An ontology methodology and

CISP

- the proposed Core Information about Scientific Papers

by
Larisa Soldatova and
Maria Liakata

Version 1.0
First published December 2007
Publisher JISC
Copyright owner The University of Wales, Aberystwyth
Status of the document: project report (intermediate)
Project: ART – an ontology based tool for preparation of articles

© UWA 2007
Executive Summary

This report has two main goals:

- To introduce a new formalism for the description of scientific papers CISP (the Core Information about Scientific Papers);
- Attract more attention to ontologies as a valuable methodology for developing metadata.

Ontologies are a basic infrastructure for the Semantic Web. The idea of the Semantic Web is based on the possibility of using shared vocabularies for describing information resources. Ontologies provide unambiguous and machine-processable semantic metadata for Semantic Web applications. Ontologies are built on rigid theoretical foundations and can be used as proof of what concepts are essential for the description of a particular domain.

We demonstrate advantages of an ontology methodology for developing metadata by applying it to the analysis of the Dublin Core metadata (DC). An ontology approach allows detecting potential weaknesses in the representation of the DC terms. Such weaknesses include overlap in the semantic meaning between the terms, logically incoherent representation of temporal and spatial relations as well as incoherence in the representation of content. An ontology can also suggest improvements to the DC.

We used an ontology methodology to construct CISP metadata about the content of papers. It makes use of an ontology of experiments EXPO proposed at the University of Wales, Aberystwyth as a core ontology, and DOLCE (a Descriptive Ontology for Linguistic and Cognitive Engineering) developed at the Laboratory for Applied Ontology, the Institute of Cognitive Science and Technology, Italy as an upper level ontology. CISP is a defined set of leaf classes from these ontologies. It includes such key classes as <Goal of investigation>, <Object of investigation>, <Research method>, <Result>, <Conclusion>.

CISP can be used to generate abstracts and summaries of papers and also to facilitate storage and retrieval of information. CISP will constitute the basis for the ART tool. The latter is an authoring tool for the semantic annotation of papers stored in digital repositories. ART is intended for the semi-automatic annotation of data and metadata describing the scientific investigation represented in a research paper. ART will also be able to aid in the expression of research results directly in both a human and machine readable format, through the composition of text using ontology-based templates and stored typical key phrases.

To find out more about ontology methodology refer to chapters 2 and 3. To learn about the proposed CISP metadata you can start reading from chapter 4 onwards.

Status of the document: this is an intermediate project report about one of the outcomes of the ART project. We plan to submit a final version of the report about CISP by the end of the project (March, 2009).
Table of Contents

1. Introduction 4
   1.1 The developing CISP metadata 4
   1.2 An ontology engineering 4

2. An Ontology methodology for the Semantic Web 5
   2.1 Introduction into ontology methodology 5
   2.2 DOLCE 6
   2.3 EXPO 8

3. Ontology methodology for the analysis of the Dublin Core metadata 12
   3.1 Analysis of DC attributes 12
   3.2 Analysis of DC terms 16
   3.2.1 Describing representations 16
   3.2.2 Types and roles 16
   3.2.3 Representing time 17
   3.3 Discussion of a Dublin Core metadata example 17

4. CISP classes and underlying ontology 19
   4.1 CISP notation 19
   4.2 The key CISP classes 19
   4.3 Candidate classes for inclusion into CISP 29
   4.4 Verification of CISP 31

5. Conclusion 33
   About the authors 34
   Appendix: CISP survey results 35
   References 36
1. Introduction

Semantic Web technologies are focussed on using semantic metadata. Rich semantic representation can improve knowledge formalization and information retrieval. “Metadata can be defined literally as "data about data," but the term is normally understood to mean structured data about digital (and non-digital) resources that can be used to help support a wide range of operations” 1. “Metadata is structured information that describes, explains, locates or otherwise makes it easier to retrieve, use, or manage an information resource” [NISO 2004].

There are various sets of metadata: the Dublin Core Metadata2, the Meta Content Framework (MCF)3, Platform for Internet Content Selection (PICS)4, RDF Site Summary (RSS)5. We will describe in this report the Dublin Core Metadata (DC) as one of the most popular and the most relevant set to our project. We will apply an ontology methodology for the analysis of the DC to demonstrate the advantages of our approach for the development of metadata in section 3.

1.1 The developing CISP metadata

Our approach is to exploit an ontology methodology with its coherent logic, clear semantics and explicit definitions of the elements used for the development of metadata to describe scientific papers.

Our analysis of papers is based on an ontological representation of investigations. A scientific paper is one of many ways (the most typical) of representing investigations. Our main assumption is that a scientific paper as a representation of the content of a scientific investigation needs to contain the key concepts for the description of investigations. First, we identified what concepts are essential for the description of scientific investigations. Second, we proposed a set of the most essential concepts for representing scientific papers – the Core Information about Scientific Papers (CISP). The principle difference between other metadata schemas and CISP is that the latter aims to represent not just what is typically reported in scientific papers, but what should be reported to convey a complete scientific investigation.

We restrict our set of investigations to ones where the research is driven by experimental methods. Our understanding of an experimental method is broad: physically executing experiments, computationally running experiments, or theoretical experiments. Section 4 has a detailed description of CISP classes.

1.2 An ontology engineering

Ontology engineering is still a relatively new research field. Therefore, many of the steps in designing an ontology remain unformalized and can be considered an “art” [Schulze-Kremer 2001]. We give a brief description of the basic ideas of an ontology methodology in section 2. We explain the major components of typical ontologies as well as the principles and different approaches to ontology design.

1 http://www.ukoln.ac.uk/metadata/
2 http://uk.dublincore.org/
3 http://www.textuality.com/mcf/NOTE-MCF-XML.html
4 http://www.w3.org/PICS/
5 http://www.techxtra.ac.uk/rss_primer/
2. An Ontology methodology for the Semantic Web

This section describes ontology as one of the basic representatives of Semantic Web technologies. An ontology approach provides semantic clarity, explicitness, and facilitates the reusability of represented information and knowledge. We explain the major components of typical ontologies, and the principles and different approaches to ontology design. We describe then two ontologies DOLCE (a Descriptive Ontology for Linguistic and Cognitive Engineering) and EXPO (an Ontology of scientific EXPeriments) that we used for the development of CISP (the Core Information about Scientific Papers).

The use of formalized semantic representation can also facilitate natural language processing for intelligent information analysis and retrieval. Therefore ontology based knowledge representation opens new perspectives for text mining techniques and logic inference.

2.1 Introduction into ontology methodology

Semantic Web technologies are based on using semantic metadata. Rich semantic representation can improve knowledge formalization and information retrieval. The full power of Semantic Web applications can only be efficiently exploited when the knowledge they work with is formalized. The first step in formalizing knowledge is to define an explicit ontology. An ontology can be used to define what metadata are required for the representation of information resources. The exact definition of what an ontology is varies between disciplines. We follow Schulze-Kremer's definition: “a concise and unambiguous description of what principle entities are relevant to an application domain and the relationships between them” [Schulze-Kremer 2001].

An ontology consists of four main components: classes, a hierarchical structure (is-a relations), relations (other than is-a relations), and axioms (axioms are used to associate classes and their properties with specifications and to give other logical information about classes and properties). Ontology engineering aims to provide a methodology for ontology development and maintenance. Riichiro Mizoguchi, one of the leading experts of ontological engineering, lists the fundamental tenets of ontological classes, instances, and is-a relations [Mizoguchi 2004b]:

1. Intrinsic property. The intrinsic property of a thing X is a property which is essential to the thing X such that it loses its identity when the property changes.

2. The ontological definition of a class. X is a class if and only if (iff) each element x of X satisfies the intrinsic property of X. If and only if (iff) this definition holds then the relation <x instance-of X> is true.

3. Is-a relation. <class A is-a class B> relation holds between classes if and only if (iff) every instance of the class A is also an instance of the class B.

Ontology engineering is still a relatively new research field. Therefore, many of the steps in designing an ontology remain unformalized and can be considered an “art” [Schulze-Kremer 1997]. Existing ontologies vary in the level of explicitness, formality and expressiveness. The two major approaches to developing an ontology are bottom-up and top-down [Mizoguchi 2004a]. In the bottom-up approach, developers usually start with an existing problem, and a list of domain concepts or a controlled vocabulary. The concepts are then organized into ontological classes and individuals, with the addition of relations between classes, and axioms.

In the top-down approach to ontology development, ontology designers also start with the

---

http://www.w3.org/TR/owl-features/
problem, but use an appropriate upper ontology to guide the developing ontology. There are a number of upper ontologies available: SUMO (The Suggested Merged Upper Ontology), OpenCYC, BFO (The Basic Formal Ontology). SUMO and OpenCYC come from general AI (artificial intelligence), whereas BFO is an upper level ontology for biomedical ontologies; BFO provides basic ontological elements and relations that are required for biomedical domains [Fielding et. al., 2004]. None of these upper level ontologies constitutes an ideal representation of the world - and perhaps we will never have a perfect one. Therefore, an ontology engineer has to make a compromise between the imperfection of the model representation, and one's practical needs. The advantages of an upper ontology as a reference model for designing a domain ontology include:

- A template structure of terms, and relations, along with key definitions and axioms;
- An approved way of determining the top level concepts;
- Compliance with other ontologies enabling cross ontology use and inference.

Any imperfections in an upper level ontology are usually less damaging to the structure of a the domain ontology than the absence of any upper ontology. Selection of the most appropriate upper ontology can significantly contribute to the quality of the final ontology, ease the design process, and guarantee its wide reusability. Use of upper ontologies on a design stage as a guidance for domain ontologies helps to avoid many of typical errors and can ease the ontology validation process.

The proposed CISP formalism follows the ontology methodology. It uses EXPO as a core ontology and DOLCE as an upper level ontology.

2.2 DOLCE

DOLCE (a Descriptive Ontology for Linguistic and Cognitive Engineering) is an upper level ontology. Here we give only a brief overview of DOLCE. To fully appreciate DOLCE advanced knowledge of logics is required. The interested reader is advised to visit the web site of the Laboratory for Applied Ontology, where it is possible to download the DLP (DOLCE-Lite-Plus) and check the Wonder Web project deliverables.

DOLCE has a cognitive bias. It aims to represent major classes for reasoning over human commonsense and natural language. DOLCE introduces classes for cognitive artefacts depending on human perception, cultural imprints and social conventions [Gangemi et. al., 2002]. DOLCE is based on the fundamental division between continuant and occurrent entities (or enduring and penduring entities in terms of DOLCE). Continuant entities are time independent, they are fully present at any time point of their existence. Examples of continuant entities are physical objects and qualities. Occurrent entities are time dependent, they are only partially present at any time point. Examples of occurrent entities are events and processes. DOLCE also has abstract entities, entities that do not have spatial nor temporal qualities. Figure 1 shows a fragment of the ontology.

DOLCE has a rich set of axioms and is suitable for logic inference. Some of the axioms can not be expressed in the standard ontology language OWL, and they are coded in KIF.

8 [http://www.opencyc.org/](http://www.opencyc.org/)
9 [http://ontology.buffalo.edu/bfo/BFO.htm](http://ontology.buffalo.edu/bfo/BFO.htm)
10 [http://www.loa-cnr.it/DOLCE.html](http://www.loa-cnr.it/DOLCE.html)
11 [http://www.w3.org/TR/owl-features/](http://www.w3.org/TR/owl-features/)
(knowledge interchange format). KIF is a computer-oriented language for the interchange of knowledge among disparate programs. It has declarative semantics (i.e. the meaning of expressions in the representation can be understood without appeal to an interpreter for manipulating those expressions); it is logically comprehensive (i.e. it provides for the expression of arbitrary sentences in first-order predicate calculus); it provides for the representation of knowledge about the representation of knowledge; it provides for the representation of nonmonotonic reasoning rules; and it provides for the definition of objects, functions, and relations.

![Diagram](http://www-ksl.stanford.edu/knowledge-sharing/kif)

Figure 1: A fragment of DOLCE in the Protégé ontology editor.

In contrast with ‘lightweight’ ontologies, which focus on a minimal terminological structure, the main purpose of foundational ontologies like DOLCE “is to negotiate meaning, either for enabling effective cooperation among multiple artificial agents, or for establishing consensus in a mixed society where artificial agents cooperate with human beings” [Gangemi et al., 2002]. The necessity of an upper level formalism is demonstrated in an example: “Law, which has two hyperonyms Legal document and Rule. According to DOLCE, we consider Legal document as being subsumed by Non-Agentive Physical Object, and Rule as subsumed by Non-Agentive Social Object. So the two categories are

http://www-ksl.stanford.edu/knowledge-sharing/kif/

http://protege.stanford.edu/
disjoint. Consequently, this multiple hyperonymy generates a logical incoherence, which could not be detected without an explicitly axiomatized upper-level” [Gangemi et. al., 2002].

According to one's needs, an upper level ontology can be used either in a light version for computationally intensive applications, or as a fully axiomatized theory. For the development of CISP and further within the ART project, we use DOLCE in a light version. In fact, a generic or domain ontology built as an extension of an upper level ontology, ‘hides’ the top classes of the former from the users. The users of the lower level ontology can be unaware of the logics behind it and work only with the leaf classes. We used this principle to develop CISP – it is a set of leaf classes from an ontology for the description of papers, which is built as an extension of DOLCE.

2.3 EXPO

In this section we describe an ontology of scientific experiments EXPO that was developed at the University of Wales, Aberystwyth [Soldatova & King 2006]. EXPO provides a clear structured framework for a consistent and shareable description of experiments for both humans and computer systems. EXPO links a general upper ontology SUMO with subject-specific ontologies of experiments by formalizing the generic concepts of experimental design, methodology, and results representation. EXPO is expressed in the W3C standard ontology language OWL-DL. EXPO has received a lot of publicity with articles appearing in the New Scientist and the Chronicle of Higher Education, and received a 2006 nomination for World Technology Award (software).

EXPO is based on ideas from the philosophy of science (logical, probabilistic, methodological, epistemological, etc.) [Curd 1988; Toulmin 2004], the theory of knowledge representation [Sowa 2000], the analysis of existing ontologies14 including bio-ontologies15, and the theory of experiment design [Boniface 1995; Fisher 1956].

Despite their different subject matter, all the sciences organize, execute, and analyze experiments in similar ways; they use related instruments and materials; they describe experimental results in identical formats, dimensional units, etc. From a knowledge management and ontology engineering point of view, this knowledge should be stored in only one place, a generic ontology of experiments, to ensure knowledge consistency and easy update [Soldatova & King 2005].

We have therefore proposed a general ontology of scientific experiments EXPO [Soldatova & King 2006]. EXPO defines general classes including <ScientificExperiment> (“a research method which permits the investigation of cause-effect relations between known and unknown (target) variables of the field of study (domain). An experimental result cannot be known with certainty in advance”), <ExperimentGoal> (“the state that a plan is intended to achieve and that (when achieved) terminates behaviour intended to achieve it” [Collins 1993]), <ExperimentTechnology> (“the total knowledge (theory, methods, and practices), and machinery available to any object of an experiment”16), <ExperimentResult> (“the set of facts and conclusions, obtained as a result of the interpretation of the experimental observations, which increase/decrease the probability of a research hypothesis of the experiment”), etc. Figures 2 and 3 show fragments of EXPO.

14 suo.ieee.org/
15 obo.sourceforge.net
16 http://www.cogsci.princeton.edu/cgi-in/webwn2.0?stage=1&word=goal
Figure 2: EXPO class <ScientificExperiment>, where p/o is a part-of relation, a/o is an attribute-of relation.

The EXPO domain is a general formalized representation of scientific experiments. EXPO is able to describe computational and physical experiments, experiments with explicit and implicit hypotheses. EXPO follows the SUMO naming convention, e.g. NameOfClass. Definitions are based on desiderates that date back as far as Aristotle. This is consistent with arguably one of the best ontologies FMA (the Foundational Model of Atonomy) [Rosse & Mejino 2003]. EXPO used SUMO as a prototype ontology and has such top classes as <Abstract> and <Physical> entity, <Proposition>, <Attribute>, <Role>, <Representation>, <Object>, <Process>. Such an upper hierarchy allows easy and flexible structuring of the classes for the description of experiments.
Figure 3: The Ontology of Scientific Experiments (a fragment), where p/o is a part-of relation, a/o is an attribute-of relation, and an arrow with an empty label corresponds to is-a relation.

The class <Abstract> combines such subclasses as <Proposition> for formalizing experiment goals, tasks, hypotheses, methods, etc.; <Role> for description of actor roles (a submitter, a user, a performer of an experiment, an object of an experiment), functional roles for description of functionality of experiment equipment, etc. The class <Physical> describes objects (materials, groups, artifacts) and processes (experiment actions, scientific activities as experiment designing, result interpreting, etc.). The class <Representation> combines various representations: representation of experiment model (logical, mathematical), representation of experiment observations and results.

There are related projects to represent knowledge about experiments. Functional Genomics Investigation Ontology (FuGO) and its successor OBI (ontology for biomedical investigations)\(^{17}\) is developing an integrated ontology for the description of biological and medical experiments and investigations. This includes a set of 'universal' terms, which are applicable across various biological and technological domains, and domain-specific terms relevant only to a given domain\(^{17}\). EXPO provides a representation for domains wider than biomedical domains, but intends to be compliant with OBI formalism.

EXPO was developed in the Hozo Ontology Editor [Kozaki et. al., 2002] and then automatically translated into standard OWL format. EXPO is publicly available at: sourceforge.net/projects/expo.

\(^{17}\) [http://obi.sourceforge.net/](http://obi.sourceforge.net/)
EXPO was used as a core ontology to develop CISP metadata. EXPO defines important classes for representing scientific investigations, and therefore can be used for description of papers reporting results of scientific investigations. The ART project will use EXPO as a core ontology, and a number of other ontologies, for marking-up scientific papers stored in digital repositories.
3. Ontology methodology for the analysis of the Dublin Core Metadata

The Dublin Core Metadata Initiative\(^1\) is an open organization engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models. The Dublin Core Metadata (DC) is arguably the most popular set of metadata for the annotation of various resources including papers, or e-Prints in many of JISC projects terminology\(^2\). We consider DC metadata as the most suitable metadata for the goals of our project. In this section, we give an ontology analysis of DC in order to demonstrate the advantages of an ontology approach, how this approach can suggest improvements for the current set of DC terms and propose more meaningful semantic representation for some of them.

We would like to make a number of remarks before the analysis of DC to avoid possible misunderstandings. This analysis has the purpose of demonstrating the ontology methodology on an example of a well known metadata set, and not to criticise the DC. The DC is not an ontology, and formally should not be analysed from an ontology point of view. We are doing so in order to show what the DC could look like, if it employed an ontology methodology. To our estimates the development process of the DC is moving in the ‘ontology direction’. The DC already has many ontology components. There already exists a DC ontology, see for example Protégé\(^3\). The DC ontology is only a list of fifteen DC terms and it does not have a hierarchical structure or defined properties, so it is not ‘a true’ ontology as we understand it. It is not difficult to re-code the current full version of the DC into an ontology, and perhaps the developers will do it at some point. It is a matter of agreement and not a technical issue.

We based our analysis on the DC version from the Dublin Core Metadata Initiative webpage: \(\text{http://dublincore.org/documents/dcmi-terms/}\) (accessed in December 2007). We use the most common ontology terminology consistent with the language OWL and ontology editor Protégé. We use italic font for properties and angle brackets to indicate classes and instances in the text, names of classes starting with a capital letter. We use pseudo code instead of OWL to simplify the examples.

From an ontology methodology point of view the DC includes the following ontological components:

- **Class** – the majority of DC elements, element refinements, encoding schemes, and vocabulary terms can be considered as ontological classes. We provide analysis of some problems in the representation of DC classes in sec. 3.2.
- **Property** - many of DC attributes are properties from an ontology point of view. We provide a detailed analysis of DC attributes below. Such DC terms as `hasPart`, `isPartOf` are classic well defined ontological properties; such DC terms as `hasFormat`, `hasVersion` and many others are also properties.

### 3.1 Analysis of DC attributes

Here we list DC attributes along with their names and DC definitions and explain how to change the DC representation into an ontology one. We show that the DC representation of attributes in many cases can be simplified and improved. Special languages for writing ontologies, such as OWL, are especially designed to describe ontological terms and

\(^1\) \text{http://dublincore.org/}

\(^2\) \text{http://www.ukoln.ac.uk/}

\(^3\) \text{http://protege.stanford.edu/}
already include many required components. Many of the DC attributes would be simply redundant in an ontology representation. An ontology analysis shows that some of the DC attributes do not capture the intended meaning of the term (for example <DC: References>, see below) or could be represented better, more explicitly and intuitively.

The DC has the following attributes:

1. **Name**: ‘the unique token assigned to the term’. There would be no need to define name as a property because the property `hasLabel` with the same semantics has already been defined for each ontological class.

2. **URI**: “the Uniform Resource Identifier used to uniquely identify a term.” There would be no need to define URI as a property because it is already defined for each ontological class.

3. **Label**: “the human-readable label assigned to the term”. This is a duplication, because the DC attributes <name> and <label> have the same semantics.

4. **Definition**: “a statement that represents the concept and essential nature of the term.” It is a property. It would be represented in an ontology as `has Definition`.

5. **Type of Term**: “The type of term, such as Element or Encoding Scheme, as described in the DCMI Grammatical Principles.” This attribute would be redundant in an ontology. Types of ontological components like class and instance are already defined. The name ‘type’ is not a good ontological name, because each class is a type of something.

6. **Status**: “Status assigned to term by the DCMI Usage Board, as described in the DCMI Usage Board Process.” It is a property. It would be represented in an ontology as `has Status`.

7. **Date issued**: “Date on which a term was first declared.” It is a property. It would be represented in an ontology as `has Date of issue`.

8. **Comment**: “Additional information about the term or its application.” There would be no need to define this attribute. The property `has Comment` is already defined for each ontological term.

9. **See**: “A link to authoritative documentation.” It is a property, but the name is not an ontological name, the name of this attribute is ambiguous. It is better to change it to, for instance `has Reference`.

10. **References**: “A citation or URL of a resource referenced in the Definition or Comment.” According to the definition, this attribute describes not the DC term, but another entity: Definition or Comment. Thus it is NOT a property (or attribute) of DC term. Definition and Comment are not defined as DC terms at all. To represent that a definition and a comment can have references, first they have to be defined as separate entities: the class `<Definition>` and the class `<Comment>`, second `has Reference` should be defined as a property of the class `<Definition>` and as a property of the class `<Comment>`. The cardinality of these properties is multiple, so a definition and a comment can have several references, and it should not be expressed by plural form of the name References. There will be no clashes with the previous property `<DC: see>` if we rename it to `has Reference`, because different classes can have references.

11. **Refines**: “A reference to a term refined by an Element Refinement”. This attribute tries to represent the structure. It would be replaced in an ontology by an `is-a` relation with clear formalized semantics.
12. **Qualifies**: “A reference to a term qualified by an Encoding Scheme”. We would suggest a different representation to capture the meaning of this element, see below.

13. **Broader Than**: “A reference from a more general to a more specific Vocabulary Term”. This attribute also tries to represent the structure. It would be replaced in an ontology by an *is-a* relation with clear formalized semantics.

14. **Narrower Than**: “A reference from a more specific to a more general Vocabulary Term”. This attribute tries to represent the structure, see the comment above.

In general, the DC attributes might be represented by defining a property *has Attribute* and the corresponding class `<Attribute>`.

For example each DC term must have a status with possible values: registered or recommended. A pseudo code for this example:

```plaintext
class: Status
property: has Status
domain of property: registered; recommended;

<DC Term> has Status only Status
Cardinality: 1
```

We would suggest a better representation for the DC term `<Qualifies>`. For example, note that the DC term `<Subject>` does not have the attribute `<Qualifies>`. DC gives only a comment: “Typically, the topic will be represented using keywords, key phrases, or classification codes. Recommended best practice is to use a controlled vocabulary. To describe the spatial or temporal topic of the resource, use the Coverage element.” A user has to infer from the description of the term `<Qualifies>` that subject should be represented by DCC (Dewey Decimal Classification) or other classification code.

We would formalize this fragment differently. Subject can have several representations, and it can be represented by key words or by a classification code. There are several classification systems: LCC (Library of Congress Classification), NLM (National Library of Medicine Classification), MESH (Medical Subject Headings), and LCSH (Library of Congress Subject Headings). We also would add one more useful classification system: RSUK classification (a classification of domains used by Britain's councils for academic research). In order to simplify explanations, we do not discuss here the fact that `<Subject>` is actually a role played by another entity. Our representation is shown in the figure 4 below:
Figure 4: An ontology-based representation of the situation where the class <Subject> can be represented by classification codes and key words.

A pseudo code for this example:

```plaintext
class: Subject
has representation: Classification code
has representation: Key word

property: has Representation
cardinality of property: multiple

class: Representation
class: Classification code
class: LCC
class: NLM
class: MESH
class: LCSH
class: RSUK classification
```

Classification by one or more widely excepted classification systems gives better identification of what the area of the resource is than the key words provided by the authors. An unambiguous classification number associated with the domain of the resource can be used then for search engines, or for other tasks, for example - selecting a reviewer for a submitted paper. It is easy to associate each area covered by the classification systems with a set of key words and to give key words if necessary. The inverse operation (assigning key words to a subject) is not always successful, because one word can be related to several domains, and it is unlikely that authors or experts can manually provide a set of key words good enough to correctly associate a domain of a paper to a classification number. This is a topic for further investigation.

Other cases of <Qualifies> correspond to different relations that we would define in a way similar to the one described above.
3.2 Analysis of DC terms

The analysis of DC terms is more complicated than the analysis of DC attributes. Here we show some weaknesses in representing DC terms that can cause problems in future extensions and applications, but we do not give analysis of all the terms.

3.2.1 Describing representations

Let us consider the following DC terms:

- **Format**: “The file format, physical medium, or dimensions of the resource. Examples of dimensions include size and duration. Recommended best practice is to use a controlled vocabulary such as the list of Internet Media Types [MIME]”.

- **Type**: “Recommended best practice is to use a controlled vocabulary such as the DCMI Type Vocabulary [DCMITYPE]. To describe the file format, physical medium, or dimensions of the resource, use the Format element.”

- **Extent**: “the size or duration of the resource”.

- **Medium**: “the material or physical carrier of the resource”.

These terms have duplications and overlaps in their semantics. It is enough to give only a few comments on the first definition to show weakness in the representation: a format is not a physical medium, a file format and dimension of the resource are completely different entities. We leave it to the reader to find out other overlaps in the listed above terms.

An ontology solution for representing <Proposition> (a piece of information) is to define the classes <Content>, <Representation> and <Medium> [Mizoguchi 2004b]. The same content can have various representations, and the DC already has terms <Image>, <MovingImage>, <Text>, etc. The class <Medium> is correctly defined in DC, but it has overlaps with other DC terms and it does not have sub-classes describing typical media of the resources: book, CD, etc. All attributes of the medium as a physical carrier of the resource, like size, material, should be defined as properties of this class. From this point of view, the DC term <Language> is a characteristic of the class <Representation>, not the resource itself as it is defined in DC now.

Such an ontological representation will solve a problem of polymorphism, that one digital object can be i.e. a journal and a learning object. An ontology can easily deal with this situation: the digital object has the same content, but different representations and intended audiences.

3.2.2 Types and roles

Let us consider the following DC terms:

1. **Contributor**: “an entity responsible for making contributions to the resource; examples of a Contributor include a person, an organization, or a service”.

2. **Creator**: “an entity primarily responsible for making the resource; examples of a Creator include a person, an organization, or a service”.

3. **Publisher**: “an entity responsible for making the resource available; examples of a Publisher include a person, an organization, or a service”.

16
Such a representation does not handle well the situation where the same person can be for example a creator and a publisher. It is not good practice to input data for the same person twice in different places, as it is a recipe for inconsistency and errors. The reason why DC representation is not logically coherent, is that the terms <Contributor>, <Creator>, <Publisher> are not types (as classes should be), but roles. The standard ontological solution is to define the class <Person> and roles that individuals of this class can play in different situations: <Contributor>, <Creator>, <Publisher>. In this case you need to input data for a person only once and it is easier to keep them updated.

3.2.3 Representing time

The DC has many terms representing temporal relations: <Modified>, <Created>, <dateAccepted>, <dateCopyrighted>, <Available>. Why are the names not <dateCreated>, <dateAvailable> like the other names of the terms? They would be more meaningful. DC defines <Available> as a “date (often a range) that the resource will become or did become available”, but it is possible to assume from the name that the meaning is whether the resource is available or not.

We would strongly recommend using Time Ontology to represent temporal relations. Time Ontology in OWL is a part of the HYPERLINK "http://www.w3.org/2001/sw/Activity" W3C Semantic Web Activity. The ontology provides a vocabulary for expressing facts about topological relations among instants and intervals, together with information about durations, and about datetime information. Time Ontology is logically consistent and can be used not only for a better representation of DC terms, but also for reasoning over temporal relations.

The usage of Time Ontology will ensure the correct logical representation of DC temporal terms and provide compliance with Semantic Web applications.

Some others DC terms, for example <Box>, <Point> for representing spatial relations, or the term <Relation> can be better represented by exploiting an ontology approach.

3.3 Discussion of a Dublin Core metadata example

An ontology methodology is a valuable approach for the development and improvement of metadata sets. In this section we have considered DC metadata as an example and suggested how to improve the DC representation. We have suggested the following:

1. Use an explicit well defined semantics of relations between terms.
2. Simplify representation of attributes.
3. Avoid duplication and overlap in semantic meaning of DC terms.
4. Use more meaningful names and definitions.
5. Improve representing of spatial and temporal relations by re-using ontology formalisms.
6. Distinguish types and roles.
7. Use the best practise in formalizing of representations of contents.

We have already performed similar analyses and demonstrated how an ontology

http://www.w3.org/TR/owl-time/

http://www.w3.org/2001/sw/Activity
methodology can improve representations [Soldatova & King, 2005; Schierz et al., 2007]. We conclude that the same approach can be applied to improve the DC metadata and for the development of other metadata sets.
4. CISP classes and underlying ontology

4.1 CISP notation

In this section we follow a format familiar to the reader for describing metadata terms. However many of the listed properties are already built into OWL. CISP follows the naming convention proposed by MSI (The Metabolomics Standards Initiative) working group: <class of instances>.

Each CISP class has the following mandatory properties:

- Definition (and Definition reference);
- Location in paper (value = # paper section);
- Representation (default value = natural language text).

Each CISP class also has the following properties and we list them in the description of the CISP classes, even though they are already built into OWL:

- Class name;
- Class ID;
- Cardinality;
- Parent class;
- Subclass;
- Comment.

CISP class can also have the following elements:

- Informal explanation;
- Example;
- and other properties.

CISP includes eight key classes. CISP applications assume mark-up of all these classes in the papers. Many of the key classes have subclasses and properties. They are given in CISP for the better understanding of the semantic meaning of instances that can be found in papers. Mark-up of papers that uses the subclasses and properties of the key classes would provide more semantics to the annotation, but it is not required. For example, if a research goal in the paper is marked-up as <discover-goal>, not just <goal of investigation>, it would provide more semantics about the nature of the goal and about the investigation. Because <discover-goal> is a subclass of the class <goal of investigation>, any computer system will be able to map this mark-up to the key CISP classes.

4.2 The key CISP classes

We propose the following eight key classes for the description of scientific papers:

23 http://msi-ontology.sourceforge.net/
4.2.1 Goal of investigation

<table>
<thead>
<tr>
<th>Class name:</th>
<th>&lt;goal of investigation&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class ID:</td>
<td>CISP 1</td>
</tr>
<tr>
<td>Cardinality:</td>
<td>multiple</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td>What does an investigation aim to show? What problem does it aim to solve?</td>
</tr>
<tr>
<td>Definition:</td>
<td>a goal of an investigation is the target state of the investigation where intended discoveries are made, approaches are tested, problems are demonstrated, tasks formulated etc.</td>
</tr>
<tr>
<td>Definition reference:</td>
<td>CISP</td>
</tr>
<tr>
<td>Comment:</td>
<td>the definition is derived from the definition of the parent class &lt;goal&gt;</td>
</tr>
<tr>
<td>Parent class:</td>
<td>&lt;goal&gt;</td>
</tr>
<tr>
<td>Definition:</td>
<td>“A goal is the state that a plan is intended to achieve and that (when achieved) terminates behaviour intended to achieve it”.</td>
</tr>
<tr>
<td>Subclass:</td>
<td>&lt;confirm-goal&gt;, &lt;explain-goal&gt;, &lt;demonstrate-goal&gt;, &lt;discover-goal&gt;, &lt;observe-goal&gt;, &lt;compute goal&gt;</td>
</tr>
<tr>
<td>Example:</td>
<td>development a new approach</td>
</tr>
<tr>
<td>Location in paper:</td>
<td>paper section: introduction</td>
</tr>
<tr>
<td>Representation:</td>
<td>natural language text</td>
</tr>
</tbody>
</table>
In the literature the most typical goals of an investigation are: to ascertain, to establish, to venture, to discover, to investigate, to infer a fact or theories, the existence of something; to examine, to verify, to falsify, to test a hypotheses, theories, ideas, causal relationships; to gather, to take measurements, to observe data, facts, and other outcomes, etc.

We summarize the goals of an investigation in the list:

1) to check or support a theory on an axiom-deductive basis (a theory-driven approach);
2) to discover a cause-effect dependency on an abductive/inductive basis (it is not driven by a theory, but a theory might be suggested to explain the experiment results);
3) to demonstrate a known truth (Aristotelian investigation [Medawar, P.B.]);
4) to “find out what happens” in “artificially created situation which allows researcher to manipulate variables” (it is an experiment in Baconian understanding [Medawar, P.B.]);
5) to observe a phenomena;
6) to compute values of an entity of interest.

We call these goals correspondingly: to confirm, to explain, to demonstrate, to discover, to observe, and to compute.

We define the follows subclasses of the class <goal of investigation>:

- **<confirm-goal>**
  Definition: a confirm-goal is a goal of an investigation that uses hypothesis-driven experiments to achieve the goals of the investigation.
  Definition reference: EXPO

- **<explain-goal>**
  Definition: an explain-goal is a goal of an investigation that uses hypothesis-forming experiments to achieve the goals of the investigation.
  Definition reference: EXPO

- **<demonstrate-goal>**
  Definition: a demonstrate-goal is a goal of an investigation that uses Aristotelian experiments (to demonstrate a known truth) to provide evidence for already known knowledge.
  Definition reference: CISP

- **<discover-goal>**
  Definition: a discover-goal is a goal of a Baconian investigation (find out what happens).
  Definition reference: EXPO

- **<observe-goal>**
Definition: an observe-goal is a goal of an investigation that uses a descriptive method of research.

Definition reference: CISP

Definition: a compute-goal is a goal of an investigation that uses computational experiments to achieve the goals of the investigation.

Definition reference: EXPO

Representation of a goal is usually natural language text, but it can be a logic expression like in the case of the Robot Scientist experiments [King et. al., 2004].

An investigation can have several different goals and a goal usually can be decomposed into subgoals. For example if an investigation includes three experiments, then the goal of the investigation can be decomposed into a set of sub-goals, and each of them constitutes a goal of the experiment. Example: a goal of an investigation is ‘development of a new approach’. In this example, the goal of the investigation is to achieve the target state of the investigation where a new approach has been developed. This goal can be decomposed onto sub-goals: to test the method properties, to compare to other methods, etc.

### 4.2.2 Motivation

<table>
<thead>
<tr>
<th>Class name:</th>
<th>&lt;motivation of investigation&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class ID:</td>
<td>CISP2</td>
</tr>
<tr>
<td>Cardinality:</td>
<td>multiple</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td>Why is an investigation important?</td>
</tr>
<tr>
<td>Definition:</td>
<td>Motivation for an investigation is the stimulus for achieving the goal of the investigation, the reason to carry out the investigation.</td>
</tr>
<tr>
<td>Definition reference:</td>
<td>CISP</td>
</tr>
<tr>
<td>Comment:</td>
<td>the definition derived from the definition of the parent class concept &lt;motivation&gt;</td>
</tr>
<tr>
<td>Parent class:</td>
<td>&lt;motivation&gt;: The (conscious or unconscious) stimulus for action towards a desired goal, esp. as resulting from psychological or social factors; the factors giving purpose or direction to human or animal behavior. Now also more generally (as a count noun): the reason a person has for acting in a particular way, a motive.</td>
</tr>
<tr>
<td>Definition reference:</td>
<td>OED</td>
</tr>
<tr>
<td>Subclass:</td>
<td>&lt;application to new domain&gt;</td>
</tr>
<tr>
<td>Example:</td>
<td>application of the method to quantum mechanics</td>
</tr>
</tbody>
</table>
Motivation for the investigation is often stated in the introduction and discussion sections of a paper. There can be several motives.

### 4.2.3 An object of the investigation

<table>
<thead>
<tr>
<th>Class name:</th>
<th>&lt;object of investigation&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class ID:</td>
<td>CISP3</td>
</tr>
<tr>
<td>Cardinality:</td>
<td>multiple</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td>About which entities do we seek to gain new knowledge, by means of the investigation?</td>
</tr>
<tr>
<td>Definition:</td>
<td>An object of an investigation is the principal entity on which the investigation is based.</td>
</tr>
<tr>
<td>Parent class:</td>
<td>&lt;role&gt;</td>
</tr>
<tr>
<td></td>
<td>role holder: &lt;method&gt;, &lt;process&gt;, &lt;object&gt;</td>
</tr>
<tr>
<td>Example:</td>
<td>model of yeast metabolism (an instance of the class &lt;abstract object&gt;)</td>
</tr>
<tr>
<td>Advantage property of object:</td>
<td>features of the object that give advantages in certain situations compared to other objects</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td></td>
</tr>
<tr>
<td>Disadvantage property of object:</td>
<td>features of the object that give disadvantages in certain situations compared to other objects</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td></td>
</tr>
<tr>
<td>Location in paper:</td>
<td>paper section: introduction, methods</td>
</tr>
<tr>
<td>Representation:</td>
<td>natural language text;</td>
</tr>
<tr>
<td></td>
<td>link to a classification system of objects</td>
</tr>
</tbody>
</table>

The word ‘object’ in the name of the class should not be understood directly. It does not mean that an object of an investigation is necessarily some physical object. The class <object of investigation> represents what an investigation is about. An object of an investigation is regarded as a role played by some entity (a role holder) in a particular situation: an investigation. Different entities can play a role of an object of an investigation: a method, a process, an animal, a robot (see [Sunagawa et. al., 2005]) for more details about roles). Consider an example where an object of an investigation is a robot. The investigation can be about testing of how this robot is robust and efficient in
extreme situations. But in other investigations the same robot can play a role of equipment, e.g. to collect some samples. Consider an example where an object of an investigation is a new method. A class <method> is ontologically defined as <proposition>, not as <object>, but method can have different roles (be a role holder) including <object of investigation>. An investigation about a new method would include studying and testing properties of the method (i.e. accuracy, efficiency), demonstration of areas of application, comparison of its performance to other methods.

An investigation can have several objects of investigation and is often classified according to the object (and it is another role: a base of a classification system).

### 4.2.4 Method of the investigation

<table>
<thead>
<tr>
<th>Class name:</th>
<th>&lt;research method&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class ID:</td>
<td>CISP4</td>
</tr>
<tr>
<td>Cardinality:</td>
<td>multiple</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td>How did the authors achieve the goal of the investigation?</td>
</tr>
<tr>
<td>Definition:</td>
<td>A research method is a way to solve a scientific task &quot;based upon or regulated by science, as opposed to mere traditional rules or empirical dexterity.&quot;</td>
</tr>
<tr>
<td>Parent class:</td>
<td>&lt;method&gt;</td>
</tr>
<tr>
<td>Subclass:</td>
<td>&lt;experimental method&gt;, &lt;analytical method&gt;</td>
</tr>
<tr>
<td>Advantage of method:</td>
<td>features of the method that give advantages in certain situations compared to other methods</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td></td>
</tr>
<tr>
<td>Disadvantage of method:</td>
<td>features of the methods that give disadvantages in certain situations compare to other methods</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>the DNA detection method</td>
</tr>
<tr>
<td>Location in paper:</td>
<td>paper section: methods; methods and materials</td>
</tr>
<tr>
<td>Representation:</td>
<td>natural language text; protocol</td>
</tr>
</tbody>
</table>

CISP includes the following research methods:

- <experimental method>
An experimental method is a way to solve a scientific task by designing and executing a scientific experiment.

Definition reference: EXPO

> <analytical method>

Definition: an analytical or theoretical method is a way to solve a scientific task by operating with abstract entities like models, theories on deductive basis.

Definition reference: CISP

What is the difference between an experimental method and other scientific methods such as analytical (or theoretical) methods, descriptive methods, and quasi-experimental methods? The value and power of experimental methods derive from the fact that they allow researchers to detect the laws (cause-and-effect relationships) of nature. On the contrary, descriptive research methods describe phenomena as they occur without aiming to manipulate or control the phenomena in order to establish cause-and-effect relationships (Davis, J.). Examples of descriptive methods are naturalistic observation and case study.

### 4.2.5 Experiment

<table>
<thead>
<tr>
<th>Class name:</th>
<th>&lt;experiment&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class ID:</td>
<td>CISP5</td>
</tr>
<tr>
<td>Cardinality:</td>
<td>multiple</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td>If the method of investigation is experimental, what types of experiments were executed? What were the experimental conditions, controls, protocols?</td>
</tr>
<tr>
<td>Definition:</td>
<td>A scientific experiment is a procedure which permits the investigation of cause-effect relations between known and unknown (target) variables of the domain. Experimental results cannot be known with certainty in advance.</td>
</tr>
<tr>
<td>Definition reference:</td>
<td>EXPO</td>
</tr>
<tr>
<td>Subclass:</td>
<td>&lt;computational experiment&gt;: &lt;computer simulation&gt;; &lt;physical experiment&gt;: &lt;Baconian experiment&gt;, &lt;Galilean experiment&gt;: &lt;hypothesis-driven experiment&gt;, &lt;hypothesis-forming experiment&gt;</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Location in paper:</td>
<td>paper section: methods</td>
</tr>
<tr>
<td>Representation:</td>
<td>natural language text; protocol</td>
</tr>
</tbody>
</table>

CISP supports the following subclasses of the class <experiment>:  
> <computational experiment>
Definition: A computational experiment is a scientific experiment which investigates cause-effect relations between known and unknown (target) variables by manipulating the computational (non-physical) domain adequate to the real-world domain.

Definition reference: EXPO

<computer simulation>
Definition: Computer simulation is a computational experiment where the real domain of study is modelled by a computer program imitating “the internal processes and not merely the results of the thing being simulated”.


Parent class: <computational experiment>

<physical experiment>
Definition: A physical experiment is a scientific experiment which investigates cause-effect relations between known and unknown variables by manipulating the real-world (physical) domain.

Definition reference: EXPO

<Baconian experiment>
Definition: A Baconian experiment is a scientific experiment which involves no explicit hypothesis.


Parent class: <physical experiment>

<Galilean experiment>
Definition: A Galilean experiment is a scientific experiment which involves explicit hypotheses.


Parent class: <physical experiment>

<hypothesis-driven experiment>
Definition: A hypothesis-driven experiment is a Galilean experiment designed to confirm or reject a given hypothesis. The deductive consequences of the hypothesis are compared with the experimental result, and the probability of the hypothesis is either increased (confirmed) or decreased (rejected).

Definition reference: EXPO

Parent class: <Galilean experiment>

<hypothesis-forming experiment>
Definition: A hypothesis-forming experiment is a Galilean experiment. It includes a hypotheses formation stage in which one or more hypotheses are formed using abduction or induction.
The class <experiment> can additionally have the following properties:

- **has Protocol**, where the class <protocol> is defined as:
  
  Definition: An experiment protocol is an explicit detailed specification of an experiment which describes a plan of experiment actions to achieve an experiment goal.

  Definition reference: EXPO
  Parent class: <procedure>

- **has Equipment**, where the class <equipment> is defined as:
  
  Definition: Experiment equipment is the set of tools, devices, materials, computer systems assembled for performing the experiment.


- **has Design**, where the class <design> is defined as:
  
  Definition: An experiment design is a structured, organized method for determining the relationship between factors affecting cause-effect relations between known and unknown variables.


- **has Experiment Factor**, where the class <experiment factor> is defined as:
  
  Definition: An experiment factor is a known variable of the model of the domain which the object of the experiment can control/vary in order to determine a value of target variables.

  Definition reference: EXPO

- **has Observation**, where the class <observation> is a key CISP class and defined in the section below.

- **has Result**, where the class <result> is a key CISP class and defined in the section below.

The formal description of experiments for efficient analysis, annotation, and sharing of results is a fundamental part of the practice of science. The above listed properties are important for providing information about the experiments executed within the investigation reported in a paper: what type of experiments were executed, what were the factors, how they were designed, what equipment was used, characteristics of the latter etc. CISP does not require mark-up of these properties, but it will add more semantics to the paper’s mark-up output. There are several ontology-based projects in bio-medical domains investigating how to record such information\(^\text{24,25}\), and metadata standards are appearing in many other sciences, e.g. in Physics\(^\text{26}\). Probably the best known attempt to formalise the description of experiments is that developed by the Microarray Gene

\(^\text{24} \text{mged.sourceforge.net/} \)
\(^\text{25} \text{psidev.sourceforge.net/} \)
The MGED Ontology (MO) was designed to formalise the descriptors required by MIAME (Minimum Information About a Microarray Experiment) standard for capturing core information about microarray experiments. MO aims to provide a conceptual structure for microarray experiment descriptions and annotation. A number of ontological developments related to MO also exist. The HUPO PSI General Proteomics Standards and Mass Spectrometry working groups are building an ontology that will support proteomic experiments. The MSI (Metabolomics Standards Initiative) ontology working group is seeking to facilitate the consistent annotation of metabolomics experiments by developing an ontology to help enable the scientific community to understand, interpret and integrate metabolomic experiments. More generally, the Functional Genomics Investigation Ontology (FuGO), now is known as OBI (an ontology of bio-medical investigations) is developing an integrated ontology that provides both a set of “universal” terms, i.e. terms applicable across functional genomics, and domain-specific extensions to terms.

CISP aims to provide semantic mark-up for investigations from various domains, while being consistent with already existing (or in progress) representations for specific domains, like OBI.

### 4.2.6 Observation

<table>
<thead>
<tr>
<th>Class name:</th>
<th>&lt;observation&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class ID:</td>
<td>CISP6</td>
</tr>
<tr>
<td>Cardinality:</td>
<td>multiple</td>
</tr>
</tbody>
</table>

**Informal Explanation:** What data/phenomena were recorded within an investigation? How are the data represented, in what format, where are they stored?

**Definition:**
An experimental observation is a direct observation of nature, the set of values of target variables (or other variables of the domain), "prior to analysis; interpretation*" (compare with results).

**Definition reference:**

**Example:**
Optical density readings

**Location in paper:**
paper section: results and discussion

**Representation:**
natural language text

**External location:**
data base

The class <observation> can provide an explicit link between the paper and the data. The semantic mark-up of the paper in future aims to contain all necessary information about

---

26 www.ph.ed.ac.uk/ukqcd/community/the_grid/QCDm11.1/ConfigDoc/ConfigDoc.html
27 msi-ontology.sourceforge.net/index.htm
28 fugo.sourceforge.net/
the data stored in data base or other public resource: location, access rights, version, supporting systems, etc.

### 4.2.7 The results

<table>
<thead>
<tr>
<th>Class name:</th>
<th>&lt;result&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class ID:</td>
<td>CISP7</td>
</tr>
<tr>
<td>Cardinality:</td>
<td>multiple</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td>What are the main outcomes of an investigation?</td>
</tr>
<tr>
<td>Definition:</td>
<td>results of the investigation are the set of facts, obtained through the interpretation of the observations.</td>
</tr>
<tr>
<td>Definition reference:</td>
<td>EXPO</td>
</tr>
<tr>
<td>Subclass:</td>
<td>&lt;experiment result&gt;</td>
</tr>
<tr>
<td>Example:</td>
<td>An average curve representing the growth of wild type yeast</td>
</tr>
<tr>
<td>Location in paper:</td>
<td>paper section: results and discussion</td>
</tr>
<tr>
<td>Representation:</td>
<td>natural language text</td>
</tr>
<tr>
<td>External location:</td>
<td>data base</td>
</tr>
</tbody>
</table>

The class <Results> (as does <Observation>) can provide an explicit link between the paper and the data. The semantic mark-up of the paper in future will contain all necessary information about the results stored in the data base or other public resource: location, access rights, version, supporting systems, etc.

The distinction between the classes <Result> and <Observation> is debatable. Many researchers consider them as synonyms. We would like to stress within the CISP formalism the difference between direct observations of some phenomena and processed or interpreted data. The status of such data is different from the point of view of the level of evidence they both provide towards reaching conclusions.

### 4.2.8 Conclusion

<table>
<thead>
<tr>
<th>Class name:</th>
<th>&lt;conclusion&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class ID:</td>
<td>CISP8</td>
</tr>
<tr>
<td>Cardinality:</td>
<td>multiple</td>
</tr>
<tr>
<td>Informal Explanation:</td>
<td>What new knowledge has been discovered? Has the goal of an investigation been achieved? Has a hypothesis been confirmed?</td>
</tr>
<tr>
<td>Definition:</td>
<td>A conclusion of an investigation is a statement inferred from observations, results, assumptions, and facts to support or reject a research hypothesis.</td>
</tr>
</tbody>
</table>
It is important to record under what assumptions, restrictions the conclusions were made; what facts and evidences are there to support such conclusions.

## 4.3 Candidate classes for inclusion into CISP

The following classes were considered for inclusion into the list of the key CISP classes:

- **<hypothesis>**
  - **Informal Explanation:** What is a hypothesis of an investigation, an experiment, or model? What was tested in the investigation?
  - **Definition:** A hypothesis is a statement about cause-effect relations between known and unknown (target) variables of the domain of the investigation "that shall be in accordance with known facts" to be verified by the experiment.
  - **Example:** A particular yeast strain will have a higher growth rate than a wild strain.

- **<background fact>**
  - **Informal Explanation:** A neutral or widely accepted statement about the knowledge domain.
  - **Example:** The addition of benzotriazole dyes to oligonucleotides either using a dye phosphoramidite, or using post oligonucleotide synthesis via suitable linker has been reported [#ref]. The post synthetic methods are easier to use in practice although lower yields are obtained.

- **<problem>**
  - **Informal Explanation:** The difficulties, restrictions when trying to achieve the goal of an investigation.
  - **Example:** DNA does not meet the requirements for SERRS due to the lack of a suitable visible chromophore.

- **<example>**
  - **Informal Explanation:** The examples given to demonstrate the authors'
findings or to explain an approach.

- **<model>**
  
  Informal Explanation: What theoretical model was used within an investigation? How is the model represented: as a system of equations, as logical rules and facts?
  
  Example: Logical model of yeast metabolism

- **<domain of investigation>**
  
  Informal Explanation: To what area of research an investigation belongs. To what area of knowledge the main results contribute.
  
  Example: Physical Chemistry
  
  Map to: <DC: Subject>

We would like to give some remarks about the class `<domain of investigation>`. This class is not directly used for the annotation of papers with CISP, and DC has a similar term `<DC: Subject>`. CISP application needs to ‘know’ what the domain of the annotated paper is, in order to download corresponding domain ontologies and data bases. It is not a trivial task to define what a domain of an investigation is. There can be several domains that are associated with an investigation. We suggest dividing them onto two main classes:

- **<main domain of investigation>**
  
  Informal Explanation: They are of research in which new knowledge is discovered
  
  Parent class: <domain>

- **<supplementary domain of investigation>**
  
  Informal Explanation: associated/ auxiliary domain
  
  Parent class: <domain>

The following classification system will be additionally used by our system:

- **<research councils UK classification>**
  
  Informal Explanation: The research councils UK classification is a classification of domains used by Britain's councils for academic research.

### 4.4 Verification of CISP

In order to define CISP, the Core Information about Scientific papers, we first chose a subset of the general scientific concepts (GSCs) described in EXPO. Our choice was based on the results of interviews with the experts and preliminary annotation of papers. Three experts were asked to annotate four papers using this set and at the end of this preliminary annotation the list of concepts was refined to include the following: **Goal of Investigation, Object of Investigation, Method of Investigation, Experiment, Observation, Hypothesis, Results, Conclusion, Motivation, Background, Problem, Example.**
The reasoning behind this choice was to determine a set of concepts that would describe the scientific investigation represented in the paper as integrally as possible, in terms of its objectives, the methodology of the approach, the outcomes and the pre-existing work which sets the scene for the current investigation.

Before proceeding on a large scale annotation of papers using the above concepts we wanted to assess whether the research community felt in agreement with our constructed set of terms and which of the GSCs were considered most informative and therefore indispensable.

We conducted an online survey (not anonymous) where each concept was presented as a candidate for inclusion in CISP, along with a short definition and an example of its use. The survey is available at:

http://www.aber.ac.uk/compsci/Research/bio/art/news/survey/

and everyone is invited to answer on our questions about CISP. We asked participants from a range of research disciplines from the UK and Japan to vote for the concepts they thought should be part of CISP. At the time of writing we have received feedback from 33 researchers.

We have ranked below the GSCs in descending order of popularity according to the survey participants. The five highest scoring concepts and the short description given on the website are:

1) Conclusion (the new knowledge that has been discovered or whether the goals of an investigation have been met);
2) Results (the actual outcomes of the investigation);
3) Goal of the investigation (what the investigation aims to show);
4) Method of the investigation (how the authors set out to pursue the goals of the investigation);
5) Object of the investigation (entity about which we seek to gain new knowledge).

The above are followed by <Experiment> (the types of experiments executed) and <Observation> (what data/phenomena were recorded within an investigation).

These were mostly the concepts from the proposed CISP. However, judging from the votes obtained, <Experiment> and <Observation> are not considered as significant in describing the methodological approach and outcomes of the investigation respectively.

In addition, Motivation which we considered to be highly relevant to an investigation, did not score as many votes. For example, some researchers saw it as being hard to distinguish from <Goal of an investigation>. Two thirds of our participants thought we should also include the hypothesis of an investigation in CISP but in general they hold concepts pertaining to pre-existing work and the state of the art <Problem>, <Background> to be less crucial. Finally, the theoretical <Model> employed and any examples used to illustrate the approach are considered by most to be too detailed aspects of the methodology.

A comment made by several of our participants was that our analysis may not be suited to all kinds of scientific papers, for instance review papers or papers that simply showcase systems may lack most of the concepts in CISP. We are taking this on board and are planning to focus on papers that are original contributions with a determinable investigation and results. Also, many people added that not all science has an explicit hypothesis either. However, even if the hypothesis of an investigation is not stated as such
in a scientific paper, it can almost always be inferred, which is what we plan to do in our analysis. Another participant mentioned how they often find there is an overlap between \(<\text{Goal}\>\), \(<\text{Object}\>\) and \(<\text{Method}\>\). It is often the case that one concept may subsume the other, e.g. sometimes developing a method can also be the object of an investigation but the details of the method are interesting in their own right. Nevertheless, we will make sure the distinction between the latter three concepts is as clear as possible to our annotators.

Overall we have found this survey to be very useful as it has helped us obtain a third party view on CISP and the importance of GSCs. Many of the comments have been invaluable and we will definitely take them into account. The survey results are summarized in the table (see the Appendix).
5. Conclusion

We have proposed the CISP metadata for the description of papers about scientific investigations. The principle difference between CISP and other metadata schemas is that the former aims to represent not just what is typically reported in scientific papers, but what should be reported to convey a complete scientific investigation. We restricted our set of investigations to ones where the research is driven by experimental methods. Our understanding of an experimental method is broad: physically executing experiments, computationally running experiments, or theoretical experiments.

CISP is a defined set of leaf classes from DOLCE and EXPO. CISP has eight key classes: <Goal of investigation>, <Motivation>, <Object of investigation>, <Research method>, <Experiment>, <Result>, <Observation>, and <Conclusion> and we have provided the detailed description of each CISP class.

CISP has sound theoretical foundations. The methodology used to construct CISP, ensures logic coherence and compliance to other formalisms. As so far, CISP has been validated by experts in physical chemistry and by researchers from different fields who participated in an online survey. Within the ART project we plan to apply CISP to a set of 200 papers to verify our approach.

We argue that ontologies are important for the development of metadata. Ontology engineering provides semantic clarity, explicitness, and facilitates the reusability of represented information and knowledge. The use of formalized semantic representation can also facilitate natural language processing for intelligent information analysis and retrieval. Therefore ontology based knowledge representation opens new perspectives for text mining techniques and logic inference.

Ontologies with their explicit definitions and clear structure are also a valuable resource for educational applications.
About the authors

**Dr Larisa Soldatova** is a RCUK Fellow at the University of Wales, Aberystwyth, Department of Computer Science. Over the past ten years, she has worked on ontology development and knowledge representation in Russia, Japan and the UK. The focus of her research is the development of formal methods for knowledge representation and their applications to the natural and social sciences. She developed EXPO, a general ontology of scientific experiments. So far she has applied EXPO to biology, physics, computer science, data base design, and has developed an ontology of tests (exams).

Larisa has a PhD in computer science from the Far Eastern Technical University (Russia) on the topic of automated generation of problems.

She can be contacted at: [lss@aber.ac.uk](mailto:lss@aber.ac.uk).

Dr Maria Liakata has been a research associate with the Computational Biology group at the University of Wales, Aberystwyth since 2005. She is a computer scientist with a mathematical and Natural Language Processing (NLP) background. Maria's research interests include Computational Semantics, machine learning for knowledge discovery and the use of ontologies in NLP applications. She has also worked on data mining from *Arabidopsis* mass spectrometry data and on the computational modelling of yeast growth curves.

Maria has a DPhil in Computational Linguistics from the University of Oxford on the topic of Inducing Domain Theories.

She can be contacted at: [mal@aber.ac.uk](mailto:mal@aber.ac.uk).
Appendix: Survey Results

*Table: The summary of the survey results conducting for the ranking of CISP terms.*

<table>
<thead>
<tr>
<th></th>
<th>Conclusion</th>
<th>Counts: 33</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Results</td>
<td>Counts: 32</td>
</tr>
<tr>
<td>3</td>
<td>Goal of Investigation</td>
<td>Counts: 30</td>
</tr>
<tr>
<td>4</td>
<td>Method of Investigation</td>
<td>Counts: 29</td>
</tr>
<tr>
<td>5</td>
<td>Object of Investigation</td>
<td>Counts: 28</td>
</tr>
<tr>
<td>6</td>
<td>Experiment</td>
<td>Counts: 26</td>
</tr>
<tr>
<td>7</td>
<td>Observation</td>
<td>Counts: 22</td>
</tr>
<tr>
<td>8</td>
<td>Hypothesis</td>
<td>Counts: 21</td>
</tr>
<tr>
<td>9</td>
<td>Model</td>
<td>Counts: 12</td>
</tr>
<tr>
<td>10</td>
<td>Example</td>
<td>Counts: 7</td>
</tr>
</tbody>
</table>
References


Schulze-Kremer, S., 2001, Ontologies for Molecular Biology. Computer and Information Sci. 6(21)


