

by Hubert Garavel, Mihaela Sighireanu, Inria Rhone-alpes, Dyade Vasy Group, H. Garavel, M. Sighireanu.

Architecture of the Evaluator 4. The table below summarizes the size in lines of code of each compiler. In these approaches, Asts are implicit not directly visible to the programmer and it is not necessary to specify the order of attribute evaluation, which is inferred from the dependencies. On the contrary, our approach requires the explicit Ast specification and attribute computation ordering. Practically, this is not too restrictive, since the user is usually aware of these details. Lotos NT is an hybrid between imperative and functional languages. Unlike the object-oriented approach e. Compared to lower-level imperative languages such as C, Lotos NT avoids tedious and error-prone explicit pointer manipulation. Compared to functional languages such as Haskell or Caml7 for which the Happy8 and CamlYacc parser generators are available , Lotos NT does not allow higher-order functions nor polymorphism. In practice, we believe that these missing features are not essential for compiler construction; instead, Lotos NT provides useful mechanisms such as strong typing, function overloading, pattern-matching, and sequential composition. Contrary to functional languages specifically dedicated to compiler construction such as Puma9 and Gentle [15], Lotos NT is a general-purpose language, applicable to a wider range of problems. Our experience in using Lotos NT for developing three compilers demonstrated the efficiency and robustness of this pragmatic approach. Since , the Traian compiler is available on several platforms Windows, Linux, Solaris and can be downloaded on the Internet. The three Traian-based compilers are or will be available soon: A Generator of Interactive Environments Tools. The Logic of Typed Feature Structures. Cambridge Tracts in Theoretical Computer Science, 32, A Protocol Validation and Verification Toolbox. A Scripting Language for Compositional Verification. International Standard , To appear in Science of Computer Programming. The Term Processor Generator Kimwitu.

Chapter 2 : Mihaela Sighireanu - Publications

H. Garavel and M. Sighireanu, 3rd International Workshop on Formal Methods for Industrial Critical Systems FMICS'98, May , Amsterdam, The Netherlands Model-Checking Verification of the LOTOS Descriptions of the Invoicing Case Study.

Our semantics handles both synchronous and asynchronous communication. We define a global state automaton We define a global state automaton from an MSC, by first defining a transition system of global states, and from that a Buchi automaton. We propose a method using temporal logic formulas to specify the required liveness properties. The Argonaute system is specifically designed to describe, specify and verify reactive systems such as communication protocols, real-time applications, man-machine interfaces,. It is based upon the Argos graphical language, whose syntax relies on the Higraphs formalism by D. Harel [HAR88], and whose semantics is given by using a process algebra. Automata form the basic notion of the language, and hierarchical or parallel decompositions are given by using operators of the algebra. The complete formalization of the language inherits notions from both classical process algebras such as ccs [MIL80], and existing programming languages used in the same field such as Esterel [BG88] or the Statecharts formalism [HAR87]. Concerning complex system description, Argos allows to describe intrinsic states directly with the basic automaton notion and only them: The Argonaute system allows to describe reactive systems graphically, to spe Show Context Citation Context As these languages were not designed while taking into account the model checking approach, they must be adapted in order to allow the user to relate the system descriptions with the specification f Sighireanu - University of Nantes , " Process algebras are often advocated as suitable formalisms for the specification of telecommunication protocols and distributed systems. However, despite their mathematical basis, despite standardization attempts most notably the Formal Description Technique Lotos , and despite an ever growing However, despite their mathematical basis, despite standardization attempts most notably the Formal Description Technique Lotos , and despite an ever growing number of successful case-studies, process algebras have not yet reached a wide acceptance in industry. On the other hand, description languages such as Promela or Sdl are quite popular, although they lack a formal semantics, which should prohibit their use for safety-critical systems. Taking the international standard Lotos as a basis, we suggest several enhancements, which fall into three Each approach has its own problems and limitations. The inclusion of ActOne abstract data types in Lotos has been often criticized by both users an This paper presents the results of an industrial case-study concerning the use of formal methods for the validation of hardware design. The specification language Lotos Iso International Standard was used to describe formally the main components of this architecture processors, memory controller and bus arbiter. Four correctness properties were identified, which express the essential requirements for a proper functioning of the arbitration algorithm, and formalized in terms of bisimulation relations modulo abstractions between finite labelled transition systems. Kritzinger , " Predicting the performance of a communication protocol from a formal description was first proposed about a decade ago. Such performance prediction involves two issues: In this paper we address the first issue: The semantics to give time and how to introduce it into the model. The approach we propose does not affect the syntax of the formal description technique and does not depend on the FDT used, but we illustrate our ideas using SDL and derive performance measures from a simulation or execution of the specification. In many cases we could equally well use Markov or queueing theory or other proposals in the literature. The problem of determining the values of the appropriate parameters in those cases remain however. We also describe a software tool which implements our proposals and an example to illustrate their application in practice. FDT-based system and protocol engineering, Extensions of Workshop UML98 , " The development of correct OO distributed software is a daunting task as soon as the distributed interactions are not trivial. This is due to the inherent complexity of distributed systems latency, error recovery, etc. The OO technology is ill equipped to deal with this dimension of the problem. On the other hand, the willingness of mastering this complexity in the context of telecommunication protocols gave birth to specific formal verification and validation tools. The aim of this paper is to explore how the underlying

technology of these tools could be made available to the designer of OO distributed software. We propose a framework allowing the integration of formal verification and validation technology in a seamless OO lifecycle based on UML, the Unified Modeling Language. While the interest of formal techniques is wide

Chapter 3 : CADP'97 - Status, Applications and Perspectives - CORE

*This "Cited by" count includes citations to the following articles in Scholar. The ones marked * may be different from the article in the profile.*

Chapter 4 : Radu Mateescu - Home Page

H. Garavel and M. Sighireanu. Towards a Second Generation of Formal Description Techniques - Rationale for the Design of E-LOTOS. In Proc. of FMICS' CWI.

Chapter 5 : Table of contents for Library of Congress control number

by Mihaela Sighireanu, Hubert Garavel, In this paper we propose a solution for the module system for E-Lotos. This solution allows export and import, hiding, and generic modules.

Chapter 6 : French-Romanian Contributions to the Definition of E-LOTOS

M. Aguilar Cornejo, H. Garavel, R. Mateescu, and N. de Palma, Proceedings of the 3rd IFIP WG International Working Conference on Distributed Applications and Interoperable Systems DAIS' (Krakow, Poland), pages Kluwer Academic Publishers, September