Undefined 0 (0) 1
IOS Press

Ontologies and Languages for Representing Mathematical Knowledge on the Semantic Web

Christoph Lange

Computer Science, Jacobs University Bremen, Germany

E-mail: ch.lange@jacobs-university.de

Abstract. Mathematics is a ubiquitous foundation of science, technology, and engineering. Specific areas, such as numeric and symbolic computation or logics, enjoy considerable software support. Working mathematicians have recently started to adopt Web 2.0 environment, such as blogs and wikis, but these systems lack machine support for knowledge organization and reuse, and they are disconnected from tools such as computer algebra systems or interactive proof assistants. We argue that such scenarios will benefit from Semantic Web technology.

Conversely, mathematics is still underrepresented on the Web of [Linked] Data. There are mathematics-*related* Linked Data, for example statistical government data or scientific publication databases, but their *mathematical* semantics has not yet been modeled. We argue that the services for the Web of Data will benefit from a deeper representation of mathematical knowledge.

Mathematical knowledge comprises logical and functional structures – formulæ, statements, and theories –, a mixture of rigorous natural language and symbolic notation in documents, application-specific metadata, and discussions about conceptualizations, formalizations, proofs, and (counter-)examples. Our review of approaches to representing these structures covers ontologies for mathematical problems, proofs, interlinked scientific publications, scientific discourse, as well as mathematical metadata vocabularies and domain knowledge from pure and applied mathematics.

Many fields of mathematics have not yet been implemented as proper Semantic Web ontologies; however, we show that MathML and OpenMath, the standard XML-based exchange languages for mathematical knowledge, can be fully integrated with RDF representations in order to contribute existing mathematical knowledge to the Web of Data.

We conclude with a roadmap for getting the mathematical Web of Data started: what datasets to publish, how to interlink them, and how to take advantage of these new connections.

Keywords: mathematics, mathematical knowledge management, ontologies, knowledge representation, formalization, linked data, XML

1. Introduction: Mathematics on the Web – State of the Art and Challenges

A review of the state of the art of mathematics on the Web has to acknowledge Web 1.0 sites that are in day to day use: review and abstract services such as Zentralblatt MATH [30] and MathSciNet [35], the arχiv pre-print server [39], libraries of formalized and machine-verified mathematical content such as the Mizar Mathematical Library (MML [9]), and reference works such as the Digital Library of Mathematical Functions (DLMF [3]) or Wolfram MathWorld [8].

These sites have facilitated the *access* to mathematical knowledge. However, (i) they offer a limited degree of interaction and do not facilitate collaboration, and (ii) the means of automatically retrieving, using, and adaptively presenting knowledge through automated agents are restricted. Concerning the Web in general, problem (i) has been addressed by Web 2.0 applications, and problem (ii) by the Semantic Web. This section reviews to what extent these developments have been adopted for mathematical applications and suggests a new combination of Web 2.0 and Semantic Web to overcome the remaining problems.

1.1. How Working Mathematicians have Embraced Web 2.0 Technology

An increasing number of working mathematicians has recently started to use Web 2.0 technology for collaboratively developing new ideas, but also as a new publication channel for established knowledge. Research blogs and wiki encyclopedias are typical representatives.

1.1.1. Research Blogs and Wiki Encyclopedias

Researchers have found blogs useful to gather feedback about preliminary findings before the traditional peer review. Successful collaborations among mathematicians not knowing each other before have started in blogs and converged into conventional articles [47]. GOWERS, an active blogger, initiated the successful Polymath series, where blogs are the exclusive communication medium for proving theorems in a massive collaborative effort [14,100,50], including the recent collaborative review of a claimed proof of $P \neq NP$ [15]. Compared to research blogs, the MathOverflow forum [7], where users can post their problems and solutions to others' problems, offers more instant help with smaller problems. By its reputation mechanism, it acts as an agile simulation of the traditional scientific publication and peer review processes.

For evolving ideas emerged from a blog discussion, or for creating permanent, short, interlinked descriptions of topics, wikis have been found more appropriate. The nLab wiki [28], a companion to the n-Category Café blog [27], is a prominent example for that, and also for the emerging practice of *Open Notebook Science*, i.e. "making the entire primary record of a research project public", including "failed, less significant, and otherwise unpublished experiments" [188]. The Polymath maintainers have also set up a companion wiki for "collect[ing] pertinent background information which was no longer part of the active 'foreground' of exchanges on the [...] blog entries" [50]. Finally, where MathOverflow focuses on concrete problems and solution, the Tricki [21], also initiated by GOWERS, is a wiki repository of general mathematical techniques – reminiscent of a Web 2.0 remake of PÓLYA's classic "How to Solve It" [161].

Wikis that collect *existing* mathematical knowledge, for educational and general purposes, are more widely known. PlanetMath [159], counting more than 8,000 entries at the time of this writing, is a mathematical encyclopedia. The general-purpose Wikipedia with 15 million articles in over 250 languages also covers mathematics [187]. Targeting a general audience, it omits most formal proofs but embeds the pure mathematical knowledge into a wider context, including, e.g., the history of mathematics, biographies of mathematicians, and information about application areas. The lack of proofs is sometimes compensated by linking to the technically similar ProofWiki [17], containing over 2,500 proofs, or to PlanetMath. Finally, Connexions [72], technically driven by a more traditional Content Management System, is an open web repository specialized on courseware. Connexions promotes the contribution of small, reusable course modules – more than 17,000, about 4,000 from mathematics and statistics, and about 6,000 from science and technology – to its *content commons*, so that the original author, but also others can flexibly combine them into collections, such as the notes for a particular course.

The wikis mentioned so far have been set up from scratch, hardly reusing content from existing knowledge bases, but maintainers of established knowledge bases are also starting to employ Web 2.0 frontends – for example the recently developed prototypical wiki frontend for the Mizar Mathematical Library (MML) [178], a large library of formalized and machine-verified mathematical content. The wiki intends to support common workflows in enhancing and maintaining the MML and thus to disburden the human library committee.

1.1.2. Critique – Little Reuse, Lack of Services

Web 2.0 sites facilitate collaboration but still *require* a massive investment of manpower for compiling a knowledge collection. Machine-supported intelligent knowledge reuse, e.g. from other knowledge collections on the Web, does not take place. Different knowledge bases are technically separated from each other by using document formats that are merely suitable for knowledge presentation but not for *re*presentation, such as XHTML with LaTeX formulæ. The only way of referring to other knowledge bases is an untyped hyperlink. The proof techniques collected in the Tricki cannot be automatically applied to a problem developed in a research blog, as neither of them are sufficiently formalized.

Intelligent information retrieval, a prerequisite for finding knowledge to reuse and to apply, is poorly supported. For example, Wikipedia states the Pythagorean theorem as $a^2 + b^2 = c^2$ and files it into the categories "Articles containing proofs" and "Mathematical theorems" [189]. The LaTeX representation of the formulæ does not support search by functional structure.

Putting the fact aside that Wikipedia cannot search formulæ at all, a search for equivalent expressions such as $x^2 + y^2 = z^2$ or $c = \sqrt{a^2 + b^2}$ would not yield the theorem, unless they explicitly occur in the article. From the categorization it is neither clear for a machine (albeit very likely for a human) whether the article contains a proof of *that* theorem, nor whether it is correct. Similarly, the Polymath collaborators had to search previous publications of refutations of $P \neq NP$ "proofs" by keyword.

Formalized repositories such as the MML use specialized search engines [48]. They do support internal knowledge reuse by formalizing new mathematical concepts of existing ones and proving new theorems by applying ones that have already been proven, but they do not support links to external repositories. Thus, the maintainers of each knowledge collection, informal or formalized, hope to receive a critical mass of contributions that makes it sufficiently self-contained for the desired application.

Finally, the integration of mathematical Web 2.0 sites with automated reasoning and computation services is scarce. Interactive computation is available in mathematical e-learning systems, such as ActiveMath [32] or MathDox [142] – where document authors have sufficiently formalized the underlying mathematics in separate editing tools before publishing –, but less so in general-purpose digital libraries and collaboration environments. Mashups, which have otherwise been a driving force of Web 2.0 development, scarcely exist for mathematical tasks.[1]

1.2. Early Adoption of the Semantic Web in Mathematical Knowledge Management on the Semantic Web

In the early 2000s, when XML was increasingly used for mathematics, particularly formulæ (cf. section 3.2.1), the first building blocks of the Semantic Web vision, such as RDFS, approached standardization. These developments sparked interest in the emerging interdisciplinary mathematical knowledge management (MKM) community, consisting of computer scientists, computer-savvy mathematicians, and digital library researchers, whose objective is "to develop new and better ways of managing mathematical knowledge using sophisticated software tools" [93][2], or, more specifically, "to serve (i) mathematicians, scientists, and engineers who produce and use mathematical knowledge; (ii) educators and students who teach and learn mathematics; (iii) publishers who offer mathematical textbooks and disseminate new mathematical results; and (iv) librarians and mathematicians who catalog and organize mathematical knowledge" [93][1]. They hoped that Semantic Web technologies would help to address their challenges. This seemed technically feasible, particularly due to the common foundation of XML and URIs [140].

The two main lines of applying Semantic Web technologies to MKM focused on *digital libraries* – improving information retrieval and giving readers access to automated reasoning and computation services –, and *web services* – providing self-describing interfaces to automated reasoning and computation on the Web, so that they could solve problems sent to them by humans or other agents.

1.2.1. Digital Libraries – MathNet, HELM, and their Spin-Offs

Mathematical institutes participating in MathNet [6], an early effort to build "a distributed, efficient and user-driven information and communication system for mathematics" [76], were advised to put up uniformly structured homepages and publishing preprints and annotate both with RDF. Some of the 180 MathNet homepages that existed in 2002 [171] are still online; however, the central services, including a preprint search engine[2] and a browser for MathNet pages, have been out of order since 2007.

Independently, HELM, the Hypertextual Electronic Library of Mathematics [5,41], aimed at "integrat[ing] the current tools for the automation of formal reasoning and the mechanization of mathematics [. . .] with the most recent technologies for the development of web applications and electronic publishing" [5]. In contrast to MathNet and other traditional digital libraries, HELM intended to explicitly represent the fine-grained structures of mathematical expressions to expose them to, e.g., automated reasoners, but also to enrich their publication on the Web. For example, mathematical formulæ were rendered in MathML in such a way that actions could be performed on them, e.g. simplifying a selected (sub)expression using an automated reasoning backend attached to the library. HELM completely relied on XML and RDF not only for publishing, but also for its internal knowledge representation. Formalizations of mathematical statements and proofs were encoded in one XML dialect

[1] enumeration added by the author

[2] . . . which actually featured the first working implementation of Dublin Core [160]!

per underlying logical system; primarily, the library of the Coq higher-order proof assistant (cf. section 3.3) was used in HELM. Relevant structural properties, interrelations, and metadata were represented as RDF (cf. section 3.4).

The HELM developers had to make a lot of foundational research and development, as suitable reusable implementations were not available for many of the planned features. As none of the prototypical RDF query engines available in 2003 satisfied the HELM requirements[3], a new one was developed [110,108]. As browsers did not sufficiently support MathML, a MathML rendering widget suitable for embedding into desktop applications was developed [156].

1.2.2. Web Services – MONET and Related Architectures

The MONET project pioneered an architecture for mathematical web services built on Semantic Web technologies [146,68]. MONET services give access to numeric and symbolic computation systems; access to proof assistants or digital libraries was envisaged but not pursued. MONET services come with a machine-comprehensible description and can be registered with a central broker. Mathematical expressions in queries or computation requests to the broker were represented by their functional structure using OpenMath (cf. section 3.4). MONET also required foundational work to be done. OWL and the RACER reasoner were already found suitable for the internal description of services and problems and computing matches. However, the frontend XML languages for service descriptions and queries (which the broker then translated to OWL) had to be designed from scratch. Furthermore, the OWL reasoners of that time could not efficiently deal with a large number of instances (here: concrete problems instantiating problem descriptions), which required a specific database/reasoner hybrid to be developed, but then, again, the separate treatment of classes and instances constrained the design of the MONET ontologies in that they had to model every object as a class [69]. Part of MONET's query language are still used in the MathDox e-learning system [142,73]. More importantly, MONET and the competing MathBroker architecture for symbolic computation web services [49] influenced each other. The latter, however, made less use of Semantic Web service technologies. The MathServe architecture, influenced by both of the former but focusing on automated reasoning, made extensive use of more recent Semantic Web service technologies, such as OWL-S service profiles [191].

1.2.3. Critique – Frustration and Discontinuation

Semantic Web approaches to MKM have so far failed to fulfill the hopes set in them, the aftermath of HELM and MONET being an instructive example. Both groups of researchers were initially enthusiastic about the possibilities of the emerging Semantic Web, but then it turned out that few stable and reusable implementations existed, and hence a considerable amount of resources had to be invested into developing fundamental building blocks.[4]

After 2004, when the HELM and MONET activities had ended, the MKM community has given up using Semantic Web technologies on a large scale, and the Semantic Web community has focused on different application areas. It has been suggested that, after the pioneering phase, it was hard to obtain research funding for applications of Semantic Web technologies to MKM.[3]

From the MKM perspective, the further development was as follows: Parts of the HELM technology have survived in an interactive desktop proof assistant [42], whereas the web frontend and the RDF-based components have been discontinued. Contemporary mathematical web services work without Semantic Web technology. While large parts of the influential OpenMath community had been involved into MONET, which heavily relied on Semantic Web technologies, the current driving force of research symbolic computation web services, the SCIEnce project (Symbolic Computation Infrastructure for Europe [169]), does not use "standard" Semantic Web service technologies at all: SCSCP (Symbolic Computation Software Composability Protocol [112]) is a lightweight XML protocol using TCP sockets, or alternatively SOAP, whose communication semantics heavily relies on a custom OpenMath vocabulary.

1.3. Mathematics on the Semantic Web – Why Retry Now?

Web 2.0 applications are attracting an increasing number of working mathematicians. The usage of Semantic Web technologies to improve MKM has been investigated, albeit without becoming mainstream yet. This section argues why a new combination of Web 2.0 and Semantic Web technologies is needed to address them, and why such a solution is now feasible.

[3]personal communication with MICHAEL KOHLHASE on 2010-11-12

1.3.1. Combining Sem. Web and Web 2.0 for MKM

The combination of Web 2.0 and Semantic Web technology has already proven successful in some fields, including semantic wikis and Linked Data mashups [37]; however, it has hardly been applied to MKM yet. ZACCHIROLI gave two reasons why a hypothetical retry of the above-mentioned HELM would benefit from Web 2.0 technology [190]: Mathematical content would become editable directly on the Web, and projects like PlanetMath have proven that there is "a community of people interested in collaboratively authoring rigorous mathematics on the web" [190].[5] Furthermore, HELM or MONET would now benefit from a much wider availability of stable libraries and tools. With SPARQL, for example, there is now a standardized and widely supported query language for RDF.

1.3.2. What MKM can Contribute to the Sem. Web

Conversely, there are now also opportunities for MKM to give back to the Semantic Web. Mathematical semantics is needed to improve, or even enable, certain applications of Linked (Open) Data. Mathematics is an ubiquitous foundation of science, technology, and engineering; a concrete example on the Web of Data is given by statistical datasets. OMITOLA et al. have, for example, used datasets published by the UK government to answer queries for public sector information in the user's home region by aggregating data about, e.g., political representatives of the local constituencies, crime statistics for the local county, and waiting list statistics of local hospitals [153]. In statistical datasets, there are values that have been derived from ground values or from other derived values using mathematical functions. Planning data collection and interpreting collected data requires mathematical models. Therefore, statistical datasets need a notion of mathematical semantics.[6] That means, publishers of statistical datasets need a mathematical vocabulary. The quality of linked data vocabularies – often designed in an ad hoc mapping of existing database structures to RDF – and hence of the linked datasets is often low, as has been observed, e.g., by JAIN et al. [119]. ZIMMERMANN has observed the following reasons for vocabularies being of low quality [192]:

1. ontologies defining the domain of interest do not exist;
2. they exist but are difficult to find because developed by small groups for experimentation, lacking advertisement;
3. they exist and can be found but they are of poor quality, not complying with standards or best practices;
4. they exist and can be found but there are too many, of mixed quality, and it is difficult to assess which ones are appropriate for a specific use case.

High-quality machine-readable vocabularies for mathematics do exist: The official OpenMath 2.0 Content Dictionaries (CDs), for example, defining 260 mathematical symbols – operators, functions, sets, constants –, have undergone a strict human-driven review process (cf. section 3.2.2), and there are large machine-verified libraries of formalized mathematics (cf. section 3.3). Large parts of the MKM community accept them as standard vocabularies for representing mathematical expressions, but for the rest of the world – including the publishers and ultimately the consumers of linked data – ZIMMERMANN's criterion (2) applies. Besides a technical mismatch – they are not available as RDF[4] – there is a *cultural* mismatch. Mathematics, due to its practice of rigorously reasoning about abstract concepts in a self-contained way using a symbolic notation, is generally perceived as hard and inaccessible (see, e.g., [91]). The average computer scientist, whose work builds on a very restricted area of applied discrete mathematics, is not immune to such stereotypes. By integrating mathematics into the Web of Data, using the techniques explained in this article, we can take it out of the Ivory Tower.

1.3.3. Structure of this Article

This article reviews ontologies and languages that are suitable for contributing mathematics to the Semantic Web, particularly the Web of [Linked] Data. The remaining sections are structured as follows: Section 2 provides an abstract overview of the structures of mathematical knowledge and thus the background knowledge about the domain that is needed to assess the aptitude of existing ontologies and languages for representing mathematical knowledge adequately to the applications described above. Section 3 provides a comprehensive review and concludes with recommendations on what ontologies and languages should be used on the Web of Data. Section 4 explains techniques for integrating non-RDF representations, which are still ubiquitous in the MKM domain, into the Web of Data, using the ontologies reviewed. Section 5 tries

[4]That can be accommodated for, as will be explained in section 4.1.

to predict the benefits of that and points out further research directions.

2. Background: Structures of Mathematical Knowledge

Before we can represent mathematical knowledge on the Semantic Web, we have to understand its structures. Realistic applications, even in pure mathematics, do not only operate on logical and functional structures but also require information about non-mathematical aspects of the respective application scenario, about project organization and management, about discussions that authors and users hold *about* the mathematical knowledge, etc. There is little literature about these structures. Working mathematicians often use them without reflecting them. Computer scientists and knowledge engineers have to reflect them but often do so from the point of view of a system specialized for a particular task – e.g. checking first-order logic proofs – and its particular conceptual model and representation language. Thus, the review given here is influenced by literature on concrete systems, models, languages, and ontologies, but tries to abstract from that.

2.1. Logical and Functional Structures

Mathematical knowledge has a three-layered *logical* structure of objects – composed of symbols –, statements, and theories[5]. Symbols comprise operators, functions, sets, and constants. New mathematical concepts (i.e. symbols) can be defined, possibly based on concepts defined previously. A mathematical *object* can be a single symbol, or a compound, such as a complex number, an application of a function to arguments, or a derivative. Some of their properties are specified as axioms. Axioms are expressed as formulæ in a logical language, e.g. first-order logic (FOL). By applying rules of that logic, other properties of the mathematical concepts can be inferred. In a usual mathematical document, such properties are first asserted and then proven – or refuted. Often, the choice of what properties of a concept to model as axioms is arbitrary and merely follows established conventions. All kinds of properties of concepts are sometimes subsumed under the term *statements*. This is the case in the OMDoc representation language (cf. section 3.2.3), which distinguishes symbol declarations and axioms, definitions[7], assertions (theorems, lemmas, corollaries, etc.), proofs (which prove assertions by applying inference rules to axioms and previously proven theorems), and examples. Not all assertions in a realistic mathematical knowledge base have to be true: There can be conjectures whose truth is not yet known, as well as wrong assertions that have been refuted by counterexamples but are kept for instructive purposes. Groups of closely related symbols and their properties form *theories*. When reusing mathematical symbols, their names are often qualified by their theory for disambiguation, i.e. theories act as namespaces for symbols; this is also reflected by speaking of the "home theory" of a statement. For example, both the theory of real numbers and the theory of functions on real numbers have an "addition" operator. The latter can be defined pointwise in terms of the former, but both remain different; for example, one cannot use either of them to add a number to a function.

In the context of theories, statements can be distinguished more precisely into *constitutive* statements (axioms and definitions), which determine the meaning of a theory, and *non-constitutive* ones, such as theorems or proofs, which "only illustrate the mathematical objects in the theory by explicitly stating the properties that are implicitly determined by the constitutive statements" [124, chapter 15.1]. Moreover, the logical language used to express the statements in a theory can itself be modeled as a theory, then called *meta-theory*. For example, the theory of commutative groups can be formalized with FOL as a meta-theory. FOL provides the universal quantifier that is needed for stating the axiom of commutativity as $\forall a, b \in G. a \circ b = b \circ a$.

For knowledge management tasks, which involve, e.g., reuse of theories or management of theory changes, it has been found useful to build theories on a minimal set of axioms and model a whole field of mathematics as a strongly interconnected graph of "little theories" reusing each other (cf. [92]). The connections are called *theory morphisms* or *views*. Some of them are given by definition – then called *imports* –, others are postulated and then have to be proven. This use of theories extends beyond mere namespacing and has particularly been adopted for the structured specification and verification of software (see, e.g., [45]).

Logical/functional structures can be expressed at different *levels of formality*: Often, an author starts a document by sketching a few formulæ and some textual notes. Later, the content is elaborated both into the formal and into the informal direction: A sloppy for-

[5] reusing the terminology introduced by KOHLHASE in [124, chapters 2.3 and 3.2] and refined in [122]

mula is written more rigorously, rigorous text is formalized, taking previously formalized knowledge into account, and natural language explanations are added to formalized knowledge (see, e.g., [124, chapter 4]). Both directions can, in principle, be automated: *Natural language processing* techniques can aid formalization (see, e.g., [96]), whereas proof explanation helps to generate natural language from formalized knowledge (see, e.g., [94]). These solutions, however, can not yet cope with the full complexity of mathematical knowledge as it occurs in practice. Particularly the automated disambiguation of symbolic notation (see below) is hard, as the surrounding text often has to be taken into account for disambiguation [96].

One aspect that is not restricted to logical/functional structures but has been investigated most deeply for them is *dependency*. For the purpose of managing mathematical knowledge, dependency can be defined such that B depends on A in the way d_p iff a change to A may have an impact on the property p of B. To make this definition precise, one has to fix the property p. Different conceptual models and representation languages have done that in different ways. The formal language MMT, which constitutes a subset of the above-mentioned OMDoc, considers dependency w.r.t. logical well-formedness [162, chapter 8.4]; examples for that will be given in 3.4.2. The MathLang language (cf. section 3.2.4) considers dependency w.r.t. the reader's ability to understand a knowledge item [165,120].

2.2. *Rigorous Language and Symbolic Notation*

Mathematics has developed its own style of natural language (cf. figure 1 and [193,177]). General models for representing the discourse structure of a text, such as Rhetorical Structure Theory (RST [139]) can also be applied. RST divides a text into spans, often down to the level of subordinate clauses. RST has a rich vocabulary of relations between *nuclei* (essential text spans) and their *satellites* (spans that provide additional information); for example, a satellite can give evidence to a nucleus, provide background information to facilitate understanding, or define the context in which the nucleus is to be interpreted. Figure 1 models a phrase from figure 2 according to RST.

For sections and chapters on the upper levels of a document, several models of discourse in scientific publications have introduced more convenient coarse-grained blocks that correspond to the usual sections of a publication, and reserve RST for fine-grained markup [107]. A typical document in one of these models starts with an abstract and a motivation and ends with a conclusion and a list of references, and has some sections in between that provide background knowledge, explain the actual contribution of the paper, demonstrate practical applications, summarize the results of experiments or evaluations, review the state of the art and related work, etc.

Mathematical formulæ employ a two-dimensional notation, whose complexity is owed to the possibility to define new symbols at will. Choosing an intuitive notation for the concepts dealt with is of great importance to understanding and communication (see, e.g., [161]). The notation of a symbol is usually introduced with its first declaration, typical phrases being "We will denote by Z the set …", "The notation aRb means that …", etc. [177]. Notation can be conceived as a one-to-many mapping of structures of mathematical knowledge – primarily logical/functional structures – to an arrangement of glyphs on paper. The notation chosen for a particular object in a particular document is determined by a number of presentation context dimensions (examples taken from [144] and [149]):

language and culture: the French/Russian notation of the binomial coefficient $\mathcal{C}_n^k$ vs. the German/English notation $\binom{n}{k}$; see [138] for details

level of expertise: the explicit notation of multiplication as $a \cdot b$, which is common in primary school, vs. the more advanced omission of the operator symbol in the notation ab

area of application: The square root of -1 is written as i in most fields, whereas electrical engineers write it as j to distinguish it from the current I.

community of practice: People with a set theory background tend to include 0 in the set of natural numbers $\mathbb{N}$, whereas people with a number theory background tend to start with 1.[8]

individual preference: Some mathematicians, who prefer completely idiosyncratic notations when working on their own, translate other articles into their own notation and translate their own articles to a more conventional notation before publication [116, pp. 166–167].

The greatest notational variety has been observed for mathematical symbols. From the level of statements upwards, notation – such as the font chosen for keywords like "Definition" – is more standardized and therefore usually not a subject of research. However, *which* definition of the same concept, which proof for

Let M be	Suppose that ...... Assume that ...... Write	. Then ......,	provided $m \neq 1$. unless $m = 1$. with g a constant satisfying ......

Fig. 1. Typical phrase patterns for theorems [177]

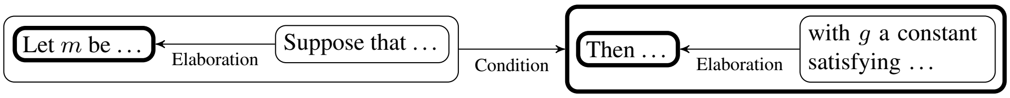

Fig. 2. RST markup of a theorem (nuclei with thick outline)

the same theorem, which example for the same thing, i.e. which *re*presentation of a thing to present to a user – these questions do depend on context (see, e.g., [123] on the context-sensitivity of examples, and [149] on generating documents from snippets using contextual information).

2.3. Mathematics-Specific Metadata

In an expressive representation of mathematical knowledge, it is hard to draw a line between data and metadata. This article, in line with prior work on metadata in MKM [98], considers all of the previously mentioned structures *data* of primary interest, whereas the remaining, mainly administrative and application-specific information will be considered *metadata*.

Metadata can be embedded into the data they describe, or point to the data from outside ("standoff markup"). This section focuses on metadata that are so closely related to the data that embedding them makes most sense in the interest of uniform management workflows. In mathematical practice, subjects annotated with metadata can be very fine-grained, as the following blackboard-style example shows:

$$b^{-1}((\boxed{a^{-1}a})b) = b^{-1}(eb) = \ldots$$

↖*We learned that last week*

Administrative metadata describe the lifecycle and revision history of a resource, the data format and the usage requirements, copyright information, as well as other general-purpose information. The most widely used metadata vocabulary is Dublin Core [22], which covers general bibliographical information, but also elementary licensing and versioning information. Its semantics is rather weak, but it is widely supported, e.g. by search engines. The Dublin Core Metadata Element Set [89] provides a basic vocabulary, which the more modern DCMI Metadata Terms vocabulary extends in a backwards-compatible way [83]. Dublin Core can be complemented by domain-specific classification schemes, whose entries are usually alphanumeric codes. For mathematical publications, the MSC (Mathematics Subject Classification [31]) prevails; this article would be classified as 68T30, where 68 is computer science, 68T is artificial intelligence, and 68T30 is "knowledge representation". GAMS, the Guide to Available Mathematical Software [4], classifies more fine-grained things, namely mathematical problems, for example, H2a1 = "one-dimensional finite interval quadrature".

Further metadata vocabularies cover the settings in which mathematical knowledge is applied. For example, the Learning Object Metadata (LOM [117]) describe educational properties of resources, such as their level of difficulty or interactivity, their coverage of topics (e.g. in terms of classification systems), and their intended audience. A vocabulary inspired by LOM has been used in the mathematical e-learning system ActiveMath [144].

Again, the application environment can be modeled as data rather than metadata, given an appropriate conceptual model of the application domain. Two notable examples are SWEET and GeoSkills. The SWEET OWL ontology (Semantic Web Earth and Environmental [19]) describes 4600 concepts in 150 modules from fields related to mathematics, such as physics, chemistry, biology, geology, and astronomy. These modules build on a foundation of general concepts of mathematics (e.g. functions), natural science (e.g. units), and space (e.g. coordinates). SWEET's model of mathematics does not intend to be as elaborate as the structural ontologies reviewed here, but SWEET provides a good showcase of how to integrate knowledge about mathematics with knowledge about its scientific application domains. One example of how SWEET integrates mathematics and science is the concept of a gravity field, defined as a vector field whose force is gravity. A vector field is a subconcept of a function whose result is a vector is defined as an array of scalar elements. GeoSkills is an OWL ontology describing topics, competencies and educational contexts related to interactive geometry [137], albeit without a connection to a structural model.

2.4. Discussions in Mathematical Collaboration

Previous research has produced ontologies for two kinds of scientific discourse: One kind is embedded into scientific publications, which, e.g., make claims and argue about claims made in other, cited publications. This has often been studied in combination with rhetorical structures; see [107] for an overview.

The other kind of scientific discourse is held externally of the representations of its subjects, e.g. in discussion forums. This perspective has been studied in the context of collaborative problem solving; the most common models employed in knowledge engineering have been derived from IBIS (Issue-Based Information System [126]). The DILIGENT argumentation model has been developed in the context of the namesake collaborative ontology engineering methodology with the design goal of making arguments more focused than in plain IBIS in order to make design decisions more traceable [175,176]. A DILIGENT argumentative thread starts with raising an issue, e.g. verbalizing a requirement for the ontology to be designed or pointing out a problem with its current state. An issue can be resolved by implementing a proposed and approved solution idea. About issues and ideas, the participants can state objective arguments or their subjective position.

The generic notion of issues and ideas can be refined to capture domain-specific problem and solution types. The most common problem with items of mathematical knowledge in knowledge collections, as reported by the 25 participants of a survey that we have conducted among domain experts, is that they are wrong, followed by being incomprehensible, their truth being uncertain, being underspecified, or redundant [135]. Further cases include knowledge items of which it was not clear whether they were useful, and knowledge items expressed in an uncommon style. Problems are most commonly solved by directly improving the affected knowledge item, by splitting it into more than one, or by deleting it altogether. Knowledge items, issues, and ideas cannot be combined arbitrarily. For example assertions, proofs, and examples can be wrong, whereas a notation can rather be inappropriate, misleading, or hard to read and write. Then, if some knowledge item is wrong, it could be deleted, or fixed in place, or kept as an instructive bad example, whereas splitting it into two parts would not solve that problem.

Combining both perspectives on discourse remains to be done for mathematics but would allow for capturing further important mathematical practices. In his work on "Proofs and Refutations" [127], LAKATOS has studied how discussions about mathematical knowledge items materialize into new mathematical knowledge. Consider a discussion thread in which a problem with a proof is pointed out, e.g. that it only covers a specific case and should be generalized. This discussion provides the rationale for a later, generalized restatement of the respective theorem and its new proof and therefore could be integrated into the text that encloses the theorem and its proof.

3. Review of Languages and Ontologies

This section reviews existing languages and ontologies for representing mathematical knowledge. Besides Semantic Web ontologies in the narrow sense, other machine-comprehensible representation languages are taken into account. This is because they are widely used in science, technology, engineering, and mathematics, even preferred over ontologies in certain settings, and most existing machine-comprehensible mathematical knowledge is available in these languages rather than RDF. We do, however, pay attention to the possibility to translate such representations to RDF.

3.1. Requirements for Representing Mathematical Knowledge on the Semantic Web

From the review of the state of the art of mathematics on the Web, we can infer as design goals for Semantic Web applications for MKM the ability to reuse knowledge across knowledge bases, information retrieval adequate to the structures of the knowledge, integration with mathematical services, such as automated reasoning and computation, without compromising comprehensibility for human end-users. Having reviewed the structures of mathematical knowledge in section 2, we are now ready to specify more precise requirements for representation languages[6]:

[6]Capitalized keywords are used in accordance with RFC 2119 [64].

S: All of the previously reviewed structures of mathematical knowledge SHOULD be supported; where this is impossible, missing dimensions MUST be compensated for by language extensions along the criteria L.E and L.→ below. We subdivide this criterion as follows:

- **S.L.{O,S,T}:** logical/functional structures: mathematical objects, statements, theories
- **S.R:** rigorous language or rhetorical structures
- **S.N:** notation
- **S.M:** metadata
- **S.D:** discussions

F: Mathematical knowledge occurs in different degrees of formality; applications targeting human users and automated agents require both formal and informal representations. Therefore,

- **F.R:** the language SHOULD be able to represent knowledge in a wide range from informally to fully formalized, and
- **F.C:** many degrees of formality SHOULD be able to coexist in one document, interlinked with each other.

L: In real-world applications, mathematical knowledge is combined with multiple dimensions of non-mathematical knowledge. Therefore, a language SHOULD support interlinking of these dimensions by rich annotation facilities, but also give authors the freedom to represent some knowledge by external means and link it to representations in the given language. In detail,

- **L.A:** the language MUST allow for attaching non-mathematical metadata and annotations to mathematical knowledge items, regardless of their granularity, and
- **L.→[7]:** for linking mathematical knowledge items to external mathematical or non-mathematical resources, and
- **L.←[8]:** it MUST be possible to address all mathematical knowledge items expressed in the given language from outside, in order to link external representations to them, for example standoff markup or existing representations in different languages.

C: Knowledge represented in a language SHOULD be comprehensible

- **C.A:** to arbitrary automated agents – therefore, the knowledge SHOULD be self-describing in a machine-comprehensible way.
- **C.H:** to human users – therefore, published human-comprehensible documents generated from representations in the given language SHOULD retain semantic annotations, so that assistive services can retrace the original knowledge and make it available to the user on request, e.g. integrated into a user interface.

[7] read "L out[going]"

[8] read "L in[coming]"

Tables 1 and 2 at the end of this section summarize how well the languages and ontologies reviewed satisfy the requirements listed in the previous section.

3.2. XML-based Semantic Markup Languages

XML languages share the URI foundation with RDF. All semantic XML languages allow for assigning IDs for the inner nodes of their tree-structured representation (i.e. for the *elements*), via XML ID [141]. Together with the URI of the XML document, that allows for global identification and linking and thus satisfies requirement **L.←**. XPointer [101] allows for identifying more complex subsets of an XML representation, such as node or text ranges, but is rarely supported. Note, however, that additional work is required to integrate semantic XML markup with RDF-based Linked Data; this will be discussed in section 4.1.

3.2.1. Content MathML 3 and OpenMath 2 Objects

MathML (Mathematical Markup Language [44]) is an XML language that was originally conceived for embedding mathematical formulæ into HTML web pages. It features a presentation-oriented sublanguage (Presentation MathML) but also a semantics-oriented one (Content MathML); the latter will be covered here. MathML allows for mixing semantic and presentation markup, even including embedded or linked non-mathematical information. However, the task of defining notation, i.e. mapping semantic structures to a human-readable presentation, is left to other, non-mathematical languages such as XSLT.

The related OpenMath [67] language has originally been invented in the mid-1990s to facilitate data exchange between computer algebra systems (CAS) but has then been aligned closely with Content MathML, leaving only syntactic differences in the latest versions of both languages [79].

Both languages are limited to representing the functional tree structure of mathematical objects but are often integrated into host languages that cover further structures. Many XML languages already embed MathML or OpenMath officially (see below), whereas others allow for extending their vocabularies accordingly. Listing 1 shows a document that contains MathML and OpenMath objects.

Mathematical objects consist of numbers, variables, symbols, and applications of objects to other objects. Content MathML comes with a default supply of symbols that cover high school and introductory university education. Mathematical objects reference these symbols by URI. Their semantics is defined in external vocabularies called *OpenMath content dictionaries* (CDs); authors can create and use additional CDs as needed. The machine-comprehensibility of MathML/OpenMath representations depends on the degree of formality of the CDs. Suggested alternative RDF representations of MathML will be discussed in section 3.4.2.

3.2.2. Extensible Mathematical Vocabularies: OpenMath 2 Content Dictionaries

An OpenMath CD is a collection of (usually closely related) definitions of symbols. The abstract model of a CD covers mathematical objects – used to formally represent properties of a symbol, as opposed to plain text descriptions –, a weak variant of axioms/definitions and examples on the statement level, and a basic notion of theories, plus a limited metadata vocabulary [67, section 4]. The reference encoding of that model is a lightweight XML language, roughly comparable to RDFS in expressivity; the OMDoc language (cf. section 3.2.3) offers a more expressive alternative. Alternatively, the model has been implemented by ontologies (cf. section 3.4.2).

The *official* CD collection reviewed by the OpenMath Society (cf. [67, section 4.5]) defines 260 symbols from arithmetics, set theory, FOL, algebra, calculus, as well as transcendental and statistical functions [23]. These CDs do not provide a full formalization; instead, developers of, e.g., CAS are supposed to use the CDs as specification manuals when implementing *phrasebooks*, which translate OpenMath objects into the native languages of such systems.

3.2.3. More Expressive CDs and Documents: OMDoc 1.3

OMDoc (Open Mathematical Documents [12,124]) is an XML language for representing mathematical knowledge that has a particularly rich vocabulary for logical/functional structures. It supports MathML and OpenMath objects and adds vocabulary for formal and informal statements, modular theories, narratively ordered documents, rhetorical structures, notation definitions, and hosts RDFa [33] for arbitrary metadata and links [132,133]. On each structural level, OMDoc supports a wide range of degrees of formality, from unstructured text to a full formalization – thus eliminating the need for phrasebooks at least on object level –, which can coexist in a literate programming style to serve the needs of human- and machine-oriented applications.

Listing 1 gives an example of the syntax of OMDoc 1.3, whose specification is currently being finalized. An OWL ontology covering a large subset of OMDoc will be reviewed in section 3.4.2.

OMDoc has been used for exchanging knowledge between systems doing structured specification, automated verification, and interactive theorem proving, for documenting Semantic Web ontologies [133], for publishing human-readable documents for interactive browsing [81] and adapted to different audiences [149], and for e-learning in the ActiveMath system mentioned before.

3.2.4. Narrative Documents: MathLang

MathLang [121] is similar to OMDoc but puts an even higher emphasis on formalization of informal, but highly conventionalized mathematical text. From its structural annotations, “proof skeletons” can be generated, i.e. templates in languages for formalized mathematics [121]. MathLang compares to OMDoc in its coverage of object- and statement-level logical/functional structures, narrative document structures, and literate-programming-style combinations of text and formalizations. However, there are no theory level and no rhetorical structures, the metadata vocabulary is restricted, and links to non-mathematical knowledge are not supported. MathLang has an XML encoding that is used for most processing tasks except authoring and presentation.

MathLang’s “Document Rhetorical aspect” (DRa), which, despite the name, does not cover rhetorical structures but rather statement-level logical structures and narrative document structures, has also been implemented as an OWL ontology (cf. section 3.4.2).

3.2.5. Languages for Books and Manuals

A formal view on technical specifications reveals structural similarities to mathematical theories. While fully formalized specifications can be written in the same languages as general formalized mathematics (cf. section 3.3) and then be verified automatically, there are different XML languages targeting human audiences, such as engineers implementing a specification, or developers using an API; DocBook, TEI, and DITA will be reviewed here. Similar languages ex-

Listing 1: An OMDoc theory with a declaration (type given in OpenMath) and implicit definition of the exponential function (given in Content MathML)

```
<theory xml:id="transc">
 <imports from="sts#sts"/>                                   <!-- Small Type System -->
 <imports from="setname1#setname1"/>        <!-- numbers and other basic sets -->
 <symbol name="exp">
  <meta property="dc:description">the exponential function</meta>
  <type><!-- ℝ → ℝ -->
   <om:OMOBJ>
    <om:OMA>          <!-- OMA applies a constructor or function to arguments -->
     <om:OMS cd="sts" name="mapsto"/>
     <om:OMS cd="setname1" name="R"/>
     <om:OMS cd="setname1" name="R"/>
    </om:OMA>
   </om:OMOBJ>
  </type>
 </symbol>
 <definition xml:id="exp-def" for="exp" type="implicit">
  <CMP>The exponential function equals its derivative and evaluates to 1
   for an argument of 0.</CMP>
  <FMP><!-- exp′ = exp ∧ exp(0) = 1 -->
   <m:math> <!-- here, we use the symbol vocabulary built into Content MathML -->
    <m:apply>  <!-- as an alternative to explicitly referring to imported CDs -->
     <m:and/>                                <!-- equivalent “strict” markup: -->
     <m:apply>                     <!-- <m:csymbol cd="logic1">and</m:csymbol> -->
      <m:eq/>
      <m:apply>
       <m:diff/>
       <m:csymbol cd="transc">exp</m:csymbol>
      </m:apply>
      <m:csymbol cd="transc">exp</m:csymbol>
     </m:apply>
     <m:apply>
      <m:eq/>
      <m:apply>
       <m:csymbol cd="transc">exp</m:csymbol>
       <m:cn type="integer">0</m:cn>
      </m:apply>
      <m:cn type="integer">1</m:cn>
     </m:apply>
    </m:apply>
   </m:math>
  </FMP>
 </definition>
</theory>
```

ist for e-books and courseware; we review EPUB/DTBook and CNXML/CollXML.

None of these languages is directly suitable for representing mathematical knowledge other than objects. All of them support MathML either natively or as an extension, some have limited built-in support for statement-level logical structures, none supports theories. We first review the individual languages, then discuss further extension possibilities. Generally, these languages do not have a machine-comprehensible semantics, and the publication tools available for them do not support generating semantically annotated human-comprehensible documents.

DocBook 5: DocBook, the most widely used XML language for technical manuals [181,182], focuses on representing structures pertinent to its main application area of software documentation. Besides MathML objects, titled equations, and examples, DocBook does not support mathematical structures, and it has a fixed idiosyncratic metadata vocabulary with a coverage similar to Dublin Core. It hardly has native markup for representing knowledge in different degrees of formality and interlinking such representations. Embedding arbitrary literal-valued metadata into a document is not supported.

A notable application of DocBook in MKM is the MathDox e-learning system [142,75], whose compound document format combines DocBook with OpenMath objects and further XML vocabularies for programming constructs, requesting user input, queries to MONET services (cf. section 1.2.2), and exercises.

TEI P5: TEI (Text Encoding Initiative [65]) is, due to its focus on digitalization and edition of paper-born documents from the humanities, social sciences, and linguistics, obviously not suited for representing *mathematical* knowledge. Nevertheless, it is a prime example of an expressive semantic markup language. The TEI guidelines recommend using any available representation language for mathematics, according to the requirements, and explicitly mentions MathML, OpenMath, even OMDoc [65, chapter 14.2]; a combined TEI+MathML XML schema is available. In its own domain of literature, TEI can express knowledge in a wide range of degrees of formality. It supports fine-grained interlinking of different representations of the same knowledge. TEI has an elaborate but finite metadata vocabulary for representing the provenance of documents and even smallest fragments of text. Arbitrary additional information can be embedded into a document, or provided as standoff markup pointing into the original document, whereas linking to external resources is restricted in that no link types are supported. Certain sub-vocabularies of TEI have been given a formal semantics by mapping them to relevant domain ontologies [20,155].

DITA 1.1: DITA (Darwin Information Typing Architecture [86]) does not support any particular application scenario by default but is rather intended to offer a framework for developing languages for topic-based technical documentation that are specialized to a particular domain of application.[9] The topic paradigm is in contrast to DocBook's focus on contiguous, narratively ordered manuals. DITA can be extended by MathML and supports [definitions of] concepts and examples. DITA's built-in metadata vocabulary primarily focuses on the context in which an object can be (re)used, such as the intended audience or keywords. DITA performs as badly as DocBook w.r.t. the **S** and **F** requirements, except that topics can be interlinked. However, DITA offers stronger support for adding arbitrary metadata and links.

EPUB 2.0.1 and DTBook 3: EPUB is a standard for general-purpose e-books, not primarily technical manuals. A complete e-book is a bundle of content files with a Dublin Core metadata record [11,13]. Besides XHTML – which could carry RDFa –, DTBook (DAISY[10] Digital Talking Book [1]) is the recommended format for them. DTBook is a semantically structured format inspired by DocBook but simpler and with less support for customization. RDFa support is planned for the next versions of EPUB and DTBook.

CNXML 0.7 and CollXML: CNXML, the language of the course modules of Connexions (cf. section 1.1.1) [25], is comparable to a subset of DocBook in expressivity. CNXML recommends using Content MathML for mathematical objects and supports definitions, "rules" – comprising, e.g., axioms and theorems – and examples; thanks to its educational focus, it also supports exercises [164].

Course modules written in CNXML are combined into collections represented in the CollXML container format [25]. CollXML was preceded by a partial representation of a collection's structure in RDF [18]. A CollXML document models dependencies between modules – so-called "featured links", which can be

[9] That is where the reference to CHARLES DARWIN comes from.

[10] Digital Accessible Information Systems

of type “prerequisite”, “supplemental”, or “example”, in three degrees of strength. There is an idiosyncratic metadata vocabulary with a coverage similar to Dublin Core.

Extensibility Towards Further Mathematical Structures: There are two principal approaches to introducing further mathematical structures: (i) literally reusing elements of sufficiently expressive mathematical markup languages, such as OMDoc, or, (ii) reusing an appropriate ontology for mathematical structures (cf. section 3.4) – provided that the host language supports referencing arbitrary metadata vocabularies on any relevant structural level without first introducing new container elements for them via approach (i), i.e. if there is an RDFa-like infrastructure (cf. section 3.4.1).

Approach (i) works in DocBook, TEI, and DITA; via that extension path, DUCHARME has proposed integrating RDFa into DocBook and DITA [90]. DocBook, TEI, and DITA offer varying degrees of support for approach (ii). There is a workaround for adding RDF-compatible annotations to DocBook: Any DocBook element can carry XLink attributes, which can have a role (= predicate), and from which RDF can be harvested [77]. TEI documents can reference external objects by XPointers, but without any possibility to specify a predicate type; thus, is does not allow for harvesting RDF. DITA offers the *othermeta* element for arbitrary key–value pairs, for which URIs could be used to emulate RDF, or, even more appropriately, the *data* element, which allows for constructing nested data structures and supports RDF’s distinction of URI- and literal-typed object, as well as datatypes. Similarly, DITA supports links with arbitrary roles from topics to related topics.

3.3. Languages for Formalized Mathematics

Languages for formalized mathematics, such as those of the proof assistants Mizar [10], Isabelle [183], or Coq [26], are of interest for us insofar as they also support informal content. They obviously support logical/functional structures of mathematical knowledge on the object and statement levels, but less so on the theory level (cf. [162, chapter 1.3]). Symbols and statements have identifiers, which are not compatible with URIs; however, for exchange purposes, the systems often offer an XML export (cf. [162]). Except for notation definitions, there is little support for other structures; the “lowest common denominator” is to put such information into comment lines, which are post-processed by a specialized tool. Isabelle and Coq formalizations can be interspersed with informal text, and certain parts of the formalized content can be marked as hidden for human-readable output. In Isabelle, informal text can contain formalized expressions as antiquotations, which the proof assistant evaluates when exporting the document [183, chapter 4].

These languages do not support links out of or into formalizations. Each language comes with its own set of services that understand formalizations in the respective language, which have a strong model- or proof-theoretic semantics for logical and functional structures. These languages are usually committed to a particular first- or higher-order logical foundation and therefore generally hard to translate into other languages for reuse. Existing translations have usually been hard-coded for a pair of two specific languages (cf. [162, chapter 1.1.3.3] and the Hets system [147,148]).

Each of the libraries that ship with the above-mentioned systems comprises several hundreds of theory files (cf. [184] for exact figures) of a very high quality: Firstly, it has been estimated that fully formalizing something takes an author about ten times as long as writing it down in rigorous textbook style; secondly, these formalizations have been machine-verified. These libraries usually have a good coverage of discrete mathematics; for example, Isabelle’s library covers elementary number theory, algebra, set theory, but also analysis. In the Flyspeck project for developing a machine-verified proof of the Kepler Conjecture[9], which employed several proof assistants, most of the required formalizations of trigonometry, geometry, topology, measure theory, etc., first had to be developed by the members of the project [111]. After HELM, no serious effort has been undertaken to fully integrate such libraries into the Semantic Web.

3.4. Structural Ontologies for Representing Mathematical Knowledge in RDF

While the languages reviewed so far are machine-comprehensible in their own ways, they do not integrate into the Semantic Web without translation to RDF (cf. section 4). Representing mathematical knowledge in RDF not only makes it accessible to Semantic Web agents, but also offers powerful ways of interlinking mathematical and non-mathematical knowledge, formal and informal representations, etc. RDF, when published in compliance with the Linked Data principles [56], is always machine-comprehensible in the sense that a machine can simply retrieve information about resources by dereferencing URIs. However, the informative value of the latter information, and the power of RDF in general, stands and falls by the availability of appropriate vocabularies, i.e. ontologies.

This section reviews ontologies that allow for representing mathematical knowledge natively in RDF, or that offer themselves as translation targets for knowledge originally represented in non-RDF languages. The review includes obsolete ontologies insofar as aspects of their design are still instructive today.

3.4.1. Different Approaches to Representing Mathematical Knowledge in RDF

Usually, when representing knowledge in RDF, one finds or develops an appropriate vocabulary and chooses a suitable RDF serialization, e.g. RDF/XML or XHTML+RDFa. In the presence of mathematical objects, this decision becomes harder due to their inherent complexity. Therefore, we will briefly discuss possible representations before reviewing concrete ontologies.

Complete RDF Representations: Due to their n-ary ordered tree structure, mathematical objects are not amenable to a straightforward representation as RDF triples. With the narrative order of (not only) mathematical text, one faces a similar challenge. The use of linked lists or ordered sets, either the collections or sequences built into RDF [52] or self-made remakes, is unavoidable. However, such data structures are not generally supported by RDF software, and they do not go well along with DL reasoning[10] and querying[11]. The N3 Vocabularies reviewed below demonstrate this approach for mathematical objects, the SALT ontology for rhetorical structures.

Mathematical Objects as XML Literals: Compared to RDF triples, XML offers a much more intuitive representation of n-ary ordered trees. With Content MathML and OpenMath, there are standardized semantic XML representations of mathematical objects, which are widely understood by mathematical software (e.g. CAS phrasebooks). Therefore, reusing them as XML literals of *rdf:XMLLiteral* datatype while representing other structures of mathematical knowledge as RDF triples suggests itself (cf. the OpenMath CD ontology in section 3.4.2 for an example). From a Semantic Web perspective, this has, however, the drawback that XML literals are largely opaque to contemporary RDF tools. The Virtuoso triple store [154] allows for filtering XML literals matched by a SPARQL graph pattern by XPath node tests [53]. The Corese RDF engine can additionally reuse variables from the proper SPARQL part of a query in XPath expressions [74]. None of these extensions has made it into the SPARQL standard yet.

Embedding RDFa into XML: RDFa is a set of XML attributes for embedding RDF graphs into X(HT)ML documents [33]. That allows for focusing on those structures that can easily be represented in RDF, while leaving the representation of n-ary and ordered structures to XML. However, queries that need both kinds of information have to be implemented separately. The upcoming RDFa 1.1 API [172], which remains to be implemented by browsers, will at least give in-browser scripts similar means of accessing embedded RDF as the Document Object Model (DOM) offers for XML. The XSPARQL [34] query language combines SPARQL and XQuery; however, such a query would still rely on a separate service that makes the RDFa annotations available as queryable RDF.

The first languages that officially hosted RDFa were the presentation-oriented XHTML and SVG languages [36], which allow human-comprehensible documents to carry as much semantic annotation as needed by agents, such as assistive services. RDFa can also be embedded into semantic markup languages; that has been done for OMDoc [132,133]. MathML has supported fine-grained annotation of presentational or semantic markup long before RDFa, with a similar expressivity (e.g. `<annotation definitionURL="link-type" src="link-target"/>`). OpenMath has a similar annotation syntax, albeit without URI support. Neither the MathML nor the OpenMath developers are currently planning to support the RDFa syntax. When using RDFa in semantic markup, one has to take care that the RDFa annotations do not interfere with the native semantics of the host language.[12]

Standoff Markup: Finally, one can maintain parallel representations of the same concepts both in RDF and in one of the specialized languages reviewed above. In such a setting, the RDF graph acts as standoff markup pointing to fragments of the other representation and adding information to them, such as additional metadata, links, or semantic abstractions not supported by the original language. Conversely, information about n-ary structures and order would only be represented in the latter language. Most of the knowledge is usually represented redundantly in RDF and the other lan-

Listing 2: The Pythagorean Theorem in N3

```
:ABC :side1 3 ; :side2 4 .

{?triangle :side1 ?a ; :side2 ?b .
 ?c is math:exponentiation of
  (((?a 2)!math:exponentiation
    (?b 2)!math:exponentiation)
   !math:sum 0.5) . }
=> { ?triangle :side3 ?c } .
```

guage – one of them possibly generated by automatic translation from the other one – to provide a maximum amount of information to agents that only understand one representation. This has so far been the most common approach in MKM (cf. section 3.4.2).

3.4.2. Logical and Functional Structures

Few approaches to completely representing logical/functional structures of mathematical knowledge in RDF have been made so far. A larger number of ontologies exists for representing mathematical statements, whereas the theory level has rarely been covered so far. The ontologies reviewed in this section have most commonly been used in standoff markup for XML representations.

N3 Vocabularies and RDF Encodings of Content MathML: The cwm [54] and Euler [84] reasoners natively use the N3 [55] superset of RDF. The standard N3 vocabularies cover a limited subset of object- and statement level structures, constrained to FOL as a meta-theory. Beyond domain knowledge, i.e. a library of basic mathematical functions (cf. section 3.4.6 for details), the N3 "math" vocabulary provides weak formalizations of general structural concepts such as the concept of a function. When a concrete function f is used as the predicate of an RDF triple, whose subject is a collection `(x1 ... xn)` holding the arguments, the reasoner infers $f(x_1, \ldots, x_n)$ as the value of the object. When the object is identified by a URI or blank node ID, it can be reused in the subject of another mathematical expression. Listing 2 shows a sample set of facts and rules yielding `:ABC :side3 5`. Few RDF processors support the full N3 syntax. When an N3-aware processor is not available, the n-ary ordered tree structure or mathematical formulæ has to be broken down into explicit RDF triples. Combining RDF reification and N3's "reason" vocabulary, which models the structure of proofs, allows for partially capturing the statement level. The coverage of the N3 vocabularies is determined by the needs of a FOL reasoner and thus not suitable for representing *arbitrary* mathematical knowledge. The semantics of mathematical functions is not fully specified in N3; cwm and Euler merely have built-in support for evaluating them.

Two RDF encodings of Content MathML have been suggested independently from N3. These representations look similar to N3, except that the application of a function is usually modeled with the [reified] application being the subject and the function symbol and the arguments being the object(s). That makes nested expressions easier to write without the additional syntactic sugar of N3. An encoding proposed by MARCHIORI [140][13] has obvious design flaws – such as an inconsistent way of modeling applications and referencing symbols in CDs –, which another, similar representation independently developed by ROBBINS avoids [166]. Both suggestions have neither been implemented nor been taken up by the MKM community.[14]

As an advantage of representing formulæ in RDF, MARCHIORI points out that it allows for making references to bound variables more explicit: Indeed, a bound variable is always represented as a unique RDF resource, be it on declaration or on usage. Content MathML, however, optionally supports a similar explication by making occurrences of the bound variable refer to the place where it is declared via *@xref* and *@id* attributes. MARCHIORI developed an ad hoc vocabulary from the Content MathML element and attribute names, which has little value from a Linked Data perspective. ROBBINS only uses a special vocabulary for the object constructors of Content MathML but the canonical OpenMath CD URIs (e.g. `http://www.openmath.org/cd/arith1#plus`) for symbols. The latter are compatible with Linked Data, as will be explained in section 4.2.

Ontologies for OpenMath CDs: In an early phase of the above-mentioned MONET project, an RDFS vocabulary for representing OpenMath Content Dictionaries (CDs) was developed [66]. The RDFS vocabulary covered logical/functional structures on the theory and statement levels, as well as metadata, by classes and properties, and represented mathematical objects as XML literals.

More recently, we have developed a more expressive OWL ontology [129], which covers more of the theory and statement levels. However, its representation of mathematical objects only covers flat occurrences of symbols, following the approach of the OMDoc ontology that will be explained below.

MONET Problem Ontology: Rather than original structures of mathematical knowledge, the MONET OWL ontologies (cf. section 1.2.2 for MONET) describe mathematical problems and the software used to solve them. It is, however, instructive to study how the MONET problem ontology represents mathematical objects. It focuses on the operator or constructor symbol at the root of the functional tree representation of a mathematical objects as a tree. Suppose the MONET broker knows a web service for computing definite integrals constructed with the *oms:calculus1#defint* symbol [68]. The type of problem that that service solves can be modeled as follows:

$$\textit{p:definite\_integration} \sqsubseteq p{:}\textit{Problem} \sqcap g{:}\textit{GamsH2a}$$
$$\sqcap =1 p{:}\textit{openmath\_head.oms}{:}\textit{calculus1\#defint}$$

The deeper structure is only represented in OpenMath; it is not used for service matching, but sent to a matching service for computation.

HELM: The HELM system (cf. section 1.2.1) generates from an original formalized representation in a non-XML language both a full XML representation and a standoff RDF graph containing a structural outline of properties relevant for searching. HELM's OWL ontologies distinguish terms (corresponding to mathematical objects in our terminology), objects (roughly corresponding to statements), and theories. There is a notion of dependency, such as a corollary being a consequence of a theorem (*hth: isConsequenceOf*), or a lemma being a prerequisite of a theorem (*hth:isPremiseOf*). Terms can have occurrences of other HELM objects, i.e. symbols. Such an occurrence is reified as a resource, which has an *h: position* and an integer *h:depth* counting the number of premises, including universal quantifiers. Among the positions that have been found relevant for answering queries, e.g. for finding applicable theorems for proving something (cf. [168,109]), there are the following, explained using the example of the theorem $\forall a : \mathbb{N}.\forall b : \mathbb{N}.\forall c : \mathbb{N}.a \leq b \wedge b \leq c \Rightarrow a \leq c$:

h:MainHypothesis: the head symbol of a hypothesis; here: $\wedge$ (depth 3)
h:InHypothesis: any other symbol anywhere else in a hypothesis; here, either of the two $\leq$
h:MainConclusion: the head symbol of the conclusion; here: $\leq$ (depth 4)

From the point of view of representing general mathematical knowledge, the HELM ontologies do not sufficiently abstract from the native knowledge representation of the Coq library (cf. section 3.3). Examples are the implicit relation between theories and their statements and the idiosyncratic mechanism for identifying theory items.

MoWGLI: An RDFS ontology with a wide coverage of logical/functional structures, including informal representations, educational content, and a rich set of metadata, was developed in the MoWGLI project [99]. MoWGLI reused vocabulary from the HELM ontologies, existing general and educational metadata ontologies, the XML schema of the OMDoc markup language (cf. section 3.2.3), and the metadata vocabularies of the ActiveMath e-learning system. For the latter two, an RDFS model was newly developed. The MoWGLI ontology (merely called "metadata model" due to its standoff usage) does not make further assumptions about the format in which the full knowledge is represented.

Summarizing, MoWGLI serves as an instructive example of a comprehensive integrated mathematical ontology. However, various shortcomings[15] make it technically unusable. It is not clear whether it has ever been used; except for its specification, no trace in the form of annotated documents is left.

OMDoc 1.3: The OMDoc OWL ontology [131, 132] has been modeled after the conceptual model and XML schema of the OMDoc language (cf. section 3.2.3). While not yet as comprehensive as the OMDoc language[16], the ontology has a richer statement- and theory-level vocabulary and more notions of dependency than the other ontologies reviewed.

Figure 3 shows the core classes and properties. Some of the depicted classes have subclasses. Definitions can, e.g., be pattern-based, implicit, or recursive, and types can be declared or asserted. Assertions comprise theorems, lemmas, and corollaries, and they can have different truth values. Moreover, the ontology covers sub-statement structures such as proof steps. Different degrees of formality are distinguished by a property. The definition in listing 1 is formal but not fully computerized to a degree an automated theorem prover would understand; actually, it consists of a formal and an informal part.[17]

There are three orthogonal properties that relate mathematical knowledge items to each other, each with a hierarchy of subproperties. Whole-part properties link, e.g., theories to their statements and proofs to their steps. The two parts of the definition in listing 1 are related by a verbalizes/formalizes relation; simi-

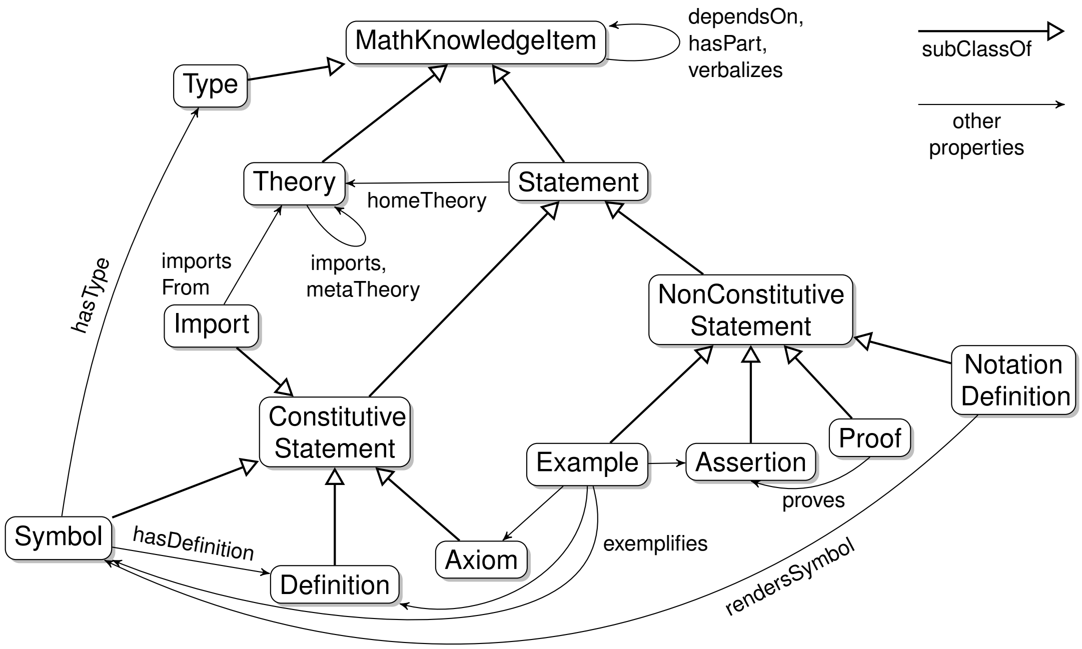

Fig. 3. The core of the OMDoc ontology (slightly simplified) [132]

lar relations can occur on all structural levels. Thirdly, there is dependency w.r.t. logical well-formedness, validity, and presentation. If, for example, one symbol is defined in terms of other symbols, such as the exponential function in terms of differentiation, its well-formedness depends on them. This is reflected by the following property hierarchy:

$$\begin{aligned}&o{:}hasDefinition \circ o{:}usesSymbol\\ &\sqsubseteq o{:}hasOccurrenceOfInDefinition\\ &\sqsubseteq o{:}wellFormedNessDependsOn\\ &\sqsubseteq o{:}dependsOn\end{aligned}$$

The *o:usesSymbol* property flattens the functional structure of a mathematical object by treating all occurrences of symbols equally, regardless of the depth of the expression tree in which they occur. For dependency w.r.t. validity, there is so far merely a proof of concept, namely the dependency of a proof on an inference rule or any other axiom or proven assertion used to justify a proof step:

$$\begin{aligned}&o{:}hasStep \circ o{:}stepJustifiedBy\\ &\sqsubseteq o{:}validityDependsOn \sqsubseteq o{:}dependsOn\end{aligned}$$

The OMDoc ontology also covers notation definitions for symbols; for example, the $\exp$ symbol from our example could be defined to render as e^x. When an OMDoc document is published, the presentation of any formula using that symbol *possibly* depends on that notation definition:

$$\begin{aligned}&o{:}usesSymbol \circ o{:}hasNotationDefinition\\ &\sqsubseteq o{:}possiblyUsesNotationDefinition\\ &\sqsubseteq o{:}presentationDependsOn \sqsubseteq o{:}dependsOn\end{aligned}$$

By means of the ontology, this cannot be decided definitely, as the knowledge base might have alternative notations for different presentation contexts. Static context matching exceeds the expressivity of a DL ontology, and in a dynamic setting, where the presentation context depends on the profile of the user viewing the published document, its notational dependencies can only be determined at runtime.

MathLang DRa: The "Document Rhetorical aspect" (DRa)[18] of the MathLang representation language covers larger chunks of mathematical text – document sections as well as mathematical statements – and their interrelations, such as a proof justifying a theorem [165,121]. A generic dependency relation has been defined, which is used for validating whether the narrative order of a document respects the logical dependencies. Conceptually, this is similar to the statement level of the OMDoc ontology. The OWL implementation of the DRa vocabulary merely serves as a formal specification of the DRa semantics, whereas the validator processes an XML representation of the DRa [165]. A drawback of the DRa ontology is that it cannot easily be extended by, e.g., additional statement types and additional dependency relations.

PML: PML (Proof Markup Language), an "interlingua for sharing explanations generated by various automated systems such as hybrid web-based question answering systems, text analytics, theorem proving, task processing, web services execution, rule engines, and machine learning components" [143], has been implemented as an OWL ontology consisting of modules for provenance, information manipulation or justification, and trust. PML assumes that facts and proofs have been written in some other language and merely adds standoff markup. Resources annotated that way can be referenced by URI or, in the case of text-based languages such as KIF, by byte offset. The justification module supports unproven conclusions or goals, assumptions, direct assertions, and antecedent→consequent justifications backed by inference rules. The provenance module has a vocabulary for describing inference rules – again, not down to the object level. So far, this is similar to the OMDoc ontology. Finally, the trust module allows for expressing degrees of belief in informations and trust in agents.

3.4.3. Scientific Documents

While logical/functional structures of mathematical knowledge may occur on their own, e.g. in formalized knowledge bases, rhetorical structures are usually studied in the context of documents written in, e.g., LaTeX or an XML language. Two very similar families of ontologies suitable for modeling rhetorical structures in mathematical documents are SALT [105,106] and OntoReST [150]; further related models and ontologies have been reviewed in [107]. SALT and OntoReST are relevant for the following reasons:

- Both have a good coverage of RST-style rhetorical structures.
- Either use case is related to mathematical collaboration: SALT focuses on annotating and linking scientific publications on the Web. OntoReST focuses on consistency checking in concurrent collaborative writing.
- Both allow for an arbitrarily fine-grained annotation of phrases. SALT additionally focuses on cross-document links for justifying statements by citing the claims made [and justified] in external publications [106].
- Both are, in principle, open for integration with arbitrary domain knowledge – which would be mathematical knowledge in our case.

Both approaches comprise three ontologies; here, we explain the model of SALT:

The Document Ontology models the outline of the document – sections, paragraphs, sentences, and text chunks (in OntoReST: "spans") below sentence level [103]. The latter remain in the original representation of the document; SALT provides standoff markup via start and end pointers to their positions in the full text. Additionally, one can represent the linear order of document units by numbering them.

The Annotation Ontology connects instances of the document ontology with annotations of their rhetorical structure and with background domain knowledge, such as the topic of a section [102]. While rhetorical structures are the primary focus of SALT, the mechanism is sufficiently general to also permit annotation of other structural dimensions.

The Rhetorical Ontology covers RST-style rhetorical relations [104]. Their nuclei and satellites, subsumed as "rhetorical elements", are linked to text spans in the document via the annotation ontology. Coarse-grained rhetorical blocks that can be applied on top level of a document are offered as an alternative. OntoReST provides a stronger OWL formalization of RST that supports consistency checking [150].

3.4.4. Scientific Discourse Ontologies

Ontologies formalizing discourse inside scientific publications are closely related to the above-mentioned ontologies for rhetorical structures; in fact, SALT supports both. GROZA et al. provide an overview of further related document formats and ontologies [107].

The DILIGENT argumentation model introduced in section 2.4 has been implemented in several variants [175,176,85]. The SIOC (Semantically Interlinked Online Communities [62,59,61]) ontology, which models user-generated content on the Web and is widely supported by Web 2.0 applications, has a DILIGENT-inspired argumentation module that al-

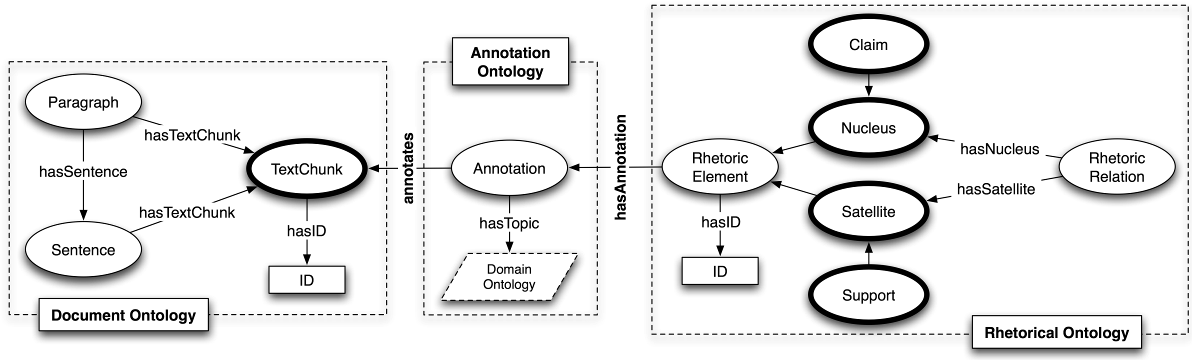

Fig. 4. The three-layered architecture of the SALT ontologies (simplified) [106]

lows for a slightly more flexible thread structure than the original DILIGENT implementations, which makes it applicable in a wider range of Web 2.0 settings [134]. An ontology of mathematical problems and solutions has been provided as an extension of the SIOC argumentation module [135].

Combining both models of scientific discourse in one ontology has been pioneered by the alignment of SWAN (Semantic Web Applications for Neuromedicine [107]) with SIOC [158]. SWAN models scientific discourse, not exactly in publications, but in a distributed knowledge base by pointers to bibliographic records and entities from domain ontologies. Its primary target is neuromedicine, but, as SALT, it supports arbitrary domain ontologies in principle.

3.4.5. Mathematical Metadata Vocabularies

Most of the metadata vocabularies mentioned in section 2.3 have been implemented as ontologies. This is the case with Dublin Core [151], LOM [118,152]. The mathematics-specific metadata vocabulary of OpenMath CDs (cf. section 3.2.2) has been implemented as an extension to Dublin Core [129]. Their review status is documented by metadata fields such as status (official, experimental, private, or obsolete), version, and the date of the next review.

When a classification scheme is not available as an ontology, one can use the identifiers of its categories as literal values of metadata fields such as *dc:subject*. A proper ontology implementation, where each category is a resource of its own, has further advantages: The hierarchy of categories can be made explicit, and URIs can be used more flexibly in queries. In MONET project, a simple class hierarchy of the GAMS problems has been implemented [145]. DOLOG et al. have turned the ACM CCS, a computer science classification scheme, into an ontology [87], drawing on the "classification" vocabulary of LOM. Similarly, the MSC 2010 is currently being translated to a taxonomy based on the SKOS ontology[11] [29].

3.4.6. Pure and Applied Mathematical Domain Ontologies

Any instance document of the XML languages and formalized languages and any ABox of the structural ontologies reviewed in this section can be considered a mathematical *domain ontology*. In particular, the following knowledge collections have been designed for reuse, reviewed or machine-verified to ensure a high quality, and published with stable identifiers – however, usually neither satisfying the Linked Data criteria nor linked to other collections: The official OpenMath CDs have been mentioned in section 3.2.2. Reusable OMDoc implementations of a large number of logics and translations between them have been published in a "Logic Atlas", including various variants of first-order, higher-order, modal, and description logic [125]. The collection of mathematical, scientific and technological courseware in the Connexions repository has been mentioned in section 1.1.1, libraries of formalized mathematics in section 3.3; however, the latter do not use URIs, as discussed in section 1.1.2. The N3 "math" vocabulary (cf. section 3.4.2) defines a fixed set of basic mathematical functions, roughly corresponding to the *arith1*, *relation1*, and *transc1* OpenMath CDs.

Finally, we mention domain ontologies from fields related to mathematics: GAMS (cf. section 3.4.2) features a directory of software that solves mathematical problems [4], but only a subset has been made

[11] personal communication with PATRICK ION, 2010-07-30

available within MONET. The SWEET domain ontology for science and GeoSkills for interactive geometry have already been reviewed in section 2.3.

3.5. Conclusion

Table 1 shows at a first glance that no single language satisfies all requirements for representing mathematical knowledge on the Semantic Web, but that expressive XML languages and RDF complement each other. The XML languages, headed by OMDoc (including MathML or OpenMath objects), lead the way w.r.t. coverage of mathematical structures and combining formal and informal representations. Moreover, they are reasonably well accepted by the MKM community, as opposed to RDF, and most of today's mathematical domain knowledge is available in these XML languages, or in formalized languages that have XML translations. RDF, above all, has superior linking capabilities. Table 2 show that RDF is capable of covering all structural aspects of mathematical knowledge, given a combination of suitable ontologies.

In MKM practice, RDF standoff markup pointing to full XML representations has so far proven most useful. Particularly in the case of mathematical objects, only selected information relevant for, e.g., information retrieval, is represented in RDF, as opposed to representing full n-ary ordered trees using RDF collections. RDFa embedded into XML has so far only been used in a few OMDoc documents but also seems a promising way to satisfy the given representation requirements.

Besides these considerations, the availability of tools also advises a division of responsibilities between XML and RDF.[19] Editors and publishing tools for semantic representations of mathematical knowledge are almost exclusively available for XML or formalized languages. For most XML languages, translations to the native languages of computer algebra systems or proof assistants have been implemented. Conversely, RDF is preferable for information retrieval, except on the object level. Both representations are good to have for a thorough validation; browsers also exist for both.

4. Representing Mathematical Knowledge on the Semantic Web

The previous section has concluded with the finding that both XML and RDF representations of mathematical knowledge are needed on the Semantic Web. This section explains techniques for integrating both. While the concrete examples are taken from representations for mathematical knowledge, the results apply to any domain where semantic XML markup is in use, such as the humanities with TEI.

4.1. Bridging Mathematical XML Markup and Ontologies by Translation

Why not Combined XML and RDF Queries? One possibility to utilize knowledge that has been represented partly in XML and partly in RDF is to employ a query language, such as XSPARQL, that supports both representations (cf. section 3.4.1). The other possibility is translating from one representation to the other one. Pure XML or RDF representations enjoy wide tool support and allow developers to focus on one query language and one object model, whereas, with a combined query language, one would always have to consider two variants in the worst case that some contributors to a knowledge base choose to represent a fact in XML, whereas others would represent the same in RDF.

Rationale for Translating RDF to XML: In this section, we consider XML→RDF translations. The reverse translation has the drawback that XML-based querying approaches handle abstraction and links less well. Most triple stores offer at least RDFS entailment in queries. Assuming, for example, the two OMDoc+RDFa fragments . . .

```
<theory about="#t">
 <imports about="#i" from="#u"/>
```

and

```
<proof about="#p">
 <derive about="#step">
  <FMP>¬⊤ = ⊥</FMP>
  <method><!-- proof by known axiom -->
   <premise xref="#axiom1"/>
```

. . . it would require a considerable effort of declaring, e.g., XML Schema datatypes and implementing XQuery functions to determine that both *#t* depends on *#u* and *#p* on *#axiom1*, whereas an OMDoc→RDF translation would generate the triples . . .

```
<#t> o:hasImport <#i> .
<#i> o:importsFrom <#u> .
<#p> o:hasStep <#step> .
<#step> o:stepExternallyJustifiedBy <#axiom1> .
```

Table 1

How the languages reviewed satisfy the knowledge representation requirements

Requirement	Structures							Formality		Linking			Compr.	
	S.L.*			S.R	S.N	S.M	S.D	F.R	F.C	L.A	L.→	L.←	C.A	C.H
	O	S	T											
MathML	++	–	–	–	–	–	–	++	+	+	+	+	–	+
OpenMath Objects	++	–	–	–	–	–	–	+	○	○	–	+	–	○
OpenMath CDs	++[a]	○	○	–	–	○	–	○	○	–	–	–	–	○
OMDoc	++[a]	++	++	++	+	++[c]	–	++	+	○	–	+	–	–
MathLang	++	++	–	–	–	–	–	++	+	–	–	○	○	–
DocBook	++[a]	–	–	–	–	+[d]	–	–	–	–	+	+	–	–
TEI	++[b]	–	–	–	–	○	–	++	+	+	–	+	–	–
DITA	++[b]	–	–	–	–	+[d]	–	–	–	+	+	+	–	–
EPUB/DTBook	++[b]	–	–	–	–	+	–	–	–	–	–	+	–	–
CNXML/CollXML/mdml	++[a]	+	–	–	–	○	–	–	–	–	–	+	–	–
Formalized languages	++	++	○	–	+	–	–	○	○	–	–	–	+	–
RDF(a) 1.0/1.1	(depends on vocabulary, see table 2)							○	+	++	++	++	○	+

[a] built-in support for MathML/OpenMath objects

[b] via MathML extension

[c] Dublin Core and similar vocabularies built in, others available via built-in RDFa

[d] Dublin Core and similar vocabularies built in, others available via non-RDFa extensions

Table 2

Structural coverage of vocabularies/ontologies

Structures	Logical/functional			Rhetorical	Notation	Metadata	Discussion
	Objects	Statements	Theories				
N3 Vocabularies	+	○	–	–	–	–	–
OpenMath CD	○	○	○	–	–	○	–
HELM	+	+	○	–	–	–	–
MoWGLI	+	++	○	–	–	+	–
OMDoc	○	++	+	–[b]	+	–	–
MathLang DRa	–	+	–	–	–	–	–
PML	–	++[a]	–	–	–	–	–
SALT	–	–	–	++	–	–	+
OntoReST	–	–	–	++	–	–	–
DILIGENT	–	–	–	–	–	–	+
SIOC Argumentation	–	–	–	–	–	–	++
Dublin Core	–	–	–	–	–	++	–

[a] proofs only

[b] intentionally delegated to SALT

... from which a query engine with DL entailment support would infer ...

```
<#t> o:dependsOn <#u> .
<#p> o:dependsOn <#axiom1> .
```

Requirements for Translating Mathematical XML Markup to RDF: In our previous work on translating OMDoc documents and OpenMath CDs to RDF, we have identified the following general requirements for translating semantic XML markup to RDF, independently from the XML language and the ontology [132]:

All structural entities that correspond to concepts covered by the given ontologies MUST be given an *identifier* by applying the first of the following rules that matches:

1. If the XML language hosts RDFa, the identifier – URI or blank node ID – MUST be determined according to the RDFa processing rules for identifying a *new subject* [33, section 7.5].
2. If the XML language specifies how to generate a URI for an entity represented by an XML fragment, that URI MUST be used.
3. If the XML language specifies how to generate an ID for an entity, e.g. via XML ID [141], that ID MUST be used as a fragment ID if possible w.r.t. the syntax of URIs [57]; appending it to the document's URI yields the URI.
4. If the XML language specifies how to generate an ID for an entity, which does not qualify as a fragment ID, the translator MUST generate a fragment ID, which SHOULD reflect the original ID.
5. Otherwise, the translator MUST mint an URI; this MAY be a URI or a blank node IDs. Minted URIs MUST NOT conflict with URIs generated for other entities in the XML document.

In practice, most semantic XML markup languages support IDs on all elements, but authors only use them when an element is a target of an explicit link in the markup. Many RDF properties, such as whole-part relations, are, however, not represented by explicit XML links but by a parent-child relation, but triples using these properties require identifiable subjects and objects.

For authors and developers, reusing the identifiers from the XML markup in the RDF representation emphasizes the correspondence of both representations. For agents, it improves retrievability, e.g., of RDF standoff markup for an XML representation: If a structural entity always has the same identifier, regardless of the representation format – semantic markup, RDF, or even a human-comprehensible presentation –, and if its different representations are published according to the “cool URI” best practices [167], all of them can be made available under the same URI. A client – agent or browser – would select the desired representation via HTTP content negotiation.

The complexity of semantic XML markup languages for mathematical knowledge entails a number of challenges to the declaration and implementation of an XML→RDF mapping, for example[20]:

URI Format Differences: OpenMath specifies a canonical URI syntax for symbols. The “namespace base URI”, called *CDBase*, may – and, in practice, usually is – omitted and defaults to `http://www.openmath.org/`. However, when OpenMath objects occur inside OMDoc theories, the default base URI of a symbol is determined from the theory from which the symbol has been imported.

Mapping Elements to Classes: Generally, OMDoc XML elements correspond to classes from the OMDoc ontology. However, the ontology has been designed with its utility for RDF-based applications in mind, not necessarily to represent the OMDoc XML markup literally. Therefore, instances of some subclasses are represented by the same element, only differing in the value of a certain attribute, or even by elements with different names.

Markup Choices for Representing Relations: Relations between two entities can be represented as a parent-child relation in XML markup, as a sibling relation, or by URI- or ID-valued attributes. As stated for classes above, the exact type of a relation is sometimes influenced by additional attributes on the same element.

Markup Choices for Representing Literal-valued Properties: Literal-valued properties can be represented by text-valued immediate child elements, by descendant elements nested more deeply, or by attributes.

Implicit Structures: The target ontologies reify certain concepts that do not have an explicit representation in the semantic markup. This is, e.g., the case with informal/formal property pairs in OpenMath CDs, and with document units, annotations, and rhetorical relations in the mapping of OMDoc's rhetorical markup to SALT.

Alternative Representations of Classification Schemes: Where a classification scheme has been implemented as an ontology, its categories are represented as classes or individuals. Otherwise, they are represented as literals. Similarly, metadata with a finite value space can be represented as RDF literals or as instances of an enumerated class.

In our previous work, we have found existing declarative XML→RDF mappings too restricted and instead chose to implement a library of XSLT convenience functions and templates, which facilitates the implementation of frequently occurring translation patterns but gives access to the full power of XSLT if necessary [136,128].

4.2. Contributing Mathematics to the Web of Data

Benefits … of publishing knowledge as Linked Data include easier development of interactive mashups (see, e.g., [114,174]) and the possibility to detect previously unknown links (see, e.g., [115]). Given that mathematical knowledge is likely to be available partly in XML and partly in RDF, as explained in section 3.5, data providers should publish both representations – which is possible, as outlined in section 4.1.

… for agents … In [180,130], we describe a scenario where an agent accesses both RDF datasets and OpenMath CDs by dereferencing URIs: The rules for computing derived values in statistical datasets are represented as RDF annotations pointing to a function – a symbol from an OpenMath CD – and other values from the dataset that should be passed as arguments to the function, using the SCOVOLink vocabulary [180]. When an agent wants to verify the derived value, it has to construct an OpenMath object from this RDF representation and send it to an OpenMath-aware computation service (cf. sections 1.2.2 and 1.2.3). When the function is not called using positional arguments or an argument list, but using named arguments, the agent has to consult the XML representation of the CD to get their order right.

… and humans: The semantic representations for mathematical knowledge reviewed in this article allow for preserving the full semantics in documents published for human readers, so that, e.g., assistive services can utilize it. For anything except mathematical objects, i.e. formulæ, XHTML+RDFa is a suitable publication format. Assistive services for formulæ have previously been realized for Presentation MathML with Content MathML or OpenMath annotations [95,157]. With HTML 5 becoming mainstream, more browsers can soon be expected to support MathML.[21] We have implemented a library that publishes OMDoc as XHTML+MathML+RDFa [81].

Challenges: In our previous work on publishing and consuming OMDoc documents and OpenMath CDs as Linked Data [81,130], we have identified three challenges to publishing mathematical knowledge as Linked Data: specifying MIME types for XML languages, bad practices of authors, and restrictions in the URI formats of XML languages.

The HTTP Content Negotiation mechanism outlined in section 4.1 distinguishes representation formats by MIME type. MathML 3, for example, has officially registered MIME types [44], OMDoc specifies an unofficial one [124], whereas MIME types for OpenMath objects and CDs have merely been proposed so far [130].

Authoring practices that are bad from a Linked Data point of view result from the fact that, where semantic XML languages for representing mathematical knowledge support URIs, authors use them wrongly or not at all. For example, hardly any OpenMath CD that has been contributed to `openmath.org` specifies a CDBase URI or references symbols by full URIs, which indicates a lack of awareness. The fallback value `http://www.openmath.org/` is not suitable for non-official CDs mainly used by one research group, even independently from Linked Data considerations, as they do not control the `openmath.org` domain. Finally, if authors are aware of the fact that CDs and symbols have URIs, they usually merely consider it a globally unique *name*, but not a means of retrieving information about these resources [130].

Thirdly, while RDF publishers can freely choose URIs (see, e.g., [58]), the URI formats of non-RDF languages often impose restrictions that complicate Linked Data publishing and have to be worked around. For example, OpenMath's schema of *cdbase/cd#name* symbol URIs, which has also been adopted by Strict Content MathML, complies well with linked data practices – unless CDs grow large. As resolving fragments after the # (hash) in a URI is up to the client, the consequent use of hash URIs for OpenMath symbols forces clients to always download a complete CD from the server, in which could then locate the symbol with the desired name. Publishers of large CDs would have to set up a redirect, where an initial request for a hash URI would result in an RDF graph that merely redirects, via *rdfs:seeAlso* links, hash URIs to slash URIs, from which the client could be able to retrieve the desired fine-grained information. The URIs of the upcoming OMDoc 1.6 have a slash-like format [163], but, again, without alternatives. Another problem of OMDoc 1.3 and its hash URI format is that symbols and theories have to be declared as fragments of the same document, which is not compatible with OpenMath's *cdbase/theory#symbolname* schema, even though OMDoc uses OpenMath objects. Combined with the possibility of having multiple theories in a document, redirect workarounds may not be possible.[22] A final problem with old languages such as the OpenMath CD language, is that certain entities cannot be given IDs. An XML→RDF translator might generate some, but an agent interested in retrieving XML representations would also need them. As a use case that would require such fine-grained links, consider the DLMF [3]. It contains a large number of equations describing or defining mathematical functions, which could be linked to the corresponding mathematical properties in OpenMath CDs.

5. Research Directions Towards a Mathematical Web of Data

Large collections of mathematical knowledge exist in non-RDF representations. The ontologies reviewed in section 3.4 and the translation techniques outlined in section 4 now enable us to contribute them to the Web of Data and fill a gap that existing Linked Datasets about, e.g., statistical government data or scientific publications, have left.

The probably most foundational dataset that has to be published as Linked Data in order to get mathematics on the Web of Data started is the official OpenMath CDs. An initial publication is feasible with the technology available and planned for spring 2011. This is, however, only the first step in making the knowledge contained in the CDs accessible; the second step is creating links from mathematics-related existing datasets into the OpenMath CDs, so that services for these existing datasets can be extended by mathematical functionality. We have sketched one use case for statistical datasets in section 4.2. The inevitable DBpedia [82] is a further candidate, as that would offer its users a more formal perspective on mathematics.

Mathematical knowledge collections that are already available on the Web, but not currently in a semantic representation, should also be semantically annotated – not necessarily as deeply as, e.g., OMDoc documents, but at least with mathematical metadata and links to relevant OpenMath CDs. For example, the DLMF [3] could benefit from access to computation services via OpenMath, whereas the benefit for PlanetMath, which is currently being overhauled to make it more interactive and more semantic [80], would be similar as for DBpedia.

The availability of true mathematical knowledge as Linked Data would also allow for taking a serious view on the April fool's joke "Linked Open Numbers", a huge dataset describing billions of natural numbers [179]. It provided descriptions as trivial as the name of each number in natural language, its predecessor and its successor. But how about a dataset of non-trivial properties of numbers? Accessing, for example, prime factor decompositions of large numbers – an information relevant for cryptography – in a linked dataset, could be much faster than computing it once more, provided a supercomputer has already done the computation once and published the results. Another source of non-trivial knowledge about numbers, which deserves being published as Linked Data, is the Online Encyclopedia of Integer Sequences [170].

An issue related to the combination of information retrieval and computation is the development of suitable query languages and reasoners. N3 reasoners already support a basic set of mathematical functions (cf. section 3.4.2). The upcoming SPARQL 1.1 supports basic arithmetics. Additionally, many query processors allow for defining extension functions; a path for supplying arbitrary functions to query processors via OpenMath should be investigated. Taking a formal semantics and computational complexity into account, such an extension could even be specified as an entailment regime [97], which makes a query return a well-defined set of additional, *entailed* results beyond the information that is explicitly encoded in the RDF graph being queried.

A possible gateway into annotating the mathematical semantics of scientific publications is the arχiv [39]. Its documents are mostly available as presentation-oriented LaTeX; however, a long-term effort to automatically annotate their mathematical structure is in progress [96], the translation of 300,000 of the 500,000 publications to XHTML+MathML, which has at least more semantic structure than LaTeX, being a first success [173]. Publishing a basic metadata record for each arχiv publication as Linked Data is feasible, as the metadata are available as XML, and the publications have stable URIs. Next, much harder steps would be interlinking with publication databases already existing as Linked Data, such as DBLP [2], and identifying mathematical symbols that could be linked to the OpenMath CDs. With an identification of statement- and theory-level logical/functional structures, this would ultimately enable powerful machine support for looking up information relevant for the next collaborative Web-based review of a $P \neq NP$ proof, or the next collaborative effort to formalize a proof of a theorem like the Kepler Conjecture.

Notes

[1] ProgrammableWeb [16], a directory of mashups, lists 3 mashups tagged with "math", out of nearly 5,000 mashups overall. The recently released "widgets" for the Wolfram Alpha "computational knowledge engine" [24] are a first step towards more mashups, albeit limited to acting as frontends to Wolfram Alpha.

[2] This notion of the term "knowledge management" is wider than that of its traditional definition as "a range of practices used in an organisation to identify, create, represent, distribute and enable adoption of insights and experiences. Such insights and experiences comprise knowledge, either embodied in individuals or embedded in organisational processes or practice." [186]

[3] independence of a concrete RDF syntax (such as RDF/XML), disjunction, data source identification, and a well-defined formal semantics [108]

[4] The HELM developers made no secret out of their frustration: "It is a pity that [. . .] most of the expectations about XML technologies [including RDF] have not been fulfilled due to intrinsic deficiencies in their design and implementation. MathML failed to be adopted by major browsers; [. . .] and RDF never really went beyond the project phase." [43] Personal communication with ASPERTI on 2010-07-09 confirmed that that statement referred to the immaturity of these technologies at the time of developing HELM.

[5] Similarly, BAEZ suggests that the release of a TEX formula editor plugin for the popular WordPress blog engine was a major incentive for mathematicians to start blogging [47].

[6] We provide concrete examples and a proof-of-concept implementation in [180,130].

[7] Definitions typically occur in textbook-style mathematics. From a formal point of view they are merely a variant of axioms.

[8] This can also be considered a difference w.r.t. the area of application. For example, in theoretical computer science it is advantageous to include 0, as many of the required induction proofs start at 0, whereas negative integers are rarely needed.

[9] This conjecture, posed in 1611, states that the density of a packing of unit spheres in 3 dimensions is at most $\pi/(3\sqrt{2})$. This reflects the intuitive observation that the way, in which, e.g., oranges in a market booth are stacked, is optimal. However, it turned out exceedingly complex to prove.

[10] In an OWL setting, one has to avoid RDF collections, as the RDF encoding of OWL uses them internally for representing n-ary DL expressions. Instead, one has to create one's own linked lists [88].

[11] At least support for querying RDF collections, which some query processors already support by non-standard extensions, will be standardized in the upcoming SPARQL 1.1 [113].

[12] See [132] for a discussion of concrete examples from OMDoc.

[13] His encoding differs from the N3 encoding in that order is represented using RDF's built-in container membership properties *rdf:_n* ($n = 1, 2, \dots$) instead of RDF collections, but that is a secondary issue.

[14] A possible explanation in MARCHIORI's case is that his proposal did not originate out of the MKM community but that he was an external (Semantic Web) expert invited to give a keynote.

[15] ambiguities and errors in its own modeling, tampering with the semantics of reused vocabularies (such as DCMES), limited documentation, and use of bad RDFS modeling practices (cf. [132] for detailed examples)

[16] The ontology does not cover complex theory morphisms, abstract datatypes, and presentation contexts.

[17] The names *CMP* = Commented Mathematical Property and FMP = Formal Mathematical Property are for historical reasons and OpenMath compatibility.

[18] Despite the name, this is not related to rhetorical structures in the sense of RST.

[19] [132, chapter 6] provides a comprehensive overview.

[20] see [132] for details

[21] At the moment, only Mozilla/Firefox supports MathML well enough to allow for interactive manipulation via scripts.

[22] See [132] for a detailed discussion.

References

[1] Specifications for the Digital Talking Book. Technical report, DAISY Consortium. URL `http://www.daisy.org/z3986/2005/Z3986-2005.html`. ANSI/NISO Z39.86-2005.

[2] D2r server publishing the dblp bibliography database. URL `http://dblp.l3s.de/d2r/`.

[3] Digital Library of Mathematical Functions. URL `http://dlmf.nist.gov`.

[4] GAMS: Guide to Available Mathematical Software. URL `http://gams.nist.gov`.

[5] HELM. URL `http://helm.cs.unibo.it`.

[6] Math-Net, . URL `http://www.math-net.org`.

[7] MathOverflow, . URL `http://mathoverflow.net`.

[8] Wolfram mathworld, . URL `http://mathworld.wolfram.com`.

[9] Mizar mathematical library, . URL `http://www.mizar.org/library`.

[10] Mizar system, . URL `http://mizar.org/system/`.

[11] OEBPS Container Format (OCF). Recommended specification, International Digital Publishing Forum. URL `http://www.idpf.org/ocf/ocf1.0/download/ocf10.htm`.

[12] OMDoc. project page at `http://omdoc.org`.

[13] Open Packaging Format (OPF). Recommended specification, International Digital Publishing Forum. URL `http://www.idpf.org/2007/opf/OPF_2.0_final_spec.html`.

[14] The polymath blog, . URL `http://polymathprojects.org/`.

Table 3
Namespace prefix↦URI bindings used in this article

Prefix	URI	Language/Vocabulary
dc	`http://purl.org/dc/elements/1.1/`	Dublin Core Metadata Element Set
g	`http://gams.nist.gov#`	GAMS
h	`http://www.cs.unibo.it/~schena/schema-h.rdf#`	HELM objects
hth	`http://www.cs.unibo.it/~schena/schema-hth.rdf#`	HELM theories
math	`http://www.w3.org/2000/10/swap/math#`	N3 math functions
o	`http://omdoc.org/ontology#`	OMDoc ontology
oms	`http://www.openmath.org/cd/`	OpenMath symbols
p	`http://monet.nag.co.uk/problems/`	MONET problems

[15] Deolalikar P vs NP paper, . URL http://michaelnielsen.org/polymath1/index.php?title=Deolalikar_P_vs_NP_paper&oldid=3654.

[16] ProgrammableWeb, . URL http://www.programmableweb.com.

[17] ProofWiki, . URL http://www.proofwiki.org.

[18] Rhaptos Trac: Collection structure redesign / inception. URL https://trac.rhaptos.org/trac/rhaptos/wiki/CollectionStructureRedesign/Inception?version=54.

[19] Semantic web for earth and environmental terminology (sweet). URL http://sweet.jpl.nasa.gov/.

[20] TEI – ontologies SIG. URL http://www.tei-c.org/Activities/SIG/Ontologies/.

[21] Tricki. URL http://www.tricki.org.

[22] Dublin Core Metadata Initiative, . URL http://www.dublincore.org.

[23] OPENMATH content dictionaries, . URL http://www.openmath.org/cd/.

[24] Wolfram|Alpha widgets. URL http://developer.wolframalpha.com/widgets/.

[25] Connexions – XML languages. URL http://cnx.org/help/authoring/xml.

[26] The coq proof assistant. URL http://coq.inria.fr/.

[27] The n-Category Café. URL http://golem.ph.utexas.edu/category/.

[28] nLab. URL http://ncatlab.org/.

[29] Skos simple knowledge organization system. URL http://www.w3.org/2004/02/skos/.

[30] Zentralblatt MATH. URL http://www.zentralblatt-math.org.

[31] Mathematics Subject Classification MSC2010, 2010. URL http://msc2010.org.

[32] ActiveMath. ACTIVEMATH. URL http://www.activemath.org.

[33] Ben Adida, Mark Birbeck, Shane McCarron, and Ivan Herman. RDFa core 1.1. W3C Working Draft, World Wide Web Consortium (W3C). URL http://www.w3.org/TR/2010/WD-rdfa-core-20101026/.

[34] Waseem Akhtar, Jacek Kopecký, Thomas Krennwallner, and Axel Polleres. XSPARQL: Traveling between the XML and RDF worlds – and avoiding the XSLT pilgrimage. In Bechhofer et al. [51].

[35] American Mathematical Society. MathSciNet Mathematical Reviews on the Net. URL http://www.ams.org/mathscinet/.

[36] Ola Andersson, Robin Berjon, Erik Dahlström, Andrew Emmons, Jon Ferraiolo, Anthony Grasso, Vincent Hardy, Scott Hayman, Dean Jackson, Chris Lilley, Cameron McCormack, Andreas Neumann, Craig Northway, Antoine Quint, Nandini Ramani, Doug Schepers, and Andrew Shellshear. Scalable Vector Graphics (SVG) Tiny 1.2 specification. W3C Recommendation, World Wide Web Consortium (W3C), December 2008. URL http://www.w3.org/TR/2008/REC-SVGTiny12-20081222/.

[37] Anupriya Ankolekar, Markus Krötzsch, Thanh Tran, and Denny Vrandečić. The two cultures: Mashing up Web 2.0 and the Semantic Web. *Web Semantics*, 6(1):70–75, 2008.

[38] Lora Aroyo, Grigoris Antoniou, Eero Hyvönen, Annette ten Teije, Heiner Stuckenschmidt, Liliana Cabral, and Tania Tudorache, editors. *The Semantic Web: Research and Applications (Part I)*, number 6088 in Lecture Notes in Computer Science, 2010. Springer Verlag.

[39] ArXiv. arXiv.org e-Print archive. URL http://www.arxiv.org.

[40] Andrea Asperti, Bruno Buchberger, and James Harold Davenport, editors. *Mathematical Knowledge Management, MKM'03*, number 2594 in LNCS, 2003. Springer Verlag.

[41] Andrea Asperti, Luca Padovani, Claudio Sacerdoti Coen, Ferruccio Guidi, and Irene Schena. Mathematical knowledge management in HELM. *Annals of Mathematics and Artificial Intelligence, Special Issue on Mathematical Knowledge Management, Kluwer Academic Publishers*, 38(1–3):27–46, May 2003.

[42] Andrea Asperti, Claudio Sacerdoti Coen, Enrico Tassi, and Stefano Zacchiroli. User interaction with the Matita proof assistant. *Journal of Automated Reasoning*, 39(2):109–139, 2007.

[43] Andrea Asperti, Herman Geuvers, and Raja Natarajan. Social processes, program verification and all that. *Mathematical Structures in Computer Science*, 19(5):877–896, October 2009.

[44] Ron Ausbrooks, Stephen Buswell, David Carlisle, Giorgi Chavchanidze, Stéphane Dalmas, Stan Devitt, Angel Diaz, Sam Dooley, Roger Hunter, Patrick Ion, Michael Kohlhase, Azzeddine Lazrek, Paul Libbrecht, Bruce Miller, Robert Miner, Murray Sargent, Bruce Smith, Neil Soiffer, Robert Sutor, and Stephen Watt. Mathematical Markup Language

(MathML) version 3.0. W3C Recommendation, World Wide Web Consortium (W3C). URL http://www.w3.org/TR/2010/REC-MathML3-20101021.

[45] Serge Autexier, Dieter Hutter, Till Mossakowski, and Axel Schairer. Maya: Maintaining structured documents. In OMDoc – *An open markup format for mathematical documents [Version 1.2]* Kohlhase [124], chapter 26.12. URL http://omdoc.org/pubs/omdoc1.2.pdf.

[46] Serge Autexier, Jacques Calmet, David Delahaye, Patrick D. F. Ion, Laurence Rideau, Renaud Rioboo, and Alan P. Sexton, editors. *Intelligent Computer Mathematics*, number 6167 in LNAI, 2010. Springer Verlag.

[47] John Baez. Math blogs. *Notices of the AMS*, page 333, March 2010. URL http://www.ams.org/notices/201003/rtx100300333p.pdf.

[48] Grzegorz Bancerek. Information retrieval and rendering with MML Query. In Borwein and Farmer [63], pages 266–279.

[49] Rebhi S. Baraka. *A Framework for Publishing and Discovering Mathematical Web Services*. PhD thesis, Johannes Kepler Universität Linz, 2006. URL http://www.risc.uni-linz.ac.at/publications/download/risc_2945/06-04.pdf.

[50] Michael J. Barany. '[b]ut this is blog maths and we're free to make up conventions as we go along': Polymath1 and the modalities of 'massively collaborative mathematics'. In Phoebe Ayers and Felipe Ortega, editors, *Proceedings of the 6th International Symposium on Wikis and Open Collaboration (WikiSym)*, ACM Press, 2010. URL http://www.wikisym.org/ws2010/Proceedings/.

[51] Sean Bechhofer, Manfred Hauswirth, Jörg Hoffmann, and Manolis Koubarakis, editors. *The Semantic Web: Research and Applications*, number 5021 in Lecture Notes in Computer Science, 2008. Springer Verlag.

[52] Dave Beckett. RDF/XML syntax specification (revised). W3C recommendation, World Wide Web Consortium (W3C), February 2004. URL http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/.

[53] Anders Berglund, Scott Boag, Don Chamberlin, Mary F. Fernández, Michael Kay, Jonathan Robie, and Jérôme Siméon. XML Path Language (XPath) 2.0. W3C recommendation, World Wide Web Consortium (W3C). URL http://www.w3.org/TR/2007/REC-xpath20-20070123/.

[54] Tim Berners-Lee. cwm, . URL http://www.w3.org/2000/10/swap/doc/cwm.html.

[55] Tim Berners-Lee. Notation 3 (N3) – a readable RDF syntax. Technical report, World Wide Web Consortium (W3C), . URL http://www.w3.org/DesignIssues/Notation3.html.

[56] Tim Berners-Lee. Design issues: Linked data, 2006. URL http://www.w3.org/DesignIssues/LinkedData.html.

[57] Tim Berners-Lee, Roy T. Fielding, and Larry Masinter. Uniform resource identifier (URI): Generic syntax. RFC 3986, Internet Engineering Task Force (IETF), 2005. URL http://www.ietf.org/rfc/rfc3986.txt.

[58] Diego Berrueta and Jon Phipps. Best practice recipes for publishing RDF vocabularies. W3C Working Group Note, World Wide Web Consortium (W3C), August 2008. URL http://www.w3.org/TR/2008/NOTE-swbp-vocab-pub-20080828/.

[59] Diego Berrueta, Dan Brickley, Stefan Decker, Sergio Fernández, Christoph Görn, Andreas Harth, Tom Heath, Kingsley Idehen, Kjetil Kjernsmo, Alistair Miles, Alexandre Passant, Axel Polleres, and Luis Polo. *SIOC Core Ontology Specification*, March 2010. URL http://rdfs.org/sioc/spec/.

[60] Chris Bizer, Sören Auer, and Gunnar AAstrand Grimnes, editors. *Scripting and Development for the Semantic Web (SFSW)*, number 449 in CEUR Workshop Proceedings, May 2009. URL http://CEUR-WS.org/Vol-449/.

[61] Uldis Bojārs and John G. Breslin. SIOC core ontology specification. W3C Member Submission, World Wide Web Consortium (W3C), June 2007. URL http://www.w3.org/Submission/2007/SUBM-sioc-spec-20070612/.

[62] Uldis Bojārs, John G. Breslin, Vassilios Peristeras, Giovanni Tummarello, and Stefan Decker. Interlinking the Social Web with Semantics. *IEEE Intelligent Systems*, 23(3), 2008.

[63] Jon Borwein and William M. Farmer, editors. *Mathematical Knowledge Management (MKM)*, number 4108 in LNAI, 2006. Springer Verlag.

[64] Scott Bradner. Key words for use in RFCs to indicate requirement levels. RFC 2119, Internet Engineering Task Force (IETF), 1997. URL http://www.ietf.org/rfc/rfc2119.txt.

[65] Lou Burnard and Syd Bauman. TEI P5: Guidelines for electronic text encoding and interchange. Technical report, TEI Consortium. URL http://www.tei-c.org/release/doc/tei-p5-doc/en/html/.

[66] Stephen Buswell. RDF/XML test cases for RDF logic, web ontology and maths content. Deliverable 5.3b, Semantic Web Advanced Development for Europe (SWAD-Europe), 2001. URL http://www.w3.org/2001/sw/Europe/reports/xml_test_cases/wp53.html.

[67] Stephen Buswell, Olga Caprotti, David P. Carlisle, Michael C. Dewar, Marc Gaëtano, and Michael Kohlhase. The Open Math standard, version 2.0. Technical report, The OpenMath Society, 2004. URL http://www.openmath.org/standard/om20.

[68] Olga Caprotti, Mike Dewar, and Daniele Turi. Mathematical service matching using description logic and OWL. In Andrea Asperti, Grzegorz Bancerek, and Andrej Trybulec, editors, *Mathematical Knowledge Management, MKM'04*, number 3119 in LNAI, pages 73–87. Springer Verlag, 2004.

[69] Olga Caprotti, Mike Dewar, and Daniele Turi. Mathematical service matching using description logic and OWL. Technical report, The MONET Consortium, 2004. URL http://monet.nag.co.uk/monet/publicdocs/monet_onts.pdf.

[70] Jacques Carette, Lucas Dixon, Claudio Sacerdoti Coen, and Stephen M. Watt, editors. *MKM/Calculemus Proceedings*, number 5625 in LNAI, July 2009. Springer Verlag.

[71] Tim Clark, Joanne S. Luciano, M. Scott Marshall, Eric Prud'hommeaux, and Susie Stephens, editors. *Semantic Web Applications in Scientific Discourse (SWASD)*, number 523 in CEUR Workshop Proceedings, 2009. URL http://CEUR-WS.org/Vol-523/.

[72] CNX. Connexions. URL http://cnx.org.

[73] Arjeh M. Cohen, Hans Cuypers, and Rikko Verrijzer. Mathematical context in interactive documents. *Mathematics in Computer Science*, 3(3):331–347, 2010.

[74] Olivier Corby, Leila Kefi-Khelif, Hacène Cherfi, Fabien Gandon, and Khaled Khelif. Querying the semantic web of data using sparql, rdf and xml. Technical Report 6847, INRIA Sophia Antipolis, February 2009. URL http://hal.inria.fr/docs/00/36/23/81/PDF/RR-6847.pdf.

[75] Hans Cuypers, Arjeh M. Cohen, Jan Willem Knopper, Rikko Verrijzer, and Mark Spanbroek. Mathdox, a system for interactive mathematics. In *Proceedings of the World Conference on Educational Multimedia, Hypermedia & Telecommunications 2008 (ED-MEDIA'08)*, pages 5177–5182. AACE, June 2008. URL http://go.editlib.org/p/29092.

[76] Wolfgang Dalitz, Wolfram Sperber, and Winfried Neun. Math-Net, a model for information and communication systems in sciences. In Jan von Knop, Peter Schirmbacher, and Viljan Mahnic, editors, *EUNIS*, number 13 in LNI, pages 140–145. GI, 2001. ISBN 3-88579-339-3.

[77] Ron Daniel Jr. Harvesting RDF statements from XLinks. W3C Note, World Wide Web Consortium (W3C), September 2000. URL http://www.w3.org/TR/2000/NOTE-xlink2rdf-20000929/.

[78] James H. Davenport, editor. *22nd OpenMath Workshop*, July 2009.

[79] James H. Davenport and Michael Kohlhase. Unifying Math Ontologies: A tale of two standards. In Carette et al. [70], pages 263–278. URL http://kwarc.info/kohlhase/papers/mkm09-MMLOM3.pdf.

[80] Catalin David, Deyan Ginev, Michael Kohlhase, and Joe Corneli. eMath 3.0: Building blocks for a social and semantic web for online mathematics & ELearning. In Ion Mierlus-Mazilu, editor, *1st International Workshop on Mathematics and ICT: Education, Research and Applications*, 2010. URL http://kwarc.info/kohlhase/papers/malog10.pdf. submitted.

[81] Catalin David, Michael Kohlhase, Christoph Lange, Florian Rabe, Nikita Zhiltsov, and Vyacheslav Zholudev. Publishing math lecture notes as linked data. In Lora Aroyo, Grigoris Antoniou, Eero Hyvönen, Annette ten Teije, Heiner Stuckenschmidt, Liliana Cabral, and Tania Tudorache, editors, *The Semantic Web: Research and Applications (Part II)*, number 6089 in Lecture Notes in Computer Science, pages 370–375. Springer Verlag, 2010.

[82] DBpedia. DBpedia. URL http://dbpedia.org.

[83] DCMI Usage Board. DCMI metadata terms. DCMI recommendation, Dublin Core Metadata Initiative, 2008. URL http://dublincore.org/documents/2008/01/14/dcmi-terms/.

[84] Jos De Roo. EulerSharp. URL http://eulersharp.sourceforge.net/.

[85] Klaas Dellschaft, Hendrik Engelbrecht, José Monte Barreto, Sascha Rutenbeck, and Steffen Staab. Cicero: Tracking design rationale in collaborative ontology engineering. In Bechhofer et al. [51], pages 782–786.

[86] DITA. OASIS Darwin Information Typing Architecture (DITA). URL http://www.oasis-open.org/committees/dita/.

[87] Peter Dolog, Rita Gavriloaie, Wolfgang Nejdl, and Jan Brase. Integrating adaptive hypermedia techniques and open RDF-based environments. In *Proceedings of the 12th WWW conference*. ACM Press, 2003.

[88] Nicholas Drummond, Alan Rector, Robert Stevens, Georgina Moulton, Matthew Horridge, Hai Wang, and Julian Sedenberg. Putting owl in order: Patterns for sequences in owl. In Bernardo Cuenca Grau, Pascal Hitzler, Connor Shankey, and Evan Wallace, editors, *OWL: Experiences and Directions (OWLED)*, November 2006.

[89] Dublin Core Metadata Element Set. Dublin Core metadata element set. DCMI recommendation, Dublin Core Metadata Initiative, 2008. URL http://dublincore.org/documents/2008/01/04/dces/.

[90] Bob DuCharme. Using RDFa with DITA and DocBook. URL http://www.devx.com/semantic/Article/42543/.

[91] Hans Magnus Enzensberger. *Drawbridge up*. A K PETERS, LTD., 1999. German original: Zugbrücke außer Betrieb. Die Mathematik im Jenseits der Kultur. English translation by Tom Artin.

[92] William Farmer, Josuah Guttman, and Xavier Thayer. Little theories. In D. Kapur, editor, *Proceedings of the 11th Conference on Automated Deduction*, number 607 in LNCS, pages 467–581, Saratoga Springs, NY, USA, 1992. Springer Verlag.

[93] William M. Farmer. MKM: A new interdisciplinary field of research. *Bulletin of the ACM Special Interest Group on Symbolic and Automated Mathematics (SIGSAM)*, 38(2):47–52, 2004.

[94] Armin Fiedler. *P.rex*: An interactive proof explainer. In Rajeev Goré, Alexander Leitsch, and Tobias Nipkow, editors, *Automated Reasoning — 1st International Joint Conference, IJCAR 2001*, number 2083 in LNAI, pages 416–420, Siena, Italy, 2001. Springer Verlag.

[95] Jana Giceva, Christoph Lange, and Florian Rabe. Integrating web services into active mathematical documents. In Carette et al. [70], pages 279–293. URL https://svn.omdoc.org/repos/jomdoc/doc/pubs/mkm09/jobad/jobad-server.pdf.

[96] Deyan Ginev, Constantin Jucovschi, Stefan Anca, Mihai Grigore, Catalin David, and Michael Kohlhase. An architecture for linguistic and semantic analysis on the arXMLiv corpus. In *Applications of Semantic Technologies (AST) Workshop at Informatik 2009*, 2009. URL http://www.kwarc.info/projects/lamapun/pubs/AST09_LaMaPUn+appendix.pdf.

[97] Birte Glimm and Chimezie Ogbuji. SPARQL 1.1 entailment regimes. W3C working draft, World Wide Web Consortium (W3C). URL http://www.w3.org/TR/2010/WD-sparql11-entailment-20101014/.

[98] George Goguadze. Metadata for mathematical libraries. Deliverable D3.a, MoWGLI, 2003. URL http://mowgli.cs.unibo.it/misc/deliverables/metadata/D3a_metadata_for_math/math_metadata.pdf.

[99] George Goguadze. Metadata model. Deliverable D3.b, MoWGLI, 2003. URL http://mowgli.cs.unibo.it/misc/deliverables/metadata/D3bmetadata_model/metadata_model.pdf.

[100] Timothy Gowers and Michael Nielsen. Massively collaborative mathematics. *Nature*, 461(15):879–881, 2009.

[101] Paul Grosso, Eve Maler, Jonathan Marsh, and Norman Walsh. W3c xpointer framework. W3C recommendation, World Wide Web Consortium (W3C), March 2003. URL http://www.w3.org/TR/2003/REC-xptr-framework-20030325/.

[102] Tudor Groza and Siegfried Handschuh. Salt annotation ontology. Technical report, Digital Enterprise Research Institute (DERI), 2009. URL http://salt.semanticauthoring.org/ontologies/sao.

[103] Tudor Groza and Siegfried Handschuh. Salt document ontology. Technical report, Digital Enterprise Research Institute (DERI), 2009. URL http://salt.semanticauthoring.org/ontologies/sdo.

[104] Tudor Groza and Siegfried Handschuh. Salt rhetorical ontology. Technical report, Digital Enterprise Research Institute (DERI), 2009. URL http://salt.semanticauthoring.org/ontologies/sro.

[105] Tudor Groza, Siegfried Handschuh, Knud Möller, and Stefan Decker. SALT – semantically annotated LaTeX for scientific publications. In Enrico Franconi, Michael Kifer, and Wolfgang May, editors, *The Semantic Web: Research and Applications*, number 4519 in Lecture Notes in Computer Science, pages 518–532. Springer Verlag, 2007. ISBN 978-3-540-72666-1.

[106] Tudor Groza, Knud Möller, Siegfried Handschuh, Diana Trif, and Stefan Decker. SALT: Weaving the claim web. In Karl Aberer, Key-Sun Choi, Natasha Fridman Noy, Dean Allemang, Kyung-Il Lee, Lyndon J. B. Nixon, Jennifer Golbeck, Peter Mika, Diana Maynard, Riichiro Mizoguchi, Guus Schreiber, and Philippe Cudré-Mauroux, editors, *ISWC/ASWC*, number 4825 in Lecture Notes in Computer Science, pages 197–210. Springer Verlag, 2007. ISBN 978-3-540-76297-3.

[107] Tudor Groza, Siegfried Handschuh, Tim Clark, Simon Buckingham Shum, and Anita de Waard. A short survey of discourse representation models. In Clark et al. [71]. URL http://CEUR-WS.org/Vol-523/.

[108] Ferruccio Guidi. *Searching and Retrieving in Content-based Repositories of Formal Mathematical Knowledge*. PhD thesis, Università di Bologna, 2003.

[109] Ferruccio Guidi and Claudio Sacerdoti Coen. Querying distributed digital libraries of mathematics. In Thérèse Hardin and Renaud Rioboo, editors, *Proceedings of the 11th Symposium on the Integration of Symbolic Computation and Mechanized Reasoning (Calculemus 2003)*, Rome, Italy, September 2003. URL http://www.calculemus.net/meetings/rome03/Proceedings/final.pdf.

[110] Ferruccio Guidi and Irene Schena. A query language for a metadata framework about mathematical resources. In Asperti et al. [40], pages 105–118.

[111] Thomas C. Hales, John Harrison, Sean McLaughlin, Tobias Nipkow, Steven Obua, and Roland Zumkeller. A revision of the proof of the kepler conjecture. *Discrete and Computational Geometry*, 2010.

[112] Kevin Hammond, Peter Horn, Alexander Konovalov, Steve Linton, Dan Roozemond, Abdallah Al Zain, and Phil Trinder. Easy composition of symbolic computation software: A new lingua franca for symbolic computation. In *Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation (ISSAC)*. ACM Press, 2010.

[113] Steve Harris and Andy Seaborne. SPARQL 1.1 query language. W3C Working Draft, World Wide Web Consortium (W3C). URL http://www.w3.org/TR/2010/WD-sparql11-query-20101014/.

[114] Olaf Hartig, Hannes Mühleisen, and Johann-Christoph Freytag. Linked data for building a map of researchers. In Bizer et al. [60]. URL http://CEUR-WS.org/Vol-449/.

[115] Philipp Heim, Steffen Lohmann, and Timo Stegemann. Interactive relationship discovery via the semantic web. In Aroyo et al. [38], pages 303–317.

[116] Bettina Heintz. *Die Innenwelt der Mathematik. Zur Kultur und Praxis einer beweisenden Disziplin*. Springer Verlag Wien, 2000.

[117] IEEE Learning Technology Standards Committee. Standard for Learning Object Metadata. Technical Report 1484.12.1, IEEE, 2002.

[118] IEEE Learning Technology Standards Committee. Standard for Resource Description Framework (RDF) binding for Learning Object Metadata data model. Technical Report 1484.12.4, IEEE, 2002.

[119] Prateek Jain, Pascal Hitzler, Peter Z. Yeh, Kunal Verma, and Amit P. Sheh. Linked data is merely more data. In Dan Brickley, Vinay K. Chaudhri, Harry Halpin, and Deborah L. McGuinness, editors, *Proceedings of the AAAI Spring Symposium on Linked Data Meets Artificial Intelligence*, 2010.

[120] Fairouz Kamareddine, Manuel Maarek, Krzysztof Retel, and J. B. Wells. Narrative structure of mathematical texts. In Manuel Kauers, Manfred Kerber, Robert Miner, and Wolfgang Windsteiger, editors, *Towards Mechanized Mathematical Assistants. MKM/Calculemus*, number 4573 in LNAI, pages 296–312. Springer Verlag, 2007.

[121] Fairouz Kamareddine, J. B. Wells, and Christoph Zengler. Computerising mathematical text with mathlang. *Electron. Notes Theor. Comput. Sci.*, 205:5–30, 2008. ISSN 1571-0661. doi: http://dx.doi.org/10.1016/j.entcs.2008.03.063. URL http://www.cedar-forest.org/forest/papers/drafts/mathlang-coq-short.pdf.

[122] Andrea Kohlhase and Michael Kohlhase. Semantic knowledge management for education. *Proceedings of the IEEE; Special Issue on Educational Technology*, 96(6):970–989, June 2008. URL http://kwarc.info/kohlhase/papers/semkm4ed.pdf.

[123] Andrea Kohlhase and Michael Kohlhase. Spreadsheet interaction with frames: Exploring a mathematical practice. In Carette et al. [70], pages 341–256. URL http://kwarc.info/kohlhase/papers/mkm09-framing.pdf.

[124] Michael Kohlhase. OMDoc – *An open markup format for mathematical documents [Version 1.2]*. Number 4180 in LNAI. Springer Verlag, August 2006. URL http://omdoc.org/pubs/omdoc1.2.pdf.

[125] Michael Kohlhase, Till Mossakowski, and Florian Rabe. Latin: Logic atlas and integrator. http://latin.omdoc.org. URL http://latin.omdoc.org.

[126] Werner Kunz and Horst W. J. Rittel. Issues as elements of information systems. Working paper 131, Institute of Urban and Regional Development, University of California, Berkeley, July 1970.

[127] Imre Lakatos. *Proofs and Refutations*. Cambridge University Press, 1976.

[128] Christoph Lange. Krextor – an extensible XML→RDF extraction framework. In Bizer et al. [60]. URL http://kwarc.info/projects/krextor/pubs/sfsw09-krextor.pdf.

[129] Christoph Lange. wiki.openmath.org – how it works, how you can participate. In Davenport [78].

[130] Christoph Lange. Towards OpenMath content dictionaries as linked data. In Michael Kohlhase and Christoph Lange, edi-

tors, *23rd OpenMath Workshop*, July 2010.

[131] Christoph Lange. The OMDoc document ontology, 2010. URL `http://kwarc.info/projects/docOnto/omdoc.html`.

[132] Christoph Lange. *Integrated Semantic Web Collaboration on Semiformal Mathematical Knowledge*. PhD thesis, Jacobs University Bremen, 2011. URL `https://svn.kwarc.info/repos/swim/doc/phd/phd.pdf`. submission expected in December 2010, draft available.

[133] Christoph Lange and Michael Kohlhase. A mathematical approach to ontology authoring and documentation. In Carette et al. [70], pages 389–404. URL `https://svn.omdoc.org/repos/omdoc/trunk/doc/blue/foaf/mkm09.pdf`.

[134] Christoph Lange, Uldis Bojārs, Tudor Groza, John Breslin, and Siegfried Handschuh. Expressing argumentative discussions in social media sites. In John Breslin, Uldis Bojārs, Alexandre Passant, and Sergio Fernández, editors, *Social Data on the Web (SDoW), Workshop at the 7th International Semantic Web Conference*, number 405 in CEUR Workshop Proceedings, 2008. URL `http://ceur-ws.org/Vol-405/paper4.pdf`.

[135] Christoph Lange, Tuukka Hastrup, and Stéphane Corlosquet. Arguing on issues with mathematical knowledge items in a semantic wiki. In Joachim Baumeister and Martin Atzmüller, editors, *Wissens- und Erfahrungsmanagement LWA (Lernen, Wissensentdeckung und Adaptivität) Conference Proceedings*, volume 448, October 2008.

[136] Christoph Lange et al. Krextor – the KWARC RDF extractor. URL `http://kwarc.info/projects/krextor/`.

[137] Paul Libbrecht. Cross curriculum search through the geoskills ontology. In D. Massart, J.-N. Colin, F. Van Asche, and M. Wolpers, editors, *Proceedings of the 2nd International Workshop on Search and Exchange of e-le@rning Material (SE@M), located at EC-TEL-2008*, CEUR Workshop Proceedings. CEUR-WS.org, 2008. URL `http://ceur-ws.org/Vol-385/`.

[138] Paul Libbrecht. Notations around the world: Census and exploitation. In Autexier et al. [46], pages 398–410.

[139] William C. Mann and Maite Taboada. Rhetorical structure theory. URL `http://www.sfu.ca/rst/`.

[140] Massimo Marchiori. The mathematical semantic web. In Asperti et al. [40]. Keynote.

[141] Jonathan Marsh, Daniel Veillard, and Norman Walsh. `xml:id` version 1.0. W3C recommendation, World Wide Web Consortium (W3C), September 2005. URL `http://www.w3.org/TR/2005/REC-xml-id-20050909/`.

[142] MathDox. MathDox – interactive mathematics. URL `http://www.mathdox.org`.

[143] Deborah L. McGuinness, Li Ding, Paulo Pinheiro da Silva, and Cyntha Chang. PML 2: A modular explanation interlingua. In Thomas Roth-Berghofer, Stefan Schulz, and David B. Leake, editors, *Proceedings of the AAAI Workshop on Explanation-Aware Computing (ExaCt)*, 2007. URL `http://www.aaai.org/Papers/Workshops/2007/WS-07-06/WS07-06-008.pdf`.

[144] Erica Melis, Giorgi Goguadze, Paul Libbrecht, and Carsten Ullrich. Culturally adapted mathematics education with ActiveMath. *AI & Society*, 24(3):251–265, 2009.

[145] MKMNET. MKMNET (Mathematical Knowledge Management Network). URL `http://monet.nag.co.uk/mkm/`.

[146] Monet. MONET – Mathematics on the net. web page at `http://monet.nag.co.uk`. URL `http://monet.nag.co.uk`.

[147] Till Mossakowski. Hets: the Heterogeneous Tool Set. URL `http://www.informatik.uni-bremen.de/cofi/hets/`.

[148] Till Mossakowski, Christian Maeder, and Klaus Lüttich. The heterogeneous tool set. In Orna Grumberg and Michael Huth, editors, *Proceedings of the 13th International Conference on Tools and Algorithms for the Construction and Analysis of Systems TACAS-2007*, number 4424 in LNCS, pages 519–522, Berlin, Germany, 2007. Springer Verlag.

[149] Christine Müller. *Adaptation of Mathematical Documents*. PhD thesis, Jacobs University Bremen, 2010. URL `http://kwarc.info/cmueller/papers/thesis.pdf`.

[150] Hala Naja-Jazzar, Nishadi de Silva, Hala Skaf-Molli, Charbel Rahhal, and Pascal Molli. Ontorest: A rst-based ontology for enhancing documents content quality in collaborative writing. *INFOCOMP Journal of Computer Science*, 8(3):1–10, September 2009.

[151] Mikael Nilsson, Andy Powell, Pete Johnston, and Ambjörn Naeve. Expressing Dublin Core metadata using the Resource Description Framework (RDF). DCMI recommendation, Dublin Core Metadata Initiative. URL `http://dublincore.org/documents/2008/01/14/dc-rdf/`.

[152] Mikael Nilsson, Matthias Palmér, and Jan Brase. The LOM RDF binding – principles and implementation. In *3rd Annual Ariadne Conference*, 2003.

[153] Tope Omitola, Christos L. Koumenides, Igor O. Popov, Yang Yang, Manuel Salvadores, Martin Szomszor, Tim Berners-Lee, Nicholas Gibbins, Wendy Hall, mc schraefel, and Nigel Shadbolt. Put in your postcode, out comes the data: A case study. In Aroyo et al. [38], pages 318–332.

[154] OpenLink Software. OpenLink universal integration middleware – Virtuoso product family. URL `http://virtuoso.openlinksw.com`.

[155] Christian-Emil Ore and Øyvind Eide. TEI and cultural heritage ontologies: Exchange of information? *Literary and Linguistic Computing*, 24(2):161–172, 2009.

[156] Luca Padovani. GtkMathView. URL `http://helm.cs.unibo.it/mml-widget/`.

[157] Luca Padovani and Stefano Zacchiroli. From notation to semantics: There and back again. In Borwein and Farmer [63], pages 194–207.

[158] Alexandre Passant, Paolo Ciccarese, John G. Breslin, and Tim Clark. Swan/sioc: Aligning scientific discourse representation and social semantics. In Clark et al. [71]. URL `http://CEUR-WS.org/Vol-523/`.

[159] PlanetMath.org. PlanetMath.org – math for the people, by the people. URL `http://planetmath.org`.

[160] Judith Plümer. Erinnerungen an Prof. Dr. Schwänzl, 2004. URL `http://d-mathnet.preprints.org/research/wissinfo/NachrufRS.html`.

[161] George Pólya. *How to Solve it*. Princeton University Press, 1973.

[162] Florian Rabe. *Representing Logics and Logic Translations*. PhD thesis, Jacobs University Bremen, 2008. URL `http://kwarc.info/frabe/Research/phdthesis.pdf`.

[163] Florian Rabe and Michael Kohlhase. A web-scalable module system for mathematical theories. Manuscript, to be submitted to the Journal of Symbolic Computation, 2010. URL https://svn.kwarc.info/repos/kwarc/rabe/Papers/omdoc-spec/paper.pdf.

[164] Ricardo Radaelli-Sanchez and Connexions. The intermediate CNXML. URL http://cnx.org/content/m9006/2.22/.

[165] Krzysztof Retel. *Gradual Computerisation and Verification of Mathematics*. PhD thesis, Heriot-Watt University, Edinburgh, April 2009.

[166] Andrew Robbins. Semantic MathML. URL http://straymindcough.blogspot.com/2009/06/semantic-mathml.html.

[167] Leo Sauermann and Richard Cyganiak. Cool URIs for the semantic web. W3C Interest Group Note, World Wide Web Consortium (W3C), December 2008. URL http://www.w3.org/TR/2008/WD-cooluris-20081203/.

[168] Irene Schena. *Towards a Semantic Web for Formal Mathematics*. PhD thesis, University of Bologna, March 2002. Technical Report UBLCS–2002–6.

[169] SCIEnce. The SCIEnce project – Symbolic Computation Infrastructure for Europe. URL http://www.symbolic-computation.org/.

[170] Neil J. A. Sloane. The on-line encyclopedia of integer sequences. *Notices of the AMS*, 50(8):912, 2003.

[171] Wolfram Sperber. Math-Net international and the Math-Net page. In Fengshan Bai and Bernd Wegner, editors, *Electronic Information and Communication in Mathematics*, number 2730 in Lecture Notes in Computer Science, pages 169–177. Springer, 2003.

[172] Manu Sporny, Benjamin Adrian, and Mark Birbeck. Rdfa api. W3C Working Draft, World Wide Web Consortium (W3C). URL http://www.w3.org/TR/2010/WD-rdfa-api-20100608/.

[173] Heinrich Stamerjohanns, Michael Kohlhase, Deyan Ginev, Catalin David, and Bruce Miller. Transforming large collections of scientific publications to XML. *Mathematics in Computer Science*, 3(3):299–307, 2010. URL http://kwarc.info/kohlhase/papers/mcs09.pdf.

[174] Carlos Tejo-Alonso, Sergio Fernández, Diego Berrueta, Luis Polo, María Jesús Fernández, and Víctor Morlán. eZaragoza, a tourist promotional mashup. In Adrian Giurca, Brigitte Endres-Niggemeyer, Christoph Lange, Lutz Maicher, and Pascal Hitzler, editors, *AI Mashup Challenge at ESWC*, June 2010. URL http://sites.google.com/a/fh-hannover.de/aimashup/home/ezaragoza.

[175] Christoph Tempich, H. Sofia Pinto, York Sure, and Steffen Staab. An argumentation ontology for DIstributed, Loosely-controlled and evolvInG Engineering processes of oNTologies (DILIGENT). In Asunción Gómez-Pérez and Jérôme Euzenat, editors, *The Semantic Web: Research and Applications*, number 3532 in Lecture Notes in Computer Science, pages 241–256. Springer Verlag, 2005. ISBN 3-540-26124-9.

[176] Christoph Tempich, Elena Simperl, Markus Luczak, Rudi Studer, and H. Sofia Pinto. Argumentation-based ontology engineering. *IEEE Intelligent Systems*, 22(6):52–59, 2007. ISSN 1541-1672.

[177] Jerzy Trzeciak. *Writing Mathematical Papers in English*. Gdańskie Wydawnictwo Oświatowe, 1995.

[178] Josef Urban, Jesse Alama, Piotr Rudnicki, and Herman Geuvers. A wiki for Mizar: Motivation, considerations, and initial prototype. In Autexier et al. [46], pages 455–469.

[179] Denny Vrandečić, Markus Krötzsch, Sebastian Rudolph, and Uta Lösch. Leveraging non-lexical knowledge for the Linked Open Data Web. In Rolodphe Héliot and Antoine Zimmermann, editors, *The Fifth RAFT, the yearly bilingual publication on nonchalant research*, 2010. URL http://km.aifb.kit.edu/projects/numbers/linked_open_numbers.pdf.

[180] Denny Vrandečić, Christoph Lange, Michael Hausenblas, Jie Bao, and Li Ding. Semantics of governmental statistics data. In *Proceedings of WebSci'10: Extending the Frontiers of Society On-Line*, 2010. URL http://journal.webscience.org/400/.

[181] Norman Walsh. The DocBook schema. Committee specification, OASIS, August 2008. URL http://www.docbook.org/specs/docbook-5.0-spec-cs-01.html.

[182] Norman Walsh and Leonard Muellner. *DocBook 5.0: The Definitive Guide*. O'Reilly, 2008.

[183] Makarius Wenzel, Clemens Ballarin, Stefan Berghofer, Timothy Bourke, Lucas Dixon, Florian Haftmann, Gerwin Klein, Alexander Krauss, Tobias Nipkow, David von Oheimb, Larry Paulson, and Sebastian Skalberg. *The Isabelle/Isar Reference Manual*. URL http://isabelle.in.tum.de/doc/isar-ref.pdf.

[184] Freek Wiedijk. Statistics on digital libraries of mathematics. *Studies in Logic, Grammar and Rhetoric*, 18(31), 2009.

[185] Wikipedia. Wikipedia, the free encyclopedia. URL http://www.wikipedia.org.

[186] Wikipedia: Knowledge management. Knowledge management, December 2009. URL http://en.wikipedia.org/w/index.php?title=Knowledge_management&oldid=329227520.

[187] Wikipedia: Mathematics. Portal: Mathematics. URL http://en.wikipedia.org/w/index.php?title=Portal:Mathematics&oldid=329137789.

[188] Wikipedia: Open Notebook Science. Open Notebook Science. URL http://en.wikipedia.org/w/index.php?title=Open_Notebook_Science&oldid=372235042.

[189] Wikipedia: Pythagorean theorem. Pythagorean theorem, November 2009. URL http://en.wikipedia.org/w/index.php?title=Pythagorean_theorem&oldid=328597679.

[190] Stefano Zacchiroli. *User Interaction Widgets for Interactive Theorem Proving*. PhD thesis, Università di Bologna, 2007.

[191] Jürgen Zimmer. *MathServe – A Framework for Semantic Reasoning Services*. PhD thesis, Universität des Saarlandes, July 2008.

[192] Antoine Zimmermann. Ontology recommendations for the data publishers. In Mathieu d'Aquin, Alexander García Castro, Christoph Lange, and Kim Viljanen, editors, *1st Workshop on Ontology Repositories and Editors*, number 596 in CEUR Workshop Proceedings, 2010. URL http://ceur-ws.org/Vol-596/.

[193] Claus Zinn. *Understanding Informal Mathematical Discourse*. PhD thesis, Technischen Fakultät der Universität Erlangen-Nürnberg, 2004.