Design of an ontology for humin substances

Article · December 2012

DOI: 10.1145/2428736.2428806

4 authors, including:


Martin Rimnac
The Czech Academy of Sciences


Roman Špánek
Technical University of Liberec


Július Štuller
The Czech Academy of Sciences

Design of an Ontology for Humin Substances

Pavel Tyl
Technical University of Liberec
Liberec, Czech Republic
pavel.tyl@tul.cz

Roman Špánek
Technical University of Liberec
Liberec, Czech Republic
roman.spanek@tul.cz

Martin Řimnáč
Institute of Computer Science
Czech Academy of Sciences
Prague, Czech Republic
rimnacm@cs.cas.cz

Július Štuller
Institute of Computer Science
Czech Academy of Sciences
Prague, Czech Republic
stuller@cs.cas.cz

ABSTRACT

Knowledge can be used for description, sorting and searching of useful data. Knowledge, on the other hand, may be very narrowly connected with a person, who may leave a company. Such situation can be avoided by using of ontology as a tool of knowledge engineering or in general as a tool for knowledge representation.

The paper presents on ontology design to peculiarities of humin substances based on analysis of existing similar type ontologies. It enables to preserve, make accessible and reuse gained experiences, for instance for design of functionally rich, flexible and easy integrable systems and applications.

In introduction this work presents an overview of selected formalisms for interception of data structure and processes in humin substances area with respect to using ontologies and their possibilities. The paper is especially a case study of ontology application for evidence and data organization of humin substances, their measurement and experimentations, where use of ontologies is not yet common. It shows a potential of ontologies and illustrates parameters of generally applicable model for such area.

Keywords

ontology, humin substances

1. MOTIVATION

Humin acids, in general humin substances, can higly influence a growth of various crop-plants. Humin substances (HS) could be used in the area of fytoremediation of contaminated soil. Influence of using HS for soil fytoremediation in dependance on used crop-plant was investigated within the scope of project Advanced Remedial Technologies (ARTEC). Influence of interaction between HS and plant substrates on growth of selected crop-plants and remediation dynamics (intensity of contaminants collecting from soil by crop-plant) is observed. There is a real assumption that HS (thank to their stimmulative effects on growth of plants) could dramatically influence the growth of their root system and also contaminants transformation intensity to the plant that can be used as biofuel after the harvest.

Previously mentioned activities are only a part of goals of ARTEC Research Center that deals with remediation technologies. Recently it happens to apply obtained results about particular technologies and their testing on real tasks.

The paper describes pilot project, whose goal was to design and to construct an ontology for humin substances production and for description of experimentations with humin substances include their results.

iiWAS2012, 3-5 December, 2012, Bali, Indonesia.

1.1 Why to Use Ontology?

In recent years the development of ontologies – explicit formal specifications of the terms in the domain and relations among them [3] – has been moving from the realm of artificial intelligence area to the area of interest of domain experts. Ontologies have become common on the Internet. The ontologies on the Web range from large taxonomies to categorizations of products with their properties. Many disciplines now develop standardized ontologies that domain experts can use to share and annotate information in their fields. For example in the biochemistry topic was developed big standardized structured vocabulary and onotlogy of chemical entities related to biology – ChEBI [1]. The WWW Consortium [14] is developing the Resource Description Framework and later OWL and OWL 2.0 languages [8], languages for encoding knowledge of objects (entities) on Web pages to make it understandable to electronic agents searching for information.

An ontology defines a common vocabulary for groups of users (people or software agents) who need to share data, information and knowledge in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them.

Why would someone want to develop an ontology? Some of the reasons are:

- *To share common understanding of the structure of information among people or software agents* – one of the most often reasons for ontology development allows software agents to summarize extracted information from more overlapping sources and answer user's queries.

- *To enable reuse of domain knowledge* – if we want to create new ontology, we can use the existing ones and integrate them, decrease or increase according to our needs.

- *To make recent general view of domain knowledge* – unambiguous specification of domain knowledge is important for new users who need to know the meaning of ontology terms. If pieces of knowledge are described in a ontologz, we can easilt find them or change. Changing and looking for all software agents with fixed knowledge (inner logic of every agent) would be much more difficult.

- *To analyze domain knowledge* – this can be done, if we have available exact meaning of particular ontologz terms. It can be used to the following increasing and reuse existing ontologies.

- *To separate domain knowledge from the operational knowledge* – other commmon use of ontologies. We can for example describe a configuration of some product from its components according to required specification and create a program that makes such configuration independent on concrete product and components themselves.

Often an ontology of the domain is not a goal in itself. Developing an ontology is akin to defining a set of data and their structure for other programs to use. Problem-solving methods, domain-independent applications, and software agents use ontologies and knowledge bases built from ontologies as data inputs. In this context, an ontology of humin substances serves for both internal cooperation in research team and for presentation of knowledge in the area of humin substances to other developers [6].

1.2 What is in an ontology?

Literature knows more definitions of an ontology. For the purposes of this paper an ontology is a formal explicit description of notions in a domain of discourse, in our case the domain of humin substances. Ontology will contain *objects* (called *concepts*, sometimes called *classes*), their *properties* (called *roles*, sometimes called *slots*) and given *restrictions* of *objects* and *properties* (*roles restrictions*, sometimes called *aspects*). An ontology together with a set of *concept instances* and *roles* constitutes a *knowledge base.* A fine line where the ontology ends and the knowledge base begins is often very vague.

Concepts are the focus of most ontologies. Concepts describe basic notions in the domain. For example, a concept *Material* represents general material. Specific existing materials are ***instances*** of this concept. A concept can have *subconcepts* that represent notions that are more specific than the superconcept alone. For example the concept *Experiment* has subconcepts *Parcel_experiment* and *Container_experiment*. can be divided the class of all wines into red, white, and rosÃl' wines. ***Hierarchy*** of concepts (relationship subconcept–superconcept) is transitive.

Roles describe properties of concepts and instances. Every *Material* has for instance particular *majority_element* and *observed_element.* Property *majority_element* can have several different values (*Al*, *Ca*, *Cl*, *Fe*, . . .). Property *observed_element* can have other possible values (*As*, *Cd*, *Co*, *Cr*, . . .). At the concept level, we can say that instances of the concept *Material* will have role *majority_element* and *observed_element* and so on.

All **instances** of the concept *Material* have role *majority_element* giving a value of concentration of particular elements (instances of the concept *Element*) in *Material* (see Fig. 1).

In practical terms, developing an ontology includes:

- *defining concepts* in the ontology,

- *arranging concepts* in a taxonomic (subconcept–superconcept) hierarchy,

- *defining roles* and *describing allowed values and restrictions* for these *roles* and

- sometimes *topping up instances* and *values for roles* (see Fig. 2).

2. ONTOLOGY DESIGN METHODOLOGY

Some fundamental rules in ontology design to follow and to keep in mind are these:

1. There is no "one correct" way or methodology for developing ontology – there are always *viable alternatives.* "The best" solution almost always depends on the application that you have in mind and the extensions that you anticipate.

2. Ontology development is necessarily an *iterative process* – we start with a rough first pass at the ontology, we then revise and refine the evolving ontology and fill in the details (along the way, we discuss the modeling decisions that a designer needs to make, as well as the pros, cons, and implications of different solutions).

3. Notions of the ontology should be close to real objects in described domain. (*Concepts* will be probably represented by *nouns*, *roles* among them by *verbs* and all together will create simple sentences describing our domain of interest.)

Deciding what we are going to use the ontology for, and how detailed or general the ontology is going to be will influence its modeling. Among several viable alternatives, we will need to determine:

- which one would work better for the projected task,

- which one would be more intuitive,

- which one would be more extensible and

- which one would be more maintainable.

After we define an initial version of the ontology, we can evaluate and debug it by using it in applications or problem-solving methods or by discussing it with experts in the field. As a result, we will almost certainly need to revise the initial ontology. This process of iterative design will likely continue through the entire lifecycle of the ontology.

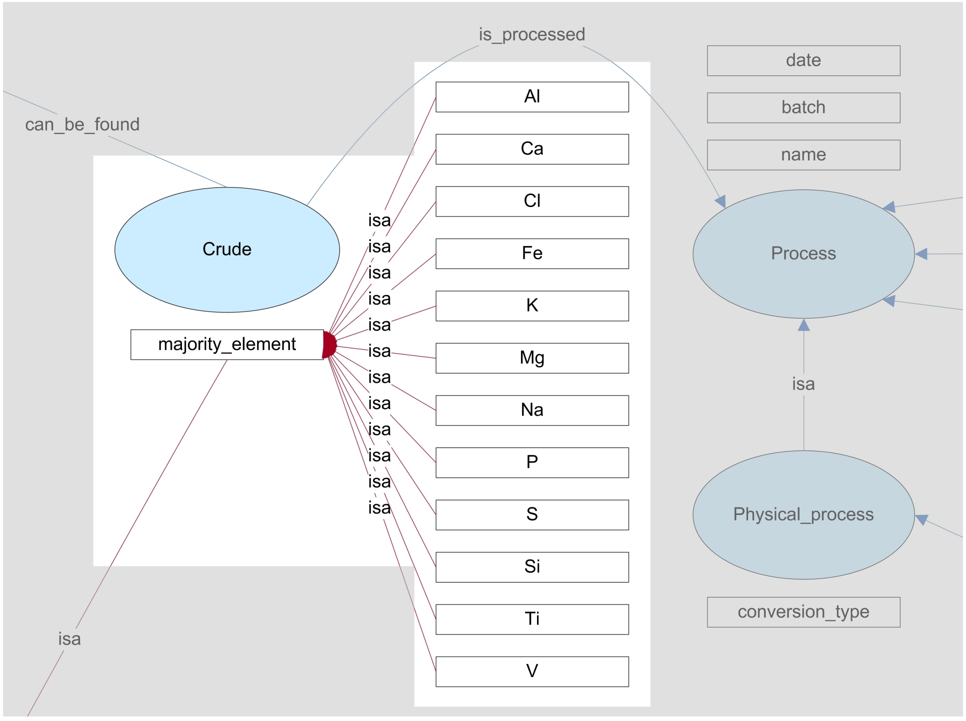

Figure 1: Concept *Material* and role *majority_element*.
Concepts are depicted in elliptic form, datatype roles and instances as rectangles. Arrows symbolize object roles, arrows with word *is-a* represent role subconcept–superconcept (subproperty–superproperty respectively).

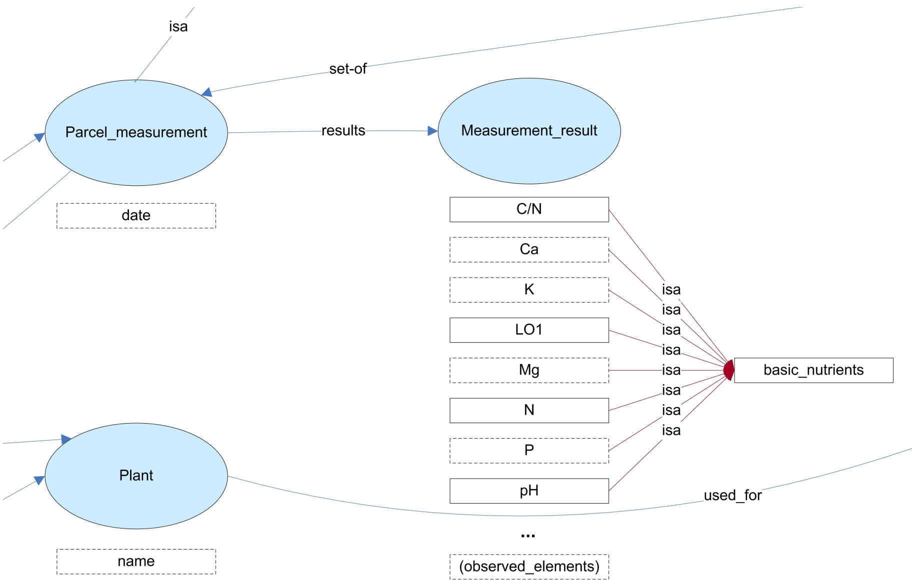

Figure 2: Some concepts, their roles and relationships among them in the humin substances domain.

2.1 Determine the domain and scope of the ontology

For determining the domain and scope of the ontology we have to answer several basic questions:

- What is the domain that the ontology will cover?
- For what we are going to use the ontology?
- For what types of questions the information in the ontology should provide answers?
- Who will use and maintain the ontology?

The answers to these questions may change during the ontology design process, but at any given time they help limit the scope of the model.

Domain of our ontology are humin substances, which are used (in the scope of this paper) as an additive to plant substrates, but in general they can be used for specific applications in given chemical or biochemical processes. Ontology could be used for:

1. qualitative assessing and selection of deposits of primary carbonaceous materials in humin acids, or in general in humin substances (oxyhumolits, lignit, peat),

2. processing procedure of primary carbonaceous materil to products (mechanicaly modified oxyhumolit, alkaline humat, pure humin acids) with required qualitative parameters of humin acids contained (HA content, relative molecule materialities spectra, biological activity level, functional groups content with basic effect to their chemical and bilogical activity [–COOH a fenolic –OH groups], ashes content, etc.),

3. assessing of propriety for precesses application of humin substance with given quality parameters, with defined conditions of their ordering and vice versa, what qualitative parameters humin substance has to have to achieve requirements of our defined process.

If the ontology we are designing will be used to assist in natural-language processing of articles about chemical substances, it may be important to include synonyms and part-of-speech information for concepts in the ontology. If the ontology will be used to help to infer economical costingness, we need to include costs of particular experiments and measurements, or availability of particular materials and chemicals. If the people who will maintain the ontology describe the domain in a language that is different from the language of the ontology users, we may need to provide the mapping between the languages.

2.1.1 Competency questions

One of the ways to determine the scope of the ontology is to sketch a list of questions that a knowledge base based on the ontology should be able to answer, *competency questions* [4]. These questions will serve as the test later:

- Does the ontology contain enough information to answer these types of questions?

- Do the answers require a particular level of detail or representation of a particular area?

These competency questions are just a sketch and do not need to be exhaustive. In the humin substances ontology are the competency questions for example following:

- What energetic plant should we plant, if some area is contaminated by "any of observed elements"?
  Selection fo suitable plants depends on concrete case and conditions (type and sort of contaminant, clima, soil, humidity, used technological system, i. e., phytoremediation as a part of other remediation applications, technology of particular plants cultivation, possibility of further using of cropped plants, etc.).

- What financial costs will be needed for decontamination of an area in five years horizont?
  It depends on several factors – chosen plant and its planting technology severity or costs of using various mechanization means, type and way of application of humin substances, further using of grown phytomass, climatic factors, etc. From this point of view multiannual plants are prefered. On the contrary possible profit from grown phytomass is lower.

- What majority elements are contained in the material which is produced by some concrete chemical process?

- What observed elements are figuring in parcel measurement results, if any?

Judging from this list of questions we can get an overview, what possible answers (in the form of appropriate data) we can obtain from database, information system or knowledge base which are build up the developed ontology – for example the selection of the the most suitable energetic plant while risk of contamination, financial and time costs for decontamination etc.

2.1.2 Consider reusing existing ontologies

It is almost always worth considering what someone else has done and checking if we can refine and extend existing sources for our particular domain and task. Reusing existing ontologies may be a requirement if our system needs to interact with other applications that have already committed to particular ontologies or controlled vocabularies. Many ontologies are already available in electronic form and can be imported into an ontology-development environment that we are using. The formalism in which an ontology is expressed often does not matter, since many knowledge-representation systems can import and export ontologies. Even if a knowledge-representation system cannot work directly with a particular formalism, the task of translating an ontology from one formalism to another is usually not a difficult one.

There are libraries of reusable ontologies on the Web and in the literature. For example, we can use the Ontolingua ontology library [7] or the DAML ontology library [2]. There are also a number of publicly available commercial ontologies, e. g., UNSPSC [12]. Already mentioned ontology library ChEBI [1] focuses on classification of chemical substances of biological importance. There also exists a taxonomy of chemical substances created by collaboration of HCLS (Semantic Web Health Care and Life Sciences Interest Group) and W3C [5].

By importing of these ontologies our ontology would be too diverted from experiments and measurements problems, economical monitoring and other aspects relating humin substances. Let us assume that no relevant ontologies already exist for our purpose.

2.2 Ontology basic concepts and terms

The following section gives a brief overview on concepts and terms that are important for designed ontology. Each term is followed by a summarization of its properties and a short description.

For example we will use the following terms:

- *Material*,

- *Locality*,

- *Biologic_activity*,

- *Plot*,
- *Plant*,
- *Substrate*,
- *Energetic_plant*,
- *Physical_process*,
- *is_measured*,
- *is_processed*,
- ...

The list ought to be complete, extensive; terms may overlap.

2.2.1 Concepts definition and concepts hierarchy

Concepts hierarchy can defined in three possible ways – *top-down*, *bottom-up* and *combination of both* [13]. Selecting the best method cannot be defined explicitly, but it has to follow ontology designer experience, topic peculiarities, etc. As the topic of Humic compounds is relatively complex, the combination of *top-down* (used for concretion of hierarchy) and *bottom-up* (used for wrapping of a concept by more universal one) was used.

The first step during an ontology design is a specification of terms – state-full terms are chosen from the set of important terms (see section 2.2) at first step, followed selection of terms describing the state-full ones more in details. The state-full terms are in an ontology described by concepts. Concepts hierarchy is then created from a set of concepts instances that are inevitably instances of another concept (e. g., included one). For example, *price* of any term defined in the ontology is inevitably *economical_parameter*.

2.2.2 Concept properties/roles definition

Concepts themselves cannot be used for answering competence questions (see section 2.1.1). It is necessary to describe also properties and relationships between defined concepts. Roles candidates have been already chosen from a set terms (e. g.,*is_measured*, *is_processed*, *can_be_found*, *belongs_to_locality*, ...

For each role in the set a concept has to be selected and assigned. The roles are sticked to concepts. For example, concept *Material* has roles *label*, *description*, *overall_acidity*, and other roles from *physically-chemic_parameters*.

Properties that can be roles in an ontology can be classified into the following:

- inner property – e. g., *overall_acidity Materials*,
- outer property – e. g., *label Localities*,
- parts – for structured (real or abstractly) objects, e. g., *relative_increment Biologic_activity*,
- relationships – on connection of particular instances (of given concepts), e. g., *Material is_processed Process*.

Thus, concept *Material* will have the following roles:*label*, *overall_acidity*, *carboxyd_group*, *dust*, *dryBasis*, *water*, etc. Every sub-concepts inherit roles from parent-concept. Each sub-concept of *Process* concept in our ontology *Physical_process* and *Chemical_process*) will have *date*, *label* a *rank*. Role by ought to be connected to the most general concept in the hierarchy.

2.2.3 Definition of roles constraints

Different types of constraints (e. g., allowed data types, values, cardinality, etc.) can be defined for roles. For example, value of *majority_element* role is a String (up to 3 letters). Role *can_be_found* can have multiple values from set of *place_of_sample* instances.

The most common constraints are listed bellow:
Role cardinality – defines a multiplicity of a role. Some systems can distinguish only between one and more cardinalities, some systems are also able to define maximum and minimum.
Role Data type – express data types possible for role values. Most common data types are String, number, truth value, enumeration and instance.
Role Domain – the most general concept describing roles. *Material* is a domain of *is_processed* role.
Role Range – the most general concept, which instances are valid values of the role. *Process* is range of *is_processed* role. If there are more concepts being a range of a role (*is_processed* role has both *Process* and *Chemical_process*) than the most general concept is used (*Process*).

2.3 Creation of instances of concepts

Instances of concepts are in some cases created as the last step of an ontology design. Instance definition requires the following:

1. concept selection,
2. creation of instance of the concept,
3. filling with values of roles.

For example, assume that an instance of *Container_experiment* (particular experiment) is to be created. At first the following set of values of roles has to be set (it is not a complete set of values but only an example set):

- *date*: 2010-04-05,
- *fertilization*: Before a plant is seeded, substrate which is going to be used for container experiments is analyzed and based on the analysis results fertilization (addition nutrients – biogenic elements) is set up. Nutrients are added with in watering,
- *used_soils (substrates)*:
  - Locality *Prague–Ruzyně* (label *R*),
  - Locality *Milevo* (label *M*),
- *container_label*:
  - *R* (variants 1–10),
  - *M* (variants 11–20),
- *number_of_repetition*: 2 (a, b),
- *experiment_schema*:

Id	Substrate	Container	Plant	Fertilizat.	Coal [g/cont.]
1	R	1R	–	–	–
2	R	2R	–	true	–
3	R	3R	–	true	60
4	R	4R	–	true	120
5	R	5R	–	true	240
6	R	6R	true	–	–
7	R	7R	true	true	–
8	R	8R	true	true	60
9	R	9R	true	true	120
10	R	10R	true	true	240
11	M	11M	–	–	–
12	M	12M	–	true	–
13	M	13M	–	true	60
14	M	14M	–	true	120
15	M	15M	–	true	240
16	M	16M	true	–	–
17	M	17M	true	true	–
18	M	18M	true	true	60
19	M	19M	true	true	120
20	M	20M	true	true	240

- *coal*: temporal label *JV21* (mine Družba, Sokolov),
- *watering*: up to 50 % of the maximal water capacity,
- *experiment_support*: Experiment location provides protection against lost of inflowed solution (due to high precipitation) from a bowls beneath containers,
- *used_plant*: each soil type is used in set of container with and without plant (instance of *Plant* concept)

	2009	2010	2011	2012
1. plant	–	Spring Wheat	Spring Wheat	Oat
next plant	Corn Seeded	Field Pie	Corn Seeded	–

- *used analytic methods*:
  - *Soil samples*
    pH/KCl
    Leach Mehlich II. – content of P, K, Ca, Mg [mg/kg]
    Content of humic [%]
    Overall organic mass in dry basis [%]
    $HS_A + HS_B$ in dry basis [%]
  - *Plant samples*
    Estimation of majority elements concentration (N, P, K, Ca, Mg) in dry basis [%],
- *has_result*: Res_ER_2010-04-05_001 (instance of *Result_of_experiment* concept).

2.3.1 Created instances notes

2010

Container experiments have show no change of volume of humic compounds as a result of coal transformation during the first year of the experiments. Results also showed (with respect to an acceptable error) that content of organic mass was increased in correlation to requirements. Results of analysis and weight of dry basis in harvested corn do not corresponds to incremented ratio of coal. Decrease of concentration of elements in containers with oxyhumolitic coal corresponds to dilution effect inducing by increase of dry basis.

2011

Harvest in 2011 showed that high concentration of coal decreases weight of harvested plants. Wheat analysis (wheat grew in both soil types) revealed the fact, that concentration of N, P, and Ca increased in all containers with coal with no correlation to coal amount. Field pie, on the the hand, increase of elements concentration was identified only in containers with substrate from Ruzyně. Substrate analysis done at the end of the containers experiment revealed that increased coal amount caused increase of content of humic and decrease of substrate pH.

2012

Results of dry basis of samples took from wheat in the first midyear of 2012 were not effect by coal in the containers. The experiments followed by planting corn into containers. Weight of dry basis was increased by running high amount of oxyhumolitic coal. Analysis of substrate did in Autumn 2011 showed correlation between contain of organic mass in dry basis and amount of oxyhumolitic coal.

Established container experiment with oxyhumolitic coal fulfills prerequisites for successful continuation; results are yet to be final and will be replenished.

2.4 Concepts definition, concepts hierarchy – common errors, recommendations

This section contains common errors that may occur during concept definition, definition of concept hierarchy. The section also contains recommendations helping proper ontology and ontology hierarchy definitions.

The designed ontology hierarchy may depend on possible application of the ontology, level of details necessary for application processing, but it may also depend on compatibility with the existing models. On the other hand, there exists a methodology, that should be followed and therefore we would like to revise the designed ontology and do some back-checking.

2.4.1 Is designed hierarchy correct?

IS-A relationships – Hierarchy of concepts represents an IS-A (KIND-OF) relationship: concept *Chemical_process* is a subconcept of *Process*, each concrete chemical process is also a process. Subconcept represents a term, which somehow represents parent concept. Designing the concept *Parcel_measurement* as a subconcept of *Parcel_experiment* would be an error. Despite the fact that parcel experiments are composed of s set of (SET-OF) parcel measurements, concrete *Parcel_measurement* cannot be considered as *Parcel_experiment* – it does not follow the reality.

Transitivity of hierarchy relationships – relationships of subconcept is transitive. Let *Cu* is an *Observed_element* and *Observed_elements* are *Quantitative_parameters*, then *Cu* is also a *Quantitative_parameter*. *Cu* is also direct subconcept of *Observed_element*.

Evolution of concepts hierarchy – maintain concepts hierarchy in consistent state may be a complicated if topic of interest is under continual evolution. Consistency is on the other hand crucial, if an ontology has to reflect chosen portion of the reality from the selected domain.

Avoidance of cycles in the hierarchy – A hierarchy contains a cycle if concept *B* is a subconcept of concept *A* and concept *A* is also a subconcept of concept *B*. It logically leads to equivalence of concepts *A* a *B*.

Analysis of siblings in concepts hierarchy – Siblings are direct subconcepts of a parent concept. Each of them should be designed with the same level of abstraction. For example, *Majority_elements* and *Hg* cannot be of the same level of abstraction. *Majority_elements* are more abstract compared to *Hg* term. Concept close to the root of the hierarchy represent basic terms and such concepts are usually direct subconcepts of a very abstract concept (e. g., *Thing*). Thus, such concepts are not required to be an the same level of abstraction.

Too few or too much siblings? – There is no rule how to set number of siblings. Even though, if there is only one subconcept. it usually means that ontology is not complete or here is an error in modeling. As an example let us mention a list – a list should not contain only one element. If there are about 12 subconcepts, it is recommended to add a interleaved category. List concepts of hierarchy are exceptions. which can be substituted by instances (in our case labels of elements – *As*, *Cd*, *Co*, *Cr*, . . .). If there are not interleaving categories, we should no create on, as the reality should be given preference

2.4.2 Multiple inheritance

Majority of systems for creation, maintenance and processing of ontologies (e. g., ontologic editors Protégé [9], SWOOP [11], . . .) enable multiple inheritance – one concept can a subconcept of several concepts. For example, set of *physically-chemical* processes which instances are elements from concept *Physical_process* and also *Chemical_process*. Such concept inherits from both parents.

2.5 Creating new concepts

One of important decision in the ontology modelling is a question if to create new concepts or to express semantic difference by different roles. It is a quite difficult to understand the ontology with many nested concepts or to understand the ontology with a small amount of concepts, but having much data expressed by the roles. A sound ontology design well arranges these two extremal approaches. Several rules of creating new concepts in a hierarchy exists. Commonly, subconcepts of any concept

- have more propeties, which the original concept do not have
- have constraints differing to supconcept
- assign another relationships than supconcept.

For example in our case, the concept *Chemical_process* has the role *reaction_duration*, which is not used for a description of the supconcept *Process*. New concept in the hierarchy will be defined in the case, if exists anything, which is valid for a concept, but not for a subconcept. Especially, each new concept should be extended by new roles or should have defined a new values of roles or should precise a inherited roles. On the other hand, subconcepts may not be created, when these subconcepts cannot be used for inference except the cases, when this new concept simplify a navigation in a taxonomies. A concept may be also created in situations, when a domain expert distinguish given terms and does not model the difference between the terms.

2.5.1 Creating new concepts or properties?

The specific differences can be modelled by roles or a set of concepts according to the ontology domain. For example in our ontology, the *Parcel_experiment* and *Container_experiment* can created as subconcepts of *Experiment*, or alternatively the concept *Experiment* can be extended by the role *Experiment_type* having specified domain of *Parcel* and *Container*. Since the results of *Parcel_experiment* and *Container_experiment* are totally different (the experiments are connected with *Parcel_experiment Result*, respectively *Container_experiment Result*, a new concept should be used (to avoid at least to connect *Parcel_experiment Result* with *Experiment* of *Experiment_type* to be set to *Parcel* . In other words, if concepts with different values of roles becomes to be limited for different roles, new subconcept is created; otherwise a type is distinguished by especial role, expressing a kind of the concept, to be created in the concept. In these cases, it is recommended to potentioaly create a corresponding instances.

2.5.2 Creating instances or concepts?

A decision to use a concept or instance depends on assumed application of the ontology. The difference between concepts and instances is the lowest level in a hierarchy of terms. The candidates to instances are the terms, which are the most specific from the domain of the interest (to be selected in searching competence as presented above in Section 2.1.1). The important difference between concepts and instances is, that instances are not ordered in hierarchy (instead of concepts), a subinstance is not commonly used in ontology modelling. If exists a hierarchy between the terms, the concepts should be used. For example, *Energetic_plant* grows on *Parcels*. These *Parcels* are distinguised by the place they are located, for example southern or northen parcels - in such a particular case, this will be modeled by new subconcepers. In our case it fully sufficient to use the concept *Parcel* (Fig. 3), which has instances as *Parcel_OH_2009_N* or *Parcel_COOH_2010_NO*.

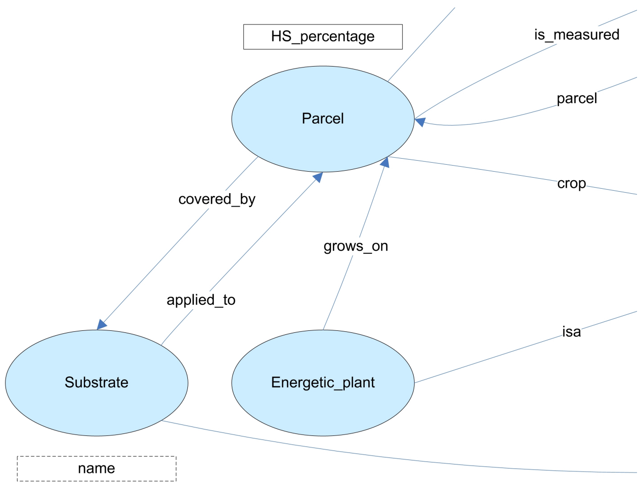

Figure 3: Concept *Parcel* need not further subconcepts (instances are enough).

2.5.3 Limiting the scope

The ontology should contain all available data about the descripted domain, but it may not specify other (relevant) terms, which are not used by any ontology application. In our case, it is not necessary to know a material of tools for collecting samples or an age of the person, which collects the samples. Similarly, the ontology may not include all roles and hierarchy contraints of concepts, which are not used by the application (in our case, there are modeled only compounds, which are relevant; compounds, which are not relevant are not provided). Only the most important (aimed to answer the question of competence) roles of concepts are given. All these decisions, if the roles or concepts are necessary, should be noted in an ontology documentation to simply understanding the ontology by other users.

2.5.4 Disjoint concepts

Most ontology designers allows to explicitly specify disjoint concepts. The concepts are disjoint, if they have no joined instances. For example, *Measurement_result* and *Experiment_result* are not disjunct because there may exists such experiment results, which are ident with *Measurement_results*.

2.5.5 Inverse roles

A value assigned to a role may depend on another role. For example, *Substrate* is on the *Parcel* and *Parcel* contains *Substrate*. These two roles are inverse. Keeping both inverse roles is rendant. If a given instance of *Substrate* has a role *is_on* assigned to a given instance of *Parcel*, the application using the ontology iffer also an inverse value of the relationship. This feature may be used for verification of modeled relationships (where a user can naturally read both relationships).

2.5.6 Default values

Many systems based on frames (for example [10]) allows to specify default values of roles. If the value of any role is identical for most instances of a given concept, such value can be defined as default. In such a case, if any instance do not explicitly redefine an assigned value (to another allowed value), a default value is automatically used.

2.5.7 Namimg convention

Naming convection and consequenlty its usage plays an important role for sound ontology understanding and often avoids common mistakes during extending ontology. Ofcourse, there exists manu convections. The convection includes a rules, how to call concepts and roles, several rules are given by a selection of tool for designing the ontology (for example Protégé [9]). These are

- different namespaces for concepts, roles and instances,
- letter case sentivenes,
- which special characters (for example _, !, etc.) may be used to name the concept?

These rules can be applied into a policy

- selecting separator, for example *ChemicalProcess*, *Chemical_process*, *chemical-proces*, *Chemical process*, ...),
- plular or singular, for example *ChemicalProcess* or *ChemicalProcesses*,
- role prefixes and postfixes, for example *is...*, *has-a*.

The recommended rule is not to call concepts, roles as *Class...* and not to abbreviate the names.

3. CONCLUSION

The paper presents a description of a design of the ontology for humine compounds. The paper was aimed by presenting rules for modelling the ontology and to discuss a chosen solution (as in Figures 1, 2 and 3). Each descibed step was extended by a teoretical background leading to creation of concepts, roles and their values.

The ontology design significantly depends on the designer; if two designers try to create an ontology from the same domain and application, both will follow all relevant rules, the designed ontologies can be totally different. The ontology quality can be, unfortunatelly, expressed by its sound usage in the proposed application.

4. ACKNOWLEDGMENTS

This project is supported by the grant GAP202/10/0761 “Web Semantization” and partly supported by the student grant SGS 2012/7821 “Interactive Mechatronics Systems Using the Cybernetics Principles”.

5. REFERENCES

[1] ChEBI. Ontology for chemical entities of biological interest. http://www.ebi.ac.uk/chebi, 2012. [Online].
[2] DAML. Darpa agent markup language. http://www.daml.org, 2012. [Online].
[3] T. Gruber. Ontology. In *Encyclopedia of Database Systems*, pages 1963–1965. Springer US, 2009.
[4] M. Grüninger and M. Fox. Methodology for the Design and Evaluation of Ontologies. In *IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing, April 13, 1995*, 1995.
[5] HCLS. Semantic web health care and life sciences interest group. http://www.w3.org/2001/sw/hcls, 2012. [Online].
[6] N. F. Noy and D. L. McGuinness. Ontology development 101: A guide to creating your first ontology. *Development*, 32(1):1–25, 2001.
[7] Ontolingua. Database and ontology. http://www.ksl.stanford.edu/software/ontolingua, 2012. [Online].
[8] OWL. Web ontology language. http://www.w3.org/2007/OWL, 2012. [Online].
[9] Protégé. Ontology editor and knowledge acquisition system. http://protege.stanford.edu, 2012. [Online].
[10] Protégé-Frames. Frame-based domain ontology editor. http://protege.stanford.edu/overview/protege-frames.html, 2012. [Online].
[11] SWOOP. Hypermedia-based owl ontology browser and editor. http://code.google.com/p/swoop, 2012. [Online].
[12] UNSPSC. The united nations standard products and services code. http://www.unspsc.org, 2012. [Online].
[13] M. Uschold and M. Grüninger. Ontologies: principles, methods, and applications. *Knowledge Engineering Review*, 11(2):93–155, 1996.
[14] W3C. The World Wide Web Consorcium. http://www.w3.org, 2012. [Online].