Available online at www.sciencedirect.com
ScienceDirect
Journal of Applied Logic 4 (2006) 414–427
JOURNAL OF APPLIED LOGIC
www.elsevier.com/locate/jal

MizarMode—an integrated proof assistance tool for the Mizar way of formalizing mathematics

Josef Urban

Department of Theoretical Computer Science, Charles University, Malostranske nam. 25, Praha, Czech Republic

Available online 10 November 2005

Abstract

The Emacs authoring environment for Mizar (MizarMode) is today the authoring tool of choice for many (probably the majority of) Mizar authors. This article describes the MizarMode and focuses on the proof assistance functions and tools available in it.

We start with the explanation of the design principles behind the Mizar system, and show how these design principles—mainly the concentration on simple and intuitive human-oriented proofs—have helped Mizar in developing and maintaining a very large body of formalized mathematics.

Mizar is a non-programmable and non-tactical verifier: the proofs are developed in the traditional "write—compile—correct" software programming loop. While this method is in the beginning more laborious than the methods employed in tactical and programmable proof assistants, it makes the "proof code" in the long-run more readable, maintainable and reusable. This seems to be a crucial factor for a long-term and large-scale formalization effort.

MizarMode has been designed with the aim to facilitate this kind of proof development by a number of "code-generating", "code-browsing" and "code-searching" methods, and tools programmed or integrated within it. These methods and tools now include, e.g., the automated generation of proof skeletons, semantic browsing of the articles and abstracts, structured viewing, proof advice using trained machine learning tools like the Mizar Proof Advisor, deductive tools like MoMM, etc. We give an overview of these proof-assistance tools and their integration in the MizarMode, and also discuss some emerging and future extensions such as integration of external theorem proving assistance.

E-mail address: urban@kti.ms.mff.cuni.cz (J. Urban).

1570-8683/$ – see front matter
doi:10.1016/j.jal.2005.10.004

Keywords: Mizar; MizarMode; Proof assistance

1. Introduction

The Mizar [7,12,13] project is today probably the largest effort towards the computer assisted formalization of mathematics. The Mizar Mathematical Library (MML) contains at the moment of writing this article almost 900 formalized articles from many mathematical fields. With the current rate of growth it should reach 1000 articles by the end of 2006.

The Mizar system differs quite significantly from many other proof assistants. It has grown for more than 30 years in relative isolation, and there are many differences to other systems both in the philosophy and in the implementation. Due to these differences, Mizar is probably not considered to be a prototype of a "proof assistant" in the most common usage of this term. It is not interactive, tactical or programmable, it is "just" a verifier, and a set of utilities for a very well designed language and proof formalism. In the next section we explain the philosophy and implementation of Mizar a bit, and conjecture that the strong emphasis on the quality of the language and the readability of the proof presentation are in the long run very important for the development of a large body of formalized mathematics.

In interactive proof assistants a lot of "authoring assistance" (e.g., proof advice) can be directly implemented inside their kernels. The compiler-like operation of the Mizar verifier is the main reason why all such functions for Mizar have to be taken up by various external tools and authoring environments. In the remaining sections we give an overview of the Emacs authoring environment for Mizar (MizarMode), and the various proof-assistance tools that have been programmed or integrated in it exactly for this purpose.

2. The Mizar way of formalizing mathematics and its consequences for proof assistance

2.1. The emphasis on readability

Mizar focuses on the development of formal, yet sufficiently high-level and human-readable proofs. There are a growing number of natural language constructs in the Mizar language. Mizar is based on the Jaskowski [4,8] natural deduction style, which is quite intuitive and similar to standard textbook proofs, and usually clear even to people with no theoretical knowledge of natural deduction. The Mizar view is that the proof language should first concentrate on clarity, human readability and closeness to standard mathematical proofs, and generally be very careful about adding an excessive number of "tactics" or even allowing users to program their own extensions that make the resulting proofs hard to understand. Today, several other proof assistants that had originally started with the opposite view, i.e., with programmable tactics and other programmable extensions, have