# COMBINING SEMI-FORMAL AND FORMAL NOTATIONS IN SOFTWARE SPECIFICATION: AN APPROACH TO MODELLING TIME-CONSTRAINED SYSTEMS

**by**

**Sergiu-Mihai Dascalu**

**Submitted**
**in partial fulfillment of the requirements**
**for the degree of**

**DOCTOR OF PHILOSOPHY**

**Major Subject: Computer Science**

**at**

**DALHOUSIE UNIVERSITY**

**Halifax, Nova Scotia** **September, 2001**

## Dalhousie University

## Faculty of Computer Science

**The undersigned hereby certify that they have examined, and recommend to the Faculty of Graduate Studies for acceptance, the thesis entitled "Combining Semi-Formal and Formal Notations in Software Specification: An Approach to Modelling Time-Constrained Systems" by Sergiu-Mihai Dascalu in partial fulfillment of the requirements for the degree of the Doctor of Philosophy.**

**Dated :** _____

**Supervisor:**

_____

**Dr. PETER HITCHCOCK**

**External Examiner:**

_____

**Dr. GREGORY BUTLER**

**Examiners:**

_____

**Dr. PETER BODORIK**

_____

**Dr. TREVOR SMEDLEY**

_____

**Dr. WILLIAM PHILLIPS**

**Dalhousie University**

**Faculty of Computer Science**

**DATE: September 14, 2001**

| | |
|---|---|
| **AUTHOR:** | **Sergiu-Mihai Dascalu** |
| **TITLE:** | **Combining Semi-Formal and Formal Notations in Software Specification: An Approach to Modelling Time-Constrained Systems** |
| **MAJOR SUBJECT:** | **Computer Science** |
| **DEGREE:** | **Doctor of Philosophy** |
| **CONVOCATION:** | **October, 2001** |

**To my daughter Diana, my wife Alexandra,**
**and my parents Niculina and Vasile**

# Table of Contents

# List of Tables

# List of Figures

# List of Abbreviations

| | |
|---|---|
| **ACTS** | **Automatic Camshaft Testing System** |
| **ADF** | **Algorithm for Deformalisation** |
| **AFCD** | **Algorithm for Formalising Class Diagrams** |
| **AFSD** | **Algorithm for Formalising State Diagrams** |
| **CASE** | **Computer-Aided Software Engineering** |
| **CD** | **Class Diagram** |
| **CLS** | **Class Specification** |
| **CLSTD** | **Class State Diagram** |
| **COMP** | **Class Compound** |
| **CSP** | **Communicating Sequential Processes** |
| **DFD** | **Data Flow Diagrams** |
| **ERD** | **Entity Relationship Diagrams** |
| **IDE** | **Integrated Development Environment** |
| **ISE** | **Integrated Specification Environment** |
| **GUI** | **Graphical User Interface** |
| **ELS** | **The Elevator System** |
| **OMG** | **Object Management Group** |
| **OMT** | **Object Modelling Technique** |
| **OO** | **Object-Oriented** |
| **OOA** | **Object-Oriented Analysis** |
| **OOD** | **Object-Oriented Design** |
| **OOSE** | **Object-Oriented Software Engineering** |
| **OOZ** | **Object-Oriented Variant of Z** |
| **ROOM** | **Real-Time Object Oriented Modelling** |
| **RT** | **Real-Time** |
| **RTL** | **Real-Time Logic** |
| **RTS** | **Real-Time Systems** |
| **SC** | **Scenario** |
| **SCG** | **Scenario Group** |
| **SQD** | **Sequence Diagram** |
| **SQDG** | **Sequence Diagram Group** |
| **TCS** | **Time-Constrained Systems** |
| **TL** | **Temporal Logic** |
| **UC** | **Use Case** |
| **UCD** | **Use Case Diagram** |
| **UML** | **Unified Modelling Language** |
| **ZPPC** | **Z++ Class** |
| **ZSPEC** | **Z++ Specification** |

# Acknowledgements

**First of foremost I would like to express my deepest gratitude to my thesis supervisor, Dr. Peter Hitchcock, who guided my research with infinite wisdom and patience, as well as with unabated confidence in my ability of completing the work presented here. I am much indebted to Dr. Hitchcock for introducing me to the rigorous, admirable world of formal notations, and for spending countless hours with me guiding the shaping of the approach described in this thesis.**

**I would also like to express my heartfelt gratitude to the members of my examining committee, Dr. Peter Bodorik, Dr. Trevor Smedley, and Dr. William Phillips, for their guidance and support throughout the course of my studies. In particular, I would like to thank Dr. Bodorik for supervising my earlier coursework and research, to express my gratitude to Dr. Smedley for opening for me the realm of visual notations, and to thank Dr. Phillips for his advice and encouragement.**

**I would also like to extend my thanks to the external examiner, Dr. Gregory Butler, from Concordia University, Montreal, for his thorough review and constructive critique of the thesis. Dr. Butler has pointed out with great clarity both the merits and the limitations of our thesis, his observations about the latter being extremely helpful for us to better evaluate our work and to define its future directions.**

**I would also like to express my gratitude to the former School of Computer Science and the former Technical University of Nova Scotia for the financial support I received in the form of scholarships and awards during my doctoral studies. I would like to thank both the former School of Computer Science at TUNS and the current Faculty of Computer Science at Dalhousie University, in particular to their directors, Dr. Jonathan Barzilai and, respectively, Dr. Jacob Slonim, for trusting me with teaching appointments as lecturer for the courses Software Development with Ada, Software Engineering, Database Management Systems, and Computer Organization for Electrical Engineers. Teaching these courses offered me the privilege of being constantly stimulated by the inquiring minds of my students and helped me consolidate the foundation of the present work.**

**Among the students who steadily motivated and encouraged me in my work Paul Evans, Greg Power, Chad Seward, Patrick Lee, Silvano Da Ros, Carmen Wong, Fernand Boudreau, Stephen Nickerson, Atreya Basu, Samer Mansour, and Jas Singh are the ones to whom I am especially grateful. Many thanks are due as well to my graduate colleagues Dawn Jutla, Eve Rosenthal, Marcel Karam, Krys Gawetski, and Arun Sood who provided me, in various phases of my studies, with a friendly and stimulating environment.**

**I am deeply indebted to Dr. Traian Ionescu, my former team leader and head of department at the Polytechnic University of Bucharest, Romania, who mentored my earlier professional steps with wisdom, goodwill, and grace. To him and his wife, Mrs. Mihaela Ionescu, I owe special thanks for their guidance and friendship throughout many years.**

**I was very fortunate to work with and be guided by several other senior professors at the Polytechnic University of Bucharest. I would like to take this opportunity and thank Dr. Radu Dobrescu, Dr. Sergiu Iliescu, and Dr. Aurelian Stanescu for giving me, years ago, through their personal examples, an excellent definition of the word academia.**

**To my longtime friend, Cristian Mierlea, I would like to say thank you for many things but especially for teaching me, many years ago, that in essence the computers are an extraordinary combination of mathematics and literature. And for inducing me, in those high school years, to value them dearly. For their constant support and encouragement throughout the years my heartfelt thanks go as well to my friend Mike Cailean and to his wife, Rodica Cailean. I would also like to express my thanks to a special friend, close to my heart and mind, Jorge Luis Borges, the great Argentine writer, presently in Heaven, in all probability narrating one of his fabulous stories. The beauty and finesse of his writings lifted my spirit in countless nights.**

**I have many special thanks for my parents, my wife, and my daughter. Words can hardly express my gratitude for them, but I trust they know how deeply I have appreciated their support throughout my studies. My daughter Diana grew up with a student father, quietly accepting to have me away from her in so many occasions. Finally, I would like to particularly thank my wife for her unfailing understanding, extended sacrifice, and unconditional love. I would like her to know that without her support this thesis would never have happened.**

# Abstract

**This thesis is about the integration of semi-formal, graphical representations with formal notations within a modelling approach aimed at the construction of time-constrained systems (TCS). We believe that the two types of notation, graphical (semi-formal) and, respectively, formal, can efficiently complement each other and provide the basis for a software specification approach that can be both rigorous and practical. Although many authors have envisaged the advantages of combining informality with formality in software construction, there are very few reports that address the problem within the context of object-orientation and project its solution over the canvas of TCS modelling.**

**The pillars of our approach are the following: the combination of formal and semi-formal notations for specification purposes, the integration into an object-oriented approach of modelling capabilities that target properties of TCS, the elaboration of detailed algorithms for UML to Z++ translations, and the proposal of a procedural frame for effective and reliable development of TCS. Principles and an outline of an algorithm for the reverse translation, from Z++ to UML, are also included in the approach.**

**While the graphical notation employed is a subset of the UML, the formal notations used are Lano's Z++, an object-oriented variant of Z, and Jahanian and Mok's Real Time Logic. Both structural and dynamic aspects of the system are considered and a new modelling element denoted class compound is proposed.**

**From a methodological point of view, after several UML-based modelling steps are completed the formalisation process can take place, the result being a formal specification derived from the graphical representations obtained in the earlier steps. The integrated, semi-formal and formal model of the system can be subsequently enhanced while the designed translation mechanisms allow changes in the graphical representations to be reflected into the formal specifications as well as modifications of the formal specifications to be fed back into the diagrammatic descriptions of the system.**

**A case study, an Elevator System, is included in the thesis to illustrate the application of the proposed approach and the GUI-centred design of Harmony, an integrated specification environment intended to support the approach, is also presented.**

**Although we believe the proposed approach offers a viable solution for modelling software systems, it has nevertheless a number of limitations that need be pointed out. Firstly, the translation of UML constructs is restricted to a subset of the notation, and the treatment of state diagrams is confined to sequential, non-composite executions (composite states and aspects related to concurrency are not covered). Secondly, although timing constraints can be attached to structural UML constructs in the regular way, we have not tackled their mechanised translation to Z++, and there is a limited incorporation of such constraints in the state diagrams considered. Thirdly, the formal language employed, Z++, is currently lacking in supporting tools, which could be an impediment to the use of the proposed approach in industrial applications. Fourthly, for the formalisation algorithms a set of rules for well-formedness and a set of principles for translation are given without using meta-models for UML and Z++/RTL, yet the use of these meta-models would have probably allowed a more concise and precise description of the algorithms.**

**However, our belief is that, through future work, the above limitations can be overcome and our proposal can thus become a stronger contender in the landscape of object-oriented approaches for modelling TCS.**