DELIVERABLE D.T.1.3.1

Elaboration of a transnational methodology based on the evaluation of the tools implemented

Version 1.0

01-2018

Table of Contents

Introduction

The road towards achievement of the climate protection goals requires, among the rest, a thorough rethinking of the energy planning tools (and policies) at all levels, from local to global.
Nevertheless, it is in the cities where the largest part of energy is produced and consumed, and therefore it makes sense to focus the attention particularly on the cities as they yield great potentials in terms of energy consumption reduction and efficiency increase. As a direct consequence, a comprehensive knowledge of the demand and supply of energy resources, including their spatial distribution within urban areas, is therefore of utmost importance.
Precise, integrated knowledge about urban space, energy infrastructures, buildings' functional and semantic characteristics, and their mutual dependencies and interrelations play a relevant role for advanced simulation and analyses[1].

As reported by the Joint Research Centre of the European Commission in "Location data for buildings related energy efficiency policies" [2] "*to implement and monitor energy efficiency policies effectively, local authorities and Member States are required to report on baseline scenarios (e.g. the Baseline Emissions Inventories in the Covenant of Mayors initiative) and on progress made at regular intervals (Annual Reports for the Energy Efficiency Directive and the Energy Performance of Buildings Directive and Monitoring Emissions Inventories every two years for the CoM)*".
Indeed, reporting tools are already available to local authorities and Member States, but they are very basic and only allow users to input aggregated and approximated values (for example, local authorities may rely on national data when local data are not available) for planning and monitoring progress towards targets.
Therefore, a **common framework for monitoring of energy efficiency policies**, with harmonised data from building to district and ending at national level could improve the interoperability of the different directives / initiatives.

[1] G. Agugiaro et al., 2018, The Energy Application Domain Extension for CityGML: enhancing interoperability for urban energy simulations, forthcoming

[2] European Commission DG JRC, 2015, Location data for buildings related energy efficiency policies, https://ec.europa.eu/isa2/sites/isa/files/actions/documents/lbna27411enn.pdf

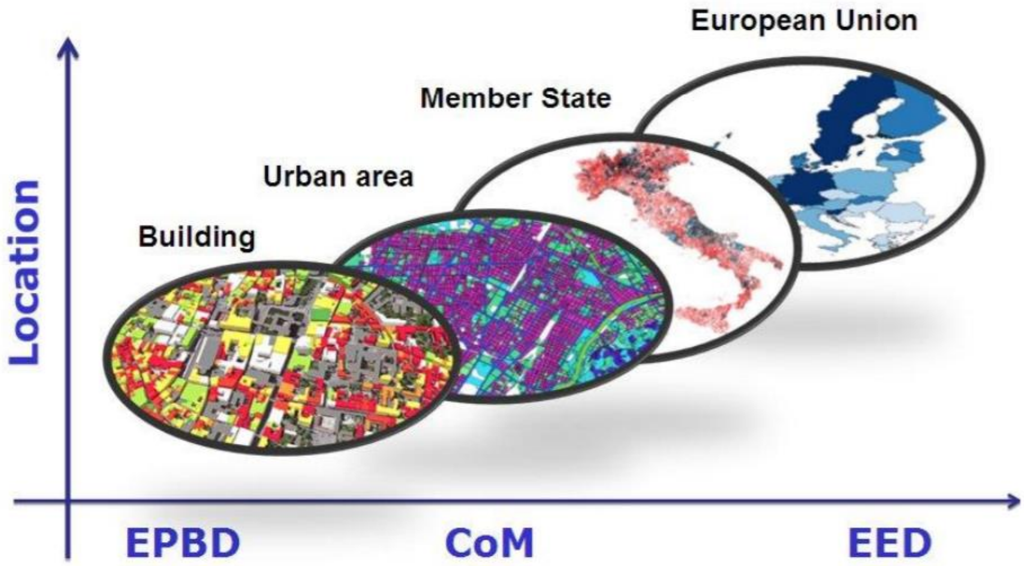

Figure 1 – Scaling and relation between EU Directives and location (EC JRC, 2015)

Within such a framework, geo-referencing all the relevant building data accurately and consistently will significantly improve data quality and reliability, enable effective scenario modelling to fill gaps in data, and support the overall policy process. Furthermore, from a potential market perspective, web-based tools providing access to the energy performance of geo-referenced buildings could improve territorial knowledge, and support, for example, the activities of energy service companies and companies involved in construction / renovation of buildings.

Scope and targets of this document

In the CitiEnGov project, Participant Partners have been asked to collect energy-related data about buildings, transport and public lighting and made them available (whole or subset) defining a harmonized "energy data model" together with ICT services for sharing energy-related data.
This document describes a **transnational methodology**, based on one hand on the evaluation of tools implemented by CitiEnGov partners, and on the other hand on standards and technologies already available at European scale for sharing interoperable energy-related data.

Due to the technical nature of the document, the text presented here is mainly addressed to ICT and geo-ICT experts, with sufficient skills on:

- Data and database modelling, data extraction/transformation/load
- Web services for presenting and sharing data
- Standards for interoperability, in particular related to geographic information

1. The data model

In CitiEnGov the three main “sectors” considered are:

- Buildings
- Mobility
- Public lighting

Actually, as described in the Covenant of Mayors’ website, “*action plans (SEAPs or SECAPs) should include actions that cover the sectors of activity from both public and private actors, covering the whole geographical area of the local authority committed*”[3].
Signatories are free to choose their main areas of action. In principle, it is anticipated that most action plans will cover the sectors that are taken into account within the emission inventory and risk and vulnerability assessment (for SECAP only).
For the **mitigation part** (both SEAP and SECAP), it is recommended to include actions targeting the Covenant key sectors:

- Municipal buildings, equipment/facilities
- Tertiary (non municipal) buildings, equipment/facilities
- Residential buildings
- Transport
- Industry
- Local electricity production
- Local heat/cold production
- Others (e.g. Agriculture, Forestry, Fisheries)

For the **adaptation part** (SECAP only), the identification of the sectors to increase the resilience in a city is highly contextual; some of the main sectors that can improve the resilience of cities include:

- Infrastructure
- Public Services
- Land Use Planning
- Environment & Biodiversity
- Agriculture & Forestry
- Economy

The idea presented here is to build up the “transnational template” starting from initiatives already defined at European level by the data specifications related to the INSPIRE Directive.

The conceptual model starts from the Data Specifications defined by the INSPIRE Directive as baseline, and considers all requirements and characteristics of energy data that partners provided.
Even though the implementation of INSPIRE data models is not the focus neither the goal of CitiEnGov they will be used as a starting point and as a common approach to get a common view and common semantics about energy-data.

[3] http://www.eumayors.eu/support/faq_en.html?id_faq=18

Therefore, the objective of this activity will be twofold:

1. a common conceptual data model, to be considered as a possible target schema for exporting and sharing data outside the local context and outside the organization;
2. a reference implementation, as SQL-based relational database (possibly for Oracle and PostGIS platforms)

It is noteworthy that the final goal is not to force CitiEnGov partners to change the way they use energy-related data internally, but to help them to generate a neutral and standardized semantics.

The importance of sharing the same semantics about energy-related data can be simply clarified with the following example: on March 2017, during a CitiEnGov videoconference (SIPRO, GOLEA, DEDAGROUP PUBLIC SERVICES) it was discussed a practical requirement coming from Slovenian regions, where data about energy consumptions are usually shared from utilities (data providers/custodians) and Public Authorities. Data about consumption are:

- temporally aggregated on annual basis
- divided by fuel (e.g. gas, electricity, DH, …)
- divided by "building" categories
- …

In the case of building "categories" GOLEA mentioned that they usually get these data divided in terms of "**uses of buildings**":

- residential
- industrial
- offices
- commerce
- …

Indeed, even though these categories are quite similar in different countries, often they do **not** have the same meaning.
That's why we need to look at **INSPIRE** in terms of **semantics** (and not merely in terms of Directive's principles, data requirements or technical specifications); semantics practically means that we already have some basic concepts like buildings' typologies, or (better) "**uses of buildings**" already defined by INSPIRE: http://inspire.ec.europa.eu/codelist/CurrentUseValue

The codelist above contains what INSPIRE conceives when we think of "**uses of buildings**". This codelist is:

- not closed, but can be **extended** (see for example this extended one: http://hub.geosmartcity.eu/registry/codelist/CurrentUseValue/ or even this second one http://www.locationframework.eu/codelist/CurrentUseValue.html or this third one http://inspire-sandbox.jrc.ec.europa.eu/codelist)
- available in different EU languages … therefore users can switch from English to German or Slovenian or Polish and get the clear definition of each value (e.g. http://inspire.ec.europa.eu/codelist/CurrentUseValue/publicServices) in national languages

Of course, this is just a simple example of what we mean when talking about "**semantics**" related to energy data.

In the deliverable DT1.2.1 project partners already shared a common definition of other “concepts” like:

- energy type (primary, estimated, final, …)
- energy source (biogas, natural gas, electricity, solid fuels, warm water o stream, …)
- heating systems (central heating, district heating, electric radiators, solar heating, stove, …)
- … etc

A first conceptual version of the data model has been provided to CitiEnGov partners in **July 2017**.
To facilitate the understanding and the further agreement of the conceptual model (by September 15th, 2017), CitiEnGov partners have been provided 2 different documents:

- PowerPoint slides, explaining the rationale of the proposed data model
- Excel spreadsheet, containing the list of classes/tables and their attributes needed to cover all possible aspects of “energy database” related to buildings, transport and public lighting

The data model consists of 3 main classes (that will be tables in the physical database implementation) corresponding to the 3 sectors the project is focused on:

- building
- transport
- installation (public light)

A physical implementation of the data model has been developed in CitiEnGov with a standard SQL structure provided to all CitiEnGov partners; the CitiEnGov SQL data model is available for the two spatial relational database platforms mostly used: Oracle and PostgreSQL/PostGIS.

1.1 The conceptual model

As aforementioned, the data model relies on the INSPIRE Data Specifications: for instance for the "Buildings" sector the Technical Guidelines considered are available at:
http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_BU_v3.0.pdf

The following tables contain the draft version of the conceptual model provided to partners.

Table 1 - Building

Attribute Name	Attribute Description	Data type	Notes
UUID	Primary ID	text string	Unique ID for the record (building unit , building or district)
UUID_BDG2BDG	Parent building ID	text string	The primary ID of the element (building unit , building or district) of which the record is part: e.g. a building

Attribute Name	Attribute Description	Data type	Notes
			unit pointing to the building it is part of.
GEOMETRY	Geomerty	Geometry	Mandatory. Can be either point, 2D polygon
IDENTIFIER_ID_NAME	Dataset namespace	text string	Allows to identify the records that refer to different sets of building data, e.g. the dataset of municipal buildings, the dataset of private buildings aggregated at district level, etc.
IDENTIFIER_ID_LOC	Dataset namespace ID	text string	An optional local ID to identify records within the same dataset namespace
EXT_REF_INF_SYS_NAME	External information system name	text string	Allows to link the record with an external information system where the same element is also listed: e.g. National Building Cadastre Database
EXT_REF_IDENTIFIER	External information system URL	text string	E.g. www.nationalcadastre.eu
EXT_REF_REFERENCE	External information system ID	text string	The ID that the element described in the record has in the external information system. This field could be used also to store the installation address.
NAME	Name of of the element in the record	text string	E.g. "Primary School 'Antonio Vivaldi'", or "City District 7"
LIFESPAN_BEGINNING	Date of record creation	date	Describes when the record was created

Attribute Name	Attribute Description	Data type	Notes
LIFESPAN_END	Date of record validity end	date	Optional. Describes when the information in the record stops to be valid.
BUILDINGTYPE	Architectural typology	CODELIST	E.g. Single-family house, Monument, Office building, Gas station, etc
ROOF_TYPE	Roof type	CODELIST	
DATE_C_BEGINNING	Construction year (begin)	integer number (yyyy)	If the exact construction year is known it should be written here. If only a reference period is known, write here the year at the beginning of the period.
DATE_C_END	Construction year (end)	integer number (yyyy)	Optional. If only a reference period is known, write here the year at the end of the period.
HEIGHT_HEIGHT_LOW	Lower reference for height value	CODELIST	Where the height is considered to be starting from: e.g. "from the ground level"
HEIGHT_HEIGHT_REF	Higher reference for height value	CODELIST	Where the height is considered to be finishing at e.g. "at the rooftop"
HEIGHT_HEIGHT_STAT	Heigth value type	text string	How was determined the height: measured, estimated, etc
HEIGHT_HEIGHT_VAL	Height value (m)	real number	
FLOORS	Number of floors	integer number	
H_FLOOR	Average floor height	real number	
UNITS	Number of building units	integer number	
USE_S	Predominant use	CODELIST	

Attribute Name	Attribute Description	Data type	Notes
OCCUPANTS	Number of occupants	integer number	
USE_HOURS	Hours of use per day	real number	Average over the behaviour of all the occupants along all the days of the week
OWNERSHIP	Ownership	CODELIST	
VOLUME_SOURCE	Volume type	text string	E.g. "external geometrical volume", "heated volume", etc
VOLUME_VALUE	Volume (m^3)	real number	
SURFACE_SOURCE	Surface type	text string	E.g. "usable surface", "heated surface", etc
SURFACE_VALUE	Surface (in m^2)	real number	
VERT_SURF	External vertical surface (in m^2)	real number	The extent of the vertical surface of the building or unit that is exchanging energy with the environment
VERT_SURF_TO_VOL	Surface to Volume ratio (m^2/m^3)	real number	
ENVELOPE_Q	Average envelope transmittance ($W/m^2°K$)	real number	
DATE_R_BEGINNING	Renovation year (begin)	integer number (yyyy)	If the exact renovation year is known it should be written here. If only a reference period is known, write here the year at the beginning of the period.
DATE_R_END	Renovation year (end)	integer number (yyyy)	Optional. If only a reference period is known, write here the year at the end of the period.
ENERGYPERFORMANCE_PERF_DATE	Energy performance date	date	
ENERGYPERFORMANCE_PERF_VALUE	Energy performance value	real number	

Attribute Name	Attribute Description	Data type	Notes
ENERGYPERFORMANCE_PERF_UOM	Energy performance unit of measure	CODELIST	
ENERGYPERFORMANCE_PERF_CLASS	Energy performance class	text string	
ENERGYPERFORMANCE_PERF_METHOD	Energy performance method	text string	

Table 2 - Building' materials (1:n)

Attribute Name	Attribute Description	Data type	Notes
UUID_ENV2BDG	Parent building ID	text string	The primary ID of the element (building unit , building or district) to which the material refers
ENV_ELEMENT_TYPE	Envelope element type	CODELIST	E.g. roof, wall, ground, windows
ENV_ELEMENT_MAT	Material	text string	
ENV_ELEMENT_Q	Trasmittance (W/m^2°K)	real number	
ENV_ELEMENT_RENOV	Renovation year	integer number (yyyy)	Optional, only if known

Table 3 - Building's uses (1:n)

Attribute Name	Attribute Description	Data type	Notes
UUID_USE2BDG	Parent building ID	text string	The primary ID of the element (building unit , building or district) to which the use refers
USE_M_CURRENTUSE	Current use	CODELIST	
USE_M_PERCENTAGE	Current use percentage	real number	Percentage with respect to the total spectrum of uses of the element (building unit , building or

Attribute Name	Attribute Description	Data type	Notes
			district) to wich this record refers

Table 4 - Installation

Attribute Name	Attribute Description	Data type	Notes
UUID	Primary ID	text string	Unique ID for the installation
UUID_INST2BDG	Parent building ID	text string	The primary ID of the element (building unit , building or district) to which the installation refers
GEOMETRY	Geomerty	Geometry	Mandatory. Can be either point, 2D or 3D polygon
IDENTIFIER_ID_NAME	Dataset namespace	text string	Allows to identify the records that refer to different sets of installation data, e.g. the dataset of municipal thermal plants, the dataset of private district network substations, etc.
IDENTIFIER_ID_LOC	Dataset namespace ID	text string	An optional local ID to identify records within the same dataset namespace
EXT_REF_INF_SYS_NAME	External information system name	text string	Allows to link the record with an external information system where the same element is also listed: e.g. National HVAC Cadastre Database
EXT_REF_IDENTIFIER	External information system URL	text string	E.g. www.nationalHVACcadastre.eu
EXT_REF_REFERENCE	External information system ID	text string	The ID that the element described in the record has in the external information system. This field could be used also to store the installation address.
NAME	Name of of the element in the record	text string	E.g. "Central Heating plant for Primary School 'Antonio Vivaldi'"

Attribute Name	Attribute Description	Data type	Notes
LIFESPAN_BEGINNING	Date of record creation	date	Describes when the record was created
LIFESPAN_END	Date of record validity end	date	Optional. Describes when the information in the record stops to be valid.
INSTALLATION_NATURE	Type of installation	CODELIST	E.g. heating (boiler, plant, etc), cooling, ventilation, photovoltaic panel, solar thermal panel, energy meter, lamp, light line, etc
DATE_ACTIVATION	Begin of installation activity	date	If a precide date is unknown, just the year is enough
POWER	Maximum power (kW)	real number	Optional, only if the installation has a maximum power (e.g. heating system)
SURFACE	Collecting surface (m^2)	real number	Optional, only if the installation has a relevant surface (e.g. photovoltaic panels)
LAMP_TYPE	Lamp type	text string	Optional, only if the installation has a lamp type (e.g. public/private lighting)
DATE_C_BEGINNING	Construction year (begin)	integer number (yyyy)	If the exact construction year is known it should be written here. If only a reference period is known, write here the year at the beginning of the period.
DATE_C_END	Construction year (end)	integer number (yyyy)	Optional. If only a reference period is known, write here the year at the end of the period.
DATE_R_BEGINNING	Renovation year (begin)	integer number (yyyy)	If the exact renovation year is known it should be written here. If only a reference period is known, write here the year at the beginning of the period.
DATE_R_END	Renovation year (end)	integer number (yyyy)	Optional. If only a reference period is known, write here the year at the end of the period.

Table 5 - Transport

Attribute Name	Attribute Description	Data type	Notes
UUID	Primary ID	text string	Unique ID for the transport type record
GEOMETRY	Geomerty	Geometry	Optional. Can be either point or 2D
TRANSPORTS_TYPE	Transport type	text string	E.g. municipal fleet, public transport, private/commercial transport, electric/hybrid transport, etc
TRANSPORTS_NVEH	Number of vehicles	integer number	Number of vehicles (measured or estimated) belonging to the specific transport type
TRANSPORTS_RENOV	Vehicle yearly renovation rate	real number	The ratio of renewd vehicles per yearto the total number of vehicles of the specific transport type

Table 6 - Energy amount

Attribute Name	Attribute Description	Data type	Notes
UUID	Primary ID	text string	Unique ID for the energy record
UUID_JOIN	Join ID	text string	The primary ID of the element (of table Building, Installations or Transports) to which the energy consumption / production refers
UUID_JOIN_CLASS	Reference to join class	CODELIST	Buildings, Installations or Transports

Attribute Name	Attribute Description	Data type	Notes
ENERGYAMOUNT_E_YEAR	Date (begin)	date	If the energy measurement refers to an istantaneous moment in time (e.g. a meter reading) write the timestamp here. If the measurement refers to an interval in time (e.g. the energy consumed in a month), write the start date of the interval here.
ENERGYAMOUNT_E_YEAR_END	Date (end)	date	Optional. Use it if the measurement refers to an interval in time (e.g. the energy consumed in a month) to write the end date of the interval.
ENERGYAMOUNT_E_AMOUNT	Energy value	real number	For buildings and installations: The amount of energy consumed or produced For transports: The amount of fuel consumed
ENERGYAMOUNT_E_UOM	Energy unit of measure	CODELIST	
ENERGYAMOUNT_E_SOURCE	Energy carrier	CODELIST	Energy carrier /Fuel type E.g. electricity, hot water (district heating), GPL, Gasoline, Natural gas, etc
ENERGYAMOUNT_E_USE	Energy use	CODELIST	E.g. production (renewables), heating, cooling, ventilation, domestic hot water, illumination, domestic appliances, transportation, etc
ENERGYAMOUNT_E_TYPE	Energy type	CODELIST	E.g. primary, final, demand, estimated To distinguish between energy production (at plants, primary), consumption (final), needs (demand) and simulations (estimated).
ENERGYAMOUNT_E_TARIFF	Energy tariff	text string	Energy tariff or fuel price
ENERGYAMOUNT_E_METHOD	Accounting method	text string	How the consumption is determined. E.g. for transportation: direct measurement, estimated from total Km, etc E.g. for buildings/intallations: from bills, from smart meters, etc
ENERGYAMOUNT_ESTIMATEDCO2	CO2 equivalent	real number	

1.2 The physical implementation

As aforementioned, the physical implementation of the data model will be a reference implementation based on two different platforms mostly used: Oracle and PostgreSQL/PostGIS.

The physical implementation of the CitiEnGov harmonized data model will be used to populate the database with data already available at partners' premises or collected during the CitiEnGov project.

These data will be transformed by CitiEnGov partners using ETL (Extract, Transform, Load) tools. Different options do exist to achieve this data transformation:

- using SQL or PL/SQL (or PL/pgSQL) scripting language
- Kettle software
- FME software
- HALE software

The physical data model will be provided to partners containing the following SQL statements:

- **CREATE** statements for all tables of the "SCC solutions database" in SQL creates an object in a relational database management system (RDBMS). In the SQL 1992 specification[4], the types of objects that can be created are schemas, tables, views, domains, character sets, collations, translations, and assertions. Many implementations extend the syntax to allow creation of additional objects, such as indexes and user profiles.
- **ALTER** statements to add constraints related to Primary Keys; in SQL changes the properties of an object inside of a relational database management system (RDBMS).
- **INSERT INTO** statements, used to insert new records in a tables corresponding to codelists; the INSERT specifies both the column names and the values to be inserted.

The SQL scripts for creating tables in both Oracle and PostGIS platform are available on the CitiEnGov online toolkit:The SQL scripts for creating tables in both Oracle and PostGIS platform are available on the CitiEnGov online toolkit:

http://toolkit.citiengov.eu/index.php?title=Transnational_methodology#Physical_implementation_of_data_model

[4] http://www.contrib.andrew.cmu.edu/~shadow/sql/sql1992.txt

2. ICT services to share energy data

The sharing of energy-related data will rely on the deployment of web **geo-ICT services** based on open standards.

These web services will span from catalogue services for browsing and searching data in distributed metadata catalogues, to services for visualizing or accessing data.

Client applications that will be implemented by CitiEnGov partners to present energy-related data (e.g. portals) need to use these web services directly by connecting them with standard interfaces/protocols.

Data services are services related to data **ingestion**, **management**, **view** and **access**; from the data provided/publisher point of view (and also according to the ISO19119 taxonomy), the data services can be grouped in the following macro-categories:

- discovery services
- viewing services
- access services configuration (download)
- processing services (subsetting, ordering, filtering)

These web **geo-ICT services** based may be implemented using proprietary solutions like Esri ArcGIS Server (http://server.arcgis.com/en/) or open source ones like GeoServer (http://geoserver.org/).
It is crucial that the solution chosen by the partner is implementing open standard protocols like the ones mentioned hereafter.

2.1 Discovery services

The discovery of energy datasets is usually performed through searching functionalities in metadata catalogues; metadata describe the general characteristics of each dataset, independently from the distribution formats or from the availability of services that operate on the dataset.

One dataset, being a geographical one or tabular or other, may have different representations; in the case of geographical data, the "discovery metadata" may provide a general but structure description (responsible parties, dates, licenses, lineage, …) and refer to one or more "resources".

For instance, a metadata regarding a geographical dataset may refer to one or more of the following "resources" in different possible formats and standard protocols:

- a CSV or XLS formatted file containing the tabular representation of data
- a ZIP file containing vector representation of data (e.g. SHP with DBF for attributes), to allow Geographic Information Systems' users to easily work on simple flat datasets
- a KML encoded file, for being represented in Google Earth or other 3D / globe viewers

- a GML encoded file, in case of complex spatial data to be provided in an interoperable and open standard format
- a web service conformant to OGC WMS standard interface, to allow the visualisation of maps in web or desktop map viewers
- a web service conformant to OGC WFS standard interface service, with dynamic outputs based on the same formats (SHP/ZIP, KML, GML, …) so to allow the downloading of subsets of data based on filters, or for the downloading of frequently updated data

2.2 View services

Since sometimes data visualization may be misunderstood as data access, it may be appropriate to highlight here the principle differences:

- **accessing** data involves the possibility of querying, sub-setting and filtering (it's a necessary condition, but not sufficient since certain view services have the capabilities of expressing a filter);
- accessing data necessarily use a physical data format, but does not depend on it; the representation of data instead is an integral part of viewing services;
- very often in **viewing** services, the representation of data completely hides underlying data making it impossible to recover them (approximations, portrayal, simplifications, generalization, aggregation etc.; usually these are part of the viewing service).
- data coming from an access service can be subsequently **elaborated** without loss or without the need of particular pre-elaboration.

CitiEnGov partners may expose services for view energy-related data via web services based on well-known APIs for representing tabular data, or through WMS / WMTS protocols defined by the Open Geospatial Consortium (OGC) for maps.

For spatial data, **viewing** means producing an image from the data applying a set of rendering rules, otherwise viewing a classic alphanumeric dataset can be achieved producing a tabular representation or a graphical one. The different infrastructural components are optimized to treat the different type of data and this results in various protocols and standards used in the data services.

In the same way, **accessing** data can have several implementations: WFS for spatial data, CSV for tabular one, SPARQL endpoint for linked (see the following section).

The CitiEnGov partners may also offer functionalities to let clients visualise:

- tabular data, with filtering/searching capabilities to extract or sort subset of datasets
- graphics (dashboards), based on open source Javascript libraries to render statistical data with high quality diagrams and presentation styles

2.3 Download services

In the context of CitiEnGov project, different representations of energy data are foreseen:

- **tabular data**, with records and rows to present data in CSV, XLS or other formats
- **geographic vector**, with spatial features representing buildings, transport networks or public lighting with vectors
- **geographic coverage**, with raster images of spatial phenomena (e.g. energy production may be provided as spatial data in the form of a raster layer, with regular grid containing cells with different values of energy consumption)
- **geographic sensor**, with near real-time data coming from sensors (e.g. energy consumption at single municipal buildings level)

As per INSPIRE definitions, a download service for vector geographic data is equivalent to a web service implementing the OGC WFS standard interface; the intention being that the user is given access to the raw data values instead of a cartographic representation as is the case with e.g. WMS requests that only return a map image.

Access to the raw data enables two key benefits:

1. the ability to perform calculation and analysis using the vector geometries or raster cell data
2. the ability to draw non-pixelated map images at all scales using client-side rendering

CitiEnGov partners may implement an extended set of download services that goes beyond the INSPIRE requirements. Each extension provides a specific performance benefit and the total implementation includes the following protocols and formats.

Service protocol	Data format, transport	Benefits
Web Feature Service (WFS)	GML/XML, GeoJSON, CSV	Provides interoperable methods to access and work with remote spatial data sources.
SPARQL	RDF/XML, RDF/JSON	Provides a basis for easy extension of any dataset through RDF triple assertion. Provides Linked Data publishing.

Custom vector data service	TileJSON	The vector equivalent of tile map services for raster data. To remove the overhead of clipping custom extents for vector data, tiles are pre-generated. Client applications can buffer neighboring tiles into memory in order to provide smooth panning experiences.
Custom table data service	WebCSV, JSON, XML	This type of service can provide access to non-spatial tabular data in one of the three formats listed. WebCSV is the lightest format but has limited support in client libraries. JSON has relatively low overhead and is widely supported by browser based end-user clients. Finally, XML is very easy to parse using any software technology despite a significant markup overhead.

2.4 Processing services

In the context of CitiEnGov, processing services are "partner-driven" web services linked to the detailed requirements coming from each partner in terms of data processing and user engagement.

Several use cases aim to perform e.g. calculations on data about buildings, transport network, public lighting. This relies, of course, on well-known data models that contain the information that is required to run the appropriate equations/algorithms. Data will be read from the partner data store and will be consumed by the processing service where the actual analysis code is implemented.

Therefore, the processing services may be a set of independent end-user applications that will consume their business logic via the APIs and (optionally) the client-side JavaScript libraries implemented at partners' level.

The following table summarizes a list of possible operations that can be performed for different categories of processing services:

Table 7 - Categories of processing services

Type of data	Visualization	Querying	Processing
Non-spatial graph	X	X	N/A
Non-spatial table	X	X	N/A
Spatial graph	X	X	Proximity
Spatial raster	X	N/A	N/A
Spatial table	X	X	Proximity, overlay
Spatial table: building	X	X	Energy performance
Spatial table: network	X	X	Route calculation
Spatial table: point	X	X	Interpolation

3. Technical references for services

This final chapter contains the technical references about interfaces, versions, operations, etc. required at server or client levels.
Indeed, the details of these technical references are based on previous EU projects (e.g. eENVplus, GeoSmartCity) available at the deliverables public access pages.

Technical references are divided in three main sections:

- **client**: set of requirements related to client software (desktop or web) directly used by human beings to search/discover, view, access energy-related data

- **server**: set of requirements related to server components, to be made available at partners' level
- **interface**: set of requirements related to standard interfaces and protocols to be considered at client and/or server side levels to guarantee interoperability

3.1 Metadata catalogue services (CS)

Table 8 - Technical details for metadata catalog services - client

ID	DESCRIPTION
1.	CS clients shall allow users to search the catalogue either through hierarchical browsing, or by expressing queries with a combination of parameters defined by the user through the controls (edit boxes, select boxes, etc.) available in the discovery GUI
2.	CS clients shall enable the user to filter search results by selecting / setting a value for the following features: • INSPIRE themes (e.g., Addresses, Administrative Units, etc.) or other topic categories • Country • Resource type (e.g. dataset, layer, service, series) • Service type (e.g. discovery, view, download, invoke, transformation, other) • Spatial resolution • Original language • Access constraint • Responsible organization • Timestamp • Defined timeframe (decade, year, or month, based on “Creation date”, “Temporal extent”, “Publication Date”, “Revision Date”, or all dates) • Bounding box (geographical constraint) • Geographical name (based on gazetteer)
3.	CS client shall allow users to sort the results by various criteria such as name and date or geographic ranking, which is based on a similarity function between the area of the rectangle of the found items and the geographical constraint.
4.	In the case of discovery of viewable layers, CS client shall allow users to visualise them on a geographic viewer (View Service clients), which enables the user to browse these layers over a background map (pan / zoom) and switch on/off the visibility of the layers
5.	CS clients shall allow users to add and combine layers to the map (geographic viewer) in the following ways: • from a search of the Geoportal’s Catalogue Service • from local (cached) resources
6.	CS Client shall perform automatic multi-lingual keyword suggestions in the GUI (concept of interactive discovery and auto-completion) given by the internal vocabulary and code-list, using GEMET and additional thesauri (e.g. eENVplus Thesaurus Framework)

Table 9 - Technical details for metadata catalog services - server

ID	DESCRIPTION

7.	CS server shall be able to return a list of identifiers for corresponding features for a request expressed in an OGC query language (aka “Discovery.GetRecords” in CSW specification).
8.	CS server shall be able to return associated meta information instances of features selected by user, managed by the catalogue and returned by a previous query (getRecordById).
9.	CS server shall provide a mean that allows harvesting resources over the network (Manager.Harvest).
10.	CS server shall provide the ability to push information into the catalogue. This operation receives the meta information to be stored and returns information about the update of the catalogue (Manager.Transaction).
11.	CS server shall accept an area of interest (geographic zone definition) and a time period in input parameters
12.	CS server shall accept the lower limit and the maximum number of results by catalogue
13.	CS server shall check metadata inserted, updated, or harvested.
14.	CS server shall allow search in distributed catalogues.

Table 10 - Technical details for metadata catalog services - interface

ID	DESCRIPTION
15.	Discovery services shall support one of the following interfaces to make their data discoverable: • OGC CSW 2.0.2 ISO Application Profile compliant • DCAT-AP • CKAN API

3.2 View services (VS)

Table 11 - Technical details for view services - client

ID	DESCRIPTION
1.	VS client shall allow users to view and browse a map (zoom, pan)
2.	VS client shall allow users to query and select data
3.	VS client shall allow users to access Table of Contents and switch layers on/off
4.	VS client shall allow users to retrieve maps and/or data provided by remote nodes, adding layers as: • OGC WMS • OGC WFS • OGC WCS
5.	VS client shall allow users to add layers to the map in the Table of Content (ToC) in the following ways: • from a URL (OWS GetCapabilities) • from a search on Catalogue Services • from local resources (local file systems or RDBMS)
6.	VS client shall allow users to remove layers from the ToC
7.	VS client shall allow users to change the portrayal of a layer, changing Styled Layer Descriptor (SLD) documents available for the layer itself
8.	VS client shall allow users to add and apply external Styled Layer Descriptor (SLD) documents to layers
9.	VS client shall allow users to remove Styled Layer Descriptor (SLD) documents out from layers
10.	VS client shall allow users to change the opacity of each layer
11.	VS client shall allow users to export and save locally the context of the map in the following graphic formats: • PNG • GIF • JPEG • PDF • GeoTIFF
12.	VS client shall allow users to export and save locally the context of the map as hyperlink (permalink)
13.	VS client shall allow users to export and save locally vector data as GML file
14.	VS client shall allow users to export and save locally vector data as OGC KML file

15.	VS client shall allow users to export and save locally vector data as SHP file
16.	VS client shall allow users to change the Coordinate Reference System (CRS) of the map. The minimum set of CRSs available to the user shall be the one defined in INSPIRE Data Specifications.
17.	VS client shall allow users to change the CRS of the data exported (by means of Coordinate Transformation Service)
18.	VS client shall allow users to view metadata of selected layers by accessing metadata through the CSW interface of Catalogue Service
19.	VS client shall allow users to view non-geographical information associated to layers, by clicking on the map (WMS GetFeatureInfo operation).

Table 12 - Technical details for view services - server

ID	DESCRIPTION
20.	VS server shall support the following operations (OGC WMS): • GetCapabilities • GetMap • GetFeatureInfo
21.	VS server shall allow the loading and serving at least one of the following vector data formats: • PostGIS • GML (also according to INSPIRE Technical Guidelines) • ESRI Shapefile
22.	VS server shall allow the loading and serving maps in the following formats: • PNG • JPEG • TIFF • GeoTIFF
23.	VS server shall support cascading Web Feature Service (WFS)
24.	VS server shall allow the representation of attributes values related to geographical phenomena as choropleth maps; choropleth maps are thematic maps in which areas are shaded or patterned in proportion to the measurement of the statistical variable being displayed on the map, such as population density or per-capita income
25.	VS server shall allow the representation of attributes values related to geographical phenomena as external graphics overlaid to the map

Table 13 - Technical details for view services - interface

ID	DESCRIPTION

26.	VS shall comply with the "Commission Regulation (EC) No 976/2009 of 19 October 2009 implementing Directive 2007/2/EC of the European Parliament and of the Council as regards the Network Services" http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:32009R0976
27.	VS shall comply with with INSPIRE View Service technical guidance http://inspire.jrc.ec.europa.eu/documents/Network_Services/...
28.	VS shall support OGC SLD specification http://www.opengeospatial.org/standards/sld
29.	VS shall support OGC FE specification http://www.opengeospatial.org/standards/filter
30.	VS shall support CQL specification http://www.loc.gov/standards/sru/cql/spec.html
31.	VS shall support OGC GML specification (3.2.1) http://www.opengeospatial.org/standards/gml
32.	VS client shall support functionalities to invoke Catalogue Service (INSPIRE Discovery Service Technical Guidelines)

3.5 Download services - tabular data (DS-tab)

Table 14 - Technical details for download services (tabular data) - client

ID	DESCRIPTION
1.	DS-tab client shall allow users to extract (portions of) tabular data from DS-tab services and provide functionalities to download them locally.
2.	DS-tab client shall be able to compose queries using SQL and/or other filtering functionalities.
3.	DS-tab client shall allow users to sort tabular data on one or more attributes.

Table 15 - Technical details for download services (tabular data) - server

ID	DESCRIPTION
4.	DS-tab server shall be able to perform queries and provide subset of dataset as results of filtering operation.

5.	DS-tab server shall be able provide tabular data in the original format or in an equivalent one (CSV)

Table 16 - Technical details for download services (tabular data) - interface

ID	DESCRIPTION
6.	DS-tab shall provide well documented APIs for reading, searching and filtering data without the need to download the entire dataset
7.	DS-tab shall accept a JSON dictionary in an HTTP POST request to an API URL, and the API shall return its response in a JSON dictionary

3.3 Download services - geographic vector data (DS-vect)

Table 17 - Technical details for download services (geographic vector data) - client

ID	DESCRIPTION
1.	DS-vect client shall allow users to extract features from DS-vect services and provide functionalities to download (subset of) datasets locally.
2.	DS-vect client shall be able to compose queries using CQL and/or OGC FE filters.
3.	DS-vect client shall allow users to change the Coordinate Reference System (CRS) of the map. The minimum set of CRSs available to the user shall be the one defined in INSPIRE Data Specifications.

Table 18 - Technical details for download services (geographic vector data) - server

ID	DESCRIPTION
4.	DS-vect shall support operations (OGC WFS): • GetCapabilities • DescribeFeatureType • GetFeature

5.	DS-vect shall inform the client about the common and specific capabilities of a download access service.

Table 19 - Technical details for download services (geographic vector data) - interface

ID	DESCRIPTION
6.	DS-vect shall comply with the "Commission Regulation (EC) No 1088/2010 of 23 November 2010 amending Regulation (EC) No 976/2009 as regards download services and transformation services" http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:02009R0976-20101228
7.	DS-vect shall comply with the INSPIRE Download Service technical guidance http://tinyurl.com/no33tnj
8.	DS-vect shall support OGC FE specification http://www.opengeospatial.org/standards/filter
9.	DS-vect shall support CQL specification http://www.loc.gov/standards/sru/cql/spec.html
10.	DS-vect shall support OGC GML specification (3.2.1) http://www.opengeospatial.org/standards/gml
11.	DS-vect client shall support functionalities to invoke Catalogue Service (INSPIRE Discovery Service Technical Guidelines)

3.4 Download services - geographic coverage data (DS-cov)

Table 20 - Technical details for download services (geographic coverage data) - client

ID	DESCRIPTION
1.	DS-cov client shall allow users to extract (portions of) grid coverages from DS-cov services and provide functionalities to download them locally.
2.	DS-cov client shall be able to compose queries using CQL and/or OGC FE filters.
3.	DS-cov client shall allow users to change the Coordinate Reference System (CRS) of the map. The minimum set of CRSs available to the user shall be the one defined in INSPIRE Data Specifications.

Table 21 - Technical details for download services (geographic coverage data) - server

ID	DESCRIPTION
4.	DS-cov server shall support the following operations (OGC WCS): • GetCapabilities • GetCoverage • DescribeCoverage
5.	DS-cov server shall allow the loading and serving of the following raster data formats: • PNG • JPEG • TIFF
6.	DS-cov server shall support reprojection, subsetting, format transcoding, subsampling on the fly.

Table 22 - Technical details for download services (geographic coverage data) - interface

ID	DESCRIPTION
7.	DS-cov shall implement at least OGC WCS 1.0 and optionally WCS 1.1.2
8.	DS-cov should optionally implement WCS extension for CF-netCDF 3.0 encoding (0.2.2) http://www.opengeospatial.org/standards/wcs

3.4 Download services - sensor data (DS-sens)

Table 33 - Technical details for sensor data streaming services - client

ID	DESCRIPTION
1.	The DS-sens clients shall allow the users to specify a Bounding Box spatial filter in a Map
2.	The DS-sens clients shall allow the users to determine a time window filter with a (From, To) datetime fields.

3.	The DS-sens clients shall allow the users to filter specific sensors (or in OGC – SOS terms, filtering by procedures or offerings).
4.	The SSS clients shall allow the users to filter specific type of observation property.
5.	The DS-sens clients shall allow the users to request existing observations for the content, spatial and temporal filters defined in previous requirements.
6.	The DS-sens clients shall provide a mechanism to subscribe to streaming generated values defined with previously specified filters.
7.	The DS-sens clients shall allow the user to explore the time series observation results for a single sensor.
8.	The DS-sens clients shall show the available features of interest in a map

Table 34 - Technical details for data streaming services - server

ID	DESCRIPTION
9.	DS-sens shall support the following OGC SOS profiles: • GetCapabilities • DescribeSensor • GetObservation • InsertSensor • DeleteSensor • InsertObservation • Spatial Filtering • EnhancedExtension • GetFeatureOfInterest
10.	DS-sens shall support the following format Bindings: • XML Encoding • KVP Binding
11.	DS-sens shall provide a mean to subscribe to streaming data in order to avoid clients to develop a mechanism to decide whether new data is available.
12.	DS-sens shall be able to work with different RDBMS systems, such as Postgresql, MS SQL Server, Oracle and mySQL.
13.	DS-sens shall comply with INSPIRE directive, by providing required extensions to OGC SOS, such as: • Include CRS as a parameter.

	• Enable multilingual support • Allow the request of service metadata
14.	DS-sens shall provide a mechanism to connect the service to underlying SCADA storage systems. This way sensor data does not need to be duplicated.

3.6 Processing services - geographic (PS)

Table 23 - Technical details for processing services (geographic) - client

ID	DESCRIPTION
1.	The PS clients shall allow the users to specify the (mandatory and optional) input parameters of the process.
2.	PS clients shall allow chaining two or more processes to facilitate the creation of repeatable workflows.
3.	PS clients shall allow the user to access and visualise the results of the process. The information returned typically contains URL references to output dataset.

Table 24 - Technical details for processing services (geographic) - server

ID	DESCRIPTION
4.	PS services shall support the following operations (OGC WPS): • GetCapabilities • DescribeProcess • Execute
5.	If applicable, PS shall minimally support the following raster formats for input and output: • GeoTIFF • JPEG2000
6.	The target coordinate reference system used by PS shall be provided as a valid EPSG-code.
7.	If applicable, the PS which manipulate vector spatial data should minimally support the following formats as input and output: • ESRI Shapefile

	• CSV • GML 3.2.1 • KML • PostGIS (local) • Oracle spatial (local)
8.	The following feature types should be supported by PS: • point • line string • polygon
9.	The PS shall allow receiving input data files either: • by reference, through FTP and HTTP URLs • by reference, providing the parameters to obtain the input dataset from an OGC Web Service (e.g. for WFS input parameters would consist of base URL, feature Type and filter statement). • by reference, selecting the path of the resources on client side
10.	PS shall inform the client about the common and specific capabilities of a process service.
11.	PS shall inform the client about the process status.
12.	PS shall load information on available processes from a configuration file or from a database.
13.	PS shall allow to wrap existing geoprocessing functionality and existing tools to enable them to be served using WPS, by providing XML that meets the interface requirements of the OGC WPS communication schema.
14.	PS shall store the results as web-accessible URLs or embedded in the response.

Table 25 - Technical details for processing services (geographic) - interface

ID	DESCRIPTION
15.	PS shall comply with the OGC WPS specification

	http://www.opengeospatial.org/standards/wps

Annex A - List of acronyms

Table 26 - Acronyms and definitions

Term	Description
API	In computer programming, an Application Programming Interface (API) is a set of routines, protocols, and tools for building software applications. An API expresses a software component in terms of its operations, inputs, outputs, and underlying types.
CityGML	CityGML is a common information model for the representation of sets of 3D urban objects. It defines the classes and relations for the most relevant topographic objects in cities and regional models with respect to their geometrical, topological, semantical and appearance properties. Included are generalization hierarchies between thematic classes, aggregations, relations between objects, and spatial properties.
CSV	A comma-separated values (CSV) (also sometimes called character-separated values) file stores tabular data (numbers and text) in plain-text form. Plain text means that the file is a sequence of characters, with no data that has to be interpreted as binary numbers.
DCAT	DCAT is an RDF vocabulary designed to facilitate interoperability between data catalogs published on the Web. This document defines the schema and provides examples for its use. By using DCAT to describe datasets in data catalogs, publishers increase discoverability and enable applications easily to consume metadata from multiple catalogs. It further enables decentralized publishing of catalogs and facilitates federated dataset search across sites. Aggregated DCAT metadata can serve as a manifest file to facilitate digital preservation.
DCAT-AP	The DCAT Application profile for data portals in Europe (DCAT-AP) is a specification based on the Data Catalogue vocabulary (DCAT) for describing public sector datasets in Europe. Its basic use case

Term	Description
	is to enable cross-data portal search for data sets and make public sector data better searchable across borders and sectors.
EPSG	The European Petroleum Survey Group (EPSG) is a structured dataset of Coordinate Reference Systems and Coordinate Transformations, accessible through an online registry (www.epsg-registry.org) or, as a downloadable zip file.
ETL	In computing, Extract, Transform and Load (ETL) refers to a process in database usage and especially in data warehousing that: - Extracts data from homogeneous or heterogeneous data sources - Transforms the data for storing it in proper format or structure for querying and analysis purpose - Loads it into the final target (database, more specifically, operational data store, data mart, or data warehouse)
FTP	The File Transfer Protocol (FTP) is a standard network protocol used to transfer computer files from one host to another host over a TCP-based network, such as the Internet.
GeoJSON	GeoJSON[1] is an open standard format for encoding collections of simple geographical features along with their non-spatial attributes using JavaScript Object Notation. The features include points (therefore addresses and locations), line strings (therefore streets, highways and boundaries), polygons (countries, provinces, tracts of land), and multi-part collections of these types. GeoJSON features need not represent entities of the physical world only; mobile routing and navigation apps, for example, might describe their service coverage using GeoJSON.
GET (HTTP)	Requests using GET should only retrieve data and should have no other effect. (This is also true of some other HTTP methods.)
GML	The Geography Markup Language (GML) is the XML grammar defined by the Open Geospatial Consortium (OGC) to express geographical features. GML serves as a modelling language for

Term	Description
	geographic systems as well as an open interchange format for geographic transactions on the Internet.
GTFS	General Transit Feed Specification that defines a common format for public transportation schedules and associated geographic information.
HTML	HyperText Mark-up Language, originally a subset of SGML
HTML5	HTML5 is a core technology markup language of the Internet used for structuring and presenting content for the World Wide Web. As of October 2014 this is the final and complete fifth revision of the HTML standard of the World Wide Web Consortium (W3C).
HTTP	The Hypertext Transfer Protocol (HTTP) is an application protocol for distributed, collaborative, hypermedia information systems.[1] HTTP is the foundation of data communication for the World Wide Web. Hypertext is structured text that uses logical links (hyperlinks) between nodes containing text. HTTP is the protocol to exchange or transfer hypertext.
JSON	JavaScript Object Notation; a text based data representation format
KML	Keyhole Markup Language (KML) is an XML notation for expressing geographic annotation and visualization within Internet-based, two-dimensional maps and three-dimensional Earth browsers.
OGC	Open Geospatial Consortium, an industry standardization organization
Oracle	Oracle Database (commonly referred to as Oracle RDBMS or simply as Oracle) is an object-relational database management system produced and marketed by Oracle Corporation.

Term	Description
OSM	OpenStreetMap, a collaborative high-resolution road centric global base map database
PDF	Portable Document Format – Adobe format for accurate reproduction of documents.
POST (HTTP)	In computing, POST is one of many request methods supported by the HTTP protocol used by the World Wide Web. The POST request method is designed to request that a web server accepts the data enclosed in the request message's body for storage.
PostgreSQL	PostgreSQL, often simply Postgres, is an object-relational database management system (ORDBMS) with an emphasis on extensibility and on standarDS-covompliance. As a database server, its primary function is to store data securely, supporting best practices, and to allow for retrieval at the request of other software applications.
PostGIS	PostGIS is an open source software program that adds support for geographic objects to the PostgreSQL object-relational database. PostGIS follows the Simple Features for SQL specification from the Open Geospatial Consortium (OGC).
RDF	The Resource Description Framework (RDF) is a family of World Wide Web Consortium (W3C) specifications originally designed as a metadata data model. It has come to be used as a general method for conceptual description or modeling of information that is implemented in web resources, using a variety of syntax notations and data serialization formats. It is also used in knowledge management applications.
REST, RESTful	REpresentational State Transfer, an architectural model for implementation of Web Service APIs
SHP	The shapefile format is a popular geospatial vector data format for geographic information system (GIS) software. It is developed and regulated by Esri as a (mostly) open specification for data interoperability among Esri and other GIS software products.

Term	Description
SLD	A Styled Layer Descriptor (SLD) is an XML schema specified by the Open Geospatial Consortium (OGC) for describing the appearance of map layers. It is capable of describing the rendering of vector and raster data. A typical use of SLDs is to instruct a Web Map Service (WMS) of how to render a specific layer.
SPARQL	SPARQL is an RDF query language, that is, a semantic query language for databases, able to retrieve and manipulate data stored in Resource Description Framework format. It was made a standard by the RDF Data Access Working Group (DAWG) of the World Wide Web Consortium, and is recognized as one of the key technologies of the semantic web.
SSN	Semantic Sensor Network ontology. OWL 2 ontology created by W3C used to describe sensors and observations — the SSN ontology, available at http://purl.oclc.org/NET/ssnx/ssn. The SSN ontology can describe sensors in terms of capabilities, measurement processes, observations and deployments
SWE	The OGC's Sensor Web Enablement (SWE) standards enable developers to make all types of sensors, transducers and sensor data repositories discoverable, accessible and useable via the Web. Exif:
SWIG	SWIG stands for Simplified Wrapper and Interface Generator, ant it is a software development tool that connects programs written in C and C++ with a variety of high-level programming languages. SWIG is used with different types of target languages including common scripting languages such as Javascript, Perl, PHP, Python, Tcl and Ruby.
SVG	Scalable Vector Graphics (SVG) is an XML-based vector image format for two-dimensional graphics with support for interactivity and animation. The SVG specification is an open standard developed by the World Wide Web Consortium (W3C) since 1999.

Term	Description
UTF-8	UTF-8 (U from Universal Character Set + Transformation Format—8-bit[1]) is a character encoding capable of encoding all possible characters (called code points) in Unicode.
URL	A uniform resource locator (URL) is a reference to a resource that specifies the location of the resource on a computer network and a mechanism for retrieving it. A URL is a specific type of uniform resource identifier (URI).
UX	User eXperience involves a person's behaviours, attitudes, and emotions about using a particular product, system or service. User experience includes the practical, experiential, affective, meaningful and valuable aspects of human–computer interaction and product ownership. Additionally, it includes a person's perceptions of system aspects such as utility, ease of use and efficiency. User experience may be considered subjective in nature to the degree that it is about individual perception and thought with respect to the system.
WFS(T)	The Open Geospatial Consortium Web Feature Service Interface Standard (WFS) provides an interface allowing requests for geographical features across the web using platform-independent calls. The basic Web Feature Service allows querying and retrieval of features. A transactional Web Feature Service (WFS-T) allows creation, deletion, and updating of features.
WMS	A Web Map Service (WMS) is a standard protocol for serving georeferenced map images over the Internet that are generated by a map server using data from a GIS database.[2] The specification was developed and first published by the Open Geospatial Consortium in 1999.
WMTS	A Web Map Tile Service (WMTS) is a standard protocol for serving pre-rendered georeferenced map tiles over the Internet. The specification was developed and first published by the Open Geospatial Consortium in 2010.

Term	Description
WPS	The OGC Web Processing Service (WPS) Interface Standard provides rules for standardizing how inputs and outputs (requests and responses) for invoking geospatial processing services, such as polygon overlay, as a Web service. The WPS standard defines how a client can request the execution of a process, and how the output from the process is handled. It defines an interface that facilitates the publishing of geospatial processes and clients' discovery of and binding to those processes.
XHTML	Extensible Hypertext Markup Language (XHTML) is a family of XML markup languages that mirror or extend versions of the widely used Hypertext Markup Language (HTML), the language in which Web pages are formulated.
XML	Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. It is defined by the W3C's XML 1.0 Specification and by several other related specifications, all of which are free open standards.