early a stage but they are of a form which can be expanded subsequently. One of the strengths of the adopted approach is that graphical and textual representations are developed in parallel. The graphical representation is very readable giving a good overview while the textual representation is written in a BNF which is reassuringly code-like. The syntax is followed by examples which normally relate to one of the case studies which punctuate the text and neatly tie up any loose ends.

PETER STRAIN-CLARK

J. W. O. BRINKWORTH
Software Quality Management. Prentice Hall. 0-13-818444-5. £22.95

I have just finished writing a book on software quality assurance and can confirm that it lacks the excitement of developing books which describe more technical subjects. Brinkworth's book is an attempt to overcome the high yawn factor in the vast majority of books on this topic. It just about succeeds but, in the process, the author has sacrificed a degree of cohesion and structure. First, let me say that the book is quite readable and is a good introduction to many of the topics that are important for software quality assurance; topics such as documentation standards, testing standards, configuration management and the key role of the requirements specification. These are described in a fairly relaxed way, sometimes too relaxed; the preponderance of exclamation marks, cliches and sentences such as "This is where I enumerate the classic heuristics of structuring code. Since you are a mature adult (I presume!), I offer you a choice ...", "this is where we enter the black art of estimating ..." and "Let's say you have a spanking new factory that crochets high-fibre, multi-vitamin cardigans ..." tend, in the end, to irritate, rather than promote a laid back approach. It is somewhat disappointing that a copy editor was not stronger with the author and excised his excesses.

Another mistake the author makes is in not providing some overall vision of what quality assurance is about: that it is the process of identifying quality factors and project-specific quality controls which ensure that these factors are present. If he had spent a few pages providing an adequate structural lifeline and overview, then everything would have been easier to fit into the book and seen as being relevant. As it is, one is faced with the impression of a loosely coupled set of topics, reasonably written but ultimately reflecting a piecemeal attitude to quality assurance.

I was also disappointed by the amount of space devoted to ISO 9000. Software developers are desperate for information about this standard. ISO 9003 does not provide much and the Tickit guide is totally inadequate. Unfortunately the author deals with the subject in three pages in a slightly eccentric way. For example, his interpretation of process control in ISO 9000 is that it covers the coding process.

If you want to read about individual topics in isolation then this is a good book to use. However, if you are interested in the nature of quality assurance, how it fits into a company, the role of the quality system, the relationship between quality factors and the quality plan, then this is not the book for you.

DARREL INCE

SUSAN FOLKES and SUE STUBENVOLL
Accelerated Systems Development. Prentice Hall. 0-13-006073-9. £22.95

You MUST buy this book. While this book is written by practitioners for practitioners, academics and the suppliers of methods and tools could equally well benefit by understanding how their offerings are used, and more importantly why they are used, what software development requirement they meet.

This book takes as its theme the need to develop systems more rapidly and cheaply, spurred on by a more rapidly changing world that demands system solutions more quickly, when there is a growing backlog of requests for systems waiting for development. The solution is to use a number of appropriate methods and tools to accelerate development. The book includes short descriptions of these approaches, with advice on when particular approaches are appropriate.

The contribution of the book is the framework within which to select methods and tools. The first three chapters motivate and then introduce a simple risk assessment method, which enables one to analyse a project and determine its needs across a number of quality features and identify where risks could arise. The method then focuses on selecting ways of accelerating development without increasing the risk unnecessarily, taking whatever trade-offs between methods and tools that are appropriate. An example is worked through in an appendix. The authors emphasize, rightly, that the method is a tool, to help you make a decision, not to make the decision for you.

The rest of the book then describes the methods and tools available, under issue focussed titles "fuzzy definitions" (requirements problems), "complex systems", "accelerating construction", "project management" and "managing corporate commitment". Prototyping, 4GLs, expert systems, object orientation, CASE tools, software asset reuse, management tools and methods are all covered, along with human issues. References are given to where you can find out more if you so wish.

Of course the book does have some minor drawbacks. The text does not always flow as it should—at times it uses bullet points on a grand scale, as if I were reading a presentation, and there are some failures in the copy editing where passages are joined together inappropri-