PhD Dissertation

International Doctorate School in Information and Communication Technologies

DISI - University of Trento

MANAGING UBIQUITOUS SCIENTIFIC KNOWLEDGE OBJECTS

Hao Xu

Advisor:

Prof. Fausto Giunchiglia

Università degli Studi di Trento

November 2011

Abstract

Scientific discourse, as the basic unit of dissemination and exploitation of research results, has steadily enhanced their accessibility and reusability in response to the advancement of web technologies. A highly semantic enriched publication always makes its information and data much easier to search, navigate, disseminate and reuse, whereas most online articles today are still electronic facsimiles of linear structured papers, with shallow metadata descriptions, lacking in semantic knowledge and interlinked relationships between elementary modules of content.

In this dissertation, we propose a Scientific Knowledge Objects (SKO) framework in terms of a theory of structural knowledge- SKO Types, a methodology for scientific discourse representation- SKO Patterns, a tool for semantic authoring and annotation- SKO TeX, and an application of SKO management- the Conference of the Future, in the context of the emerging Social Web and Semantic Web.

Keywords

Knowledge Management, Scientific Publishing, Digital Library, Metadata Schema, Pattern, Discourse Representation, Semantic Annotation, Semantic Authoring, Social Network Services

Contributions and Publications

This work has been *gasified*, *liquefied* and *solidified* in collaboration with many other people as the publications indicate and in particular with Prof. Fausto Giunchiglia, Ronald Chenu, Denys Babenko and Aliaksandr Birukou.

This dissertation makes the following contributions:

- A detailed survey on the state of the art of metadata schemas, discourse representation models and publishing platforms in the scientific domain.
- Proposing an entity-oriented theory, namely *SKO Types*, for representing and linking *Scientific Knowledge Objects* by defining entities, relationships between entities, and the attributes of each entity in the scientific domain.
- Design and development of a general discourse representation model, namely *SKO Patterns*, especially for knowledge management in the emerging Social Web and Semantic Web.
- Design and development of a semantic editing tool - *SKO TeX* - along with sets of implemented macros and processors for IJCAI (International Joint Conference on Artificial Intelligence) project.
- Launching the *Conference of the Future Initiative*, along with its high-level prototype and interface implementations.

Part of the material of this thesis has been published as journal papers, conference papers and technical reports (in order of appearance):

[30] **Hao Xu**. Managing Ubiquitous Scientific Knowledge on Semantic Web. Lecture Notes in Computer Science, Volume 6059: 421-430, Springer, AST/UCMA/ISA/ACN 2010, Miyazaki, Japan, 2010

[80] Giuseppe Veltri, Matteo Turrini, Luc Schneider, Jordi Sabater-mir, Diego Ponte, Nardine Osman, Matus Medo, Ronald Chenu, Aliaksandr Birukou, Marcos Baez, Joseph Rushton Wakeling, **Hao Xu**. State of the Art in Scientific Knowledge Creation, Dissemination, Evaluation and Maintenance. E-Print, University of Trento, Italy, Technical Report # DISI-09-067, 2009 http://eprints.biblio.unitn.it/archive/00001773/01/LiquidPub_D1-1_TechRep.pdf

[82] Fausto Giunchiglia, Ronald Chenu, **Hao Xu**, Aliaksandr Birukou, Enzo Maltese. Design of the SKO Structural Model and Evolution Version 2. LiquidPub Project Deliverable 1.2, 2010 https://dev.liquidpub.org/svn/liquidpub/papers/deliverables/LP_D1.2v2.pdf

[84] Fausto Giunchiglia, Ronald Chenu-Abente, **Hao Xu**, Uladzimir Kharkevich. A Metadata-Enabled Scientific Discourse Platform. E-Print, University of Trento, Italy, Technical Report # DISI-10-069, 2010 http://eprints.biblio.unitn.it/archive/00001939/01/069.pdf

[85] Fausto Giunchiglia, **Hao Xu**, Aliaksandr Birukou, Ronald Chenu. Scientific Knowledge Object Patterns. Proceedings of 15th European Conference on Pattern Languages of Program, EuroPLoP 2010, Irsee Monastery, Bavaria, Germany, 2010

[94] **Hao Xu**. A Semantic Pattern for Scientific Discourse Representation. Journal of Computational Information Systems. Volume 6(13): 4223-4228, 2010

[103] Fausto Giunchiglia, **Hao Xu**. Conference of the Future. Proceedings of 16th European Conference on Pattern Languages of Programs, EuroPLoP 2011, Irsee Monastery, Bavaria, Germany, 2011

Acknowledgement

In an instant, the 22 years of my student life is coming to the end. It was the beginning with me dragging luggage through the Trento train station four years ago that remains still fresh in my memory. Now, however, I am feeling as excited and proud as the first time I walked into this beautiful campus with longing and expectation.

I would like to express the greatest gratitude to my supervisor Professor Fausto Giunchiglia. I can still remember all my tension the first time I met this internationally renowned scientist, in an unfamiliar environment, facing brand-new technologies. I will never forget those countless weekends, numerous early mornings, many nights, in which we continued the discussion about projects and papers. Your dedication with regard to research not only helped me learn the true meaning of science, but also brought me into the temple of science. Thank you for showing me how to be patient whenever I showed fickleness, and for teaching me persistence whenever I faced a setback. By your words and deeds, and careful guidance, I felt the great and proud feeling of being "on the shoulders of giants”.

Special thank goes to Professor Liang Yanchun, who always showed his fatherly care and edification. Your rigorous scholarship, and your kindness, is an example for my life. I will always be grateful to my teachers who give their endless support to my life and academic growth – Professor Zhou Chunguang, Professor Zhang Changhai, Professor Fabio

Casati, Professor Maurizio Marchese, Professor Marco Ronchetti, Professor Niculae Sebe, Professor Enrico Blanzieri, Professor Paris Avgeriou, Professor Jin Ying, Professor Zhang Jingying and Doctor Yao Yuchun, I wish to show my sincere gratitude to you all.

My appreciation also goes to all my colleagues and friends ever stood by my side in Trento. Accompanied by you, I became stronger and no longer lonely in a foreign country. Zhou You, Sun Caitang, Liu Xin, Li Kainuo, Wang Kangping, Wang Zhe, Zhang Rui, Gao Wenjuan, Zhou Xu, Wu Chunguo, Huang Lan, Sun Shuyang, Yang Bin, Song Jia, Ma Da, Liu Fangfang, Chen Xin, Gao Wenzheng, Jin Ran, Zhang Chen, Du Wei, Peng Tao, Huang Xiaojuan, Jiang Ye, Chang Jian, Zhang Yan, Ouyang Ruochuan, Dong Sa, Ai Yang, Huang Haifeng, Yu Yang, Liu Huaxiao, Ren Yanjiao, Meng Xiangyu, Niu Bingshuai, Sun Youcheng, Zhang hanyu, Liu Qinan, Xu Guoshan, Yue Yanchao, Jia Mofei, Fang Enzhu, Ronald Chenu， Aliaksandr Birukou, to all of you, I would like to show my sincere thanks.

I owe the deepest gratitude to my dearest parents and beloved. Thank you for your eternal love for me. My great appreciation to you! I believe that I eventually learned how to cherish your love by being apart and through missing you.

Thank you all, the ones who love me and whom I love.

Last but not least, I am grateful to the China Scholarship Council, the *East Web* European Project, the *Bridging The Gap* European Project and the *Liquid Publications* European Project, that supported and assisted my work in this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Context

Scientific publishing is currently undergoing significant paradigm shifts, as it makes the transition from print to electronic format [1], from subscribers only to open access [2,3] and from static information to a dynamic (collaborative) knowledge space [4,5]. Although the processes of scientific publishing, including submission format, review and distribution, vary greatly from journal to journal, conference to conference, publisher to publisher and field to field, we believe that the development of information and communication technology becomes one of the most underlying drivers which is leading its trends and revolutions.

During the past five decades, the theory of metadata [6,7] has been developed in a variety of directions, such as the cataloging of archived literature in libraries [8,9,10,11]. However, metadata in digital libraries is traditionally focused on a description created by librarians or web designers, which can be shallow and non-collaborative [12]. The advent of Web 2.0 [13,14] has had a significant impact on scientific knowledge discovery and dissemination, especially on information retrieval [15,16], knowledge sharing [17,18], web mining [19,20] etc. More importantly, it allows users to participate in the content management. Users are now becoming contributors of metadata, e.g. tagging and annotating, instead of only being consumers. Of course, an ideal way of gaining metadata is to generate it automatically by computers in the form of Semantic Web [21], in which describing things can be understood by machines and ubiquitous data can be linked together [22].

A major concern in the scientific publication research community today is the continued improvement of semantics during the entire lifecycle of scientific artifacts [23], i.e. creation, dissemination, evaluation, publication and reuse. The concept of externalization [24] has been investigated intensively in recent years. Externalization represents the process of articulating tacit knowledge into explicit concepts which was proposed by Nonaka [25]. Cognitive externalization makes scientific publications much easier to disseminate, navigate, understand and reuse in research communities. In the last decade, a handful of models [24] targeting the externalization of the rhetoric and argumentation captured within the discourse of scientific publications were proposed based on Cognitive Coherence Relations [26] or the Rhetorical Structure Theory [27].

Moreover, computer science and web technology are also revolutionizing the scientific publishing systems where diverse scientific knowledge is produced and disseminated. Such publishing platforms not only provide tools for strategic reading or annotating [28], but also establish community based environments for social networking and open science [29].

1.2 The Problem

To date, prestigious publishers always provide a highly recognizable format and form of presentation for their published papers. However, most of them haven't changed much over several decades. In the mid-1990s, the advent of the Internet offered amazing opportunities for scientific journals. Online publishing thoroughly revolutionized searchability and information discovery, tremendously increased the breadth and ease of access, and gradually allowed for the dissemination of supplementary materials such as large data sets, comments and some related citation links online, which could not be obtained in traditional printed publications. However, few have tackled the problem of how best to bring the magic of the new ICT technologies, especially of Web 2.0 and Semantic Web technologies, to bear on the structure, representation, organization and presentation of the article itself. Thus, for most publishers, the online publication of today remains an electronic copy of the traditional print paper. Cell[1] has made a successful attempt to promote the direction of “the Article of Future”, but it is restrained respectively by its narrow discipline and types of literature, which is difficult to apply to all kinds of scientific publications, certainly for more general potential readers.

The initial motivating example comes from the narrative of writing a PhD qualifying paper. To start with, the student uses Google Scholar, Citeseer and DBLP to accumulate his background knowledge to arrive at the state of the art in his field and to generate a tentative *gas* idea. Subsequently, he discusses it with his supervisor and colleagues face to face or via email. Meanwhile, he attends interesting seminars, courses, related workshops and conferences, and begins to draft his *liquid* paper.

[1] A scientific journal: http://www.cell.com

After several iterations, he finishes organizing and writing the qualifying paper using LaTex, and then sends the *solid* PDF[2] file to the committee. He gets feedbacks from the reviewers and checks the review forms, item by item, in terms of his paper in order to make final modifications.

Although some technical progress has been made in such a scenario, at least several obstacles must be overcome before a semantic framework can be realized. Firstly, how to write a PhD qualifying paper. Essentially, what the structure of a qualifying paper should be, and how to prepare both background knowledge and writing skills for it, are practical questions for every doctoral student. Although some experienced students have achieved a degree of expertise from previous courses or practices, an empirical pattern is generally appreciated. Secondly, the state-of-the-art tools are not efficient enough for collaborative work in this use case. Since the qualifying paper itself evolves and changes during its lifecycle in a distributed production environment, several versions are generated, and various comments and reviews are mixed. A supervisor could give some general comments by email, while commenters and reviewers might suggest several detailed critiques or referenced materials with un-unified formats of files. There is still no standard schema and container to describe, comment on, and review SKOs in order to facilitate collaboration, version management and metadata sharing. Thirdly, when the student hunts for background knowledge about his research topic, it frequently happens that he wants to check particularly interesting references for further in-depth reading directly, such as the result of an evaluation experiment, a definition of a novel concept, an impressive figure, etc. To date, scientific publications are always applied as basic indivisible units such as a PDF document, which needs a specific modularity for the SKO's rhetorical structure and

[2] PDF: portable document format

interlinked knowledge representation. Fourthly, when the student finds some interesting related works, e.g. a reference, a relevant project, or even a researcher mentioned in a paper, he has to input their titles or names to the search engines in order to begin a time-consuming navigation. Using such an approach, months of work might only result in a 10-page paper, which will dramatically benefit others in the event of sharing. Instead of such a paper disappearing from view, marking them up as entities and annotating them with Uniform Resource Identifiers (URI)[3], along with sets of attributes, could definitely facilitate the efficiency for SKO search and navigation. Enriching papers semantically is still a difficult problem, yet to be adequately resolved. Papers always lack semantics both during authoring and during the post-publication period. To help readers attain a rhetorical block which describes background, contribution or discussion easily and intuitively, is another research issue that has yet to be tackled [30].

Within the scope of this thesis, the four prime issues that we focus on can be summarized:

1. Current scientific metadata schemas focus on describing data, but not entities. They are descriptive, but few of them are structural and administrative. They provide a rare mechanism for linking entities and describing relationships between them.

2. Modularity patterns for semantically modelling different kinds of SKOs are needed, for both reading and writing purposes.

3. Existing editing tools for SKOs such as LaTex and Microsoft Office are not fit for semantic authoring and annotating.

4. Current review models have been heavily criticized in various

[3] Uniform Resource Identifiers (URI): http://www.w3.org/Addressing/URL/URI_Overview.html

scientific communities in terms of, for example, superficial reviews, a lack of social connectedness, comments and discussions about papers which can hardly be kept track of, etc.

Developments to tackle the above difficulties are real challenges faced by researchers.

1.3 The Solution

In this dissertation, we propose a Scientific Knowledge Objects (SKO) framework in terms of a theory, a methodology, a tool, and an application for SKO management, in the context of the emerging Social Web and Semantic Web. The main contribution of this research can be summarized as follows:

1. SKO Types: A Theory - From Linked Data to Linked Entity

SKO Types specifies sets of bibliographically related entities, relationships, attributes and services, intended to describe ubiquitous scientific knowledge objects semantically, and to facilitate their dissemination, collaboration, evolution and reuse. It comprises six categories of attributes. The *general category* groups the general information that describes the SKO as a whole. The *lifecycle category* groups the characteristics associated to the history and current status of this SKO, and those who have affected this SKO during its evolution. The *relational category* groups features that define the relationship between the SKO and other entities. The *technical category* groups the technical requirements and technical characteristics of the SKO. The *rights category* groups the intellectual property rights, authorship, copyrights and conditions of use for the SKO. Finally, the *meta-metadata category* groups data of the metadata instance itself, rather than the SKO that the metadata instance describes.

2. SKO Patterns: A Methodology - From Linear Structure to Rhetorical Structure

We propose the Scientific Knowledge Object Patterns (SKO Patterns) in terms of a general discourse representation model, especially for the purpose of knowledge management in the emerging Social Web and Semantic Web. Such model not only draw on the essence of the existing rhetorical structured models, but also extend the capabilities of semantic annotation, semantic search, and strategic authoring, grounded on logical reasoning (i.e. deduction, induction, and abduction). We modularize a scientific paper by the logical functions of the information, and reorganize it by rhetorical structure as our pattern solution for discourse representation. Above all, we divide a discourse into Metadata and Data parts. Herein, the Metadata consists of bibliographic information, abstracts, reference sets, annotations, etc., while the Data is the main body of the paper that is constructed using the general scientific method.

3. SKO TeX: A Tool - From Syntax Tagging to Semantic Annotation

We provide a tool, namely SKO TeX, for authoring and annotating semantic documents. SKO TeX is a LaTex-like editing environment, and supports the creation of both content data and related metadata for scientific publications. PDF format is an ideal container for SKO semantics, since it can be considered as the de facto standard in terms of electronic publishing. The vision of SKO TeX aims at SKOs' creation, distribution, collaboration and evaluation. This will be enabled by the use of SKO Types and SKO Patterns. We would strongly argue that the best way to present a narrative to a computer is to let the author explicitly create a rich semantic structure for the SKO during writing. SKO TeX provides a viable way for authoring and annotating semantic documents using SKO Patterns. With SKO TeX, readers can quickly glance through the contribution and skip to the section they are interested in. The writing at the syntax level in SKO TeX will be compatible with regular LaTex commands. In addition, the specific annotation commands are proposed as a mark-up language. All these commands provide the support for creating rhetoric elements, creating implicit and explicit visual annotations and for inserting arbitrary annotations in SKOs. In fact, semantic annotation creates a bridge between the actual SKO and its metadata.

4. Conference of the Future: An application - From Open Access to Open Science

The "Conference of the Future" Initiative aims to establish a new way to submit, evaluate, revise, publish, comment on and reuse, in future papers, the contents of the papers published in a conference. Such conferences enable researchers to communicate much more interactively, while the live presentation is only one stage of the interaction, even if the most important, in terms of what happens before and after the conference. The referee feedback is provided as part of the reviewing process. For those papers which are initially accepted, the reviewing, shepherding, commenting on and revision process keeps going until after the conference, when the paper is finalized. Even after publication, the papers can be commented upon and become the topic of online discussion leading eventually to the submission of new papers.

1.4 Structure of the Thesis

The remainder of this thesis is organized as follows:

Chapter 2 investigates and analyses the state-of-the-art of metadata schemas, discourse representation models, and publishing platforms in the scientific domain.

Chapter 3 proposes an entity-oriented theory, namely SKO Types, for representing and linking Scientific Knowledge Objects by defining entities, relationships between entities, and the attributes of each entity in the scientific domain.

Chapter 4 describes SKO Patterns in terms of a general discourse representation model, especially for knowledge management in the emerging social and semantic webs.

Chapter 5 presents a semantic editing tool - SKO TeX - along with sets of implemented macros and processors for IJCAI[4].

Chapter 6 launches the Conference of the Future Initiative, along with its high-level prototype and interface implementations.

Chapter 7 concludes the consideration of SKO theory and its applications, and points out our future trajectory.

[4] IJCAI- International Joint Conference on Artificial Intelligence: http://ijcai.org/

Chapter 2

State of the Art

In this chapter, we first introduce the metadata schemas applied in scientific publishing with the context of Web 2.0 and Semantic Web. We summarize the predominant existing metadata schemas and illustrate how metadata facilitates the evolution of scientific publishing, along with well-known applications, being enriched with features of semantic technologies. Following this, we investigate a handful of models targeting the externalization of the rhetoric and argumentation captured within the discourse of scientific publications. We will then discuss several tremendously promising online publishing systems and projects as intuitive case studies.

2.1 Scientific Metadata Schemas

Metadata is generally defined as "data about data" or "information about data"[7], which is used to facilitate resource discovery, e-resources organization, interoperability, digital identification, archiving and preservation. There are three main types of metadata, i.e. descriptive metadata, structural metadata, and administrative metadata [31].

During the past fifty years, many metadata schemas have been developed in a variety of disciplines. Standards for metadata in digital libraries include Dublin Core[5], EAD (Encoded Archival Description) [32], MARC (Machine Readable Catalogue) bibliographic records [33], METS (Metadata Encoding and Transmission Standard)[6] [34], PREMIS (PREservation Metadata: Implementation Strategies) schema[7] [35], OAI-PMH (Open Archives Initiative - Protocol for Metadata Harvesting)[8] [36], CIDOC-CRM (The CIDOC conceptual reference module) [37], FRBR (Functional Requirements for Bibliographic Records) [38], etc. Moreover, FOAF (Friend of a Friend)[9] defines an open, decentralized technology and metadata schema for connecting social web sites, and the people they describe. LOM (Learning Object Metadata) [39] focuses on learning objects, digital or non-digital, and their management, location and evaluation. In addition to this, major search engines, such as Google[10], Yahoo[11] and Bing[12], also provide their own metadata schemas for archiving and searching. Those aforementioned standards constitute the metadata foundation for scientific publication management.

Meanwhile, metadata promotes the evolution of semantic technologies, e.g. ontology, mark-up language, semantic search, semantic matching and so forth. Ontology is a formal representation of a set of concepts. It focuses on a specific domain and the relationships between concepts within it, which is applied to reason about the metadata of that domain or to define the domain [40]. In the field of scientific publications, a set of bibliographic ontologies have been proposed to support information retrieval and text mining, e.g.

[5] http://dublincore.org/
[6] http://www.loc.gov/standards/mets/
[7] http://www.oclc.org/research/projects/pmwg/
[8] http://www.openarchives.org/OAI/openarchivesprotocol.html
[9] http://www.foaf-project.org/
[10] http://www.google.com/
[11] http://www.yahoo.com/
[12] http://www.bing.com/

Bibliographic Ontology [41], FaBiO (the FRBR-aligned Bibliographic Ontology) [42], MarcOnt[13] [43], etc. A mark-up language is an artificial language comprising metadata, markup and data content [44]. It is used to describe the information in relation to the structure of text or its display, which has already been popularly used in annotating a text, such as HTML (Hypertext Markup Language) [45], XML (Extensible Markup Language) [46], RDF (Resource Description Framework)[14] [47] and OWL (Web Ontology Language)[15][48], etc. Additionally, semantic matching [49,50,51,52,53,54] and semantic searches [55,56,57] have improved the search process by leveraging XML, RDF and OWL data to produce highly relevant results. The essential difference between a semantic search and a traditional search is that a semantic search is based on semantics, while a traditional search is mainly resulted by keywords mapping.

Recently, applications of scientific publication search engines have proliferated, examples include Google Scholar[16], Citeseer[17], DBLP[18] and so on. With the advent of semantic browsers [58,59,60] , semantic wiki [61,62,63] and semantic digital libraries [64,65], users may enjoy more conveniences brought by semantic web and social network services.

In this section, we delve into five state-of-the-art metadata schemas that are widely used in scientific publishing areas and most related to our research, i.e. Dublin Core, LOM, BiBTeX, Schema.org and Google Scholar.

---

[13] MarcOnt Specification: http://semdl.info/books/2/appendices/G
[14] http://www.w3.org/RDF/
[15] http://www.w3.org/2004/OWL/
[16] http://scholar.google.com
[17] http://citeseer.ist.psu.edu/
[18] http://www.informatik.uni-trier.de

2.1.1 Dublin Core

In March 1995, the Online Computer Library Center (OCLC)[19], located in Dublin, State Ohio, United States, proposed a metadata element set to describe online information, which, in fact, means to be able to describe all objects on the web. This metadata is named Dublin Core (DC). After 10 years of development, Dublin Core has been popularized as a metadata standard by Dublin Core Metadata Initiative (DCMI)[20] and has widely been adopted around the world. Furthermore, it is the most widely used metadata standard in libraries, museums, governmental agencies and commercial organizations.

DCMI specification [66] provides a one-stop source of up-to-date definitions on metadata terms, including the classic Dublin Core Metadata Element Set [67] and the DCMI Type Vocabulary and Resource Classes [68] used as formal domains and ranges. The Dublin Core Metadata Element Set contains fifteen elements which are broad and generic in order to describe a wide range of resources as follows:

Term Name	Definition
Title	A name given to the resource.
Creator	An entity primarily responsible for making the resource.
Subject	The topic of the resource.
Description	An account of the resource.
Publisher	An entity responsible for making the resource available.
Contributor	An entity responsible for making contributions to the resource.
Date	A point or period of time associated with an event in the lifecycle of the resource.
Type	The nature or genre of the resource.
Format	The file format, physical medium, or dimensions of the resource.
Identifier	An unambiguous reference to the resource within a given

[19] http://www.oclc.org/
[20] http://dublincore.org/

	context.
Source	A related resource from which the described resource is derived.
Language	A language of the resource.
Relation	A related resource.
Coverage	The spatial or temporal topic of the resource, the spatial applicability of the resource, or the jurisdiction under which the resource is relevant.
Rights	Information about rights held in and over the resource.

Table 2.1 Dublin Core Metadata Element Set (DCMES)[21]

So far, Dublin Core has been adopted by a large number of prestigious scientific publishers, e.g. Oxford Universty Press[22], Nature Publishing[23], Sage[24], HighWire Press[25], Sciencemag[26], Ingenta[27] and Biomedcentral[28], etc.

2.1.2 Learning Object Metadata

Learning Object Metadata (LOM) is an internationally recognized open standard developed by IEEE working group[29] for describing learning objects and similar digital resources used to support learning, education and training. The purpose of LOM is to facilitate the reusability, discoverability and interoperability of learning objects.

As shown in Figure 2.1, LOM comprises a hierarchy of grouped elements. At the first level there are nine categories, i.e.

[21] Source: http://dublincore.org/documents/dces/
[22] http://global.oup.com/
[23] http://www.nature.com/
[24] http://www.sagepub.com/
[25] http://highwire.stanford.edu/
[26] http://www.sciencemag.org/
[27] http://www.ingentaconnect.com/
[28] http://www.biomedcentral.com/
[29] http://grouper.ieee.org/

(1) General

(2) Life Cycle

(3) Meta-Metadata

(4) Technical

(5) Educational

(6) Rights

(7) Relation

(8) Annotation

(9) Classification

And each of them contains several sub-elements.

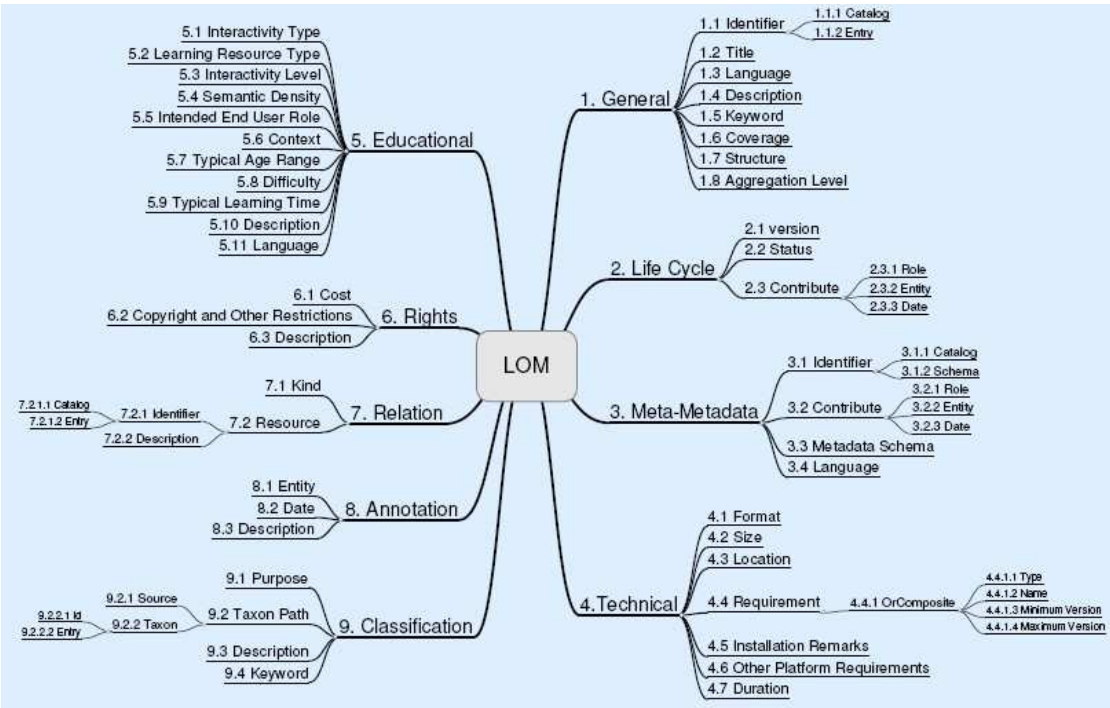


Figure 2.1 A Schematic Representation of the Hierarchy of Elements in the LOM Data Model [69]

2.1.3 BibTeX

BibTeX[30], mostly in conjunction with LaTeX[31], works as a tool and a file format for processing reference entries. By separating the bibliographic contents from its format, BibTeX simplifies the process of citing. This is similar to LaTeX's function of separating the information and the format of information.

BibTeX also becomes to be a de facto metadata schema (Table 2.2) since it is widely used and provided by both authors and digital libraries.

Entry Types	Article, Book, Conference, Inbook, Incollection, Inproceedings, Manual, Mastersthesis, Misc, Phdthesis, Proceedings, Techreport, Unpublished
Bibliography Items	Address, Annote, Author, Booktitle, Chapter, Crossref, Edition, Editor, Eprint, Howpublished, Institution, Journal, Key, Month, Note, Number, Organization, Pages, Publisher, School, Series, Title, Type, Url, Volume, Year

Table 2.2 BibTeX Metadata Schema

2.1.4 Schema.org

Supported by the three major search engines, i.e. Google, Yahoo! and Bing, Schema.org[32] is a joint effort to improve web searches by creating a shared structured data markup schema that helps optimize the display of search results and effective navigation for web users. On-page markup enables search engines to improve their understanding of the information on web pages, and provide more accurate, heuristic and richer search results.

[30] http://www.bibtex.org/
[31] http://www.latex-project.org/
[32] Schema.org: http://schema.org/

Schema.org maintains a collection of markup vocabularies, where schemas are a set of "types", each associated with a set of properties. Figure 2.2 illustrates the type hierarchy.

```
DataType
  |  Boolean
  |  Date
  |  Number
  |     |  Float
  |     |  Integer
  |  Text
  |     |  URL

Thing
  |  CreativeWork
  |  Event
  |  Intangible
  |  Organization
  |  Person
  |  Place
  |  Product

CreativeWork
  |  Article
  |     |  BlogPosting
  |     |  NewsArticle
  |     |  ScholarArticle
  |  Blog
  |  Book
  |  ItemList
  |  Map
  |  MediaObject
  |  Movie
  |  MusicPlaylist
  |  MusicRecording
  |  Painting
  |  Photograph
  |  Review
  |  Sculpture
  |  TVEpisode
  |  TVSeason
  |  TVSeies
  |  WebPage
  |  WebPageElement
```

Figure 2.2 Type hierarchy of Schema.org

The data model used by Schema.org is generic, extensible [70] and easily mapped into RDF Schema[33].

(1) *Types* are arranged in a multiple inheritance hierarchy where each type may be a sub class of multiple types.

(2) Each *property* may have one or more types as its domains, while this property may be used for instances of any of these types. Each *property* may have one or more types as its range, while value(s) of this

[33] http://www.w3.org/TR/rdf-schema/

property should be instances of at least one of these types [71]. Table 2.3 presents an example of property of “Scholarly Article” defined in Schema.org. For the full description of Scholarly Article properties, please refer to [72].

Hierarchy	Property
Properties from “Thing”	description, image, name, url
Properties from “CreativeWork”	about, aggregateRating, audio, author, awards, contentLocation, contentRating, datePublished, editor, encodings, genre, headline, inLanguage, interactionCount, isFamilyFriendly, keywords, offers, publisher, reviews, video
Properties from “Article”	articleBody, articleSection

Table 2.3 Property of “Scholarly Article” defined in Schema.org

2.1.5 Google Scholar

Google Scholar[34] is the most commonly used search engine in today's field of science. It helps users find academic literature, including journal articles, dissertations, books, preprints, abstracts and technical reports. The content covers the natural sciences, humanities, social sciences and other disciplines. Google Scholar’s literature rank is in strict accordance with the article’s academic value, the reference factors, which includes the authoritative of literature, authors and publishers and the reference frequency. Generally, the first choice of a reader who uses network resources to fulfill his or her information needs is to use search engines, such as Google, to do large-scale searches, followed by the use of specialized academic databases and finally the reading of academic journals. This sequence has formed a social habit. Therefore, more and more publishers and authors have begun to focus on Google Scholar's metadata schema in order to make their article more accurately indexed

[34] http://scholar.google.com/

by Google Scholar.

The following table is an official Google Scholar metadata tagging schema. It also supports Eprints[35], Digital Commons[36], DSpace[37] and many other formats.

Meta tag name	Description
title	The paper title
date	The official publication date
online_date	The online publication date
author	An author name. Multiple occurrencees of this tag are allowed
pdf_url	The full paper
conference_title	The conference name or the proceedings title (for conference and workshop papers)
journal_title	The journal name (for journal papers)
volume	The volume (for journal papers)
issue	The issue number (for journal papers)
issn	The journal ISSN (for journal papers)
isbn	ISBN number
firstpage	The first page of the article
lastpage	The last page of the article
dissertation_institution	The university name (for master's and Ph.D. thesis)
technical_report_institution	The institution name (for technical reports)
technical_report_number	The technical report number (for technical reports)

Table 2.4 Google Scholar Metadata Schema [73]

2.2 Scientific Discourse Representation

This subchapter presents a succinct review of existing dominant scientific publication representation models and projects. Conceptually, all of them share a similar representation form with the features of coarse-grained rhetorical structure, fine-grained rhetorical structure,

[35] http:// eprints.org/
[36] http:// digitalcommons.bepress.com/
[37] http:// dspace.org/

relations, domain knowledge and shallow metadata support [24]. Specifically, the ScholOnto (Scholarly Ontologies) project [74] and the SWAN (Semantic Web Applications in Neuromedicine)[38] project focuses more on modeling the argumentation. However, in this thesis, we are more interested in some of the approaches to modeling the rhetorical structure of publications, i.e. Harmsze Model, ABCED and SALT.

2.2.1 Harmsze Model

In 2000, Harmsze from the University of Amsterdam proposed a modularized structure to represent electronic papers on experimental sciences in her doctoral dissertation [75]. This is one of the first comprehensive models of rhetorical structure representation. Harmsze's model comprises of two parts: the *Modules* and the *Links*.

- Modules

A module is a self-contained functional information unit. Its composition does not depend on its length but is decided by the consistency and integrity of the information it contains. Similarly, the relationship between the modules can not only be achieved through the links but also through the complex modules. Here we can make an analogy: the equivalent of a basic module is an atomic entity, and they can be used to form more complex modules, which is the molecular entity.

At the same time, two different types of complex modules need to be distinguished: compound module and cluster module. As shown in Figure 2.3, in a compound module, the relevant module will set into a higher level module. For example: "Experimental methods" is consisted of a number of lower level modules. The cluster module is a

[38] SWAN: http://swan.mindinformatics.org/ontology.html

generalization of concepts. Module "Raw Data" is a cluster module. The division of modules is mainly based on the characteristics of the information and the conceptual function, which include Positioning, Methods, Results, Interpretation, Outcome and Meta-Information.

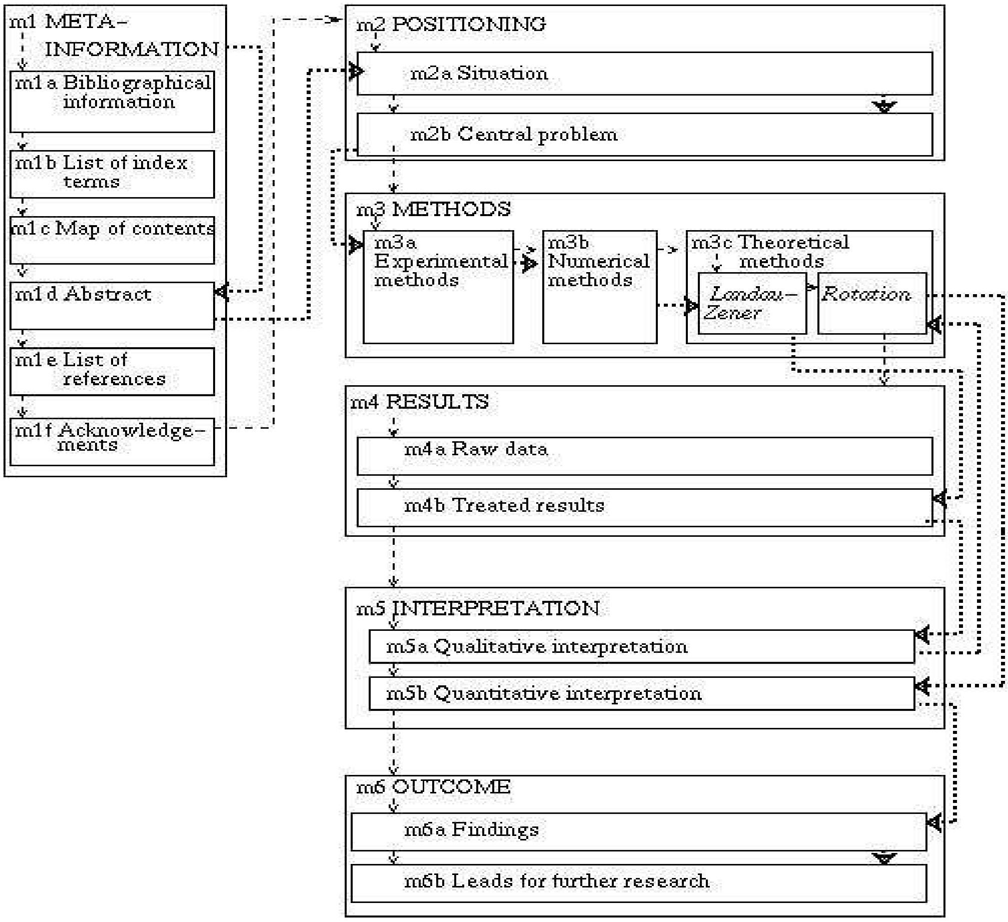


Figure 2.3 Harmsze Model[39]

- Links

In the traditional hypertext links, the relationship between the reader's linked objects is often unclear. Readers can only judge a standard hyperlink with impressions. For example, we know that a blue font and underlined text often provides us with a hyperlink.

In the Harmsze model, an author may define two categories of

[39] http://www.science.uva.nl/projects/commphys/papers/infwet/infwet.html

relationship, i.e.

(1) The organizational relations: Hierarchical, Proximity-based, Range-based, Administrative, Sequential and Representational.

(2) The scientific discourse relations: communicative function and content relations.

2.2.2 ABCDE Format

ABCDE Format is proposed by De Waard *et al*. in 2006, which provides an open standard and widely reusable format for creating rich semantic structures for the articles during writing. The "ABCDE" is an abbreviation which represents the following terms: Annotation, Background, Contribution, Discussion and Entities [76]. Using this format, people can easily mark papers semantically, especially in the LaTeX editing environment. To be specific,

- Annotation: Every article contains a set of metadata which can be used for retrieval, classification and so on. The most familiar one for us is the Dublin Core standard, which is also widely used in library management. For example, the article title, creator, identifier, date etc. They tend to be a part of the text of the article, but can also be relatively independent.
- Background: mainly used to introduce the background of the article, which includes the purpose and significance of research and development of the status quo and the core issues to be resolved.
- Contribution: mainly used to introduce the texts. The information within this section may include the contributions the study authors and the scientific community have made for academia. It may also discuss what new methods, theories or discoveries have been made

and the subsequent conclusions etc.

- Discussion: This part is mainly used to evaluate the work described in the article. It allows the article to be compared to similar research articles and discusses the impact of this institute and the direction of the research.
- Entities: throughout the whole text we will find that a large number of entities exist in the content of any article. The clearest examples of an entity are the references, as well as the names mentioned in the article, the project's website and so on. These entities are often found in footnotes, endnotes or references modalities. Usually, we can convert these entities to RDF format through data mining algorithms. In these cases, the RDF can include entity name, entity URI and the type of entity (such as reference, person or project).

De Waard *et al*. believe that any article is composed by the five ABCDE elements described above. Here, *abstract* is considered as a stress sentence. This set of sentences should come from the content that is covered by BCD. We can provide readers with the summary and general ideas of articles through the way of mark.

Meanwhile, De Waard *et al*. also conducted a study of semantic annotation. They developed an ABCDE structured style file and successfully applied it to Springer's LaTeX template (llncs.cls)[40].

2.2.3 SALT

SALT (Semantically Annotated LaTeX)[41] is developed by the Digital Enterprise Research Institute (DERI)[42]. It provides a semantic authoring

[40] http://www.springer.com/computer/lncs
[41] http://salt.semanticauthoring.org/
[42] http://www.deri.ie/

framework which aims to enrich scientific publications with semantic annotations and could be used during the authoring and post-publication process. It consists of three ontologies, i.e. Document Ontology, Rhetorical Ontology and Annotation Ontology [77], which deal with annotating linear structure, rhetorical structure and metadata of the document respectively.

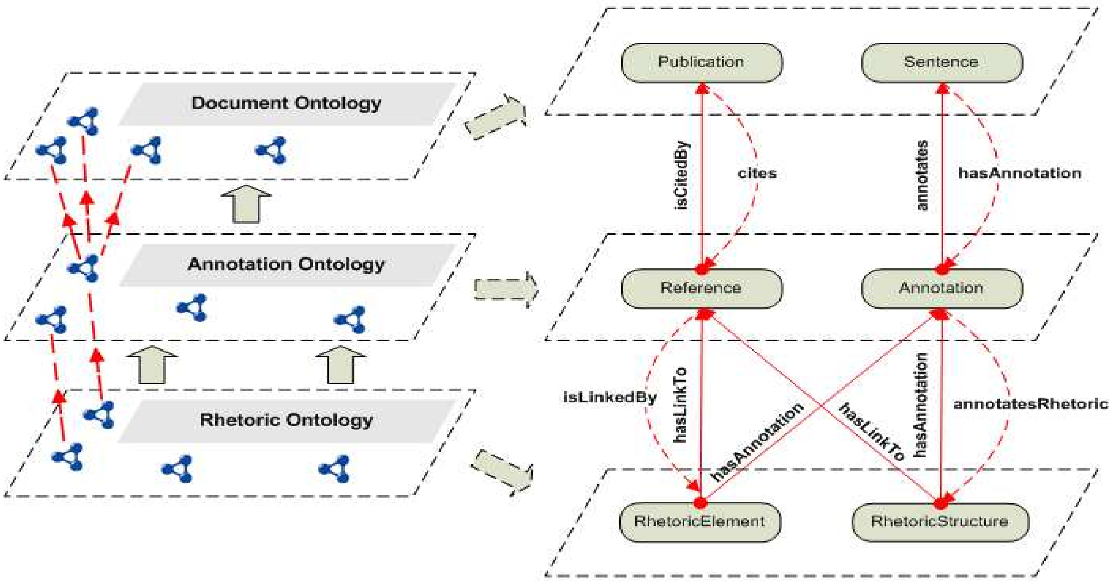


Figure 2.4 SALT Model[43]

In Figure 2.4, we can also see the relations defined in the SALT model. For example, in Rhetoric Ontology, "Rhetoric Element" and "Rhetoric Structure" exists, and the "Rhetoric Element" has a "hasLinkTo" relationship with the "Reference" found within Annotation Ontology. It also has a "hasAnnotation" relationship with the "Annotation" in Annotation Ontology. Similarly, "Reference" has a "isCitedBy" relationship with the "Publication" found in the Document Ontology; "Annotation" has a "annotates" relationship with the "Sentence" also found in Document Ontology. The various relationships between the definition achieve the three links of ontology.

[43] http://salt.semanticauthoring.org/

2.3 Scientific Publishing

In this section, we introduce four online publishing platforms which represent significant development and change. "Article of the Future" is dedicated to breaking the traditional linear reading of the paper structure. PLoS ONE focuses on "Publish first, Judge later". Nature Precedings create a pre-publication of the "post" platform to ensure the real author gets a wider range of comments and feedback before publishing the paper. The Liquid Publication Project mainly investigates the life cycle management of Scientific Knowledge Objects.

2.3.1 Article of the Future

From the first issue in 2010, the journal of Cell[44] began to launch a new format for online presentation of all research articles. The "Article of the Future" initiative aims to evolve the concept of a scientific publication in step with the development of new technologies and functionalities.

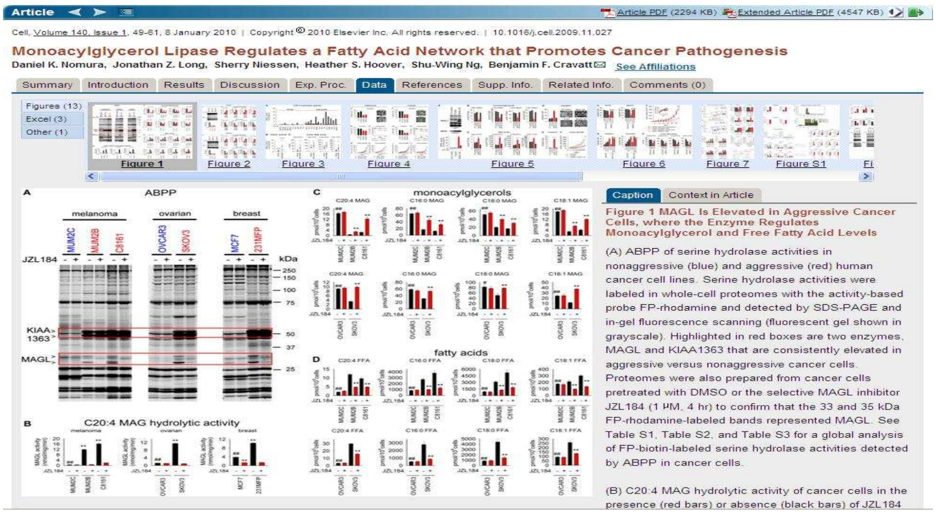


Figure 2.5 The "Data" of an Article with Presentation by "Article of Future" Format[45]

[44] Cell: http://www.cell.com

[45] Source: http://www.cell.com/abstract/S0092-8674%2809%2901439-1#Data

The Cell journal aims to develop an online format which breaks from the restraints of traditional linear structured paper and allows individual readers to create a personalized path through the discourse's content based on one's own interests or needs. "Article of the Future" proposed a new approach to organizing the traditional sections of the article by moving away from a strictly linear structure, required by print, towards a more integrated and linked structure. Tabbed and hyperlinked navigation through the Summary, Introduction, Results, Discussion, Experimental Procedures, Data, References, Supplemental Information, Related Information and Comments allows subject-area researchers to quickly access in-depth information on a specific experiment result, while providing more general readers a choice to gain the conceptual insights without being overwhelmed by additional details.

In addition to this, there are exciting functions that can be found within this designed architecture. For instance, Figure 2.5 shows the "Data" part of the paper. When a reader selects the Data tab, a film strip of thumbnails for all of the figures in the paper are collected and organized together which allows the reader to easily and rapidly scan through the data and then connect from an individual figure to the related context or textual discussion of findings. The Results tab offers a reader to view an enlarged figure and the associated Results text on a single screen. Additionally, Graphical Abstract and Highlights provided by this new format complements the traditional text Abstract and promotes paper browsing with a visual summary and bullet-points that effectively highlight and convey the main take-home messages of the article.

2.3.2 PLoS ONE

PLoS ONE[46] is an open-access, peer-reviewed, online journal published by Public Library of Science (PLoS)[47], which is the most prominent publisher in the open-access movement. PLoS ONE covers all disciplines within science and medicine, and the key idea of it is to "Publish first, judge later" [78]. This journal is built in a conceptually different way in comparison to traditional peer-reviewed scientific publishing.

Figure 2.6 PLoS ONE Platform[48]

Every paper submitted to PLoS ONE is reviewed by at least one editorial board member. The decision of acceptance or rejection is not assessed by the perceived importance and significance of a paper, instead, PLoS ONE only evaluates whether technical methods were conducted rigorously. It leaves future verification to the community-based peer review, following its online publication, which involves annotation, discussion and rating.

The PLoS ONE online platform provides features such as Online

46 http://www.plosone.org/
47 http://www.plos.org/
48 Source: http://www.plosone.org/article/info:doi/10.1371/journal.pone.0004018

Discussions, Ratings, Trackbacks[49], Open-access[50], Fast publication times, Post-publication tools, indicating quality and impact, Community-based dialogue on articles and Worldwide media coverage, etc.

2.3.3 Nature Precedings

Nature Precedings[51] is a permanent, citable, open-access repository for pre-publication research and preliminary findings in the fields of biomedical sciences, chemistry and earth sciences. Copyrights of publications submitted to Nature Precedings are retained by authors. It is an express channel for publishing findings at its beginning stage to distribute preliminary results, seek community opinions and prove originalities of findings. It complements the traditional review models and allows easier access to the content for citing, sharing and archiving. [79]. (Figure 2.7)

Figure 2.7 Nature Precedings[52]

[49] For example, if you link to a PLoS ONE article in your post, that article will display a link back to your blog post.

[50] Freely accessible online, authors retain copyright

[51] http://precedings.nature.com/

[52] http://precedings.nature.com/documents/6113/version/1

2.3.4 Liquid Publications

The Liquid Publication (LiquidPub) European Project[53] proposes a paradigm shift in the way scientific knowledge is created, disseminated, evaluated and maintained. This shift is enabled by the notion of *Liquid Publications* [80] which are evolutionary, collaborative and composable scientific contributions [81]. In 2009, Prof. Giunchiglia *et al*. proposed a formal model of Scientific Knowledge Object (SKO) and its associated structures [82]. Being a theoretical foundation of LiquidPub, the approach they presented is based on three organization levels (SKOnode, SKO and SKOset), three states (Gas, Liquid, and Solid), and four layers (File, Semantic, Serialization and Presentation, see Figure 2.8) that regulate the metadata and operations allowed at each level [83].

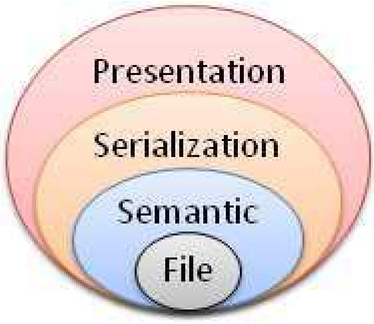


Figure 2.8 Four-layer Structure of SKO[54]

The term of *Liquid* borrows the concept of a physical liquid. As is known, the physical state of an object includes gas, liquid and solid. We can metaphor the generated knowledge and the text in the process of scientific publishing in the same way. We believe that the article's argumentations, research methods and research objectives are not quite clear in the process of our envisioning a paper, and we consider this

[53] http://liquidpub.org/

[54] Source: Giunchiglia Fausto and Chenu Ronald. Scientific knowledge objects v.1. Technical report, University of Trento, Italy, 2009.

period as *gas*. Similarly, when we start drafting a document or developing a project or a software product, we will first have a variety of drafts and then there will be a series of test versions. The entire article or products are constantly upgraded and improved before issuing. We call this phase *liquid*. After the articles are published, we cannot re-modify the article content. At this time, copyrights will be transferred to the publishers from the author. We call this the state of *solid*.

In the traditional scientific publishing field, we often face a reality in which an article is published or rejected. Once published, an article ends its life cycle. When the author has a new expansion of the experimental data or a new improved algorithm to obtain better results in a certain time, he or she cannot reuse the old article and a new article must be written. It is hoped that a scientific paper can be compatible with software engineering and have its own development process. This would mean that when an improvement is made, a completely new product does not need to be launched, instead, an updated version can be introduced, small bugs can be fixed or new features can be added. We hope to let these scientific publications be in a state of *Liquid*. This is the origin of the name *Liquid Publications* and is also the mission of this project. This thesis is partially supported by Liquid Publications Project.

Chapter 3

SKO Types

SKO Types is an entity-oriented theory for representing and linking Scientific Knowledge Objects by defining entities, relationships between entities, and attributes of each entity in the scientific domain. In SKO management, SKO Types serve as the basis for relating entities, entity components, aggregated entities, relationships and attributes to various tasks, e.g. linked entity, rhetorical structuring, strategic reading, semantic annotating, etc., that users may perform when consulting ubiquitous SKOs.

This chapter is organized as follows:

Section 3.1 defines the entity types used in SKO Types and elaborates on their nature and scope, including SKO, SKO Set, SKO Node, and SKO-related entities such as *Researcher*, *Conference*, *Institution*, and *Project*.

Section 3.2 delineates the relationships that operate between entities (or specific instances of entities), such as *Syntactic Relationships*, *Content Relationships*, *Rhetorical Relationships*, *Part/Whole Relationships* and *Entity Relationships*.

Section 3.3 provides the definition of attributes associated with the entities defined for the SKO Types.

Section 3.4 maps SKO Types to some prominent bibliographic metadata standards that we intend to support and interoperate with.

3.1 SKO, SKO Set and SKO Node

The entity types that have been defined for SKO Types represent the key objects of bibliographically related data in a scientific domain, including SKO, SKO Set, SKO Node, and SKO-related entities.

3.1.1 SKO

An SKO, an abbreviation for Scientific Knowledge Object, is a type of entity of intellectual and artistic endeavour, which is defined as:

$$SKO = < T, \{A\}, \{R\}, \{S\} >$$

where

- T is one of the entity types in an SKO hierarchy.
- {A} is a non-empty set of attributes A, while there are several mandatory attributes, e.g. URI.
- {R} is a set of relationships R.
- {S} is a set of services S.

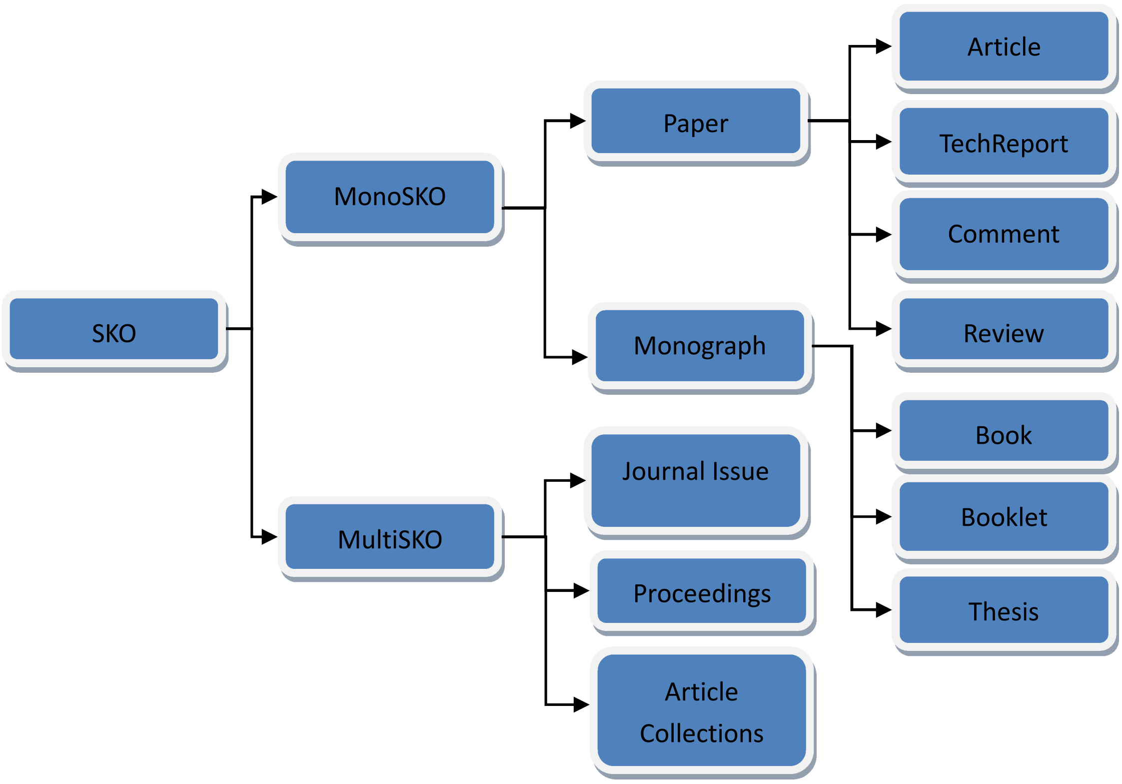


Figure 3.1 Entity Types in an SKO Hierarchy

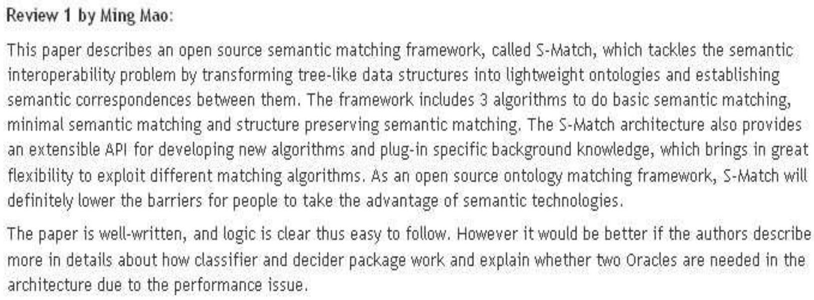
Figure 3.1 illustrates the entity types in an SKO hierarchy. SKO, as an entity type, has been divided into two subtypes, i.e. *MonoSKO* and *MultiSKO*. MonoSKO comprises *Paper* and *Monograph*, while MultiSKO consists of *Journal Issue*, *Proceedings* and *Article Collections*. Furthermore, *Paper* contains subtypes of *Article*, *TechReport*, *Comment*, and *Review*. Monograph includes *Book*, *Booklet* and *Thesis*.

In this hierarchy tree, the father entities are more generic than the children entities. In addition, the lattice makes the children nodes inherit all the attributes, relationships and services that their ancestors have.

Figure 3.2 and Figure 3.3 show a *Review* and a *Thesis* as instances of SKOs.

Examples

- **Review**



Review 1 by Ming Mao:

This paper describes an open source semantic matching framework, called S-Match, which tackles the semantic interoperability problem by transforming tree-like data structures into lightweight ontologies and establishing semantic correspondences between them. The framework includes 3 algorithms to do basic semantic matching, minimal semantic matching and structure preserving semantic matching. The S-Match architecture also provides an extensible API for developing new algorithms and plug-in specific background knowledge, which brings in great flexibility to exploit different matching algorithms. As an open source ontology matching framework, S-Match will definitely lower the barriers for people to take the advantage of semantic technologies.

The paper is well-written, and logic is clear thus easy to follow. However it would be better if the authors describe more in details about how classifier and decider package work and explain whether two Oracles are needed in the architecture due to the performance issue.

Figure 3.2 Review[55]

- **Thesis**

PhD Dissertation

International Doctorate School in Information and Communication Technology

DIT - University of Trento

ITERATIVE SCHEMA-BASED SEMANTIC MATCHING

Pavel Shvaiko

Figure 3.3 Thesis[56]

3.1.2 SKO Set

The SKO Types model permits us to represent aggregated SKOs as a whole, i.e. SKOset, and the component SKO as an integral unit, i.e. SKOnodes, in the same way as we present SKOs.

From a logical perspective, SKO sets and SKO nodes share the same characteristics as SKOs. For example, they express scientific knowledge, and they also have subject, author/editor, publisher, etc.

[55] http://www.semantic-web-journal.net/content/s-match-open-source-framework-matching-lightweight-ontologies

[56] http://static.digns.com/uploads/doctoral_school/documents/phd-thesis/XVIII/shvaiko_pavel.pdf

An SKO set is a set of SKOs whose attributes answer a *query*, and it is defined as:

SKO Set = < N, {T}, Q, {R}, {S} >

where

- N is the name of the SKO set.
- {T} is a set of entity types that the elements in this SKO set must belong to.
- Q is the query Q = < {A} > where {A} is a set of attributes.
- {R} is a set of relationships R.
- {S} is a set of services S.

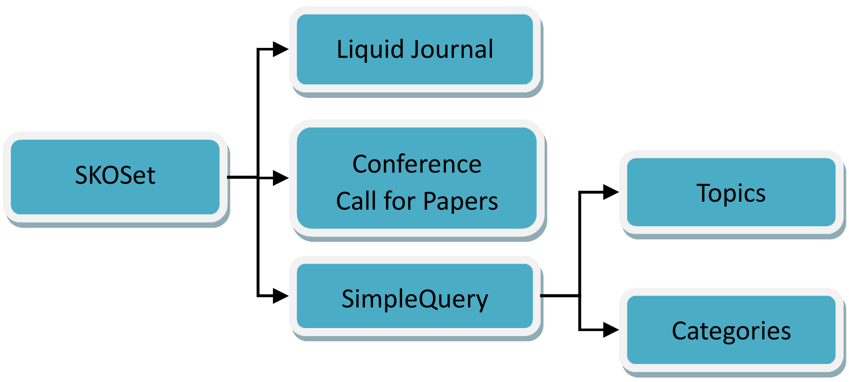


Figure 3.4 SKO set types and subtypes

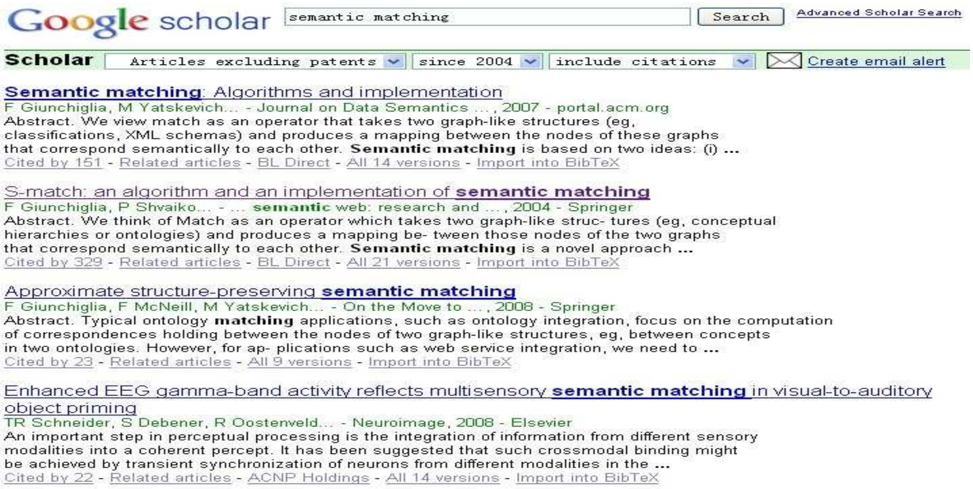
As shown in Figure 3.4, we define three types of SKOsets at the first level, i.e. *Liquid Journal*[57], *Conference Call for Papers*, and *Simple Query*, where Simple Query can be done using *Topics* or *Categories*.

Figure 3.5 and Figure 3.6 illustrate a *SimpleQuery* and a *Conference Call for Papers* as instances of SKOsets.

57 A research area of Liquid Publications European Project: http://project.liquidpub.org/research-areas/liquid-journal

Examples:

- **SimpleQuery**


Google scholar | semantic matching | Search | Advanced Scholar Search

Scholar | Articles excluding patents | since 2004 | include citations | Create email alert

Semantic matching: Algorithms and implementation
F Giunchiglia, M Yatskevich... - Journal on Data Semantics ..., 2007 - portal.acm.org
Abstract. We view match as an operator that takes two graph-like structures (eg, classifications, XML schemas) and produces a mapping between the nodes of these graphs that correspond semantically to each other. **Semantic matching** is based on two ideas: (i) ...
Cited by 151 - Related articles - BL Direct - All 14 versions - Import into BibTeX

S-match: an algorithm and an implementation of **semantic matching**
F Giunchiglia, P Shvaiko... - ... **semantic** web: research and ..., 2004 - Springer
Abstract. We think of Match as an operator which takes two graph-like struc- tures (eg, conceptual hierarchies or ontologies) and produces a mapping be- tween those nodes of the two graphs that correspond semantically to each other. **Semantic matching** is a novel approach ...
Cited by 329 - Related articles - BL Direct - All 21 versions - Import into BibTeX

Approximate structure-preserving **semantic matching**
F Giunchiglia, F McNeill, M Yatskevich... - On the Move to ..., 2008 - Springer
Abstract. Typical ontology **matching** applications, such as ontology integration, focus on the computation of correspondences holding between the nodes of two graph-like structures, eg, between concepts in two ontologies. However, for ap- plications such as web service integration, we need to ...
Cited by 23 - Related articles - All 9 versions - Import into BibTeX

Enhanced EEG gamma-band activity reflects multisensory **semantic matching** in visual-to-auditory object priming
TR Schneider, S Debener, R Oostenveld... - Neuroimage, 2008 - Elsevier
An important step in perceptual processing is the integration of information from different sensory modalities into a coherent percept. It has been suggested that such crossmodal binding might be achieved by transient synchronization of neurons from different modalities in the ...
Cited by 22 - Related articles - ACNP Holdings - All 14 versions - Import into BibTeX

Figure 3.5 SimpleQuery[58]

- **Conference call for papers**


Call for Papers ESWC 2011 Tracks

In-Use Tracks

- Semantic Web In-Use (Olmedilla Daniel, Telefonica I+D, Spain-ES; Shvaiko Pavel, TasLab - Informatica Trentina S.p.A., Italy-IT)

Research Tracks

- Social Web and Web Science (Vrandecic Denny, KIT, Germany-DE; Passant Alexandre, DERI, Ireland-IE)
- Ontologies (D´Aquin Mathieu, Open University, United Kingdom-UK; Stuckenschmidt Heiner, University of Mannheim, Germany-DE)
- Reasoning (Hitzler Pascal, Kno.e.sis Center, Wright State University, Dayton, Ohio, United States-US; Della Valle Emanuele, Politecnico di Milano, Italy-IT)
- Semantic Data Management (Polleres Axel, DERI, Ireland-IE; Christophides Vassilis, FORTH-ICS and University of Crete, Greece-GR)
- Linked Open Data (Consens Mariano, University of Toronto, Canada-CA; Groth Paul, Free University of Amsterdam, Netherlands-NL; Lehmann Jens, University of Leipzig, Germany-DE)
- Software, Services, Processes and Cloud Computing (Norton Barry, KIT, Germany-DE; Stollberg Michael, SAP Research, Germany-DE)
- Natural Language Processing (Cimiano Philipp, University of Bielefeld, Germany-DE; Witbrock Michael, Cycorp, Slovenia-SI)
- Sensor Web (Alani Harith, KMI, Open University; Mottola Luca, Swedish Insitute of Computer Science, Sweden-SE)
- Mobile Web (Lassila Ora, Nokia, Finland-FI; Toninelli Alessandra, INRIA, France-FR)

Figure 3.6 Conference call for papers[59]

[58] http://scholar.google.com/scholar?hl=en&q=semantic+matching&as_sdt=1%2C5&as_ylo=2004&as_vis=0

[59] http://www.eswc2011.org/content/cfp

3.1.3 SKO Node

An SKO node is a component entity encapsulated in SKOs that semantically represent scientific knowledge as an integral unit.

An SKO node is defined as:

SKO node = < N, T, {A}, {R}, {S} >

where

- N is the name of the SKO node.
- T is the type of SKO that the SKOnode belongs to.
- {A} is a set of attributes.
- {R} is a set of relationships R.
- {S} is a set of services S.

Figure 3.7 describes the types of SKO nodes. The first level includes *TextChunk*, *Video*, *Audio* and *Data*. TextChunk can be further divided into two groups, namely *Syntactic Partition* and *Rhetorical Partition*. Syntactic Partition comprises *Chapter*, *Section*, *Paragraph*, *Sentence*, *Figure*, *Formula* and *Table*. Rhetorical Partition comprises *State of the Art*, *Problem Statement*, *Solution*, *Discussion*, *Methods*, *Material*, *Results* and *Evaluation*.

An SKO node is the smallest object in SKO Types that:

- Has a unique identifier.
- Was created independently.
- Can be cited independently.
- Can be reused autonomously.
- Can be published or distributed separately.
- Has separable copyright.

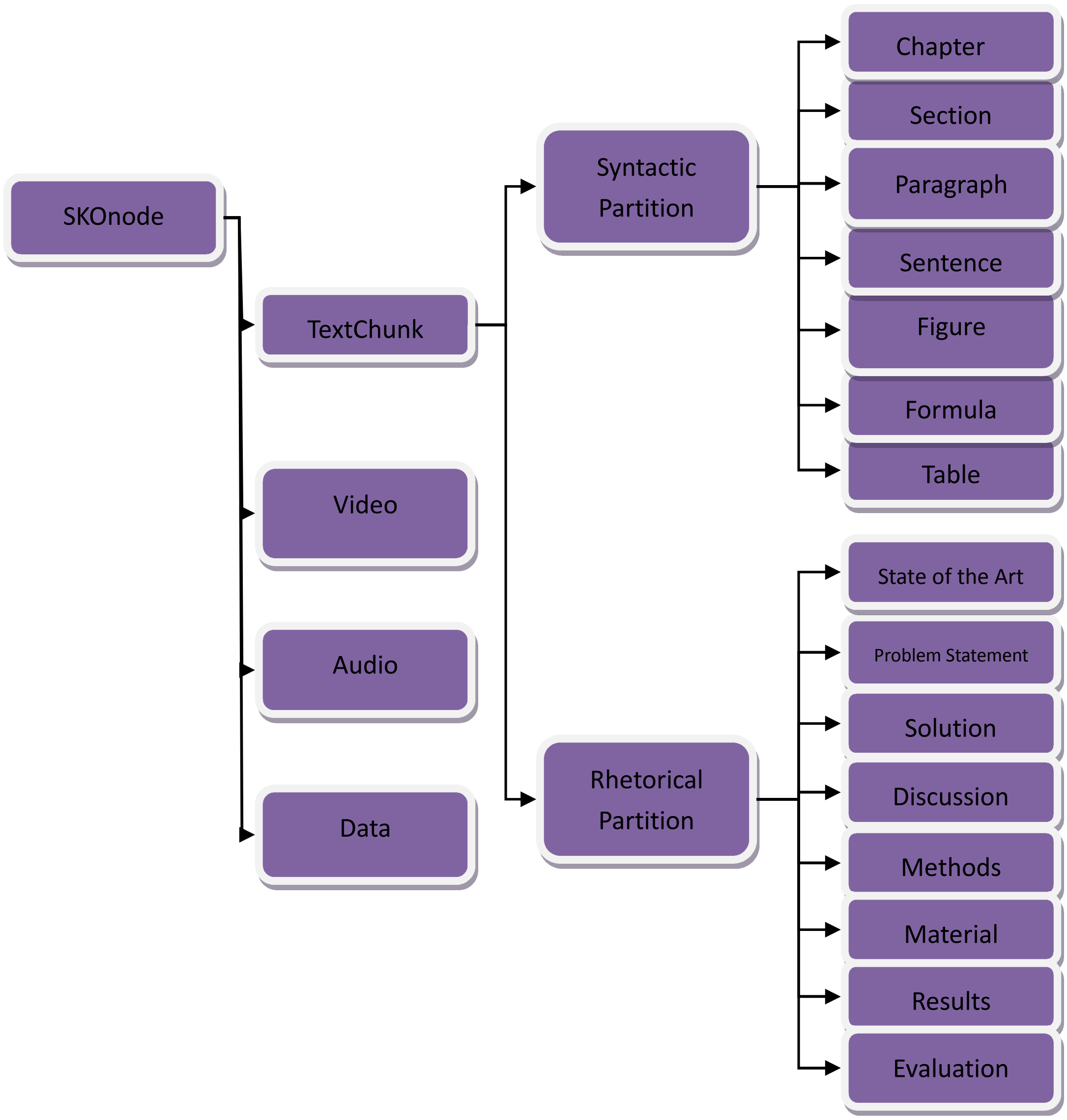


Figure 3.7 SKO node types and subtypes

Figure 3.8 and Figure 3.9 give two SKOnode instances of *Abstract* and *Video*.

Examples:

- **Abstract**

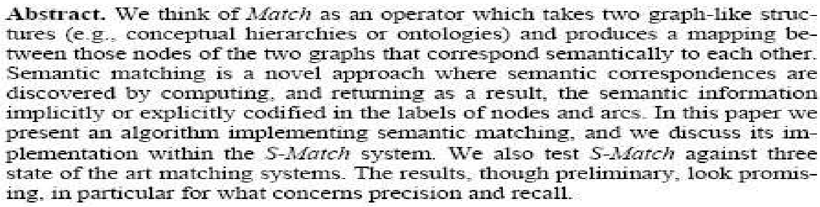

Abstract. We think of *Match* as an operator which takes two graph-like structures (e.g., conceptual hierarchies or ontologies) and produces a mapping between those nodes of the two graphs that correspond semantically to each other. Semantic matching is a novel approach where semantic correspondences are discovered by computing, and returning as a result, the semantic information implicitly or explicitly codified in the labels of nodes and arcs. In this paper we present an algorithm implementing semantic matching, and we discuss its implementation within the *S-Match* system. We also test *S-Match* against three state of the art matching systems. The results, though preliminary, look promising, in particular for what concerns precision and recall.

Figure 3.8 Abstract[60]

- **Video**

Figure 3.9 Video[61]

3.1.4 SKO-related Entites

In the scientific universe, there are several other entities which are tightly related to SKOs, SKOsets, or SKOnodes, that are responsible for the production, dissemination, or custodianship of knowledge such as *Researcher*, *Conference*, *Institution* and *Project*.

Note that the full definitions of these SKO-related entities are not the main scope of this thesis, although such entities may appear throughout this thesis.

[60]http://www.springerlink.com/content/vhu9mfhql6dveu94/fulltext.pdf
[61]http://videolectures.net/eswc2011_antoniou_shvaiko_award/

Generally speaking, an entity can be defined as:

Entity= < T, {A}>

where

- T is one of the entity types.
- {A} is a set of attributes A.

Actually, *Researcher* is a role of *Person*, *Conference* and *Project* are subtypes of *Event*, while *Institution* is a subtype of *Organization*. Full specifications should refer to the tech report of Entitypedia Project[62] conducted by the KnowDive group[63].

3.2 Relationships

Relationships abound in the scientific world. These may be educational, economic, social, legal, and so on. The relationships addressed herein are restricted to those involved in the representation and management of SKOs, including:

- Syntactic relationships: text structure, hyperlink.
- Content relationships: equivalent, derivative, descriptive, sequential, accompanying, shared characteristic.
- Whole/part relationships: whole-whole, whole-part, part-whole, part-part.
- Rhetorical relationships: state of the art, problem statement, solution, discussion, material, methods, results, evaluation.
- Entity relationships: relationships between SKO and SKO-related

[62] Entitypedia: http://entitypedia.org/
[63] KnowDive Group at University of Trento, Italy: http://disi.unitn.it/~knowdive/

entities.

Note that these five categories are not necessarily mutually exclusive, and we have endeavoured to attain and keep alignment with other relevant terminology systems such as FRBR[64], SPAR[65], etc. In SKO Types, we view a relationship as a particular kind of attribute, i.e. a relational attribute. In this subchapter, we describe these relationships accompanying sets of concrete instances, while the formal definition of (relational) attributes are proposed in Chapter 3.3.

3.2.1 Syntactic Relationships

- **Text Structure relationships:** these capture the linear document structure. For example (Figure 3.10), a paper may consist of some sections, and a section may have subsections, paragraphs, tables, algorithms or sentences. We use several relational attributes to describe this kind of syntactic structure relationship, such as hasTextChunk, hasChapter, hasSection, hasParagraph, hasSentence, hasStartPointer, hasEndPointer and so on.

Example:

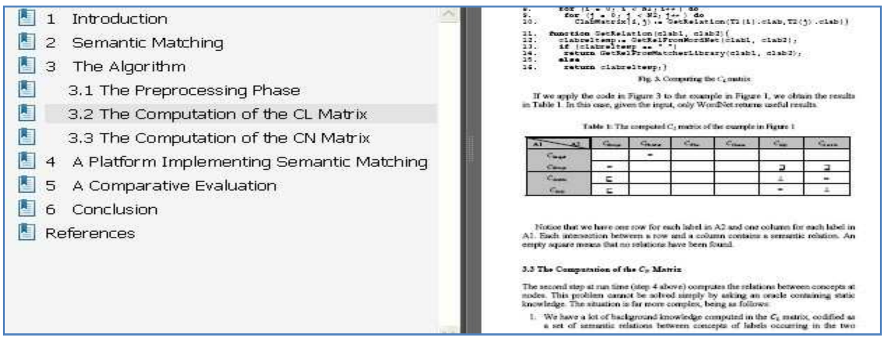


Figure 3.10 Syntactic Relationships

---

[64] Functional Requirements for Bibliographic Records: http://www.ifla.org/publications/functional-requirements-for-bibliographic-records

[65] The Semantic Publishing and Referencing Ontologies: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/SPAR/index.html

- **Hyperlink Relationships:** there are two types of hyperlinks, i.e. intratextual and intertextual. As the names imply, an intratextual hyperlink connects the source and target in the same SKO/ SKOset (Link1, Figure 3.11), while an intertextual hyperlink is a link between different SKOs/SKOsets/SKOnodes (Link2, Figure 3.11).

Example:

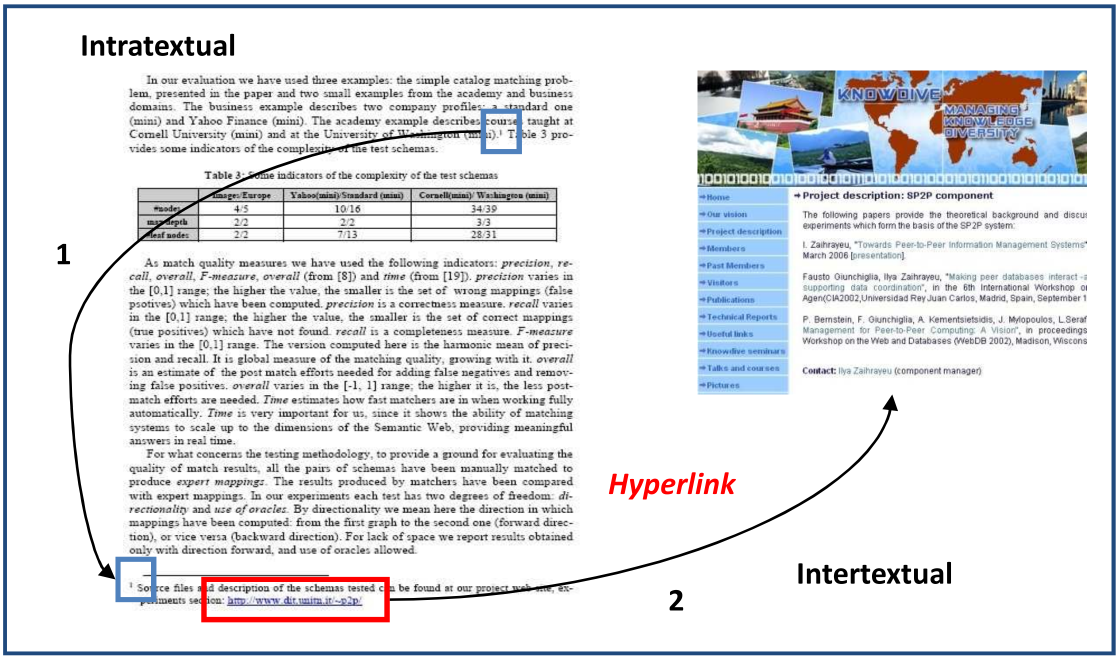


Figure 3.11 Hyperlink Relationships

3.2.2 Content relationships

The definitions of content relationships in SKO Types are derived from Tillett's dissertation (1987) [84], while the explanations are expressed in the context of the SKO Model introduced in Chapter 2.

- **Equivalent Relationships**, which hold between entities having

  (1) same data

  (2) same semantics

(3) same serialization

(4) different presentations

Example:

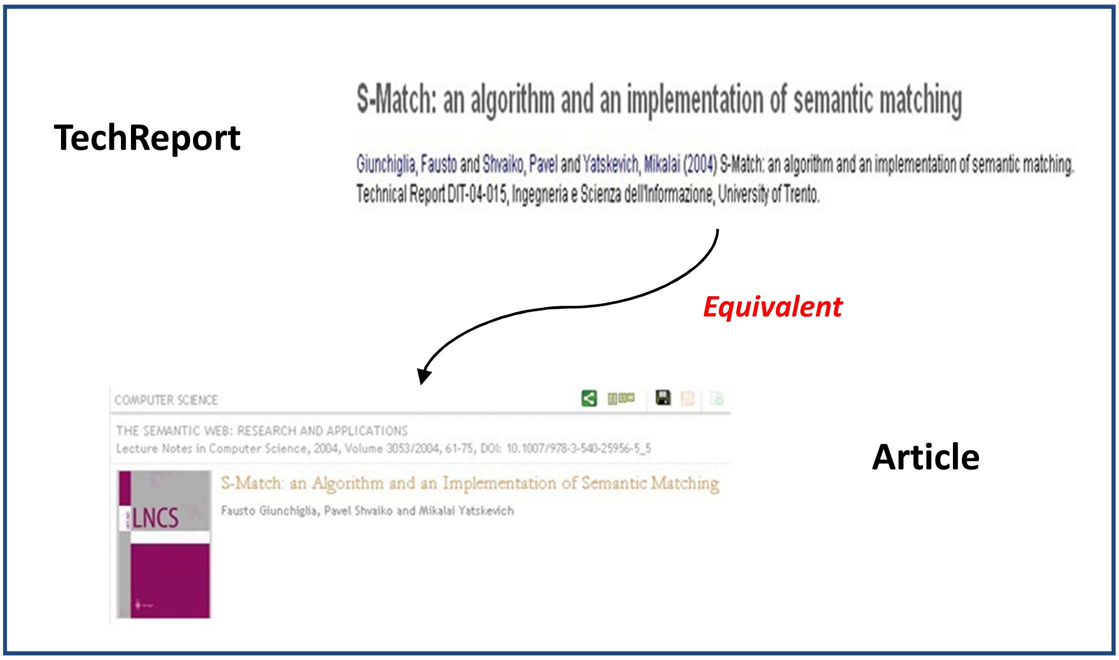


Figure 3.12 Equivalent Relationship

When submitting an article to a conference or a journal, we always formulate the manuscript as a tech report for internal or wider discussion, distribution and citation. This tech report may have the same data, the same semantics, and even the same serialization as a final publication in a conference or a journal, while it allows them having differences such as typesetting format or bibliographic metadata (e.g. publication date, publisher, etc.). In Figure 3.12, the DISI tech report "S-Match: an algorithm and an implementation of semantic matching" is *Equivalent* to a conference paper published in ESWC 2004 with the same title.

- **Derivative Relationships**. These hold between entities having the

  (1) same data

(2) same semantics

(3) different serialization

or

(1) same data

(2) enhanced semantics

Example:

There are entities based on the same semantics while having different serializations. For example, a presentation (PPT) of "S-Match" is *Derivative* from the conference paper of S-Match, as are the presentation video, a PhD thesis, and a book (Figure 3.13).

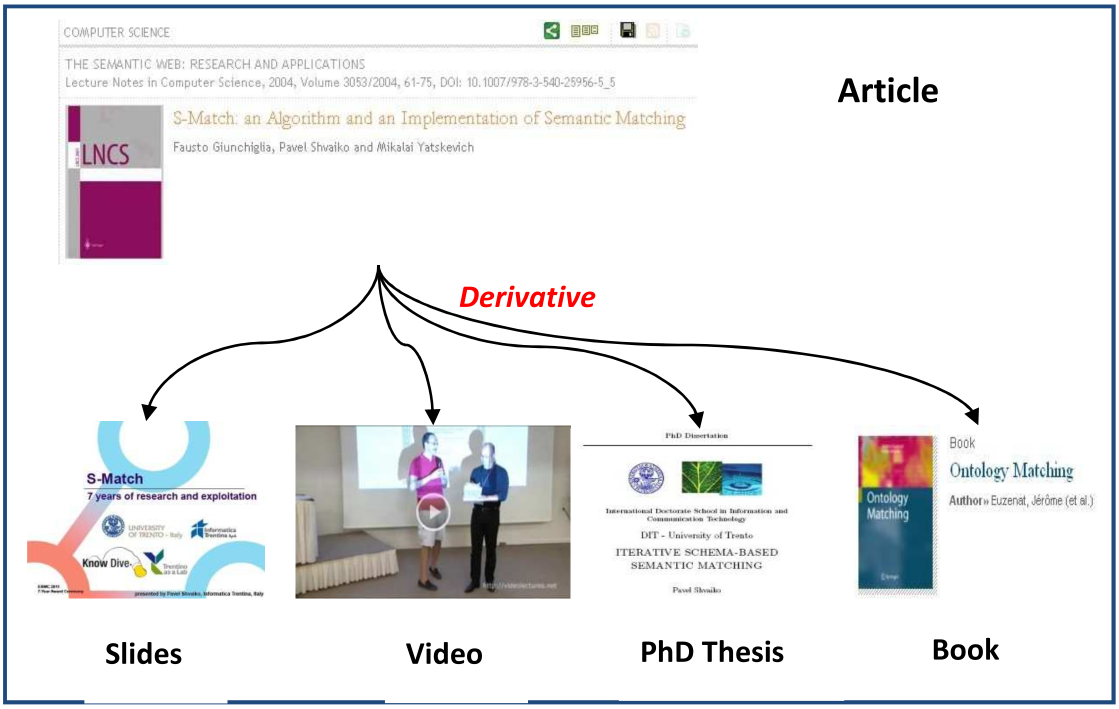


Figure 3.13 Derivative Relationships

- **Descriptive Relationships**, which hold between entities having the

  (1) same data

  (2) detailed semantics

There are always some SKOs based on the same data, although one provides a more detailed explanation or analysis, such as a description, a criticism, an evaluation or a review of the other.

Example:

A comment on "S-Match" and a review on "S-Match" have *Descriptive* relationships with the conference paper of S-Match (Figure 3.14).

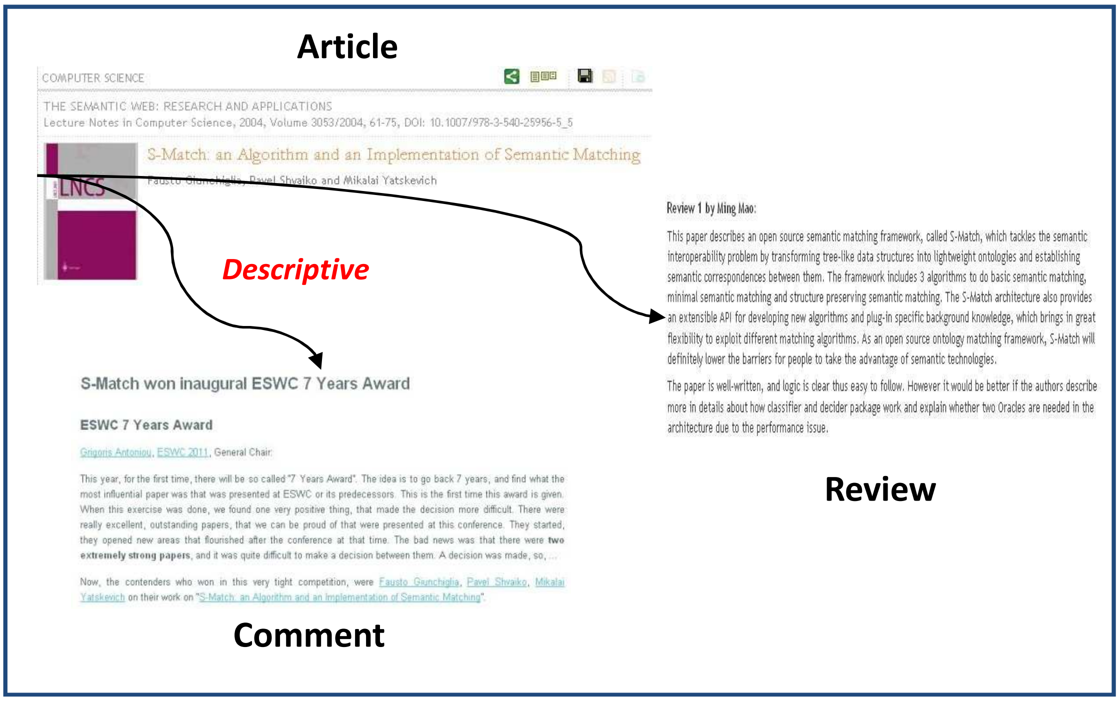


Figure 3.14 Descriptive Relationships

- **Accompanying Relationships**. These hold between entities which

  (1) augment each other equally or

  (2) in which one entity augments the other predominant entity.

Example:

During Pavel's presentation at the ESWC[66] 7-year award ceremony, there were also some *accompanying* videos or images (Figure 3.15).

[66] Extended Semantic Web Conference: http://www.eswc2011.org/

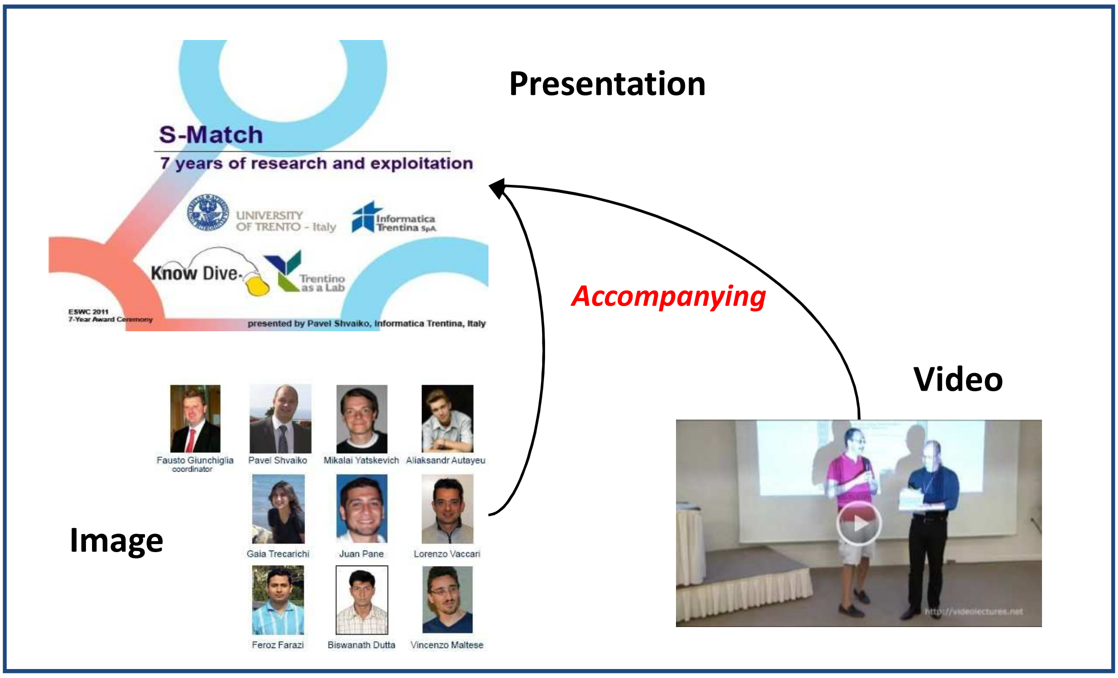


Figure 3.15 Accompanying Relationships

- **Sequential Relationships**. These hold between the SKOnodes continuing or preceding one another. In SKO Types, we consider two kinds of sequential relationships: one is the syntactical sequential relationships for ordering sections or pages as shown in Figure 3.16. The other are logical sequential relationships such as deduction, induction and abduction, which we delve into in Chapter 4- SKO Patterns.

Example:

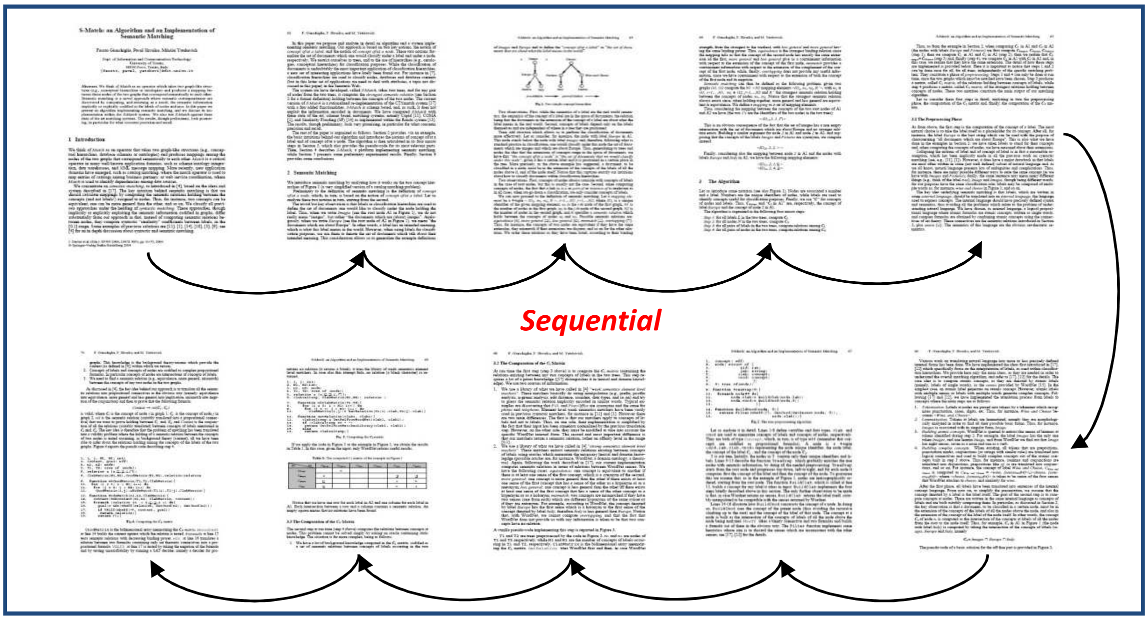


Figure 3.16 Sequential Relationships

- **Shared characteristic relationships**. These hold between entities having common attributes such as author, or title, as shown in Figure 3.17.

Example:

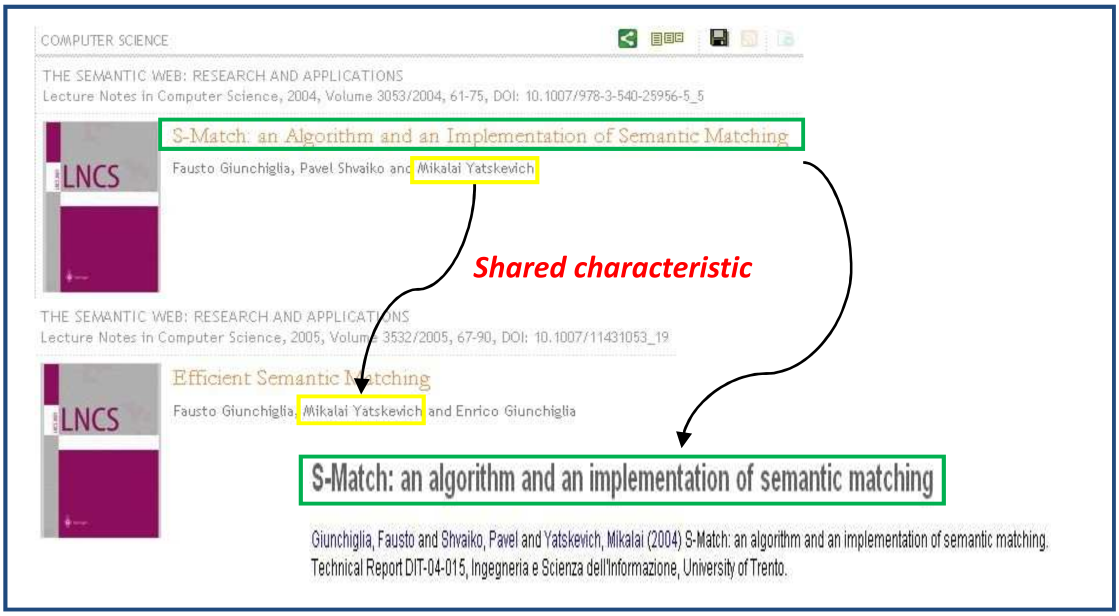


Figure 3.17 Shared Characteristic Relationships

3.2.3 Part/Whole Relationships

There are four kinds of part/whole relationships for bibliographic relations. These are whole-whole, whole-part, part-whole, and part-part. For example, a relationship from an SKOnode to an SKO could be considered as a part-whole relationship, such as when a paragraph cites a conference paper as a reference. In the same way, relationships between "SKO and SKO", "SKO and SKOnode", "SKOnode and SKO" are "whole-whole", "whole-part", "part-whole" in our theory. The reason for clarifying this is that there are various relationships among SKOnodes, SKOs and SKOsets, and we hope to denote the subject and object of a relation explicitly.

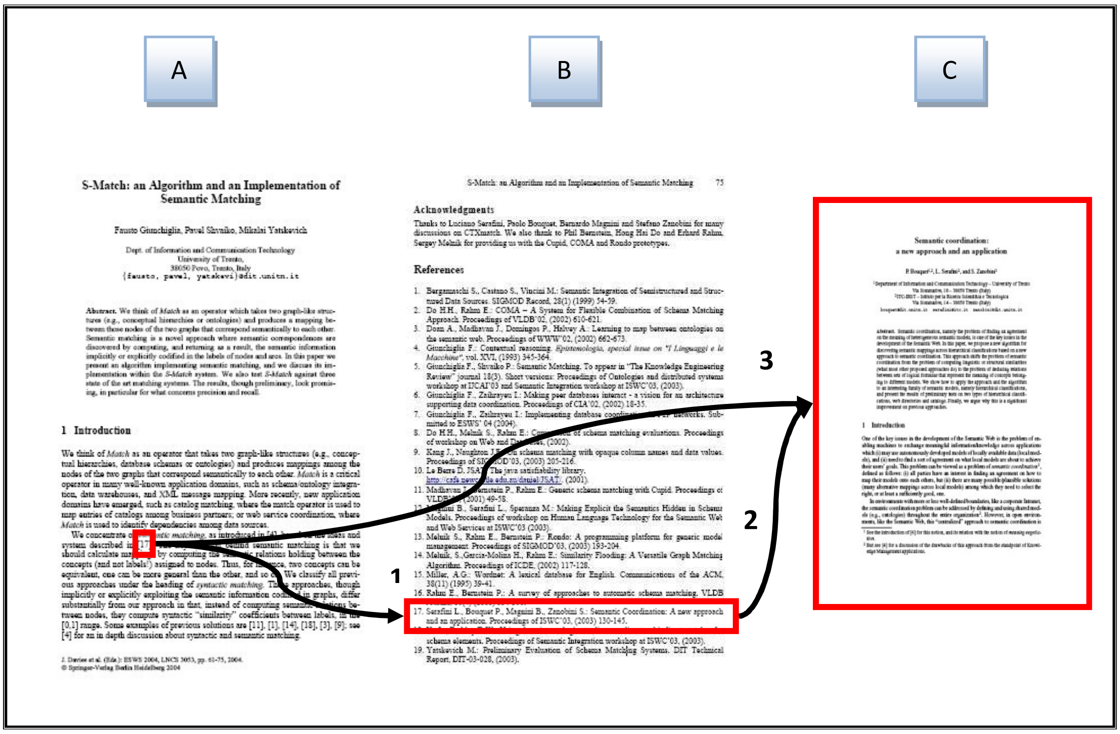


Figure 3.18 A Concrete Example of Part-Whole Relationships

Figure 3.18 illustrates a concrete example of part/whole relationships. "A" and "B" are parts of one paper entitled "S-Match: an Algorithm and an Implementation of Semantic Matching", and "C" is another referenced research paper “Semantic Coordination: A new approach and an application as a “whole”. In the “Introduction” section of A, there appears to be a citation “[17]”. In this scenario, two links are created from this text chunk to both “B” and “C”. Link1 is an internal part-part Relation which is from a citation to a piece of reference items at the end of the same paper. Meanwhile, Link3 is an external part-whole relation between a part of a paper and another whole paper. Link2 is also a part-whole relation from the reference item to the whole referenced paper.

We will specify these part-whole relationships in the following attributes definition section for each of SKO attributes as a column in the specification.

3.2.4 Rhetorical relationships

These relationships modularize the semantic structure of a document. We use these to denote the modularity of a paper. The attributes to realize this purpose include: state of the art, problem statement, solution, discussion, material, methods, results, evaluation. We elaborate on these rhetorical relationships in Chapter 4.

3.2.5 Entity relationships

As is shown in Figure 3.19, an SKO may have many relationships with other SKO-related entities. For example, an SKO and a Researcher may have a relationship of “author”, an SKO may “acknowledge” a Project. An SKO may be “submittedTo” a Conference. And a Researcher may have an “affiliation”, which is to an Institution.

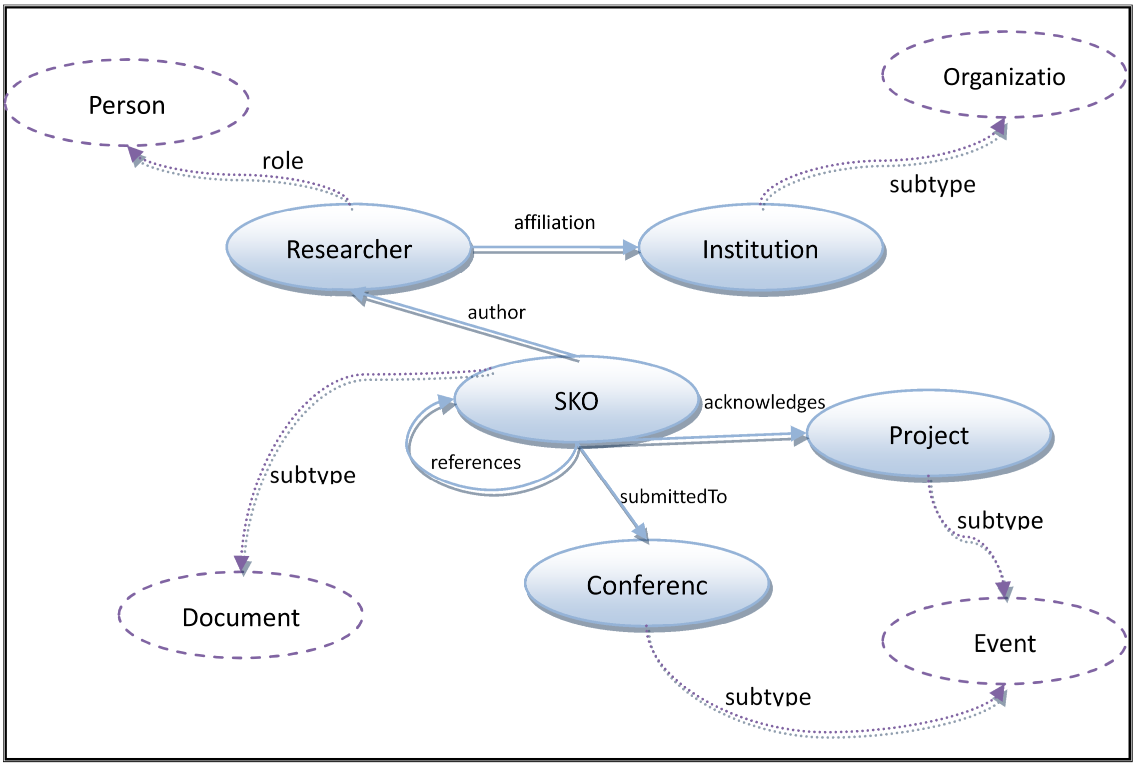


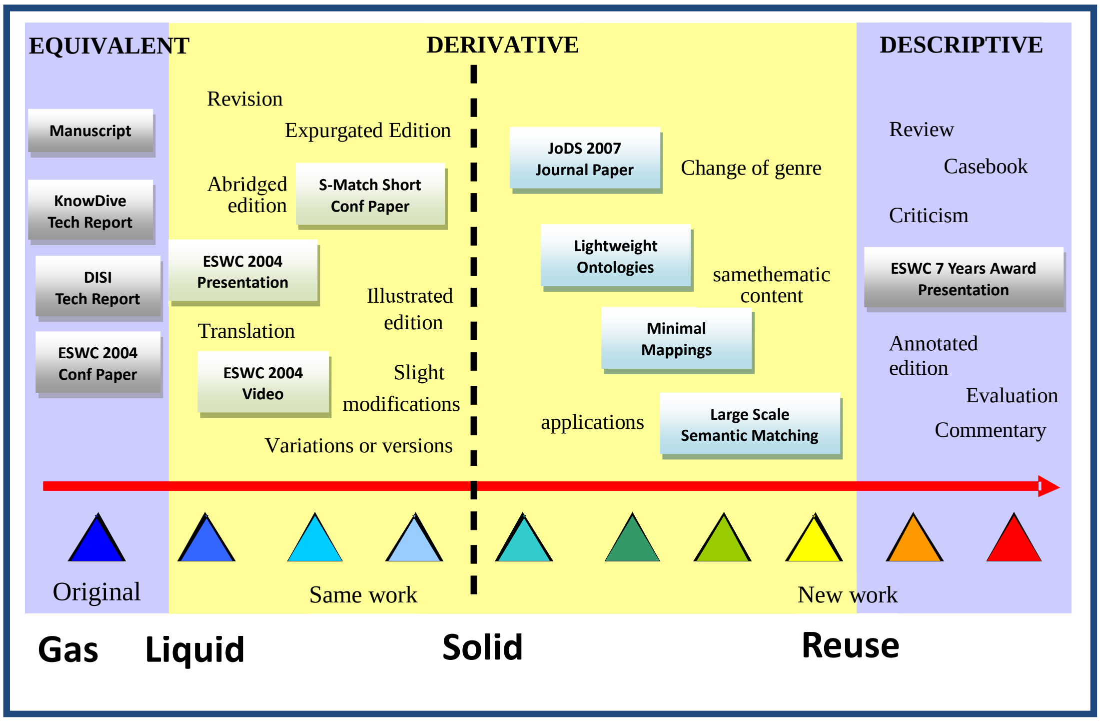
Figure 3.19 Entity Relationships

3.2.6 Family of SKOs: An Example

One of the distinctive features of SKO theory is that it keeps evolving during its entire lifecycle, namely gas, liquid and solid. Figure 3.21 gives a concrete story of the work "S-Match". When the ideas and manuscripts of S-Match are discussed and distributed internally in the KnowDive group, it exists in the gas stage. The milestone of its liquefaction is when it is published openly to communities with modalities of a DISI tech report and an ESWC conference paper. Then, more SKOs are derived from the original work of "S-Match" such as an abridged edition, a conference presentation, or some slight modifications, while all of these are based on the same work (semantic) and become more stable. Along with its solidification, "S-Match" keeps evolving and being reused in terms of new work or topics, e.g. Lightweight Ontologies, Minimal

Mapping, Large Scale Semantic Matching, etc. In addition, more descriptive SKOs appear, including Review, Evaluation, Annotations, Commentary, and so forth.

Figure 3.20 Family of SKOs[67]



3.3 Attributes

Each of the entities defined in SKO Types has associated with it a set of attributes. An attribute A is defined as:

$$A = < N, V >$$

where

- N is an attribute name

[67]This figure is based on the presentation of "Relationships in FRBR" (Page 12), by Barbara Tillett, at FRBR workshop, 2005.
http://www.oclc.org/research/activities/past/orprojects/frbr/frbr-workshop/program.htm

- V is an attribute value

In SKO Types, an attribute name is a concept, which means that there cannot exist two attributes with the same name. The attribute value domain consists of *Boolean*, *Integer*, *Float*, *Date*, *Duration*, *Semantic Less String*, *Semantic String*, *Entity* and *URL*. Note that an attribute definition allows multiple values and polymorphism, in which the data type domain can be a single data type, an array or a list of different data types.

For example, the attribute value of "*author*" is "*Researcher [] or Organization []*".

3.3.1 Abstract Model

Figure 3.22 specifies an abstract model for SKO Types. It defines the nature of the elements used and illustrates how those elements are combined to create structured knowledge representation. The model is presented here using a UML class diagram[68]:

- Each *SKO* is described using one or more *attributes*.
- Each *attribute* is made up of one *name* and one *value*.
- Each *name* is a *concept*.
- Each *value* is any of an *attributive value*, a *textual value* or a *relational value*.
    - An *attributive value* is a *value* which is a *concept*, e.g. the data type is Semantic String.

[68] Lines ending in a block-arrow should be read as 'is' or 'is a' and that lines starting with a block-diamond should be read as 'contains a' or 'has a'.

- A *textual value* is a *value* which is a *text* which carries an implicit semantic, e.g. the data type is Boolean, Integer, Float, Duration, or Semantic Less String.
- A *relational value* is a *value* which is a physical or digital *entity*, e.g. the data type is URL or Entity.

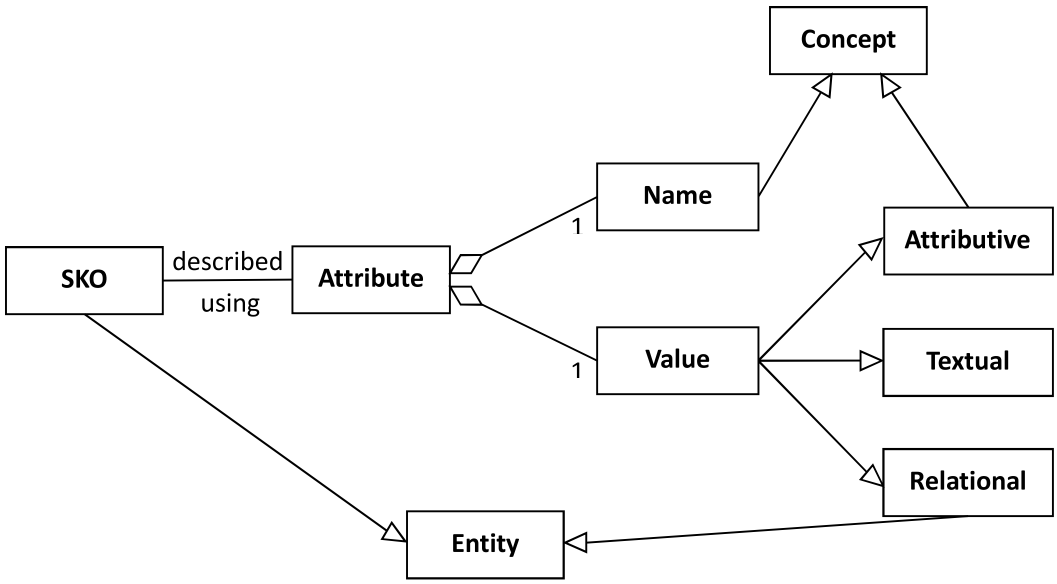


Figure 3.21 The Abstract Model for SKO Types

3.3.2 Attribute Specification

The attributes defined for SKO Types were derived from a comparative analysis of state-of-the-art metadata schemas such as DC, FOAF, LOM, etc. The scope of attributes included in our theory is intended to be comprehensive but not exhaustive.

For the focus of this research, the attributes for the other entities *Conference*, *Project*, *Researcher* and *Institution* include only those that are conventionally displayed as part of the Scientific Knowledge per se. Additional logical attributes are not included in this thesis.

We group related attributes into six categories as follows.

(1) The *general category* groups the general information that describes the SKO as a whole.

(2) The *lifecycle category* groups the features related to the history and current state of this SKO, and those who have affected this SKO during its evolution.

(3) The *relational category* groups features that define the relationship between the SKO and other entities.

(4) The *technical category* groups the technical requirements and technical characteristics of the SKO.

(5) The *rights category* groups the intellectual property rights, authorship, copyrights and conditions of use for the SKO.

(6) The *meta-metadata category* groups information about the metadata instance itself, rather than the SKO that the metadata instance describes.

Each attribute is specified by the following properties:

- ID: the unique identifier of an attribute.
- Name: the name of an attribute in NL.
- Data Type Domain: Boolean, Integer, Float, Date, Duration, Semantic Less String (SLS), Semantic String (SS), Entity, URL.
- Kind: Strictly Mandatory, Mandatory, Suggested, Permanent, Temporal, Computed, Transitive, Symmetric.
- Overrides: specifies a more general attribute name that this attribute “oversides”.
- Reference: for example, Dublin Core, SALT, FOAF, etc.

- Description: a brief account of an attribute in NL.
- Concept ID: the name of an attribute in FL.
- Whole/Part: indicates an attribute may apply in SKOs, SKOsets, or SKOnodes.
- Example: indicates when and how to use an attribute.

The following gives the current version of *SKO Types Specification*, which is being encoded and employed in the SWeb system[69] and AISN platform[70] [85].

General

Name	Datatype	W/P	Reference	Description	Example
identifier	URL	W&P	DC	An unambiguous reference to the resource within a given context.	www.liquidpub.org/doc/SKOTypes V1.9
description	SS	W&P	DC	An account of the resource.	This work is a branch of EType Theory.
language	SS	W&P	DC	A language of the resource.	English
keywords	SS []	W&P	DC: subject	The topic of the resource.	Taxonomy Mapping, Semantic Matching, Mapping Evaluation
coverage	SS	W&P	DC	The spatial or temporal topic of the resource, the spatial applicability of the resource, or the jurisdiction under which the resource is relevant.	16-19 century, Italy
creator	Person[] or Organization[]	W&P	DC	An entity primarily responsible for making the resource.	Hao Xu
source	URL	W&P	DC	A related resource from which the described resource is derived.	www.sweb.com/0001.pdf
title	SS	W&P	DC	A name given to the resource.	Scientific Knowledge Objects Types Version 2.0
alternative	SS	W&P	DC	An alternative name for	SKOTypes V2.0

[69] A Semantic Web system being developed by KnowDive Group.http://disi.unitn.it/~knowdive/description.php

[70] AI Social Network: http://disi.unitn.it/~knowdive/aisn.php

				the resource.	
pattern	SS	W	DC: conformTo	An established standard to which the described resource conforms.	SKO Pattern 001
author	Person[] or Organization[]	W&P	DC: contributor	A set of authors of this SKO.	Fausto Giunchiglia Ronald Chenu
editor	Person[] or Organization[]	W&P	DC: contributor	A set of editors of this SKO. Note:sometimes there is no author for an SKO like an article collection, but editors.	Hao Xu
references	SKO[] Or SKOnode []	P	DC	A related resource that is referenced, cited, or otherwise pointed to by the described resource. Note: internal reference is form Part to Part, while external one is from Part to Whole.	SKO Definition V3.0
serialization	URL	W		An SKO's serialization.	Skotypes.serial.xml

Table 3.1 Attribute Specification: General

Lifecycle

Name	Datatype	W/P	Reference	Description	Example
state	Enumeration< SS>	W&P		The current state of this SKO.Note: the attribute value should be one of “Gas”, “Liquid”, or “Solid”.	Liquid
hasVersion	SKO[]	W&P	DC	A related resource that is a version, edition, or adaptation of the described resource.	SKOTypes Version1.0 SKOTypes Version2.0 SKOTypes Version2.9
created	Date	W&P	DC	Date of creation of the resource.	01/01/08
dateOfSolidification	Date	W&P	DC: date	Date of solidification.	06/08/08
dateOfPublication	Date	W&P	DC: issued	Date of formal issuance (e.g., publication) of the resource.	06/08/08
publisher	Person[] or Organazition[]	W	DC	An entity responsible for making the resource available.	DISI
conditions	SS	W	DC: accrualMethod	The method by which items are added to a collection.	author=”Fausto”
dateOfAccept	Date	W	DC	Date of acceptance of the resource.	06/08/08
dateCopyrighted	Date	W&P	DC	Date of copyright.	06/08/08

dateSubmitted	Date[]	W	DC	Date(s) of submission of the resource.	06/08/08
submittedTo	Conference or CFP	W&P	DC	Resource(s) where this resource submitted to.	ESWC 2008
modified	Date	W&P	DC	Date on which the resource was changed.	06/08/08

Table 3.2 Attribute Specification: Lifecycle

Relational

Name	Datatype	W/P	Reference	Description	Example
hasTextChunk	SKOnode	P	SALT	Has a text chunk.	Foreword
hasChapter	SKOnode or SKO or SKOset	W&P	SALT	Has a chapter.	Chapter 1
hasSection	SKOnode or SKO or SKOset	W&P	SALT	Has a section.	Scection 1
hasParagraph	SKOnode	P	SALT	Has a paragraph.	Paragraph 1
hasSentence	SKOnode	P	SALT	Has a sentence.	Sentence 1
hasFigure	SKOnode	P	SALT	Has a figure.	Figure 1
hasTable	SKOnode	P	SALT	Has a table.	Table 1
hasFormula	SKOnode	P	SALT	Has a formula.	Formula 1
hasStartPointer	SKOnode	P	SALT	Has a start pointer.	In this section...
hasEndPointer	SKOnode	P	SALT	Has a end pointer.	... in the future.
isAbstract	SKOnode	P	SALT	It is an abstract rhetorical chunk.	Abstract
isBackground	SKOnode	P	SALT	It is a background rhetorical chunk.	Background
isMotivation	SKOnode	P	SALT	It is a motivation rhetorical chunk.	Motivation
isContribution	SKOnode	P	SALT	It is a contribution rhetorical chunk.	Contribution
isDiscussion	SKOnode	P	SALT	It is a discussion rhetorical chunk.	Discussion
isEvaluation	SKOnode	P	SALT	It is an evaluation rhetorical chunk.	Evaluation
isConclusion	SKOnode	P	SALT	It is a conclusion rhetorical chunk.	Conclusion

Table 3.3 Attribute Specification: Relational

Technical

Name	Datatype	W/P	Reference	Description	Example
format	SS	W&P	DC	The file format, physical medium, or dimensions of the resource.	Text
size	Integer	W&P	DC	The size of the SKO.	1024

Table 3.4 Attribute Specification: Technical

Rights

Name	Datatype	W/P	Reference	Description	Example
copyRight	SS	W&P	DC: rights	The copyright of this SKO.	Copyright to this paper in the Liquid Pub Platform remains with the authors or their assignees.
licence	SS	W&P	DC	A legal document giving official permission to do something with the resource.	This paper is provided under the terms of this creative commons public licence.

Table 3.5 Attribute Specification: Rights

Meta-metadata

Name	Datatype	W/P	Reference	Description	Example
creator	Person	W&P	DC	The person who creates this metadata record.	Hao Xu
timestamp	Date	W&P		The time that metadata is created or modified.	12:06, 01/03/11

Table 3.6 Attribute Specification: Meta-metadata

3.4 SKO Types and Previous Formalizations

Interoperability is one of the most important factors that we should consider during the practical development and implementation processes, since the SKO Types, along with the SKO Patterns and SKO TeX that we define in the latter chapters will be mainly applied in various digital libraries, while for the existing legacy of scientific publications and their associated metadata schemas, we are required to build up a compatible mechanism. This will be one in which the original metadata can be imported into our system on the one hand, generated according to the SKO Types metadata schema, while on the other hand, in order to promote our standard, we hope to provide more convenient updating methods for harmonizing with different kinds of libraries.

Here, we have already compared and matched SKO Types with the current metadata standards in several mainstreams and finally, have attempted to find a mutually compatible mechanism.

3.5.1 SKO Types and Dublin Core

In this section, we compare SKO Types with the Dublin Core. In SKO Types, the so called element which is defined in the Dublin Core is named "Attribute". In the Dublin Core, there is no definition of the relationship between "whole and part", neither the definition of semantic data types for attribute values, nor the definition about "Category". Such definitions which are used for the relationships and entities in the Semantic Web are the core concepts in SKO Types.

In this chapter, we have introduced a total of 15 basic elements in Dublin Core, called "DC Basic Element". These are already labeled in the column "Note".

Dublin Core Element	SKOType Attribute	Whole /Part	DateType	Category	Note
contributor	author editor	W&P	Person[] or Organization[]	General	DC Basic Element
coverage	coverage	W&P	Formula	General	DC Basic Element
creator	creator	W&P	Person or Organization	General	DC Basic Element
date	dateOfSolidification dateOfPublication	W&P	Date	LifeCycle	DC Basic Element
description	description	W&P	Formula	General	DC Basic Element
format	format	W&P	Formula	Technical	DC Basic Element
identifier	identifier	W&P	SURL	General	DC Basic Element
language	language	W&P	Formula	General	DC Basic Element
publisher	publisher	W	Person or Organization	LifeCycle	DC Basic Element
relation					DC Basic Element (all the relational

					attributes in SKOTypes)
rights	copyRights	W&P	Formula	Intellectual Property	DC Basic Element
source	source	W&P	SURI	General	DC Basic Element
subject	keywords	W&P	Formula []	General	DC Basic Element
title	title	W&P	Formula	General	DC Basic Element
type	kind	W&P	Enumeration< Formula >	General	DC Basic Element W: see BibTex P: see LaTex
abstract					
accessRights	accessRights	W&P	Person[]	Technical	
accrualMethod	conditions	W	Formula	LifeCycle	for SKOsets
accrualPeriodicity					
accrualPolicy					
alternative	alternative	W&P	Formula	General	
audience					
available					Service(T)
bibliographicCitation					Service(G)
conformsTo	Pattern	W	Formula	General	
created	created	W&P	Date	LifeCycle	
dateAccepted	dateAccepted	W	Date	LifeCycle	SKOs in SKOsets
dateCopyrighted	dateCopyrighted	W&P	Date	LifeCycle	
dateSubmitted	dateSubmitted	W	Date[]	LifeCycle	SKOs in SKOsets
educationLevel					
extent					
hasFormat					Service(T)
hasPart					Service(G)
hasVersion	hasVersion	W&P	SKO	LifeCycle	
instructionalMethod					
isFormatOf					Service(T)
isPartOf					Service(G)
isReferencedBy					Service(G)
isReplacedBy					
isRequiredBy					
issued					
isVersionOf					Service(L)
license	license	W&P	Formula	Intellectual Property	
mediator					
medium					
modified	modified	W&P	Date	LifeCycle	
provenance					
references	references	W&P	SKO[]	General	

replaces					
requires					
rightsHolder					Service(R)
spatial					
tableOfContents					SKOnodeType
temporal					
valid					Service(L)
	serialization	W		General	
	state	W&P	Enumeration< Formula >	LifeCycle	
	submittedTo	W&P	SURL[]	LifeCycle	

Table 3.7 Comparison between SKO Types and Dublin Core

3.5.2 SKO Types and LaTeX

As is well known, LaTex is an important tool for word processing and typesetting. Especially in science and engineering, including in Computer Sciences, LaTex is widely applied by scholars and graduate students. The process of using LaTex is different from what is done in Office Word, such as focusing on typesetting, setting font size, and numbering for chapters and references. Instead, it is completed by using one group of control commands and macros from LaTex. In LaTex, we need to construct the article by using labels, which is quite similar to the type of already defined SKOnode in SKO Types. In this respect, we are going to make comparisons between LaTex Label and the SKOnode kind as follows.

LaTex Label	SKOnode Kind	Note
title		Global
author		Global
institution		attribute of Author
email		attribute of Author
abstract		Global
keyword		Global
chapter	Chapter	
section/subsection/subsubsection	Section	
figure	Figure	

table	Table	
align	Formula	
ack		Entity[]
reference		SKO[]
Tableofcontents		SURL[]
	Video	
	Audio	
	Data	
	Text Chunk	
	Paragraph	
	Sentence	

Table 3.8 Comparison between SKO Types and LaTeX

In the table shown above, we can find that a set of command tags for the document structuring has been defined in LaTex, including title, author, institution, email, abstract, keyword, and chapter, section, subsection, sub-subsection, figure, table, align, acknowledgement, reference and table of content. By comparison, in the SKOnodes classification, we find that the main corresponding ones include chapter, section, figure, table and formula. We view the first six tags from LaTex which are applied in the model of SKO Types as metadata instead of content data. This theory will be introduced later, when SKO Patterns are described. Similarly, for the conceptions of video, audio, data, text chunk, paragraph, sentence in SKOnode kind, we will introduce them mainly for two reasons. One is that such SKOnode types can be the extension from single Article Form to Multimedia Articles, which includes audio, video and other supporting information. The other reason is that by offering more detailed classifications, can we achieve the required semantic structure, semantic annotation and other features.

3.5.3 SKO Types and BibTeX

BibTex is a tool to manage references and is usually used together with

LaTex. It is also viewed as a small database system, by which we can either manually add or can also directly import .bib files. In such a system, the cited entries is recorded when the paper is written, including metadata such as author, title, publications, pages, the press and published time, etc. of the cited article. In BibTex, references are classified in several groups, containing article, tech-report, book, booklet, and manual, master's thesis and so on, which is quite similar to the definition of SKO entity types by us in SKO Types, and therefore comparisons are made in this section.

<table>
<tr><th>BibTex</th><th>SKO Kind</th><th>SKO Kind</th><th>Note</th></tr>
<tr><td>article</td><td rowspan="2">Paper</td><td rowspan="9">Simple SKOs</td><td rowspan="9">We could consider a comment or a review as a kind of paper from the metadata point of view.</td></tr>
<tr><td>techreport</td></tr>
<tr><td>book</td><td rowspan="5">Monograph</td></tr>
<tr><td>booklet</td></tr>
<tr><td>manual</td></tr>
<tr><td>Master's thesis</td></tr>
<tr><td>PhD thesis</td></tr>
<tr><td rowspan="2"></td><td>Review</td></tr>
<tr><td>Comment</td></tr>
<tr><td>journal</td><td>Journal Issue</td><td rowspan="3">Complex SKOs</td><td rowspan="3"></td></tr>
<tr><td>proceedings</td><td>Proceedings</td></tr>
<tr><td>collection</td><td>Article Collection</td></tr>
</table>

Table 3.9 Comparison between SKO Types and BibTeX

Note that here we have to distinguish between differences in terms of metadata between simple and complex SKOs:

a) Simple SKOs have only *Authors*, both on the level of the whole (SKO) and that of the parts (SKOnodes). Complex SKOs have *Editors* on the level of the whole (complex SKO), and *Authors* on the level of the parts (constituent SKOs).

b) Complex SKOs are associated to SKOsets: Journal Issues to *Journals*, Proceedings to *Conferences*, and Article Collections to *Simple Queries.*

c) The topmost parts complex SKOs are SKOs, while the topmost parts of simple SKOs are SKOnodes.

d) Otherwise, there are no differences in terms of metadata between simple and complex SKOs.

Chapter 4

SKO Patterns

Emerging web services technology is driving profound changes in the methods of scientific communication in academic circles. Scientific discourse, as the basic unit of dissemination and the exploitation of research results, have steadily enhanced their discoverability and reusability in response to the advancement of Web 2.0, the semantic web, data-driven science and open source science. When a publication is highly semantically enriched, its information and data are always much easier to search, navigate, disseminate, and reuse, whereas most online articles today are still electronic facsimiles of linearly structured papers with descriptions of shallow metadata, lacking semantic knowledge and interlinked relationships among elementary modules of content.

In the last few years, a handful of models have been proposed for scientific discourse representations which aim to externalize the rhetoric and argumentation within publications [24]. Harmsze's model [75] is one of the first comprehensive models which attempted to present the rhetorical structure of scientific information in electronic articles. The ABCDE format [76] organizes papers into five types of rhetorical blocks: *Annotation*, *Background*, *Contribution*, *Discussion* and *Entities*. This is similar to the IMRD (*Introduction*, *Methods*, *Results*, *Discussion*) structure

[94]. SALT (Semantically Annotated LaTeX) [77] is constituted by three ontologies (Document Ontology, Rhetorical Ontology, Annotation Ontology) and is dedicated to an authoring framework targeting the enrichment of scientific discourses with metadata. Conceptually, all of these representation models for rhetorical structuring are analogous, while the theoretical foundations, such as the Rhetorical Structure Theory (RST) [27] or Cognitive Coherence Relations [26] are different.

In this chapter, we propose Scientific Knowledge Object (SKO) Patterns towards a general discourse representation model, especially for knowledge management in the emerging social web and semantic web. Such a model not only draws on the essence of the above-mentioned rhetorical structured models, but also extends the capabilities of semantic annotation, semantic search, and strategic authoring grounded on logical reasoning (i.e. *deduction*, *induction* and *abduction*) [86].

With reference to the SKO Model, the SKO Patterns mainly work in the semantic and serialization layers in order to help pattern users establish semantic documentation with flexible rhetorical structures, along with extendable and interoperable metadata schemes. Potential users of our proposed patterns include scientific publishers, digital libraries, knowledge base developers, or even individual researchers and authors who want to make scientific publications more modularized, expressive, semantic and reusable.

This chapter is organized as follows:

Section 4.1 discusses the correlations of scientific method and scientific writing through the use of a parallel hourglass model.

Section 4.2 introduces some background knowledge of *pattern*, along with its definition conventions that are applied throughout this chapter.

Section 4.3 presents a pattern for the structure of a typical scientific paper based on the IMRD model.

Section 4.4 proposes the SKO Patterns for representing the rhetorical structure of scientific discourses on the semantic web. Three types of sub-patterns, namely deduction, induction, and abduction have been considered in depth.

Section 4.5 overviews this chapter and makes a comparison between proposed patterns.

4.1 Scientific Method and Scientific Writing

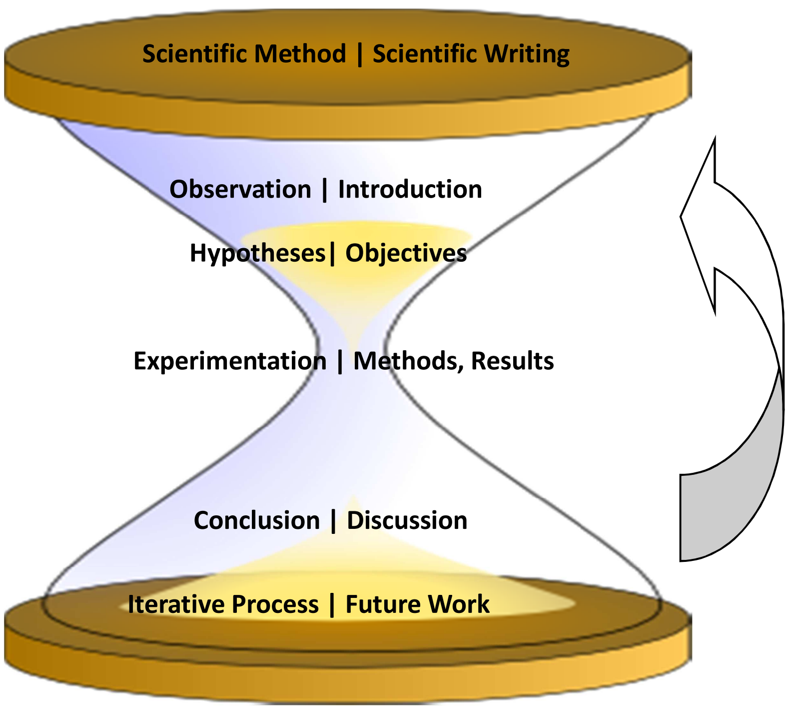


Figure 4.1 Parallel Hourglass Model for Scientific Method and Scientific Writing [87]

Along with the alteration in genre and topic, scientific writing supplies its own approach, a parallel process known as the scientific method. Generally, the scientific method partitions science from non-science, whereas scientific writing outlines the steps of scientific method both to the audiences of scientists and non-scientists.

Nevertheless, the structure of these two processes, scientific writing and the scientific method, is strikingly similar as can be seen from Figure 4.1, where we compare and understand them in terms of each other in a parallel hourglass model [88].

Four essential elements of the scientific method are iteration, recursion, interleaving and orderingin terms of the following [89]:

- Observation: An observation is the act of noting and recording something with instruments. Observations help scientists decide how certain variables might affect the problem.
- Hypothesis: A hypothesis is a tentative explanation that accounts for a set of facts and can be tested by further investigation.
- Experimentation: An experiment is an examination under controlled conditions that is made to show a known fact, or verification of a hypothesis.
- Conclusion: A conclusion is the result or outcome of an act or process.

Similarly, most scientific publications contain four main sections [94], namely:

- Introduction: Define the problem and position it in terms of background knowledge and the state of the art within the context.

- Methods: The method is the process or steps used in an experiment. This should be very detailed and include the materials needed.
- Results: The results are the facts or data that the researcher collects from his/her experiment.
- Discussion: This compares and evaluates the results with related work.

Basically, as demonstrated in the hourglass model, both scientific method and scientific wiring follow a general (background) – specific (certain problem solution) – general (discussion with others) pattern. As if the hourglass is upended, the whole process can be iterated illustrated by the arrow connecting two parts of the hourglass. The main components of scientific method and scientific writing are shown in Table 4.1.

Scientific Method	Scientific Writing
Observation	Introduction
Hypotheses	Objectives
Experimentation	Methods Results
Conclusion	Discussion
Iterative Process	Future Work

Table 4.1 Component Mapping between Scientific Method and Scientific Writing

4.2 Pattern

When a designer designs something such as a building, a program, or a piece of furniture, etc., s/he always comes out with a set of possible solutions for solving certain problems. A *pattern* is informally defined as *a type of theme of recurring events or objects, sometimes referred to as* *elements of a set of objects*[71]. In other words, a pattern describes an occurring problem and provides a reusable solution which facilities making decisions from well-known uses within a field of expertise. Christopher Alexander, an architect, first coined the term *pattern language* in 1977 in his book “A Pattern Language: Towns, Buildings, Construction” [90] derived from timeless entities called patterns. A pattern language, formed by a set of patterns, indicates relationships between the patterns therein, and helps designers to better understand related problems that must be solved. Although patterns originated as an architectural concept, the concept gained popularity in computer science following the publication of the book “Design Patterns: Elements of Reusable Object-Oriented Software” [91] in 1994. Software engineers very often use design patterns as a bible for handling programming problems which recur over and over. Moreover, patterns have also been applied to construct and modularize ontologies that guarantee the adoptability and maintainability of concepts in complex and heterogeneous scenarios [92].

In our case, we use patterns to represent how a scientific discourse can be structured by its semantics and rhetorics. Such an SKO can be segmented into SKOnodes and into the links between them, while an SKOnode can be manipulated independently and reused in other SKOs or SKOsets. Instead of defining a large number of complex and diversified structures, we have identified a small number of structures/patterns with regard to a general reusable solution that is sufficient to express what most users need. Such a low number of patterns is capable of capturing the most relevant document structures and is compatible with SKOTypes and other metadata standards.

By convention [93], pattern definition may be described in terms of

[71] http://en.wikipedia.org/wiki/Pattern

the *context* of use, the *problem* that the pattern addresses, the *forces* of the scenario, the *solution* to the problem, the *rationale* of the mechanism, the *benefits* of the solution that resolves the forces, the *liabilities* of such a solution, along with the *examples* of existing related projects and applications.

4.3 A Typical Pattern for a Scientific Paper

4.3.1 Context

- A scientific paper reports original empirical and theoretical work in the natural and social sciences as the basic functional unit of scientific knowledge dissemination among researchers. This pattern guides the authors towards a typical writing style that is widely accepted by various publishers.
- Papers written in this pattern also facilitate reading. Sections are well organized and structures are clear to understand.

4.3.2 Problem

How to structure a scientific paper?

4.3.3 Solution

The general structure of a paper comprises four major sections: introduction, methods, results and discussion [94]. The introduction leads the reader from general motivations and a broad subject area to a particular research question to be dealt with in the paper. Then the paper stays within a tight thematic scope, describes the research methods and results in detail. Finally, the discussion section aims to draw general conclusions from the particular results. Besides, there are additional parts of a paper which are of equal importance: title, author, abstract, keywords, acknowledgement and references as the meta-information of a paper, as shown in Figure 4.2 [95].

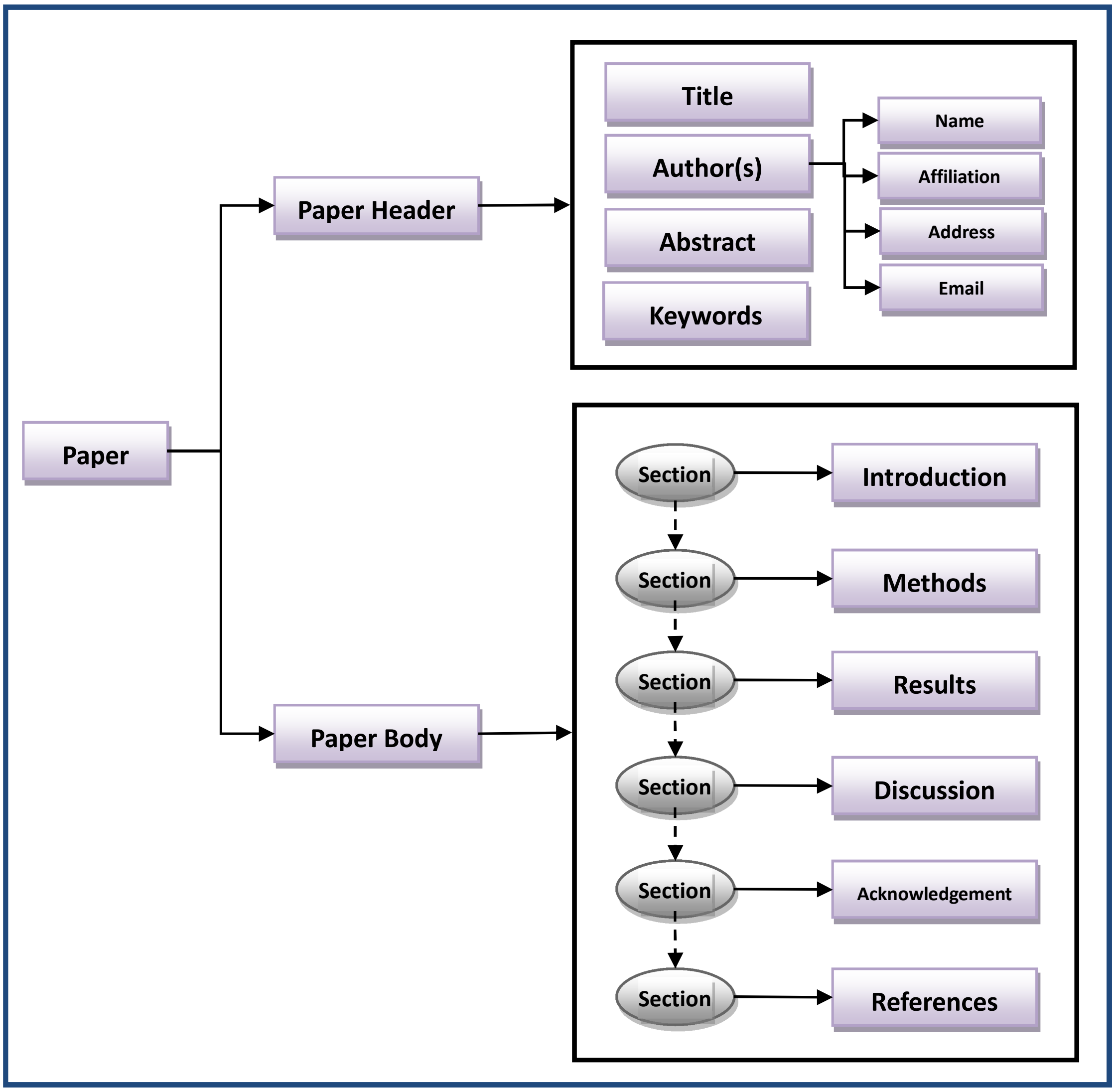


Figure 4.2 A Typical Paper Pattern

- *Paper Header*

  Title

  Author, containing name, affiliation, address and email information

Abstract

Keywords

- *Paper Body*

  1. Introduction: Define the problem and position it into background knowledge and the state of the art within the context.

  2. Methods: The method is the process or steps of your experiment. This should be very detailed and include materials needed.

  3. Results: The results are the facts or data that you collect from your experiment.

  4. Discussion: Compare the results with related work as evaluation.

  5. Acknowledgement: An expression of gratitude for assistance in the paper.

  6. References: A list of bibliographies cited in the paper.

4.3.4 Examples

- Exploiting Background Knowledge to Build Reference Sets for Information Extraction. Matthew Michelson and Craig A. Knoblock. IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009. 2076-2082 [96]

  Abstract

  1. Introduction

  2. Seed-Based Reference Set Construction

  3. Experiments and Results

4. Related Work
5. Conclusion
References

4. 4 SKO Patterns

4.4.1 Context

People want to publish a research paper and make it easy for others to read, search, and reuse.

A scientific publication is always written and read in a linear structure as an indivisible knowledge unit. Its complex composition makes it hard for readers to access the target information directly, especially non-expert readers. A rhetorical structure unveils precise semantics of the paper under the processes of intuitive thinking. Moreover, metadata as supportive material link related data and knowledge. These would definitely facilitate the reading, dissemination, information retrieval, and semantic search.

4.4.2 Problem

A traditional paper does not represent its rhetorical structure explicitly and lacks semantic information.

4.4.3 Forces

- A traditional paper is always a self-contained narrative with a linear structure ordered by sections.
- A traditional paper has shallow metadata support for navigation and search.

- In a traditional paper, the conceptual structure is implicitly expressed to readers.
- It is difficult to automatically extract information and meta-information from a traditional paper.
- It is difficult to import, export, or integrate annotations of a paper by other researchers.
- In traditional papers, text is not linked to the underlying data.
- Different audiences are interested in different parts of a paper, and it is hard to access these parts directly in a traditional papers.
- A traditional publication has low capabilities in terms of social dissemination and collaboration, for example tagging, commenting, annotating, and sharing.

4.4.4 Solution

Compose an SKO paper with rhetorical structure and semantic metadata.

We modularize a scientific paper by logical functions of the information and reorganize it by rhetorical structure as our pattern solution for discourse representation. Above all, we divide a discourse into Metadata and Data parts. Herein, the Metadata consist of *bibliographic information*, *abstract*, *reference set*, *annotation*, and so on, while the Data part is the main body of a paper that is constructed via the general scientific method. The basic element of rhetorical structure is called the Rhetorical Block in our methodology. Figure 4.3 gives an overview of the SKO Patterns for scientific papers.

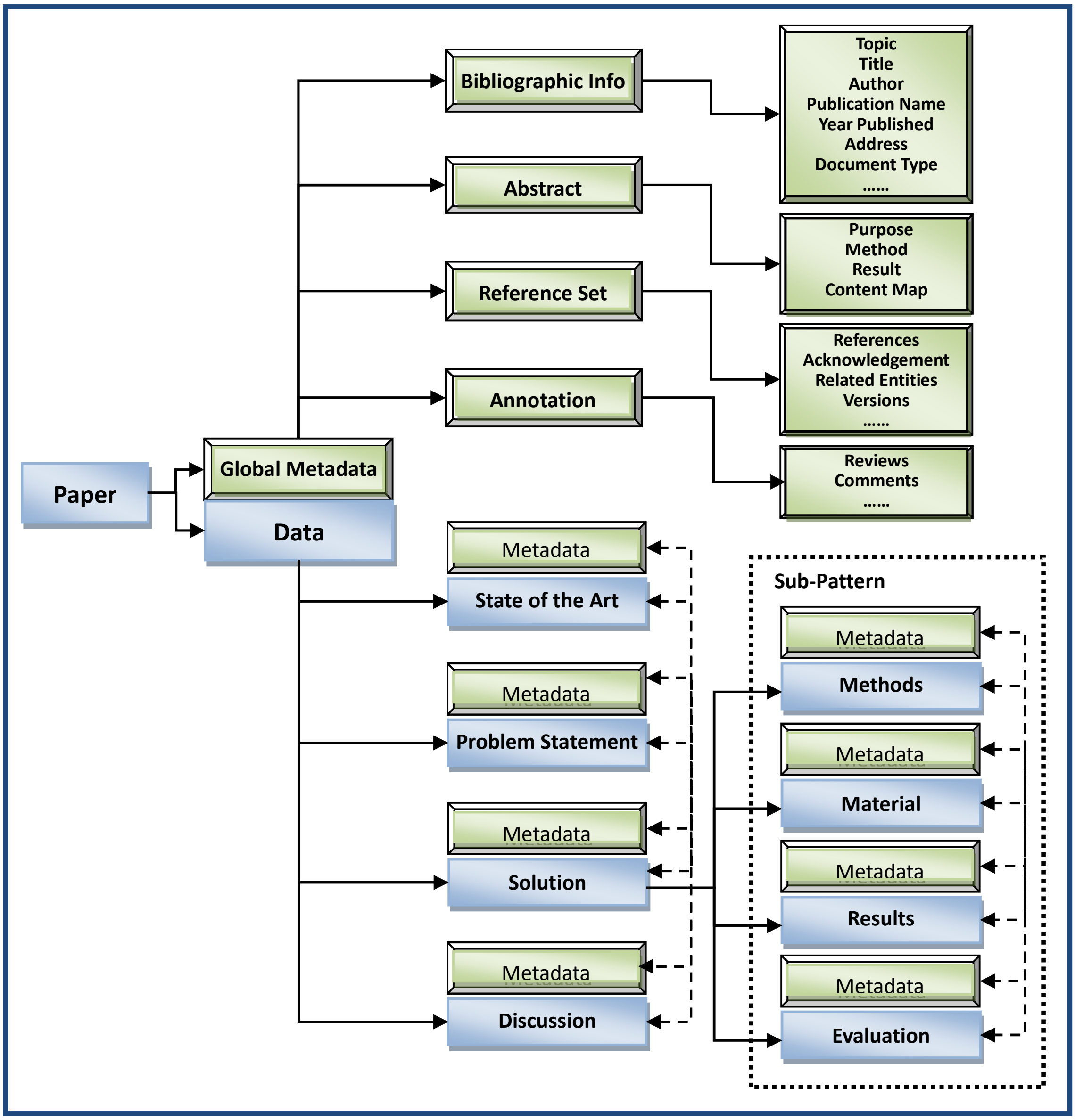


Figure 4.3 SKO Pattern

Metadata

- Bibliographical Information: Topic, Title, Author/Editor (Name, Affiliation, Email), Keywords, Category, Source (Journal, Conference, Inproceedings, Inbook, Article, Thesis, Techreport, Misc, Other), Publisher, Year, Volume, Number, Pages, Series, Edition, Month, Document Type, and so on.

- Abstract: a brief description of the paper including Purpose, Method, Result, and Content Map.
- Reference Set: A set of referenced entities, such as a list of “References”, Persons and Projects mentioned in “Related Work”, and “Acknowledgement”, a set of URLs or other entities in the Footnotes and Endnotes, and so on.
- Annotation: Comment, Review, Tag, and so on.

Data

- State of the Art: Observations of phenomena, situations, foundational theories, and related work where the contextualized scientific problem is addressed.
- Problem Statement: The description and an active challenge faced by researchers which the discourse aims to solve.
- Methods: The specific techniques or methodology used in conducting a particular experiment.
- Material: Data collection, pretreatment, and analysis.
- Results: The outcome or the findings of the research.
- Evaluation: The evaluation methodology and its associated results.
- Discussion: Comparison of the results with related solutions or observations.

SKO Patterns provide a semantic approach for scientific discourse representation. Rhetorical blocks constitute the composition of metadata and data of discourse. Essentially, these rhetorical blocks are unordered – they always have types of relations between each other instead of a linear order. Examples of such relations include explanation relations, argumentation relations, and so on. It is impossible to convince researchers to follow a uniform structure for writing various types of publications. However, there always are some sequential relations among the rhetorical blocks. For instance, we commonly address the problem first and find the solution next as a problem-solving scientific method. To find the solution, we need to collect data, carry out the experiment, and obtain the results. The further sequential relations (orders) of rhetorical blocks, which are based on three strategies of logical reasoning, will be discussed in the following subsection, Rationale.

4.4.5 Rationale

The Rhetorical Blocks are derived from general scientific methods and three fundamental logical reasoning methods (Deduction, Induction, and Abduction).

The SKO Patterns are constituted by unordered rhetorical blocks with links through semantic metadata and relations. In this subsection, we sequentially discuss the rationale and some possible solutions for ordering these atomic rhetorical blocks in an intuitive way for both writing and reading.

We derive three fundamental patterns for serialization of scientific discourse from the three basic types of logical reasoning method, that is, Deduction, Induction, and Abduction. A logical reasoning contains three elements for inferences, that is, Precondition, Rule, and Conclusion.

$$\text{Precondition} \xrightarrow{\text{Rule}} \text{Conclusion}$$

- Deduction is a process of applying the Rule to the Precondition and determining the Conclusion. For example, "When it rains, the road gets wet” is the Rule. “It rains” is the Precondition. Then we can deduce the Conclusion “The road is wet". Mathematicians are commonly associated with this style of reasoning.

- Induction is using the Precondition and Conclusion to find the Rule that can explain the transition, for example, "The road has been wet every time it has rained. Therefore, when it rains, the road gets wet". Scientists are commonly associated with this style of reasoning.
- Abduction is using the Rule and the Conclusion to support the proposition that the Precondition could explain the Conclusion, for example, "When it rains, the road gets wet. The road is wet; therefore, it may have rained". Diagnosticians and detectives are commonly associated with this style of reasoning.

In practice, when we do research and write a paper, problems always have to be solved by steps (states). We take a deduction as an instance:

We start from State 0 (S_0) as the Precondition and Theory 0 (T_0) as the Rule. Using T_1 and S_0 we may deduce S_1 as the intermediate Conclusion, while the rest may be deduced by analogy. So we can reach the Final State (S_F) as the Conclusion.

$$T_0, S_0 \xrightarrow{T_1, S_0} S_1 \xrightarrow{T_2, S_1} S_2 \ldots \ldots \xrightarrow{T_i, S_{i-1}} S_i \ldots \ldots \xrightarrow{T_F, S_{F-1}} S_F$$

During these reasoning periods, we also need to make the Observation, formulate the Hypothesis, and conduct the Experimentation for obtaining and validating the related States and Theories. In the following subsections, we propose three rhetorical structure patterns according to the three logical reasoning methods.

- **Deduction**

The Deductive Method (Figure 4.4) works from a general rule or principle to a specific solution. (1) Theory and Observation: the method begins with a theory and observation of our interest. (2) Hypothesis: we then narrow them down to a specific hypothesis that may solve the problem we face. (3) Experimentation: we narrow it down further to test the hypothesis by specific experimentation. (4) Conclusion: a conclusion follows logically from the available theory and observations.

Deductive Pattern

1. State of the Art: Observe S0, T0, set i = 1;

Investigate existing Theories and Observations. Related phenomena, development, and analysis construct the Initial State (S0). Selected theories and techniques will support inference and argumentation as T0.

2. Problem Statement: Hypothesis SF, state the problem P = |SF| – |Si-1|;

Predict a Target State SF as a hypothesis for further testing and confirmation. The problem statement presents the gap between SF and Si-1.

3. Methods: Propose Ti such that |Ti| > |Ti-1|;

This is the method of designing, refining, or applying a Theory Ti, which leads Si-1 $\longrightarrow$ Si. The method could be an experimental, numerical, or theoretical method, for example.

4. Material: Compute Si = Ti (Si-1);

The material includes all the raw data, intermediary data, and pretreated data collected from the State of the Art that are used for

Experimentation by the proposed Method.

5. Evaluation: Evaluate Si. if (|SF| – |Si| > ε) i = i + 1, go to (2) ;

Compare Si with SF. If Si does not satisfy the expectations, repeat the loop 3–4–5 with the modifications of Theories until the ideal Si is obtained. Here some new problem may arise during the whole loop 3–4–5. If this happens, go to 2, making a new sub-problem statement and continue in recursion. When Si is (approximately) equal to SF, then break and go on to the next step, 6.

6. Results: SF = Si;

Present Final State SF.

7. Discussion: Discuss SF and |SF| – |S0|;

Compare SF and S0 with related observations and findings from other scientists, always together with an old theory which is confirmed or applied within a new context.

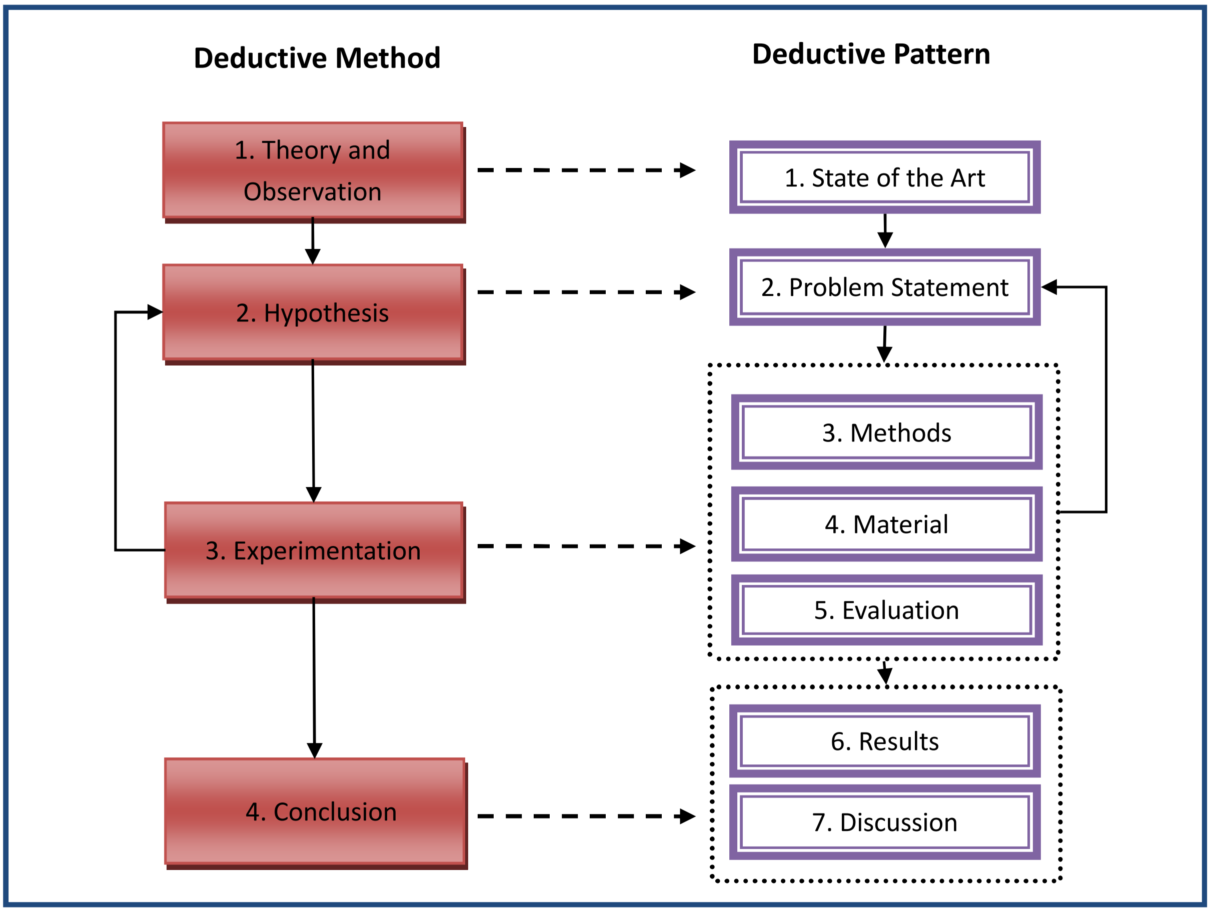


Figure 4.4 Deductive Method and Deductive Pattern

- **Induction**

The Inductive Method works from specific observations towards general theories and principles. (1) Observation: we begin with specific observations. (2) Hypothesis: we then formulate a generalized hypothesis to explore. (3) Experimentation: we detect the patterns and regularities via various measures and experimentations. (4) Theory: finally, we develop some general theories.

Inductive Pattern

1. State of the Art: Observe T0, S0, SF, i = 1;

Investigate existing Observations along with their theoretical explanations, and set them as T0, S0, SF.

2. Problem Statement: Hypothesis TF, P = |TF| – |T0|;

Pose some phenomena as a Final State SF which cannot be explained by existing theories or described by existing models. The problem statement aims at finding a Theory TF which possibly implies that S0 ⟶ SF.

3. Discussion: Discuss Property (SF) and |SF| – |Si-1|;

Observe and analyse the specific phenomena and particular scenario in Si-1 and SF. Generalize and patternize a more general solution for a series of separate problems.

4. Methods: Propose Ti such that |Ti| > |Ti-1|;

The scientific methodology, logic, or philosophical approach for deriving a Theory from transmission Si-1⟶Si.

5. Material: Compute Si = Ti (Si-1);

Evidences, intermediate data, observations, and so on which support analysis and evaluation via the proposed Method.

6. Evaluation: Evaluate Si. if (Si != SF) i = i +1, go to (3);

Compare Si with St. Repeat the loop 3–4–5–6 with modifications of Ti until the ideal Theory is obtained.

7. Results: TF = Ti;

A new theory TF is proposed.

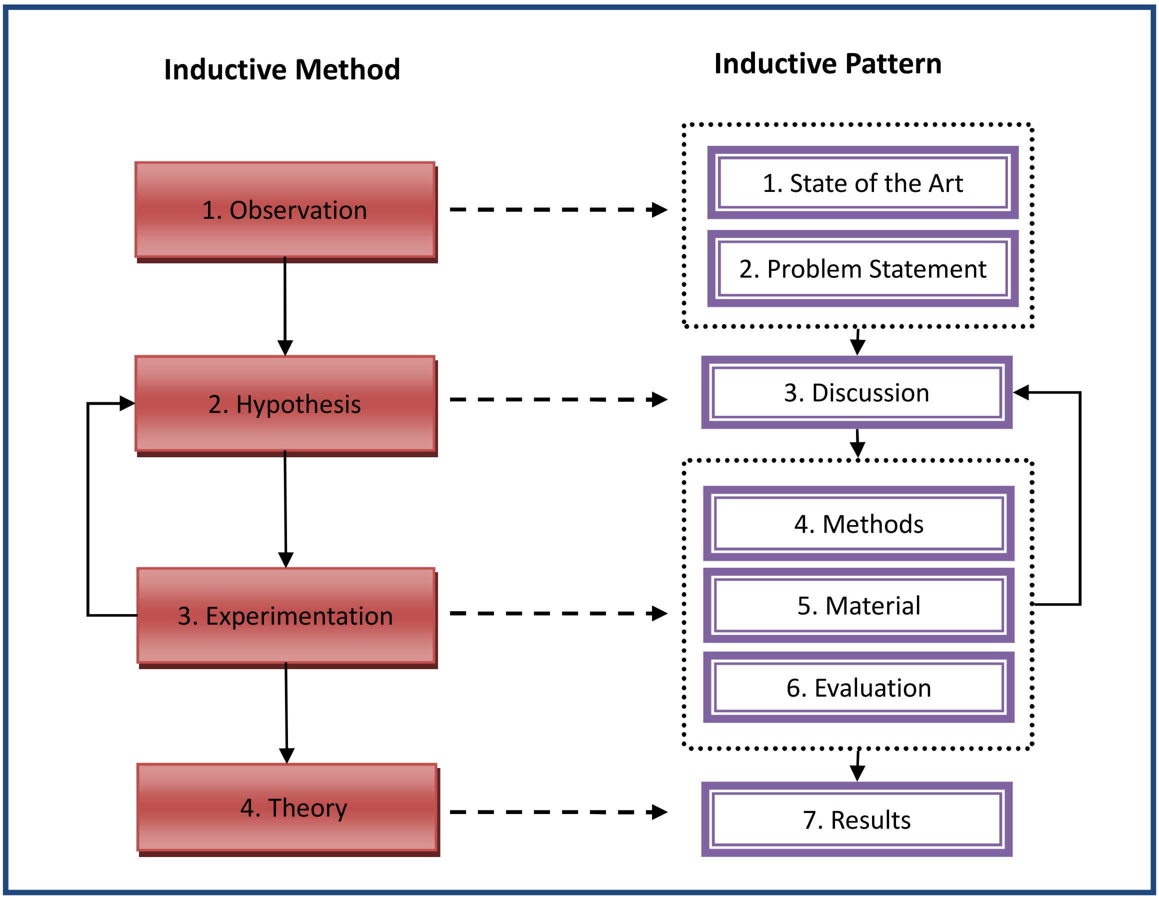


Figure 4.5 Inductive Method and Inductive Pattern

- **Abduction**

The Abductive Method is the process of inference that produces a hypothesis as its end result. (1) Observation: observe a set of seemingly unrelated facts, armed with an intuition that they are somehow connected. (2) Theory: move then to the related theories or principles that may explain some features of facts. (3) Experimentation: infer a possible precondition as an explanation of observable facts judging by existing theories. (4) Hypothesis: a hypothesis is detected.

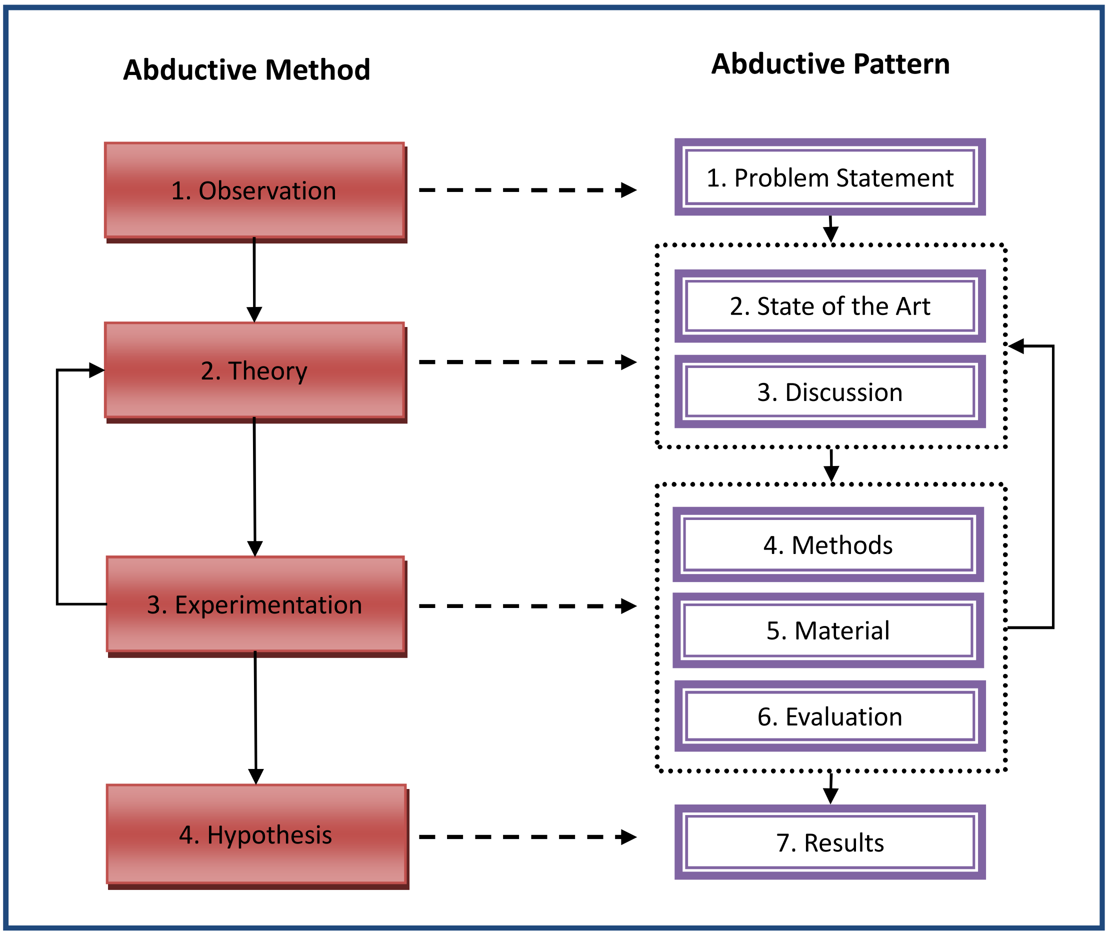


Figure 4.6 Abductive Method and Abductive Pattern

Abductive Pattern

1. Problem Statement: Pose a problem in order to derive explanations E of observations O according to theories T, namely

(1) $T \cup E \models O$ and

(2) $T \cup E$ is consistent.

2. State of the Art: Investigate related observations, phenomena, and facts, and set them as the Final State SF.

3. Discussion: Observe and analyse the set of seemingly unrelated facts

and discuss various possibilities whereby an Initial State Si could be an explanation of SF, where

Si $\longrightarrow$ SF.

4. Methods: The way in which Si is derived, for example, enumerative method, exclusive method, and so on.

5. Material: Evidences, facts, observations, and so on which support analysis and backtracking according to the existing Rule.

6. Evaluation: Compare T(Si) with St. Repeat the loop 2–3–4–5–6 with the modifications of methods and replacement of rules until the ideal Si is obtained.

7. Results: Phenomena detection or theory generation, development, or appraisal.

4.4.6 Benefits

- Rhetorical structured papers facilitate strategic reading.
- Rhetorical blocks enhance the discoverability of elementary knowledge within the context.
- Metadata and other annotated semantic information enable linking of scholarly literature with research data.
- SKO Patterns can be employed in various platforms or services, such as publishing workflow tools, semantic web tools, metadata exchange, social networks, linked data, and authoring and reviewing tools.
- SKO Patterns are compatible with other prominent scientific annotation ontologies.

4.4.7 Liabilities

- High cost of metadata generation.
- High cost of metadata maintenance.

4.4.8 Examples

Deduction

- Automated composition of Web services via planning in asynchronous domains. Piergiorgio Bertoli, Marco Pistore, Paolo Traverso. Source: Artificial Intelligence 174 (2010) 316–361 [97]

  Abstract
  1. Introduction
  2. The problem
  3. Processes as state transition systems
  4. Modeling the composition problem
  5. The synthesis algorithms
  6. Experimental evaluation
  7. Related work
  8. Conclusions
  References

- Model Checking Syllabi and Student Careers, Roberto Sebastiani, Alessandro Tomasi, Fausto Giunchiglia. TACAS2001, Tools and Algorithms for the Construction and Analysis of Systems, Genova, Italy, April 2001. LNCS , N. 2031, Springer [98]

  Abstract

1. Motivations and goals
2. The Problem
3. Formalization into Model Checking
4. A prototype implementation
5. Preliminary empirical results
6. Ongoing and future work

References

Induction

- Sampling community structure. Arun S. Maiya and Tanya Y. Berger-Wolf. Source: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010: 701-710 [99]

  Abstract
  1. Introduction and Motivation
  2. Related Work
  3. Preliminaries
  4. Proposed Method
  5. Experimental Evaluation
  6. Conclusion
  7. Acknowledgement
  8. References

- Local models semantics, or contextual reasoning = locality + compatibility. Chiara Ghidini, Fausto Giunchiglia. Source: Artificial intelligence 127, 2001: 221-259 [100]

  Abstract
  1. Introduction

2. Two examples
3. Local Models Semantics
4. The two examples - model theory
5. The proof theory: MC systems
6. The two examples - proof theory
7. Other frameworks - a comparison
8. Conclusion
References

Abduction

- Hypermedia and the Semantic Web: A Research Agenda. Jacco van Ossenbruggen, Lynda Hardman and Lloyd Rutledge. Source: Journal of Digital information, volume 3 issue 1 [101]

  Abstract
  1. Introduction
  2. Current Semantic Web Infrastructure
  3. Relation with Hypermedia Research
  4. Open Research Questions
  5. Conclusion
  Acknowledgements
  References

- Web Service Composition - Current Solutions and Open Problems. Biplav Srivastava and Jana Koehler. Source: In: ICAPS 2003 Workshop on Planning for Web Services, 2003, 28-35 [102]

  Abstract
  1. Introduction
  2. An Example Scenario

3. Web Services
4. Modeling Flow Composition
5. Related Work
6. Conclusion and Future Work
References

4.5 Discussion

In this chapter, we propose Scientific Knowledge Object Patterns for solving problems of explicit representation in terms of the semantics of scientific discourse. The patterns mainly serve in the semantic layer of SKOs, and three possible serialization patterns derived from logical reasoning -deduction, induction and abduction - have also been discussed.

Currently we are initiating a project entitled “Conference of the Future” (see Chapter 6) which will be the first comprehensive scientific publishing platform equipped with SKO Patterns, along with metadata schemes. Our ultimate goal is to provide a high-level pattern language for the externalization of the rhetoric and argumentation captured within Scientific Knowledge Objects such as papers, which will facilitate discovery, dissemination, and the reuse of scientific knowledge in research communities.

As exhibited in Table 4.2, comparing a Typical Pattern with an SKO Pattern indicates that the latter provides more metadata support and a more flexible rhetorical representation structure as an alternative.

Typical Pattern	SKO Pattern
Title	GlobalMetadata. BibliographicInfo. Title
Author	GlobalMetadata – BibliographicInfo. Author
Abstract	GlobalMetadata. Abstract
Keywords	GlobalMetadata. BibliographicInfo. Topic
Introduction	GlobalMetadata. Abstract. ContentMap

	State of the Art	
	Problem Statement	
Methods	Solution	Methods
		Material
Results		Results
		Evaluation
Discussion	Discussion	
Acknowledgement	GlobalMetadata. ReferenceSet. Acknowledgement	
References	GlobalMetadata. ReferenceSet. References	

Table 4.2 Comparison between Typical Pattern and SKO Pattern

From Table 4.3, we can see clearly the comparison of functionalities between the Typical Pattern and the SKO Pattern. In the Typical Pattern, a scientific paper is composed of sections, subsections, paragraphs, and sentences, while in the SKO Pattern the basic content unit is an SKOnode (rhetorical block). Concerning the structuring, a Typical Pattern follows a linear structure, including components such as sections, paragraphs and sentences that are ordered as a one way linked list. SKO Patterns adopt the representing way of rhetorical structure, while SKOnodes are linked via various relationships. With respect to metadata support, the Typical Pattern provides only shallow metadata in the paper header part, while the SKO Pattern supplies both global metadata describing the whole SKO, and local metadata that depicts the SKOnode. Furthermore, in the SKO Pattern, we define three sub-patterns enabling strategic reading and writing, i.e. deduction, induction and abduction. SKO Patterns are also extendable, customized, configurable and interoperable in terms of importing other ontologies.

	Typical Pattern	**SKO Pattern**
Component	Section, Subsection, Paragraph, Sentence	SKO node as a basic functional unit (for authorship, citation, search, reuse, even copyright)
Content Organization	Linear structure	Rhetorical Structure
Metadata Support	Few metadata in	Global Metadata and

	paper header	Local Metadata
Strategic Reading/Writing	No	Yes Deduction, Induction, Abduction
Ontology-Based	No	Yes

Table 4.3 Functionality Comparison between Typical Pattern and SKO Pattern

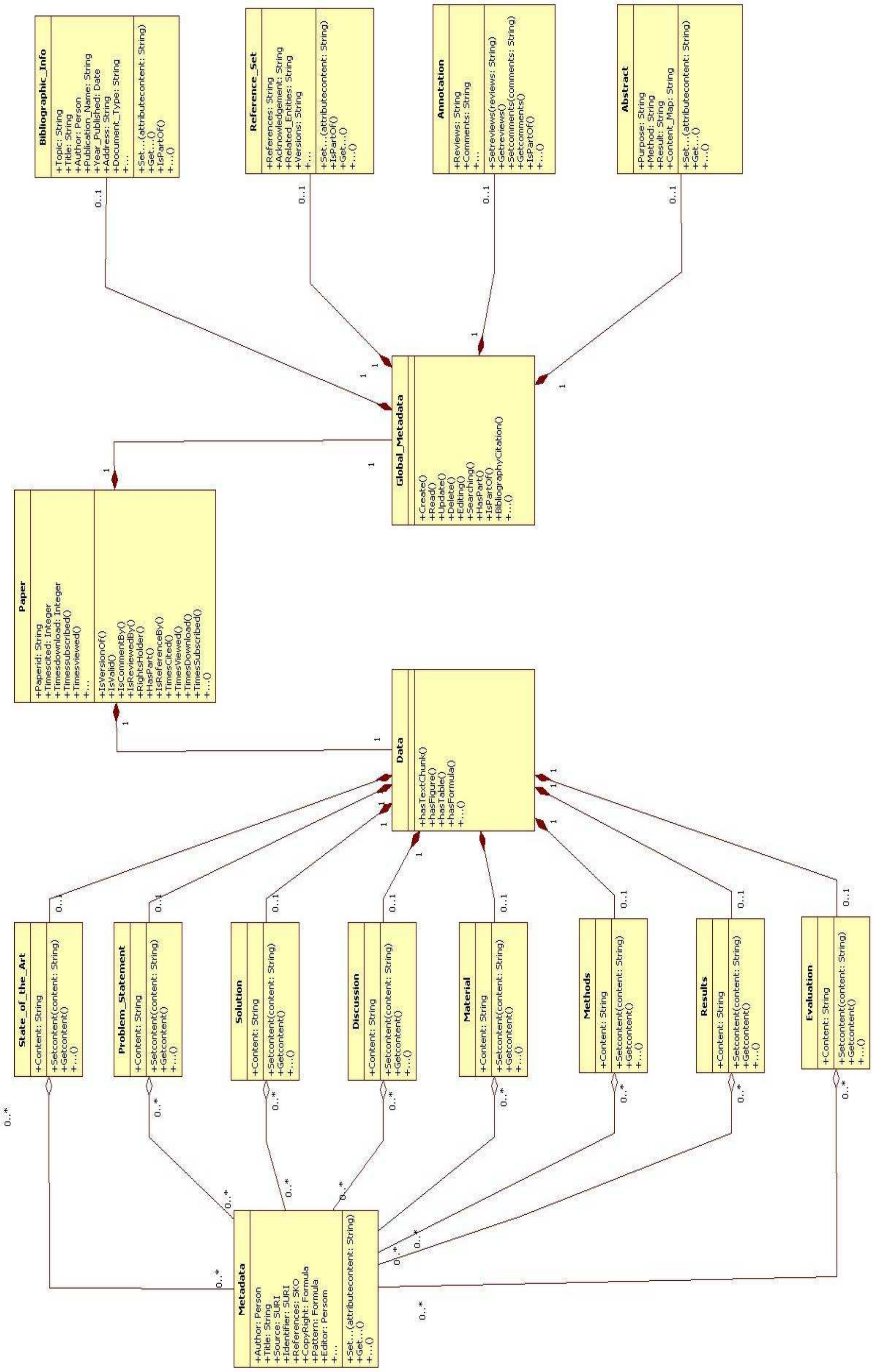


Figure 4.7 E-R Diagram for SKO Patterns

Chapter 5

SKO TeX

Reading and writing scientific articles are integral components of daily scientific activity. The most practical task is that we need to search, browse, peruse and digest useful knowledge for our research; meanwhile, we are also trying to publish what we have found, and to share it with others. In such a course of events, we need to read fluently and effectively. For example, we may wish to retrieve items or data more precisely and find articles with valuable information in an efficient manner. As part of this, we hope to identify the relevant references as well as opinions and comments of others more speedily. Nowadays, the traditional scientific publishing model, which is to download PDF articles into personal computers or iPads to read, is apparently far behind peoples' expectations. In particular, this is not co-developing with the existing Web 2.0 and Semantic Web technological development.

We believe that semantic annotation will undermine the traditional way of reading and knowledge dissemination. People can obtain knowledge from simple PDF files. This is either from more detailed supplementary information provided by the author, such as a data set applied to the experiment and the program codes, which cannot be completely included in the articles even though they very important for readers to understand, or from views and comments by the readers after they have finishing reading. Such comments and suggestions may support or question parts of the articles, but can also provide more clues and thinking space for readers.

In Chapter 3 and Chapter 4, we respectively defined an entity oriented metadata schema and rhetorical structure patterns for scientific discourse representation, namely SKO Types and SKO Patterns, which constitute the theoretical foundations of SKO management. In this chapter, we introduce the SKOTeX, whose name is derived from LaTeX and BibTeX, respectively an editing tool which enables users to generate semantic enriched documentation, and a file format that specifies sets of annotating commands and storage forms similar to those used in LaTeX and BibTeX.

SKO TeX can be applied in such a way as to take charge of the whole lifecycle management of SKOs, which includes the establishment, release, and annotation, re-use, and so on. At the present stage of implementation, we are developing and defining SKO TeX for the purposes of the IJCAI project. These include:

- several macro packages which are used to define different kinds of semantic annotating commands;
- a processor, which is applied to process the initial .tex files and interact with internet databases or local database files, according to the annotating commands from authors or readers. It can also be used to extract the correlative entities or metadata information and generate extended .tex files containing more semantic information. All these .tex files can be translated, edited and compiled using

common LaTeX tools.

- an improved traditional LaTeX reference management tool BibTeX. The administration of bibliographies can be extended to SKO-related entities management by SKOTeX, for instance, author, project, conference, SKOnode and so on. In other words, using SKOTeX, we may cite entities similar to references in a traditional LaTeX environment.

Of course, the entity citing mentioned above may include all properties such as attributes, relationships, etc., that can be automatically obtained through the use of the SKO TeX processor. This can help to realize the semantic annotations and perfect the semantic editing environment.

This chapter is organized as the following:

In Section 5.1 we define sets of entries for SKO TeX, based on BibTeX and SKO Types.

Section 5.2 presents cases showing how SKO TeX facilitates the authoring and annotating of semantic publications.

Section 5.3 describes the implementation of SKO TeX for IJCAI72.

5.1 SKO TeX Entries

As introduced in Section 2.1.3, BibTeX is a tool and a file format which is used to describe and process lists of references, mostly in conjunction with LaTeX documents[73]. It mainly consists of a set of files as follows:

[72]IJCAI- International Joint Conference on Artificial Intelligence: http://ijcai.org/
[73]BibTeX: http://www.bibtex.org/

- A .bib file is a database that stores all reference entries that authors might cite. This file is always maintained by the author.
- A .bst file specifies the presentation style with regard to references, and defines the format of individual entries. This file is commonly provided by the publisher.
- Other intermediate files such as .aux and .bbl files.

In SKO TeX, we extend BibTeX to support more types of entries for citation and annotation. For instance, in BibTeX, it only defines part of SKO. Table 5.1 presents all predefined entry types in SKOTeX, and also makes a comparison between BibTeX and SKOTypes.

BibTeX	SKOTeX	SKOTypes
Article/ Inproceedings/ Incollection/ Inbook/ Misc/ Unpublished	Article	SKO
Book	Book	
Booklet	Booklet	
Proceedings,	Proceedings	
Journal	JournalIssue	
Masterthesis/ Phdthesis,	Thesis	
Techreport/ Manual	TechReport	
	ArticleCollections	
	Comment	
	Review	
	LiquidJournal, ConferenceCallforPapers Topics, Categories	SKOset
	Chapter, Section, Paragraph, Sentence, Figure, Formula, Table, StateoftheArt, ProblemStatement, Solution, Discussion, Methods, Material, Results, Evaluation	SKOnode
	Author, Editor	Person

	Institution, Publisher, Conference-Organizer	Organization
	Conference, Project	Event
	Location	Location

Table 5.1 Entries Types for SKOTeX

In SKOTeX, there are 37 types of predefined entries that can be mapped to 7 SKO types. In contrast to BibTeX, SKOTeX extends the capabilities of storage and can process more types of entities that are defined in SKOTypes.

Each SKOTeX entry is specified by *Type Name*, *Description*, *Required fields*, *Optional fields*, and an *Example* in accordance with BibTeX. Some of the fields (values) in the entries are marked in blue, which means that those fields are *Relational* fields. More specifically, those field values are *Entities* instead of *String*s. In practice, when an author creates/modifies the .sko.bib files, s/he can simply input the entity name to the SKOTeX entry, and the SKOTeX processor will convert these entity names to SURLs or AISN-IDs during the compiling phase. Alternatively, authors are also encouraged to use SURLs and AISN-IDs directly when they compose their .sko.bib files. Note that a tag's name in the SKOTeX file is NOT case-sensitive. We give several examples of entry definitions herein, which may be used in the following subsections.

Authors may create and manage the SKOTeX entries either via a simple text file, or by using customized off-the-shelf BibTeX tools such as JabRef(see Figure 5.1).

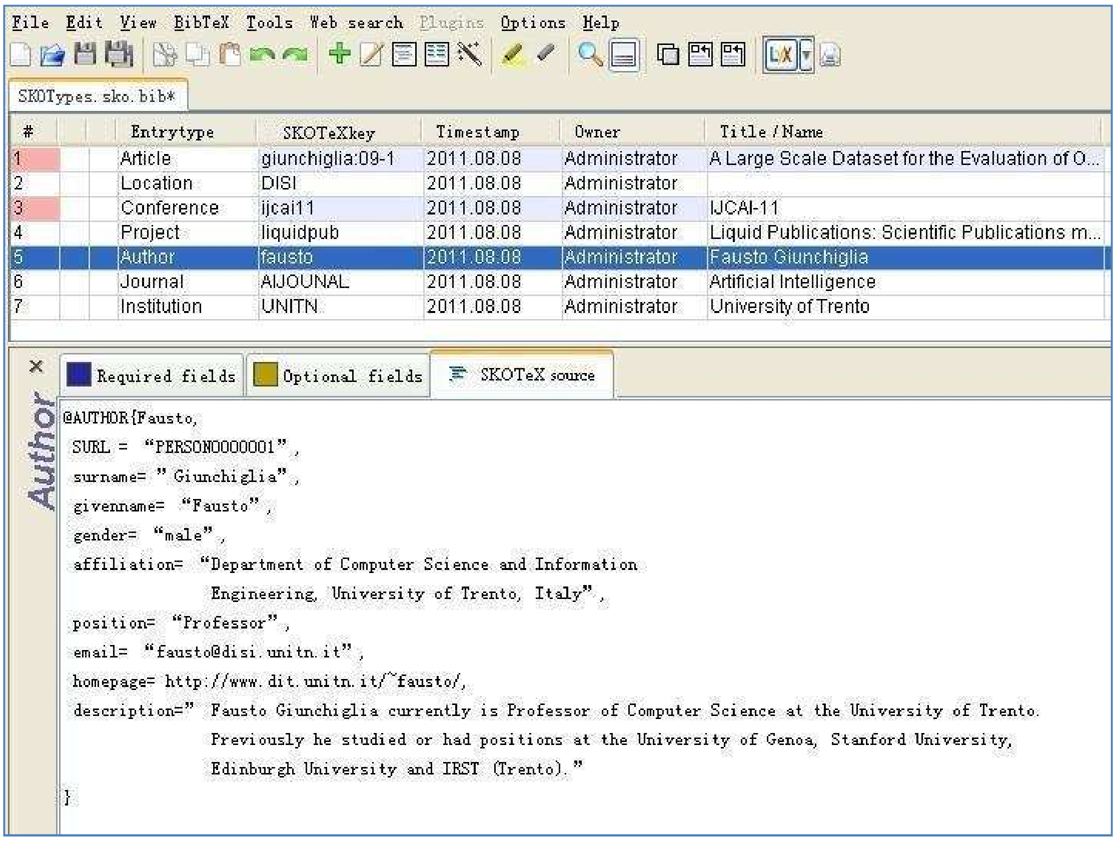


Figure 5.1 SKOTeX Entries Management in JabRef

5.2 Use Cases

Although SKO TeX is not restricted to processing LaTeX/BibTeX source files, we believe that the LaTeX-like commands for citing and annotating is ideal, or at least comparatively easy way, for SKO TeX users to adapt. Also, SKO TeX can seamlessly process normal LaTeX/ BibTeX files and generate semantic documentations.

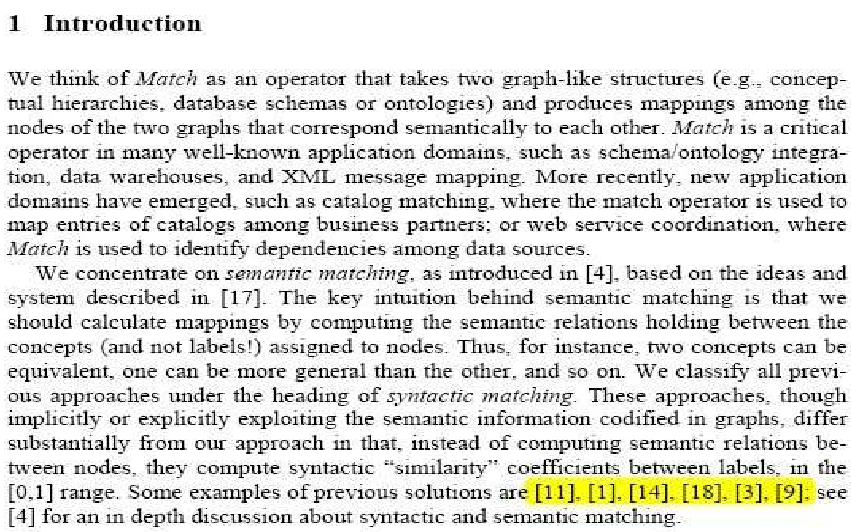
5.2.1 Cite Article Collections

When we write a paper, we invariably cite a handful of works which together describe a certain topic. For example, as shown in Figure 5.2, the authors enumerated a set of references as previous work on *syntactic matching*.

In the LaTeX editing environment, we need to cite these references as follows:

> *…Some examples of previous solutions are /cite{cupid}, /cite{SIGMOD}, /cite{similarityFlooding}, /cite{domainOntology}, /cite{mapOntologies}, /cite{schemaMatching}; see /cite{contextualReasoning} for an in depth discussion about syntactic and semantic matching.*

However, the ideal solution is that an author may cite these articles as a whole, using shorter commands, especially when this article collection can be easily maintained, updated, and can even be retrieved automatically by some simple queries.



1 Introduction

We think of *Match* as an operator that takes two graph-like structures (e.g., conceptual hierarchies, database schemas or ontologies) and produces mappings among the nodes of the two graphs that correspond semantically to each other. *Match* is a critical operator in many well-known application domains, such as schema/ontology integration, data warehouses, and XML message mapping. More recently, new application domains have emerged, such as catalog matching, where the match operator is used to map entries of catalogs among business partners; or web service coordination, where *Match* is used to identify dependencies among data sources.

We concentrate on *semantic matching*, as introduced in [4], based on the ideas and system described in [17]. The key intuition behind semantic matching is that we should calculate mappings by computing the semantic relations holding between the concepts (and not labels!) assigned to nodes. Thus, for instance, two concepts can be equivalent, one can be more general than the other, and so on. We classify all previous approaches under the heading of *syntactic matching*. These approaches, though implicitly or explicitly exploiting the semantic information codified in graphs, differ substantially from our approach in that, instead of computing semantic relations between nodes, they compute syntactic "similarity" coefficients between labels, in the [0,1] range. Some examples of previous solutions are [11], [1], [14], [18], [3], [9]; see [4] for an in depth discussion about syntactic and semantic matching.

Figure 5.2 Cite Article Collections

In SKO TeX, the citation commands can be shortened as follows:

(Note that *syntacticMatching* is an article collection of [11,1,14,18,3,9])

> *…Some examples of previous solutions are /cite{syntacticMatching}; see /cite{contextualReasoning} for an in depth discussion about syntactic and semantic matching.*

5.2.2 Cite Authors

S-Match: an Algorithm and an Implementation of Semantic Matching

Fausto Giunchiglia, Pavel Shvaiko, Mikalai Yatskevich

Dept. of Information and Communication Technology
University of Trento,
38050 Povo, Trento, Italy
{fausto, pavel, yatskevi}@dit.unitn.it

Figure 5.3 Cite Authors

Every time we write a scientific paper, we are obliged to supply information about the author(s). Basically, it always contains names, affiliations, addresses, and the emails of the authors. Actually, all this information are the attributes of the authors. Specifically, an address is not the address of an author, but of an institution or organization. Although this kind of authoring involves neither a great deal of time or thought, in SKO TeX we can retrieve this information automatically from a linked database file, e.g. a .bib file, using a simple command.

We can compare the LaTeX source and the SKO TeX source for writing paper headers as shown in Figure 5.3.

LaTeX

\title{S-Match: an Algorithm and an Implementation of Semantic Matching}

\author{FaustoGiunchiglia, PavelShvaiko, MikalaiYatskevich }

\institute{Dept. of Information and Communication Technology

University of Trento,

38050 Povo, Trento, Italy

\email{\{fausto, pavel, yatskevi\}@dit.unitn.it}}

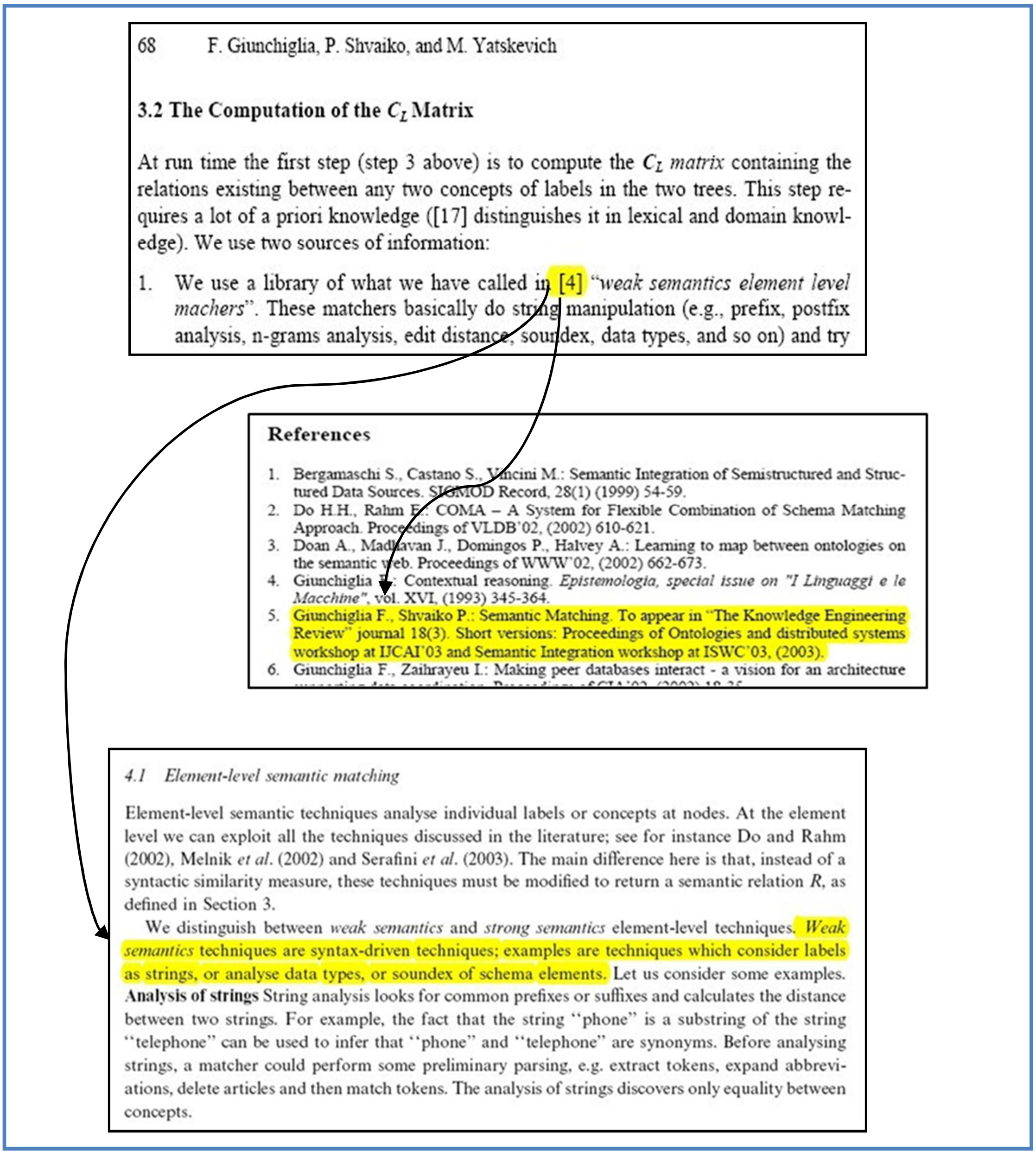
SKOTeX

\title{S-Match: an Algorithm and an Implementation of Semantic Matching}

\author{\citeAuthor{Fausto}, \citeAuthor{Pavel}, \citeAuthor{Mikalai}}

5.2.3 Cite SKOnode

Sometimes we cite a reference in order to recommend the whole paper to readers which may provide more detailed explanations. More frequently, an author may cite references just because segments of the references, e.g. SKOnodes, may be of interest to the readers. Traditionally, LaTeX doesn't provide such a mechanism and functionality. In SKO TeX, a SKOnode can be cited as a normal reference by using the same citation commands.


68 F. Giunchiglia, P. Shvaiko, and M. Yatskevich

3.2 The Computation of the C_L Matrix

At run time the first step (step 3 above) is to compute the C_L *matrix* containing the relations existing between any two concepts of labels in the two trees. This step requires a lot of a priori knowledge ([17] distinguishes it in lexical and domain knowledge). We use two sources of information:

1. We use a library of what we have called in [4] "*weak semantics element level machers*". These matchers basically do string manipulation (e.g., prefix, postfix analysis, n-grams analysis, edit distance, soundex, data types, and so on) and try

References

1. Bergamaschi S., Castano S., Vincini M.: Semantic Integration of Semistructured and Structured Data Sources. SIGMOD Record, 28(1) (1999) 54-59.
2. Do H.H., Rahm E.: COMA – A System for Flexible Combination of Schema Matching Approach. Proceedings of VLDB'02, (2002) 610-621.
3. Doan A., Madhavan J., Domingos P., Halvey A.: Learning to map between ontologies on the semantic web. Proceedings of WWW'02, (2002) 662-673.
4. Giunchiglia F.: Contextual reasoning. *Epistemologia, special issue on "I Linguaggi e le Macchine"*, vol. XVI, (1993) 345-364.
5. Giunchiglia F., Shvaiko P.: Semantic Matching. To appear in "The Knowledge Engineering Review" journal 18(3). Short versions: Proceedings of Ontologies and distributed systems workshop at IJCAI'03 and Semantic Integration workshop at ISWC'03, (2003).
6. Giunchiglia F., Zaihrayeu I.: Making peer databases interact - a vision for an architecture

4.1 Element-level semantic matching

Element-level semantic techniques analyse individual labels or concepts at nodes. At the element level we can exploit all the techniques discussed in the literature; see for instance Do and Rahm (2002), Melnik *et al.* (2002) and Serafini *et al.* (2003). The main difference here is that, instead of a syntactic similarity measure, these techniques must be modified to return a semantic relation *R*, as defined in Section 3.

We distinguish between *weak semantics* and *strong semantics* element-level techniques. *Weak semantics* techniques are syntax-driven techniques; examples are techniques which consider labels as strings, or analyse data types, or soundex of schema elements. Let us consider some examples.

Analysis of strings String analysis looks for common prefixes or suffixes and calculates the distance between two strings. For example, the fact that the string ''phone'' is a substring of the string ''telephone'' can be used to infer that ''phone'' and ''telephone'' are synonyms. Before analysing strings, a matcher could perform some preliminary parsing, e.g. extract tokens, expand abbreviations, delete articles and then match tokens. The analysis of strings discovers only equality between concepts.

Figure 5.4 Cite SKOnode

5.2.4 Cite Dataset

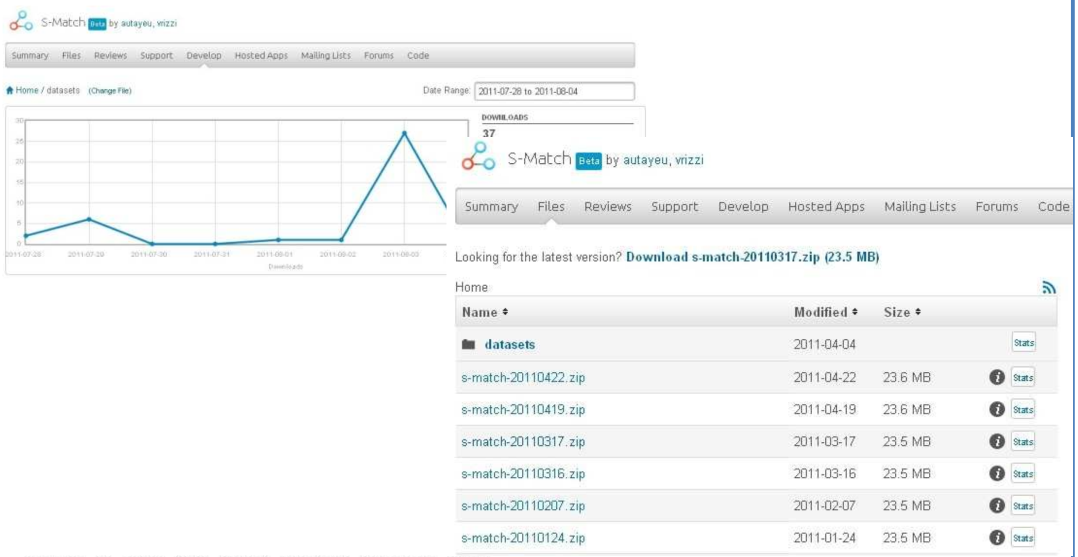
It is always not possible to publish, at least in a paper per se, all datasets or programming codes used in the research that is presented in the article. However, these may be essential for reader to understand, digest, reuse, and compare the work. A simple hyperlink may solve this problem, while some dynamic analysis may definitely help the reader also.

3 The Algorithm

Let us introduce some notation (see also Figure 1). Nodes are associated a number and a label. Numbers are the unique identifiers of nodes, while labels are used to identify concepts useful for classification purposes. Finally, we use "*C*" for concepts of nodes and labels. Thus, C_{Europe} and "C_2 in A1" are, respectively, the concept of label *Europe* and the concept of node 2 in A1.

The algorithm is organized in the following four macro steps:

Step 1: for all labels L in the two trees, compute C_L
Step 2: for all nodes N in the two trees, compute C_N
Step 3: for all pairs of labels in the two trees, compute relations among C_L
Step 4: for all pairs of nodes in the two trees, compute relations among C_N




```
package it.unitn.disi.smatch.matchers.structure.tree;

import it.unitn.disi.smatch.components.Configurable;
import it.unitn.disi.smatch.components.ConfigurableException;
import it.unitn.disi.smatch.data.mappings.IMappingFactory;
import it.unitn.disi.smatch.matchers.structure.node.INodeMatcher;
import org.apache.log4j.Logger;

import java.util.Properties;

/**
 * Base class for tree matchers. Needs the following configuration parameters"
 * <p/>
 * nodeMatcher string parameter which should point to a class implementing a
 * {@link it.unitn.disi.smatch.matchers.structure.node.INodeMatcher} interface.
 * <p/>
 * mappingFactory string parameter with a class implementing
 * {@link it.unitn.disi.smatch.data.mappings.IMappingFactory} interface.
 *
 * @author Aliaksandr Autayeu avtaev@gmail.com
 */
public class BaseTreeMatcher extends Configurable {

    private static final Logger log = Logger.getLogger(BaseTreeMatcher.class);

    private static final String NODE_MATCHER_KEY = "nodeMatcher";
    protected INodeMatcher nodeMatcher = null;

    private static final String MAPPING_FACTORY_KEY = "mappingFactory";
    protected IMappingFactory mappingFactory = null;
```

Figure 5.5 Cite Dataset

5.2.5 Rhetorical Structure

We insist that the best way to present a narrative to a computer is to let the author explicitly create a rich semantic structure for the SKO during the writing process. SKO TeX provides a viable way for authoring and annotating semantic documents using SKO Patterns. With SKO TeX, readers can quickly glance through the contribution and skip to the section they are interested in. The writing at syntax level in SKO TeX will be compatible with regular LaTeX commands. And the specific annotation commands are proposed as a mark-up language as follows. All these commands provide the support for creating rhetoric elements, creating implicit and explicit visual annotations, and for inserting arbitrary annotations in SKOs. In fact, semantic annotation creates a bridge between the actual SKO and its metadata.

We propose a pseudo mark-up language in Figure 5.6, which describes a semantic writing and reading environment. Ideally, after annotating an entity like a person or a project, we could get its attributes automatically by the system without another single search. For example, in Fig.5.6 when we click on the Person "Fausto Giunchiglia", the system retrieves his attributes such as "name", "affiliation", "email" and so forth which are predefined in SKO Types. Alternatively, an author may also choose a traditional way of writing as shown in Figure 5.6.

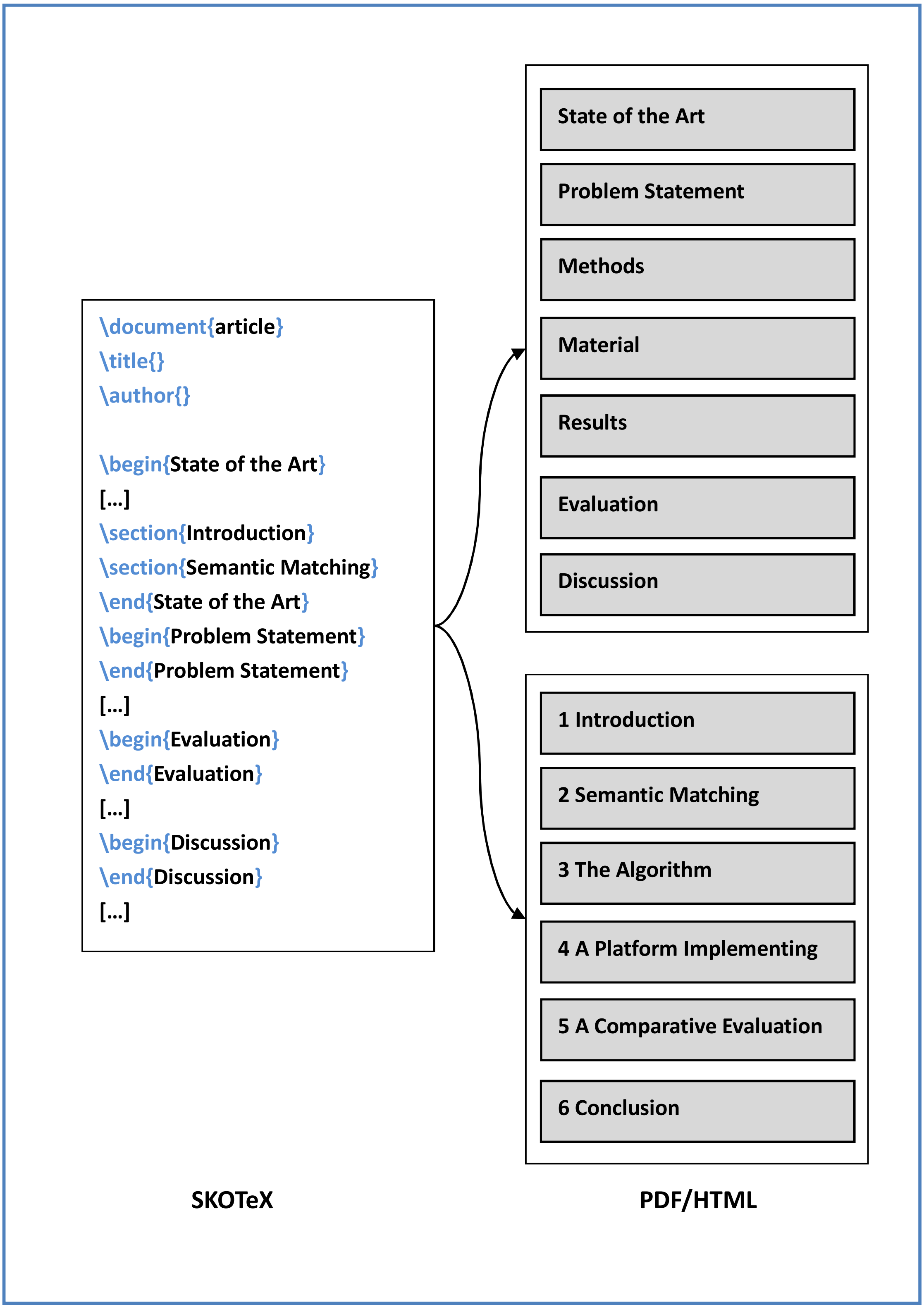


Figure 5.6 Rhetorical Structure

5.3 SKO TeX for IJCAI

We implement a customized SKO TeX for IJCAI as a case study. Currently, we haven't implemented all functionalities, whereas some preliminary development can give us a better idea as to how SKO TeX improves users' experience.

The reasons of choosing IJCAI includes the fact that the AI community is one of the most active communities that is dedicated to improving knowledge mining and dissemination. More importantly, our group is building a social network project for IJCAI, and we have collected and cleaned the metadata from all papers in IJCAI since 1969. All these metadata are being classified and encoded to the SKO Types format, and can be imported or cited directly by SKO TeX without further processing.

SKO TeX provides a set of macros that enable an author to compile and use our customized commands in a normal LaTeX editing environment, generates entity annotations and links enriched files. The process can be done locally or via the internet.

5.3.1 Architecture

The input of SKO TeX are a set of LaTeX source files, e.g. .tex file, .bib files, and style files such as .sty and .bst files. To be specific, a .sty file is always used to define general formats and commands, while a .bst file describes the format of citations and reference entries according to a chosen bibliographic style. In our case, we modified the previous macros[74] provided by IJCAI-2011[75] with our featured functionalities.

74 http://ijcai-11.iiia.csic.es/files/ijcai11.tar
75 http://ijcai-11.iiia.csic.es/

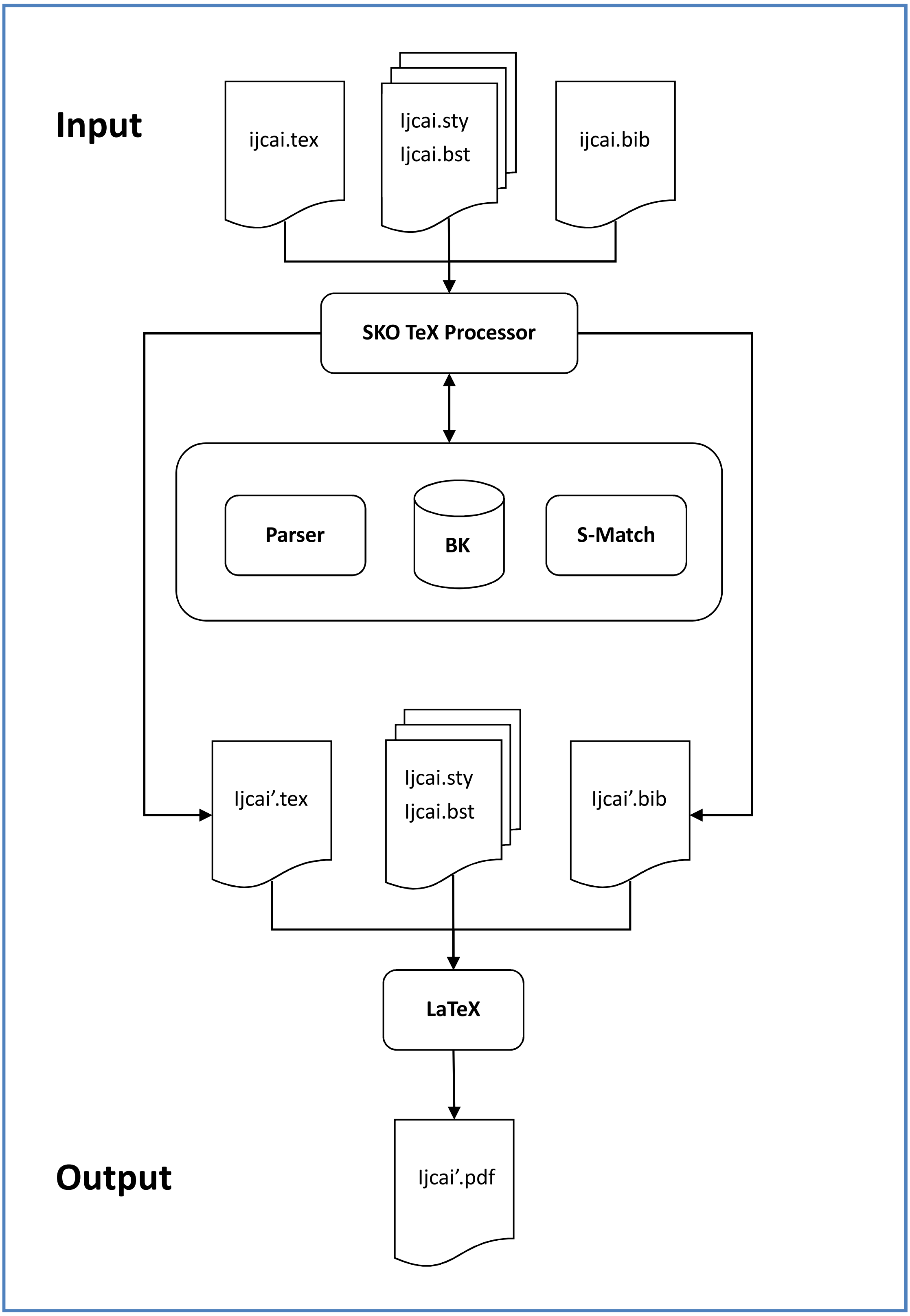


Figure 5.7 Architecture of SKO TeX for IJCAI

Current implementations for the SKO TeX processor is a Java application. In our case, it processes the original files provided by authors, i.e. the ijcai.tex file and the ijcai.bib file, into new files ijcai'.tex and ijcai'.bib. Ideally, the processor should contain a parse that may automatically detect entities embedded in the ijcai.tex, whereas so far we make use of semi-automatically performing semantic functionalities. That is, users need to annotate using our customized commands, and the processor will only perform these focused parts without further parsing. All the information for entities is retrieved from a knowledgebase developed by our group, entitled BK. We are also considering beginning to integrate the S-Match algorithms and applications to the SKO Tex as one of the processor cores.

After processing, two files are generated, i.e. ijcai'.tex and ijcai.bib. We will illustrate and contrast the concrete sources of input and output files in the following sections.

Finally, the files, ijcai'tex and ijcai.bib, are compiled using ordinary LaTeX to create the file, ijcai'.pdf.

5.3.2 Implementation

The procedures associated with processing and compiling are as indicated in the following six steps, where we concurrently exhibit the source files and the generated files.

(1) The author creates the LaTeX source file, ijcai.tex.

To begin with, an author needs to create a .tex file to begin authoring. We take the "IJCAI-11 Formatting Instructions" as an ongoing example in the subsection that follows. Figure 5.8 shows parts of original LaTeX source file excerpted for the "IJCAI-11 Formatting Instructions".

Conventionally, the author needs to input all information that a paper header requires, such as author, affiliation, email, etc.

```
\title{IJCAI--11 Formatting Instructions\thanks{These match the formatting instructions of IJCAI-07.
The support of IJCAI, Inc. is acknowledged.}}
\author{Toby Walsh \\
NICTA and UNSW\\
Sydney, Australia \\
pcchair11@ijcai.org
\And
Fausto Giunchiglia\\
DISI, UNITN\\
Trento,Italy\\
fausto@disi.unitn.it
}

\section*{Acknowledgments}

The preparation of these instructions and the \LaTeX{} and Bib\TeX{} files that implement them was
supported by Schlumberger Palo Alto Research, AT\&T Bell Laboratories, and Morgan Kaufmann
Publishers. Preparation of the Microsoft Word file was supported by IJCAI.  An early version of this
document was created by Shirley Jowell and Peter F. Patel-Schneider.  It was subsequently modified by
Jennifer Ballentine and Thomas Dean, Bernhard Nebel, and Daniel Pagenstecher. These instructions are
the same as the ones for IJCAI--05, prepared by Kurt Steinkraus, Massachusetts Institute of
Technology, Computer Science and Artificial Intelligence Lab.
```

Figure 5.8 Original Tex file of “IJCAI-11 Formatting Instructions”

As was mentioned before, SKO TeX specifies a set of commands that cooperate with the processor to semi-automatically provide annotating services. Benefitting from these mechanisms, an author can reduce the effort involved in some non-scientific tasks. For instance, instead of inputting details of an author, an author may simply use an SKO TeX command “\citeauthor{}” as shown in Figure 5.9. This would call for services from the processor which would retrieve all metadata with regard to the cited author for both content importing and formatting. By the same token, in order to attain a further description or explanation in terms of metadata about an entity, an SKO TeX user may cite entities in the same way as citing references, using commands such as

\citeins{}: cite an institution, e.g. University of Trento

\citeorg{}: cite an organization, e.g. Springer Publisher

\citeperson{}: cite a person, e.g. FaustoGiunchigalia

\citeconf{}: cite a conference, e.g. IJCAI-2011

Figure 5.9 ijcai.tex

```
\title{\citeconf{IJCAI--11} Formatting Instructions\thanks{These match the formatting instructions of
\citeconf{IJCAI-07}. The support of \citeorg{IJCAI, Inc.} is acknowledged.}}
\author{
\ctieauthor{Toby}
\And
\citeauthor{Fausto}
}


\section*{Acknowledgments}

The preparation of these instructions and the \LaTeX{} and Bib\TeX{}
files that implement them was supported by \citeins{Schlumberger Palo Alto
Research}, \citeins{AT\&T Bell Laboratories}, and \citeorg{Morgan Kaufmann Publishers}.
Preparation of the Microsoft Word file was supported by \citeorg{IJCAI}.  An
early version of this document was created by \citeperson{Shirley Jowell} and \citepersion{Peter
F. Patel-Schneider}.  It was subsequently modified by \citeperson{Jennifer
Ballentine} and \citeperson{Thomas Dean}, \citeperson{Bernhard Nebel}, and \citeperson{Daniel
Pagenstecher}.
These instructions are the same as the ones for \citeconf{IJCAI--05}, prepared by
\citeperson{Kurt Steinkraus}, \citeins{Massachusetts Institute of Technology, Computer
Science and Artificial Intelligence Lab}.
```

(2) The author creates the BibTeX source file, ijcai.bib.

It is always the case that a BibTeX file is not disposable, and it can be maintained by authors as a personal favourite. So, when an author composes a paper, s/he either creates a brand new .bib file or adds some entries to an existing .bib file. An SKO TeX user needn't pay for the extra overhead, and simply does it in the same way as in a traditional LaTeX/BibTeX editing environment. Basically, such BibTeX entries can be simply download from various sources such as Citeseer, Google Scholar, etc., and can easily be imported to a ijcai.bib file. The information provided by ijcai.bib helps the processor to semantically identify and match the entities in our knowledgebase BK.

```
@book{ abelson-et-al:scheme,
  author = "Harold Abelson and Gerald~Jay Sussman and Julie Sussman",
  title = "Structure and Interpretation of Computer Programs",
  publisher = "MIT Press",
  address = "Cambridge, Massachusetts",
  year = "1985"
}

@article{ brachman-schmolze:kl-one,
  author = "Ronald~J. Brachman and James~G. Schmolze",
  title = "An overview of the {KL-ONE} knowledge representation system",
  journal = "Cognitive Science",
  volume = "9",
  number = "2",
  pages = "171--216",
  month = "April--June",
  year = "1985"
}

@article{ gottlob:nonmon,
  author = "Georg Gottlob",
  title = "Complexity results for nonmonotonic logics",
  journal = "Journal of Logic and Computation",
  volume = "2",
  number = "3",
  pages = "397--425",
  month = "June",
  year = "1992"
}
```

Figure 5.10 ijcai.bib

(3) The author downloads the ijcai.sty and ijcai.bst from IJCAI which is revised by us.

The style control files, such as ijaci.sty and ijcai.bst, which define the general formatting and commands for IJCAI papers, are provided by IJCAI with contributions from us. Authors can downloads these files along with instructions from the conference website, in conjunction with other LaTeX files.

(4) SKO TeX processes ijcai.bib and generates ijcai'.bib.

```
@book{ abelson-et-al:scheme,
  author = "\href{http://groups.csail.mit.edu/mac/users/hal/hal.html}{Harold Abelson}
and \href{http://groups.csail.mit.edu/mac/users/gjs/}{Gerald~Jay Sussman} and
\href{http://mitpress.mit.edu/catalog/author/default.asp?aid=830}{Julie Sussman}",
  title = "\href{http://mitpress.mit.edu/sicp/full-text/book/book.html}{Structure and
Interpretation of Computer Programs}",
  publisher = "\href{http://mitpress.mit.edu/main/home/default.asp}{MIT Press}",
  address = "\href{http://en.wikipedia.org/wiki/Cambridge,_Massachusetts}{Cambridge,
Massachusetts}",
  year = "1985"
}

@article{ brachman-schmolze:kl-one,
  author = "\href{http://en.wikipedia.org/wiki/Ronald_J._Brachman}{Ronald~J. Brachman}
and \href{http://www.cs.tufts.edu/~schmolze/}{James~G. Schmolze}",
  title = "\href{http://linkinghub.elsevier.com/retrieve/pii/S0364021385800148}{An
overview of the {KL-ONE} knowledge representation system}",
  journal = "\href{http://www.cognitivesciencesociety.org/journal_csj.html}{Cognitive
Science}",
  volume = "9",
  number = "2",
  pages = "171--216",
  month = "April--June",
  year = "1985"
}

@article{ gottlob:nonmon,
  author = "\href{http://www.comlab.ox.ac.uk/people/georg.gottlob/}{Georg Gottlob}",
  title = "\href{http://logcom.oxfordjournals.org/content/2/3/397.full.pdf}{Complexity
results for nonmonotonic logics}",
  journal = "\href{http://logcom.oxfordjournals.org/}{Journal of Logic and
Computation}",
  volume = "2",
  number = "3",
  pages = "397--425",
  month = "June",
  year = "1992"
}
```

Figure 5.11 ijcai'.bib

Figure 5.11 shows an extract from the file ijcai'.bib. After processing by the SKO TeX processor, entities encapsulated within the entries have been enriched with hyperlinks that are realized by sets of "\href" commands. The ijcai'.bib file is completely compatible with LaTeX.

(5) SKO Tex processes ijcai.tex and generatesijcai'.tex.

Meanwhile, the SKO TeX processor generates the file ijcai'.tex, which compiles customized commands such as \citeauthor{}, \citeconf{}, etc. into common LaTeXsyntactic and enriched content as illustrated in Figure 5.12.

```
\title{\href{http://ijcai-11.iiia.csic.es/}{IJCAI--11} Formatting Instructions\thanks{These match the
formatting instructions of \href{http://www.ijcai-07.org/}{IJCAI-07}. The support of
\href{http://www.ijcai.org/}{IJCAI}, Inc. is acknowledged.}}
\author{\hyperlink{Toby}{Toby Walsh}\\
\href{http://www.nicta.com.au/}{NICTA} and \href{http://www.unsw.edu.au/}{UNSW}\\
\href{http://en.wikipedia.org/wiki/Sydney}{Sydney}, \href{http://en.wikipedia.org/wiki/Australia}
{Australia} \\
\href{mailto:pcchair11@ijcai.org}{pcchair11@ijcai.org}
\And
\hyperlink{Fausto}{Fausto Giunchiglia}\\
\href{http://disi.unitn.it}{DISI}, \href{http://www.unitn.it}{UNITN}\\
\href{http://en.wikipedia.org/wiki/Trento}{Trento}, \href{http://en.wikipedia.org/wiki/Italy}
{Italy}\\
\href{mailto:fausto@disi.unitn.it}{fausto@disi.unitn.it}
}

\section*{Acknowledgments}

The preparation of these instructions and the \LaTeX{} and Bib\TeX{}
files that implement them was supported by \href{http://www.parc.com/}{Schlumberger Palo Alto
Research}, \href{http://www.corp.att.com/attlabs/}{AT\&T Bell Laboratories}, and
\href{http://www.mkp.com/}{Morgan Kaufmann Publishers}.
Preparation of the Microsoft Word file was supported by \href{www.ijcai.org}{IJCAI}.  An
early version of this document was created by \href{http://www.linkedin.com/pub/shirley-
jowell/17/a44/453}{Shirley Jowell} and \href{http://ect.bell-labs.com/who/pfps/}{Peter
F. Patel-Schneider}.  It was subsequently modified by Jennifer
Ballentine and \href{http://www.cs.brown.edu/~tld/}{Thomas Dean}, \href{http://www.informatik.uni-
freiburg.de/~nebel/}{Bernhard Nebel}, and Daniel Pagenstecher.
These instructions are the same as the ones for \href{http://ijcai.org/~ijcai05/}{IJCAI--05},
prepared by
Kurt Steinkraus, \href{http://web.mit.edu/}{Massachusetts Institute of Technology},
\href{http://www.csail.mit.edu/}{Computer
Science and Artificial Intelligence Lab}.
```

Figure 5.12 ijcai'.tex

(6) LaTex processes ijcai'.tex andijcai'.bib, and generates ijcai'.pdf.

As shown in Figure 5.13, in contrast to the original file provided by IJCAI-11, the ijcai'.pdf, processed by SKO TeX, is enriched with plenty of links. In the current implementation, these links are still hyperlinks that connect the content to our knowledgebase BK or to other webpages.

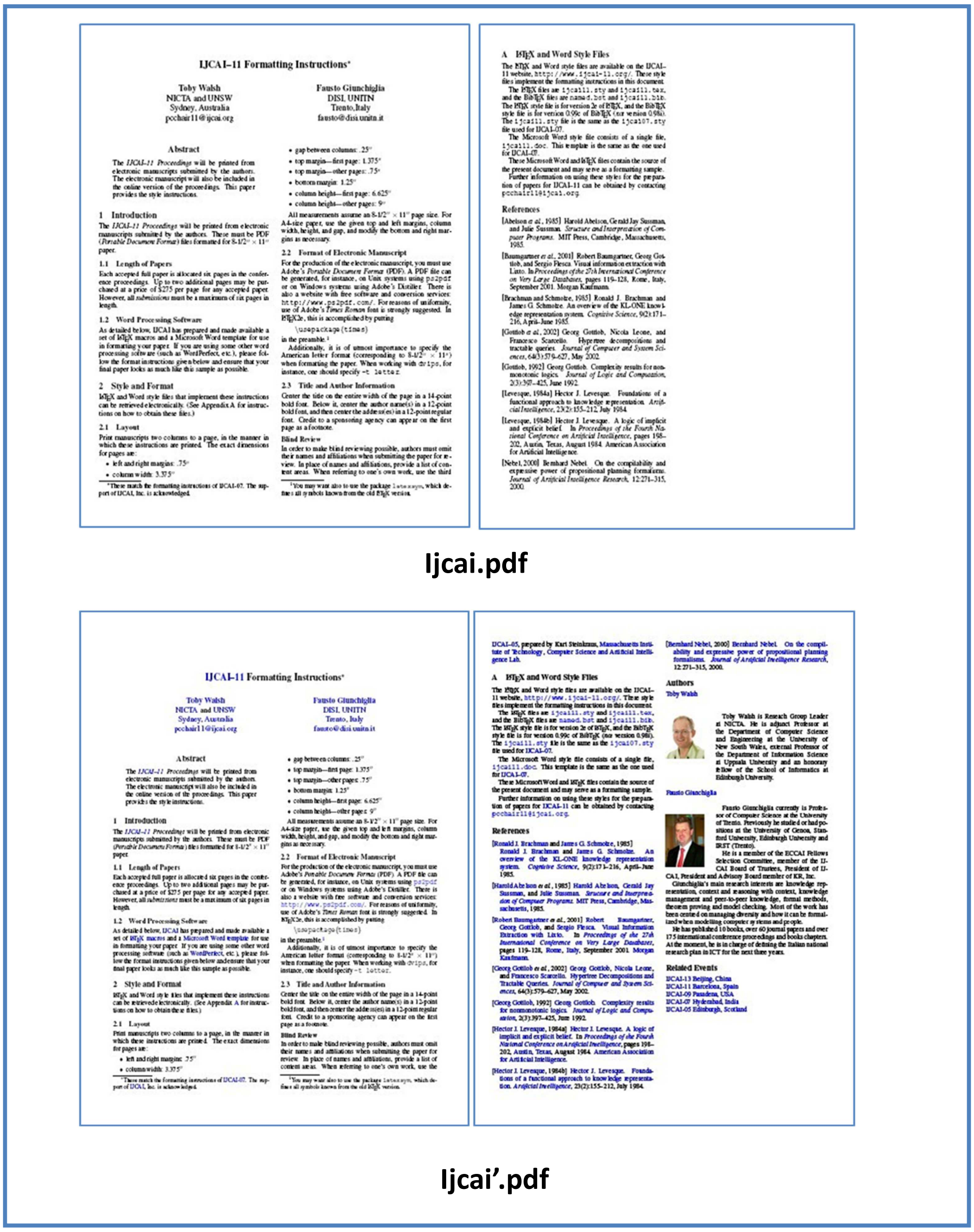


Figure 5.13 Comparison between ijcai.pdf and ijcai'.pdf

Chapter 6

Conference of the Future

Current conference models have been heavily criticized in various scientific communities[103,104,105] in terms of, for example, superficial reviews, a flood of syntactically correct yet meaningless papers, a lack of social connectedness, financial and logistical restraints, comments and discussions about papers which can hardly be kept track of, etc.

The "Conference of the Future" Initiative aims to establish a new way to submit, evaluate, revise, publish, comment on and reuse, in future, the contents of the papers published in a conference. Such conferences enable researchers to communicate much more interactively, with the live presentation being only one stage of the interaction, albeit the most important, in terms of what happens before and after the conference. Referee feedback is provided as part of the reviewing process. For those papers which are initially accepted, the reviewing, shepherding, commenting on, and revision process keeps going until after the conference, when the paper is finalized. Even after publication, the papers can be commented upon and become the topic of online discussion, leading eventually to the submission of new papers [106].

In this chapter, we propose a high-level prototype for the "Conference of the Future", the initial inspiration for which came from

EuroPLoP [107] and Liquid Conference [108]. Our focus is to merge emerging web technologies, i.e. social network services and the semantic web, into a revised conference model based on existing ones, and finally develop a semantic platform for managing Scientific Knowledge Objects (SKO)[109], in particular conferences, scientific publications, and researchers. Also, this is a follow up on the work of SKO Patterns [110] discussed in Chapter 5, in which we defined a general coarse-grained rhetorical structure and semantic annotation schemes for scientific discourse.

The rest of chapter is organized as follows.

Section 6.1 investigates three state-of-the-art conference models.

Section 6.2 addresses the problems we face today.

In Section 6.3, we propose a preliminary process and functional design as a high-level solution.

Section 6.4 discusses some benefits and constraints.

Finally, Section 6.5 points out our current implementations.

6.1 Current Models

The traditional conference model is widely used in various research communities nowadays, while two distinctive conference models, i.e. Liquid Conferences and a PLoP series of conferences, are also introduced here.

- **Traditional Conferences**[76]

Traditional conferences are usually composed of a set of formal presentations. They tend to be short and concise, with a time span of

[76] Conference: http://en.wikipedia.org/wiki/Academic_conference

about 10 to 30 minutes; presentations are usually followed by a discussion. The work may be bundled in written form as academic papers and published as the conference proceedings. Usually a conference will include keynote speakers (often, scholars of some standing, but sometimes individuals from outside academia). The keynote lecture is often longer, lasting sometimes up to an hour and a half, particularly if there are several keynote speakers on a panel.

In addition to presentations, conferences also feature panel discussions, round tables on various issues, and workshops.

Prospective presenters are usually asked to submit a short abstract of their presentation, which will be reviewed before the presentation is accepted for the meeting. Some disciplines require presenters to submit a paper of about 6–15 pages, which is peer reviewed by members of the programme committee or referees chosen by them.[111]

- **Liquid Conferences**[77]

Liquid Conferences aim to provide an alternative in the form of virtual conferences where presentations and discourse take place in a dedicated online environment. Invited authors present papers for discussion within the community. In response to this discourse, the participants may revise or adapt their papers; community members with interesting comments can be invited to expand them into full articles; and both discourse and revisions are all archived in perpetuity for future updates and reference purposes.

Key features:

(1) Effective online environment for virtual meetings, which do not carry logistical costs and do not require all participants to be in

[77] Liquid Conferences: http://project.liquidpub.org/research-areas/liquid-conferences

the same place at the same time.

(2) Both conventional peer review and post-publication community feedback.

(3) Moderated, intelligent open discourse to surround each presented paper.

(4) Easy opportunities for spontaneous invitations to present papers.

(5) The resulting collections of articles and associated (or selected) commentary can be turned into books, journal special issues, or other forms of publication [112].

- **PLoP Series of Conferences**[78]

The core of a PLoP Conference is a series of writers' workshops where authors work together to improve their papers. Before patterns or other papers are accepted for a writers' workshop, they are shepherded (non-anonymously). This means that an experienced author will discuss your submission with you, so that you can refine your paper prior to the conference. All submissions will be peer-reviewed.

Post-shepherding papers may be accepted directly into a conference workshop, or into a writing group. Writing Group papers will receive additional face-to-face shepherding at the conference itself. Writing Group papers reaching the required standard will be considered for workshop review on the final day of the conference.

After the conference, authors get more feedback and inspiration from writers' workshops or through on-site shepherding. They keep working on the papers continuously. Half a year later, they submit final

[78]XPLoP Conferences: http://hillside.net/europlop/europlop2011/cfp.html

versions for post-conference proceedings [107, 113].

6.2 Current Problems

Emerging web technologies are revolutionizing the way scientific knowledge is produced and disseminated. However, current models of academic conferences are comparatively limited, lacking in collaborative networked discussions between authors, reviewers, commenters, and readers. Specifically, in a traditional conference, an accepted paper is simply reviewed by two or three referees, and discussed by a limited number of participants during a half-hour presentation, while an author who is always engaged in presenting slides and answering questions has little chance of recording the feedback.

Existing conference models are heavily criticized in [103,104,105,112] from the point of view of different roles.

Authors: lack of fairness, lack of transparency, low quality or superficial reviews, biased reviewers, reviews based on half-read papers, decisions based on one or two reviews only, author feedback with zero impact, overfocus on getting details right, overformalized papers[79], overselling[80], and frustration- especially for PhD students [99].

Readers: flood of syntactically correct yet meaningless papers, delta papers[81], fostering of niche topics[82], over-polished papers, suppression of

[79]For example, a paper with excessive notations that obscures what's really going on.

[80]Currently a considerable portion of the paper writing process goes intoselling, i.e. justifying the work in the Introduction, contrasting it with other related work, and makingsure it is different or has some other twist that was not investigated before.

[81]A way of transmitting data in the form of differences between sequential data rather than complete papers.

dissent with mainstream ideas, crushing of unpolished yet interesting research ideas and directions, topic killing, missing re-experimentation, no publishing of negative results, biased experimentation, dataset and query picking, long review times, slow innovation process [99].

Reviewers: review overload at few times a year, missing reviewing standards and guidelines, huge investment in reading a long paper [99].

Conference Organizers: conference centres have to be booked, accommodation found, financial support has to be obtained, and despite video and audio recording technology, *most of the discourse gets lost.* The presented papers are documented in the respective proceedings, while the discussions about these papers are usually not kept track of [105].

6.3 The New Conference

The solution for "Conference of the Future" consists of three parts, i.e.

(1) Submission Format and Types

A new submission format and types made up of rhetorical blocks and associated global/local metadata.

(2) Review Process

A social and transparent review process open to entire research communities.

(3) Conference Structure

[82] Currently we often see arguments like "Although paper X provided a generalsolution for problem Y, it did not consider the case where <whatever>. This paper Z fills the gap."

A focused and interactive conference structure.

6.3.1 Submission Format and Types

All submissions are encouraged to be written following the SKO Patterns format (Figure 4.3). This is made up of a set of global metadata, e.g. bibliographic information, abstracts, reference sets, annotations, etc., and a set of rhetorical blocks, e.g. State of the Art, Problem Statement, Solutions, Methods, Materials, Results, Evaluations, Discussions, along with a set of local metadata associated with each rhetorical block.

Note: Generally, all these authoring and annotating can be done on our proposed online platform. Specifically, for LaTeX users, the tagging of paper structure and metadata can be easily done using SKOTeX. We also integrate existing parsers and converters as LaTeX plug-ins that help us to produce PDF, or HTML format of papers. For those authors/contributors who are used to writing papers using Microsoft Word (or others), we plan to implement some templates, e.g. the .dot file, in future work.

It’s not necessary to submit an article that consists of all rhetorical blocks and metadata sets. *Our idea is to allow these rhetorical blocks (paper parts) to be submitted, reviewed, commented on, and published individually.* We welcome various types of submissions for particular interest groups in the conference. For instance, the *Submission Types* may include:

(1) “State of the Art” Papers, e.g. survey papers;

(2) “Problem Statement” Papers, e.g. PhD symposium papers;

(3) “High-Level Solution” Papers, e.g. vision papers, poster papers;

(4) “Research” Papers, e.g. papers containing detailed descriptions of “Methods”, “Material”, “Results”, and “Evaluation”.

(5)…etc.

In addition, *all the discussions about submissions can be tracked* and permanently archived (with URL or DOI) on the "Conference of the Future" platform, including all versions of papers, reviews, shepherdings, comments, conversations, presentations, even audios and videos during pre-conference, at-conference, and post-conference phases.

Authors are encouraged to collaborate to the maximum possible extent with other researchers and REUSE existing research outputs.

We take this paper as an example of a possible submission to "Conference of the Future". We currently ignore the copyright and licensing issues on reuse that have already been discussed in the Liquid Pub project.

Figure 6.1 roughly illustrates the composition of this chapter. The blue blocks show the skeleton of this paper, i.e. "State of the Art", "Problem Statement", "Solution", and "Discussion". The pink blocks (SotA1-3, PS1-4, S1) indicate those content that have been reused from other sources, while the purple blocks (S2-3, D1-3) are our original work. In the header of this paper, we may mark the "Document Type" of our paper as "High-Level Solution Paper", while readers can directly see our main contribution and can read strategically. We will invent better notations for clearly indicating "Reuse" parts and "Original" parts in a later paper.

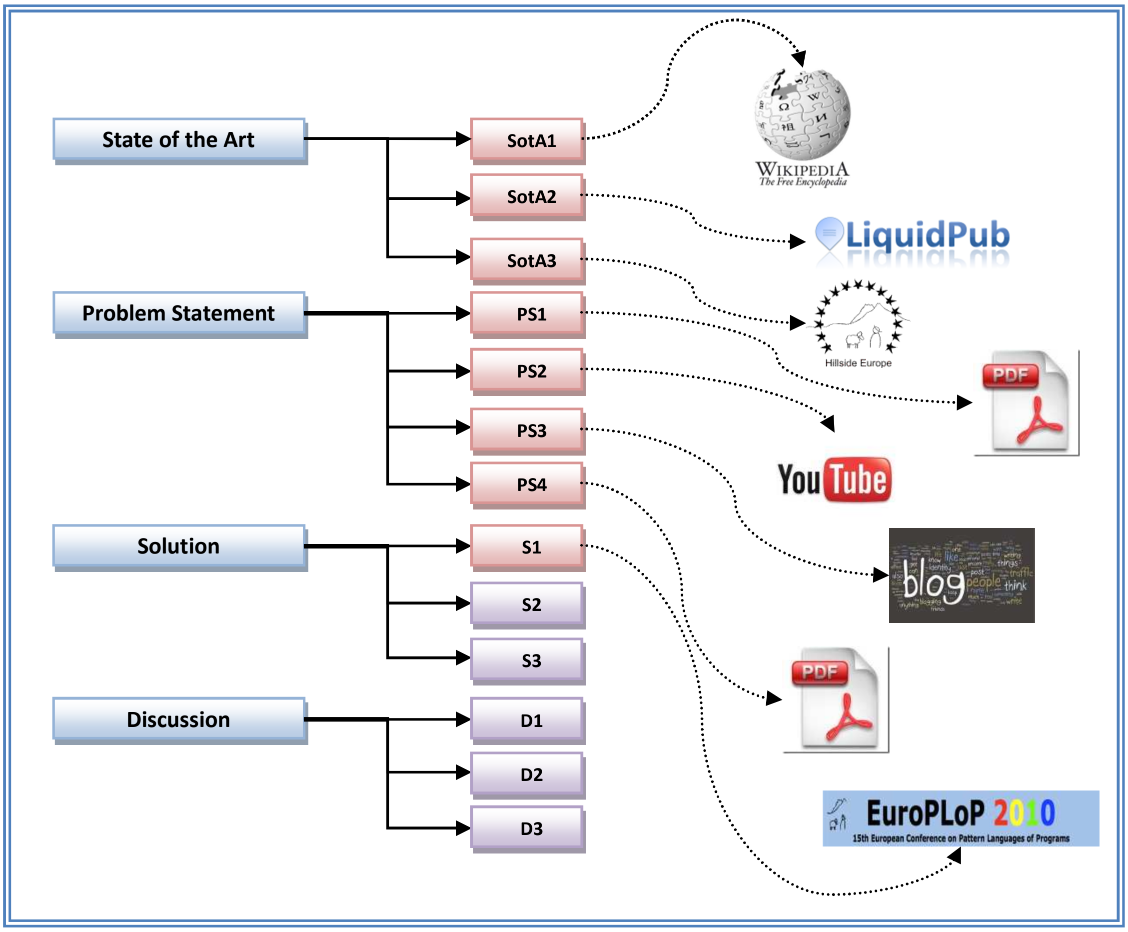


Figure 6.1 Discourse Composition

Since the paper evolves during its whole lifecycle, more commenters, readers, reviewers, and shepherds will become co-authors of the paper, or contributors to certain paper parts. SKO patterns provide this capability using metadata/entity annotations. Such submissions/publications will be enriched with more *related/recommended information* for reading and also facilitate a *semantic search*.

6.3.2 Review Process

Basically, there are three rounds of review - review for publishing online, review for presentation (on-site discussion/shepherd) and review for publication.

(1) Review for publishing online (pre-publication)

Authors submit articles using the “Conference of the Future” online platform. Submissions are screened to filter out articles which clearly fail to match the quality and suitability criteria for the conference - these are rejected outright.

Other *articles are made available online that are immediately open to research communities for discussion*. The Programme Chair assigns a shepherd to each article.

Each paper is open to discussion, usually for a period of 30-60 days. Shepherd and readers comment on the paper, authors respond and revise their paper, and moderators (Area Chairs) moderate messages from the larger audience.

Each paper, with its discussion, is then archived and is kept available for reading, commenting on, and annotating on the website.

(2) Review for presentation (conference proceedings)

The Area Chair solicits three reviews for each paper, usually to be completed within 2-4 weeks. The Area Chair also openly invites other researchers in the area to write public reviews for the latest version of the manuscript.

The reviews of the solicited reviewers are posted on the platform – usually together with their names and affiliations. Reviewers may choose to remain anonymous. Any other researcher can choose to become a reviewer for the article by posting a non-anonymous review on the platform. These reviews will be made available as soon as the solicited reviews are online.

Area Chairs make a first pass and identify all papers which are sure accepts or sure rejects for conference presentation.

The Programme Chair plus Area Chairs meet and make the final decision on any controversial papers, and this may require some additional reviews. They also consider possible controversies which might have arisen during the rebuttal phase.

The reviews plus rebuttals plus possibly added reviews, plus final decisions get sent to the authors.

(3) Review for publication (post-proceedings/journals)

After the conference, authors get more feedback and keep working on the papers. They submit a complete and mature work for Post-Proceedings/Journals review. This version always involves more co-authors and contributors who collaborated with the original authors during the last two review periods and the conference.

If the article is accepted, all shepherds and reviewers who were substantially involved are named in the final version.

6.3.3 Conference Structure

A conference can be held by getting people together, *either at a certain venue or via the internet.*

Rather than presenting every accepted paper, the "Conference of the Future" should *select certain paper parts (blocks) for presentation*. We needn't repeat the same "State of the Art" or similar "Problem Statement" in many different presentations. Since all the conference papers have been available and discussed online for months, people in the conference should more focused on innovation and collaboration.

Besides, some ideas from the Writers' Workshop, Writing Groups, and Focus Groups from EuroPLoP offer quotable experiences that may also be adopted in the "Conference of the Future", such as "feedback oriented" discussion instead of presentation, authors and shepherds' face-to-face, detailed shepherding, birds-of-a-feather sessions, etc.

6.4 Discussion

Benefits

No early crushing of high-level ideas: high-level ideas may be proposed as *"high-level solution"* papers. Neither details nor algorithms are required.

Shorter publications: paper parts can be submitted, reviewed, commented on, and published individually.

Accelerated innovation process: All the papers, reviews, comments are "open source".

Versioning: a paper can be updated and evolved over time just as is the case with software.

Collaboration from different paper parts: this involves a simple reuse instead of rephrasing and rewriting.

Better assessment of researchers: this explicitly shows who of the authors contributed to which parts of a long publication.

Strategic reading: read the most interesting parts instead of whole papers.

Semantic search: search by metadata and entities.

Constraints and Open Issues

Copyright and licensing issues on reuse: how to distinguish reuse and plagiarism.

High cost of metadata generation: how to generate high-quality metadata - by authors or by readers? No unified metadata standard yet.

High cost of data/metadata maintenance: this is also an open issue with regard to the "Semantic Web". Data disaster can become metadata disaster.

Limited available shepherds: it's really hard to find a shepherd for each paper, even in a medium-sized conference.

Editing tools: we started with LaTeX, but parsing Word and PDF files will mean more challenges. It is certainly the case that our tools cannot support all the editing environments.

6.5 Current Implementation

In this section we briefly report the current implementation of the "Conference of the Future" platform. We took two papers as exemplars of semantic enhancements of scientific discourses, while the interface has been implemented in Javascript with Dojo.

As shown in Figure 6.2, the interface of the Conference of the Future comprises three main parts. These are:

Part1: a tool bar on the top, including "Settings", "Roles", "My Conference", "ETypes", "Format".

Part2: a side bar on the left, including "Metadata" and "Data" that are defined by SKO Patterns.

Part3: a view bar on the right

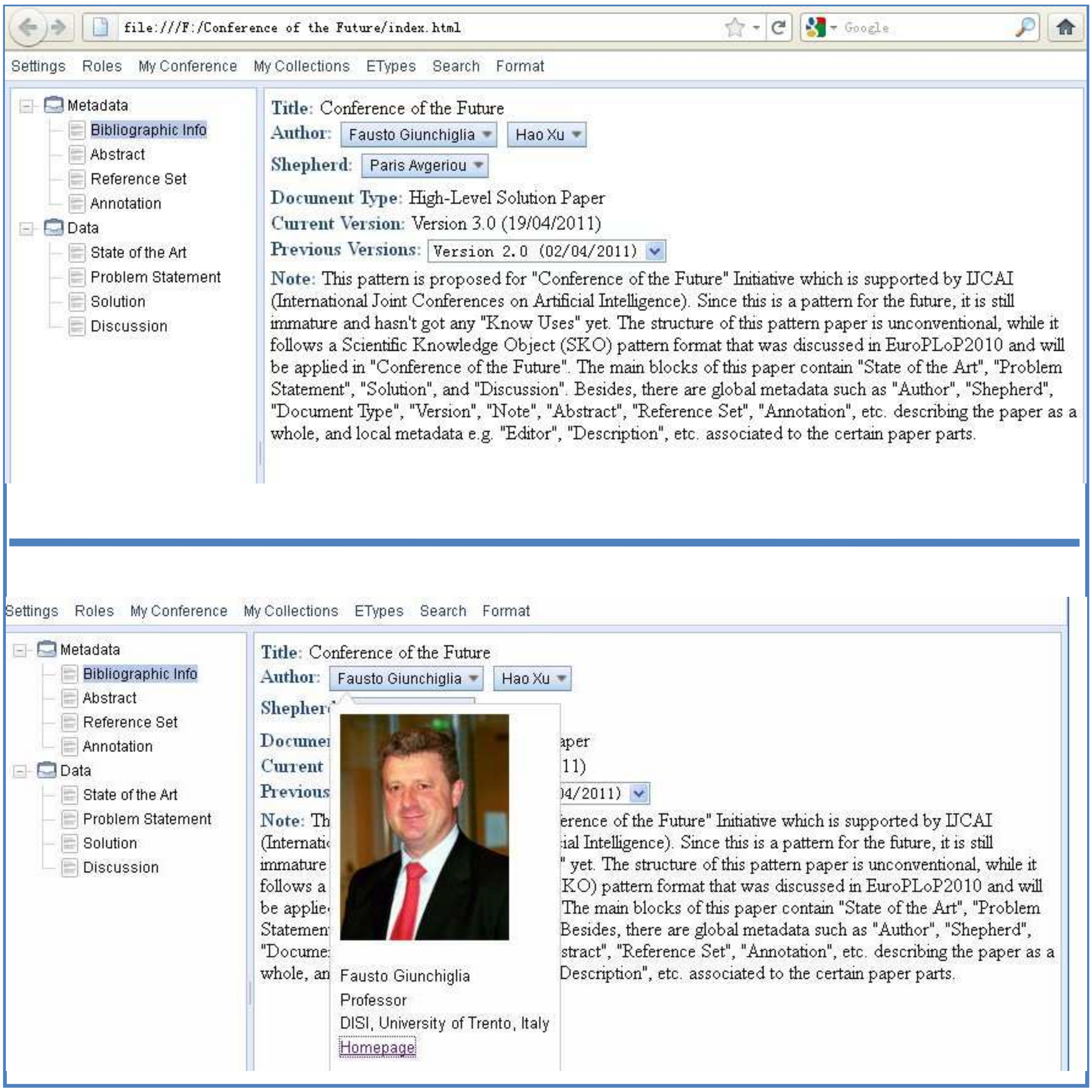


Figure 6.2 Bibliographic Information

Figure 6.2 illustrates the Bibliographic Info of a paper entitled "Conference of the Future" that was discussed in EuroPLoP 2011. From the navigation area, readers have been provided with a set of metadata such as title, author, shepherd, document type, version information, etc. Some entities have already been enriched with further meta-information. For example, when we click on one of the authors "*Fausto Giunchiglia*", a small dialogue window pops up with the metadata of *Fausto* retrieved from AISN.

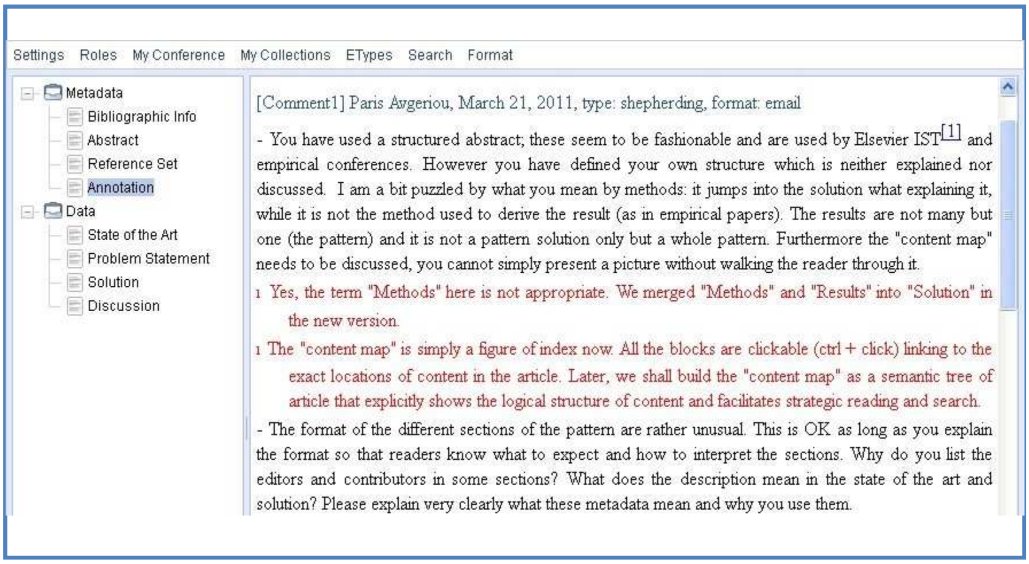


Figure 6.3 Annotation

Another example as presented in Figure 6.3 is that we collect all annotations together as the global metadata of an SKO. As a real practice, we tracked all comments from reviewers, shepherds, conference participants, and other readers of the paper "Conference of the Future". Generally, all kinds of format of annotations can be embedded into the platform, e.g. text, email, image, video, audio, etc. Withal, several comments are focused on some specific segments of the SKO or SKOnode. We linked these sources and targets together via hyperlinks that facilitate reading. Figure 6.4 exhibits a commenting environment on an SKOnode.

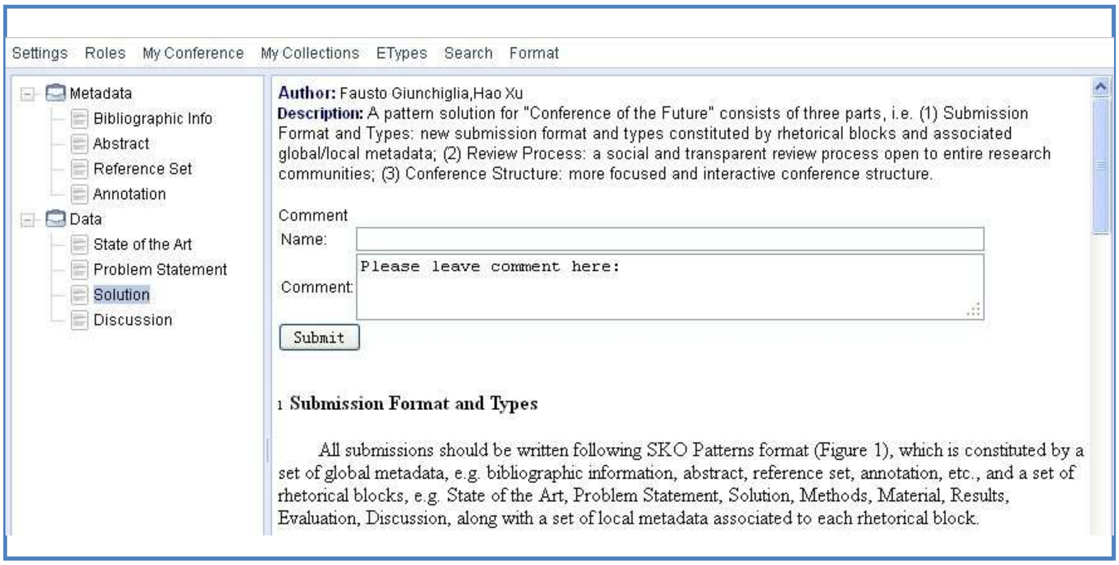


Figure 6.4 Comment

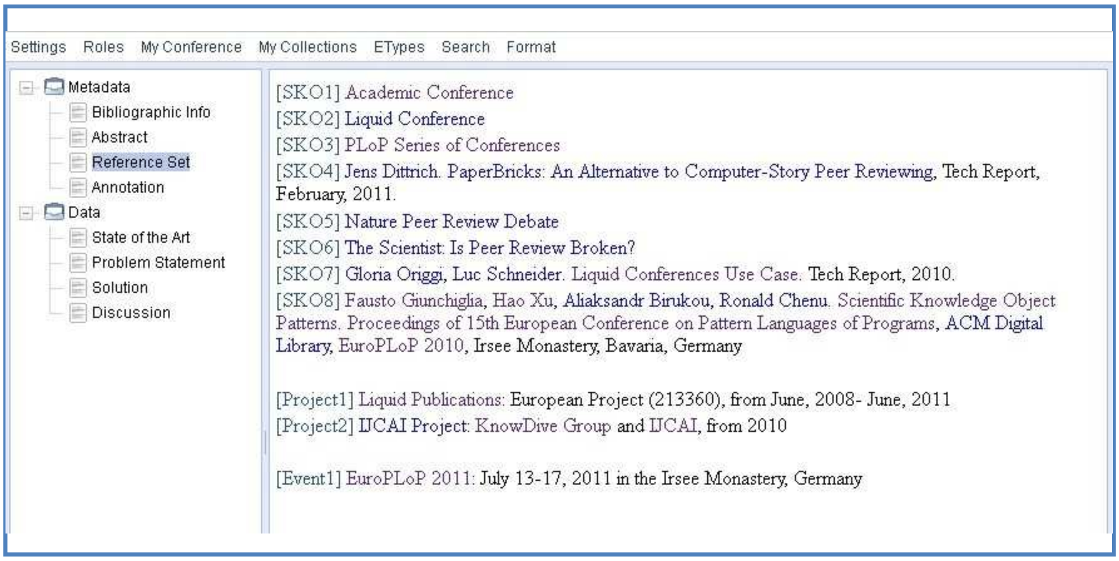
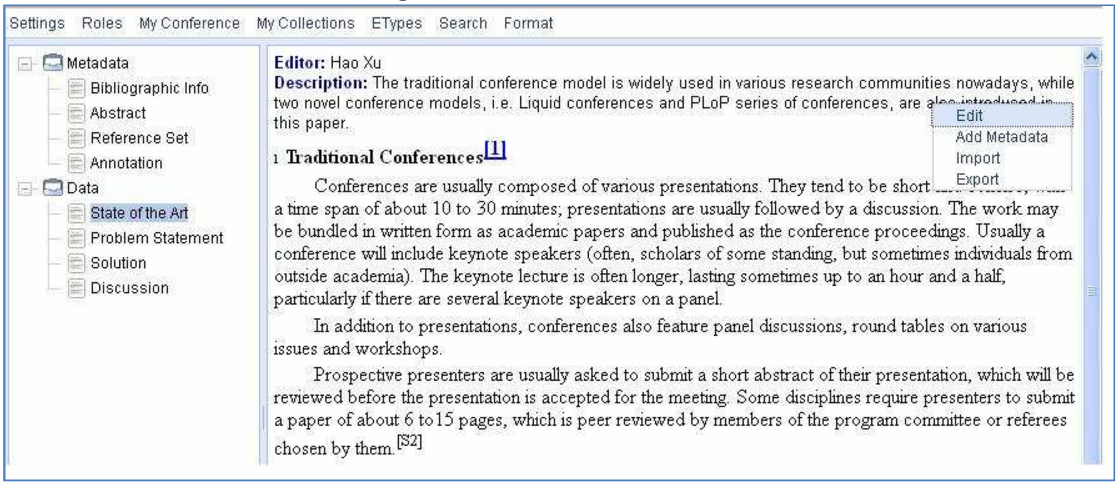


Figure 6.5 Reference Set

One of major functional enhancements to a CotF paper is that we use a Reference Set to replace traditional References. This means that the citations are extended to all the entity types defined in SKO Types from basic bibliographies. Actually, the current version of SKO TeX has already provides such features. In this case, a reader may easily access our Entitypedia or AISN and be efficiently fed with more nutrition during the reading process. In Figure 6.5, we depict a set of entities with the basic information that is embedded, with hyperlinks to knowledge bases and social networks such as SKO, Project, Event, etc.

Figure 6.6 State of the Art



As we know, besides automatic or semi-automatic extraction and explanation, semantic enhancements are always done by annotating manually. All CotF users may have permission to edit metadata rather than the data of the original content. Moreover, such metadata can be imported and exported for knowledge sharing purposes. Specifically, in a CotF paper, it’s common to have editors for certain SKOnodes. One reason should be reuse. This means that the content therein are not created by the contributor, but are copy-pasted or rephrased from other sources, while an editor is the person who organizes the collection of this data similarly to the situation with Wikipedia. The other reason for an “Editor” is from the perspective of the metadata. A person who provides valuable metadata that aggregates the original data as one of the contributors to this SKO, can be consider to be an editor, as is shown in Figure 6.6.

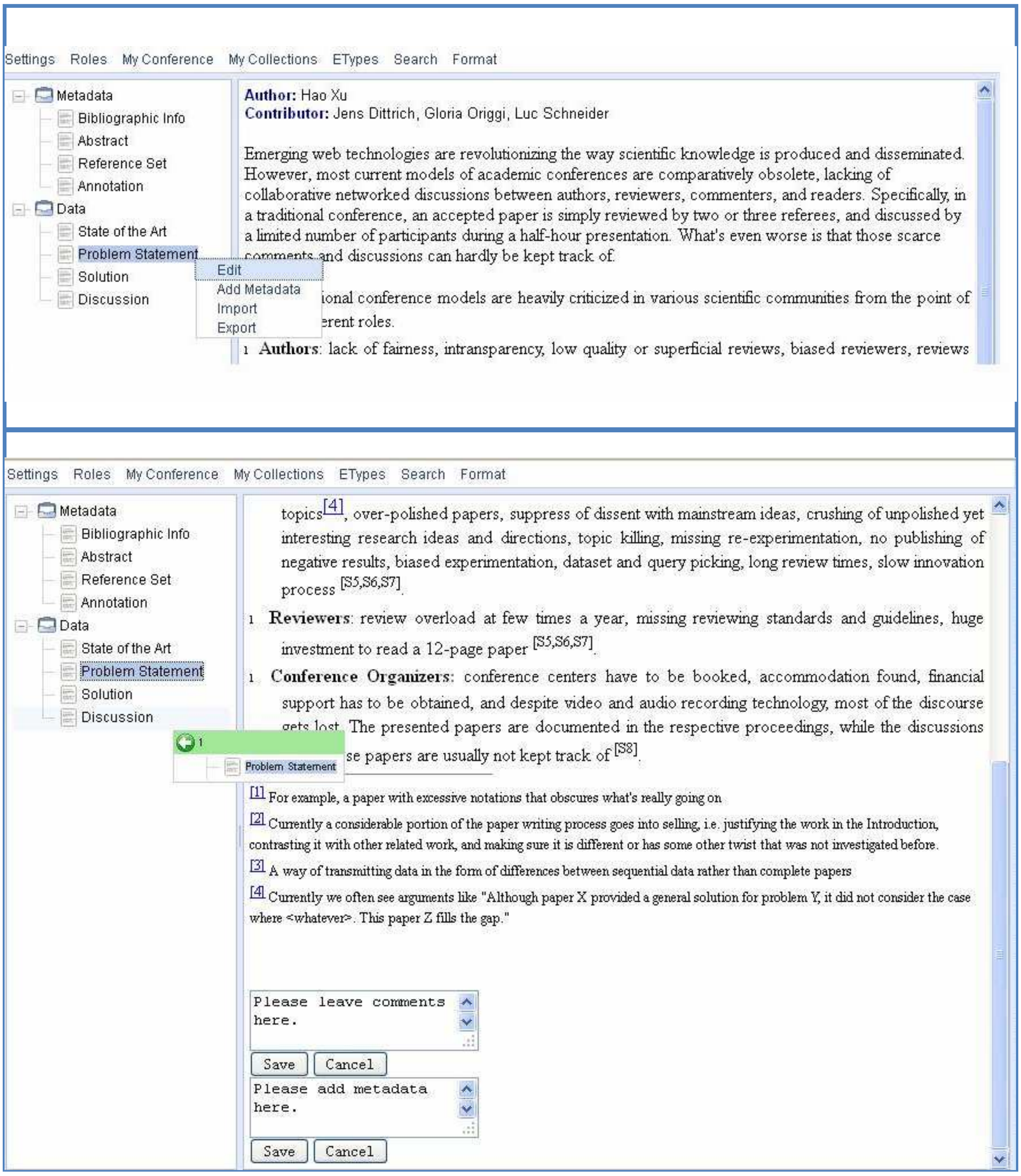


Figure 6.7 Management of Rhetorical Structure

We can also manage the rhetorical structure on the left side bar as shown in Figure 6.7. The present structure is the default one of SKO Patterns provided to its users. Basically it satisfies most articles in terms of the coarse-grained rhetorical structure and metadata schema. However, once a user needs to extend or modify such a structure, the

CotF platform supports such structure management by using right click or drag-and-drop.

The functionalities provided by the tool bar on the top of the interface as shown in Figure 6.8, are similar to other publishing tools such as easychair, etc. Setting controls private information management e.g. accounts, and so on. Also it will be equipped with RelBAC for access control in the near future. In Roles, it defines various roles with different access permissions such as Author, Reviewer, Shepherd, PC member, Chair, Reader. My Conference and My Collections are two SKO sets for managing personal submissions, events, or other interests. When we click on ETypes, a control bar pops up in the view bar. This has several entity selections including “turn all highlighting on”, “person”, “SKO”, “conference”, “project”, “institution”, and “location”, each of which are covered with one distinct colour. When users choose one or more coloured selection buttons, the corresponding types of entities in the article will be highlighted. In search we will employ the efforts from S-Match and Concept Match, while in Format we can define the format of export files such as PDF, XML, etc.

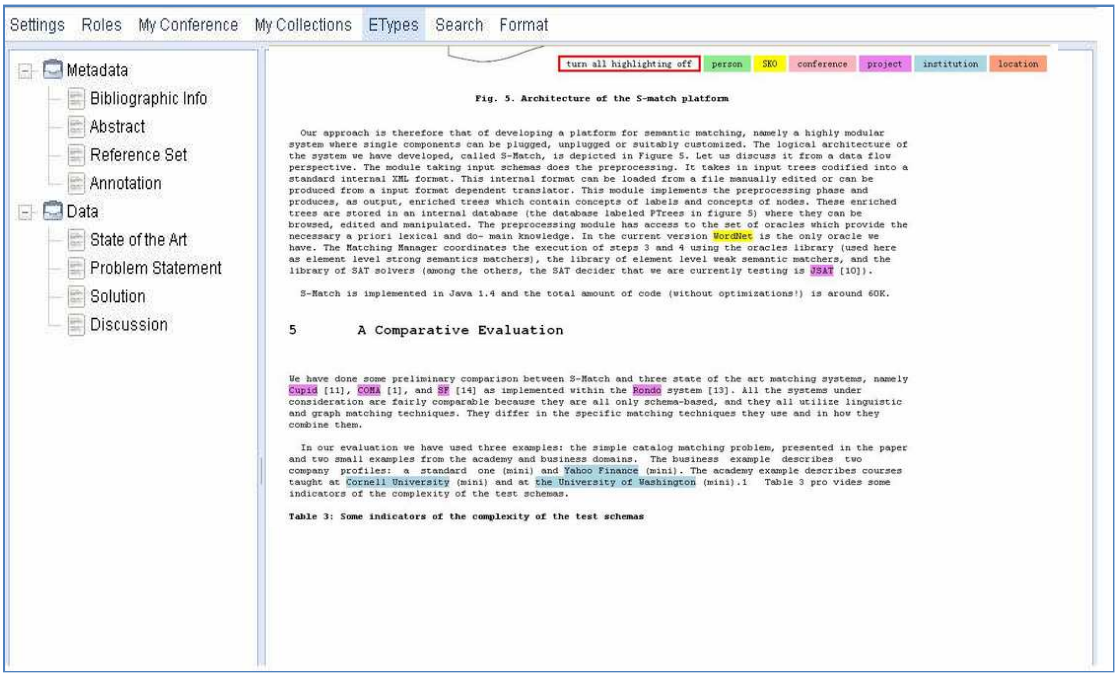


Figure 6.8 Tool Bar

Chapter 7

Conclusion

In this dissertation, we investigate existing widely-used metadata schemas, several prominent discourse representation models, and some emerging scientific publishing applications. We analyse the interoperability mechanisms between various metadata schemas, and summarize the underlying theoretical foundations in terms of models and applications of scientific discourse representation.

We propose a Scientific Knowledge Object (SKO) Framework in terms of a theory, a methodology, a tool, and an application for SKO management, in the context of an emerging social and semantic web. The main contribution of this research can be summarized as follows:

1. SKO Types: A Theory of Structural Knowledge

SKO Types specifies sets of bibliographically related entities, relationships, attributes and services, intended to describe ubiquitous scientific knowledge objects semantically, and to facilitate their dissemination, collaboration, evolution and reuse.

2. SKO Patterns: A Methodology for Discourse Representation

SKO Patterns not only draw on the essence of the existing rhetorical structured models, but also extend the capabilities of semantic annotation, semantic search, and strategic authoring, grounded on logical reasoning, i.e. Deduction, Induction, and Abduction.

3. SKO TeX: A Tool for Semantic Authoring and Annotation

SKO TeX is an editing environment, a file format and an entity repository, which support the management of data, metadata and related entities for scientific publications. It provides a viable way for authoring and annotating semantic documents using SKO Patterns.

4. Conference of the Future: An application of Open Science

The “Conference of the Future” Initiative aims to establish a new way to submit, evaluate, revise, publish, comment on and reuse in future papers, the contents of the papers published in a conference. Such conferences enable researchers to communicate much more interactively, while the live presentation is only one stage of the interaction, even if the most important, in terms of what happens before and after the conference.

Despite the multiplicity of the efforts made with regard to this thesis, several incremental steps towards developing and integrating SKO theories and applications form some future trajectories. The focal point will be an extension and refinement of the SKO Patterns Framework, especially for metadata exchange mechanism, fine-grained rhetorical structure representation and an automatic semantic parser for SKO TeX. We intend to launch the “Conference of the Future” platform in IJCAI-2013, in terms of implementing an online management system for all conference submissions, discussions, and related entities/ontologies, along with an SKO Editor- a set of macros and parsing tools for authoring/annotating SKOs in the LaTeX and Office Word editing environment.

Bibliography

[1] Crawford, S.Y., Hurd, J.M. and Weller, A.C. , From Print to Electronic. The Transformation of Scientific Communication, ASIS, Medford, NJ.1996

[2] Budapest Open Access Initiative. 2002. http://www.webcitation.org/query?id=3046. Accessed 17 Nov 2005 .

[3] Peter Suber, "Open Access Overview: Focusing on Open Access to Peer-Reviewed Research Articles and Their Preprints," http://www.earlham.edu/~peters/fos/overview.htm.

[4] Charlie Schick . The Future Of Scientific Publishing. Talk at the 3rd WLE Symposium, London, 2009

[5] Sebastian Ryszard Kruk, Semantic Digital Libraries, Tutorial on Semantic Digital Libraies at ICSD'09. 2009 http://www.slideshare.net/knowledgehives/tutorial-on-semantic-digital-libraries-at-icsd09

[6] "The notion of "metadata" introduced by Bagley". Solntseff. N, Yezerski. A , A survey of extensible programming languages, Annual Review in Automatic Programming, 7, Elsevier Science Ltd, pp. 267–307, 1974

[7] NISO Press. Understanding Metadata, available at www.niso.org, National Information Standards Organization, ISBN: 1-880124-62-9. 2004

[8] S. Weibel, J. Kunze, C. Lagoze, and M. Wolf. Dublin core metadata for resource discovery. 1998.

[9] E. Duval, W. Hodgins, S. Sutton, and S.L. Weibel. Metadata principles and practicalities. D-lib Magazine, 8(4):16, 2002.

[10] O. Lassila. Web metadata: A matter of semantics. Internet Computing, IEEE, 2(4):30-37, 1998.

[11] S. Weibel. Metadata: The foundations of resource description. 1995.

[12] R. Gartner, H. L'Hours, and G. Young. Metadata for digital libraries: state of the art and future directions. JISC, 2008.

[13] T. Oreilly. What is web 2.0. Design patterns and business models for the next generation of software, 30:2005, 2005.

[14] P. Miller. Web 2.0: building the new library. Ariadne, 45(1), 2005.

[15] A. Hotho, R. Jaschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. The Semantic Web: Research and Applications, pages 411-426, 2006.

[16] D. Zhou, J. Bian, S. Zheng, H. Zha, and C.L. Giles. Exploring social annotations for information retrieval. In Proceeding of the 17th international conference on World Wide Web, pages 715-724. ACM, 2008.

[17] L.A. Adamic, J. Zhang, E. Bakshy, and M.S. Ackerman. Knowledge sharing and yahoo answers: everyone knows something. In Proceeding of the 17th international conference on World Wide Web, pages 665-674. ACM, 2008.

[18] C.M. Chiu, M.H. Hsu, and E.T.G. Wang. Understanding knowledge sharing in virtual communities: an integration of social capital and social cognitive theories. Decision Support Systems, 42(3):1872-1888, 2006.

[19] R. Kosala and H. Blockeel. Web mining research: A survey. ACM SIGKDD Explorations Newsletter, 2(1):1-15, 2000.

[20] J. Han, M. Kamber, and J. Pei. Data mining: concepts and techniques. Morgan Kaufmann Pub, 2011.

[21] T. Berners-Lee and J. Hendler. Scienti_c publishing on the semantic web. Nature, 410:1023-1024, 2001.

[22] T. Berners-Lee. Linked data-the story so far. International Journal on Semantic Web and Information Systems, 5(3):1-22, 2009.

[23]Marcos Baez, Fabio Casati, and Maurizio Marchese. Universal resource lifecycle management. In Data Engineering, 2009. ICDE '09. IEEE 25th International Conference on, pages 1741-1748, 2009.

[24] Tim Clark Simon Buckingham Shum Tudor Groza, Siegfried Handschuh and Anita de Waard. A short survey of discourse representation models. In Proceedings 8th International Semantic Web Conference, Workshop on Semantic Web Applications in Scientific Discourse. Lecture Notes in Computer Science, Springer Verlag: Berlin, 26 Oct 2009, Washington DC. 2009.

[25] Takeuchi H. Nonaka, I. The knowledge-creating company: How japanese companies create the dynamics of innovation. Oxford University Press, 1995.

[26] Spooren W.P.M. Noordman L.G.M. Sanders, T.J.M. Coherence relations in a cognitive theory of discourse representation. Cognitive Linguistics, 4(2):93-133, 1993.

[27] Thompson S.A. Mann, W.C. Rhetorical structure theory: A theory of text organization. Technical report, Information Science Institute, 1987.

[28] A.H. Renear and C.L. Palmer. Strategic reading, ontologies, and the future of scientific publishing. Science, 325(5942):828, 2009.

[29] J. Willinsky. The unacknowledged convergence of open source, open access, and open science. First Monday, 10(8):1396-0466, 2005.

[30] Hao Xu. Managing Ubiquitous Scientific Knowledge on Semantic Web. Lecture Notes in Computer Science, Volume 6059: 421-430, Springer, AST/UCMA/ISA/ACN 2010, Miyazaki, Japan, 2010

[31] National Information Standards Organization. Understanding metadata. NISO Press, Bethesda, USA, 2004.

[32] D.V. Pitti. Encoded archival description: An introduction and overview. ESARBICA Journal: Journal of the Eastern and Southern Africa Regional Branch of the International Council on Archives, 20(0), 2005.

[33] T. Delsey, Library of Congress. Network Development, and MARC Standards Office. Functional analysis of the MARC 21 bibliographic and holdings formats. Network Development and MARC Standards Office, 2002.

[34] L. Cantara. Mets: The metadata encoding and transmission standard. Metadata: a cataloger's primer, page 237, 2005.

[35] R. Guenther. Premis-preservation metadata implementation strategies update 2: Core elements for metadata to support digital preservation. RLG DigiNews, 8(6), 2004.

[36] C. Lagoze, H. Van de Sompel, M. Nelson, and S.Warner. Open archives initiative-protocol for metadata harvesting-v. 2.0. 2002.

[37] M. Doerr. The cidoc conceptual reference module: an ontological approach to semantic interoperability of metadata. AI magazine, 24(3):75, 2003.

[38] E.T. ONeill. Frbr: Functional requirements for bibliographic records. Library resources & technical services, 46(4):150-159, 2002.

[39] Ieee learning technology standards committee (ltsc). IEEE P1484.12 Learning Object Metadata Working Group, 2000.

[40] N. Guarino and Istituto (Roma) Consiglio nazionale delle ricerc. Formal ontology in information systems, 1998.

[41] B. DArcus and F. Giasson. Bibliographic ontology specification. Retrieved October, 8, 2010.

[42] David Shotton, Silvio Peroni, FaBiO, the FRBR-aligned Bibliographic Ontology). http://speroni.web.cs.unibo.it/cgi-bin/lode/req.py?req=http:/purl.org/spar/fabio

[43] M. Synak and S.R. Kruk. Marcont initiative-the ontology for the librarian world. In 2nd European Semantic Web Conference ESWC 2005. Citeseer, 2005.

[44] J.H. Coombs, A.H. Renear, and S.J. DeRose. Markup systems and the future of scholarly text processing. Communications of the ACM, 30(11):933-947, 1987.

[45] D. Raggett, A. Le Hors, I. Jacobs, et al. Html 4.01 specification. W3C recommendation, 24, 1999.

[46] T. Bray, J. Paoli, C.M. Sperberg-McQueen, E. Maler, F. Yergeau, et al. Extensible markup language (xml) 1.0, 2000.

[47] O. Lassila and R.R. Swick. Resource description framework (rdf) model and syntax. World Wide Web Consortium, http://www. w3.org/TR/WD-rdf-syntax.

[48] D.L. McGuinness, F. Van Harmelen, et al. Owl web ontology language overview. W3C recommendation, 10:2004-03, 2004.

[49] F. Giunchiglia, P. Shvaiko, and M. Yatskevich. S-match: an algorithm and an implementation of semantic matching. The semantic

web: research and applications, pages 61-75, 2004.

[50] F. Giunchiglia and P. Shvaiko. Semantic matching. The Knowledge Engineering Review, 18(03):265-280, 2003.

[51] F. Giunchiglia, M. Yatskevich, and P. Shvaiko. Semantic matching: Algorithms and implementation. In Journal on Data Semantics IX, pages 1-38. SpringerVerlag, 2007.

[52] F. Giunchiglia, P. Shvaiko, and M. Yatskevich. Semantic schema matching. On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE, pages 347-365, 2005.

[53] F. Giunchiglia, P. Shvaiko, and M. Yatskevich. Discovering missing background knowledge in ontology matching. In Proceeding of the 2006 conference on ECAI 2006: 17th European Conference on Artificial Intelligence August 29,September 1, 2006, Riva del Garda, Italy, pages 382-386. IOS Press, 2006.

[54] F. Giunchiglia, M. Yatskevich, and E. Giunchiglia. E_cient semantic matching. The Semantic Web: Research and Applications, pages 272-289, 2005.

[55] R. Guha, R. McCool, and E. Miller. Semantic search. In Proceedings of the 12th international conference on World Wide Web, pages 700-709. ACM, 2003.

[56] F. Giunchiglia, U. Kharkevich, and I. Zaihrayeu. Concept search: Semantics enabled syntactic search. Semantic Search, page 109, 2008.

[57] F. Giunchiglia, U. Kharkevich, and I. Zaihrayeu. Concept search. The Semantic Web: Research and Applications, pages 429-444, 2009.

[58] M. Dzbor, J. Domingue, and E. Motta. Magpie-towards a semantic web browser. The SemanticWeb-ISWC 2003, pages 690-705, 2003.

[59] DA Quan and R. Karger. How to make a semantic web browser. In

Proceedings of the 13th international conference on World Wide Web, pages 255-265. ACM, 2004.

[60] M. Hildebrand, J. van Ossenbruggen, and L. Hardman. /facet: A browser for heterogeneous semantic web repositories. The Semantic Web-ISWC 2006, pages 272-285, 2006.

[61] S. Schaffert. Ikewiki: A semantic wiki for collaborative knowledge management. 2006.

[62] M. Bu_a, F. Gandon, G. Ereteo, P. Sander, and C. Faron. Sweetwiki: A semantic wiki. Web Semantics: Science, Services and Agents on the World Wide Web, 6(1):84-97, 2008.

[63] M. Volkel, M. Krotzsch, D. Vrandecic, H. Haller, and R. Studer. Semantic wikipedia. In Proceedings of the 15th international conference on World Wide Web, pages 585-594. ACM, 2006.

[64] S.R. Kruk and B. McDaniel. Semantic digital libraries. Springer Verlag, 2009.

[65] S.R. Kruk. Semantic Digital Libraries: Improving Usability of Information Discovery with Semantic and Social Services. Lulu. com, 2010.

[66] DCMI Specification: http://dublincore.org/specifications/

[67] Dublin Core Metadata Element Set: http://dublincore.org/documents/dces/

[68] DCMI Type Vocabulary: http://dublincore.org/documents/dcmi-terms/

[69]Barker Phil. What is IEEE Learning Object Metadata / IMS Learning Resource Metadata? CETIS standards briefings series, the Centre for Educational Technology Interoperability Standards, May 2005.

[70] Extension Mechanism- Schema.org:
http://schema.org/docs/extension.html

[71] Data Model- Schema.org: http://schema.org/docs/datamodel.html

[72] Scholarly Article Properties: http://schema.org/ScholarlyArticle

[73] Accurate bibliographic metadata and google scholar.

http://www.monperrus.net/martin/accurate+bibliographic+metadata+and+google+scholar

[74] Mancini, C., Shum, S.J.B.: Modelling discourse in contested domains: A semiotic and cognitive framework. International Journal of Human-Computer Studies 64(11) ,1154-1171, 2006

[75] Harmsze, F.A.P.: A modular structure for scientific articles in an electronic environment. PhD thesis, University of Amsterdam , 2000

[76] Anita de Waard and Gerard Tel. The abcde format enabling semantic conference proceedings. In SemWiki, 2006.

[77] Tudor Groza, Siegfried Handschuh, Knud MRnller, and Stefan Decker. Salt - semantically annotated latex for scientific publications. In ESWC, pages 518-532, 2007.

[78] Jim Giles. Open-access journal will publish first, judge later. Nature 445, 9, 2007
http://www.nature.com/nature/journal/v445/n7123/full/445009a.html

[79]Andrea Gawrylewski. New site pits “published” vs. “posted”. The Scientist - Magazine of the Life Sciences, 19th June 2007
http://www.the-scientist.com/news/display/53294/#ixzz1Se0JuVJr

[80] Fabio Casati, Fausto Giunchiglia, and Maurizio Marchese. Publish and perish: why the current publication and review model is killing research and wasting your money. Ubiquity, 8(3):1, 2007.

[81] Giuseppe Veltri, Matteo Turrini, Luc Schneider, Jordi Sabater-mir, Diego Ponte, Nardine Osman, Matus Medo, Ronald Chenu, Aliaksandr Birukou, Marcos Baez, Joseph Rushton Wakeling, Hao Xu. State of the Art in Scientific Knowledge Creation, Dissemination, Evaluation and Maintenance. E-Print, University of Trento, Italy, Technical Report # DISI-09-067, 2009 http://eprints.biblio.unitn.it/archive/00001773/01/LiquidPub_D1-1_TechRep.pdf

[82] Giunchiglia Fausto and Chenu Ronald. Scientific knowledge objects v.1. Technical report, University of Trento, Dipartimento di Ingegneria e Scienza dell'Informazione,Trento, Italy, 2009.

[83] Fausto Giunchiglia, Ronald Chenu, Hao Xu, Aliaksandr Birukou, Enzo Maltese. Design of the SKO Structural Model and Evolution Version 2. LiquidPub Project Deliverable 1.2, 2010 https://dev.liquidpub.org/svn/liquidpub/papers/deliverables/LP_D1.2v2.pdf

[84] Tillett, B.A.B. Bibliographic relationships: Toward a conceptual structure of bibliographic information used in cataloging. Unpublished doctoral dissertation, Graduate School of Library and Information Science, University of California, Los Angeles. 1987

[85] Fausto Giunchiglia, Ronald Chenu-Abente, Hao Xu, Uladzimir Kharkevich. A Metadata-Enabled Scientific Discourse Platform. E-Print, University of Trento, Italy, Technical Report # DISI-10-069, 2010 http://eprints.biblio.unitn.it/archive/00001939/01/069.pdf

[86] Fausto Giunchiglia, Hao Xu, Aliaksandr Birukou, Ronald Chenu. Scientific Knowledge Object Patterns. Proceedings of 15th European Conference on Pattern Languages of Program, EuroPLoP 2010, Irsee Monastery, Bavaria, Germany, 2010

[87] Gordana Dodig-Crnkovic. Scientific Methods in Computer Science. Conference for the Promotion of Research in IT at New Universities and at University Colleges in Sweden (2002)

[88] Schulte, B.A. (2003). Scientific writing and the scientific method: Parallel ‘‘hourglass’’ structure in form and content. The American Biology Teacher, 65, 591–594.

[89] Brandon C. Beltz, Edna Monroe, and Denise Williford. The scientific method. http://www.brandonbeltz.com/scimeth/index.htm

[90] A Pattern Language: Towns, Buildings, Construction

[91] Erich Gamma Richard Helm Ralph Johnson John Vlissides: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1995.

[92] Introduction to SPAR: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/SPAR/index.html

[93] A Pattern Language for Pattern Writing, http://hillside.net/index.php/a-pattern-language-for-pattern-writing

[94] Swales, J.M. 1990, Genre Analysis: English in Academic and Research Settings, Cambridge University Press.

[95] Hao Xu. A Semantic Pattern for Scientific Discourse Representation. Journal of Computational Information Systems. Volume 6(13): 4223-4228, 2010

[96] Matthew Michelson and Craig A. Knoblock.Exploiting Background Knowledge to Build Reference Sets for Information Extraction. IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial

Intelligence, Pasadena, California, USA, July 11-17, 2009. 2076-2082

[97] Piergiorgio Bertoli, Marco Pistore, Paolo Traverso. Automated composition of Web services via planning in asynchronous domains. Source: Artificial Intelligence 174 (2010) 316–361

[98] Alessandro Tomasi, Fausto Giunchiglia.Model Checking Syllabi and Student Careers, Roberto Sebastiani, TACAS2001, Tools and Algorithms for the Construction and Analysis of Systems, Genova, Italy, April 2001. LNCS , N. 2031, Springer

[99] Arun S. Maiya and Tanya Y. Berger-Wolf. Sampling community structure. Source: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010: 701-710

[100] Chiara Ghidini, Fausto Giunchiglia.Local models semantics, or contextual reasoning = locality + compatibility. Source: Artificial intelligence 127, 2001: 221-259

[101] Jacco van Ossenbruggen, Lynda Hardman and Lloyd Rutledge.Hypermedia and the Semantic Web: A Research Agenda. Source: Journal of Digital information, volume 3 issue 1

[102] Biplav Srivastava and Jana Koehler. Web Service Composition - Current Solutions and Open Problems. Source: In: ICAPS 2003 Workshop on Planning for Web Services, 2003, 28-35

[103] Nature Peer Review Debate, http://www.nature.com/nature/peerreview/debate/index.html.

[104] The Scientist: Is Peer Review Broken?, http://www.the-scientist.com/article/display/23061/.

[105] Jens Dittrich. PaperBricks: An Alternative to Computer-Story Peer

Reviewing, Tech Report, February, 2011.
http://arxiv.org/abs/1102.3523v1

[106] Fausto Giunchiglia, Hao Xu. Conference of the Future. Proceedings of 16th European Conference on Pattern Languages of Programs, EuroPLoP 2011, Irsee Monastery, Bavaria, Germany, 2011

[107] PLoP Series of Conferences:
http://hillside.net/europlop/europlop2011/cfp.html

[108] Liquid Conference:
http://project.liquidpub.org/research-areas/liquid-conferences

[109] Fausto Giunchiglia, and Ronald ChenuAbente, Scientific Knowledge Objects v.1. Technical Report DISI-09-006, Ingegneria e Scienza dell'Informazione, University of Trento. 2009

[110] Fausto Giunchiglia, Hao Xu, Aliaksandr Birukou, Ronald Chenu. Scientific Knowledge Object Patterns. Proceedings of 15th European Conference on Pattern Languages of Programs, ACM Digital Library, EuroPLoP 2010, Irsee Monastery, Bavaria, Germany

[111] Academic Conference:
http://en.wikipedia.org/wiki/Academic_conference

[112] Gloria Origgi, Luc Schneider. Liquid Conferences Use Case. Tech Report, 2010.
http://project.liquidpub.org/research-areas/liquidconferences.pdf

[113] Lise B. Hvatum: Some Patterns for Organizing PLoP Events, Proceedings of 17th Conference on Pattern Languages of Programs, PLoP 2010, Reno, Nevada, USA