Mathematical Knowledge Management: Transcending the One-Brain-Barrier with Theory Graphs

Michael Kohlhase (Jacobs University Bremen, Germany)

We present the emerging discipline of Mathematical Knowledge Management (MKM), which studies the possibility of computer-supporting and even automating the representation, cataloguing, retrieval, refactoring, plausibilization, change propagation and in some cases even application of mathematical knowledge.
We focus on theory graph technology here, which supports modular and thus space-efficient representations of mathematical knowledge and allows MKM systems to achieve a limited mathematical literacy that is necessary to complement the abilities of human mathematicians and thus to enhance their productivity.

1 Introduction

Computers and Humans have complementary strengths. While the former can handle large computations and data volumes flawlessly at enormous speeds, humans can sense the environment, react to unforeseen circumstances and use their intuitions to guide them through only partially understood situations. In mathematics we make use of this complementarity by letting humans explore mathematical theories and come up with novel proofs, while delegating symbolic/numeric computation and typesetting of documents to computers.

There is, however, one area where the specific strengths of computers are not commonly exploited in mathematics: the *management of existing mathematical knowledge*. In contrast to the creation of new knowledge, which requires (human) insights and intuition, the management (cataloguing, retrieval, refactoring, plausibilization, change propagation and in some cases even application) of mathematical knowledge can be supported by machines and can even be automated in the near future given suitable representation formats and algorithms.

With more than 100 thousand articles published annually, the times where a human could have an overview over – let alone have working knowledge in – all of mathematics are long gone. Even web-based information systems with global reach like the Cornell e-print arXiv [ArX] and reviewing services like the AMS reviews [MR] or Zentralblatt Math [ZBM] cannot prevent duplication of work and missed opportunities for the application of mathematical results.

Even if they make mathematical documents available at the click of a mouse, they cannot transcend a fundamental invariant: to do mathematics with any of that knowledge, it must pass through a human brain, which has a very limited capacity – compared to the volume, breadth and diversity of mathematical knowledge currently accessible in documents. We will call this the **one-brain-barrier** (**OBB**) of mathematics. Of course the OBB applies to all areas of human knowledge but the highly networked nature and rigorous presentation of mathematical knowledge make it a particularly appealing subject for studying the possibilities of machine support in knowledge management.

The effect of the OBB is particularly noticeable for the mathematical practice of *framing*, i.e. establishing that an object a (of class A) can be viewed as an incarnation of a class B of objects (via an *interpretation mapping* $\iota : B \to A$); see [KK09] for an introduction. Framing is exceedingly successful in establishing seemingly new mathematical knowledge in bulk: proving a single theorem T about objects b in B yields the theorem $\iota(T)$ about objects $\iota(b)$ in A "for free". Influential interpretations (where the theory of B is particularly rich) are often referred to as "representation theorems" as they cross-fertilise between mathematical areas by bridging represenational differences. But the method of framing is useful also in the small: we use it every time we apply a theorem, only that we usually do not state necessary interpretations explicitly but leave their reconstruction to the reader. The ability to to this without conscious thought is part of every mathematical education. The utility and ubiquity of framing lets some mathematicians consider it a defining characteristic of mathematical practice.

Note that the proof that any A can be interpreted to be a B may be highly non-trivial; but the problems with finding interpretations and representation theorems begin way before that: a prerequisite is that working knowledge about both As and Bs is co-located in one brain. There are two kinds of situation where interpretations are needed:

1. We have established a new result about a class B of objects and want to apply it to all known objects a that can be interpreted to be of class B.
2. We study object b (of class B) and want to see whether there are classes A it can be interpreted as, so that we can take advantage of the knowledge about A.

Both induce a search problem for an interpretation $\iota : B \to A$, where A ranges over "all mathematical knowledge". So, even if we assume that the knowledge about B is already available – a questionable assumption in 2 – the OBB hits us with full force. Note that even though the proof of the representation theorem for ι usually requires (human) insight and intuition, the application of the frame (the transport of the knowledge from B to A) is a largely syntactical process, which can be machine-supported.

In this article, we want to survey the field of Mathematical Knowledge Management, its basic ideas and methodological basis in general (next section), then focus on a subset of methods that put the notion of framing at the centre (Section 3) and show how they can be used to transcend the one-brain barrier in Section 4. Section 5 concludes the article.

2 Mathematical Knowledge Management

Mathematical Knowledge Management (see [MKM; Far11]) is a young research area at the intersection of mathematics, artificial intelligence, computer science, library science and scientific publishing. The objective of MKM is to develop new and better ways of managing sophisticated mathematical knowledge based on innovative technology of computer science, the internet and intelligent knowledge processing.

MKM is expected to serve mathematicians, scientists and engineers who produce and use mathematical knowledge; educators and students who teach and learn mathematics; publishers who offer mathematical textbooks and disseminate new mathematical results; and librarians who catalogue and organise mathematical knowledge.

Even though MKM focuses on (MK)M (the "management of mathematical knowledge"), its methods touch on M(KM) (the "mathematics of knowledge management"). It is a fundamental tenet of MKM that important aspects of knowledge can be formalised into a state where their management can be supported by machines. Conversely, it is assumed that *only* by making such aspects of knowledge formal – e.g. by semantics extraction – will they become amenable to automation. As a consequence, semantisation is an important concern for MKM.

As the "client list" of MKM above already suggests, the notion of knowledge management is rather inclusive. Correspondingly, the aspects of knowledge studied by MKM and the "depth of formalisation" are varied, ranging from full formalisation of the mathematical content in a logical system [Har08; Wie08] to machine-readable versions of the Mathematics Subject Classification [Lan+12].

It has always been an aim of the MKM community to build a digital mathematical library – envisioned to be universal in [Far11]. Based on this, sophisticated mathematical software systems could help humans articulate, organise, disseminate and access mathematical knowledge to allow them to concentrate on those parts of doing mathematics that need human cognitive facilities – creating new mathematical knowledge and appreciating the beauty of the existing canon.

In this paper we focus on ways to structure a digital mathematical library into modular theory graphs and explain salient mathematical practices in this setting. Our exposition here follows [RK13], which introduces OMDoc/MMT, currently the strongest formulation of theory graph ideas, which were initially introduced in [FGT92] and extended to a comprehensive representation format for mathematical knowledge and documents (OMDoc: Open Mathematical Documents [Koh06]).

3 Theory Graphs & Computer-Supported Framing

The main idea of the theory graphs paradigm in MKM is to take the idea of interpretations seriously and use it as a structuring principle in a modular representation methodology for mathematical knowledge. Neither the modularity principle nor the use of interpretations are particularly new in mathematics – they have been the main structural mechanism behind Bourbaki's systematic redevelopment of mathematics. But the MKM community is developing ways of formalising them and is making use of these structures for machine support. We will now explore the basic notions and show how these relate to the mathematical practice of framing.

Theory graphs build on two central notions: theories (the nodes of the graph) and theory morphisms (the edges). The former are just collections of symbols denoting indivisible mathematical objects and axioms stating the assumptions this theory makes about them.

> **Theory morphisms** are mappings from symbols in the source theory to expressions in the target theory, such that all axioms are mapped to theorems of the target theory. (*)

Figure 1 shows a simple example of a theory graph. It consists of a theory monoid of monoids, which has the symbols op for the binary composition operation and unit for its neutral element. Symbols and axioms (which we omit in Figure 1 for simplicity) have tripartite global names of the form g?t?c in OMDoc/MMT, where g is the URL of the document that contains the theory t and c is the name of a symbol in t. Note that such tripartite names (**MMT URIs**) are valid uniform resource identifiers, which is a crucial prerequisite for being used in web-based MKM systems. In Figure 1, we have assumed that all theories are in the same document and we can therefore omit the URL g throughout.

Inheritance Now, a theory cgp of Abelian groups can be obtained from a monoid by adding an inverse operation inv. In OMDoc/MMT we can avoid duplication of representations by utilising an inclusion morphism (visualised by an arrow $\hookrightarrow$) that "copies" all symbols and axioms from the source theory monoid to cgp. Note that inclusion morphisms are trivially theory morphisms, since the target is defined to make them so. OMDoc/MMT treats incoming inclusions like symbols and names them in their target theory, so that the inclusion of the monoid in Figure 1 is globally accessible to MKM systems via the MMT URI g?cgp?mon. The inherited symbols are available in cgp via the name of the inclusion that supplied them, e.g. composition as mon/op or globally as g?cgp?mon/op. Axioms are inherited similarly.

In the next step we can obtain a theory ring via two inclusions: ring?add and ring?mul (and adding a distributivity axiom). Note that ring has two distinct binary operations: the additive ring?add/mon/op and the multiplicative ring?mul/op, two neutral elements: ring?add/mon/unit and ring?mul/unit, and the additive inverse ring?add/inv. Note that the respective copies of the axioms of mon and cgp are inherited as well. ring inherits eight axioms: two closure, two associativity and two neutrality axioms, plus additive inverse and commutativity. We see that OMDoc/MMT can mimic the efficiency of mathematical vernacular in this respect.

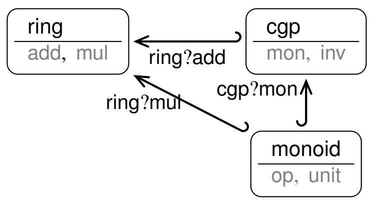


Figure 1: A simple Theory Graph

Notation Before we go on with our exploration of Figure 1, a clarification of the level of representation is in order. We have seen that the use of inclusion morphisms gives us theory structures that systematically supply URIs for the objects we expect. We should think of OMDoc/MMT as a kind of "machine language" for representing mathematical knowledge. In particular, URIs as names are good for communication of knowledge over the internet and its manipulation by machines but they are woefully inadequate for communication with humans, who expect to see, for example, $\circ$ instead of MMT URI g?cgp?mon/op in Abelian groups and may use $+$ for ring?add/mon/op and $\cdot$ for ring?mul/op in rings. To account for this, OMDoc/MMT offers a system of **notation def-**

initions that allow us to associate human-oriented notations with symbols and regain the familiar and helpful notations of mathematical discourse. Indeed, making notation definitions first-class citizens of the representation format allows us to model them and their effects in modular theory graphs: they are inherited along inclusions and thus available wherever the inherited symbols are visible. In MKM systems that communicate with humans, we use notation definitions to generate formula presentations for the user but keep the symbol URIs to render additional services, e.g. definition lookup.

Views We have seen that inclusions can be used for a modular development of, for example, the elementary algebraic hierarchy. But the real power of theory graphs comes from another kind of theory morphisms: **views**. In contrast to inclusions they link two pre-existing theories and thus we have to prove that all the source axioms are theorems of the target theory to establish the theory morphism condition (*). Moreover, views usually have non-trivial symbol mappings. In Figure 2 we extend the theory graph from Figure 1 with a theory of integers and two views. v_1 maps the monoid operation to integer addition and the neutral element to the integer number zero. To establish this mapping as a view we have to establish (*) by proving closure of $\mathbb{Z}$ under $+$, associativity of $+$, and $x + 0 = x$ for all $x \in \mathbb{Z}$ (the **proof obligations** induced by v_1) from the axioms of integers.

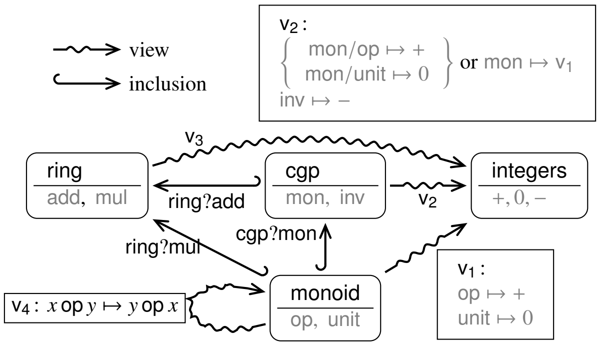


Figure 2: A Theory Graph with Views/Examples

The view v_2 is similar, only that it additionally maps the inverse operation to $-$ and has proof obligations for the inverse axiom and commutativity. Note that v_2 can be made modular by mapping the inclusion mon to v_1 instead of the direct mapping of (inherited) symbols. This also allows us to inherit/reuse the proof obligations of v_1 in v_2. A view v_3 from ring to integers can be established similarly.

Theory Morphisms Transport Proofs and Theorems But more important than the mechanics of theory graphs is the realisation that in the presence of a theory morphism $S \xrightarrow{\sigma} T$ by (*) any S-theorem t becomes a T-theorem $\sigma(t)$ as the S-proof π of t can be translated along σ: any appeal to an S-axiom in π becomes an appeal to a T-theorem in $\sigma(\pi)$ by (*). In particular, $\sigma(t)$ can be referred to as $T?[v]/\mathsf{t}$, just like induced symbols, only that v is the MMT URI of the view σ.

In this context, note that the pragmatics of **structures** (inclusions with non-trivial mappings) and views are complementary. Structures are constitutive to a theory and restrict the "applicability" of a theory, since they introduce axioms that must be fulfilled by proof obligations, while views are (logically) redundant but allow us to "apply" a theory elsewhere. The subgraph induced by structures must be acyclic, while views may induce cycles. Indeed, theory isomorphisms (pairs of inverse theory morphisms) are a good way to represent equivalent theories, alternative definitions, etc.

Views at Work in Mathematics In our example in Figure 2, this means that all group theorems apply to $(\mathbb{Z}, +)$ (and all theories that are reachable from it via theory morphisms). This already explains one mathematical practice: proving a conjecture in the greatest possible generality. In theory graph terms, this means establishing a theorem as low as possible (in terms of the pre-order induced by theory morphisms) in the graph, since the "cone of influence" gets bigger this way.

Another mathematical practice is to abbreviate proofs by meta-arguments, e.g. "(*iv*) the proof of this case follows from case (*iii*) by symmetry". In most cases, such proofs by meta-arguments can be seen as making the reader aware of a theory-endomorphism like v_4 that can be used to transport the proof of (*iii*) into one of (*iv*). Finally, mathematical examples can be interpreted as theory morphisms in theory graphs (another consequence of (*)). In our example, $(\mathbb{Z}, +, 0)$ is an example of a monoid by v_1, $(\mathbb{Z}, +, 0, -)$ is an example of an Abelian group via v_2 and $(\mathbb{Z}, +, 0, -, *, 1)$ is an example of a ring via v_3. An example becomes "non-obvious" if it has at least one view component.

Framing In conclusion, we note that theory morphisms are *very natural candidates* for representing the structures underlying the *mathematical practice of framing* highlighted in the introduction. Indeed, we can see the condition (*) as being the essence of framing. We are currently exploring the logical and cognitive consequences of this idea. It seems that theory morphisms directly account for representation theorems like Stone's theorem or Ken Ribet's link from the Taniyama-Shimura conjecture to Fermat's last theorem, which made Wiles' proof possible. But framing also seems to be at work in less obvious places, e.g. in mathematical synonyms. The concepts of "nodes" and "edges" in a graph are often called "points" and "lines", borrowing terminology from geometry. I believe that this metaphoric usage of names is licensed by a partial view from (linear, ordered) geometry to theory graphs. A problem in the study of framing is that it is a cultural skill of mathematics and that mathematical literacy requires mastering framing to a level where it becomes almost subconscious, and therefore hard to observe via introspection.

MKM for Theory Graphs Even our simple example in Figure 2 shows that theory graphs are very information-rich objects whose structure can be used to explain many mathematical practices. But the full power of theory graphs only comes out when they are implemented in a software system that supports these practices. OMDoc/MMT has been implemented in the MMT API [Rab13], which supports services including notation-definition-based presentation (see above), incremental **flattening** (i.e. computation of the symbols, axioms and theorems of a theory S induced by the theory graph), as well as type- and proof-checking (an important aspect of MKM we have neglected in this article).

Such systems allow us to make use of the space efficiency afforded by the modular representation in theory graphs: we only need to represent (and store) a small subset of the knowledge available to an experienced mathematician. In our experience, the induced knowledge outweighs the represented by a factor of 50/1 (if it is finite at all) even for relatively small theory graphs. Note that the quotient of the size of the induced knowledge over the represented knowledge is a good measure not only for the space efficiency of the representation system but also for the mathematical literacy of a human. Experienced mathematicians induce much more (implicit) knowledge from what they read than inexperienced ones – because the former, it is conjectured, have more densely connected mental theory graphs at their disposal. OMDoc/MMT provides standardised identifiers for all induced knowledge items and the MMT API can compute their values on demand and reason with them. In this sense, the MMT API can be considered to possess a certain amount of "mathematics literacy" that allows it to render higher-level mathematical services.

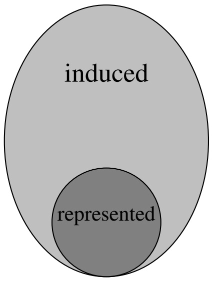


Figure 3: Knowledge Addressable in MMT

4 Overcoming the One-Brain-Barrier

We have seen above that with theory graph technology, MKM systems can achieve a certain level of mathematical literacy even without the ability for automated reasoning – another important aspect of MKM we will not cover in this article. But in the foreseeable future it seems clear that no machine will even come close to a human mathematician's creative ability of inventing new mathematics. But we can make use of the high storage capacity and cheap computation that MKM offers. It is claimed that even with limited mathematical literacy, machines can (*i*) complement human mathematicians and (*ii*) even uncover – rather than discover – novel mathematics by overcoming the OBB, if we have theory graphs that contain more knowledge than the average human mathematician. In this section, we will assume such theory graphs will become available and sketch one example for (*i*) and two for (*ii*) – which may provide motivation for organising a community effort to build such graphs. Starting points could be theorem prover libraries like the Mizar library [Miz], which has more than 1000 theories which together contain more than 50,000 formal theorems and definitions (see Figure 4 for the inheritance graph – the Mizar theory language only supports inclusions). We are currently in the process of exporting half-a-dozen theorem prover libraries into the OMDoc/MMT format to create a large theory graph (the Open Archive of Formalizations; OAF) to experiment on.

Complementing Humans by Searching the Induced Knowledge Space The idea is very simple: we use a formula search engine like MathWebSearch (see [KMP12; Koh+13]) and

Figure 4: The Mizar Inheritance Graph

instead of indexing all the represented knowledge items with their respective URIs, we extend this to the set of induced items and their MMT URIs (which the MMT API can compute by flattening). Consider the situation in Figure 5, where we are searching for the associativity formula $X + (Y + Z) = (X + Y) + Z$, and MathWebSearch returns the MMT URI http://latin.omdoc.org/math?IntAryth?assoc, which – together with the underlying theory graph – contains enough information to generate an explanation of the reason + is associative on $\mathbb{Z}$.

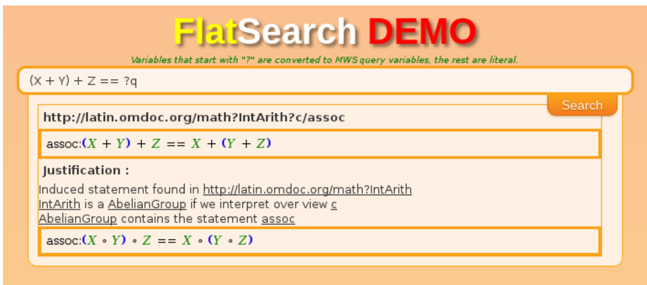


Figure 5: Searching for Induced Knowledge Items

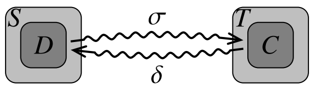
In essence, this experimental search engine searches the space of (induced) mathematical knowledge rather than just the space of mathematical documents.

Making Bourbaki Accessible Ideas like the ones above can solve one of the problems with the Bourbaki book series, which is written in such a very concise and modular manner that it can only be understood if one has all the previous parts in memory. We have extracted the theory graph underlying the first 30 pages of Algebra I [Bou74]. It contains 51 theories, 94 inclusions and 10 views. The theories contain 82 Symbols, 38 axioms, 30 theorems and 17 definitions [Lau07]. For knowledge items higher up in the graph, there is no (single) place in the book which states all their axioms or properties. With the MMT API, we can generate flattened descriptions for reference and with FlatSearch we can search for their properties. We conjecture that the simple explanation feature from Figure 5 can be extended into a "course generator" that generates a self-contained – modulo the reader's prerequisite knowledge – document that explains all aspects necessary for understanding the search hits. With theory graph technology, Bourbaki's Elements can be read by need, as a foundational treatise should be, instead of being restricted to beginning-to-end reading.

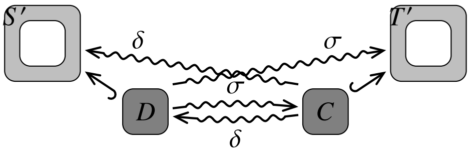
Uncovering Theorems We can now come back to the discussion on the OBB from the introduction. One way to overcome the problem of missing interpretations is to systematically search for them – after all, machine computation is comparatively cheap and the number of potential theory morphisms is bounded by the square of the number of theories times the number of symbols in theories. In particular, the MathWebSearch formula search engine [KMP12] can efficiently search for substitutions (which are essentially the same as the symbol-to-expression mappings of theory morphisms). We have explored this idea before the theory graph technology was fully developed, and the TheoScrutor system [Nor08] found a considerable number of simple views in the Mizar library. While these were relatively syntactic and obvious – after all, for a **view finder** to work, the proofs for the proof obligations have to be part of the library already – they had not previously been noticed because of the OBB, even though the Mizar project has a "library committee" tasked with finding such linkages. We expect that once the OAF is sufficiently stable, a renewed experiment will yield many more, and more interesting views and possibly novel theorems – after all, the HOL Light and Isabelle Libraries contain Tom Hales' proof of the Kepler Conjecture and the Coq library contains the proof of Feit-Thomson's Odd-Order Theorem.

Note that once a set of views has been found by the viewfinder, we can generate the induced theorems in all accessible theories and iterate the view finding process, which might find more views with the induced theorems.

Refining Theory Graphs Finally, even partial views – which should be much more numerous than total ones in a theory graph – can be utilised. Say we have the situation below with two theories S and T, a partial theory morphism $S \xrightarrow{\sigma} T$ with domain D and codomain C, and its partial inverse δ.



Then we can pass to the following more modular theory graph, where $S' := S \backslash D$ and $T' := T \backslash C$. In this case we think of the equivalent theories D and C as the intersection of theories S and T along σ and δ. Note that any views out of S and T now have to be studied, if they can be pulled back to C and D.



We have observed that many of the lesser known algebraic structures in Bourbaki naturally arise as theory intersections between better known structures. We hope to explain the remaining ones via other category-theory-inspired theory graph transformation operations.

Note that operations like theory intersections apply theory graph technology to the problem of theory graph maintenance, which is itself a problem greatly hampered by the OBB without MKM techniques.

5 Conclusion

In this paper we have explored opportunities to lift the one-brain barrier in mathematics, which limits the application of mathematical knowledge both inside the mathematical domain as well as in other disciplines. We propose that the way forward is to employ computer systems that can systematically explore immense knowledge spaces, if these are represented in sufficiently content-oriented formats. Together with the creation, curation and application of digital mathematical libraries (DMLs), this is one of the central concerns of the new field of Mathematical Knowledge Management (MKM).

We have presented the theory graphs approach as a representation paradigm for mathematical knowledge that allows us to make its modular and highly networked structure explicit and therefore machine-actionable. We have seen that theory graphs following the "little theories" approach contain the information structures necessary to explain – and thus ultimately support by computer – many mathematical practices and culture skills. This has the potential to significantly extend the "mathematical literacy" of mathematical knowledge management systems and consequently make them more suitable as tools that complement human skills.

We have explored three exemplary mathematical applications of theory graph technologies and one MKM-internal to give an intuition of what services we can expect if we embark on the enterprise of representing large bodies of mathematical knowledge and its network structures in machine-actionable formats. The availability of such DMLs is currently the largest bottleneck for overcoming the OBB in mathematics. We are currently experimenting with establishing an open archive of formal mathematics (the OAF project [OAF]) by integrating theorem prover libraries. But formalisation often poses a high burden on the author and forces decisions about logical foundations that are irrelevant mathematically. Therefore, we are currently researching ways the theory graph methods presented here can be extended to representations of mathematical knowledge in which the degree of formalisation is flexible. Flexiformal representations – see [Koh13] for a discussion – are much closer to mathematical vernacular, which mixes informal parts (natural language) with formal parts (e.g. formulae and functional markup for mathematical statements) and are therefore easier to obtain in practice. But mathematical literacy may be limited by the availability of formal/machine-actionable parts; therefore, we are additionally investigating methods for automated semantics-extraction from mathematical documents, which would greatly enhance the reach of the methods described in this paper.

Acknowledgements Work on the concepts presented here has been partially supported by the Leibniz association under grant SAW-2012-FIZ_KA-2 and the German Research Foundation (DFG) under grants KO 2428/9-1 and KO 2428/13-1.

References

[ArX] `arxiv.org` *e-Print archive*. URL: http://www.arxiv.org (visited on 06/12/2012).

[Bou74] Nicolas Bourbaki. *Algebra I*. Elements of Mathematics. Springer Verlag, 1974.

[Far11] William M. Farmer. "Mathematical Knowledge Management". In: *Encyclopedia of Knowledge Management*. Ed. by David Schwartz and Dov Te'eni. 2nd ed. Idea Group Reference, 2011, pp. 1082–1089.

[FGT92] William M. Farmer, Josuah Guttman, and Xavier Thayer. "Little Theories". In: *Proceedings of the 11th Conference on Automated Deduction*. Ed. by D. Kapur. LNCS 607. Saratoga Springs, NY, USA: Springer Verlag, 1992, pp. 467–581.

[Har08] John Harrison. "Formal Proof – Theory and Practice". In: *Notices of the AMS* 55.11 (2008), pp. 1395–1406. URL: http://www.ams.org/notices/200811/tx081101395p.pdf.

[Jeu+12] Johan Jeuring et al., eds. *Intelligent Computer Mathematics*. Conferences on Intelligent Computer Mathematics (CICM). (Bremen, Germany, July 9–14, 2012). LNAI 7362. Berlin and Heidelberg: Springer Verlag, 2012.

[KK09] Andrea Kohlhase and Michael Kohlhase. "Spreadsheet Interaction with Frames: Exploring a Mathematical Practice". In: *MKM/Calculemus Proceedings*. Ed. by Jacques Carette et al. LNAI 5625. Springer Verlag, July 2009, pp. 341–356. URL: http://kwarc.info/kohlhase/papers/mkm09-framing.pdf.

[KMP12] Michael Kohlhase, Bogdan A. Matican, and Corneliu C. Prodescu. "MathWebSearch 0.5 – Scaling an Open Formula Search Engine". In: *Intelligent Computer Mathematics*. Conferences on Intelligent Computer Mathematics (CICM). (Bremen, Germany, July 9–14, 2012). Ed. by Johan Jeuring et al. LNAI 7362. Berlin and Heidelberg: Springer Verlag, 2012, pp. 342–357. URL: http://kwarc.info/kohlhase/papers/aisc12-mws.pdf.

[Koh+13] Michael Kohlhase et al. "Zentralblatt Column: Mathematical Formula Search". In: *EMS Newsletter* (Sept. 2013), pp. 56–57. URL: http://www.ems-ph.org/journals/newsletter/pdf/2013-09-89.pdf.

[Koh06] Michael Kohlhase. OMDoc – *An open markup format for mathematical documents [Version 1.2]*. LNAI 4180. Springer Verlag, Aug. 2006. URL: http://omdoc.org/pubs/omdoc1.2.pdf.

[Koh13] Michael Kohlhase. "The Flexiformalist Manifesto". In: *14th International Workshop on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2012)*. Ed. by Andrei Voronkov et al. Timisoara, Romania: IEEE Press, 2013, pp. 30–36. URL: http://kwarc.info/kohlhase/papers/synasc13.pdf.

[Lan+12] Christoph Lange et al. "Reimplementing the Mathematics Subject Classification (MSC) as a Linked Open Dataset". In: *Intelligent Computer Mathematics*. Conferences on Intelligent Computer Mathematics (CICM). (Bremen, Germany, July 9–14, 2012). Ed. by Johan Jeuring et al. LNAI 7362. Berlin and Heidelberg: Springer Verlag, 2012, pp. 458–462. arXiv: 1204.5086 `[cs.DL]`.

[Lau07] Bastian Laubner. "Using Theory Graphs to Map Mathematics: A Case Study and a Prototype." MA thesis. Bremen: Jacobs University, Aug. 2007. URL: https://svn.eecs.jacobs-university.de/svn/eecs/archive/msc-2007/blaubner.pdf.

[Miz] *Mizar*. URL: http://www.mizar.org (visited on 02/27/2013).

[MKM] *International Conference on Mathematic Knowledge Management*. URL: http://www.mkm-ig.org/ (visited on 08/12/2012).

[MR] *Mathematical Reviews*. URL: http://www.ams.org/mr-database (visited on 12/16/2013).

[Nor08] Immanuel Normann. "Automated Theory Interpretation". PhD thesis. Bremen, Germany: Jacobs University, 2008. URL: https://svn.eecs.jacobs-university.de/svn/eecs/archive/phd-2008/normann.pdf.

[OAF] *The OAF Project*. URL: http://mathhub.info/OAF-project (visited on 01/25/2014).

[Rab13] Florian. Rabe. "The MMT API: A Generic MKM System". In: *Intelligent Computer Mathematics*. Conferences on Intelligent Computer Mathematics. (Bath, UK, July 8–12, 2013). Ed. by Jacques Carette et al. Lecture Notes in Computer Science 7961. Springer, 2013, pp. 339–343. DOI: 10.1007/978-3-642-39320-4.

[RK13] Florian Rabe and Michael Kohlhase. In: *Information & Computation* 0.230 (2013), pp. 1–54. URL: http://kwarc.info/frabe/Research/mmt.pdf.

[Wie08] Freek Wiedijk. "Formal proof – Getting Started". In: *Notices of the AMS* 55.11 (2008), pp. 1408–1414. URL: http://www.ams.org/notices/200811/tx081101408p.pdf.

[ZBM] *Zentralblatt MATH*. URL: http://www.zentralblatt-math.org/zbmath/ (visited on 06/12/2012).

Michael Kohlhase [m.kohlhase@jacobs-university.de] is a full professor for computer science at Jacobs University Bremen and an associate adjunct professor at Carnegie Mellon University. He studied pure mathematics at the Universities of Tübingen and Bonn (1983–1989) and continued with computer science, in particular higher-order unification and automated theorem proving (PhD 1994, Saarland University). His current research interests include knowledge representation for mathematics, inference-based techniques for natural language processing and computer-supported education.