

SRI International

AD-A256 528



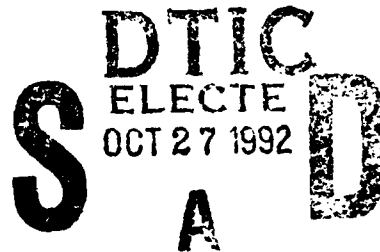
Progress Report 0002/A001 • October 12, 1992

LOGICS AND MODELS FOR CONCURRENCY AND TYPE THEORY

Prepared by:

José Meseguer, Principal Scientist
Computer Science Laboratory

SRI Project 6729



Prepared for:

Chief of Naval Research
Code 1133/Annual Report
Ballston Tower One
800 North Quincy Street
Arlington, VA 22217-5000

Attn: Dr. Ralph Wachter, Code 1133
Director, Computer Science Division

Contract No. N00014-88-C-0618

92-27450



1048

This document has been approved for public release and sale; its distribution is unlimited.

1 Productivity Measures

Principal Investigator Name: José Meseguer
PI Institution: CS Lab, SRI International
PI Phone Number: (415) 859-3044
PI E-mail Address: meseguer@csl.sri.com
Contract Title: Logics and Models for Concurrency and Type Theory
Contract Number: N00014-88-C-0618
Reporting Period: 1 Oct 91 - 30 April 92

- Refereed papers submitted but not yet published: 2
- Refereed papers published: 5
- Unrefereed reports and articles: 0
- Books or parts thereof submitted but not yet published: 1
- Books or parts thereof published: 1
- Patents filed but not yet granted: 0
- Patents granted: 0
- Invited presentations: 8
- Contributed presentations: 1
- Honors received¹: 9
- Prizes or awards received: 0
- Promotions obtained: 0
- Graduate students supported: 2
- Post-docs supported: 1
- Minorities supported: 0

Accession For	
NTIS CRA&I	J
DTIC TAB	
Unannounced	
Justification	
By <i>per A252737</i>	
Distribution/	
Availability	
Dist	Avail. and/or
	Special
<i>A-1</i>	

¹A description of these honors is included in Section 3.

2 Detailed Summary of Technical Progress

Principal Investigator Name: José Meseguer

PI Institution: CS Lab, SRI International

PI Phone Number: (415) 859-3044

PI E-mail Address: meseguer@csl.sri.com

Contract Title: Logics and Models for Concurrency and Type Theory

Contract Number: N00014-88-C-0618

Reporting Period: 1 Oct 91 - 30 April 92

This reporting period is the last period of the project. During this last period substantial progress has been made in the following areas:

- Linear logic and concurrency
- Rewriting logic and concurrency
- Concurrency models

2.1 Linear Logic and Concurrency

A key goal in this project has been to study logics that can express important aspects of concurrent computation. In particular, we have studied Girard's linear logic, placing special emphasis on its applications to concurrent computation.

The paper [1] by Martí-Oliet and Meseguer has now appeared in print. It contains a simple axiomatization of linear logic models based on category theory and studies some equationally defined subclasses of models, including poset models, and models in which the token game of Petri nets can be generalized to games with negative resources that can be canceled out by positive ones.

The paper [2] by Martí-Oliet and Meseguer has also appeared in print. It presents a detailed survey—aimed at researchers and students who would like a gentle introduction to the ideas of linear logic and their applications to concurrency—of recent work on the applications of linear logic to concurrency, with special emphasis on Petri nets and on the use of categorical models. In particular, it presents a synthesis of all the previous work on this project by Martí-Oliet and Meseguer on the systematic correspondence between Petri nets, linear logic theories, and linear categories, and explains its relationships to work by many other authors. Throughout, the computational interpretation of the linear logic connectives is discussed and the ideas are illustrated with examples.

2.2 Rewriting Logic and Concurrency

Our quest for logics that can naturally express concurrent computations has not been restricted to linear logic. A new logic called *rewriting logic* whose models are concurrent systems and whose deduction corresponds to concurrent computation in such systems has also been proposed and studied in this project. The concurrent systems that naturally appear as models of rewriting logic are very general, and in fact many previously known models of concurrency appear as special cases in a very direct and simple way.

By comparison with linear logic, rewriting logic is considerably more expressive for accounting for the possibly quite complex structure of the distributed state of a concurrent system. This structure can be expressed by what might be called user-definable propositional connectives and by axioms chosen for each application, instead of by a fixed set of connectives. However, although

extensions of rewriting logic for specification purposes seem natural and are very much worth exploring, rewriting logic is primarily a logic of computation; therefore, rewriting logic itself does not have counterparts for linear logic connectives whose primary use is specification.

Since rewriting logic is a logic of computation, it can be used as a programming language to program concurrent systems declaratively. This has led to the design of Maude, a concurrent programming language whose modules are theories in rewriting logic and whose concurrent computation is performed by logical deduction. Thanks to a map of logics that systematically relates equational logic to rewriting logic, an equational style of functional programming is supported in a sublanguage of Maude essentially identical to the equational language OBJ. Concurrent object-oriented programming is also supported and is conceptually expressed in rewriting logic terms. For an extension of Maude called MaudeLog whose modules are also rewriting logic theories but that allows queries with logical variables, Horn logic programming is also supported thanks to a map of logics embedding Horn logic into rewriting logic.

Rewriting logic's rules of deduction, model theory, and soundness and completeness theorems, as well as its use as a very general model of concurrency, have been published in [3].

A detailed study of Maude's concurrent object-oriented facet and of how the basic concepts of object-oriented programming can be naturally expressed in rewriting logic in a fully declarative way is presented and is illustrated with examples in the paper [4].

The paper [5] presents a general axiomatic notion of "logic programming language" based on the previous work of Meseguer on General Logics² and discusses methods for defining multiparadigm logic programming languages. The Maude and MaudeLog languages are viewed as specific examples of a multiparadigm logic programming language satisfying the general requirements proposed in the paper and unifying functional programming, concurrent object-oriented programming, and (in MaudeLog's case) Horn logic programming.

The paper [6] by Meseguer, Futatsugi, and Winkler gives an overview of Maude and rewriting logic with special emphasis on the software technology innovations made possible by this approach, including concurrency and machine independence, declarativeness and wide-spectrum capabilities, multiparadigm nature, highly reusable and adaptable modules, and support for decentralized cooperative problem solving.

The paper [9] studies the capabilities of Maude as a machine-independent parallel programming language. Maude is a wide spectrum language supporting specification, rapid prototyping and parallel programming. Parallel programming is carried out in a sublanguage called Simple Maude that can be executed with reasonable efficiency on a wide variety of parallel architectures, including SIMD, MIMD, and SIMD/MIMD. In addition, Simple Maude can incorporate modules written in conventional code, and can also integrate heterogeneous systems in a parallel framework.

2.3 Concurrency Models

Place/Transition (PT) Petri nets are one of the most widely used models of concurrency. However, they still seem to lack a satisfactory semantics: on the one hand the "token game" is too intensional, even in its more abstract interpretations in terms of nonsequential processes and monoidal categories; on the other hand, Winskel's basic unfolding construction, which provides a coreflection between nets and finitary prime algebraic domains, works only for safe nets.

In the papers [7, 8] Meseguer, Montanari, and Sassone extend Winskel's result to PT nets. They start with a rather general category **PTNets** of PT nets, introduce a category **DecOcc** of decorated (nondeterministic) occurrence nets, and define adjunctions between **PTNets** and

²José Meseguer, "General Logics," in H.-D. Ebbinghaus et al., eds., *Logic Colloquium '87*, pages 275-329. North Holland, 1989.

DecOcc and between **DecOcc** and **Occ**, the category of occurrence nets. The role of **DecOcc** is to provide natural unfoldings for PT nets, i.e., acyclic safe nets where a notion of family is used for relating multiple instances of the same place.

The unfolding functor from **PTNets** to **Occ** reduces to Winskel's when restricted to safe nets; moreover, the standard coreflection between **Occ** and **Dom**, the category of finitary prime algebraic domains, when composed with the unfolding functor above, determines a chain of adjunctions between **PTNets** and **Dom**.

It is worth observing that, while the categories of (marked) PT nets introduced so far in the literature do not have coproducts, **PTNets** has both products and coproducts. The same result holds for **DecOcc**.

3 List of Publications, Presentations, Reports and Awards/Honors

Principal Investigator Name: José Meseguer

PI Institution: CS Lab, SRI International

PI Phone Number: (415) 859-3044

PI E-mail Address: meseguer@csl.sri.com

Contract Title: Logics and Models for Concurrency and Type Theory

Contract Number: N00014-88-C-0618

Reporting Period: 1 Oct 91 - 30 April 92

3.1 Publications and Reports

References

- [1] Narciso Martí-Oliet and José Meseguer. An algebraic axiomatization of linear logic models. In G.M. Reed, A.W. Roscoe, and R. Wachter, editors, *Topology and Category Theory in Computer Science*, pages 335–355. Oxford University Press, 1991.
- [2] Narciso Martí-Oliet and José Meseguer. From Petri nets to linear logic through categories: a survey. *Intl. J. of Foundations of Comp. Sci.*, 2(4):297–399, 1991.
- [3] José Meseguer. Conditional rewriting logic as a unified model of concurrency. *Theoretical Computer Science*, 96(1):73–155, 1992. Also Technical Report SRI-CSL-91-05, SRI International, Computer Science Laboratory, February 1991.
- [4] José Meseguer. A logical theory of concurrent objects and its realization in the Maude language. To appear in G. Agha, P. Wegner, and A. Yonezawa (eds.) *Research Directions in Object-Based Concurrency*, 1992.
- [5] José Meseguer. Multiparadigm logic programming. In H. Kirchner and G. Levi, editors, *Proc. 3rd Intl. Conf. on Algebraic and Logic Programming*, pages 158–200. Springer LNCS 632, 1992.
- [6] José Meseguer, Kokichi Futatsugi, and Timothy Winkler. Using rewriting logic to specify, program, integrate, and reuse open concurrent systems of cooperating agents. To appear in *Proceedings of the 1992 International Symposium on New Models for Software Architecture*, Tokyo, Japan, November 1992, 1992.
- [7] José Meseguer, Ugo Montanari, and Vladimiro Sassone. On the semantics of Petri nets. In W.R. Cleaveland, editor, *Proceedings of the Concur'92 Conference, Stony Brook, New York, August 1992*, pages 286–301. Springer LNCS 630, 1992.
- [8] José Meseguer, Ugo Montanari, and Vladimiro Sassone. On the semantics of place/transition Petri nets. Submitted for publication, 1992.
- [9] José Meseguer and Timothy Winkler. Parallel Programming in Maude. In J.-P. Banâtre and D. Le Métayer, editors, *Research Directions in High-level Parallel Programming Languages*, pages 253–293. Springer LNCS 574, 1992. Also Technical Report SRI-CSL-91-08, SRI International, Computer Science Laboratory, November 1991.

3.2 Presentations

The following is a list of lectures given at professional conferences and seminars during the last academic year:

1. Lecture by J. Meseguer on "Maude and the Rewrite Rule Machine" at the California Institute of Technology, Pasadena, California, October 1991.
2. Lecture by J. Meseguer on "Declarative Programming" at the Center for the Study of Language and Information, Stanford University, Stanford, California, November 1991.
3. Lecture by J. Meseguer on "Parallel Programming in Maude" at the Kestrel Institute, Palo Alto, California, March 1992.
4. Lecture by J. Meseguer on "Parallel Programming in Maude" at the University of Pisa, Italy, May 1992.
5. Lecture by J. Meseguer on "Maude and the Rewrite Rule Machine" at the University of Southern California, Los Angeles, California, June 1992.
6. Lecture by J. Meseguer on "Maude and the Rewrite Rule Machine" at the Supercomputer Research Center, Bowie, Maryland, June 1992.
7. Lecture by J. Meseguer on "Maude and the Rewrite Rule Machine" at the Hewlett-Packard Research Laboratories, Palo Alto, California, July 1992.
8. Lecture by V. Sassone on the paper "On the Semantics of Petri Nets" by Meseguer, Montanari and Sassone at the CONCUR'92 International Conference, Stony Brook, New York, August 1992.
9. Invited Lecture by J. Meseguer on "Multiparadigm Logic Programming" at the Algebraic and Logic Programming International Symposium, Volterra, Italy, September 1992.

3.3 Awards/Honors

1. Editorial board of the Journal on Mathematical Structures in Computer Science (J. Meseguer).
2. Member of the IFIP Working Group 14.3 (Foundations of Systems Specification) (J. Meseguer).
3. Member of the GI (Gesellschaft fuer Informatik) Working Group 0.1.7 (Specification and Semantics) (J. Meseguer).
4. Program Committee of LICS'93 (Symposium on Logic in Computer Science) to be held in Montreal, Canada, June 1993 (J. Meseguer).
5. Program Committee of the Dagstuhl Workshop on Specification and Semantics to be held in Dagstuhl Castle, Germany, May 1993 (J. Meseguer).
6. Visiting Professor, holding the BBV Foundation Chair, University of Navarre, Spain, Fall 1992 (J. Meseguer).
7. Invited Lecture on "Multiparadigm Logic Programming" at the Algebraic and Logic Programming International Symposium, Volterra, Italy, September 1992 (J. Meseguer).

8. Session Chairman at the Algebraic and Logic Programming International Symposium, Volterra, Italy, September 1992 (J. Meseguer).
9. Invited lecture on "Using Rewriting Logic to Specify, Program, Integrate, and Reuse Open Concurrent Systems of Cooperating Agents" at the Second International Symposium on New Models for Software Architecture, Tokyo, Japan, November 1992 (J. Meseguer).

4 Transitions and DOD Interactions

Principal Investigator Name: José Meseguer

PI Institution: CS Lab, SRI International

PI Phone Number: (415) 859-3044

PI E-mail Address: meseguer@csl.sri.com

Contract Title: Logics and Models for Concurrency and Type Theory

Contract Number: N00014-88-C-0618

Reporting Period: 1 Oct 91 - 30 April 92

The researchers in the project are making a serious effort to transmit the project's results and ideas to the research community through publications, lectures, courses, etc. as summarized in this report. In this regard, in addition to the publications listed—which include a very comprehensive survey paper on linear logic and concurrency—it is worth pointing out the delivery of a good number of invited presentations—including invited talks at international conferences and a forthcoming invited course.

Fruitful interactions related to the Navy during the last academic year include:

- Technical discussions with Dr. Ralph Wachter, of the Office of Naval Research, in March and June 1992, on the research conducted under this contract and under contract N00014-90-C-0086
- Technical discussions with Dr. Keith Bromley, of the Naval Ocean Systems Center, in June 1992, on research conducted under a separate ONR contract on the Rewrite Rule Machine

In addition, the project leader has engaged in a variety of technical discussions with research officers at other DOD agencies, including DARPA, SDIO, and NSA.

5 Software and Hardware Prototypes

Principal Investigator Name: José Meseguer

PI Institution: CS Lab, SRI International

PI Phone Number: (415) 859-3044

PI E-mail Address: meseguer@csl.sri.com

Contract Title: Logics and Models for Concurrency and Type Theory

Contract Number: N00014-88-C-0618

Reporting Period: 1 Oct 91 - 30 April 92

No such prototypes have been developed under this contract.