Building an Information Service for Mathematical Software – the SMATH Project

Sebastian Bönisch, Gert-Martin Greuel, Wolfram Sperber

We report about a project to create, with the help of Zentralblatt MATH, a comprehensive open access database SMATH that will provide broad coverage of mathematical software and, at the same time, will guarantee high quality and relevance of the referenced software packages.

Present Situation and Final Goal

Scientific publications are no longer the only form of mathematical knowledge. During the past three decades, mathematical software has become an increasingly important tool for mathematical education and research. It is often an indispensable tool in many active areas of current mathematical research fields like numerics or statistics. But mathematical software is also used in some parts of pure mathematics, to check hypotheses, to construct counter-examples, or to illustrate and sometimes even prove new mathematical results. Moreover, since mathematical software is operational knowledge, it is the most important link between mathematics and its applications.

For mathematical publications, comprehensive online collections and information services are available, as well as established classification systems and standard citation schemes. Unfortunately, the same is not true for mathematical software, where the information situation is much less complete. While there are many specialized online collections of mathematical software, a unified and commonly accepted approach to information retrieval is still missing. The SMATH project aims at filling this gap.

There are three main differences between SMATH and existing software collections or portals. SMATH

- intends to be fairly complete,
- links the software with the publications using it,
- links the publication with the used software,

where the links are realized at the moment with the help of the publication database Zentralblatt MATH.

Difficulties Encountered

Nowadays, every student of mathematics or a related discipline is trained to become familiar with computer algebra systems and numerical software packages like Maple[1], Mathematica[2] or MATLAB[3]. But there are many more, in fact several thousand, mostly specialized software systems and packages in use. The specific requirements of potential software users differ very much depending on their individual background and the character of the problem to be solved. However, certain basic questions constantly arise in almost any context:

- How to find a suitable software for a given mathematical problem?
- Which mathematical methods does the software employ for solving that problem?
- How about the quality of a software?
- For which problem classes was the software used successfully in the past and by whom?
- What are the license terms for a certain software package? Is the source code available, e.g. “open source” or not?
- Which programming languages are used?
- Which interfaces to other systems does the software provide?
- Is the software compatible with a certain operating system?

A detailed description of a software package necessarily has to be much more complex than is the case with an article. In addition to describing the goals and areas of application of the software, one has to specify technical parameters (such as programming languages, operating systems, interfaces etc.), license terms or terms of use, as well as information about the current version. Finding all these metadata for a certain package can be very involved, since information about software is typically inhomogeneous and spread across different media.

In many cases, part of the required information can be found in certain special articles. The ideal case, however, is to have an informative and up-to-date website which is carefully maintained by the software developers and at any time reflects the current state of the software. Often the amount of information available depends heavily on the size and the development status of a software package. For many small and highly specialized mathematical software libraries, there is no such thing as a describing website.

Another problem is the dynamic character of any software development process. Unlike bibliographical metadata, information about software is subject to frequent changes during the whole software life cycle. Not only the technical parameters but also the areas of application of a software can change, as can the software authors (i.e. the developer team). These considerations may explain of why there is still no commonly accepted and widely used standard for the description of mathematical software resources, even though there is no lack of theoretical work or standardization attempts.

The SMATH Project

The SMATH project is scheduled for three years (2011-2013). It is directed by the second author of this article and jointly carried out by Mathematisches Forschungsinstitut Oberwolfach (MFO)[4] and Fachinformationszentrum Karlsruhe (FIZ)[5]. MFO is

[1] http://www.maplesoft.com/products/maple/
[2] http://www.wolfram.com/mathematica/
[3] http://www.mathworks.de/products/matlab/index.html
[4] http://www.mfo.de/
[5] http://www.fiz-karlsruhe.de

already developing a database of selected high quality software packages, called ORMS, while FIZ is responsible for developing and hosting the publication database Zentralblatt MATH. Further institutions with proven expertise in the area of mathematical software support the project: Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB)[6], the DFG Research Center MATHEON[7], the Weierstraß-Institute for Applied Analysis and Stochastics (WIAS)[8], and the Felix-Klein-Zentrum für Mathematik (FKZ)[9] in Kaiserslautern.

Identification and detailed description of as many mathematical software packages as possible are essential project goals. But SMATH goes one step further: For any software package referenced in SMATH, all mathematical publications referring to that software will be identified and referenced, too.

There are essentially two categories of such publications. The first category consists of articles written by the main software developers with a view to presenting the software to the public ("developer's article", see Figure 1). Such articles typically contain a more or less detailed description of a certain software. The second category comprises articles written on a certain mathematical topic by an author who used an existing software package to obtain or illustrate his results and who therefore cites the software as an auxiliary tool ("user's article", see Figure 2).

[MATRIX] Zbl 0902.14040
Greuel, G.-M.; Pfister, G.; Schönemann, H.
SINGULAR: A computer algebra system for singularity theory, algebraic geometry and commutative algebra. (English)
Comput. Sci. J. Mold. 5, No.1, 3-9 (1997).

SINGULAR, as pointed out by the authors is a computer algebra system designed especially for commutative algebra, algebraic geometry and singular theory. For almost all computations SINGULAR requires a base ring. Computations are possible over such base rings as: polynomial rings, localizations of polynomial rings, quotient of such rings, exterior algebras, Weyl algebras and tensor products. The main algorithms are very general standard Gröbner basis algorithms, and include algorithms for the usual ideal theoretic operations, for different syzygy problems, for computing invariants of singularities and semiuniversal deformations of isolated singularities. SINGULAR contains several libraries to which the user may add more procedures.
Reviewer: Aigli Papantonopoulou (Ewing)

MSC 2010
14Q99 Computational aspects of algebraic geometry
13P99 Computational aspects of commutative algebra
68W30 Symbolic computation and algebraic computation

Keywords
SINGULAR

Figure 1. Example of a "developer's article" on the SINGULAR software (source: Zentralblatt MATH).

[MATRIX] Zbl 0932.13021
De Jong, Theo
An algorithm for computing the integral closure. (English)
J. Symb. Comput. 26, No.3, 273-277 (1998).

Finding efficient algorithms for computing the integral closure of a commutative noetherian ring R is one of the big challenges of computational algebra. The paper under review contains an algorithm that is based on a result by Grauert and Remmert, which has as a consequence the fact that, if $I \subset R$ is a radical ideal which contains a non-zero divisor, then R is normal if and only if $R = \mathrm{Hom}\ R(I,I)$. Several known criteria for normality used in existing algorithms are special cases of this. The algorithm terminates if the normalization is finitely generated as an R-module, e.g., for affine rings. It has been implemented in MACAULAY and SINGULAR.
Reviewer: Reinhard Laubenbacher and Abdul Jarrah (New Mexico)

MSC 2010
13P99 Computational aspects of commutative algebra
13B22 Integral closure; integrally closed rings, related rings
13E05 Noetherian rings and modules
68W30 Symbolic computation and algebraic computation
13-04 Machine computation, programs (commutative algebra)

Keywords
reduced noetherian ring; computing the integral closure; algorithm; MACAULAY; SINGULAR

Figure 2. Example of a "user's article" on the SINGULAR software (source: Zentralblatt MATH).

At the moment we use mainly the database of Zentralblatt MATH for finding articles referring to a given software, but an extension to other data sources would of course be possible. The use of Zentralblatt has three main advantages:

- First of all, the database will be updated continuously and automatically whenever a new software or version is cited in an article indexed by Zentralblatt. Since all indexed articles are peer-reviewed, this is sufficient to guarantee the quality of the software.
- Every user who is searching for information about some software is linked to all articles indexed in Zentralblatt MATH where this software is used. These articles provide valuable information about the software which is available nowhere else.
- Everyone who reads a review in Zentralblatt has immediate access to basic information about the software mentioned via a link to SMATH.

It is an interesting and useful observation that we can use the articles in Zentralblatt MATH to identify a large number of additional software packages. To this end, we developed and implemented heuristic tools for the automatic identification of software-relevant articles in Zentralblatt MATH and elsewhere. Examples of such automatically identified articles are:
"KNITRO: an integrated package for nonlinear optimization."
"PDETool: A multi-formalism modelling tool for discrete-event systems based on SDES description."
"QuBE++: An efficient QBF solver."

The key ingredient of the heuristics is the fact that articles describing software often exhibit a characteristic title structure: The titles of such articles often contain software-specific keywords (e.g. "software", "package", "solver" etc.) as well as the name of the software package. The characteristic spelling of many software names and their position within the title (mostly at the beginning) can be used for automatic extraction. So far we identified about 3,500 software packages by means of heuristic analysis tools.

Further Schedule

In the first year the SMATH team defined a provisional metadata model for mathematical software and has set up the general framework and the technical infrastructure for SMATH. A focus of the second year is on further filling and consolidating the database. To this end, the publication-based heuristics approach will be combined with the input and specific know-how of the cooperating institutes; they will add relevant software from their fields of expertise to SMATH. For this purpose they will develop tools for the semantic analysis of software websites in order to reduce the manual effort to obtain software metadata. Another focus of the current project year is on the design and implementation of a functional and ergonomic web user interface for the future SMATH users.

The first preliminary production version of SMATH is scheduled for mid-2012.

[6] http://www.zib.de/
[7] http://www.matheon.de/
[8] http://www.wias-berlin.de
[9] http://www.felix-klein-zentrum.de/