Navigation in Mathematical Documents

Andrea Kohlhase

Computer Science, Jacobs University Bremen
`http://kwarc.info`

Abstract. Mathematical documents are not only hard but fun to write, they are equally hard but fun to read. Unfortunately, the "fun" part comes only in, when one has mastered the art of using mathematical vernacular and grasped all relevant context dimensions. Even though this seems an immanent issue for mathematical authors only, it is also for mathematical readers. In this paper, we argue for the need for more reader assistance in mathematical documents. In particular, we believe navigation to be a top candidate in this regard. Thus, we elaborate design opportunities for (semantically) helping the reader to navigate, which we anchor around the PlanetMath website.

1 Introduction

In recent years, the traditional notion of "document" changed drastically from a finished product accessible only to a selected group of people to a potentially open access, widely distributable, collaborative and flexible artifact. Documents are considered modern if they are **active documents**, i.e., documents that use a presentation engine to (re- or inter-)actively adapt the surface structure of the document to the environment or user input. Spreadsheets are the paradigmatic example for such an active document type: If a spreadsheet author adds a column to a spreadsheet, all cell references in underlying formulas are automatically updated by the presentation engine. Active documents exploit the distinction between form and content (like cell layout vs. computed cell values in spreadsheets).

Information on the Web is also often encoded in active documents (e.g. HTML- or XML-files). But Web documents frequently carry yet another property which non-Web documents don't have: The linear communication structures of traditional documents are replaced by multidimensional, multifunctional and multimedial hypertext structures, in which information is distributed according to the "Net Communication Model" (see e.g. [Flu96] or Sect. 2.1).

The central idea of this paper is the observation, that mathematical documents rely heavily on the Net Communication Model, which makes them hard to read without supporting services. Thus, we want to suggest potential reading services for modern math documents.[1] We first point out the underlying Net

---

[1] Please note that this is a workshop paper aiming at discussions around this topic. Thus, we do not consider the ideas fully worked out yet.

Communication Model in mathematical documents (Sect. 2). In order to read (and understand) a mathematical document, this either requires a highly active reader's mind or active documents that offer elaborate reader assistance. In Sect. 3 we look at an exemplary such service, which draws on the Net Communication Model in a specific category of active math documents: Semantic navigation in spreadsheets. Finally, we envision more navigational features in active mathematical documents like articles on the PlanetMath website[2] in Sect. 4.

2 The Net Communication Model in Math Documents

The idiosyncracies of mathematical vernacular have often been discussed. Many researchers started out hoping to find a nice model for it, so that the process of either reifying a human's mathematical knowledge into reviewable documents ("codification") or converting existing math documents into formal documents ("formalization") could be supported or even automated. The same kind of problem arises when math educators teach math: young mathematicians have to learn "*. . . becoming conversant in the language of mathematical discussion.*" [Day11].

Mathematical vernacular turns a math document into an **intellectual artifact**, i.e. according to [dS05, p. 10],

- *"it encodes a particular understanding or interpretation of a problem situation*
- *it also encodes a particular set of solutions for the perceived problem situation*
- *the encoding of both the problem situation and the corresponding solutions is fundamentally linguistic (i.e., based on a system of symbols [. . .]); and*
- *the artifact's ultimate purpose can only be completely achieved by its users if they can formulate it within the linguistic system in which the artifact is encoded"*

In particular, DE SOUZA points out that intellectual artifacts require that producer and consumer use the same language. Math documents as intellecutal artifacts make use of a linguistic encoding system. As mathematical vernacular itself does not seem to be too systematic (otherwise all the research already done would have resulted in finding this system), it only seems to support the encoding system.

Therefore, in this section, we take a look at the underlying discourse structure of mathematical vernacular and find an underlying Net Communication Model.

2.1 The Net Communication Model (NCM)

Media-theorist VILEM FLUSSER suggested several discourse models for communication in [Flu96]. He distinguishes four communication discourse types:

Pyramid The **pyramid discourse** preserves information very well, as it is centered around getting confirmation about every obtained and possibly recoded information by its sender. Edited articles can serve as an example for this discourse type, as the information is checked by the author in the final proof copy before it is published.

---

[2] `www.planetmath.org`, relaunched shortly based on Planetary services (see [?])

Theatre The **theatre discourse** is characterized by a sender and a receiver exchanging information, protected against communication noise and thus misunderstanding by a concave theater screen. As long as the actors constrain to the rule to be within the protection area of the screen while communicating, the information entropy is very low. A document handed from the author to colleague in his field fits this description. The safeguarding screen here is the resp. Community of Practice (CoP) (see [LW91]).

Graph The tree or rather **graph discourse**, FLUSSER describes as sciences' discourse. The original sender (=author) creates information to be communicated, this information is broken down into information fragments, that in turn are picked up, re-coded and further communicated. FLUSSER calls it "centrifugal information distribution" and points to its inherent information creation while not conserving the information's original content. In many related work sections in scientific documents we can find such distributed information fragments, that give the setting for innovative ideas, i.e., new information.

Amphitheatre The final discourse type, FLUSSER presents is the **amphitheatre discourse**. Here, the information is neither send by one sender (but by many) nor is there a screen available against which the communication happens. This is the modern, web discourse type, the **Net Communication Model (NCM)**. The information entropy is extremely high, but information is communicated extremely fast and broadly.

2.2 The NCM in Math Documents

Now we want to look at the discourse type of math documents.

- **Pyramid:** As it is very important for mathematicians to conserve the "objectivity" and "truth" of statements, this safest-of-all discourse type seems highly attractive for mathematical communication in documents. Publication in a journal is still the most honorable publication format for mathematicians, and here the author indeed carefully checks its proof before it is published. But this is only a check of the form. The real question concerns a document's content, that is, whether used information fragments by other authors are checked by those authors? Here, the answer is no. In particular, the pyramid discourse type seems not to fit math documents.
- **Theatre:** The "objectivity" and "truth" in math documents are kept by a more or less rigid reviewing system. This is organized within the concerned Community of Practice, which serves as a theatre screen for the safety of the communication of mathematical information. Therefore, we conclude at first glance that mathematicians prefer to practice this communication discourse type. Unfortunately, this type does not support information creation, but mathematicians love the creativeness in math, for example:

  > "*It is somewhat remarkable that a subject with such high and objective standards of logical correctness should at the same time provide such opportunity for the expression of playful genius. This is what attracts*

> *many people to the study of mathematics, particularly those of us who have made it our life's work.*" [Day11]

  So the theatre discourse type doesn't seem to fit too well to math documents.

- **Graph:** But the graph discourse type supports creativeness, so we can consider this a fitting type for math documents? One of the most successful tools of mathematicians is their practice of aligning different discourses according to their purpose, for instance:

  > "*Mathematics is about* ideas*. In particular it is about the way that different ideas relate to each other.*" [Ste90, p. 2]

  Moreover, modularization is another key mathematical practice, which enables fine-tuned aligning of theories. We conclude, that mathematicians use the graph discourse type in their documents.

- **Amphitheatre:** But how do mathematicians preserve "objectivity" and "truth" in their documents at the same time? They screen the information by their assumption of a math document to be self-contained. In particular, any reader can persuade herself of the truthfulness of inferences/conclusions. Therefore, even if a miscommunication happened in the discourse at some point, the derived statements are true in their local context. STEWART gave an example of the vernacularous depth of mathematics by letting a fictious mathematician (in a discussion with a fictional layman) exclaim

  > "*You can't get a feeling for what's going on without understanding the technical details. How can I talk about manifolds without mentioning that the theorems only work if the manifolds are finite-dimensional para-compact Hausdorff with empty boundary?*" [Enz99, p. 45]

  This example demonstrates that the self-containedness has its natural limits. Those links outside math documents are part of the amphitheatre discourse type.

In summary, math documents are built on basis of the amphitheatre discourse type, but including parts of the theatre and the graph discourse types locally. In particular, they yield the Net Communication Model with local creativity-enabling provenances and local safe-guards.

3 Math Reader Assistance: Semantic Document Navigation

The difficulty of reading math documents becomes clear directly: To understand the locally used theatre screens, the reader has to be either part of the resp. Community of Practice or has to has access to its knowledge in all of its dimensions (see e.g. [KKL10]). Moreover, math documents frequently contain holes that refer to the implicit knowledge of the resp. CoP, which makes that specific part in the document potentially ambiguous based on the graph discourse type. This means that readers may fill in the missing parts incorrectly or not at all. Therefore, we argue for more reader assistance in mathematical documents in general.

3.1 Navigation

The showcased issues centered around a reader's orientation problems, thus, we consider navigation support a central service for these readers. In [IM00] navigation was introduced as a service that allows users "*to browse a huge document [...] without problems*". Originally, especially in the Nautics discipline, it describes the "*process of monitoring and controlling the movement of a craft or vehicle from one place to another*" [Wik11]. When transferred to the "*sea of information*" (e.g. [McI03]) available on the Web, navigation becomes the (elaborate) process of steering towards looked-for information goals.

Dourish and Chalmers introduced in 1994 the term "**semantic navigation**". In particular they stated that the

> *spatial organisation of data has been a highly visible component of a number of information systems.[...] Indeed, it's the notion of spatial arrangements which encourages (and legitimises) the notion of navigation of information systems. However, in navigation (as opposed to organisation), this use of the "spatial" is a convenient gloss for a different organisation, which we refer to here as "semantic".[DC94]*

They realized that in hypertext systems the primary form of navigation is of a semantic nature, in particular, depending on the semantic properties of existing links. Going beyond one-dimensional hyperlinks we therefore like to transfer this well-known notion of navigation to non-Web, active documents.

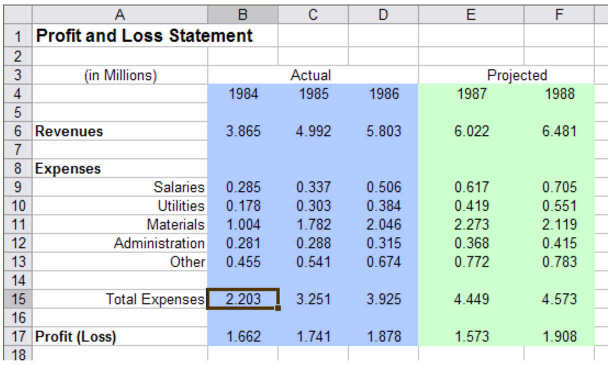
3.2 Semantic Document Navigation in `SACHS`

A possible way to provide reader assistance was presented for spreadsheets as active, mathematical documents in [KK11]. Even though they do not contain the usual mathematical vernacular, they also operate in the same discourse type fashion. Their local screens are built for the data of the document, their local provenance as well. Underlying formulae and charts are computed resp. drawn on the spot, so that the presented data are locally "true" data. The reader's interpretation of the data depend on the locally given, provenanced-dependent context information like headers. The known usability problems wrt. to spreadsheets were analyzed to be either false formulae, false data interpretation or input data errors (e.g. [KK10]).

We realized with the `SACHS` system [KK11] a semantic help system for spreadsheet documents, which includes interesting document navigation features, that we summarize next.

Let us start with introducing our running example document (see Fig. 1): An MS Excel spreadsheet based on [Win06], which can be considered a simple controlling system. It shows a profit-loss statistics over a time period, including cost and revenue data. The purpose and meaning of the spreadsheet seem clear enough, but as soon as we ask

- for which company this is a controlling system, or



	A	B	C	D	E	F
1	**Profit and Loss Statement**					
2						
3	(in Millions)		Actual		Projected	
4		1984	1985	1986	1987	1988
5						
6	**Revenues**	3.865	4.992	5.803	6.022	6.481
7						
8	**Expenses**					
9	Salaries	0.285	0.337	0.506	0.617	0.705
10	Utilities	0.178	0.303	0.384	0.419	0.551
11	Materials	1.004	1.782	2.046	2.273	2.119
12	Administration	0.281	0.288	0.315	0.368	0.415
13	Other	0.455	0.541	0.674	0.772	0.783
14						
15	Total Expenses	2.203	3.251	3.925	4.449	4.573
16						
17	**Profit (Loss)**	1.662	1.741	1.878	1.573	1.908
18						

Fig. 1. A Simple Controlling System Using MS Excel after [Win06]

– whether the numbers are given in millions of Dollars or Yen, or
– what the is definition of "projected data"

we realize right away that even in such a simple document the explicit knowledge is just the tip of the iceberg of the (background) knowledge necessary to interpret it. The `SACHS` system was designed to draw on a semi-formal formalization of this background knowledge as domain ontology.

The spreadsheet objects that carry meaning are the cells. They are interpreted by the user both wrt. the grid layout (like within a table with an assigned row and column specification) and via the underlying formula. With SACHS we offer a third dimension of interpretation by providing access to the background knowledge based on the alignment of cells with concepts in a domain ontology. In particular, we realized an embedded user assistance method by using cell clicks as entry points for the help system, that is, every click on a cell generates help.

Here, we are only interested in the help features "Dependency Graph" and "Search" box, since these two allow the user semantic document navigation.

The Dependency Graph

Let us first look at what happens when the user clicks on cell [B15] (Total Expenses, 1984). Then a new window (as seen in Fig. 2) is opened displaying at the top the concept connected to the selected cell. All concepts, which this top concept directly depends on, are shown on the second level.

If the user wants to elaborate on a specific concept like "Utility Costs", then a click on the corresponding node expands it by another level. The user is free to drill down into ever more abstract information available in the domain ontology; see Fig. 3 for an example path.

In a nutshell, the dependency graph enables the user to explore the background knowledge according to her own mental map of the concerned knowledge, her experience with it, her situation-dependent interests and time-frames. Cells are reinterpreted as hyperlinks to a domain ontology and moreover, the nodes in the dependency graphs are themselves links to further concepts in this ontology.

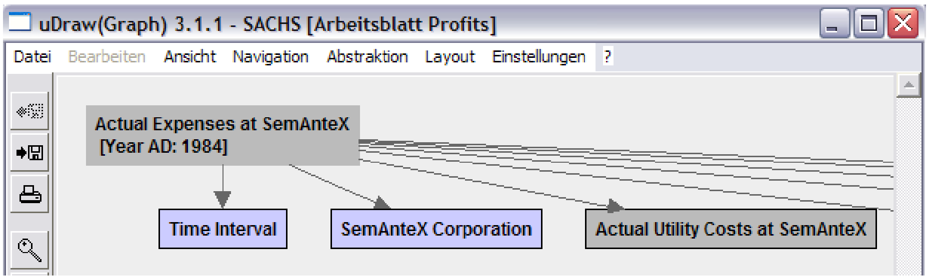


Fig. 2. Dependency Graph for Cell [B15]: Level 1 and 2

Iske [Isk02] distinguishes three navigational options based on a Network Communication Model: un- and directed browsing, and guided tours. A spreadsheet reader can explore the document and its previously hidden knowledge on her own, therefore SACHS enables *undirected browsing* as navigational service. Moreover, it offers *directed browsing* features as it is designed around the central semantic object for the interpretation of data within a spreadsheet — the cell (see [KK09]). The author of the spreadsheet and presumably of the resp. background knowledge governs the form of the dependency graph, thus the graph feature in SACHS can be also considered to promote paths or *guided tours* as a navigation option.

One can argue that this kind of navigation is mostly happening in the “document behind the original document”: the background knowledge captured in a domain ontology. But the navigation functionality was also extended to cross-modality navigation by taking the alignment of concepts and cells as semantic document navigation cues.

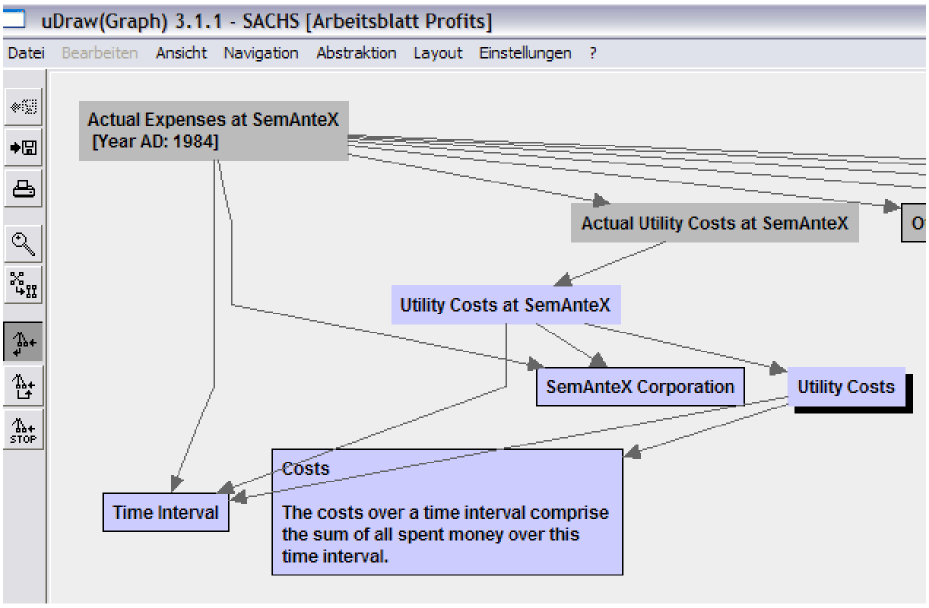
In SACHS we mashed-up the graph-based interface with spreadsheet focus operations to enable **spreadsheet navigation** via the definitional structure of the intended functions in the structured background ontology. Note that the nodes e.g. in Fig. 3 have distinct colors. This color-coding indicates whether the concept in a node is connected to a specific cell in the workbook: An aligned concept node in the dependency graph carries a hyperlink to the resp. cell.

Let us look at Fig. 3 for example. There are two nodes in darker grey: “Actual Expenses at SemAnteX” and “Actual Utility Costs at SemAnteX”. Clicking the node “Actual Utility Costs at SemAnteX” moves the spreadsheet focus to cell [B10]. Then the user can switch back to the original position in the spreadsheet by clicking the top node, here the “Actual Expenses at SemAnteX” node.

This way a user can get a good *orientation* on how the spreadsheet works and an overview over the various dependencies between cells.

Searching in the Background Knowledge

The search feature enables the user to reach through to the information available in background knowledge.



The costs over a time interval comprise the sum of all spent money over this time interval.

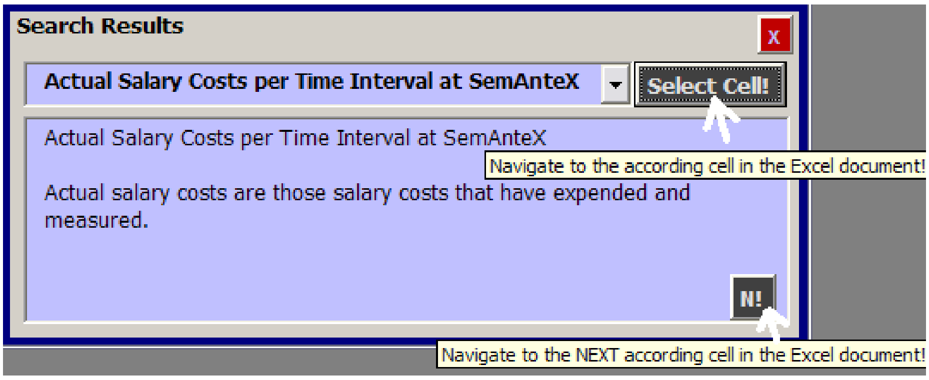
Fig. 3. Dependency Graph for Cell [B15]: More Levels

Concretely, users can type into the search box a string of characters which is used to search all concept titles for this string.

Once an element in the search result is selected by a user, the according explanation is presented in this area. For instance, she likes to know more about the concept “Steady State Prognosis”, so she clicks on the according item in the list and its definition is shown (see Fig. 4). The concept “Actual Salary Costs per Time Interval at SemAnteX” is aligned to the cell [B9] in our running example. A click on the “Select Cell!” button would thus result in a navigation to this aligned cell.

On the lower right hand side of the “Search Results” area we can also find the “N!” command button. Pushing this results, on the one hand, in the selection of the next aligned concept in the search list, and on the other hand, in the navigation to the resp. cell in the according spreadsheet. The order is determined by the current cursor position within the list downwards and on user request starting on its top again. This feature enables the user who is unfamiliar with the concepts generally or concept titles particularly to judge by herself whether the shown concept is the one she was looking for to begin with.

In summary, we observe that both the concept graph as well as the hit list (the list of search results) can be used by the user as “navigation panels”. The former is arguably more semantic (as it uses the conceptual dependency structure), whereas the latter is more mnemonic (it goes via the concept names). Their utility is largely due to the fact that they allow the user to focus on a particular



Actual Salary Costs per Time Interval at SemAnteX

Actual Salary Costs per Time Interval at SemAnteX

Navigate to the according cell in the Excel document!

Actual salary costs are those salary costs that have expended and measured.

Navigate to the NEXT according cell in the Excel document!

Fig. 4. Explanation for “Actual Salary Costs at SemAnteX” from Background Knowledge

aspect — the dependency cone of concepts leading up to the value of a particular cell or the concepts mentioning a particular string — and use these as a jump list for complex document collections. Indeed, commentators on the SACHS system always highlighted the added-value of being able to disregard worksheet and potentially even workbook limitations when navigating.

4 Math Reader Assistance of the Future: More Navigation

In order to envision more helpful navigation features, we imagine them for a web-based collection of documents as gathered on the PlanetMath website [**?**].

4.1 Articles Graph

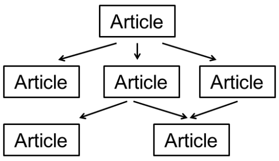


Fig. 5. Articles Graph

Consider for example the short article `http://planetmath.org/PrimeAn1.html`. The blue strings indicate links to other PlanetMath articles. The representation of the dependencies of this article on the other articles as dependency graph on the side, would give a clear indication of what this article is about. In contrast to the semantic document navigation in `SACHS`, the development of the graph level by level doesn’t seem sensible as the user can do this by her own by using the hyperlink structure itself. But using this given hyperlink structure the reader quickly looses the overview and maybe doesn’t even remember where she started at (think of the “magical number 7” in interaction design rules). A graph would be very helpful, in particular when deciding that a followed branch wasn’t worth the effort.

4.2 Bibliographic Graph

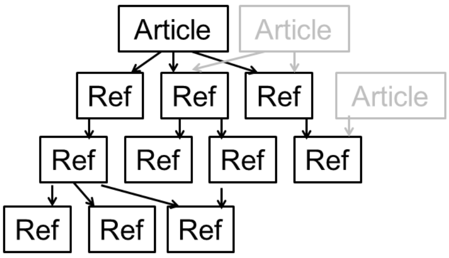


Fig. 6. Bibliographic Graph

Whenever one sees a list of bibliographic references, one is wondering which of all are the important ones. A hint could be delivered if the bibliographic references were presented in a "bibliographic graph", a graph containing all the references of an article and all recursively discovered references for each reference. The important ones are the likeliest to be referenced from the present references themselves (somewhat in analogy to Google's pagerank algorithm). If other articles referring to some of the present references, then this information may also be included into a graph. Naturally, this information could also be used to rank the references of an article.

4.3 Formula Graph

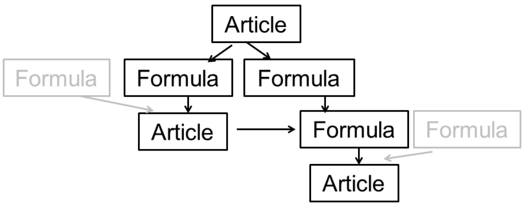


Fig. 7. Formula Graph

Other objects that appear frequently in math documents are formulae. If we take these serious as semantic objects, then a graph visualization also seems sensible. With the recent progress in math search facilities, structurally equivalent formulae can be found in other articles. Moreover, subformulae used in other articles can be discovered as well. If an article uses the same formula as the original article and additional ones, then these can also be used as a hyperlink in a respective graph.

5 Conclusion

In this paper we address the issue of the Net Communication Model together with some math specific extensions as a discourse model for math documents. This specific model makes it on the one hand hard to write and read math documents, but on the other hand it also provides the means to create a document that can only be understood from a holistic point of view. Once the author or the reader have been enabled to enjoy this view, fun is also part of the consuming process for that document. The holistic view depends on several dimensions of the underlying Net Communication Model. A math document can only be created if the author has mastered these dimensions, but not every reader has done so. Here, navigational features appear naturally as help services, as they serve as access points to multi-dimensional information. We gave an example for semantic document navigation in spreadsheet documents via the `SACHS` system and envisioned some more navigational design opportunities for a collection of active, web-based math documents in the PlanetMath system.

We believe that such reader assistance will be particularly useful and gives them new, and efficient access to salient parts of math documents. In turn, this will induce a better overview and a deeper understanding of the concepts in users — and for some even enable an enjoyable reading experience of math documents.

References

Day11. Martin V. Day. *An Introduction to Proofs and the Mathematical Vernacular.* Online at http://www.math.vt.edu/people/day/ProofsBook/IPaMV.pdf, 2011.

DC94. Paul Dourish and Matthew Chalmers. Running out of space - models of information navigation. In *Proc. HCI*, Glasgow, UK, 1994.

dS05. Clarisse Sieckenius de Souza. *The Semiotic Engineering of Human-Computer Interaction.* The MIT Press, 2005.

Enz99. Hans Magnus Enzensberger. *Drawbridge Up.* A K Peters, Ltd., 1999.

Flu96. Vilem Flusser. *Kommunikologie.* Bollmann, Mannheim, 1996.

IM00. Naoko Ito and Kazuhisa Manabe. XML Document Navigation Language, 2000. http://www.w3.org/TR/xdnl.

Isk02. Stefan Iske. *Vernetztes Wissen.* Bertelsmann, Bielefeld, 2002.

KK09. Andrea Kohlhase and Michael Kohlhase. Semantic transparency in user assistance systems. In Brad Mehlenbacher, Aristidis Protopsaltis, Ashley Williams, and Shaun Slatterey, editors, *Proceedings of the 27th annual ACM international conference on Design of communication (SIGDOC)*, pages 89–96, New York, NY, USA, 2009. ACM Special Interest Group for Design of Communication, ACM Press.

KK10. Andrea Kohlhase and Michael Kohlhase. What we understand is what we get: Assessment in spreadsheets. In Simon Thorne, editor, *Symp. of the European Spreadsheet Risks Interest Group (EuSpRIG 2010)*, pages 111–121. European Spreadsheet Risk Interest Group, 2010.

KK11. Andrea Kohlhase and Michael Kohlhase. Spreadsheets with a semantic layer. *Electronic Communications of the EASST: Specification, Transformation, Navigation – Special Issue dedicated to Bernd Krieg-Brückner on the Occasion of his 60th Birthday*, 2011. accepted.

KKL10. Andrea Kohlhase, Michael Kohlhase, and Christoph Lange. Dimensions of formality: A case study for MKM in software engineering. In 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, pages 355–369. Springer Verlag, 2010. http://arxiv.org/abs/1004.5071.

LW91. Jean Lave and Etienne Wenger. *Situated Learning: Legitimate Peripheral Participation (Learning in Doing: Social, Cognitive and Computational Perspectives S.).* Cambridge University Press, 1991.

McI03. Neil McIntosh. World drowning in a rising sea of information. *The Guardian*, November 2003. http://www.guardian.co.uk/technology/2003/nov/01/internationalnews.onlinesupplement.

Ste90. Ian Stewart. *From Here to Infinity.* Oxford University Press, 1990.

Wik11. Wikipedia. Navigation — wikipedia, the free encyclopedia, 2011. [Online; accessed 20-September-2011].

Win06. Terry Winograd. The spreadsheet. In Terry Winograd, John Bennett, Laura de Young, and Bradley Hartfield, editors, *Bringing Design to Software*, pages 228–231. Addison-Wesley, 1996 (2006).