Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

328

R. Bloomfield L. Marshall
R. Jones (Eds.)

VDM '88
VDM – The Way Ahead

2nd VDM-Europe Symposium
Dublin, Ireland, September 11–16, 1988
Proceedings

Springer-Verlag
Berlin Heidelberg New York London Paris Tokyo

Editors

Robin E. Bloomfield
Adelard, 28 Rhondda Grove
London E3 5AP, UK

Lynn S. Marshall
Université Catholique de Louvain
Place Sainte-Barbe 2, 1348 Louvain-La-Neuve, Belgium

Roger B. Jones
ICL (UK), Eskdale Road, Winnersh
Wokingham, Berkshire RG11 5TT, UK

CR Subject Classification (1987): F.3.1, D.2.1, D.2.4

ISBN 3-540-50214-9 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-50214-9 Springer-Verlag New York Berlin Heidelberg

Printing and binding: Druckhaus Beltz, Hemsbach/Bergstr.
2145/3140-543210

FOREWORD

VDM, the Vienna Development Method, is a formal method for software engineering. It is being applied to an increasing number of projects by companies throughout Europe and there is an active international research programme supporting this process. "*VDM: the way ahead*" is the second of a series of symposia sponsored by the *Commission of the European Communities* (CEC) and organised by VDM-Europe.

For those unfamiliar with "formal methods" it is perhaps useful to define what is meant by the term in this context. "Formal method" applies to a mathematically formal software specification and production method that has three components. It must contain a system of mathematically based notations that address the specification, design and development phases of software production. It must also contain a well-founded inference system in which formal proofs of correctness and other properties can be formulated and constructed. The third component is a methodological framework within which software may be developed from the specification in a formally verifiable manner.

Formal methods aim to increase the quality of software in two related ways: by improving the specification, and by making verification during the software production process more effective and easier to audit.

In VDM, sequential systems are modelled by a collection of operations defined in terms of pre- and post-conditions on an underlying state. Development is by a process of data reification and operation decomposition. The method includes a logical system containing the predicate calculus for the formulation of proofs, and each design step may require the designer to establish a number of proof obligations which show the correctness of the development.

Those readers requiring an introduction to VDM and an outline of its historical development should consult the proceedings of the first VDM symposium (VDM'87 LNCS 252).

The Technical Advisory Group, VDM-Europe, established by the Commission of the European Communities (CEC) meets regularly to discuss issues pertinent to VDM. These discussions focus around five main areas of interest:

-education and technology transfer

-experience and use of VDM

-tools and support environments

-method development and foundational work

-the standardisation of VDM

In addition VDM-Europe maintains a register of all VDM related activities in Europe. VDM-Europe is an open European group and anyone who is interested may attend and participate.

These five areas of interest are well represented in this symposium. The commitment to education and training and technology transfer is exemplified by the existence of the symposium and in particular by the two day tutorial to be given by D. Bjoerner and C.B. Jones (perhaps the two names most associated with VDM) as well as by the paper by Naftalin on "Correctness for Beginners".

The applications of VDM reported in this symposium are diverse, and they range from hardware test case selection through to the specification of Chinese characters as well as the more traditional use of VDM in compiler specification and development. There is also an increasing use of VDM in the definition and analysis of standards illustrated by the papers on ODA, GKS, Modula 2 and the deliberations of the BSI VDM-SL standardisation panel. In addition to the problem specific applications there are also generic issues such as the development towards a particular implementation language, Ada, as well as how to handle concurrency.

The availability of tools to support the use of VDM is essential for its successful industrialisation. The requirements for tools range from clerical support in editing and typesetting documents through type checkers and parsers to fully integrated support environments and theorem provers. The exhibition that runs in parallel with the symposium enables attendees to sample the latest developments. In these proceedings, tools are discussed in the papers on support for VDM specifications, the execution of programming language definitions, theorem proving assistants and project support environments.

The development of VDM and the definition or re-examination of its foundations are the subject of much research. Some of this, such as the structuring work for the BSI VDM-SL standard (Bear) involves extensions to VDM to assist in its application to large scale projects. The symposium also includes reports of research on logics and domain theory relevant to the theoretical foundations of VDM (Blikle; Tarlecki et al; Haxthausen).

The application and development of VDM must of course take into account work on other related formal methods. A comparison with FOREST is provided by the paper from Goldsack, while Abrial gives an invited talk on his development of B. Other work, such as the paper on the RAISE project (Nielsen, Havelund, Wagner, George), seeks to extend VDM to handle, among other things, concurrency. In addition there is the work on Metasoft, represented by the paper from Borzyszkowski et al.

Standardisation is the last of the five themes of the symposium and one of the most important. Widespread investment in VDM and associated tools requires a standard for the concrete and abstract syntax, a definition of the semantics and the associated inference system. It is imperative that the standard not only be technically adequate but also reflect the consensus of interested parties. Although some researchers may regard standardisation as merely a bureaucratic activity the BSI/VDM has had to grapple with considerable technical issues in its deliberations to date. A progress report is given by Andrews.

The title of the symposium is "*VDM: the way ahead*". We hope that the proceedings do indeed show the way ahead both for those concerned with the application of VDM and those concerned with its development.

Brussels, July 1988

R. Bloomfield
L. Marshall
R. Jones

TABLE OF CONTENTS

DAY 1

Computing is a Physical Science (Abstract)
Donald I. Good; Computational Logic, Inc., Texas, USA 1

Structuring for the VDM Specification Language
Stephen Bear; CEGB, London, UK 2

Correctness for Beginners
Maurice Naftalin; Univ. of Stirling, UK 26

Understanding an Informal Description: Office Documents Architecture, an ISO Standard
Andrzej Borzyszkowski, Stefan Sokolowski; Polish Ac. of Science, Gdańsk, Poland 48

Towards a Formal Definition of GKS and Other Graphics Standards
Clive Ruggles; Univ. of Leicester, UK 64

Report from the BSI Panel for the Standardisation of VDM (IST/5/50)
Derek Andrews; Univ. of Leicester, UK 74

Formal Methods in Standards - A Report from the BCS Working Group
Clive Ruggles; Univ. of Leicester, UK 79

DAY 2

The B Tool (Abstract)
J.R. Abrial; Paris, France 86

Specification of an Operating System Kernel FOREST and VDM Compared
S.J. Goldsack; Imperial College, London, UK 88

APPLICATIONS AND TOOLS

Compiler Prototyping with VDM and Standard ML
R.D. Arthan; ICL, Winnersh, UK 101

VDM Development with Ada as the Target Language
David O'Neill; Generics, Dublin, Ireland 116

NUSL: An Executable Specification Language Based on Data Abstraction
XinJie Jiang, YongSen Xu; Nanjing Normal Univ.; Shantou Univ., Guangdong, China 124

A Support System for Formal Reasoning: Requirements and Status
C.B. Jones, P.A. Lindsay; Manchester Univ., UK 139

The Use of VDM Within the Alvey Flagship Project
Graham Boddy; ICL, Manchester, UK 153

The Formal Definition of Modula-2 and Its Associated Interpreter
D.J. Andrews, A. Garg, S.P.A. Lau, J.R. Pitchers; Univ. of Leicester, UK 167

Test Case Selection Using VDM
G.T. Scullard; ICL, Manchester, UK 178

The VIP VDM Specification Language
Kees Middelburg; PTT, Leidschendam, The Netherlands 187

SAMPLE - A Functional Language
M. Jäger, M. Gloger, S. Kaes; Techn. Hochschule, Darmstadt, Germany 202

FOUNDATIONS AND THEORY

A Three-valued Logic for Software Specification and Validation
Beata Konikowska, Andrzej Tarlecki, Andrzej Blikle; Polish Ac. of Science, Warsaw, Poland 218

Three-valued Predicates for Software Specification and Validation
Andrzej Blikle; Polish Ac. of Science, Warsaw, Poland 243

A Set-Theoretic Model for a Typed Polymorphic Lambda Calculus. A Contribution to MetaSoft
Andrzej Borzyszkowski, Ryszard Kubiak, Stefan Sokolowski; Polish Ac. of Science, Gdańsk, Poland 267

Mutually Recursive Algebraic Domain Equations
Anna Elisabeth Haxthausen; Technical Univ. of Denmark, Lyngby, Denmark 299

Proof Rules for VDM Statements
Robert Milne; STC, Harlow, UK 318

Muffin: A User Interface Design Experiment for a Theorem Proving Assistant
Cliff B. Jones, Richard Moore; Manchester Univ., UK 337

DAY 3

The RAISE Language, Method and Tools
Mogens Nielsen, Klaus Havelund, Kim Ritter Wagner, Chris George; DDC, Lyngby, Denmark; STC, London, UK 376

Correctness Proofs for Meta IV Written Code Generator Specifications Using Term Rewriting
Bettina and Karl-Heinz Buth; Christ.-Albrechts-Univ. zu Kiel, Germany 406

Using VDM with Rely and Guarantee-Conditions. Experiences from a Real Project
J.C.P. Woodcock, B. Dickinson; Oxford Univ.; GEC, Coventry, UK 434

Software Support for the Refinement of VDM Specifications
P. Kilpatrick, P. McParland; The Queen's Univ. of Belfast, UK 459

The Use of VDM in the Specification of Chinese Characters
Ghee S. Teo, Mícheál Mac an Airchinnigh; Trinity College, Dublin, Ireland 476