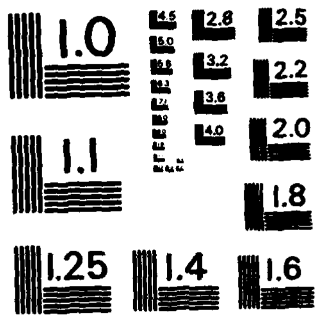
AD-A130 693 NATO/AGARD SYMPOSIUM ON SOFTWARE FOR AVIONICS(U) OFFICE 1/1
OF NAVAL RESEARCH LONDON (ENGLAND) D WEISS 26 APR 83
ONR-C-7-83

UNCLASSIFIED F/G 9/2 NL

END
DATE
FILMED
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

ADA130693

(12)

ONR LONDON CONFERENCE REPORT

C-7-83

OFFICE OF NAVAL RESEARCH

BRANCH
OFFICE
LONDON
ENGLAND

NATO/AGARD SYMPOSIUM ON SOFTWARE FOR AVIONICS

DAVID WEISS
Naval Research Laboratory, Washington, D.C.

26 April 1983

UNITED STATES OF AMERICA

83 07 26 197

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER C-7-83	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) NATO/AGARD Symposium on Software for Avionics		5. TYPE OF REPORT & PERIOD COVERED Conference
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) David Weiss Naval Research Laboratory, Washington, D.C.		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS US Office of Naval Research, Branch Office London Box 39 FPO NY 09510		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE 26 April 1983
		13. NUMBER OF PAGES 7
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release: Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Avionics
Software technology

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
The symposium was concerned with requirements, design, development, verification, and validation of avionics software. A few papers concerned research results and future technology, but most dealt with the pratical aspects of software development.

p3

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

NATO/AGARD SYMPOSIUM ON SOFTWARE FOR AVIONICS

Software for avionics was the focus of the fall 1982 AGARD Avionics Panel symposium held in Kijkduin, the Netherlands. The subject was chosen because of the problems associated with the increasing cost and complexity of software in general and avionics software in particular.

About 250 attendees participated in sessions concerning requirements, design, development, verification, and validation of avionics software. The technical program committee was chaired by Max Jacobsen and was made up of members from the US, the UK, Italy, France, and Germany.

Although a few papers concerned research results and future technology, most dealt with the practical aspects of avionics software development. As a result, the symposium had the character of a practitioners' workshop. Some sessions and related conclusions are described in this report. The appendix contains a numerical list of program speakers and the titles of their presentations.

Software Technology Tutorial

The first session was a tutorial on the current state of avionics software technology. The principal point of the first paper, presented by Dr. Willis Ware (Rand Corp.), was that software problems are developing faster than solutions. To support this contention, Ware noted that military users cannot state requirements precisely enough for software designers; the information management problem in the cockpit is getting more difficult. For example, more rapid handling of more information is needed, and aircraft will be spending longer periods of time in the air and must become more reliable.

Ware observed that techniques for alleviating software problems are just now moving from the laboratories into the development community. He stated that the most promising techniques treat software components as black boxes.

Other presentations in the session dealt with avionics software development methodologies and automated aids to support them. Discussions included the methodologies used for the F-16, the Tornado, and the MINERVE; extensions to the MASCOT methodology; and the AIGLE system. MASCOT and MINERVE are standardized methodologies used in the UK and France, respectively. MASCOT includes a set of integrated support tools; AIGLE is a set of integrated support tools now under development for MINERVE.

Requirements Analysis

The papers in Session 2 seemed to indicate that it was desirable for a requirements specification to be complete, formal, and unambiguous, and to answer the questions why and what rather than how. Paper 8 used these criteria to provide a quantitative evaluation of a manually created requirements document. Presentations 9, 10, and 11 dealt with automated aids for producing requirements with such characteristics. Interesting variations on the theme were also included in the session. Paper 6 discussed pitfalls in the requirements definition process.

One could conclude from Session 2 that good requirements specifications can be created manually, but that automated aids would make the task much easier. People using such aids are struggling to fit them into the usual software development methodologies but consider the effort worthwhile.

Software Design and Development

Much of the session on design and development was concerned with the use of high order languages for avionics. The avionics community in general has lagged in the use of such languages and is now attempting to remedy the situation. The following questions were discussed during the session:

1. Should we wait for the advent of Ada production compilers before switching to a high order language, or should we adopt some other language for which we know efficient compilers can be developed?

2. How do we solve the problem of using high order languages for microprocessors when a new generation of microprocessors comes along every 2 to 4 years and requires a new compiler?

3. Can we afford the overhead of using abstractions built into standardized methodologies and their tools, including high level languages and systems such as MASCOT?

The status of the Ada effort (paper 13) was presented by a representative of the Ada Joint Program Office.

Because of doubts about the efficiency of Ada's real-time capabilities, the possible training problems, and the costs associated with using the system and its environment, several organizations have decided to use existing languages. For example, the French language LTR, which has been a standard since 1974, is now being upgraded to handle parallel tasks (paper 14).

To deal with microprocessor problems, Messerschmitt-Bölkow-Blohm GmbH (MBB) of Germany has developed a system for producing Pascal compilers for microprocessors (paper 16). A common front end and intermediate language is used by all the compilers, requiring only the development of a new code generator to produce a new compiler for a new microprocessor.

Advantages and disadvantages of using the MASCOT system were discussed by a representative of Ferranti (paper 18). The issue that drew the most attention was an estimate of a 35 to 40% increase in overhead using MASCOT.

Verification and Validation

Many of the papers in the verification and validation session were descriptions of detailed test methodologies and associated automated tools. Issues discussed included how to manage the testing process, who should do the testing, and what are good measures of test coverage. There was agreement that a strictly disciplined approach to testing is needed. Many organizations are using similar approaches for achieving such discipline, including developing support tools to generate tests and to measure the testing process (e.g., to report on test coverage for each program being checked).

Other interesting papers on verification were included in the session. One discussed using dissimilar software to achieve high integrity systems (paper 34). In other words, two or more independently developed software components that have the same specification are used to perform a task. If all produce the same result, the probability is high that the result is correct. If one or more disagree, a vote may be taken to determine the correct result. As the number of dissimilar systems used is increased, the number of errors that each may contain--while still providing the correct result--also increases. Consequently, no individual component requires extensive verification.

Paper 30 described a state-of-the-art, semiautomated, program verification system. The system is being used experimentally to verify microprograms used to control computers. Verification requires as input a formal description of the machine emulated by the microcode, a formal description of the semantics of the micromachine on which the microcode executes, and the microcode itself. In addition, a "rationale" for the microcode is submitted by the programmer.

A symbolic execution scheme is used, and about 1,000 microinstructions per hour can be verified. The developers believe the rate can be increased to approximately 2,000 microinstructions per minute.

Conclusions

The avionics community is struggling to automate its work. Different methodologies are being used to impose discipline on avionics software development, and the methodology developers are now attempting to provide as much automated support as possible. Support tools such as requirement specification systems, high level language compilers, data-base systems, test languages, and other test tools are all being slowly integrated into the software development process. The situation can be contrasted with that 10 years ago, when software developers were thinking about what methodology to use.

Perhaps most ignored during the symposium were techniques for design--as opposed to techniques for managing development. The criteria used to organize and document a design are the basic problems in producing systems that are maintainable over long periods of time.

APPENDIX:

SESSIONS AND SPEAKERS

Session 1: Software (S/W) Technology (Tutorial)--Chairman, Dr. A.A. Callaway (UK)

1. Avionic Software: Where Are We
   Dr. W. Ware, Rand Corp., Santa Monica, CA
2. Avionics Software Design
   Dr. D.E. Sundstrom, General Dynamics, Fort Worth, TX
3. S/W Development: Design and Reality
   Dr. H. Groote, Dr. Schwegler, MBB, München, Germany
4. MASCOT Developments to Improve Software Structure and Integrity
   Dr. H.R. Simpson, British Aerospace PLC, Dynamics Group, Stevenage, UK
5. Vers un Veritable Atelier de Logiciel Avionique
   J. Perin, Electronique Marcel Dassault, St. Cloud, France

Session 2: Software and System Requirement Analysis--Chairman, Dr. Ing. L. Crovella (Italy)

6. Requirements Decomposition and Other Myths
   B. Malcolm, T. Swann, B. Hauxwell, D. Jordan, Marconi Avionic Systems Ltd., UK
7. Practical Considerations in the Introduction of Requirement Analysis Techniques
   C.P. Price, D.Y. Forsyth, British Aerospace PLC, Warton, Preston, UK
8. Evaluation of the A-7 Software Requirements Document by Analysis of Changes: Three Years of Data
   D.M. Weiss, L. Chmura, U.S. Naval Research Laboratory, Washington, D.C.
9. D.L.A.O.: Un Systeme d'Aide a la Definition de Logiciel Avioniques
   S. Chenut/Martin, Ing. F. Doladille, Electronique Marcel Dassault, St. Cloud, France
10. The Mentor Approach to Systems Development
   D. Jordan, B. Hauxwell, Marconi Avionic Systems Ltd., UK
11. The Computer Aided System Specification Easy
   N. Christensen, L. Hirschmann, Mat. Beratungs- und Programmierdienst, Dortmund, Germany

Session 3: Software Design and Development Process--Chairman, B. Mirailles (France)

12. The Impacts of Standardization on Avionic Software
   Dr. J.D. Engelland, G.R. England, General Dynamics Div., Fort Worth, TX

13. Ada Status and Outlook
    L/Cdr. J.F. Kramer, Ada Joint Program Office, Arlington, VA
14. Standardisation du LTR Pour Calculateurs Embarques--le Present et le Futur
    ICA de Montcheuil, Direction Technique des Engins, Paris-Armees, France
15. Use of High Order Language for OFP Programming With Emphasis on the Use of Ada
    Dr. R. Pendleton, Dr. J.J. Zenor, Naval Weapons Center, China Lake, CA
16. An Approach to a Portable Pascal Language for Different Onboard Computer Systems
    Dr. W. Wiemer, Mr. Reitz, MBB, München, Germany
17. Use of High Order Languages on Micro-Processors
    R.M. Boardman, Marconi Avionic Systems Ltd., UK
18. Software Design and Development Using MASCOT
    R. Dibble, G. Cram, D. Milledge, Ferranti Computer Systems Ltd., UK
19. Safety Critical Fast-Real-Time Systems
    Dr. B. Gusmann, O.F. Nielsen, R. Hansen, MBB, München, Germany
20. Usability of Military Standards for the Maintenance of Embedded Computer Software
    Prof. N. Schneidewind, Naval Postgraduate School, Monterey, CA
21. Software Configuration Management at Work
    Jan T. Pedersen, A/S Kongsberg Våpenfabrikk, Norway
22. Configuration Management and the Ada Programming Support Environment
    Chf. Eng. K. Pulford, Marconi Avionic Systems Ltd., UK
23. Practical Software Fault Tolerance for Real-Time Systems
    Dr. John Knight, Dr. T. Anderson, Department of Applied Mathematics and Computer Science, Univ. of Virginia
24. Electronic Warfare Software
    R. Shaw (AFWAL/AAWP), Wright-Patterson AFB, OH

Session 4: Software Verification and Validation--Chairman, R.O. Mitchell (US)

25. An Eight Point Testing Strategy for Real-Time Software
    R. Wilson, N. Higson, Marconi Avionic Systems Ltd., UK
26. Tornado Flight Control Software Validation: Methodology and Tools
    Dr. Ing. R. Pelissero, AERITALIA, Gruppo Equipaggiamenti, Torino, Italy

27. Applications of Network Logic Modeling and Analysis to System Validation and Verification
Mr. G. Sundberg, Tracor, Inc., Warminister, PA
28. Software Test Language and Related Tools
Eng. P. Taillibert, G. Lamarche, Electronique Marcel Dassault, St. Cloud, France
29. Software Verification of a Civil Avionic AHP System
Dr. M. Kleinschmidt, Dr. N. Sandner, Litton Technische Werke der Hellige GmbH (LITEF), Freiburg, Germany
30. Progress in Verification of Microprograms
Dr. S.D. Crocker, The Aerospace Corporation, Los Angeles, CA
31. Validation of Software for Missile to Aircraft Integration
J.R. McManis, Naval Weapons Center, China Lake, CA
32. Implementing High Quality Software
E. Dowling, Ferranti Computer Systems Ltd., Gwent, UK
33. La Qualite Des Logiciels Avioniques--Specification et Evaluation
Prof. M. Galinier, G. Germain, IGL, Paris, France
34. Dissimilar Software in High Integrity Applications
Dr. D.J. Martin, Marconi, UK
35. The Cost of Software Fault Tolerance
G.E. Migneault, AESB/FED, US

Session 5: Software Life Cycle Considerations--Chairman, Dr. H. Hessel (Germany)

36. Management of Large Real-Time Military Avionics Software Programs
Dr. P.J. Carrington, R.M. Gisbey, K.F.J. Manning, Marconi Avionics, Rochester, Kent, UK
37. F/A-18 Avionics Software--A Case Study
T.V. McTigue, McDonnell Aircraft, St. Louis, MO
38. A Life Cycle Model for Avionic Systems
Wis. Dir. Dipl. Ing. Schaff, Bundesakademie für Wehrverwaltung, und Wehrtechnik, Mannheim, Germany
39. Avionics Software Support Cost Model
D.V. Ferens (AFWAL), Wright-Patterson AFB, OH (presented by R. Shaw)
40. A Software-Cost Data Base for Aerospace Software Development
G.J. Dekker, National Aerospace Research Laboratory, Amsterdam, Netherlands

41. The Military User View of Software Support Throughout the In-service Life of Avionics Systems
    Wg. Cdr. S. Barker, RAF; Sqn. Ldr. B. Hambling, RAF; London, UK
42. Design of a Software Maintenance Facility for the RAF
    J. Whalley, T.H. Scott-Wilson, British Aerospace PLC, Stockport, Cheshire, UK
43. A Software Engineering Environment for Weapon System Software
    H.G. Stuebing, Software and Computer Directorate, US Naval Air Development Center, Warminster, PA
44. On Aircraft Software for First Line Maintenance
    Dr. H. Klenk, MBB, München, Germany

ME

- 8