MODELLING CHOICES
AND ONTOLOGY EXPLANATION
FOR GO!

DOCUMENT OF TECHNICAL SPECIFICATIONS AND MODELLING CHOICES
FOR THE MODULES:

GO-TOP ~ GO-PHY ~ GO-HUM ~ GO-FAR

REVISION LIST

Version	Date	Revision description
1.0	25/09/2015	First draft
1.1.	24/01/2018	update to the IRIs of GO! modules

SUMMARY

1. Introduction to the project GeoLat and to the GO! ontology

GO! (Geographical Ontology) ontology has been developed by Claudia Corcione, Paola De Caro and Silvia naro, with the collaboration of Diego Magro, Timothy Tambassi and Maurizio Lana for the research project “Geolat – geography for Latin literature”. It describes the geographical locations, with a particular attention to the description of the Ancient World, especially to give the opportunity of having a link between the places mentioned in the texts, especially ancient, and their identification and correspondence with contemporary ones. For classical scholars this correspondence of ancient / contemporary modelling is of undisputed interest, both for the study of the habits of the most ancient peoples, and for the most various themes of literary interest. Through ontologies you can build maps of the ancient world and compare them to contemporary ones, annotate historical, geographical, cultural details connected to the place, indicate in which ancient text the place is mentioned and as which author discloses the details. These are just some ideas for research that can be developed, but the scenario that opens through these connections will be much larger.

The GO! modules contain numerous classes and relations and differ in the specific entities defined in them, and are connected by a Top Level ontology - GO TOP:

• an ontology that describes the physical and natural places, GO-PHY ;

• an anthropic ontology, where are specified all the administrative bodies and artifacts created by human activity, GO-HUM

• an ontology for the ancient world, which describes its specific aspects, GO-FAR where FAR means For Ancient Resources.

The GO! ontology serves as an information base for the platform of the project GeoLat.

The GO! modelling choices took into account the needs which the ontology must meet, allowing to add a range of additional information about the geographical place, through the inclusion of ad hoc relationships, in particular it is possible to express:

• the correspondence with the places listed by Pleiades (historical online gazetteer http://pleiades.stoa.org/)

• the physical and cultural characteristics shown in the Barrington Atlas

• the source where the ancient place is mentioned (with philological reference)

• the geographical coordinates of corresponding contemporary sites

• a description of historical events (wars, defeats ...)

• the changes of the place (e.g. a village which becomes a city)

• the hypotesized location of imaginary places (such as Hades)

• the physical and geopolitical description of the place

These are only some of the potentialities of the GO! ontology, which incorporates some standard ontologies (for example GeoSPARQL), so as to be more easily shared and reused, because the quality of the ontology and the project in its entirety resides in its widespread use, in order to become a benchmark for the projects that link to the geographical description.

2. The prerequisites of the GO! ontology

Here below are the main requirements that have been identified for the construction of ontologies. The choice of these requirements dictated the modelling choices of the GO! ontology:

1. the ontology is in English, with the exception of the Latin terms. Labels and descriptions, in RDF Schema are both in English and Italian.

2. all the classes related to the real world are inserted in the ontology, however there is also a class for imaginary places;
3. the geographical entities which have no identification in contemporary geography (eg imaginary places) will not have the properties latitude and longitude;
4. the relationship that connects the place in modern geography and in the antiquity. It will be instantiated (using the same reference system, since there are currently no GIS reference systems for ancient geography), but it will be possible however to specify coordinates also in not classic reference systems.
5. 3 more specific ontologies (GO-PHY, GO-HUM and GO-FAR) connected to a top level ontology (GO-TOP) have been defined. The tree structure was chosen, compared to the forest, because it is a structure that keeps everything under a common root, which are the GO-TOP ontology classes. The ontology modules were assigned the following IRIs:
    - for the top level ontology https://w3id.org/geolit/ontologies/GO-TOP
    - for physical and natural ontology https://w3id.org/geolit/ontologies/GO-PHY
    - for the ontology of human activity and products https://w3id.org/geolit/ontologies/GO-HUM
    - for the ontology for ancient resources https://w3id.org/geolit/ontologies/GO-FAR
6. the standard GeoSparql ontology has been imported into the Top Level ontology module;
7. to connect to the other standard ontologies an external mapping file was employed;
8. all ontologies are expressed in OWL2.

3. Modelling choices

When you start to think about how to represent a certain reality or domain knowledge, you must, in addition to studying the existing representations of the same domain and read up about what to represent, understand which is the purpose of modelling. There are in fact many possible representations of the same reality, which depend on the point of view of the objective and of those who represent it, ie, in this case, the group of ontologists. For this reason it is important, before you start thinking, to identify the purpose and the possible applications of the representation you're starting to create. The purposes and uses are the compass for the modelling choices, the ontological commitments, ie assumptions that ontologists must take to reality to be represented before approaching their work.

The ultimate objectives of the four ontological modules of the project GeoLat are mainly two:

- providing an inventory of classes and relationships in order to semantically annotate Latin texts;
- describing, using the ontological representation, the geographic knowledge in the ancient and contemporary world.

To fulfill these tasks and to meet other requirements expressed by the buyer, the representation is divided into four ontological modules containing respectively:

- GO-TOP: the set of classes and relationships shared by all modules, that is the general elements used to describe the geographical entities and situations in which they are involved;
- GO-PHY: the set of the classes organized into a taxonomy which constitutes an inventory of geographic entities that are found in nature;

- GO-HUM: the set of the classes are organized into a taxonomy, which constitute an inventory of geographical entities created by human activity (both in material sense, such as artifacts, and in intangible sense, such as the places that have a institutional value) ;
- GO-FAR: the set of classes, organized into a taxonomy, which make up an inventory of geographical features created by humans in ancient times, with particular reference to ancient Rome;

Despite the presence of elements imported from other ontologies (first of all the standard ontology GeoSPARQL entirely imported), the modelling choices are peculiar to the GeoLat project and do not reflect models and structures of the imported ontologies.

In the following sections we describe in detail the modelling choices adopted in each module.

4. The four ontological modules of GO!

4.1. Top Level module GO-TOP

This module contains all the more general elements describing the geographical entities and the situations in which they are involved. In particular, all classes of general and high-level belong the Top Level module, and all the relationships (both object property and date property), used by the other three modules.

4.1.1. Classes modelling

We describe below the high level classes, to explain the modelling from the point of view of the entities involved:

- **EntityName:** collects the names of entities. This class was created to describe the place names on the base of the name of other places, people, or persons related to it. For example, the Etruscan people gave its name to the region of Etruria.

- **God** is the class that contains the elements deities.

- **Event:** is the class that shapes the types of events. It has two subclasses: HistoricalEvent: events that depend on human action; and NaturalEvent: natural events, which do not depend on human action.

- **HistoricalEntity:** is the class that contains the different types of actors. It has two subclasses: HistoricalCharacter, which contains the different types of historical personalities (such as console and emperor) and HistoricalGroup, which contains various types of groups of people (such as armies and populations) and is divided again into two subclasses: CivilPeople (groups of civil people) and MilitaryGroup (military groups).

- **Path:** is a bridge class that serves to associate a route to an event, which will then have a length, measured with a measurement unit.

- **Length:** is a bridge class to bind a path to a numerical value that indicates the length of a path, and to units of measure.

- **UnitOfMeasure:** is the class that contains all the types of unit of measure by which one can measure a length (dimension used mostly in this ontology to connote a path).

- **MithologicalEntity:** collects the entities that have mythological value in certain texts.

- SpatialObject >> Feature >> **GeographicEntity:** the Spatial Object class and its subclass Feature are two classes of the standard ontology GeoSPARQL. Our class GeographicEntity, was defined as a subclass of Feature, namely the class of all those physical elements to which a geometry can be associated. So, to all the geographical entities present in the ontology it can be associated geometry, using the object property (of GeoSPARQL) hasGeometry.

- **AnthropicEntity:** This class is a subclass of class GeographicEntity and contains all features built by man from ancient times to today (such as bridge, road building).

- **PhysicalEntity:** This class is a subclass of class GeographicEntity and contains all features found in nature (such as mountains, hills, river).

- **SpatialObject** >> **Geometry**: Geometry class belongs to ontology GeoSPARQL and contains all kinds of geometries, this class is linked to Feature and therefore also to all GeographicEntity.

- **SpatialObject** >> **Location:** it is a class bridge to connect a geographical entity at a value of latitude and longitude, without passing through the concept of geometry.It was introduced to facilitate the operation of the annotators, which may not be experts in spatial modelling in the technical sense.

- **SpatialReferenceSystemIdentifier:** is the class that contains the various possible reference systems, that is the unique values used to identify unambiguously the spatial coordinates.

- **Temporal Entity:** This class contains the time periods that are not expressed in the form of precise dates (it's a possibility allowed by the presence of specific data properties as inDate, beginningPeriod etc ..), but rather as time spans, sometimes identified by influential people (for example the Augustan period).

- **UnrealEntity:** contains all entities that are not real, or fantasy beings or objects, regardless of whether they have or not a mythological value.

4.1.2. Object Properties Modelling

The object properties are relationships between elements of classes. These properties have been created to model the main relationships between the classes described in the previous paragraph. The modelling choice operated on the object properties has been to maintain the highest degree of generality, in order to reuse the same relationship to describe more facts or events. Each property is defined by a domain and codomain, respectively the set of entities from which the relationship begins and the set of entities in which it ends. When the properties have a highly general character, the domain and codomain was not specified, just to facilitate this general character and to not impose restrictions in advance to users / annotators. For simplicity sake we will arrange these properties on the basis of the size and will explain the ess intuitive relationships l.

- **Spatial dimension:** above, below, borderToTheEast, borderToTheNorth, borderToTheSouth, borderToTheWest, hasLocation, hasRealPlace, hasSRID, identify, inPlace, inSRID, isUnder, leftOf, nearby, RightOf, visibleFrom (it refers to a feature visible from another geographical entity, such as: the constellation X + visible from the summit of the mountain Y), beginningPlace, endingPlace (these two properties indicate either the place of beginning and end of an event such as a battle or a pilgrimage), partof.

- **Time dimension:** after, before, occurIn.

- **Nominal size:** deriveFrom (this property is used to describe the dependence of the meaning of a name by another name or an entity), hasName, nameOf.

- **Size of the actors:** Becomes, wins, composedBy (this property is used to describe by which factions / peoples consists an array / alliance), owns, foughtBetween (this property links a battle or a war to the involved parties), hasPath, hasStopOver (indicates the possibility that a path be split into stages), Involves, isStopOverOf (represents a stage of a path), objectOf, subjectOf (these two properties will be better illustrated in the section dedicated to events), to, by, passesThrough, controls, belongsTo.

- **Size measures:** hasLength; measuredBy (it binds a length to a unit of measurement).

4.1.3. Data Properties modelling

The date properties are relationships among elements of the classes and types of data, such as numeric or date values.

As for the object properties, the modelling choice resulted in the search of the maximum generalization and, for this reason, often the domain and the type of data that constitute the codomain of the date properties has not been defined. Below there is a list of the date properties (which in the case of counterintuitive relations are explained), divided by type of size:

- **Spatial dimensions:** latitude, longitude.

- **Time dimension:** beginningPeriod, temporalPeriod, endingPeriod, inDate, validSince, validUntil (the latter two properties are used to bind a man-made entity, often with institutional and conventional character, to a date or a time when that institution / convention is valid).

- **Area of nouns**: etymology (it serves to bind to the entity name the respective etymological definition), name (it is a very generic property, for which you neither domain or codomain is specified, just because it can be linked to any class and contain different values, numerical or not).

- **Area of measures:** hasValue (including an individual class length and a numeric value).

4.1.4. Events modelling

It was considered crucial to shape, in the TOP module also the possible events that involve the geographical entities found in the Latin texts. The approach adopted for this modelling was that of aiming at a maximum of generalization in the structure of the events, generalization necessary given the variety of the situations and the complexity of representation of the events.

To structure this representation in the simplest possible way there are a superclass **Event** which contains different types of events (e.g. foundation, destruction, birth, step, fight etc. …); **a class HistoricalEntity** to have an inventory of the possible players involved (part of this class are both historical personalities, and groups of people, such as armies or populations); **a class GeographicEntity** containing all possible geographical entities found in the Latin texts and useful to characterize an event from the point of view of the place. As for the time-related information describing an event, you can specify it in two ways: by using individuals of the **TemporalEntity class,** or by binding the instance of an event to a specific date or period of time through a given temporal property (beginningPeriod , temporalPeriod, endingPeriod, inDate).

Each event can have a **subject** (who/what does the action described in the event) and an **object** (someone/something who bears the action described in the event) according to the type of event that is being described. For example:

The birth of Caesar: `<GO: Caesar> <GO: subjectOf> <GO: birth_1>`

The construction of the Imperial Palace: `<GO: August> <GO: subjectOf> <GO: constructing_1>`

```
<GO: palazzoImperiale_1> <GO: objectOf> <GO: constructing_1>
```

Each event can be associated with a **time period** (beginningPeriod, endingPeriod, inDate) and with one or more **locations** (inPlace). For example date and site could be associated to the events previously mentioned:

```
<GO: birth_1> <GO: inDate> "101 BC" <GO: birth_1> <GO: inPlace> <GO: rome>

<GO: constructing_1> <GO: beginningPeriod> "10 BC"

<GO: constructing_1> <GO: endingPeriod> "8 BC"
```

It is also possible that other actors are involved, in addition to the subject and the object of an event, and for that the object properties **"to"** and **"by"** (inverse to each other) were created. This is for example the case of events as Conquest, in which intervene a subject who wins, a conquered object and the previous owner of the object, to which is conquered. For example:

```
<GO: Roman> <GO: subjectOf> <GO: conquest_1>

<GO: conquest_1> <GO: objectOf> <GO: Carthage>

<GO: conquest_1> <GO: to> <GO: Phoenician>
```

Through these properties and generic classes you can model almost all kinds of events. A special case occurs in the event of war or battle. Here there may be multiple subjects involved and connected to each other, such as more allied peoples in a conflict, with no object. This is the reason why the concept/class Alliance was created (as a subclass of MilitaryGroup) and the object properties **foughtBetween** (including a battle/war and alliances) and composedBy (between alliance and historical entities that compose it). For example:

```
<GO: alleanza_1> <GO: composedBy> <GO: popolo_1>

<GO: alleanza_1> <GO: composedBy> <GO: popolo_2>

<GO: alleanza_2> <GO: composedBy> <GO: popolo_3>

<GO: alleanza_2> <GO: composedBy> <GO: popolo_4>

<GO: guerra_1> <GO: foughtBetween> <GO: alleanza_1>

<GO: guerra_1> <GO: foughtBetween> <GO: alleanza_2>
```

This is an effective way to overcome the expressive limit of binary relations (typical of ontological formalisms) and to be able to claim that two or more people fight together against another alliance of peoples within a single event.

Finally, the event of **"becoming",** that is of change/evolution of an entity into another one (for example, a temple which becomes a warehouse) has not been modeled with the general structure of the events, but with the object property **Becomes** which connects two instances of the same class or of two different classes, one that evolves in the other. The evolution is understood both from the ontological point of view (if an instance changes over time its ontological class that it initially is an instance of a given class and later it is an instance of another class) and from the point of view of use (the two instances will always belong to the same ontological category, while assuming different roles).

```
<GO: tempio_1> <rdf: type> <geo Temple>

<GO: tempio_1> <GO: inDate> <1860>

<GO: magazzino_1> <rdf: type> <GO: Warehouse>

<GO: magazzino_1> <GO: inDate> <1930>

<GO: tempio_1> <GO: Becomes> <GO: magazzino_1>
```

4.2. Physical geography module GO-PHY

The PHY module contains the classes that represent geographical entities with physical-natural aspects. Within this module the GO-TOP module was imported, to allow the description of the geographical phenomena. The upper classes too belong to the GO-TOP module (GeographicEntity and PhysicalEntity), in order to make the modules connected to each other and at the same time independent should they had to be individually reused, even for different purposes.

4.2.1. Classes modelling

All the classes of GO-PHY belong to the PhysicalEntity over-class of GO-TOP**;** below there is the description of the classes and the explanation of the choices of entities modelling:

AstronomicalEntity: it contains all the entities of astronomical type, affecting the astral bodies and placed into extraterrestrial space.They are necessary because very often the astral bodies serve as reference points for the routes and for guidance.

TerrestrialPhysicalEntity: it contains all the physical entities referring to the land. Within this class they contain all entities that physically describe the geographic locations on Earth. Within this class some subclasses were created to improve the organization and the research, in particular:

Desert: it has subclasses that describe the type of desert

HydrographicEntity: it collects all subclasses relating to hydrological entity. Some hydrological classes have in turn subclasses, because they specify the type of Lake **(Lake),** mouth **(Mouth)** and inlet **(Inlet).**

Pass: it specifies the type of pass

Relief: it specifies the type of relief

Valley: it has a subclass to specify a specific type, the canyon.

4.3. Anthropic module GO-HUM

The module GO-HUM contains all entities produced by anthropic activity, including artifacts. In this module the GO-TOP module is imported, to allow the modelling of the events and to be in connection with other GO ontology modules; it was also imported the standard ontology GeoSPARQL.

4.3.1. Classes modelling

The classes of GO-HUM are born as subclasses of the GO-TOP **AnthropicEntity** class**.**

Here are the top-level classes and the modelling choices adopted:

AncientEntity: in this class all the entities belonging to the ancient world are represented, in particular it refers to period of the Latin and Ancient Greek texts. This class is disjointed from the ContemporaryEntity, MedievalEntity and **PrehistoricEntity.**

ContemporaryEntity: in this class all the entities belonging to the contemporary world are represented. This class is disjointed from the classes **AncientEntity, MedievalEntity** and **PrehistoricEntity.**

MedievalEntity: this class represents all the entities belonging to the medieval world. This class is disjointed from the classes **ContemporaryEntity, AncientEntity** and **PrehistoricEntity.**

PrehistoricEntity: this class represents all the entities belonging to the prehistoric period, and it is disjointed from classes **ContemporaryEntity, MedievalEntity** and **AncientEntity.**

Artifact: this class includes all the artificial, not natural, entities that is all the entities made by man. Inside this class, with an IRI on the GO-HUM all the artifacts produced by man are inserted. It contains two important subclasses:

> **ArchitecturalElement:** this class brings together all the entities related to architectural elements.
>
> **Building:** this class brings together all entities related to the various types of buildings.
>
> **InstitutionalEntity:** in this class all the man-made institutional entities are entered such as forms of government (eg. monarchy and republic), boundaries, types of human settlement etc.
> It has two subclasses:
>
> **Boundary:** the class that represents the boundaries and their different types (administrative boundaries, maritime, political, forestry).
>
> **MilitaryEntity:** the class that represents the various kinds of military entity (eg. the command, the trench, the barracks etc.)

4.4. Ancient module (GO-FAR - For Ancient Resources)

The module GO-FAR was created to describe the resources related to the ancient world, of all the forms is the one with the most specific temporal connotation, as it aims to be able to be used to record and describe places, people and events of ancient Rome. For this reason it underwent modelling choices different from those of the previously described modules, also many class names have been labeled with their Latin name. In this case too the GO-TOP module has been imported, as well as the standard ontology GeoSPARQL.

Also some classes of GO-HUM (AncientEntity; Artifact and InstitutionalEntity) were included as they refer to the human and socio-political sphere, but within them there are subclasses that are specific of the ancient world. In this case it was not done a comprehensive import of the GO-HUM module but specific classes were inserted through their GO-HUM IRI. This is because if the GO-HUM module had been integrally imported into GO-FAR also all those classes which are anachronistic for the Ancient world would have been imported into GO-FAR(e.g. Airport, , Cable car, Highway, etc.); therefore the GO-HUM elements also present in the Ancient World have been identified and only those have been brought into GO-FAR simply inserting them with their IRI (and including them appropriately in the taxonomy).

This modelling choice allowed to maintain the GO-FAR IRI for all those entities typical of the Roman time (e.g. Mausoleum, Obelisk, etc.) and to specify the adjective "Roman" for all those entities with homonyms, such as Villa (the Villa of today is not the same thing as a RomanVilla, even if they share the same name).In this case, the conceptual division was a must and was made in the ontology. The distinction between a contemporary instance of a bridge and an ancient one will be done on the level of individuals, that is the bridge X, found in the Latin text of the author Y, will be both an ancient artifact and an ancient entity and this can be expressed by defining the bridge X as an individual both of the class "Bridge" and of the class "AncientEntity".

4.4.1. Classes modelling

As it was already explained, in the GO-FAR module only the specific classes for the Ancient World were included, that is those contained in the **GO-HUM AncientEntity** class. In particular we point out:

RomanEntity: combines all the classes related to specific entities of Roman type (eg.**Domus, RomanBasilica, RomanStreet** etc.).

RomanVilla: this class contains entities related to the Roman villa, which as it was already mentioned has peculiar characteristics.

GreekEntity: this class collects all the entities related to the ancient Greek civilization.

In the **GO-FAR** module, from the GO-HUM Artifact class only the classes of entities that also belong to the Ancient World have been inserted, but by modelling choice they maintained the original GO-HUM IRI. The same method of modelling (keeping the GO-HUM IRI) was adopted for the classes **ArchitecturalElement, Building** and **InstitutionalEntity,** which contain entities that can either be used for the description of the Ancient World, and for the description of any other entity not connected to a particular historical period, because they refer to a wider anthropic context.

5. Mapping with external ontologies

The other standard ontologies were mapped externally and are contained in a separate file.

In particular the ontologies that have been included in the mapping file are:

xmlns: schema = http://schema.org/

xmlns: current = http://erlangen-crm.org/current/

xmlns: geo = http://rdf.insee.fr/def/geo#

xmlns: resource = http://wikidata.dbpedia.org/resource/

xmlns: ontology3 = http://www.geonames.org/ontology#

xmlns: places = http://purl.org/ontology/places#

xmlns: ontology = http://dbpedia.org/ontology/

xmlns: DUL = http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#

xmlns: ontology2 = http://linkedgeodata.org/ontology/

xmlns: place-types = http://pleiades.stoa.org/vocabularies/place-types/

To allow the mapping with PLEIADES (http://pleiades.stoa.org/vocabularies/place-types/) the classes were reified the relationship OWL:sameAs with its individual belonging to the Pleiades class was instantiated.

Doc: Document Geolat 1.0.docx