with its own closed-source reasoner, but I don't know how to extract domain specific sub-ontologies from Cyc.
- Princeton Wordnet 3.0 in RDF : the famous and wonderful English dictionary is a way to associate an URI to word senses, thus it is a natural way for a priori understanding between applications
- http://owl.cs.manchester.ac.uk/repository/

- http://babel.informatik.uni-bremen.de/
- http://eulergui.svn.sourceforge.net/viewvc/eulergui/trunk/eulergui/html/documentation.html#Finding
- Device related ontologies: https://marinemetadata.org/community/teams/ontdevices/ontdevrel
- More ontologies (some related with sensors):
  - https://marinemetadata.org/conventions/ontologies-thesauri
  - https://marinemetadata.org/taxonomy/term/320?sort=desc&order=Modified
  - Delegation Ontology: An OWL ontology to describe delegation concepts in the context of Grid computing. http://www.nada.kth.se/~mehrana/Delegation.owl
  - Dependable Systems Ontology: Ontology about resilient and dependable systems including threats, failures, faults and errors as used in the ReSIST project. http://www.ecs.soton.ac.uk/~aoj04r/resist.owl
  - FOAF Ontology: An ontology describes people, the links between them and the things they create and do. Contributed by Dan Brickley and Libby Miller.
  - Geographic Information Metadata - ISO 19115: An ontology representing Geographic Information Metadata - (ISO 19115). http://loki.cae.drexel.edu/~wbs/ontology/
  - office-env1.owl,office-env2.owl: Ontologies for office environment. Contributed by Prakash Kadel
  - http://ise.icu.ac.kr/Ontologies/office-env1.owl
  - http://ise.icu.ac.kr/Ontologies/office-env2.owl
  - shuttle-crew-ont.owl: An ontology about the crew from a space shuttle. Source: Dynamic Research Corporation. http://protege.cim3.net/file/pub/ontologies/shuttle/shuttle-crew-ont.owl
  - Trade.owl: An OWL ontology specifically focused on the commercial management of purchase orders, in a company primarily dedicated to trade in electric, energy, and evironmental products. Contributed by Prof. Dr. Antonio Paredes-Moreno. http://protegewiki.stanford.edu/wiki/File:TradeOWLOntology.zip
  - travel.owl: A tutorial OWL ontology for a Semantic Web of tourism. Contributed by Holger Knublauch. http://protege.cim3.net/file/pub/ontologies/travel/travel.owl
- Patterns :http://ontologydesignpatterns.org/wiki/Community:ListPatterns
  - AgentRole: To represent agents and the roles they play. http://www.ontologydesignpatterns.org/cp/owl/agentrole.owl
  - Classification: To represent the relations between concepts (roles, task, parameters) and entities (person, events, values), which concepts can be assigned to. To formalize the application (e.g. tagging) of informal knowledge organization systems such as lexica, thesauri, subject directories, folksonomies, etc., where concepts are first-order elements. http://ontologydesignpatterns.org/wiki/Submissions:Classification http://www.ontologydesignpatterns.org/cp/owl/classification.owl

- ParticipantRole: to represent participants in events holding specific roles in that particular event: http://ontologydesignpatterns.org/wiki/Submissions:ParticipantRole
- Action: The purpose of the pattern is to model actions that are proposed, planned, and performed or abandoned, together with their status and durations in time. http://ontologydesignpatterns.org/wiki/Submissions:Action http://www.ontology.se/odp/content/owl/Action.owl
- Time Interval
- Task Execution: To represent actions through which tasks are executed http://www.ontologydesignpatterns.org/cp/owl/taskexecution.owl
- Sequence: To represent sequence schemas. It defines the notion of transitive and intransitive precedence and their inverses. Part of DUL.owl http://ontologydesignpatterns.org/cp/owl/sequence.owl
- RoleTask: To represent the assignment of tasks to roles http://ontologydesignpatterns.org/wiki/Submissions:Role_task; http://www.ontologydesignpatterns.org/cp/owl/taskrole.owl

ANNEX II DATASETS

Datasets Template

In order to collect the datasets CITI-SENSE will generate and use, a template has been setup for collecting information about the datasets and it is outlined below. In Annex II are two examples of datasets that have been collected so far. The list will be extended in future versions of this document.

1. Dataset name

Give a name to the dataset. A dataset is a collection of data which taken as a whole offers some useful information to a potential user. An example could be "Air quality measurements in Norwegian cities".

2. Maintainer organization and contact person

Who is the maintainer of the dataset and/or the person who can be contacted for further details about the dataset?

3. Dataset short description

Brief description about the dataset:

- what is the data about;
- who are the potential users of the data and the stakeholders;
- what is currently the data used for;
- how do you envision using the data?

4. Theme / tags

Categorise the dataset and provide some relevant keywords/tags. For example, air quality, measurements, Norwegian cities, etc.

5. Data collection

How is the data in the dataset being collected? For example, automatically via e.g. real-time sensors, manually collected, manually composed and curated from other datasets, etc.

6. Data storage and access

How is the data currently being stored? For example, in files, relational databases, distributed databases, etc.

How is the data currently being accessed? E.g. via a Web portal, Web APIs, Web services, database queries, standalone desktop tools, etc.

7. Data structures

How is the data currently structured? For example if the data is in files, is the data structured in formats like CSV, excel, pdf, relational tables, graphs, shapefiles, GML, NetCDF, coverage, etc. If the data is accessed via a Web service is the data structured in OGC WMS, WCF, WFS, etc.

8. Metadata and provenance

Is there any metadata attached to the dataset? If yes, against which specification (e.g. any specific metadata standard)? Are there any provenance information attached to the dataset (e.g. who created it, when, how, etc.)?

9. Quantity/Size

How large is the dataset? How many Mb, Gb, Tb, etc.

10. Licencing

Are there any licencing issues with the datasets? If so, describe them. For example, is the data restricted for use only for some users, freely available, etc. Do you have plans to monetize the data?

11. Update frequency

How often is the dataset updated (e.g. real-time updates, seconds, minutes, hours, daily, weekly, monthly, annually)?

12. Geographical coverage

If the dataset has a geospatial aspect, which areas, regions, countries, etc, does the data cover?

13. Quality

Provide some information about the quality of the dataset. For example, is the dataset the authoritative dataset in a certain area (country, city, region, etc.), is the dataset conformant with any directives (e.g. INSPIRE), is the dataset outdated, not maintained, is the granularity of the data good enough, is there any available documentation, usage instructions, is the data accurate, does it contain any uncertainty information, etc.

14. Dependencies

Is the dataset dependent on other datasets? If yes, which datasets? For example if one measures the air quality in certain locations, are the locations taken from 3rd party data sets (e.g. mapping agencies datasets with location in a country)?

15. Link and additional documentation

Provide a link to the location of the dataset if available (e.g. WFS end-point), or link to websites, documents, etc., with further information about the dataset.

16. Other information

If there is relevant information about the dataset not covered in the above items, please provide details here. For example if the dataset is in languages other than English, any data privacy issues, etc.

Data measurements on European air quality and background concentrations

1. **Dataset name**

Give a name to the dataset. A dataset is a collection of data which taken as a whole offers some useful information to a potential user. An example could be "Air quality measurements in Norwegian cities". Data measurements on European air quality and background concentrations.

2. **Maintainer organization and contact person**

Who is the maintainer of the dataset and/or the person who can be contacted for further details about the dataset?

European Environmental Agency (EEA): http://www.eea.europa.eu/about-us/who/staff-list

European Monitoring and Evaluation Program (EMEP): http://www.nilu.no/projects/ccc/about/

http://www.nilu.no/projects/ccc/about/

3. Dataset short description

Brief description about the dataset:

a. what is the data about;

The data contains air pollution measurements, data from background, traffic, industrial stations, and background concentrations taken from static and portable micro sensors.

b. who are the potential users of the data and the stakeholders;

Stakeholders are citizens, citizen's groups, policy makers, and the Global Monitoring for Environment and Security Initiative (GMES).

c. what is currently the data used for;

Data is currently used to provide environmental data products on air pollution and meteorological conditions to the various stakeholders.

d. how do you envision using the data?

Collect raw data, transform to RDF, link to other existing related data, and visualize reports and graphics

4. Theme / tags

Categorize the dataset and provide some relevant keywords/tags. For example, air quality, measurements, Norwegian cities, etc.

The European Environmental Agency (EEA), the European Monitoring and Evaluation Program (EMEP), air quality, background concentrations, Airbase database, EBAS database, urban, rural areas.

5. Data collection

How is the data in the dataset being collected? For example, automatically via e.g. real-time sensors, manually collected, manually composed and curated from other datasets, etc.

Airbase database (used by EEA) takes data from a total number of 7965 stations, of which 4280 are classified as urban stations, 2208 as suburban stations, and 1339 as rural stations. 138 stations are unclassified in the metadata. EMEP uses EBAS database. There are 321 stations in EMEP.

Static and portable micro sensors which collect environmental data are used. The data collected from such sensors is used in combination with existing data sources using data assimilation techniques, in order to provide a value-added product to citizens.

6. Data storage and access

How is the data currently being stored? For example, in files, relational databases, distributed databases, etc.

There are two databases:

- Airbase: used by the EEA
- EBAS: used by EMEP

How is the data currently being accessed? E.g. via a Web portal, Web APIs, Web services, database queries, standalone desktop tools, etc.

Data can be accessed via a web portal. The entire Airbase database, raw data are provided in ASCII format and all metadata are available and can be downloaded from the website: http://www.eea.europa.eu/data-and-maps/data/airbase-the-european-air-quality-database-6

EMEP data can be accessed through the EBAS website operated by NILU: http://ebas.nilu.no

After filtering, the number of datasets matching the criteria is shown in the lower right hand of the screen and can be displayed by clicking the "List datasets" button. Subsequently, the user can select the dataset of interest and download it formatted in the ASCII-based NASA Ames Format for Data Exchange.

7. Data structures

How is the data currently structured? For example if the data is in files, is the data structured in formats like CSV, excel, pdf, relational tables, graphs, shape files, GML, NetCDF, coverage, etc. If the data is accessed via a Web service is the data structured in OGC WMS, WCF, WFS, etc.

Data is structured in CSV and NASA-AMES files.

8. Metadata and provenance

Is there any metadata attached to the dataset? If yes, against which specification (e.g. any specific metadata standard)? Are there any provenance information attached to the dataset (e.g. who created it, when, how, etc.)?

Metadata about existing European air quality monitoring networks and similar data sources are available at: http://www.eea.europa.eu/data-and-maps/data/airbase-the-european-air-quality-database-6#tab-metadata

The primary focus for the detailed description is on two main air quality monitoring networks in Europe, namely the Airbase database operated by the European Environmental Agency, and the European Monitoring and Evaluation Program network of air quality stations.

Additional metadata are provided for other datasets, in particular those provided by related EU projects.

9. Quantity/Size

How large is the dataset? How many Mb, Gb, Tb, etc.

- For EEA:
    - AirBase measurement configurations 643.16 KB
    - AirBase stations 303.47 KB
    - AirBase statistics 21.29 MB

10. Licensing

Are there any licensing issues with the datasets? If so, describe them. For example, is the data restricted for use only for some users, freely available, etc. Do you have plans to monetize the data?

EMEP:

Information about data policy of EMEP can be found here: http://ebas-submit.nilu.no/DataPolicy/tabid/10309/Default.aspx

EEA:

EEA standard re-use policy: unless otherwise indicated, re-use of content on the EEA website for commercial or non-commercial purposes is permitted free of charge, provided that the source is acknowledged (http://www.eea.europa.eu/legal/copyright). Copyright holder: Directorate-General for Environment (DG ENV), United Nations Framework Convention on Climate Change (UNFCCC).

11. Update frequency

How often is the dataset updated (e.g. real-time updates, seconds, minutes, hours, daily, weekly, monthly, annually)?

In EEA stations the data is updated every hour, while in the EMEP the data is updated daily.

12. Geographical coverage

If the dataset has a geospatial aspect, which areas, regions, countries, etc, does the data cover?

EEA covers Europe and some European territories worldwide and EMEP covers Europe.

13. Quality

Provide some information about the quality of the dataset. For example, is the dataset the authorities dataset in a certain area (country, city, region, etc), is the dataset conformant with any directives (e.g. INSPIRE), is the dataset outdated, not maintained, is the granularity of the data good enough, is there any available documentation, usage instructions, is the data accurate, does it contain any uncertainty information, etc?

The data is maintained. Additional information about the data can be found on the following websites:

http://www.nilu.no/projects/ccc/qa/index.htm

http://www.eea.europa.eu/data-and-maps/data/airbase-the-european-air-quality-database-6

The datasets are conformant with the INSPIRE Directive.

Detailed information about the INSPIRE Directive can be found at the INSPIRE website at

http://inspire.jrc.ec.europa.eu/.

14. Dependencies

Is the dataset dependent on other datasets? If yes, which datasets? For example if one measures the air quality in certain locations, are the locations taken from 3rd party data sets (e.g. mapping agencies datasets with location in a country)?

For data assimilation tasks, two data sets were chosen : EEA and EMEP.

The data collected from sensors are planned to be used in combination with existing data sources (EEA and EMEP) in order to provide a value-added product to citizens.

15. Link and additional documentation

Provide a link to the location of the dataset if available (e.g. WFS end-point), or link to websites, documents, etc., with further information about the dataset.

Other information can be found at:

http://www.eea.europa.eu/data-and-maps/data/airbase-the-european-air-quality-database-6

http://ebas.nilu.no

16. Other information

If there is relevant information about the dataset not covered in the above items, please provide details here. For example if the dataset is in languages other than English, any data privacy issues, etc.

Smartphones sensors measurements in Barcelona

1. Dataset name

Give a name to the dataset. A dataset is a collection of data which taken as a whole offers some useful information to a potential user. An example could be "Air quality measurements in Norwegian cities".

Different measurements (e.g. temperature, location, acceleration, velocity, etc.) provided from smart phone's sensors in Barcelona.

2. Maintainer organization and contact person

Who is the maintainer of the dataset and/or the person who can be contacted for further details about the dataset? SINTEF: http://www.sintef.no/

3. Dataset short description

Brief description about the dataset:

a. What is the data about;

   Information collected from mobile phone's sensors (gyroscope, accelerometer, etc.) regarding position, speed, weight, light, etc.

b. Who are the potential users of the data and the stakeholders;

   Telephone companies, people interested in city surroundings

c. What is currently the data used for;

   Statistical purposes, information about city surroundings

d. How do you envision using the data?

   Collect raw data, transform to RDF, link to other existing related data, and visualize reports and graphics

4. Theme / tags

Categorise the dataset and provide some relevant keywords/tags. For example, air quality, measurements, Norwegian cities, etc.

Mobile phone sensor data collection, Barcelona, measurements : speed, elevation, weight, light, temperature, battery level, Wifi signal etc.

5. Data collection

How is the data in the dataset being collected? For example, automatically via e.g. real-time sensors, manually collected, manually composed and curated from other datasets, etc. The data is collected real-time using the registered sensors incorporated in the smart phones.

6. Data storage and access

How is the data currently being stored? The data is stored in a MongoDB database on Cloud (Amazon server).

How is the data currently being accessed?

- The database can be accessed directly on server :

  First server: ec2-54-228-9-253.eu-west-1.compute.amazonaws.com
  Second server: ec2-54-228-19-44.eu-west-1.compute.amazonaws.com

- Data can be accessed through a web portal:

http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com:8080/SensAppGUI/
http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com:8080/SensAppGUI/

- There's also a new service as part of SensApp called the converter. It is located at this address: http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com/sensapp/converter/toCSV

Data can be retrieved from several sensors and group the values according to their timestamp. To generate the table below, you have to POST at the previous address a JSON describing the sensors you are interested in as described below:

Timestamp	GPS lat	GPS long	ACC x	ACC y	ACC z
0:00:01	• etc	• etc	• etc	• etc	• etc
• 0:00:02	• etc	• etc	• etc	• etc	• etc
• 0:00:03	• etc	• etc	• etc	• etc	• etc


```
{
  "datasets" : [{
        "url":"http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com:80/sensapp/databases/raw/data/ACC-z-ParticipantID001"
    }, {
        "url":"http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com:80/sensapp/databases/raw/data/ACC-x-ParticipantID001"
    }, {
        "url":"http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com:80/sensapp/databases/raw/data/ACC-y-ParticipantID001"
    }, {
        "url":"http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com:80/sensapp/databases/raw/data/GPS-latitude-ParticipantID001"
    }, {
      "url":"http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com:80/sensapp/databases/raw/data/GPS-longitude-ParticipantID001"
    } ],
```

"separator" : ","
}

7. Data structures

How the data is currently structured? Data format relies on HTTP negotiation mechanisms:

- application/json: Usual SenML representation
- text/xml: XML «flavor» of SenM

8. Metadata and provenance

Is there any metadata attached to the dataset? If yes, against which specification (e.g. any specific metadata standard)? Are there any provenance information attached to the dataset (e.g. who created it, when, how, etc.)? n/a

9. Quantity/Size

How large is the dataset? How many Mb, Gb, Tb, etc.

One sensor sends 86400 messages daily (one per second). One SenML message weights in average 100 bytes. This means that one sensors produces 9 MB of data daily (3 GB per year). The current system collects data from 16 sensors.

10. Licensing

Are there any licensing issues with the data sets? If so, describe them. For example, is the data restricted for use only for some users, freely available, etc. Do you have plans to monetize the data? There are no restrictions of using the data. The application is open-source and provides access to the data.

11. Update frequency

How often is the dataset updated (e.g. real-time updates, seconds, minutes, hours, daily, weekly, monthly, annually)?

The data is updated every second.

12. Geographical coverage

If the dataset has a geospatial aspect, which areas, regions, countries, etc, does the data cover? The data refers to information taken from Barcelona city.

13. Quality

Provide some information about the quality of the dataset. For example, is the dataset the authoritative dataset in a certain area (country, city, region, etc), is the dataset conformant with any directives (e.g. INSPIRE), is the dataset outdated, not maintained, is the granularity of the data good enough, is there any available documentation, usage instructions, is the data accurate, does it contain any uncertainty information, etc?

The dataset is maintained, and updated every second and contains relevant and accurate information related to Barcelona’s surroundings. The fields are meaningfully subdivided using SenML representation (sensor name, base unit, base timestamp, measurement : name, value, time). Documentation and usage instructions are available in the attached documents: 2013-01-24_SENSAPP.ppt and 2012-06-14_SensApp_MODERATES_Workshop.docx

14. **Dependencies**

Is the dataset dependent on other datasets? If yes, which datasets? For example if one measures the air quality in certain locations, are the locations taken from 3rd party data sets (e.g. mapping agencies datasets with location in a country)?

The data is not dependent on other datasets.

15. **Link and additional documentation**

Provide a link to the location of the dataset if available (e.g. WFS end-point), or link to websites, documents, etc., with further information about the dataset.

In order to access the data set, the following link can be used: http://ec2-54-228-9-253.eu-west-1.compute.amazonaws.com:8080/SensAppGUI/

Additional information can be found in the:

- Scientific Article: http://www.i3s.unice.fr/~mosser/_media/research/micas12.pdf
- SenML representation documentation : http://tools.ietf.org/html/draft-jennings-senml-08

ANNEX II Guidelines for modelling ontologies within CITI-SENSE

This annex shall give an overview to the modelling guidelines that should be taken into account when modelling core ontologies for usage within the CITI-SENSE platform. The purpose of these guidelines is to have a homogeneous style over all ontologies used in the framework.

6.1 Syntactical guidelines

This section will give an overview of different syntactical issues and what restrictions have to be taken into account when modelling ontologies within the CITI-SENSE platform. First goal of these guidelines is to have a consistent set of ontologies which are easy to read in the same manner. Second one is to have a guideline for documenting the parts of ontology in order to have well documented ontologies.

6.1.1 Namespaces within CITI-SENSE:

Namespaces are used in order to differentiate between parts of CITI-SENSE. We suggest the official URL from the coordinator extended by the project shortcut to be the base namespace

Base **http://www.citi-sense.eu/2013/cityobservatories**

6.1.2 Concept naming within CITI-SENSE:

Concept names should clearly reflect the human language in order to allow an intuitive access to the semantic of the concepts within the ontology.

As the distinction between different structures within the framework is done by namespaces we do not suggest reflecting this in the concept naming as well.

General rules:

- Concept names should start with a capital letter
- No shortcuts should be used for concept names
- If the concept name consists of two or more words, the concept name is written in one word and the single parts are separated by writing the first letter of each word in capital
- Concept names should always be in singular
- The length should not contain more than ___ letters

Examples:

Correct examples:

- Person
- Wine
- WhiteWine

Incorrect examples:

- Dogs: it is in plural, but singular should be used (Dog)
- Animal: first letter is small but capital ones should be used (Animal)
- Srvc: no shortcut should be used (Service)

- Whitewine: consists of two words that should be divided by capital letter for the first letter of each word (WhiteWine)

6.1.3 Attribute and relation naming

As well as in the concept naming attribute and relation names should reflect very well the semantics of the attribute/relation

General rules:

- Attribute/relation names should start with a small letter
- No shortcuts should be used
- If the attribute/relation name consists of two or more words, the attribute/relation name is written in one word and the single parts are separated by writing the first letter of each word in capital (except of the first)
- The length should not contain more than ___ letters

Examples:

Correct examples:

- name
- hasFirstName

Incorrect examples:

- Knows - Attribute/relations should not start with a capital letter (knows)
- hasmember - consists of two words that should be divided by capital letter for the first letter of each word except first word (hasMember)
- isCplt - no shortcut should be used (isComplete)

6.1.4 Comments (descriptions)

Comments should be delivered for each concept as well as for each relation/attribute. The comment should – beside the name of the concept/attribute/relation – give a clear idea what the modeller has in mind when modelling the concrete concept/attribute/relation. This will not only help for people to get a quicker idea of the semantics but also helps to avoid misunderstanding or ambiguousness in the ontology. The comments should contain the change history of the concept/relation/attribute as well as the shortcut for the modeller.

General rules:

- The comment should contain the shortcut of the modeller
- The comment should contain the date of creation of the concept/attribute/relation
- The comment should contain a change history
- The length should not contain more than __ letters

Example:

Concept : Person

Modeller : HT

Creation date : 2013-09-19

Description:

A person is defined by philosophers as a being who is in possession of a range of psychological capacities that are regarded as both necessary and sufficient to fulfill the requirements of personhood. These are, in general, that it is capable of reasoning, that it is self-conscious, and that it has an identity that persists through time.

Change history:

Date : 2013-09-19

Modeller : HT

Change desc : Person now subclasses LivingEntity

6.2 Concept hierarchy policy

This chapter shall give some guidelines towards the modelling of the concept hierarchy for ontologies in the CITI-SENSE project. On the one hand side they should help to avoid certain types of problems already occurred in ontology modelling and on the other hand side they should give a help for decision finding on how to model certain things.

6.2.1 Concept specification

Concepts in a hierarchy always describe keywords of the domain and sub-concepts are always the same kind of their super concepts but more specific.

If a concept B is a sub-concept of a concept A the following sentence should always be a true statement: **"Every instance of concept B is also an instance of concept A."**.

If the relation between concept A and concept B has another semantic (like "An instance of concept A consists of instances of concept B" or "An instance of concept A is the result of an instance of concept B") then this is not a sub-concept relation and they should not be in the same path of the concept hierarchy. In this case there should be a relation modelled between concept A and concept B.

Basic rules:

- If a concept has only one child, there may be a modeling problem – often a sign that a definition is incomplete
- Subclasses of a class usually have
  - Additional properties
  - Additional slot restrictions
  - Participate in different relationships
- Compare to bullets in a bulleted list

Examples:

If we have a concept `Wine` then we might sub-concept this by `WhiteWine` and `RedWine` for example, because every instance of `WhiteWine` or `RedWine` is an instance of `Wine` at the same time. They are more specific than their superclass, because they take into account the color of the wine.

Instead `Grape` is not a sub-concept of `Wine` but here the relation `isMadeOf` between `Wine` and `Grape` is possible.

6.2.2 Creating levels and subclasses

Purpose of the concepts is at one side to get a common vocabulary for the given domain of interest. But on the other side it is also about structuring the vocabulary in a concept-sub-concept relationship (taxonomy). This will lead to different levels of abstraction within the ontology. Some basic rules for the levels are given here.

Basic rules:

- If a class has a large number of subclasses, it may be useful to define intermediate levels
- For example, in the domain of wines, there are natural groupings around wine color
- However, if no natural classification exists, the long list may be appropriate

6.2.3 Inheritance, naming and synonyms

Inheritance states that all relations and attributes from a superclass are also valid for each sub-concept. Synonyms are used to give a concept alternative names which can be used as well for referring to the same concept. This helps to keep the number of concepts smaller and the concept hierarchy to be more concise.

Basic rules:

- A “wine” is not a subclass of “wines”
- A particular vintage should be classified as an instance of the class Wines
- Class names should be either
  - all singular
  - all plural
- Synonym names for the same concept are not different classes
- Many systems, metadata standards support synonymous terms as part of a class definition

6.2.4 Transitivity of the hierarchical relations

A sub-concept relationship is transitive:

If B is a sub-concept of A and C is a sub-concept of B, then C is a sub-concept of A

Example:

We can define a concept Wine, and then define a concept WhiteWine as a subconcept of Wine. Then we define a concept Chardonnay as a sub-concept of WhiteWine. Transitivity of the sub-concept relationship means that the concept `Chardonnay` is also a sub-concept of `Wine`. Sometimes we distinguish between direct sub-concepts and indirect sub-concepts. A **direct subconcept** is the “closest” sub-concept of the class: there are no concepts between a concept and its direct sub-concept in a hierarchy. That is, there are no other concepts in the hierarchy between a concept and its direct super-concept. In our example, `Chardonnay` is a direct sub-concept of `WhiteWine` and is not a direct sub-concept of `Wine`.

6.2.5 Evolution of a class hierarchy

Maintaining a consistent concept hierarchy may become challenging as domains evolve. For example, for many years, all Zinfandel wines were red. Therefore, we define a concept Zinfandel wines as a sub-concept of the `RedWine` concept. Sometimes, however, wine makers began to press the grapes and to take away the color-producing aspects of the grapes immediately, thereby modifying the color of the resulting wine. Thus, we get “whitezinfandel” whose color is rose. Now we need to break the

`Zinfandel` concept into two classes of zinfandel — `WhiteZinfandel` and `RedZinfandel` — and classify them as sub-concepts of `RoseWine` and `RedWine` respectively.

Remark: this sample shows that the classification has to be thought of very carefully. The reason to create sub-concepts should be given by attributes that are as stable as possible!

6.2.6 Multiple inheritance

Most knowledge-representation systems allow **multiple inheritances** in the concept hierarchy: a concept can be a sub-concept of several concepts. This concept will inherit the attributes and relations from all its super-concepts.

Example:

Suppose we would like to create a separate concept of dessert wines, the DessertWine concept. The Port wine is both a red wine and a dessert wine. Therefore, we define a concept Port to have two superconcepts: RedWine and DessertWine. All instances of the Port concept will be instances of both the RedWine concept and the DessertWine concept. The Port concept will inherit its attributes and relations from both its parents.

6.2.7 Concept versus instance

This chapter describes the way to find a decision whether or not a certain notion should be modelled as a concept of its own or as an individual of an already given concept.

As in the previous chapter the proposed usage plays an important role and also the granularity that has to be expressed.

Example:

If you develop ontology for car manufacturers and you are only interested in the different manufacturers that are known you might model it like the following:

Concept : Car

Instances : Audi

BMW

Hyundai

…

This allows you to query the ontology about the manufacturers of cars and use the concept Car as range of a property.

But if you are interested in the different types of cars that each manufacturer provides a different modell might be more suitable:

Concept : Car

SubConcept : Audi

BMW

Hyundai

…

Instances : Audi A8

BMW 730

Hyundai Sonata

…

6.2.8 Prevention of cyclic definitions

We say that there is a cycle in a hierarchy when some concept A has a sub-concept B and at the same time B is a super-concept of A.

Creating such a cycle in a hierarchy amounts to declaring that the concepts A and B are equivalent: all instances of A are instances of B and all instances of B are also instances of A. Indeed, since B is a sub-concept of A, all B's instances must be instances of the concept A. Since A is a sub-concept of B, all A's instances must also be instances of the concept B.

Within the CITI-SENSE framework we should avoid cycles in the concept hierarchy for different reasons. First reason is that having two or more different concepts for exactly the same meaning but with different names means that there is some ambiguity which may lead to misunderstanding or even wrong modeling and results. Another reason is that it makes already complex ontologies even more complex without any need.

If a cycle occurs during modeling the according modelers of the domain should come together and find a common notion for the concepts that resolves the cycle and includes all needs of the different modelers.

6.2.9 Siblings in the concept hierarchy

Definition: Siblings in the concept hierarchy are all concepts that are sub-concepts of the same concept.

All siblings in the hierarchy must be at the same level of generality. This holds not for the concepts at the root level.

Example:

The concepts KoreanCar and Hyundai should not be sub-concept of the same concept Car because Hyundai is a special car manufacturer whereas KoreanCar is a more general type classifying a group of certain car manufacturers.

However the concepts at the root of the hierarchy represent major divisions of the domain and do not have to be similar concepts.

6.3 Attribute and relation definition guidelines

6.3.1 Relation versus Attribute

A simple distinction between attributes and relations can be described as follows:

- Attributes are describing the nature of a given concept with a basic data type
- Relations are describing the relation between two given concepts where each concept again may have more attributes and/or relations

So whenever something describes the nature of a concept **and it can be expressed sufficiently with a basic data type** it might be expressed with an attribute. As soon as the value that describes the concept **has to have more detailed information on its own** this values needs a representation as a concept and therefor it becomes necessary to use a relation in order to give a relation between the concept and the value.

Another helpful source of information what to use for the modelling of the properties can be the available information given in databases or other external information systems that should be used as facts source for the project.

6.3.2 Sub-concept versus Relation/Attribute

In order to differentiate between types of a concept there are always two possibilities. Taking an attribute/relation describing the difference between two given instances or create a more detailed sub concept that encapsulates the distinction on the concept level. This section is about how to find out which way is the proper way for different types of modelling tasks.

Example:

If we have a concept Wine there is a difference within the color of a wine. In order to be able to express this distinction between different wines it is possible to

- Use an attribute hasColor that describes the color of the wine
- Create a sub-concept for each possible color type of wines (like RedWine, WhiteWine,…)

So, what are the points for getting a decision on how to model it in a concrete situation? Mostly there are the following points to take into account when looking for the proper way:

- Usage of the distinction as a restriction for properties in other concepts:
    - If the distinction is important in different parts of the ontology (so it is used as a range of properties in other concepts and the distinction is important in this context) it should be modelled as a new collection of sub-concepts.
      Note: In this case this decision has even effect on the performance of the system. It is faster to have the distinction on the concept level rather than first get the instance and then query for the attribute/relation describing the distinction.
    - If the distinction is not important within the scope of the domain then it should not be modelled with sub-concepts but with properties describing the distinction. This will be the case when the distinction will mostly be used in order to display the value to the user but not as a condition for rules for example.

Example:

For a domain model used in a factory producing wine labels, rules for wine labels of any color are the same and the distinction is not very important.

Alternatively, for the representation of wine, food, and their appropriate combinations a red wine is very different from a white wine: it is paired with different foods, has different properties, and so on. Similarly, color of wine is important for the wines knowledge base that we may use to determine wine-tasting order. Thus, we create a separate concept for Whitewine.

6.3.3 Inverse relation

Two relations R1 and R2 may be declared to be inverse if the following statements are true:

- The domain D of R1 is the range R of R2 and vice versa.
- If i1, i2 are two instances and i1 is related to i2 via R1 then i2 is related to i1 via R2 and vice versa.

Example:

If a Wine was producedBy a Winery, then the Winery produces that Wine. So, if we know that a certain wine was producedBy a certain Winery we can follow that the Winery produces that wine. And if we know that a certain Winery produces a certain Wine we can conclude that this Wine was producedBy this Winery.

Inverse relations allow avoiding redundancy in the system on the one hand side. On the other hand side most of the time facts are already given in available database systems and there the data is stored in one way or the other. So the ontology helps to get more flexible ways to access the facts in a natural way of language (as humans are able to ask both directions and don't have to care about the underlying storage in their mind).

6.3.4 Symmetric relation

A relation R is symmetric if for each instances i1 and i2 that are related via R the following statement is true:

- If i1 is related to i2 via R then i2 is also related to i1 via R

Example:

The relation that two persons are related is symmetric. Another example might be the relation that a person knows another person is symmetric, although this must not hold for each application.

6.3.5 Transitive relation

A relation R is called transitive if for each given three instances i1, i2 and i3 the following statement is true:

- If i1 is related to i2 via R and i2 is related to i3 via R then i1 is also related to i3 via R

Example:

Again the relation that two persons are related to each other is an example for this type. Another example is the concept hierarchy in ontologies itself: if A is a sub-concept of B and B is a sub-concept of C then A is also a sub-concept of C.

6.4 Definition of usable axioms for the core ontology

Within the CITI-SENSE project several groups will work on different parts of the ontologies and a wide range of tools will be used by them. In order to allow a transfer of the models between tools with as less hassle as possible a defined set of axioms will be described within this section. It should minimize the problems of the transfer of the ontologies on the one hand side and give a good range of expressiveness on the other hand side.

Note: this section is currently under discussion and might undergo changes

Currently agreed upon axioms

RDF Schema Features:

rdfs:subClassOf

rdfs:domain

rdfs:range

rdf:type

Property Characteristics:

owl:ObjectProperty

owl:DatatypeProperty

owl:inverseOf

owl:TransitiveProperty

owl:SymmetricProperty

Annotation Properties:

rdfs:comment

Class Axioms:

rdfs:subClassOf (applied to class expressions)

6.5 Further readings

The following webpages and papers may help to get more information about how to build ontologies. Some parts from this guideline have been taken from these sources.

Ontology Development 101: A Guide to Creating Your First Ontology (http://ksl.stanford.edu/people/dlm/papers/ontology101/ontology101-noy-mcguinness.html)

OntoClean (http://hcs.science.uva.nl/SIKS/GuarinoWelty2004.pdf#search=%22ontoclean%22)

ANNEX III Upper Ontologies

6.1 DOLCE ontology

Source

The DOLCE ontology has been developed during the WonderWeb project (http://www.loa-cnr.it/DOLCE.html) which is a EU funded IST project.

Availability

The DOLCE lite ontology is available originally as a full owl file (version 3.9) and has been transformed to a DLP version in owl and oxml/flogic. There exist several extensions for different fields of interest that have not yet been transformed.

Description

A detailed description of the ontology is available in the WonderWeb Deliverable D18. Subsequent an overview the concepts, relations and the semantics will be given for the DOLCE-Lite ontology.

High level concepts

On the first level two concepts are given, namely **"spatio-temporal-particular"** and **"abstract"**. This leads to a distinction between things that exist in space (like a **"physical-object"**) or in time (like an **"event"**) on the one hand side and abstract concepts like a **"region"** or **"time-interval"**.

Within the **"spatio-temporal-particular"** part of the concept hierarchy the main distinction is made between so called **"endurant"** and **"perdurant"** concepts. In the paper D18 this is described as follows

"Classically, the difference between enduring and perduring entities (which we shall also call *endurants* and *perdurants*) is related to their behavior in time. Endurants are *wholly* present (i.e., all their proper parts are present) at any time they are present. Perdurants, on the other hand, just extend in time by accumulating different temporal parts, so that, at any time they are present, they are only *partially* present, in the sense that some of their proper temporal parts (e.g., their previous or future phases) may be not present. E.g., the piece of paper you are reading now is wholly present, while some temporal parts of your reading are not present any more."

In the next level **"endurant"** is mainly subclassed by **"non-physical-endurant"** and **"physical-endurant"**. For **"non-physical-endurant"** is only one subclass given, the **"non-physical-object"** which is the most concrete concept given by DOLCE in this part of the ontology. For the **"physical-endurant"** there are **"amount-of-matter"**, **"physical-object"** and **"feature"** given as most concrete concepts.

At the same time **"perdurant"** is distiguished into the subconcepts **"event"** (with subconcepts **"accomplishment"** and **"achievement"**) and **"stative"** (subconcepts **"process"** and **"state"**).

It should be mentioned that there are further extensions available that model certain parts of the DOLCE ontology in more details, e.g. there is an ontology about plans, social units and others. All of them are refering towards the DOLCE ontology.

"Qualities can be seen as the basic entities we can perceive or measure: shapes, colors, sizes, sounds, smells, as well as weights, lengths, electrical charges. . . 'Quality' is often used as a synonymous of 'property', but this is not the case in DOLCE: qualities are particulars, properties are universals.

Qualities *inhere* to entities: every entity (including qualities themselves) comes with certain qualities, which exist as long as the entity exists.18 Within a certain ontology, we assume that these qualities belong to a finite set of *quality types* (like color, size, smell, etc., corresponding to the "leaves" of the quality Taxonomy shown in Figure 2), and are characteristic for (*inhere in*) specific individuals: no two particulars can have the same quality, and each quality is *specifically constantly dependent* (see below) on the entity it inheres in: at any time, a quality can't be present unless the entity it inheres in is also present. So we distinguish between a quality (e.g., the color of a specific rose), and its "value" (e.g., a particular shade of red). The latter is called *quale*, and describes the position of an individual quality within a certain *conceptual space* (called here *quality space*) [39]. So when we say that two roses have (exactly) the same color, we mean that their color qualities, which are distinct, have the same position in the color space that is they have the same *color quale*."

Space and time locations as special qualities. In our ontology, space and time locations are considered as individual qualities like colors, weights, etc. Their corresponding qualia are called *spatial* (*temporal*) *regions*. For example, the spatial location of a physical object belongs to the quality type *space*, and its quale is a region in the geometric space. Similarly for the temporal location of an occurrence, whose quale is a region in the temporal space. This allows one a homogeneous approach that remains neutral about the properties of the geometric/temporal space adopted (for instance, one is free to adopt linear, branching, or even circular time).

Direct and indirect qualities. We distinguish in DOLCE two kinds of quality inherence: *direct* and *indirect* inherence. The main reason for this choice comes from the symmetric behavior of perdurants and endurants with respect to their temporal and spatial locations: perdurants have a well-defined temporal location, while their spatial location seems to come indirectly from the spatial location of their participants; similarly, most endurants (what we call *physical endurants*, see below) have a clear spatial location, while their temporal location comes indirectly from the that of the perdurants they participate in. Another argument for this distinction concerns complex qualities like colors, which – according to Gardenfors – exhibit multiple *dimensions* (hue, luminosity, etc.). We model this case by assuming that such dimensions are qualities of qualities: the quality *color of rose#1* has a specific hue that directly inheres to it, and indirectly inheres to *rose#1.*

Parts of qualities. As a final comment, we must observe that no parthood relation (neither temporal nor atemporal) is defined for qualities in the DOLCE ontology. This seems to us a safe choice, since apparently we do not need to reason about parts of qualities (while we certainly do need to reason on parts of quality regions). So we do not have to commit on a single kind of parthood relationship for them (maybe some of them need a temporal parthood, while others do not). Since no parthood is defined, qualities are neither endurants nor perdurants, although their persistence conditions may be similar, in certain cases, to those of endurants or perdurants.

Evaluation

The DOLCE ontology is on a high level of abstraction – and therefor well usable as an upper ontology as it is meant to be independent from the application level. It is assumed to be unchanged during the whole process of modelling and gives the frame for all ontologies to be developed in future.

Nevertheless due to the level of abstraction it is to be expected that mapping the other ontologies to the DOLCE ontology may be not an easy task to do. And at the same time it is not delivering a concrete usable vocabulary (such as "person", "contact" or similar).

6.2 SUMO ontology

Source

SUMO (Suggested Upper Merged Ontology) is being created as part of the IEEE Standard Upper Ontology Working Group.

Availability

The original ontology ist delivered in KIF format which is not readable for OntoStudio (and not for Protégé as well). There is an OWL translation available that is a subset of the original ontology according to the webpage (KIF can also contain rules which are not expressable in OWL) and there is an RDF(S) available form the Smartweb project (with applied changes to the ontology due to the needs of the project). Additionally there is a Protégé project file available.

Description

The content of the SUMO can be divided by topic into the following sections:

1. Principal Distinctions
2. Objects
3. Processes
4. Abstract Entities
5. Structural Ontology
6. Basic Binary Relations
7. Artifacts
8. Spatial Relations
9. Number
10. Measure
11. Organic
12. Temporal Concepts
13. Mereology
14. Semiotics

Principal Distinctions

The root node of the SUMO is, as in many ontologies, Entity, and this concept immediately subsumes Physical and Abstract. The former class includes everything that has a position in space/time, and the latter class includes everything else. Under the concept of Physical, we have the disjoint concepts of Object and Process. It is important to note that this distinction embodies a controversial ontological claim. In effect, it means the SUMO assumes a so called 3D orientation, rather than a 4D orientation. According to those who adopt a 3D orientation (or “endurantists”, as they are sometimes called), there is a basic, categorial distinction between objects and processes. According to those who adopt a 4D orientation (the “perdurantists”), on the other hand, there is no such distinction. The 3D orientation posits that objects, unlike processes, are completely present at any moment of their existence, while a 4D orientation regards everything as a space-time worm (or a slice of such a worm). On the latter view, paradigmatic processes and objects are merely opposite ends of a continuum of spatio-temporal phenomena. The concepts of **Object** and **Process** (and their subclasses) are discussed in detail in following sections.

Objects

Immediately under the concept of **Object**, there are four concepts, viz. **SelfConnectedObject**, **Region**, **Collection**, and **Agent**. A **SelfConnectedObject** is any **Object** whose parts are all mediately or immediately connected with one another. The two most important concepts under **SelfConnectedObject** are **Substance** and **CorpuscularObject**. A **Substance** is an **Object** in which every part is similar to every other in every relevant respect. More precisely, something is a **Substance** when all of its parts (down to an unspecified level of granularity) have the properties of the whole. Thus, for example, water and clay would be subclasses of **Substance**. The class **CorpuscularObject** is disjoint from the class **Substance**, and, in particular, its instances can be divided into separate components. Disjoint from **SelfConnectedObject** is the class **Collection**. Each instance of **Collection** consists of disconnected parts, and the relation between these parts and the whole is known as **member** in the SUMO. Note that the **member** predicate is different from the **instance** and **element** predicates, which relate things to the classes or sets, respectively, to which they belong. Unlike **SetOrClasses**, **Collections** have a position in space-time, and **members** can be added and subtracted without thereby changing the identity of the **Collection**. Some examples of **Collections** are toolkits, football teams, and flocks of sheep.

Processes

The other child of **Physical** is the concept **Process**. The typology of processes in the SUMO was inspired by Beth Levin's well-received work entitled "Verb Classes and Alternations". Among other things, this work attempts to classify over 3,000 English verbs into 48 “semantically coherent verb classes”. Not all of these classes are represented in the indented list below, because not all of them are relevant to the task of developing a typology of processes. Some of the verb classes relate to static predicates in the ontology rather than to processes and some classes are syntactically motivated, e.g. the class of verbs that take predicative complements. In the **Processes** branch of the SUMO, we have eliminated the verb classes that do not seem to refer to genuine processes, altered the structure of the hierarchy somewhat, and developed formal axioms for the remaining concepts.

Abstract Entities

The class **Abstract** subsumes five disjoint concepts: **SetOrClass**, **Relation**, **Proposition**, **Attribute**, and **Quantity**. **SetOrClass** subsumes **Set** (the ordinary set-theoretic notion) and **Class**. A **Class** is associated with a property or conjunction of properties that constitute the conditions for membership in the **Class**, while a **Set** can be an arbitrary set of things.

The concept of **Proposition** corresponds to the notion of semantic or informational content. Unlike many other ontologies, the SUMO places no size restrictions on this content. Although some **Propositions** are expressed by single sentences, other **Propositions** are expressed by entire books or even libraries of books. This is a broader notion than is used in many ontologies, but it does not seem to be possible to make a principled distinction between the abstract content expressed by one sentence and the abstract content expressed by larger units of discourse. Some examples of **Propositions** would be the story line conveyed by a novel and the musical content denoted by a printed score.

The class of **Attributes** includes all qualities, properties, etc. that are not reified as **Objects**. For example, rather than dividing the class of **Animals** into “FemaleAnimals” and “MaleAnimals”, **Female** and **Male** are instances of **BiologicalAttribute**, which is a subclass of **Attribute**.

The class **Quantity** under **Abstract** is divided into **Number** and **PhysicalQuantity**. The former is understood as a count independent of an implied or explicit measurement system, and the latter is taken to be a complex consisting of a **Number** and a particular unit of measure. Thus, 1 meter and

39.37 inches would be two distinct instances of **PhysicalQuantity** in the SUMO, though they can be shown to be equivalent via conversion axioms in the ontology.

Structural Ontology

This section of the ontology consists of **Relations** and **Classes** that are used to frame the content of the ontology (the content that is contained in the other sections of the SUMO). It contains definitions of syntactic abbreviations that can be heuristically useful and computationally advantageous. It also covers set-theoretic predicates and functions (adapted from the kif-sets ontology on the Ontolingua server) and relation types (adapted from frame-ontology, abstract-algebra, kif-relations, and kif-extensions, also on the Ontolingua server).

Basic Binary Relations

Binary relations are the most familiar and most useful type of relation in any ontology, and the SUMO contains a wide variety of them. There are case roles (**CaseRole**) for representing the relations between a **Process** and something essentially involved in the Process, predicates to express part/whole relationships, and various locative relations, among others.

Artifacts

This section covers objects that are intentionally produced by human beings.

Number

This section contains a hierarchy of number types, as well as **Predicates** and **Functions** that are defined over numbers.

Measure

This section of the SUMO incorporates the relations in the Quantities ontology (developed by ITBM-CNR) and the units of measure in the "Standard Units" and "Standard Dimensions" ontologies on the Ontolingua server. This section has been extensively revised by Helena Sofia Pinto of the Instituto Superior Tecnico in Portugal. The sources for these revisions were as follows: Barry Taylor, NIST Special Publication 811, Guide for the Use of the International System of Units (SI), 1995, and Encyclopaedia Britannica (on-line version at http://www.britannica.com).

Organic

This is essentially a taxonomy of high-level biological categories. The original version of the taxonomy was inspired by the Natural Kinds ontology developed by ITBM-CNR.

Temporal Concepts

This section contains measures of time, as well as predicates and functions for linking such measures to themselves and to instances of 'Physical'. The original version of this section was derived largely from James Allen's work and from the Simple-Time ontology on the Ontolingua server. Much of this section was extensively revised on the basis of comments from Pat Hayes. Additional suggestions were provided by Jerry Hobbs.

Mereology

The general predicates in this section are adapted from Barry Smith's and Nicola Guarino's papers on the subject. The stuff relating to holes is adapted from Casati and Varzi's book on the subject.

Spatial Concepts

This section of the SUMO was inspired by the content in the Positions ontology of ITBM-CNR. This content is a class of attributes and a class of relations.

Semiotics

This section covers concepts related to the representation and communication of information. Most of this content was developed on an ad hoc basis with input from the SUO Working Group.

Evaluation

Compared to the DOLCE ontology the SUMO ontology has more concrete concepts available which seems to be easier to use on the first view. On the other hand the concept hierarchy seems to be at some points unsharp and the distinction between different concepts is sometimes not very intuitiv. Another problem that was also observed during the usage of the SUMO ontology in the SmartWeb project is that there are several concepts that inherit from different superconcepts which may lead to ambiguity and therefor to difficulties.

Nevertheless the ontology might be used as source of ideas for the CITI-SENSE upper ontology anyway. But it is not recommended to take it as it is.

6.3 SOUPA

Source

The SOUPA ontology has been developed by a group of people around Harry Chen with regard to ubiquitos computing (http://pervasive.semanticweb.org/soupa-2004-06.html).

Availability

The ontology is divided into several subfiles in OWL. There exists a core part and several extensions. There seems to be currently some problems in the conversion of these file or even in the correct display (testet with OntoStudio and Protégé). Therefor currently no DLP version is available.

Description

SOUPA Core

This set of ontologies consists of vocabularies for expressing concepts that are associated with person, agent, belief-desire-intention (BDI), action, policy, time, space, and event. The ontologies are grouped into nine distinctive ontology documents.

Person

This ontology defines typical vocabularies for describing the contact information and the profile of a person. The OWL class per:Person is defined to represent a set of all people in the SOUPA domain, and is equivalent

to the foaf:Person class in the FOAF ontology (i.e., the owl:equivalentClass property holds between the per:Person and foaf:Person class). An individual of the class can be described by a set of properties, which include basic profile information (name, gender, age, birth date, etc.), the contact information (email, mailing address, homepage, phone numbers, instant messaging chat ID, etc.), and social and professional profile (people that a person is friend of, organizations that a person belongs to). In addition, all property vocabularies that are applicable to describe a person in the FOAF ontology can also be used to describe an individual of the per:Person class. This is because all individuals of the per:Person class are also individuals of the foaf:Person class.

Policy & Action

Security and privacy are two growing concerns in developing and deploying pervasive computing systems. Policy is an emerging technique for controlling and adjusting the low-level system behaviors by specifying high-level rules.

The SOUPA policy ontology defines vocabularies for representing security and privacy policies and a description logic based mechanism for reasoning about the defined policies. The defined vocabularies in this ontology are influenced by the Rei policy language. A policy is a set of rules that is specified by a user or a computing entity to restrict or guide the execution of actions. For example, in the context of system security, a system administrator may use policies to define who has the right to execute what services; in the context privacy protection, a user may use policies to restrict the type of personal information that can be shared by the public services.

The ontology representation of an action is defined in the action ontology document. The class act:Action represents a set of all actions. Individuals of this class can have a set of property values, which include

(i) act:actor – the entity that performs the action,

(ii) act:recipient – the entity that receives the effect after the action is performed,

(iii) act:target – the object that the action applies to,

(iv) act:location – the location at where the action is performed,

(v) act:time – the time at which the action is performed,

(vi) act:instrument – the thing that the actor uses to perform the action.

In SOUPA, a policy consists of rules that either *permit* or *forbid* the execution of certain described actions. Defined in the policy ontology document, the pol:Policy class represents a set of all policies. For a given policy individual, it may be associated with one or more pol:permits or pol:forbids properties. The range of these two properties are the pol:PermittedAction class and the pol:ForbiddenAction class, respectively.

The policy ontology also defines vocabularies for describing meta information about individual policies. This information includes the author of a policy (pol:creator), the entity that enforces a policy (pol:enforcer), the creation time of a policy (pol:createdOn), and the default reasoning mode of a policy (pol:defaultPolicyMode).

The design of the SOUPA policy exploits *classification* as a means to reason about policies. A typical process flow of the system implementation is the following:

(i) a user or a system administrator defines a policy,

(ii) the policy is transmitted to the appropriate policy enforcer (e.g, a security or a privacy protection agent),

(iii) before the policy enforcer can permit other agents to perform an action, it creates an explicit representation of the action using the SOUPA action ontology,

(iv) this represented action is then loaded into a description logic reasoner along with the associated ontology,

(v) the policy http://www.cs.man.ac.uk/˜horrocks/FaCT/ enforcer will permit the execution of the action if the action is classified as type of pol:PermittedAction, and it will forbid the execution of the action if the action is classified as type of pol:ForbiddenAction.

In case if an input action is classified as both pol:PermittedAction and pol:Forbidden-Action, then the policy enforcer will report there is an inconsistency in the policy and may forbid the execution of the action by default. In case if the action cannot be classified as either classes, the policy enforcer will decide whether the action should be permitted or forbidden based on the default policy mode (see the above example). If the mode is pol:RequiresExplicit-Permission, then the action will be forbidden. If the mode is pol:RequiresNoExplicitPermission, then the action will be permitted.

Agent & BDI

When building intelligent pervasive computing systems, sometimes it is useful to model computing entities as *agents*. In SOUPA, agents are defined with a strong notion of agency, which is characterized by a set of *mentalistic* notions such as knowledge, belief, intention, and obligation. In this ontology, both computational entities and human users can be modeled as agents.

When the goals, plans, desires, and beliefs of different agents are explicitly represented in the ontologies, this information can help independently developed agents to share a common understanding of their "mental" states, helping them to cooperate and collaborate. The explicitly represented human user's mental states can help computing agents to reason about the specific needs of the users in a pervasive environment.

Two ontology documents are related to this ontology:agent and bdi. The agt:Agent class represents a set of all agents in the SOUPA domain and is associated with three properties that can be used to characterize an agent's "mental" state: agt:believes, agt:desires, and agt:intends. The respective range values of these properties are the bdi:Fact, bdi:Desire, and bdi:Intention classes. The goals of an agent are considered to be a special type of desire, which is expressed by defining the agt:hasGoal property as a sub-property of the agt:desires property.

The bdi:Fact class is a subclass of the rdf:Statement class, which represents a set of reified RDF statements. A reified RDF statement consists of the rdf:subject, rdf:object, and rdf:predicate properties.

The bdi:Desire class defines a set of world states that agents desire to bring about. Every instances of this class can be characterized by the property bdi:endState. The range restriction of this property is unspecified in the bdiontology document. Application developers are responsible for defining the representation of different world states.

Some suggested representations are

(i) symbolic names, e.g., a set of pre-defined RDF resource URI and

(ii) metarepresentation, e.g., each world state description is a set of reified RDF statements.

The bdi:Intention class represents a set of plans that agents intend to execute. Plans are defined in terms of actions, pre-conditions, and effects. The bdi:Plan class is defined as a subclass of the act:Action class with additional properties, namely bdi:preCondition and bdi:effect.The representation of pre-conditions and effects are unspecified in this ontology, and it is left to be defined by the application ontologies.

Sometimes it may be useful to describe whether or not different desires of an agent are in conflict of each other, and whether or not certain desires are achievable. The cause of desire conflicts may be due to inconsistent beliefs in the knowledge base or conflicting user preferences or systems policies. The cause of unachievable desires may be due to the change of situational conditions. In the bdi ontology document, different subclasses of the bdi:Desire class, bdi:ConflictingDesire, bdi:Non-ConflictingDesire, bdi:AchievableDesire, and bdi:UnachievableDesire, are defined for classifying different types of agent desires.

Time

SOUPA defines a set of ontologies for expressing time and temporal relations. They can be used to describe the temporal properties of different events that occur in the physical world.

Part of the SOUPA ontology adopts the vocabularies of the DAML-time and the entry sub-ontology of time. The basic representation of time consists of the tme:Time-Instant and tme:TimeInterval classes. All individual members of these two classes are also members of the tme:TemporalEntity class, which is an OWL class that is defined by taking the union of the tme:Time-Instant and tme:TimeInterval classes. The set of all temporal things that are divided into two disjoint classes: tme:InstantThing, things with temporal descriptions that are type of time instant, and tme:Interval-Thing, things with temporal descriptions that are type of time interval. The union of these two classes forms the tme:TemporalThing class.

In order to associate temporal things with date/time values (i.e., their temporal descriptions), the tme:at property is defined to associate an instance of the tme:Instant-Thing with an XML xsd:dateTime datatype value (e.g., 2004-12-25T12:32:12), and the tme:from and tme:to properties are defined to associate an instance of the IntervalThing with two different tme:Time-Instant individuals.

For describing the order relations between two different time instants, the ontology defines the following properties: tme:before, tme:after, tme:beforeOr-At, tme:afterOrAt, and tme:sameTimeAs. Both tme:before and tme:after properties are defined of type owl:TransitiveProperty. The tme:same-TimeAs property expresses that two different time instants are associated with equivalent date/time values and is defined of type owl:SymmetricProperty.

For describing the order relations between two different temporal things (i.e., time instants and time intervals),

the ontology defines the following properties:

tme:startsSoonerThan, tme:startsLater-Than, tme:startsSameTimeAs, tme:ends-SoonerThan, tme:endsLaterThan, tme:ends-SameTimeAs, tme:startsAfterEndOf, and tme:endsBeforeStartOf. The first three properties respectively express that for any two given temporal things A and B, the starting time of A is before the starting time of B, the starting time of A is after the starting time of B, and the starting time of A is the same as the starting time of B. The next three properties respectively express that for any two given temporal things A and B, the ending time of A is before the ending time of B, the ending time of A is after the ending time of B, and the ending time of A is the same as the ending time of B. The tme:starts-AfterEndOf property expresses that the beginning of one temporal thing is after the ending of another temporal thing, and the tme:endsBeforeStartOf property expresses the inverse of this property.

In the future we plan to extend this ontology to adopt or map to additional vocabularies from the RDF Calendar ontologies3 for modeling time intervals that may contain repeating time intervals or instants. This new feature will be useful for representing recurrent events such as weekly meetings and classes.

Space

This ontology is designed to support reasoning about the spatial relations between various types of geographical regions, mapping from the geo-spatial coordinates to the symbolic representation of space and *vice versa*, and http://www.w3.org/2002/12/cal/ the representation of geographical measurements of space.

Part of this ontology vocabularies are adopted from the spatial ontology in OpenCyc and the OpenGIS vocabularies.

Two ontology documents are related to this ontology: space and geo-measurement. The first ontology document defines a symbolic representation of space and spatial relations, and the second document defines typical geospatial vocabularies (e.g., longitude, latitude, altitude, distance, and surface area).

In the symbolic representation model, the spc:SpatialThing class represents a set of all things that have spatial extensions in the SOUPA domain. All spatial things that are typically found in maps or construction blueprints are called spc:GeographicalSpace. This class is defined as the union of the spc:GeographicalRegion, spc:FixedStructure, and spc:SpaceInAFixed- Structure classes.

An individual member of the spc:Geographical-Region class typically represents a geographical region that is controlled by some political body (e.g., the country US is controlled by the US government). This relation is expressed by the spc:controls property, the domain of which is spc:GeopoliticalEntity and the range of which is spc:GeographicalRegion. Knowing which political entity controls a particular geographical region, a pervasive computing system can choose to apply the appropriate policies defined by the political entity to guide its behavior. For example, a system may apply different sets of privacy protection schemes based on the policies defined by the local political entities.

To support spatial containment reasoning, individual members of the spec:GeographicalSpace class can relate to each other through the spc:spatially-Subsumes and spc:spatiallySubsumedBy properties. For example, a country region may spatially subsume a state region, a state region may spatially subsumes a building, and a building may spatially subsumes a room. Knowing the room in which a device is located, we can infer the building, the state and the country that spatially subsumes the room.

In the geo-spatial representation model, the individual members of the spc:SpatialThing class are described by location coordinates (i.e., longitude, latitude, and altitude). This relation is expressed by the spc:hasCoordinates property, the range of which is the geo:LocationCoordinates class. In this model, multiple location coordinates can be mapped to a single geographical region (e.g., a university campus typically covers multiple location coordinates.). This relation is useful for defining spatial mapping between different geographical locations and GPS coordinates. This information can enable a GPS-enabled device to query the symbolic representation of its present location for a given set of longitude, latitude, and altitude.

Event

Events are event activities that have both spatial and temporal extensions. An event ontology can be used to describe the occurrence of different activities, schedules, and sensing events. In the event ontology document, the eve:Event class represents a set of all events in the domain. However, the definition of this class is silent about its temporal and spatial properties.

The eve:SpatialTemporalThing class represents a set of things that have both spatial and temporal extensions, and it is defined as the intersection of the tme:TemporalThing and spc:Spatial-Thing classes. To specifically describe events that have both temporal and spatial extensions, eve:Spatial-TemporalEvent class is defined as the intersection of the eve:SpatialTemporalThing and eve:Event classes.

SOUPA Extension

The SOUPA Extension ontologies are defined with two purposes:

(i) define an extended set of vocabularies for supporting specific types pervasive application domains, and

(ii) demonstrate how to define new ontologies by extending the SOUPA Core ontologies. At present, the SOUPA Extension consists of experimental ontologies for supporting pervasive context-aware applications in smart spaces and peer-to-peer data management in a pervasive computing environment.

Meeting & Schedule

For describing typical information associated with meetings, event schedules, and event participants.

Document & Digital Document

For describing meta-information about documents and digital documents, e.g., the creation date and the author of a document, the source URL of a digital document, file size, and file type.

Image Capture

When a camera phone takes a picture, this event is type of image capturing event. This ontology defines vocabularies for describing image capturing events (where and when a picture is taken, which device has taken the picture, etc.).

Region Connection Calculus

A spatial ontology that supplements the core space ontology. Based on the Region Connection Calculus, this ontology defines vocabularies for expressing spatial relations for qualitative spatial reasoning.

Location

For describing sensed location context of a person or an object. The location context is information that describes the whereabouts of a person or an object, which includes both temporal and spatial properties.

Evaluation

Similar to the SUMO there are more specific concepts than in the DOLCE-Lite ontology but also abstractions available. There are needed definitions for things like agent, time but also emotional states can be described. The range of the SOUPA ontology seems to fit quiet well to the scope of the CITI-SENSE but as with all ontologies it should only be taken into account as a delivering source for an own upper ontology.

6.4 FOAF Ontology

Source

Latest Version: http://xmlns.com/foaf/0.1/

FOAF Basic Description

FOAF (Friend-Of-A-Friend) is RDFS/OWL ontology for expressing information about persons and their relationships. The FOAF project is based around the use of *machine readable* Web homepages for people, groups, companies and other kinds of thing

Availability

The FOAF ontology is available originally as an full owl file and has been transformed to an DLP version in owl and oxml

High Level Concept

1. Class: foaf:Agent

Agent - An agent (eg. person, group, software or physical artifact).

Status:	unstable
in-range-of:	Foaf:maker, foaf:member
in-domain-of:	Foaf:mbox, Foaf:mbox_sha1sum, Foaf:gender, Foaf:jabberID, Foaf:aimChatID, Foaf:icqChatID, Foaf:yahooChatID, Foaf:weblog, Foaf:tipjar, Foaf:made, Foaf:holdsAccount, Foaf:birthday

The foaf:Agent class is the class of agents; things that do stuff. A well-known sub-class is foaf:Person, representing people. Other kinds of agents include foaf:organization and foaf:Grop

The foaf:Agent class is useful in a few places in FOAF where foaf:Person would have been overly specific. For example, the IM chat ID properties such as jabberID are typically associated with people, but sometimes belong to software bots.

2. Class: foaf:Document

Document - A document.

Status:	testing
in-range-of:	Foaf:homepage, Foaf:weblog, Foaf:tipjar, Foaf:workplaceHomepage, Foaf:schoolHomepage, Foaf:interest, Foaf:publications, Foaf:isPrimaryTopicOf, Foaf:page, Foaf:accountServiceHomepage
in-domain-of:	Foaf:sha1, Foaf:topic, Foaf:primaryTopic

The foaf:Document class represents those things which are, broadly conceived, 'documents'.

3. Class: foaf:OnlineAccount

Online Account - An online account.

Status:	unstable
in-range-of:	Foaf:holdsAccount
in-domain-of:	Foaf:accountServiceHomepage, foaf:accountName

A foaf:OnlineAccount represents the provision of some form of online service, by some party (indicated indirectly via a foaf:accountServiceHomepage) to some foaf:Agent. The foaf:holdsAccount property of the agent is used to indicate accounts that are associated with the agent.

4. Class: foaf:Project

Project - A project (a collective endeavour of some kind).

Status: unstable

The foaf:Project class represents the class of things that are 'projects'. These may be formal or informal, collective or individual. It is often useful to indicate the foaf:homepage of a foaf:Project.

Overview FOAF

FOAF terms, grouped in broad categories.

FOAF Basics

- **Agent**
- **Person**: The foaf:Person class represents people. Something is a foaf:Person if it is a person. We don't take care about whether they're alive, dead, real, or imaginary. The foaf:Person class is a sub-class of the foaf:Agent class, since all people are considered 'agents' in FOAF.
- **name**: FOAF provides some other naming constructs. These are under development. While foaf:name does not explicitly represent name substructure (family vs given etc.) it does provide a basic level of interoperability.
- **nick**: property, such as those use in IRC chat, online accounts, and computer logins.
- **title**: Property, Values such as 'Mr', 'Mrs', 'Ms', 'Dr' etc. are expected.
- **homepage**: The foaf:homepage property relates something to a homepage about it.
- **mbox**: Property, personal mainbox
- **mbox_sha1sum**: personal mailbox URI name
- **img**: property, foaf:img does not have any restrictions on the dimensions, colour depth, format etc of the foaf:Image it references.
- **depiction(depicts):** Property, extended to deal with multimedia content (video clips, audio
- **surname**
- **familyname**
- **givenname**
- **firstName**

Personal Info

- **weblog**: Property
- **knows**: Property, A person known by this person
- **interest**: The foaf:interest property represents an interest of a foaf:Agent, through indicating a foaf:Document whose foaf:topic(s) broadly characterises that interest.
- **currentProject**: Property, this person work on
- **pastProject**
- **plan**
- **based_near**
- **workplaceHomepage**
- **workInfoHomepage**
- **schoolHomepage**
- **topic_interest**: Property, A thing of interest to this person

- **publications**
- **geekcode**
- **myersBriggs**: A Myers Briggs (MBTI) personality classification, The values for foaf:myersBriggs are the following 16 4-letter textual codes: ESTJ, INFP, ESFP, INTJ, ESFJ, INTP, ENFP, ISTJ, ESTP, INFJ, ENFJ, ISTP, ENTJ, ISFP, ENTP, ISFJ. If multiple of these properties are applicable, they are represented by applying multiple properties to a person.
- **dnaChecksum**

Online Accounts / IM

- **OnlineAccount**
- **OnlineChatAccount**
- **OnlineEcommerceAccount**
- **OnlineGamingAccount**
- **holdsAccount**
- **accountServiceHomepage**
- **accountName**
- **icqChatID**
- **msnChatID**
- **aimChatID**
- **jabberID**
- yahooChatID

Projects and Groups

- **Project**
- **Organization**: The foaf:Organization class represents a kind of foaf:Agent corresponding to social instititutions such as companies, societies etc
- **Group**
- **member**
- **membershipClass**
- **fundedBy**
- **theme**

Documents and Images

- **Document**
- **Image**: Digital images (such as JPEG, PNG, GIF bitmaps, SVG diagrams etc.) are examples of foaf:Image
- **PersonalProfileDocument**
- **Topic(page)**
- **primarytopic**
- **tipjar**
- **sha1**
- **made(maker)**
- **thumbnail**
- **logo**

Evaluation

FOAF ontology is useable for describing the basic information about persons and their relations. FOAF ontology allows amending it with the information contained in person's profiles. This information may be useable to many businesses, including human resources management and social networks. We wouldn't use "project" concept in FOAF for CITI-SENSE and should more detail about person's profile and preferences. We can combine with person concept in SOUPA ontology.

6.5 Dublin Core

Source

The "Dublin" in the name refers to Dublin, Ohio, USA, where the work originated from a workshop hosted by OCLC, a library consortium which is based there. The "Core" refers to the fact that the metadata element set is a basic but expandable "core" list.

The semantics of Dublin Core were established and are maintained by an international, cross-disciplinary group of professionals from librarianship, computer science, text encoding, the museum community, and other related fields of scholarship and practice.

The Dublin Core Metadata Initiative (DCMI) is an organization providing an open forum for the development of interoperable online metadata standards that support a broad range of purposes and business models. DCMI's activities include consensus-driven working groups, global conferences and workshops, standards liaison, and educational efforts to promote widespread acceptance of metadata standards and practices.

Availability

One Document Type Definition based on Dublin Core is the Open Source Metadata Framework (OMF) specification. OMF is in turn used by ScrollKeeper, which is used by the GNOME desktop and KDE help browsers and the ScrollServer documentation server. PBCore is also based on Dublin Core. The Zope CMF's Metadata products, used by the Plone and the Nuxeo CPS Content management systems, also implement Dublin Core.

DCMI also maintains a list of projects using Dublin Core on its website.

Description

Simple Dublin Core

The Simple **Dublin Core Metadata Element Set (DCMES)** consists of 15 metadata elements:

1. Title

Label	Title
Definition	A name given to the resource.
Comment	Typically, Title will be a name by which the resource is formally known.

2. Creator

Label	Creator
Definition	An entity primarily responsible for making the content of the resource.

Comment	Examples of Creator include a person, an organization, or a service. Typically, the name of a Creator should be used to indicate the entity.

3. Subject

Label	Subject and Keywords
Definition	A topic of the content of the resource.
Comment	Typically, Subject will be expressed as keywords, key phrases or classification codes that describe a topic of the resource. Recommended best practice is to select a value from a controlled vocabulary or formal classification scheme.

4. Description

Label	Description
Definition	An account of the content of the resource.
Comment	Examples of Description include, but is not limited to: an abstract, table of contents, reference to a graphical representation of content or a free-text account of the content.

5. Publisher

Label	Publisher
Definition	An entity responsible for making the resource available
Comment	Examples of Publisher include a person, an organization, or a service. Typically, the name of a Publisher should be used to indicate the entity.

6. Contributor

Label	Contributor
Definition	An entity responsible for making contributions to the content of the resource.
Comment	Examples of Contributor include a person, an organization, or a service. Typically, the name of a Contributor should be used to indicate the entity.

7. Date

Label	Date
Definition	A date of an event in the lifecycle of the resource.
Comment	Typically, Date will be associated with the creation or availability of the resource. Recommended best practice for encoding the date value is defined in a profile of ISO 8601 [W3CDTF] and includes (among others) dates of the form YYYY-MM-DD.

8. Type

Label	Resource Type
Definition	The nature or genre of the content of the resource.
Comment	Type includes terms describing general categories, functions, genres, or aggregation levels for content. Recommended best practice is to select a value from a controlled vocabulary (for example, the DCMI Type Vocabulary [DCT1]). To describe the physical or digital manifestation of the resource, use the FORMAT element.

9. Format

Label	Format
Definition	The physical or digital manifestation of the resource.
Comment	Typically, Format may include the media-type or dimensions of the resource. Format may be used to identify the software, hardware, or other equipment needed to display or operate the resource. Examples of dimensions include size and duration. Recommended best practice is to select a value from a controlled vocabulary (for example, the list of Internet Media Types [MIME] defining computer media formats).

10. Identifier

Label	Identifier
Definition	An unambiguous reference to the resource within a given context.
Comment	Recommended best practice is to identify the resource by means of a string or number conforming to a formal identification system. Formal identification systems include but are not limited to the Uniform Resource Identifier (URI) (including the Uniform Resource Locator (URL)), the Digital Object Identifier (DOI) and the International Standard Book Number (ISBN).

11. Source

Label	Source
Definition	A Reference to a resource from which the present resource is derived.
Comment	The present resource may be derived from the Source resource in whole or in part. Recommended best practice is to identify the referenced resource by means of a string or number conforming to a formal identification system.

12. Language

Label	Language

Definition	A language of the intellectual content of the resource.
Comment	Recommended best practice is to use RFC 3066 [RFC3066] which, in conjunction with ISO639 [ISO639]), defines two- and three-letter primary language tags with optional subtags. Examples include "en" or "eng" for English, "akk" for Akkadian", and "en-GB" for English used in the United Kingdom.

13. Relation

Label	Relation
Definition	A reference to a related resource.
Comment	Recommended best practice is to identify the referenced resource by means of a string or number conforming to a formal identification system.

14. Coverage

Label	Coverage
Definition	The extent or scope of the content of the resource.
Comment	Typically, Coverage will include spatial location (a place name or geographic coordinates), temporal period (a period label, date, or date range) or jurisdiction (such as a named administrative entity). Recommended best practice is to select a value from a controlled vocabulary (for example, the Thesaurus of Geographic Names [TGN]) and to use, where appropriate, named places or time periods in preference to numeric identifiers such as sets of coordinates or date ranges.

15. Rights

Label	Rights
Definition	Information about rights held in and over the resource.
Comment	Typically, Rights will contain a rights management statement for the resource, or reference a service providing such information. Rights information often encompasses Intellectual Property Rights (IPR), Copyright, and various Property Rights. If the Rights element is absent, no assumptions may be made about any rights held in or over the resource.

Each Dublin Core element is optional and may be repeated. The DCMI has established standard ways to refine elements and encourage the use of encoding and vocabulary schemes. There is no prescribed order in Dublin Core for presenting or using the elements.

Qualified Dublin Core

Subsequent to the specification of the original 15 elements, an ongoing process to develop exemplary terms extending or refining the Dublin Core Metadata Element Set (DCMES) was begun. The additional terms were identified, generally in working groups of the Dublin Core Metadata Initiative, and judged by the DCMI Usage Board to be in conformance with principles of good practice for the qualification of Dublin Core metadata elements.

Element refinements make the meaning of an element narrower or more specific. A refined element shares the meaning of the unqualified element, but with a more restricted scope. The guiding principle for the qualification of Dublin Core elements, colloquially known as the Dumb-Down Principle [1], states that an[132] application that does not understand a specific element refinement term should be able to ignore the qualifier and treat the metadata value as if it were an unqualified (broader) element. While this may result in some loss of specificity, the remaining element value (without the qualifier) should continue to be generally correct and useful for discovery.

In addition to element refinements, Qualified Dublin Core includes a set of recommended encoding schemes, designed to aid in the interpretation of an element value. These schemes include controlled vocabularies and formal notations or parsing rules. A value expressed using an encoding scheme may thus be a token selected from a controlled vocabulary (e.g., a term from a classification system or set of subject headings) or a string formatted in accordance with a formal notation (e.g., "2000-12-31" as the standard expression of a date). If an encoding scheme is not understood by an application, the value may still be useful to a human reader.

DCMI also maintains a small, general vocabulary recommended for use within the element Type. This vocabulary currently consists of 12 terms [1]:

- Collection
- Dataset
- Event
- Image
- InteractiveResource
- MovingImage
- PhysicalObject
- Service
- Software
- Sound
- StillImage
- Text

DCMES Element		**Element Refinement(s)**	**Element Encoding Scheme(s)**
main elements	Title	Alternative	-
	Creator	-	-
	Subject	-	LCSH MeSH DDC

[132] Dublin Core Metadata Registry

DCMES Element		Element Refinement(s)	Element Encoding Scheme(s)	
			LCC UDC	
	Description	Table Of Contents Abstract	-	
	Publisher	-	-	
	Contributor	-	-	
	Date	Created Valid Available Issued Modified Date Copyrighted Date Submitted	DCMI Period W3C-DTF	
	Type	-	DCMI Type Vocabulary	• Collection • Dataset • Event • Image • InteractiveResource • Service • Software • Sound • Text • PhygicalObject
	Format	-	IMT	
		Extent	-	
		Medium	-	
	Identifier	-	URI	
		Bibliographic Citation	-	
	Source	-	URI	
	Language	-	ISO 639-2	

DCMES Element		Element Refinement(s)	Element Encoding Scheme(s)
			RFC 3066
	Relation	Is Version Of Has Version Is Replaced By Replaces Is Required By Requires Is Part Of Has Part Is Referenced By References Is Format Of Has Format Conforms To	URI
	Coverage	Spatial	DCMI Point ISO 3166 DCMI Box TGN
		Temporal	DCMI Period W3C-DTF
	Rights	Access Rights	-
Extending main element	Audience	Mediator Education Level	-

Evaluation

Dublin Core is useable for describing the Content information like document, multimedia files, etc… Widely used RSS also use Dublin Core. The more RSS contents increase, the more Dublin Core ontology become useful.