System Description: MathWeb, an Agent-Based Communication Layer for Distributed Automated Theorem Proving

Andreas Franke and Michael Kohlhase

FB Informatik, Universität des Saarlandes
afranke|kohlhase@ags.uni-sb.de

1 Introduction

Real-world applications of theorem proving require open and modern software environments that enable modularization, distribution, inter-operability, networking, and coordination. This system description presents the MathWeb[1] approach for distributed automated theorem proving that connects a wide-range of *mathematical services* by a common, *mathematical software bus*. The MathWeb system provides the functionality to turn existing theorem proving systems and tools into mathematical services that are homogeneously integrated into a networked proof development environment. The environment thus gains the services from these particular modules, but each module in turn gains from using the features of other, plugged-in components.

2 Implementation

The MathWeb system is an object-oriented toolbox that provides the functionality for building a society of software agents that render mathematical services by either encapsulating legacy deduction software or their own functionality. In the current implementation the software bus functionality is realized by a model quite similar to the *Common Object Request Broker Architecture* (CORBA [Sie96]) in which a central *broker* agent provides routing and authentication information to the mathematical services (see [SHS98] for details). The agents are realized in a distributed programming system mOZart[2], which provides the full infrastructure to write distributed applications.

Furthermore, MathWeb provides the mOZart shell (Mosh), a tool for launching and administering multiple mOZart applications (the agents) within only one mOZart process. It combines some frequently used shell commands (for files, processes and environment) with some (thread-related) mOZart commands. These allow (remotely) administering the mathematical services across the Internet, since the administrator can connect to remote Mosh demons[3], launch and terminate services. This also allows for a limited form of self-organization of mathematical services, since these can use Mosh scripts themselves to launch and administer other services.

[1] The system is available at `http://www.ags.uni-sb.de/~omega/www/mathweb.html`.
[2] See `http://mozart.ps.uni-sb.de`
[3] which run continually at the host providing the services

mOZArts main advantage as a basis for MathWeb comes from its network transparency, i.e., the full support of remote computations in the base language (lexical scoping, logical variables, objects, constraints,...), and its network awareness, i.e., the full control over network operations, such as the choice between stationary and mobile objects, which make it easy to 'agentify' arbitrary applications.

3 Existing Mathematical Services

In this section we will briefly list and categorize the currently available mathematical services.

Automated Theorem Provers MathWeb currently features the first-order theorem provers bliksem, EQP, Otter, ProTeIn, Spass, WaldMeister (see [SS97] for references), and the higher-order systems TPS [ABI+96] and $\mathcal{LEO}$ [BK98]. Furthermore, there is a service `competitive-atp` that calls sets of ATP concurrently as competing services (this strategy is known to yield even super-linear speedups in practice).

Computer Algebra Systems There are services wrapping the systems Maple, MagMa, GAP and μCAS (see [KKS98] for references). Here, the MathWeb approach is particularly interesting, since a licensee of commercial software systems like Maple and MagMa can export the corresponding services to the deduction community.

Mediators are mathematical services that transform mathematical knowledge from one format to another. The agent-oriented MathWeb approach allows to encapsulate the zoo of conversion programs currently available[4] to generally available mathematical services and avoid duplication of efforts.

Proof Transformers are rather substantial mediators that transform between proof formats. Currently MathWeb features a proof transformation service from the proof formats of the theorem provers mentioned above [HF96,Mei99] to the natural deduction calculus.

Knowledge bases MathWeb currently only includes the MBase service, a simple web-based mathematical knowledge base system that stores mathematical facts like theorems, definitions and proofs and can perform type checking, definition expansion and semantic search. It communicates with other mathematical services by mediators and with humans by the interaction unit OctOpus.

Human Interaction Units are MathWeb services that provide visualization and control features for the user interaction. Currently, MathWeb includes the $\mathcal{L}\Omega\mathcal{UI}$ graphical user interface for interactive theorem provers [SHB+98], the OctOpus front-end for MBase and the ProVerb proof presentation system [HF96], which can transform ND proofs to natural language. The Doris system (see section 4) is a MathWeb client from outside the domain of deduction systems.

[4] e.g. at the TPTP library `http://wwwjessen.informatik.tu-muenchen.de/~tptp`

4 Applications and Experiences

ΩMEGA*:* The work reported in this paper originates in the development effort of the ΩMEGA-system [BCF+97], a mathematical assistant system with the ultimate goal of supporting theorem proving in main-stream mathematics and mathematics education. To provide the necessary reasoning and symbolic computation facilities this system incorporates most the mathematical services listed in section 3. The MATHWEB approach has been a key factor in keeping the system maintainable [SHS98,FHJ+99] and the near future will see further modularization and agentification of system components, which will lead to simpler system maintenance and a more open development model.

DORIS*:* Apart from this application, MATHWEB has been tested in the DORIS[5] system, a natural language understanding system that uses first-order automated theorem provers and model builders as external mathematical services to solve the consistency and entailment problems pertaining to various disambiguation problems in text and dialogue understanding. DORIS generates between 1 and ca. 500 deduction problems for each sentence it processes, distributes them to competing mathematical services (over a network of workstations) and collects the results to obtain the desired result. Using the MATHWEB approach, the integration of the theorem provers was very simple: the only new parts were a socket connection from Prolog on the DORIS side and a new service module for the `doris` service[6] on the MATHWEB side. Experience with this application shows that distribution using MATHWEB does not come for free: A test with ca. 1300 DORIS deduction queries yielded the following timings:[7]

30–1250 ms pure theorem proving time

50-120 ms spent in the service module (opening an inferior shell, creating files,...). This depends strongly on the efficiency of the server file system.

5–500 ms Internet latency (we have measured inter-department (in Saarbrücken) and international (Saarbrücken/Amsterdam) connections)

However, the large number of deduction problems and the possibility of coarse-grained parallelization by distribution lead to a significant increase in overall system performance, compared to an earlier centralized, sequential architecture. In particular, the timings also show that it can pay off for a client in Saarbrücken to delegate deduction problems to faster machines in Amsterdam or vice versa.

5 Conclusion and Future Work

The MATHWEB system provides a transport layer for distributed theorem proving and set of mathematical services, which will grow over time. The authors

[5] See `http://www.coli.uni-sb.de/~bos/atp/doris.html` for a web-based interface that acts as a MATHWEB client

[6] I.e. a small (60 line) MOZART program that relays problems, results and statistics between the DORIS program and the `competitive-atp` service.

[7] These times have been measured on a collection of SUN Ultra machines running Solaris 5 in Saarbrücken and Amsterdam (all timings given in total elapsed time; normalized to our fastest machine, a SUN Ultra 4 at 300 MHz).

would like to encourage the automated deduction community to supply further mathematical services.[8]

The current CORBA-like distribution model in MathWeb is sufficient in an agent society, where services and their abilities are relatively fixed and well-known, which is reasonable for the relatively closed projects described in section 4. As the number of available services will grow (MathWeb has for instance been adopted by other projects building on Doris), this design will become too inflexible. Therefore the logical next step will be to adopt a more general truly agent-based approach. We have started to extend MathWeb so that it uses the Kqml interlingua (*Knowledge Query and Manipulation Language* [FF94]) as the agent interaction language and the OpenMath [Cap98] standard as a content language.

This move will result in a "plug-and-play" architecture for theorem proving and (in the future) for doing mathematics and program verification on the web.

References

[ABI+96] Peter B. Andrews, M. Bishop, et al. TPS: A theorem proving system for classical type theory. *Journal of Automated Reasoning*, 16(3):321–353, 1996.

[BCF+97] C. Benzmüller, L. Cheikhrouhou, et al. Ωmega: Towards a mathematical assistant. In William McCune, editor, *Proc. CADE'97*, number1249 in LNAI, pages252–255, 1997. Springer Verlag.

[BK98] Christoph BenzmüllerandMichael Kohlhase. LEO, a higher order theorem prover. In Claude KirchnerandHélène Kirchner, editors, *Proc. CADE'98*, number1421 in LNAI, pages139–144, 1998. Springer Verlag.

[Cap98] The open math standard. 1998. at `http://www.openmath.org`

[FF94] T. FininandR. Fritzson. KQML — a language and protocol for knowledge and information exchange. In *Proceedings of the 13th Intl. Distributed Artificial Intelligence Workshop*, pages127–136, Seattle, WA, USA, 1994.

[FHJ+99] Andreas Franke, Stephan M. Hess, Christoph G. Jung, Michael Kohlhase, andVolker Sorge. Agent-oriented integration of distributed mathematical services. submitted, 1999.

[HF96] Xiaorong HuangandArmin Fiedler. Presenting machine-found proofs. In M.A. McRobbieandJ.K. Slaney, editors, *Proc. CADE'96*, number1104 in LNAI, pages221–225, 1996. Springer Verlag.

[KKS98] M. Kerber, M. Kohlhase, andV. Sorge. Integrating computer algebra into proof planning. *Journal of Automated Reasoning*, 21(3):327–355, 1998.

[Mei99] Andreas Meier. Translation of automatically generated proofs at assertion level. Technical reportforthcoming, Universität des Saarlandes, 1999.

[SHB+98] Jörg Siekmann, Stephan Hess, et al. A distributed graphical user interface for the interactive proof system OMEGA. In Roland C. Backhouse, editor, *UITP'98*, number98-08 in CS Reports, pages130–138, Dept. of Mathematics and Computing Science, Eindhoven Technical University, 1998.

[SHS98] M. Kohlhase S. Hess, Ch. JungandV. Sorge. An implementation of distributed mathematical services. In *6th CALCULEMUS and TYPES Workshop*, 1998. `http://www.win.tue.nl/math/dw/pp/calc/proceedings.html`.

[Sie96] Jon Siegel. *Corba: Fundamentals and Programming*. Wiley, 1996.

[SS97] *Journal of Automated Reasoning*, special issue on CASC-14, 18(2), 1997.

[8] Either by using MathWeb directly, or by cooperating with the authors.