Read Online Automatic Verification Of Behavioral Specifications In Software Intensive Systems

KEY=BEHAVIORAL - MOODY NORRIS

Automatic Verification of Behavioral Specifications in Software Intensive Systems Towards Systematic Software Security Hardening Marc-André Laverdière **In this thesis, we report our research on systematic security hardening. We see how the software development industry is currently relying on highly-qualified security experts in order to manually improve existing software, which is a costly and error-prone approach. In response to this situation, we propose an approach that enables systematic security hardening by non-experts. We first study the existing methods used to remedy software vulnerabilities and use this information to determine a classification and definition for security hardening. We then see how the state of the art in secure coding, patterns and aspect-oriented programming (AOP) can be leveraged to enable systematic software security improvements, independently from the users' security expertise. We also present improvements on AOP that are necessary in order for this approach to be realizable. The first improvement, GAFlow and GDFlow, two new pointcut constructors, allow the injection of code that precedes or follows any of the points in the input set, facilitating the development of reusable patterns. The second, ExportParameter and ImportParameter, allow us to safely pass parameters between different parts of the program. Afterwards, we leverage our previous findings in the definition of SHL, the Security Hardening Language. SHL is designed in order to permit language-independent expression of security hardening plans and security hardening patterns in an aspect-oriented manner which enables refinement of patterns into concrete solutions. We then demonstrate the viability of this approach by applying it to add a security feature to the APT package acquisition and management system. Proceedings of the International Research Training Groups Workshop 2006 6 - 8th November 2006, Dagstuhl** GITO mbH Verlag **Object-Oriented Behavioral Specifications** Springer **Object-Oriented Behavioral Specifications encourages builders of complex information systems to accelerate their move to using the approach of a scientific discipline in analysis rather than the approach of a craft. The focus is on understanding customers' needs and on precise specification of understanding gained through analysis. Specifications must bridge any gaps in understanding about business rules among customers, Subject Matter Experts, and `computer people', must inform decisions about reuse of software and systems, and must enable review of semantics over time. Specifications need to describe semantics rather than syntax, and to do that in an abstract and precise manner, in order to create software systems that satisfy business rules. The papers in this book show various ways of designing elegant and clear specifications which are reusable, lead to savings of intellectual effort, time, and money, and which contribute to the reliability of software and systems. Object-Oriented Behavioral Specifications offers a fresh treatment of the object-oriented paradigm by examining the limitations of traditional OO methodologies and by describing the significance of competing trends in OO modeling. The book builds on four years of successful OOPSLA workshops (1991-1995) on behavior semantics. This book deals with precise specifications of `what' is accomplished by the business and `what' is to be done by a system. The book includes descriptions of successful use of abstract and precise specification in industry. It draws on the experience of experts from industrial and academic settings and benefits from international participation. Collective behavior, neglected in some treatment of the OO paradigm, is addressed explicitly in this book. The book does not take `reuse' of specifications or software for granted, but furnishes a foundation for taking as rigorous an approach to reuse decisions as to precise specifications in original developments. New Trends in Software Methodologies, Tools and Techniques Proceedings of the Eighth SoMeT_09** IOS Press **"Papers presented at the Eighth International Conference on New Trends in Software Methodologies, Tools and Techniques, (SoMeT 09) held in Prague, Czech Republic ... from September 23rd to 25th 2009."--P. v. Collective Specification and Verification of Behavioral Models and Object-oriented Implementations We present a finite-state-machine-based language, iFSM, to seamlessly integrate the behavioral logic and implementation strategies of object-oriented applications to prevent their design and implementation from being out-of-sync. The**

language allows developers to focus on higher-level abstractions to support software analysis and design instead of focusing on language or architecture specific details. To support the language, we provide a transformation engine which automatically translates iFSM specifications to lower-level C++/Java implementations of object-oriented classes. The auto-generated implementations from iFSM are similar in style to the manually written code, so that they are readable and easily integrable with the existing code. Our experiments show that their performance is similarly comparable to that of manually written programs. We have automatically verified that these implementations are consistent with their behavioral models by translating iFSM specifications into the input language of model checker NuSMV. Component-Based Software Engineering 11th International Symposium, CBSE 2008, Karlsruhe, Germany, October 14-17, 2008, Proceedings Springer **On behalf of the Organizing Committee we are pleased to present the p- ceedings of the 2008 Symposium on Component-Based Software Engineering (CBSE). CBSE is concerned with the development of software-intensivesystems from independently developed software-building blocks (components), the - velopment of components, and system maintenance and improvement by means of component replacement and customization. CBSE 2008 was the 11th in a series of events that promote a science and technology foundation for achieving predictable quality in software systems through the use of software component technology and its associated software engineering practices. WewerefortunatetohaveadedicatedProgramCommitteecomprisingmany internationallyrecognizedresearchersandindustrialpractitioners.Wewouldlike to thank the members of the Program Committee and associated reviewers for their contribution in making this conference a success. We received 70 subm- sions and each paper was reviewed by at least three Program Committee m- bers (four for papers with an author on the Program Committee). The entire reviewing process was supported by the Conference Management Toolkit p- vided by Microsoft. In total, 20 submissions were accepted as full papers and 3 submissions were accepted as short papers. Specification, Algebra, and Software Essays Dedicated to Kokichi Futatsugi** Springer **This Festschrift volume, published in honor of Kokichi Futatsugi, contains 31 invited contributions from internationally leading researchers in formal methods and software engineering. Prof. Futatsugi is one of the founding fathers of the field of algebraic specification and verification and is a leading researcher in formal methods and software engineering. He has pioneered and advanced novel algebraic methods and languages supporting them such as OBJ and CafeOBJ and has worked tirelessly over the years to bring such methods and tools in contact with software engineering practice. This volume contains contributions from internationally leading researchers in formal methods and software engineering. Software Engineering Research, Management and Applications** Springer **The purpose of the 11th International Conference on Software Engineering Research, Management and Applications (SERA 2013) held on August 7 - 9, 2012 in Prague, Czech Republic was to bring together scientists, engineers, computer users, and students to share their experiences and exchange new ideas and research results about all aspects (theory, applications and tools) of Software Engineering Research, Management and Applications, and to discuss the practical challenges encountered along the way and the solutions adopted to solve them. The conference organizers selected 17 outstanding papers from those papers accepted for presentation at the conference in order to publish them in this volume. The papers were chosen based on review scores submitted by members of the program committee, and further rigorous rounds of review. Cloud Computing and Security First International Conference, ICCCS 2015, Nanjing, China, August 13-15, 2015. Revised Selected Papers** Springer **This book constitutes the proceedings of the International Conference on Cloud Computing and Security (ICCCS 2015) will be held on August 13-15, 2015 in Nanjing, China. The objective of ICCCS 2015 is to provide a forum for researchers, academicians, engineers, industrial professionals, students and government officials involved in the general areas of information security and cloud computing. Electronic Design Automation for IC System Design, Verification, and Testing** CRC Press **The first of two volumes in the Electronic Design Automation for Integrated Circuits Handbook, Second Edition, Electronic Design Automation for IC System Design, Verification, and Testing thoroughly examines system-level design, microarchitectural design, logic verification, and testing. Chapters contributed by leading experts authoritatively discuss processor modeling and design tools, using performance metrics to select microprocessor cores for integrated circuit (IC) designs, design and verification languages, digital simulation, hardware acceleration and emulation, and much more. New to This Edition: Major updates appearing in the initial phases of the design flow, where the level of abstraction keeps rising to support more functionality with lower non-recurring engineering (NRE) costs Significant revisions reflected in the final phases of the design flow, where the complexity due to smaller and smaller geometries is compounded by the slow progress of shorter wavelength lithography New coverage of cutting-edge applications and approaches realized in the decade since publication of the previous edition—these are illustrated by new chapters on high-level synthesis, system-on-chip (SoC) block-based design, and back-annotating system-level models Offering improved depth and modernity, Electronic Design Automation for IC System Design, Verification, and Testing provides a valuable, state-of-the-art reference for electronic design automation (EDA) students, researchers, and professionals. Advances in Software Engineering, Education, and e-Learning Proceedings from FECS'20, FCS'20, SERP'20, and EEE'20** Springer Nature **This book presents the proceedings of four conferences: The 16th International Conference on Frontiers in Education: Computer Science and Computer Engineering + STEM (FECS'20), The 16th International Conference on Foundations of Computer Science (FCS'20), The 18th International Conference on Software Engineering Research and Practice (SERP'20), and The 19th International Conference on e-Learning, e-Business, Enterprise Information Systems, & e-Government (EEE'20). The conferences took place in Las Vegas, NV, USA, July 27-30, 2020 as part of the larger 2020 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE'20), which features 20 major tracks. Authors include academics, researchers, professionals, and students. This book contains an open access chapter entitled, "Advances in Software Engineering, Education, and e-Learning". Presents**

the proceedings of four conferences as part of the 2020 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE'20); Includes the tracks Computer Engineering + STEM, Foundations of Computer Science, Software Engineering Research, and e-Learning, e-Business, Enterprise Information Systems, & e-Government; Features papers from FECS'20, FCS'20, SERP'20, EEE'20, including one open access chapter. System-Level Synthesis Springer Science & Business Media **System-Level Synthesis deals with the concurrent design of electronic applications, including both hardware and software. The issue has become the bottleneck in the design of electronic systems, including both hardware and software, in several major industrial fields, including telecommunications, automotive and aerospace engineering. The major difficulty with the subject is that it demands contributions from several research fields, including system specification, system architecture, hardware design, and software design. Most existing book cover well only a few aspects of system-level synthesis. The present volume presents a comprehensive discussion of all the aspects of system-level synthesis. Each topic is covered by a contribution written by an international authority on the subject. Hardware/Software Co-Design Principles and Practice** Springer Science & Business Media **Introduction to Hardware-Software Co-Design presents a number of issues of fundamental importance for the design of integrated hardware software products such as embedded, communication, and multimedia systems. This book is a comprehensive introduction to the fundamentals of hardware/software co-design. Co-design is still a new field but one which has substantially matured over the past few years. This book, written by leading international experts, covers all the major topics including: fundamental issues in co-design; hardware/software co-synthesis algorithms; prototyping and emulation; target architectures; compiler techniques; specification and verification; system-level specification. Special chapters describe in detail several leading-edge co-design systems including Cosyma, LYCOS, and Cosmos. Introduction to Hardware-Software Co-Design contains sufficient material for use by teachers and students in an advanced course of hardware/software co-design. It also contains extensive explanation of the fundamental concepts of the subject and the necessary background to bring practitioners up-to-date on this increasingly important topic. Embedded Systems Handbook** CRC Press **Embedded systems are nearly ubiquitous, and books on individual topics or components of embedded systems are equally abundant. Unfortunately, for those designers who thirst for knowledge of the big picture of embedded systems there is not a drop to drink. Until now. The Embedded Systems Handbook is an oasis of information, offering a mix of basic a A Functional Start to Computing with Python** CRC Press **A Functional Start to Computing with Python enables students to quickly learn computing without having to use loops, variables, and object abstractions at the start. Requiring no prior programming experience, the book draws on Python's flexible data types and operations as well as its capacity for defining new functions. Along with the specifics of Python, the text covers important concepts of computing, including software engineering motivation, algorithms behind syntax rules, advanced functional programming ideas, and, briefly, finite state machines. Taking a student-friendly, interactive approach to teach computing, the book addresses more difficult concepts and abstractions later in the text. The author presents ample explanations of data types, operators, and expressions. He also describes comprehensions—the powerful specifications of lists and dictionaries—before introducing loops and variables. This approach helps students better understand assignment syntax and iteration by giving them a mental model of sophisticated data first. Web Resource The book's supplementary website at http://functionalfirstpython.com/ provides many ancillaries, including: Interactive flashcards on Python language elements Links to extra support for each chapter Unit testing and programming exercises An interactive Python stepper tool Chapter-by-chapter points Material for lectures Hardware and Software: Verification and Testing 9th International Haifa Verification Conference, HVC 2013, Haifa, Israel, November 5-7, 2013, Proceedings** Springer **This book constitutes the refereed proceedings of the 9th International Haifa Verification Conference, HVC 2013, held in Haifa, Israel in November 2013. The 24 revised full papers presented were carefully reviewed and selected from 49 submissions. The papers are organized in topical sections on SAT and SMT-based verification, software testing, supporting dynamic verification, specification and coverage, abstraction and model presentation. Models in Software Engineering Workshops and Symposia at MODELS 2009, Denver, CO, USA, October 4-9, 2009. Reports and Revised Selected Papers** Springer **This book constitutes a collection of the best papers selected from 9 workshops and 2 symposia held in conjunction iwth MODELS 2009, the 12 International Conference on Model Driven Engineering Languages and Systems, in Denver, CO, USA, in October 2009. The first two sections contain selected papers from the Doctoral Symposium and the Educational Symposium, respectively. The other contributions are organized according to the workshops at which they were presented: 2nd International Workshop on Model Based Architecting and Construction of Embedded Systems (ACES-MB'09); 14th International Workshop on Aspect-Oriented Modeling (AOM); Models@run.time (Models@run.time); Model-driven Engineering, Verification, and Validation: Integrating Verification and Validation in MDE (MoDeVVa09); Models and Evolution (MoDSE-MCCM); Third International Workshop on Multi-Paradigm Modeling (MPM09); The Pragmatics of OCL and Other Textual Specification Languages (OCL); 2nd International Workshop on Non-Functional System Properties in Domain Specific Modeling Languages (NFPinDSML); and 2nd Workshop on Transformation and Weaving OWL Ontologies and MDE/MDA (TWOMDE2009). Each section includes a summary of the workshop. SOFSEM 2007: Theory and Practice of Computer Science 33nd Conference on Current Trends in Theory and Practice of Computer Science, Harrachov, Czech Republic, January 20-26, 2007, Proceedings** Springer Science & Business Media **This book constitutes the refereed proceedings of the 33rd Conference on Current Trends in Theory and Practice of Computer Science, SOFSEM 2007, held in Harrachov, Czech Republic in January 2007. The 69 revised full papers, presented together with 11 invited contributions were carefully reviewed and selected from 283 submissions. The papers were organized in four topical tracks. Deductive Software Verification: Future Perspectives Reflections on the Occasion of 20 Years of KeY** Springer Nature **This book presents reflections on the occasion of 20 years on the KeY project that**

focuses on deductive software verification. Since the inception of the KeY project two decades ago, the area of deductive verification has evolved considerably. Support for real world programming languages by deductive program verification tools has become prevalent. This required to overcome significant theoretical and technical challenges to support advanced software engineering and programming concepts. The community became more interconnected with a competitive, but friendly and supportive environment. We took the 20-year anniversary of KeY as an opportunity to invite researchers, inside and outside of the project, to contribute to a book capturing some state-of-the-art developments in the field. We received thirteen contributions from recognized experts of the field addressing the latest challenges. The topics of the contributions range from tool development, effciency and usability considerations to novel specification and verification methods. This book should offer the reader an up-to-date impression of the current state of art in deductive verification, and we hope, inspire her to contribute to the field and to join forces. We are looking forward to meeting you at the next conference, to listen to your research talks and the resulting fruitful discussions and collaborations. Computer Information Systems and Industrial Management 14th IFIP TC 8 International Conference, CISIM 2015, Warsaw, Poland, September 24-26, 2015, Proceedings Springer **This book constitutes the proceedings of the 14th IFIP TC 8 International Conference on Computer Information Systems and Industrial Management, CISIM 2015, held in Warsaw, Poland, in September 2015. The 47 papers presented in this volume were carefully reviewed and selected from about 80 submissions. The main topics covered are biometrics, security systems, multimedia, classification and clustering with applications, and industrial management. Behavioral Modeling for Embedded Systems and Technologies: Applications for Design and Implementation Applications for Design and Implementation** IGI Global **"This book provides innovative behavior models currently used for developing embedded systems, accentuating on graphical and visual notations"--Provided by publisher. Software Specification Methods** John Wiley & Sons **This title provides a clear overview of the main methods, and has a practical focus that allows the reader to apply their knowledge to real-life situations. The following are just some of the techniques covered: UML, Z, TLA+, SAZ, B, OMT, VHDL, Estelle, SDL and LOTOS. An Integrated Approach to Software Engineering** Springer Science & Business Media **It is clear that the development of large software systems is an extremely complex activity, which is full of various opportunities to introduce errors. Software engineering is the discipline that provides methods to handle this complexity and enables us to produce reliable software systems with maximum productivity. An Integrated Approach to Software Engineering is different from other approaches because the various topics are not covered in isolation. A running case study is employed throughout the book, illustrating the different activity of software development on a single project. This work is important and instructive because it not only teaches the principles of software engineering, but also applies them to a software development project such that all aspects of development can be clearly seen on a project. Software Engineering and Formal Methods SEFM 2015 Collocated Workshops: ATSE, HOFM, MoKMaSD, and VERY*SCART, York, UK, September 7-8, 2015. Revised Selected Papers** Springer **This book constitutes revised selected papers from the workshopscollocated with the SEFM 2015 conference on Software Engineering andFormal Methods, held in York, UK, in September 2015.The 25 papers included in this volume were carefully reviewed and selected from 32 submissions. The satellite workshops provided a highly interactive and collaborative environment for researchers and practitioners from industry and academia to discuss emerging areas of software engineering and formal methods.The four workshops were: ATSE 2015: The 6th Workshop on Automating Test Case Design, Selection and Evaluation; HOFM 2015: The 2nd Human-Oriented Formal Methods Workshop; MoKMaSD 2015: The 4th International Symposium on Modelling and Knowledge Management Applications: Systems and Domains; VERY*SCART 2015: The 1st International Workshop on the Art of Service Composition and Formal Verification for Self-* Systems. Rapid Integration of Software Engineering Techniques First International Workshop, RISE 2004, Luxembourg-Kirchberg, Luxembourg, November 26, 2004, Revised Selected Papers** Springer Science & Business Media **This book constitutes the thoroughly refereed postproceedings of the First International Workshop on Rapid Integration of Software Engineering Techniques, RISE 2004, held in Luxembourg-Kirchberg, Luxembourg in November 2004. The 12 revised full papers presented together with an invited paper went through two rounds of reviewing and improvement and were selected from 28 initial submissions. Among the topics addressed are software architecture, software process, component-driven design, dynamic service verification, model checking, model-based testing, exception handling, metamodeling, UML, state machines, and model-centric development. Verified Software: Theories, Tools, Experiments First IFIP TC 2/WG 2.3 Conference, VSTTE 2005, Zurich, Switzerland, October 10-13, 2005, Revised Selected Papers and Discussions** Springer Science & Business Media **A Step Towards Verified Software Worries about the reliability of software are as old as software itself; techniques for allaying these worries predate even James King's 1969 thesis on "A program verifier. " What gives the whole topic a new urgency is the conjunction of three phenomena: the blitz-like spread of software-rich systems to control ever more facets of our world and our lives; our growing impatience with deficiencies; and the development—proceeding more slowly, alas, than the other two trends—of techniques to ensure and verify software quality. In 2002 Tony Hoare, one of the most distinguished contributors to these advances over the past four decades, came to the conclusion that piecemeal efforts are no longer sufficient and proposed a "Grand Challenge" intended to achieve, over 15 years, the production of a verifying compiler: a tool that while processing programs would also guarantee their adherence to specified properties of correctness, robustness, safety, security and other desirable properties. As Hoare sees it, this endeavor is not a mere research project, as might normally be carried out by one team or a small consortium of teams, but a momentous endeavor, comparable in its scope to the successful mission to send a man to the moon or to the sequencing of the human genome. Fundamental Approaches to Software Engineering 19th International Conference, FASE 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands,**

April 2-8, 2016, Proceedings Springer **This book constitutes the proceedings of the 19th International Conference on Fundamental Approaches to Software Engineering, FASE 2016, which took place in Eindhoven, The Netherlands, in April 2016, held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016. The 23 full papers presented in this volume were carefully reviewed and selected from 90 submissions. They were organized in topical sections named: concurrent and distributed systems; model-driven development; analysis and bug triaging; probabilistic and stochastic systems; proof and theorem proving; and verification. A System-Theoretic Safety Engineering Approach for Software-Intensive Systems** Cuvillier Verlag **Software safety is a crucial aspect during the development of modern safety-critical systems. However, safety is a system level property, and therefore, must be considered at the system-level to ensure the whole system's safety. In the software development process, formal verification and functional testing are complementary approaches which are used to verify the functional correctness of software; however, even perfectly reliable software could lead to an accident. The correctness of software cannot ensure the safe operation of safety-critical software systems. Therefore, developing safety-critical software requires a more systematic software and safety engineering process that enables the software and safety engineers to recognize the potential software risks. For this purpose, this dissertation introduces a comprehensive safety engineering approach based on STPA for Software-Intensive Systems, called STPA SwISs, which provides seamless STPA safety analysis and software safety verification activities to allow the software and safety engineers to work together during the software development for safety-critical systems and help them to recognize the associated software risks at the system level. Verified Software: Theories, Tools, and Experiments 7th International Conference, VSTTE 2015, San Francisco, CA, USA, July 18-19, 2015. Revised Selected Papers** Springer **This volume constitutes the thoroughly refereed post-conference proceedings of the 7th International Conference on Verified Software: Theories, Tools and Experiments, VSTTE 2015, held in July 2015 in San Francisco, CA, USA. The 12 revised full papers presented were carefully revised and selected from 25 submissions. The goal of this conference is to advance the state of the art in the science and technology of software verification, through the interaction of theory development, tool evolution, and experimental validation and large-scale verification efforts that involve collaboration, theory unification, tool integration, and formalized domain knowledge. Handbook of Research on Software Engineering and Productivity Technologies: Implications of Globalization Implications of Globalization** IGI Global **"This book provides integrated chapters on software engineering and enterprise systems focusing on parts integrating requirements engineering, software engineering, process and frameworks, productivity technologies, and enterprise systems"--Provided by publisher. System-level Test and Validation of Hardware/Software Systems** Springer Science & Business Media **New manufacturing technologies have made possible the integration of entire systems on a single chip. This new design paradigm, termed system-on-chip (SOC), together with its associated manufacturing problems, represents a real challenge for designers. SOC is also reshaping approaches to test and validation activities. These are beginning to migrate from the traditional register-transfer or gate levels of abstraction to the system level. Until now, test and validation have not been supported by system-level design tools so designers have lacked the infrastructure to exploit all the benefits stemming from the adoption of the system level of abstraction. Research efforts are already addressing this issue. This monograph provides a state-of-the-art overview of the current validation and test techniques by covering all aspects of the subject including: modeling of bugs and defects; stimulus generation for validation and test purposes (including timing errors; design for testability. Algebraic Methodology and Software Technology 7th International Conference, AMAST'98, Amazonia, Brazil, January 4-8, 1999, Proceedings** Springer **AMAST's goal is to advance awareness of algebraic and logical methodology as part of the fundamental basis of software technology. Ten years and seven conferences after the start of the AMAST movement, I believe we are attaining this. The movement has propagated throughout the world, assembling many enthusiastic specialists who have participated not only in the conferences, which are now annual, but also in the innumerable other activities that AMAST promotes and supports. We are now facing the Seventh International Conference on Algebraic Methodology and Software Technology (AMAST'98). The previous meetings were held in Iowa City, USA (1989 and 1991), in Enschede, The Netherlands (1993), in Montreal, Canada (1995), in Munich, Germany (1996), and in Sydney, Australia (1997). This time it is Brazil's turn, in a very special part of this colorful country – Amazonia. Thus, "if we have done more it is by standing on the shoulders of giants." The effort started by Teodor Rus, Arthur Fleck, and William A. Kirk at AMAST'89 was consolidated in AMAST'91 by Teodor Rus, Maurice Nivat, Charles Rattray, and Giuseppe Scollo. Then came modular construction of the building, wonderfully carried out by Giuseppe Scollo, Vangalur Alagar, Martin Wirsing, and Michael Johnson, as Program Chairs of the AMAST conferences held between 1993 and 1997. Design Automation of Cyber-Physical Systems** Springer **This book presents the state-of-the-art and breakthrough innovations in design automation for cyber-physical systems.The authors discuss various aspects of cyber-physical systems design, including modeling, co-design, optimization, tools, formal methods, validation, verification, and case studies. Coverage includes a survey of the various existing cyber-physical systems functional design methodologies and related tools will provide the reader unique insights into the conceptual design of cyber-physical systems. Microprocessor Based Protection Systems** Springer Science & Business Media **From a symposium, or perhaps a series of symposia (no information is provided) 15 papers discuss the use of computers to control potentially hazardous industrial processes. The sections cover guidelines, standards, and design; reliability analysis; software production and research; and industrial ap Asian Test Symposium Verification and Validation in Systems Engineering Assessing UML/SysML Design Models** Springer Science & Business Media **At the dawn of the 21st century and the information age, communication and c- puting power are becoming ever increasingly available, virtually pervading almost every aspect of modern socio-economical interactions. Consequently, the potential for realizing a signi?cantly greater number of technology-mediated activities has emerged. Indeed, many of our modern**

activity ?elds are heavily dependant upon various underlying systems and software-intensive platforms. Such technologies are commonly used in everyday activities such as commuting, traf?c control and m- agement, mobile computing, navigation, mobile communication. Thus, the correct function of the forenamed computing systems becomes a major concern. This is all the more important since, in spite of the numerous updates, patches and ?rmware revisions being constantly issued, newly discovered logical bugs in a wide range of modern software platforms (e. g. , operating systems) and software-intensive systems (e. g. , embedded systems) are just as frequently being reported. In addition, many of today's products and services are presently being deployed in a highly competitive environment wherein a product or service is succeeding in most of the cases thanks to its quality to price ratio for a given set of features. Accordingly, a number of critical aspects have to be considered, such as the ab- ity to pack as many features as needed in a given product or service while c- currently maintaining high quality, reasonable price, and short time -to- market. Feature Interactions in Telecommunications Systems IOS Press **Features are modifications to the control of telecommunications services. A feature interaction occurs when the behaviour of another, which can lead to unexpected or undesired behaviour, which affects the quality of service. The goal of this volume is to generate a combination of techniques through protocol engineering, software testing, formal techniques and AI and applications to telecommunications services. Computing and Software Science State of the Art and Perspectives** Springer Nature **The papers of this volume focus on the foundational aspects of computer science, the thematic origin and stronghold of LNCS, under the title "Computing and Software Science: State of the Art and Perspectives". They are organized in two parts: The first part, Computation and Complexity, presents a collection of expository papers on fashionable themes in algorithmics, optimization, and complexity. The second part, Methods, Languages and Tools for Future System Development, aims at sketching the methodological evolution that helps guaranteeing that future systems meet their increasingly critical requirements. Chapter 3 is available open access under a Creative Commons Attribution 4.0 International License via link.springer.com. FME 2001: Formal Methods for Increasing Software Productivity International Symposium of Formal Methods Europe, Berlin, Germany, March 12-16, 2001, Proceedings** Springer Science & Business Media **This book constitutes the refereed proceedings of the International Symposium of Formal Methods Europe, FME 2001, held in Berlin, Germany, in March 2001. The 32 revised full papers presented together with abstracts of three invited talks were carefully reviewed and selected from a total of 72 submissions. Focusing on increasing software productivity, all current aspects in formal methods are covered. Among the application areas addressed are avionics, smart cards, financial engineering, E-commerce, middleware, security, telecommunications, etc.**