This is a collection of articles written by leading experts in their fields covering a wide area of structural links between logic and information flow. These links are not, of course, exhausted here. One might also include nonmonotonic logic, many agents logics and logics for communication and concurrency, to name just a few formalisms which touch on the multisided connection of logic and information. As Van Eijk and Visser mention, 'the time is not ripe for a unified theory of information, information linking, action and information flow', and perhaps there should not be because the particular formalization might well depend on the particular context or application. However this book succeeds in identifying the subject and giving us a glimpse of some problems associated with it.


Konstantinos Georgatos
Institute of Mathematical Sciences
Madras, India

Dov M. Gabbay, Ian Hodkinson & Mark Reynolds

Temporal logic describes the evolution of a system in time. It has emerged from mathematics and philosophy, and has gained a lot of attention in computer science over the last two decades. It is widely used as a specification language for expressing the requirements and constraints of systems such as communication protocols, databases and real-time environments, and is also used in applications of AI and natural languages. The theory behind temporal logic has been put into practice in various tools that can verify largely automatically the correctness of algorithms and hardware circuits.

This book presents the results of more than two decades of intensive research by its authors in the area of temporal (modal) logic. In particular, it reflects the broad research results of the first author, Dov Gabbay. The two main subjects of this book are the axiomatization and the expressiveness of various temporal logic systems. Such systems can differ from one another with respect to the class of models they address and the temporal operators that can be used to express properties. Classes of models can vary according to their respective admissible relation on time. For example, in linear time, each pair of time points is ordered, i.e. one comes before the other. On the other hand, branching time allows many possible unordered futures, but only one past (the future is yet unknown, so there are many possibilities, but the past has already happened). Classes of models can also vary according to the time domain, that is, whether the time points correspond to the natural numbers, the integers or the reals, for example.

In this book, the expressiveness of propositional temporal logics is studied with respect to associated first order languages in which they are embedded. Thus, first order logic is viewed as a yardstick to compare the expressiveness of temporal logics; the property of being expressibly complete means 'as expressive as the associated first order language in which it is embedded'. The book presents some fundamental theorems which help to prove or disprove the expressiveness completeness of a temporal logic, or to show that for a given class of models, expressiveness completeness cannot be achieved, regardless of the number of temporal operators that are added. Logics that are more expressive than first order logic are also discussed. These include propositional temporal logic with fixpoint operators or quantifiers which have the expressive power of second order logic.

The book also deals with temporal logics over first order structures, admitting function and relation symbols. This is generalized by presenting a generic construction for adding a temporal dimension to various logics. This construction introduces the ability to express changes over time to an existing logic. Connections between the original logics and their temporalized versions are shown with respect to expressiveness and the axiomatization. This topic is written as a separate research paper by Dov Gabbay and Marcelo Finger. Results about the complexity of deciding the validity of temporal formulae for various systems are surveyed in the last chapter of the book.

A very noticeable feature of the book is the great generality in which the results are presented: special attention is made to present each topic as broadly as possible. While most textbooks on temporal logic present various temporal systems, this book also generalizes the presentation in other ways. For example, it studies temporal logics with multiple time dimensions, where additional dimensions are used as reference points.

Although the book is focused mainly on the authors' research, the breadth of their research allows them to cover a lot of interesting material. They also include
other relevant fundamental results such as Salqvist’s correspondence theorem, which allows characterizing classes of models using certain temporal formulae, and the connection between expressive completeness and Ehrenfeucht games. One exception to the general breadth of this book is the theory of finite automata over infinite words. This is only briefly covered and the reader is referred for results to relevant research papers. This theory is one of the most important tools used today in the research and applications of linear temporal logic, and does not receive here the coverage it deserves.

The book is not an easy-to-read introduction to temporal logic, and some existing knowledge of temporal logics is useful; it is geared more towards experts in temporal logic who want to broaden their knowledge. It would be hard to use this book as a textbook for an undergraduate level course. However, it can be used as a teacher’s reference for an advanced graduate level course on temporal logic.

The style of writing in this book is similar to that of a research paper. It contains proofs for most of the theorems presented. The proofs are comprehensible, but are not always trivial to follow, and some maturity in understanding proofs in logic can be very helpful. It is natural for an exposition of this length to have a number of typos, but it is hoped they will be corrected in later printings. In the typesetting, the same fonts are used for temporal operators and propositions, making such expressions sometimes difficult to parse. The bibliography deserves some more attention in order to be more useful for locating the relevant referenced papers.

In general, this new book is a very comprehensive study of temporal logics, their expressiveness and axiomatization. It is not an easy text for beginners in this area; it is very thorough and presents the subjects of its focus in great depth. It can point practitioners of this field to existing results and thus is recommended as a good research reference.

D. PELED
AT&T Bell Laboratories

CHRIS CASEY

Development of large software is complex, time consuming and error prone. Many software engineering methodologies and techniques have been proposed to lessen these problems. But most of these methods are informal: except for the final code produced, the initial specifications and the intermediate designs are stated using imprecise, ambiguous and verbose natural language and/or diagrams and can lead to inconsistent descriptions whose completeness is uncertain. With an informal notation, we cannot talk about consistency and completeness of descriptions, quite apart from the problem of checking for these properties. Testing, used as a technique for ensuring that the produced software is free of errors, only enables one to uncover bugs. Except in some special classes of programs, exhaustive testing to certify that the developed systems is free of errors is impossible and/or infeasible.

Formal methods for software development have been proposed to overcome these deficiencies. Two important ingredients of any formal method are the use of a formal language for specifications and intermediate designs and the use of mathematical techniques for establishing that systems meet their specification. Further, the testing phase is replaced by formal verification, which involves showing rigorously, using mathematical techniques, that a given system meets its specification.

Formal methods have been in existence for quite some time but remain somewhat inaccessible to an average programmer or to students not having much background in mathematics. The aim of this text book is to introduce formal methods at a level that can be easily assimilated and used by these groups of people; as the name suggests, it adopts a programming approach to illustrate the concepts and techniques of formal methods. The author succeeds, to a greater extent, in this aim.

The book starts with a survey of the concepts and features of programming languages and software engineering, before getting into the basics of formal methods. This survey is rather brief and may serve only as a refresher intended for reviewing concepts relevant for the rest of the book.

The basic concepts of formal methods are introduced and illustrated in the next three chapters. Quite a number of important topics and techniques are touched upon in these chapters. These include symbolic logic as a specification language, correctness proofs of programs, loop invariants, proof by induction, recursion and termination. It is interesting to note that a working knowledge of many of these topics could be gained in the short span of these three chapters. All the concepts are well illustrated with a number of simple examples.

After this general introduction to the concepts of formal methods, the book gets into some details of a specific formal method, namely VDM. (This transition from generality to specifics is rather sudden and could probably have been avoided.) Four important abstract data types of VDM are discussed and their use is illustrated with a number of examples.

The important topic of data type refinement is also discussed and illustrated using a small example. This discussion is primarily in the context of VDM. One chapter is devoted to the use of formal methods in developing designs from specifications, testing and documentation. A complete chapter is devoted to a case study that illustrates the use of formal methods in developing part of a simple information system.

A nice feature about the book is that it contains a