TUTORIALS  
MONDAY, MARCH 26, 1984  
9:00 AM TO 5:00 PM

TUTORIAL 1—
Models and Metrics for Software Management & Engineering—Victor Basili

TUTORIAL 2—
A Builder’s Guide to Software Engineering Environments—William Riddle

TUTORIAL 3—
Structured Testing—Thomas J. McCabe

TUTORIAL 4—
Advances in Software Design—Peter Freeman & Anthony Wasserman

15 YEARS OF SOFTWARE ENGINEERING
The term “Software Engineering” was first used 15 years ago. It was invented when it had become evident that the cost effective development of reliable software would ultimately depend on the successful creation of a new engineering discipline. In the last 15 years new methods have been invented and evaluated, and our understanding of the problems to be solved in the future has deepened. The Seventh International Conference on Software Engineering will focus on the results of these 15 years, on the state of this new engineering discipline, and what to expect of it in the future.

TOOLS FAIR
Concurrent with the Conference, a Tools Fair will be held. Software tools, both experimental and commercial, will be demonstrated. Authors of papers describing practice and experience with particular tools will also be given an opportunity to demonstrate. For information concerning the Tools Fair contact Gerald Leach-Lewis, IEEE Computer Society, PO. Box 639, Silver Spring, Maryland, U.S.A. Telephone: (301) 589-8142—Telex 7108250437 IEEECOMPSO.
Models and Metrics for Software Management and Engineering
Instructor: Victor Basili

Audience: intended for software managers and software engineers who are looking for quantitative assistance in managing, controlling, estimating, evaluating, and contracting software development and maintenance.

Course Description: Presents a quantitative approach to software management and software engineering. Focuses on attributes that can be managed quantitatively covering both product-oriented and process-oriented attributes. Utilizes a large set of models and metrics.

Course Outline:
- Introduction: Quantitative Software Management and Metrics
  - Resources Models: classes of models, empirical models of programming estimation and measurement
  - Model validation studies: an approach to developing an empirical resource model specifically for the organization in which it will be used
  - Theoretical dynamic modeling using the Rayleigh curve

Changes and Errors: the use of changes and errors in the characterization, estimation, and evaluation of software development

Product Measures: size, control and data structure metrics, cyclomatic complexity, software science, data bindings, span metrics across time

Data Collection: establishing the environment, collecting data, data validation analysis and result reporting

Victor R. Basili is a professor and chairman of the computer science department at the University of Maryland. A consultant to several organizations and government agencies, he has been active in the design and development of several software projects and is currently involved in the measurement and evaluation of software development at NASA's Goddard Space Flight Center. He received his Ph.D. in computer science from the University of Texas at Austin in 1970.

A Builder's Guide to Software Engineering Environments
Instructor: William Riddle

Course Outline:
- Introduction: A Graduated Introduction to Environments
  - The Use of Software Engineering Environment Models
  - Structured Modularity: Using Complexity to Modularize the Design

William Riddle is an independent consultant specializing in research, education and evaluation projects in the area of software engineering environments. Previously, he directed software design and engineering work at Cray Laboratories, the research and development subsidiary of Cray Research. Prior to that, he was a professor of computer science at the Universities of Michigan and Colorado. His work has emphasized the description and analysis of software designs and the delivery of description and analysis techniques to practitioners through software engineering environments. He is a former ACM National Lecturer and is currently chair of the ACM SIGSOFT organization.

Structured Testing
Instructor: Thomas J. McCabe

Course Description: In most software systems, the central problems of reliability and maintainability are treated by past experience indicating that we often spend 50% of our time in testing and more than 70% of our dollars in maintenance. Most of these problems, no matter how, are the result of testing and modularization practices that are inadequate. This seminar presents techniques and methods that can be used to improve reliability and maintainability and which should be employed in any size software project.

Outline:
- The Role and Importance of Testing
- Levels of Testing: Against specifications, functions, performance, software modules
- The Central Problem: Software Complexity: reliability, development, maintainability
- Measurement and Quantification of Software Complexity
- Graph Theory, Alternative Methods, Language constructs
- Structured Modularity: Using complexity to modularize the design
- Structured Testing: the technique, testing specifications, tests, the code
- Effectiveness and Reliability of Testing Environments: Top-down and bottom-up testing integration

Thomas J. McCabe is President of McCabe & Associates, Inc. He is widely known as a consultant and authority in the area of software development testing and quality control. He is best known for his research and publications on software complexity and the complexity measure that bears his name. He has developed and published the Structured Testing Methodology which is being adopted extensively. He has developed and taught advanced state-of-the-art courses in Software Quality Assurance, Structured Testing, Software Specification and Design, and Software Engineering which have been used at universities from the Univ. of Connecticut, is listed in "Who's Who in the East," and has been elected to the "International Dictionary of Biography."

Advances in Software Design
Instructor: Peter Freeman and Anthony Wasserman

Course Description: Several design methods and their underlying concepts are now widely known and used by experienced software designers. While these methods and concepts are sufficiently for many instances of routine design, new techniques are being developed to handle more complex design problems. This tutorial briefly reviews design concepts and methods, and then surveys several directions in design.

Outline:
- Design by Example: The Role of Software Design
  - Basic Concepts
  - Usage
  - Methods
- Design by Contract: A Design Methodology
  - Design
  - Development
- Design by Inspection: A Design Methodology
  - Design
  - Development
- Design by Construction: A Design Methodology
  - Design
  - Development

Peter Freeman, an associate professor of information and computer science at the University of California, Irvine, has been involved in the analysis, design, and construction of advanced computer applications and the training of software engineers since 1961. He has concentrated in software design techniques and their application to the software engineering process. He has published numerous technical papers, is the author of SOFTWARE SYSTEMS PRINCIPLES (SRA, 1976), and has jointly edited two books. He received his Ph.D. in computer science from Carnegie-Mellon University in 1970.

Anthony J. Wasserman is an associate professor of medical information science at the University of California, San Francisco. He is a lecturer in the Computer Science Div. at the University of California, Berkeley. His research is focused on the use of Software Engineering (USE) project. He is also the author of 10 technical papers and editor of or co-author of six books, including "Compile Time Software Engineering Design Techniques" (Peter Freeman and "Software Engineering Design Techniques""). Wasserman received his Ph.D. in computer science from the University of Wisconsin-Madison.
TUESDAY, MARCH 27, 1984

9:00 A.M.  OPENING SESSION—KISSIMMEE HALL

9:30 A.M.-10:30 A.M.  KEYNOTE ADDRESS—Professor Herbert Simon, Nobel Laureat, "Does Software Need to be Artificially Intelligent?"

10:30 A.M.-11:00 A.M.  COFFEE BREAK—KISSIMMEE HALL

11:00 A.M.-12:30 P.M.  PANEL
SESSION CHAIR: Col. John Marcinlak, Rome Air Development Center, USA
"The National Software Engineering Projects" panelists include:
K. Fuchl, Director Research Center for New Generation Computer Technology, JAPAN
H. Hünke, Director of ESPRIT, Commission of the European Communities
E. Lieblein, Director of large systems and software, Office of the Undersecretary of Defense for Research and Adv. Technology, USA
J. C. Rault, Head of Software Eng. Dept., Agence de l’Informatique, FRANCE

12:30 P.M.-2:00 P.M.  LUNCH

2:00 P.M.-3:30 P.M.  PARALLEL SESSIONS

KISSIMMEE HALL
SESSION 2A—SOFTWARE DEVELOPMENT PARADIGMS, presentation

Presenters include:
R. Balzer, Information Sciences Institute, USA
F. L. Bauer, Technical Univ. of Munich, WEST GERMANY
T. E. Cheatham, Harvard Univ., USA
Harlan Mills, IBM, USA

4:00 P.M.-5:30 P.M.

KISSIMMEE HALL
SESSION 3A—SOFTWARE DEVELOPMENT PARADIGMS, Debate

Debaters include:
R. Balzer, Information Sciences Institute, USA
F. L. Bauer, Technical Univ. of Munich, WEST GERMANY
T. E. Cheatham, Harvard Univ., USA
Harlan Mills, IBM, USA

ORLANDO HALL
SESSION 2B—ERROR PROCESSING
Session Chair: TBA
Specification for Distributed Error Recovery Protocols, A. CIUFFOLETTI, SELENIA S.P.A., ITALY
Exception Handling: Formal Specifications and Systematic Program Construction, B. BIEBOW, M.C. GAUDEL, LAB. DE NOZAY, M. FRANCE, M. BIDOIT, LAB. DE RECHERCHE EN INFORM
Diagnostics System for Distributed Software: A Relational Database Approach, P. HSIA, K. CHONG, UNIV. TEXAS-ARLINGTON, USA

ORLANDO HALL
SESSION 3B—DISTRIBUTED SYSTEMS
Session Chair: S. Budkowski, Warsaw Polytechnic, POLAND
Multifaceted Distributed Systems Specification Using Processes and Event Sync., G.C. ROMAN, M. DAY, WASHINGTON UNIV. USA
EDMAS: A Locally Distributed Mail System, G. ALMES, A. BLACK, C. BUNJE, UNIV. WASHINGTON, USA
Resource Controller Tasks in ADA: Their Structure and Semantics, K. RAMAMRITHAM, UNIV. MASS., USA
WEDNESDAY, MARCH 28, 1984

8:30 A.M.-10:00 A.M.
KISSIMMEE HALL
SESSION 4A—MANAGEMENT ISSUES
Session Chair: J. H. Frame, ITT, USA
A Spiral-Out Approach to Software Engineering
USA
An Approach to Cooperative Software Development
by Application Engineers and Program Engineers,
K. Uemura, M. Ohori, Kozo Keikaku Eng., Inc., Japan
Fifteen Years of Psychology in Software Engineering,
B. Curtis, ITT Corp., USA

10:00 A.M.-10:30 A.M. COFFEE BREAK

10:30 A.M.-12:00 NOON
KISSIMMEE HALL
SESSION 5A—COST AND PRODUCTIVITY MEASURES
Session Chair: B. W. Boehm, TRW, USA
Quantitatively Evaluating Software Project
Effectiveness, M. Evans, Integrated Computer Engr.,
USA
Productivity Factors & Prog. Environments:
Observations from ITT Prog. Measurement Data,
B. Curtis, J. Vosburgh, R. Wolverton, ITT, USA
Programming Cost Estimate: Is It Reasonable? R.
Boydston, IBM, USA

12:00 PM.-1:30 PM. LUNCH

1:30 PM.-3:00 PM.
KISSIMMEE HALL
SESSION 6A—USER INTERFACES
Session Chair: I. Kimura, Tokyo Institute Technology,
Japan
MGEN—A Generator for Menu-Driven Programs,
B. Friman, FronTech, Sweden
Synchro: A Dataflow Command Shell for Lilith/
Modula Project, T. Demarco, Atlantic Systems Guild,
A. Soceneantu, Brigham Young Univ., USA
INVITED PAPER: Software Engineering for User
Interfaces, D.A. Norman, Univ. of Calif., San Diego,
USA

3:00 P.M.-3:30 P.M. COFFEE BREAK

3:30 P.M.-5:30 P.M.
KISSIMMEE HALL
SESSION 7A—TESTING
Session Chair: C. V. Ramamoorthy, Univ. of Calif.
Berkeley, USA
An Evaluation of Required Element Testing Strategies,
S. Natafson, Univ. Texas-Dallas, USA
Perturbation Testing for Computation Errors, S. Zeil,
Univ. Mass., USA
Assessing a Class of Software Tools, M.A. Hennell,
I.J. Ridell, Univ. of Liverpool, D. Hedley, Liverpool
Data Res. Assoc., England
An Algebra for Data Flow Anomaly Detection,
I. Forman, ITT, USA

ORLANDO HALL
SESSION 4B—PROGRAMMING ENVIRONMENTS I
Session Chair: TBA
PDAS: An Assistant for Detailed Design and
Implementation of Programs, T. Mohri, E. Ono, and
S. Uehara, Fujitsu Labs., Japan
Steps of an Advanced ADA Programming
Environment, T. Standish, R. Taylor, Univ. Calif.-
Irvine, USA
Maiday: An Environment for Guided Programming
with a Definitional Language, J. Guyard, J-P. Jacquot,
Centre de Recherche En Inform. de Nancy, France

ORLANDO HALL
SESSION 5B—PROGRAMMING ENVIRONMENTS II
Session Chair: A. Wasserman, Univ. of Calif., San
Francisco, USA
Interactive System for Structured Program
Production, H. Maezawa, M. Kobayashi, K. Saito,
Hitachi, Japan
A Large System Evaluation of SREM, P. Scheffer,
A. Stone, Martin Marietta, W. Rzepka, Griffiss AFB, USA
INVITED PAPER: The Cedar Programming Environment:
Mid-Term Report & Evaluation, W. Teitelman, Xerox,
USA

ORLANDO HALL
SESSION 6B—RELIABILITY AND COMPLEXITY MEASURES I
Session Chair: A. Marmor-Squires, TRW, USA
Comparison of Concurrent Software Reliability
Models, R. Moawad, Egypt
A Logarithmic Poisson Execution Time Model for
Software Reliability Measurement, J. Musa,
K. Okumoto, Bell Labs., USA
A Program Complexity Metric Based on Data Flow
Information in Control Graphs, K. Tai, Natl. Taiwan
Univ., China

ORLANDO HALL
SESSION 7B—RELIABILITY AND COMPLEXITY MEASURES II
Session Chair: Plo de Feo, Textron, USA
Characteristic Program Complexity Measures, J.
Elshooff, Gen. Motors, USA
A Discriminant Metric for Module Cohesion, T.
Emerson, Bell Labs, USA
Integrated Program Measurement and
Documentation Tools, A. Schroeder, Inria, France
Quantifying Software Designs, J. Beane, J. Silverman,
N. Giddings, Honeywell, USA
THURSDAY, MARCH 29, 1984

8:30 A.M.-10:00 A.M.  KISSIMMEE HALL
SESSION 8A—EDITORS

Session Chair: E. Wada, Univ. Tokyo, JAPAN
PECAN: Program Development Systems that Support Multiple Views, S. REISS, BROWN UNIV., USA
TABLE: An Experiment in Editing Complex Structures, T. BIGGERSTAFF, M. ENDRES, I. FORMAN, ITT, USA
Practical Applications of a Syntax Directed Program Manipulation Environment, V. DONZEAU-GOUJE, B. LANG, B. MELESE, INRIA, FRANCE

10:00 A.M.-10:30 A.M. COFFEE BREAK

10:30 A.M.-12:00 Noon  KISSIMMEE HALL
SESSION 9A—KNOWLEDGE BASED SOFTWARE ENGINEERING
Session Chair: D. R. Barstow, Schlumberger-Doll, USA
Knowledge-Based Communication Proceedings In Software Engineering, G. FISCHER, E. SCHNEIDER, UNIV. OF STUTTGART, W. GERMANY
Proust: Knowledge-Based Program Understanding W.C. JOHNSON, E. SOLOWAY, YALE UNIV., USA
The Intelligent Program Editor: A Knowledge Based System for Supporting Program Development and Maintenance, D. SHAPIRO, B. MCCUNE, ADVANCED INFORMATION AND DECISION SYSTEMS, USA

12:00 Noon-1:30 P.M. LUNCH

1:30 P.M.-3:00 P.M.  KISSIMMEE HALL
SESSION 10A—FORMAL SPECIFICATIONS
Session Chair: M. Nivat, TBA
Semantics of Dialogue Interfaces: A Formal Approach, R. STUDER, UNIV. OF STUTTGART, W. GERMANY

Formal Specifications and Development of an ADA Compiler, O. OEST, G. CLEMMENSEN, DANSK DATAMATIK CENTER, DENMARK
Application of Axiomatic Methods to a Specification Analyser, S. GERHART, WANG INST. USA

3:00 P.M.-3:30 P.M. COFFEE BREAK

3:30 P.M.-5:00 P.M.  KISSIMMEE HALL
SESSION 11A VERIFICATION
Session Chair: C. Ghezzi, Milan Politecnich, ITALY
Structure Charts and Program Correctness Proofs, S.C. CHYOU, TEKTRONIX, USA

Assigning Meanings to Program Verification Methods: A Classifying Survey, A. MILI, UNIV. LAVAL, CANADA

Spec. and Verif. of Distributed Systems Using Prolog Interpreted Petri Nets, P. AZEMA, C. JUANOLE, E. SANCHIS, LAB. D’ANALYSE DES SYSTEMES, FRANCE

5:30 P.M. CLOSING SESSION

ORLANDO HALL
SESSION 8B—SOFTWARE ENGINEERING AND SOCIAL RESPONSIBILITY

Panel
Panelists: TBA

SESSION 9B—DESIGN METHODS
Session Chair: R. T. Yeh, Univ. of MD, USA
Communication System Design Using ADA, A. DUNCAN, J. HUTCHINSON, J. BAILEY, GEN. ELEC., USA

INVITED PAPER: The Modular Structure of Complex Systems, D.L. PARNAS, UNIV. OF VICTORIA, CANADA

SESSION 10B—PROGRAMMING METHODOLOGIES
Session Chair: M. Zelkowitz, Univ. of MD, USA
A New Program Structure to Improve Accuracy and Readability of Pascal Software, W. ROGERS, WAIKATO UNIV., NEW ZEALAND

Functional Programming and Logical Programming for Telegram Analysis Problem, K. TORII, ETL, Y. SUGIYAMA, OSAKA UNIV., Y. MORISAWA, NIPPON UNIVAC, JAPAN

Prototyping vs. Specifying: A Multi-Project Experiment, B. BOEHM, TRW, T. GRAY, T. SEEWALDT, UCLA, USA

SESSION 11B—DEVELOPMENT MODELS AND METHODS
Session Chair: R. Bourgonjon, Phillips, THE NETHERLANDS
A Method of Large-Scale Software Development, M. HOZUMI, Y. SEKINE, H. EJIMA, FUJITSU, JAPAN
On What Exactly Is going on when Software Is Developed Step-by-Step, T.S.E. MAIBAUM, IMPERIAL COLLE, ENGLAND, W.M. TURSKI, WARSAW UNIV., POLAND

Three Paradigms for Developing Information Systems, B. BLUM, JOHNS HOPKINS UNIV., USA
ADVANCE HOTEL REGISTRATION FOR HYATT ORLANDO

WARNING: We have reserved 1000 rooms for the Software Engineering Conference. Because of the popularity of Walt Disney World and the EPCOT Center, if you do not reserve your room by March 14, 1984 you will not be able to get a room within 25 miles of the conference. Reserve your room early. You can always cancel prior to your arrival date and receive a full refund. Your priority for a room reservation expires March 14, 1984. Be sure to mention the Software Engineering Conference in order to get the special room rate and your room reservation prior to March 14, 1984.

Arrival Date ___________________ Departure Date ___________________

Arrival Time ___________________ Flight _________________________

Number of Rooms ___________________ Number in Party _______________

Adults ___________________ Children ___________________

Special Request __________________

Name __________________________

Address _________________________

City/State/Zip ______________________

Daily Room Rates for the Conference $64 single or double
Children under 18 years of age free. Maximum number of persons in room is four.
All rates are subject to 7% room tax.
Reservations must be accompanied by a one night room deposit.

Please charge my one night deposit to my
Visa _____ Master Charge _____ American Express _______
Card No. _______________________
Expiration Date ________________

Signature _______________________

For those who wish to arrive early and/or extend their stay, the $64 rate will apply to two nights before and after the conference.

Advance Registration (prior to 3/12/84) Late Registration (After 3/12/84)
Member Non-Member Member Non-Member

(CIRCLE APPLICABLE FEE)

TUTORIAL 1
Models and Metrics for Software Management & Engineering Victor Basili
$125 $150 $140 $165

TUTORIAL 2
A Builder's Guide to Software Engineering Environments William Riddle
Structured Testing Thomas J. McCabe
$125 $150 $140 $165

TUTORIAL 4
Advances in Software Design P Freeman & A. Wasserman
Conference Only $100 $125 $115 $135
Tutorial 1 & Conference $200 $250 $215 $265
Tutorial 2 & Conference $200 $250 $215 $265
Tutorial 3 & Conference $200 $250 $215 $265
Tutorial 4 & Conference $200 $250 $215 $265

Notes:
Membership registration is available to members of ACM, IEEE, or the IEEE Computer Society or employees of the U.S. Government. All others pay non-member registration fee.
Student Discount is available for Conference Only (fee $55). To receive a discount, students must be IEEE or ACM Members, not employed full-time, and must show membership card at door.
Tutorial registration fee includes tutorial attendance, notes, lunch and coffee breaks.
Conference registration fee includes conference attendance, one copy of the conference proceedings, coffee breaks, and reception on conference days.
Request for refunds must be in writing and postmarked prior to March 22, 1984.
Late registration will be accepted at the Hyatt Orlando starting 4 PM Sunday, March 24, 1984.