FREDERICK T. SHELDON

785 SE Ridgeview 509/595-1179 (C), 509/535-6138 (O)
Pullman, Washington 99163 sheldon@wsu.edu (www.eecs.wsu.edu/~sheldon)

PROFESSIONAL BEARING

Research to develop / validate methods and supporting tools for the creation of safe and correct software.

EDUCATION

University of Texas at Arlington, College of Engineering
Ph.D. in Computer Science (Mjr. Prof. Krishna M. Kavi) May 1996
MS in Computer Science (Mjr. Prof. Krishna M. Kavi) August 1988

University of Minnesota, Institute of Technology
Bachelor of Science in Computer Science December 1983

University of Minnesota, College of Biological Science
Bachelor of Science in Microbiology July 1977

EMPLOYMENT

Washington State University, College of Engineering and Architecture From/To: 06/99 - present
Assistant Professor, School of Electrical Engineering and Computer Science

University of Colorado, College of Engineering and Applied Science From/To: 08/96 – 06/99
Assistant Professor, Computer Science Department

National Research Council, NASA Langley Research Center From/To: 06/96 - 08/96
Research Associate, Flight Electronics Technology Division: Formal specification of stochastic properties for an integrated air/ground system to support low visibility landing and surface operations at commercial airports. Investigated advanced design techniques for ensuring that high-performance computing systems are produced in a timely fashion and meet stringent real-time reliability and fail-safe requirements.

University of Texas at Arlington, College of Engineering From/To: 01/93 - 05/96
Assistant Instructor, Computer Science and Engineering: Taught Methods in Software Engineering, Discrete Structures in Computer Science (+Pascal lab) and Formal Methods for Software. Dissertation title: Specification and analysis of stochastic properties for concurrent systems expressed using CSP. Developed a CSP to Stochastic Petri net translator and investigated automated techniques for decomposition/composition of system models. Studied simulation techniques for verification of performance / dependability ($66K NASA fellowship).

General Dynamics, Fort Worth Division From/To: 01/90 - 01/93
Engineering Specialist (promoted 1992), Avionics R&D: Technical lead for $2-million Contracted R&D project to define a generic Integrated Diagnostics (ID) Software Development Process to address problems associated with functional deficiencies of avionics software and software maturation using as a basis, the Software Development Integrity Program (Mil-Std-1803, plus Mil-Std-2167A, 2168, 1815 and 800-xx Series). We developed a software engineering process model (to specify, develop and verify diagnostic software) and recommendations for updates to Mil-Std-1814. I authored the Software Risk Management Practices and the F-22 Software/System Engineering Environments prime requirements document (System Segment Spec.) and provided tools training. Participated on three proposal writing teams.

General Dynamics, Fort Worth Division From/To: 10/88 - 01/90
Senior Engineer, Advanced Avionics Development: FS/X Japan tailoring F-16 derivative avionics systems components. Lead Engineer for YF-22 VMS Kernel and prime author of it's software requirements specification. Collaborated on the stores management operational flight program, built-in-test, Software Development Plan (2167A), computer resource integration, processing architecture evaluations and white papers, computer data security, and communication interfaces. Developed memory usage estimation algorithm for Ada (Secret clr).

Texas Instruments, Defense Systems and Electronics Group From/To: 9/87 -10/88
Test Engineer (promoted 1988), Test Automation Department: On loan to Military Computer Systems as designer for built-in-test software on the Multipurpose Interface module for the Advanced Tactical Fighter (YF-22) Mission Display Processor using DoD-STD-2167 (Secret clr).

Texas Instruments, Defense Systems and Electronics Group From/To: 7/84 -09/87
Software Design Engineer, Avionics Systems/Radar Division: Designer for hardware fault diagnostic, built-in-self-test, calibration and acceptance test software for the Tornado Nose Radar. Various functionalities included terrain following / ground mapping radar / ECCM and the infrared imaging Lantern pod (College Recruiter).

AWARDS AND RECOGNITION

- 2000 Completed Software System Safety Class [N. Leveson SE Lab, MIT Aeronautics and Astronautics]
- 2000 Completed EEF-Summerschool: Formal Methods and Performance Analysis (FMPA 2000)
- 1997-98 Fellowship: Stanford/NASA Ames Summer Faculty Research Assoc. (Amer. Soc. Engrng Ed.)
- 1996-97 Outstanding UTA Ph.D. Dissertation Award from UTA Chapter for the Society of Sigma Xi
- 1995-6 UTA School of Engineering, Dept. of CSE: Outstanding Research by a Ph.D. Student

- 1996 IEEE First Place Graduate Technical Paper Contest (Ft. Worth Sec) IEEE Region 5 cash award
- 1996 National Research Council Associateship Award at NASA/LaRC (see awarded proposals)
- 1994 University of Texas at Arlington's ACM Programming Contest Team Coach
- 1993 Runners-Up Best Paper at the Fourth Int'l Conf. on Applications of Software Measurement
- 1993-95 NASA Graduate Student Researchers Fellowship Recipient at LaRC (see awarded proposals)
- 1993-95 Member of UTA's Autonomous Aerial Vehicle team for the Aerial Robotics Competition
- 1992 Scholarship for 2 week NATO ASI on Real-time Computing (St. Maarten NL Antilles)
- 1992 Theme Article for *IEEE Software* special issue on Applications of Software Reliability Models
- 1988 Member of Upsilon Pi Epsilon honor society in the computing sciences
- 1986 Member of Tau Beta Pi international engineering honor society
- 1980-81 Elected to University of Minnesota Senate representing the Institute of Technology
- 1981-2 Selected for the University of Minnesota, Institute of Technology student board of publications
- 1981 Selected by the Institute of Technology Dean for a summer workshop on Minnesota Gov.
- 1980-2 Editor and business manager of the University of Minnesota Institute of Technology Yearbook
- 1972-4 Academic scholarship awarded from St. Johns University, Collegeville, Minnesota
- 1970-2 National Honor Society in High School (Minneapolis West, Minnesota)

REFEREED JOURNAL ARTICLES

Sheldon, F.T., Kim, H.Y. and Zhou, Z, "Specification and Analysis of a Real Time Schedule for an Embedded Spaceborn System," Submitted [Dec. '00] *IEEE Mirro (Special Issue on* Embedded Fault-Tolerant Systems) December 2000.

Sheldon, F.T. and Greiner, S.A., "Composing, Analyzing and Validating Software Models to Assess the Performability of Competing Design Candidates," *Annals of Software Engineering –Special Volume on Software Reliability, Testing and Maturity*, Vol. 8, 49 pages, 1999.

Kavi, K.M., **Sheldon, F.T.** and Reed, S.C., "Specification and Analysis of Real-Time Systems Using CSP and Petri Nets," *Int'l Journal of Software Engineering and Knowledge Engineering –Special Issue on Software Engineering Practices and Tools for Real-Time Systems*, 24 Refs., June 1996.

Sheldon, F.T., Kavi, K.M., Everett, W.W., Brettschneider, R., Yu, J.T., and Tausworthe, R.C., "Reliability Measurement: From Theory to Practice," *IEEE Software –Special Issue on Applications of Software Reliability Models*, pp. 13-20, 10 Refs., July 1992.

REFEREED PROCEEDINGS

Sheldon, F.T. and Kim, H.Y., "Hybrid Approach to System Level Validation and Component Verification for Reactive Systems," Submitted [Oct. '00] *ACM Proc. Sigmetrics*, Cambridge, MA, June 17-20, 2001.

Sheldon, F.T. and Kim, H.Y., "Software Requirements Specification and Analysis Using Zed and Statecharts," Submitted [Oct. '00] *Proc. Fundamental Approaches to Software Engineering (in European Joint Conferences on Theory and Practice of Software [ETAPS 2001])*, Genova Italy, Apr. 2-6, 2001.

Sheldon, F.T. Greiner, S.A., and Benzinger, M., "Specification, Safety and Reliability Analysis Using Stochastic Petri Net Models," *ACM Proc. Tenth Int'l Wkshp on Software Specification and Design*, pp. 123-132 Nov. 5-7 2000.

Owens, D.A., and **Sheldon, F.T.,** "A Tool-based Approach to Distributed Database Design," *ACM Symposium on Applied Computing (SAC'99)*, 10 refs., 18 pages, February 1999.

Sheldon, F.T. "Analysis of Real-Time Concurrent Systems Models Based on CSP Using Stochastic Petri Nets," *SCS Proc. 12th European Simulation Multiconference*, pp. 776-783, 22 Refs., Manchestet, UK, June 16-19, 1998.

Sheldon, F.T., Alspector, J. and Haefner, J. "Technical Education over Drive-able Distances using a Portable Classroom and Variable Bandwidth Communication Networks," IEEE Proc. Int'l Symp. on Internet Technology, Apr. 29 - May 1, 1998.

Sheldon, F.T., "Specification and Analysis of Stochastic Properties for Concurrent Systems Expressed Using CSP," Ph.D. Dissertation, *Computer Science and Engineering Dept., Univ. of TX at Arlington,* 260 Refs., May, 1996 (Sigma Xi Scientific Research Society Outstanding Ph.D. Dissertation Award).

Kavi, K.M., and **Sheldon, F.T.**, "Specification and Analysis of Real-Time Systems Using CSP and Petri Nets," *IIT Proceedings 1st Conf. on Fault-Tolerant Systems (FTS'95)*, Madras, India, Dec. 20-22, 1995.

Sheldon, F.T., and Kavi, K.M., "Linking Software Failure Behavior to Specification Characteristics II," *IEEE Proceedings Fourth International Workshop on Evaluation Techniques for Dependable Systems*, San Antonio, TX, 27 Refs., Oct. 1995.

Sheldon, F.T., Kavi, K.M., and Kamangar, F.A., "Reliability Analysis of CSP Specifications: A New Method Using Petri Nets," *AIAA Proc. Computing in Aerospace 10,* pp. 317-326, 16 Refs., March 1995.

Kavi, K.M., **Sheldon, F.T.**, Shirazi, B. and Hurson, Ali R., "Reliability Analysis of CSP Specifications Using Petri Nets and Markov Processes," *IEEE Proc. Hawaii Int'l Conf. on Systems and Sciences,* 10 Refs., Jan. 1995.

Kavi, K.M., and **Sheldon, F.T.**, "Specification of Stochastic Properties with CSP," *IEEE Proceedings Int'l Conference on Parallel and Distributed Systems*, Taiwan, ROC, pp. 288 - 293, 12 Refs., December 1994.

Sheldon, F.T., and Kavi, K.M., "Position Statement: Linking Software Failure Behavior to Specification Characteristics I," *IEEE Proceedings Third International Workshop on Integrating Error Models with Fault Injection*, Annapolis, MD, pp. 35-39, 19 Refs., April 1994.

Sheldon, F.T., Mei, Hsing, and Yang, S.M., "Reliability Prediction of Distributed Embedded Fault-Tolerant Systems," *IEEE Proc. 4th Int'l Symp. on Software Reliability Engineering,* pp. 92-102, 27 Refs., Nov. 1993.

Yang, S.M., Yoo, S.M., Kim, Y.S., Song, Y.J., and **Sheldon, F.T.**, "UTARK: An Object-Based Real-Time Kernel for Distributed Embedded Systems," *IEEE Proc. CompEuro93* (Paris), pp. 392-399, May 1993.

Sheldon, F.T., Yang, S.M., and Bornejko, T.L., "Simulation-Based Analysis for Real-Time Systems Development," *IEEE Proc. Automatic Testing Conf. (AutoTestCon92)*, pp. 361-366, 15 Refs., Sept. 1992.

Sheldon, F.T., Kavi, and Reese, R. "Software Reliability Modeling: A Case Study," *Proceedings General Dynamics Software Technology Conference*, San Diego, CA, pp. 29-1 to 29-14, 32 Refs., April 12, 1991.

Sheldon, F.T., and Kavi, K.M., "A Model for Estimating Software Reliability Based on Residual Defects," IEEE Proceedings *MidCon'90*, Dallas, TX, pp. 22-25, 11 Refs., September, 1990.

Sheldon, F.T., and Kavi, K.M., "A Comparative Evaluation of Two New Advanced Microprocessor Architectures: *The N10 (i860) and the P10 (i960) further promote the RISC trend in 1989,*" *IEEE Proceedings MetroCon'90*, Arlington, TX, 13 pages, 25 Refs., February, 1990.

Sheldon, F.T., and Kavi, K.M., "A New Software Reliability Model for Estimating Failure Rate Due to Residual Software Defects," *Proceedings ACM/IEEE Workshop on Applied Computing*, Stillwater, OK, pp. 7-13, 21 Refs., March, 1989.

INVITED ARTICLES

Sheldon, F.T. and Kim, H.Y, "Software Requirements Specification and Analysis Using Zed and Statecharts," *IEEE 3rd Wkshp on Formal Descriptions and Software Reliability*, San Jose, CA, Nov 7, 2000.

Sheldon, F.T. and Dugan, "Stochastic Petri Nets and Discrete Event Simulation: A Comparative Study of Two Formal Description Methods," *IEEE 3rd Wkshp on Formal Descriptions and Software Reliability*, San Jose, CA, Nov 7, 2000.

Sheldon, F.T., "Automatic Stochastic Analysis of CSP Specifications Using a CSP-to-Petri Net Translation Tool: CSPN," *IEEE Proc. MetroCon* (2nd place Graduate Paper Contest), Arlington. TX, Feb. 1996.

Sheldon, F.T., Mei, Hsing, and Yang, S.M., "Reliability Prediction of Distributed Embedded Fault-Tolerant Systems," Invited paper and presentation in cooperation with ASQC and Centre for Software Reliability (City University, London), *Proceedings Fourth International Conference on Applications of Software Measurement,* 10 pages, 27 Refs., 0.5 hrs., November 1993 (Awarded runners-up best paper).

Sheldon, F.T., Yang, S.M., and Bornejko, T.L., "Position Statement: Simulation-Based Analysis for Real-Time Systems Development," *Proceedings of the NATO Advanced Study Institute on Real-Time Computing*, Springer-Verlag (Berlin), St. Maarten Netherlands Antilles, 6 Refs., October 4-19 1992.

Sheldon, F.T., "Software Development and Reliability Modeling: Software Life Cycle Model," *Masters Thesis, Computer Science and Engineering Dept., Univ. of TX at Arlington*, 89 pgs, 85 Refs., Aug. 1988.

AWARDED PROPOSALS

Brake-Safe, *Proposal to Daimler-Benz AG*, Model-based Specification and Analysis (safety and reliability) Using CSP and Stochastic Petri Nets, (PI: **F.T. Sheldon**: $10,850 requested, $10,850 awarded) Dec. 1998.

Technical Distance Learning Courseware, *Proposal to the Changing the Learning Paradigm through Technology Initiative (Source: Univ. of Colorado's Office of the President)*, (PIs: J. Alspector, **F.T. Sheldon** and J. Haefner: $15,500 requested, $11,100 awarded) Mar. 1998.

A Novel Approach to Model Based Validation of Fault Tolerant Systems: *DARPA SBIR Proposal*, (PI: F.T. Sheldon: $99,000 awarded) September 1996.

Formal Specification of Stochastic Properties for an Integrated Air/Ground System to Support LVLASO: Low Visibility Landing and Surface Operations element of the Terminal Area Productivity program funded by the FAA: *Postdoc Proposal to the National Research Council Associateship Program at NASA Langley Research Center*, ($42,000/yr for 3 yrs awarded) February 1996.

Simulation-Based Analysis for Real-Time Systems Development," *Fellowship Proposal to the NASA Graduate Student Researchers Program, Langley Research Center*, $22,000/yr for 3 yrs awarded on January 1993.

GIMADS (Task 29) Software Contract Change Proposal Technical Lead: researched, developed, coordinated, authored and promoted three CCP revisions (PI: **F.T. Sheldon** $2.1M Awarded, *General Dynamics FWD* 1992).

PROPOSALS PENDING

SEDS: Software Engineering for Dependable Systems: Toolkit for Composing, Analyzing, Verifying and Validating Software Based System Models, *NSF Faculty Early Career Development Program*, (PI: **F.T. Sheldon**: $498,928 requested) July 1999.

TECHNICAL REPORTS / WHITE-PAPERS

Composing, Analyzing and Validating Software Models, (Author: **F.T. Sheldon**, *NASA ARC / Stanford – ASEE Final Report which included the CSPN Software Toolkit*, August 1998).

Specification Based Stochastic Analysis and Diagnostics of Concurrent Embedded Systems, (Author: **F.T. Sheldon**, *ASEE Final Report*, August 1997).

A Novel Approach to Model Based Validation of Fault Tolerant Systems: Addresses the limitations and issues for modeling methods and tools developed specifically for stochastic analysis, performability evaluation, and solution methods suited to systems with low latency requirements and rare events (e.g., single independent and multiple coincident failures). Goal: develop a methodology and toolset for *specification and stochastic and performability analysis* of vital DoD systems (Author: **F.T. Sheldon**, *DARPA SBIR Final Report*: May 1997).

Engineering Modernization: *Unix-Based Software Engineering Environment Feasibility Study* of OSF/1 Open Systems Technology for avionics software development including requirements/risk analysis, characterize critical capabilities, metrics, costs and performance criteria (Author: **F.T. Sheldon**, *General Dynamics FWD* 1991).

Domain Specific Software Architectures White-paper *DARPA Abstract Proposal* (Principal Author: **F.T. Sheldon**, *General Dynamics FWD* 1991).

Avionics Software Domain Analysis: *Engineering Modernization White Paper:* describing this emerging technology as applied to Avionics software synthesis and reuse (Author: **F.T. Sheldon**, *General Dynamics FWD* 1990).

INVITED / SPONSORED PRESENTATIONS

Using Models of Software to Assess the Performability and Reliability of Competing Design Candidates, Summer Faculty Research Review NASA-ASEE Ames Research Center, Moffitt Field, CA, 0.25 hrs., August 10, 1998.

Work-In-Progress Report: Toolkit for Composing and Analyzing Models of Software and Systems., Three Invited Talks at NASA Ames Research Center –Automated Software Engineering group, June-Aug. 1998 (included students Chuck Rodacker and Shane Holloway).

Work-In-Progress Report: Prototype Open Toolkit for Composing and Analyzing Models of Software and Systems., Invited Talk at Fu Jen Catholic University, Taipei, Taiwan, April 1998.

Formal Specification Based Stochastic Analysis and Design of Safety-Critical Systems, Sponsored Talk at Mercedes Benz in Stuttgart, Germany, Jan. 1998.

Formal Specification Based Stochastic Analysis and Design of Safety-Critical Systems, Sponsored Talk at Univ. of Erlangen-Nurnberg, Modeling and Process Control Group, Dept. of Computer Science, Germany, Dec., 1997.

Specification Based Stochastic Analysis and Diagnostics of Concurrent Embedded Systems, Invited Talk at NASA Ames Research Center, Moffitt Field, CA, 0.25 hrs., August 8, 1997.

Specification and Analysis of Stochastic Properties for Concurrent Systems Expressed Using CSP, Invited Talk at NASA Ames Research Center, Moffitt Field, CA, 1.0 hrs., July 2, 1997.

Specification and Analysis of Stochastic Properties for Concurrent Systems Expressed Using CSP, Sponsored Talk at University of Erlangen-Nurnberg, Modeling and Process Control Group, Department of Computer Science IV, Germany, 1.25 hrs., May 20, 1997.

Distributed Real-Time Systems Development: An Assessment, Sponsored Talk at Techniche Universitat (TU) Clausthal, Institute fur Informatik, Germany, 0.75 hrs., May 20, 1993.

Software Reliability Modeling: A Case Study, Proceedings, Integrated Diagnostics Design Technology Tools Conference, Ellicott City, Maryland, NSIA Sponsored, 0.75 hrs., pp. 146-159, May 3, 1991.

SOFTWARE RELIABILITY: Theory, Practice, and Controversy," *Proceedings Joint Logistics Management Committee General Membership and Integrated Diagnostics Meeting*, San Diego, CA, NSIA Sponsored., 0.75 hrs., 23 pages, Jan. 15, 1991.

Panel Moderator: "Software Reliability: Theory, Practice and Controversy," Panelists: Everett, W.W., AT&T Bell Labs; Yu, Jim, AT&T Bell Labs; Tausworthe, R., Jet Propulsion Lab; Brettschneider, R., Motorola; *ACM/IEEE Symposium on Applied Computing*, Fayetteville, AR, April, 1990.

Software Reliability and Quality Metrics, Invited Talk for *Dallas/Fort Worth Software Quality Association*, 1.5 hrs., **Sheldon, F.T.** and Whitfill, W.A., General Dynamics, *FWD*, March, 1990.

Estimating Failure Rate Due to Residual Software Defects and Software Quality Metrics," Talk for *Dallas IEEE Reliability Society*, 1.5 hrs., **Sheldon, F.T.** and Whitfill, W.A., General Dynamics, Oct., 1989.

TITLES OF COURSES TAUGHT

Software Specification and Analysis (CptS 580.1/483.1 – Spr00)

Software Engineering Principles (CptS 422 – Fall99)
Software Engineering (CS 330 – Spr98, Fall98, Spr99)
Software Requirements Analysis and Specification (CS 531 – Spr99)
Formal Methods of Software Systems Engineering (CS 533 – Spr97, Spr99)
Computer Architecture (CS 520/420 – Spr97, Fall97)
Software Design (CS 532 – Fall96, Fall97, Fall98)
Discrete Structures in Computer Science (CSE 2315/3315 was CSE 1442 – Fall93, Spr94, Fall94, Spr95)
Formal Methods: Software Systems (CSE 5312 substitute lectures – Fall95, Spr96)
Fundamentals of Software Engineering (CSE 3310 – Spr93)

Courses designated CSE taught at The University of Texas at Arlington (Computer Science and Engineering Dept.)
Courses designated CS taught at The University of Colorado at Colorado Springs (Computer Science Dept.)
Courses designated CptS taught at The Washington State University (School of Electrical Engineering and Computer Science)

DESCRIPTION OF COURSES TAUGHT

Software Specification and Analysis (CptsS 580.1/483.1 Taught on Wash. Higher Ed. Television – Spr00)

Introduction to formal methods used in software engineering: Formal mechanisms for specifying, validating and verifying the correctness, reliability and efficiency of software systems. The course will first introduce the broad area of formal methods including algebraic and model based specification techniques. The class will then focus on developing a working knowledge for using Z. A project will require the use of the Z language in developing a formal specification of a particular real world requirements specification. Outside readings are assigned that report on a range of independent experiments devoted to broadening the link between theory and practice (i.e., in the application of theories on an industrial scale) including case studies. Prerequisites include knowledge of modern programming languages, data structures, algorithms and discrete mathematics.. There are two exams, a term paper and term project. Distribution of all required course materials will be via the web. Texts: *Using Z: Specification, Refinement, and Proof*, by Woodcock, J., and Davies, J. PH, 1996, and *Specification Case Studies (2nd Ed)*, by Hayes, I., PH, 1993.

Software Engineering Principles (CptsS 422 – Fall99)

See the objectives for "Software Engineering" as specified below.

Software Engineering (CS 330 – Spr98, Fall98, Spr99)

Course Objectives include understanding and knowing: (1) appropriate terminology and the process of developing a large software system. (2) factors that govern the selection of a process model and that affect process maturity, (3) concepts underlying the goals of all major software development and management activity, (4) how to use at least one technique applicable to every major software phase, and finally (5) the utilization of a various distance learning techniques to facilitate the above objectives. A Term project involves development teams of 4 - 7 participants including electronic (and hard copy) submission of software life-cycle artifacts (Software Requirements Specification, Preliminary and Critical Design Reviews, Design Notebook, Test Report, Users Manual and Product Demonstration). There are three exams, several quizzes and article reading homeworks that suppliment lecture coverage and distribution of all required course materials will be via the web. Teams submit weekly progress reports via Email. Texts: Sommerville, I., Software Engineering, 5th ed., 1996, and Brooks, F. P. Jr., The Mythical Man-Month: Essays on Software Engineering, Ann. Ed., 1995.

Software Requirements Analysis and Specification (CS 531 – Spr99)

Students participate in a project involving the analysis and specification of a major software. Coverage includes methods, techniques and tools that support various languages, notations and formalizations for composing and evaluating specification properties (unambiguous, completeness, consistency, etc.) Students are evaluated on their project work (initiative, innovativeness, etc.), and oral presentations (scored on individual and team basis using self and objective assessment methods), homeworks and exams. Prerequisite: Working knowledge of modern programming languages, data structures, algorithms and discrete mathematics. Texts: Requirements Engineering: Processes and Techniques by Gerald Kotonya / Ian Sommerville, (John Wiley) Sept. 1998, and Requirements Engineering: A Good Practice Guide by Sommerville, Ian and Sawyer, Pete (John Wiley) 1997.

Software Systems Engineering Project Lab (CS 539 designed, never taught due to low enrollment)

Students participate in a project involving the development and production of a software system intended for external distribution and use. Duties include requirements and specification analysis and design, implementation, testing, quality assurance, configuration management and documentation. Projects come from the university and from outside sources. Students are evaluated on their project work (initiative, innovativeness, etc.), and oral presentations (scored on individual and team basis using self and objective assessment methods). Capstone course for the Masters of Software Engineering.

Formal Methods of Software Systems Engineering (CS 533 – Spr99)

Elements of discrete mathematics, formal mechanisms for specifying and verifying the correctness, reliability and efficiency of software systems, finite state machines, regular expression, assertions, algebraic and model based specification techniques including case studies (prerequisites include knowledge of modern programming languages, data structures, algorithms and discrete structures) Texts: Using Z: Specification, Refinement and Proof by Woodcock, J. and Davies, J. (Prentice Hall, Int'l 1996), Specification Case Studies by Ian Hayes 2nd ed.

(Prentice Hall, Int'l 1993 - Opt'l), Application of Formal Methods by Hinchey, M.G., and Bowen, J.P. (Prentice Hall, Int'l 1995 - Opt'l) Formal Methods for Real-Time Computing Edited by C. Heitmeyer and D. Mandrioli (John Wiley, 1996 - Opt'l).

Formal Methods of Software Systems Engineering (CS 533 – Spr97)

Elements of discrete mathematics, formal mechanisms for specifying and verifying the correctness, reliability and efficiency of software systems, finite state machines, regular expression, algebraic and operational specification techniques, Petri nets and Markov modeling (prerequisites include knowledge of modern programming languages, data structures, algorithms and discrete structures) Text: Linz, P., An introduction to Formal Languages and Automata, Ross, S.M., Introduction to Probability Models, Marsan, A., et. al., Modeling with Generalized Stochastic Petri Nets.

Computer Architecture (CS 420/520 – Spr97, Fall97)

Specification and logical design of digital computer systems. Examines the functional basis of structures including control, memory hierarchy, *real* instruction set architectures, processor design (pipelining, microprogramming, arithmetic), I/O and peripherals, communications and networking. Text: Heuring, V.P., and Jordan, H.F., Computer Systems Design and Architecture.

Software Design (CS 532 – Fall96, Fall97, Fall98)

Covers principles underlying a variety of methodologies and tools for design of sequential, parallel and distributed software systems, design language, viewpoints, graphical representations, data abstraction, data dictionaries, data flow analysis and structured design, and object-oriented design. Text: Witt, B.I., et. al., Software Architecture and Design; Budgen, D., Software Design; Rumbaugh, J., et. al., Object-Oriented Modeling and Design.

Fundamentals of Software Engineering (CSE 3310 – Spr93)

Software engineering principles, processes and techniques; software development approaches focusing on functional analysis and functional design methods. Configuration management, implementation strategies, and testing. Team project involving reverse engineering and forward engineering of a discrete event simulation environment. Text: Chris Gane and Trish Sarsen, Structured Systems Analysis: Tools and Techniques; Alan M. Davis, Software Requirements: Objects, Functions and States; selected videos and periodical articles.

Discrete Structures in Computer Science (CSE 2315/3315 [originally 1442] Fall93, Spr94, Fall94, Spr95)

Propositional and predicate logic, mathematical proof techniques, sets, combinatorics, functions and relations, Boolean algebra, graphs, graph algorithms, finite state machines, regular expressions, automata and formal languages, computability, complexity. A weekly three hour compulsory programming laboratory meets for implementation of a requiring assignment associated with the lecture topics. Text: Judith L. Gersting, Mathematical Structures for Computer Science.

Formal Methods: Software Systems (CSE 5312 substitute lectures 1995-6)

Abstractions used to build correct, reliable, and efficient systems. Formal techniques for specifying abstractions and for defining hierarchies of computation, regular expressions and context-free languages; formal logic and proof techniques as they relate to computer science. Texts: Sudkamp, Languages and Machines: An introduction to the Theory of Comp. Sci.; R. E. Davis, Truth, Deduction and Computation: Logic and Semantics for CS.

STUDENT ADVISEES

Major Advisor (Washington State University)

David Dugan – MS Software Engineering: Safety and Reliability Analysis using Stochastic Petri Net Models (Fall99-present)

Wen Wei – MS Software Engineering: Adaptation and Implementation and Integration of Graph Layout Algorithms for a Petri Net Graphical Editor (Fall99-present)

Major Advisor (University of Colorado)

Norb Gravelle – MS Software Systems Engineering: Petri Net Graphical Editor for C-Based Stochastic Petri net Lang. (CSPL) (Spr98-Spr99)

Charles Rodacker – MS Software Systems Engineering: Translation of Promela Specified Models into Stochastic Petri Nets (Spr98-present)

Matthew Berdine – BS Independent Study: Refine PCSP Grammar for Software Specification and Stochastic Analysis (Spr98)

David Owens – MS Software Systems Engineering: Design for Data Integrity for a Distributed Database System (Spr99-Spr98)

Committee Member (University of Colorado)

Myron Berg – MS: Pattern Detection of TINA's Canonical Representation (used Petri Nets) (Spr98)

Dixon L. Miller – MS: Software Cost Estimation (Spr98)

Stefan Greiner – Ph.D.: University of Erlangen, Germany, Software Performance Modeling (Fall99)

John Robert Billups IV – MS: Identification and Visualization of Emotions in Music (F97)

Ann Zweig – MS: Language-Independent Requirements Language Displayer (F97)

Doug Collins – MS: Off Board Simulator (OBSim): Object Oriented Design (Spr97)

Monica L. Midkiff IV – MS: SW System in C/Unix Comparison with one in Foxbase/PC (Spr97)

William B. Camp – MS: Execution Mechanism for a Train Intersection (Spr97)

PROFESSIONAL SERVICE ACTIVITIES

UCCS Service Activities and Material Gifts

- Transportation and Traffic Committee member (1997-98)
- Obtained $20,000 Software Gift-in-Kind from Mercedes Benz (1998)
- CS Undergraduate curriculum and SSE graduate curriculum committee (1996-97)
- Computer Science Newsletter: *Cyberdog,* Editor (1996 and 97)
- Active participant in the EAS Open Houses (1996-97)
- Active participant in the Dean's (on campus) Advisory Committee Meetings (1996-97)
- Very active participant in the *Software Engineering Series Certificate Program* (taught Introduction to S/E, Software Design and helped with Configuration Management) for Spr.'97
- Obtained $16,000 donation of UltraSparc1 Workstation from Sun / UniSoft Consult Fall'96
- Obtained $2,000 donation of reference books from O'Reilly and Associates, Inc. (Sept. '96)

Reviewer

IEEE Transactions on Computers
IEEE Software
Proceedings of the IEEE
Journal of Systems and Software
IEEE Transactions on Aerospace and Electronic Systems
Design Automation for Embedded Systems
Kluwer Academic Publishers
IEEE Int'l Symp on Software Reliability Engineering
IEEE Pacific Rim Int'l Symp on Fault Tolerant Systems
Society for Computer Simulation
National Science Foundation

Editor

Software Reliability Engineering Case Studies, Book published by IEEE CS Press, distributed at the International Symposium on Software Reliability Engineering, November 1997

Program Committee Member

1997 IEEE Int'l Symp on Software Reliability Engineering and Session Chair (Process and Quality Track)
1996 IEEE High-Assurance Systems Engineering Workshop
1996 IEEE 3rd Symposium on the Assessment of Software Tools

Professional Organization Membership

IEEE Computer Society (since 1985)
IEEE Technical Committee on Software Engineering (since 1995)
IEEE Reliability Society (since 1987)
Association of Computing Machinery (since 1995)
American Institute of Aeronautics and Astronautics (since 1992)
Dallas/Fort Worth Association for Software Engineering Excellence (since 1988)

TECHNICAL SPECIALTIES

Platforms

Unix, Mac OS and Windows/NT (ordered by preference)

Areas of Specialty in Software Engineering

System specification, modeling, refinement and performability analysis
Software requirements specification and analysis
Software architecture and design
Software reliability, diagnostics, testing and integration
Formal methods and tools

Application Domains

Real-time embedded systems and safety critical avionics applications
Fault-tolerant techniques and diagnostics (BIT/BIST/testability)
Operating systems and interfaces
Hardware/software co-design and implementation