There are some redundant repetitions of essentially the same material in the Handbook. For example, the properties of CPOs and continuous functions and their fixed point have their reasonable place in the chapters.

The Handbook could be the best place to arrange the terminology of Theoretical Computer Science. As an example a widespread confusion in the use of the terms "Tarski Theorem" (on complete lattice of fixpoints) and "Knaster-Tarski Theorem" (?) may be considered: both terms are used when the subject is "Knaster Theorem" (on existence of fixpoints) (see, Une théorème sur les fonctions d'ensembles, Ann. Soc. Polon. Math. 6 (1927) 133-134).

Perhaps, a great opportunity was missed by not providing a list of open problems in the Handbook.

These are the words I ought to say with an optimistic hope that the next version of the Handbook will be more perfect.

Nevertheless, the Handbook is a unique presentation of theoretical computer science and is highly recommended to those interested in it. So enjoy it as I did.

Hrant. B. Marandjian
Institute for Informatics and Automation Problems
Academy of Sciences
Yerevan, Armenia


Positive logic programming is a fragment of first-order logic enjoying nice computational and model-theoretic properties. These are essentially the completeness of a variation of linear resolution (SLD-resolution) and the existence of a minimal Herbrand model, computable as least fixpoint of an immediate consequences operator. These properties are the basis of the well-known interpretation of this fragment as a programming language. The expressive power of positive logic programs has been improved by allowing to use a form of default negation (normal logic programs) and other logical connectives in the clause bodies. Still the basic computational properties of positive logic programs can be shown to hold. The basic constraint is therefore on the structure of the clause heads, which are always forced to be (positive) atoms. Many people in the logic programming community believe this to be an essential constraint. Namely, if we relax the strong constraint on the clause heads, we are going to lose all the nice properties of logic programs.

The theory of disjunctive logic programs, developed by Jack Minker's group over several years, shows that this is not really the case. By using clauses with disjunctive heads we can represent indefinite knowledge, while preserving most of the properties of positive logic programs. In particular, there exists still a complete linear resolution strategy (SLI-resolution). Moreover the minimal Herbrand model can be replaced by
the minimal model state, which can also be computed as least fixpoint of an immediate consequences operator. The only questionable aspect is the nature of the proof procedure, which, even if linear, does not seem to naturally belong to the class of uniform proof procedures, which are sometimes considered to be the typical aspect of logic programming.

The book is a very good comprehensive description of the theory of disjunctive logic programs. The first four chapters have the same organization as John Lloyd's book on the theory of pure logic programs. They cover the declarative and procedural semantics of disjunctive logic programs and include the basic definitions and results valid for pure logic programs. The similarities are also shown by the fact that most of the proofs are either identical or very similar to those contained in Lloyd's book.

The second part of the book is even more interesting and contains topics such as constructive negation, stable, well-founded and stationary semantics that have never been systematically described in a book even for non-disjunctive programs. In particular, I found the procedural semantics for normal disjunctive logic programs very useful. Chapter 9 considers the case of disjunctive databases (function-free and range restricted disjunctive programs), from the typical viewpoint of the logic databases community, i.e. bottom-up set-oriented query evaluation methods. Finally, the last chapter contains two interesting applications still in the case of disjunctive deductive databases.

Giorgio LEVI
Università di Pisa
Corso Italia 40
56125 Pisa, Italy


This book is a marriage of Elgot's iterative algebraic theories, Pratt's dynamic logic, and the author's own theory of Boolean categories. Such a marriage has its benefits and drawbacks, and the author has done a good job developing the former while deemphasizing the latter. The intent of the book is to show that Boolean categories with iteration operators form useful models for dynamic logic and related programming logics. The book is a research monograph (it developed from a research paper that grew and grew), and a well-prepared reader of the book must have background in category theory and dynamic logic.

The text is divided into four sections: a review of iterative algebraic theories, a development of Boolean categories, "metatheory," and distributive categories. A valuable feature of the book is its 22 page Introduction, which summarizes in detail the contents of the four sections. With this material, a reader can judge clearly which parts of the book will be of most interest. Also, the reader's progress through the Introduction provides a good gauge of the reader's likely progress through the text itself!