

# Domain Ontology of Scientific Publications

In this document we describe domain ontology for scientific publications which was designed and created according to characteristics of scientific publication domain.

Domain ontology in project MAPEKUS<sup>1</sup> imports two standalone ontologies:

- Party ontology – specifies various types of organizations and individuals. It is used to describe authors, editors and publishers.
- Region ontology – is used for geographical area and country representation. It is used to describe origin of a publication or location of an event.

## 1.1 Classes and data properties

This part describes all ontology classes and their data properties. We opted for this approach because data properties are always bound to one class and are important part of its description. Because all data properties are functional there is only one attribute for each instance.

### 1.1.1 Publication Class and its subclasses

Publication class is abstract, fully defined with its subclasses and all types of publications extend it. It has following properties:

- **Title** – property of type `xsd:string` represents publication title;
- **Year** – property of type `xsd:int` represents year of appearance;
- **Month** – property of type `xsd:int` represents month of appearance;
- **Day** – property of type `xsd:int` represents day of appearance;
- **FirstPage** – property of type `xsd:int` represents first page number of publication;
- **LastPage** – property of type `xsd:int` represents last page number of publication;
- **Abstract** – property of type `xsd:string` represents publication abstract;

---

<sup>1</sup>MAPEKUS project, <http://mpekus.fiit.stuba.sk>

- **Source** – property of type `xsd:string` represents publication source e.g. doi address;
- **Web** – property of type `xsd:string` represents source address;

Figure 1.1 depicts hierarchy of classes derived from Publication class which is in other words hierarchy of all classes in ontology.

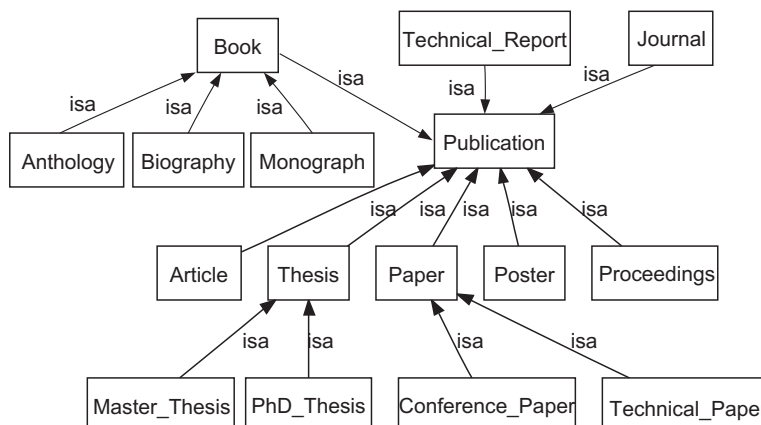


Figure 1.1: Publication Hierarchy

#### Individual subclasses of Publication Class

**Article.** Represents an article for example in magazine

**Article.** Represents an article for example in magazine

**Book.** Represents a book, it has three subclasses:

- **Anthology** – Collection of work from various authors ;
- **Biography** – Curriculum vitae;
- **Monography** – Scientific work dedicated to single domain/

It has following specific properties:

- **ISBN** – property of type `xsd:string` represents international standard book number;

**Journal.** Represents periodical dedicated to scientific community. It has following specific properties:

- **Number, volume** – property of type `xsd:string` represents identification number of periodic;

**Paper.** Represents scientific text published in journal or proceedings. It further specializes to:

- Conference paper – paper presented at conference;
- Technical paper – describes technical aspects of particular system.

**Proceedings.** Collection of scientific papers published in context of conference or other meeting of scientific community. Usually it has form of a book.

**Poster.** Poster presenting scientific content.

**Technical report.** Formal report describing contribution in domain of applied research, it contains details and results regarding solution of some scientific problem.

**Thesis.** Work describing scientific results created by candidate to master of PhD title. It has two subclasses:

- MasterThesis – final work of second level of education;
- PhDThesis – final work of third level of education.

### 1.1.2 Event Class and its subclasses

This part of ontology is used to describe an event involving scientific community. Hierarchy of event classes is depicted on figure ??.

Event class has following properties:

- **StartDate** – property of type `xsd:date` represents a starting date of event;
- **EndDate** – property of type `xsd:date` represents a finishing date of event;
- **Web** – property of type `xsd:string` represents a link to event description.

Subclasses of Event Class:

- **Activity**
- **Conference**
- **Meeting** – other scientific meeting;
- **Workshop** – event including learning and discussion about some scientific topic.

### 1.1.3 IndexTerms class hierarchy

Class `IndexTerm` and its subclasses represent division of fields of information science into hierarchical structure. This classification is based on computing classification system (CCS) used by ACM digital library. Figure 1.2 shows tree structure of individual subclasses.

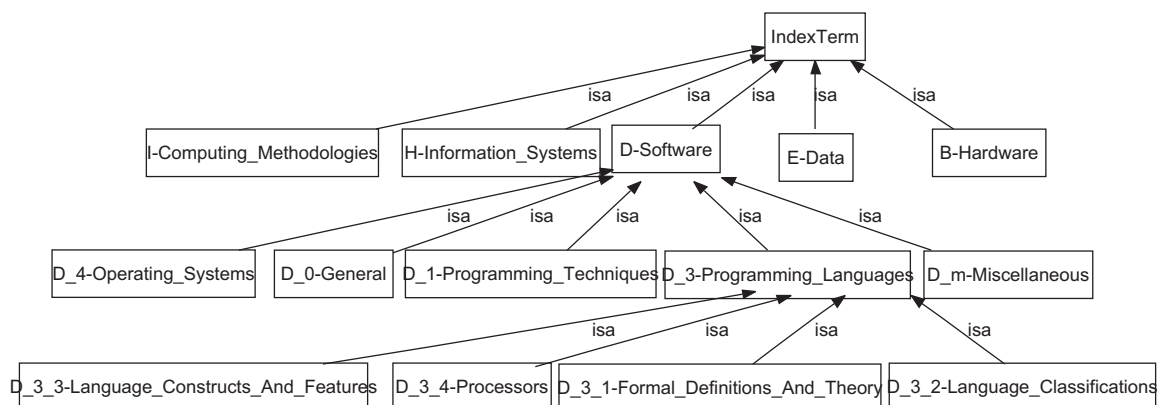


Figure 1.2: Part of hierarchy of IndexTerms class

### 1.1.4 Other classes

**Project.** Class representing a project. Name of the project is stored in the name of its instance.

**Author.** Class representing an author of at least one publication. It is derived from class `Person` (from imported *Party* ontology) which is a general representation of person. It contains the following inherited attributes:

- `GivenName` – property of type `xsd:string` represents a given name;
- `FamilyName` – property of type `xsd:string` represents a family name;

**Editor.** Class representing an editor of at least one publication. Similarly to the `Author` class, it is also derived from the imported `Person` class and contains the same inherited attributes.

**Publisher.** Represents a publisher of at least one publication. It is derived from class `Organization` imported from ontology *Party*.

## 1.2 Class Relations

This section describes relations between individual ontology classes. Each has listed a set of valid domains and ranges and also type for nonstandard relations (functional, symmetric or transitive) and name of corresponding inverse relation if such exists. Most significant classes and relations between them are depicted in figure 1.3.

**IsAuthorOf.** Relation which represents relationship between author and corresponding authored publication.

- Domain: `Person`

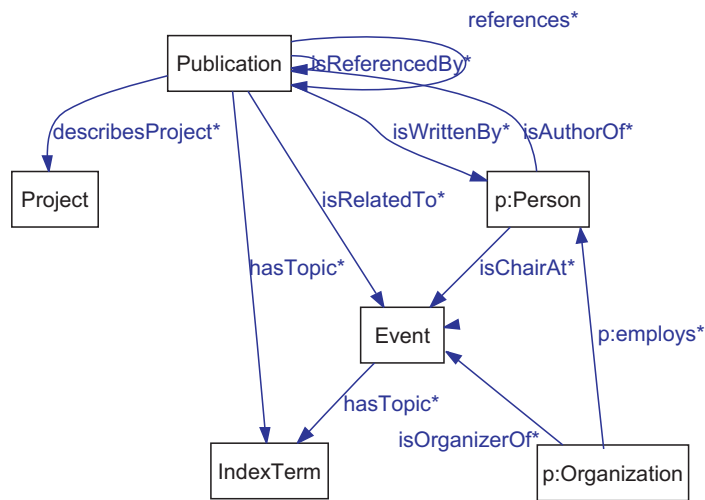


Figure 1.3: Most significant relations between classes

- Range: Publication
- Inverse relation: IsWrittenBy

**HasTopic.** Relation between publication or event and its scientific domain. Publication or event can refer to one or more topics.

- Domain: Publication, Event
- Range: IndexTerm
- Inverse relation: IsTopicOf

**IsReferencedBy.** Relation between publications of which one references the other.

- Domain: Publication
- Range: Publication
- Inverse relation: References

**IsEditorOf.** Relation between journal and person who is its editor.

- Domain: Person
- Range: Journal
- Inverse relation: HasEditor

**IsPublishedBy.** Relation between publication and its publishing organization.

- Domain: Journal, Proceedings, Books
- Range: Organization
- Inverse relation: IsPublisherOf

**IsRelatedTo.** Relation between publication and related event, e.g. publication published as part of conference proceedings.

- Domain: Person
- Range: Event
- Inverse relation: -

**IsPartOf.** Relation which represents that publications which are not published individually are contained in some other publication which is published individually.

- Domain: Article, Paper, Poster, Thesis
- Range: Journal, Proceedings, Book
- Inverse relation: Contains

**SimilarPublication.** Relation representing similarity of publications.

- Domain: Publication
- Range: Publication
- Inverse relation: SimilarPublication
- Type: Symmetric

**IsCollaboratorOf.** Relation representing collaboration of two authors on some publication.

- Domain: Author
- Range: Author
- Inverse relation: IsCollaboratorOf
- Type: Symmetric

**IsChairAt.** Relation between conference and chairman (of program committee).

- Domain: Person
- Range: Conference
- Inverse relation: -

**DescribesProject.** Relation between publication and project which it describes.

- Domain: Publication
- Range: Project
- Inverse relation: IsDescribedBy

**IsOrganizerOf.** Relation between event and its organizer.

- Domain: Organization (from Party ontology)
- Range: Event
- Inverse relation: IsOrganizedBy

**TakesPlaceAt.** Relation representing where the given event took place.

- Domain: Event
- Range: Address (from Region ontology)
- Inverse relation: -